Palestine Polytechnic University

College of Administrative Sciences and Informatics

Department Information Technology

Exploring the Traffic Sign Recognition Problem

Research team:

Ala' Ibrahem Talahmeh

Asma' Abdul Aziz Amro

Dua' Omar Mansour

Supervisor:

Dr. Alaa Halawani

This project is submitted in partial fulfillment for the bachelor's degree in Information Technology.

2009

Letter of Acknowledgment

We are very grateful to our worthy teachers and responsible colleagues who helped us during our endeavor for completion of our research. In particular, we are greatly indebted to our supervisors: Dr. Alaa Halawani and Dr. Hashem Tamimi for their valuable advice, critical comments, their inspiring intellectual guidance and optimistic encouragement through out our work.

Our appreciation to the support extended by our families whom exerted their best efforts to create a very comfortable environment for us to accomplish the research.
During the hard time of research they encouraged and provoked us to keep working to achieve our collective goal.

We would like also to express our enduring appreciation to a friend of ours Mohammad Jebriny for his unique understanding, extreme help and incredible patience.

We wish to extend our deepest appreciation to all the staff of Faculty of Information Technology in PPU for their co-operation and support throughout the course of our research. We also acknowledge our other friends and colleagues who carry out their B.Sc. Graduation Research in the same faculty.

We also thank our friends for their support and care.

Last but not least, my heartfelt appreciation to all of people who helped encouraged and supported us all the way long.

Research team

Dedication

To the cause of life …………

Our Parents

To the lights of life …………..

Our Teachers

To our unique resort in life

Our Friends

To the Oxygen ……….

Our beloved persons

To the souls that have left and that are still waiting

Research team

Declaration

The research team announces that the information contained in this research is correctly documented and that we are legally accountable if it is proved otherwise.

Research team

Abstract

In this project a technique suggested as a solution for traffic sign recognition is presented.

Scale-Invariant Feature Transform (SIFT) was examined to perform recognition (Detection and Classification).

SIFT proved to be excellent in the detection and classification of traffic signs.

Since SIFT is orientation invariant, orientation histogram used to make the feature orientation dependent.

We used two databases of traffic signs images for testing the system, the result showed a success rate of (96.67%).

Table of contents

List of figure

List of tables

Chapter One: Introduction

Overview

Research objectives

System components

Limitations & constraints

System requirements

Methodology

Literature survey

1.1 Overview

Our study is a valuable research since it snapshots a modern and critical problem which is traffic sign recognition. Such driver assistant systems are used nowadays in different transportation means.

On roads and especially in cars, such systems are more crucial. Surrounding scenes in roads are not stable and full of many things other than traffic signs that make it difficult for drivers to concentrate on signs. Specially drivers with vision problem and other kinds of disabilities that make traffic sign recognition difficult.

So, a system based on machine vision is needed, using a camera depending on image processing techniques.

Traffic signs have a distinctive appearance from all other objects in an outdoor scene. They are pure geometric shapes (e.g. circle, rectangle, triangle) filled with red, blue, yellow or white colors.

Despite the fact of uniqueness of traffic signs, researchers don't find it easy to perform the recognition. In other words there is no one unique method that performs detection and classification perfectly. Even the most successful solutions do not provide a comprehensive recognition for all traffic signs.

1.2 Research objectives

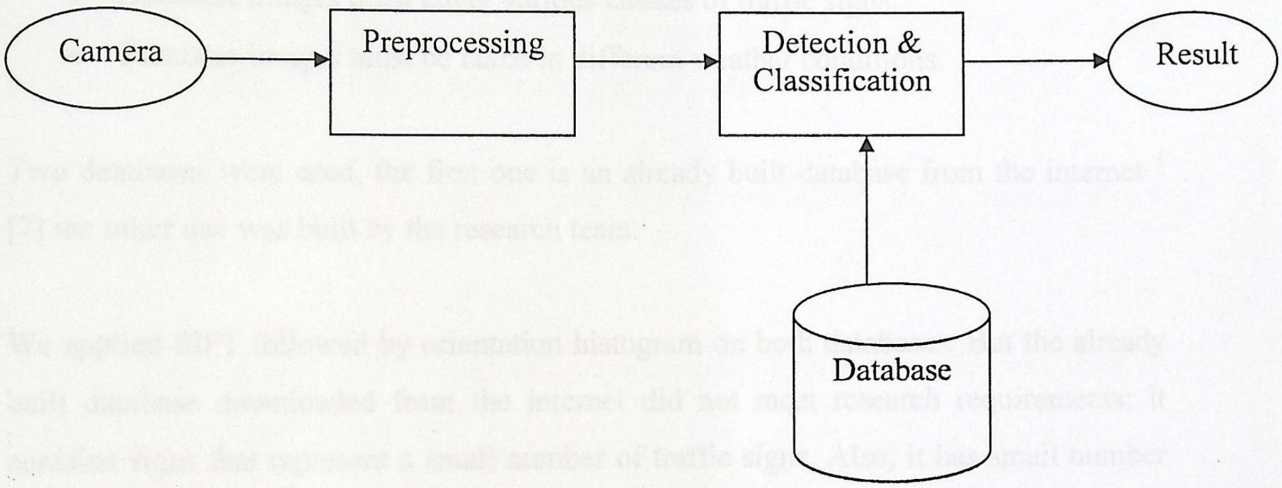
In this research the team aims to accomplish the following objectives:

1. Explore traffic sign recognition problem.
2. Apply an efficient technique for detection and classification of traffic signs and testing this technique.
3. Improve current methods and combine some of them to provide semi complete solution.
4. Study of both quality and performance of the system.

5. Highlight common problems and suggest solutions to overcome them in future work.

1.3 System components

The suggested system contains different components. The main components of the system are shown in figure(1).



Figure(1): System components

- camera: digital camera is used to capture image of a whole scene containing the traffic sign.
- Preprocessing : perform operation on query image to make it in a suitable format for next step.
- Detection &Classification: recognition of the traffc sign from the preprocessed query image, classification is performed by comparing detected sign with the image in the database.
- Database: collection of traffic images used to perform classification.

1.4 Limitations & constraints

During the research the team faced several limitations, some are listed below:

1. Database collection

One of the most important limitation and constraints that we faced in our research is to obtain an appropriate database of traffic signs images that satisfies certain conditions:

- Database must contain reasonable number of images for training and testing needs.
- Database images must cover various classes of traffic signs.
- Database images must be taken in different weather conditions.

Two databases were used, the first one is an already built database from the internet [1] [7] the other one was built by the research team.

We applied SIFT followed by orientation histogram on both databases. But the already built database downloaded from the internet did not meet research requirements; it contains signs that represent a small number of traffic signs. Also, it has small number of images.

Building a new database was time consuming. We were faced by inappropriate weather conditions that acted as negative external influences. The biggest problem is the lack of diversity of traffic sign in the area around us. In addition some signs were not clear or almost destroyed.

2. budget was limited

1.5 System requirements

Suggested system needs no extra infrastructure. It needs the following components (table (1) below) :

[1] http://www.domkrat.ru/Laws/rules/znak1.shtml.

Item	Cost
Digital cell phone camera	250$
Computer(lenovo, N100,dual sore)	1400$

Table(1): System requirements

1.6 Methodology

Our research is based on the scientific research methodology. Based on image processing, machine vision, and pattern recognition techniques.

SIFT and edge/orientation histogram were applied and tested as a comprehensive recognition solution.

Xnview program was used to perform preprocessing. Matlab program was used to perform tests and produce reports.

1.7 Literature survey

We have selected our research topic to concentrate on traffic sign recognition problem, which has been challenging problem for researchers who recognized the problem in the first place. Researchers invested non-ignorable time and effort analyzing the problem and developing methods to overcome it by taking advantage of different combined disciplines: Statistics, Mathematics, Image Processing, Human Perception, Human Computer Interaction, and many of other scientific branches.

In our research we dedicate this section to mention basic efforts and techniques adopted over several years of research to solve the problem of traffic sign recognition, to take advantage of those ideas in our attempt to understand the problem and its solutions, and to improve current methods by highlighting any existing weaknesses.

Review of literature

Traffic Sign Recognition problem has been discussed and studied by many researchers. Many solutions were provided to solve detection and/or classification problems.

In terms of the first stage in a traffic sign recognition system; detection of the traffic sign from whole complex scene. Provided solutions may be shape-based [5], color-based [2][4][6] or both[3].

In [2] Gao, et al. applied color appearance model CIECAM97s to segment traffic signs from the rest of scene. Segmentation process can be summarized as follows: to Find the range of color vectors; use equations to transform the image from RGB to standard XYZ, then using CIECAM97 model Hue, lightness and chroma are obtained.

For color segmentation Ruta, Li and Liu in [4] developed the Color Structure Code (CSC). Image segmentation is one of the most crucial phases in an image analysis system. Color image segmentation is performed to partition an image into separate regions that are homogeneous in terms of their color. The CSC is an inherent parallel color segmentation technique that also operates on distributed data. The CSC follows a hierarchical region growing on a special hexagonal topology.Detection is then –after segmentation- achieved by a fault–tolerant knowledge–based hierarchical decision graph consisting of many different feature extraction modules. In this graph a decision is not made in a binary (yes, no) but in a fuzzy way (may be) by using probability measures.

In [6] Ek¡stein and Bellada applied a detection method that depends on converting RGB colors into HSV, or HIS color models so it's easier to detect traffic signs. Since traffic signs are basically composed of the pure colors: red, blue, white and yellow.

Item	Cost
Digital cell phone camera	250$
Computer(lenovo, N100,dual sore)	1400$

Table(1): System requirements

1.6 Methodology

Our research is based on the scientific research methodology. Based on image processing, machine vision, and pattern recognition techniques.

SIFT and edge/orientation histogram were applied and tested as a comprehensive recognition solution.

Xnview program was used to perform preprocessing. Matlab program was used to perform tests and produce reports.

1.7 Literature survey

We have selected our research topic to concentrate on traffic sign recognition problem, which has been challenging problem for researchers who recognized the problem in the first place. Researchers invested non-ignorable time and effort analyzing the problem and developing methods to overcome it by taking advantage of different combined disciplines: Statistics, Mathematics, Image Processing, Human Perception, Human Computer Interaction, and many of other scientific branches.

In our research we dedicate this section to mention basic efforts and techniques adopted over several years of research to solve the problem of traffic sign recognition, to take advantage of those ideas in our attempt to understand the problem and its solutions, and to improve current methods by highlighting any existing weaknesses.

Review of literature

Traffic Sign Recognition problem has been discussed and studied by many researchers. Many solutions were provided to solve detection and/or classification problems.

In terms of the first stage in a traffic sign recognition system; detection of the traffic sign from whole complex scene. Provided solutions may be shape-based [5], color-based [2][4][6] or both[3].

In [2] Gao, et al. applied color appearance model CIECAM97s to segment traffic signs from the rest of scene. Segmentation process can be summarized as follows: to Find the range of color vectors; use equations to transform the image from RGB to standard XYZ, then using CIECAM97 model Hue, lightness and chroma are obtained.

For color segmentation Ruta, Li and Liu in [4] developed the Color Structure Code (CSC). Image segmentation is one of the most crucial phases in an image analysis system. Color image segmentation is performed to partition an image into separate regions that are homogeneous in terms of their color. The CSC is an inherent parallel color segmentation technique that also operates on distributed data. The CSC follows a hierarchical region growing on a special hexagonal topology.Detection is then –after segmentation- achieved by a fault–tolerant knowledge–based hierarchical decision graph consisting of many different feature extraction modules. In this graph a decision is not made in a binary (yes, no) but in a fuzzy way (may be) by using probability measures.

In [6] Ek¡stein and Bellada applied a detection method that depends on converting RGB colors into HSV, or HIS color models so it's easier to detect traffic signs. Since traffic signs are basically composed of the pure colors: red, blue, white and yellow.

In [5] Gavrila suggests shape- based detection; detection is performed using a template-based correlation method to identify potential traffic signs in images; this involves so-called distance transforms (DTs).

In [3] Hatzidimos provides a hybrid solution for detection by applying the following process to detect the location of the sign in the image:
Region of Interest segmentation with image thresholding, thinning and Edge Detection, Region identifying and region clustering, line Detection to check the kind of shape in the region of interest, and finally shape check and calculating angles between lines.

Classification, on the other hand is performed also using different approaches; shape-based [4], color-based [1][3], or both.

As we have mentioned before, in [1] Pixels of an interest color treated as feature pixels, all other pixels treated as non-feature pixels.

In [2] Kogan applies completely different approach; Traffic signs after being segmented from the original scene are classified by the application of a Behavioral Model of Vision.

In [3] Hatzidimos color based classification approach is applied; templates (images in the DB) are transformed to the search image coordinate System. So that templates can be matched to search images. Matching is performed by calculating cross correlation coefficient between the search image and the template.

Challenges in the previous approaches

Current methods of traffic sign recognition have some weaknesses that ban them from acting as comprehensive problem solutions. Even the fact that those methods have suggested excellent ways [1][3][4] to detect and/or classify signs; we cannot mention one comprehensive method to perform recognition.

1.-Traffic sign image database must contain representative number of images. In addition, images of a database must be containing various traffic sign classes.

2-Methods in general are not robust in terms of partial occlusions, and illuminations.

Chapter Two: Theoretical Background

Introduction

Digital image

Color models

Image features

Edge detection

Convolution

2.1 Introduction

Our project performs analysis of traffic sign recognition problem. Recognition consists of two high level image processing operations: Detection and Classification. To understand our works one must first be aware of some basic concepts in image processing.

2.2 Digital image

An image stored in 2D form and divided into a matrix of pixels, each pixel consists of one or more bits of information that represent either the brightness, or the color, of the image at that point.

2.3 Color models

RGB

"The RGB color model shown in figure(2)[8] is an additive color model in which red, green, and blue light are added together in various ways to reproduce a broad array of colors. The name of the model comes from the initials of the three additive primary colors, red, green, and blue. RGB can be described as the computer's native color space for capturing images and displaying them. As human eyes are sensitive to these primary colors red, green, and blue all colors are perceived as a combination of these three colors. The RGB color model, based on a Cartesian coordinate system.

The main purpose of the RGB color model is for the sensing, representation, and display of images in electronic systems, such as televisions and computers, though it has also been used in conventional photography. Before the electronic age, the RGB color model already had a solid theory behind it, based in human perception of colors.

Levels of R, G, and B can each range from 0 to 100 percent of full intensity. Each level is represented by the range of decimal numbers from 0 to 255 (256 levels for each color). An example of RGB component is shown in figure (3) [9].

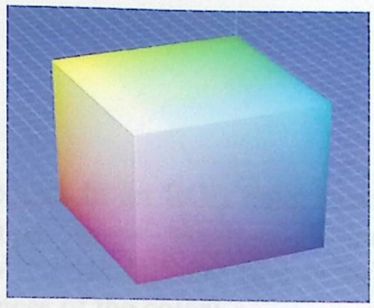


Figure (2): RGB color model.

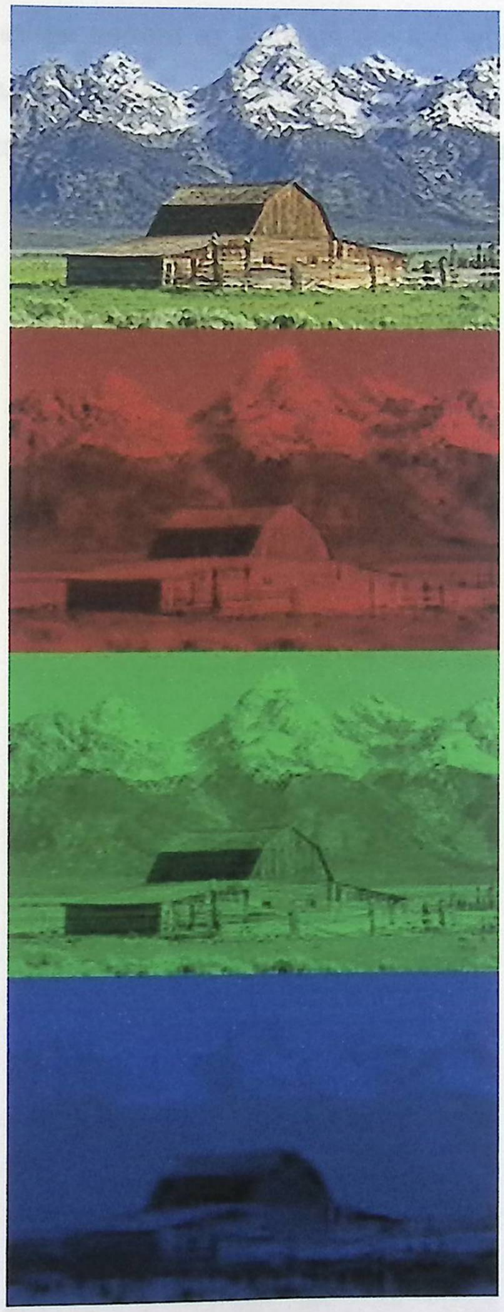


Figure (3): RGB image, with its separate R, G and B component .

HSV

HSV is a representation of points in an RGB color space, which attempt to describe perceptual color relationships more accurately than RGB, while remaining computationally simple. HSV stands for hue (the color), saturation (intensity of the color), value (brightness/darkness of the color).

HSV describes colors as points in a cylinder as shown in figure(4)[10] whose central axis ranges from black at the bottom to white at the top with neutral colors between them, where angle around the axis corresponds to “hue”, distance from the axis corresponds to “saturation”, and distance along the axis corresponds to “lightness”, “value”, or “brightness”. An example of HSV modle is shown in the figure (5)[11] .

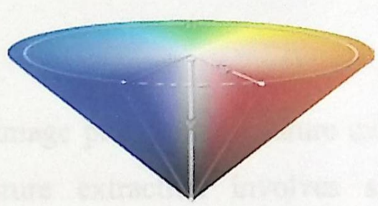
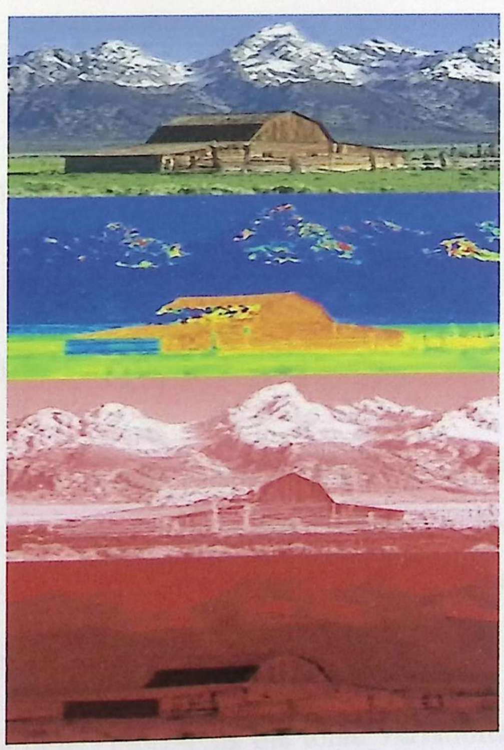


Figure (4): HSV color model.



Figure(5): HSV image, with its H,S and V component.

2.4 Image features

Definition of a feature

There is no universal or exact definition of what constitutes a feature, and the exact definition often depends on the problem or the type of application. Given that, a feature is defined as an "interesting" part of an image, and features are used as a starting point for many computer vision algorithms. Since features are used as the starting point and main primitives for subsequent algorithms, the overall algorithm will often only be as good as its feature detector. Consequently, the desirable property for a feature detector is repeatability: whether or not the same feature will be detected in two or more different images of the same scene.

Feature extraction

In pattern recognition and in image processing, Feature extraction is a special form of dimensionality reduction. Feature extraction involves simplifying the amount of resources required to describe a large set of data accurately. A feature might be global representation on an image (e.g. Histograms), or local representation (e.g. SIFT).

Histograms

We will talk about a type of histogram namely image histogram and orientation histogram.

1. Image histogram

An image histogram is type of histogram which acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance.

The horizontal axis of the graph represents the tonal variations, while the vertical axis represents the number of pixels in that particular tone.

2. Edge/Orientation histogram

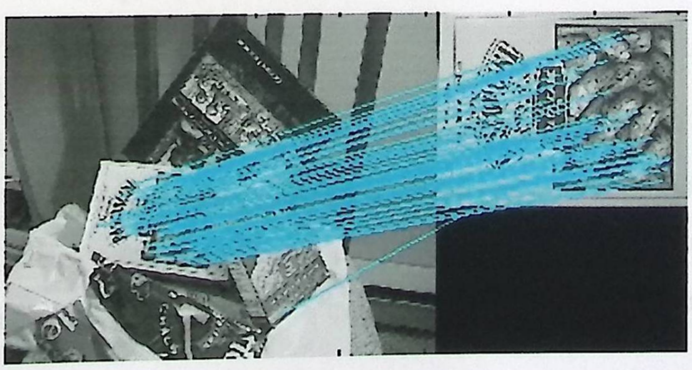
Histogram of Oriented Gradient Descriptors, or HOG descriptors, are feature descriptors used in computer vision and image processing for the purpose of object detection. The technique counts occurrences of gradient orientation in localized portions of an image.

SIFT

Scale-invariant feature transform (or SIFT) is an algorithm in computer vision to detect and describe local features in images. The algorithm was published by David Lowe in 1999.

SIFT allow us to efficiently match small portions of cluttered images under arbitrary rotations, scaling, change of brightness and other transformations by break the image into many small overlapping pieces of varying size, each of which is described in a manner invariant to the possible transformations. Then each part can be individually matched

The image below Figure(6)[12] shows an example of matching between two images. Features are extracted from each of the two images, and lines are drawn between features that have close matches.



Figure(6) : SIFT image [12].

The SIFT algorithm has four stages:-

1- Scale-space extrema detection

$$D(x,y,\sigma) = L(x,y,k_i\sigma) - L(x,y,k_j\sigma)$$

where $L(x,y,k\sigma)$ is the original image $I(x,y)$ convolved with the Gaussian blur $G(x,y,k\sigma)$ at scale $k\sigma$, i.e.,

$$L(x,y,k\sigma) = G(x,y,k\sigma) * I(x,y)$$

2- Keypoint localization

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}}\mathbf{x} + \frac{1}{2}\mathbf{x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2}\mathbf{x}$$

where D and its derivatives are evaluated at the candidate keypoint and $\mathbf{x} = (x,y,\sigma)$ is the offset from this point.

3- Orientation assignment

The Gaussian-smoothed image $L(x,y,\sigma)$ at the keypoint's scale σ is taken so that all computations are performed in a scale-invariant manner. For an image sample $L(x,y)$ at scale σ, the gradient magnitude, $m(x,y)$, and orientation, $\theta(x,y)$, are precomputed using pixel differences:

$$m(x,y) = \sqrt{\left(L(x+1,y) - L(x-1,y)\right)^2 + \left(L(x,y+1) - L(x,y-1)\right)^2}$$

$$\theta(x,y) = \tan^{-1}\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right)$$

4- Keypoint descriptor

Previous steps found keypoint locations at particular scales and assigned orientations to them. This ensured invariance to image location, scale and rotation. Now we want to compute descriptor vectors for these keypoints such that the descriptors are highly distinctive and partially invariant to the remaining variations, like illumination, 3D viewpoint, etc.

Although the dimension of the descriptor, i.e. 128, seems high, descriptors with lower dimension than this don't perform as well across the range of matching tasks, and the computational cost remains low.

Longer descriptors continue to do better but not by much and there is an additional danger of increased sensitivity to distortion and occlusion.

2.5 Edge detection

Edge detection is a terminology in image processing and computer vision, particularly in the areas of feature detection and feature extraction, to refer to algorithms which aim at identifying points in a digital image at which the image brightness changes sharply or more formally has discontinuities.

Edges characterize boundaries and are therefore a problem of fundamental importance in image processing. Edges in images are areas with strong intensity contrasts – a jump in intensity from one pixel to the next. Edge detecting an image significantly reduces the amount of data and filters out useless information, while preserving the important structural properties in an image. Figure(7) and figure(8) shows an example of edge detection using different kernel.

Figure(7): Detect edges using the Sobel method [13].

Figure(8): Detect edges using the Laplace method [13].

2.6 Convolution

Convolution: the mathematical, *local* operation is central to modern image processing.
It has the general form

- ***A*** is an image.
- ***B*** is called the *convolution kernel*
- The weights in ***B*** are the values of a filter.

The 2D convolution operation requires 4-double loop, so it isn't extremely fast, unless you use small kernel.(e.g. 3*3).

Chapter Three: *Design*

Introduction

Traffic sign image database

System description

Preprocessing

Algorithms analysis

3.1 Introduction

In this chapter research design is discussed in terms of:

- Traffic sign image database, in this part research team illustrates the properties of the images tested by SIFT and orientation histogram algorithms, and mention some conditions that a database must meet in order to fit with research requirements.
- Algorithms selected as a solution for traffic sign recognition problem. Some suggested solutions to solve the problem of traffic sign recognition are introduced in this chapter with details.
- Systems overall description. The system that delivers reasonable solution for traffic sign recognition problem.

3.2 Traffic sign image database

To apply SIFT and orientation histogram for traffic sign recognition, test, and analyze them we need a database. A database of traffic sign images with suitable properties for research requirements.

Database with enough number of images, images of whole scenes, taken by one camera, and representing a few various traffic sign classes is necessary to perform tests and obtain results.

So, in our research we dealt with two Traffic sign image database each of which consists of a set of images. Each image of the Database represents whole scene of street, people and other objects in addition to a traffic sign.

We used two databases, the first is already built database from the internet the other is built by research team.

We applied SIFT and orientation histogram on both databases. But the already built database downloaded from the internet did not meet research requirements; it contains signs that represent a small number of traffic signs. Also, it has small number of images. The already built database downloaded from the internet contains 48 images, each of size (360x270) pixels, in PNG format, each representing a traffic scene.

The images are grouped in 3 classes, each class corresponding to a different template traffic sign (pedestrian crossing, compulsory for bikes, intersection) as shown in table(2) below:

Pedestrian crossing	
Compulsory for bikes	
Intersection	

Table(2): the 3 class of the first database

We have 16 images for each class. The images of the template traffic signs and the traffic scenes were taken with a digital photo-camera.

We applied selected method and our method on the images of this database during the first period in our research time. Tests and experiments on those images were helpful as a basic to extend experiments later, since this database meets research requirements.

Then after we finished analyzing methods using the first database we built our own database which contains 300 images of size (1376x1032) pixels in JPG format, each representing a traffic scene. Despite the lack of traffic signs in our surrounding, we were able to capture images that are grouped in 10 classes, each class corresponding to a different template traffic sign, as shown in table(3) below. The traffic scenes were taken with digital cell phone camera.

Class	Sample Image
Crossr	
Walk	

Stop	
Pedestrian	
Park	

Taxi	
Straight to Right	
Straight to left	

Curved to Left	
Slop	

Table(3): the 10 classes of the second database

We have 30 images for each class. Images of this database meet the research requirement, and the database contains enough number of images with various classes.

3.3 System description

In our system we apply two consecutive methods to perform detection and classification: SIFT and orientation histogram. Preprocessing must be done on input image before applying the system.

Main steps performed in the system for detection and classification include:

1. SIFT implementation.
2. Orientation histogram extraction.
3. Distance computation

Preprocessing

Preprocessing is any operation that must be applied on an image before it can be tested in a system. The operations to make an image suitable for system requirements.

Images must be in PGM (Portable Gray Map) format for further processing using SIFT and orientation histogram. Since SIFT requires image to be in PGM format to process it. Converting images from colored images into PGM images is accomplished using Xnview software.

Algorithms analysis

We used SIFT to perform detection and basic classification step. Orientation histogram and distance calculations are made as final steps in classification.

SIFT

We use SIFT to perform detection and basic classification processes. In classification SIFT is applied to place images into the closest classes depending on number of matches between images in the database (Systems training database) and input image.

SIFT basic processes:

1. Key point descriptor extraction from two images in PGM format.
2. Perform matching between the two images depending on key point descriptors extracted from each one. This produces number of matches between the two images. However, matching is performed for all images in the system's database each at once with the input image.

The highest the number of matches the closest class to the sign in the image is recognized. The system picks the closest two images with highest two numbers of matches when compared to input image.
Figure(9) below show The result from applying SIFT is the recognition of the traffic sign from a complex scene.

Figure(9):e.g. of applying SIFT.

Input image is "matched" with all images in the training database; two images with the highest number of matches from the database are then passed to next step with the input image.

Orientation histogram

Research team selected orientation histogram as a unique feature for classification. Feature vector is created using orientation histogram for images in system's database. When an image in PGM format and it had SIFT applied on it, its orientation histogram is produced.

Sobel edge detection method was applied to perform edge detection and creating orientation histograms.

Sobel performs convolution process using the following two kernels :

-1	0	1
-2	0	2
-1	0	1

Kx

1	2	1
0	0	0
-1	-2	-1

Ky

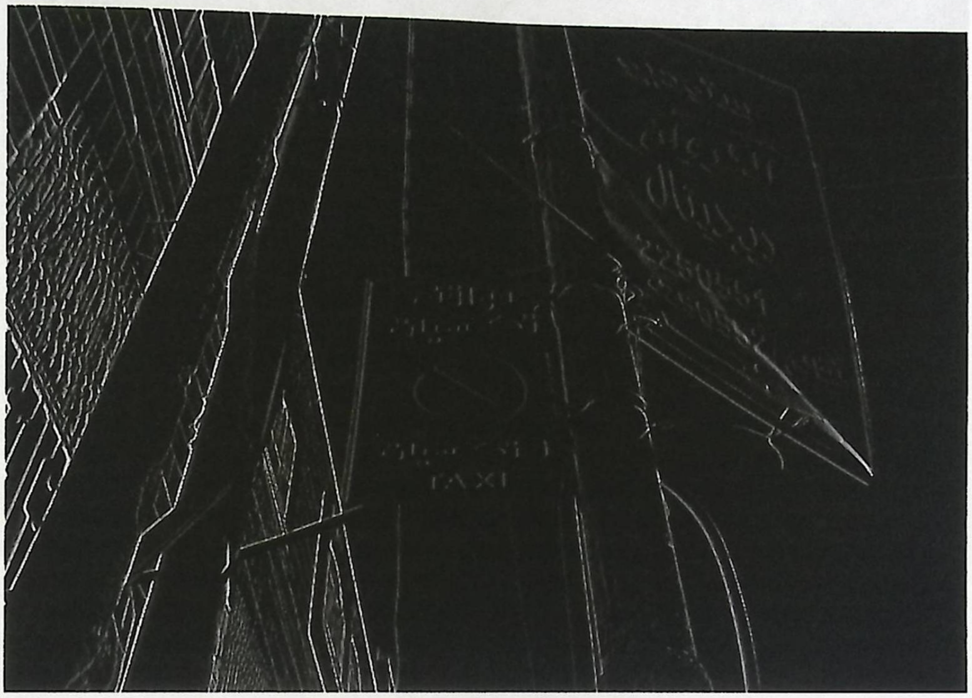
A convolution is applied to image twice, once in each direction. The method suggests that convolution is applied on the input image using Kx kernel and the Ky kernel as two separated convolution operations.

Each convolution process when applied on the input image show in figure(10) two new images are produced: result of convolution on x direction (Gx) show in figure(11) and the result on y direction (Gy) show in figure(12).

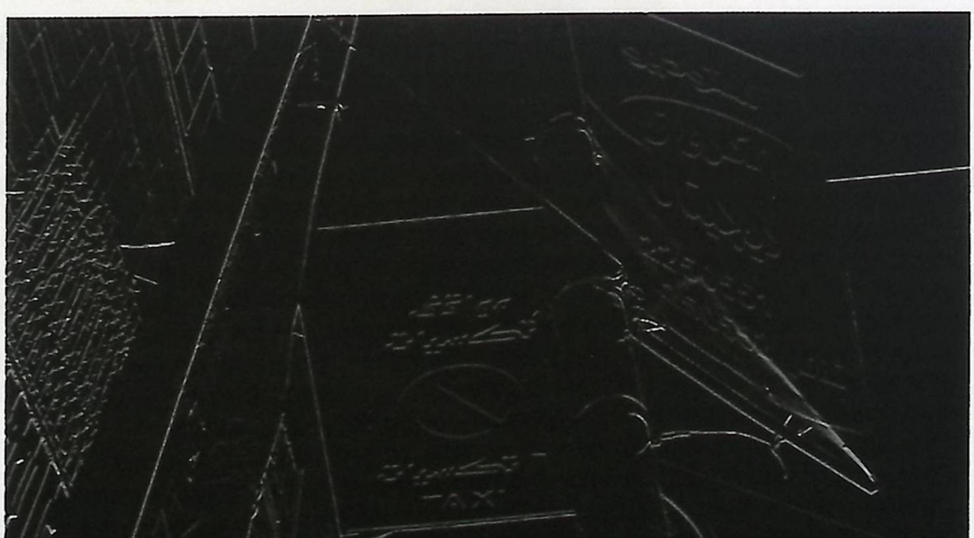
Figure (10): Query image.

Those images represent magnitude of each pixel:

$$G = \sqrt{Gx^2 + Gy^2}$$



Figure(11): x direction



Figure(12): y direction

Magnitude is used to find edges show in figure (13).

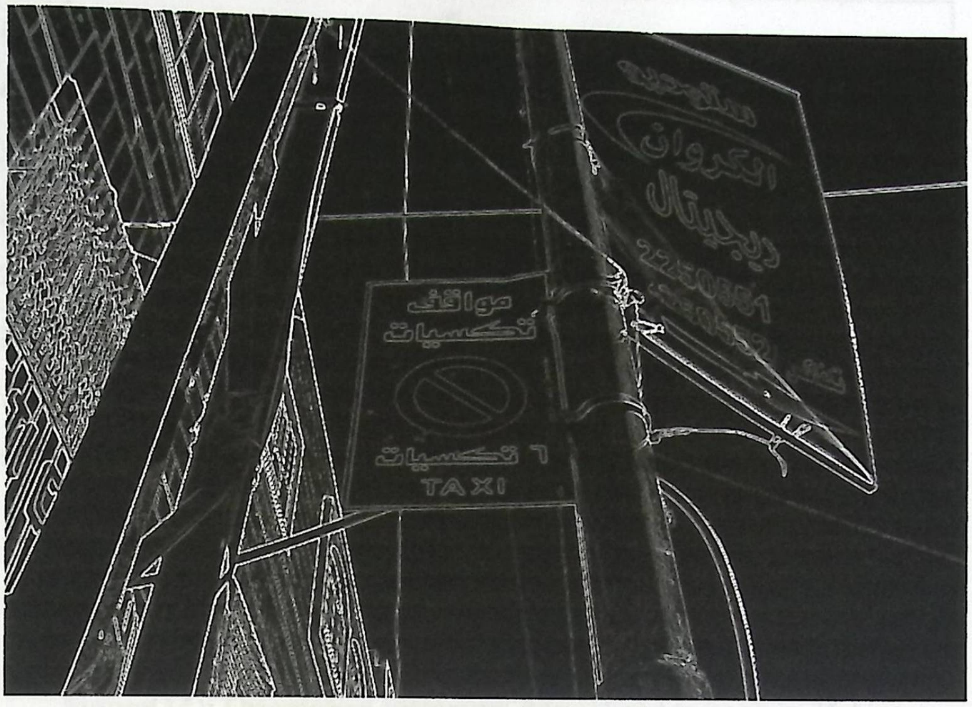


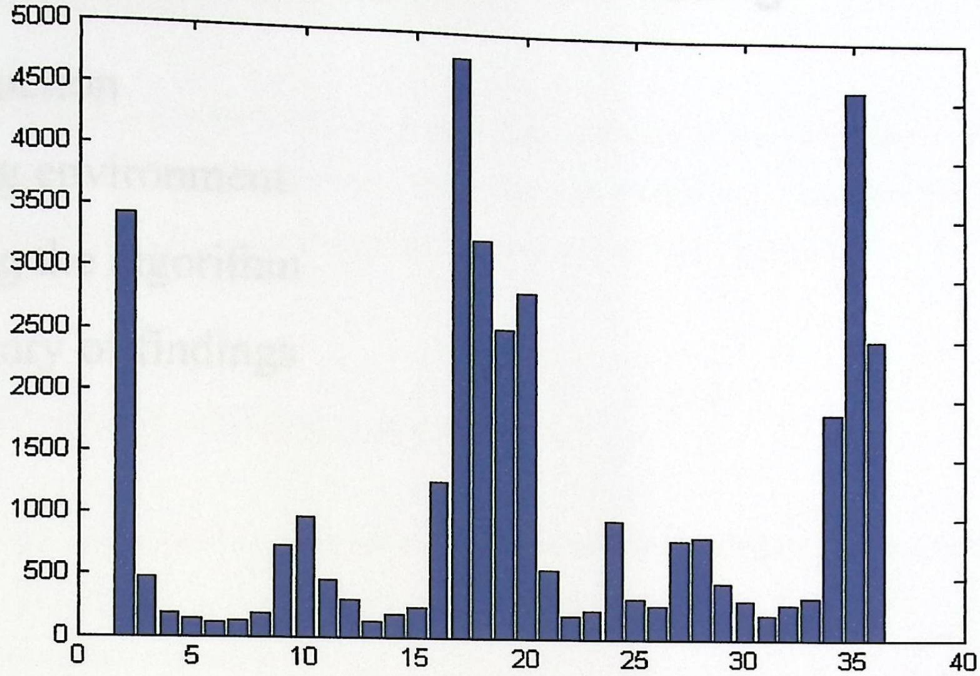
Figure (13):Edge detection

The orientation (angle) of each pixel is computed as specified in the following formula:

$$\theta = \arctan(Gy/Gx)$$

Histograms are produced using theta's values for three images: the input image and the two images from the training database with highest number of matches when applying SIFT.

An example of Orientation histogram normalized to contain 36 bins is shown in figure (14) .



Figure(14): Orientation histogram normalized to contain 36 bins.

Distance is then computed between the input image's histogram and each of the two images histogram using the following formula:

$$\textbf{\textit{Distance}} = \sqrt{\sum_{i=1}^{36} \left(OHist1(i) - OHist2(i)\right)^2}$$

Lower distance means much close image. So, less distance is selected to indicate that the input image is classified.

Combining the two algorithms improves performance of the orientation histogram and provides good classification success rate.

Chapter four: Experiment & Testing

Introduction

Testing environment

Testing the algorithm

Summary of findings

4.1 Introduction

This chapter is concerned with the experiments and tests of the performance of SIFT and orientation histogram algorithms for traffic sign recognition. Tests were performed using Matlab program and were applied on images from both databases.

4.2 Testing environment

Tests were applied on two different databases (illustrated in chapter 3). Lenovo, N100, Dual core laptop is used to conduct tests. Matlab 7.0, is the program used to test the algorithm and report results

To perform tests, a smaller database was extracted from each main database as a training set.

From the first database (the one downloaded from the internet) 12 images were selected to build the training set. Four images from each class were contained in this set.

In the second database (the database built by research team) the 300 images were tested using a training set of 70 images. Seven images were chosen to represent each class. Training set images were selected to represent a class in different conditions; illumination, rotation, and weather conditions.

To conduct tests all images in the training sets must be preprocessed and converted into PGM (Portable Gray Map) format. So, Xnview program is used to perform the conversion.

Each image in the training sets contains only a traffic sign representing one distinctive class.

4.3 Testing the algorithm

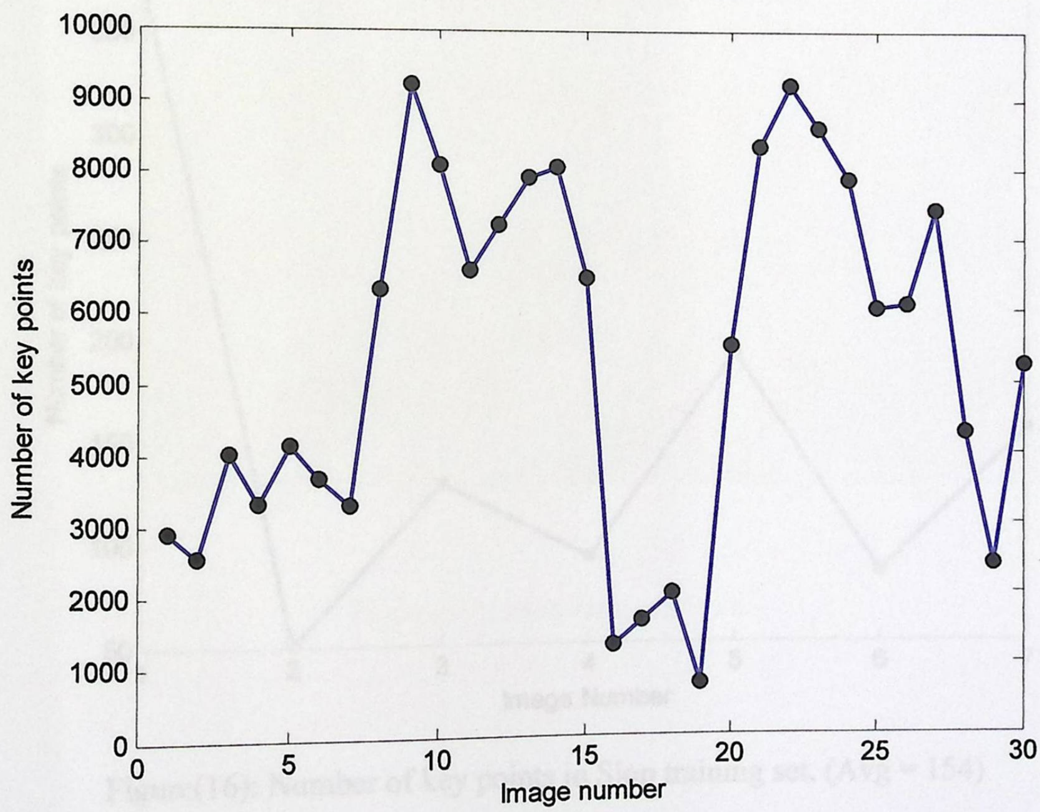
SIFT

Scale Invariant Feature Transform (SIFT) is used to perform detection and classification. The main steps performed by SIFT to accomplish classification process are:

1. Finding key points

SIFT accepts two PGM images as parameters; query image that contains the whole scene captured by a camera and the image from the training set.

Key descriptor is created for each image for further processing as shown in figure (15).

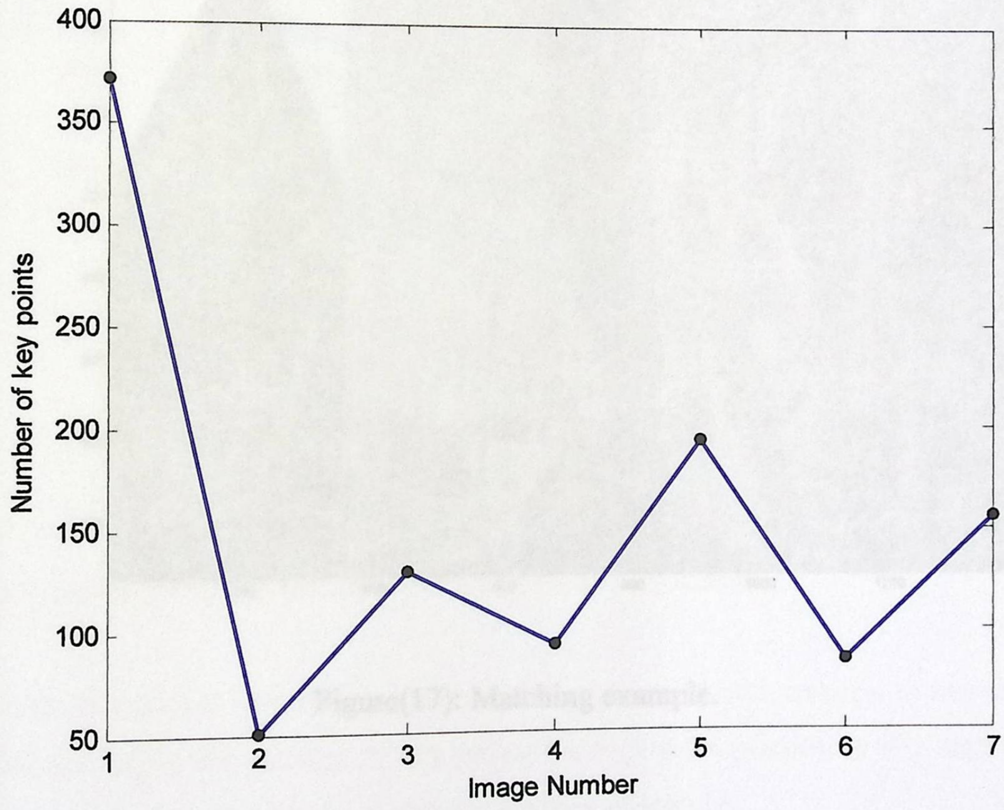


Figure(15): Key points descriptor of the Slop class.(Avg = 5.3505e+003)

Number of key points differs from one image to another due to each image's unique properties.

Different illumination conditions for example causes SIFT to produce different key descriptors for different images even if they were of the same class.

Images in a training set representing the same class also must have different number of key points to indicate the uniqueness of each image as shown in figure (16) below.



Figure(16): Number of key points in Slop training set. (Avg = 154)

Higher number of key points increases the possibility of having a successful recognition but it does not guarantee success.

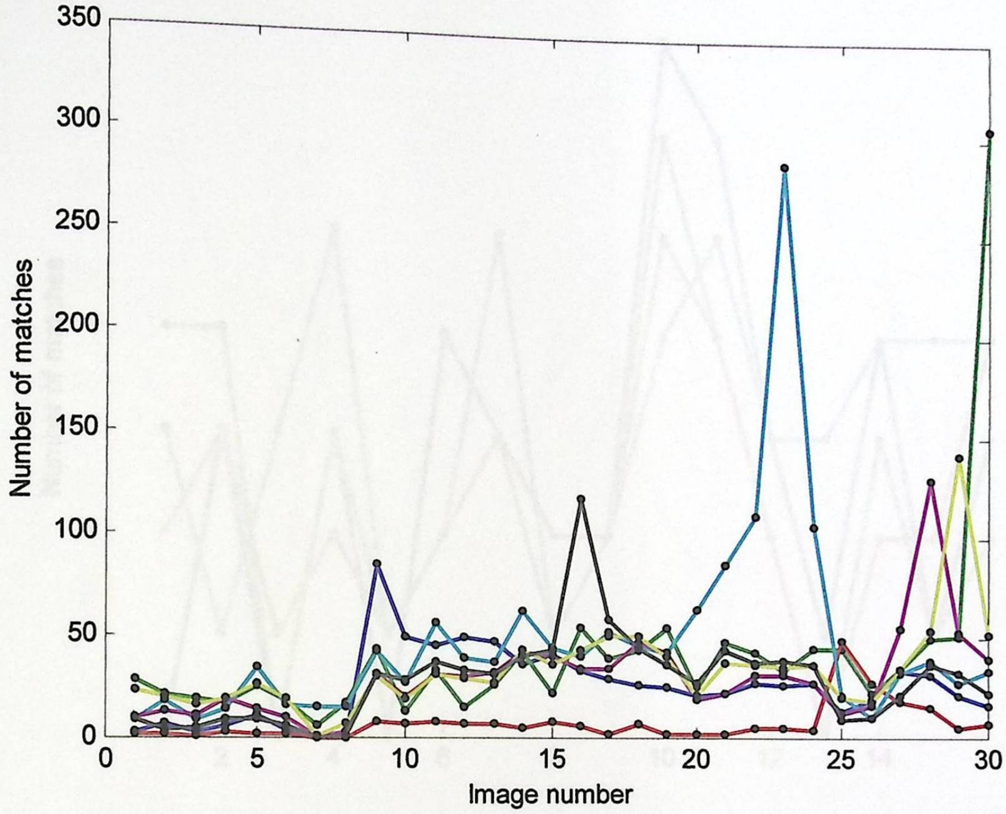
2. Matching

Matching in SIFT is the operation that follows the extraction of the two key descriptors (One for the query image and the other for the image from the training set).

Match takes the two descriptors as parameters, locates key points on both images and then link similar key points with each other as shown in figure(17) below.

Figure(17): Matching example.

The figure(18) shows number of matches between seven images representing one class from a training set and thirty images representing whole scenes.



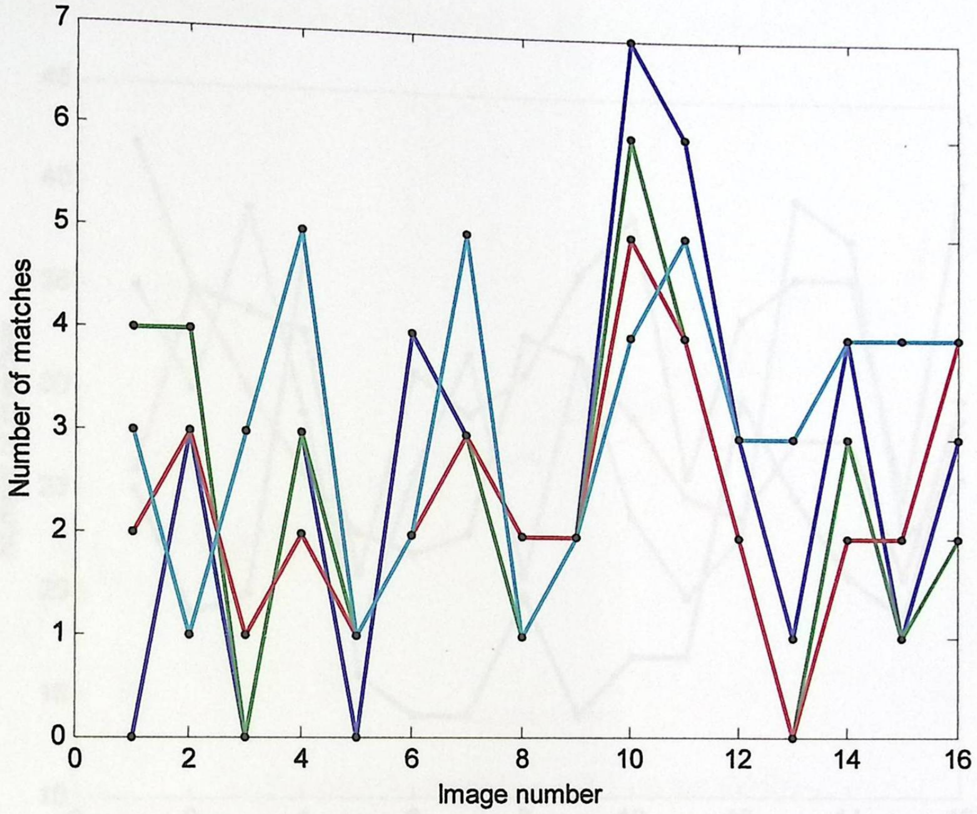
Figure(18): Number of matches example. (Avg = 30.5857)

Each colored curve in the figure above represents the number of matches between certain image from the training set and thirty different images.

Number of matches depends on number of key points in both images, as mentioned before the higher the number of key points the higher the possibility of a sign to be recognized but not always the higher the number of matches.

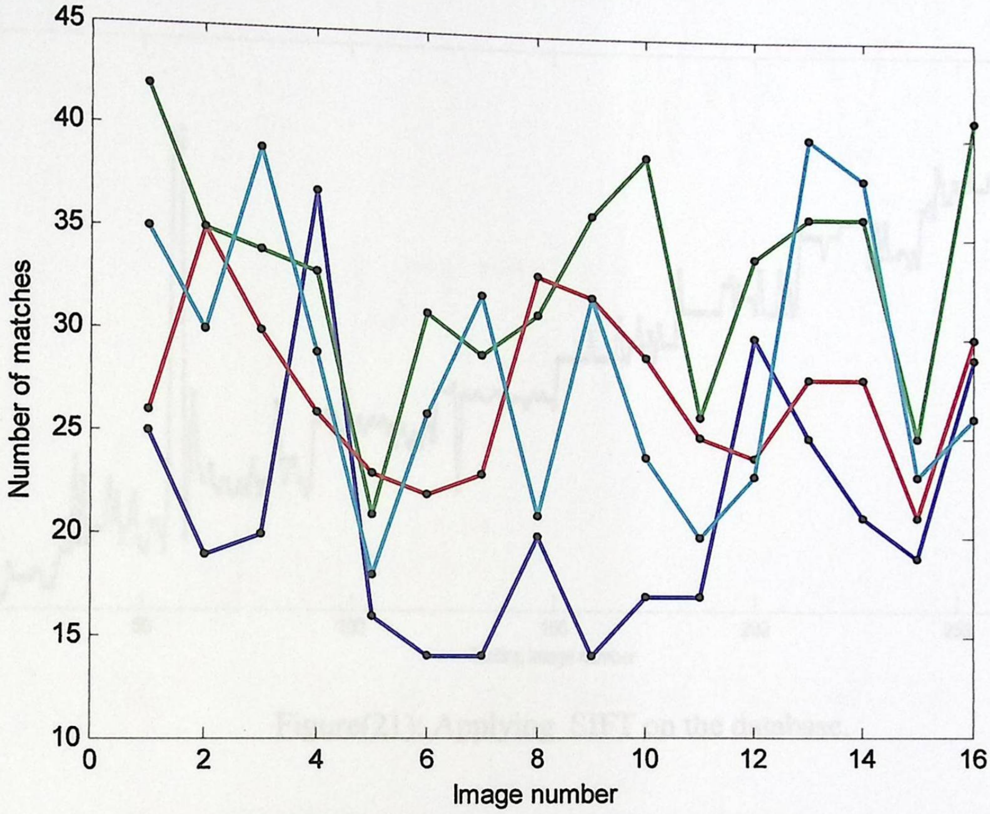
Another factor that affects number of matches is the size of images. Larger or smaller images cause number of matches to vary.

In figure (19) below number of matches of original images when matched to training set is shown.



Figure(19): Number of matches of original images

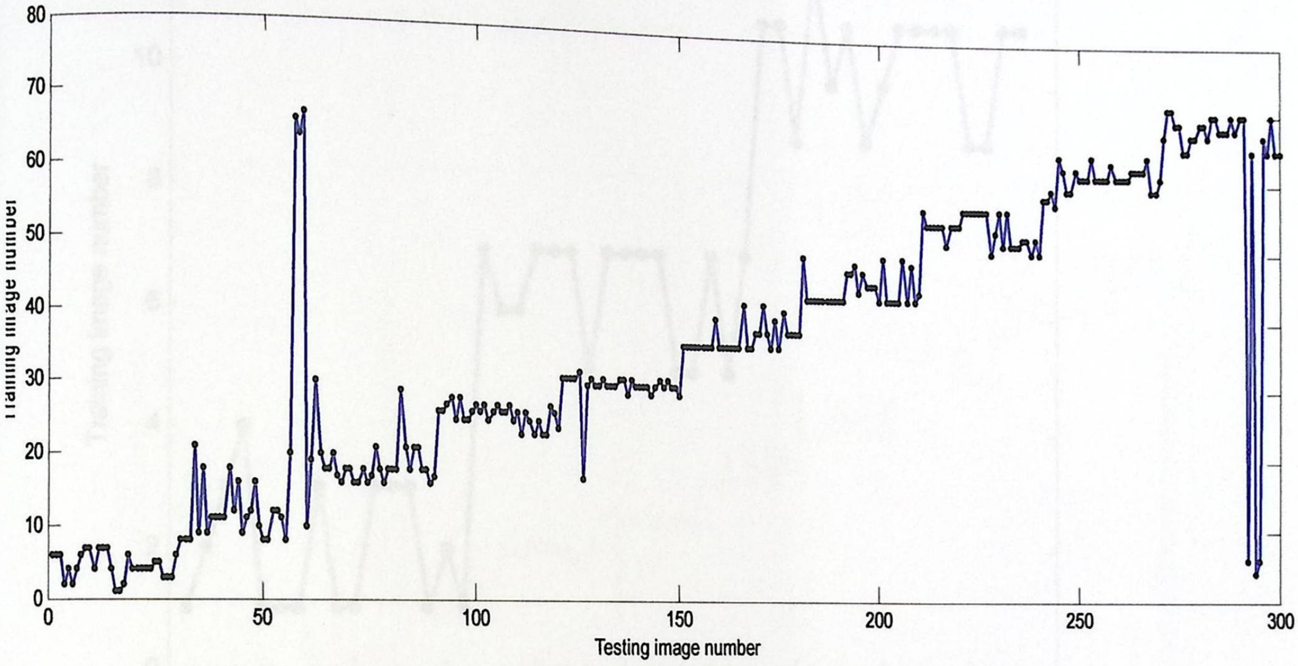
In figure(20) illustrates number of matches when the same testing images are doubled then matched against the same training set.



Figure(20) : Number of matches (Doubled size images).(Avg = 27.4531)

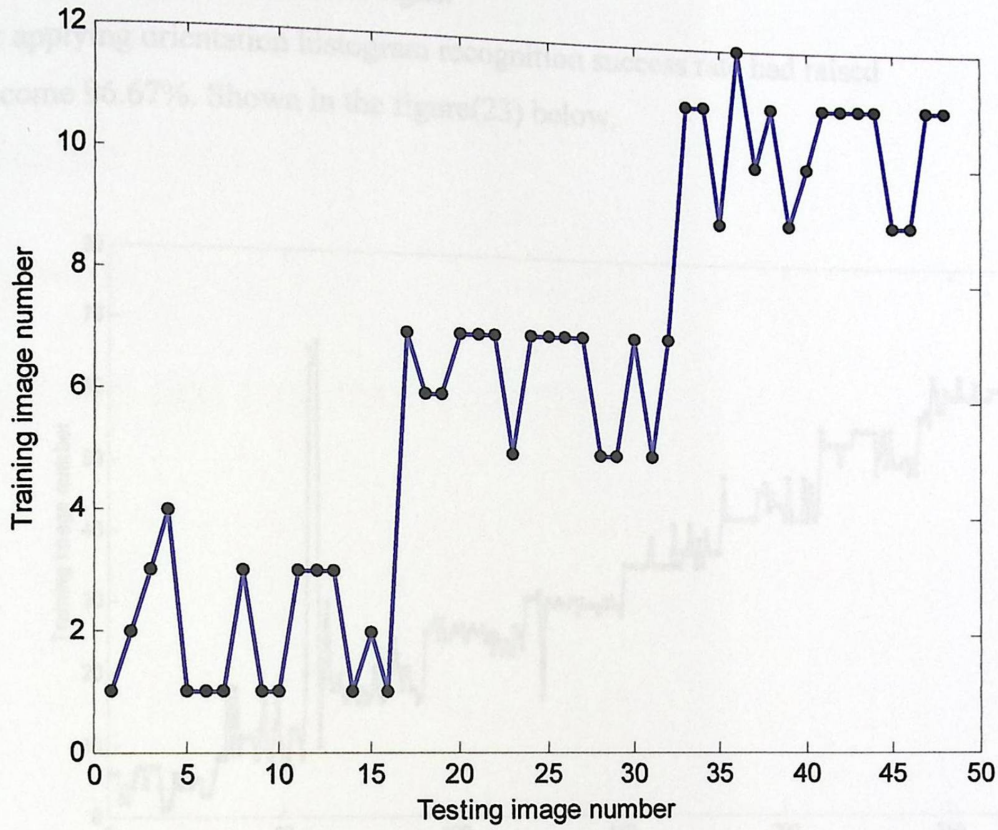
Overall recognition

SIFT testing applied on our database showed a success rate of 95%. SIFT misclassified 15 images out of 300. Recognition is illustrated in the figure(21) below.



Figure(21): Applying SIFT on the database.

When applied on the other database (downloaded from the internet) SIFT showed a success rate of 100%. As shown in figure(22) below.

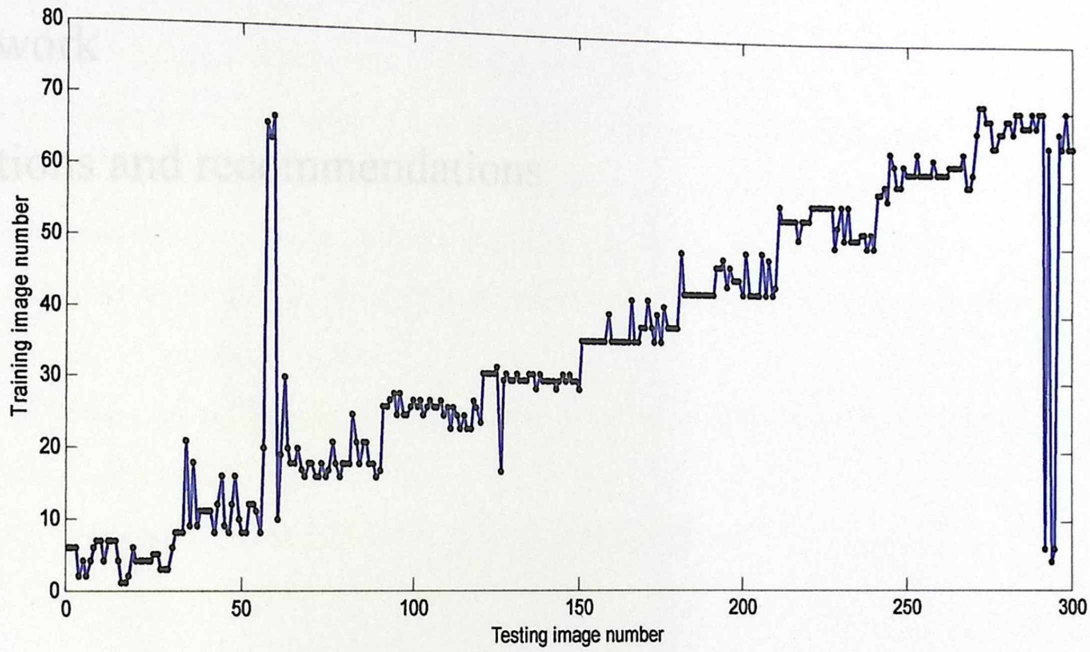


Figure(22): SIFT implication on the internet database

Orientation histogram

Research team suggests the use of orientation histogram to improve SIFT results and to give better recognition of traffic signs.

After applying orientation histogram recognition success rate had raised to become 96.67%. Shown in the figure(23) below.



Figure(23): Applying orientation histogram after SIFT on the database.

4.4 Summary of findings

Applying SIFT and edge orientation histogram results can be summarized as follows:

Applying SIFT on the images of the first database had shown success rate of (100%).

Applying edge orientation histogram then had shown success rate of (100%) with (48) images out of (48) were classified correctly.

Applying SIFT followed by Edge Orientation Histogram on the second database

Applying SIFT on the images of the second database had shown success rate of (95%). with (285) images out of (300) were classified correctly.

Applying edge orientation histogram then had shown success rate of (96,67%) . with (290) images out of (300) were classified correctly

Estimated time needed for the recognition process is 2- 3 seconds.

Chapter Five: Conclusion & future work

Introduction

Conclusion

Future work

Suggestions and recommendations

5.1 Introduction

This chapter is concerned with conclusion and future work of the research .

5.2 Conclusion

- Many people need this system to recognition the traffic sign.
- The characteristic of traffic sign, consisting of the original colors red and blue that are not secondary or composite colors makes it that easy to discover of a sign from complex image with objects other than traffic sign. Also traffic signs have distinctive shapes, which helped researchers finding different ways in their attempts to recognize signs.
- SIFT and edge orientation histogram have proven their excellence when applied on almost all traffic signs.
- We have applied these method on two different databases.

5.3 Future work

Research team developed a method to detect the traffic sign .

Our detection method uses HSV color model. Detection process is accomplished by several consecutive steps:

First: Converting to HSV color model

Conversion is applied to images: since traffic signs are composed from pure colors (Red, Blue, and Yellow), and to reduce the effect of variations in illumination.

Formulas used to perform conversion are:

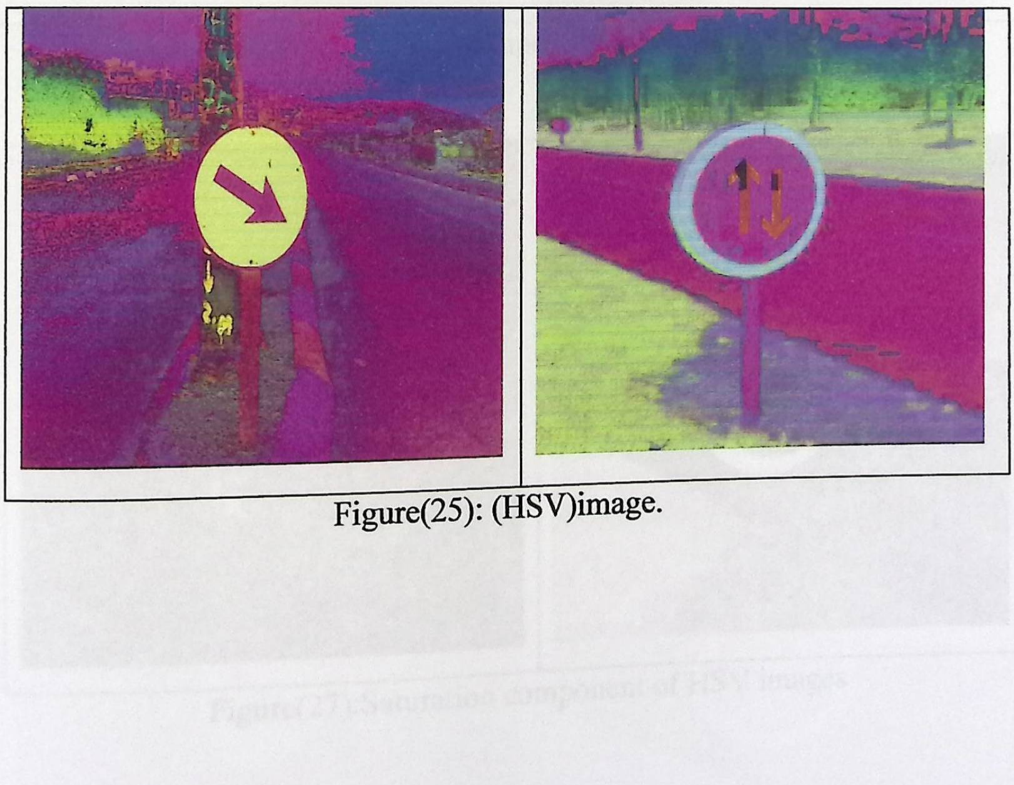
$$h = \begin{cases} 0 & \text{if } \max = \min \\ (60^\circ \times \frac{g-b}{\max - \min} + 0^\circ) \bmod 360^\circ, & \text{if } \max = r \\ 60^\circ \times \frac{b-r}{\max - \min} + 120^\circ, & \text{if } \max = g \\ 60^\circ \times \frac{r-g}{\max - \min} + 240^\circ, & \text{if } \max = b \end{cases}$$

$$s = \begin{cases} 0, & \text{if } \max = 0 \\ \frac{\max - \min}{\max} = 1 - \frac{\min}{\max}, & \text{otherwise} \end{cases}$$

$$v = \max$$

The input image in figure (24) convert to HSV image in figure(25).

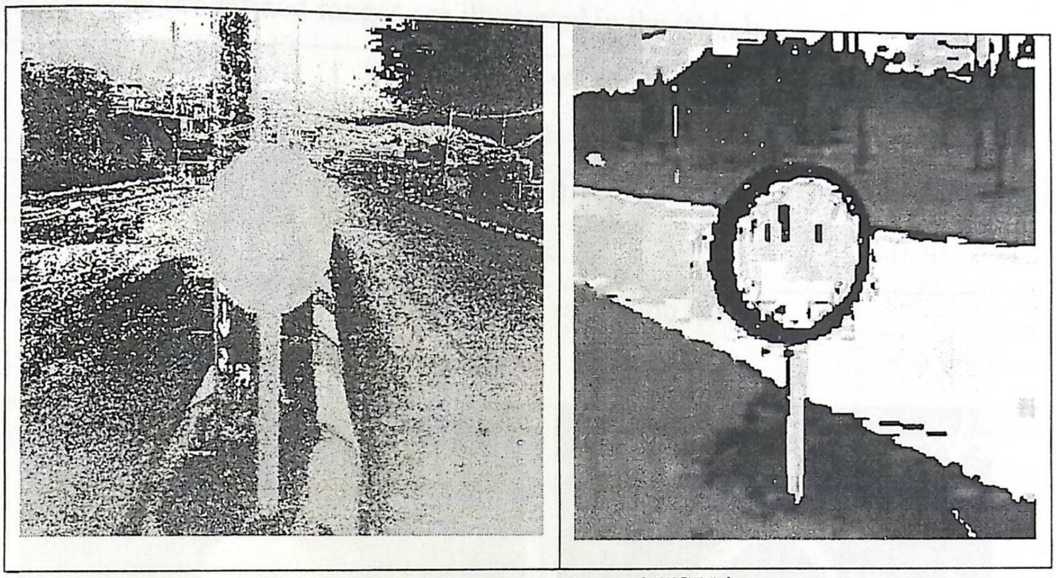
Figure (24):Input images sample(RGB).



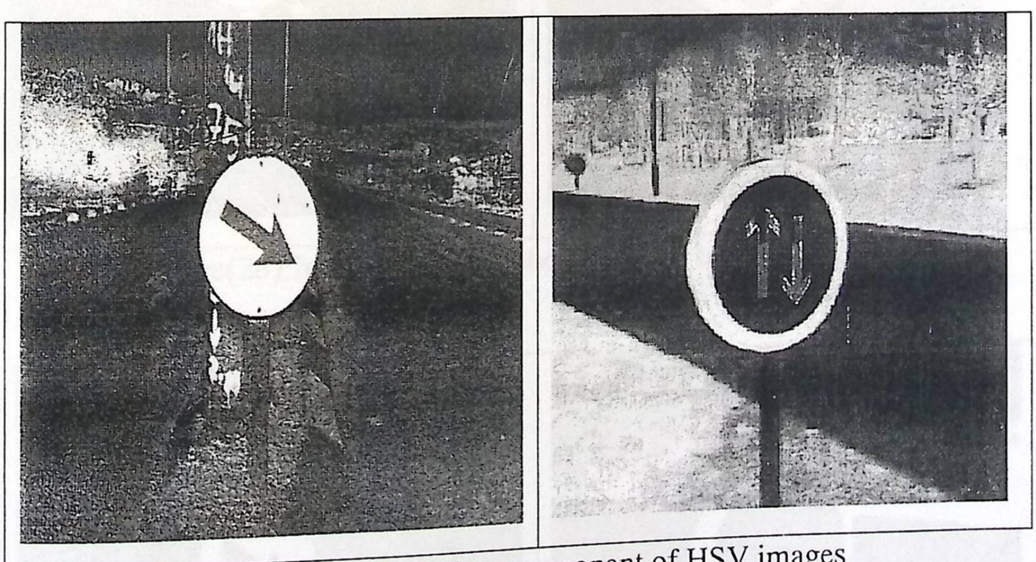
Figure(25): (HSV)image.

Second: Taking Hue, Saturation Component.

Hue is the color itself as shown in figure(26), while saturation is the depth of the color as shown in figure(27). Hue and Saturation are the most significant components of HSV color model. Extracting them as separate components from the original image after converting it into HSV color model.



Figure(26):Hue component of HSV images.



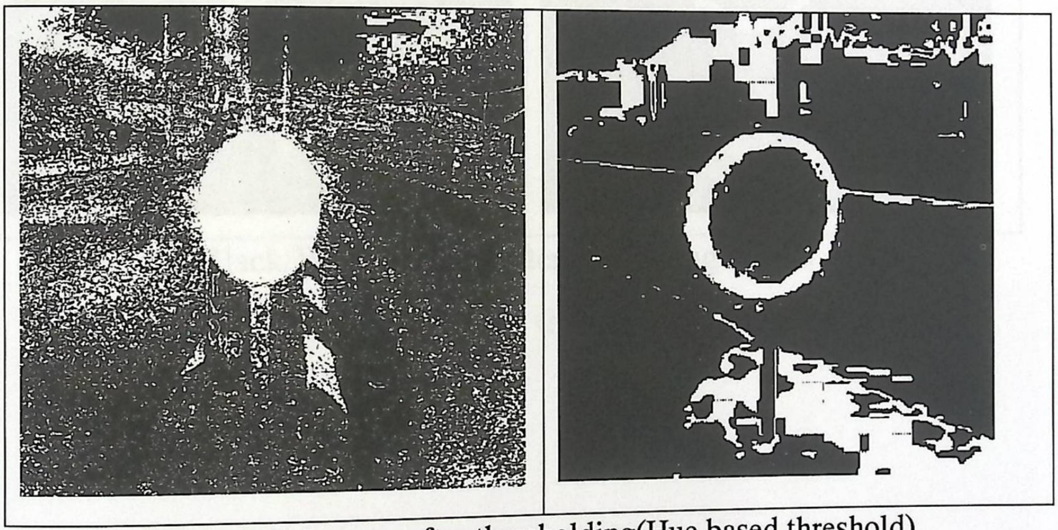
Figure(27):Saturation component of HSV images

Third: Thresholding.

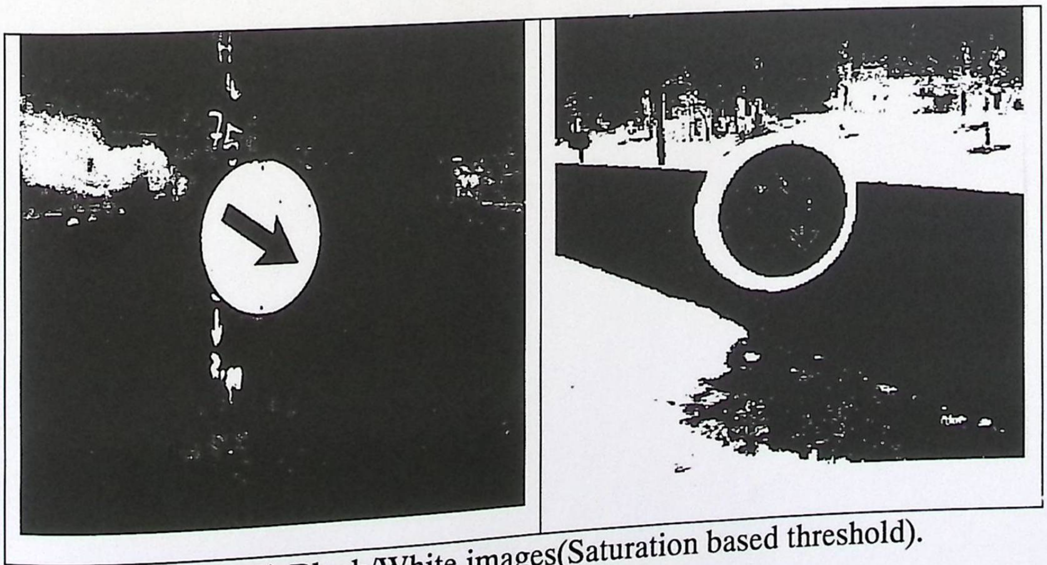
After two new images for the whole scene are produced: one is Hue image the other is Saturation, thresholds are used to produce new image containing only traffic sign . Using suitable ranges of Hue and Saturation that represent traffic sign colors: (Red, Blue) to produce two images: hue as shown in figure(28) and saturation as shown in figure(30). The predefined ranges are illustrated in the table below.

Color	Range of Hue component (V)	Range of Saturation component (S)
Red	0.00 - 0.1	> 0.40
Blue	0.50 – 0.69	

Table(4):Hue and saturation range.



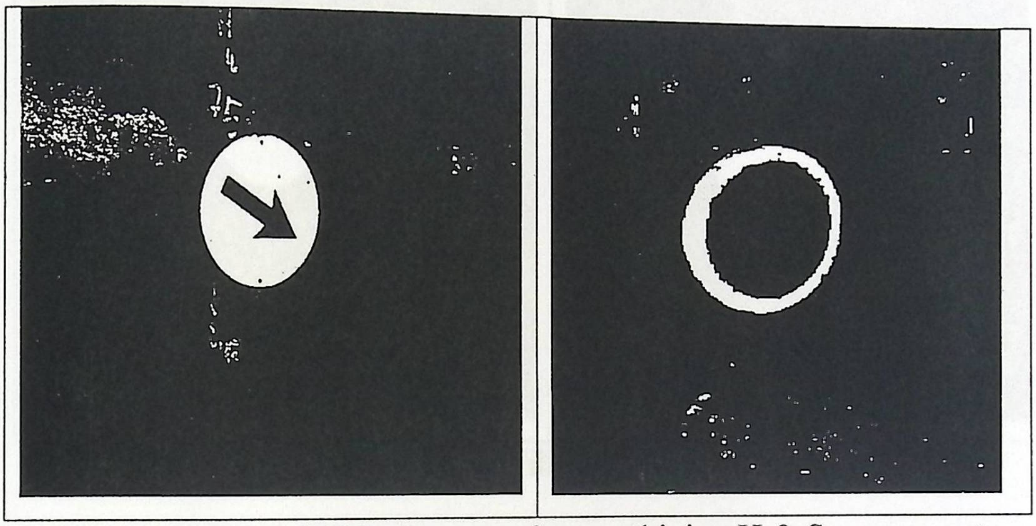
Figure(28):Images after thresholding(Hue based threshold).



Figure(29):Black/White images(Saturation based threshold).

Forth: Exporting a B/W Image

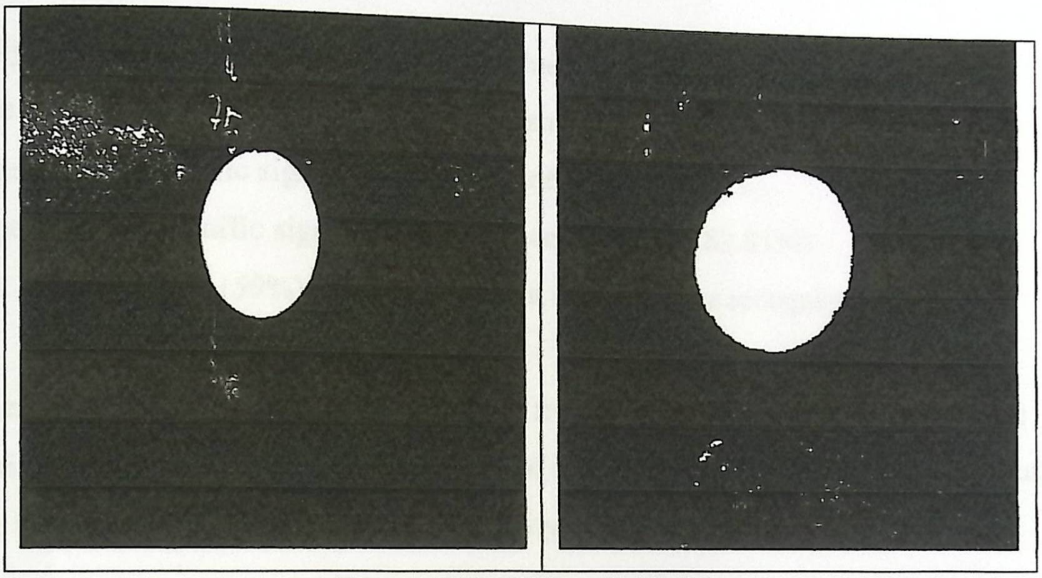
Black/White image is then produced by combining Hue and Saturation images (resulting after thresholding) in one image where the sign's position is determined. White is the shape of traffic sign only on a black background. (i.e. White empty circle on black background) as in figure(30).



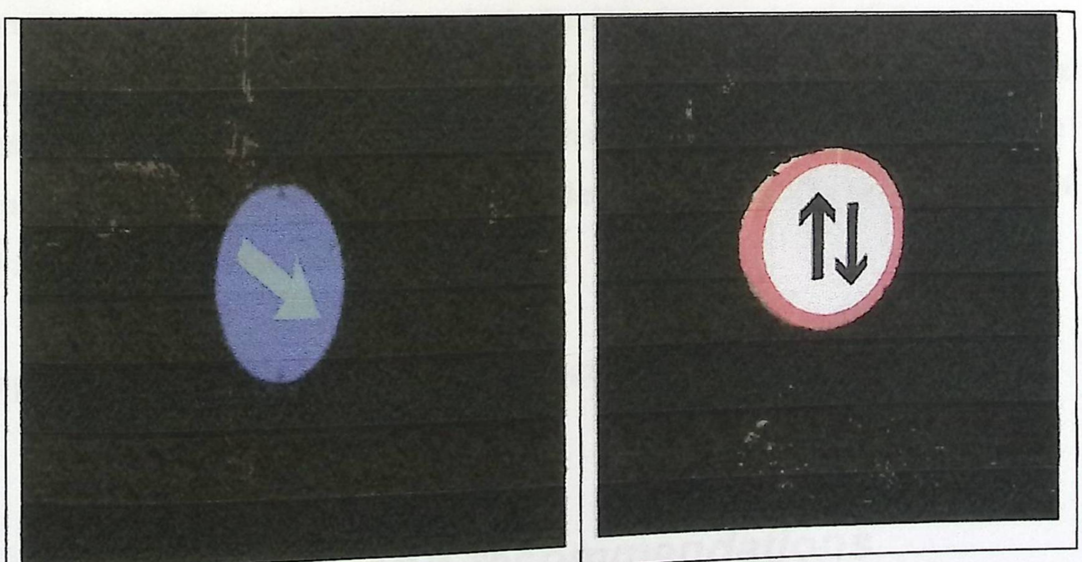
Figure(30):Black/White images after combining H & S components (using H & S thresholds).

Finally: Extraction The sign.

By filling holes in the Black/White image as shown in figure(31), and then reading pixels from the original image that matches white pixels in the Black/White images shown in figure(32).



Figure(31):Filling the holes.



Figure(32):Anding with the original image.

Applying our method on the first database

Success rate varies as sign color varies:

Images with red traffic sign have shown success rate of (23%).

Images with blue traffic sign have shown success rate of (82.85%).

Total success rate is (66.6%), (32) images out of (48) were recognized correctly.

Applying our method on the second database

Since our method depends on color models success rate varies as sign color varies:

Images with red traffic sign have shown success rate of (3.3%).

Images with blue traffic sign have shown success rate of (82.85%).

Total success rate is (59%), (177) images out of (300) were recognized correctly.

Implementation of our algorithm on images from the two databases showed that the algorithm is very effective in the detection of sign: it specifies the signs location, but there is a weakness in cases of color similarity between signs and other areas of the image.

From the result of tests, the aspects which should be improved in the future are:

• Increase the response of algorithm to light condition changes that have an important impact, because these changes affect color thresholds. Thresholding (determining the accurate range of color) is very important in the success of the sign location detection and sign's final recognition.

• Improve the algorithm so it can detect red sign more effectively.

5.4 Suggestions and recommendations

Our search would have produced more accurate results if we:

- Captured images using a different camera
- Captured images of clear signs (signs we found weren't clear and most of times features were destroyed).

References:

1- Volker Rehrmann Raimund Lakmann Lutz Priese, " A parallel system for real–time traffic sign recognition", Image Recognition Laboratory
University of Koblenz–Landau.

2 - X. Gao, N. Shevtsova, K. Hong, S. Batty, L. Podladchikova, A. Golovan, D. Shaposhnikov, V. Gusakova ," Vision Models Based Identification of Traffic Signs." Research Institute for Neurocybernetics, Rostove State University, Rostov-on-Don, Russia.

3 - John Hatzidimos, " Automatic traffic sign recognition in digital images".

4 - Andrzej Ruta Yongmin Li and Xiaohui Liu," Towards Real-Time Traffic Sign Recognition by
Class-Specific Discriminative Features", School of Information Systems, Computing and Mathematics Brunel University, Uxbridge, Middlesex UB8 3PH, UK

5- D.M. Gavrila, " Traffic Sign Recognition Revisited", Image Understanding Systems, DaimlerChrysler Research, Germany.

6- K. Ek¡stein, L. Bellada, " Recognition of Traffic Signs ", Dept. of Computer Science and Engineering,
University of West Bohemia, Plze¡n,
Czech Republic

7- http://www.domkrat.ru/Laws/rules/znak1.shtml.

8- www. Ctr.columbia.

9- en.wikipedia.org.

10- www.codeproject.com

11- en.wikipedia.org/wiki/HSL_color_space

12- en.wikipedia.org/wiki/Scale-invariant_feature_transform

13- www.pages.drexel.edu/~weg22/edge