

**TRAFFIC SIGN RECOGNITION USING SCALE  
INVARIANT FEATURE TRANSFORM AND COLOR  
CLASSIFICATION METHOD**

**M.Sc. Thesis by**

**Merve Can KUŞ, B.Sc.**

**Department : Computer Engineering**

**Programme: Computer Engineering**

**JUNE 2008**

**TRAFFIC SIGN RECOGNITION USING SCALE INVARIANT  
FEATURE TRANSFORM AND COLOR CLASSIFICATION  
METHOD**

**M.Sc. Thesis by  
Merve Can KUŞ, B.Sc.  
504061523**

**Date of submission: 05 May 2008  
Date of defence examination: 10 June 2008**

**Supervisor (Chairman) : Prof. Dr. Muhittin GÖKMEN  
Members of the Examining Committee Prof. Dr. Coşkun SÖNMEZ (Y.T.Ü)  
Yrd. Doç. Dr. Şima ETANER-UYAR**

**JUNE 2008**

**İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ**

**SIFT (SCALE INVARIANT FEATURE TRANSFORM) VE RENK  
SINIFLAMA YÖNTEMİNİ KULLANARAK TRAFİK İŞARETİ  
TANIMA**

**YÜKSEK LİSANS TEZİ  
Müh. Merve Can KUŞ  
504061523**

**Tezin Enstitüye Verildiği Tarih : 05 Mayıs 2008  
Tezin Savunulduğu Tarih : 10 Haziran 2008**

**Tez Danışmanı : Prof. Dr. Muhittin GÖKMEN  
Diğer Jüri Üyeleri Prof. Dr. Coşkun SÖNMEZ (Y.T.Ü.)  
Yrd. Doç. Dr. Şima ETANER-UYAR**

**HAZİRAN 2008**

## ACKNOWLEDGEMENTS

I would like to express my utmost gratitude to my thesis advisor, Prof. Dr. Muhittin Gökmen and Assist. Prof. Dr. Şima Uyar for their invaluable guidance throughout my work and for keeping me focussed in my research. They have shown a large and consistent interest in my project and it were our fruitful discussions and their constructive comments that have hugely contributed to my thesis work. I would not be performing this work without their continual encouragement.

Initial phases of the thesis work were conducted in the Image Processing Laboratory pertaining to the Computer Engineering Faculty of Istanbul Technical University. For this, I want to thank Prof. Dr. Muhittin Gökmen for creating me such an environment.

I am particularly thankful to my mother Ayşe and father Erten for their support throughout my life and for continuous effort to my education that always motivated me towards achieving great things in my life.

Special thanks are due to my friend and colleague Caner Kömürlü for his help and patience, expecting nothing in return during my thesis work.

Above all, I would like to thank my husband, Anar Khalilov who stood beside me and encouraged me constantly. He has always been a source of motivation and I want to thank him for assistance during my thesis work.

May, 2008

Merve Can Kuş

## TABLE OF CONTENTS

<b>ABBREVIATIONS</b>	<b>v</b>
<b>TABLE LIST</b>	<b>vi</b>
<b>FIGURE LIST</b>	<b>vii</b>
<b>SYMBOL LIST</b>	<b>viii</b>
<b>ÖZET</b>	<b>ix</b>
<b>SUMMARY</b>	<b>x</b>
<b>1. INTRODUCTION</b>	<b>11</b>
<b>2. PROPERTIES OF TRAFFIC SIGN DETECTION AND RECOGNITION SYSTEMS AND THE DIFFICULTIES</b>	<b>16</b>
2.1. Detection	16
2.1.1. Using Color Information in the Detection of Traffic Signs	17
2.1.1.1. The Hue-Saturation-Value (HSV) Color Model	17
2.1.2 Using Shape Information in the Detection of Traffic Signs	20
2.2. Recognition	20
2.3. The Difficulties	21
<b>3. SCALE INVARIANT FEATURE TRANSFORM (SIFT)</b>	<b>28</b>
3.1. Scale-Space Extrema Detection	29
3.2. Keypoint Localization	32
3.3. Orientation Assignment	35
3.4. Keypoint Descriptor	36
<b>4. IMPLEMENTATION</b>	<b>38</b>
4.1. Development Environment	38
4.2. Recognition Methods	38
4.2.1. Recognition by Using only SIFT Features	38
4.2.2. Recognition by Using SIFT Features and Color Information	39
4.2.2.1. Color Classification Method	40
4.2.2.2. Thresholding based on Euclidean distance of [H,S,V] vectors	42
4.2.2.3. Comparison	43
4.2.3. Recognition by Using SIFT features with Color and Orientation Information	46
4.3. Training Images	46
4.4. Test Images	49

<b>5. RESULTS</b>	<b>51</b>
5.1. Recognition Results of Extracted Traffic Sign Images	51
5.2. Recognition Results of Test Images without Extracting Traffic Signs	56
<b>6. CONCLUSION</b>	<b>59</b>
6.1. Future Work	60
<b>REFERENCES</b>	<b>62</b>
<b>APPENDIX A</b>	<b>69</b>
<b>APPENDIX B</b>	<b>75</b>
<b>BIOGRAPHY</b>	<b>108</b>

## **ABBREVIATIONS**

<b>GIS</b>	: Geographical Information Systems
<b>HSI</b>	: Hue-Saturation-Intensity (Color Model)
<b>HSV</b>	: Hue-Saturation-Value (Color Model)
<b>ITS</b>	: Intelligent Transport Systems
<b>L*a*b*</b>	: Lab Color Model
<b>PCA-SIFT</b>	: Principal Component Analysis-Scale Invariant Feature Transform
<b>RGB</b>	: Red-Green-Blue
<b>SIFT</b>	: Scale Invariant Feature Transform
<b>YIQ</b>	: Luminance-In phase-Quadrature
<b>YUV</b>	: Luminance-Bandwidth-Chrominance

## TABLE LIST

	<u>Page No</u>
<b>Table 4.1</b> Color Classes and Corresponding Rules .....	41
<b>Table 4.2</b> Confusion Matrix Created for the Classification of Sample Patches ..	42
<b>Table 4.3</b> Matching Results of the Two Methods for 18 Test Images .....	45
<b>Table 4.4</b> Training Set .....	47
<b>Table 4.5</b> Number of Test Images for Each Traffic Sign that Exist in the Training Set .....	50
<b>Table 5.1</b> Confusion Matrix Created for the Recognition of Test Images by the First Method .....	54
<b>Table 5.2</b> Confusion Matrix Created for the Recognition of Test Images by the Second Method .....	55
<b>Table 5.3</b> Confusion Matrix Created for the Recognition of Test Images by the Third Method .....	56
<b>Table A.1</b> Complied Color Classes .....	69
<b>Table A.2</b> H, S, V Values for Sample Patches with Known True Colors .....	71
<b>Table B.1</b> Recognition Results of Extracted Traffic Sign Images .....	75
<b>Table B.2</b> Recognition Results of Test Images without Extracting Traffic Sign Regions .....	93

## FIGURE LIST

	<u>Page No</u>
<b>Figure 2.1</b> : General Traffic Sign Detection and Recognition System .....	16
<b>Figure 2.2</b> : The Hue, Saturation, Value (HSV) Color Model .....	18
<b>Figure 2.4</b> : Examples of Variation in the Color of the Traffic Signs with Time. 21	
<b>Figure 2.5</b> : Effect of Weather Conditions on the Appearance of the Traffic Sign .....	22
<b>Figure 2.6</b> : Examples of Shadowed Traffic Signs .....	23
<b>Figure 2.7</b> : Effect of the Direction and the Strength of Light on the Color Perceived .....	24
<b>Figure 2.8</b> : Viewing Orientation Examples .....	25
<b>Figure 2.9</b> : Difficulties in Traffic Sign Recognition. (a), (b) Examples of Disoriented Traffic Signs. (c), (d), (e) Examples of Occluded Traffic Signs .....	26
<b>Figure 2.10</b> : Change in Size of a Traffic Sign Depending on the Camera Distance .....	27
<b>Figure 2.11</b> : Blurring Caused by Motion .....	27
<b>Figure 3.1</b> : Matching of Two Images which Include the Same Object .....	28
<b>Figure 3.2</b> : Example of Object Recognition for an Image Containing 3D Objects .....	29
<b>Figure 3.3</b> : Set of Scale Space Images at the Left, Difference-of Gaussian Images on the Right .....	31
<b>Figure 3.4</b> : Detection of Maxima and Minima of Difference-of-Gaussian Images .....	31
<b>Figure 3.5</b> : Keypoint Selection. (a) Original Image. (b) The Initial Locations of 832 Keypoints at Maxima and Minima of the Difference-of-Gaussian Function. (c) Remaining 729 Keypoints after Applying a Threshold on Minimum Contrast. (d) The Final 536 Keypoints that Remain after Applying Threshold on Ratio of Principal Curvatures .....	34
<b>Figure 3.6</b> : Creation of a Keypoint Descriptor .....	36
<b>Figure 4.1</b> : Example Test Image from the Test Set used Evaluations of the Thresholding Based on Euclidean Distance of [H,S,V] Vectors Method .....	43
<b>Figure 4.2</b> : Mean Euclidean Distances Calculated for the SIFT Matches with Same and Different Colors .....	44
<b>Figure 4.3</b> : Different Versions of the Same Traffic Sign .....	49
<b>Figure 4.4</b> : Two Versions of Same Traffic Sign with Different Colors and Shapes .....	49

## SYMBOL LIST

$I(x,y)$	: Arbitrary image
$\sigma$	: Scale
$\sigma^2 \nabla^2 G$	: Scale-normalized Laplacian of Gaussian
$L(x,y,\sigma)$	: Scale space function
$G(x,y,\sigma)$	: Variable-scale Gaussian function
$D(x,y,\sigma)$	: Difference-of-Gaussian function
$\hat{x}$	: Location of the extremum of $D(x)$
$D(\hat{x})$	: Function value at the extremum
$H$	: Hessian matrix
$Tr(H)$	: Trace of Hessian matrix
$Det(H)$	: Determinant of Hessian matrix
$\alpha$	: Eigenvalue of Hessian matrix with the largest magnitude
$\beta$	: Smaller eigenvalue of Hessian matrix
$r$	: Ratio between the largest magnitude eigenvalue and the smaller one of Hessian matrix
$m(x,y)$	: Gradient magnitude of keypoint
$\theta(x,y)$	: Orientation of keypoint

## SIFT (SCALE INVARIANT FEATURE TRANSFORM) VE RENK SINIFLAMA YÖNTEMİNİ KULLANARAK TRAFİK İŞARETİ TANIMA

### ÖZET

Literatürde trafik işareti tespiti ve tanıma problemi için önerilen birçok teknik ve konu üzerinde yapılan birçok çalışma bulunsa da, bu çalışmaların hiçbirinde SIFT (Scale Invariant Feature Transform) yöntemi, yöntemle tanıma başarımını arttıracak yeni özellikler katılıp kullanılmamıştır. Bu tezin amacı, yerel öznitelik tekniği olması açısından SIFT yönteminin bu problemdeki davranışını incelemektir.

SIFT, verilen bir resimde yerel değişmez öznitelikler bulur, bu öznitelikleri daha önceden oluşturulmuş bir veritabanında bulunan trafik işareti resimlerinin özniteliklerine eşler ve en çok sayıda eşleşmenin olduğu trafik işaretini bularak trafik işaretlerini tanır.

SIFT anlatıldığı şekilde kullanılarak tanıma başarımları elde edildikten sonra, SIFT yönteminde bulunan eşleşmelerin doğruluğunu kontrol ederek tanıma başarımını arttıracak yeni özellikler katılması amaçlanmıştır. Bunun için, eşleşmelerin doğruluğunu sınarken önerilen renk sınıflama yöntemi ile renk kontrolü ve SIFT özniteliklerinin yönlerinin kontrolü yapılmıştır.

Geliştirilen renk sınıflama yöntemi bazı sınıflara ayırma kuralları kullanarak piksellerin gerçek renklerini bulur. Bu yöntemin gerçek renklere karar vermekteki başarısının gerçekten iyi olduğu elde edilen sonuçlar tarafından gösterilmektedir. Ayrıca, yöntemin rengin eşleşmelerin doğruluğunu sınarken kullanılmasında akla gelebilecek diğer yöntemlerden daha iyi başarımlar sağladığı da görülmüştür. Yön kontrolü ile de eşlenen özniteliklerin aralarındaki yön farkının fazla olduğu eşleşmeler doğru kabul edilmeyerek yanlış eşleşmelerin çoğu elenmektedir.

Sonuç olarak, renk ve yön kontrollerinin eklenmesinin zaten SIFT yönteminin iyi olan tanıma başarımını ciddi oranda arttırdığı gözlenmiştir. Çeşitli açılarda dönmüş, afin dönüşümlere uğramış, hasarlı, diğer nesnelere tarafından bir kısmı örtülmüş, üzerine diğer nesnelere gölgesi düşmüş, renk değişimine uğramış, değişik hava koşullarında ve değişik aydınlatma koşullarında resimlenmiş trafik işareti resimleri için bile elde edilen tanıma sonuçlarının gerçekten çok iyi ve tatmin edici olduğu görülmüştür.

# **TRAFFIC SIGN RECOGNITION USING SCALE INVARIANT FEATURE TRANSFORM AND COLOR CLASSIFICATION METHOD**

## **SUMMARY**

Although many techniques are proposed and many implementations exist in the literature for the traffic sign detection and recognition issue, none of them added new features to Scale Invariant Feature Transform (SIFT) while using it in this issue in order to increase the recognition performance. The purpose of this thesis is investigating behavior of SIFT in this subject from the aspect that it is a local feature technique.

SIFT finds local invariant features in a given image and recognizes traffic signs by matching these features to the features of traffic sign images that exist in a database formed previously and finding out the traffic sign image that gives the maximum number of matches.

After obtaining the recognition performance as described the use of SIFT, adding new features to SIFT, which will increase the recognition performance by checking the accuracy of matches which are found by SIFT, is aimed. Color inspection by using proposed color classification method and inspecting the orientations of SIFT features are performed while investigating the accuracy of the matches.

Color classification method that is developed finds out true colors of the pixels by applying some classification rules. The results show that the performance of this method is very good at determining the colors and the method performs better than other solutions to this problem that can be thought. And also by the orientation inspection, most of the false matches are eliminated by discarding the matches between the features that have distant orientations.

As a consequence, it is observed that adding color and orientation inspections raises the recognition performance seriously, even using only SIFT without these inspections gives good results. Obtained results are very good and satisfying even for the images containing traffic signs which are rotated, have undergone affine transformations, have been damaged, occluded, overshadowed, had alteration in color, pictured in different weather conditions and different illumination conditions.

## 1. INTRODUCTION

Traffic sign detection and recognition play very important role in Intelligent Transport Systems (ITS) which contain the internet, mobile data services, smart sensors, artificial intelligence, position technologies and geographical information systems (GIS) [1]. This is due to the vital importance of the traffic signs at safety and easy drives since they warn the drivers against dangers and difficulties and help them during their navigation [2]. ITS carry out the duty of paying attention to the traffic signs in lieu of the drivers and give opportunity not to overlook any of the traffic signs where the drivers may miss many signs.

The goal of this thesis is to shed light over the subject of recognition of the traffic signs by using Scale Invariant Feature Transform (SIFT) [3] which is one of the local feature techniques. Besides, the aim is to develop new methods by adding new features to SIFT in order to increase the success at the recognition process. The main contribution of this thesis is the implementation of color inspection by using proposed color classification method and inspecting the orientations of SIFT features, which are used while investigating the accuracy of the matches which are found by SIFT. This approach seems to be a very good choice in order to increase the performance.

Local feature techniques first detect features for a given image and then a set of descriptors are computed for these features [4]. The features are the distinctive parts of an image. SIFT is selected from many local feature descriptors and is used in this thesis since it is observed that SIFT-based methods outperform many other descriptors like steerable filters, differential invariants, complex filters, moment variants, shape context, spin images and cross correlation for different type of interest points in the presence of different image transformations such as image rotation, scale changes, affine transformations and illumination changes [5, 6].

SIFT is shown to be successful in many areas like object recognition and matching, place recognition in which a mobile device or vehicle could identify its location by

recognizing familiar locations, robot localization and mapping problems [3, 7, 8]. The computation of SIFT features is also efficient, it is stated that several thousand keypoints can be extracted from a typical image with near real-time performance on standard PC hardware [3].

It is shown that extended versions of SIFT show best performance when compared to PCA-SIFT (Principal Component Analysis-Scale Invariant Feature Transform), Moment invariants and cross correlation in the pedestrian detection problem and in the recognition of objects on real images from the Caltech 101 database [4].

The superiority of SIFT to other descriptors used in literature on a number of measures is also stated in [9] which is a study that many authors who are interested in the subject of feature detectors and descriptors contribute and in the mentioned study while comparing detectors, SIFT is used as the descriptor.

In general, identification of traffic signs is achieved by two main stages: detection, and recognition. First, the location of the traffic sign found by the detection process and the outer edges of the signs are extracted, then recognition is performed in this area. However, by using a local feature technique as SIFT, detecting location of the traffic sign and extracting the outer edges as the first step are not required anymore, since SIFT features that are matched to the training image, which contains corresponding traffic sign, show the location of the sign in a given image containing a traffic sign. This situation facilitates and simplifies the whole process.

In the literature, if gray scale images are used in the detection step while extracting the outer edges of the signs, Genetic Algorithms, Hough transforms, and Neural Networks-based methods are preferred [10, 11].

In other studies colors are taken into consideration. In general, colors are used especially for the segmentation of traffic signs from the rest of a scene, different than utilization method of color information in this thesis. Color spaces which are not affected by the lightning changes as HSI (Hue-Saturation-Intensity) and HSV (Hue-Saturation-Value) which are very common and use hue-saturation, and also other color spaces like YIQ (Luminance-In phase-Quadrature), YUV (Luminance-Bandwidth-Chrominance),  $L^*a^*b^*$  (Lab color model) are used. It is stated that 70% of the segmentation approaches which benefit from color information use hue as basic color hint and 30% of them use other color spaces [1]. A small number of

authors developed databases of color pixels, look-up tables, and hierarchical region growing techniques. As color segmentation methods, simple methods which are very fast and preferable for real-time applications and complex methods like fuzzy or neural network-based methods, which are computationally costly but give more accurate results, are used. As a consequence, there is no standard method to extract colors from the color image that is under consideration [2, 12-14].

Neural networks are widely used in the detection and the recognition of the traffic signs, mostly as a classifier in the recognition step. Besides, they also play important role in color detection, shape detection, shape classification, and pictogram recognition. Back propagation neural networks are used and Kohonen maps are trained for occluded traffic signs or the signs which are rotated by small angles. ART1, ART2, Hopfield, Cellular neural networks are the examples which were used in different studies [15-17]. However, neural networks have problems as training overhead and the impossibility of the adaptation of the multi-layer neural networks for on-line application because of their architecture. A serious redesign penalty arises if there occurs an increase in the number of traffic sign classes since this architecture is fixed, and the new classes are not recognized until the entire network is retrained. As a consequence, neural networks do not have significant advantages over the template matching which is another method.

Template matching is the second widely used alternative in the recognition step [1]. It classifies the inner parts of traffic signs and it can also extract the local features of the sign when it is combined with wavelets. It is also combined with complex-log transform and 2D-FFT to get a better performance [18-20].

Some difficulties with previous algorithms like the slowness of template matching based techniques [21], the need for a large number of images for training like in the neural-based approaches [22], or the need for a priori knowledge of the physical characteristics of the lighting illumination of the signs [23] are overwhelmed by SIFT.

Clustering classifiers, nearest neighbor classifiers, Laplace kernel classifiers are the other methods used for the classification of traffic signs [1]. Other classifiers like classical classifier, weighted distance classifier, Angular histogram, matching

pursuit classifier, and Euclidean distance were also used for traffic sign classification [24].

Fuzzy classifiers and the fuzzy techniques are also good at recognizing and classifying traffic signs, but Fuzzy sets, Fuzzy classifiers and Fuzzy Inference Systems were not studied deeply in the detection and recognition of traffic signs. Fuzzy set theory was used for color detection in the study of Jiang and Choi [25]. Fuzzy rules were used to combine color and shape by Fang et al. [2]. More studies which use fuzzy techniques are needed to be done in the traffic sign recognition field. For example, neuro-fuzzy pattern recognition and Fuzzy ARTMAP classifiers are suggested for further investigation [1].

The combination of fuzzy techniques and the moments which are calculated for signs can be used as the combination of the neural networks with the moments. The central moments, invariant central moments, affine moment invariants, radial moments, and Zernike moments are the moment types which can be chosen as features descriptors to describe the signs [1].

The work presented in [26] detects traffic signs by red color thresholding and uses Breadth First Search to locate and extract the sign for recognition purpose. Then recognition is performed by using SIFT. However, this work is valid only for traffic signs which have red outer edges. This is a big restriction since there are many traffic signs having different colored edges. Another difference of this work than the work done in this thesis is that localization and extraction are not required in the latter one.

Another study proposed in [27] is based on extraction of traffic signs from the original image by using shape and specific color information. SIFT is used in the recognition step, nevertheless neither color nor orientation information of matching features are checked. Although the system has high recognition accuracy, results of this thesis demonstrate that by introducing color and orientation information the accuracy can even be higher.

As seen, many techniques are proposed and many implementations exist in the literature for the traffic sign detection and recognition issue but none of them added new features to SIFT while using it in this subject in order to increase the performance of the system.

The thesis is organized as follows.

In Section 2, components and properties of the traffic sign detection and recognition systems and the difficulties that are faced in this issue are discussed in detail. The general approaches in this subject are presented. In addition, details of the color model used are described.

SIFT which is the local feature technique used is introduced in Section 3. After a brief introduction to SIFT, steps which are performed until the keypoint descriptors are obtained are discussed. Thereafter, use of SIFT in image matching and recognition is explained.

Section 4 includes information about the development environment and the implementation of the three recognition methods. In other words, recognition of traffic signs using SIFT features are explained in detail. Next, adding color information by using color classification method and including orientation information are discussed. Lastly, information about the training and test images are given.

The results are given and discussed in Section 5.

Finally, conclusions and future work is discussed in Section 6.

## 2. PROPERTIES OF TRAFFIC SIGN DETECTION AND RECOGNITION SYSTEMS AND THE DIFFICULTIES

The identification of traffic signs in two consecutive stages in general: first detection and then recognition.

A general traffic sign detection and recognition system is given in Figure 2.1. The system implementation can be done by using either color information, shape information, or both of them. As expected, using both of them gives better results.

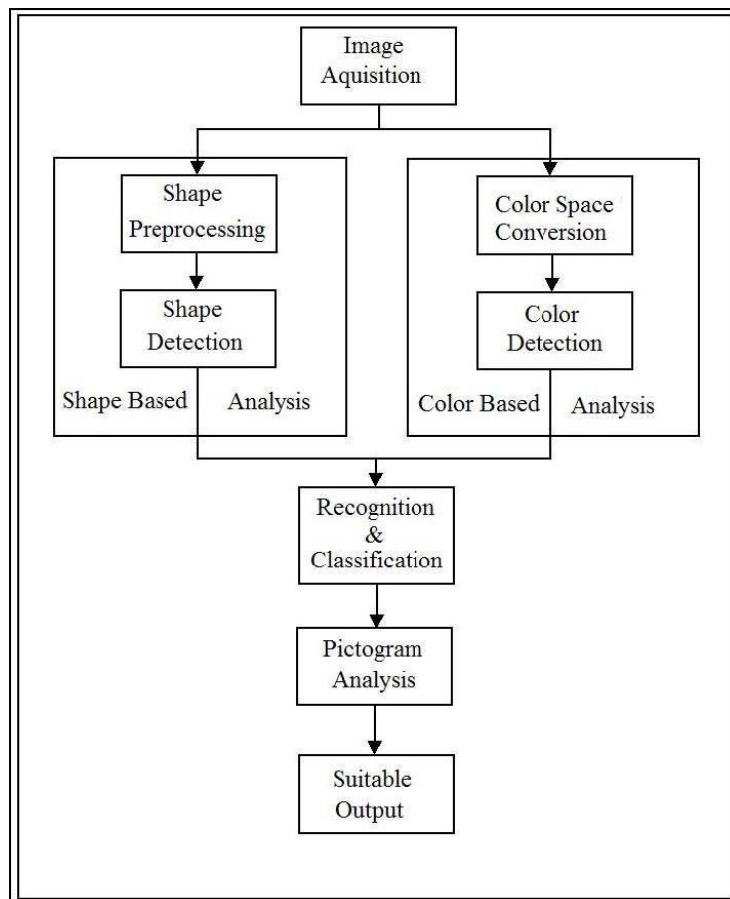


Figure 2.1. General Traffic Sign Detection and Recognition System [1]

### 2.1. Detection

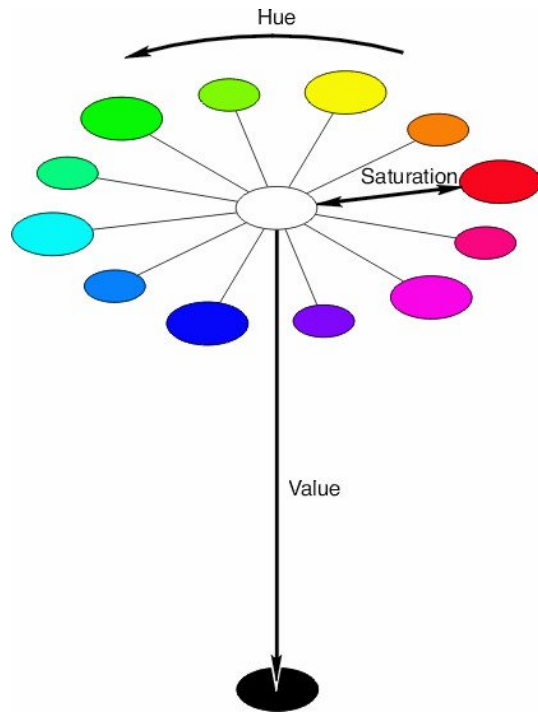
In the studies up to now, in the detection phase the images are segmented according to the sign properties like color or shape [1].

### **2.1.1. Using Color Information in the Detection of Traffic Signs**

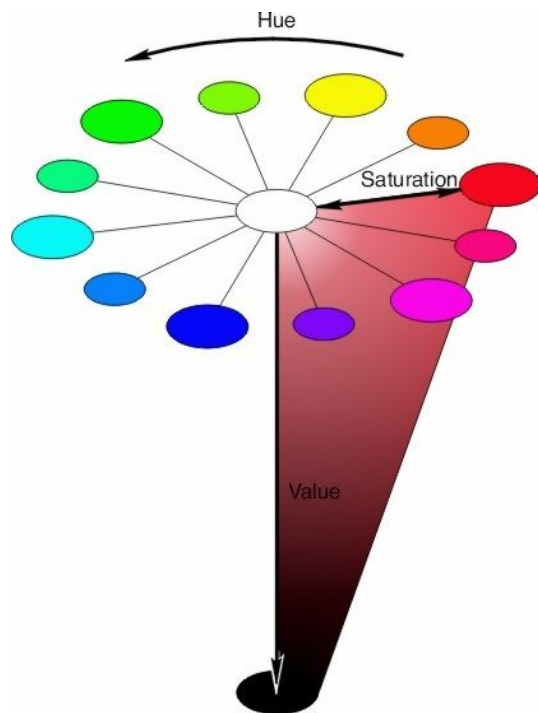
Colors give important information in the detection of traffic signs since colors are differentiating features of traffic signs. When the picture of a traffic sign is taken by a camera, the camera produces an RGB (Red-Green-Blue) image that uses RGB color space which is built as Cartesian coordinate system and the x, y, and z axis are represented by the R, G, and B respectively. But the RGB image is not appropriate for the detection and recognition of traffic signs since the coordinates of the three colors are highly related to the light intensity which affects them by shifting the cluster of colors towards the white or the black sides. So, color space conversion which means converting RGB image into an image that uses another color space is an important part of the detection and recognition systems which use color information. By color space conversion, the brightness information is separated from the color information and then this real color information can be used in the process of detection and recognition of the traffic signs. HSI, HSB,  $L^*a^*b^*$ , YIQ, and YUV are examples of the color spaces available in the literature. The hue-saturation systems which are also used in this study are the most widely used color systems in traffic sign detection and recognition systems. Details of the Hue-Saturation-Value (HSV) Color Model that is used in this thesis in order to get the right color information of the traffic signs are given in Section 2.1.1.

#### **2.1.1.1. The Hue-Saturation-Value (HSV) Color Model**

HSV color space describes how humans naturally describe and perceive colors [28]. As stated in [29], the HSV color model is like a cone which was developed in 1978 by Alvy Ray Smith and can be seen from the Figures 2.2 and 2.3. The central axis ranges from white at the top to black at the bottom and there are neutral colors between them. “Hue” is the angle around the axis, “saturation” is the distance from the axis and “value” is the distance along the axis.



**Figure 2.2.** The Hue, Saturation, Value (HSV) Color Model [28]



**Figure 2.3.** A Saturation/Value Slice of a Specific Hue in the HSV Color Model [28]

Hue is the general color class of a specific color: red, green, blue, etc [28]. Saturation shows the amount of hue dominance in the color. The colors on the central axis are desaturated which constitute the grayscale. Finally, value gives the lightness/darkness of a color.

HSV is defined by transformations between the r, g, b coordinates of the RGB color space and h, s, v coordinates of the HSV color space.

If a color in RGB space has the red, green, and blue coordinates as r, g, b between [0,1] and “max” is the greatest of r, g, b, and “min” is the least, the conversion from RGB color space to HSV color space can be done as follows [29].

$$h = \begin{cases} 0 & \text{if max} = \text{min} \\ 60^\circ \times \frac{g-b}{\text{max}-\text{min}} + 0^\circ, & \text{if max} = r \text{ and } g \geq b \\ 60^\circ \times \frac{g-b}{\text{max}-\text{min}} + 360^\circ, & \text{if max} = r \text{ and } g < b \\ 60^\circ \times \frac{b-r}{\text{max}-\text{min}} + 120^\circ, & \text{if max} = g \\ 60^\circ \times \frac{r-g}{\text{max}-\text{min}} + 240^\circ, & \text{if max} = b \end{cases} \quad (2.1)$$

$$s = \begin{cases} 0, & \text{if max} = 0 \\ \frac{\text{max}-\text{min}}{\text{max}} = 1 - \frac{\text{min}}{\text{max}}, & \text{otherwise} \end{cases} \quad (2.2)$$

$$v = \text{max} \quad (2.3)$$

The hue angle  $h$  is between [0, 360), the saturation  $s$  and the value  $v$  are between [0,1].

The conversion from HSV color model to RGB color model is given as [29]:

$$h_i = \left\lfloor \frac{h}{60} \right\rfloor \text{ mod } 6 \quad (2.4)$$

$$f = \frac{h}{60} - \left\lfloor \frac{h}{60} \right\rfloor \quad (2.5)$$

$$p = v \times (1 - s) \quad (2.6)$$

$$q = v \times (1 - f \times s) \quad (2.7)$$

$$t = v \times (1 - (1 - f) \times s) \quad (2.8)$$

Then (r, g, b) color vector can be computed as:

$$(r, g, b) = \begin{cases} (v, t, p), & \text{if } h_i = 0 \\ (q, v, p), & \text{if } h_i = 1 \\ (p, v, t), & \text{if } h_i = 2 \\ (p, q, v), & \text{if } h_i = 3 \\ (t, p, v), & \text{if } h_i = 4 \\ (v, p, q), & \text{if } h_i = 5 \end{cases} \quad (2.9)$$

### 2.1.2 Using Shape Information in the Detection of Traffic Signs

The systems which use shape information while detecting the traffic signs can also show good performance [1]. Detecting the shape of the traffic signs is a good choice in the situations where extracting the color information is very difficult. But there are some difficulties while using shape information in the detection and recognition of the traffic signs too. As an instance, there can be objects in the image which have similar shapes to the traffic signs such as windows, cars, shop names signs. Another difficulty is the change of the orientation of the traffic signs with time by the effect of humans or cars which hit them. Traffic signs can be damaged or occluded by other objects, so in these situations detecting the shape of the traffic signs and recognizing them correctly are very difficult. Detecting the shape of the sign becomes difficult if the size of the traffic sign is very small in the image which is the result of long distance between the camera and the sign. Working with shapes requires robust edge detection and matching algorithms.

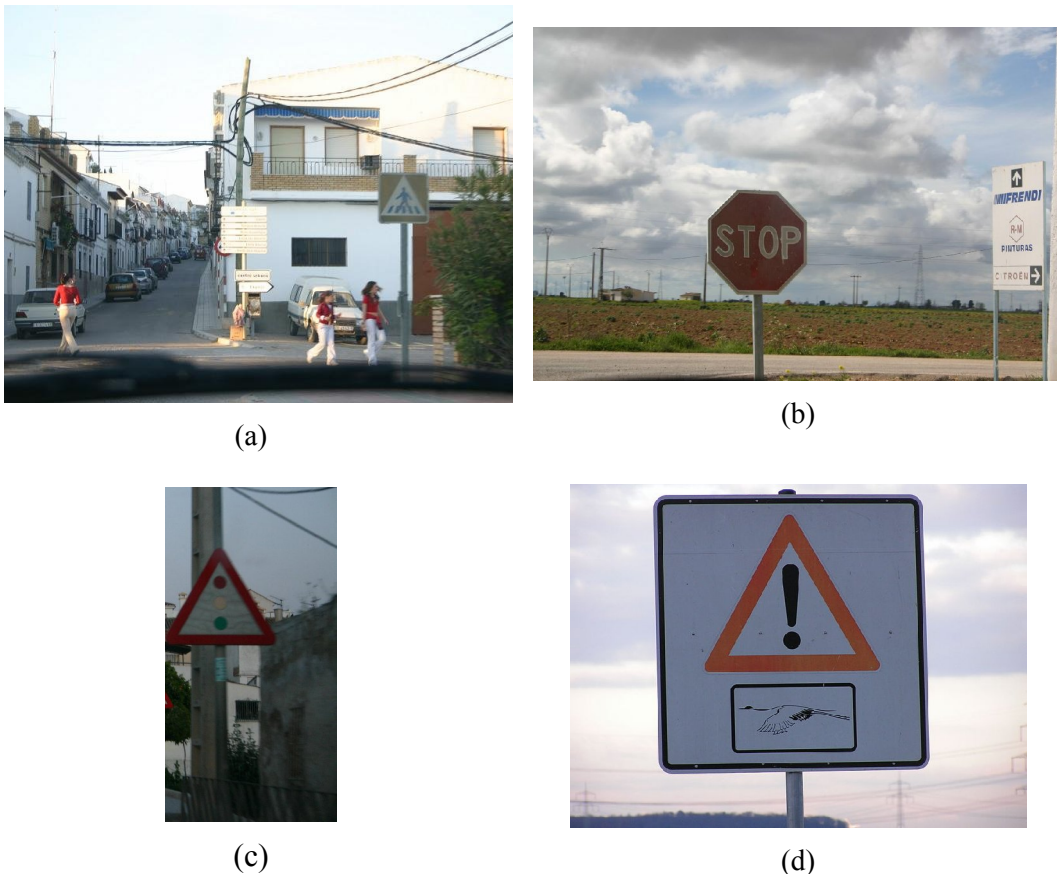
### 2.2. Recognition

The output of the detection step is a segmented image which is the region that the traffic sign may come into existence and in this region the recognition process proceeds. In the recognition step, the general used methods as template matching, neural networks, numerous classifiers find the individual class of the traffic sign by testing against a certain set of features which put emphasis on the differences among the different traffic signs. Also the signs can be put into classes such as triangles, circles, octagons, etc. Then performing pictogram analysis, the text and the shape which exist at the inner part of the sign determine the individual class of the sign under consideration. The method used in the recognition must be robust to the geometrical status, the size and the position of the traffic sign in the image.

In this thesis, shape information of the traffic signs is not utilized. Besides, utilization of color information is different than these general approaches, since there is not any separate detection component in the work done in this thesis. Color information is used while checking the accuracy of the traffic sign which is found by SIFT. This is achieved by comparing the colors of the pixels which are matched by SIFT in the input image and the training image that contains the matched traffic sign.

### 2.3. The Difficulties

There can be some difficulties while detecting and recognizing the traffic signs. For example, the color of the traffic sign can change with time by the sun light and also as a result of the reaction between the pollutants in the air and the paint of the traffic sign [30]. Examples of these type traffic signs from the test set that is used in this thesis are given in Figure 2.4.



**Figure 2.4.** Examples of Variation in the Color of the Traffic Signs with Time (a)-(c) [31]  
(d) [32]

Also the weather conditions like rain, snow, fog affect the appearance of the traffic sign as can be seen from the image that exists in the test set and in Figure 2.5, they make detection and recognition difficult and clouds can cause shadows on the sign.



**Figure 2.5.** Effect of Weather Conditions on the Appearance of the Traffic Sign [33]

Shadows can also be caused by other objects. Some example images which exist in the test set are given in Figure 2.6.



(a)



(b)



(c)

**Figure 2.6.** Examples of Shadowed Traffic Signs [31]

Besides, the direction and the strength of the light which change according to the time of the day and season vary the appearance of the traffic sign by changing the color information as can be seen in the images which are in the test set and in Figure 2.7.



(a)



(b)



(c)



(d)



(e)



(f)

**Figure 2.7.** Effect of the Direction and the Strength of Light on the Color Perceived  
(a) [31] (b) [34] (c) [31] (d) [35] (e) [36] (f) [37]

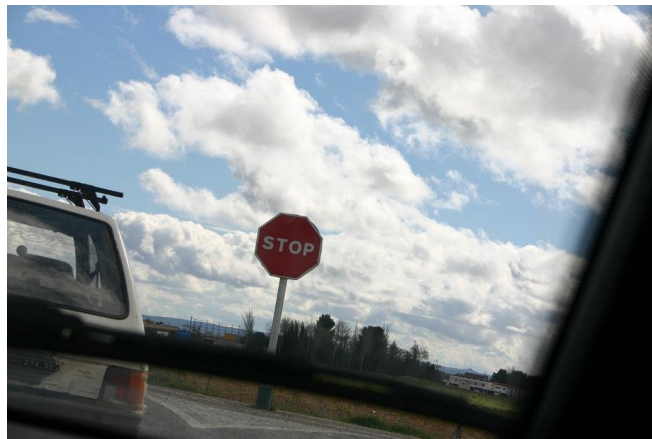
Illumination geometry and viewing geometry also affect the color information. Different viewing orientation examples from the test set are given in Figure 2.8.



(a)



(b)

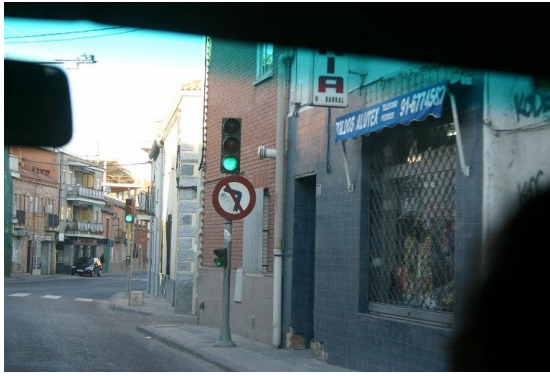


(c)

**Figure 2.8.** Viewing Orientation Examples [31]

If the color information is used to segment the traffic sign, the objects which have same color with the traffic signs like buildings, cars, pedestrians in the scene make the detection process difficult. Besides, if the shape information is used for segmentation, there can be objects which have the same shape with the traffic signs.

Detecting and recognizing disoriented, damaged or occluded traffic signs are not easy too. Two disoriented traffic sign examples used in the test set and examples of occluded traffic sign images from the test set are given in Figure 2.9.



(a)



(b)



(c)



(d)



(e)

**Figure 2.9.** Difficulties in Traffic Sign Recognition. (a), (b) Examples of Disoriented Traffic Signs [31, 38]. (c), (d), (e) Examples of Occluded Traffic Signs [31].

The size of the traffic sign change with the distance between the camera and the traffic sign, so the detection and the recognition are not performed for fix sized

traffic signs, all different sizes must be considered. Figure 2.10 show images which include different sized traffic signs which exist in the test set.



(a)



(b)



(c)



(d)

**Figure 2.10.** Change in Size of a Traffic Sign Depending on the Camera Distance  
(a), (b) [39] (c), (d) [31]

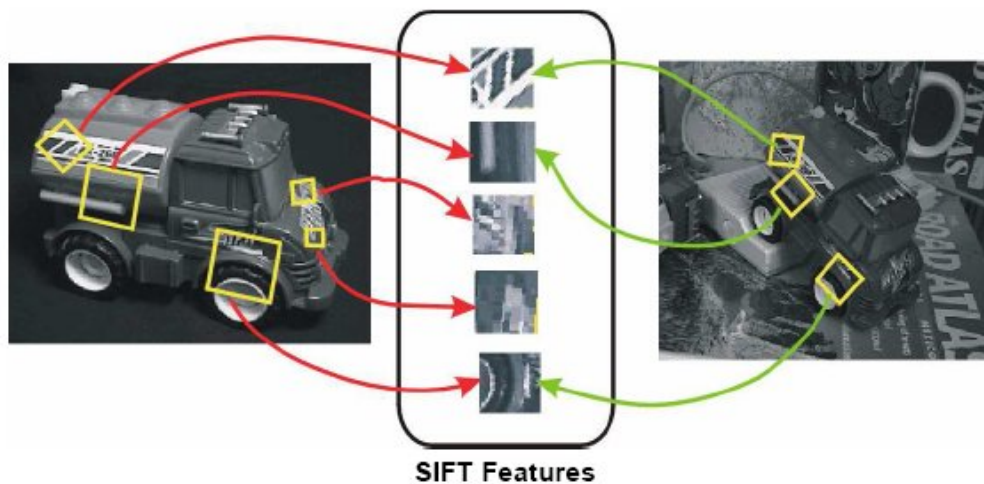
If the image is taken from a camera which exists in a moving car, then problems like motion blur and car vibration can arise [10] as can be seen from Figure 2.11 which is an image that exists in the test set. Also there is deficiency of the standard database for evaluation of the classification method which is used [24, 40].



**Figure 2.11.** Blurring Caused by Motion [31]

### 3. SCALE INVARIANT FEATURE TRANSFORM (SIFT)

SIFT [3] is a method for extracting distinctive invariant features from images and reliable matching between different views of an object or scene can be performed by it. This approach has been named the Scale Invariant Feature Transform (SIFT), since it transforms image data into scale-invariant coordinates relative to local features. The features are invariant to translation, rotation, scaling, clutter, lighting and occlusion. Also robust matching across a substantial range of affine distortion, change in 3D viewpoint, addition of noise, and change in illumination is provided. The features are highly distinctive so that a single feature can be correctly matched with high probability against a large database of features from an image database which includes many images. The recognition using SIFT features is performed by matching individual features to a database of features from known objects using a fast nearest-neighbor algorithm. Matching of two images which include different views of the same object using SIFT features is showed in Figure 3.1.



**Figure 3.1.** Matching of Two Images which Include the Same Object [41]

Figure 3.2 shows an example of object recognition for an image containing 3D objects. The training images for two objects are shown on the left. These can be recognized in an image which is shown in the middle and contains instances of these objects hidden behind others and with extensive background clutter. The results of

recognition are shown on the right. Parallelograms show each recognized object and smaller squares indicate the keypoints that were used for recognition.



**Figure 3.2.** Example of Object Recognition for an Image Containing 3D Objects [3]

The cost of extracting SIFT features is minimized by taking a cascade filtering approach, so only at locations that pass an initial test the more expensive operations applied.

Following are the major steps of SIFT to generate the set of image features:

### 3.1. Scale-Space Extrema Detection

In this first step, all scales and image locations which can be assigned under differing views of the same object are searched. Detecting locations which are invariant to scale change of the image can be done by searching for stable features across all possible scales, using a continuous function of scale known as scale space [42]. The studies of Koenderink [43] and Lindeberg [44] showed that the only possible scale-space kernel is the Gaussian function. So, the scale space of an image is defined as a function  $L(x,y,\sigma)$  which is produced from the convolution of variable-scale Gaussian,  $G(x,y,\sigma)$ , with an input image,  $I(x,y)$  [3].

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y), \quad (3.1)$$

where  $*$  is the convolution operation in  $x$  and  $y$ , and  $\sigma$  is the scale, and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \quad (3.2)$$

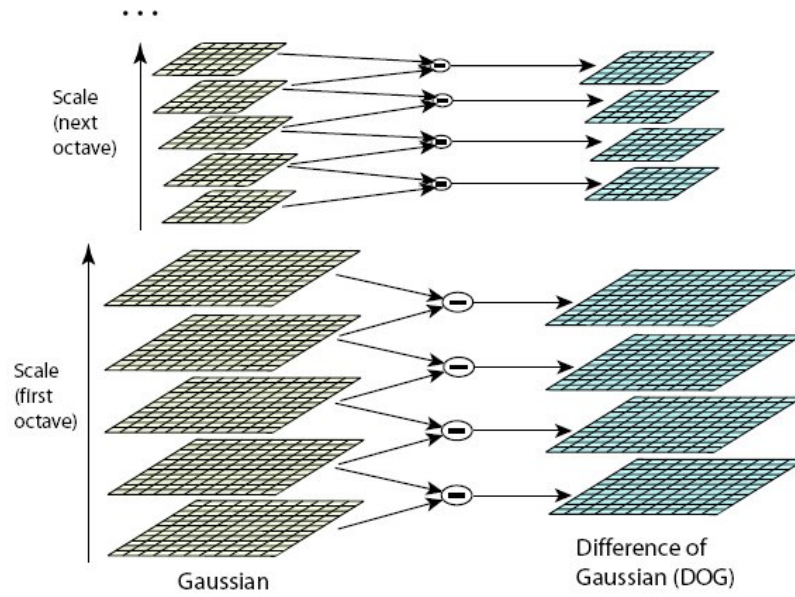
In order to detect stable keypoint locations in scale space, it is proposed in [45] using scale-space extrema in the difference-of-Gaussian function convolved with the image,  $D(x,y,\sigma)$ , which is computed from the difference of two nearby scales separated by a constant multiplicative factor  $k$ :

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned} \tag{3.3}$$

This function is an efficient function to compute since it can be computed by simple image subtraction. Besides, the difference-of-Gaussian function is a close approximation to the scale-normalized Laplacian of Gaussian,  $\sigma^2 \nabla^2 G$ , as studied by Lindeberg [44].

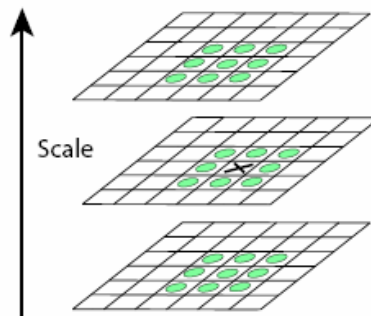
And it is showed that for true scale invariance, the normalization of the Laplacian with the factor  $\sigma^2$  is required. As found by Mikolajczyk [46], the maxima and minima of  $\sigma^2 \nabla^2 G$  produce the most stable image features compared to a range of other possible image functions, such as the gradient, Hessian, or Harris corner function.

In Figure 3.3, construction of  $D(x,y,\sigma)$  is shown. The initial image is repeatedly convolved with Gaussians to produce the set of scale space images which are shown in the left column for each octave of scale space. Then adjacent Gaussian images are subtracted to produce the difference-of Gaussian images on the right. After each octave, the Gaussian image is down-sampled by a factor of 2 by taking every second pixel in each row and column, and the process repeated.



**Figure 3.3.** Set of Scale Space Images at the Left, Difference-of Gaussian Images on the Right [3]

In order to detect the local maxima and minima of  $D(x,y,\sigma)$ , each sample point is compared to its 26 neighbors, eight neighbors in the current image and nine neighbors in the scale above and below which are shown in Figure 3.4. It is selected only if it is larger than all of these neighbors or smaller than all of them.



**Figure 3.4.** Detection of Maxima and Minima of Difference-of-Gaussian Images [3]

To reliably detect the extrema, the frequency of sampling in the image and scale domains must be determined. It is shown that [3] the highest repeatability is obtained when sampling 3 scales per octave, and this is the number of scale samples used. Matching scale is defined as  $\sqrt{2}$  of the correct scale and a matching location as being within  $\sigma$  pixels, where  $\sigma$  is the scale of the keypoint which is defined as the standard deviation of the smallest Gaussian used in the difference-of-Gaussian function. As the frequency of sampling per octave of scale space is determined, so the frequency

of sampling in the image domain relative to the scale of smoothing must be determined too. Since there is a cost to using a large  $\sigma$  in terms of efficiency,  $\sigma$  is chosen as  $\sigma = 1.6$ , which is stated that provides close to optimal repeatability.

If the image is pre-smoothed before extrema detection, the highest spatial frequencies are effectively discarded. Therefore, the image can be expanded to create more sample points than were present in the original to make full use of the input. The size of the input image is doubled using linear interpolation prior to building the first level of the pyramid by the image doubling. It is assumed that the original image has a blur of at least  $\sigma = 0.5$  which is the minimum needed to prevent significant aliasing, and so the doubled image has  $\sigma = 1.0$  relative to its new pixel spacing which means that little additional smoothing is needed before the creation of the first octave of scale space.

### 3.2. Keypoint Localization

When a keypoint candidate is found by comparing a pixel to its neighbors, a detailed fit to the nearby data for location, scale, and ratio of principal curvatures is performed in the next step. By using this information, the points which have low contrast, and therefore sensitive to noise, or are poorly localized along an edge are found and then these point are rejected.

A method is developed by Brown [47] for fitting a 3D quadratic function to the local sample points to determine the interpolated location of the maximum. His experiments showed that this method provides a substantial improvement to matching and stability. In this approach a shifted Taylor expansion (up to the quadratic terms) of the scale-space function,  $D(x,y,\sigma)$ , is used so that the origin is at the sample point:

$$D(\mathbf{x}) = D + \frac{\partial D}{\partial \mathbf{x}} \mathbf{x} + \frac{1}{2} \mathbf{x}^T \frac{\partial^2 D}{\partial \mathbf{x}^2} \mathbf{x} \quad (3.4)$$

Here  $D$  and its derivatives are evaluated at the sample point and  $\mathbf{x} = (x,y,\sigma)^T$  is the offset from this point. To determine the location of the extremum,  $\hat{\mathbf{x}}$ , the derivative of this function with respect to  $\mathbf{x}$  is taken and set to zero, which gives

$$\hat{\mathbf{x}} = -\frac{\partial^2 D^{-1}}{\partial \mathbf{x}^2} \frac{\partial D}{\partial \mathbf{x}} \quad (3.5)$$

The function value at the extremum,  $D(\hat{\mathbf{x}})$ , is useful for rejecting unstable extrema with low contrast. This value is obtained by substituting equation (3) into (2), giving

$$D(\hat{\mathbf{x}}) = D + \frac{1}{2} \frac{\partial D^T}{\partial \mathbf{x}} \hat{\mathbf{x}}. \quad (3.6)$$

If it is assumed that image pixel values are in the range [0,1], then all extrema with a value of  $|D(\hat{\mathbf{x}})|$  less than 0.03 were discarded in order to reject unstable extrema with low contrast.

For stability, it is not enough to reject keypoints with low contrast [3]. The difference-of-Gaussian function has a strong response along edges, even if the location along the edge is poorly determined and therefore unstable to small amounts of noise. So, these edge responses have to be eliminated. A poorly defined peak in the difference-of-Gaussian function will have a large principal curvature across the edge but a small one in the perpendicular direction. The principal curvatures can be computed from a 2x2 Hessian matrix,  $\mathbf{H}$ , which is computed at the location and scale of the keypoint where the derivatives are estimated by taking differences of neighboring sample points:

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (3.7)$$

The eigenvalues of  $\mathbf{H}$  are proportional to the principal curvatures of  $D$ . Since only the ratio of eigenvalues is considered, it is not needed to compute eigenvalues [48]. If the eigenvalue with the largest magnitude is named as  $\alpha$  and the smaller one as  $\beta$ , the sum of the eigenvalues can be computed from the trace of  $\mathbf{H}$  and their product can be computed from the determinant [3].

$$\begin{aligned} \text{Tr}(\mathbf{H}) &= D_{xx} + D_{yy} = \alpha + \beta, \\ \text{Det}(\mathbf{H}) &= D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta. \end{aligned} \quad (3.8)$$

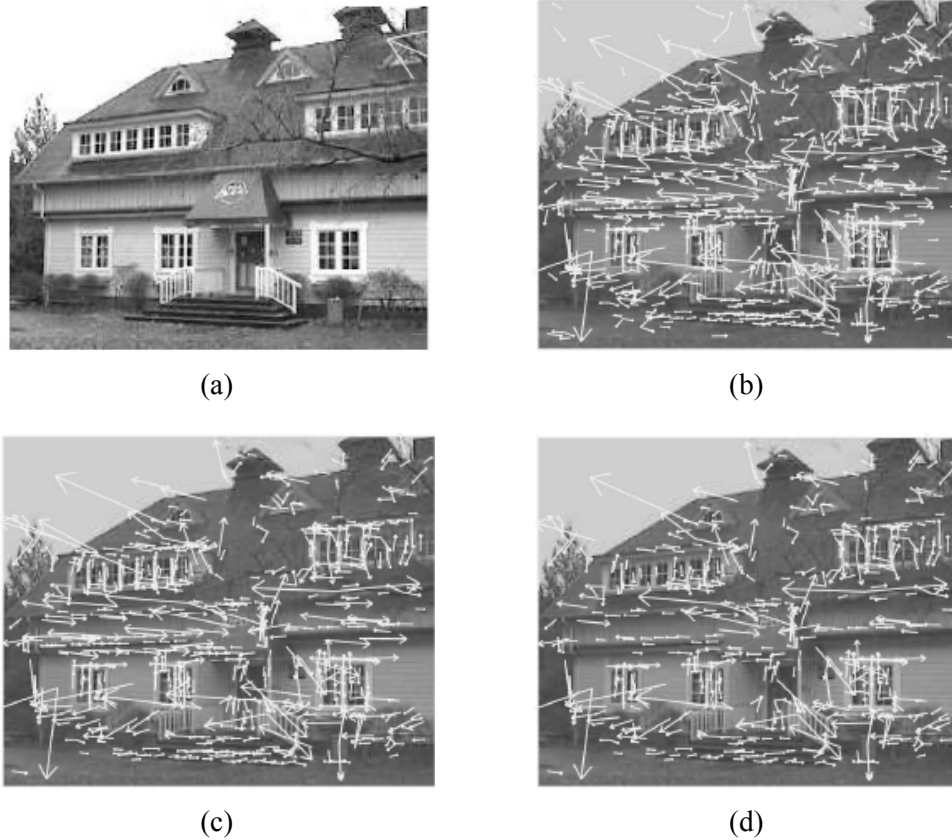
If  $r$  is the ratio between the largest magnitude eigenvalue and the smaller one of  $\mathbf{H}$ , so that  $\alpha = r \beta$ , then

$$\frac{\text{Tr}(\mathbf{H})^2}{\text{Det}(\mathbf{H})} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r + 1)^2}{r}, \quad (3.9)$$

which depends only on the ratio of the eigenvalues rather than their individual values.  $(r+1)^2/r$  is minimum when the two eigenvalues are equal and it increases with  $r$ . Then in order to eliminate these keypoints, to check that the ratio of principal curvatures is below some threshold,  $r$ , it is only needed to check

$$\frac{\text{Tr}(\mathbf{H})^2}{\text{Det}(\mathbf{H})} < \frac{(r + 1)^2}{r}. \quad (3.10)$$

$r$  is selected as  $r = 10$ , which eliminates keypoints that have a ratio between the principal curvatures greater than 10.



**Figure 3.5.** Keypoint Selection. (a) Original Image. (b) The Initial Locations of 832 Keypoints at Maxima and Minima of the Difference-of-Gaussian Function. (c) Remaining 729 Keypoints after Applying a Threshold on Minimum Contrast. (d) The Final 536 Keypoints that Remain after Applying Threshold on Ratio of Principal Curvatures [3]

Figure 3.5 shows the effects of keypoint selection on an example 233 by 189 pixel image. The keypoints are shown as vectors giving the location, scale, and orientation

of each keypoint. Figure 3.5 (a) shows the original image. Figure 3.5 (b) shows the 832 keypoints at all detected maxima and minima of the difference-of-Gaussian function, (c) shows the 729 keypoint that remain after the removal of the ones with a value of  $|D(\hat{x})|$  less than 0.03. (d) shows the final 536 keypoints which remain after the additional threshold ( $r = 10$ ) on ratio of principal curvatures.

### 3.3. Orientation Assignment

The keypoint descriptor can be represented relative to this orientation by assigning a consistent orientation to each keypoint based on local image gradient directions, so invariance to image rotation can be achieved.

The scale of the keypoint is used to select the Gaussian smoothed image,  $L$ , with the closest scale, so that all computations are performed in a scale-invariant manner. At this scale, for each image sample,  $L(x,y)$ , the gradient magnitude,  $m(x,y)$ , and orientation,  $\theta(x,y)$ , is computed using pixel differences:

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2} \quad (3.11)$$

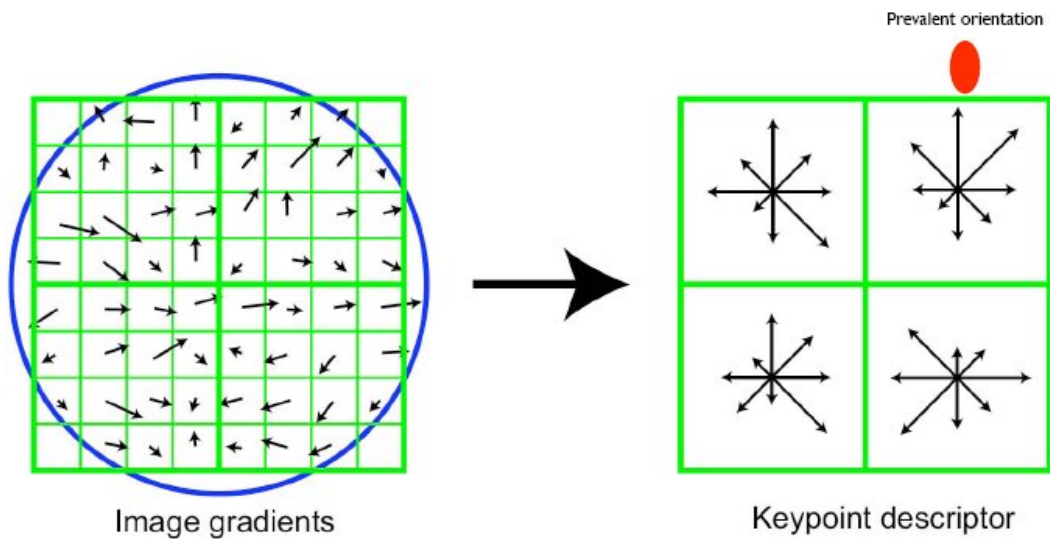
$$\theta(x, y) = \tan^{-1}((L(x, y + 1) - L(x, y - 1)) / (L(x + 1, y) - L(x - 1, y)))$$

Then, an orientation histogram is formed from the gradient orientations of sample points within a region around the keypoint. The orientation histogram has 36 bins which cover the 360 degree range of orientations and each sample added to this histogram is weighted by its gradient magnitude and by a Gaussian-weighted circular window with  $\sigma$  that is 1.5 times that of the scale of the keypoint.

Peaks in the orientation histogram give the dominant directions of local gradients. The highest peak in the histogram is detected, and then any other local peak that is within 80% of the highest peak is also used to create a keypoint with that orientation. For that reason, for locations with multiple peaks of similar magnitude, there will be multiple keypoints created at the same location and scale but their orientations will be different. In general, multiple orientations are assigned to only about 15% of points, but these provide the stability of matching. Finally, a parabola is fit to the 3 histogram values closest to each peak to interpolate the peak position for better accuracy.

### 3.4. Keypoint Descriptor

In the previous steps, an image location, scale, and orientation are assigned to each keypoint. These parameters form a local 2D coordinate system in which to describe the local image region, and therefore provide invariance to these parameters [3]. In the next step, a descriptor must be computed for the local image region which is highly distinctive and as invariant as possible to remaining variations, such as change in illumination or 3D viewpoint.



**Figure 3.6.** Creation of a Keypoint Descriptor [3]

In Figure 3.6, the computation of the keypoint descriptor is shown. A keypoint descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the keypoint location which are weighted by a Gaussian window where  $\sigma$  is equal to one half the width of the descriptor, indicated by the overlaid circle, as shown on the left. The purpose of this Gaussian window is to avoid sudden changes in the descriptor with small changes in the position of the window, and to give less emphasis to gradients that are far from the center of the descriptor, as these are most affected by misregistration errors. Then orientation histograms which summarize the contents over 4x4 sub-regions are formed from these samples, as shown on the right. Here, the length of each arrow corresponds to the sum of the gradient magnitudes near that direction within the region. In this figure a 2x2 descriptor array computed from an 8x8 set of samples is shown as an example, but 4x4 descriptors computed from a 16x16 sample array are used in SIFT. Therefore,  $4 \times 4 \times 8 = 128$  element feature vector exist for each keypoint.

The dimensionality of the descriptor may seem high, but it is found that it consistently performs better than lower-dimensional descriptors on a range of matching tasks and that the computational cost of matching remains low.

To reduce the effects of illumination change, the feature vector is firstly normalized to unit length. By vector normalization, the contrast change in which each pixel value is multiplied by a constant will multiply gradients by the same constant is cancelled.

The gradient values are not affected by a brightness change in which a constant is added to each image pixel, since they are computed from pixel differences. As a result, the descriptor is invariant to affine changes in illumination. However, camera saturation or illumination changes that affect 3D surfaces with differing orientations by different amounts can cause non-linear illumination changes. A large change in relative magnitudes for some gradients occurs by these effects, but the gradient orientations are less likely to be affected. Therefore, in order to reduce influence of large gradient magnitudes is succeeded by thresholding the values in the unit feature vector to each be no larger than 0.2, and then renormalizing to unit length. This means that matching the magnitudes for large gradients is no longer as important, and that the distribution of orientations has greater emphasis.

For image matching and recognition, SIFT features are first extracted from a set of reference images which can be named as training set and stored in a database. Then a new image, which is the test image, is matched by individually comparing each feature from the new image to this database and finding candidate matching features based on Euclidean distance of their feature vectors. The image in the training set which gives the maximum number of matches with the test image is the image that is searched for and it is in the same class with the test image.

## **4. IMPLEMENTATION**

### **4.1. Development Environment**

This thesis was developed using the MATLAB 7.0 software development environment. The program was run on a PC with Intel Pentium M 1.80 GHz CPU (Central Processing Unit). The system had 512 MB of RAM (Random Access Memory).

### **4.2. Recognition Methods**

For the recognition of traffic signs, at the beginning SIFT is used alone in the first method. The color information is included and utilized in the second method. In the third method, newly added feature is the orientation examination of matched features.

#### **4.2.1. Recognition by Using only SIFT Features**

To investigate the performance of SIFT in the subject of traffic sign recognition, in the first method, SIFT keypoints are found and stored for each image in the training set. Then the keypoints of the test image are found and matched with the keypoints of all training images. The training image which gives the maximum match with the training image is the result traffic sign.

In order to extract SIFT keypoints the “sift” function which is written in MATLAB and included in the demo software that is provided by David Lowe [49] is used. In the “sift” function, there is a call to an executable file named “siftWin32” which is a program that detects invariant keypoints. For an image “imageFile”, the use of “sift” function is as follows.

```
[image, descriptors, locs] = sift(imageFile);
```

Here, “descriptors” is K-by-128 matrix where K is the number of keypoints. Each row gives an invariant descriptor for a keypoint, so descriptor is a vector of 128 values. And “locs” is a K-by-4 matrix where each row has the 4 values for a keypoint

location as (row, column, scale, orientation). Orientation is in the range  $[-\pi, \pi]$  radians. When it is called, “siftWin32” writes invariant keypoints to “tmp.key”, and also fills “descriptors” and “locs” from “tmp.key”.

When the program is run, the descriptors of the input test image are found. Next, the SIFT keypoint descriptors of the training images are loaded from files which are previously stored in the training stage in order to save time by not extracting them at each run. And for all training images, for each descriptor in the test image, its match is selected in the training image if it exists and the number of matches is stored. The match of a descriptor of the test image in a training image is found by computing vector of dot products of the descriptor that is under consideration and all descriptors of the training image. Then, in order to calculate Euclidean distances between two descriptors, inverse cosine of dot products is taken and the results are sorted. Then the first match in this order is checked. If the distance of this first match is less than “distRatio” times the distance to the second match where “distRatio” is a constant, the first match is accepted as a true match. In this thesis, “distRatio” is taken as 0.6. This process is performed for all the descriptors of the test image.

Finally, the training image that has maximum number of matches is found and a new image is created showing the two images side by side, also lines joining the accepted matches are shown.

#### **4.2.2. Recognition by Using SIFT Features and Color Information**

The color information is included in the second method. While matching the features of a test image with the features of the training images, colors of the pixels of the features are checked, if they are same then the match is assumed to be true, else the match is considered to be false. So, some of the false matches are eliminated by checking the color. As a result, the performance is increased.

Hue-Saturation-Value (HSV) color model is used as stated in Section 2.1.1 in order to get the true color information of the traffic signs. In Matlab, the conversion from RGB to HSV can be done easily with “rgb2hsv” function [50]. A RGB image can be converted to the equivalent HSV image as follows.

```
hsv_image = rgb2hsv(rgb_image)
```

“rgb\_image” is an m-by-n-by-3 image array whose three planes contain the red, green, and blue components for the image in RGB color space and “hsv\_image” is m-by-n-by-3 image array whose three planes contain the hue, saturation and value components for the image in HSV color space. The elements of both color maps are in the range 0 to 1. Then H, S, V coordinates then can be obtained as follows.

```
H = hsv_image(:,:,1);
```

```
S = hsv_image(:,:,2);
```

```
V = hsv_image(:,:,3);
```

The corresponding color becomes red, yellow, green, cyan, blue, magenta, and again red as hue varies from 0 to 1. So there are red colors both at hue equals to 0 and hue equals to 1. As saturation varies from 0 to 1, the corresponding colors vary from unsaturated level (grayscale) to fully saturated level where colors contain no white component. As value increases from 0 to 1, so does the brightness.

#### **4.2.2.1. Color Classification Method**

The traffic signs which are used in this study include the colors, red, blue, green, white, yellow and black. In general, the differentiation of the pixels which have red, green, yellow and blue colors can be achieved by looking only their hue values. But it is not easy to determine black or white colors, since they can have any hue value. So, in order to differentiate these 6 colors, all of the components of the HSV color model, hue, saturation and value, are taken into consideration. By examining the hue, saturation and value information of many example pixels with known true colors which are taken from the images of real world traffic signs, some rules are defined in order to determine the color of a pixel is yellow, red, blue, green, black or white. But these color classes can intersect at some areas and therefore the color of a pixel is one of the colors which intersect in these areas. The color classes are given in the Table 4.1. For example, if H, S, V values of a pixel falls into into black/white class, then the color of this pixel can be either black or white. Also, a color can have more than one rule, as an instance, red color has two rules; if H, S, V values of a pixel obey one of these rules, then the color of the pixel is red.

**Table 4.1.** Color Classes and Corresponding Rules

<b>Color</b>	<b>Rules</b>		
Yellow	$0.05 < H < 0.2$ $0.52 \leq S$ $0.27 \leq V$		
Red	$H \leq 0.036$ $0.45 \leq S$ $0.23 \leq V$	$0.93 \leq H$ $0.16 \leq S$ $0.23 \leq V$	
Black	$H = 0$ $S = 0$ $V = 0$	$0.54 \leq H \leq 0.66$ $0.135 \leq S \leq 0.48$ $0.23 \leq V \leq 0.47$	
	$0.54 \leq H \leq 0.61$ $0.507 \leq S \leq 0.725$ $0.155 \leq V \leq 0.245$	$0.5 \leq H \leq 0.7$ $V \leq 0.01$	
White	$S \leq 0.61$ $0.25 \leq V$		
Blue	$0.5 \leq H \leq 0.68$ $0.23 \leq S$ $0.22 \leq V$		
Green	$0.33 \leq H \leq 0.5688$ $0.598 \leq S$ $0.21 \leq V \leq 0.58$		
Black/White	$0.35 \leq H \leq 0.715$ $0.13 \leq S \leq 0.165$ $0.24 \leq V \leq 0.275$	$0.54 \leq H \leq 0.66$ $0.13 \leq S \leq 0.48$ $0.24 \leq V \leq 0.47$	$H \leq 0.25$ $S \leq 0.38$ $V \leq 0.5$
Green/Black	$0.54 \leq H \leq 0.5688$ $0.598 \leq S \leq 0.725$ $0.21 \leq V \leq 0.245$		
Blue/Green	$0.5 \leq H \leq 0.5688$ $0.598 \leq S$ $0.22 \leq V \leq 0.58$		
White/Red	$H \leq 0.036$ $0.45 \leq S$ $0.3 \leq V$	$0.93 \leq H$ $0.16 \leq S$ $0.25 \leq V$	
Blue/Black/White	$0.54 \leq H \leq 0.66$ $0.23 \leq S \leq 0.48$ $0.3 \leq V \leq 0.47$		
White/Green	$0.33 \leq H \leq 0.5688$ $0.598 \leq S \leq 0.61$ $0.25 \leq V \leq 0.58$		
Blue/White	$0.5 \leq H \leq 0.68$ $0.23 \leq S \leq 0.61$ $0.25 \leq V$		
Yellow/Black	$0.05 < H < 0.2$ $0.85 \leq S \leq 0.96$ $0.27 \leq V \leq 0.37$		
Yellow/White	$0.05 < H < 0.2$ $0.52 \leq S \leq 0.61$ $0.27 \leq V$		

The pixels which have H, S, V values that do not fall into to a class according to these rules are checked again. If  $V < 0.4$  then the pixel is marked as black, otherwise no color is assigned to the pixel.

When a SIFT match is found, the match is accepted as a correct match if the color classes of the pixels of the two matched features also match. The complied color classes are given in Table A.1 in Appendix A.1. In this table, in the first column the color class names are given in order. In the second column, the color classes which comply with the color class that is under consideration are given.

Sample image patches whose true colors are known and which are taken from the traffic sign images that exist in the training and test sets and their H, S, V values are given in Table A.2 in Appendix A.2. Ten different image patches are given for each color: blue, yellow, red, green, black and white.

From these patches, it can be seen that the observed colors of the traffic signs can change very much because of the reasons which are described in Section 2.

The confusion matrix which is constructed for these patches and which shows the color classification of these patches according to the rules that are given in Table 4.1 is given in Table 4.2. It can be seen that all patches for all colors are classified accurately.

**Table 4.2.** Confusion Matrix Created for the Classification of Sample Patches

	<b>Red</b>	<b>Blue</b>	<b>Yellow</b>	<b>Green</b>	<b>White</b>	<b>Black</b>
<b>Red</b>	10					
<b>Blue</b>		10				
<b>Yellow</b>			10			
<b>Green</b>				10		
<b>White</b>					10	
<b>Black</b>						10

#### 4.2.2.2. Thresholding based on Euclidean distance of [H,S,V] vectors

Another probable solution that may be thought is computing Euclidean distance between [H, S, V] vectors of the two corresponding pixels when a SIFT match is found. But, it can be seen from Table A.2 that H values which belong to a color class can change very much, same can be seen for S and V values too. For example, red

colors can have large H values as 0.9993, and small H values as 0.0101-0.036. Again, blue colors can have small V values as 0.22 and large values which are bigger than 0.8. S values of black colors can be very small as 0.05 or large as 0.6667-1. So, the Euclidean distance between [H, S, V] vectors would be very big for these examples even if they have the same color.

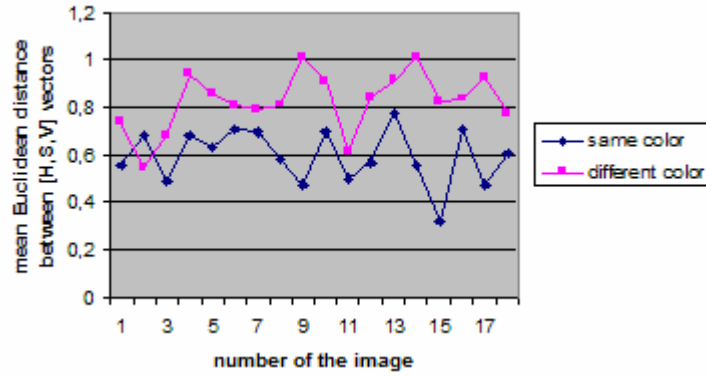
As an example, for the test image from the test set which is shown in Figure 4.1, there are 88 SIFT matches with all the training images in which the two corresponding pixels have the same color, and 37 SIFT matches with all the training images in which the two corresponding pixels have different colors. For these matches, the Euclidean distances between [H, S, V] vectors of the two corresponding pixels are computed and kept, and finally mean Euclidean distances are calculated. For the matches with the same color, the mean Euclidean distance is 0.6854 and for the matches with different colors, the mean Euclidean distance is 0.5416. The mean for the same color matches should be smaller than the mean for the different color matches, but on the contrary it is bigger.



**Figure 4.1.** Example Test Image from the Test Set used Evaluations of the Thresholding Based on Euclidean Distance of [H,S,V] Vectors Method [39]

#### **4.2.2.3. Comparison**

For a detailed investigation, 18 example images from the test set are taken into consideration. For these images, the SIFT matches with all the training images in which the two corresponding pixels have the same color and the matches with all the training images in which the two corresponding pixels have different colors are searched and the Euclidean distances between [H, S, V] vectors of the two corresponding pixels are computed and kept. Then, mean Euclidean distances are calculated and these values are given in Figure 4.2.



**Figure 4.2.** Mean Euclidean Distances Calculated for the SIFT Matches with Same and Different Colors

From this graph, it is seen that the information which is given by the Euclidean distance of  $[H, S, V]$  vectors is not so bad at all times, and in general the mean Euclidean distance for the same colors are smaller than the mean Euclidean distance for the different colors. Therefore, in order to get matching results for these 18 images, according to this graph, 0.7 is taken as the threshold that will be used for these 18 images and which will determine if the two corresponding pixels have the same color or not, since 0.7 is the appropriate value which classifies only 3 mean of same colors as different colors and 3 mean of different colors as same colors. Then, for these 18 test images, the program is run and results shown in the first two columns of Table 4.3 are obtained. In the last two columns of Table 4.3, the results of the second method of the program for the same images are given. In the results, the images are given in gray scale which can be confusing, but color control is included in the program and performed on the colored images.

**Table 4.3.** Matching Results of the Two Methods for 18 Test Images

Results of thresholding based on Euclidean distance of [H,S,V] vectors		Results of the second method	

As seen from Table 4.3, while 6 wrong matches were encountered if the method of thresholding based on Euclidean distances between [H, S, V] vectors is used, only 2 of 18 images are matched wrongly when the color classification method that is proposed is used. Therefore, by having 12 successful matches, the thresholding based on Euclidean distance method shows 66.67% performance, where the color classification method has 16 successful matches out of 18, and has 88.89% performance. The superiority of the color classification method is obvious.

#### **4.2.3. Recognition by Using SIFT features with Color and Orientation**











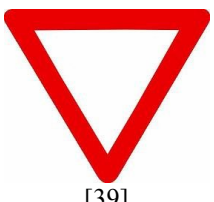




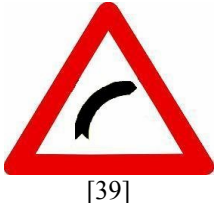




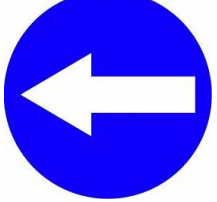



##### **Information**





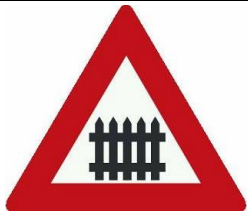


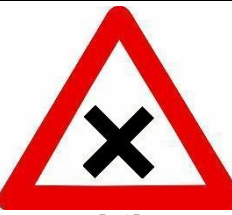



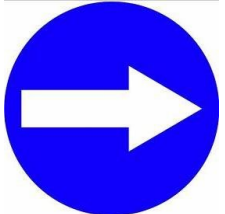









To increase the performance, both the color and the orientation information of the features are included in the third method. As in the second method, the colors of the features' pixels are checked when a SIFT match is found. If they are same, then the orientations of the matched features are checked. Orientations can have values between [-3.14, 3.14]. If two orientations have a difference bigger than 2.5, the match is assumed to be false, else it is accepted as a true match. The value of 2.5 is obtained by examining orientations of many keypoints that match and it is found that this value separates correct matches from the incorrect ones. Therefore, the performance is increased by excluding the wrongly detected matches. Orientation control provides also the correct choice of the traffic sign class if two or more training images give the same number of matches which is the maximum.

#### **4.3. Training Images**

The training set that is used and composed of 36 different traffic signs is given in Table 4.4.

Table 4.4. Training Set

1  [51]	2  [51]	3  [51]	4  [39]
5  [39]	6  [39]	7  [52]	8  [39]
9  [53]	10  [39]	11  [39]	12  [53]
13  [53]	14  [39]	15  [54]	16  [39]
17  [39]	18  [55]	19  [55]	20  [39]
21  [39]	22  [39]	23  [51]	24  [54]

25  [51]	26  [55]	27  [56]	28  [56]
29  [57]	30  [55]	31  [57]	32  [51]
33  [58]	34  [59]	35  [60]	36  [39]
37  [61]	38  [61]	39  [61]	40  [61]
41  [61]	42  [51]	43  [61]	44  [61]
45  [61]			

In general, one training image is used for each traffic sign. For some traffic signs, different versions of the considered sign in the test images can be recognized using

only one training image. As an instance, the test images shown in Figure 4.3 can be recognized using only one training image which is given in Table 4.4, numbered as 7, although this image is different than others.



**Figure 4.3.** Different Versions of the Same Traffic Sign [31, 62].

However, for the different versions of some traffic signs, more than one training image is required in order to recognize them correctly. Therefore, two training images are used for nine of the signs. Training images 6 and 41, 19 and 37, 21 and 43, 22 and 44, 23 and 45, 25 and 42, 27 and 39, 31 and 38, 36 and 40 are the two different versions of the considered traffic signs used in the training set. Two test images formed with different colors and shapes and showing the necessity of using more than one training image for the considered sign whose training images are given in Table 4.4, numbered as 25 and 42 are given in Figure 4.4.



**Figure 4.4.** Two Versions of Same Traffic Sign with Different Colors and Shapes [31, 63]

#### **4.4. Test Images**

As test images, 150 images are used. Number of images for each traffic sign is given in the Table 4.5.

**Table 4.5.** Number of Test Images for Each Traffic Sign that Exist in the Training Set

<b>Training image no.</b>	1	2	3	4	5	6, 41	7	8	9	10	11	12	13	14	15	16	17	18
<b>Number of test images</b>	2	6	3	5	8	5	8	3	14	6	4	4	6	3	2	2	4	1
<b>Training image no.</b>	19 , 37	20	21 , 43	22 , 44	23 , 45	24	25 , 42	26	27 , 39	28	29	30	31 , 38	32	33	34	35	36 , 40
<b>Number of test images</b>	3	4	2	2	4	3	6	5	3	7	2	1	5	4	2	4	3	4

Firstly, the regions containing traffic signs are cut and extracted from the images and these regions are used as the test images. The results of three methods described in Section 4.2 for these images are given in Section 5.1.

Afterwards, test images are used without cutting and extracting traffic sign regions, and recognition is performed by the third method which considers color and orientation information. The results are given in Section 5.2.

## **5. RESULTS**

The results of extracted traffic sign images and the results of test images without extracting traffic sign regions are given as follows as described in Section 4.4.

### **5.1. Recognition Results of Extracted Traffic Sign Images**

Recognition results of three methods for the cut and extracted images that contain traffic signs are given in Table B.1 in Appendix B.1.

It is observed that checking the colors of the matching features is very important at recognizing the traffic signs correctly. There are many examples in the test images that reflect this situation like 5, 10, 22, 39, 42, 70, 73, 77, 88, 90, 115, 116, 118, 144, 148. It is seen that for these images the second method which utilizes color information recognizes the traffic signs correctly while the first method can not recognize them.

Checking orientations of the keypoints also contribute to the accurate recognition of the traffic signs. For the traffic signs which have similar symmetrically, checking orientations finds the correct traffic sign among them. Test images 6, 13, 37, 40, 41, 60, 124 and 146 are the examples of this situation. The third method which utilizes orientation information recognizes these traffic signs correctly while the first and the second methods can not recognize them.

Even for the traffic signs that do not have symmetrically similar, checking orientations eliminates the wrong matches by discarding the matches that do not pass the orientation inspection. In test images 15, 30, 45, 65, 83, 84, 96, 112, 141, 149, 150, correct traffic signs are found by checking the orientations.

It can be seen from test images 43, 55 and 56 that while passing from the first method to the second method, the color inspection eliminates the wrong matches where two corresponding pixels have different colors and while passing from the second method to the third method, the orientation inspection finds the correct traffic

signs by eliminating the matches where two corresponding features have orientations which have a big distance.

Rotated traffic signs can be recognized accurately even if there is a strong rotation as in test image 52. Other example test images that contain rotated traffic signs are 8, 18, 20, 33, 51, 52, 57, 68, 97, 98, 100, 111, 121, 130. For all these test images even the first method recognizes the traffic signs accurately.

It is seen that recognition of the traffic signs which undergo affine transformations can be accomplished successfully too. There is a strong affine transformation in the test image 46, but the traffic sign can be recognized accurately even there is only one match. Test images 3, 14, 108, 128, 131, 150 are the other examples containing affine transformations and can be recognized by all three methods. Test image 108 can be recognized accurately although the traffic sign in the test image is different in shape than the corresponding training image. Test images 42, 115, 116 and 144 are the ones which undergo affine transformations and can be recognized accurately after the color inspection.

Some of the test images include "MATTON" label, which falls onto the traffic signs, indicating the website they are taken from, but even they can be recognized correctly. The examples of this situation are the test images 137, 138, 139, 140, 141, 148 and 150. Recognition of all these traffic signs can be done successfully by the third method. Some of them can be recognized even before the color and orientation inspections.

It is obvious that when the size of the test image is not small, recognition can be performed easily since there will be many features extracted by SIFT and will be matched to the corresponding training image. However, when the size of the test image is very small, few features are extracted by SIFT and the recognition process becomes harder. Test images 73, 78, 79, 80, 81, 82, 83, 84, 85, 136, 142, 143, 145 and 146 are the example images that have very small sizes, nevertheless in most of them three methods recognize traffic signs accurately. In few of them, color or orientation inspection is required for the accurate recognition.

Recognition of the traffic signs whose color has changed with time by the effects which are stated in Section 2 are performed accurately. The example test images are 29, 32, 122 where three of the methods recognize them correctly.

Test images including overshadowed traffic signs as 17 and 38 are recognized correctly by all of the three methods. Test image 56 is another example but for this image, color and orientation inspections are required, so it is recognized by the third method.

Test image 12 is an example where the effect of snow on the appearance of the traffic sign is shown, but it is seen that this situation does not prevent the accurate recognition of this sign by all three methods.

Effect of the direction and the strength of the light on the color perceived as mentioned in Section 2 can be seen from test images 21, 40, 54, 77 where the traffic signs seem darker than the test images 28, 73, 86, 87 where the traffic signs are bright. It is observed that this situation does not affect the recognition process since the color classification method includes all dark and bright tones of colors.

Occluded traffic signs as in test images 43, 44 and 45 can also be recognized correctly by the third method, where even 44 does not necessitate neither color nor orientation inspections.

Test image 48 is an example of blurring caused by motion but in spite of this situation, the traffic sign can be recognized by all three methods.

As a consequence, it is seen that the first method shows 74% performance by recognizing 111 out of 150 test images accurately. In the second method, adding color information raises the performance to 84.67% by recognizing 127 out of 150 test images accurately. The performance of the third method which also includes orientation information attains 99.33% by recognizing only one test image incorrectly and recognizing 149 out of 150 images correctly. Importance of adding color and orientation inspections is obvious in order to raise the performance so much.

The results of the first method are given in the confusion matrix which is shown in Table 5.1. Column and row names of this matrix are given as the training image no.s of the traffic signs given as in Table 4.4. The numbers colored black are the numbers of correctly detected test images. These numbers take part in the diagonal of the matrix where the same no.s intersect. The numbers colored red are the numbers of wrongly detected test images and these numbers take part outside of the diagonal of the matrix.

**Table 5.1.** Confusion Matrix Created for the Recognition of Test Images by the First Method

	1	2	3	4	5	6, 41	7	8	9	10	11	12	13	14	15	16	17	18	19, 37	20	21, 43	22, 44	23, 45	24	25, 42	26	27, 39	28	29	30	31, 38	32	33	34	35	36, 40		
1	2																																					
2	1	5																																				
3		1	2																																			
4				5																																		
5					8																																	
6, 41						3								1																								1
7			1	1	1		5																															
8							2	0												1																		
9									14																													
10					1					4											1																	
11	1										2										1																	
12												4																										
13				1		1							4																									
14	1														2																							
15																2																						
16																	1																				1	
17																		4																				
18																			1																			
19, 37																				3																		
20			2																		0	1		1														
21, 43	1					1															0	0																
22, 44						2																0																
23, 45																							4															
24	1																							1												1		
25, 42																									6													
26																										6												
27, 39																											5											
28						1															1								2									
29																													1	3								
30																																						
31, 38																																					1	
32		1																																			3	
33																																						2
34																																						4
35																																						1
36, 40	1																																					2

It is seen from Table 5.1 that there are a lot of numbers outside of the diagonal of the matrix which represent the false recognitions. For some training image no.s, none of the test images are recognized as 8, 20, 21, 22. However, some of the traffic signs which have training no as 5, 9 and 25-42 are recognized in all corresponding test images, this situation can occur from the characteristic shape of these traffic signs.. Besides, for example, test images which belong to training image no 8, can be recognized as 7 or 19-37 which are different in color since there is no color check between matching features.

The results of the second method are given in the confusion matrix which is shown in Table 5.2.

**Table 5.2.** Confusion Matrix Created for the Recognition of Test Images by the Second Method

	1	2	3	4	5	6, 41	7	8	9	10	11	12	13	14	15	16	17	18	19, 37	20	21, 43	22, 44	23, 45	24	25, 42	26	27, 39	28	29	30	31, 38	32	33	34	35	36, 40			
1	2																																						
2	1	5																																					
3	1		2																																				
4				5																																			
5					8																																		
6, 41						3								1																								1	
7				1	1		6																																
8								3																															
9									14																														
10						1				5																													
11											3																												1
12												4																											
13													6																										
14														3																									
15															2																								
16	1															1																							
17																	4																						
18																		1																					
19, 37																			3																				
20	1	2																		1																			
21, 43																					1																		
22, 44																						0																	
23, 45																							4																
24																								3															
25, 42																									6														
26																										5													
27, 39																												3											
28																													2	4									
29																																							2
30																																						1	
31, 38																																						1	3
32	1																																					3	
33																																						2	
34																																						4	
35																																						1	
36, 40																																						1	3

Decrease in the numbers outside of the diagonal of the matrix is observed in Table 5.2. As instances, recognized test image numbers increase for 8, 20 and 21. There is no corruption in the ones which are recognized by the first method. After the color check, test images which belong to training image no 8 are recognized accurately. This situation can also be seen for different signs. However, since there is not any orientation inspection, similar images with different orientations can be confused. For example, all test images which belong to training image no 22-44 are recognized as 6 which are different in orientation.

The results of the third method are given in the confusion matrix which is shown in Table 5.3.

**Table 5.3.** Confusion Matrix Created for the Recognition of Test Images by the Third Method

	1	2	3	4	5	6, 41	7	8	9	10	11	12	13	14	15	16	17	18	19, 37	20	21, 43	22, 44	23, 45	24	25, 42	26	27, 39	28	29	30	31, 38	32	33	34	35	36, 40		
1	2																																					
2		6																																				
3	1		2																																			
4				5																																		
5					8																																	
6, 41						5																																
7							8																															
8								3																														
9									14																													
10										6																												
11											4																											
12												4																										
13													6																									
14														3																								
15															2																							
16																2																						
17																	4																					
18																		1																				
19, 37																			3																			
20																				4																		
21, 43																					2																	
22, 44																						2																
23, 45																							4															
24																								3														
25, 42																									6													
26																										5												
27, 39																											3											
28																												7										
29																													2									
30																														1								
31, 38																															5							
32			1																																			
33																																			4			
34																																				2		
35																																					4	
36, 40																																					3	
																																					3	
																																					4	

It is observed in Table 5.3, that almost there is no number outside of diagonal of the matrix. Only, for training image no 3, there is one outlier, which is recognized as traffic sign that have training no 1. Similarity of the traffic signs that exist in no.s 1 and 3 in shape is obvious, so they are confused only in one image. Performing check on the orientations corrects the recognition results for training image no 22-44, all test images are recognized accurately by the third method. Importance of adding orientation inspection on recognition performance is seen.

**5.2. Recognition Results of Test Images without Extracting Traffic Signs**

Three methods are performed for the test images without extracting traffic sign regions. It is seen that the first method shows 48.67% performance by recognizing 73 out of 150 test images accurately. In the second method, adding color information raises the performance to 56.67% by recognizing 85 out of 150 test images

accurately. The performance of the third method which also includes orientation information attains 64% by recognizing 96 out of 150 images correctly. This performance is not bad, but is not good as the performance presented in Section 5.1. However, adding color and orientation information raises the performance of SIFT significantly again. Recognition results of the third method which considers color and orientation information and shows superior performance when compared to the first and the second methods are given in Table B.2 in Appendix B.2.

As it can be seen in Table B.2, for some test images, the accurate recognition is succeeded. In the recognition results of these test images, the location of the sign in the test images can be detected from the lines showing matches and the matched features. Therefore, detecting location of the traffic sign and extracting the outer edges by another algorithm as a separate detection component as the first step is not required anymore. This feature is an important advantage of the algorithm.

But in several of test images, the results are not correct. In general, if the traffic sign have a big in size in the image, it is recognized accurately. However, there are examples of incorrect recognition even the traffic sign is big in size in the image as test images 14, 139, 140. There are also accurate recognition examples in which the traffic sign is small in size and there are many other objects like test images 17, 19, 28, 31, 35. More examples can be given. Therefore, predicting if recognition will be correct or will be incorrect is very difficult for any given image.

By investigating the keypoints which are extracted by SIFT, it is found that different keypoints are found within the traffic sign regions in extracted traffic sign images and in test images without extracting traffic sign regions, even though these regions are same in size in two images. Keypoints are affected by other keypoints which exist in the image, so this situation causes difference in number and distribution of keypoints in the image. In general the number of keypoints within the traffic sign region decreases if the images without extracting traffic sign regions are used. Therefore the recognition process becomes difficult and the performance is not high as in the ones presented in Section 5.1. Besides, there occur many false matches to other objects like cars, buildings, humans, roads, trees, etc., and this situation can cause recognizing traffic signs wrongly. This situation affects the performance very much. However the performance can be increased by adding more features and

inspections. The investigations that can be performed and tried in order to increase the performance for these images are described in Section 6.1.

## 6. CONCLUSION

This thesis proposed a colored traffic sign recognition system which uses SIFT to find and recognize the traffic sign which exist in an image. Adding new features which will increase the performance of SIFT in this issue is intended. New contributions are included for the color classification method that is used for comparing the colors of the matched feature pixels and checking the orientations of SIFT features which are matched in two images.

After giving insight about traffic sign detection and recognition systems, the difficulties that are faced and SIFT, implementation details were described in Section 4 and obtained results were given in Section 5. Furthermore, comments were made on these results and explanations were provided. To sum up, the following conclusions can be made.

It is seen that adding color and orientation inspections provide a great increase in the recognition rate. After adding these two features, almost all test images are recognized accurately.

Color inspection was performed by using proposed color classification method which looks at  $H$ ,  $S$ ,  $V$  values of a pixel and put it in a color class according to the rules that are developed. Investigations which are performed and described in Section 4.2.2 showed that this color classification method performs accurately and gives better results than other approaches.

Obtained results for the extracted traffic sign images are very good and satisfying even for the test images containing traffic signs which are rotated, undergone affine transformations, damaged, occluded, overshadowed, had alteration in color, pictured in different weather conditions, pictured in different illumination conditions. Many algorithms mentioned in Section 1 suffer in this issue and the high performance reached in this thesis can not be obtained.

It is important to see that these good results are obtained by using only one training image for each version of each traffic sign. Methods like template matching do not include this feature.

Another contribution is showing that using a local feature technique as SIFT, detecting location of the traffic sign and segmenting it as the first step are not required anymore after obtaining better results for the images without extracting traffic sign regions. Because SIFT features that are matched to the training image, which contains corresponding traffic sign, show the location of the sign in a given image containing a traffic sign.

### **6.1. Future Work**

Further investigations can be done in order to increase the performance at recognition of test images without cutting and extracting traffic sign regions. There are several approaches which can be thought. For example, the matches which are crowded in some locations in the image can be accepted since it can be thought that there exists a traffic sign and the outlier matches which are far away from other matches can be rejected. However, sometimes there can be few matches between test and training image even the training image includes the correct sign. Therefore, this approach can fail if crowding of matches can not become definite. Increasing *distRatio* parameter of SIFT can be tried since it increases the number of matches between two images. Also, size of the traffic sign in the image is not fixed and not known, so determination of measure of crowding can be very difficult in this approach.

Another approach can be checking firstly the relations like distance ratios between the features in the training image where matches are found. Then the relation between the features in the test image where matches are found can be checked. If the relations are similar, considered traffic sign can be accepted as the correct result. But the difficulty that can be faced in this approach may be the difficulty of distinguishing the correct matches from the false matches, so it is a problem to know which relations between which features that are matched will be considered.

The subject that the lines which show the matches between test and training images should not intersect if the sign is recognized correctly can also be considered too in

order to increase the performance at recognition of test images without cutting and extracting traffic sign regions. However, there can be false matches between training and test images even the found training image as the result is the accurate one. This is due to the presence of different objects in the test images. There are roads, cars, humans, buildings, etc. Therefore, sets of lines of false matches and sets of lines of correct matches can intersect and again it is a problem to know which matches will be accepted as true matches and which of them will be eliminated since it is not known which of them are correct.

Hough Transform or RANSAC can also be tried in order to discard false matches.

In addition, after obtaining good results for the test images without cutting and extracting traffic sign regions, implementation of recognizing more than one traffic sign can be added as a new feature, since it is assumed that there exists exactly one traffic sign in a given image for the present.

Another assumption is that a test image always contains a traffic sign. Consequently, the recognition step is carried on and a result is produced even if there is no traffic sign in the test image. Determining existence or nonexistence of a traffic sign in an image would be a good feature that can be included.

For a more comprehensive system, the variety of traffic signs in the training set can be increased.

## REFERENCES

- [1] **Fleyeh, H. and Dougherty, M.**, 2005. Road and Traffic Sign Detection and Recognition, *Proceedings of the 16th Mini - EURO Conference and 10th Meeting of EWGT*.
- [2] **Fang, C., Chen, S. and Fuh, C.**, 2003. Road Sign Detection and Tracking, *IEEE Transactions on Vehicular Technology*, **52**, 1329-1341.
- [3] **Lowe, D.G.**, 2004. Distinctive Image Features from Scale-Invariant Keypoints, *International Journal of Computer Vision*.
- [4] **Mikolajczyk, K., Leibe, B. and Schiele, B.**, 2005. Local Features for Object Class Recognition, *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05)*, 17-21 October, pp. 1792-1799.
- [5] **Mikolajczyk, K. and Schmid, C.**, 2003. A Performance Evaluation of Local Descriptors, *Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'03)*, 18-20 June, pp. 257-264.
- [6] **Mikolajczyk, K. and Schmid, C.**, 2005. A Performance Evaluation of Local Descriptors, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**, 1615-1630.
- [7] **Se, S., Lowe, D.G. and Little, J.**, 2001. Vision-Based Mobile Robot Localization and Mapping Using Scale-Invariant Features, *International Conference on Robotics and Automation*, pp. 2051-2058.
- [8] **Se, S., Lowe, D.G. and Little, J.**, 2002. Global Localization Using Distinctive Visual Features, *International Conference on Intelligent Robots and Systems*, 30 September-5 October, pp. 226-231.
- [9] **Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T. and Van Gool, L.**, 2006. A Comparison of Affine Region Detectors, *International Journal of Computer Vision*.
- [10] **Paclik, P. and Novovicova, J.**, 2000. Road Sign Classification without Color Information, *Sixth Annual Conference of the Advanced School for Computing and Imaging*.
- [11] **Lu, S.**, 1994. Recognition of Traffic Signs Using a Multilayer Neural Network, *Canadian Conference on Electrical and Computer Engineering*, 25-28 September, pp. 833-834.
- [12] **De la Escalera, A., Armingol, J. and Mata, M.**, 2003. Traffic Sign Recognition and Analysis for Intelligent Vehicles, *Image and Vision Comput.*, **21**, 247-258.

- [13] **De la Escalera, A., Armingol, J. and Pastor, J.**, 2004. Visual Sign Information Extraction and Identification by Deformable Models for Intelligent Vehicles, *IEEE Transactions on Intelligent Transportation Systems*, **5**, 57-68.
- [14] **Fleyeh, H.**, 2004. Color Detection and Segmentation for Road and Traffic Signs, *IEEE Conference on Cybernetics and Intelligent Systems*, pp. 809-814.
- [15] **Vitabile, S., Gentile, A. and Sorbello, F.**, 2002. A Neural Network Based Automatic Road Sign Recognizer, *The 2002 Inter. Joint Conf. on Neural Networks*, 12-17 May, pp. 2315-2320.
- [16] **Vitabile, S., Gentile, A., Dammone, G. and Sorbello, F.**, 2002. Multi Layer Perceptron Mapping on a SIMD Architecture, *The 2002 IEEE Signal Processing Society Workshop*.
- [17] **Vitabile, S. and Sorbello, F.**, 1998. Pictogram Road Signs Detection and Understanding in Outdoor Scenes, *Conference on Enhanced and Synthetic Vision*.
- [18] **Piccioli, G., De Micheli, E., Parodi, P. and Campani, M.**, 1996. Robust Method for Sign Detection and Recognition, *Journal of Image and Vision Computing*, **14**, 209-223.
- [19] **Betke, M. and Makris, N.**, 1995. Fast Object Recognition in Noisy Images Using Simulated Annealing, *Fifth International Conference on Computer Vision*, 20-23 June, pp. 523-530.
- [20] **Hibi, T.**, 1996. Vision Based Extraction and Recognition of Road Sign Region from Natural Color Image by Using HSL and Coordinates Transformation, *29th International Symposium on Automotive Technology and Automation, Robotics, Motion and Machine Vision in the Automotive Industries*.
- [21] **Huang, C. and Hsu, S.**, 2000. Road Sign Interpretation Using Matching Pursuit Method, *4th IEEE Southwest Symposium on Image Analysis and Interpretation*, April, pp. 202-208.
- [22] **Fang, C., Chen, S., Yen, P. and Fuh, C.**, 2003. Road Sign Recognition System Based on Dynamic Visual Model, *IEEE Computer Conference on Computer Vision and Pattern Recognition*, pp 750-755.
- [23] **Lauziere, Y., Gingras, D. and Ferrie, F.**, 2001. A Model-Based Road Sign Identification System, *IEEE Computer Conference on Computer Vision and Pattern Recognition*, pp. 1163-1170.
- [24] **Paclik, P., Novovicova, J., Pudil, P. and Somol, P.**, 2000. Road Sign Classification Using Laplace Kernel Classifier, *Pattern Recognition Letters*, **21**, 1165-1173.
- [25] **Jiang, G. and Choi, T.**, 1998. Robust Detection of Landmarks in Color Image Based on Fuzzy Set Theory, *Fourth International Conference on Signal Processing*, 12-16 October, pp. 968-971.
- [26] **Sanjida, M., Abdul Halim, M., Shaukatul, I. and Khandoker, R.A.** Automatic Road Sign Detection and Extraction, *B.Sc Graduation*

*Project*, Northern University Bangladesh, Department of Computer Science and Engineering.

- [27] *Detection and Recognition of Road Signs Using Scale Invariant Feature Transform*. Chen, P.-F. Feb. 2008 <[http://etds.yzu.edu.tw/etdservice/view\\_metadata?etdun=U00090112200611305892&query\\_field1=keyword&&query\\_word1=.&](http://etds.yzu.edu.tw/etdservice/view_metadata?etdun=U00090112200611305892&query_field1=keyword&&query_word1=.&)>
- [28] *Color Principles – Hue, Saturation, and Value*. Feb. 2008 <[http://www.ncsu.edu/scivis/lessons/colormodels/color\\_models2.html](http://www.ncsu.edu/scivis/lessons/colormodels/color_models2.html)>
- [29] *HSL and HSV*. 20 Apr. 2008. Wikipedia. Feb. 2008 <[http://en.wikipedia.org/wiki/HSL\\_and\\_HSV](http://en.wikipedia.org/wiki/HSL_and_HSV)>
- [30] **Miura, J., Kanda, T. and Shirai, Y.**, 2000. An Active Vision System for Real-Time Traffic Sign Recognition, *IEEE Intelligent Transportation Systems*, 1-3 October, pp. 52-57.
- [31] *Photo Gallery*. Feb. 2008 <[http://roadanalysis.uah.es/page\\_files/SignsGallery.html](http://roadanalysis.uah.es/page_files/SignsGallery.html)>
- [32] *Road Signs in Germany*. 26 Aug. 2006. Wikimedia Commons. Mar. 2008 <[http://commons.wikimedia.org/wiki/Category:Road\\_signs\\_in\\_Germany](http://commons.wikimedia.org/wiki/Category:Road_signs_in_Germany)>
- [33] *Road Traffic Signs*. 2 Mar. 2005. Freefoto. Mar. 2008 <<http://www.freefoto.com/preview/41-07-90?ffid=41-07-90&k=Road+Traffic+Signs>>
- [34] *Spanish Road Sign*. 4 Jun. 2004. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-02-53?ffid=41-02-53&k=Spanish+Road+Sign>>
- [35] *No U Turn – Road Sign*. 6 Apr. 2006. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-06-91?ffid=41-06-91&k=No+U+Turn++Road+Sign>>
- [36] *Traffic Sign*. 22 Apr. 2008. Wikipedia. Apr. 2008 <[http://en.wikipedia.org/wiki/Traffic\\_sign](http://en.wikipedia.org/wiki/Traffic_sign)>
- [37] *Matton Images*. Feb. 2008 <<http://www.mattonimages.co.uk/search.php?sok=traffic+sign&lsok=&notrel=&inres=&obj=>>>
- [38] *Road Liable to Flooding – Road Sign*. 4 Oct. 2004. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-08-94?ffid=41-08-94&k=Road+Liable+To+Flooding+Road+Sign>>
- [39] *Traffic Sign*. 30 Mar. 2008. Wikimedia Commons. Feb. 2008 <[http://commons.wikimedia.org/wiki/Traffic\\_sign](http://commons.wikimedia.org/wiki/Traffic_sign)>
- [40] **Perez, E. and Javidi, B.**, 2000. Composite Filter Bank for Road Sign Recognition, *13th Annual Meeting IEEE Lasers and Electro-Optics Society*, 13-16 November, pp. 754-755.
- [41] *CPSC 425: Computer Vision Course Slides*. 18 Mar. 2007. Lowe, D. Feb. 2008 <[www.cs.ubc.ca/~lowe/425/slides/10-sift-6up.pdf](http://www.cs.ubc.ca/~lowe/425/slides/10-sift-6up.pdf)>
- [42] **Witkin, A.P.**, 1983. Scale-Space Filtering, *International Joint Conference on Artificial Intelligence*, pp. 1019-1022.

- [43] **Koenderink, J.J.**, 1984. The Structure of Images, *Biological Cybernetics*, **50**, pp. 363-396.
- [44] **Lindeberg, T.**, 1994. Scale-Space Theory: A Basic Tool for Analysing Structures at Different Scales, *Journal of Applied Statistics*, **21**, pp. 224-270.
- [45] **Lowe, D.G.**, 1999. Object Recognition from Local Scale-Invariant Features, *International Conference on Computer Vision*, 20-27 September, pp. 1150-1157.
- [46] **Mikolajczyk, K.**, 2002. Detection of Local Features Invariant to Affine Transformations, *PhD thesis*, Institute National Polytechnique de Grenoble, France.
- [47] **Brown, M. and Lowe, D.G.**, 2002. Invariant Features from Interest Point Groups, *British Machine Vision Conference*, pp. 656-665.
- [48] **Harris, C. and Stephens, M.**, 1988. A Combined Corner and Edge Detector, *Fourth Alvey Vision Conference*, pp. 147-151.
- [49] *Demo Software: SIFT Keypoint Detector*. Jul. 2005. David Lowe. Feb. 2008 <<http://www.cs.ubc.ca/~lowe/keypoints>>
- [50] *MATLAB Function Reference – rgb2hsv*. Feb. 2008 <<http://www.mathworks.com/access/helpdesk/help/techdoc/ref/rgb2hsv.html>>
- [51] *Warning Sign*. 25 Apr. 2008. Wikipedia. Apr. 2008 <[http://en.wikipedia.org/wiki/Warning\\_sign](http://en.wikipedia.org/wiki/Warning_sign)>
- [52] *Crossing Signs*. 13 Feb. 2008. Wikimedia Commons. Apr. 2008 <[http://commons.wikimedia.org/wiki/Crossing\\_signs](http://commons.wikimedia.org/wiki/Crossing_signs)>
- [53] *Manual of Traffic Signs*. 5 Jan. 2008. Moeur, R.C. Apr. 2008 <<http://trafficsign.us>>
- [54] *Diagrams of Road Signs of Italy*. 2 Dec. 2007. Wikimedia Commons. Mar. 2008 <[http://commons.wikimedia.org/wiki/Category:Diagrams\\_of\\_road\\_signs\\_of\\_Italy](http://commons.wikimedia.org/wiki/Category:Diagrams_of_road_signs_of_Italy)>
- [55] *Road Signs of Germany*. 13 Jul. 2007. Wikimedia Commons. Mar. 2008 <[http://commons.wikimedia.org/wiki/Category:Road\\_signs\\_of\\_Germany](http://commons.wikimedia.org/wiki/Category:Road_signs_of_Germany)>
- [56] *Diagrams of Road Signs of Austria*. 19 Apr. 2008. Wikimedia Commons. Apr. 2008 <[http://commons.wikimedia.org/wiki/Category:Diagrams\\_of\\_road\\_signs\\_of\\_Austria](http://commons.wikimedia.org/wiki/Category:Diagrams_of_road_signs_of_Austria)>
- [57] *Diagrams of Road Signs of the Netherlands*. 4 Jun. 2007. Wikimedia Commons. Apr. 2008 <[http://commons.wikimedia.org/wiki/Category:Diagrams\\_of\\_road\\_signs\\_of\\_the\\_Netherlands](http://commons.wikimedia.org/wiki/Category:Diagrams_of_road_signs_of_the_Netherlands)>
- [58] *Diagrams of Road Signs of Poland*. 4 Jun. 2007. Wikimedia Commons. Apr. 2008 <[http://commons.wikimedia.org/wiki/Category:Diagrams\\_of\\_road\\_signs\\_of\\_Poland](http://commons.wikimedia.org/wiki/Category:Diagrams_of_road_signs_of_Poland)>

- [59] *AOL Member Page Directory Listing*. 21 Jul. 1996. <<http://members.aol.com/rmoeuradot/200x200/guide/D9-6.gif>>
- [60] *Diagrams of Road Signs of Spain*. 4 Jun. 2007. Wikimedia Commons. Apr. 2008 <[http://commons.wikimedia.org/wiki/Category:Diagrams\\_of\\_road\\_signs\\_of\\_Spain](http://commons.wikimedia.org/wiki/Category:Diagrams_of_road_signs_of_Spain)>
- [61] *Know Your Traffic Signs*. 24 Jul. 2007. Department for Transport, Mar. 2008 <<http://www.dft.gov.uk/pgr/roads/tss/trafficsigns.pdf>>
- [62] *Spanish Road Sign*. 5 Jun. 2004. Freefoto. Mar.2008 <<http://www.freefoto.com/preview/41-02-64?ffid=41-02-64&k=Spanish+Road+Sign>>
- [63] *Freefoto*. Mar. 2008 <<http://www.freefoto.com/browse/41-03-0?ffid=41-03-0>>
- [64] *Wrong Way Road Traffic Sign*. 15 Aug. 2003. Freefoto. Mar. 2008 <<http://www.freefoto.com/preview/41-07-9?ffid=41-07-9&k=Wrong+Way+Road+Traffic+Sign>>
- [65] *Road Signs in Austria*. 17 Jul. 2007. Wikimedia Commons. Feb. 2008 <[http://commons.wikimedia.org/wiki/Category:Road\\_signs\\_in\\_Austria](http://commons.wikimedia.org/wiki/Category:Road_signs_in_Austria)>
- [66] *Webshots*. Mar. 2008 <[http://thumb4.webshots.net/t/63/463/1/50/28/2860150280052197924DCLoYZ\\_th.jpg](http://thumb4.webshots.net/t/63/463/1/50/28/2860150280052197924DCLoYZ_th.jpg)>
- [67] *Motorail Sign*. 22 Aug. 2007. Wikimedia Commons. Feb. 2008 <[http://commons.wikimedia.org/wiki/Image:Motorail\\_sign.jpg](http://commons.wikimedia.org/wiki/Image:Motorail_sign.jpg)>
- [68] *Wrong Way*. Getty Images. Mar. 2008 <<http://www.gettyimages.com/Search/Search.aspx?src=quick&contractUrl=1&family=creative&phrase=wrong%20way#>>
- [69] *Road Signs in Czech Republic*. 3 Jun. 2007. Wikimedia Commons. Mar. 2008 <[http://commons.wikimedia.org/wiki/Category:Road\\_signs\\_in\\_the\\_Czech\\_Republic](http://commons.wikimedia.org/wiki/Category:Road_signs_in_the_Czech_Republic)>
- [70] *Road Signs in the Netherlands*. 1 Sep. 2006. Wikimedia Commons. Mar. 2008 <[http://commons.wikimedia.org/wiki/Category:Road\\_signs\\_in\\_the\\_Netherlands](http://commons.wikimedia.org/wiki/Category:Road_signs_in_the_Netherlands)>
- [71] *Traffic Sign Image Database*. Feb. 2008 <[http://www.cs.rug.nl/~imaging/databases/traffic\\_sign\\_database/traffic\\_sign\\_database.html](http://www.cs.rug.nl/~imaging/databases/traffic_sign_database/traffic_sign_database.html)>
- [72] *Road Sign*. 2 Mar. 2001. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-03-8?ffid=41-03-8&k=Road+Sign>>
- [73] *No Left Turn*. 2 Mar. 2001. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-03-10?ffid=41-03-10&k=No+Left+Turn>>
- [74] *Keep Right – Road Sign*. 25 Aug. 2002. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-03-24?ffid=41-03-24&k=Keep+Right+-+Road+Sign>>
- [75] *No Right Turn*. 7 Apr. 2002. Freefoto. Mar. 2008 <<http://www.freefoto.com/preview/41-01-60?ffid=41-01-60&k=No+Right+Turn>>
- [76] *Railroad Crossing – French Road Sign*. 21 Aug. 2002. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-03-89?ffid=41-03-89&k=Signal+Automatique+-+Railroad+Crossing+-+French+Road+Sign>>

- [77] *Bicycle Route – Road Sign*. 30 May 2002. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-06-88?ffid=41-06-88&k=Bicycle+Route+-++Road+Sign>>
- [78] *Disabled Parking – Road Traffic Sign*. 10 Apr. 2003. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-07-59?ffid=41-07-59&k=Disabled+Parking+Road+Traffic+Sign>>
- [79] *Road Sign*. 18 Jun. 2005. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-08-40?ffid=41-08-40&k=Road+Sign>>
- [80] *Stop Sign*. 24 Sep. 2006. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-15-6?ffid=41-15-6&k=Stop+Sign>>
- [81] *Stop Sign*. 3 Mar. 2005. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-15-59?ffid=41-15-59&k=Stop+Sign>>
- [82] *S Bend – Road Traffic Sign*. 24 Jul. 2006. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-22-58?ffid=41-22-58&k=S+Bend+Road+Traffic+Sign>>
- [83] *30 km - Road Sign*. 4 Jun. 2004. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-02-58?ffid=41-02-58&k=30+km+Road+Sign>>
- [84] *Road Sign*. 25 Feb. 2001. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-03-53?ffid=41-03-53&k=Road+Sign>>
- [85] *No Right Turn*. 20 Jun. 2001. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-04-76?ffid=41-04-76&k=No+Right+Turn>>
- [86] *Parking Pay Here*. 27 Feb. 2004. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-01-78?ffid=41-01-78&k=Parking+Pay+Here>>
- [87] *Railroad Crossing – French Road Sign*. 22 Aug. 2002. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-03-15?ffid=41-03-15&k=Signal+Automatique++-+Railroad+Crossing+-+French+Road+Sign>>
- [88] *Give Way – Road Sign*. 2 Mar. 2001. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-03-57?ffid=41-03-57&k=Give+Way+-+Road+Sign>>
- [89] *Bicycle Route – Road Sign*. 23 Aug. 2002. Freefoto. Feb. 2008 <<http://www.freefoto.com/preview/41-06-62?ffid=41-06-62&k=Bicycle+Route+-++Road+Sign>>
- [90] *Keep Left – Road Traffic Sign*. 21 Apr. 2002. Freefoto. Mar. 2008 <<http://www.freefoto.com/preview/41-01-61?ffid=41-01-61&k=Keep+Left+Road+Traffic+Sign>>
- [91] *Keep Left – Road Traffic Sign*. 6 Jul. 2003. Freefoto. Mar. 2008 <<http://www.freefoto.com/preview/41-01-66?ffid=41-01-66&k=Keep+Left+Road+Traffic+Sign>>
- [92] *One Way Sign*. 22 Aug. 2001. Freefoto. Mar. 2008 <<http://www.freefoto.com/preview/41-04-81?ffid=41-04-81&k=One+Way+Sign>>
- [93] *Series of Bends Ahead – Road Sign*. 8 Oct. 2004. Freefoto. Mar. 2008 <<http://www.freefoto.com/preview/41-08-95?ffid=41-08-95&k=Series+of+Bends+Ahead+Road+Sign>>

- [94] *Traffic Calming Road Traffic Sign*. 19 Nov. 2003. Freefoto. Mar. 2008  
<<http://www.freefoto.com/preview/41-01-7?ffid=41-01-7&k=Traffic+Calming+Road+Traffic+Sign>>
- [95] *Matton Images*. Feb. 2008 <<http://www.mattonimages.co.uk/search.php?next=1&fopen=1&sok=road%20sign>>

**APPENDIX A**











**Table A.1.** Complied Color Classes

<b>Color of the First Pixel</b>	<b>Color of the Second Pixel</b>
yellow	yellow yellow/black yellow/white
red	red white/red
black	black black/white blue/black/white green/black yellow/black
white	white black/white blue/white white/red white/green blue/black/white yellow/white
blue	blue blue/white blue/green blue/black/white
green	green blue/green green/black white/green
black/white	black/white black white blue/white blue/black/white green/black white/red white/green yellow/black yellow/white
green/black	green/black green black black/white





















	blue/green blue/black/white white/green yellow/black
blue/green	blue/green blue/white blue blue/black/white green green/black white/green
white/red	white/red red white black/white blue/white white/green blue/black/white yellow/white
blue/black/white	blue/black/white black black/white blue/white blue blue/green green/black white white/red white/green yellow/black yellow/white
white/green	white/green white/red green/black green blue/green blue/white black/white white blue/black/white yellow/white
blue/white	blue/white black/white white blue blue/green blue/black/white white/red white/green yellow/white

yellow/black	yellow/black yellow black black/white blue/black/white green/black
yellow/white	yellow/white yellow white black/white blue/white blue/black/white white/red white/green

**Table A.2.** H, S, V Values for Sample Patches with Known True Colors

True Color	Sample Patch	H	S	V
Blue		0.6667	1	0.9961
		0.6256	0.45-0.55	> 0.52
		0.61	1	> 0.8
		0.5957	~ 0.87	0.22
		0.55-0.58	0.94-1	0.45 - 0.55
		0.55-0.59	0.61-0.8	0.22-0.29
		0.5793	1	0.56
		0.5715	1	0.9961
		0.5901	1	0.5294
		0.6766	1	0.9843

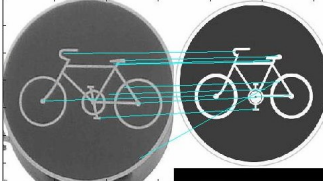
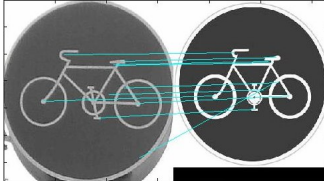
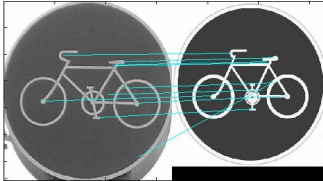
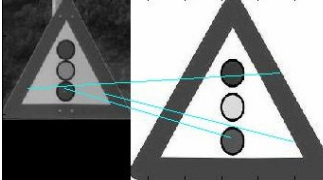
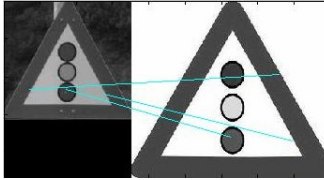
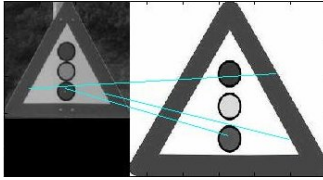



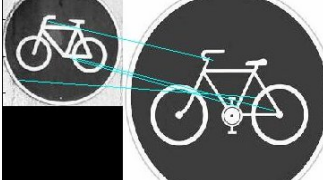
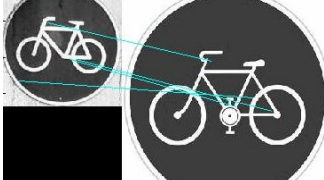
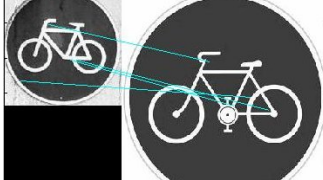
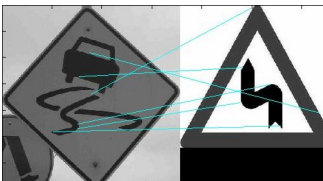


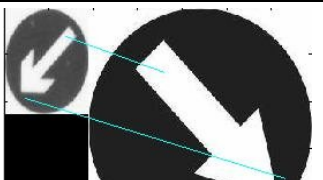
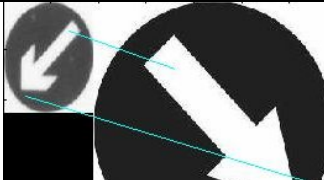

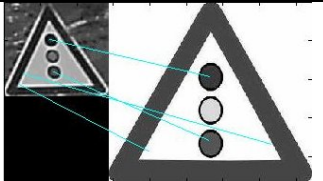
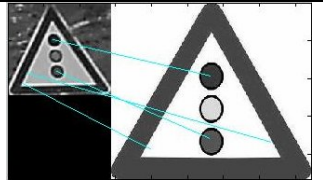
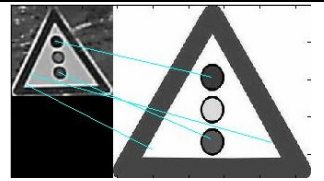
Yellow		0.1667	1	1
		~ 0.133	0.994-1	0.63-0.66
		~ 0.1	0.93-0.94	~ 0.62
		0.0853	~ 1	~ 0.75
		0.1222-0.1278	0.99-1	~ 0.81
		0.0635-0.1970	0.06-0.21	0.35-0.39
		0.1024	0.7-0.8	~ 0.4
		0.1436	~ 1	~ 0.97
		0.1444	~ 1	1
		0.1667	~ 0.99	~ 0.92
Red		0	1	0.9961
		0.9831	0.82	0.6078
		0.9655	0.82	0.29
		0.9993	~ 1	0.94-0.95
		0.015	~ 0.77	~ 0.56
		0.989	0.85	0.8392
		0.0079	0.5872-0.6407	0.8549-0.9059
		0.0101-0.036	0.4615-0.5556	0.27-0.3
		0.9811	0.6163	0.2745-0.3373
		0	0.99	0.7961

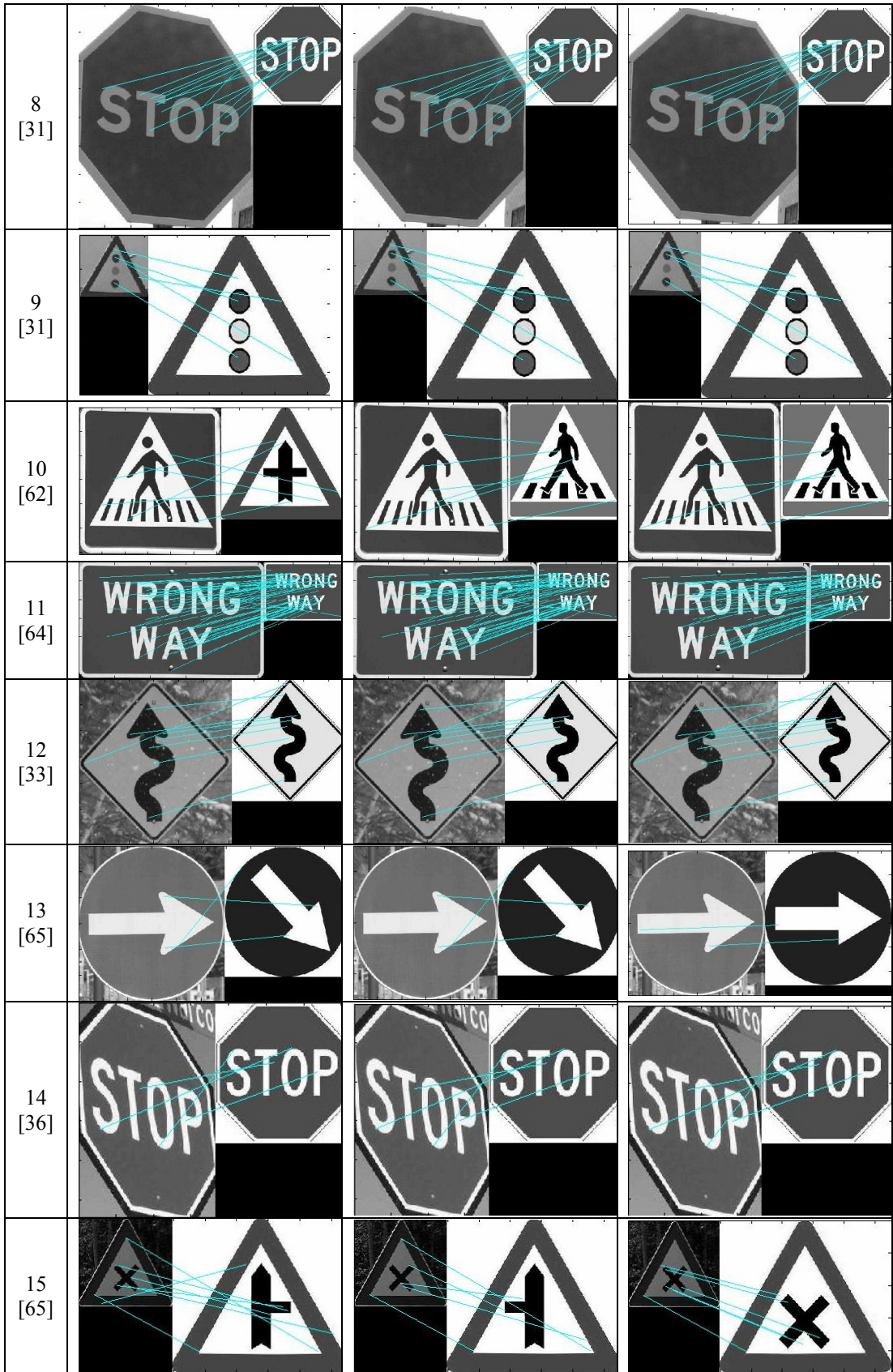
Green		0.47-0.49	0.84-0.9	0.31-0.4
		0.5-0.57	0.8-1	0.22-0.5
		0.48-0.5	~ 1	~ 0.34
		0.45-0.47	~ 1	0.21-0.22
		0.43-0.45	0.598-0.8	~ 0.42
		0.3345	1	~ 0.57
		0.45-0.46	0.74-0.84	0.41-0.45
		0.45-0.46	0.86-1	0.21-0.24
		0.42-0.44	0.6-0.8	0.24-0.3
		0.47-0.486	0.78-0.95	0.32-0.36
Black		0	0	0
		0.7735	0.527	0.29
		0.44-0.67	0.13-0.56	0.13-0.3
		0.5789	> 0.5	0.06-0.27
		0.33-0.41	0.18-1	0.03
		0.4167	0.05	0.15-0.16
		0.59-0.6	0.29-0.33	0.32-0.39
		0.6667	0.22-1	0.007-0.035
		0.6667	0.6667-1	0.0078-0.0275
		0.53-0.61	0.1-0.22	0.14-0.21

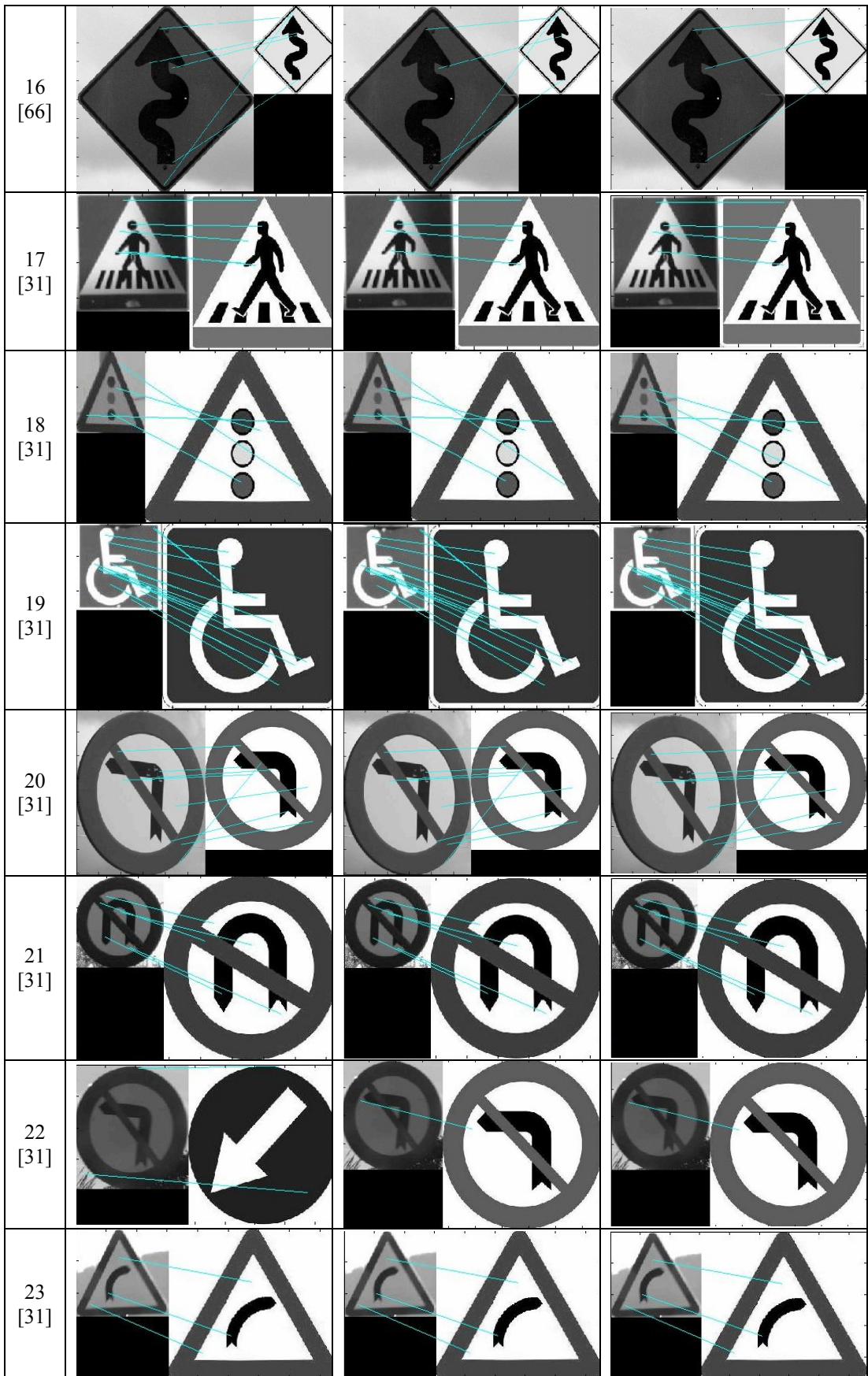
White			0	0	1
			0.6042	~ 0	> 0.7
			0.48-0.53	0.04-0.11	0.4157
			0.1944-0.2	0.041-0.0496	0.4667-0.4784
			0.1444	0.064-0.069	0.85-0.91
			0.1-0.12	0.1-0.26	0.3-0.45
			0.11-0.41	0.0198-0.129	0.36-0.42
			0.52-0.55	0.29-0.34	0.31-0.34
			0.21-0.4	0.0185-0.1	0.615-0.65
			0.21-0.55	0.025-0.063	0.8863-0.9608

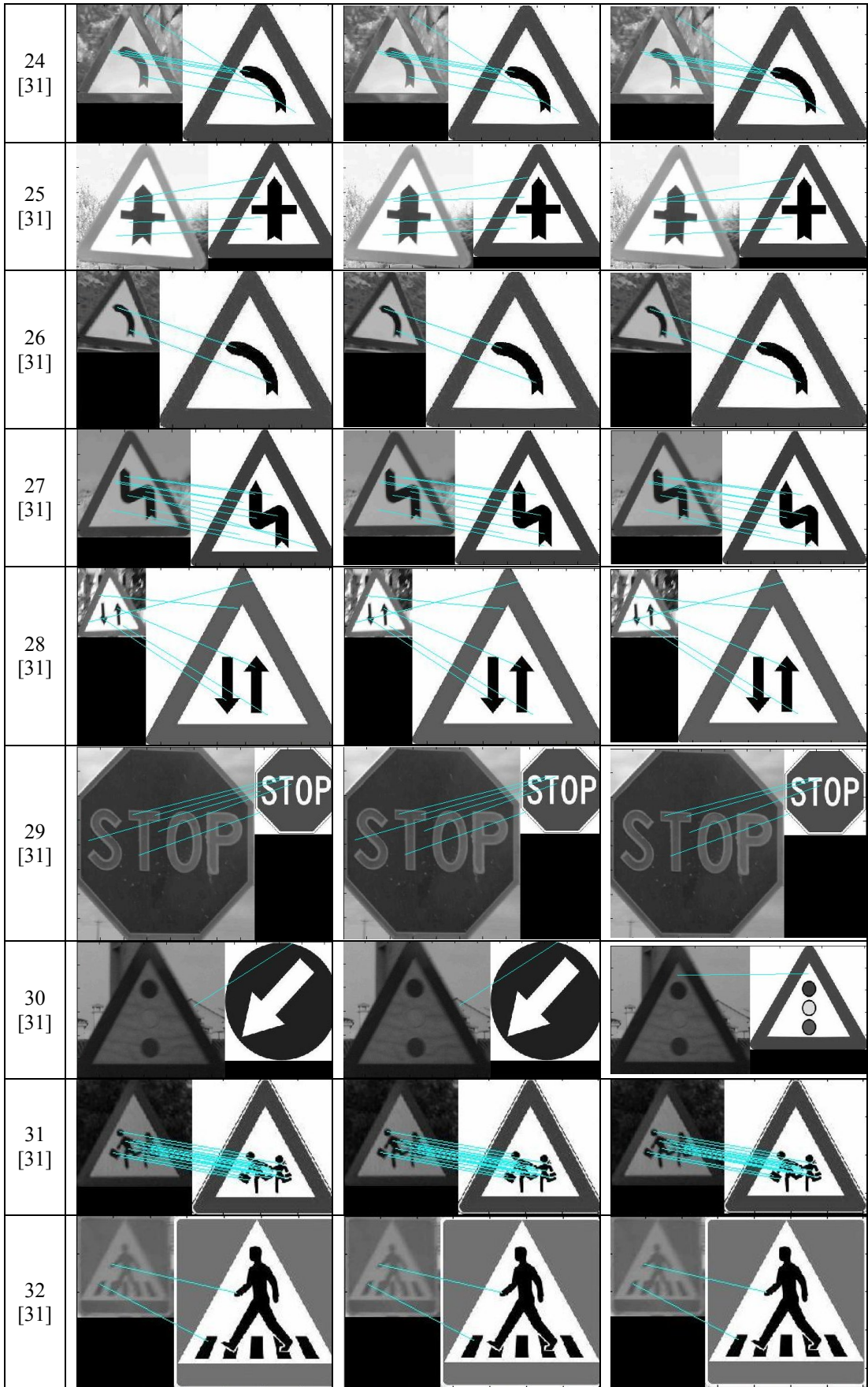
## APPENDIX B

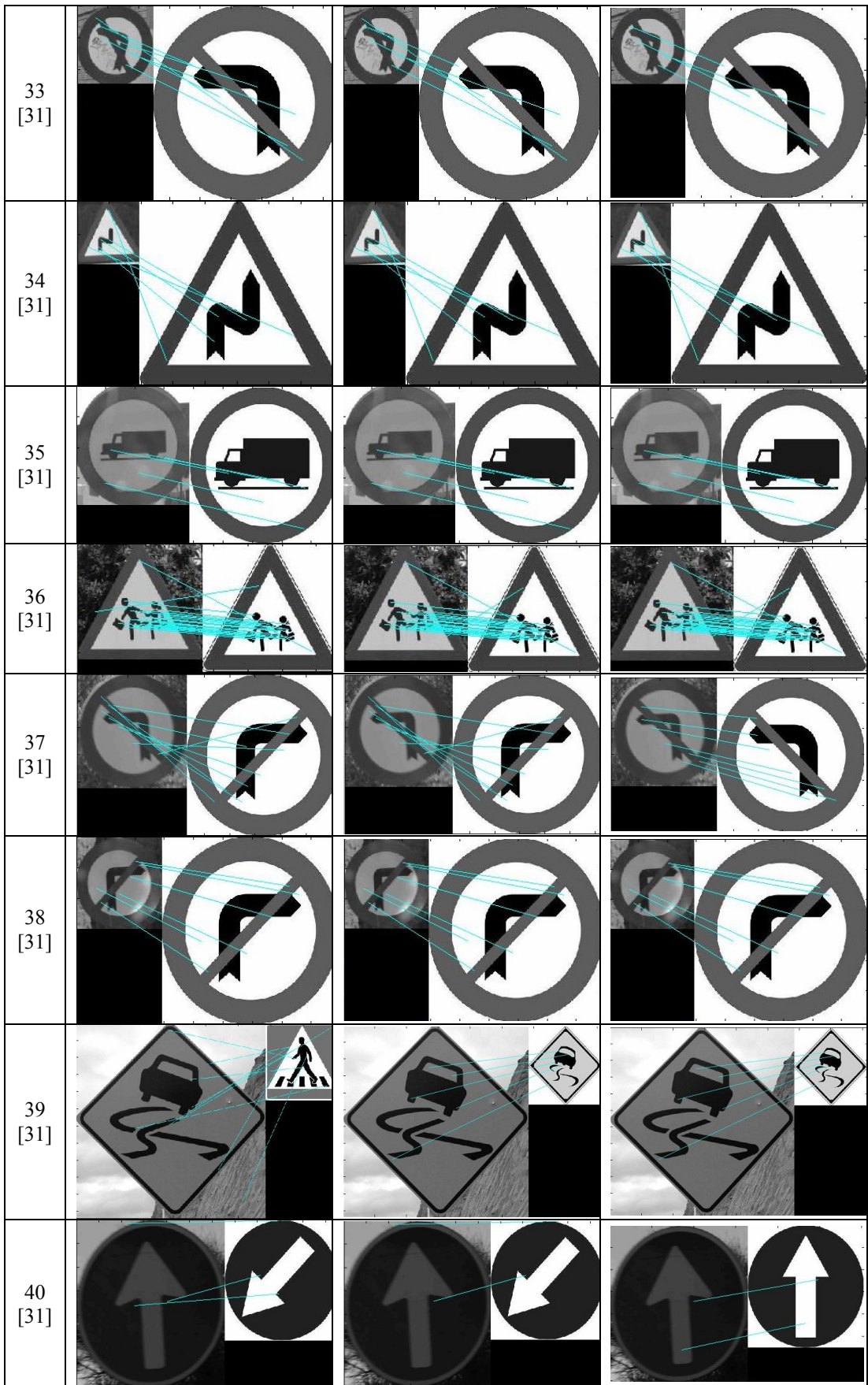
**Table B.1.** Recognition Results of Extracted Traffic Sign Images

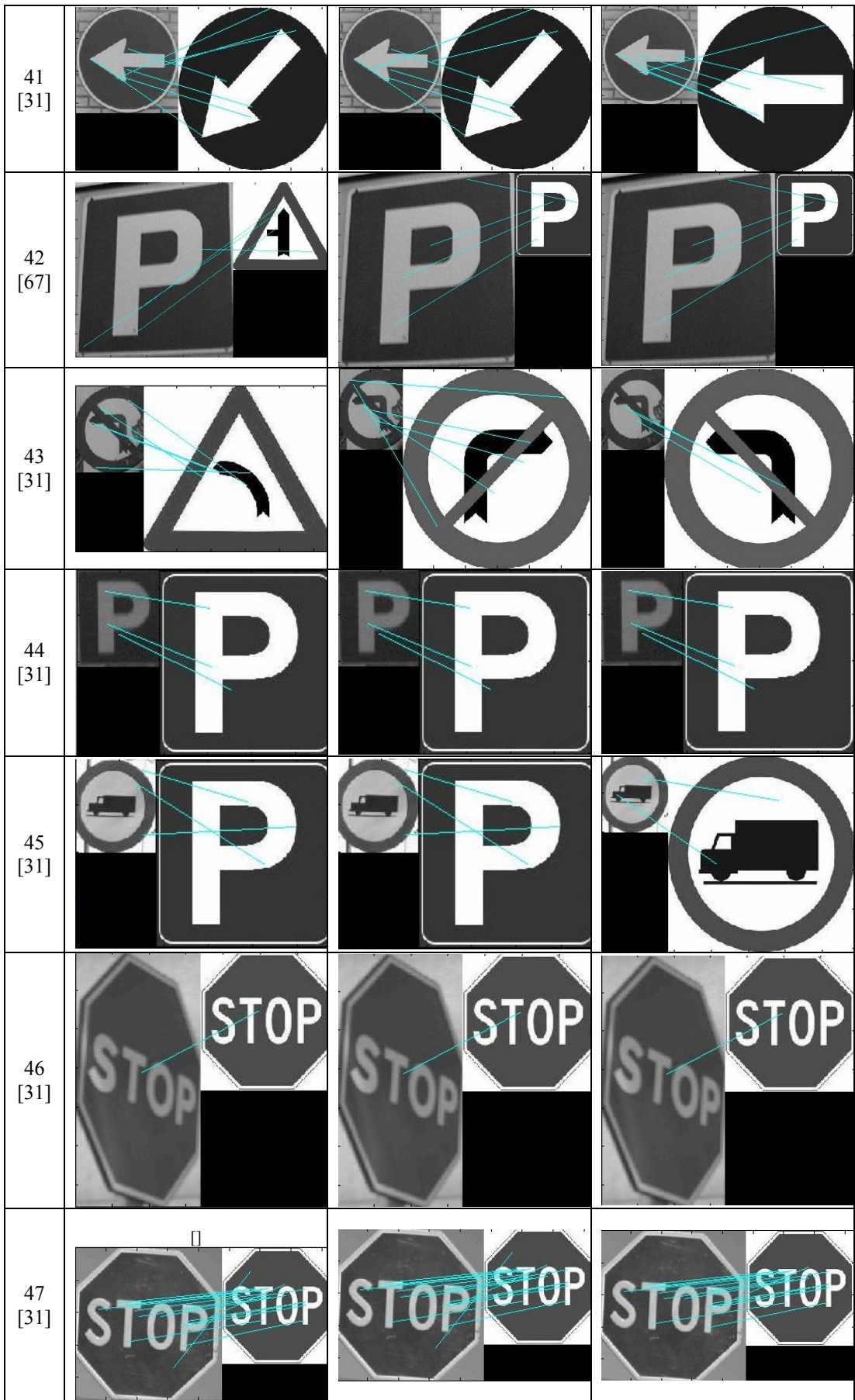
No	First Method	Second Method	Third Method
1 [39]			
2 [39]			
3 [37]			
4 [39]			
5 [36]			
6 [39]			
7 [39]			

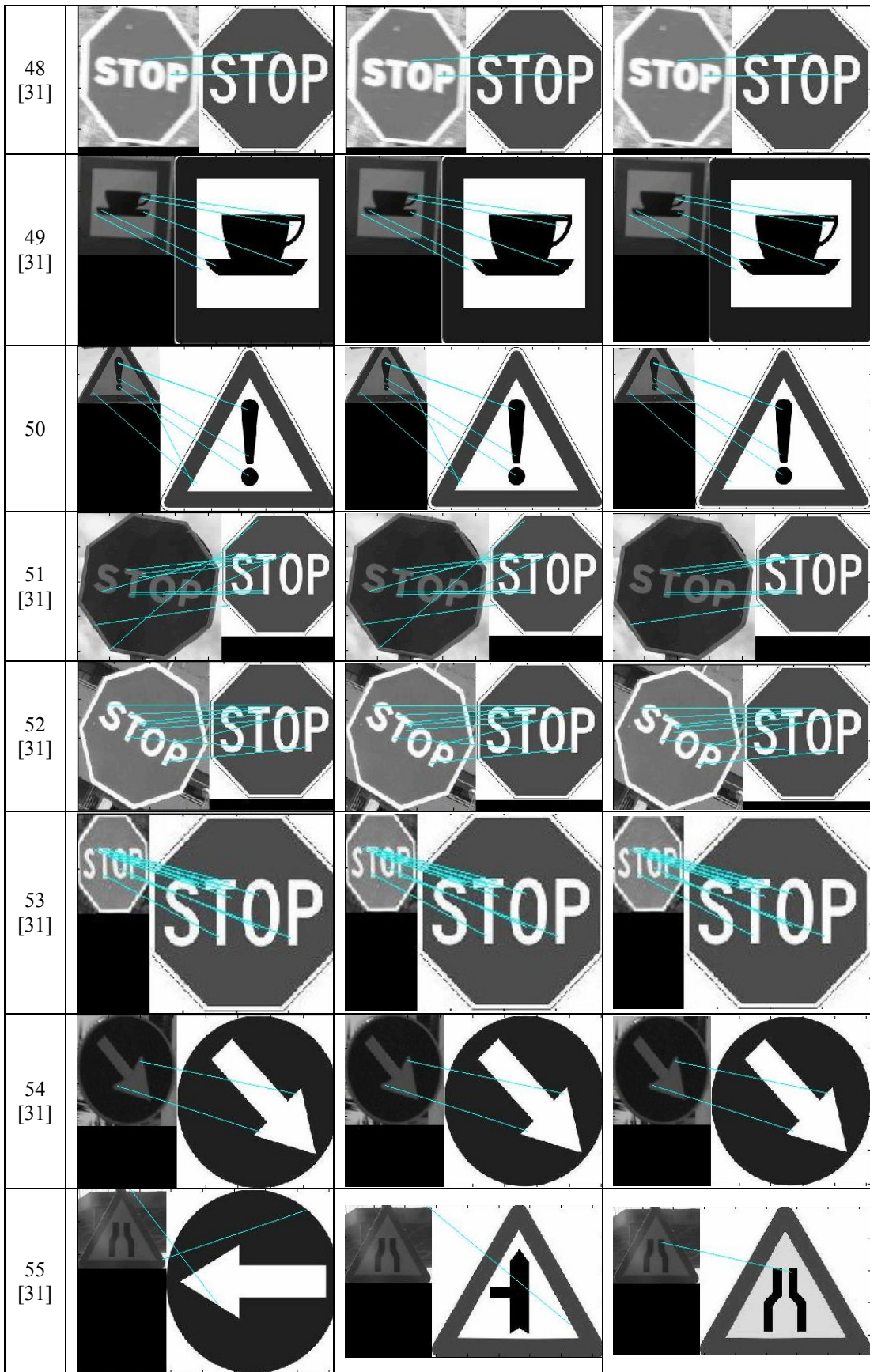


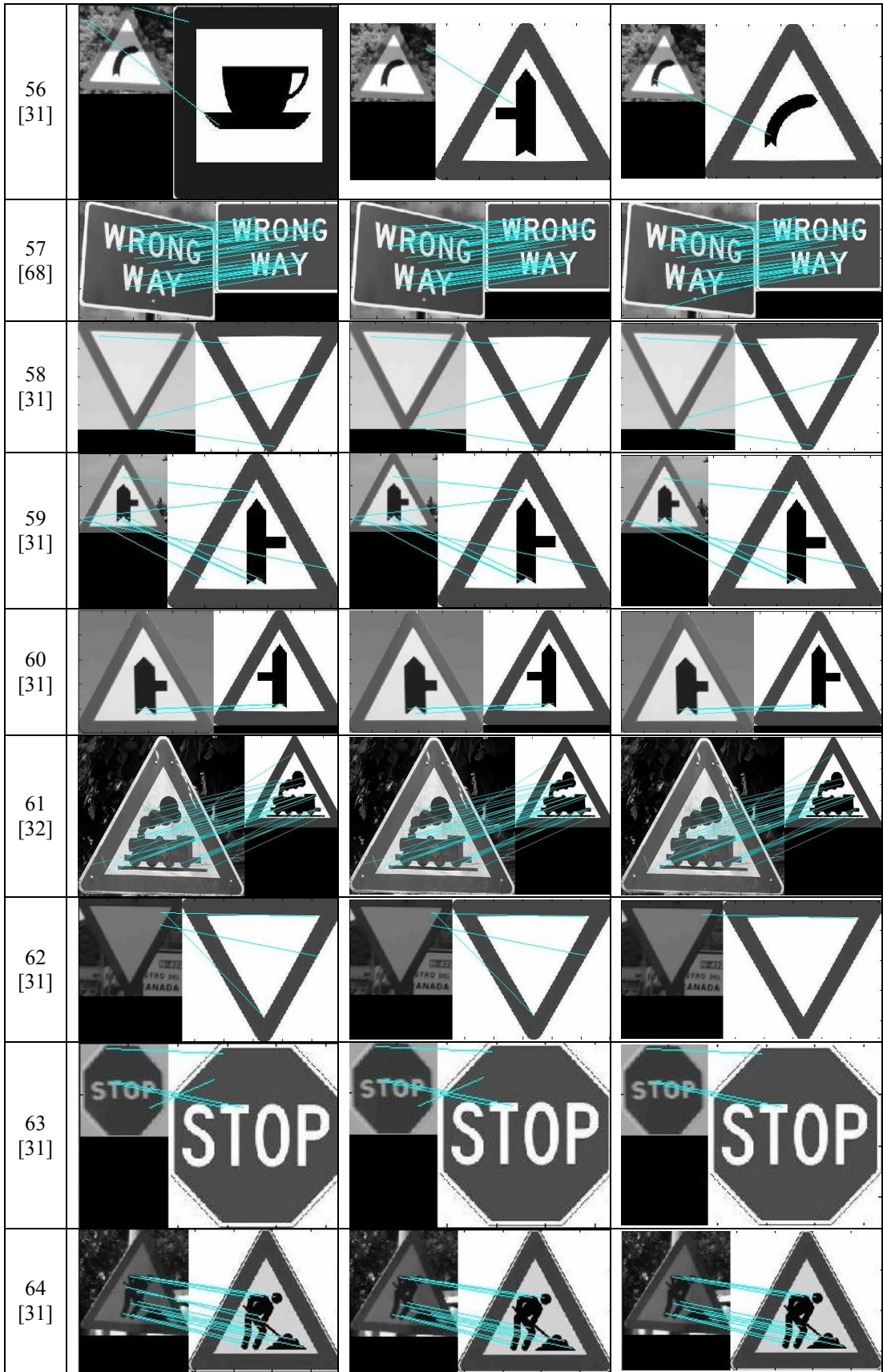


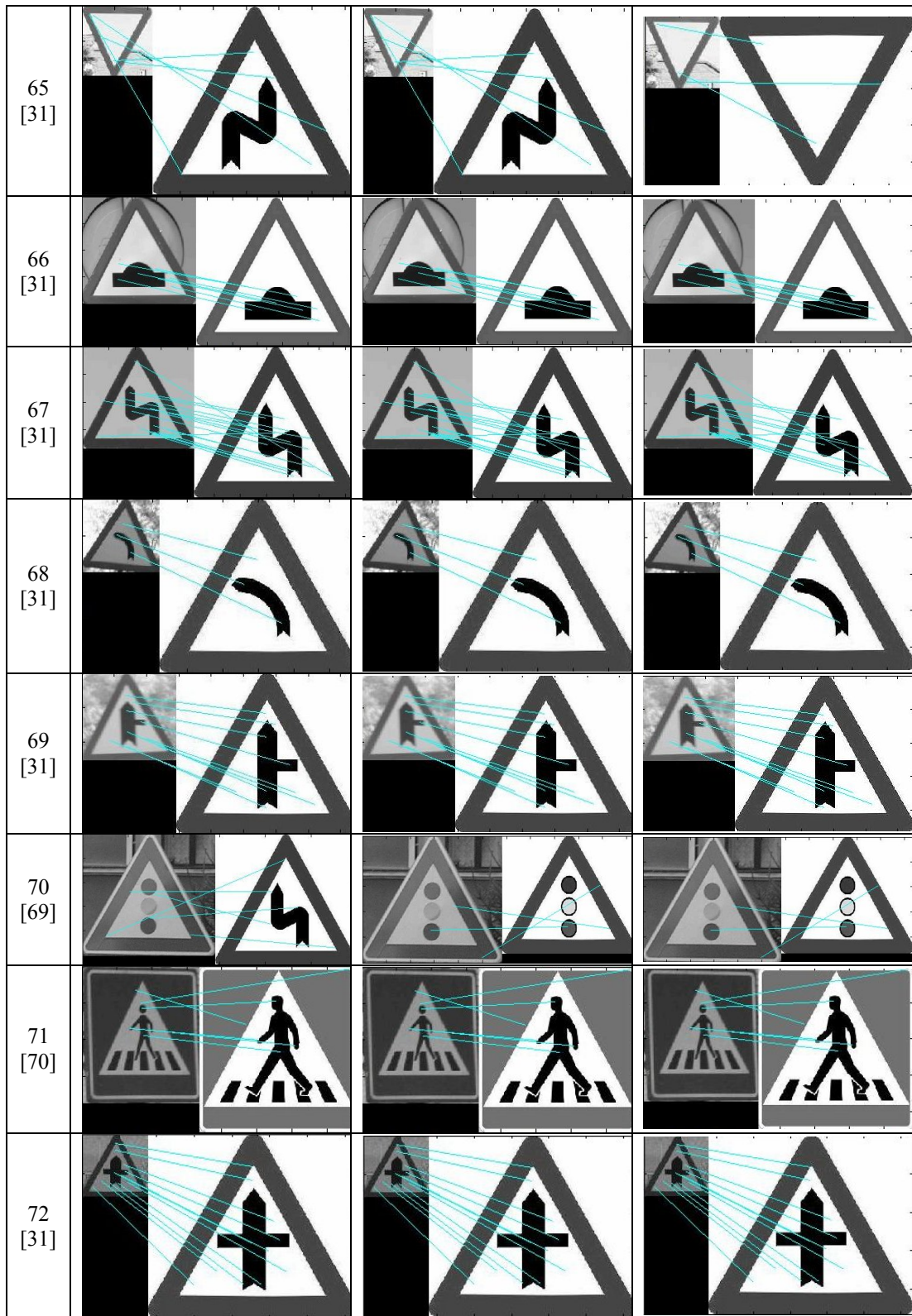


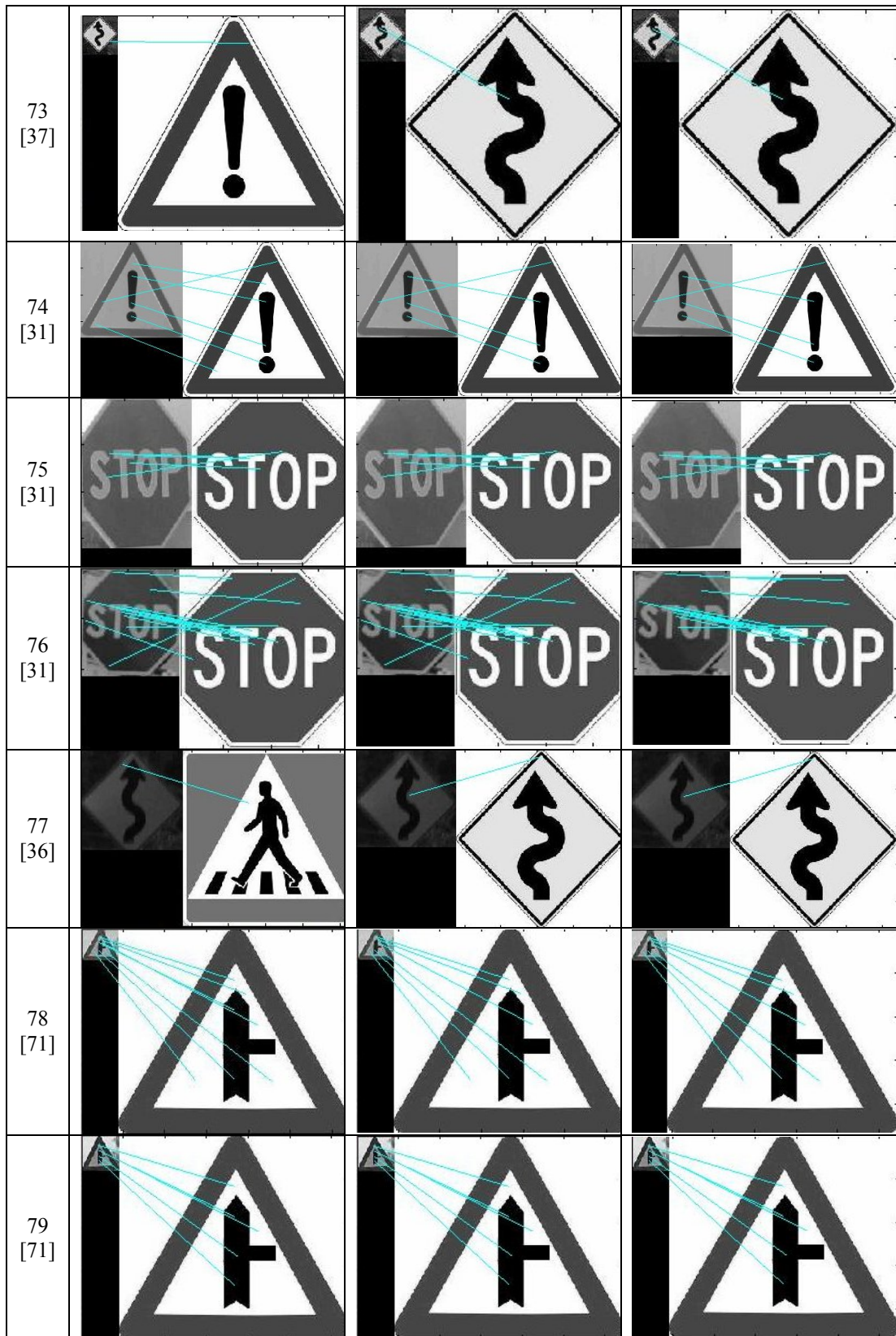


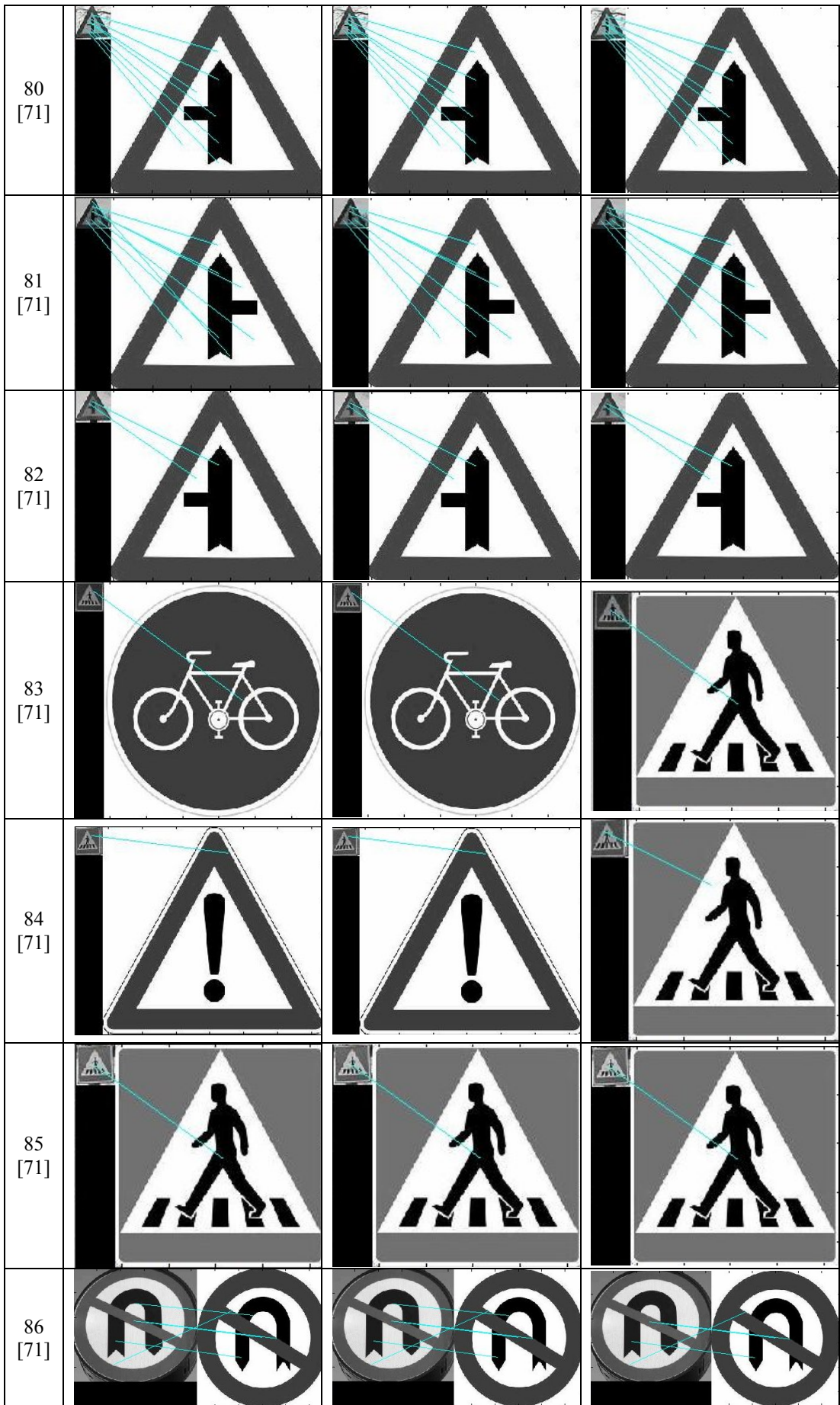


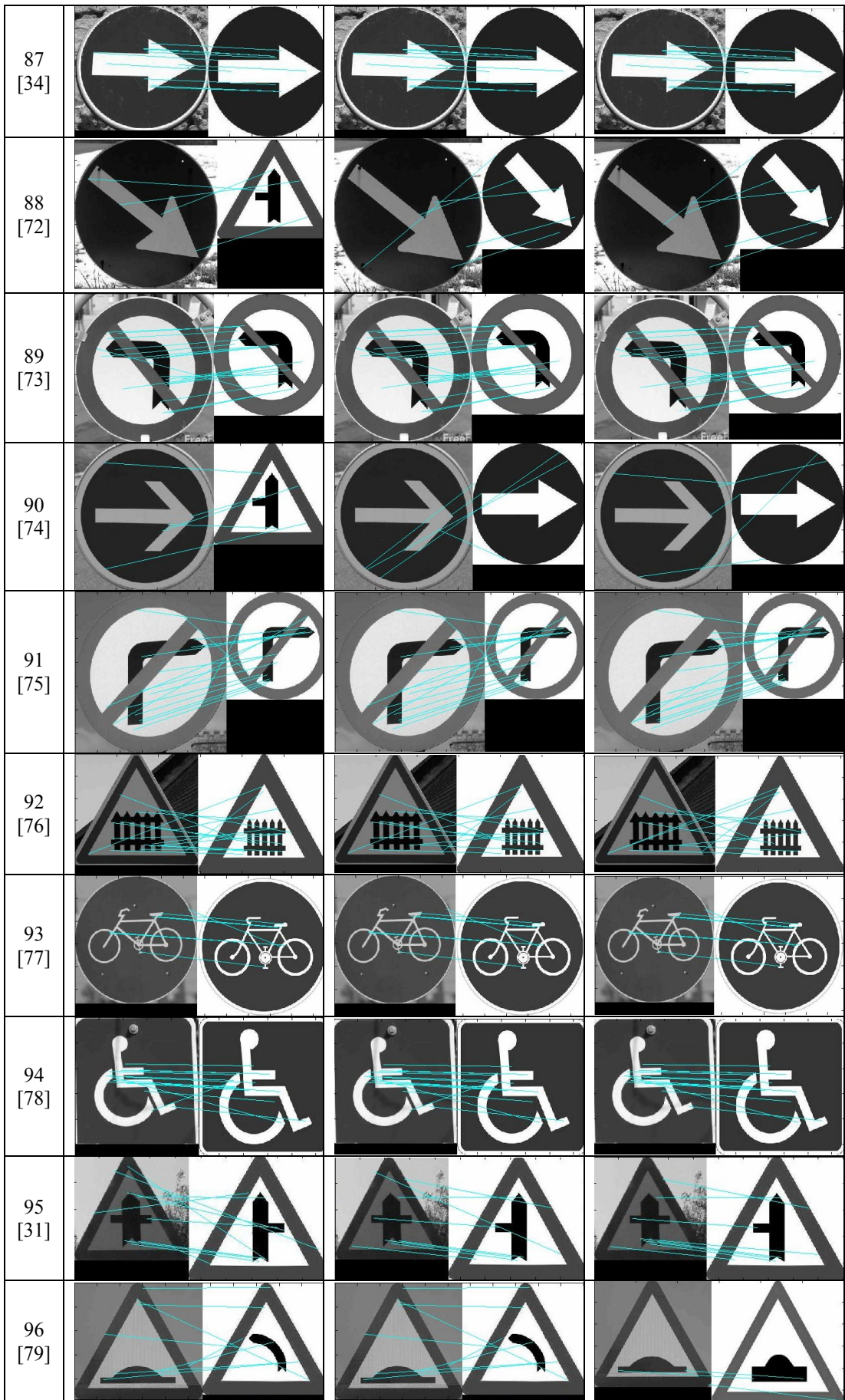


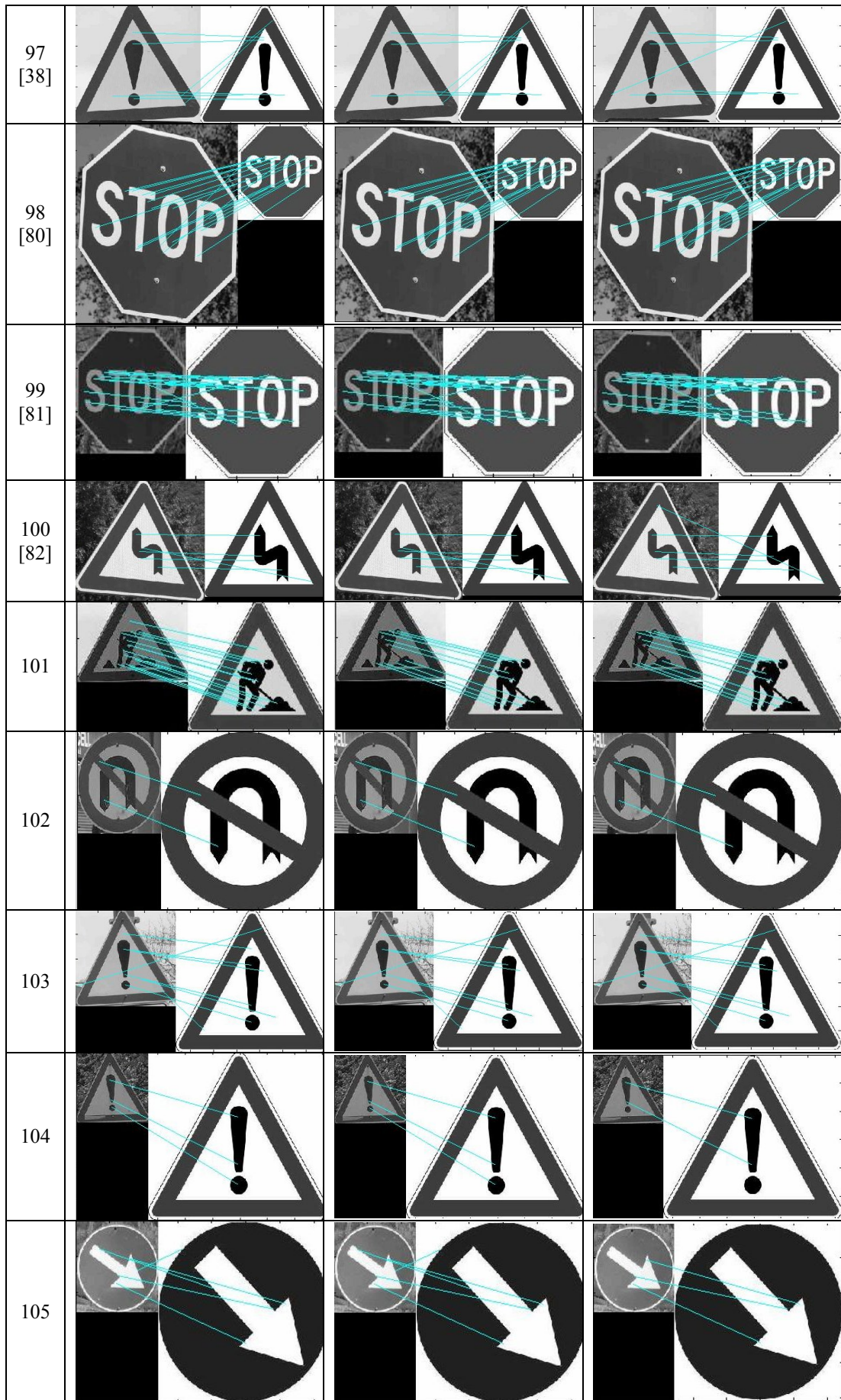


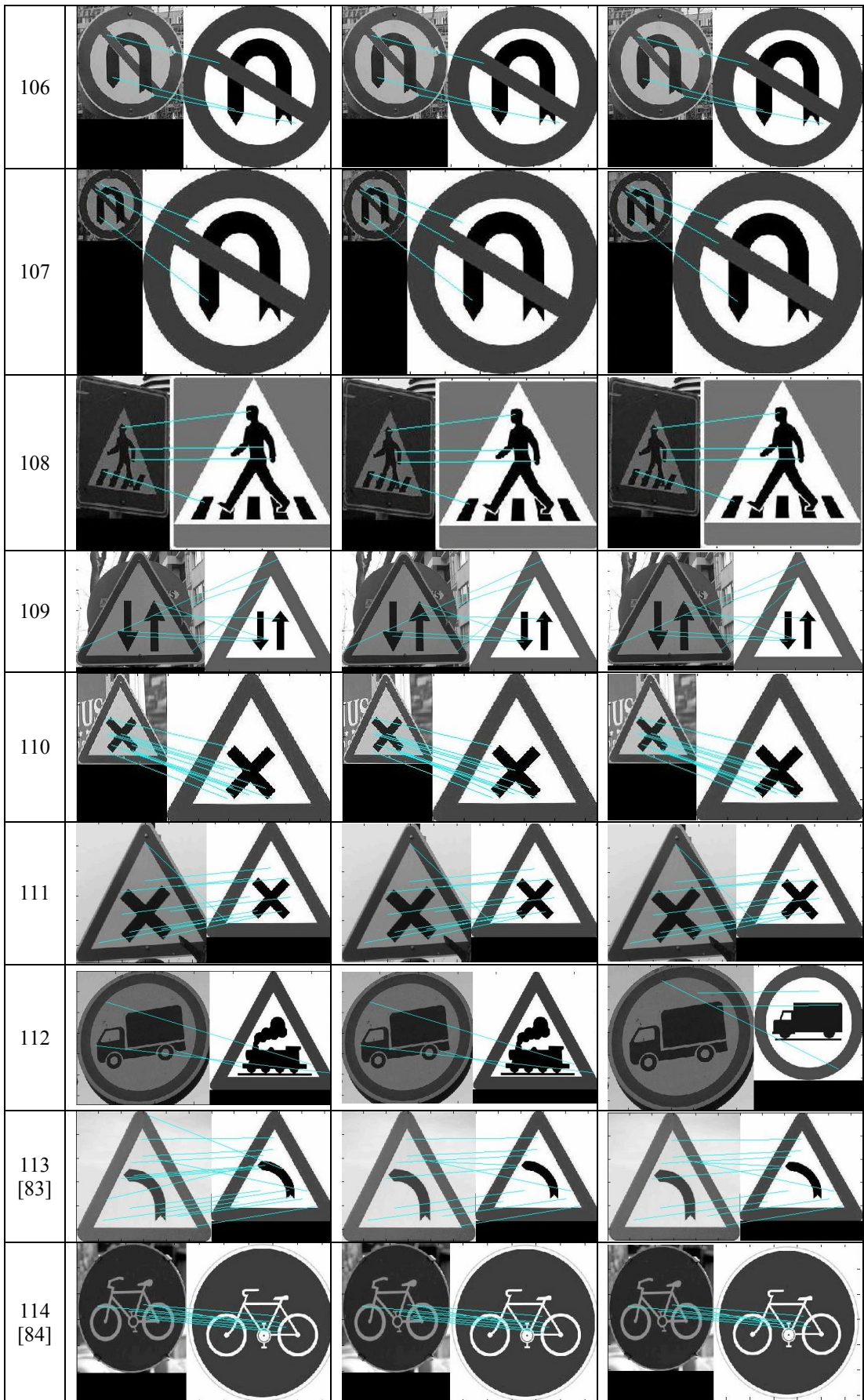


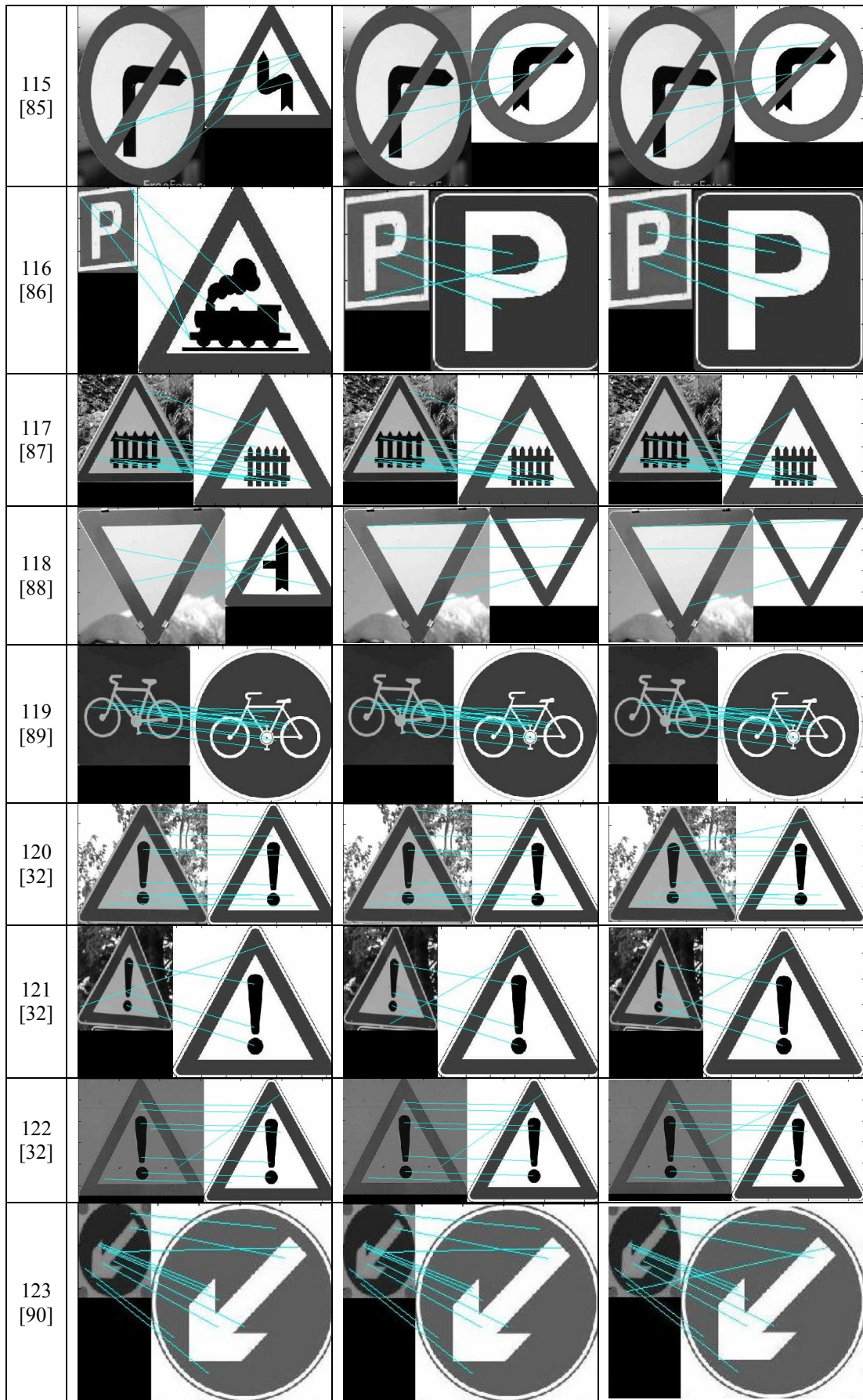


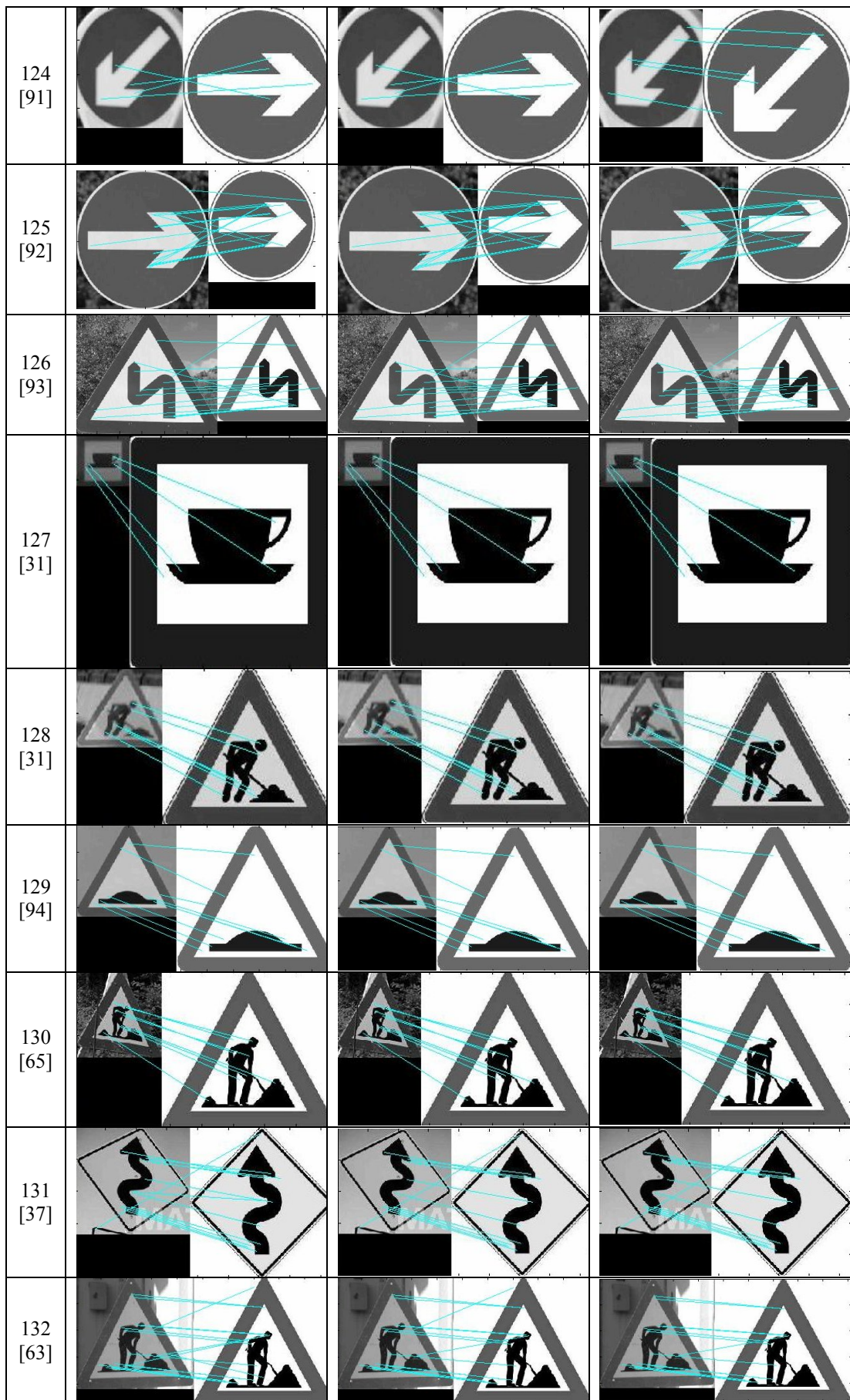


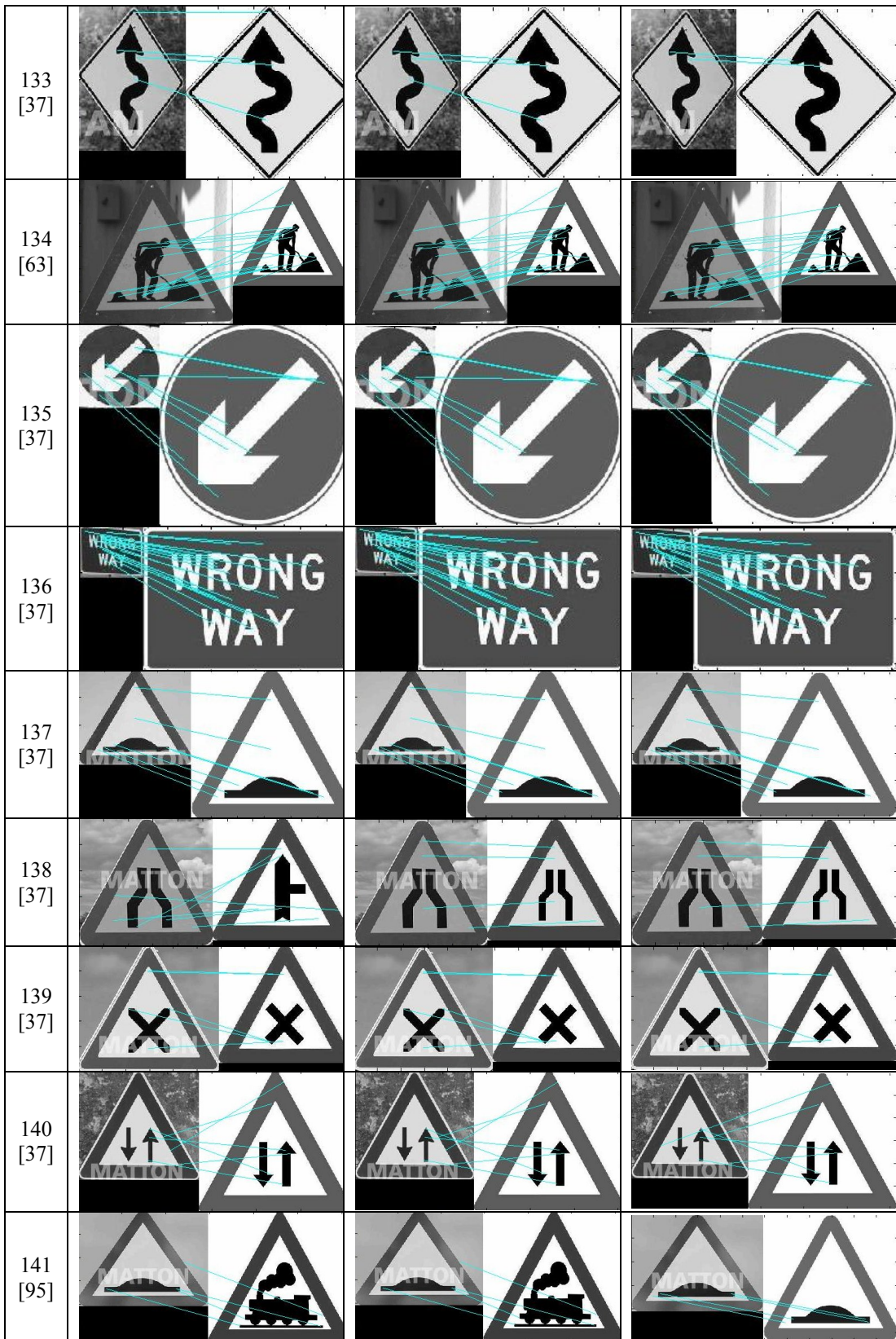


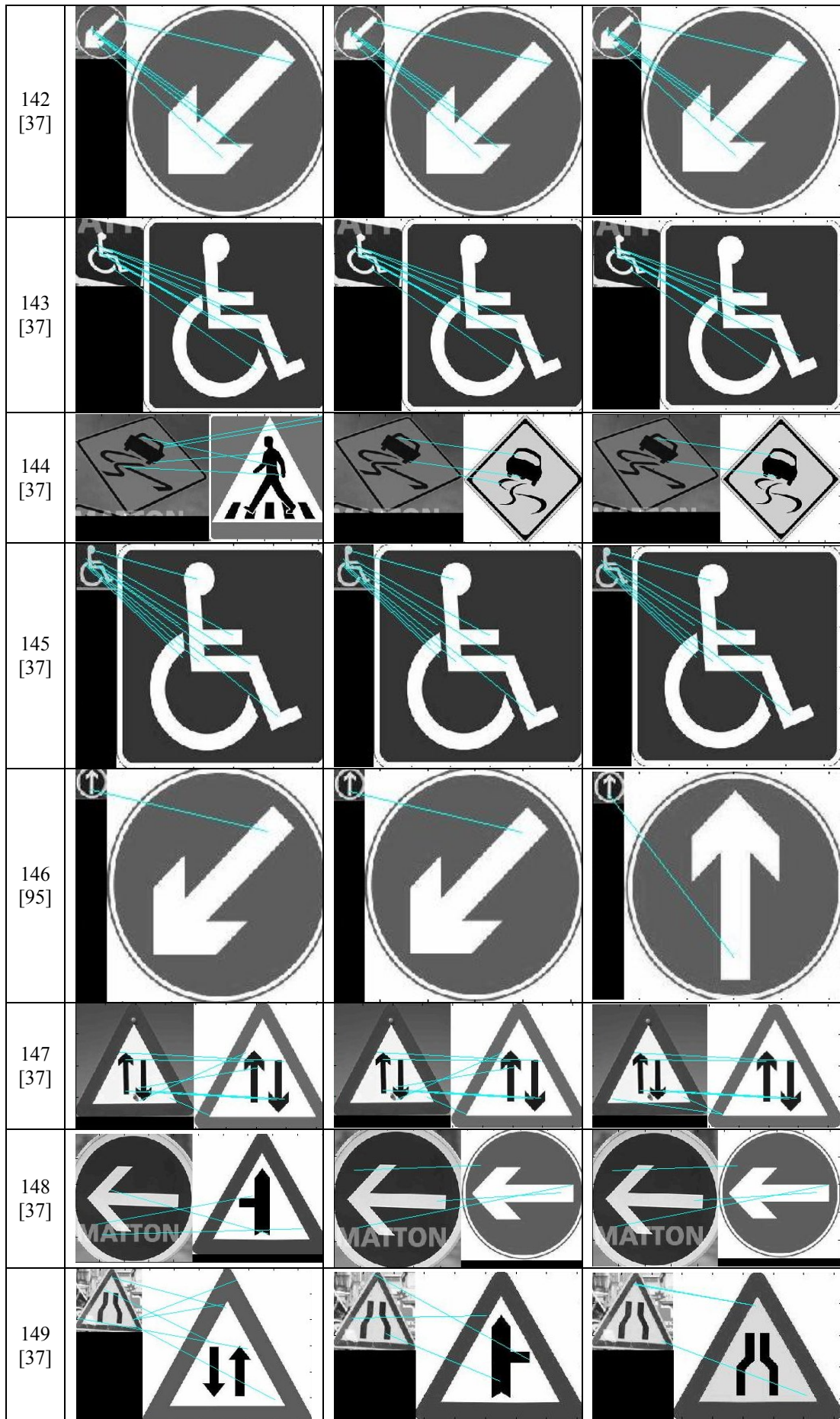


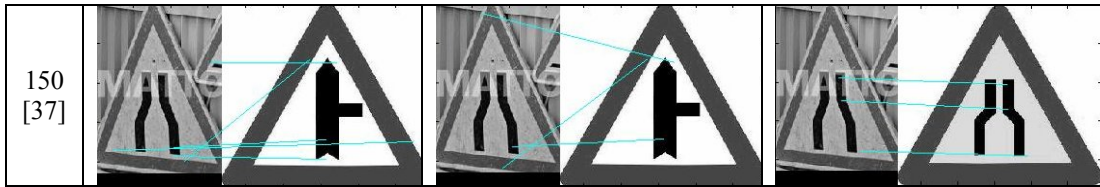






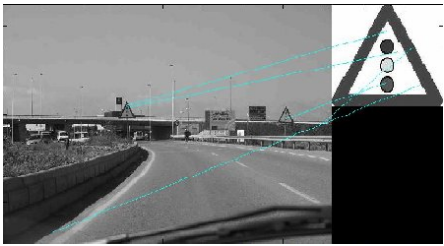
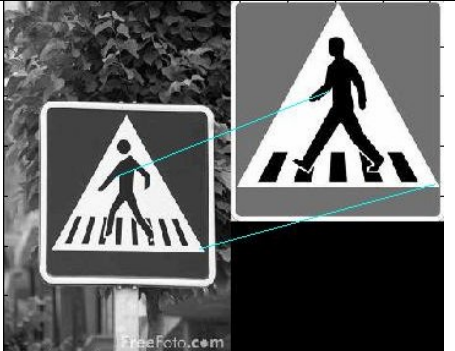

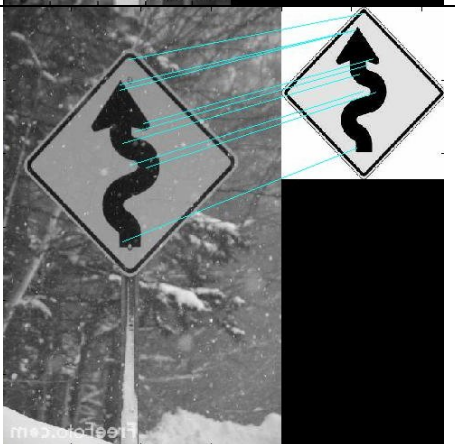
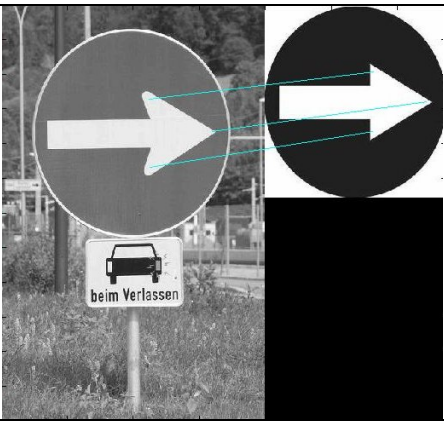
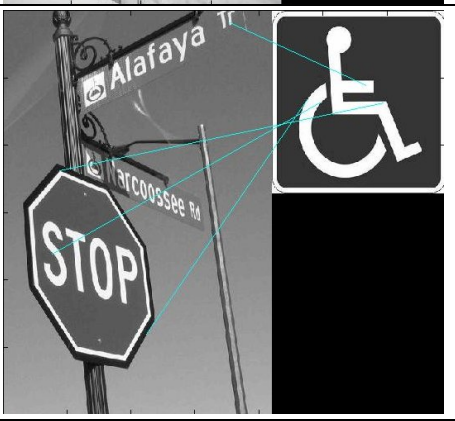
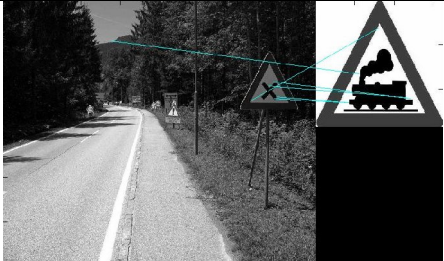
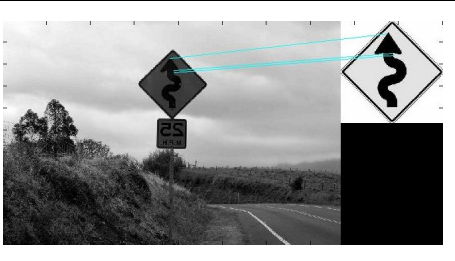


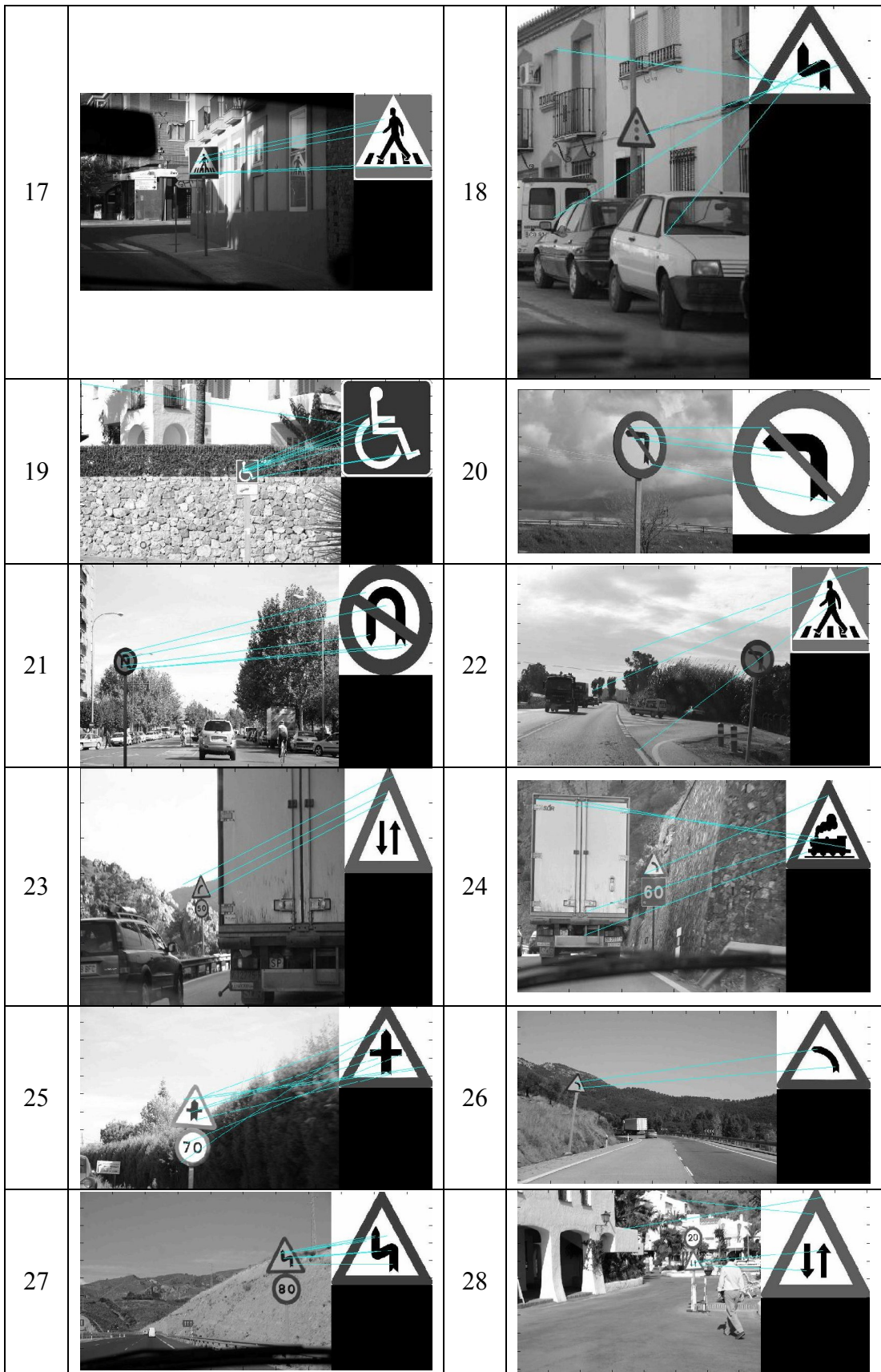




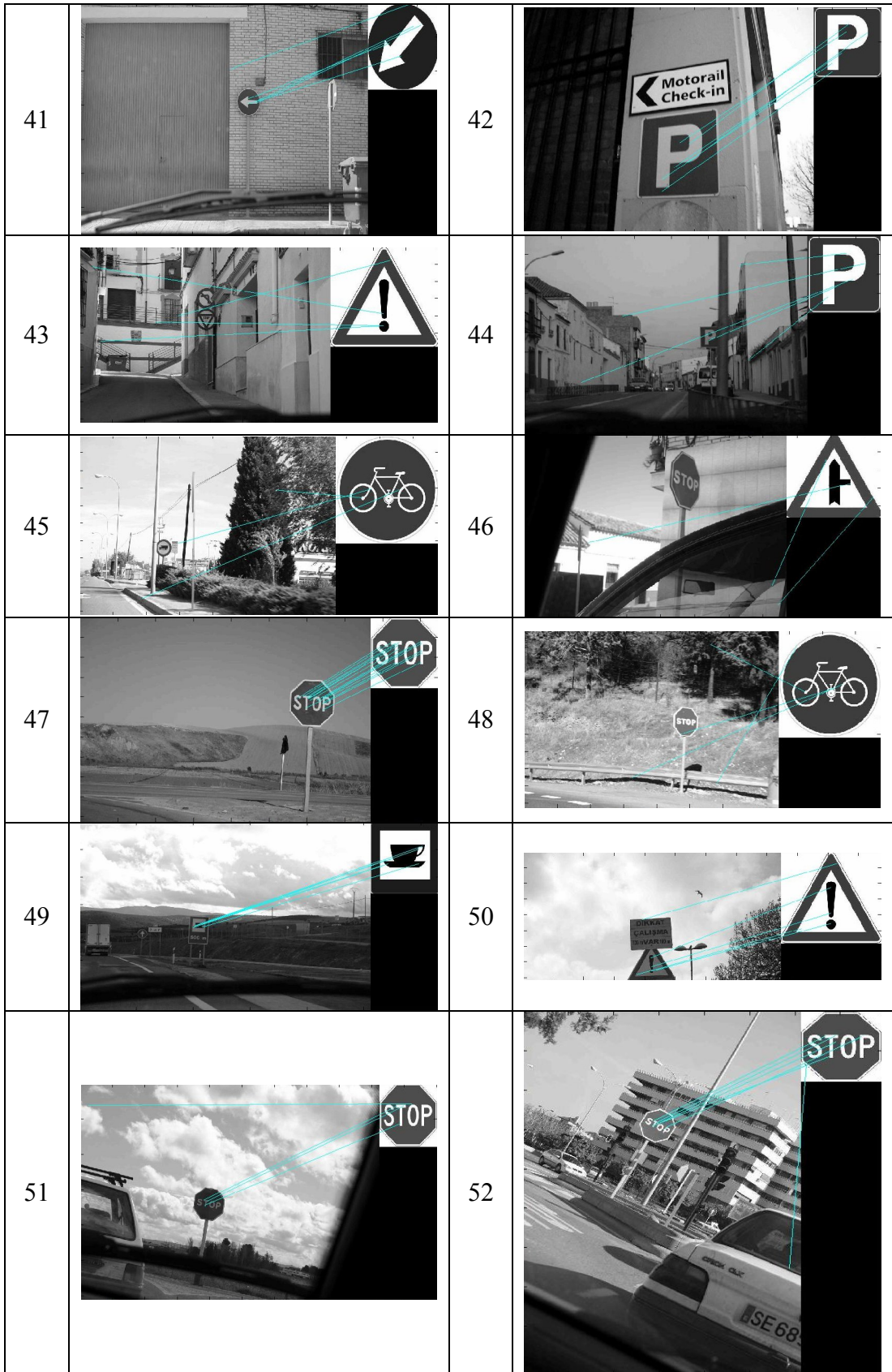
**Table B.2.** Recognition Results of Test Images without Extracting Traffic Sign Regions

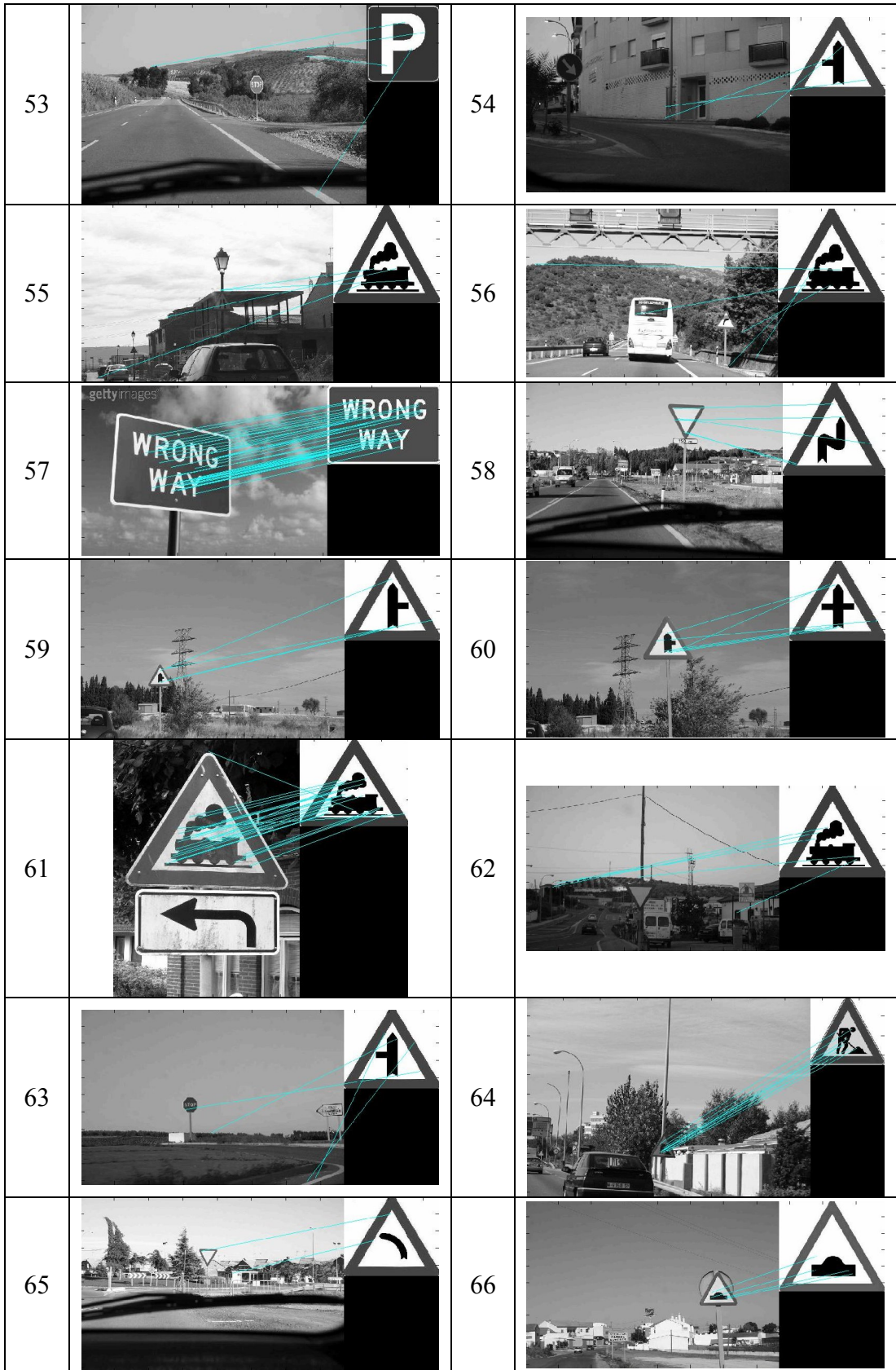
1		2	
3		4	
5		6	
7		8	


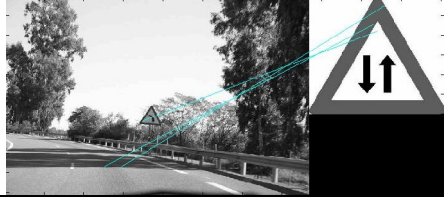

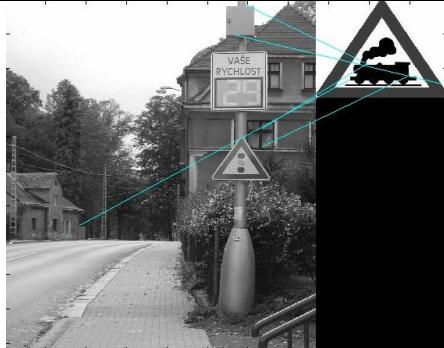
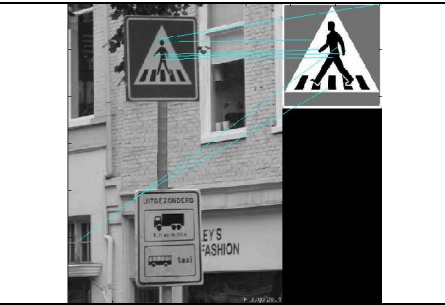

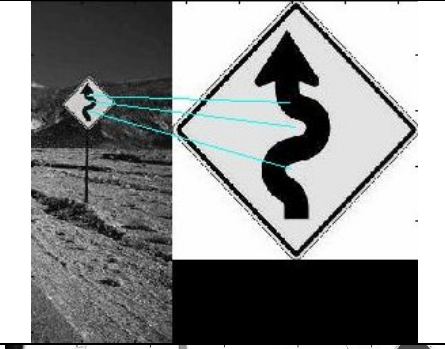
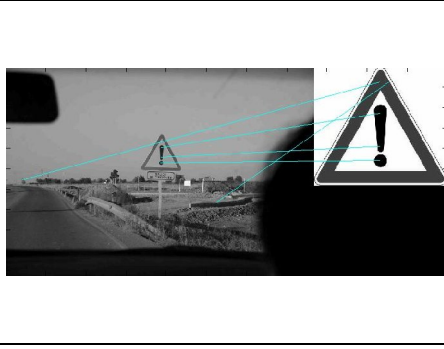

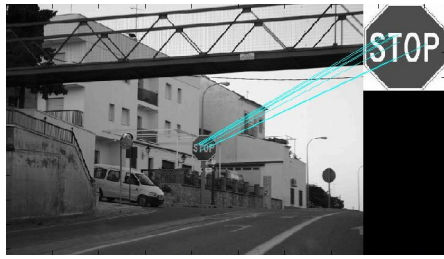
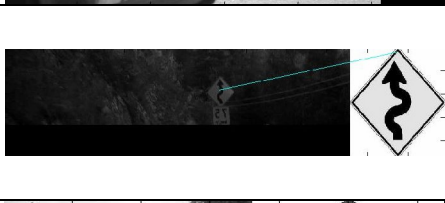
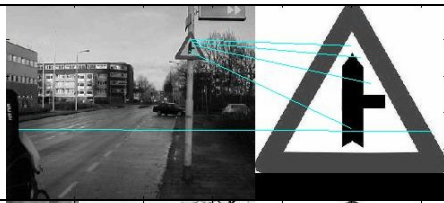
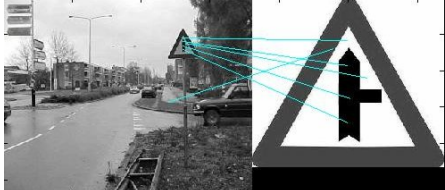
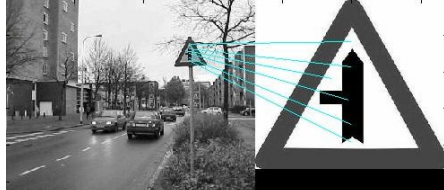
9		10	
11		12	
13		14	
15		16	

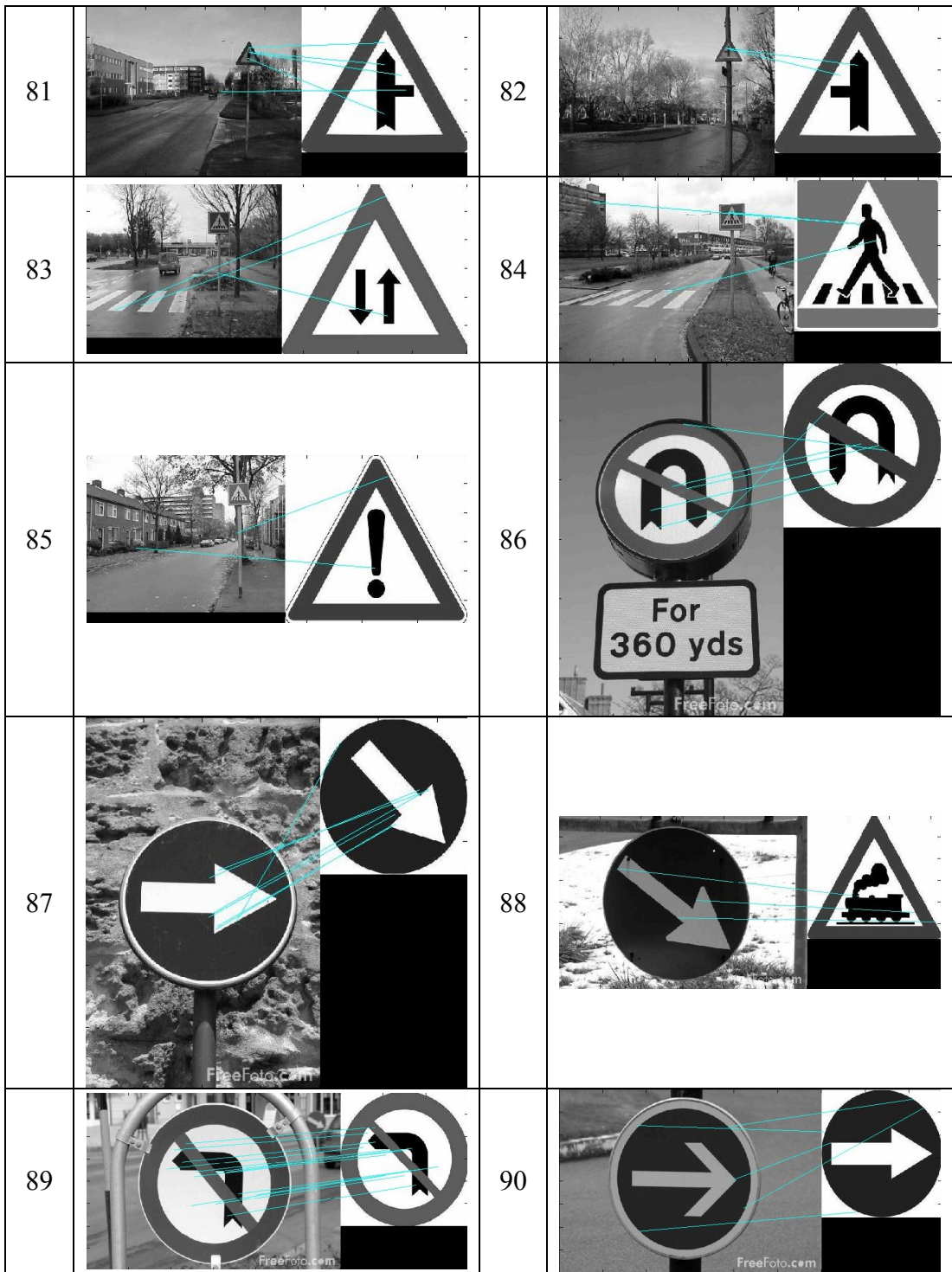


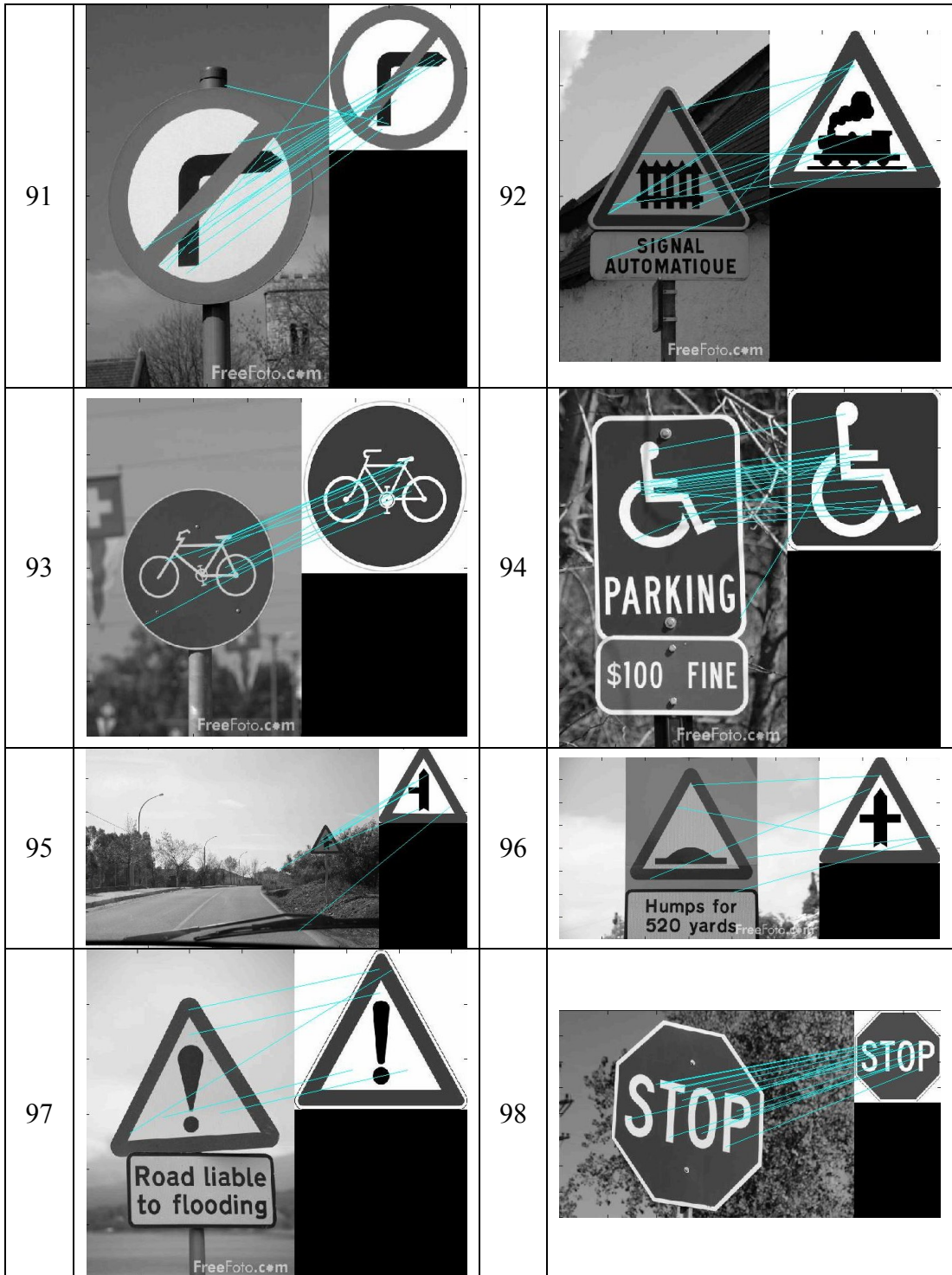


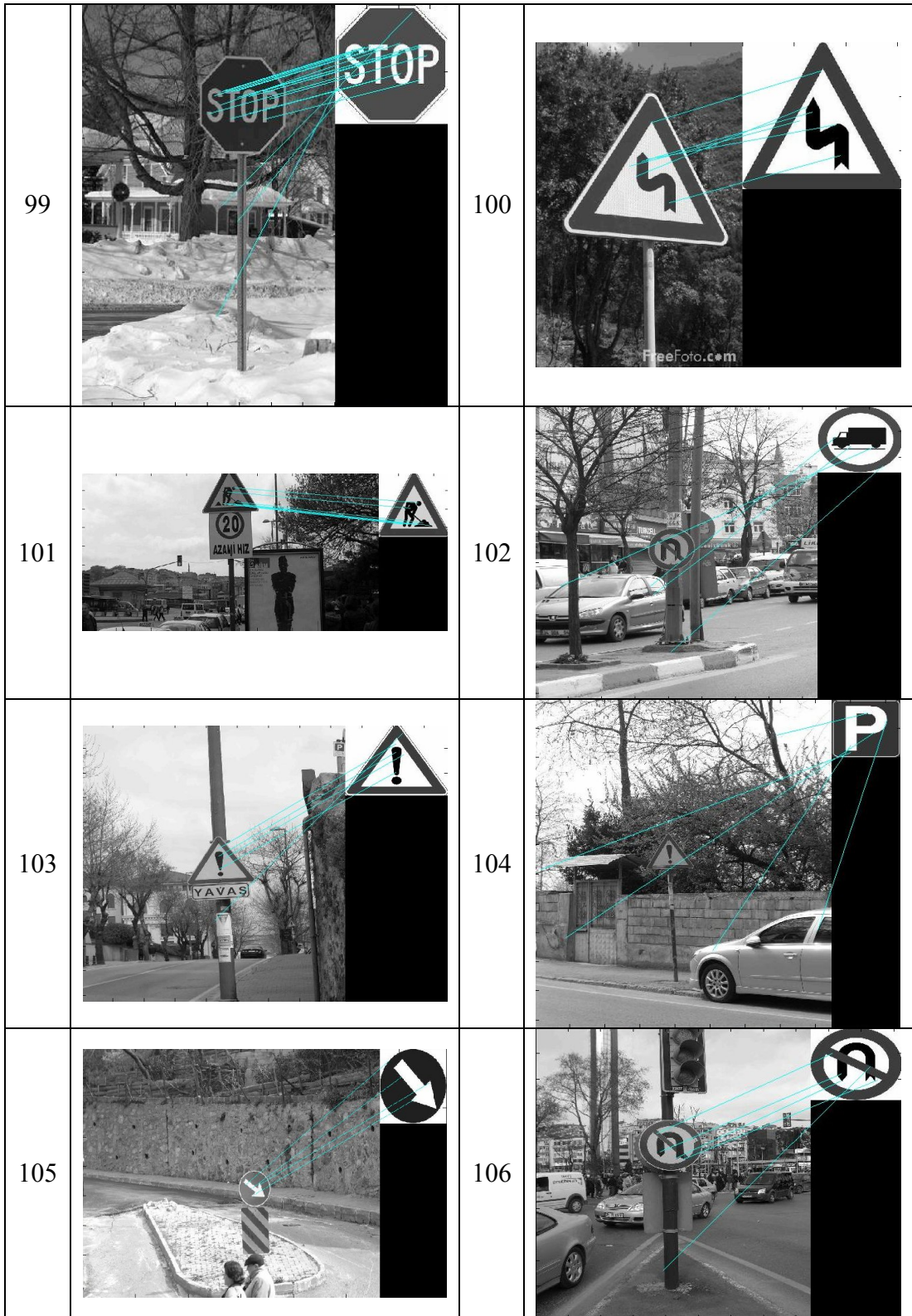


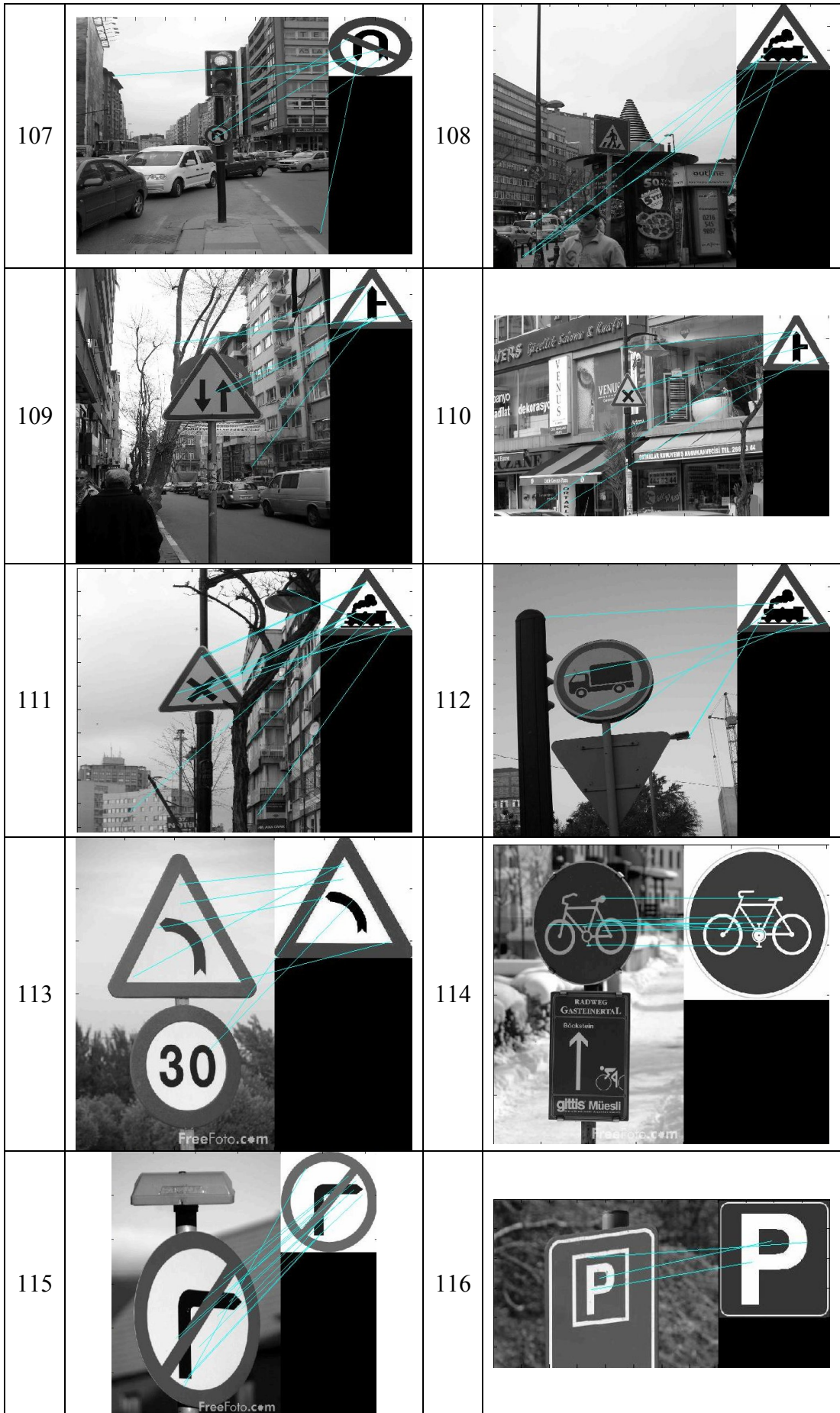


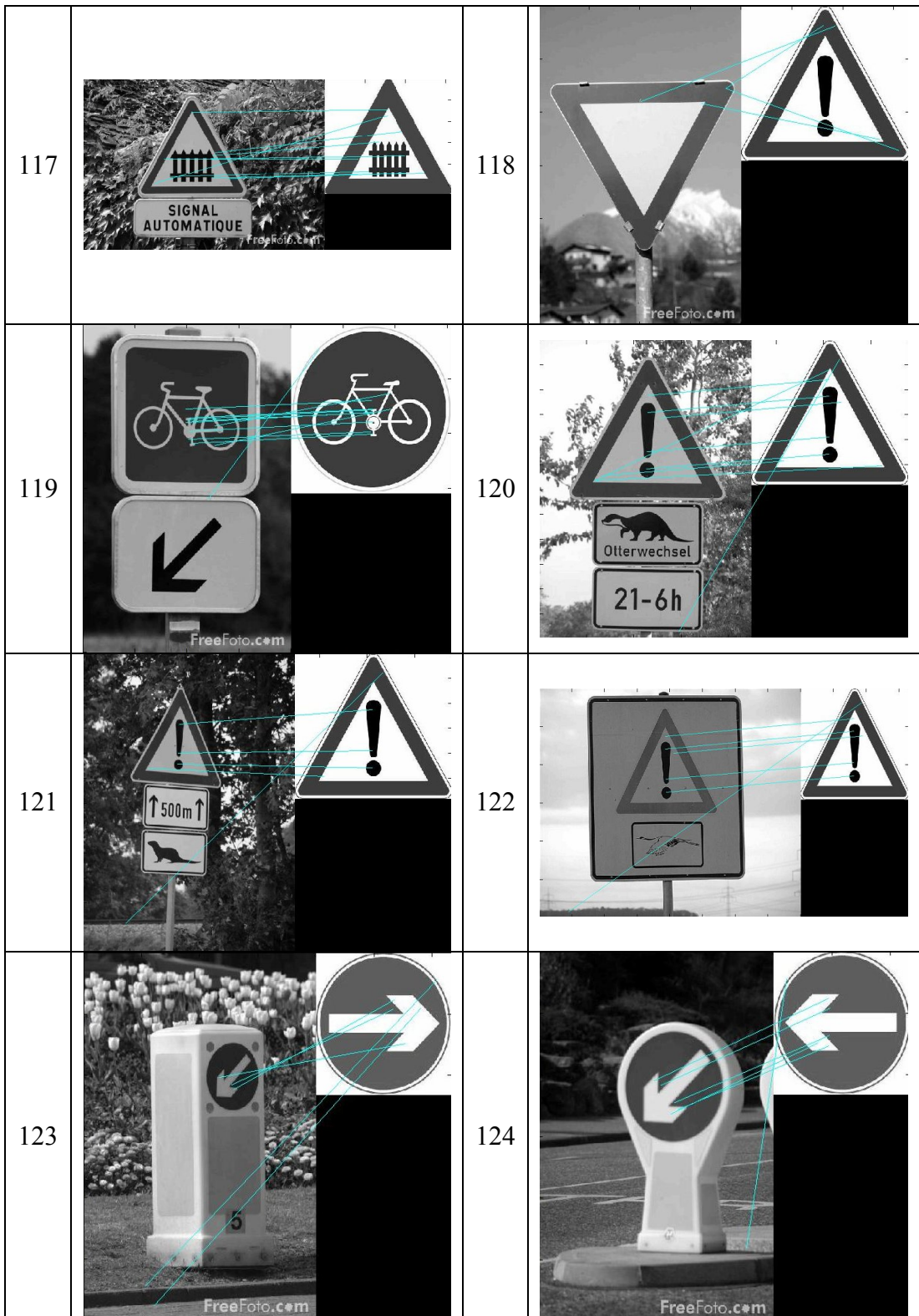
67		68	
69		70	
71		72	
73		74	
75		76	
77		78	
79		80	

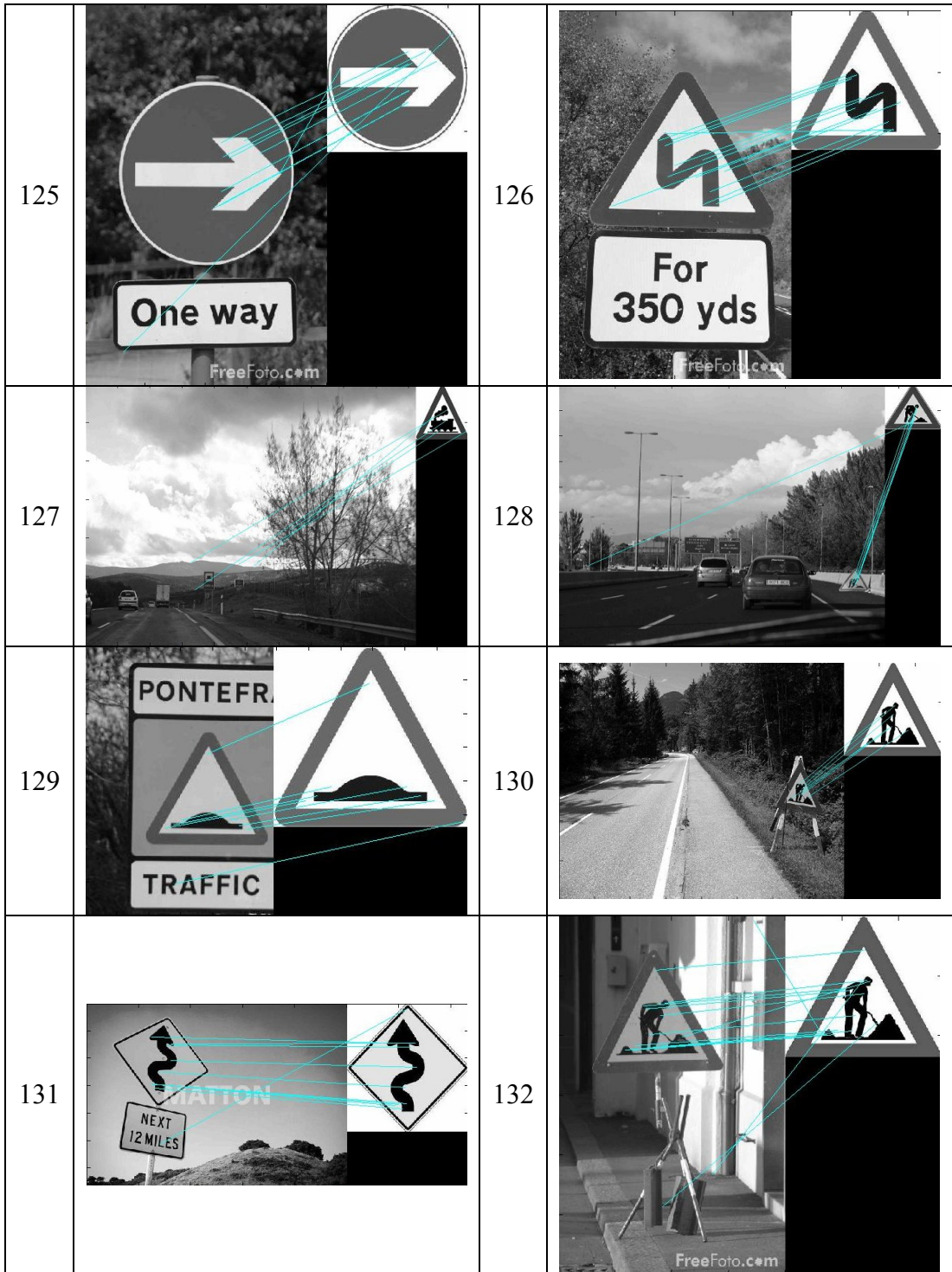


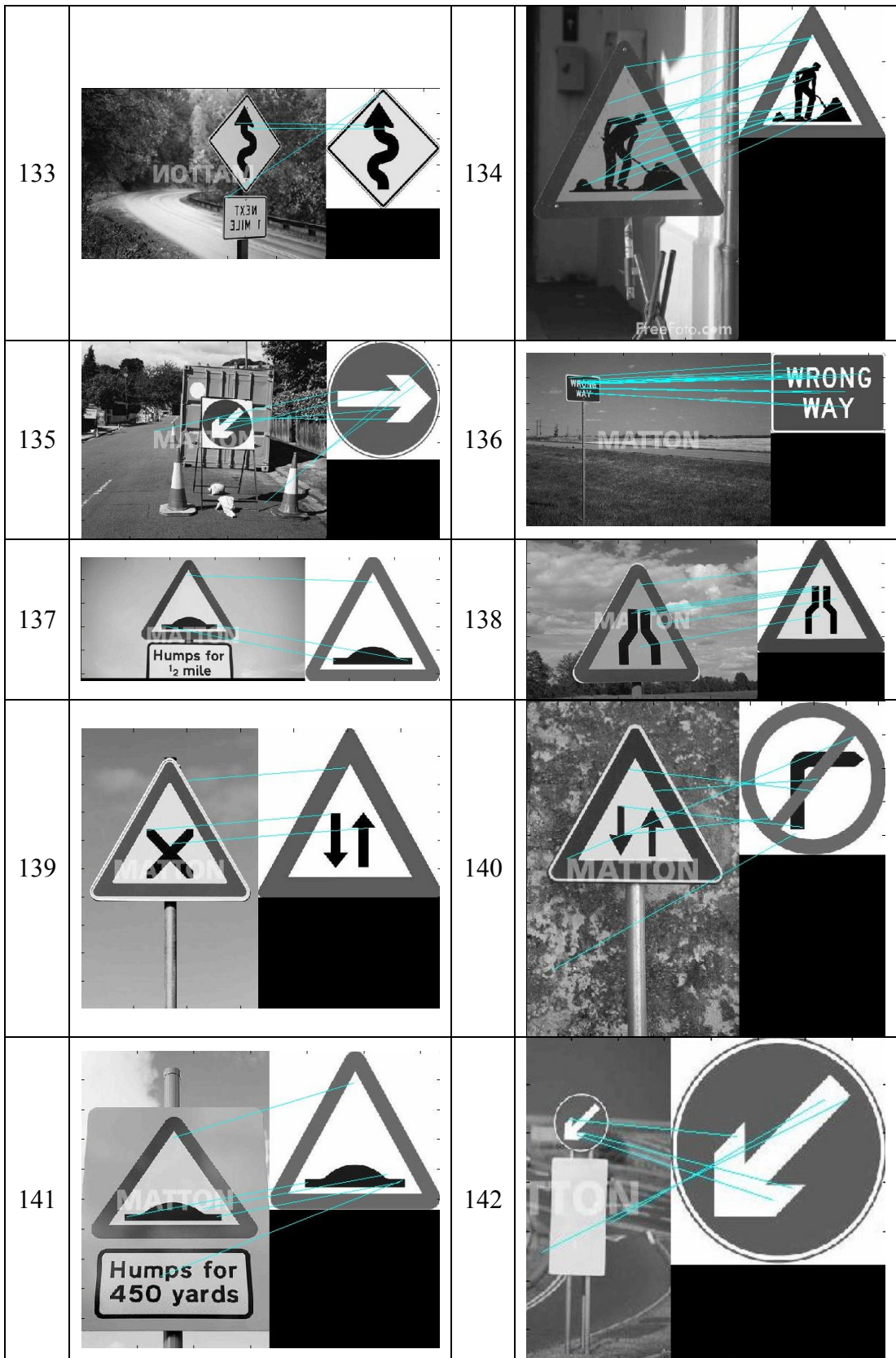












143		144	
145		146	
147		148	
149		150	

## **BIOGRAPHY**

Merve Can Kuş was born in Çanakkale, Turkey in 1984. Raised in Çanakkale, she started her education at Çanakkale 18 Mart Elementary School. Then she earned scholarship to carry on her education at Bursa Private İlkbahar Middle School. After successfully completing it, she was accepted to Izmir Private Malhun Hatun Science High School with scholarship. While continuing the high school in 2001-2002, she participated in TUBITAK Chemistry Olimpiads organized by BAYG where she became the second of Aegean Region, and she earned chance to participate in summer and winter instructional courses of TUBITAK, and she was finally selected to be one of the 8 members of Turkish Olimpiad Team's essence. Upon completing high school, she made a huge step forward in her educational career by being accepted to Istanbul Technical University where she studied Computer Engineering for four years. In these years, she earned mansion with her colleagues in the EMO Project Awards in 2006 with their graduation projects "Interactive Nature-Inspired Heuristics for Automatic Facial Composite Generation". She graduated with the highest score amongst all students in the faculty, besides won Siemens Excellence Award and after obtaining a B.Sc. degree in 2006, she carried on to study at the same university; this time she attended the M.Sc. Program in Computer Engineering. After completing her thesis and upon obtaining the M.Sc. degree, it is her wish to become highly qualified and successful in her professional work career.