# A MODEL-BASED ROAD SIGN RECOGNITION SYSTEM

Yves Bérubé Lauzière

Departement of Electrical and Computer Engineering
McGill University, Montréal

February 2002

A Thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

# Abstract

A road sign recognition system poses a real challenge for machine vision. It must recognize a wide variety of road signs under considerable variations in illumination and imaging geometry - all in real-time. This thesis presents a modular road sign recognition system relying on modelling for both detection and recognition. It divides into three main stages of processing. The first, concerned with detection, exploits the specific colors of road signs. The color constancy problem caused by the daylight illumination variations is addressed directly with a physics-based model supplemented by a calibration stage using real data. The second stage of processing, devoted to recognizing road signs in regions of interest found in the detection phase, involves a database containing more than 400 road signs arranged in a tree structure, and uses a novel correlation-based template matching technique relying on a bitwise encoding that accounts for both color labels and affine variations in the image formation process, and which also allows to build templates that are able to represent classes of objects. The content of the database used by the recognition algorithm is generated in a deterministic and automated manner by way of geometrical modelling of the image formation process starting with only model images of the road signs to be recognized. The recognition algorithm exploits color as a first logical classification step to direct the search for a road sign in the database, with the later finer steps being driven by correlation scores obtained from template matching. At the third stage of processing, a scene understanding module exploits constraints on the position of road signs along with the spatial relationships they must have in certain cases to other road signs in the image to filter out false positives. During processing, the system incorporates top-down mechanisms that use data fed back by partial recognitions, which allow to progressively gain more information about the identity of a potential road sign, thereby increasing detection and recognition robustness. Experimental results are presented which demonstrate the high overall performance of the system in a real task environment.

# Résumé

Un système de reconnaissance de panneaux routiers représente un réel défi pour la vision numérique. Une grande variété de panneaux doit être reconnue en temps réel sous des conditions changeantes d'illumination et de prise d'images. Cette thèse présente un système modulaire de reconnaissance de panneaux routiers utilisant de la modélisation pour la détection et la reconnaissance. Le traitement se divise en trois grandes parties. La première, reliée à la détection, exploite les couleurs spécifiques des panneaux routiers. Le problème de constance des couleurs causé par les variations de l'illumination du jour est abordé de façon directe à l'aide d'un modèle physique adjoint d'une étape de calibrage utilisant des données réelles. La seconde partie du traitement est chargée de reconnaître les panneaux dans les régions d'intérêt déterminées par l'étape de détection. Elle implique une base de données sous forme d'arbre contenant plus de 400 panneaux et utilise une nouvelle technique de correspondance par gabarits basée sur un calcul de correlation et exploitant un encodage de bits qui tient compte de la couleur et des variations affines dans le processus de formation des images et qui permet en outre de construire des gabarits pouvant représenter des classes d'objets. Le contenu de la base de données utilisé par l'algorithme de reconnaissance est généré de façon déterministe et automatique par modélisation géométrique du processus de formation des images en n'utilisant au départ que des modèles des panneaux devant être reconnus. L'algorithme de reconnaissance exploite la couleur dans une première phase de classification pour orienter la recherche d'un panneau dans la base de données, les phases ultérieures s'appuient quant à elles sur le degré de correspondance mesuré par corrélation par rapport aux gabarits dans la base de données. Dans la troisième partie de traitement, un module d'interprétation de scènes routières exploite des contraintes sur la position des panneaux ainsi que les relations spatiales qu'ils doivent avoir dans certains cas avec d'autres panneaux afin d'éliminer les faux positifs. Pendant le traitement, le système

utilise des mécanismes de haut niveau exploitant des données fournies par des reconnaissances partielles, ce qui permet de progressivement gagner de l'information sur l'identité d'un panneau et d'ainsi rendre plus robustes la détection et la reconnaissance. Des résultats expérimentaux sont présentés, démontrant les performances élevées du système dans un environnement réel de fonctionnement.

# Acknowledgements

First, I would like to express my gratitude to my two supervisors. To Professor Frank Ferrie for his clear thinking and insight about my work, and his so many good advices during the writing of papers. To Dr. Denis Gingras (now Professor at the Université de Sherbrooke) for having happily welcomed me at INO (Institut National d'Optique) where this research was carried, and who initiated this work on road sign recognition. The confidence they both had in me all along was highly appreciated.

I would like to thank INO for having provided me a highly stimulating environment, with state of the art equipment and computing facilities. Special thanks to the many "stagiaires", students, and colleagues at INO for their patience in holding the chart and for accompanying me to take images on the road. I can't avoid mentioning their names here: Nathalie Blanchard, Louis Gosselin, Mario Marois, Pascale Parrein, Jérôme Comallonga, François Brunet, Stéphane Chatigny, Maxime Savard, Eric Harvey, Michel Doucet, and especially Alexis Lesage "dit Victor" whose training period I supervised during the 1997 summer. Thanks go also to Denis Boulanger for his wide knowledge of image processing techniques.

I would also like to thank the Ministère des Transports du Québec, especially Pierre Cauchon and Paul Bergeron for their cooperation.

Before I went to INO to pursue my work, the early days of my Ph.D. were spent at McGill, which is a most enjoyable place to be, both intellectually and physically. Moreover, when I had to deal with the administrative staff, I have always felt that the student was lying at the center of interests. Such an appreciation of students is quite unique.

I also wish to thank Vincent Hayward for having given me a chance and good advices when I arrived at McGill, and Steven Zucker for having gotten me excited about computer vision.

I wish to acknowledge financial support from McGill University through its McGill Major Fellowships Program, in particular the J.W. McConnell Foundation, and also support from NSERC, FCAR, and INO.

Last, but not least were the encouragements of my mother and above all of my second half Brigitte. Her day after day patience, understanding and lovely smile were often the source of motivation during those long late evenings and week-ends.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# CHAPTER 1

# Introduction

## 1.1 Preliminaries

Road sign recognition represents a challenging problem, since the objects of interest are in a natural and non-controlled environment, the main uncontrolled factor being the illumination. Moreover, the objects are seen dynamically from a moving platform. Consequently, the point of view and distance with respect to road signs continuously vary. Road sign recognition is also interesting because it involves a relatively large but finite number of objects to be recognized, thus making the classification problem well-defined. The purpose of the present work is to develop a system capable of detecting and recognizing road signs autonomously in color images of road scenes, with the ultimate goal being to achieve this in real-time aboard a moving vehicle.

Nowadays automation plays an ever growing role in almost every aspect of our lives. Transportation is no exception, and the major reasons are increased safety, better efficiency, and lower costs. Programs to support and fund development in transportation technologies have been launched in the United States and Europe respectively under the names Intelligent Transportation Systems (ITS) (formerly Intelligent Vehicle and Highway Systems (IVHS)), and PROMETHEUS (PROgraM for a European Traffic with Highest Efficiency and Unprecedented Safety). Such technologies are also well funded in Japan through the Autonomous Vehicle System (AVS) program, and here in Canada these are being viewed with increasing importance by the different governmental levels.

More specifically with regard to traffic sign recognition systems, the two major avenues of applications in view are:

- Intelligent vehicles and driving assistance, and eventually autonomous vehicles.
- Inventory and maintenance of road signs on a road network.

In the first case, such systems are mostly interesting for car manufacturers, and it is no surprise that work in this area has first been done in countries with automobile industries, *e.g.*Daimler-Benz (now DaimlerChrysler) [**1**] in Germany and Peugeot-Citroën [**2**] in France. Here, the major concern is in improving passenger safety by providing vehicles with warning mechanisms if the driver's behavior is inconsistent with signposting. It could even go so far as the vehicle taking an action in such cases. In this line of thought, the development of completely autonomous vehicles is also being pursued [**3**].

The second avenue is directly aimed at governmental transportation agencies. To date, road sign inventories are performed manually either by a person taking notes directly on the field, or by viewing hours of video tapes filmed on the network. Needless to say that this is an extremely tedious task, both costly in terms of money and especially time. During discussions at the International Road Federation Conference in May 97 in Toronto, people involved in road sign surveys mentioned that the cost of such an operation ranges from $2.00 to $3.00 (US) per road sign. As an example, the city of Detroit has 75,000 road signs on its network, and here in Québec, on the road network administrated by the Ministère des Transports du Québec this number is around 320,000 [**4**]. This limits the frequency at which inventories can be made.

To automate road sign inventories, use of a road sign recognition system in conjunction with a global positioning system (GPS) could be made, thereby giving the type of road sign and its location on the globe. This would also allow a database of road signs to be built. Once such a system is available, it could be mounted to function permanently on trucks as workers drive on the road network for other purposes. An interesting point to mention is that based on the results obtained in this thesis, the Ministère des Transports du Québec has decided to invest money for the further development and integration of the system that will be described.

Also interesting for maintenance purposes is the detection of defective road signs, *i.e.*road signs that are aged, misplaced, or damaged. In fact, these are the most interesting from the authorities' perspective. This is far from being trivial as the problem of identifying road signs in good condition is already difficult. However, this is a problem worth to keep in mind because once a system capable of detecting road signs in good condition is developed, this more difficult problem can then be tackled.

A question that is often raised concerning road sign detection is: Why use vision to perform such a task? Indeed other types of systems that provide the equivalent of road sign information are also being developed (see [5] for an example). They are either based on encoded messages placed on or nearby the road signs or on vehicle-to-roadside communications. These are non-autonomous systems, and require major changes in the infrastructure, such as putting beacons or emitters on each road sign. Also, it is far from being evident that such an infrastructure can be installed on all the roads of a network, not just on highways or major roads, and this at a reasonable cost. Nowadays governments tend to strongly favor solutions in which they are the least involved. It is thus not to be expected in a near future that road networks will be fully automated. Moreover, for the eventual detection of defective road signs, since road signs aim at sending visual information to drivers, visual inspection is mandatory to insure their full functionality. These are all reasons that make autonomous vision systems far more advantageous, and provide a justification why work on such systems is pursued and valuable. Thus, before road networks are automated, "visual" road signs will remain. With constantly improving computational speed and hardware, it is hopefully possible to develop a computer vision system to efficiently perform road sign recognition in real-time with a minimum of hardware (computer, frame grabber and camera) as will be shown here.

Road sign recognition is a nice example of a complete computer vision problem with detection and shape recognition as the primary low level sub-problems, and system integration at a higher level. For detection, the specific colors of road signs will be exploited. This is, however, not as simple as it may seem at first sight, since road signs are illuminated by daylight, which is subject to wide variations depending on time of day and weather conditions. This is a problem to which part of the thesis will be devoted, and a detection scheme based on a physical model will be developed. Once detection is performed reliably, the

question of validating whether the detected objects are road signs or not arises. Road signs form a finite set of objects as mentioned above, and are particularly well suited to template matching for their recognition with the templates being part of a recognition database. As the basic tool for this, an efficient template matching technique that is able to account for spatial as well as color information has been developed. Road sign recognition is not just detection and recognition considered sequentially. These two aspects must communicate with each other. At first sight, it would appear that these must be performed in a linear way with detection feeding information to recognition. In fact, this is how a system is first implemented. However, as will be seen, specific problems arise, and the system can be endowed with feedback so that recognition influences detection thereby reaching better segmentation results in the course of the detection process. Integration issues, and others, play an important role in a system having to deal with complex situations, and have a direct impact on the overall system's performance. Such aspects will be discussed in the present work.

The previous paragraphs are just a brief introduction to what is to come in this thesis, and before introducing further in more detail the present work, previous work with a focus on road sign recognition will first be presented. Literature reviews on color detection and template matching are deferred to their respective chapters.

## 1.2 Previous Work

In this section, the literature on road sign recognition based on computer vision techniques will be reviewed and discussed. This will serve to position the present work with respect to what has been done elsewhere. The presentation follows more or less a chronological order modulated by a certain order of importance. A note regarding the processing times given in some of the works described: computing power has evolved extremely rapidly in recent years, and so the significance of the numbers given must be considered with caution. When mentioned in the original articles, the type of processor used is reproduced here.

**1.2.1 Literature Review.** Road sign detection and recognition is a relatively new application of computer vision. Earliest work dates back to 1987 with Akatsuka and Imai [**6**], who considered only speed limit road signs that consist of a red annular border

with blue digits in a white interior to indicate the speed. Images of real scenes containing such road signs are used to empirically determine regions in the normalized *rgb* color space corresponding to the aforementioned colors. Detection is performed by embodying those regions into look-up tables, and scanning the image in order to filter out unwanted colors and keep only pixels with appropriate colors, which is very fast. The scan, and hence the search for road signs, is restricted to a predetermined sub-area in the image where road signs are likely to be present to further increase processing speed. To recognize road signs, template matching is used. No time sequence information is exploited. As regards processing speed, a dedicated image processor with a 14 MHz master clock is used for the color segmentation part which is carried out in 1/60 of a second and a PC-AT-286 computer carries out the recognition part in 0.5 second. No figures are reported on the detection and recognition performance of the system.

Reference [**7**] describes an on-board (mobile) system for the purpose of measuring the retro-reflectivity of road signs for making decisions on their replacement. The measurements are made during daylight hours at speeds of up to 90 km/h. Image acquisition is done with a camera coupled to an electronic flash source which can provide a short burst of light sufficiently bright to overcome the sign's luminance due to ambient daylight illumination. Histograms of the retro-reflected intensity distribution are used to detect road signs, and measure their average retro-reflectance. A laser rangefinder provides the road sign to vehicle distance, and at an appropriate distance an image of the road sign is captured in order to build a database. In this process an operator continuously aims the optical head at an oncoming road sign to do the tracking. This is a weak part, because the system is not automatic; it requires the intervention of an operator, which is slow, and poses considerable problems when multiple road signs are present in the scene. In such cases, how can the operator aim at all of them, especially at such high speeds? The ability of the system to detect road signs is not quantified, and as regards processing time, it can be considered irrelevant since most of the work is performed by an operator.

In what appears to be an exploratory study, De Saint Blancard [**8**] restricts her work to 4 types of road signs: octagonal stop signs, danger warnings (triangles up), yields (triangles down) and round forbidding signs, which, in Europe, all have a red component. Image acquisition is performed with a black and white camera in front of which a filter cutting off the

red portion of the spectrum is positioned. This is to increase the contrast of the road signs of interest with respect to the background. Detection is performed by identifying closed contours in the image (edge detection and edge following using Freeman coding). Three edge detection techniques were studied: mathematical morphology, the Sobel operator, and the differential Nagao gradient, with the latter being favored. Closed contours of appropriate code length are selected, and passed to a decision making system for identification. Three approaches, not described in detail, were investigated for this purpose: a rule-based approach called "structured programming" (which was abandoned because of its slow execution time and the need for special programming skills; the exact meaning of "structured programming" is not made clear in the article), expert system classification (which is deemed to be potentially the most interesting approach, but which was not pursued because of the effort required and short term constraints), and a restricted Coulomb energy (RCE) neural network that uses a histogram of the Freeman coding and the average gray level in the bounding box containing the contour as classification features. The neural network (NN) was finally chosen because of its ease of use (no reason is given as to why an RCE network was chosen in particular). No image sequence information is exploited. A processing time of 0.7 second is achieved for detection and recognition on a PC equipped with an image processing card based on a digital signal processor (Texas Instruments TMS320C25). On a set of 614 road signs in real images, 94.9% were correctly identified, 0.8% were not, 2.6% were uncertain (the meaning of "uncertain" is not made clear), and 1.4% were unidentified. Here, by identification it is meant that the outer shape of the road sign is correctly classified, and so this does not correspond to the full identification of a road sign. Mention is made that in future work, the use of color will be investigated.

Krumbiegel *et al.* [**9**] present a connectionist approach to road sign recognition. It is divided into three modules: a detection module, a pattern recognition module, and a knowledge-based control unit. The detection module, which consists in a NN trained with the backpropagation procedure, finds picture areas where traffic signs are likely to occur by using the particular colors of traffic signs. These picture areas of interest are passed to the pattern recognition module, which is divided into two layers. The first layer consists of three "experts" based on NN's which are concerned with color, texture, and shape. The color expert uses the color extracts to search for typical color combinations, while the texture

expert uses the gray levels of the image, and the form expert uses an edge detector. The second layer deals with the recognition of pictograms and/or characters. The purpose of the knowledge-based unit is to perform the final classification by combining a small number of partial results in a rule-based manner. No details are given about the color space used, as well as on how the particular colors of road signs are determined. Mention is made that further work has to be done concerning the integration of the detection and recognition modules by the knowledge-based unit. Although the ideas are expounded, it is clear that at the time the article was written, the system described was still in a preliminary stage. No performance figures or processing time are given.

Betke and Makris [**10**] have presented an approach based on template matching and fast simulated annealing, in which object recognition is addressed as the problem of best describing a match between a hypothesized object and an image (or sub-image). To illustrate their approach, they consider the problem of road sign recognition. They work with gray scale images, and use the normalized correlation coefficient as a mesure of a match. This coefficient provides for a certain degree of invariance with respect to changes in the illumination[1]. Templates are generated on-line during the search for the optimum by transforming a model image of the hypothesized object using parameters relevant to the road sign viewing geometry (angle of rotation, scale changes along both coordinate axes, and position of the object in the image, which amounts to 5 parameters). Optimization for finding the parameters of the best match is carried out by resorting to a fast simulated annealing procedure in a 5-dimensional search space. When the problem of determining *which* object is present in the image, the preceding procedure is repeated for each model in the library of objects they consider, and the object for which the correlation is highest is taken as the one recognized in the scene. This ignores the situation where multiple road signs may be present in an image, which is a problem not addressed. In the results they present, only 8 road sign model images are considered for recognition and the tests are performed on 18 scene images, all containing a single road sign (except for one which contains two). To avoid false positive

[1]This invariance is exact only when the overall level of illumination is changed, *i.e.*when the spectral distribution of the illumination is multiplied by a constant factor. However, outdoors not just the overall level changes, but also the spectral composition (*i.e.*the shape of the spectral distribution of the illumination). This influences the gray levels rendered by a camera in a non-linear manner. This is often neglected by people working with gray scale images, who assume that a change in the illumination results in a constant scaling of the gray levels, which is not always true.

matches, they introduce a measure of the information content in a model image based on its autocorrelation, and propose that only models having a high information content should be considered. Models with low information content are simply ignored. With respect to processing time, the only result reported is 15 seconds for a $112 \times 77$ pixels image.

Kellmeyer and Zwahlen [**11**] have presented a system entirely based on back-propagation NNs for both color segmentation and pattern recognition. At the color detection level, a supervised NN segments the image into 8 colors that are important to highway road sign detection (red, orange, yellow, green, blue, violet, brown and achromatic). The NN was trained with a set of 2400 color examples (the authors mention that this was an extremely tedious task)[2]. Next, the system scans the image for color regions that are potential highway warning road signs. The regions of interest thus obtained are fed to a road sign recognition NN. In their system, only highway warning signs (yellow diamond-shaped) are fully processed, *i.e.*detected and recognized. Performance results are given for image sequences recorded on a video tape which contain 35 separate highway warning signs to be recognized and distributed over 55 images (as such, however, their system does not perform tracking of the road signs in image sequences). In these conditions, 86% of the road signs could be detected and located[3] with 3 false alarms (5.5%). In those statistics, only images in which the signs that are sufficiently large for their algorithm are considered. The processing time for a $640 \times 480$ image is 2 minutes 30 seconds on a 486/25MHz PC with a Targa+/64 image acquisition board.

Closer to here, Lalonde and Li from CRIM (Centre de Recherche Informatique de Montréal) have developed an algorithm for detecting road signs based on color indexing [**12**]. It is composed of two stages. Working with the $HSI$ color space, they first extract color information from the image in order to locate road signs based on their salient colors. A simple labelling scheme is adopted in which a label corresponding to a road sign color is assigned to a pixel if its color falls into a specific region of the sub-space $HS$ specified by a central point. The Euclidean distance with respect to this central point is employed to

[2]Their motivation in resorting to a NN for color segmentation, is, as they put it, the "... lack of a mathematical equation [that relates one's perception of color to the physical color spectrum] is the major reason for choosing a neural network to learn from examples the relationship between spectral color and human color perception".

[3]It is not clear if the system is only able to determine whether a road sign is yellow diamond-shaped, or can go further and also give the identity of the road sign, *e.g.*left turn.

determine whether a color falls into such a region or not. Pixels labelled in this way are then grouped together via a standard component connecting algorithm. Groups of pixels determine regions of interest (ROI's) in the image delineated by bounding boxes. These ROI's are validated (or rejected) as potential road signs by employing color indexing, which is a technique due to Swain and Ballard [**13**] for browsing a database of objects with their color attributes as the index. Each object is modeled by a normalized color histogram describing its color distribution. Comparison of two objects is done by verifying whether their histograms match or not. To evaluate a match, several measures can be envisaged, Lalonde and Li use a simple one. They mention that the effects of sunlight and its changing spectral distribution according to weather conditions and time of day pose a problem, which they have not tackled. Further, they conclude that the most significant limitation of their method must come from the labelling phase, which is sensitive to color shifts caused by daylight. As regards identification performance, since their approach uses color indexing which is based on histogram comparison, it does not allow to discriminate between road signs that have the same histograms, hence the full recognition problem is not addressed (for example two road signs which are the mirror image of one another will have the same color histogram). In these conditions, they present tests they have made on 5 images with a total of 11 road signs to be found. They are able to find 10 of the 11 road signs (90.9%), but at the same time generate a total of 23 false alarms (which, when referred to the number of images is a rate of 460%). No figures are given in terms of processing time.

Azami and Aoki [**14**] extract road signs as rectangles in gray scale images by applying the Sobel operator and combining vertical and horizontal edges using certain criteria. Examination of the interior content of the extracted rectangles is performed to discard unsuitable ones. Because of the unreliability of the edge detection method they employ in the presence of complex backgrounds, some road signs are not detected. To cope with this, the following heuristic, based on the assumption that route guidance signs are of the same size, is introduced: Using the already extracted rectangles, others of approximately the same size and lying at about the same height in the image as those already detected (horizontal proximity) are searched for (so only road signs in these particular situations are considered). The interior region of the extracted rectangles is analyzed to detect characters and arrows, which are first approximated as rectangles. Undetected characters and arrows

are found by a method similar to that for finding undetected road signs. The authors mention that in some instances difficulties in extracting road signs are encountered, and plan to improve on this by exploiting color and time sequence information. This further work is the subject of [**15**], where three methods of detection are used in parallel and combined together: rectangle detection (as in the previous paper), detection of signs in sky regions of the image, and detection using $HSV$ color coordinates (only blue and green hues are considered). Few details are given about tracking, but position prediction is made. In both these works, no information is given of the processing speed of the systems as well as on recognition performances.

French researchers have implemented a traffic sign localization system on a vehicle for highway inventories [**16**]. Two sensing solutions have been investigated for road scene analysis: a 3-CCD color camera, and a near infrared (IR) camera in conjunction with a near IR illuminator. With images from the color camera, segmentation relies on thresholding the hue and saturation bands to extract the red and blue colors of the signs involved in the demonstration. IR segmentation is based on an automatic threshold for extracting the bright spots caused by retro-reflective road signs. From the detected pixels, blobs are formed and retained based on their area and aspect ratio. No further recognition is done. These blobs are assumed to be road sign candidates, and their centroids are sent to a Kalman tracker, which also receives motion information from an on-board hybrid inertial-odometer navigation system. The 3-D location of the candidates is progressively (at each new image) reconstructed, and refined by the Kalman filter. Upon some 3-D location quality criteria (number of associations, distance with respect to the vehicle), a decision is made to trigger an acquisition with a high resolution color camera aimed at the road sign passing by. Then the 3-D location of the sign, its associated uncertainty, and its class are stored with the high resolution image file, providing a geo-referenced collection of road sign images. Their system achieves a processing rate of one frame per second (1 Hz) at a speed of 5 km/h, with detection being carried on an image processing board, and tracking on a PC. Tests on a short dedicated road section with 5 road signs are discussed, but with no assessment on detection performance figures.

Reference [**17**] reports preliminary investigations on road sign recognition in city streets carried on for CRIM in which images obtained from the Internet (Germany) or by scanning

photographs captured in Montréal with a conventional camera are used. The proposed approach proceeds in four steps. First, color segmentation of the image is performed using pixel classification based on color followed by pixel grouping to detect potential regions in the image where road signs can be found. For the color pixel classification part, cylindrical regions in RGB space corresponding to road sign colors are constructed by manually selecting pixels on road signs found in sample images (how these regions are constructed and the parameters used is not explained, and the number of road sign colors considered is not specified). The second step consists of edge-based segmentation performed on the regions found in the first step from which closed contours composed of straight segments and circular arcs are extracted. In the third step, a geometrical analysis of these contours is made (number of straight sections, angles and length ratios between straight sections, etc...) to extract those that are characteristic of road signs. Finally, for classification of the road signs, the percentage of road sign colors within the contours are exploited (as in Lalonde and Li [**12**], this does not allow to identify a road sign, but only its class because more than one road sign may have the same color signature). The types and number of road signs considered for recognition as well as recognition performance figures are not given. Results for 3 images are presented in which only red is considered for color classification (the red region in RGB space is determined from pixel values taken on one of the road signs in these images, thus making the approach non-autonomous).

In [**18**], Kehtarnavaz *et al.*focus on the recognition of stop signs based on the sequential processing of color and shape. A statistical study of the red color of a large number of stop signs is performed in the hue-saturation-brightness color coordinate system to determine a region in color space corresponding to stop signs red. Potential stop signs are segmented from images using this color space region. Median and morphological filtering, the Sobel edge detector, and the Hough transformation are combined to obtain boundary contours of the detected regions. Stop signs are identified as the contours that can be approximated by eight straight line segments. As regards performance, an experiment is conducted using 11 road signs, 5 of which are stops. The system is able to identify all stops, but these are rather fragmentary results as only a very small number of samples is considered. Concerning the processing time, no numbers are given. In pursuing his work on road sign detection and recognition, Kehtarnavaz, with Ahmad [**19**], presents a method for recognizing road

signs using both color and shape attributes. Only the red and yellow colors appearing in warning and caution road signs are considered. A discriminant analysis is carried out to determine which color coordinate system gives the highest separability of the clusters representing the different traffic sign colors (red and yellow only). They conclude that the $YIQ$ system is the most appropriate. Based on this, a so-called ART2 NN is used to perform color segmentation. To provide for rotation and scale invariance, signatures obtained from a log-polar transformation followed by a Fourier transform are used. A backpropagation NN is then trained to do the recognition using these signatures. The effects of various sources of noise on their algorithm are examined. For this latter work, no figures are given in terms of recognition performance, and as regards processing time it takes around 80 seconds to analyze an image on a Sparc 1000 server (image dimensions are not specified). In later work, Kehtarnavaz and Estevez [**20**] discuss the implementation of an algorithm on a digital signal processor (DSP) restricted to the recognition of stops, yields, and do-not-enter signs (road signs all having a red component). At the detection level, the algorithm first finds red areas in an image where edge detection is applied. A "redness" measure is defined and thresholded to determine whether a pixel is red or not. For recognition, the histogram of edge orientations for the whole image is used (only one sign can be present in the image for this method to work). On a Spectrum MDC40IC image processing board based on a Texas Instruments TMS320C40 DSP, frame rates of 9 per second are achieved in the worst case corresponding to a yield sign present in the image (yield signs require the most computations). When red objects are not present in the image, 12 frames per second can be processed. The following recognition efficiencies are reported based on images of 49 scenes captured under clear, partly cloudy, and overcast conditions. For stop signs, 50% are correctly identified, 41% are incorrectly identified as another sign (yield or do-not-enter), and 9% are not recognized. For yields, 37% are correctly identified and 63% are incorrectly identified. Finally, 94% of do-not-enter signs are correctly identified, with 6% missed. In all these works of Kehtarnavaz and his co-workers, no temporal image sequence information is used.

In [**21, 22**], a road sign recognition system is presented composed of three parts: segmentation, recognition of the contour shape, and eigenspace representation for identification. For road sign segmentation logarithmic opponent $RG$ and $BY$ values are used, and

the green channel is chosen as the intensity image from which a texture image is extracted. Texture is exploited to segment road signs in the presence of backgrounds with different texture (the surface of road signs being smooth, whereas complex backgrounds such as woods and bushes are highly textured). Empirical thresholds are used to segment the color and texture images. Segmented regions are then eroded and dilated in order to smooth their edges. To recognize the contour shape of a segmented object, two types of contours are considered: polygonal (triangles, diamonds, rectangles, pentagons, and octagons), and circular (railroad signs, do-not-enter, and no-turning signs). For polygonal shapes, the distances along horizontal and vertical scan lines in the image between the boundary of the shape and the sides of a bounding box placed around the shape are used. For each side of the bounding box and its associated distances, and given a particular shape, constraints that characterize the shape must hold between these distances. As regards circular signs, a conic is fitted to their border, with the coefficients satisfying constraints for circles. Although in many cases, it must be easy to distinguish whether an object is polygonal or circular, in others this needs not be so. For instance, a stop sign being octagonal could also be quite well approximated by a circle. In such a case, no criterion is given on how to decide if the sign is octagonal or circular. Regarding the identification of road signs, a decision tree is used, which is subdivided along the possible road sign shapes. Color is also used as a first decision step to constrain the possibilities, when the segmented object comes from either the $RG$ or the $BY$ image (*e.g.*when segmented from the $BY$ image, the object must be blue or yellow). When segmented from the texture image, no color information can be used, making the search in the database more consuming. For matching in this database, the eigenspace representation of Murase and Nayar is used [**23, 24**]. This system was tested using pictures taken with both a digital camera and an ordinary camera (then the film was digitized to make photo-CDs), and an overall identification rate of 89.8% is achieved.

De la Escalera *et al.* [**25**] have presented a road sign recognition system based on color segmentation, corner detection and neural networks. Color segmentation is performed by directly thresholding the red channel, along with the normalized green and blue channels of the input RGB image (the normalization at each pixel in these color normalized images is done with the corresponding values in the red image). Only red road signs are considered for segmentation. Corners are detected in the resulting color thresholded image by using

correlation filters. The algorithm seeks for specific corner combinations in the image corresponding to road signs (*e.g.*for triangular signs, 3 corners of approximately 60 degrees at the vertices of an equilateral triangle are sought). For detection, upward triangular, rectangular, and circular road signs are considered, but inverted triangular (yield) and octagonal (stop) signs are not. Recognition is performed with two multilayer perceptron neural networks (one for triangular signs and the other for circular signs) to which a sub-image where a putative road sign was detected, normalized to $30 \times 30$ pixels, is presented as input pattern. Training patterns are ideal signs (models) of which 5 rotated versions are computed, to each of which 3 levels of Gaussian noise are added. These are then segmented using 3 different threshold levels and displaced at 3 different positions along the $x$ axis. This results in 180 training patterns for a given sign. Nine different triangular and nine circular road signs are considered in their system. Hence, only road signs with those shapes are considered. The detection part of their algorithm is implemented on an image processing board (ITI 150/40) in a PC-486 33 MHz host computer, with the neural networks running on the host computer. Detection takes 220 ms for $256 \times 256$ images and the neural networks take 1.2 seconds. Results are shown, but detection and recognition performances are not detailed.

An Italian group based in Genova has developed a rather complete system in its overall architecture [**26, 27, 28, 29, 30**], but restricted to the detection and recognition of circular and triangular road signs. The approach comprises three stages. First, a region of the image, called the *search region*, in which road signs are most likely to be found is selected. For this, the middle right part of the image is used, but in their early work mention is made that this selection could be more sophisticated, *e.g.*based on localizing the road, the sky, or by using color (indeed in their early work, no use of color is made, and this region is determined a priori in a fixed manner as just described, whereas in later work search regions are determined by exploiting color). Second, edge detection is applied to the search region. To detect triangular shapes, edge chaining and polygonal approximation is used, whereas for circular ones annuli of variable radii are employed[4]. This leads to the extraction of clusters of edges having the desired shapes. The first two stages described thus far constitute the detection part of their system. For triangular road signs, the efficiency of this detection

[4]The question arises as to why not use "triangular" annuli in the case of triangular shapes as well.

approach is "about" 92% for a set of 600 test images containing one or more road signs with 11 false alarms (1.8%). The average processing time is 6 seconds on a SparcStation ELC. As regards circular road signs, they obtain an efficiency between 93% (first set of 200 "difficult images") and 96% (second set of 200 "easy" images), with respectively 25 and 27 false alarms for each case (12.5% and 13.5%). The processing time is 15 seconds in the case of circular signs. In the third stage, the inner region of each cluster is tested against a database of normalized road sign images for classification or rejection by template matching using a cross-correlation. To increase robustness, temporal integration between frames is exploited by tracking detected objects via Kalman filtering methods. With temporal integration, the recognition performance is "about" 98%. As such, recognition in an image is performed in an average time of 500ms on a SparcStation ELC with a database containing 60 circular and 47 triangular signs. Hence, in their system, most of the processing time is spent in the detection part. In their later work [**28**], color has been incorporated to further restrict the regions where edge detection (and geometrical analysis of the edges) is carried out. They mention substantial improvements (no numbers are given), and note that color is also very useful for database template matching, since it more efficiently directs the search in the database.

Priese *et al.* [**31**] from the University of Koblenz-Landau describe a traffic sign classifier module designed to localize traffic signs and recognize their broad classes in real-time. They proceed in two steps. In the first, the image is segmented using their Color Structure Code (CSC) segmentation method, which combines both local region growing and top-down segmentation techniques [**32, 33**]. This gives a set of color regions, and those corresponding to traffic sign colors are kept; the $HSV$ color space is used. The second step is a geometric classification of the color objects. For every object, the probability that it belongs to a certain shape class is computed. These shape classes depend on the different traffic sign colors, because for a given color not all shapes are possible. The possiblity of an object having more than one shape interpretation is embodied in their approach. The shape of an object is encoded by its convex hull, which is approximated by polygon with 24 edges. Simple recognition methods are then applied to determine the shape (*e.g.*a circle is characterized by 24 edges of nearly equal length) using probability measures. Such shape attributes are supplemented by a color histogram of the object, since geometric attributes

alone give rise to too many false alarms. They claim to have performed extensive tests of their module with the help of a database of more than 20,000 $512 \times 512$ real color images, but do not give any recognition performance figures. Their system has been implemented on a transputer image processing system using various types of digital signal processors (T805, C40 and MP601), but the best performances were obtained on a SUN SPARC 10-51. The processing time is 3.3 seconds for a complete $512 \times 512$ image, and reduces to 771ms when a sub-region of interest $200 \times 300$ in size is considered in the image (these tests were performed on a sample of 10 color images). In the previous description, only the classes of road signs are recognized. This work is further pursued in [**34**] by the addition of a module involved with the exact identification of the detected and classified road signs. For this task, ideograms are first isolated from the traffic signs, and then analyzed by different modules. The module comprises four submodules. The first detects the position and direction of arrows using skeletons, the second one recognizes numerals, and the third submodule is a general nearest neighbor classifier applied to three classes of ideograms: prohibition sign ideograms, speed limits, and arrows on mandatory signs. A fourth module based on a NN is used for prohibition signs. Some of these submodules are used competitively in order to increase recognition rates. As presented, this work on ideogram identification seems to be at an exploratory stage, and no exact numbers are given in terms of recognition performance and processing time (although, as regards processing time, they mention that their approach is slow as it stands). The work at Koblenz-Landau has been done with partial funding from, and in collaboration with Daimler-Benz.

Since 1988, Daimler-Benz has been developing a traffic sign recognition system [**1, 35, 36, 37**] in collaboration with German university laboratories (Paderborn and Koblenz). The data management of their system is organized following the Perception Machine model of Tsotsos [**38**], which exploits the temporal continuity of image sequences. The interpretation of the traffic signs is stored in a data structure, which is updated as a new image is analyzed and interpreted. This structure is a list of traffic sign object (TSO) structures, themselves containing information about the color, the shape, the pictogram, the location, and the reliability of traffic sign candidates. Their system consists in a network of modules having specialized purposes and divided into three levels of data processing. At the lowest

level are the modules called specialists, concerned with color segmentation, shape recognition (outer shape), and pictogram recognition. The specialists react in a stimulus-response form, and are dedicated to their specific task. They neither have an explicit representation of their environment, nor any memory of their history. At the second level are the detection and tracking modules which are linked to the specialists. The detection module sends new candidates to the administration module at the top level, while tracking analyzes and follows detected candidates. As long as a candidate is not fully recognized, it is tracked and analyzed in order to update its associated structure and gain information about it. However, once it is recognized only tracking takes place. The administration module synchronizes detection and tracking. In their system, information flows both in a bottom-up and top-down manner. Bottom-up, since the system extracts data from an image which is used to update the traffic sign data structure, and top-down because from the previously accumulated information, the administration makes queries to the appropriate modules of the lower levels to extract relevant data. From frame to frame, the list of TSO's is updated with the list of new TSO candidates extracted from a new image. The matching of two TSO structures is done symbolically according to their content. Concerning the specialists, color segmentation is done in two stages with three colors being considered: red, blue, and yellow (in the early work done at Daimler-Benz [**39**], color was ignored due to insufficient computing power and monochrome images were processed using the hierarchical structure coding (HSC) developed by Hartmann [**40**]). In the first step, pixel labelling is performed with a supervised polynomial classifier [**41, 42**] previously trained with traffic sign colors taken out of manually labelled traffic scene images. The normalized $Irg$ color space is used. In the second step, the color connected component algorithm of Mandler and Oberländer [**43**] allows for the description of interesting regions in terms of boundaries and color. Each region is checked against constraints on its size and color combinations with neighboring regions, and accordingly marked as interesting or not. This results in a list of bounding boxes, called regions of interest (ROI's), potentially containing road signs. The purpose of the recognition process is to verify whether an ROI is a traffic sign or not, and in the affirmative to identify the pictographic information it contains. Both verification and pictogram recognition are performed using supervised distance-weighted k-Nearest-Neighbor classifiers which are trained on manually selected examples [**44**]. In the former, training is

performed on the shape of the outer contour of road signs extracted from training images, whereas in the latter the pictogram is used as the training attribute. Daimler-Benz has built a sophisticated prototype using dedicated hardware mounted in a car with a processing capability of 5 images per second [**1, 45**][5]. It consists in a multiprocessor network based on four T805 INMOS-transputers and four PowerPCs. Their system is able to recognize speed limits, warning signs, prescriptive signs, and informative signs at a rate of about 90% with 5% of false positives. The exact number of road signs the system is able to recognize is not specified.

In a series of articles, Gavrila [**46, 47, 48, 49**][6] also working at Daimler-Benz, describes an approach to road sign recognition based on a hierarchical template matching technique using distance transforms of gray-scale images[7]. Given an input image, the initial processing steps (pre-processing) consist of computing its binary edge image, and then its distance transform (DT) image which contains at each pixel the image distance (geometrical distance in terms of pixels) to the nearest edge pixel in the edge image. Matching then corresponds to sliding the binary edge image of a sought object (the template) over the DT image and computing the correlation at each location normalized by the number of "ON" pixels in the template[8]. A small value of the correlation at a given position denotes a high degree of matching between the image and the template at that position. A user-supplied threshold determines what "small" means for deciding whether a match is found or not. Instead of using a single binary edge image in which edges are considered irrespective of their orientations, as is customarily done in DT matching, Gavrila proposes to partition edges into $M$ bins that discretize the continuum of edge directions in the interval $[0, 2\pi]$ ($M = 8$ is considered). This gives rise to $M$ edge images, one for each bin, from which $M$ DT images are computed. Similarly, $M$ templates are considered. The match measure is then obtained by sliding and correlating each template with its associated DT image

[5]As such, the image size is not specified in [**1, 45**], but presumably $512 \times 512$ images are used, because in a previous work [**36**] to those just cited, a processing time of about 3 seconds a SUN Sparc-2 is reported for images $512 \times 512$ in size. In a slightly modified version of the article in [**36**], 1 second of processing time on a SUN Sparc-10 is reported in [**37**].

[6]These articles, being very similar in theoretical content and as regards experimental results, will be described as a whole.

[7]This work seems to be carried independently as an alternative to the earlier work at Daimler-Benz described previously.

[8]This normalization is necessary, otherwise a template with many features (*i.e.*with many "ON" pixels) would tend to give higher correlation values than one with few features.

and summing the contributions corresponding to each of the $M$ bins. The motivation for distinguishing edge directions is to reduce false positives because when no distinction is made between edge orientations, there is always the possibility that an "ON" pixel in the template is being matched with a minimum distance in the DT image corresponding to an edge actually having a different orientation (*i.e.*in effect matching edges with different orientations). The previous approach considers the situation where a single object must be found in an image. In the case where the presence of multiple objects, say $N$, must be assessed in the image, brute-force could be used (*i.e.*slide and correlate the templates associated with each of the objects and determine image locations where the correlation is sufficiently low), but this would be inefficient, especially when subsets of similar objects can be considered (when there is no similarity between the objects, however, not much better than brute-force can be done). In situations were similarity is present, a matching scheme using a template hierarchy organized in a tree structure along with a coarse-to-fine search over the image considered at different grid sizes is suggested. The motivation is that at a coarse search level on a large image grid size, there is no advantage in matching similar objects individually. To construct the hierarchy, Gavrila proposes to group similar objects to form template "prototypes" representing such groups. Unfortunately, these ideas are not fully exploited since the objects considered are simple and highly dissimilar (circles and triangles (up and down)). In the examples for which experimental results are given, objects are grouped solely based on size, *e.g.*a template for up triangles 7 to 12 pixels wide, and another template again for up triangles 13 to 18 pixels wide (prototypes at the first coarser level in the template hierarchy). This is not an approach that strongly exploits similarity, and even worse, more than one template at a coarse level is needed for the *same* type of object (*e.g.*up triangles). Grouping based on size as presented has more to do with noise robustness and small object size deviations than with object similarity (although some notion of similarity is present in that objects of the same shape but with different dimensions are grouped). Nevertheless, this approach based on DT's allows to locate (or detect) potential road signs using the basic characteristic shapes of their contours in gray-scale images. It does not give, however, a final classification, which is to determine which road sign it is, *e.g.*is it a triangular road sign indicating a road narrowing, or indicating a railway crossing? For this an RBF (radial basis function) network is used. No tracking in

image sequences is performed in this system. Specifically as regards experimental results, 3 classes of road signs are considered: circular and triangular up and down. For these, a "manually" specified tree hierarchy containing 36 templates in total is used. Adjoined to this is an RBF network which is able to distinguish "about" (the author says) 10 pictograms. Road signs that appear tilted and/or skewed in the image are not considered. The image size used is $360 \times 288$ pixels. Both off-line and on-board tests have been performed. Off-line, a set of 1000 test images taken during day-time (sunny, rainy), and night-time is considered. With the previously described system configuration, DT-based road sign detection in single images achieves rates over 95% with two or less false positives per image on average, whereas as regards RBF classification, preliminary recognition rates above 90% with less than 5% false positives are reported. Mention is made that detection rates may degrade by as much as 15% in difficult environmental conditions (rain drops, partial occlusion by window wiper, and direct sunlight into the camera), when road signs are tilted, or when contrast is low (which is more significant in gray-scale than in color images). A suggested solution is to add more templates to the hierarchy tree and lower the edge detection threshold. As regards on-board tests, by optimizing the algorithm for SIMD instructions sets and using a dual-Pentium II MMX 450 MHz computer, processing rates of 10-15 images per second ($360 \times 288$ images) are achieved.

The literature reviewed thus far gives a good overview of the diversity of approaches that have been considered in the field of road sign recognition and surveys, in the author's opinion, the most significant works in that field as regards systems. For completeness, further references will now be cited, but without going as much into the details. In work done in collaboration with CRIM, Prince and Bergevin [**50**] have presented an interesting approach for road sign recognition based on affine invariant matching of interest points extracted from closed edge contours. Unfortunately, their approach poses a severe problem for integration in a real application system, as the models used in their recognition database require an extremely large amount of memory (for example, representing a stop sign requires 442 Megabytes). No performance in terms of processing time and recognition percentages are given. Hsu and Huang [**51, 52**] have presented a system based on matching pursuit (MP) filters devoted to the recognition of red triangular and circular road signs. The first step of detection consist in selecting a predetermined search region where road signs are

most likely to appear (middle right part of the image)[9]. If a circular or triangular shape is found in this region, a closer view image, called the detailed search image, is captured (how this is implemented is not explicited). Template matching based on matching pursuit (MP) filters is then used to recognize road signs in the closer view image. To test the approach, a database containing 30 different triangular and 10 circular road signs is used. For triangular road signs, 356 test samples are used, and a recognition rate of 88% is achieved. For circular road signs, a recognition rate of 86% is obtained with 125 test samples. The approach runs in 350 ms on a Pentium II 300 MHz computer. Some authors have also considered the use of optical correlators for road sign recognition [**2, 53, 54, 55, 56, 57, 58**]. Correlation is a well-known method for performing template matching in recognition applications. Optical correlation relies on the fact that it is possible to perfom a Fourier transformation by using a lens. The big advantage is that the computation takes place at the speed of light. However, to exploit an optical correlator, both pre- and post-processing are required, and this is not discussed in the cited references. Moreover, although the correlation computation is very quick, the electronic access to the hardware effectively limits this speed. These are definitely major problems which cannot be ignored, and makes these works far less interesting because they are too simplistic. This does not mean that optical correlation could not be used for the purpose of road sign recognition, but only that it cannot be used in a raw manner as these authors present it.

**1.2.2 Discussion.** In retrospect, it is seen that for detection, two types of approaches have been considered: those based on edges in monochrome images, and those based on color. Working with color images is slower and bulkier than with monochrome images, but with the ever increasing computing power, one can reasonably expect that to detect road signs, methods relying solely on monochrome information will tend to reduce in importance. Moreover, many researchers that have worked with monochrome images have come to the conclusion that color would improve their results (*e.g.*see [**8, 39, 14, 28**]), and pursue this avenue. In all cases, color detection relying on training has been favored, with no physics-based modelling approaches being investigated. With respect to recognition, the methods are various. NN's appear to be popular, and modelling has been exploited in only few works (such as in [**25**]). Concerning tracking, only a few consider it. Kalman

[9]This is analogous to [**30**].

filtering is the approach most widely used, but in [**1**] another method is considered, which is deterministic in nature.

The method of Betke and Makris [**10**] somewhat differs from the other approaches, because both detection and recognition are performed at the same time. However, a major disadvantage of their approach is that to search through the database of objects to be recognized, brute force needs to be used, *i.e.*try all the objects of the database, since no information is extracted from the image *per se*, such as from some form of pre-processing that would identify regions of interest where interesting objects could potentially be (the approach described by Gavrila [**46**] is somewhat similar to that of Betke and Makris in that a template is correlated with the image, but the computational load is reduced by matching at different image scales and by resorting to a template hierarchy). Also, in Betke and Makris, for a given image, only a single object is selected which is the one with highest correlation. Moreover only objects having a high information content are considered, the others being ignored. The issue of how to increase information content is not explored, but surely, using color cues would help in these matters. Thus the problem of *automatic* traffic sign recognition is not considered in its entirety.

From a global system's point of view, that of Daimler-Benz [**1, 37**] is quite interesting. It is a nice combination of artificial intelligence concepts and computer vision, and the interaction between the different modules is approached in a clever way. However, some of the approaches are arguable, especially regarding the detection and recognition specialists. As will be seen later, the reason for this is that these aspects of their system rely heavily on learning from examples. Moreover, not much is said about what makes their system break down and in which situations, and what are the potential remedies. Nevertheless, at the time the present doctoral thesis project was undertaken, this system appeared to be the most complete and integrated of all presented in the literature. It should perhaps be noted that after 1995, no more progress on this system was reported in the open technical literature. It is not clear as to what happened with this system since then: Is it close to the market, or has it been abandoned? Other developments on road sign recognition at Daimler-Benz, have started to be published again starting in 1998 [**46**], but, which as described above, have to do with quite a different approach. The Italian group of Genova [**30**] has also developed a nice system, but it is definitely not at the level of the preceding one. In both the work

at Daimler-Benz and of the Italians, practical problems at the integration level and where their systems break down (and why) are not discussed.

As regards the latest developements at DaimlerChrysler [**49**], many nice ideas are suggested, but much work has to be done in order that they be fully exploited. Questions remain as to how to automatically generate the template hierarchy. Simulated annealing is suggested, but results on such an avenue for road sign detection are not presented. In fact in the experiments, a hierarchy constructed manually is used. Also, only circular and triangular objects are considered, whereas stop signs, no entries, rectangular signs *etc...* are not. With these three types of road signs, the hierarchy already contains 36 templates. Moreover, the image must be scanned and correlations (8 in fact to account for edge orientations) must be computed at each location. Different grid sizes are used to reduce this computational load, but with more road sign shapes to be accounted for, will the whole approach remain computationally as efficient? Further, questions regarding how to choose the image grid sizes to work with, the object sizes to use in the template hierarchy, and geometrical deformations such as rotations, skewing, and tilting are not explicited. Notwithstanding these questions, the speed-up using scale is an interesting idea, and, moreover, this work is the only one to have at least given a few results on road sign recognition at night.

The approach of de la Escalera *et al.*is interesting with respect to the present work, because in effect, their neural networks are trained with models to which variations are added. In the present work, models will also be used, but differently and not in conjuction with neural networks. The system of de la Escalera *et al.*nevertheless stands out from other approaches in road sign recognition based on learning since the latter exclusively use examples extracted from real images. Unfortunately, de la Escalera *et al.*do not discuss the performances of their system, other than the processing speed.

In the present thesis, some general ideas concerning road sign recognition will be seen to resemble those of the works discussed and reviewed in this and the previous sections. This is to be expected because clearly in a complete system, modules for detection, recognition, and tracking must be present. However, the particular methods for detection and classification will strongly differ from those considered by these other groups. It is towards these aspects that the thesis will be oriented, as there are good opportunities for interesting work in these domains.

## 1.3 Overview

As emanates from the literature review, a road sign recognition system poses a real challenge for machine vision. It has to deal, in real-time, with a large database of objects under difficult imaging conditions, since images are acquired under widely varying illumination conditions and imaging geometry on a moving vehicle.

The main fundamental problems to be solved are the following. First, a means for reliably detecting road signs in an image under these difficult imaging conditions must be devised. Secondly, a computationally efficient method must be determined, which will allow for the accurate recognition of the road signs, taking into account geometrical deformations, and the fact that a large number of road signs need to be recognized (in the present system, more than 400 road signs are considered). Third, a system architecture which allows real-time processing, is easy to extend in the case where new road signs are added to a regulation, and which can be adapted to other regulations must be developed. Finally, at the architecture level, the complexity of integrating different modules, each with its specific functionalities, represents a challenging task in its own.

In this thesis, a model-based road sign recognition system that brings substantial improvements over previous approaches is presented. First, at the detection level, the color constancy problem caused by the illumination variations is addressed directly via a physics-based technique developed in Chapters 2 and 3, which accurately predicts the color space regions associated with road sign colors under all possible daylight illumination conditions. In this approach, the effects of illumination variability are characterized instead of trying to eliminate them as in color constancy approaches. As such, the color constancy problem is not solved, but the system is made immune to color variations by quantifying them. The manner in which regions in color space associated with road sign colors will be defined here, using a model-based technique, avoids somewhat ad-hoc approaches used in other research. It will be shown that having a precise model of the appearance of a given color under different illumination conditions provides an effective starting point for dealing with practical color constancy problems. To be more precise, the approach to be described first uses a physical model of color formation along with a semi-empirical model of the spectral distribution of daylight to predict color space regions corresponding to the colors that need

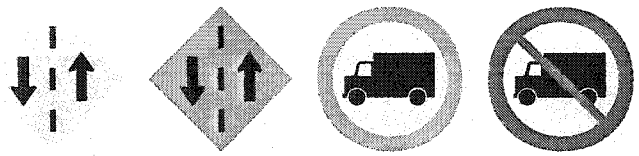



FIGURE 1.1. (a) and (b): Warning (yellow) and construction (orange) road signs having the exact same spatial attributes. (c) and (d): Very different road signs when color is considered, but monochromically very similar, and where spatial information leads to very little discrimination capability.

to be detected (road sign colors here). These predictions serve in a second step to automatically find color samples in real images from which data are extracted. With these data, the predicted color space regions are further refined to better represent real data and their variability.

As described in the literature review, color information has been used in other work on road sign recognition, all of which are based on learning or similar techniques, which can be tedious and difficult to automate, as the initial data to train the classifier must be manually extracted. The present approach does not suffer from this. Other approaches have examined road sign recognition without resorting to color, but this neglects the symbolic information color carries in road signposting (*e.g.*red relates to stop, prohibition and prescription road signs), and some authors ultimately conclude that color is a means of improving their approach. As will be seen, and as already noted in [**28**], color can be exploited as a first logical classification step to efficiently direct the search for a road sign through a recognition database in an automatic road sign recognition system. Moreover, for some regulations, such as those in Canada and in the U.S.A., some road signs are only distinguishable through their color, while others are very similar and would otherwise be difficult to differentiate solely based on monochrome information, see Fig. 1.1. Systems using only monochrome information, are not sufficiently general for these regulations. Further advantages in using color for segmentation as opposed to only intensity, are that color provides for additionnal discrimination and contrast (just think of the case in which two different colors are of the same intensity), and is also invariant with respect to viewing geometry. However, there are also achromatic (black and white) road signs which will not be dealt with in the present system. The problem is that if one tries to detect black and

white road signs with the techniques presented here, then many other objects will be detected as well. The reason is that for most objects found in outdoor scenes, the degree of saturation in their colors is usually small, and so gets easily confounded with achromatic colors. Approaches using monochrome images do not seem limited in this respect as are those using color, but there may well be color-based approaches developed in the future which will circumvent this problem, but they will probably require more subtle techniques (also advances in imaging technology, especially cameras, may well play an important role with regards to these matters). Dealing with achromatic road signs is beyond the scope of the present work, and this must be considered as part of further work to eventually be carried in line with this thesis. The philosophy adopted here is to exploit color as much as possible, to then eventually deal with achromatic road signs[10]. The problem with black and white road signs has also been reported in [**19, 37**].

A second strong point of the present system is the use of a model-based technique for the recognition part as well. The natural organization of road signs into classes lends itself well to a model-based recognition approach. Furthermore, real-time considerations dictate that efficient computational techniques be employed. For this, a novel template matching technique based on an "orthogonal" bitwise encoding which accommodates both vector labels (essential for color) as well as the affine variations in the image formation process, will be developed in Chapter 4. Moreover, through this encoding it is possible to build templates that are able to represent classes of objects. When color is present, using it as a further source of information appears as an obvious thing to do, but this has been neglected in the literature. The technique developed here differs from what is customarily done, which is to convert color information into gray-scale using diverse mappings[11], and then to apply standard cross-correlation techniques to find matches. The template matching technique developed here will play a central role in searching the database used for recognizing road signs.

[10]Incidentally, many research groups that have worked on road sign recognition in gray-scale images are actually detecting colored road signs. For instance in the results presented in [**46, 48**] only road signs with red and blue as their predominant colors are detected (although there are achromatic road signs in the European regulations for which that system was designed, no examples for those are given).

[11]Such mappings are diverse, such as considering only one of the color bands in a color image, or only use a hue image, or use a gray-scale image that enhances a particular color, *etc.*..

As regards the road sign recognition system which will be presented in Chapter 5, apart from integrating the color detection and template matching approaches summarized above, a further highlight resides in the database of road signs used by the recognition algorithm. The data content in this database, structured in a tree-like manner, is generated in a deterministic and automated manner starting with only model images of the road signs to be recognized. No examples extracted from real images as in learning are used, and consequently this approach is markedly different from those in other work which have relied on learning from examples [**1, 37, 51**]. Instead, image formation is modelled from first principles via projective geometry using those model images and a few geometrical parameters to produce encoded templates that take part in the template matching technique introduced above[12]. Such modelling offers great flexibility when it comes to modifying the database, such as considering other geometrical imaging conditions, adding road signs, or changing to a new set of road signs (*e.g.*from North American to European road signs)[13]. This potential of being easily reconfigurable is an appealing aspect of the approach proposed here which is not found in other work, primarily because the latter are based on learning from examples. Moreover, the database used in the present system is complete in that no restriction is imposed on the shapes of road signs, in contrast with [**8, 18, 30, 50, 17, 49**] where only few road sign classes are considered (triangular, circular, or octagonal). However, as mentioned previously, not all road sign colors (hence not all road sign classes) are at present considered in the system described here; in fact black and white as said earlier. This is an aspect that will be investigated in future work.

What has been described thus far about the recognition module concerns its static part. The dynamic part has to do with the algorithm that actually performs recognition. This algorithm exploits correlation scores obtained from the template matching technique alluded to above at different levels during the search through the database resulting in a list of possible matches when these scores are sufficiently high. The fact that the algorithm is able to account for multiple possibilities in the results it delivers allows to avoid the main pitfall in tree-like structures such as the database used here, which is to take the wrong

[12]This idea of explicitly embodying geometrical transformations into a template using analytic calculations is not considered in other work on road sign recognition using learning approaches, where in fact such deformations are implicitely accounted for by the variety of the examples extracted and used for training.
[13]Note that the recognition approach presented here could eventually be used in other contexts where a finite set of objects is considered, such as in some classes of character recognition problems.

branch in the tree in ambiguous situations, ending up with an incorrect recognition result. An elegant approach to alleviate this will be described.

Further features of the road sign recognition system are that it incorporates feedback mechanisms as more information is gathered about a putative road sign, and a scene understanding module which allows to eliminate impossible situations. As regards performance, the system described here compares advantageously to the best results reported by others, considering the modular model-based approach used to attain those performances, and more importantly the large scale of the database used with more than 400 road signs that can be recognized. To the author's knowledge, no system has thus far been reported with as many different road signs. In the present project, no application specific hardware was developed, and the system runs on a general-purpose computer. Of course, further gains as regards processing speed could be made with such hardware.

The design methodology of the system presented in this thesis stands somewhat midway with respect to two major trends that currently assert themselves in computer vision. On one side are modelling and physics-based approaches, which are somewhat more restricted to theoretical investigations, and on the other side are learning approaches that are completely driven by data extracted from real images, and which find widespread use in practical applications. A good example of the fact that this work stands somewhat midway is color detection. Although a model will be used in the first place to predict object colors, these predictions will be further refined with data extracted from real images, which is somewhat similar to what learning does. However, here a model will be used to generate initial computed data, that subsequently guides the localization of real data, whereas in learning the real data are provided from examples extracted manually by an operator. In the problems tackled in this work, emphasis will first be placed on understanding the physical phenomena involved and the geometry of the situations in order to be able to use models. But models are not perfect, and at some point available data will be used to cope with those imperfections.

# CHAPTER 2

# Color Detection under Daylight: Physics-Based Model

## 2.1 Introduction

Using color for identifying objects is a difficult task in outdoor scenes illuminated by daylight. Some days are overcast, some others clear, an object may face the sun, or not, illumination changes depending on time of day, seasonal factors, *etc...* Thus a whole gamut of illumination conditions, which alter the color values output by the camera by appreciable amounts for the same object, has to be taken into account. Hence the problem consists in using color as a useful means of identification, even though the output values of a camera to a given colored object change due to illumination variations. In the present case, daylight and its variations will be considered, but other illuminants could also be considered such as artificial indoor illuminants. The problem just stated is an instance of the so-called *color constancy problem.* Its solution is important for naming colors consistently, and also in applications where color is an important attribute in the detection and identification of an object, as will be the case in the recognition of road signs to be considered later. To put this chapter and the next in perspective, a survey of the work related to color identification under varying illumination conditions will now be presented.

## 2.2 Survey

In order to use color as a useful means of identification, and since the measurements taken by a camera change as the illumination varies, it is legitimate to ask: What does it

mean for an object to have a color? Physically three basic quantities come into play in the formation of the color of an object as sensed by a device. These are: the spectral reflectance of the object, the spectral response curves (also called spectral sensitivities) of the sensor used, and the spectral distribution of the illumination impinging on the object. Concerning the reflectance of materials, it is the diffuse spectral reflectance function, also called body reflection function, which is of primary interest, because it is this type of reflection which is responsible for the colors of objects [**59**]. However, specular reflection (also called interface reflection), which conveys the same spectral distribution as the illumination source[1], cannot be completely neglected as will be seen later. Thus, to answer the question from a physics point of view, it is seen that the only factor attached to an object and partaking in color formation is its spectral reflectance function. Therefore, in order to somehow estimate this spectral reflectance or have a signature of it, and given that the characteristics of the sensor do not change and can be known (although this last point is not always exploited), one avenue that has attracted much interest is to somehow discard the illuminant from the color description, or at least get sufficient information to compensate for its variations. This is the essence of so-called color constancy approaches, many variants of which have been proposed. Common to all is to find color descriptors that are independent of the illumination. Some do this by making assumptions about the illumination and the reflectance of objects, and by solving sets of linear equations for extracting coefficients related to object reflectance. Others resort to some normalization rule on the components of the color vector.

Color constancy approaches can be divided into two subsets, which could be qualified as "instrinsic" and "extrinsic". In intrinsic approaches, emphasis is put on trying to evaluate the illuminant in order to discard its effects, and this, without resorting to a white reflectance standard or a color chart artificially placed at a known position in the scene, whereas in extrinsic approaches use of such a standard or chart is made. The idea underlying the use of a white reflectance standard, is that since its reflectance function is known and approximately constant over the visible spectrum, light reflected by it should be of approximately the same spectral content as the illumination (similarly as with specular reflection). Therefore, observing such a white standard with a color camera with known spectral response curves gives partial information about the spectral distribution of the illumination.

[1]This fact is at the basis of the methods using highlights for getting information about the illuminant [**60**].

The case of a chart containing several color patches of known spectral characteristics is an extension of this giving an increased number of constraints on the possible illuminant [**61**].

Intrinsic and extrinsic approaches to color constancy will be described in Sections 2.2.1 and 2.2.2. Learning approaches to color detection which have proved to be popular in practical applications where illumination varies will then be discussed in Section 2.2.3, followed by physical modelling in Section 2.2.4.

**2.2.1 Intrinsic Approaches.** Color constancy was already known to Helmholtz [**62**], but it was Land who revived interest into it with a famous series of experiments [**63, 64, 65**]. In an attempt to explain the results of those experiments, Land and McCann [**66**] have devised an intrinsic method, called the Retinex algorithm, in which three descriptors, one for each color channel, are computed at each pixel by way of a normalization rule. For a given pixel, the normalization of the output value in a given channel depends on the output values in the same channel at other pixels along several paths in the image. The major features of the Retinex algorithm, as presented in [**66, 67**], are that it is non-local and sequential. Horn [**68**] and Blake [**69**] have further explored the ideas of Land, and proposed parallel iterative algorithms requiring only local computations. However, their algorithms are of limited validity, and, as pointed out by Brainard and Wandell [**70**], can only work in a world illuminated by monochromatic light. The reason for this is that spectral integration over the visible spectrum cannot be neglected, thereby breaking the separability assumption at the basis of the model used by Horn and Blake.

Based on work by Sällstrom [**71**], Buchsbaum [**72**], and Brill [**73**], Maloney and Wandell [**74**] use linear models in which the surface reflectance of objects and the spectral distribution of the illumination are respectively represented as linear combinations of a small number of reflectance and light basis functions. The goal then is to solve for the coefficients in those linear combinations (in a least-squares sense). In this approach non-trivial questions arise regarding the choice of the basis functions and their number in order to give an adequate representation of real objects and illuminants. Moreover, it is not clear that a universal set of such functions exists, which are appropriate to deal with a wide class of situations. Interestingly, similar approaches are used in quantitative spectrometry and chemometry.

Forsyth [**75**] has proposed two alternative intrinsic approaches to color constancy, having their root in linear models. For an image captured under unknown illumination conditions, the color constancy problem is formulated as that of recovering an $(R, G, B)$ vector for each surface in an image as would be seen if these surfaces were illuminated by a canonical illuminant (a canonical illuminant is one that serves as a reference). Mathematically the problem consists in finding a unique $3 \times 3$ matrix which will allow to find the "canonical" values of $(R, G, B)$ from actual measurements of these quantities in the image. A constraint on the possible matrices for solving the problem under the given illumination conditions is that they must map the actual measurements into the gamut of possible colors under the canonical illuminant. The first approach is an extension of the work done by Maloney and Wandell [**74**], and requires to solve for 9 matrix coefficients. To reduce the complexity of the problem, the set of matrices is restricted to diagonal ones in the second approach. This is motivated by the von Kries coefficient law [**76, 77**], in which the system independently adjusts the gain of the receptors it employs in each color channel in a fashion thought to be analogous to adaptation in human vision. The matrix solving the problem is chosen as that which satisfies the aforementioned constraint, and which maximizes the volume of the convex hull of the actual color values measured from the image and mapped by that matrix into the canonical color gamut. This matrix can be seen as that which makes the image most colorful when mapped in the canonical color gamut. Finlayson [**78**], has developed an approach based on Forsyth's ideas working with 2-dimensional "perspective" color coordinates defined by $(r\prime, g\prime) = (R/B, G/B)$ and $2 \times 2$ diagonal matrices. An additional constraint on these matrices is added in order that they be related to real illuminants. This is done by considering a single standard reflectance taken to be a perfect white viewed under a large set of representative illuminants (within which is a subset of daylight illuminants). As in Forsyth's work, the solution to the color constancy problem is the diagonal matrix which makes the mapped image most colorful under this new constraint and that of Forsyth transposed to 2-D perspective coordinate, *i.e.*the matrix must map the actual measurements into the gamut of possible colors under the canonical illuminant. The rationale behind using perspective coordinates is to factor out color intensity from the color description of an object, which appears to cause problems in Forsyth's approach due to specular and shading effects. Unfortunately in Finlayson's results, the approach is tested

under the same conditions as Forsyth (a Macbeth color checker of matte samples is used with no shading effects or specularities). It is thus difficult to assess how well his approach truly performs under effects related to intensity.

More specifically as regards color constancy and the daylight model of Judd, MacAdam and Wyszecki [**79**] which will play a central role in the present work and will be described later, Ohta and Hayashi [**80**] have presented an approach based on linear models for recovering the full spectral distribution of the illuminant and spectral reflectance of objects. In their approach, two objects must be identified in two different images illuminated by different illuminants. The two objects do not need to be placed at the same location in the two images, but both must be present in the two images and must somehow be put into correspondence (identified) from one image to the other, a fact which limits the applicability of the approach. The Judd *et al.*model is used to constrain the possible illuminants. A similar approach in spirit is presented in [**81**], in which two color camera measurements of a given surface must be made under two spectrally different illuminations. However, here the outputs of the algorithm are not spectral quantities, but rather perspective coordinates as would be obtained if the surface were illuminated under a reference illuminant, similarly to Forsyth's work.

To the author's knowledge, no intrinsic solution has emerged as a standard technique and found widespread use in practical applications. Even in the highly idealized world in which the problem is usually studied, the so-called Mondrian world, which consists of flat scenes containing randomly distributed matte color patches (see Fig. 2.1), no satisfactory intrinsic solution exists. Nonetheless, there is a definite need for such intrinsic solutions to the color constancy problem, as color is more and more used in applications, and people face the problem of color changes under variations in the illumination. An intrinsic solution would offer great flexibility. There is certainly large interest here, and color constancy is an important and interesting problem. However, it may be that color constancy is too hard of a problem to solve, and that perhaps it is *not* the problem to solve, at least with regards to how intrinsic solutions have been envisaged thus far.

**2.2.2 Extrinsic Approaches.** Novak and Shafer [**61**] have presented an extrinsic method they called supervised color constancy in which a chart (in fact the Macbeth color checker) with a large number of colors samples of a priori known spectral reflectances is used

FIGURE 2.1. Example of a Mondrian scene with rectangular color patches.

in order to give enough constraints to calculate the spectral distribution of the illuminant to the desired degree of accuracy. Two approaches are examined for this. The first resorts to a linear model for representing the illuminant spectral distribution with Legendre polynomials as basis functions and uses least-squares to find the expansion coefficients. This is a parametric estimation approach, and in this case the spectral responses of the camera must be known. The other approach uses a neural network to evaluate the illuminant spectral distribution altogether. This is a non-parametric estimation approach, and here there is no need to know the spectral responses of the camera. These approaches have the advantage of being simple in principle, but are somewhat cumbersome to implement in certain applications, since the chart has to be put in the field of view of the camera. This is easily achieved in a fixed environment such as a laboratory or production line, but not for a mobile platform in a dynamic environment, because the chart should be at about the same location as the object to be detected (which of course is not known), so that the same lighting is effective. This is especially important outdoors when, for example, clouds are present. A cloud could be hiding the sun from an object, but not from the chart, and vice versa. Therefore, for practical reasons, it is to be expected that the method will not be as efficient in an outdoor environment.

Methods based on somehow discarding the effects of the illuminant will not be dwelled on any further. Rather, a second avenue will be explored, which is to find a description

that explicitly takes illumination changes into account. In this line of thought, two possible approaches can be considered, the first being learning, and the second physical modelling.

**2.2.3 Learning.** In the first approach, a classifier is trained to learn the colors of the objects of interest from images of typical scenes containing them and captured under different illumination conditions. In this process, a database of training objects manually retrieved by an operator from real images is constructed, and their pixel colors extracted. These colors form clusters in color space, which are used to mathematically define regions (which can be fuzzy) that are characteristic of the different colors of the objects to be detected. Once learning is completed, the pixels in an image of a scene potentially containing objects of interest are classified using the color regions, and image regions are formed using this classification by grouping neighboring pixels that have been assigned the same color label. This is the approach followed in [**41**] and [**44**].

This learning from examples approach is based on the assumption that after having been presented enough examples, the system should be able to recognize certain features; in the present case particular colors. In the context of color detection, this type of learning is somewhat unsatisfying because one is left with the impression of not having understood the physical processes behind the phenomenon. Indeed, learning from examples is a way of bypassing that understanding, and when the system fails, it is difficult to identify the reasons for failure. Such questions arise as: Did it fail because not enough examples were presented, should more be used? Or did it fail because the number of degrees of freedom (dimensionality) of the neural network used is too small to properly map the stochastic process (*e.g.*varying outdoor illumination spectrum)? Moreover, it requires human intervention to select training examples, and so for instance if the camera settings are changed, or even the camera itself, everything has to be done all over again. The database of training objects has to be thrown away, and a new one constructed using images captured with the new camera settings or camera. This is because the intrinsic information about the camera is also embedded in the training process, and is hardly separable. Constructing such a database represents a large amount of work, and this approach is thus not very flexible in these regards. This is definitely a limitation, and it appears difficult to improve on this situation

since some parameters are not separable[2]. The reason for this is that there is no theory on which such an approach can firmly rely, and so no parameters can be changed. The fact that a human operator must manually select the training examples from a large number of images is time consuming, but necessary to break the chicken-and-egg problem, since for the learning task, the objects must first be recognized and identified as objects of interest. Only then can their colors be extracted[3]. However, there are not just disadvantages to learning since it certainly takes more factors into account than model-based approaches, but in an implicit and less well understood manner. Depending on the complexity of the learning system, *i.e.*dimensionality of the neural network and the number of training images, more or less factors will be taken into account.

**2.2.4 Physical Modelling.** The paradigm of physics-based vision makes use of existing physical theories for different vision applications (see [**82**] and the many articles collected therein); it is a model-based approach. Its most prominent features are to stress the importance of understanding physical processes and exploit the physical structure involved in a given imaging situation, thereby gaining a firm grasp on the physical parameters partaking in these processes and their interrelation via a model. Physical modelling is a means of compressing information and helps to separate specifics (*i.e.*parameter values) from generalities (model structure), which cannot be achieved in learning as alluded to above. In physics-based vision, as pointed out by Klinker [**83**] "Since the control parameters of the algorithm are related to physical scene properties, there is an objective criterion for setting these values". Also, since physics-based vision is a modelling approach, the parameters involved can be changed, and in principle the model's predictions can be adapted to new situations, which is a more difficult task in learning[4]. Nevertheless, model-based approaches also have their limitations. In modelling, as opposed to learning, simplifying assumptions about the real situation are always made to reduce the complexity and render the problem tractable. Hence, one should be aware that not all factors are and can be taken

[2]In this particular case, one could think of pre-processing the input training data prior to feeding it to the neural network in order to compensate for the effect of the camera spectral responses. However, this pre-processing would also need to be performed in real-time when the neural network is used in its actual classification application.

[3]These problems can be overcome if the starting point is a model, as will be shown in next chapter.

[4]For instance, in the case where the camera used is changed, then only parameters related to the camera need to be changed in the model.

into account in such paradigms. Models are at best approximations of the physical world, and are thus by no means perfect.

It should be noted at this point that many of the color constancy approaches that were reviewed above are physics-based. Recall that in such approaches, the objective is to extract color descriptors that are independent of the illumination based on a model. The work to be presented in this chapter takes a different point of view. Indeed, a stochastic physical model will rather be used to make predictions about the colors of an object under daylight and its variations (any other illumination source could be considered as well, but in view of the application of the approach to the detection of road signs, the work concentrates on daylight). Now, as said in the previous paragraph, models have limitations, and to cope with this, these predictions will be calibrated using real data. This will be the subject of next chapter.

## 2.3 Proposed Method and Overview

In this chapter, a physical model for predicting the colors of specific objects assumed to have known body spectral reflectances will be developed for daylight illumination. The predictions will then be applied to the detection of these objects. These predictions are founded directly on the color formation equations, which serve to model the color of an object as sensed by a camera under a given illuminant[5].

While assuming known body spectral reflectances may appear somewhat restrictive at first sight, in many applications the objects to be detected are known beforehand, and so their body spectral reflectance can be measured (using measured reflectances has also been exploited in other contexts [**61, 81, 85**]). This is the class of problems considered here, and an example of this situation is the problem of finding occurences of traffic signs in color images of road scenes which will be considered in Chapter 5. In fact, samples of road sign colors will be used in this and the following chapters, and the results will be readily applicable in Chapter 5.

The color of an object sensed by a camera is related to how its environment is illuminated. In the present context, daylight, which obviously changes in time, is considered.

[5]The work in this and the next chapters has been presented in condensed form in [**84**].

Now, if a model is to be developed for color formation under daylight illumination conditions, then the question is: Is it possible to predict how daylight varies; is there a model for this? Obviously, no control can be exerted over daylight, and this likely explains the fact that approaches based on the color formation equations have been neglected and learning methods preferred in outdoor applications of computer vision. To the author's knowledge, nobody has attempted to exploit those equations for detection purposes in an outdoor environment such as will be proposed here.

A considerable amount of experimental work has been carried out to characterize the spectral composition of daylight illumination [**86, 87, 88, 89, 90, 91, 92**]. This work has been synthesized by Judd, MacAdam and Wyszecki [**79**] through a semi-empirical model (*i.e.*a model founded on experimental data) which allows to calculate a whole range of relative spectral power distributions that are representative of the different illumination conditions found outdoors during the day. This semi-empirical model will play a central role in the present work, as it will be used in conjunction with the color formation equations to predict the colors of objects under daylight.

As mentioned in the previous section, the work of Judd *et al.*has been used in color constancy approaches by some groups in computer vision [**80, 81, 78**], but in this thesis another point of view is adopted. The approach developed here is somewhat a compromise between methods based on estimating the illuminant to eliminate its effects (*i.e.*discarding it to some point) and learning approaches. As in the former, which almost always rely on physical properties of the color formation process, the present approach uses a physical model of color formation. However, as in learning, no attempt will be made at determining the illuminant or the spectral reflectance of objects. Instead, all the expected color variations will be taken into account as a whole. But in opposition to learning, which uses examples to achieve this, a physical model will be resorted to instead to make *predictions* about the different values of the color vector for a given camera and material under the various illumination conditions considered. The equivalent of choosing learning examples here is to assume that the objects are of known spectral reflectances. This is much less of a burden than having to choose several examples of the same color because it requires less work, and all the information is condensed in a single well defined physical quantity. Also, the fact

that the physical processes are clearly understood and embodied into the approach appears to be much less brute-force than learning.

With respect to approaches that attempt to recover the spectral reflectance of objects, the present one is less general. However, the former methods have not found widespread use in uncontrolled environments, and require a large amount of on-line computation, which makes them less interesting for many practical applications. The present approach is devoted to applications requiring fast color identification. This explains why no attempt is made at determining the details of which daylight conditions prevail, which would require real-time parameter estimation and would simply be too time consuming in such applications. The main goal is that the detection be robust to the variations of daylight and gives accurate results. The advantages of the present approach are that it is flexible as it could easily be adapted to other materials and illumination conditions than those considered here, and it allows a fast implementation via a look-up table, which is important in practical applications.

Specifically, the detection approach developed in this chapter is along the following lines. In next section, color formation will be discussed, and the color formation equations presented. Each term entering those equations will be described. These are: the spectral reflectance coefficient of the materials out of which objects are made, the spectral response curves of the color camera, and the spectral distribution of the illumination, here daylight. The first two can be measured, and are discussed in Sections 2.4.1 and 2.4.2. Note that although it is body reflection which is responsible for the colors of objects, specular reflection will also be accounted for in the model (which is not the case in color constancy approaches which have been exclusively restricted to the study of mate colors). Specular reflection cannot be ignored since some amount is always present. As concerns daylight, the features relevant to the present work of the semi-empirical model of Judd, MacAdam, and Wyszecki [**79**] for describing its spectral distribution will be presented in Section 2.4.3. Using this, a model giving a prediction of the colors of an object under all possible illumination conditions found outdoors is developed in Section 2.4.4. Model uncertainties are introduced in those predictions in Section 2.5. These predictions and their uncertainty form a set of points in color space associated with a particular material. From such a set, color space region can be determined, with which detection can be performed. Detection results

using this model-based approach will be presented in Section 2.6. The approach performs efficiently in many cases, but in others it must be made more robust to noise and in terms of the accuracy of the detected image regions. This can be traced back to the limits of the model, which can only be an approximation for describing a real situation (modelling errors).

Having reached the model's limitations, real data will be resorted to at this level in order to calibrate the model-based color regions. This will be described in next chapter. In the present and next chapters, although a small number of colors is considered, the challenge is to be able to efficiently detect them under daylight and the wide variations it presents.

## 2.4 Color Formation and Daylight

When an object is illuminated, the physical process responsible for its color as sensed by a detector is governed by three factors: the reflectance properties of the material with which the object is made, the spectral responses of the sensor used (a color camera in the present case), and the spectral distribution of the illumination. Mathematically, this is modelled by the so-called *color formation equations* [6]

$$\mathbf{C} = \begin{pmatrix} C_R \\ C_G \\ C_B \end{pmatrix} = cst \cdot \begin{pmatrix} \int \rho(\lambda) L(\lambda) s_R(\lambda) d\lambda \\ \int \rho(\lambda) L(\lambda) s_G(\lambda) d\lambda \\ \int \rho(\lambda) L(\lambda) s_B(\lambda) d\lambda \end{pmatrix}, \tag{2.4.1}$$

with $\mathbf{C}$ being the color vector composed of the values $C_R$, $C_G$, and $C_B$ output by the sensor in the $R$, $G$, and $B$ channels, $\rho(\lambda)$ the spectral reflectance of the colored object under view, $L(\lambda)$ the spectral distribution of the light illuminating the object, and $s_R(\lambda)$, $s_G(\lambda)$, and $s_B(\lambda)$ the spectral responses of the camera in each color channel. Integration is carried over part of the spectrum where the camera has non-zero responses. The common constant factor in front of the integrals embodies geometrical factors (such as shading effects), and the fact that the spectral distribution of the illumination and the spectral sensitivities of the camera are usually specified as relative values[7]. A quantity related to the color vector that

[6] The interested reader will find a rigourous derivation of those equations in Appendix C, along with some radiometric terminology and details on the physical units involved.

[7] Formally $\rho(\lambda)$ and $s_R(\lambda)$, $s_G(\lambda)$ and $s_B(\lambda)$ are unitless quantities, whereas $L(\lambda)$ is specified in Watts per square meter per steradian; see Appendix C.

will be of interest in the sequel and which is often used in color work is the *color intensity*[8] defined as

$$I = C_R + C_G + C_B. \tag{2.4.2}$$

To be able to use the color formation equations to predict the colors of objects for detection purposes as will be done in the following, the three factors participating in color formation need to be discussed.

**2.4.1 Material Reflectance.** In the present work, optically inhomogeneous materials are considered [9]. These consist in a dielectric medium comprising the bulk of the material, in which colorant particles are embedded (paints and colored plastics are examples). Such materials are known to produce two types of reflections: body and interface [**59, 83**]. Body reflectance is responsible for the colors of objects through the colorant particles. Interface reflection, governed by Fresnel's equations, accounts for the glossiness of a surface due to the change in index of refraction at the interface between the surrounding medium (assumed to be air in the present case) and the material's dielectric bulk.

When light propagating in the surrounding medium is incident on such a material, because of the change in the refraction index at the interface, part of the light is reflected back (interface reflection), and part is transmitted through the material. In the material, transmitted light may be scattered or selectively absorbed by colorant pigments. A fraction of the scattered light eventually reaches the interface again and exits the material (body reflection). With respect to the light that entered the medium, the spectral content of this exiting light has been modified through selective absorption by the pigments, which is responsible for the characteristic color of the material. Body reflection can to a good approximation be considered diffuse, *i.e.*light is equally likely exiting in all directions, since scattering is responsible for the light that exits the material. Moreover, apart from the

[8]One should be cautious about the different meanings loosely assigned to the word intensity in the literature. The first is related to the concept of the intensity of a color mathematically expressed as in 2.4.2. The second refers to the intensity of the illumination as defined in optical physics which is related to the strength of the illumination in energy units ($Watts/m^2$) (more correctly, this is called irradiance). The first meaning will almost be exclusively used in this thesis, unless otherwise stated.

[9]More precisely, the materials to be considered later in this work are those used in the manufacturing of road signs (samples of such materials were kindly provided by the Ministère des Transports du Québec), and consist of flat colored adhesive pellicles made of plastic and small flat aluminum plates coated with paint. These materials will be the ones used in this chapter and the next. However, the discussion is not solely restricted to such materials.

usual cosinusoidal foreshortening projection factor, which determines the amount of light impinging on the surface, body reflection is independent of the angle of incidence of the illuminating beam, that is, the bidirectional reflectance distribution function (BRDF) does not depend on the angle of incidence, see Appendix C.

Interface reflection is dependent on the angle of incidence, and occurs mostly around the ideal specular direction; mostly but not entirely because surfaces are usually not prefectly optically flat, *i.e.*have a certain degree of roughness [**93, 94, 95**] [10]. Interface reflection is responsible for highlights, and is governed by the Fresnel reflection coefficient, and for dielectrics that are of interest in color appearance (paints, plastics), it is usually approximately constant over the visible spectrum [**59**], because the refractive index of those dielectrics does not depend on wavelength in any significant way in the visible range. Typical variations are only of a few percent in the visible spectrum for most common materials of interest in color appearance, and so interface reflectance is usually assumed constant [11]. This is why interface reflection is often considered to be of the same color as the illuminant, a fact exploited in highlight analysis [**60**].

Reflection as just described accounts for many common materials, such as most types of paints and varnishes, paper, ceramics, and plastics. In fact any material composed of an isotropic dielectric of nearly constant index of refraction in the visible spectrum, and in which colorant pigments are embedded, is covered by this discussion. However, homogeneous materials such as metals and crystals require other treatments [**96, 97, 98**], but these will not be considered in this work.

The spectral reflectance $\rho(\lambda)$ of a material as considered here can be expressed as

$$\rho(\lambda) = m_b\rho_b(\lambda) + m_i\rho_i(\lambda), \tag{2.4.3}$$

where $\rho_b(\lambda)$ and $\rho_i(\lambda)$ give the spectral dependence of body and interface reflectance, and the coefficients $m_b$ and $m_i$ determine the amount of each type of reflection. The spectral

[10] To be more precise, when light is incident at a point on an interface and is reflected there, the angle of reflection is equal to the angle of incidence with respect to the normal to the interface at that point (microscopic normal). However, the surface of most materials is not optically flat, and as a result, macroscopically, the light reflected from a small portion of interface tends to be more or less scattered around the macroscopic normal of that portion, which may differ from the individual microscopic normals. The amount of scatter depends on the degree of roughness of the interface.

[11] For instance, acrylic plastic, which is often used in paints, has an index of refraction of 1.485 at $400nm$ and 1.505 at $700nm$, leading to Fresnel reflection coefficients of 3.8% and 4.1% at normal incidence [**59**].

body reflectance $\rho_b(\lambda)$ can be measured with a spectrophotometer using the appropriate geometry [**77**]. Measurements across the visible spectrum for a few material samples are shown in Fig. 2.2.

Concerning the spectral dependence of interface reflection, since the index of refraction generally varies with wavelength, this is also the case for $\rho_i(\lambda)$. However, as said previously, interface reflectance can be assumed constant for the materials of interest in color appearance. As a consequence of Eq. 2.4.3, body color is always more or less "diluted" towards achromatic colors by interface reflection, and is the reason why interface reflection cannot generally be neglected in color analysis. Here the constant value of $\rho_i(\lambda)$ is chosen to be 1.0. Such a choice can be made, because as will be seen later, only relative values are of relevance here. Regarding the coefficients $m_b$ and $m_i$, they depend on the geometry of the light source (*e.g.* point source, extended source, *etc...*), its position with respect to the material sample, and on the viewing geometry; $m_i$ also depends on the roughness of the material considered. Since these factors depend on the particular situation considered, a general expression for the relative weight of $m_b$ versus $m_i$ can hardly be given. Suffice it to say at this point, that interface reflection may be present, in which case it must be accounted for. As a result, the color formation equations can be separated into two components, one pertaining to body reflection, and the other to interface reflection[12].

**2.4.2 Camera Spectral Responses.** The second quantity entering the color formation equations concerns the camera spectral responses $s_R(\lambda)$, $s_G(\lambda)$, and $s_B(\lambda)$. For the present purposes, these have been measured experimentally using a monochromator. In performing such measurements, the problems associated with the non-ideal behavior of real color cameras must be accounted for in order to do the appropriate corrections. These problems have been thoroughly investigated by Shafer, Novak, and Willson [**99**].

A technique for measuring the spectral responses of color cameras, which takes the aforementioned problems into account, has been developed. Details on this have been presented in [**100**], and are deferred to Appendix A. Such measurements are shown in Fig. 2.3 for the camera settings that were used (the camera was a Panasonic GP-US502 3-CCD micro head color video camera).

[12]Note that Eq. 2.4.3 is an approximation, since generally wavelength and geometry cannot be decoupled as is assumed (for the interface part, recall the Fresnel equations). However, such an assumption is of minimum impact in real images [**59**].

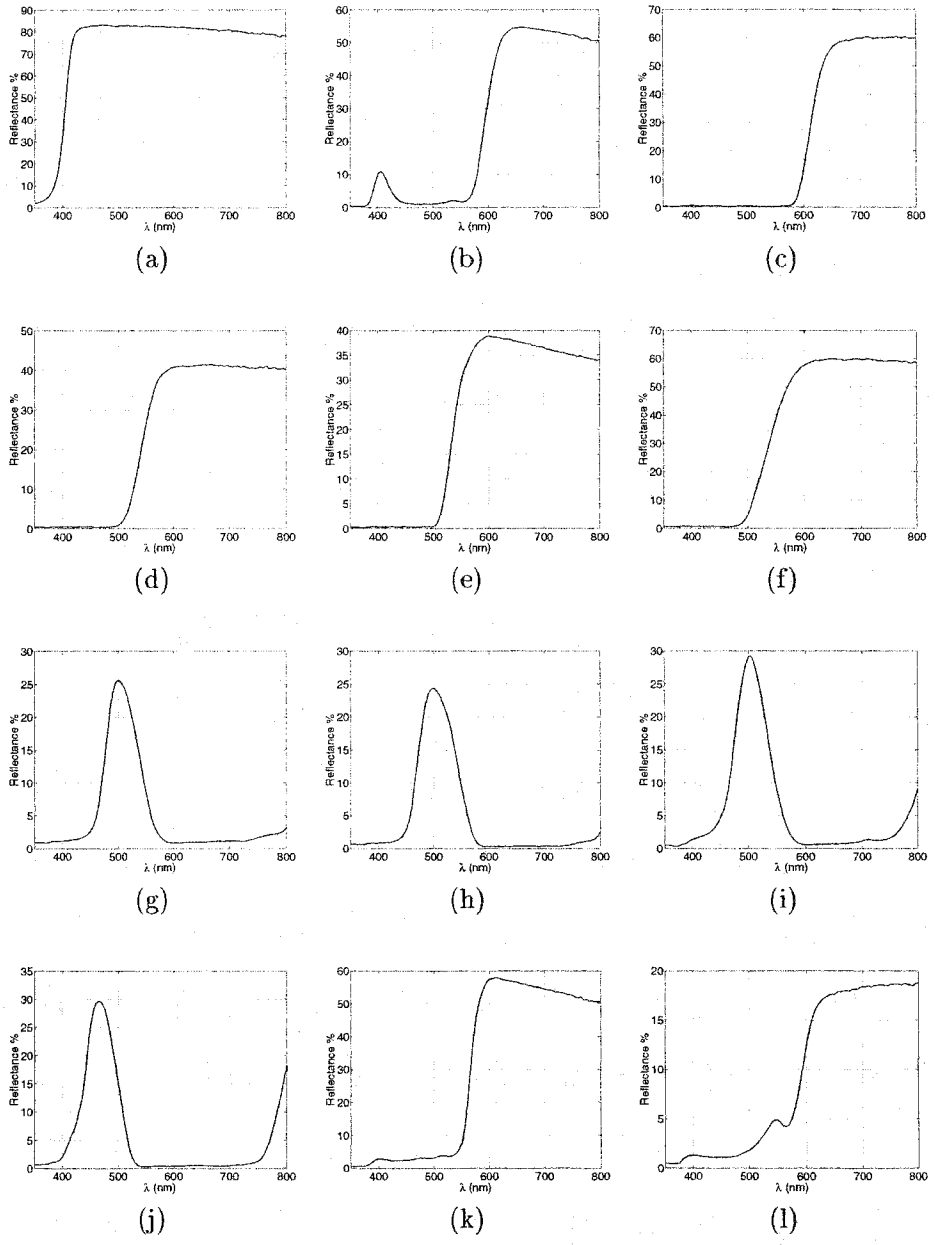



FIGURE 2.2. Graphs of spectral reflectance factors for different material samples plotted as percentage of reflectance versus wavelength. Note that the scale on the ordinate is not the same for all graphs. In (a) is the spectral reflectance for a material having a white appearance, (b), and (c) correspond to red materials, (d), (e), and (f) to yellow materials, and (g), (h), and (i) to green ones. Graph (j) depicts the spectral reflectance of a blue sample, whereas, (k) and (l) correspond to orange and brown samples respectively. Notice that for orange and brown, the transitions (rising slope) of the reflectance occur in approximately the same range of wavelengths, confirming that brown is indeed a dark orange, a fact often mentioned in the literature.

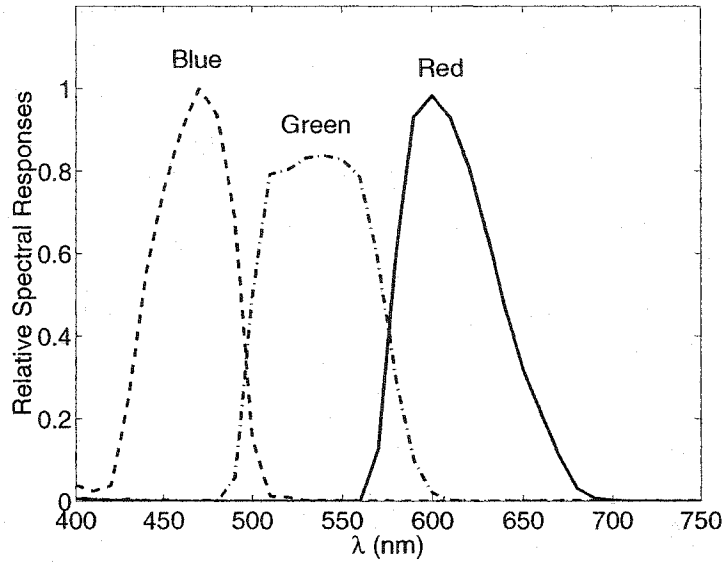



FIGURE 2.3. Measured spectral responses for the camera with an objective.

**2.4.3 Daylight.** In this work, the illumination considered is daylight. To make the terminology clear, by daylight is meant light composed of direct sunlight and/or sunlight scattered by the atmosphere and/or clouds.

Based on experimental data, Judd *et al.* [**79**] have developed a semi-empirical model for reconstituting spectral power distributions representative of the different phases of daylight. The parameter for specifying a given phase is the so-called correlated color temperature $T_c$ [**77**]. In order not to hamper the text with details, only the relevant calculations pertaining to the Judd *et al.*model will be presented here. Some further details, as well as the definition of the correlated color temperature and its numerical values related to typical daylight illumination conditions are given in Appendix B.

Given a correlated color temperature, specified in degrees Kelvin ($K$), the spectral distribution of daylight is expressed as

$$S_D(\lambda;T_c) = S_0(\lambda) + M_1(T_c)S_1(\lambda) + M_2(T_c)S_2(\lambda). \tag{2.4.4}$$

Here, $S_0$, $S_1$, and $S_2$ are respectively the mean, and the first and second principal components of the experimental data. They are illustrated in Fig. 2.4 (numerical values are given in Table B.1 of Appendix B). $M_1(T_c)$ and $M_2(T_c)$ are coefficients that depend on the

correlated color temperature through the chromaticity coordinates $(x_D, y_D)$ [**79, 77**] (the reader is assumed to be familiar with the CIE 1931 $(x, y)$-chromaticity coordinates; if not [**77, 101**] are good references to consult),

$$M_1 = \frac{-1.3515 - 1.7703x_D + 5.9114y_D}{0.0241 + 0.2562x_D - 0.7341y_D},$$
$$M_2 = \frac{0.0300 - 31.4424x_D + 30.0717y_D}{0.0241 + 0.2562x_D - 0.7341y_D}. \tag{2.4.5}$$

(2.4.6)
$$x_D = \begin{cases} -4.6070\frac{10^9}{T_c^3} + 2.9678\frac{10^6}{T_c^2} + 0.09911\frac{10^3}{T_c} + 0.244063 & \text{if } 4000K \leq T_c < 7000K \\ -2.0064\frac{10^9}{T_c^3} + 1.9018\frac{10^6}{T_c^2} + 0.24748\frac{10^3}{T_c} + 0.237040 & \text{if } 7000K \leq T_c \leq 25000K \end{cases}$$

The second chromaticity coordinate is related to the first by

$$y_D = -3.000x_D^2 + 2.870x_D - 0.275. \tag{2.4.7}$$

This last equation defines the "CIE daylight locus" in the CIE 1931 $(x, y)$-chromaticity diagram.

Relative spectral power distributions given by the daylight model for different correlated color temperatures ranging from $4000K$ to $25000K$ are illustrated in Fig. 2.5. Note that as the temperature increases, the curves get more and more crowded, this being due to the dependence in powers of $1/T_c$ of the chromaticity coordinates (Eq. (2.4.6).

To get an idea of how the correlated color temperature and the Judd *et al.*model in general connect to real daylight illumination conditions, the reader is referred to Appendix B.

**2.4.4 Predicting Object Colors under Daylight.** Combining the color formation equations and the model of Judd *et al.*allows to predict the response of the camera to an object with spectral reflectance $\rho(\lambda)$ illuminated by daylight of correlated color temperature $T_c$,

$$\mathbf{C} = \begin{pmatrix} C_R \\ C_G \\ C_B \end{pmatrix}(T_c) = cst \cdot \begin{pmatrix} \int \rho(\lambda) S_D(\lambda; T_c) s_R(\lambda) d\lambda \\ \int \rho(\lambda) S_D(\lambda; T_c) s_G(\lambda) d\lambda \\ \int \rho(\lambda) S_D(\lambda; T_c) s_B(\lambda) d\lambda \end{pmatrix}. \tag{2.4.8}$$

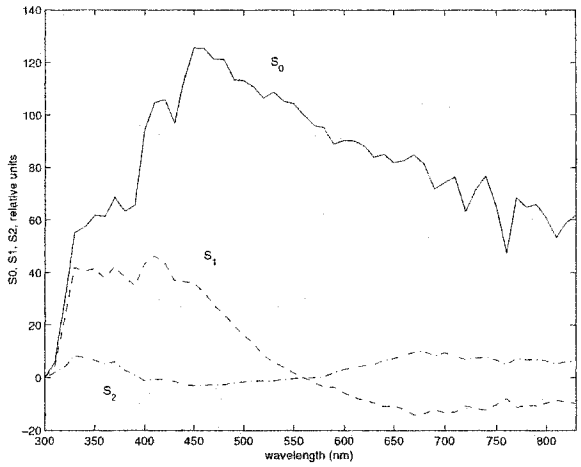



FIGURE 2.4. Showing the mean, $S_0$, and characteristic vectors $S_1$, and $S_2$. Notice that the characteristic vectors are zero at the normalizing wavelegth $\lambda = 560\,nm$, and that the mean is set to a value of 100. The data is available on the CIE web sites at http://www.ping.at/cie/, or http://www.hike.te.chiba-u.ac.jp/ikeda/CIE/.

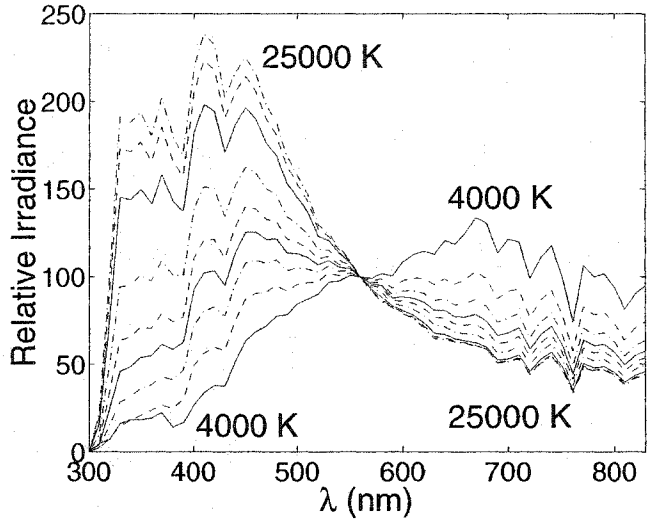



FIGURE 2.5. Typical daylight spectral distributions for different correlated color temperatures plotted in relative units versus wavelength. Below the normalizing wavelength at 560 $nm$, the lowest curve corresponds to 4000 Kelvins ($K$) and the top to 25000 $K$. The curves in between correspond to intermediate temperatures (5000, 6000, 7000, 8000, 10000, 15000, and 20000$K$ respectively). The situation is reversed above the normalizing wavelength.

Use of normalized color coordinates $\mathbf{c} = (c_r, c_g, c_b)$ is made to eliminate the common constant factor to all three components of the color vector which is not of interest here[13]. These are given by

$$\begin{aligned} c_r &= \frac{C_R}{C_R + C_G + C_B} = \frac{C_R}{I}, \\ c_g &= \frac{C_G}{C_R + C_G + C_B} = \frac{C_G}{I}, \\ c_b &= \frac{C_B}{C_R + C_G + C_B} = \frac{C_B}{I}. \end{aligned} \tag{2.4.9}$$

Such coordinates are of common usage in applications involving color. Their use is usually justified by saying that they are invariant under changes of illumination levels, which is somewhat imprecise for an explanation, and is true only if the changes are constant throughout the illumination spectrum[14]. More precisely, the factors that are eliminated by doing such a normalization are: geometrical factors related to the imaging geometry, and the absolute level of the spectral distribution of the illumination (illuminance). These factors are contained in the common constant in front of the integrals in Eqs. 2.4.1 and 2.4.8. Due to the constraints $c_r, c_g, c_b >= 0$, and $c_r + c_g + c_b = 1$, use of such coordinates also has the advantage of reducing the dimensionality of the space for describing colors, their sum amounting to one, only two are independent. When plotted in 3-D space, a point specified by these coordinates falls into the planar triangle illustrated in Fig. 2.6 (a), in which an orthogonal system of coordinates $(p, q)$ with origin at the center of the triangle can be defined. Fig. 2.6 (b) shows the axes of this coordinate system. The correspondence between $(p, q)$ and $(c_r, c_g, c_b)$, which can be deduced from simple geometry, is given by

$$\begin{aligned} p &= \frac{c_g - c_r}{\sqrt{2}}, \\ q &= \frac{3c_b - 1}{\sqrt{6}}. \end{aligned} \tag{2.4.10}$$

[13]This factor is studied in shape from shading work, see Appendix C.

[14]Although illumination changes found outdoors are much more complicated than just a multiplication by a constant factor (it is in fact the spectral composition of daylight that changes for the various illumination conditions), this type of invariance is relevant.

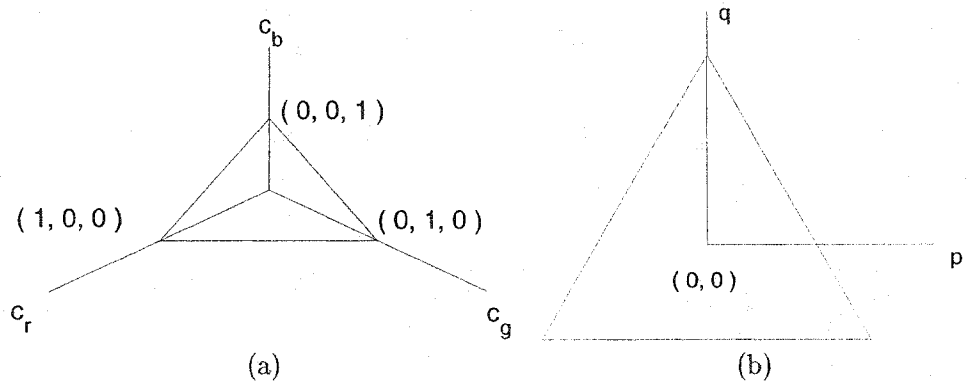



FIGURE 2.6. (a) Plane in 3-D space in which $(c_r, c_g, c_b)$ points are located. (b) Orthogonal coordinate system $(p, q)$ attached to this plane. In both (a), and (b), the region within the triangle is that of real colors.

From Eqs. 2.4.3 and 2.4.8, the color vector can be separated into two components, one pertaining to body reflection, and the other to interface reflection,

$$\mathbf{C} = m_b \mathbf{C}_b + m_i \mathbf{C}_i. \tag{2.4.11}$$

Here, $\mathbf{C}_b$ and $\mathbf{C}_i$ have the same expressions as $\mathbf{C}$ in Eq. 2.4.8, but with $\rho(\lambda)$ replaced by $\rho_b(\lambda)$ and $\rho_i(\lambda)$ respectively. Regarding the factors $m_b$ and $m_i$, when use of normalized coordinates is made, only their ratio matters. Since body reflection is always present (even if completely masked by interface reflection) $m_b \neq 0$, and the ratio $\kappa = m_i/m_b$ can be introduced. For normalized coordinates, the vector given by

$$\tilde{\mathbf{C}} = \mathbf{C}_b + \kappa \mathbf{C}_i \tag{2.4.12}$$

is equivalent to that of Eq. 2.4.11. It is an easy exercise to show that the normalized color of an object lies on the line joining its normalized body color $\mathbf{c}_b$ and the normalized color of the illuminant $\mathbf{c}_i$ (due to interface reflection) in the color triangle [**60**], the position along this line being determined by $\kappa$.

Since the goal here is to find objects with specific colors in an image, some bound on $\kappa$ must be set, otherwise achromatic colors will also be detected. As pointed out by Shafer [**59**], there is no obvious way to scale the factors $m_b$ and $m_i$ (and thus determine $\kappa$).

Their relative amounts depend on the particulars of the illuminating and viewing conditions and on the optical roughness of the surface (as concerns $m_i$). Values of $\kappa$ as high as of the order of $10^4$ can be attained when an object is seen in the specular direction of the reflected rays from the source (assuming a point source and an optically flat material). With such values, interface reflection completely masks body reflection. In the present case, it will be assumed that the objects are never seen from the specular direction. However, this does not mean in practice that interface reflection is never present. In fact, due to the optical roughness of materials, and since an extended light source is considered (the sky), some amount of interface reflection, which plays the role of a bias, must be accounted for. For this, a small maximal value for $\kappa$ is chosen, which is equal to 2% of the maximum of the body spectral reflectance for a given object, $\kappa_{max} = 0.02 \cdot \max \rho_b(\lambda)$[15].

Traces of color predictions made by the model are shown in Fig. 2.7 (a) for the material with red appearance having the body reflectance plotted in Fig. 2.2 (b). The trace of red stars gives the predictions when only body color is considered (no interface reflection, $\kappa = 0$), whereas that of black crosses is for $\kappa = \kappa_{max}$. In both cases, the points correspond to daylight illuminants having temperatures ranging from 4000 $K$ to 25000 $K$ in steps of 1000 $K$. Traces of predictions with $\kappa = 0$ are illustrated in Fig. 2.7 (b) for some of the samples appearing in the chart in (c) having various color appearances (red, orange, brown, yellow, green, and blue). Also of importance in the sequel are the colors of the different phases of daylight which appear at the center of the color triangle in (b) and are labelled "achromatic" (these could also be called whites). These are obtained by setting the spectral reflectance constant to 1 throughout the spectrum.

## 2.5 Introducing Uncertainty

As such, the predictions just presented cannot readily be used to do color detection. Actual measurements cannot be expected to fall exactly on the traces of predictions, since sources of variations in the different quantities entering the color formation equations are always present and must be accounted for, and because a model can only be an approximation for describing a real situation. Therefore, a certain scatter from the predictions is to be expected. Two sources of variations can be considered: those pertaining to measurements,

[15]This small value was determined empirically. It is sufficient to account for the interface reflectance typically seen on the color samples considered.

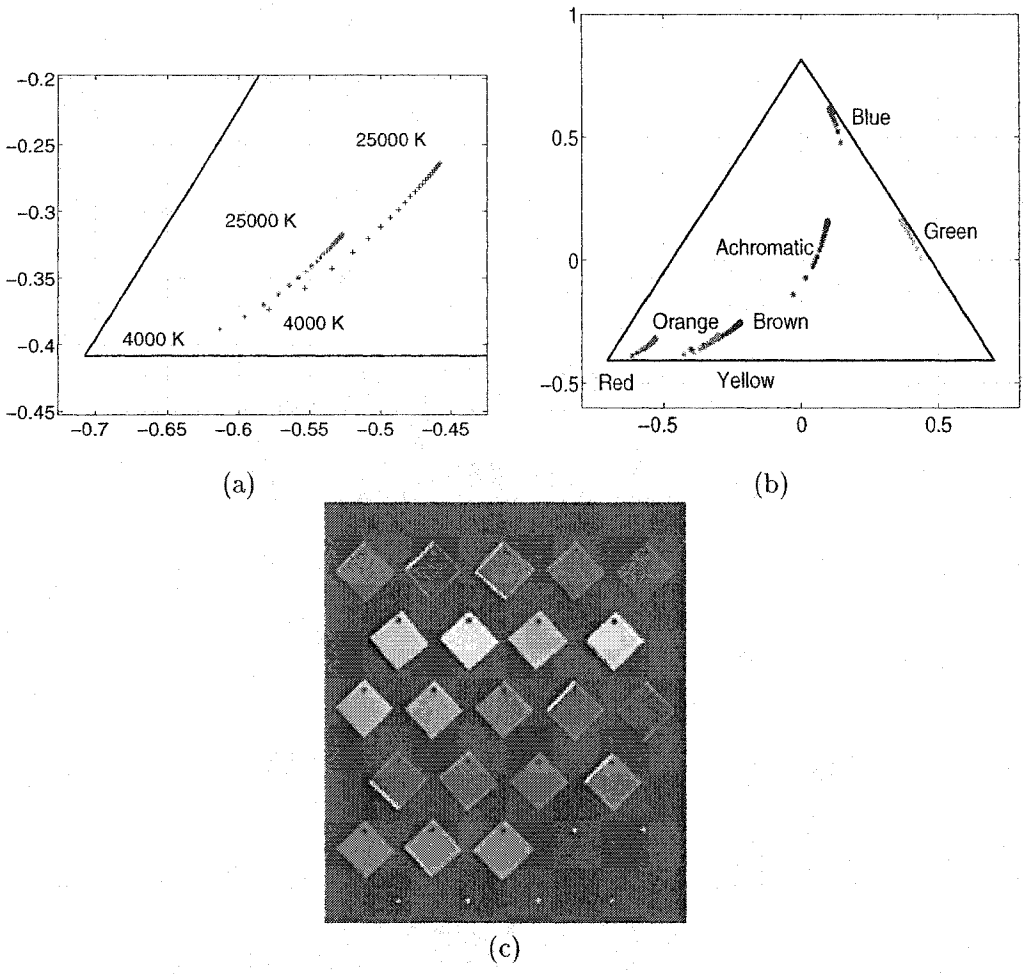



FIGURE 2.7. (a) Predicted body color points under daylight for a material with red appearance. (b) Traces of predicted body colors (with $\kappa = 0$) for some samples of the chart in (c) having various colors. The colors of the traces are choosen to correspond to the materials they represent, except for the magenta, and black traces which correspond to orange and brown materials. In (c), the samples correspond to different road sign colors. Although perhaps not apparent in the figure, the samples for a given color are of different varieties. Some contain microspheres and other microprisms which serve as retroreflectors. By numbering the samples from left to right and top to bottom (the leftmost orange sample is then number 10), samples 1 to 5, 7, 8, 10 to 15, and 17 to 20 contain microspheres, while samples 6, 9, 16, and 21 contain microprisms. Microshperes and microprisms are of no concern in this work because the sun or the sky is the light source, hence the retroreflection being in its direction. Only body reflection is of interest. The white edges seen at the border of some of the samples (1, 2, 13, 15, and 18) is due to the fact that these samples were not colored at those borders.

which can be estimated and embodied into the model as statistical variations, and those due to modelling errors (these will be taken care of by the calibration approach to be discussed in next chapter).

Errors in the measurements of the material body spectral reflectance were determined by calculating the standard deviation of measurements made on several samples of the same material. In the worst case, this was on the order of 7%. For the camera spectral responses, the only source of errors lies in the measurements, and in the worst case these were on the order of 11%. Regarding the daylight model, it too is subject to variations. As Judd *et al.* put it, their model is a recommendation for "satisfactorily" representing typical daylight spectral distributions. Unfortunately, in their study, no analysis is provided on the typical errors that can be expected when the model's predictions are compared to experimentally measured spectral distributions. However, from some of the experimental data they provide and references they cite, 20% variations are not unusual. This will be the value taken here. An obvious source of variations in the model is that only the first two principal components of the experimental data are considered. Also, deviations due to weather conditions, which are never exactly the same, and others due to climatic changes since the measurements used by Judd *et al.* were made probably play a role. Another factor of importance for the spectral composition of daylight under cloudy conditions is reflection from the ground as pointed out by Middleton [**91**]. All the previous points regarding daylight will be considered as modelling errors.

To model the measurement errors on the quantities appearing in the color formation equations, Monte Carlo simulations are used as suggested by Ohta and Wyszecki [**102**]. The error $\Delta X(\lambda)$ on a given spectral quantity $X(\lambda)$ is modeled as a uniform random variable (RV) of specified percentage value. The resulting varied quantity is given by

$$\begin{aligned} X'(\lambda) &= X(\lambda) + \Delta X(\lambda), \\ \Delta X(\lambda) &= \delta_X(\lambda) \cdot X(\lambda), \end{aligned} \tag{2.5.1}$$

where $\delta_X(\lambda)$ is a uniform RV in $[-\delta_{X,0}, \delta_{X,0}]$, $\delta_{X,0}$ being the percent deviation specified (*e.g.*7% for the material samples). By combining a large number of such variations for each spectral quantity in the color formation equations (which is the essence of a Monte Carlo simulation), a cloud of points around each trace of predictions is obtained, which serves to

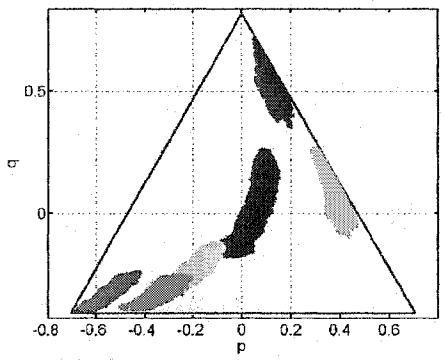



FIGURE 2.8. (a) Clouds of points obtained with Monte Carlo simulations for all the samples shown in Fig. 2.7 (c). The clouds for samples having the same color are grouped to form a unique cloud of points. Blue, green, red, and yellow sample color points are illustrated as such, whereas brown ones are illustrated in cyan, orange ones in magenta, and achromatic ones in black.

model the expected color variations resulting from measurement and modelling errors. A total of 3000 variations for each point on a trace of predictions was considered in the Monte Carlo simulations.

The clouds of points for the samples having the same color appearance, *e.g.*red, are grouped together to form a unique cloud associated to that color. Fig. 2.8 illustrates these clouds of points for the different color samples considered here (see Fig. 2.7 (c)).

## 2.6 Model-Based Color Detection

The clouds of points obtained in the previous section can serve as the basis for a simple detection technique under daylight illumination conditions. It consists of first determining color space regions from the clouds for each of the colors considered. These regions are then used to label the pixels in an image. The details follow.

**2.6.1 Color Space Regions.** Clouds of points corresponding to different colors which overlap are first separated with an approach based on Fisher's linear discriminant method [**103**], except for the brown and orange clouds for which the overlap is of a different nature[16]. Separating brown and orange requires a different treatment. A scheme for this will be presented later in Section 2.6.2.

[16]In fact as was said earlier, see caption of Fig. 2.2, brown and orange are the same color, only their color intensity differentiates them, brown being a dark version of orange

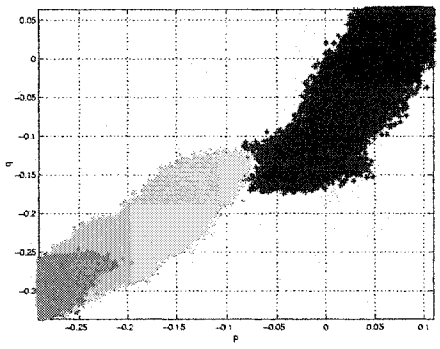



FIGURE 2.9. Overlap of the theoretically predicted brown and achromatic clouds of points shown in cyan and black respectively. In the present case, the clouds do not overlap by a significant amount, but nevertheless they need to be separated, and is the reason for using an objective computational approach such as Fisher's.

Recall that in 2-D, Fisher's approach allows to find the direction of the best line that separates two clouds of overlapping points. Note that only the direction of this line can be determined with this approach, not its ordinate at the origin (which fixes its absolute position). There must be a further means to find this position once the direction is found. In the present case, only the clouds for brown and achromatic colors overlap (and not by a large extent). This is illustrated in Fig. 2.9. The details of the separation approach developped here will be illustrated with the latter clouds (the same approach will be used in next chapter). The steps are as follows (see Fig. 2.10). First, the direction of the separation line is determined with Fisher's approach. Fig. 2.10 (a) shows the result. Next, the coordinate system is rotated so that the $x$ axis of the new coordinate system is along the direction of the separation line. The coordinates in this new system will be denoted by $x\prime$ and $y\prime$. Fig. 2.10 (b) shows the data in this system, in which the optimal separation line is parallel to the $x\prime$ axis, its equation being $y\prime = constant$. It is readily seen in Fig. 2.10 (b) that the value of the constant should be around 0.6. This value must now be precisely determined. To achieve this, the $y\prime$ axis is discretized into regular intervals, and for each of these intervals, the number of data points of each type (here brown and achromatic) for which the projection falls into this interval is determined. In other words, a histogram for each type of data is constructed, which gives an estimate of the density of these data along the $y\prime$ axis. Fig. 2.10 (c) illustrates these histograms. Again, it is seen that the value of $y\prime$

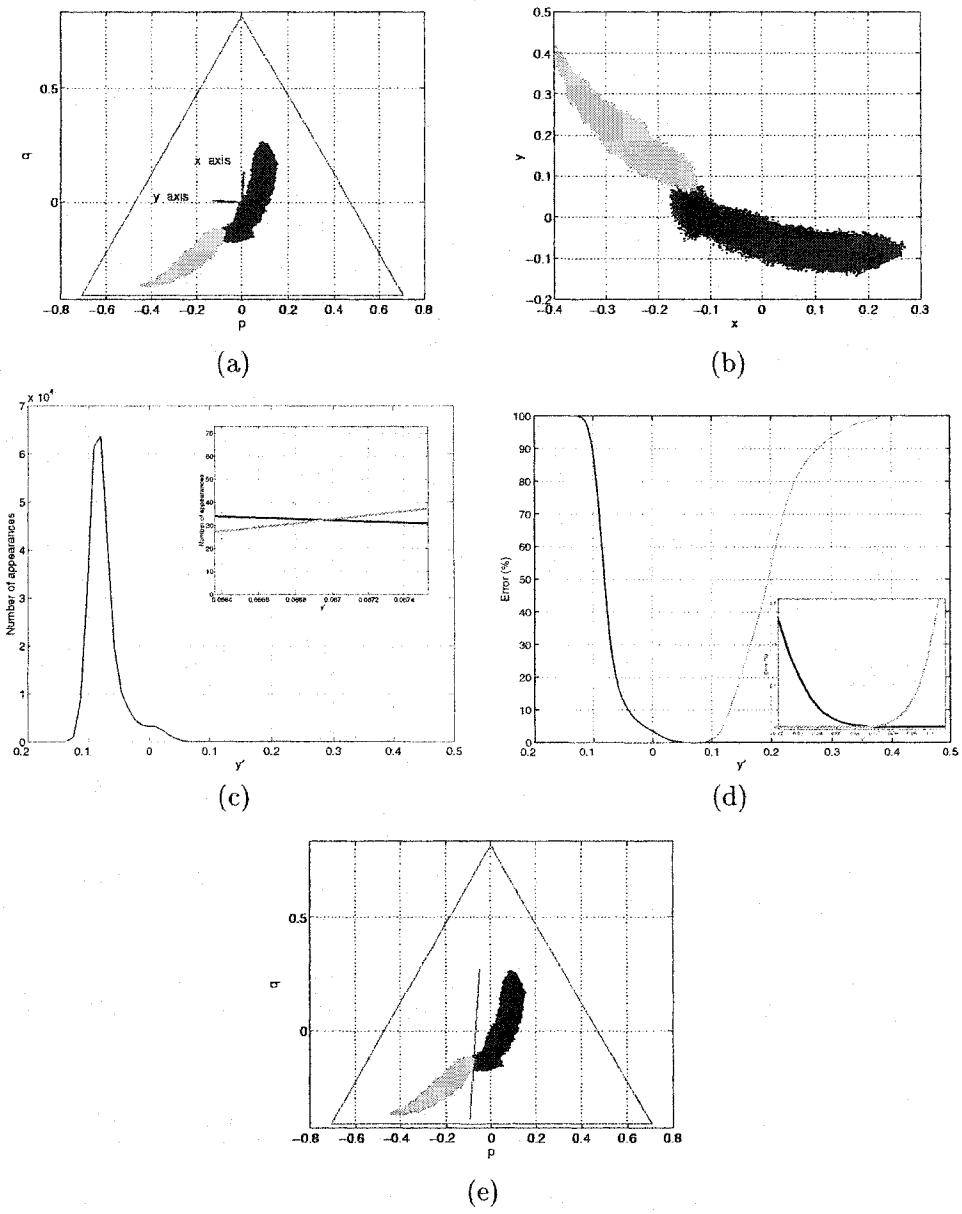



FIGURE 2.10. Separation of the brown and achromatic clouds. (a) Brown and achromatic data in the $(p, q)$ coordinate system, and axes of the $(x\prime, y\prime)$ coordinate system obtained via Fisher's method. (b) Data displayed in the $(x\prime, y\prime)$ system. (c) Histograms of the brown and achromatic data projected onto the $y\prime$ axis. The inset shows a zoom on the crossign of the histograms. (d) Classification errors as a function of the threshold chosen to separate the data. The inset shows a zoom on the crossign of the error curves. (e) Data and optimal Fisher discrimination line in the $(p, q)$ system.

where the histograms should be separated is somewhere around 0.6. To find this value, the classification errors of the two data sets (brown and achromatic) when the discrimination line is positioned at different values of $y\prime$ are computed. For a given data set and position of the discrimination line, these errors, expressed as percentage values, are given by the number of points that are located on the wrong side of the line divided by the total number of points in that data set. These errors are illustrated in Fig. 2.10 (d). The value of $y\prime$ at which the two error curves cross is the best compromise, and is taken as the optimal value (this crossing is difficult to see in the figure, because the curves are quite flat around the optimal value). In the present case, this occurs for $y\prime = 0.0674$ at which value the errors are 0.005%. Fig. 2.10 (e) shows the optimal separation line in the $(p, q)$ coordinate system for that value of $y\prime$. Visually this line appears to be optimal. The flatness of the error curves around 0.0674 indicates that the errors are not very sensitive to the value of $y\prime$ chosen to separate the data around 0.0674.

The line that separates two overlapping clouds also separates the color space in two half-spaces. The color region associated with a given cloud is then taken as the convex hull of the points of the cloud located in its half-space. Fig. 2.11 (a) shows the resulting color space regions thus obtained for the brown and achromatic clouds. These regions will be called the brown and achromatic regions respectively. For the clouds that do not overlap with others, such as the blue, green, red, and yellow ones, there is no need for using Fisher's approach in defining their associated regions, taking their convex hull is sufficient. This is also done for the orange cloud, although it overlaps with the brown one; brown-orange separation will be discussed shortly. Fig. 2.11 (b) shows the regions obtained for the different colors considered here.

**2.6.2 Pixel Labelling and Grouping.** With the color space regions just determined, each pixel in an image is labelled (or classified) with the color name associated with a given region if its associated $(p, q)$ color coordinates fall into the region. This labelling technique can be efficiently implemented with a look-up table, and the calculations to generate this table can be carried off-line. This has the advantage of being a very fast labelling scheme, which will be of importance for road sign detection. After all pixels in an image are labelled, neighboring pixels with the same label are grouped to form image regions. In this grouping stage, 8-connectivity was found to provide for superior performance than

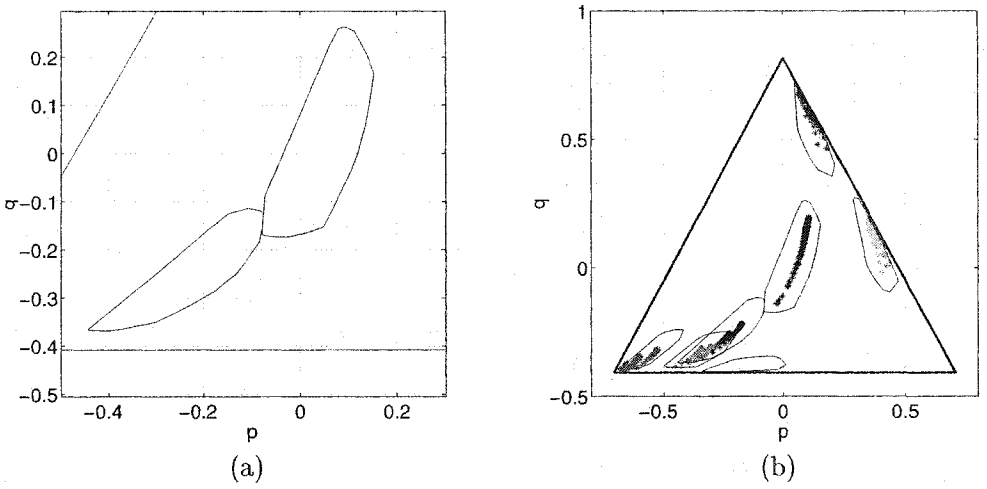



FIGURE 2.11. (a) Color space regions for brown and achromatic colors. (b) Color space regions for all colors illustrated with the traces obtained from the model without uncertainty, see Section 2.4.4.

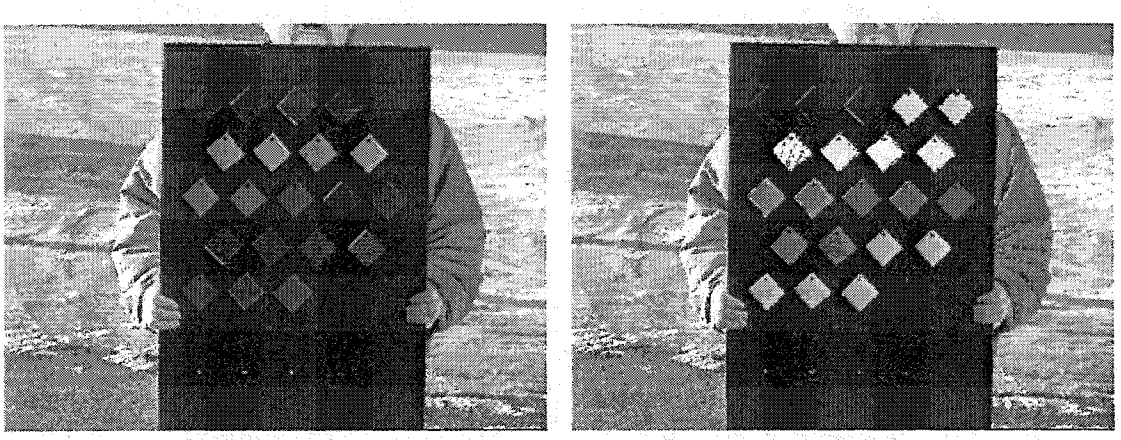
4-connectivity as regions tend to be less fragmented with 8-connectivity (this is particularly important for road sign detection).

As regards brown and orange, pixels which have their $(p, q)$ coordinates fall in the brown and orange detection regions are first labelled as being of the same color, call it brown-orange. They can then be further discriminated by considering the histogram of their color intensity $I = C_R + C_G + C_B$. Orange being more intense than brown[17], this histogram should be bimodal. To separate the modes, Otsu's method [**104**] is used to automatically determine a threshold[18]. One might ask why orange and brown are not separated as was done in the brown-achromatic case (previous section). For orange and brown, it appears more appropriate to use color intensity for separation because brown is a dark orange, and so color intensity is what more fundamentally distinguishes them[19]. In the brown-achromatic case, these are basically distinct colors, and moreover it is always

[17]The reason for this is that there is more contrast for orange than for brown between the plateau where the spectral reflectance reaches high values (above approximately 610 nm) compared to where it is low (below approximately 500 nm).

[18]It should be stressed that Otsu's approach allows to find the threshold *automatically*.

[19]There might be little spectral differences between orange and brown as was seen in Fig. 2.2 and which also shows up in Fig. 2.7 (b) for the samples considered here (the traces of predicted body colors overlap significantly, but are not exactly coincident). Nevertheless, brown and orange can be considered as almost the same color due to their significant overlap in color space.



(a) (b)

FIGURE 2.12. Results using the detection scheme discussed in text. (a) Original image. (b) Detection results. To ease visualisation (due to the color printing process), pixels labeled brown, orange, and yellow are depicted in cyan, magenta and white respectively. For the other labels (blue, red, and green), the colors correspond to the labels. This coloring will be used throughout the following, unless otherwise stated. Here the image was captured under clear sky conditions with direct sunlight coming from behind at the left, the chart was thus not directly illuminated by the sun.

possible to find a gray that will have the same color intensity as brown. It is thus more appropriate to separate brown and achromatic colors directly in color space.

## 2.7 Results

The detection scheme just described has been tested on a large number of images (272 to be precise) containing the chart shown in Fig. 2.7 (c). These images were taken at different times of the year and under a wide range of daylight illumination conditions.

Fig. 2.12 shows a typical test image and the detection results. It is seen that superfluous pixels have been detected as green on the left arm of the person holding the chart, along with some as orange on his moustache. These pixels do not belong to objets of interest, and will be termed superfluous (although from a pure color detection perspective, these pixels are not superfluous as their colors are appropriate for being detected). As such, the present method does not allow to eliminate such superfluous detections. This will have to be done by other mechanisms using other type of information depending on the application. For example, too small clusters of detected pixels can be discarded by using a threshold on the number of pixels. Also, in applications where the objects to be detected based on

color must also be identified using shape, such as in road sign recognition, the identification phase will determine whether or not a detected region is relevant. These matters shall not be considered any further at this point, as they will be treated later in the thesis. It is not the purpose of the present detection method to assert whether a region is relevant or not, but rather to detect regions of pixels that are of potential interest on the basis of color given an application where color is a useful attribute for the task to be performed. So the primary quality of this first detection step is sensitivity; more weight is given to avoid miss error than to avoid false alarm error, since in the early stages of processing the former is more costly than the latter. Referring back to the figure, it is seen that the detection results on the color samples are excellent (on the leftmost yellow sample, there are perhaps a very few pixels that have not been labelled, but their number is not significant).

Figures 2.13, 2.14, 2.15, 2.16, 2.17, 2.18, 2.19, and 2.20. show results for a wide gamut of daylight illumination conditions (see figure captions for the description of the results). Again, it should be emphasized that these images have been captured during the different seasons of the year and at different times during the day. Overall, the method gives satisfying results for a first step, but in some instances problems occur and improvements must be made. This will be the subject of next chapter[20].

These problems consist in missing pixels either in the form of detected regions that should have been larger, and/or that contain undetected pixels. A few samples in Fig. 2.22 exemplify these problems (see caption for the description). More severe cases are depicted in Figs. 2.23 and 2.24.

Problems are also seen to occur when the image gets darker (mostly for the brown samples, and sometimes for the red ones). The reason for this is that as colors get darker, they all tend to black, and are thereby more sensitive to noise (assuming sensor noise remains constant). This is seen to be the case in Figures 2.14 (b), 2.16 (c), 2.17 (f), and for a few brown samples of some of the images in Fig. 2.18, where pixels are seen to be missing on some of the samples.

Of course the cause for the pixels that are missed is that their values do not fall in the color regions defined using the model (Fig. 2.11), and thus cannot be detected. Superfluous detections are also seen to occur, but as mentioned above, not much can be done, and it

[20]Since the method will be refined in next chapter, quantitative results of extensive tests will be presented there.

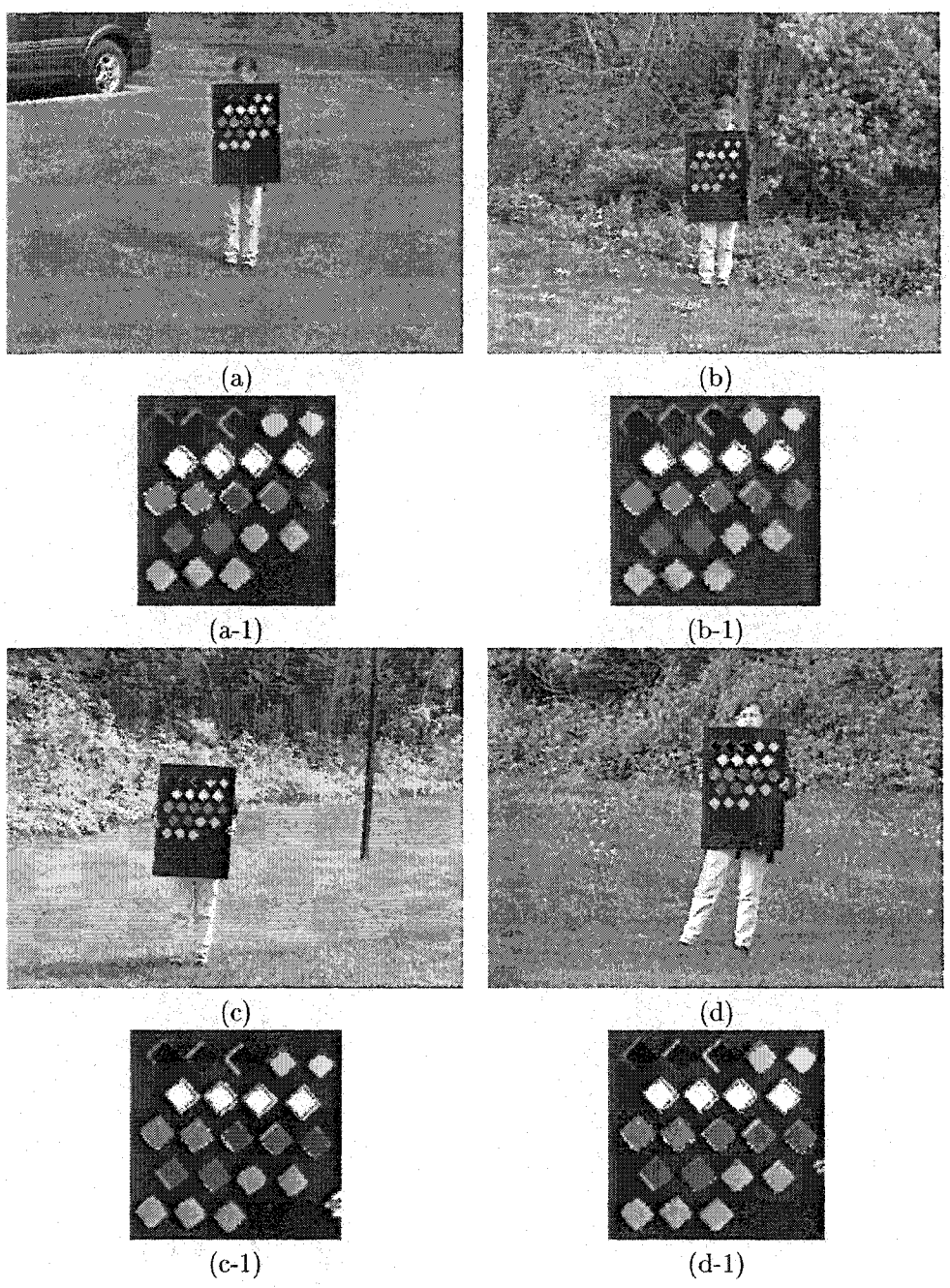



FIGURE 2.13. (a), (b), and (c) were captured at the end of the spring within a few minutes in the morning under a rapidly varying sky; (d) was taken on the same day in the afternoon. (a-1), (b-1), (c-1), and (d-1) are zooms on the chart. Some pixels not having the color of the samples are seen to be detected in the vicinity of sample boundaries. This is due to the finite resolution of cameras, which causes averaging to take place at objects borders. Hence, at borders detection can possibly not be reliable. Such detections could easily be filtered out. This has not been done here in order to illustrate raw results, however see Section 3.2.2 of next chapter and Section 5.4.3 in Chapter 5, for background cleaning.

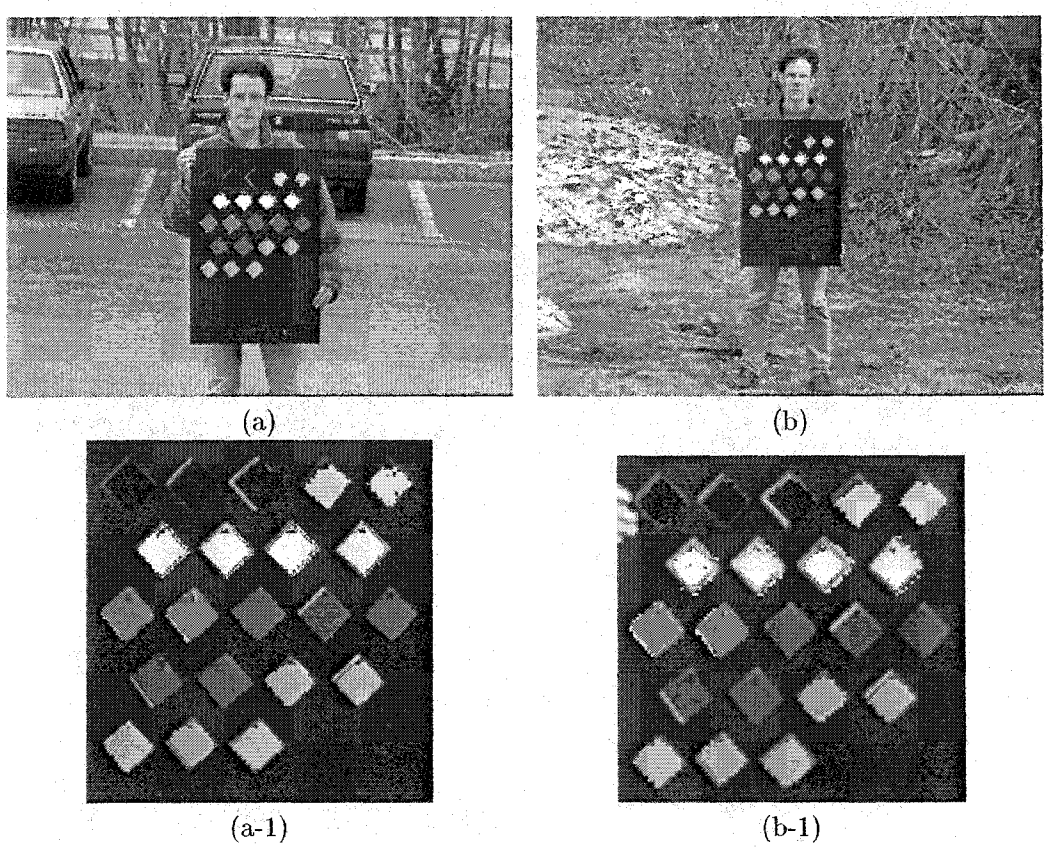



FIGURE 2.14. Results for images taken during the spring on different days. For both (a), and (b) the overcast was heavy. For (b), it is seen that two red samples were less well detected.

is also not critical at an early processing stage as other mechanisms may be considered to resolve these at later stages. Figures 2.17 (c), and (d) show examples where superfluous detections are more prominent (blue and red pixels on cars).

Bright images may also suffer from saturation problems (also called clipping), as illustrated in Fig. 2.21. Conceptually, clipping is not a problem of the detection method itself, since it is due to the finite dynamic range of cameras, which is technical in nature. For this, not much can be done at the image processing level for real-time applications. To avoid clipping, the sole avenues are at the hardware level. A first avenue is to use the automatic gain control (AGC) of the camera. However, this was not done here because of its potential effects on the spectral sensitivity curves of the camera. Indeed, if the gain is changed, because of the gamma parameters (see Appendix A), the spectral sensitivity curves do not

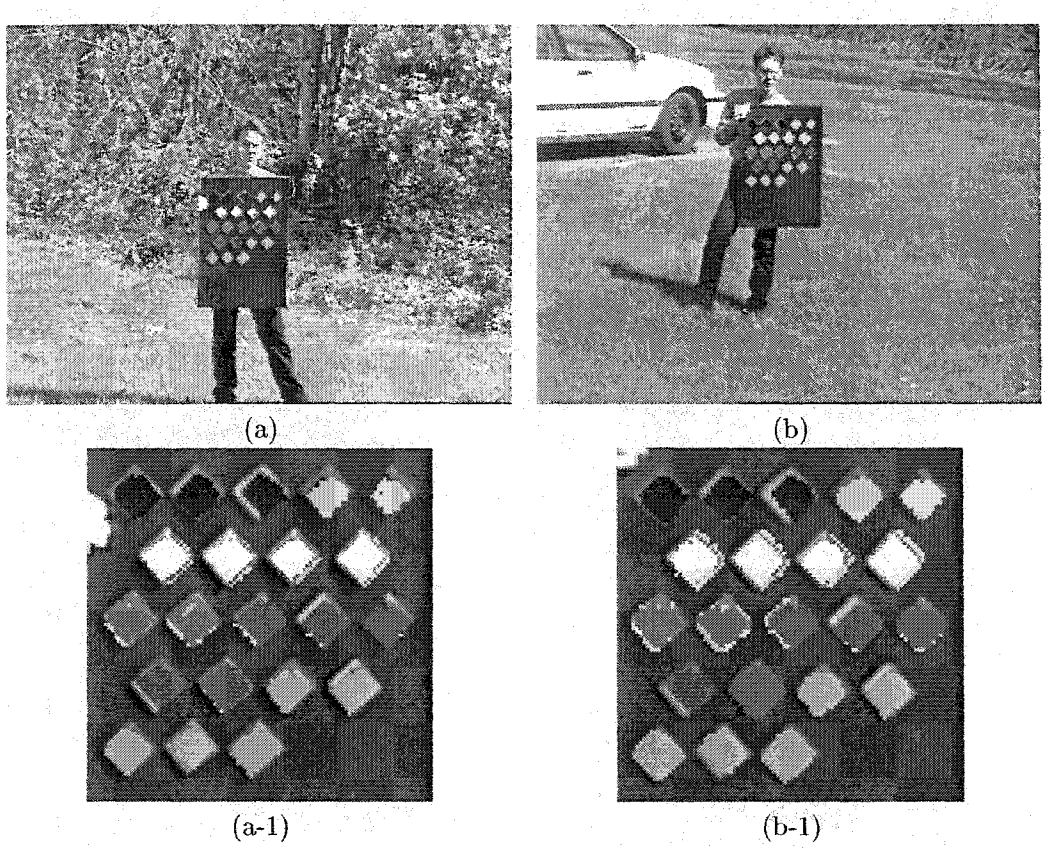



FIGURE 2.15. Detection results for clear sky conditions. The images were taken within the same minute on a summer day morning. As can be seen from the shadows, the light from the sun comes from the right in (a), whereas in (b) the chart faced the sun. Zooms on the chart are given in (a-1) and (b-1). To avoid saturation problems, the aperture of the camera objective was set smaller in (b). However, the zoom in (b-1) shows that saturation problems still occured close to the border of two yellow samples. See Fig.2.21 and text for comments on saturation problems.

scale linearly. Compensating for this would require to have access to the value of the gain used by the camera, which was not possible with the camera used in this work. Other avenues consist in either adjusting the iris of the camera objective so that the image is not too bright, or reducing the integration time of the sensor (either approach could eventually be actively performed in real-time by feedback control with an automatic iris or shutter). In doing this, however, the settings should be such that the image is also not too dark. There is thus a trade-off here. In Fig. 2.16 the influence of changing the iris aperture of the camera lens is illustrated. Results tend to degrade if the image becomes too dark.

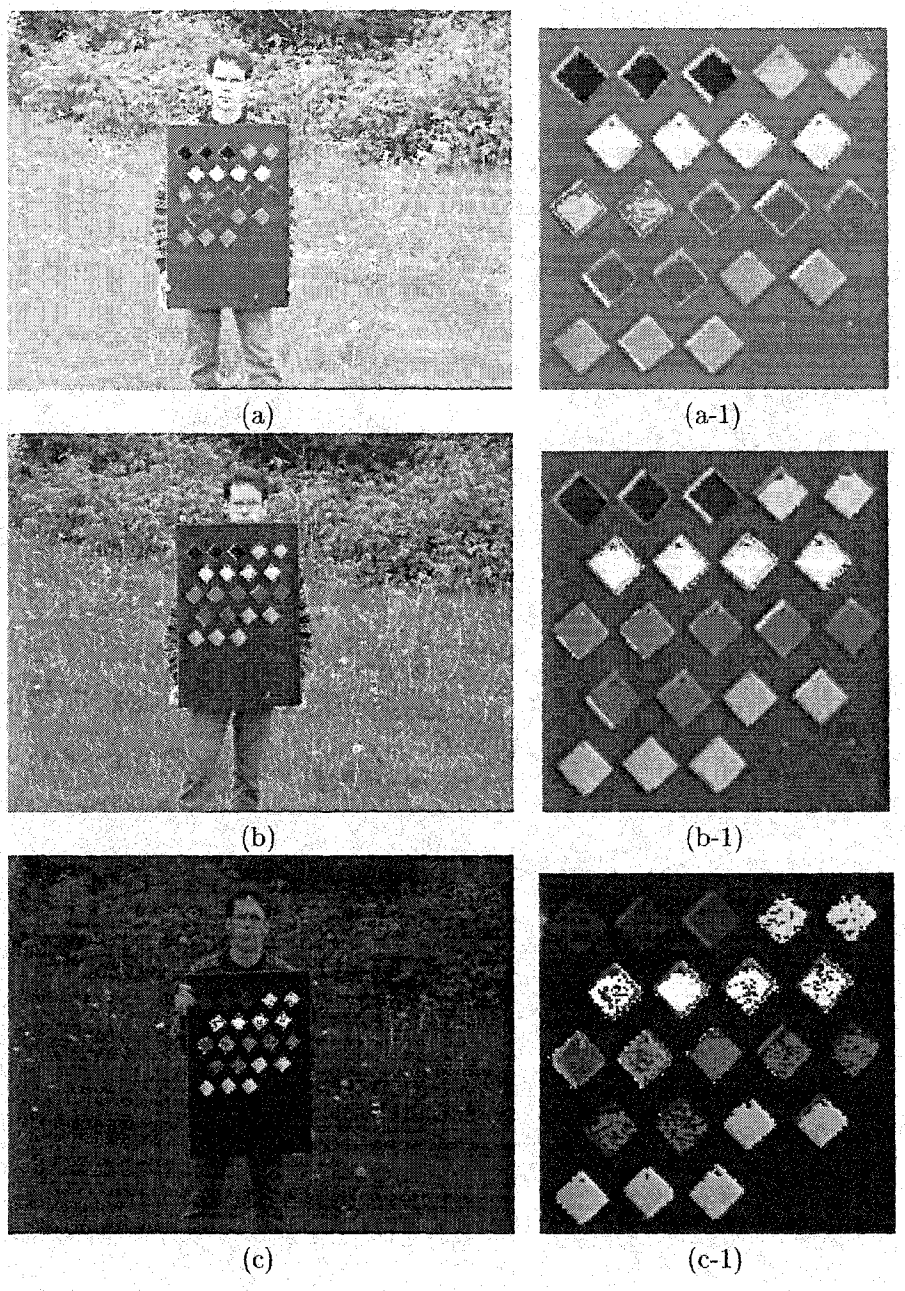



FIGURE 2.16. Effects of changing the aperture of the camera objective. The images were captured within a few seconds during the spring under a cloudy sky. When the image is too bright (a) (aperture too large) clipping problems occur (on the second and fourth yellow samples, and on the first and second orange samples. On the other hand, when the image is too dim (c), the results tend to degrade, and appear noisier. Notice, however, in this last case that pixels are still detected on every patch, and so with further work it should be possible to detect all pixels on every patch. This will be done in next chapter.

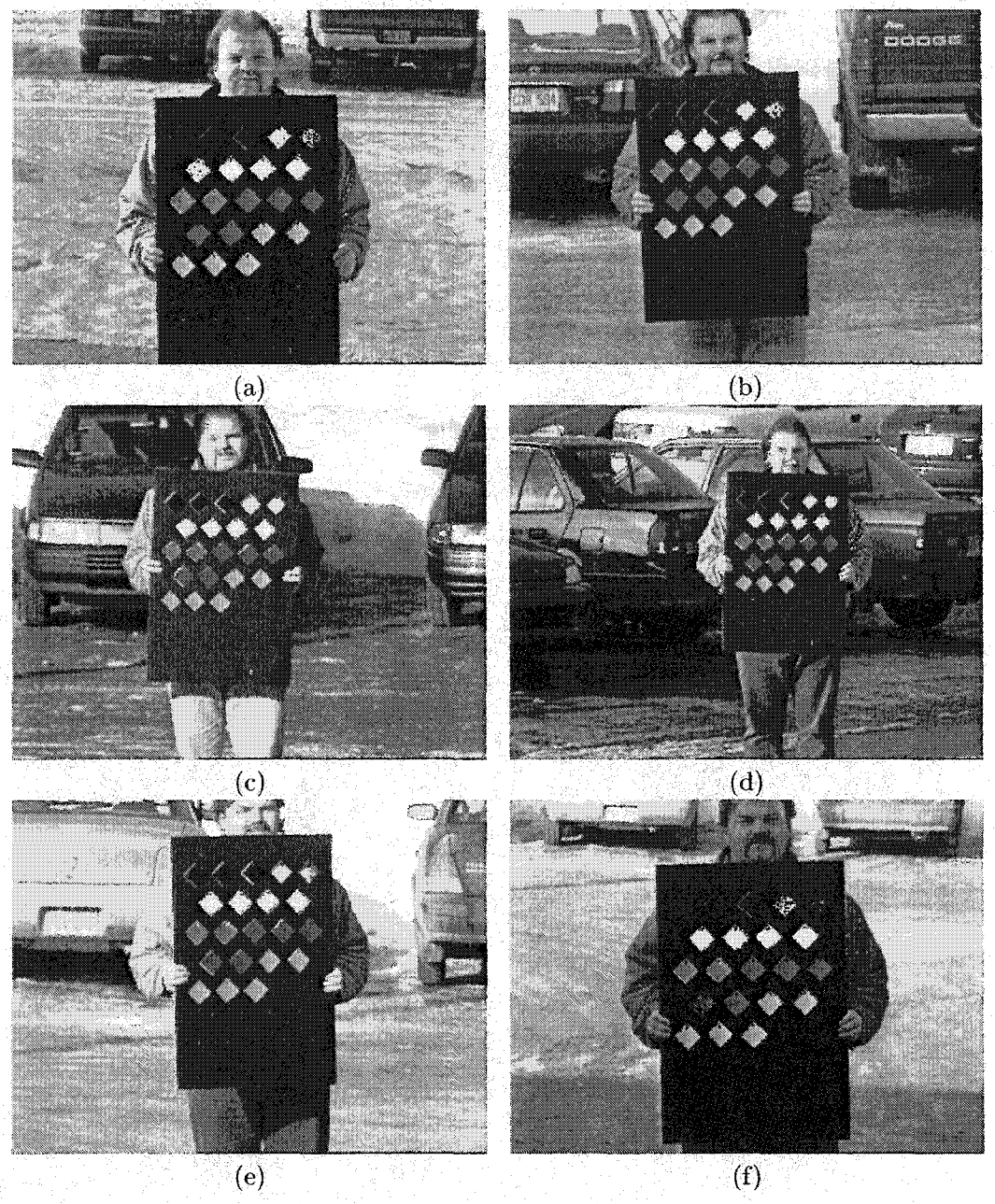



FIGURE 2.17. Images taken in winter under sunny conditions except for (b), shown here because it was captured on the same day as (a) ((a) was taken in the afternoon under sunny conditions with a light overcast, and (b) in the morning under a completely overcast sky). For (c), and (d) (different days), the sky was clear, with (c) taken in the morning, and (d) late in the afternoon. For the last two images, taken at almost the same time, the difference is that in (f) the chart was shadowed by a building. Only a few pixels were detected on the top-right brown sample due to the darkness of the sample.

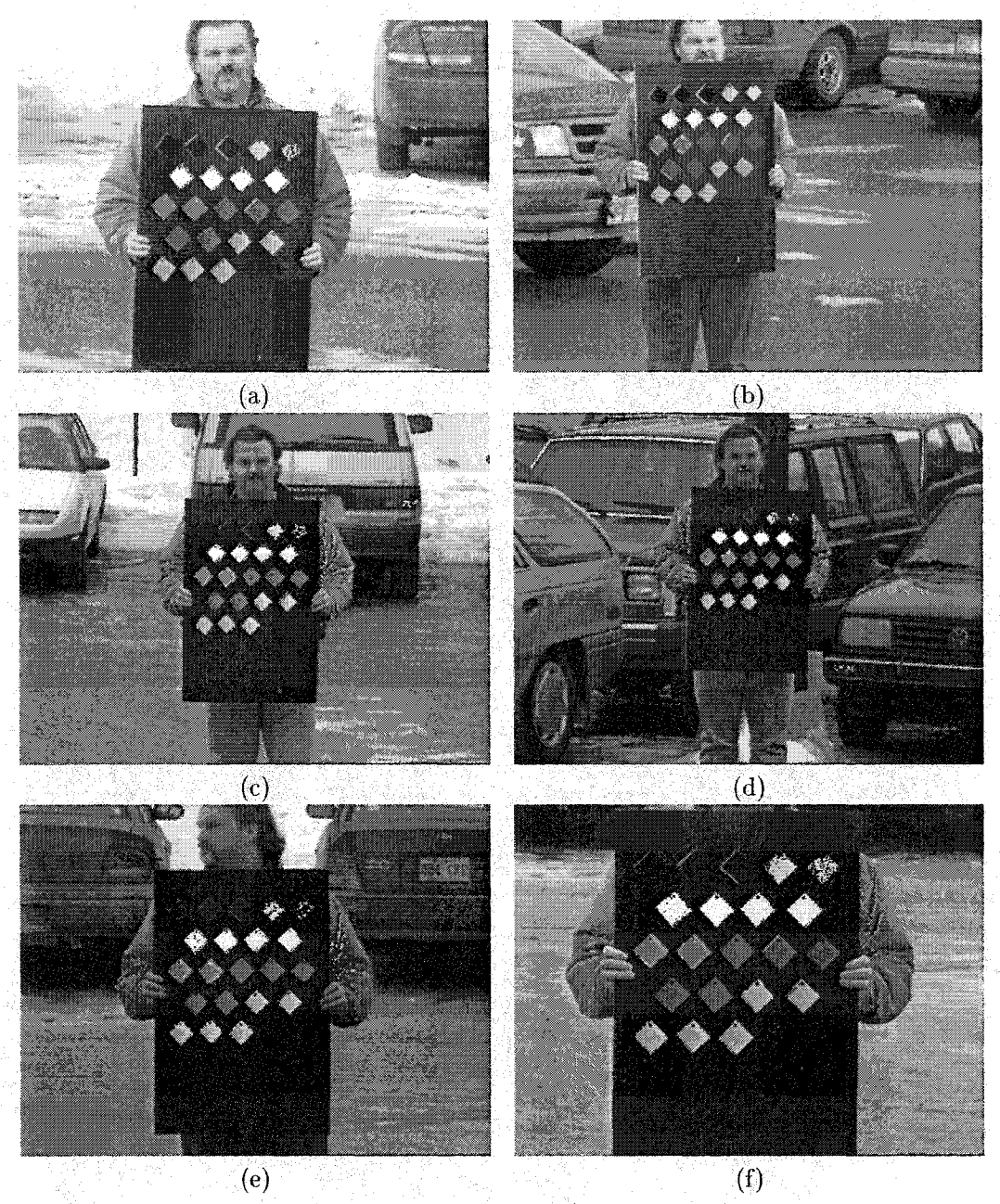



FIGURE 2.18. Images taken during the winter with different overcast conditions. In (a), and (b) the sky was completely overcast with white clouds, the day was clear. Images (c), and (d) were captured under heavy overcast (dark gray clouds), it was about to snow. For (e) and (f) the overcast lied somewhat between the previous conditions (light to dark gray clouds).

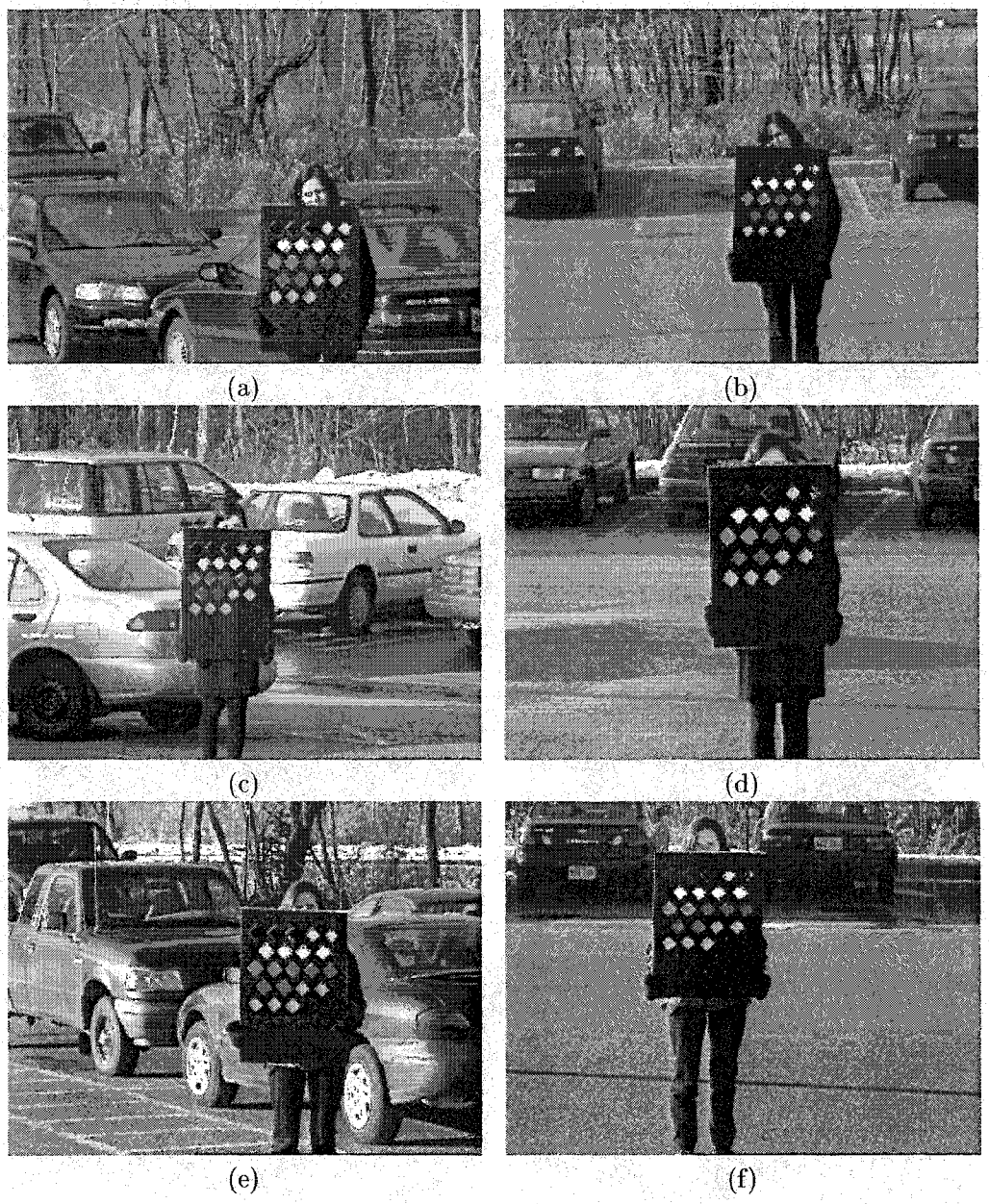



FIGURE 2.19. Images captured in fall under sunny sky conditions. (a) and (b) were taken around noon on the same day, with the chart almost facing the sun in (a), and placed sideways to the sun in (b). Some clouds were present, but the images were taken when the sun was not shadowed. (c) and (d) were captured by the end of the afternoon under a very light overcast. In (c) the chart directly receives light from the sun; this is not the case in (d). For (e), and (f) the time of capture and the chart's position with respect to the sun are the same as in (c) and (d), but the images were taken on a different day, with the sky being completely clear, and the light being more reddish. In (f), the upper right brown sample is less well detected, but it will be recovered in next chapter.

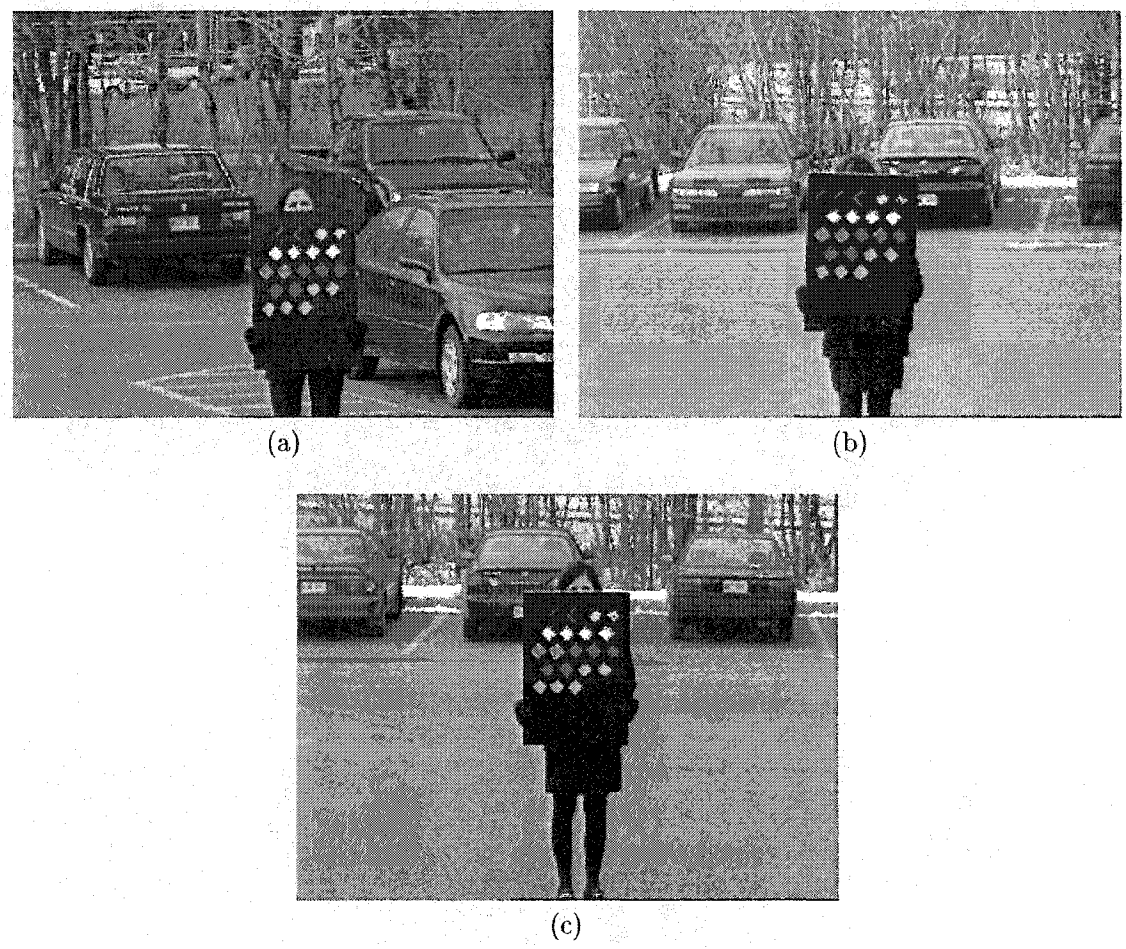
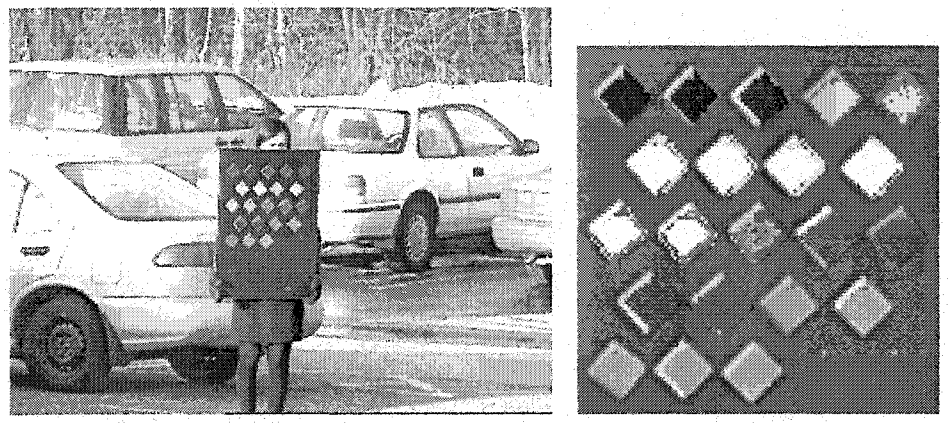



FIGURE 2.20. Images taken during the fall under cloudy conditions. The image in (c) was captured in the morning on the same day as that of Fig. 2.19 (c) and (d) (sunny conditions). The conditions in the morning were cloudy.

Although the model-based detection approach suffers from some problems as discussed above (missing pixels, problems when image gets darker, and saturation), its salient feature is that it is successful in detecting clusters of pixels on objects of interest. In the tests that were performed using it, only rarely does it fail to at least find small clusters of pixels on objects of interest, and this even in the most severe problematic cases (see again Figs. 2.23 and 2.24). This means that the approach has good sensitivity, and that the theoretical predictions are sufficiently close to model what is actually observed in real images, but that the variations of real data are not completely accounted for by the model. To illustrate this on a specific example, which is typical of the situation, Fig. 2.25 shows color pixel values



(a) (b)

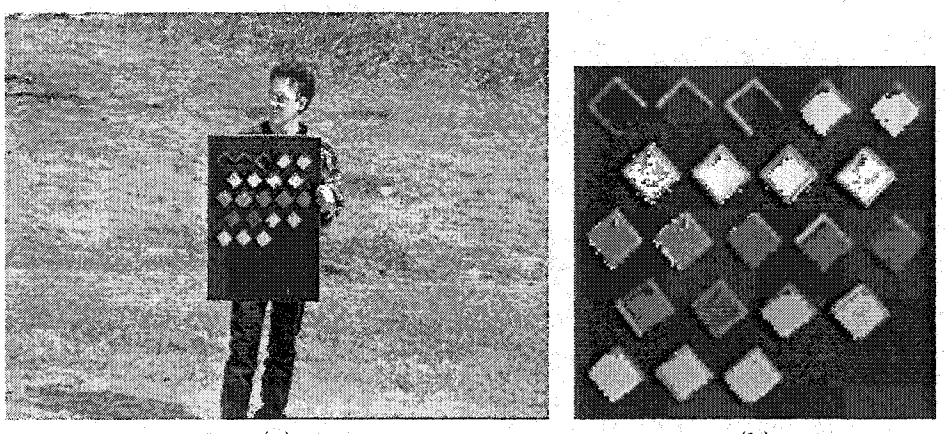
FIGURE 2.21. Images showing detection results with problems due to clipping. Original image and detected pixels are illustrated in (a) and (b) respectively. Three yellow samples are not detected, and a red one is only partly detected. It is also seen in (b) that pixels which should be labelled as orange are instead labelled as yellow. In all these cases, it is the red channel that has saturated.

that were not detected in Fig. 2.22. It is readily seen, that these values are not far from the model's predictions.

This suggests to further refine the regions predicted theoretically in order to obtain the color regions associated with the considered materials that will account for those variations. This refinement will be performed in next chapter via a calibration procedure which will resort to real data. By using real data to calibrate the theoretical regions, the variations encountered in real images will be taken into account and more robust detection will be achieved. This is important if the detection results are to be used in further processing, *e.g.*as in road sign recognition which will be considered later. A main advantage of starting with theoretical predictions that are sufficiently close to model the behavior of real data is that the subsequent calibration procedure can be automated, as will be shown.

## 2.8 Conclusion

The theoretical model presented in this chapter, despite the problems discussed in the previous section, provides a solid basis on which to found work related to color detection under daylight. It would also be applicable to other types of light sources, not just daylight. The model allows to predict color regions in a consistent and well-defined manner, since the



(a) (b)

FIGURE 2.22. Typical detection problems. (a) Original image with detected pixels overlayed. (b) Zoom on the chart. The first and fourth yellow samples from left in the second row, and the second red sample from left in the previous to last row are cases where the detected regions contain undetected pixels. The second yellow sample from the right shows an instance where the region should have been detected larger.

values of the parameters it involves are known. Such a model is a valuable tool as it gives a solid starting point with which to compare actual data.

As a final remark, here color space regions were defined with the convex hulls of clouds, and detection was performed using these regions and a look-up table. Another detection scheme would be to use a Bayesian approach in which each pixel's color would be assigned a probability of being of a given color. This could be an interesting avenue to examine, but in view of the application of the approach to road sign detection, pixel color labelling must be performed quickly, and an approach that does require a minimum of on-line computation is priviledged. A Bayesian approach would require on-line computations, and this is the reason why such a paradigm has not been studied (this remark is also valid for the work presented in next chapter). Nevertheless, a Bayesian scheme based on the model presented in this chapter could be an interesting avenue to study in other contexts.

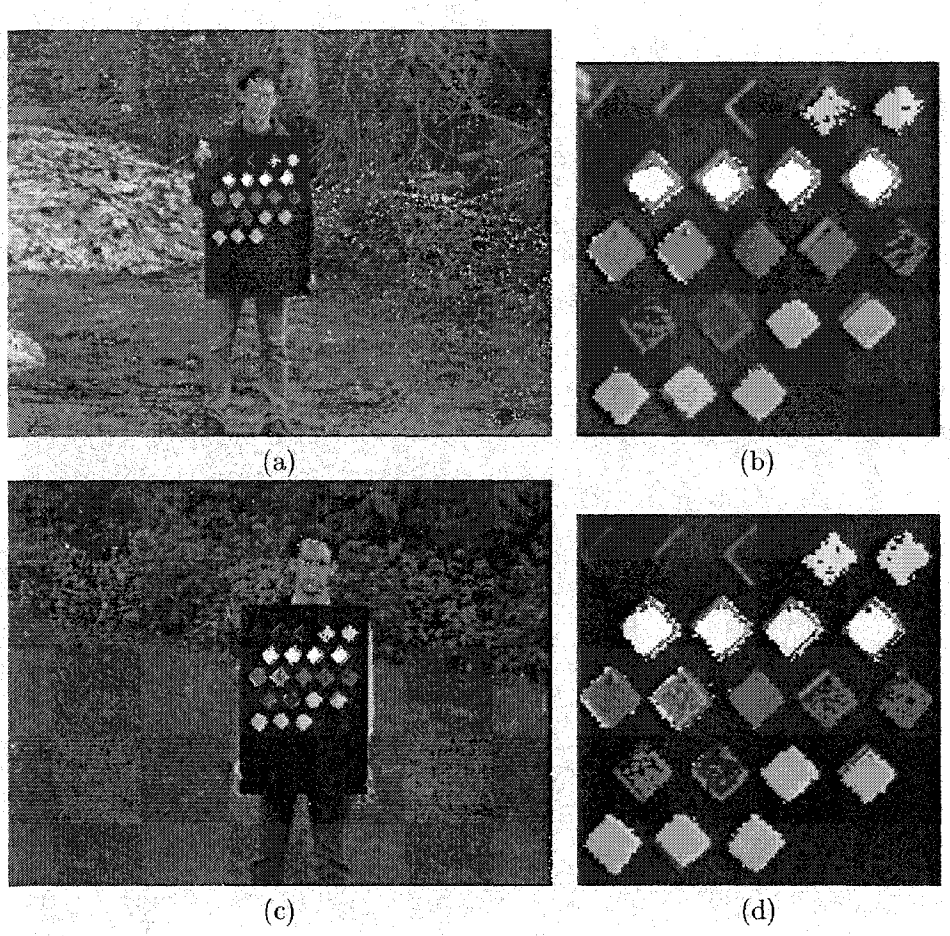



FIGURE 2.23. Severe detection problems. (b) and (d) are zooms on the chart of (a) and (c) respectively. The image in (a) is a darker version of Fig 2.14 (b) (aperture of the camera objective set smaller). In (b) and (d), although pixels of the appropriate color are detected on all samples, a significant amount of pixels are missing on some of them. This will be corrected for in next chapter.

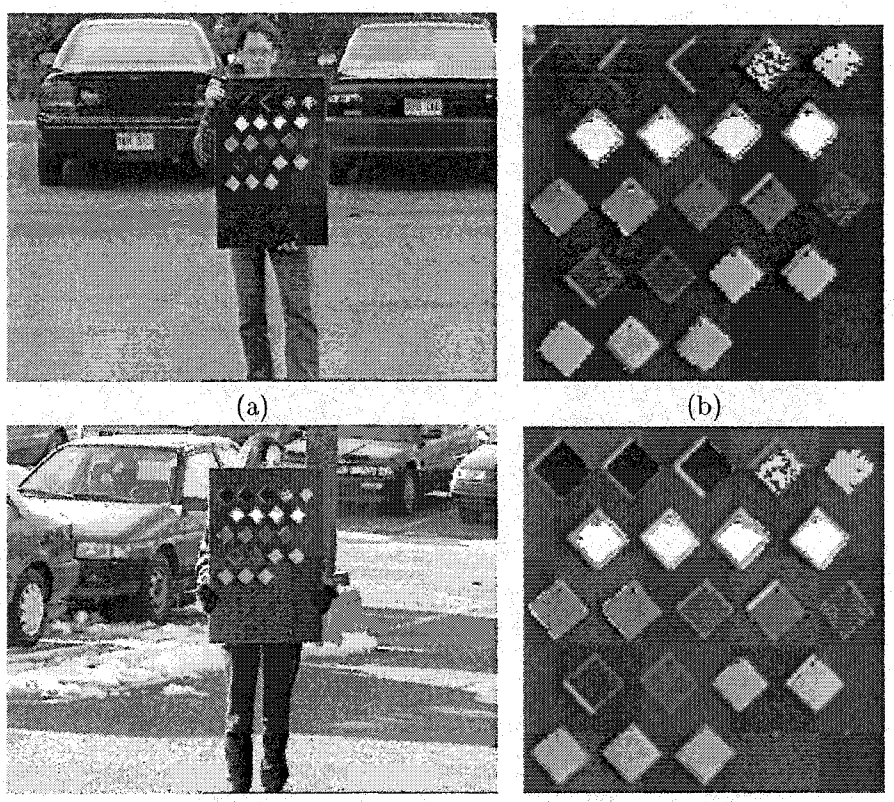



FIGURE 2.24. Other examples where more pronounced detection problems occur. As in the previous figure, (b) and (d) are zooms on the chart of (a) and (c) respectively.

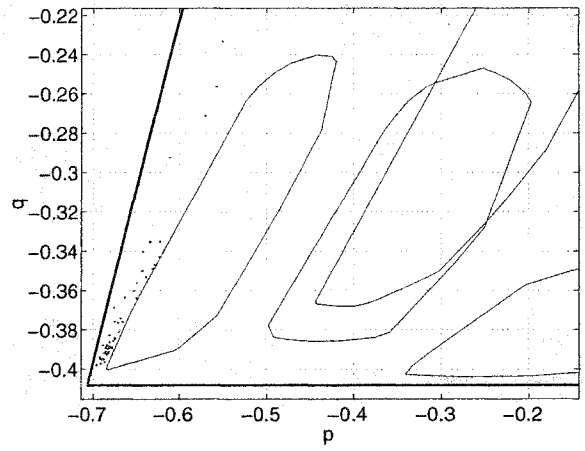



FIGURE 2.25. Plotting the color coordinates of pixels that were not detected in Fig. 2.22 as compared to the color regions predicted by the model. The undetected values fall close to the theoretically predicted regions.

# CHAPTER 3

# Physics-Based Color Calibration under Daylight

## 3.1 Introduction

In the preceding chapter, a theoretical model was developed for predicting the color of objects of interest in images captured under daylight illumination. From this, color space regions could be determined and used for detection purposes. Experimental results have shown that in some instances the theoretical regions do not account for enough variation compared to observed data. However, on samples where problems occur, detection based on those regions almost never fails completely, as it is able at least to find clusters of pixels on those samples. This motivates the use of the model-based regions for subsequent development. It seems difficult at this point to go any further with the model itself and try to refine it, *e.g.*by supplementing it with additional parameters, without introducing ad-hoc hypotheses. Instead, what will be done is to use real data to better adapt the color regions prediced by the model. Exploiting real data is a standard technique in learning approaches where a classifier is trained directly with that data [**37**].

At first sight, it might be tempting to conclude that model-based detection is a failure and is not useful, and that perhaps a learning approach should have been considered at the forebeginning. This is not so for two reasons. Firstly, because the theoretical regions are close to model what is actually observed in real images, it will be shown in this chapter that it is possible to calibrate them and reach high detection efficiency with the resulting calibrated regions. Secondly, in learning the first task is to determine examples of objects (to be found in a set of predetermined "training" images) from which to extract the data

needed to train the classifier (here color values). This is usually done manually by a person, which is a tedious task representing a large amount of work. In what follows, finding examples will be automated, and the model-based regions will be exploited in that process. The approach to be presented will be seen to bear resemblance with learning approaches, and some terminology will be borrowed from these (*e.g.*learning and training sets), but the fundamental difference is that the process relies on a model to be instantiated and is automatic, which is generally not the case in learning.

Finding examples is a typical recognition problem consisting of two parts (when done manually, the person actually solves this problem). In the first part, objects that can potentially serve as examples are detected. In the present approach, it is at this point that model-based detection will be used. However, just as it was not sufficient for finding all pixels on a color sample in some cases, this will also be the case here, since the same type of images will be considered. It is precisely those missing pixels that must now be extracted in order to calibrate the model-based regions and increase their detection capabilities. To overcome these problems, model-based detection will be supplemented by a region growing technique which uses broad color regions (broader than those determined from the model) in its merging criterion. This will allow the previously missed pixels to be found, and thereby examples to be reliably extracted. However, other objects will also be extracted in this process. The second part for finding examples then consists in validating the detected objects using shape recognition to keep only those from which pixel color values should be extracted (that is, find the sought color examples). The values so obtained must be further validated to filter out outliers. The remaining data, called the training data, are then used to define color space regions representative of the colors to be detected.

The organization of this chapter is as follows. The details of the calibration procedure that has just been described, will be given in Section 3.2. In Section 3.3, color regions resulting from the calibration procedure will be given, and in Section 3.4 experimental results on the detection efficiency using calibrated regions will be discussed. Section 3.5 will conclude.

To summarize this introduction, what is suggested is to first use a broad extension of the model to instantiate an automatic calibration process to better adapt to real data the color regions obtained from that model. In a way, the model will be self-adapting to real

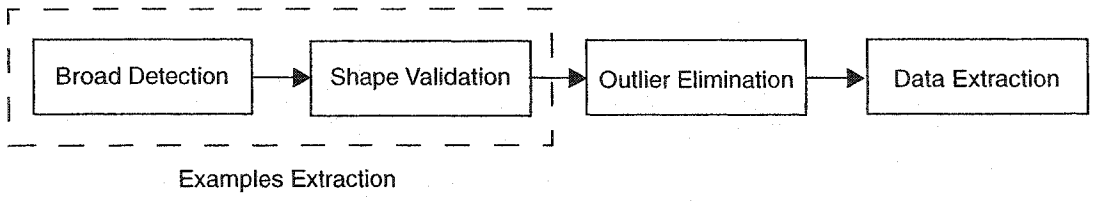



FIGURE 3.1. Salient steps of the proposed color calibration approach. The broad detection and validation stages serve to find examples of objects from which to extract data. Prior to obtaining the final data, outliers are eliminated.

data. This is done through a detection procedure based on the model which finds objects of potential interest (*i.e.*hypotheses), and which are then validated, so that the training data extracted from them is reliable. This is sketched in the flow chart of Fig. 3.1.

## 3.2 Calibration Procedure

In this section, the details of the different steps involved in the calibration procedure as outlined in the previous section will be described. The discussion follows the flow chart of Fig. 3.1.

**3.2.1 Broad Detection.** Given a training image from which to extract pixel colors, the detection part of autonomous examples extraction comprises two stages. The first consists in scanning the image and determining at each pixel if its value falls in one of the model-based color regions (Fig. 2.11). In the affirmative, a check is made to determine if the considered pixel belongs to a cluster of neighboring pixels (8-connectivity) of prescribed minimal size, all having values falling in the same model-based color region. This minimal size is a parameter provided to the algorithm. This is to decide if the pixel has enough support from its neighbors to be considered a reliable *seed*[1]. Notice that it is at this point that the model-based regions are exploited for their ability to find small reliable cluster of pixels having colors of interest, as was discussed in the previous chapter. Once a reliable seed is found, the second part of detection is to perform a region growing starting at that seed. The criterion for merging a new pixel to an image region being grown uses enlarged color regions which are significantly broader than those obtained with the model (Fig. 3.2). Before justifying the use of such regions, let the mechanics of the region growing be first

[1]Here this value was set to 10. With the resolution of the images used (640 × 480), clusters smaller than this size are not significant.

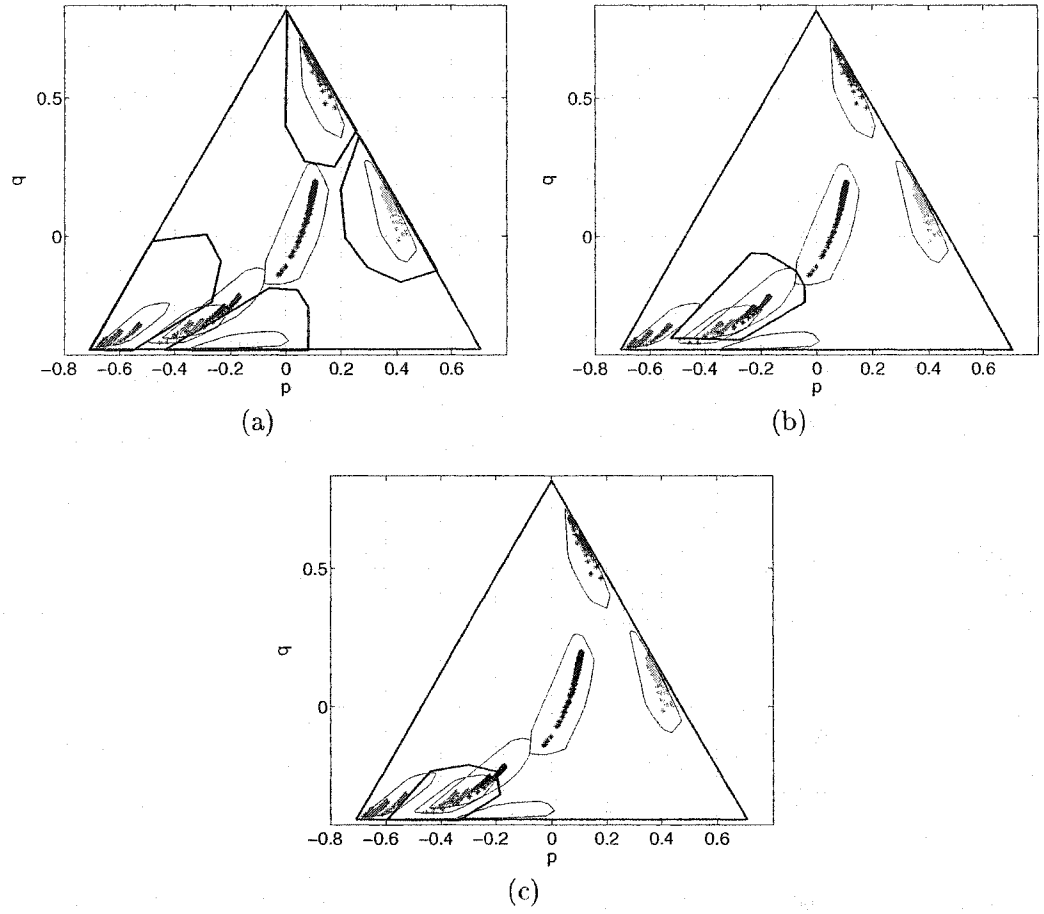



FIGURE 3.2. Enlarged color regions used in the region growing part of detection. They are depicted with bold boundary lines. For comparison, the regions obtained via the model are also shown. (a) Blue, green, red, and yellow enlarged regions, (b) brown enlarged region, and (c) orange enlarged region.

described. This is best done with an example. Suppose a red seed is found. A neighbor is merged to the region provided its value falls in the enlarged region corresponding to red; the criterion is applied recursively to successive neighbors of this new pixel until no more neighboring pixels are found satisfying this criterion.

The rationale behind the use of enlarged color regions is to allow the region growing to get the pixels that are missed by the model-based approach. Obviously, this is important if colors with their variability are to be calibrated. However, there is a trade-off regarding these enlarged regions. While their extent should be sufficiently broad in order that the color

regions to be obtained from the calibration are not limited beforehand, this extent must also be restricted, otherwise the region growing will spill significantly over the border of objects in the image, resulting in inaccurate detections (some small spillover will nevertheless still need to be corrected for as will be seen later). For the present purposes, the enlarged regions illustrated in Fig. 3.2 are a good compromise. These were determined by following certain guidelines as follows. First, and obviously, the enlarged regions must enclose those obtained from the physics-based model. Second, the enlarged regions need to be limited towards the center of the color triangle. This limit is set by the region of achromatic colors obtained with the physics-based model, which is shown again in Fig. 3.2. More precisely, the extended regions of the colored objects should not go beyond the boundary of the achromatic region (or perhaps only slightly as is the case for brown), otherwise a significant amount of color data pertaining to achromatic objects would also be gathered through the region growing. As a result a large amount of outlier data would contaminate the calibration data and would be difficult to discard (see Section 3.2.3 for outliers). The third guideline concerns potential overlapping between different extended regions. This is allowed, but the trace of predictions of a given color should not be included into the broad region associated with another color. The fourth guideline says that the broad regions should be made as large as possible within the constraints imposed by the previous guidelines. This is to ensure, as said above, that the calibrated regions be not restricted beforehand. The regions illustrated in Fig. 3.2 were designed to conform to these guidelines.

In the present case, it is seen that following these guidelines pretty much constrains the shape and extent the broad regions can have, not much lattitude is left. To illustrate some of the guidelines, notice in Fig. 3.2 how the blue, brown, and yellow regions are limited by the achromatic region (guideline two), and also how the orange region is limited at the left by the red traces, and at the right by the yellow ones (guideline three). Similarly, guideline three was also applied for the determination of the yellow, brown, and red regions.

**3.2.2 Shape Validation.** With the previous broad detection scheme, objects are detected, some of which are not of interest. This is illustrated in Fig. 3.3 (b) for the image in Fig. 3.3 (a). Spurious objects are discarded by a validation module using shape information. As a result, only objects of interest are retained (Fig. 3.3 (c)). In the present case, simple diamond-shaped objects are considered, and the template matching approach that will be

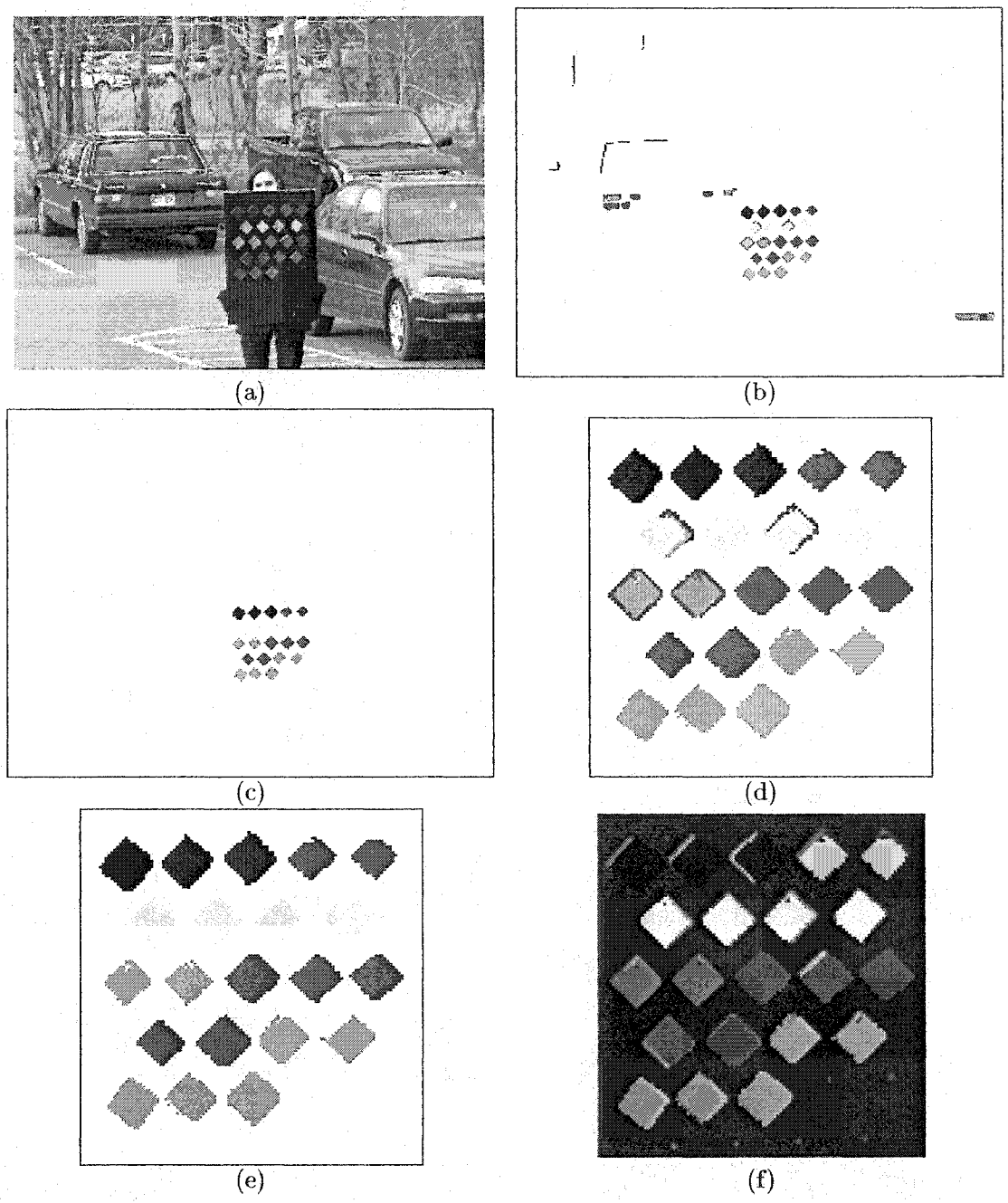
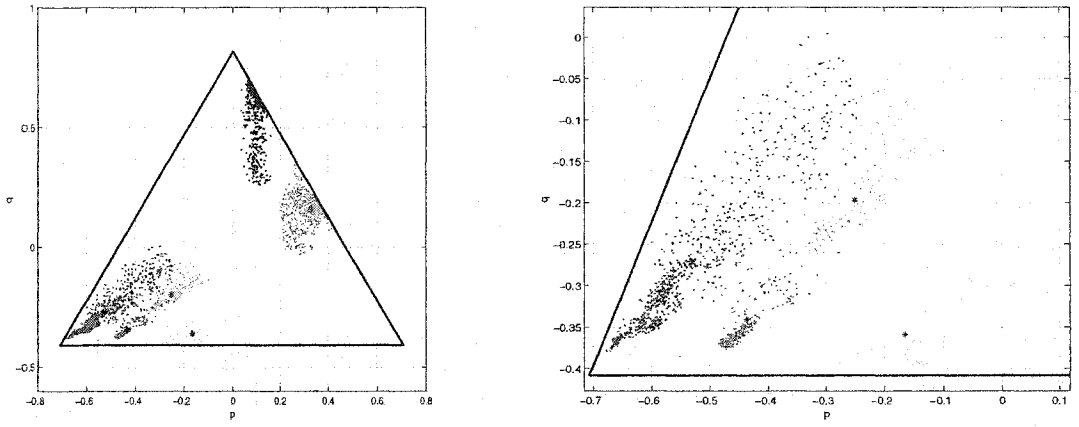



FIGURE 3.3. (a) Typical training image. (b) Objects obtained with the broad detection scheme. (c) Remaining objects after validation. (d) Zoom of the results in (b). (e) Zoom of the results in (c). (f) Same as (e), but with the results overlayed on the corresponding part of the original image. Note that in (b), (c), (d) and (e) the pixels detected as brown, orange and yellow are highlighted as such.




(a) (b)

FIGURE 3.4. The clouds of color values extracted from the image of Fig. 3.3 are shown in (a). Those for red, yellow, orange and brown are illustrated more closely in (b). The black stars correspond to the mean values of the clouds.

described in Chapter 4 was used for determining which detected objects to retain[2] (see also [**105**]). It is seen in Fig. 3.3 (d) that for some objects border pixels are detected as having a different color than the object; the leftmost yellow sample is an example. This is due to the finite resolution of cameras, which causes the color of border pixels to be a blend of the colors on each side of the border. Groups of pixels labelled with different colors are considered as different objects. Hence, the brown and orange groups of pixels at the border of the leftmost yellow sample are not validated as being diamond-shaped, and so do not appear in Fig. 3.3 (e) which shows the results after shape validation.

Once objects of interest are determined, pixel values are extracted from them. Fig. 3.4 illustrates the clouds of values so obtained for the image of Fig. 3.3 (a). The values forming these clouds are those of the pixels highlighted in Fig. 3.3 (c). The clouds are composed of regions of high density, as well as of values that are scattered away from these regions (best seen in Fig. 3.4 (b)). Values that are too much scattered away can be considered as outliers as they are not within the bulk of values. Indeed, they will be shown to mainly result from unreliable data gathered by the calibration approach due to the broad detection scheme and can be eliminated.

[2]Note that due to the simplicity of the objects, this could have been done by other means, such as contour analysis, but since an efficient method was developed in the present thesis project and was already at hand, it was used for the present purposes.

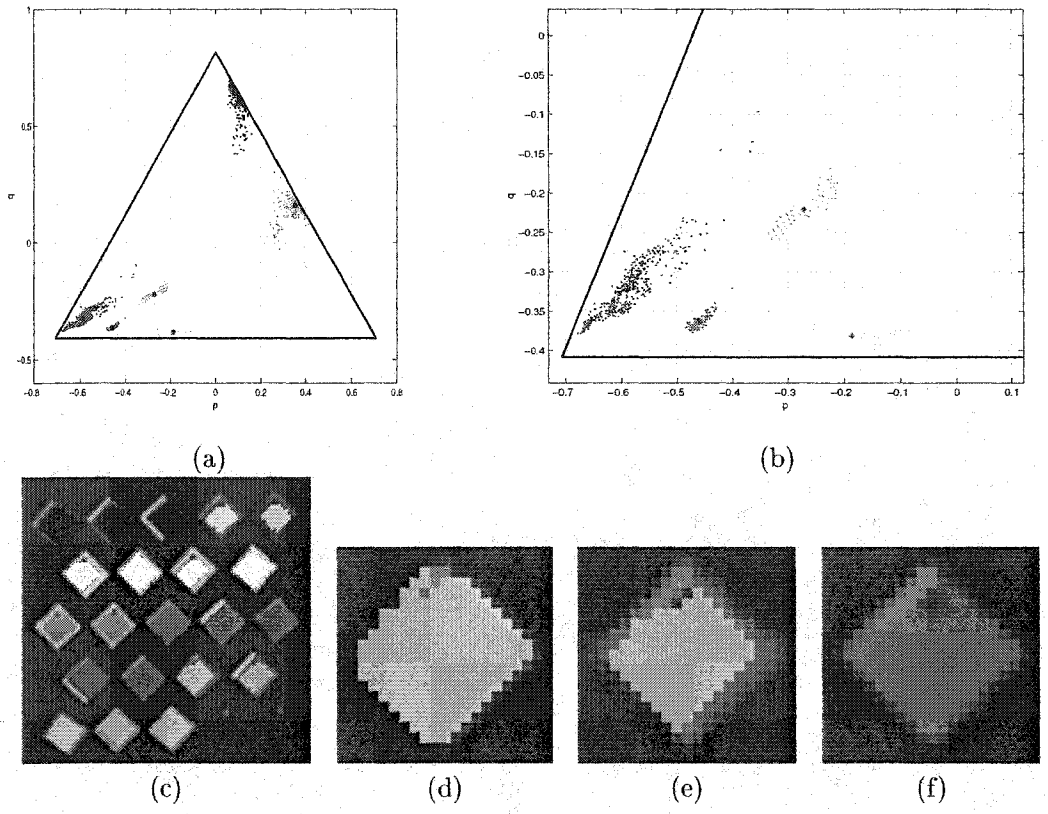
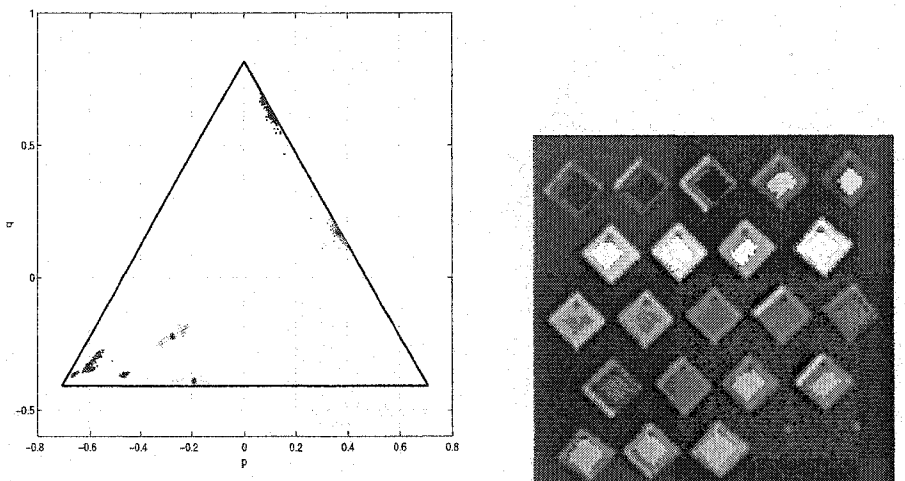



FIGURE 3.5. Using erosion to eliminate outliers resulting from border inclusion. The clouds obtained after one erosion are shown in (a) and (b). They should be compared with those in Fig. 3.4 (a) and (b). The pixels extracted after the erosion are depicted in (c). This should be compared with Fig. 3.3 (f). Border inclusion is illustrated more closely in (d), (e), and (f) on an example, which is the leftmost green sample on the fourth row on the chart. Highlighted pixels when no erosion is done are shown in (d), whereas in (e) one erosion was performed. When compared with the original sample in (f), there was clearly border inclusion in (d).

**3.2.3 Outlier Elimination.** The first idea that perhaps comes to mind to eliminate outliers is to resort to so-called robust statistical techniques [**106, 107, 108**]. However, doing this would overlook the causes for their appearance, which can in large part be corrected for at the source.

A first cause has to do with the enlarged color regions used in the detection. Due to this, the region growing has the tendency to include border pixels of objects (border inclusion), which is not reliable. A simple remedy is to erode the detected regions before pixels are extracted. Doing this eliminates most outliers and improves the results as Fig. 3.5




(a) (b)

FIGURE 3.6. Clouds obtained when two erosions are applied (a), which is too drastic as (b) shows. Moreover, some outliers still remain in (a) especially for green.

shows. Nevertheless, some still remain (see the red cloud in Fig. 3.5 (b)). A single erosion is perhaps not sufficient to correct for border inclusion, but doing two is too severe as Fig. 3.6 shows. Pixels should be retained as much as possible on objects of interest, but without retaining those that are unreliable (*i.e.*pixels clearly not on the object). It is in this sense that using two erosions is too severe, since too many pixels on the objects are eliminated. Moreover, using two erosions would not completely solve the problem, since some outliers still remain (some green values in Fig. 3.6 (a)). Another cause of unreliability is clipping as pointed out by Novak *et al.* [**99**] [3]. A clipped pixel value is easily detected and eliminated by verifying if one of the color bands has saturated. Of course, image acquisition should ideally be made so that no clipping occurs, but this is not always possible, and so clipping must be taken care of (clipping problems were discussed in Section 2.7). In the present case, clipping did not occur due to specular reflection (because the samples are flat), but simply because the samples (especially the yellow ones) were too bright at times when outdoor illumination was strong.

[3]To borrow from their words, "anytime the measured pixel value is equal to the camera maximum in one or more (color) bands, this value ... should be considered suspect".

With clipped values eliminated, if any, the fact that one step of erosion does much of the clean-up suggests that border inclusion is very likely the main cause for the appearance of outliers, but probably not the sole[4]. At this point, it is difficult to find another reason for the occurence of outliers, or to control their appearance. Henceforth, the remaining ones will be taken as random events, and at this stage it is reasonable to resort to statistics to further sieve the data. This is done by calculating the covariance matrix of the $(p, q)$ pixel color values on each detected and validated object, and by using the Mahalanobis distance [**103**] to discard values that are too distant from the mean of the cloud associated to that object. A Mahalanobis distance of 3.0 was employed. For 2-D data, and assuming Gaussian statistics this corresponds to covering more than 99% of the data. A dimension of color that has not been used so far to filter the data is color intensity $I$ (Eq. 2.4.2). Statistical filtering on this variable has also been performed. Pixels having color intensities that differ too much from the mean color intensity are discarded. A bound for this difference was set as 2 standard deviations, which for 1-D data and Gaussian statistics corresponds to cover more than 99% of the data. Fig. 3.7 shows the result when one erosion is used, clipped values are eliminated, and statistical filtering is performed.

This completes the different aspects that needed to be discussed so that reliable color data be automatically extracted from a single image. The next step is to extract data from a large number of images, and this is the subject of the following section.

## 3.3 Calibration Results

A large number of training images will now be processed with the autonomous data extraction procedure just described to obtain color values from which color space regions will be generated. The training set is composed of 141 images similar to that of Fig. 3.3 (a) and captured under a wide variety of illumination conditions during the years 1998 and 1999 (winter, spring, summer, fall, sunny, cloudy, morning, mid-day, afternoon, *etc...*). The clouds of points obtained from these images are illustrated in Fig. 3.8 (a). It is seen in Fig. 3.8 (b) that there is a small amount of overlap for the red and brown clouds, and for

[4]As a matter of fact, in the present case, the color samples on the chart were held to the black board by screws. These appear as tiny black spots at the top corner of each color sample, which the reader may have noticed, see Fig. 3.5 (f) for instance. Pixels on these spots are frequently included by the region growing, and often cannot be eliminated via erosion.

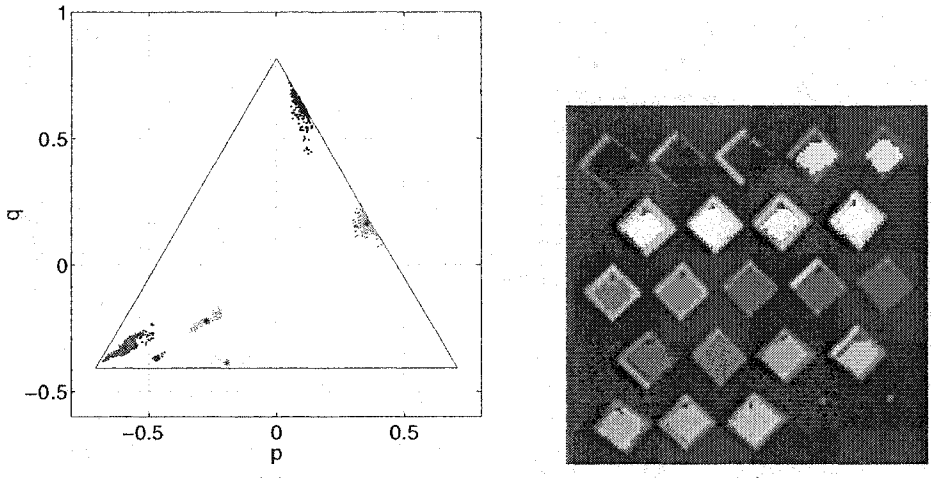



FIGURE 3.7. (a) Results when one erosion is used, clipped values are eliminated, and statistical filtering via the covariance matrix and a Mahalanobis distance of 3 is used. Statistical filtering on color intensity with 2 standard deviations is also performed. As seen in (b), the color samples are now better covered, compared to Fig. 3.6 (b).

the red and orange ones. The color space regions associated with the clouds of the different colors obtained from the calibration procedure were determined in the same manner as for the clouds obtained from the theoretical model, *i.e.*convex hulls, and Fisher's linear discriminant approach for the clouds for which there is overlap (see Section 2.6.1). These regions, to be called the calibrated color regions, are depicted in Fig. 3.8 (c). Convex hulls are sufficient here because no concavities are apparent in the clouds[5].

In Section 3.2.1, it was mentioned that the broad detection regions should ideally not constrain the data to be gathered during the calibration process. In Fig. 3.9, the calibrated regions are depicted with the contours of the broad detection regions (Fig. 3.2). Recall that the latter were determined using the guidelines enunciated in Section 3.2.1. They are seen to fullfill the previous requirement since the calibrated regions are well within the boundaries of the broad ones, except perhaps for the brown and orange calibrated regions. However, in these cases, the boundaries only graze those of the corresponding broad regions, indicating that the broad boundaries must be approximately coinciding with those of the

[5]However, in other applications where the present calibration approach might be used and in which concavities in the clouds are apparent, it may be necessary to determine the contours more precisely. Determining the outline of a set of points when concavities need to be accounted for is a non-trivial problem, which does not admit a unique solution[**109, 110, 111, 112**].

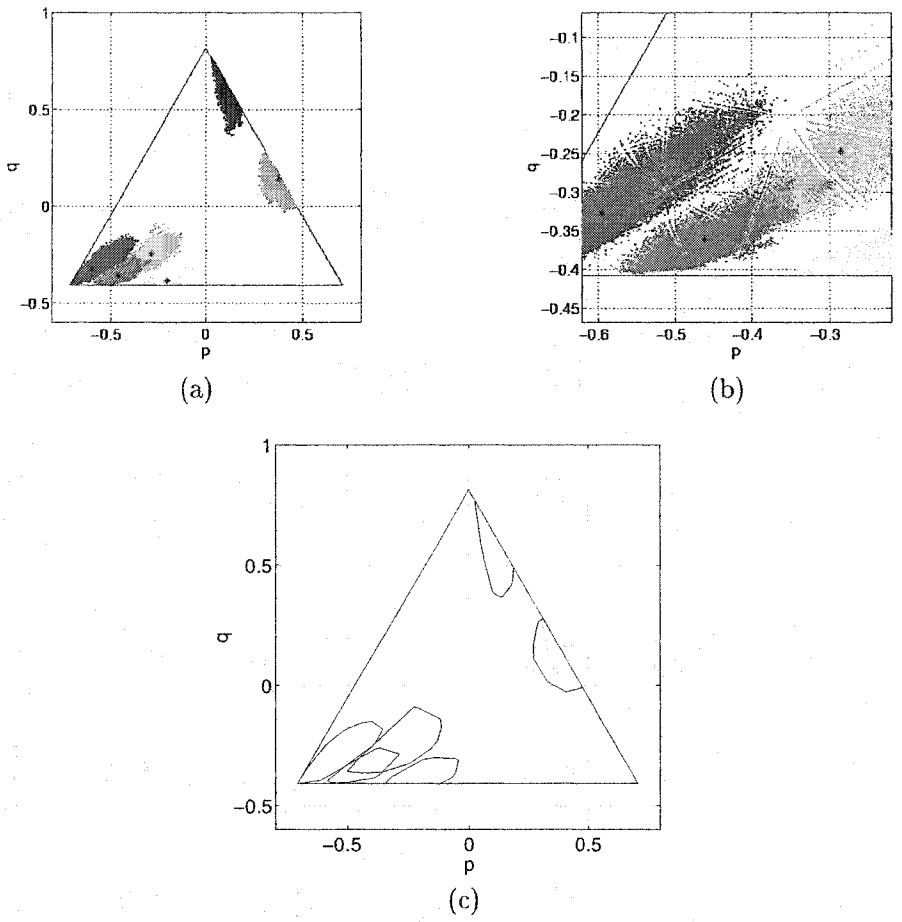



FIGURE 3.8. (a) Clouds of points obtained from the automatic data extraction procedure. (b) Illustrating the little degree of overlap between the red and brown clouds and the red and orange ones. These were separated based on Fisher's approach as was done in Section 2.6.1 for the brown and achromatic clouds of the theoretical predictions. (c) Contours of the color regions obtained from the clouds in (a). Note that in (b), the star-shaped lines are due to quantization effects since the $C_R$, $C_G$ and $C_B$ assume discrete values, and hence the $(p, q)$ values as well.

real data where grazing occurs. Had the calibrated color regions been constrained to a significant degree, the calibrated boundaries would then not just be grazing those of the broad regions, but would instead be clipped (not to be confused with saturation clipping). In fact this can serve as a criterion for determining if the enlarged regions were designed appropriately. Significant clipping indicates that the enlarged regions are too restrictive. In such a case, the calibration procedure should be ran again with enlarged regions made broader without contradicting the previous guidelines. If this is impossible, then it means

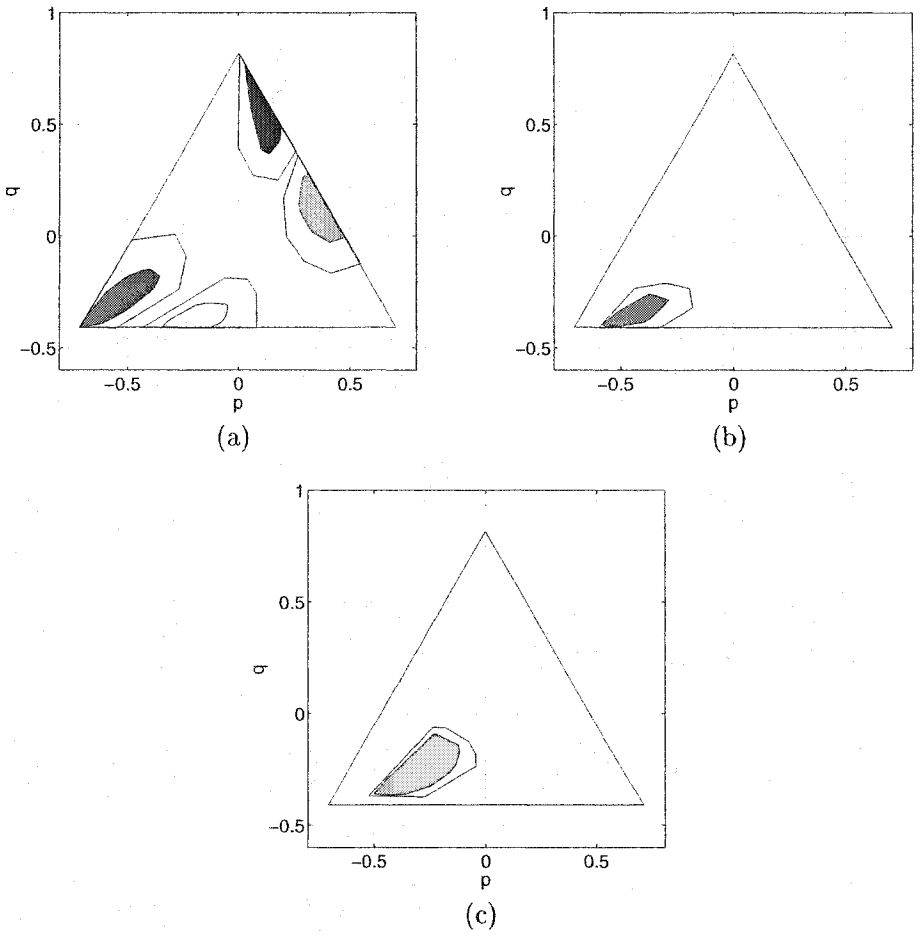



FIGURE 3.9. Calibrated regions compared to the enlarged ones. In (a) these are shown for blue, green, red, and yellow, in (b) for orange, and in (c) for brown.

that the colors that need to be discriminated are too close in color space with respect to the discrimination allowed by the camera used. Either should these colors be considered as a single one, or simply a better camera with more quantization levels in each color channel should be resorted to, when available[6].

A last comment on the choice of the enlarged detection regions. It is seen that the details of their exact shape are unimportant, for changing them by not too large amounts

[6]With the advent of always better cameras, color discrimination capabilities will keep on improving, since having more quantization levels (more "bits per pixel" in the jargon) allows for finer discrimination between nearby colors.

would lead to the same clouds of points, provided there is no region clipping, since the clouds are located within the broad regions, and are not constrained by them.

## 3.4 Experiments

This section presents results and statistics on the detection efficiency when the calibrated color regions of Fig. 3.9 are used to label pixels in images.

To obtain statistics on the performance figures, the approach has been tested on a set containing 272 images (the "testing" set) captured under a wide variety of illumination conditions. The results are presented in Table 3.1. A small amount of images in the testing set had clipped pixel values on some of the color samples (column three of the table), mostly for yellow and orange samples which are brighter; clipping in the red channel was then involved. In such cases, the method can do nothing. This is the reason for giving statistics with and without samples that suffered from clipping in columns five and six. As described previously in Section 2.6.2, Otsu's method was used for separating orange and brown. At times, it could not threshold the orange-brown color intensity histogram appropriately (column four), which resulted in some detected orange samples being wrongly labelled as brown, and vice-versa (to a lesser extent). These cases were considered as failures.

As Table 3.1 shows, overall the method gives high detection rates. For the brown samples, the results are somewhat lower than those for the other colors. The reason for this is that the brown samples are darker, and are thus more affected by noise and also by specular reflection[7]. Nevertheless, there was still a partial detection in 47.3% of the cases in which they were not adequately detected (by not adequately deteted is meant that a significant amount of pixels on the sample are not detected), and as such these are not complete detection failures. Fig. 3.10 shows such a case.

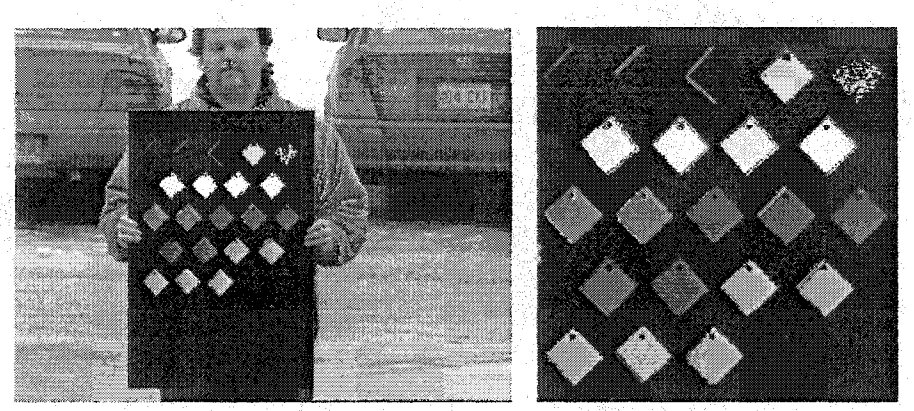
The next figures to be described show detection results with the calibrated color regions for some of the images that were used in Section 2.7 of the previous chapter, and for which detection with the model-based regions had some problems (missing pixels). This is to illustrate the improvements brought by the calibrated regions. For comparison, the results of the previous chapter are shown again, and the figure captions give the corresponding

[7]To be more precise, since specular reflection is independent of the darkness of the body color of an object, then the darker the body color, the higher specular reflection becomes relative to body reflection. Hence, the more towards achromatic colors do the dark objects tend in the presence of specular reflection.

| | nb. of samples | not detect. (total) | not detect. due to to clipping | wrongly labeled by Otsu's method | % detection (all) | % detection clip. samples excl. |
|---|---|---|---|---|---|---|
| blue | 816 | 3 | 0 | NA | 99.6 | 99.6 |
| brown | 544 | 53 | 0 | 0 | 90.3 | 90.3 |
| green | 1360 | 2 | 0 | NA | 99.9 | 99.9 |
| orange | 544 | 26 | 24 | 2 | 95.2 | 99.6 |
| red | 1360 | 26 | 0 | NA | 98.1 | 98.1 |
| yellow | 1088 | 66 | 57 | NA | 93.9 | 99.1 |

TABLE 3.1. Results obtained by testing the detection approach on a set 272 images using the calibrated color regions. The chart, already shown in several figures, is composed of 3 blue samples, 2 brown, 5 green, 2 orange, 5 red, and 4 yellow. This is why the number of samples varies depending on color. Column two gives the total number of samples that were not detected for each color. From the samples that were not detected, column three gives the number of those which were not detected due to saturation. Column four, which only applies to orange and brown, shows the number of samples that were wrongly labeled by Otsu's method. Columns five and six give percentages associated to the preceding values. In five, clipped samples are considered as failures, whereas in six these are excluded from the statistics.

figures in the previous chapter. Consider Fig. 3.11 (a), which shows the detection results using the calibrated color regions for the same image as in Fig. 2.14. In (a-1), a close-up on the chart of these results is given, and (a-2) gives the results that were obtained for that image in the previous chapter. Notice how the yellow samples (in particular the leftmost yellow one) are now more completely detected (*i.e.*more pixels detected towards the border), and also how the pixels that were missed on the red samples of the previous to last row are now detected. The same sequence of results is given in (b), (b-1), and (b-2) for the image that appeared in Fig. 2.16 (c). In this case, the improvements are more pronounced. The approach with the calibrated regions is much more robust, and gives better detection. Notice in (a-1), and (b-1) that there are superfluous pixels at the border of the samples, but with some work which depends on the application, these can be cleaned. A technique for this will be described in Chapter 5 on road sign detection. Figs. 3.12, 3.13, 3.14, and 3.15 (a) show similar results under different illumination conditions. In those cases, the problems were mainly with the brown samples. In particular, the top right brown sample in Fig. 3.12 (b-2) is hardly detected, whereas a large portion of it is recovered in (b-1). In this case, the sample is even difficult to see with the human eye; there is almost no contrast with the black background. Fig. 3.15 (b) depicts an example where detection with the



(a) (b)

FIGURE 3.10. Example where the top-right brown sample is considered not adequately detected, but for which there is nevertheless partial detection.

model-based color regions had difficulties with several samples. Figs. 3.16, and 3.17 show cases were the problems were even more severe. Detection with the calibrated color regions has resolved these problems.

As mentioned earlier, the color samples considered correspond to traffic sign colors. As a preview to later chapters concerned with road sign detection and recognition, Fig. 3.18 shows a few detection results when the calibrated color regions are used for road sign detection. It is these regions that will be used in the road sign detection and recognition system to be described in Chapter 5.

## 3.5 Conclusion

In this chapter, a procedure for calibrating the color regions obtained from the physics-based model of the previous chapter has been presented. Experimental results have shown that the calibrated regions provide for efficient detection under a wide gamut of daylight illumination conditions. These regions will be exploited when road-sign detection is considered later in the thesis. The main contributions of this and the previous chapters are to provide a theoretical understanding of color formation under daylight, and to exploit this to arrive at a hybrid method conciliating the best of both modelling and the use of real data as in learning from examples. The model part of the approach embodies a priori knowledge while remaining impredictabilities and modelling errors are taken care of by the calibration procedure. In a way, the calibration technique developed in this chapter to obtain color

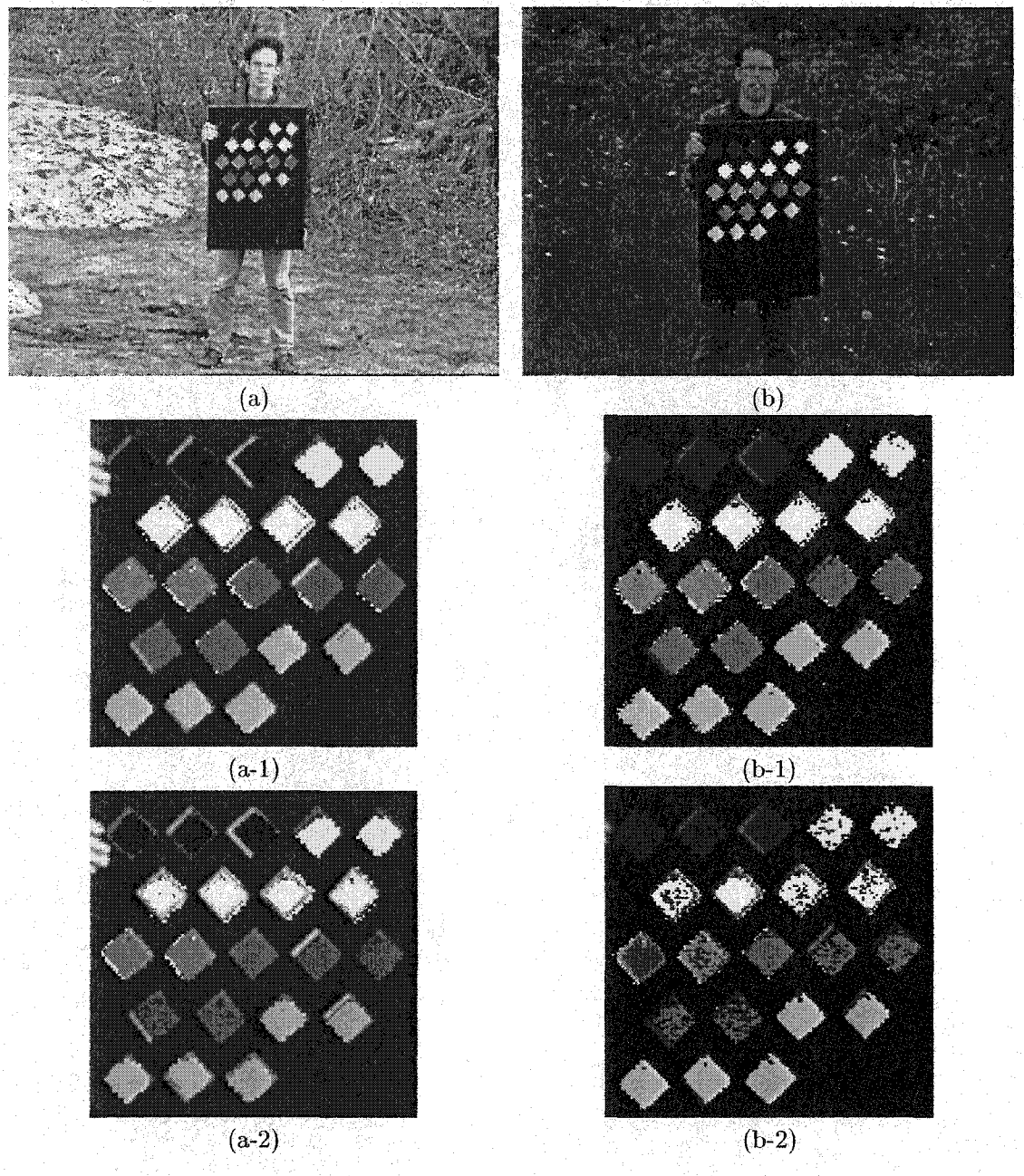



FIGURE 3.11. (a) Detection with the calibrated regions for the same image as in Fig. 2.14 (b) of the previous chapter. (a-1) Close-up of the results in (a). (a-2) Results obtained in the previous chapter for comparison. The sequence for (b) is the same as for (a), but for the image of Fig. 2.16 (c) of the previous chapter.

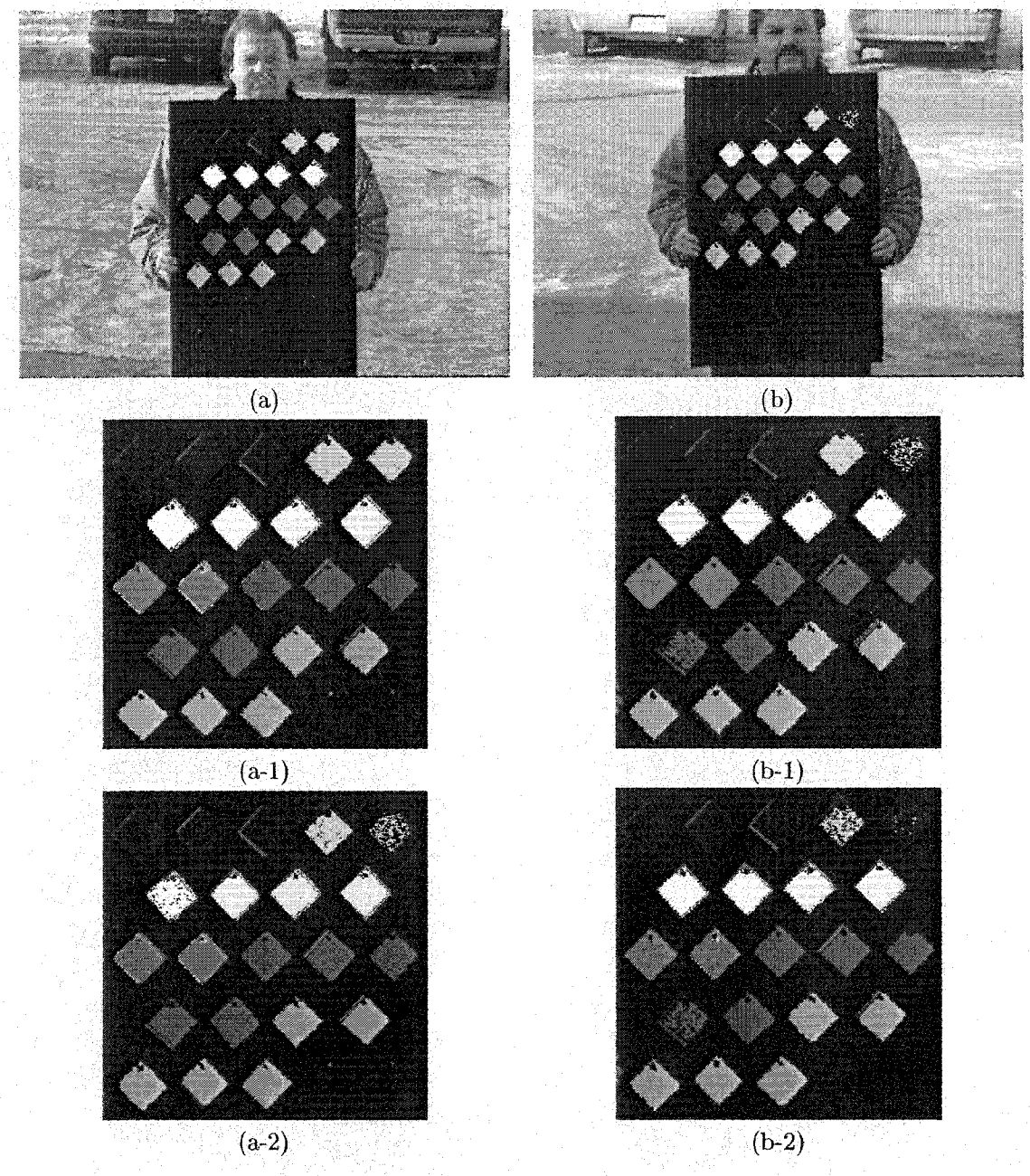



FIGURE 3.12. Same sequence of results as described in Fig. 3.11. (a) and (b) respectively correspond to Fig. 2.17 (a) and (f). For the top right brown sample in (b-1), although detection is not perfect, there are definitely more detected pixels than there were before with the model-based color regions. The leftmost red sample on the previous to last row is also not perfectly detected, but only little holes are present.

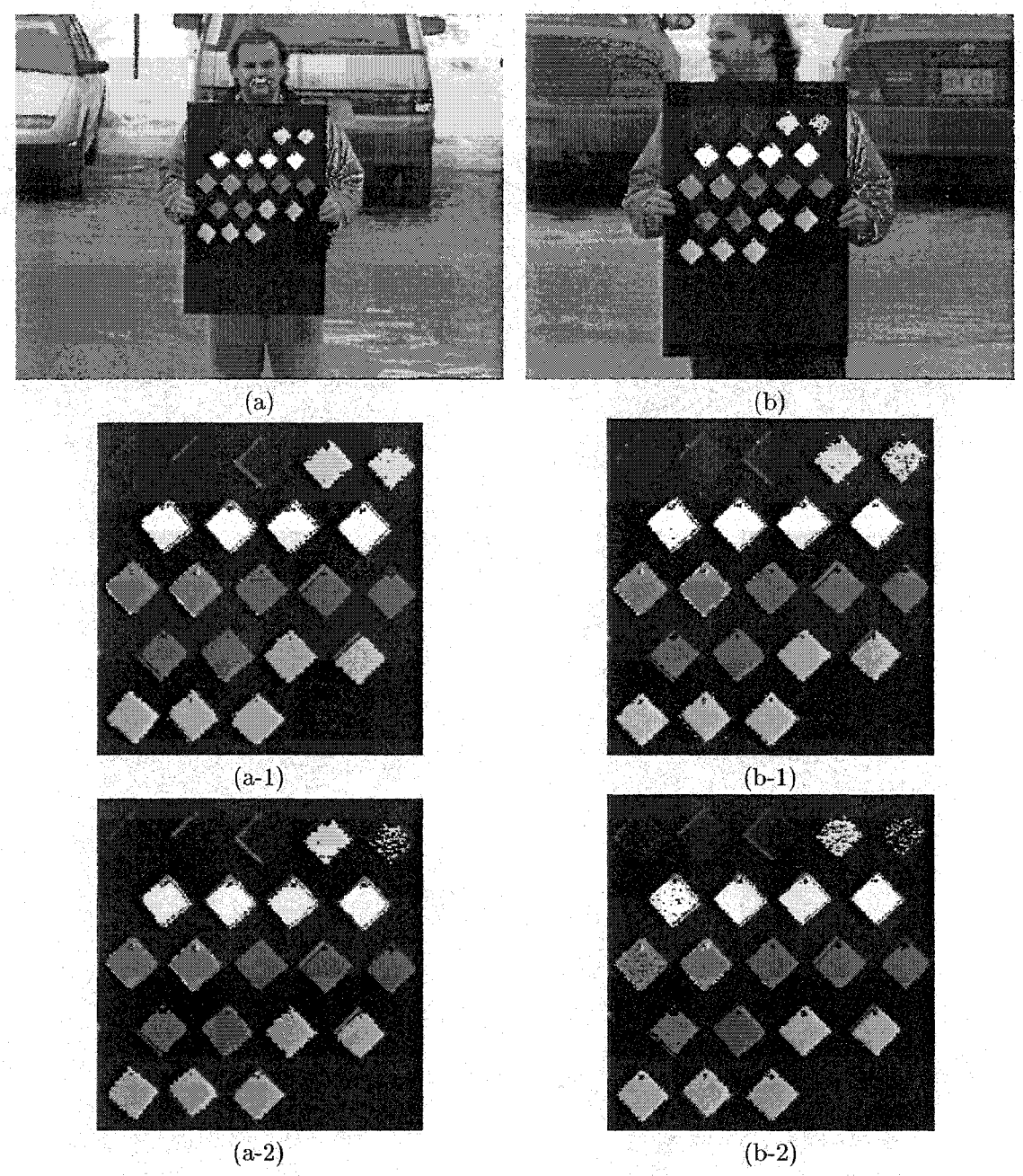



FIGURE 3.13. Same sequence of results as described in Fig. 3.11. (a) and (b) respectively correspond to Fig. 2.18 (c) and (e). Mainly problems with the rightmost brown sample were present with model-based detection.

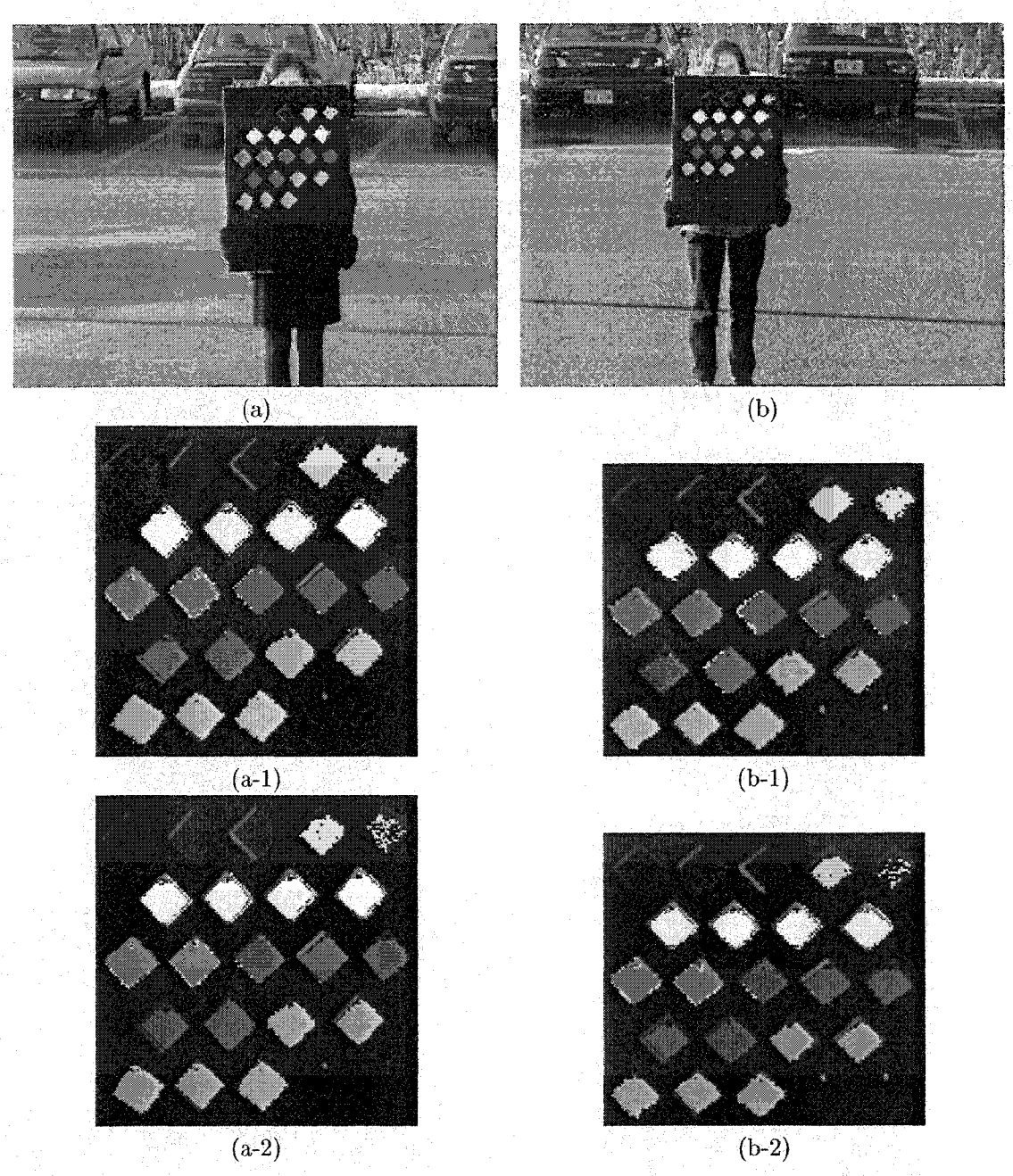



FIGURE 3.14. Same sequence of results as described in Fig. 3.11. (a) and (b) correspond to Fig. 2.19 (d) and (f) respectively. Mainly problems with the rightmost brown samples were present with model-based detection.

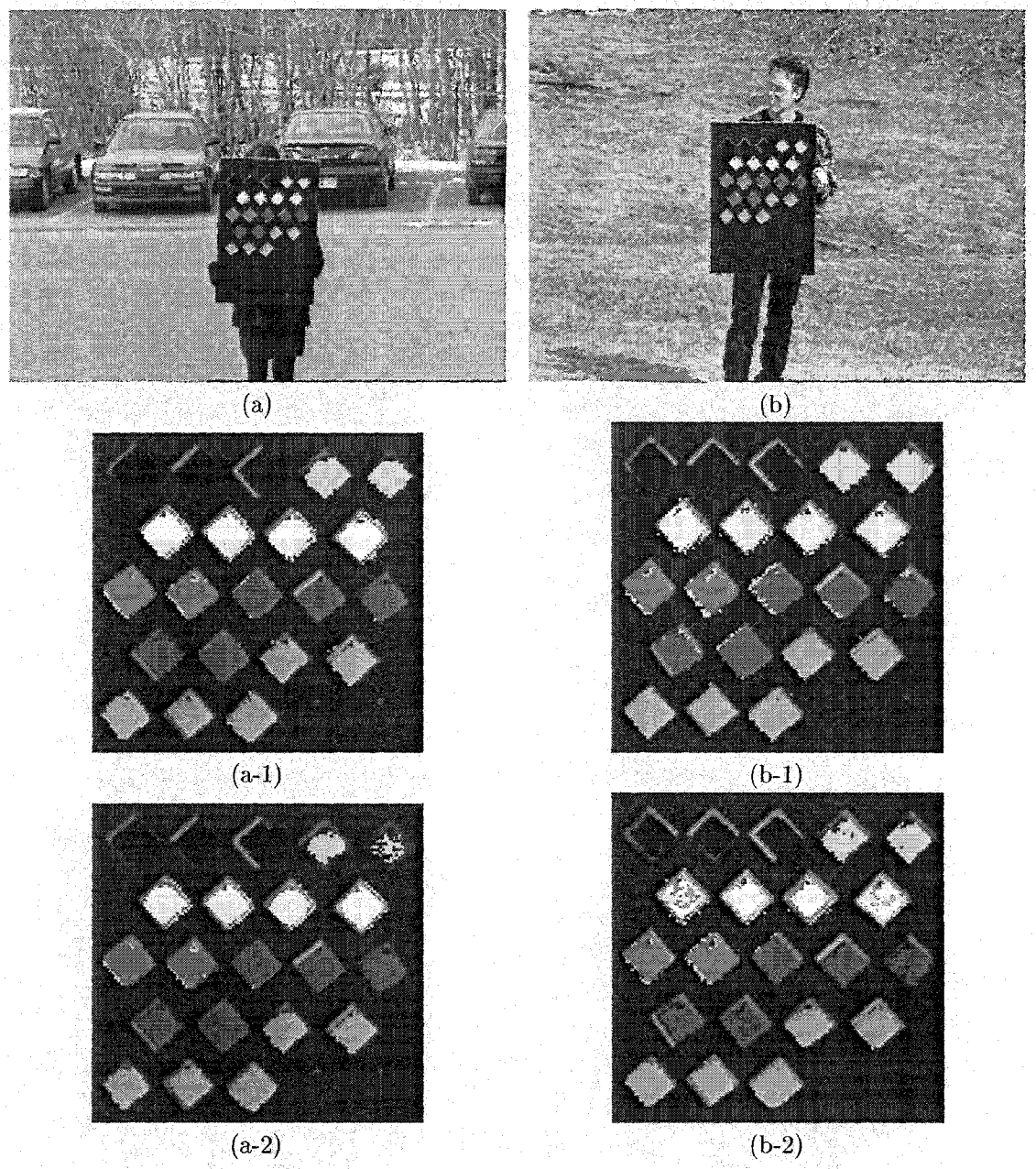



FIGURE 3.15. (a) corresponds to Fig. 2.20 (b), and (b) to Fig. 2.22 (a). In (b), model-based detection missed pixels on several of the samples, which is no longer the case with use of the calibrated color regions.

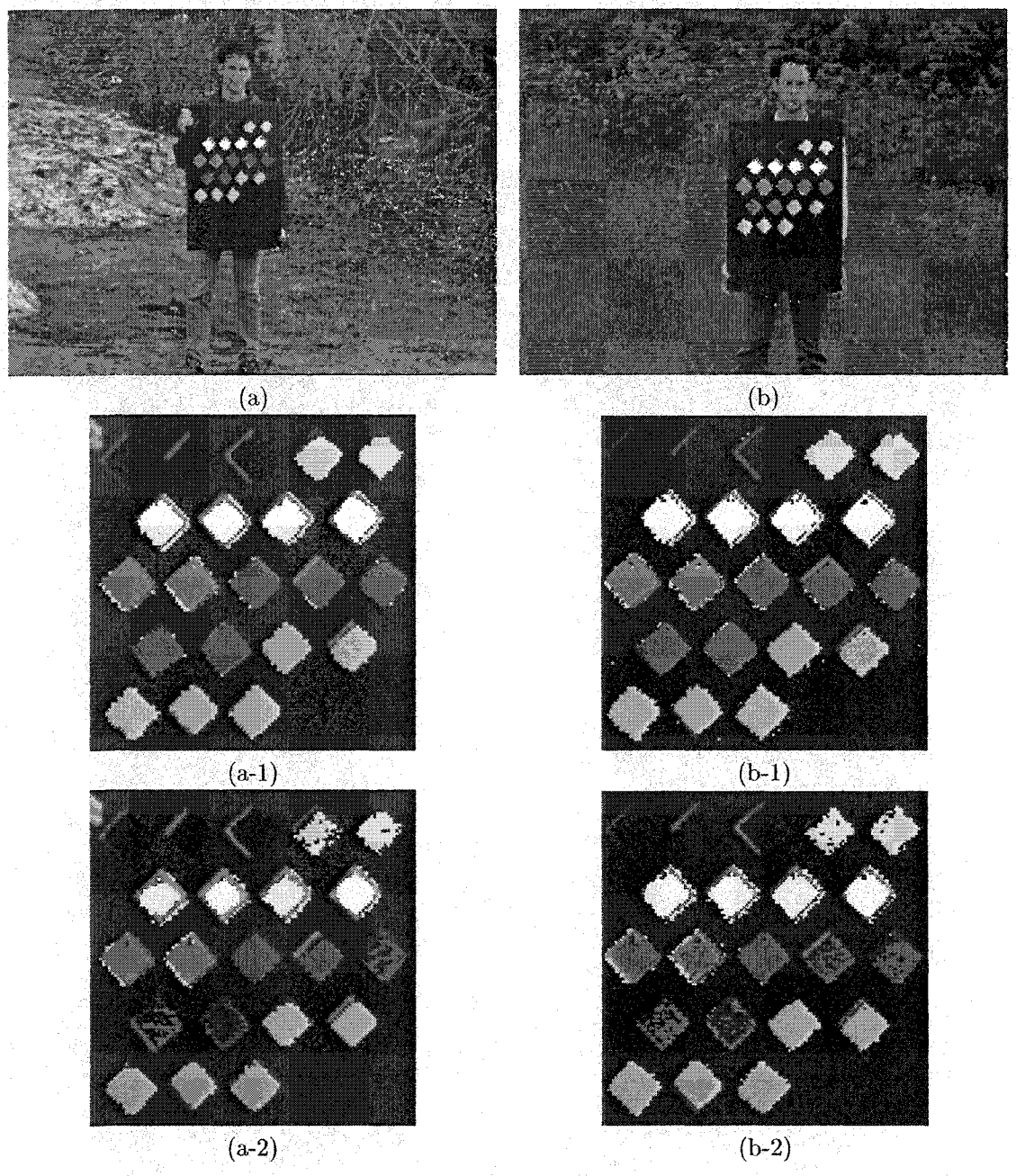



FIGURE 3.16. (a) and (b) correspond to Fig. 2.23 (a), and (c) respectively. Fig. 2.23 (b) and (d) are reproduced here in (a-2), and (b-2) respectively. Model-based detection suffered from severe problems for red and orange samples. Also, when (a-1) and (a-2) are compared, the yellow samples are more completely detected towards their border in (a-1).

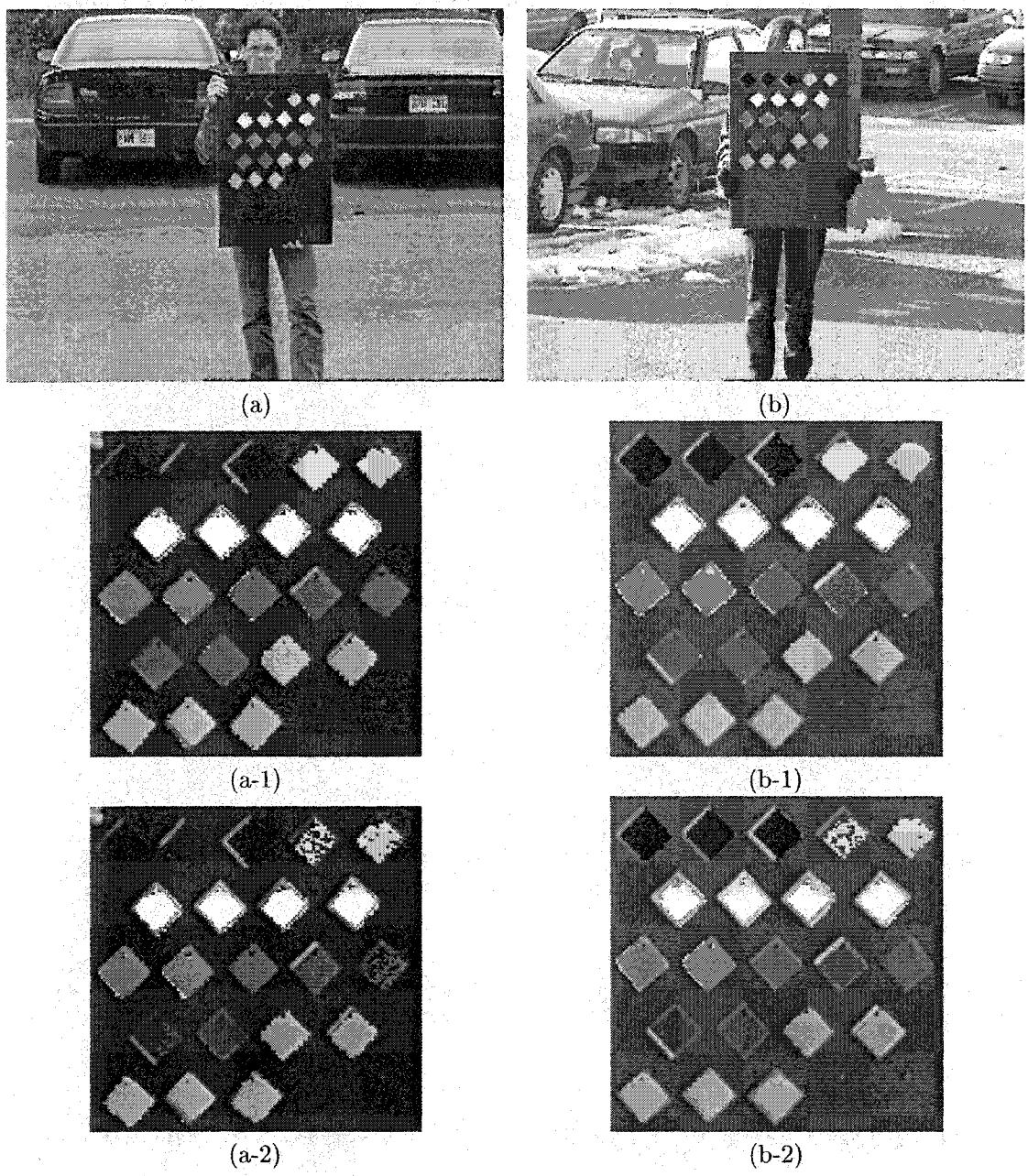



FIGURE 3.17. Similar problems to the previous figure, but for different illumination conditions. (a) and (b) correspond to Fig. 2.24 (a) and (c) respectively. Fig. 2.24 (b) and (d) are reproduced here in (a-2) and (b-2).

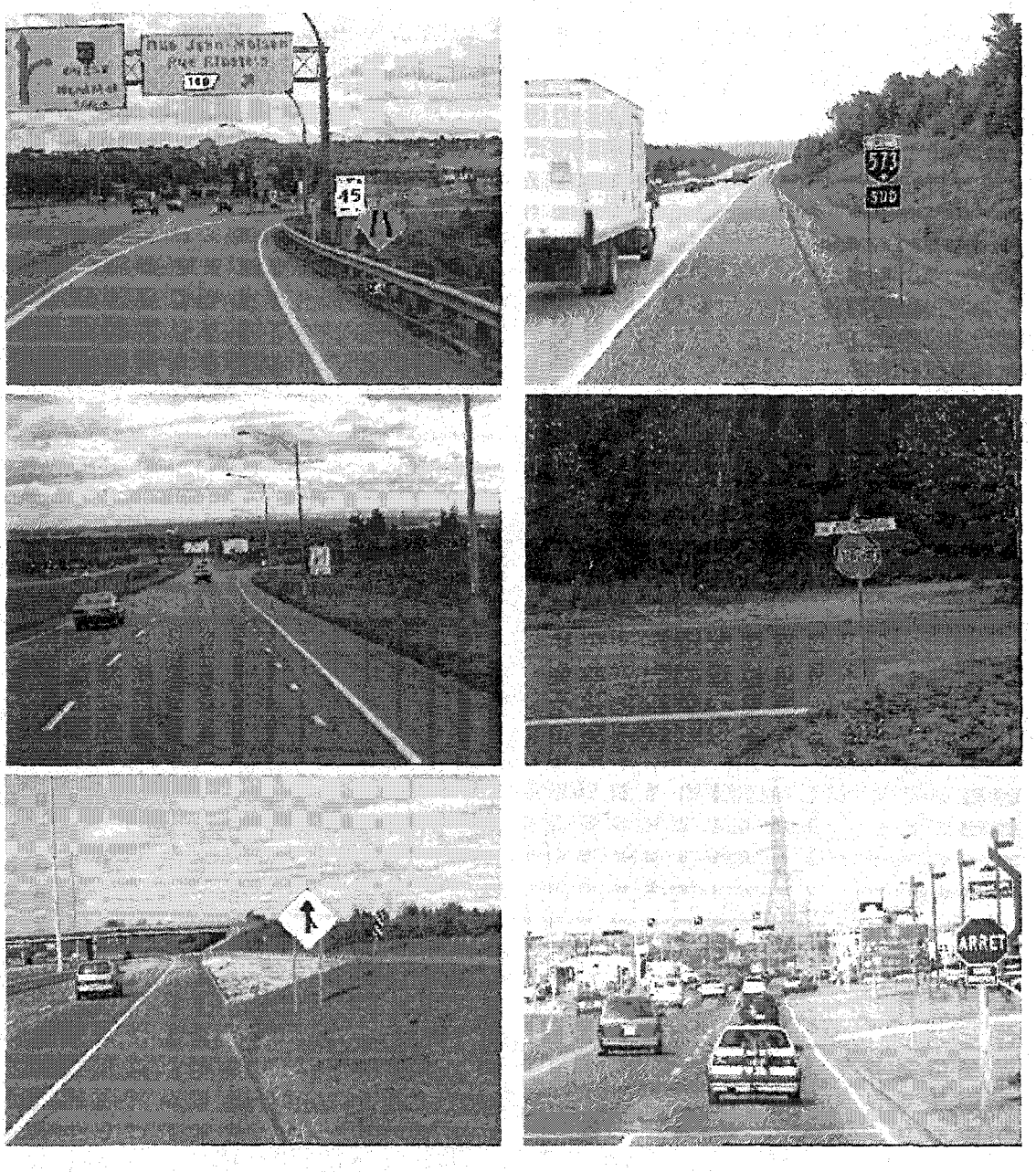



FIGURE 3.18. Use of the calibrated color regions to detect road signs. The images were captured under various sky conditions. In the top left image, the road sign highlighted in magenta is an orange construction road sign. Yellow road signs are highlighted in white.

space regions can be thought of as a learning approach, but with the fundamental difference that the initial data is obtained from a model rather than having been determined manually by an operator. Real data is exploited at a higher level here since it is used to refine a model, whereas in learning it is used right from the start. This distinction is of significant importance because in the present approach, the initial conditions (*i.e.*the data to start with) are precisely and reproducibly determined since a priori information in the form of a physical model is being exploited, whereas this is not the case when the initial data are determined by an operator, since from one operator to the other there may be variations which may hardly be reproduced.

The fact that the calibration method is highly autonomous constitutes a definite advantage over learning approaches (be it supervised or unsupervised approaches) requiring examples to be extracted prior to train the neural network with those examples. This examples extraction phase generally requires the intervention of an operator in a significant manner. The approach developed here cannot, however, be considered completely autonomous for a few reasons. First, to determine the color space regions in the modelling part of the approach, the spectral reflectances of the considered objects must be measured as well as the spectral sensitivities of the camera used. Also, a maximum value for $\kappa$ (Eq. 2.4.12) must be provided to the model. Apart from these, the model requires no further information. As regards the calibration part, the broad regions used in the region growing are determined by hand. However, once these are determined, the calibration procedure can be launched and work on its own. In that sense it is truly autonomous, and once set, it allows for the processing of large amounts of images for the extraction of data, much larger than what can reasonably done manually. Finally, concerning the application of Fisher's approach to separate some of the color clouds, in the present system this has not been automated, and the algorithm must be told by an operator which clouds to separate (this requires a minimum effort, but nevertheless needs to be done). However, this could be automated by first determining if two clouds overlap, *e.g.*by finding if their convex hulls intersect. In the affirmative, the Fisher separation algorithm could be launched on these two clouds.

# CHAPTER 4

# Bit Encoded Labelling and Quasi-Invariant Spatial Pattern Matching

## 4.1 Introduction

In many computer vision problems, once objects have been detected on the basis of certain attributes, they usually need to be further analyzed by a classification module. This is necessary to both filter out objects whose attributes fulfill the detection criteria, but which are not objects of interest, and to classify the ones of interest if some are found. In short, validation and possibly classification must be performed. As an example, road sign recognition is the practical problem considered in the present work, and color is the attribute of interest for detection. Obviously, other objects may possess road sign colors in an image, and thereby be detected. Shape attributes must clearly be resorted to at some point in the validation and classification of the detected objects. Road signs constitute rigid planar shapes. It is thus natural to consider an appearance-based template matching approach (also called matched spatial filtering) for their identification. This has motivated the development of the template matching technique described in the present chapter, although the technique could be applied to objects other than road signs as well.

Appearance-based template matching has proved to be a popular approach for determining if an image contains instances of predetermined objects taken out of a database. This has been extensively studied in the context of monochrome images (gray-scale or binary). A measure often employed for computing a match between part of an image (sub-image)

and an image of a given object in the database (called a template[1]) is the normalized cross-correlation between the two. The rationale behind this is that large values of the correlation should indicate the presence of the object in the input image[2]. Correlation-based approaches date back to the infancy of image analysis [**103, 115**]. An interesting comparative study appears in [**116**]. These techniques have remained a vigorous topic of research in recent years (see [**117**] and the references therein). Another approach that was discussed in the introductory chapter is that based on distance transforms [**49**]. Correlation-based techniques have also attracted much interest in the optical pattern recognition community due to the availability of optical correlators [**118**].

In many applications, objects of potential interest are extracted from the image prior to being compared with those of the database for finding a match. The situation considered here is when objects are extracted as sub-images contained in bounding boxes obtained through pixel labelling followed by a grouping of neighboring pixels having the same label. An example of this is when color labels (or names) are attached to pixel values in color images as was described in the previous chapters. The question that then arises is how to directly use the labelled sub-images instead of resorting to a gray-scale or binary image representation for finding matches as is customarily done. For instance, in the case of color images, only intensity information is usually considered for computing matches [**44**], and so the color information and its spatial distribution is in large part lost. In the present work, color information plays an important role, and the main motivation is to find some means of further exploiting it in an appearance-based template matching approach based on a correlation computation. To the author's knowledge, the only work that has been done in this area consists in applying the correlation techniques alluded to above independently in different color planes. For example, in [**55**], a color image is first separated into its $R$,

[1] In optical pattern recognition, the template is called the filter. Both terms will be used interchangeably in this introduction.

[2] What is referred to here should not be confused with moment-based template matching, which also relies on calculating a correlation (or more precisely a convolution, which is equivalent). In this case the filter is designed to extract some moment from the image, the correlation function representing the value of that moment. The matching is then performed by comparing combinations of such moments with those of a database and representing particular objects. See [**113, 114**] for examples. This is an instance of feature-based matching in which more abstract features of an object than its spatial appearance are used for the matching. This type of matching will not be considered in the present work, although the correlation computation that will be developed here could be adapted to compute moments. However, from now on template matching will be used with the meaning of being appearance-based.

$G$, $B$, and intensity ($I = R + G + B$) component images, which are gray scale, and given that a certain object is searched for, these images are correlated with a filter representing that object. This results in four correlation images of the same dimensions as the original color image. Then the assumption is that if the object is present in the initial color image, then there should be a correlation peak at the object's location in each correlation image. By thresholding the correlation images to obtain binary images and ANDing these, non-null values in the resulting binary image should determine where the object is located. A problem with this is that if the object's color is such that it generates no signal or only a faint one in one of the color planes (for example a red object will not lead to a strong signal in the blue component image), then no peak will appear in the corresponding correlation image, and the final image will not contain non-zero values at the object's location[3].

In this chapter, an efficient template matching technique that uses the labels assigned to each pixel in the object extraction phase will be described. This technique will play a major role in the classification module of the road sign identification system to be elaborated in Chapter 5[4]. The key to the approach resides in a novel encoding of the different labels based on which a correlation-like computation can be defined and used for finding matches. This encoding allows to further exploit the information conveyed by the labels and their spatial distribution in such a way that different labels cannot be confounded when computing a match, as is the case when intensity values are used (for instance, two different colors may have the same intensity). More generally, it will be realized that this technique can be used whenever rigid planar shapes are considered, and their detection involves a labelling of pixels in terms of one or more attributes, of which color is an example.

A basic problem with correlation-based template matching is that in general it will be difficult to recognize a pattern subject to geometrical transformations (*e.g.*rotations) with a single template, unless the distortions introduced by these transformations are small. This is the so-called invariance problem, to which much effort has been devoted (see the review article [**119**] and the references therein). Without entering too much into the details, some approaches resort to a change of coordinates to convert a given geometrical transformation

[3]Also, such an approach is subject to scale problems, requiring several filters at different scales in order to recognize an object, which is a severe drawback.

[4]Note, however, that the classification module will not solely rely on template matching. Color as such, and other shape attributes will also be used.

to a translation in the new coordinate system [**120, 121, 122**], *e.g.*in log-polar coordinates $(\log r, \theta)$ a rotation by an angle $\theta_0$ in cartesian coordinates corresponds to a displacement by $\theta_0$ along the $\theta$ axis, whereas a stretch by a factor $\alpha$ corresponds to a displacement by a distance $\log \alpha$ along the $\log r$ axis. The idea for changing to coordinates where transformations correspond to translations is that the correlation function is only translated as a whole under translations of the underlying coordinate system, its values remain unchanged. While such approaches allow to gain invariance with respect to changes in some geometrical parameters, invariance is lost with respect to other parameters, *e.g.*in the previous example by changing to log-polar coordinates, invariance to translations in cartesian coordinates is lost. For a given group of geometrical transformations, the required coordinate change can be determined with the help of Lie group techniques [**123**]. Another type of approach is to resort to moments or descriptors that are invariant to geometrical transformations [**124**]. Such descriptors are obtained from the coefficients of the decomposition of an object onto a set of basis functions (discrete or continuous). The basis functions are chosen according to the required invariance. For instance, for rotation invariance, circular harmonic functions are used (which are no more than complex exponentials with integer frequency, and in fact the decomposition in this case corresponds to a Fourier expansion of the angular dependence of the considered object expressed in polar coordinates), and for scale invariance the Mellin transform is resorted to. Projection invariance has also been considered in [**125**]; in this case logarithmic harmonics are used.

In the present approach, it will be shown that the proposed encoding allows templates to be combined, and thereby tolerance to geometrical transformations to be embedded in the templates in a manner similar to the composite filters used in optical pattern recognition [**126**]. Moreover, the possibility of combining templates will also allow to construct templates that are representative of classes of objects, which is a novel idea brought by the present work in the context of template matching. This allows to determine if an object belongs to a class of similar objects using a correlation-based approach, and will be exploited in the recognition of road signs. This is a novel approach that has not been studied in the literature.

In this chapter, the focus will be on the template matching technique itself and the principles underlying it, and not on search and retrieval from the database which is application specific, and which will be discussed in detail in Chapter 5 for the case of road sign recognition. Moreover, extensive results on the efficiency of the template matching approach will also be presented in the context of road sign recogntition in Chapter 5, as this provides a real framework to test the approach.

## 4.2 Overview

The structure of this chapter is as follows[5]. In next section, a brief review of correlation and why it is considered in pattern recognition will be made. Some comments will be made regarding the use of correlation in the context of color processing. Based on these preliminaries, a technique for encoding color labels will be presented. In Section 4.4, an example of how a sub-image and a template are encoded will be given for a real case. Section 4.5 will present operations that can be performed on the encoding previously introduced leading to the definition of a correlation computation compatible with that encoding. A discussion on how this encoding can be exploited to obtain quasi-invariance of the correlation with respect to geometrical transformations, as well as to define classes of objects will follow in Section 4.6. In Section 4.7, avenues for generalizations of the approach will be presented, and Section 4.8 will present experimental results (more results will be given in Chapter 5). Section 4.9 will conclude.

## 4.3 Correlation in Template Matching and Label Encoding

To provide some background to the template matching technique that will shortly be developed, the motivation for using the usual correlation function in template matching will first be reviewed.

Since it is assumed in the present work that sub-images have been extracted and resized to the same size as the template prior to computing a match (in this manner scale problems are eliminated), only correlation at the origin needs to be discussed. Given two real-valued functions $I(x, y)$ (to be seen as the sub-image) and $T(x, y)$ (to be seen as the template) defined on the same grid, a standard matching measure is the $L^2$ norm (or Euclidean

[5]Part of the work in this chapter has been presented in [105].

distance)

$$M = \sum_{(x,y)} (I(x,y) - T(x,y))^2. \tag{4.3.1}$$

For a match, this measure should have a small value. Expanding out the right-hand side of this equation, gives

$$\begin{aligned} M &= \textstyle\sum_{(x,y)} I^2(x,y) \\ &+ \textstyle\sum_{(x,y)} T^2(x,y) \\ &- 2 \cdot \textstyle\sum_{(x,y)} I(x,y) \cdot T(x,y). \end{aligned} \tag{4.3.2}$$

In this expression, the first two terms are the image and template energy. They influence the value of the match (in absolute terms), but do not convey any information about whether or not the image and the template bear some similarity. They simply act as biases on the absolute value of the match[6]. The interesting contribution to the measure comes from the sum in the third term, which is the correlation of $I$ and $T$ to be denoted by

$$(I \odot T) = \sum_{(x,y)} I(x,y) \cdot T(x,y). \tag{4.3.3}$$

Clearly, when the value of the measure $M$ is small, this term is large. Hence, large values of the correlation should be looked for in order to find matches. This is what motivates the use of correlation as a means for measuring matches. Related to the correlation is the normalized correlation defined as

$$Cor(I,T) = \frac{(I \odot T)}{\sqrt{(I \odot I)}\sqrt{(T \odot T)}}. \tag{4.3.4}$$

The normalized correlation is often used instead of the correlation in order to eliminate the energy bias alluded to above, the quantity $F \odot F$ being the energy in $F$. An important consequence of the Cauchy-Schwartz inequality is

LEMMA 4.1 (Cauchy-Schwartz Inequality for Correlation).

$$|Cor(I,T)| \leq 1. \tag{4.3.5}$$

The question that shall now be investigated concerns how the information conveyed by labels and their spatial distribution can be embodied in a matching computation. This

[6]It should be noted, however, that such biases pose some problems, and so must usually be gotten rid of.

pre-supposes that the image has been labelled, and that sub-images containing objects of potential interest have been extracted prior to looking for matches. In the present work, the labelling is that described in the previous two chapters. The normalized correlation just presented is valid for real-valued images where pixel values can be multiplied. In this case, the pixelwise matches are measured by a multiplication, and these are summed to obtain a global match measure. Here images of labels are considered, and multiplying labels makes no sense. To circumvent this problem a possibility could be to assign a different gray level to each label. The problem with this approach is twofold. Firstly, the labels are not treated on equal grounds: Labels assigned higher values contribute more to the correlation. Secondly, different labels will give a contribution to the correlation when their corresponding values are multiplied, *i.e.*discrimination between different labels is not complete. Different labels should have orthogonal representations, and thus not contribute to the match value when compared. This suggests the use of orthonormal vectors in some $n$-dimensional space for representing the different labels ($n$ being the number of labels considered), and resort to the standard scalar product for computing the pixelwise match. The usual basis vectors of $n$-dimensional Euclidean space would be adequate for this. In this manner no label is favored, and two pixels, one in the image and the other in the template, will give a non-zero contribution to the correlation only if their values correspond to the same label. Having as much discrimination as possible between different labels is important for discriminating similar objects, but of different colors.

Using the vector representation of labels, the match measure is then given by the correlation of normed vector functions,

$$\hat{\mathbf{S}} \odot \hat{\mathbf{T}} = \frac{1}{dim_x \times dim_y} \sum_{(x,y)} \langle \hat{\mathbf{S}}(x,y), \hat{\mathbf{T}}(x,y) \rangle, \tag{4.3.6}$$

$\hat{\mathbf{S}}$ and $\hat{\mathbf{T}}$ being the normed vector functions representing the labels in the image and the template, and $\langle , \rangle$ being the Euclidean scalar product. In this equation, the dimensions $dim_x$ and $dim_y$ of the template (or of the sub-image, both being the same) appear as normalization factors. This is equivalent to normalizing by the energy of the template times that of the image, since normed vectors are considered. Consequently, a perfect match will give a value of 1. Such a perfect score is of course highly improbable in practice,

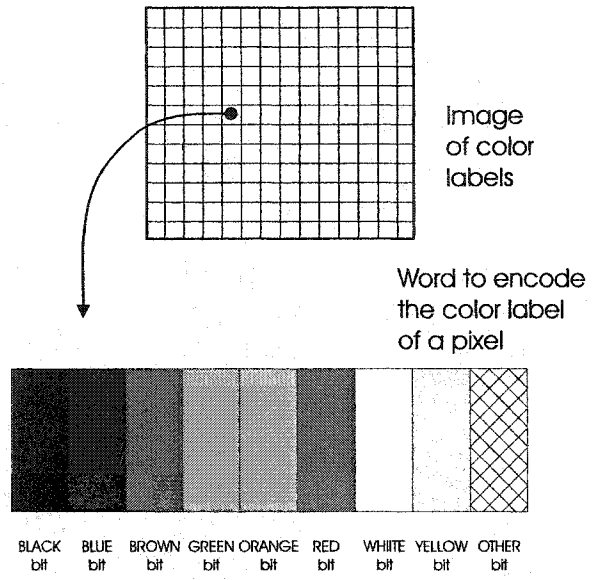



FIGURE 4.1. Encoding the color label of a pixel using a bit-word of sufficient length. Each bit corresponding to a different label. There is also a bit "OTHER" for pixels which do not have values of interest.

but this is a convenient way to normalize the correlation, since it can then be compared against the reference value of 1.

The previous vectorial representation is bulky, because high dimensional vectors are needed. A more compact alternative is to resort to bit words of sufficient length in which each bit represents a label, see Fig. 4.1. The word associated with a given label is such that its corresponding bit is set to one and the other bits to zero. This is a way to bit encode labels which allows computations to be performed based on labels as will be shown shortly. Thus, instead of images of unit vectors, images of words will be used for computing matches. These images will be termed *bit encoded images* (BEI's). Now, the manner in which two BEI's are correlated must be specified, and this requires a few definitions. However, before proceeding with these, the encoding of images and templates will be illustrated on a specific example to clarify the ideas presented thus far.

## 4.4 Encoding Images and Templates: An Example

Let the encoding of an image be first considered. This will be illustrated for the color image shown in Fig. 4.2 (a) in which road signs must be found. Once the pixels have been labelled using the color classification approach developed in Chapters 2 and 3, the labelled

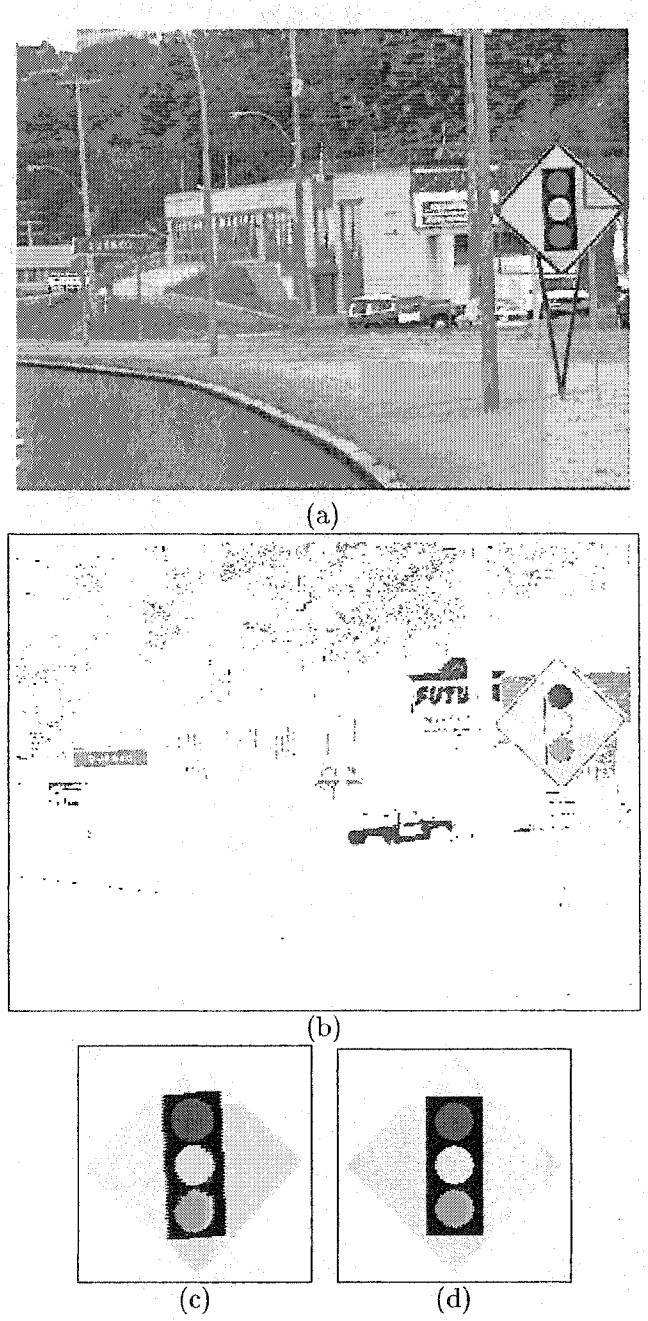



FIGURE 4.2. Illustrating the encoding of an image. (a) Original image along with the bounding boxes of found objects. (b) Labelled image using colors representative of each label (cartoon colors). The "OTHER" label is shown in white. (c) Resized labelled sub-image of the yellow object extracted. (d) Image used to generate the template. More details on how (c) is obtained starting from (b) will be given in Chapter 5.

image shown in Fig. 4.2 (b) is obtained[7]. Neighboring pixels having the same label are grouped to form regions as was described in Section 2.6.2, and the bounding box of each of these regions is computed (these bounding boxes are shown in Fig. 4.2 (a)). Sub-images of the labelled image corresponding to bounding boxes having sufficiently large dimensions are further processed to finally be extracted and resized to standard dimensions[8]. These constitute objects of potential interest. This is illustrated in Fig. 4.2 (c). The last step is to encode each resized labelled sub-image using the words representing the different labels as depicted in Fig. 4.1.

As concerns the generation of a template corresponding to a given road sign, this is done from a model drawn using standard drawing software[9] as follows. First, the model is rasterized to the same standard dimensions as above. Since the colors in such models are standardized, it is easy to label the raster image and then obtain the encoded template of the object. The raster model used to generate the template corresponding to the yellow road sign appearing in Fig. 4.2 (b) is shown in (d) (as such the template itself cannot be displayed as an image).

## 4.5 Operations on Words and Bit Encoded Images

Simple operations that can be done on words and on BEI's will now be defined. This will be useful in defining the correlation of BEI's, and for generating templates that will provide for invariance to sets of transformations.

The $AND$ ($\bigwedge$) and $OR$ ($\bigvee$) of two words $W_1$ and $W_2$ is the word $W_3$ obtained by applying the logical $AND$ and $OR$ operator to each corresponding bits of $W_1$ and $W_2$. This is respectively denoted by $W_3 = W_1 \bigwedge W_2$, and $W_3 = W_1 \bigvee W_2$. A remark about the words resulting from an $OR$: These may have bits set to one in more than one position. Consequently, a word may represent several labels at the same time. This property of the $OR$ operation will be exploited when tolerance to geometrical transformations is discussed

[7]Further details about color labelling specific to the context of road sign detection will be presented in Chapter 5.

[8]The minimal width and height for a bounding box have been chosen to be 25 and 25 in 640 × 480 images. Bounding boxes having smaller width or height do not contain enough detailed information to be used. This is why some color blobs in Fig. 4.2 (b) have no corresponding bounding boxes highlighted in (a). For resizing, the standard dimensions have been selected to be 64 × 64.

[9]The models for all the road signs appearing on the Québec road network were kindly supplied by the Ministère des Transports du Québec. They were drawn by a specialist using CorelDraw$^{TM}$.

in next section. Another operation that will be useful is the ***IF*** operator of a word $W$ which simply tests whether a word is nil (word with all zeros) or not.

$$\boldsymbol{IF}(W) = \begin{cases} 0 & \text{if } W \text{ is the nil word} \\ 1 & \text{otherwise.} \end{cases}$$

The ***IF*** operator is the one of the C Programming Language [**127**] p.56.

Similarly to operations on words, the $AND$ ($\bigwedge$) and $OR$ ($\bigvee$) of two BEI's $I_1$ and $I_2$ is the BEI $I_3$ obtained by applying the $AND$ and $OR$ word operator pixelwise to each corresponding pixels of $I_1$ and $I_2$. This will be respectively denoted by $I_3 = I_1 \bigwedge I_2$, and $I_3 = I_1 \bigvee I_2$. In all operations on BEI's, it is to be understood that the BEI's involved are of the same dimensions.

With these preliminary definitions, the normalized correlation of bit encoded images can be defined.

DEFINITION 4.1. *The normalized correlation of $I_1$ and $I_2$, to be denoted by $I_1 \triangle_n I_2$, is defined by*

$$I_1 \triangle_n I_2 = \frac{1}{dim_x \times dim_y} \sum_{(x,y)} \boldsymbol{IF}\left(I_1(x,y) \bigwedge I_2(x,y)\right). \tag{4.5.1}$$

The value of the correlation readily gives the percentage of pixels that match between the image and the template. For a perfect match, this value should be 1. Analogously to the usual correlation, a Cauchy-Schwartz inequality can be stated:

LEMMA 4.2 (Cauchy-Schwartz Inequality for Correlation of BEI's).

$$I_1 \triangle_n I_2 \leq 1 \tag{4.5.2}$$

*with the equality being satisfied if and only if $I_1$ and $I_2$ can be written as $I_1 = J \bigvee A_1$ and $I_2 = J \bigvee A_2$ where $A_1$ and $A_2$ are arbitrary*[10].

PROOF. Obviously, for the correlation to be equal to 1 it must be true that

$$\boldsymbol{IF}(I_1(x,y) \bigwedge I_2(x,y)) = 1 \text{ for all}(x,y),$$

[10]Recall that for the usual Cauchy-Schwartz inequality, the equality is satisfied if and only if the two images are a multiple of one another.

otherwise the correlation is smaller than 1. So in the case of equality, the question amounts to determining when the $AND$ of two words $W_1$ and $W_2$ is not equal to the nil word. For this to occur, $W_1$ and $W_2$ must have at least one bit set to one at a common place. Let $V$ be the word containing bits set to one where both $W_1$ and $W_2$ have bits set to one, *i.e.*$V = W_1 \bigwedge W_2$. Then $W_1$ and $W_2$ can be written as $W_1 = V \bigvee U_1$ and $W_2 = V \bigvee U_2$, where $U_1$ is the word having ones where $W_1$ has, but $W_2$ does not ($U_1$ can be the nil word), and conversely for $U_2$. This translates directly to the case of images of words, where $A_1$, $A_2$, and $J$ respectively play the role of $U_1$, $U_2$, and $V$, and ends the proof. □

The correlation just defined can be seen as a generalization of the correlation of binary images often used in morphological image analysis [**128**] in which case the words are composed of a single bit. However, the multi-bit encoding of Fig. 4.1 is much richer than simple binary values, and will now be exploited.

## 4.6 Geometrical Transformations and Classes of Objects

So far the encoding was introduced as a means of representing labels which allows them to be taken into account in a completely discriminated manner in the correlation introduced in Eq. 4.5.1. As mentionned in the introduction to this chapter, a problem arising with the use of correlation is that of having invariance with respect to geometrical transformations since these alter the correlation score even for small values of the transformation parameters in some cases. Consider the object in Fig. 4.3 (a) for rotations about different axes. In the first case, the object is rotated in-plane for roll angles varying from $-180°$ to $+180°$ (see Fig. 4.3 (b) for an example at 13° clockwise). The result of the correlation between the unrotated object, and the different rotated versions of it is given in (d). The center peak is seen to decrease very rapidly. Indeed, the correlation drops by 20% (correlation value = 0.8) at angles of only $\pm 13°$. The other peaks in the graph correspond to the approximate symmetries of the object when rotated by multiples of 90°. The effect of out-of-plane rotations is less severe as Fig. 4.3 (e) shows. Here, the yaw angle was varied from $-90°$ to $+90°$, and the correlation reaches a value of 0.8 for angles of $\pm 39°$ (see Fig. 4.3 (c) for the object rotated by $-39°$). In (e), the variation is shown for yaw, but the behavior is similar when the pitch angle is varied. Although a particular object is considered in Fig. 4.3, it

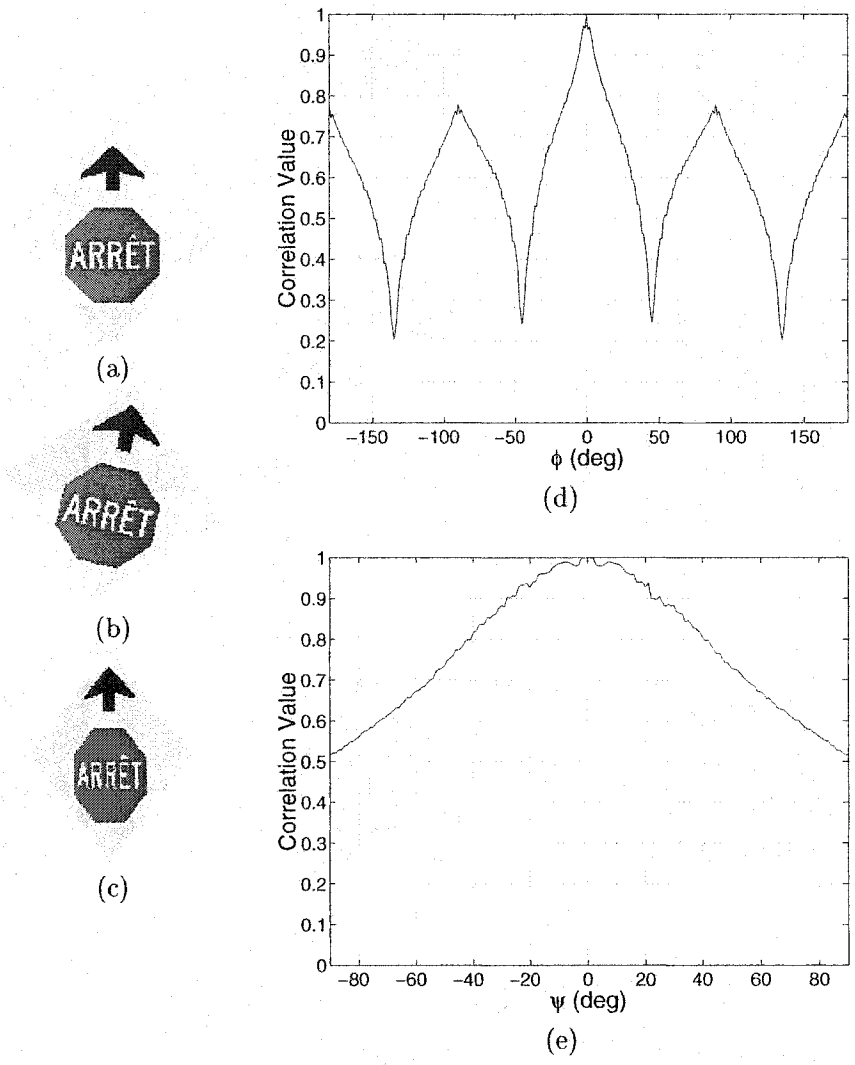



FIGURE 4.3. Variations of the correlation in the case of rotations. (a) Un-rotated object. (b) Object rotated clockwise in-plane (roll angle $\phi = 13°$). (c) Object rotated out-of-plane (yaw angle $\psi = -39°$. The dependence of the correlation is shown in (d) for in-plane rotations from $-180°$ to $+180°$, and in (e) for out-of-plane rotations from $-90°$ to $+90°$.

gives some intuition as to how the correlation may vary in other cases, the difference from one object to another being in large part determined by their symmetries.

The definition for correlation to be invariant under a given set of transformations is as follows:

DEFINITION 4.2. *A template is said to be* ***correlation-invariant*** *to an object and a given set of transformations if the value of the correlation between the template and the object transformed by any element of the set remains constant to* 1[11].

This says that by using a *single* template, the value of the correlation should remain the same when the object is transformed by any element in the set of transformations. This is an idealization. In practice the value is never exactly constant, and so it is perhaps more realistic to speak in terms of quasi-invariance, or tolerance, and use a range for the values of the correlation. For this, the notion of $\epsilon$-invariance is introduced, which is a slight modification of the previous definition.

DEFINITION 4.3. *Given an error* $\epsilon$*, a template is said to be* $\epsilon$***-invariant*** *to an object and a given set of transformations if the value of the correlation between the template and the object transformed by any element of the set remains in the interval* $[1-\epsilon, 1]$.

Here $\epsilon$ can be seen to set a bound on the range of correlation values for which a match is declared to be found.

Using the encoding introduced in Fig. 4.1, invariance to a set of transformations can be embedded in a template by combining the individual templates of transformed versions of an object with the OR operator (in short by "OR"ing the individual templates). In this manner the resulting template is representative of these transformed versions because it correlates strongly (in fact ideally retains its value of one) with any of them. This is how invariance is achieved in the present work. Such a template will be called a *composite encoded template* (CET). Mathematically, let $T_i$, $i = 1, \ldots, N$, be the templates corresponding to transformed versions of an object, where $i$ is an index over the considered transformations. The CET of the object embedding the given transformations is given by

$$CET = \bigvee_{i=1}^{N} T_i. \tag{4.6.1}$$

Such a template is reminescent of the so-called composite filters used in optical pattern recognition [**126**], except that here the combination is non-linear, whereas in composite filters, templates are combined linearly. The present method for embedding invariance

[11] Note that as long as the correlation value is constant, *i.e.*not necessarily having the value of 1, it could be said that invariance is achieved. However, since correlation is used for the purpose of finding matches, 1 is the reference.

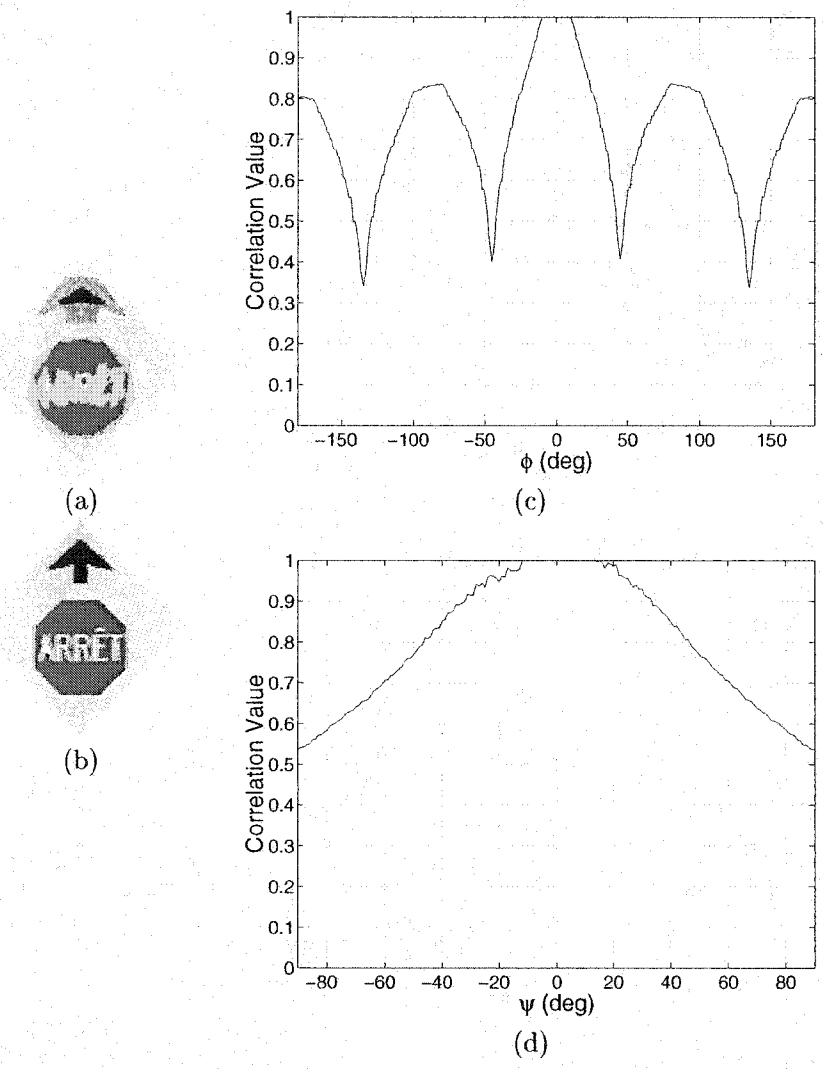



FIGURE 4.4. (a) CET for obtaining invariance to roll angles between $-10°$ and $+10°$. (b) Same as (a) but for yaw angles between $-10°$ and $+10°$. Variation of the correlation for (c) in-plane rotations from $-180°$ to $+180°$, and (d) out-of-plane rotations from $-90°$ to $+90°$. (c) and (d) should be compared with Fig. 4.3 (d) and (e).

finds its source in the logical operations that can be performed on the encoding, the fundamental point being the possibility of having a word represent several labels concurrently. Fig. 4.4 gives examples of the use of CET's for achieving invariance with respect to rotations. Fig. 4.4 (a) shows a CET obtained by the combination of templates of the object rotated (in-plane) from $-10°$ to $+10°$ in steps of $1°$. Pixels for which the word resulting

from the combination represents more than one label are colored using the mean of the individual labels' associated colors. This explains why the CET appears as "washed out" when represented in color. The graph in (c) was obtained by correlating this CET with encoded images of computed rotated versions of the object as was done for Fig. 4.3 (d). The center peak has now been flattened to a value of one between $-10°$ to $+10°$, thereby illustrating invariance in this range of angles. This range can be fully controlled by combining appropriate templates (here angles from $-10°$ to $+10°$ have been considered, but any other range could have been chosen, and the corresponding templates combined). Fig. 4.4 (b) shows a CET composed of templates with yaw angle varied from $-10°$ to $+10°$ in steps of $1°$; (d) is the associated graph.

In the present approach, the problem of invariance is reduced to combining templates. Which transformations should be considered for building composite templates depends on the application, and the degree of tolerance required. Examples will be given in next section.

Thus far, CET's have been presented as a means for embodying geometrical transformations of a given object in a single template. However, a CET can also be used to represent a class of objects by having it be a combination of templates of different objects. For example, in the case of road signs, many are very similar, and a CET can be used to represent them all at once for the purpose of performing an initial classification. Formally, given a template, an $\epsilon$-class of objects associated to it can be defined as follows.

DEFINITION 4.4. *Let $T$ be a template and $I$ the encoded image of an object $\mathcal{O}$. Then $\mathcal{O}$ is said to be in the $\epsilon$-class of $T$ if*

$$I \triangle_n T \in [1-\epsilon, 1]$$

In the case where a CET is used to represent a class of objects, it will be called a class CET (CCET). It can be seen at this point that CET's and CCET's naturally lend themselves for gradual recognition from classes of objects down to individual objects. Such a possibility in the form of a tree structure will be exploited in next chapter.

Before closing this section, a point worth mentioning is that combining a great many templates of widely differing objects would in the limit lead to a CET that would match everything. However, for road signs the geometrical transformations to consider do not differ too significantly from the identity, and for CCET's, the combined templates correspond

to similar road signs (see Section 5.3.1). Indeed, in any application, CCET's should be composed with similar objects.

## 4.7 Generalizations

There are a further directions in which the previous discussion can be generalized. It is not the purpose here to investigate these in full detail, which would lead too far away, but rather to give outlines for other potential applications.

The most important direction for generalization concerns the labelling phase. Thus far the discussion has been phrased in terms of color labelling, but this need not necessarily be the case. As such, the template matching approach that was developed only requires images and templates to be encoded, and it is conceivable to do this based on other types of labellings, even though here, the initial motivation came from encoding colors orthonormally for matching purposes. For instance, the problem could involve planar objects that present different textures, which can be identified (detected), and to which labels could be associated.

Still concerning the encoding, it could represent multiple sources of information, not just one. Indeed, suppose that $N$ attributes, which distribute spatially, must be taken into consideration, and that it is required to perform spatial pattern matching based on these attributes. Suppose further that each attribute comprises $n_i$ possibilities (hence $n_i$ labels), $i = 1...N$. Then the information can be encoded in words containing $n_1 + n_2 + ... + n_N$ bits, with the $n_1$ first bits representing the possibilities for the first attribute, the next $n_2$ bits representing those of the second attribute, *etc...* The advantage of this is that several sources of information can be used at once in a single correlation, which could be useful in sensor fusion. This would not be so easy with techniques as those used in optical pattern recognition. As before, templates embodying geometrical transformations or representing classes of objects can be obtained by using the OR operation.

A last point concerns the dimensionality of the images that are encoded. Here 2-D images are dealt with, however the encoding can be used in higher dimensional images, for which the correlation introduced in Eq. 4.5.1 can also be trivially generalized. For example template matching in 3-D images could be considered, and in this case the voxel values would be encoded using words such as those introduced in this chapter.

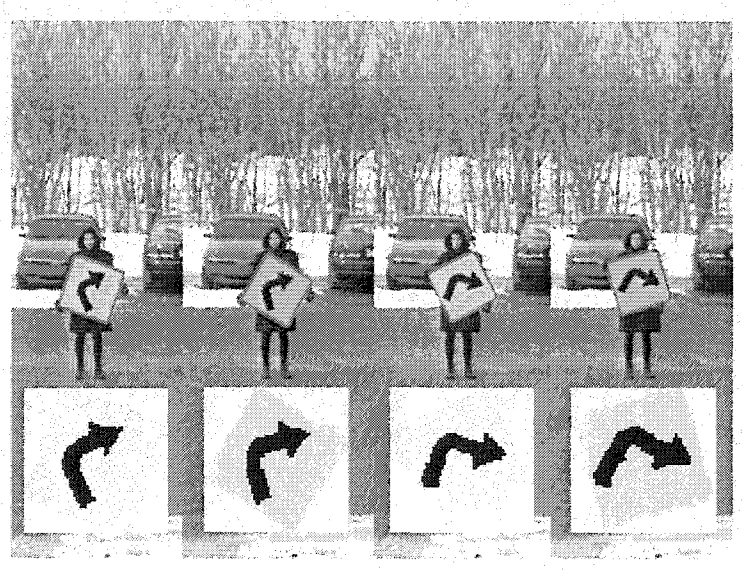
## 4.8 Experiments

As a first example, consider the road sign extracted from Fig. 4.2 (a). In this case there is no real need to resort to a composite template since the road sign was not distorted much by geometrical transformations. Correlating the encoded sub-image and the template corresponding to Fig. 4.2 (c) and (d) respectively gives a value of 0.962, which is sufficiently high for concluding on the presence of the road-sign.

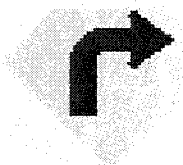
The next example in Fig. 4.5 illustrates the improvement brought by using a CET for a road sign imaged outdoors, and for which the roll was purposedly varied between $-45°$ and $+45°$ (only a few examples are actually shown in (a)). The blue curve in (d) depicts the values of the correlation when the template representing only the object in the upright position (shown in (b)) is correlated with the different rotated versions of the road sign extracted from the images. The red curve illustrates the same situation, but with a CET, shown in (c), composed of rotated templates for angles between $-10°$ to $+10°$. In the latter case, the curve remains above 0.9 for rolls in the range from $-19°$ to $+15°$. This means that by setting $\epsilon$ to 0.1, it is possible to obtain epsilon-invariance in that range when this CET is used. In the case of the blue curve, the value of the correlation remains above 0.9 only in the range between $-7°$ to $+5°$. For comparison, theoretical results are also shown on the graph: The green curve depicts the correlation values when the template of (b) is correlated with computed rotated versions of the object for rolls from $-45°$ to $+45°$, whereas the black curve depicts the same, but with the template of (c) (these two latter curves are analoguous to the curves shown in Figs. 4.3 (d) and 4.4 (c)).

Transformations in normal conditions are of course not as severe as those presented in the previous example. Very rarely is a road sign rotated by as much as $20°$. Actually in typical road sign imaging situations in-plane rotations of at most $\pm 10°$ need to be considered. Nevertheless invariance is important even for small angles because it helps in maintaining a high correlation when other sources of error come into play, such as noise and missing detected pixels.

As a preview to the application of the template matching approach developed in the present chapter to the road sign recognition system to be described in next chapter, Fig. 4.6 shows recognition results for a real image of a road scene. The details of how the template matching is incorporated in the system as well as more extensive results will be given in

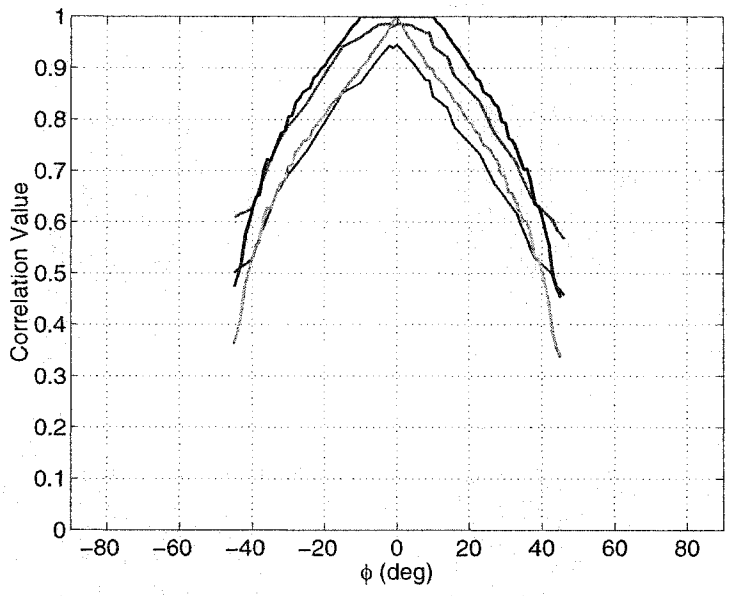


(a)



(b)

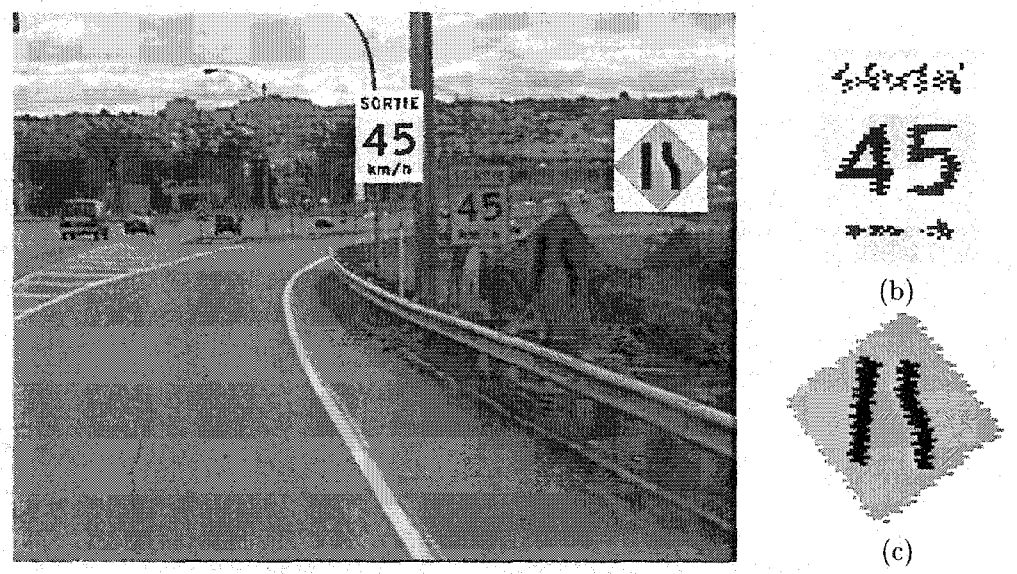
(c)




(d)

FIGURE 4.5. Illustrating the use of CET's for obtaining increased invariance. This is depicted for a road sign imaged outdoors and positioned at different rolls.

Chapter 5. At this point, suffice it to say that the system exploits both CET's to embody geometrical transformations as well as CCET's to define classes of road signs.




(a) (b) (c)

FIGURE 4.6. Example of a road sign classification result based on the template matching developed in this chapter. In (a), the original image is depicted along with the road signs that have been recognized, whereas (b) and (c) show the labelled sub-images corresponding to the detected objects. The jitter in (b) and (c) occurs because the image was captured at relatively high speed using an interlaced camera. In the future, a progressive scan camera will be used.

## 4.9 Conclusion

A template matching technique for labelled images was developed. It is based on an orthonormal encoding of the labels, on which operations can be performed. From this, a correlation computation between encoded images can be defined, which serves as a match measure. The interesting aspect of the encoding is to allow templates to be combined, and this was exploited to obtain invariance (or tolerance) of the correlation to geometrical transformations. The degree of tolerance is controlled by the templates of transformed versions of the object that are combined. This was illustrated for the problem of identifying road signs in color images. In this case, it is the particular colors of the road signs that serve to label images (more results to come in next chapter), but, as was discussed, other types of attributes could be encoded (such as textures). Invariance was studied in the case of rotations, and the approach was seen to provide for tolerance. The possibility to combine templates also allows classes of objects to be defined in the case where groups of very similar objects are considered. Both these aspects will be exploited in the sequel.

# CHAPTER 5

# Road Sign Recognition

## 5.1 Overview

The purpose of the present chapter is to describe the overall architecture of a road sign recognition (RSR) system for the processing of a single image, along with the techniques used in the different processing steps, and to extract quantitative results on the system's recognition abilities in real situations[1].

The sections of the chapter are organized along the processing flow shown in Fig. 5.1, with the system composed of three primary subsystems: detection, recognition, and scene understanding, respectively described in Sections 5.2, 5.3, and 5.5. An input color image is first processed by the detection subsystem responsible for identifying *regions of interest* (ROI's) on the basis of color space labelling and connectivity (Section 5.2.1). The novel contribution here is the use of the physics-based approach to the color labelling problem under daylight described in Chapters 2 and 3. Extraction of ROI's is accomplished by clustering pixels with similar color labels (Section 5.2.2). However, in some cases this fails to capture the full extent of a road sign (RS). An additional ROI grouping stage consisting of an original hypothesis-verification procedure is then carried in the context of the recognition process (since this requires recognition, this is described later in the chapter in Section 5.4.1). Once ROI's are determined, the sub-image associated with each ROI is re-labelled more accurately (Section 5.2.3) than what can be performed by the initial labelling of the whole image.

[1] A condensed version of the work discussed in this chapter has been accepted for presentation at CVPR 2001 [**129**].

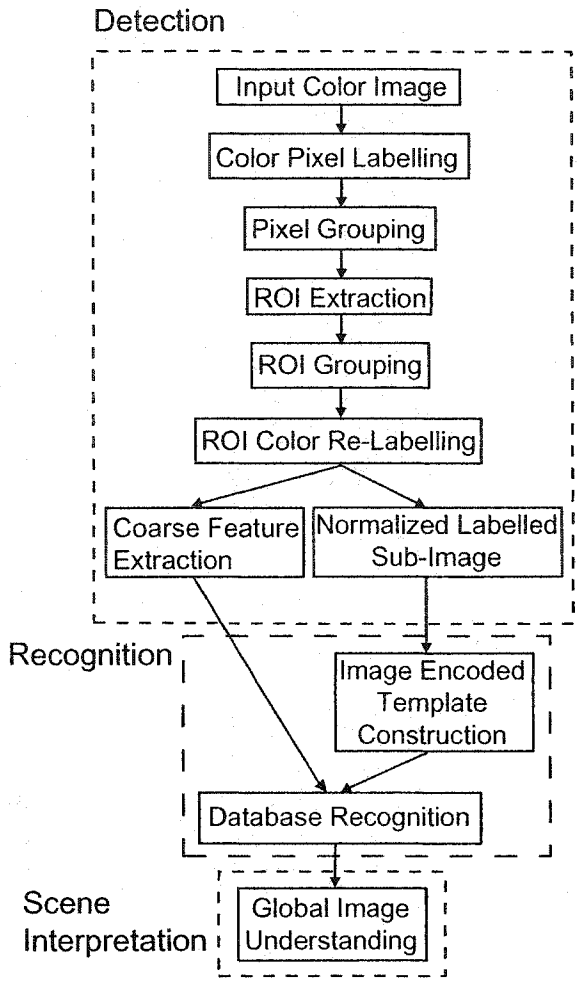



FIGURE 5.1. RSR System Block Diagram.

As depicted in Fig. 5.1, each ROI serves to extract coarse features from the image (aspect ratios and percentage of road sign colors in the ROI), and to compute a normalized labelled sub-image. These constitute the output of the detection subsystem and are subsequently used by the recognition subsystem which relies on a database to perform its task. The coarse features serve to prune the search space, restricting template matching (TM) to viable candidates in the RS database using the image encoded template obtained from the normalized labelled sub-image. As mentioned in the introductory chapter, one of the salient features of the present system resides in that the recognition database is generated automatically through modelling using few geometrical parameters. The details regarding

the recognition subsystem and the underlying database are described in Section 5.3. Some further implementation details concerned with detection and recognition are discussed in Section 5.4.

The third sub-system, described in Section 5.5, is concerned with the understanding of road sign scenes. It is devoted to the analysis of the recognition results based on constraints on the position of road signs along with their spatial relationships to other road signs. The goal is to further sieve out false positives.

Experimental results are presented in Section 5.6, and Section 5.7 concludes with a summary and directions for further work.

**5.1.1 Context of the Work.** The conditions under which RSR has been studied are as follows: 1) RS's are detected using color information. RS's of all colors are considered, except those containing only black and white, which are not considered in this work. The structure is there to recognize such road signs, but it is at the detection level that further work must be done. The problem with black and white is that natural colors tend not to be very saturated, and so are also detected and get mixed with black and white when detection is performed. 2) Arbitrary daylight illumination conditions are considered; night detection is however not treated. 3) Rainy, snowy, and foggy weather conditions will not be considered. 4) Because of the finite resolution (pixel size) of cameras, the size of a detected object in an image (potentially a road sign) should be large enough for it to be considered. 5) As concerns full recognition, pictographic RS's are considered (this already requires over 400 RS's to be recognized in North American regulations). RS's requiring optical character recognition (OCR) for a complete interpretation, such as direction, motorist services, recreational, and cultural interest guidance and advertisement RS's, are not yet fully processed, although the system architecture can eventually accommodate an OCR back end. 6) The present work concentrates on the processing of individual images.

## 5.2 Road Sign Color Detection

In the context of RSR, color is not as easy to exploit as it first appears because outdoor illumination conditions vary widely depending on time of day and weather. Although RS materials are made of standardized materials, the color measured by a camera can change significantly for a given RS due to these variations. Recall that the work presented in

Chapters 2 and 3 was precisely carried to overcome such problems. This is exploited in the RSR system described here.

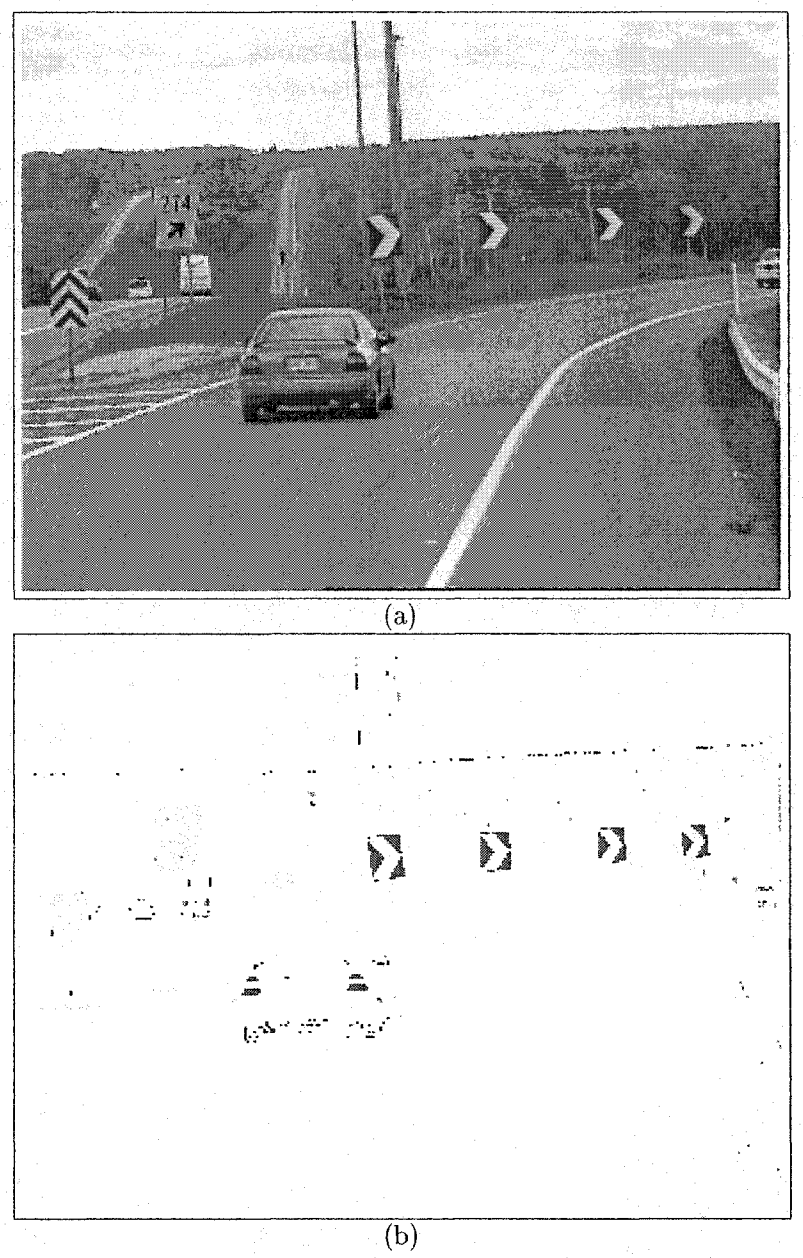
**5.2.1 Pixel Labelling and Grouping.** In the present RSR system, the color space regions obtained in Chapter 3 after calibration (see Fig. 3.8) are used to perform pixel classification (or labelling). With these color space regions, the pixel labelling and pixel grouping steps of Fig. 5.1 are performed as was described in Section 2.6.2 [2]. Recall that a look-up table is used to perform pixel classification, which is very fast. The result of the pixel classification step is an image of labels (to be called the "labelled image" in the sequel) having the same width and height as the input color image. Fig. 5.2 shows an example. Note that pixels that do not have RS colors are labelled *OTHER*. As regards the grouping step, the result is a set of pixel regions in the image which are potential RS's, and to each of which is associated an RS color label. Fig. 5.3 (a) depicts in random colors the pixels regions for the image of Fig. 5.2 obtained from the pixel grouping.

**5.2.2 ROI Extraction.** The bounding box (BB) of each pixel region in the image found by the grouping stage is computed. Such a region and its BB constitute an ROI. The color associated with the pixel region is called the ROI's *principal color*[3]. Fig. 5.3 depicts the final result of the ROI extraction step (Fig. 5.1 should perhaps be recalled again at this point). Note that a first step of size filtering has been performed at this stage. In fact ROI's having BB's smaller than $5 \times 5$ pixels in size are discarded (this is why some small color regions appearing in Fig. 5.2 (b) have no counterpart in either Fig. 5.3 (a) or (b)). It should be mentioned, however, that not too much filtering can be performed at this point, otherwise this prevents fragmented RS's to be recovered by the ROI grouping step to be discussed later (Section 5.4.1 below). Another step of size filtering will be carried after ROI grouping (Section 5.4.4).

**5.2.3 ROI Color Re-labelling.** Once an ROI is determined, the sub-image contained in its BB is re-labelled. This is primarily to extract the black and white pixels it

[2]As mentioned in Section 2.6.2, 8-connectivity was found to provide for superior performance than 4-connectivity in the pixel grouping step as regions tend to be less fragmented with 8-connectivity, although fragmentation may still occur, see Section 5.4.1 below.

[3]This terminology is used because RS's are made of color combinations. As explained later in Section 5.3.1, it is the principal color that serves in the initial classification of an RS for its recognition.

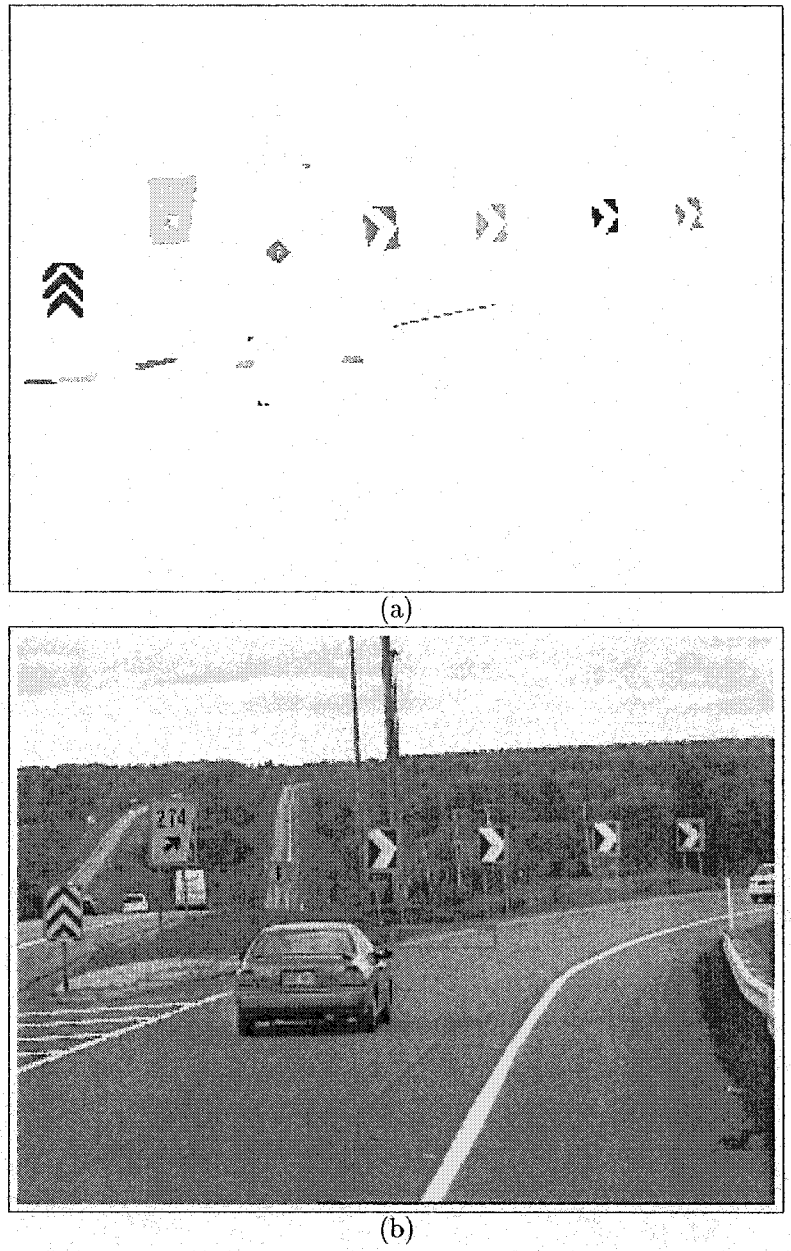


(a)

(b)

FIGURE 5.2. Illustrating pixel classification. (a) Original input color image. (b) Labelling of input image (labelled image). The labels are illustrated with their corresponding colors; pixels labelled as $OTHER$ are displayed in white. Neighboring pixels with the same label are grouped in this image, see Fig. 5.3 (a).

contains. This is important since the pictographic information carried by most RS's is displayed in black and/or white. Fig. 5.4 (a) shows a labelled sub-image associated with an



(a)

(b)

FIGURE 5.3. (a) Pixel regions displayed in random colors for the labelled image shown in Fig. 5.2. These regions serve to determine the BB's of the extracted ROI's depicted in pink in (b). Note that the ROI grouping stage to be described in Section 5.4.1 has been carried here in determining some of these BB's, and also that some small groups of pixels that appeared in Fig. 5.2 (b) do not appear here as a result of a first step of size filtering (see text).

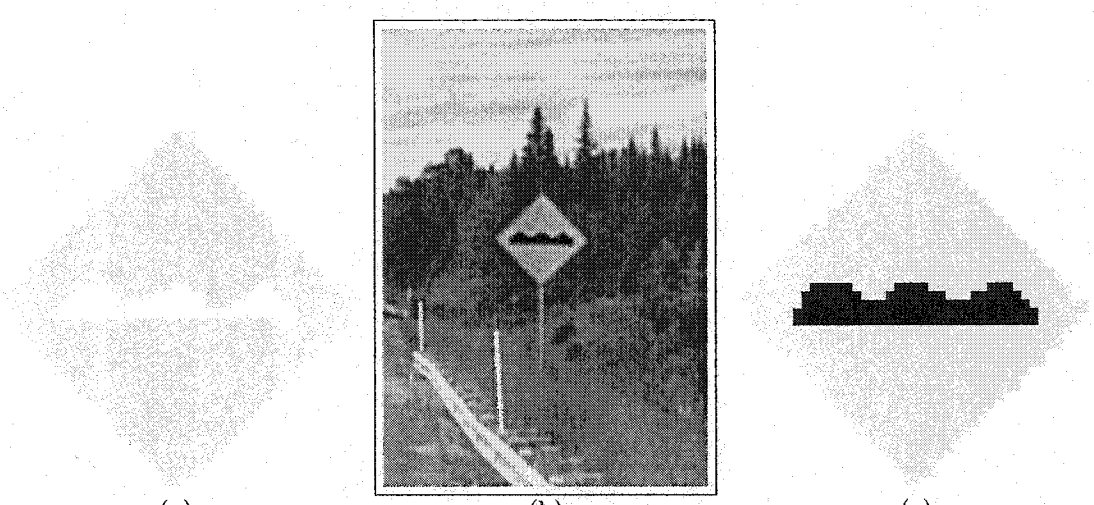



FIGURE 5.4. (a) Labelled sub-image of the yellow RS extracted from the image shown in (b) (original image) prior to re-labelling. (c) Labelled sub-image after re-labelling. The black pixels are recovered (in this case, no white pixels are to be recovered).

ROI prior to re-labelling. At this stage black and white pixels are not yet extracted, and color intensity information can be resorted to for this purpose as follows. First the mean $\mu$ and standard deviation $\sigma$ of the color intensity of pixels in the sub-image having the principal color are computed (in the example of Fig. 5.4 (a), the principal color is yellow and all yellow pixels contribute to $\mu$ and $\sigma$). With these statistics, the sub-image in the original image is scanned and each pixel is re-labelled following the algorithm shown in Fig. 5.5, where $Label_i(x, y)$ and $Label_f(x, y)$ are the labels of the pixel at position $(x, y)$ prior to and after re-labelling, $I(x, y)$ is the color intensity of pixel $(x, y)$ in the original image, and $PRINCIPAL\_COLOR\_LABEL$ is the label of the ROI's principal color, *e.g.*$YELLOW$ in Fig. 5.4 (a). For pixels having their initial label as that of the principal color, a deviation of $2\sigma$ from the mean $\mu$ is used to determine whether they should be re-labelled as black or white or retain the initial label. Such a deviation is sufficiently large (assuming Gaussian statistics) to insure that pixels having the principal color label are not wrongly re-labelled as black or white.

The algorithm shown in Fig. 5.5 is referred to as the generic re-labelling algorithm, because for some principal colors, the lines that re-label a pixel as $BLACK$ or $WHITE$ must be omitted since for those colors pictographic information cannot appear as $BLACK$

```
if Label_i(x,y) = PRINCIPAL_COLOR_LABEL
  if I(x,y) < μ − 2σ
    Label_f(x,y) = BLACK
  else if I(x,y) > μ + 2σ
    Label_f(x,y) = WHITE
  else
    Label_f(x,y) = Label_i(x,y)
else if Label_i(x,y) = OTHER
  if I(x,y) < μ
    Label_f(x,y) = BLACK
  else if I(x,y) > μ
    Label_f(x,y) = WHITE
  else
    Label_f(x,y) = Label_i(x,y)
else
  Label_f(x,y) = Label_i(x,y)
```

FIGURE 5.5. Generic re-labelling algorithm.

or $WHITE$. For example, on blue RS's, only white symbols or characters can appear, thus no pixel should be re-labelled as $BLACK$.

Notice that no re-labelling is done on pixels having initial labels different than that of the ROI's principal color or $OTHER$ except in cases where this does not lead to what will be called "color combination conflicts" as will be explained shortly. The reason for not re-labelling such pixels is that an RS may be of a color combination that contains more than just the principal color, black, and/or white. Fig. 5.19 shows an example of such an RS, which is yellow, and which also contains red and green. Re-labelling red and green pixels as either black or white would be a wrong thing to do in this case since this would be conflicting with the fact that the RS does indeed contain red and green. These other colors must be left unaltered by the re-labelling, otherwise they would be lost. Since the recognition procedure relies on color combinations in searching the class of an RS (refer to Section 5.3), these other colors must be retained.

As just alluded to, in some cases it is possible to re-label pixels that do not have the principal color of the ROI or are not labelled as $OTHER$ when this does not lead to conflicts. The purpose of re-labelling in such cases is to correct for pixels that were wrongly labelled by the initial labelling stage (most often, wrongly labelled pixels occur at the border of color changes in the image because of pixelization effects). This can be done

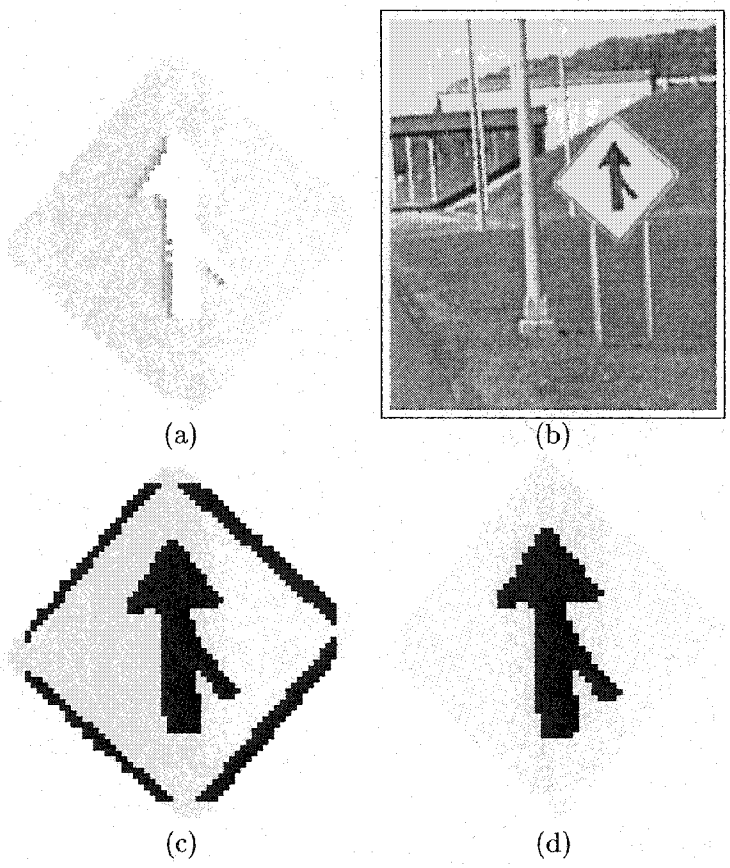



FIGURE 5.6. (a) Labelled sub-image of the yellow RS extracted from the image shown in (b) (original image). Some pixels on the border of the arrow were initially labelled as orange. (c) Re-labelling taking into account that orange pixels must be re-labelled since no orange should be present on yellow RSs, and no color combination conflict can occur if orange pixels are re-labelled. Notice also the yellow pixels on the contour of the RS that have been re-labelled as black. A technique for determining more accurately the contour of objects will be described later in Section 5.4.2 as it relies on recognition. As a preview, (d) shows the result that will be obtained.

when the combination of the principal color and the color associated with the label of such pixels is impossible, in which case it will be said that no color combination conflict occurs if re-labelling is performed. For example, it is not possible for an RS with principal color yellow to have pictographic information displayed in orange on it. Thus, in such a case it is possible to re-label orange pixels if some appear. Fig. 5.6 shows an example where correction is possible[4], whereas Fig. 5.7 shows a case where correction of red pixels is not

[4]In Fig. 5.6 (a) and similar cases, where orange pixels appear on a yellow RS, they must be re-labelled in the same manner as yellow pixels. Hence, the first line of the generic re-labelling algorithm must be replaced by: if $Label_i(x,y) = YELLOW\ OR\ Label_i(x,y) = ORANGE$.

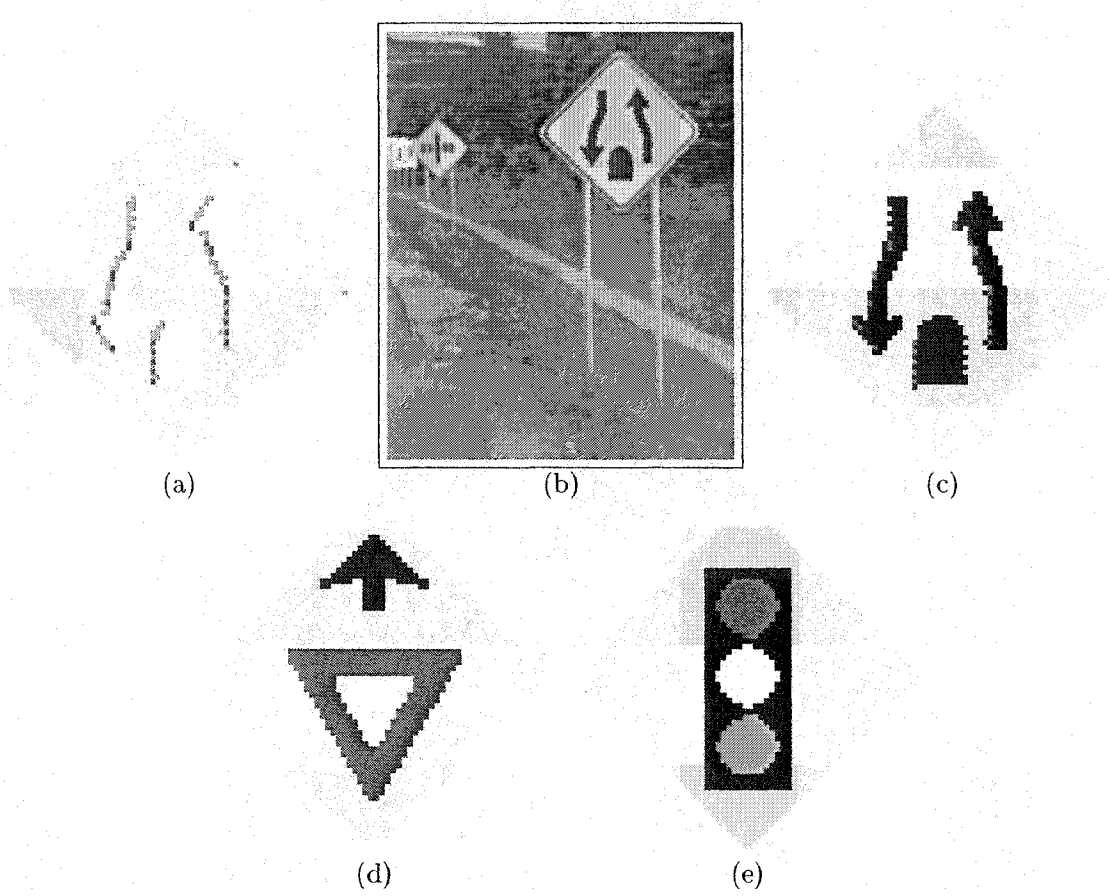



FIGURE 5.7. (a) Example where re-labelling would lead to a color combination conflict. Here some pixels are initially labelled as orange and red, which is wrong. (b) Original image. Pixels labelled as orange can be corrected for by re-labelling as previously described. However, it is not possible to re-label as yellow or black, pixels that were initially labelled as red (as they should be in the present case), unless more information about the RS is obtained. The reason is that some classes of yellow RS's may have information displayed in red on them, and in such cases re-labelling red pixels as yellow or black would be wrong ((d) and (e) show examples). There are also pixels labelled as white for which the situation is similar. (c) Re-labelling result with remaining red and white pixels (one must look carefully to see the white pixels).

possible because there is not yet enough prior information about the identity of the RS at this stage of processing. How to obtain more information for resolving such cases will be discussed in Section 5.4.2.

**5.2.4 Coarse Feature Extraction and Normalized Labelled Sub-Image.** For each ROI, the detection subsystem computes the aspect ratio of the ROI's BB (width divided by height), and extracts the percentages of each RS color present in the ROI's re-labelled

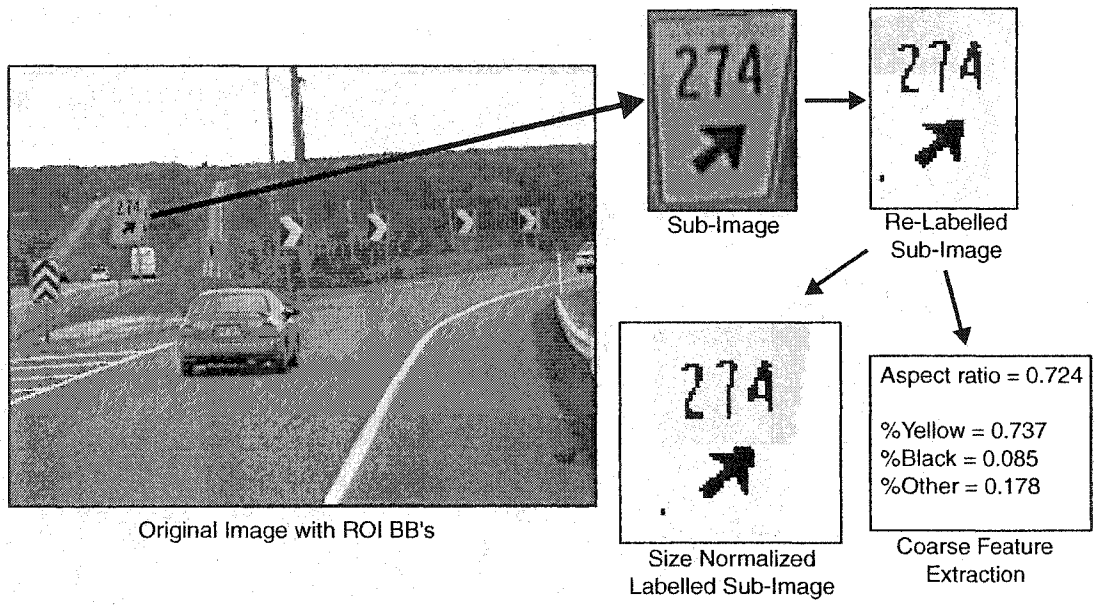



FIGURE 5.8. Processing steps after the ROI's are found. It is from the re-labelled sub-image that the color presences are computed. Here, the sub-image and re-labelled sub-image are $42 \times 58$ in size, whereas the normalized labelled sub-image is $64 \times 64$ with the object centered in it.

sub-image. The re-labelled sub-image is then resized to standard dimensions (in the current implementation, the largest dimension (width or height) is resized to 64 pixels in size and the other is resized accordingly so as to preserve the aspect ratio of the object). This size normalization avoids scale problems when recognition is considered. The aspect ratio, color presences, and size normalized labelled sub-image for each ROI constitute the output of the detection subsystem used as input by the recognition subsystem. To recapitulate, Fig. 5.8 shows pictorially the processing steps involved in detection for one of the RS's found in the image of Fig. 5.2 starting at the point where ROI BB's have been determined (*i.e.*following Fig. 5.3 (b)).

As a conclusion to this section, it should be pointed out that the technique used in the present work to determine the color space regions associated with RS colors is significantly different from that used by others. In many cases, these color regions are obtained from examples, such as in [**37**], where no modelling of color formation is made.

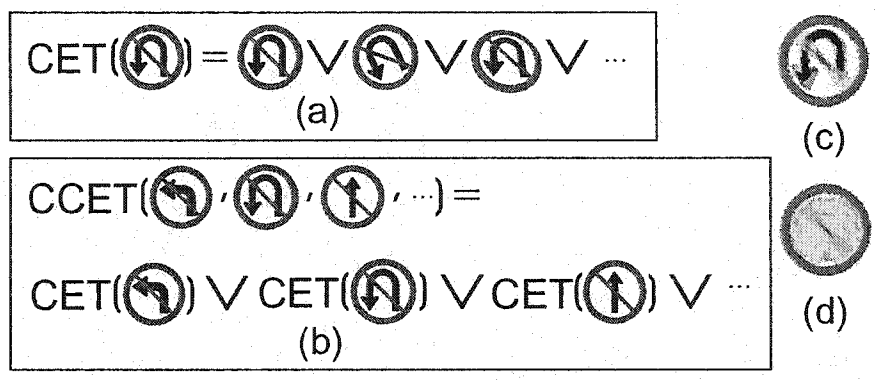



FIGURE 5.9. CET construction for: (a) embodying geometrical invariance, and (b) defining a RS class (a CCET). (c) shows the resulting CET for (a), and (d) for (b). In (b) and (d), pixels for which the word resulting from the combination represents multiple labels are colored using the mean of the individual labels' associated colors. This is why the CET appears "washed out".

## 5.3 Recognition

Once potential RS's are detected, they must be recognized. Since RS's have a definite structure, they can be arranged in a database to facilitate their recognition. For a potential RS extracted from an image, the recognition problem consists in finding a match in this database. Central to the RS recognition scheme discussed in this section is the template matching (TM) technique for planar objects developed in the previous chapter.

As a reminder, recall that the technique relies on an encoding of the different considered colors (here RS colors). With this encoding, a template is built for an object (either extracted from an image or obtained from a model, see Section 4.4), and a correlation computation can be defined, which serves as a measure for computing matches between templates. The encoding has the important property of embodying color information as well as allowing different templates to be combined. Composite templates can be exploited to obtain tolerance to geometrical deformations when computing the correlation, and to define templates that are representative of classes of objects. Advantage of both aspects will be taken in the present RSR system, and are illustrated pictorially in Fig. 5.9.

**5.3.1 Database Organization and Content.** Fig. 5.10 depicts the structure of the database used in the present system. It consists in three levels. At the first level, RS's are divided along their principal colors. Within these color classes, at the second level, RS's are further subdivided into RS classes according to their shapes and color content which

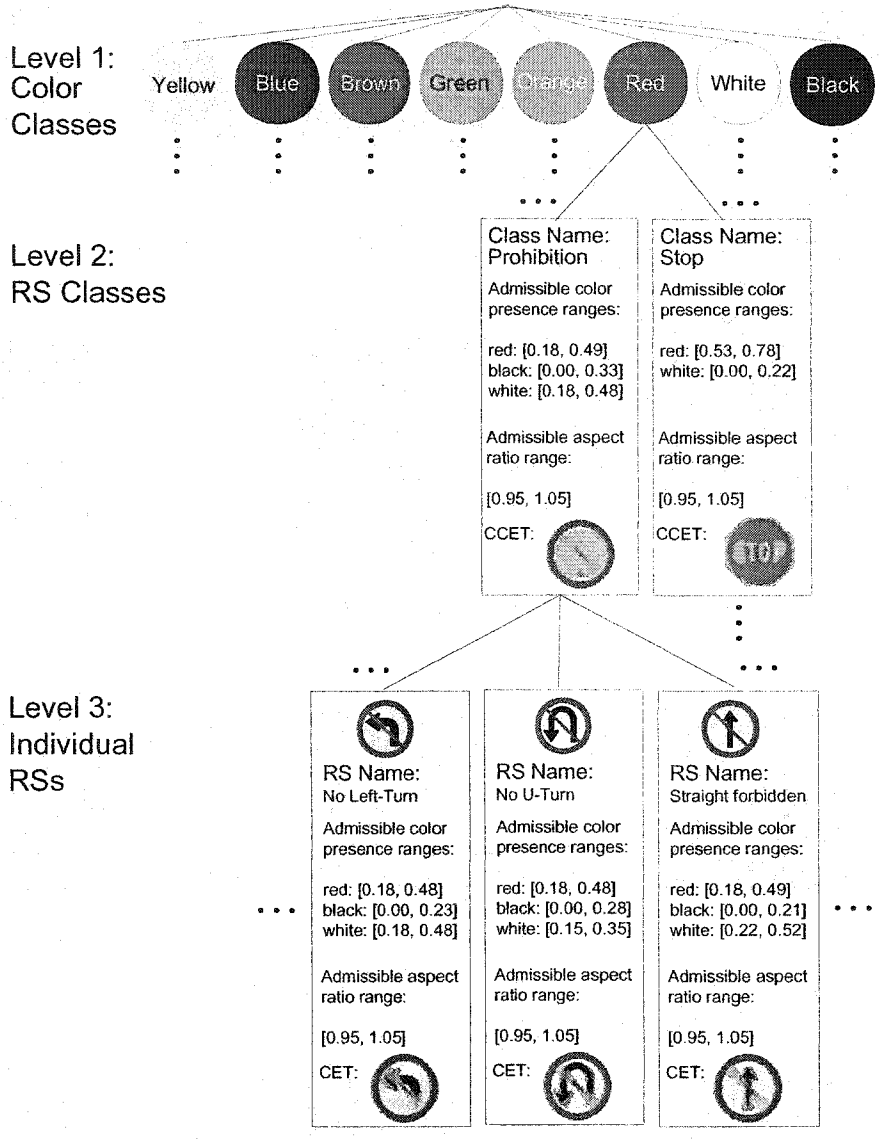



FIGURE 5.10. Database structure.

correspond to RS's having different meanings. RS's having the same shape and colors are grouped together, *e.g.*yellow diamond-shaped RS's with black pictographic symbols. The shape aspect of the RS classes is represented by a range of admissible aspect ratios, and a CCET which is a combination of all the CET's for the RS's belonging to the class (as in Fig. 5.9 (b)). Regarding the color content, the mean and standard deviations of the percentages of presence of the colors contained on the RS's belonging to the class are used

to define a broad range of admissible percentages for each of these colors (the CCET also contains color information through the encoding). These shape and color features appear in the enclosing rectangles of the classes at level 2 in Fig. 5.10. They serve to determine if an unknown object can belong to a given class or not. This is explained in more detail below. Finally, at the third level are the individual RS's. The features associated with a given RS (similar to those for an RS class) are: a CET containing geometrical deformations of the RS (which is a combination of templates of geometrically deformed models as shown in Fig. 5.9 (a)), a range of admissible aspect ratios, and a range of admissible color presence percentages for each color found on the RS. Appendix D shows images of RS models considered in the present system along with their grouping into classes. In the actual implementation, these models are placed in a directory tree. As regards the automatic generation of the information contained in the database starting from those models (*i.e.*the templates, color percentages, *etc...*), a program scans these directories recursively, and processes the models as will be described in Section 5.3.3. This needs to be done only once, unless the database is modified (*e.g.*adding an RS). Once this information has been generated, it is loaded at start-up by the RSR program each time it is ran.

**5.3.2 Database Recognition.** The previous section described the data structure underlying recognition, whereas this section will concentrate on the algorithm to perform recognition (the two should be distinguished, although of course the latter depends on the former). To illustrate the interplay between the data structure and the algorithm, a typical recognition example will be used. Consider the beacon which has been extracted by the detection sub-system at the left in Fig. 5.3 (a). It must be determined if a match can be found for it in the database. Recall that the color presences and the aspect ratio have been computed by the detection subsystem; these will be compared against the ranges of admissible color presences and aspect ratios at levels 2 and 3 in the recognition database.

Also output by the detection subsystem is the normalized labelled sub-image. Before recognition begins, this sub-image is encoded by the recognition subsystem using 9-bit binary words as was described in Section 4.4. The resulting IET (image encoded template) will be used in the course of recognition.

At this point, recognition *per se* may begin, and the algorithm to perform this will now be described, pursuing with the beacon example. From the detection process, it is

readily known to which color class the object belongs; yellow in the present case. The first level of recognition in Fig. 5.10 (color recognition) is thus already carried, and the search is restricted to a few RS classes having the appropriate color. The second level, called *class recognition*, is then entered. At this level, all RS classes within the color class are scanned, and for each RS class a check is made to see if the computed aspect ratio and all color presences fall in their respective admissible ranges (this can be viewed as a first recognition step using rough attributes). In the negative, the object is assumed not belong to this class, and next class is examined. In the affirmative, the correlation between the IET of the object and the CCET of the class is computed. If the correlation score is higher than a given threshold to be called the *class selectivity threshold*[5], the class is retained and this score is assigned to it. If the score is too low, the class is ignored. After all classes have been scanned, the one with highest score is chosen (provided there is at least one class for which the correlation score is above the class selectivity threshold) along with those for which the score is above threshold and sufficiently close to that of the class with highest score. The scores of two classes are considered sufficiently close when their values differ by less than a specified tolerance to be called the *class ambiguity tolerance*[6]. The reason for maintaining multiple possibilities at the class recognition level, and thus exploit ambiguity, when the above criteria are met is to avoid situations in which the class with highest score is not the right one and where the right one has almost the same score (*i.e.*just fell short). Simply choosing the class with highest score would lead to a wrong recognition result in such situations, whereby this can be avoided in many cases by maintaining multiple possibilities. Such situations occur at times in the present system, and must be handled. An example and further details on this point will be given below, but before this, the next level of recognition must be discussed. At this point, suffice it to say that the decision on which class is finally chosen is made after the RS's in each of these classes have been examined.

Once a set of potential classes is determined, the third level of recognition, called *individual RS recognition*, is entered where the object is compared against all RS's of each

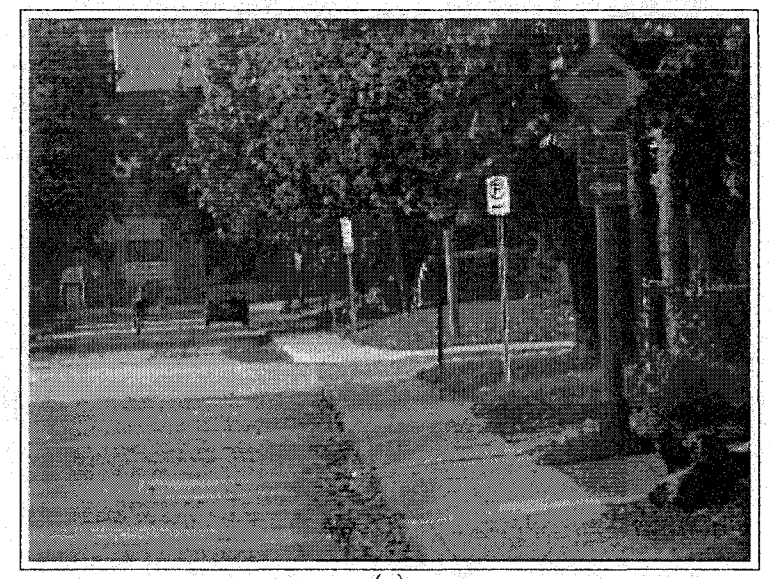
[5]Its value has been determined empirically by tests on a large set of real test images. There is a trade-off in its choice, as is always the case with thresholds. It should not be set too high, otherwise no class will be considered, and it should not be too low, for then there will be no selectivity. The threshold value was chosen so as to optimize the recognition performances on a set of test images. The empirical value found is 0.7. In effect, the class selectivity threshold selects only classes that have sufficiently high correlations scores.

[6]Its value is 0.01. This is also a parameter that was determined empirically over a set of test images.

of the possible classes in the same manner as it was compared with all classes at the second level, the sole difference being that the IET of the object is now correlated with CET's of individual RS's instead of CCET's. RS's for which the correlation score is sufficiently high are retained, and the others ignored. This results in a list of possible RS's along with their correlation scores and respective classes, which are sorted in descending order with respect to the correlation score. Finally, only the elements having correlation scores near that with highest score are kept (similarly to what was done at the class level, an *RS selectivity threshold* and an *RS ambiguity tolerance* are defined). These constitute the final possibilities. In most cases, the final recognition result is taken as the RS with highest correlation score in that final list (first in the list because of the sorting), and hence the final class is that associated with this RS (it should perhaps be recalled that this list may contain RS's across the possible classes, although most often only RS's from one class are present).

Referring back to the discussion on maintaining multiple classes, it is now clear that the choice of the class is made along with the choice of the final RS only after individual RS's have been examined. A question that should then perhaps be answered at this point is why is it more reliable to make the decision about the class at the individual RS level? The reason for this is that the CCET's used at the class recognition level are fuzzier than the CET's at the individual RS recognition level since they are composed of several such individual CET's. Hence, it is to be expected that a CET will give a more precise result than a CCET as regards resemblance to a given object. The pitfall in tree-like recognition structures such as the one used here is to take the wrong branch in ambiguous situations without being able to go back (which would be the case if only the class with highest score was retained). Here, this is avoided by considering all possibilities that are almost equally likely (or ambiguous) with the degree of ambiguity tolerated being quantified by the class ambiguity tolerance.

Considering the beacon example of Fig. 5.3 (a) again, in this case only one class is selected at the class recognition level, and at the individual RS level, only one RS is selected as well. This is a trivial example since this beacon class is quite dissimilar from the other classes pertaining to yellow and contains only one RS, namely the beacon itself.



(a)

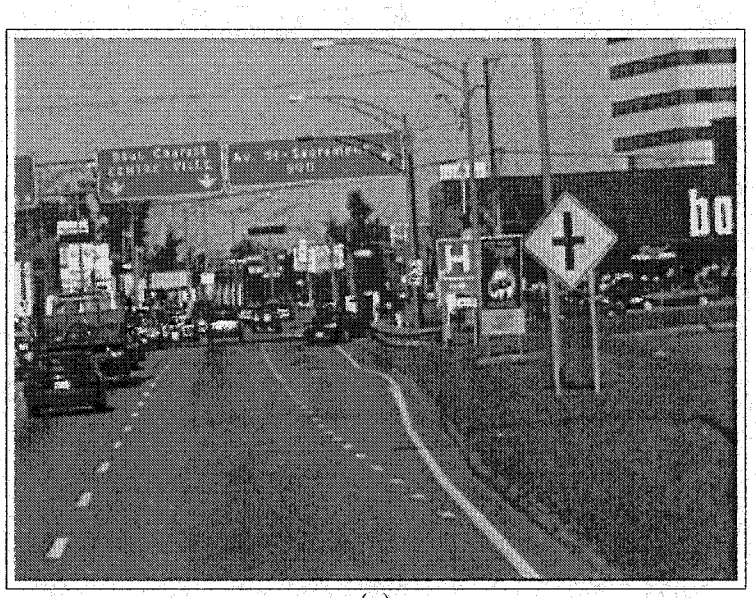
correlation score = 0.9897 correlation score = 0.9812

(b) (c)

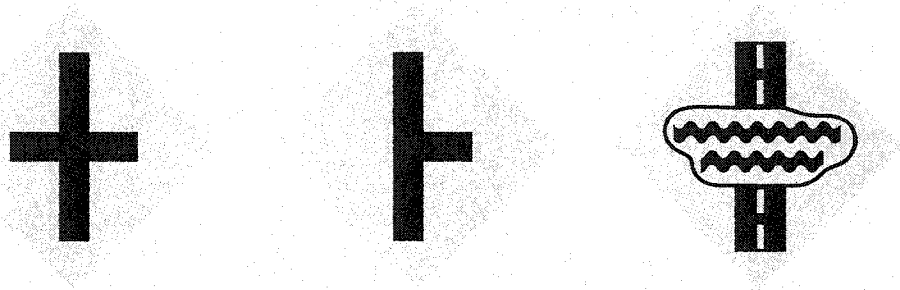
FIGURE 5.11. Recognition result where the final list at the individual RS recognition level contains two possibilities. (a) Original image. (b) RS with highest correlation score in the final list (this is the recognition result). (c) Second RS in the final list.

A somewhat less trivial example is shown in Fig. 5.11 for the yellow diamond-shaped RS at the top-right corner of the image shown in (a) captured in an urban scene. Here only one class is selected at the class recognition level, but the final list obtained at the individual RS level contains two possibilities, with the one having highest score being the correct recognition result. These possibilities along with their scores are given below the image in the figure in (b) and (c).

As a final example in this section, consider the yellow diamond-shaped RS indicating a crossing in the image in Fig. 5.12. For this RS two classes are retained at the class



(a)



correlation score = 0.9988 (b)

correlation score = 0.9890 (c)

correlation score = 0.9819 (d)

FIGURE 5.12. Recognition result where class recognition gives two possibilities, with that of highest score not being the correct one. (a) Original image. (b) RS with highest correlation score in the final list (this is the recognition result). (c) Second RS in the final list. Both (b) and (c) belong to the class *LosJn3*. (d) RS that would have been recognized, had the class of highest correlation score been chosen (*i.e.LosJn2*).

recognition level. To name them, these are *LosJn2* with a correlation score of 0.9999 and *LosJn3* with a correlation score of 0.9998 (see the figures in Appendix D for the listing of the classes and the RS's they contain; in the present case, the difference between the two considered classes is the amount of black they contain, with *LosJn2* containing more black than *LosJn3*). Here the correct class is *LosJn3*, and it is seen that it is very close to that with highest correlation score. By going deeper to the individual RS recognition level,

the ambiguity is easily settled as the final list of RS's only contains two elements, both belonging to the class *LosJn3*, with the correct one having highest score (the fact that the class *LosJn2* was initially identified as the one of highest evidence is due to the fuzzyness of the CCET's at the class recognition level). The results contained in the final list along with their scores are given in (b) and (c) below the original image in (a), and (d) shows the RS that would have been recognized, if the class *LosJn2* had been chosen[7].

To give typical numbers about the multiple possibilities retained by the algorithm in practice during recognition, at the class recognition level only rarely is there more than one class. When more are retained, 2 or 3 are typical. More than 3 has never occured in the present system and would be highly surprising since all RS colors typically do not contain more than 3 classes that are similar (see Appendix D), except perhaps for yellow, which has more similar classes than is the case for the other colors due to the variety of diamond-shaped classes it contains. At the individual RS's level, the number of possibilities can go up to 10, but 2 or 3 is more typical. It is easy to understand that for RS's within classes the number of retained possibilities may reach higher numbers than at the class level since within classes, the RS's are very similar: that is to say that differences are typically greater between than within classes.

Another point that should be mentioned with regards to the final list of possibilities is that eventually if tracking is implemented in this RSR system, it will be possible to update the scores associated with the elements in that list and use consistency constraints between these lists from subsequent images to achieve even further recognition robustness.

The TM approach introduced in the previous chapter is seen to play a key role in the recognition scheme just presented, which is very efficient due to the simplicity of the correlation (the correlation does not require complicated and lengthy computations). Results on the efficiency of this recognition approach will be given in Section 5.6. With the encoding, the templates generally introduce a high degree of discrimination between RS's of different classes and within classes, which is a desirable property considering the large number of RS's that must be handled.

Note also the important role played by color in this recognition scheme as it allows to efficiently direct the search in the database. If color were not used, the search would

[7]This RS is also a yellow with black diamond-shaped RS. Again, the reason it belongs to a different class is the amount of black content as explained in Appendix D.

be longer, as many more classes would need to be compared with, indeed all, unless a completely different approach were used, for instance, one which would group RS's in terms of exterior shape instead of color in a first coarse grouping stage as in [**49**]. However, as mentionned in the introduction of the thesis, such approaches suffer from the problem of having RS's that are identical in shape (or nearly), but different in color; refer again to Fig. 1.1. Also, as regards RS's, color appears to be intuitively more appealing for a first coarse grouping stage. Moreover, by using color it is possible to separate road signs into more classes than would be possible if only monochrome information was used. For instance, Appendix D shows that there are several yellow diamond-shaped classes, which are distinguished through their color content. This would hardly be feasible with monochrome information.

As a last remark, and independently of color, the idea of grouping road signs into classes has also been exploited in [**49**] (where it is called a hierarchy). In this latter case, a scale argument is used to justify the grouping, the motivation being that as scale gets coarser, similar objects get more similar. The approach presented here is different. Here also, templates of a set of similar objects are combined to define a compound template that represents the set (which is equivalent to the "prototypes" in [**49**]), but the motivation resides in a novel encoding that allows for such combinations. It may perhaps be claimed, that to an end-user knowing nothing about pattern recognition, the present approach would appear easier to use and closer to his intuition.

**5.3.3 Automatic Database Generation.** A question arising at this point is how the database is generated, and what parameters are involved. An important feature of this work is that the database content described in Section 5.3.1 is entirely generated off-line automatically by modelling the image formation process for each RS, as opposed to learning from examples techniques usually employed, and which use examples extracted by hand from images. This is a powerful feature of the present system as it allows any desired number of RS's to be considered in the database, and moreover human intervention is brought to a minimum in the generation of the database. From the modelling, all necessary parameters for recognition are computed (admissible aspect ratio and color presence ranges), as well as CET's and CCET's at levels 3 and 2 in the database. For modelling image formation, a model of an RS (noiseless reference image) placed at different locations in space and rotated

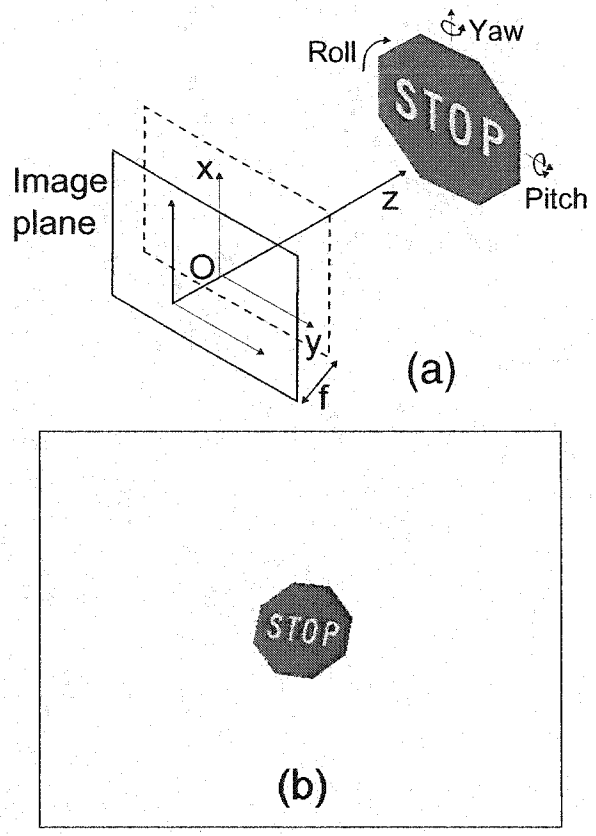



FIGURE 5.13. (a) Modelling image formation. The $(x, y, z)$ coordinate system is attached to the camera moving with the vehicle with $z$ along the vehicle and the $xy$-plane parallel to the image plane at a "focal distance" $f$ behind the $xy$-plane. A model raster image of the RS, here a stop $512 \times 512$ in size, is projected (mathematically imaged) onto the image plane. (b) Imaging example of a stop sign placed at $z = 10m$, with roll, pitch, and yaw angles of respectively 7.5, 15, and 15 deg.

to different degrees is mathematically projected onto an image plane. Fig. 5.13 (a) shows the geometry. The pinhole camera model is assumed (full camera calibration is not required in the present case), and the calculations are done using projective geometry [**130**]. Regarding the models used in the RS recognition database (Section 5.3), these have been drawn using commercial drawing software. They are then rasterized to standard dimensions. All model rasterized images have their greatest dimension scaled to 512 pixels with the other dimension scaled accordingly so as to preserve the aspect ratio of the object (in the stop sign example of Fig. 5.13, since the aspect ratio of a stop sign is 1, these dimensions are 512×512). These rasterized model images are the ones used to perform the modelling.

In normal conditions RS's are upright (parallel to the $xy$-plane) and located either at the right, left, or above the road. The viewing conditions considered are when the RS's are at least 5 meters away from the vehicle along $z$. At such distances with the camera optics used, $x$ and $y$ displacements have no significant perspective effects, on the image of an RS, unless it is for large values of $x$ and/or $y$, which are then out of the cameras's view[8].

In the modelling that has been performed, image formation has been considered for the RS's placed at discrete distances $z$ of 10 and 20 meters, for roll angles from $-7.5$ to $+7.5$ deg in increments of 2.5 deg, and for pitch and yaw angles from $-15$ to $+15$ deg in increments of 7.5 deg[9]. This results in 350 geometrical configurations, all of which are computed off-line for each RS, and which are typical for RS's along the road. Note that the number of configurations considered in the modelling must be constrained, otherwise the computation time to generate the database, which at present contains more than 400 RS's, becomes daunting. Fig. 5.13 (b) shows one such configuration for an RS placed at $z = 10m$, with roll, pitch, and yaw angles of 7.5, 15, and 15 degrees respectively in a $640 \times 480$ synthetic image (same size as the images grabbed from the camera). For each configuration, the BB around the synthetically imaged RS is determined, and the sub-image contained in this BB extracted. The aspect ratio and color presences are computed, and a normalized version of the sub-image resized to a maximal dimension of 64 (as in Section 5.2.2) is encoded to obtain a MET. The mean and standard deviations of the aspect ratios and color presences are computed over these configurations to determine the admissible ranges discussed earlier. Also, the MET's for these configurations are ORed together as in Eq. 4.6.1 of Chapter 4, and as depicted in Fig. 5.9 (a), to give the CET associated with the RS embedding the typical geometrical deformations of an RS when imaged along the road.

As seen, the whole process of detection, extraction and encoding is performed for those synthetic images in much the same way as is done for real images of road scenes, thereby imitating the real situation. By applying such a procedure to each RS model in a class, the CET's, aspect ratios and color presences of all RS's in that class are obtained, which are then used to compute the CCET, as well as the admissible ranges for this class. In this

[8]The effect of $x$ and $y$ displacements of the RS can also respectively be nearly compensated by considering pitch and yaw angles.

[9]It should be noted that the effect of pitch and yaw are not as significant as for roll on the correlation, see [**105**] and Section 4.6 for a discussion on this point.

manner, all the data necessary to perform database recognition is generated; everything is computed and modelled from first principles. A main advantage of the model-based method just presented is that the database is scalable; adding RS's or modifying it is straightforward, as only models need be added or parameters be changed. Then it is a matter of letting the computer do the calculations. There is no need to go through a whole learning process, and no supervision is required.

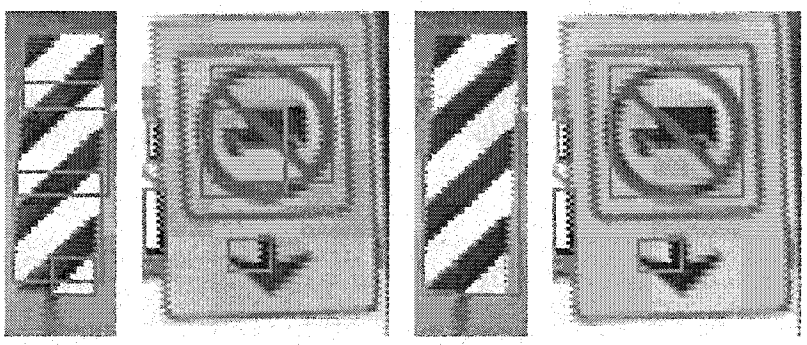
## 5.4 Improvements

During the development and implementation of a system for a real-world application, particular problems are always encountered, which were not, and could not be accounted for in the initial overall system design. Such problems must be dealt with to further increase the system's performance. Here, the following problems were encountered: 1) fragmented RS's requiring ROI grouping, 2) ROI's necessitating further re-labelling for more accurate detection results, 3) sub-image background cleaning prior to recognition, and 4) ROI's that need to be filtered out because they are too small in size or do not have interesting content. Keeping the latter ROI's slows down the recognition process, hence the need to eliminate them. These problems will now be described along with their remedies. It should be noted that in the case of 1) and 2), the solutions involve a partial recognition, in the former case, in the form of a hypothesis-verification procedure, and in the latter as a feedback mechanism of recognition over color segmentation (or labelling).

**5.4.1 ROI Grouping.** In some cases, RS's are fragmented by the detection. This is why an "ROI Grouping" processing step appears in Fig. 5.1. Fragmentation occurs, because in some cases the color detection procedure cannot detect an RS as a single connected component. There are two instances of this problem of a different nature: *i*) the problem is inherent as in beacons where the RS is composed of multiple connected components, and *ii*) detection fails to detect an RS as a single connected component. Fig. 5.14 (a) and (b) shows instances of both cases[10].

The problem is solved by resorting to a hypothesis-verification procedure. We ask what would happen if the fragments were grouped (hypothesis), would an RS class be recognized

[10]In the case of prohibition RS's as in (b), the problem is due to the fact that the color detection approach is unable to detect the red circle as a single connected region because the circle in very thin, often only one to two pixels wide in the image.



(a) (b) (c) (d)

FIGURE 5.14. Fragmented RS's. (a) Beacon composed of 4 yellow connected components. (b) Prohibition RS for which the detection algorithm could not extract the red region as a single connected component due to the thinness of the circle. (c) and (d), same as (a) and (b), but after ROI grouping.

(verification)?[11] In the affirmative, the fragments are grouped together for good to form an RS. Knowing the RS classes for which fragmentation problems occured, the procedure targets those particular cases. For each of these cases, the grouping of the fragments is carried by grouping their BB's following some adjacency rules specific to each RS class involved. Results of applying this procedure to the examples of Fig. 5.14 are given in (c) and (d). Fig. 5.15 depicts in random colors the grouping that has taken place for the image of Fig. 5.3; five RS's required grouping in that case: the beacon at the left of the image and the four chevrons at the right.

**5.4.2 Re-Labelling with Partial Recognition Feedback.** Referring back to Fig. 5.7 (c), if the class of the RS could be determined, then a re-labelling of the red pixels would be possible since for the class to which the RS belongs, no red pixels should be present. This leads to re-labelling based on partial recognition (class recognition) which will now be described. The purpose of such a re-labelling here is to obtain a better color labelling of the potential RS to ultimately improve recognition at the individual level. By knowing the class of an RS through a partial recognition, it is possible to restrict the set of colors to be considered for re-labelling (*e.g.*for the RS in Fig. 5.7, this set contains only yellow and black). This approach can be seen as a feedback (or top-down) mechanism,

[11] Class recognition is sufficient for the purpose of verification.

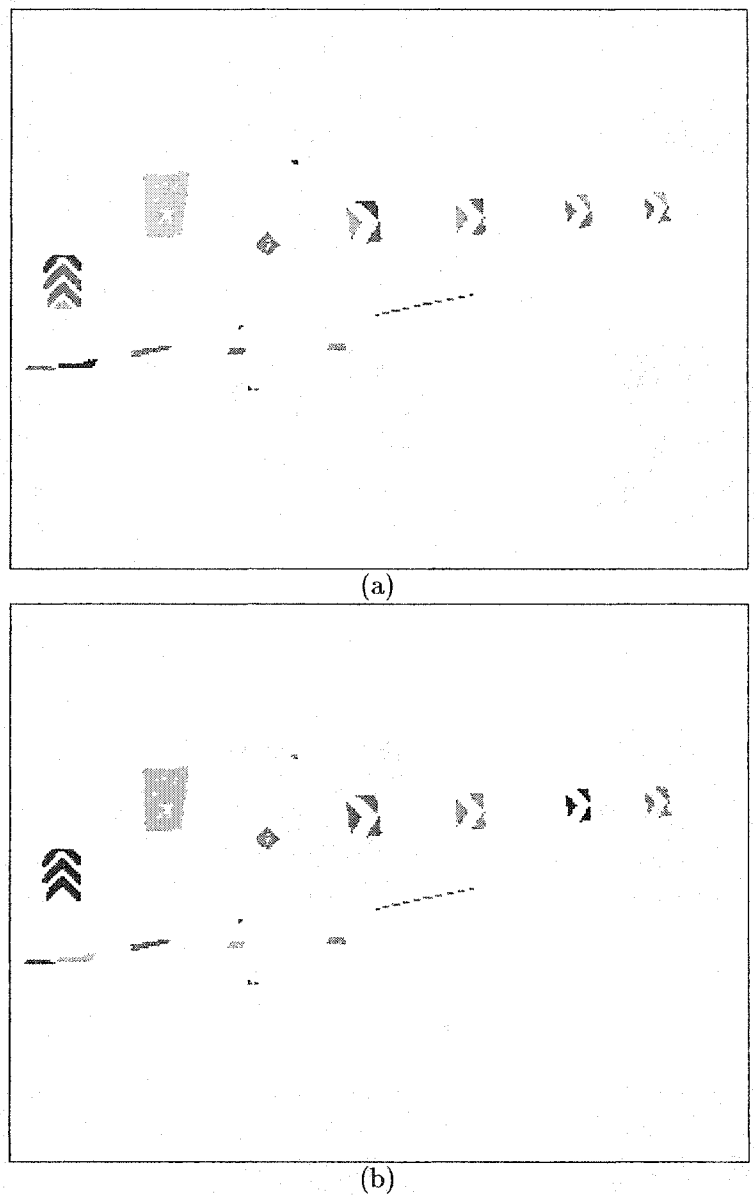
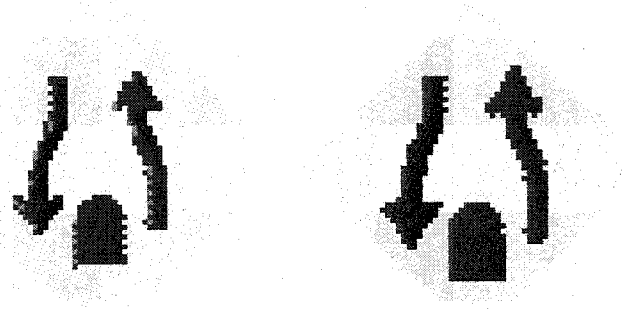


FIGURE 5.15. Illustrating grouping for the image of Fig. 5.3. (a) Extracted color regions prior to ROI grouping shown in random colors. (b) Color regions after ROI grouping.

because the information provided by the partial recognition (higher level information) is fed back to the color labelling mechanism in order to impose consistency[12].

[12]This is a limited feedback in the sense that only one loop of feedback is carried, the procedure is not iterative. This is somewhat different from what is usually seen in feedback control systems, where the feedback loop is carried iteratively until some convergence is achieved. Here, one can consider that convergence is attained after only one loop.



(a) (b)

FIGURE 5.16. (a) Initial labelling (same as Fig. 5.7 (c)). (b) Re-labelling using class recognition information. Red pixels are now correctly re-labelled. Notice that the border of the RS has changed due to the re-labelling.

In Fig. 5.7, pixels initially labelled as red are either re-labelled as yellow or black depending on their color intensity in the original image: if the color intensity is smaller than the mean color intensity of yellow pixels, then the pixel is re-labelled as black, otherwise it is re-labelled as yellow. Fig. 5.16 shows the final re-labelling result. Notice that white pixels have also been re-labelled (their situation is similar as that of red pixels, recall the caption of Fig. 5.7).

By knowing the class of an RS, it is even possible to push the processing further by exploiting the spatial structure of the RS in performing a re-labelling. Fig. 5.17 shows an example: (a) is the original image, (b) shows the initial pixel classification, (c) gives the initial re-labelling, and (d) depicts the RS that would be recognized if only the initial labelling were performed. Clearly, in the initial re-labelling many pixels were wrongly labelled as black (pixels along the diagonal and along the inner side of the red enclosing annulus). These contribute to the wrong recognition result shown in (d). Knowing the class of the RS, and thus its spatial structure, the black pixels along the diagonal and the inner side of the enclosing annulus can be re-labelled as red. This is depicted in (e), and (f) shows the recognition result after re-labelling with the RS being correctly identified.

Fig. 5.18 shows further examples using partial recognition feedback for re-labelling. For the image displayed in (a), (b) shows that the initial re-labelling detected black pixels along the inner side of the green enclosing annulus. This is corrected for in a similar manner as was done for the image in Fig. 5.17. Also, green pixels were detected on the truck. Knowing

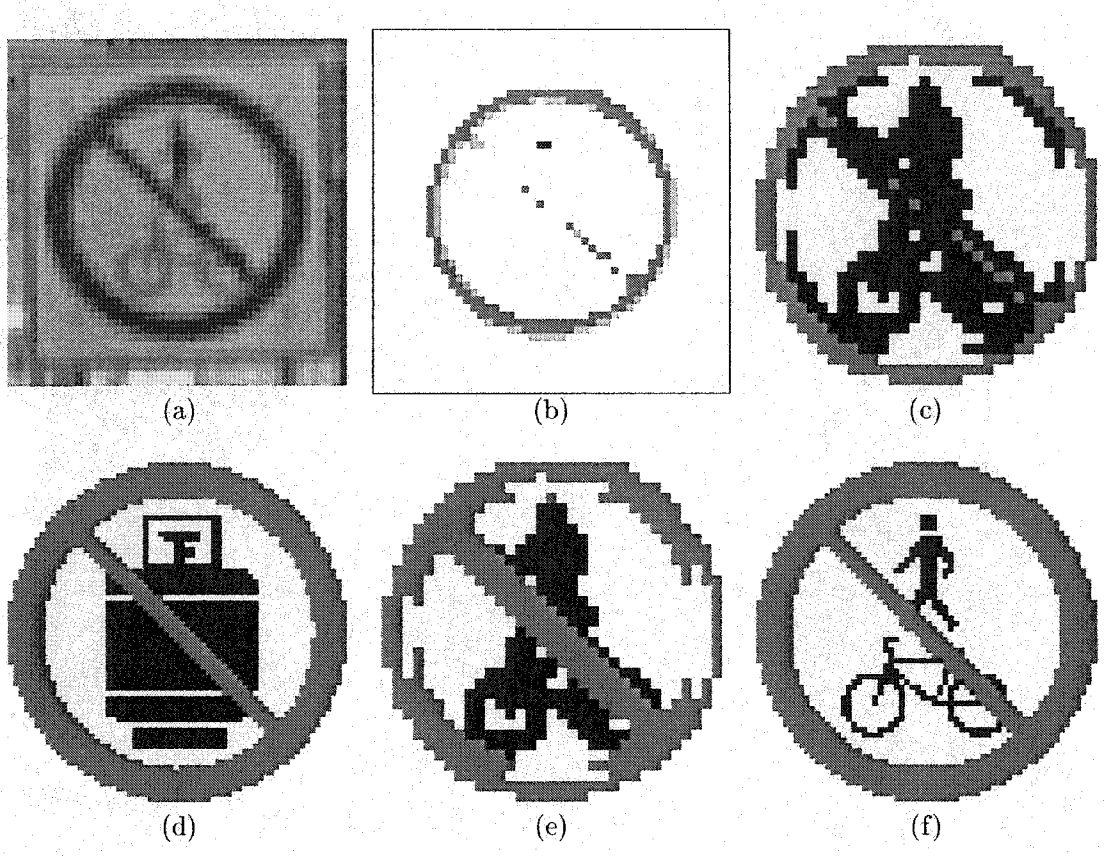



FIGURE 5.17. Exploiting the spatial structure for re-labelling from the information gained through a partial recognition. (a) Original image. (b) Initial pixel classification. (c) Initial re-labelling (Section 5.2.3). (d) Object recognized when only initial labelling is performed. (e) Re-labelling using partial recognition. (f) Object recognized when re-labelling is performed.

that green pixels cannot be present in the center region for the RS class to which the RS belongs , these pixels can be re-labelled correctly; (c) depicts the end result. As regards (e), white pixels are incorrectly labelled as black by the initial labelling. In this case, the problem is that in the generic algorithm used in the initial labelling (see Fig. 5.5), pixels are labelled as black or white based on the intensity statistics of the green pixels. It is possible to re-label black and white pixels based on the statistics of the white pixels instead, and (f) depicts the final result, showing a clean final labelling.

Another use of partial recognition is to better determine the contour of objects. This is illustrated on an example. Consider Fig. 5.6 (c) again which is the result of the initial re-labelling. It is seen that black pixels are present along the outer side of the border, and

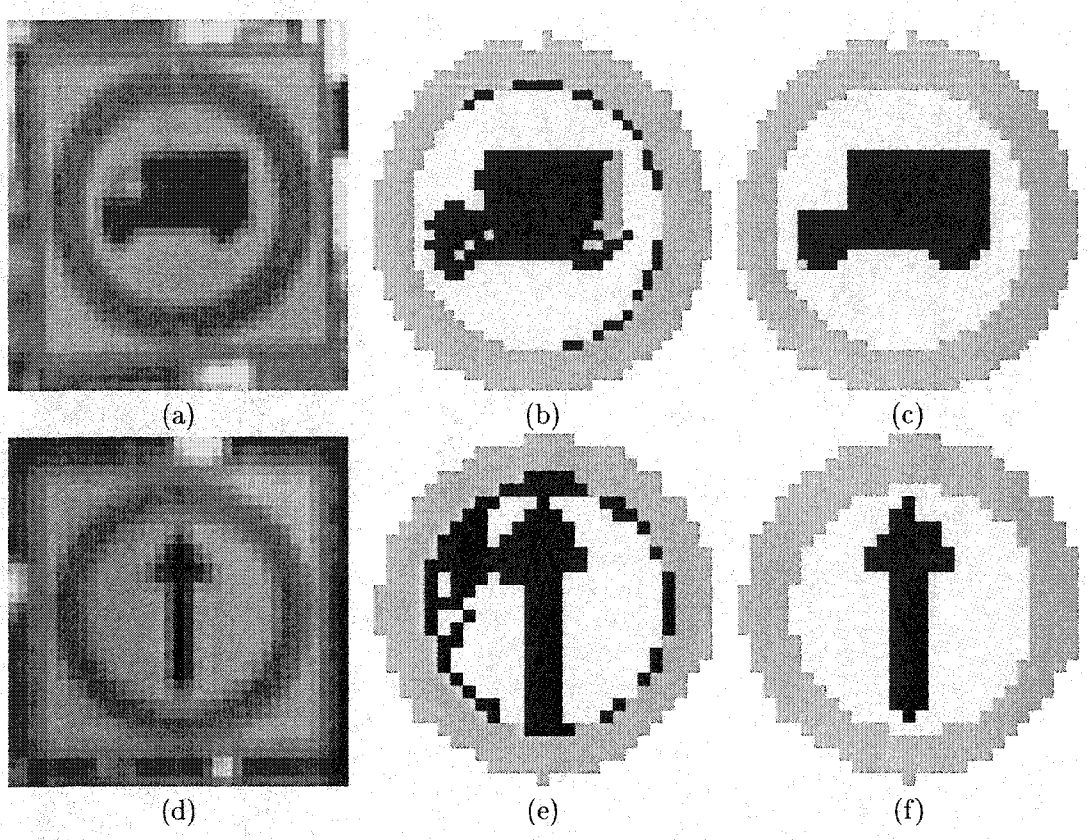



FIGURE 5.18. Further examples exploiting the spatial structure for re-labelling from the information gained through a partial recognition. See text for details. (a) Original image. (b) Initial labelling. (c) Re-labelling using partial recognition. (d) Original image. (e) Initial labelling. (f) Re-labelling using partial recognition.

also that the vertices of the diamond-shaped object are not sharply determined (in that case, yellow pixels could not be re-labelled as black by the generic re-labelling algorithm). Even if the contour is somewhat inaccurate, a partial recognition is nevertheless able to determine that the object is diamond-shaped. This is then used as a priori information to redetermine its contour. The algorithm starts by finding the vertices of the object (by prior knowledge of the shape searched for, this is an easy task), and then a line is fitted by least-squares to the contour between each vertex by using the border pixels having the object's principal color. With the contour of the object determined in this manner, pixels oustide that contour are re-labelled as $OTHER$, and those inside the contour in its neighborhood as the principal color. Fig. 5.6 (d) depicts the result of such a processing.

As a final note on using partial recognition for labelling purposes, obviously the information fed back must be reliable in order that the mechanism be sucessful. In the present system, class recognition, which is used for partial recognition, is very efficient; only very seldomly is the class of an RS wrongly determined. This is the reason it can be relied on, and has motivated the partial recognition feedback approach presented here (and also the use of class recognition for grouping ROIs).

**5.4.3 Sub-Image Background Cleaning.** In some instances, pixels having RS colors are in the background of an RS. These must be discarded prior to recognition. Fig. 5.19 illustrates the point.

In the present system, once an ROI is determined, all pixels in the background of the ROI's BB which do not have the ROI's principal color are cleaned. This is done by starting at the border pixels of the ROI's BB and scanning inwards. As each pixel is encountered, its label is set to $OTHER$, unless it has the ROI's principal color label, in which case the scan stops. Such scanning is repeated for all pixels on the BB's border (horizontal scan for pixels on left and right borders, and vertical for upper and lower ones). Fig. 5.19 (d) shows the region in (c) after cleaning.

**5.4.4 ROI Filtering.** Prior to recognition, a final step of ROI filtering is carried. Two types of filtering are considered: 1) filtering on the size of ROI's BB's (recall that an initial ROI size filtering step was carried at the ROI extraction stage), and 2) filtering on the color presence percentage of the principal color.

As regards size filtering, ROI's with either dimension (width or height) smaller than 25 pixels are discarded as they do not contain enough detailed structural information for reliable recognition.

With respect to filtering based on the presence percentage of the principal color, ROI's for which this percentage is below a given minimal value are eliminated. This minimal value depends on the principal color considered, and is obtained from the RS class for which the principal color is least present. The theoretical value of color presence obtained while generating the RS recognition database is used as a starting value. This theoretical value is then reduced by a fixed amount to account for the fact that in real situations the extracted color presence percentages are often smaller than their theoretical counterparts.

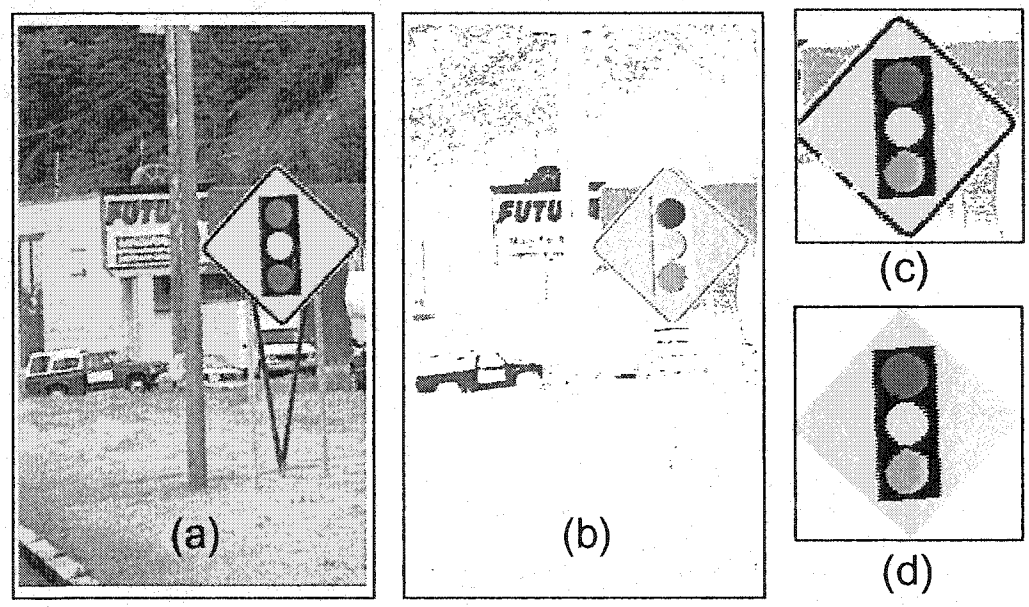



FIGURE 5.19. (a) Original image. (b) Labelled image. Notice the green pixels in the back of the yellow RS. (c) Extracted ROI without background cleaning, and (d) with background cleaning.

For instance, for red RS's, there are 8 classes. Of those classes, the one having lowest theoretical red color presence is the prohibition class with 32% (a representative of this class is the RS shown in Fig. 5.14 (b)). The final value chosen is 8% (which is conservative) for the minimal red color percentage that a red ROI should have in order to be considered for recognition.

The filtering described here considerably speeds up recognition by having it not to consider superfluous ROI's that are either too small or do not contain a sufficiently high percentage of principal color. Fig. 5.20 shows the remaining ROI's after ROI filtering and prior to recognition for the image shown in Fig. 5.3. Notice how small BB's that contained no useful information have been eliminated (size filtering), and also how part of the yellow line marking on the road in front of the car has been discarded (color presence filtering).

## 5.5 Global Image Understanding

Thus far, RS's have been considered as isolate entities without reference to their context in the whole image and without exploiting the relationship they might have to other RS's that may be present in the image. Such type of higher level information can be used

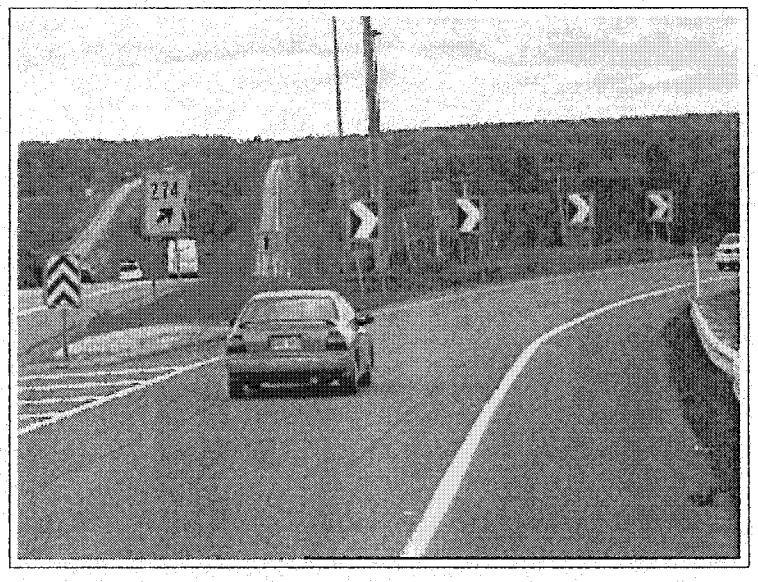


FIGURE 5.20. Remaining ROI's after ROI filtering for the image of Fig. 5.3.

to impose further constraints on the RS's that are recognized by the system in order to eliminate false positives or incorrect recognition results. In this sense, the processing presented in this section is termed global image understanding because it considers the whole image in the decision making process after recognition has been performed on the potential individual RS's found in the image (this could have also been termed RS scene interpretation). For instance, RS's belonging to certain classes never appear isolated, but are rather always accompanied by other RS's nearby (most usually above). Hence, this can be used as a constrain, and if such an RS is detected isolated in the image, it can be considered a false positive (as will be seen, in some cases this constraint can be supplemented by a further constraint on the absolute position of the RS in the image in order to increase the confidence in the decision). In other instances, the absolute position of the RS in the image is sufficient for not considering it. It is not the purpose to describe exhaustively all the constraints that can be imposed on RS's belonging to certain classes, but rather to illustrate the previous discussion with a few specific examples that will now be given.

Fig. 5.21 shows an image where the green RS behind the yellow diamond-shaped one is recognized as a small green RS belonging to the class *RectPet2* (see Appendix D), whereas it is in fact a big directions advertising RS located far away; (a) is the original image and (b) shows the result. The fact that this RS is recognized as a member of the class *RectPet2*

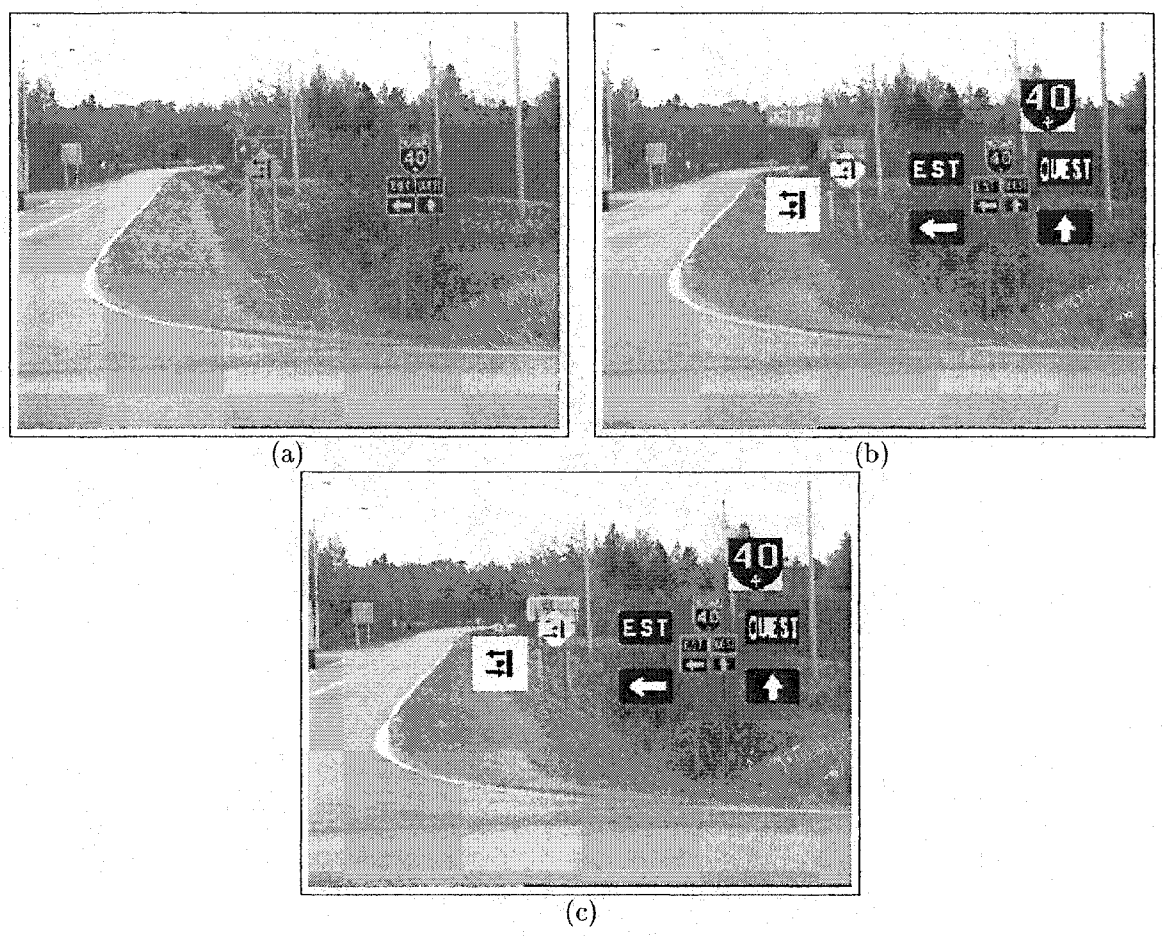



FIGURE 5.21. (a) Original image. (b) Recognition result without imposing an adjacency constraint. The green RS behind the diamond-shaped yellow one is recognized as a junction RS, which is not correct. In fact, the green RS is a guidance sign (which are not at present interpreted by the system). (c) Imposing adjacency constraints resolves the wrong recognition of the green RS. Objects detected by the system and identified as RS's are enclosed by a magenta bounding box whereas those not identified as RS's are enclosed by a light orange bounding box.

is not a flaw of the system in itself since the RS has the appropriate aspect ratio, color content, and correlates sufficiently strongly with the model templates in that class. Further information must be provided to the system for it not to recognize the RS as a member of this class. In fact, the RS's in this class are always located underneath another green RS having the same width, and with which the left and right borders are well aligned. This is an adjacency constraint that must be imposed on RS's belonging to this class. Providing this information to the system, the RS is not recognized anymore as a member of *RectPet2*.

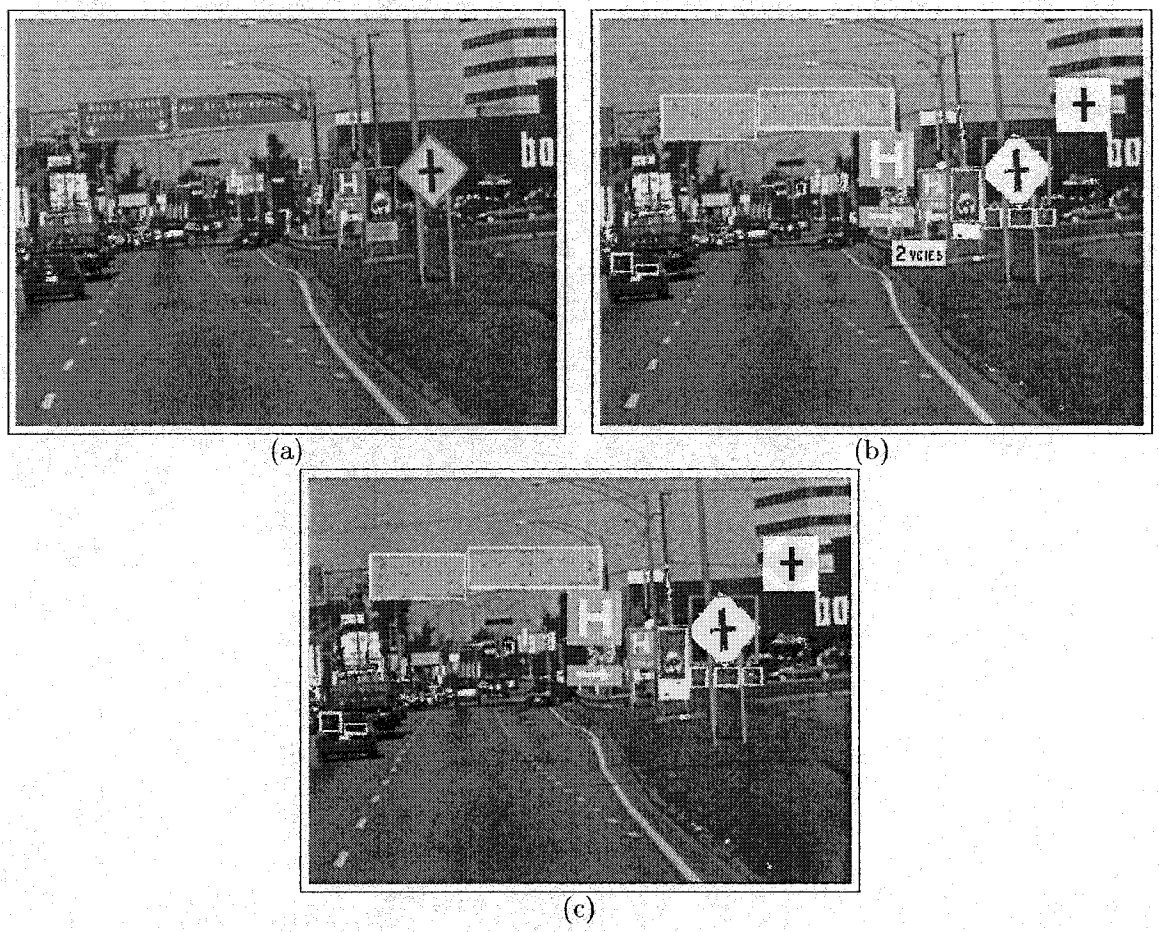



FIGURE 5.22. (a) Original image. (b) Recognition result without imposing an adjacency constraint. (c) Imposing adjacency constraints resolves the wrong recognition of the yellow RS placed underneath an object detected as red (and not recognized as an RS).

This result is given in (c) (in the result images presented here and in the following, objects detected by the system are enclosed in a magenta bounding box if they are identified as an RS and in a light orange bounding box if they are not; the pixels that have initially served to detect the objects are also displayed in their respective colors). Here the RS is still detected by the system, but it is not identified as any of the RS's in the database. As mentioned previously, these big green advertising RS's are not interpreted by the current system (as are also blue and yellow big ones). Fig. 5.22 shows another example where the same condition is imposed, but to a yellow RS class (the class *RectAnn*, see Appendix D). Without the constraint, (b) shows that a yellow rectangular sign is recognized (which is not

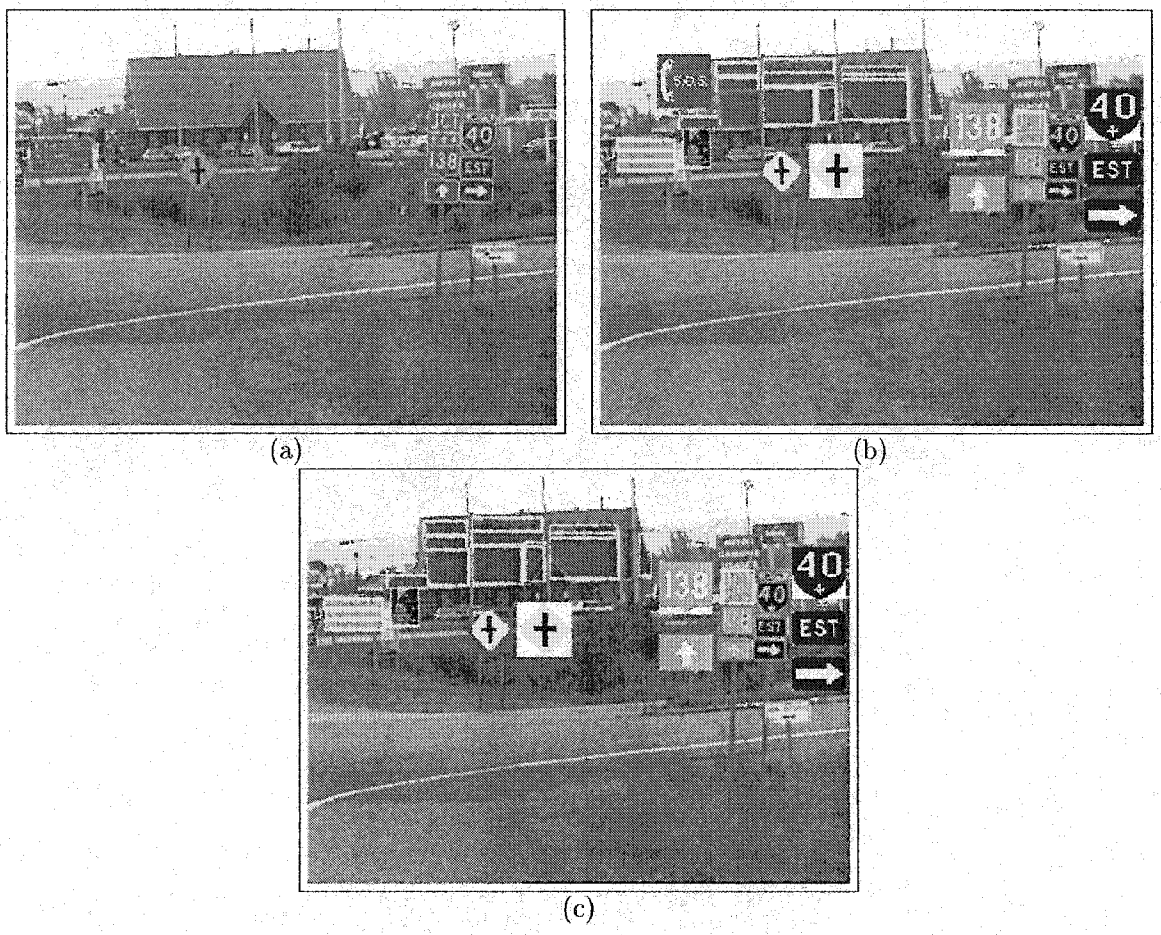



FIGURE 5.23. (a) Original image. (b) Recognition result without imposing a constraint on the position in the image. (c) Imposing a position constraint resolves the false positive recognition of an RS.

an RS), whereas (c) shows that with the constraint the same sign is not recognized anymore (this sign is underneath another sign with their left and right borders well aligned, but the sign above is a red one).

Whereas the previous examples showed situations where the recognized objects that needed to be filtered out from the list of recognized objects were signs that were argued legitimate to be recognized without constraints, the next example in Fig. 5.23 presents a case where a true spurious detection (which cannot be avoided from a color detection perspective since the pixel colors fall in the red color detection region) is recognized as an RS. This is the red SOS sign shown in (b), which in fact corresponds to a part of a red roof.

This recognition can be considered fortuitous for the aspect ratio falls into the admissible range of the SOS class, whereas this is not the case for other parts of the roof that are also detected. The constraint to be imposed on RS's in the SOS class is that they should not be located in the central upper part of the image (except perhaps in exceptional circumstances which were never seen). With this constraint, the recognition of an SOS sign is eliminated as (c) shows.

In applications where a vision system needs to analyze complex scenes such as those considered here, there will always be spurious detections and/or false positives, and structural constraints such as those discussed in this section provide further mechanisms to filter out spurious data or unwanted recognitions. In this system, sensitivity has been maximized in order to loose the least RS's possible in the early stages of processing, but the compromise to this is that in the end more filtering must be performed, in which global image understanding takes part.

As a final note to the present section, it should be mentioned that constraints related to the absolute position of an RS in an image or to its relationship with other RS's in the image, such as proximity, have been exploited in other work, but for different purposes. For instance, Azami and Aoki [**14**] use horizontal proximity to find other road signs in order to cope with the unreliability in the detection scheme they employ. In [**6, 30, 40, 51**], processing is restricted to a predetermined sub-region in the right part of the image to increase processing speed (note that as a by-product this also eliminates false positives that could potentially be found in non-searched regions). Here, constraints on position are less stringent than in those other works as they are not imposed to be the same for all road signs. Indeed, the allowed positions with respect to the road are not necessarily the same for all road signs, as assumed in those other approaches, and this is taken into account here. For instance, some road signs (such as yields) are allowed to be placed to left of the road, while for others this is not possible. With the approach in [**30**], which restricts the search for RS's to the right central part of the image, RS's at the left in the image cannot be found.

As regards Azami and Aoki, this is an interesting idea that could be applied to find other RS's, especially achromatic ones, but as concerns colored RS's, the present approach

has proved sufficiently reliable for not having to resort to such techniques. Here, proximity is used to prune out impossible situations, and hence avoid false positives.

## 5.6 Results

In this section, performance figures in terms of the detection and recognition ability of the system will be given. Next, the issue of processing time will be discussed. Finally, a few case studies will be presented in detail to give an idea of the type of images processed by the system and their degree of complexity. Examples have already been presented in the previous section, and more will be given here.

**5.6.1 Detection and Recognition Performance.** The images used to test the system's performance were taken along the road with a 3-CCD color camera (Panasonic GP-US502) in an enclosure mounted on the roof of a truck[13]. A computer with frame grabber (Matrox Meteor RGB board) in the truck acquired the images at a rate of 10 frames per second (fps). The images were captured at speeds between 0 and 70 km/h[14] in different wheather conditions (sunny, cloudy) and at different times of the day (morning, afternoon). Image sequences have also been acquired. In fact, most acquisitions were in the form of image sequences (10 to 20 images per sequence). Although their processing is not the subject of the present work, image sequences are at hand for the eventual development of a tracking module.

To measure the system's performances, a set of 860 test images has been considered. An important factor in testing are the minimal ROI dimensions. A first series of tests was performed by requiring all ROI's down to dimensions of $25 \times 25$ to be considered for recognition. This corresponds to a total of 1460 RS's that should be recognized in the test images (many images contained more than one RS, and also some contained no RS in order to test for false positives). The first two rows in Table 5.1 give the results for $25 \times 25$ minimal dimensions as indicated in the leftmost column (column 1). The third column,

[13]The author wishes to acknowledge the Ministère des Transports du Québec for having lent this truck numerous times.

[14]At higher speeds, acquisition was too much affected by bumps on the road and vibrations. Also, the camera used was of the interlaced type. For such cameras, an image (or frame) is composed of two fields (even and odd) which are acquired 1/60 sec. apart. At high speeds, the effect of interlacing becomes too severe. In further work, a progressive scan camera which does not lead to interlacing problems shall be used. At 70 km/h with a frame rate of 10 fps, this corresponds to taking an image every 1.9 meters.

| Minimal Dims | | Color Detection | Class Recognition | Individual Recognition | False Class | False Individual | False Positives | Nb. of RS's |
|---|---|---|---|---|---|---|---|---|
| 25×25 | Absolute | 1407 | 1302 | 1261 | 85 | 116 | 71 | 1460 |
| | Percentage | 96.4% | 89.2% | 86.4% | 5.8% | 7.9% | 8.3% | |
| 35×35 | Absolute | 1007 | 978 | 946 | 25 | 44 | 34 | 1031 |
| | Percentage | 97.7% | 94.9% | 91.7% | 2.4% | 4.3% | 4.0% | |

TABLE 5.1. System performance figures.

entitled "Color Detection", shows the system's efficiency for detecting RS's using color in both absolute and percentage values. Only a few percent are not detected. Some examples of this will be given in Section 5.6.3. The fourth column gives the system's efficiency in recognizing RS classes, *i.e.*class recognition (Section 5.3.2). For nearly 90% of the RS's, the class is correctly recognized. Column five gives the efficiency at the individual level (individual RS recognition): a little more than 85% of the RS's are correctly identified. Column 6 gives the false recognitions at the class level, *i.e.*an RS is detected and a class is recognized, but this class is wrong. False recognitions at the individual level (*i.e.*an RS is wrongly recognized) are given in column 7. Errors are of 7.9%. Of those, 5.8% are caused by class recognition errors (if the class is not well recognized, the individual RS cannot either). Also of importance are false positives, *i.e.*something is recognized as an RS (down to the individual level), although it is not. These are of 8.3% as given in column 8. Note that the percentages for false positives is computed as the number of false positives (71) referred to the total number of images (860), whereas the other percentages given thus far are referred to the total number of RS's to be recognized (1460). For false positives, referring to the total number of images appears to be a better measure, because false positives have no direct relationship to the number of RS's there are to be recognized. The rate for false positives is somewhat high, although recognition rates are relatively high. Most of the errors ("False Class", "False Individual", and "False Positives") occur for small ROI's, meaning that requiring recognition down to $25 \times 25$ pixels is demanding (this is small in $640 \times 480$ images). A second series of tests was carried by relaxing this constraint to $35 \times 35$ pixels. The performance figures are seen to improve significantly (rows 3 and 4 in Table 5.1; the number of RS's to be recognized is then smaller). Hence, considering sizes down to $25 \times 25$ was perhaps too stringent, but shows the system's limits (such tests help in characterizing the system). Requiring larger dimensions means that the vehicle must get closer to an RS

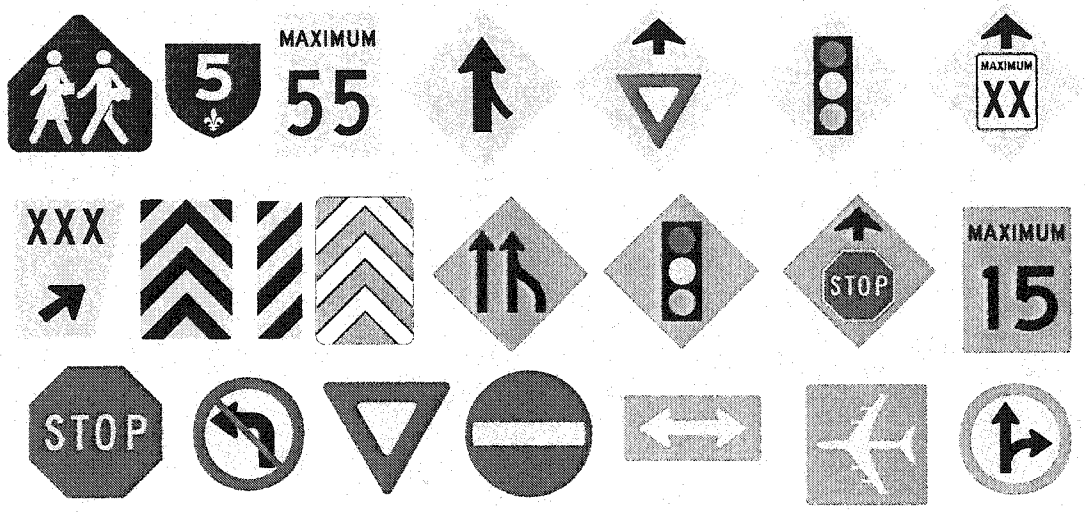



FIGURE 5.24. RS's representative of a subset of all considered classes.

before it can recognize it, which is not a problem as the vehicle is moving. Also, in further work tracking should be added to the system which will further improve the performance figures.

The results for $35 \times 35$ minimal dimensions compare advantageously with systems that are considered to constitute current state-of-the-art. In [**46, 47, 48, 49**] and [**30**], recognition rates respectively above 90% and of 90% are claimed, but this is for a limited set of RS's (circular and triangular).

Somewhat lower results, about 90%, are reported in [**1, 37**], but with a larger database (square and chevron RS's being also considered). The main point for the results of the present work is that they were obtained with what is to the author's knowledge the largest RS recognition database reported thus far. In all there are 41 classes distributed over the different RS colors containing a total of 447 RS's (not considering black and white RS's). Fig. 5.24 shows 21 RS's representative of a subset of all the RS classes considered in the present system, and for more details the reader is again referred to Appendix D.

**5.6.2 Processing Time.** Tests have been carried to determine the processing speed of the current system. The average processing time of a $640 \times 480$ image is currently 0.65 second on a 1 GHz Pentium III computer. At present, no significant efforts have been put on optimizing this aspect of the system, and in some parts of the system general-purpose

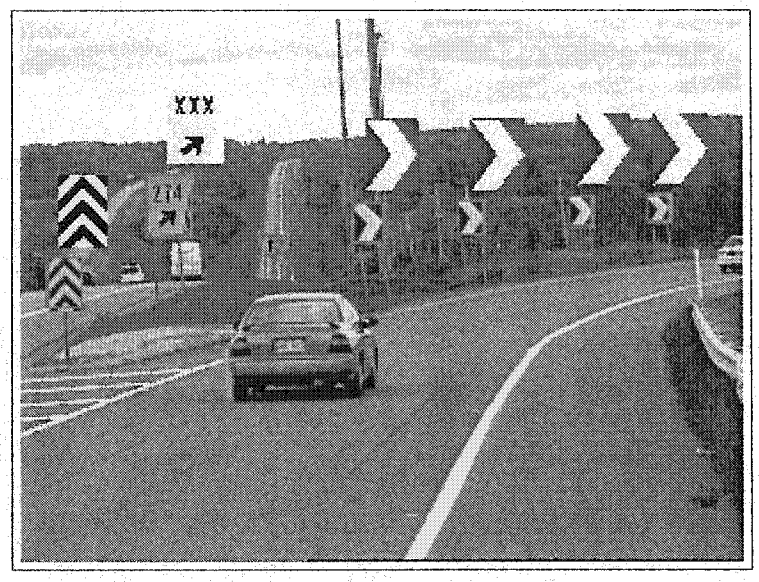


FIGURE 5.25. Recognition result for the image of Fig. 5.3.

algorithms are used, which can be further tailored to the specific RSR application. Also avenues exploiting the parallelism allowed for some operations on MMX technology and using a dual-processor computer are still to be explored.

**5.6.3 Case Studies.** As a first example, Fig. 5.25 shows the final recognition results for the image of Fig. 5.3 that has served for illustrative purposes since the beginning of this chapter. All RS's are correctly recognized. The yellow RS above the car, although detected, is not considered for recognition because it is smaller than 35×35. The red chevrons and the black and yellow beacon required ROI grouping prior to recognition. The number appearing on the exit RS (274) at the left of the car is not identified. As said earlier, the current system does not perform OCR.

The next series of examples illustrates situations where the system encounters difficulties. Figures 5.26 and 5.27 depict cases where the system failed because it detected other objects in the background of an RS which have the same color as the RS. This will be called background interference. Such objects maybe other RS's as Fig. 5.26 shows or natural objects as in Fig. 5.27. Background interference prevents an RS to be appropriately detected and segmented leading to no class recognition at all (Fig. 5.26 (d) and Fig. 5.27 (a)) or to a faulty individual recognition (Fig. 5.26 (a)). In some instances, such as in Figs. 5.26 (b)

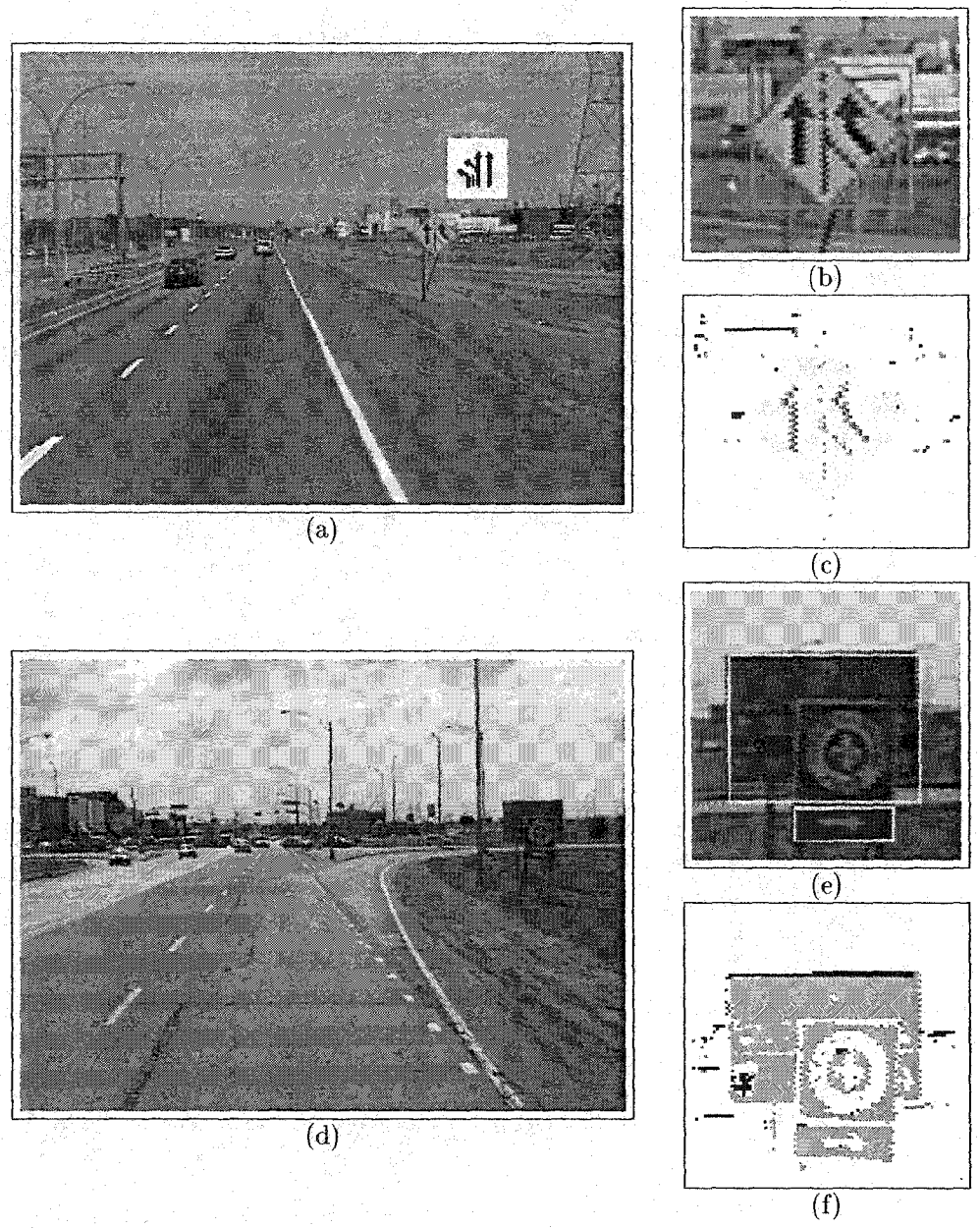



FIGURE 5.26. Examples of background interference with other RS's as interfering objects. (a) Original image with recognition result, which is wrong. (b) Zoom on the interfering RS's. The presence of a small RS at the right of the larger one can be seen. As a note, one can see the effect of interlacing in this zoom. (c) Zoom on the initial pixel labelling of the RS's. (d) Other example (original image). Here, interference does not lead to wrong recognition. The square RS with white annulus is detected as part of the bigger rectangular RS (which is a RS that would require OCR for its full interpretation). Strictly speaking, the square RS is not detected. (e) Zoom on the interfering RS's. (f) Zoom on the initial pixel labelling of the RS's.

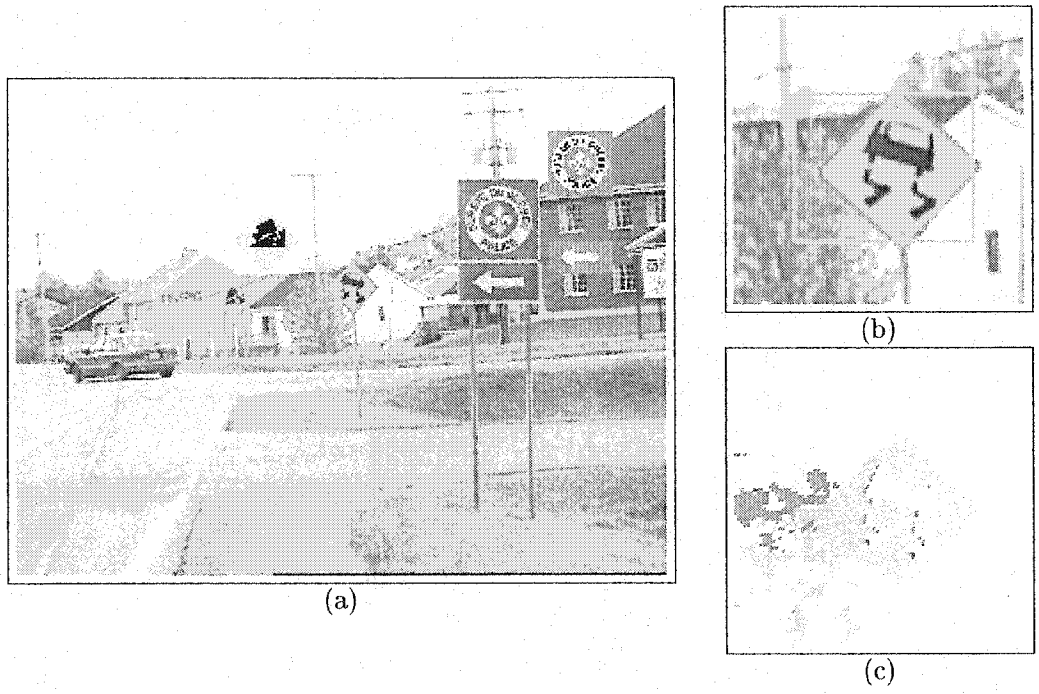
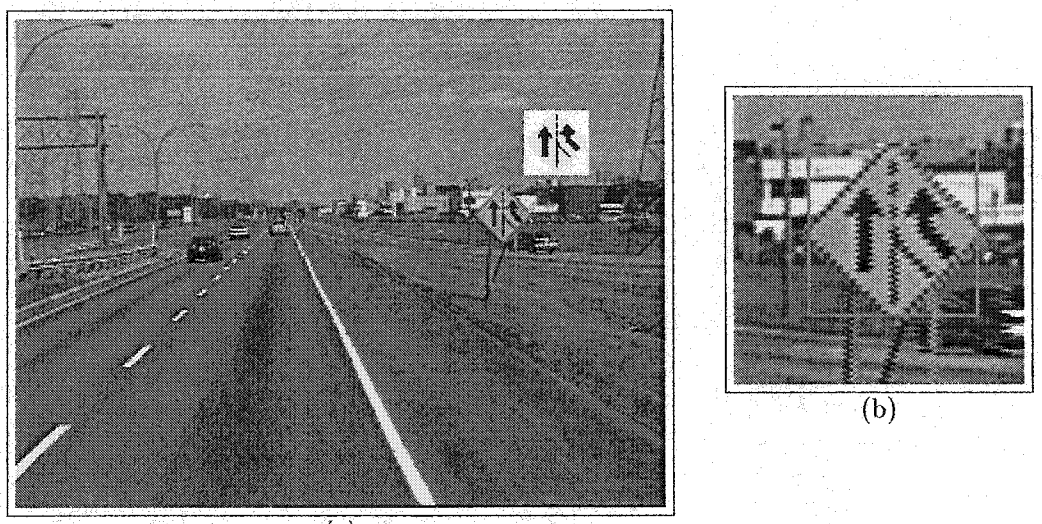



FIGURE 5.27. Examples of background interference with natural objects. (a) Original image with recognition result. (b) Zoom on the RS not recognized because of interference with the presence of yellow leaves. (c) Zoom on the initial pixel labelling around the RS.

and 5.27 (b), the RS is shifted into its BB due to this interference (as compared to where it should be in its BB if there were no interference). In many cases, background interference is a momentary problem, because as the vehicle is moving it utlimately disappears. To illustrate this, Fig. 5.28 shows the next frame to the image in Fig. 5.26 (a), in which the RS is now correctly recognized. Such cases, although considered as faults in the present system, should be resolved by a future tracking module.

Another problem maybe occlusion. Fig. 5.29 (a) shows an example where an RS is partly occluded leading to a false individual recognition (similarly to background interference, occlusion shifts the RS in its BB as compared to if there was no occlusion). In other instances, the RS is too much occluded, a situation which leads to no recognition at all, Fig. 5.29 (b) depicts such an example. In RS scenes, occlusions are most of the time momentary since the vehicle is moving, and again tracking will resolve many of these cases. Fig. 5.29 (c) illustrates this fact for a subsequent frame to the image shown in (a).



(a) (b)

FIGURE 5.28. (a) Next frame to the image shown in Fig. 5.26 (a). The RS that caused interference is now behind the correctly recognized RS. (b) Zoom on the RS.

As final examples that illustrate problems, the next ones have to do with background cleaning and re-labelling. Fig. 5.30 (a) is an instance where background cleaning "etches" a little bit too hard, resulting in an incorrect individual recognition (the class is, however, correctly determined). A zoom on the RS is given in (b). By comparing the initial color labelling in (c) with the background cleaned and re-labelled sub-image in (d), it is seen that background cleaning has penetrated inside the arrow and erased it (admittedly, the border is rather thin and even part of it is incomplete to the right of the arrow; moreover one notices that interlacing is present). Although what is written on the RS is hardly readable, primarily due to interlacing, as (b) shows, one would expect the system to recognize an RS which has a black arrow at its top and which is more similar than the one actually recognized (there are a few such RS's in the database; see Appendix D) even if it is not the exact correct one. Because the arrow is erased, this is not possible. In Fig. 5.30 (e), the RS is not correctly recognized at the individual level. The problem here is that the uniformity assumption at the basis of yellow-black resegmentation does not hold. The left corner of the RS is bent which makes it darker. The rest of the RS is highly uniform (small standard deviation). So during resegmentation, the corner is re-labelled as black. This is clear from (f) and (g), which respectively show the initial pixel labelling and the re-labelling.

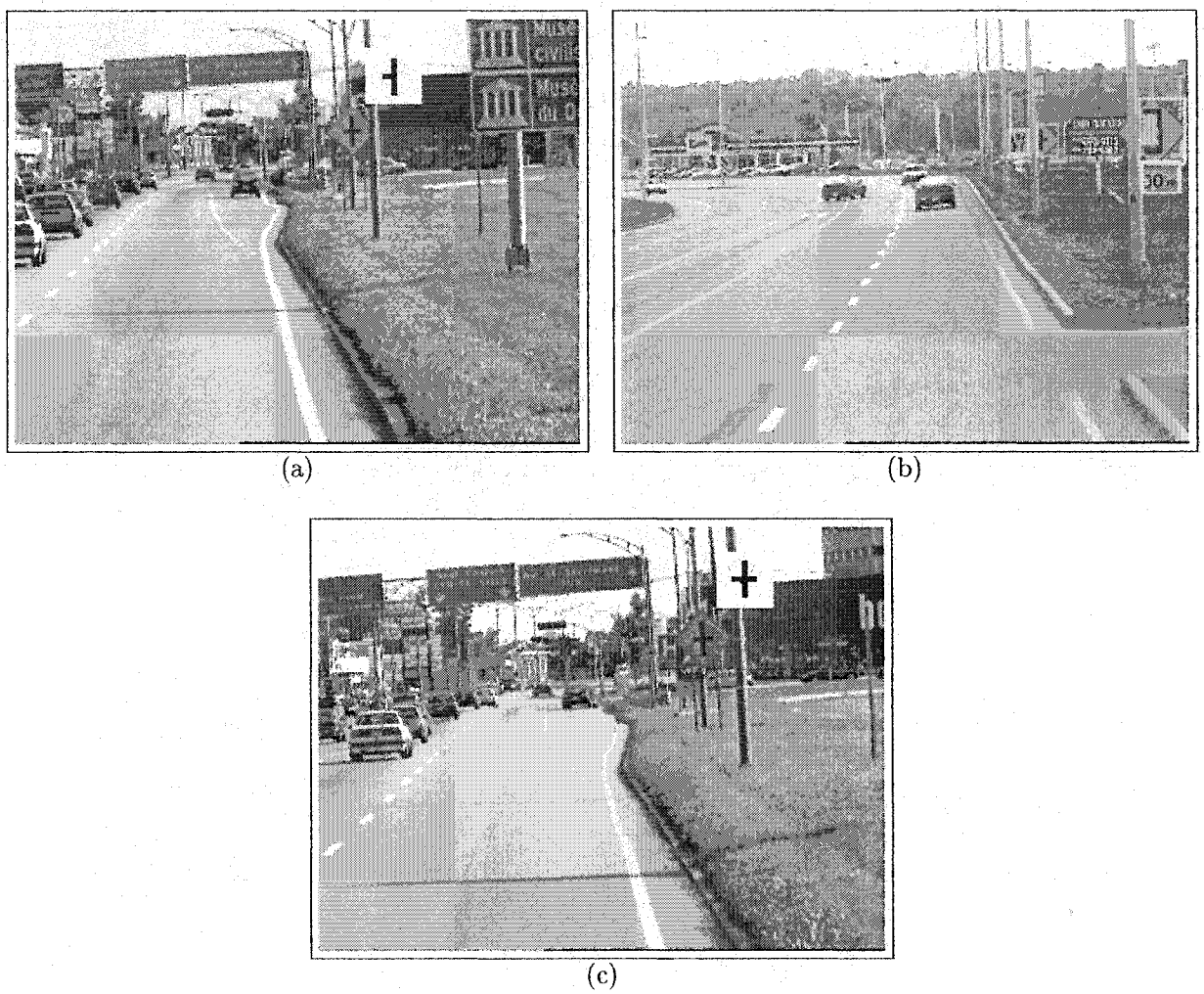


FIGURE 5.29. Occlusion examples. (a) RS partly occluded leading to an incorrect individual recognition result. (b) RS's severely occluded leading to no recognition at all (however, here the system cannot be blamed!) (c) Subsequent frame to the image shown in (a). The RS is now correctly recognized.

Complex examples have already been given in Section 5.5 (Figures 5.21, 5.22, and 5.23). To close this section, Figures 5.31 and 5.32 further illustrate the system's capability for handling complex situations. The reader is referred to the figure captions for the details about these examples. Finally, Fig. 5.33 gives results for images that are part of a sequence. In this latter case, notice how the beacon at the left and the 45km/h sign at the right of the first image get out of sight as the truck from which acquisition is performed progresses,

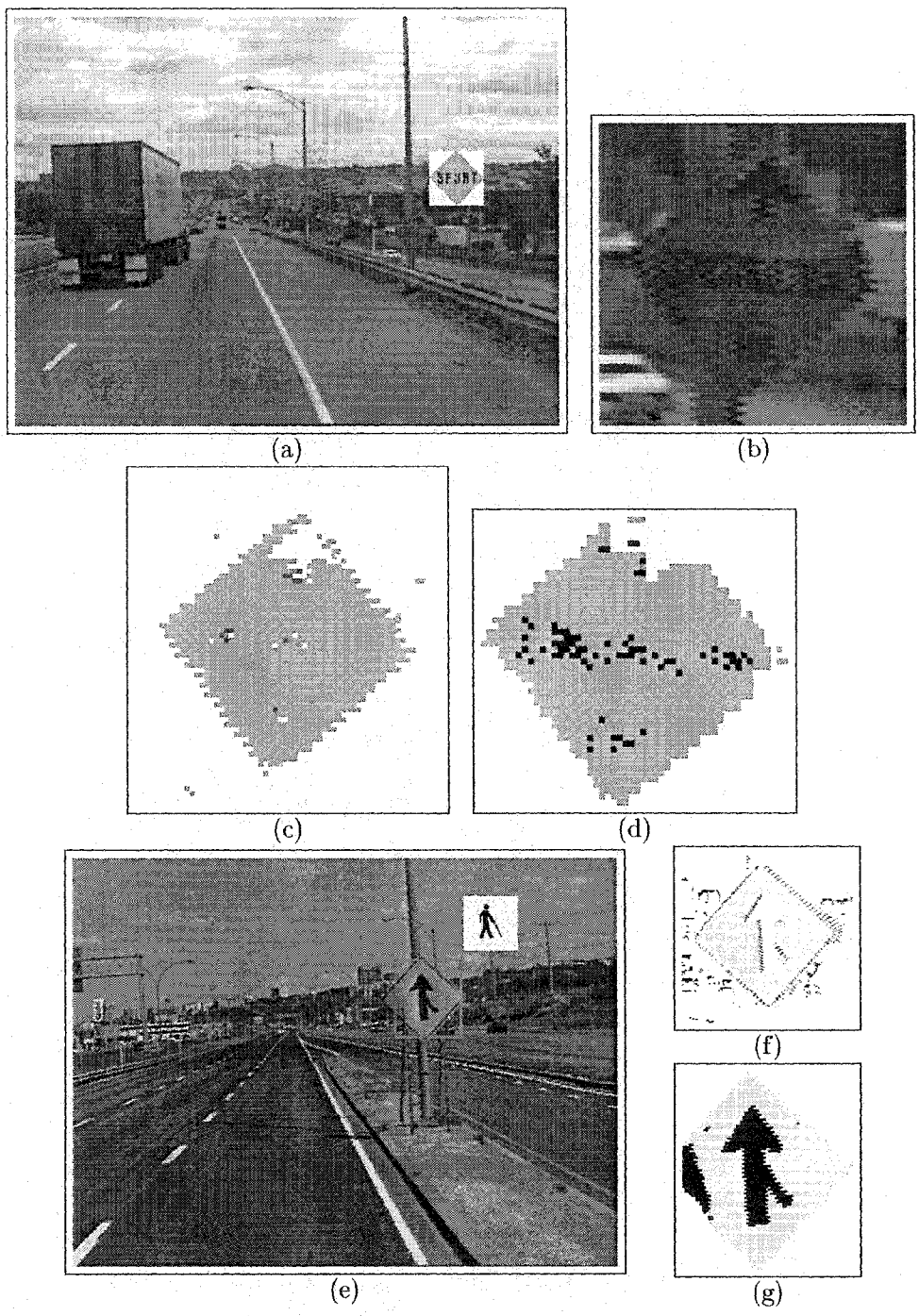



FIGURE 5.30. (a) Original image with individual recognition result. (b) Zoom on the RS. (c) Initial color labelling. (d) Background cleaned and re-labelled image. (e) Original image with individual recognition result. (f) Zoom on the initial color labelling. (g) Re-labelled sub-image with the corner labelled as black.

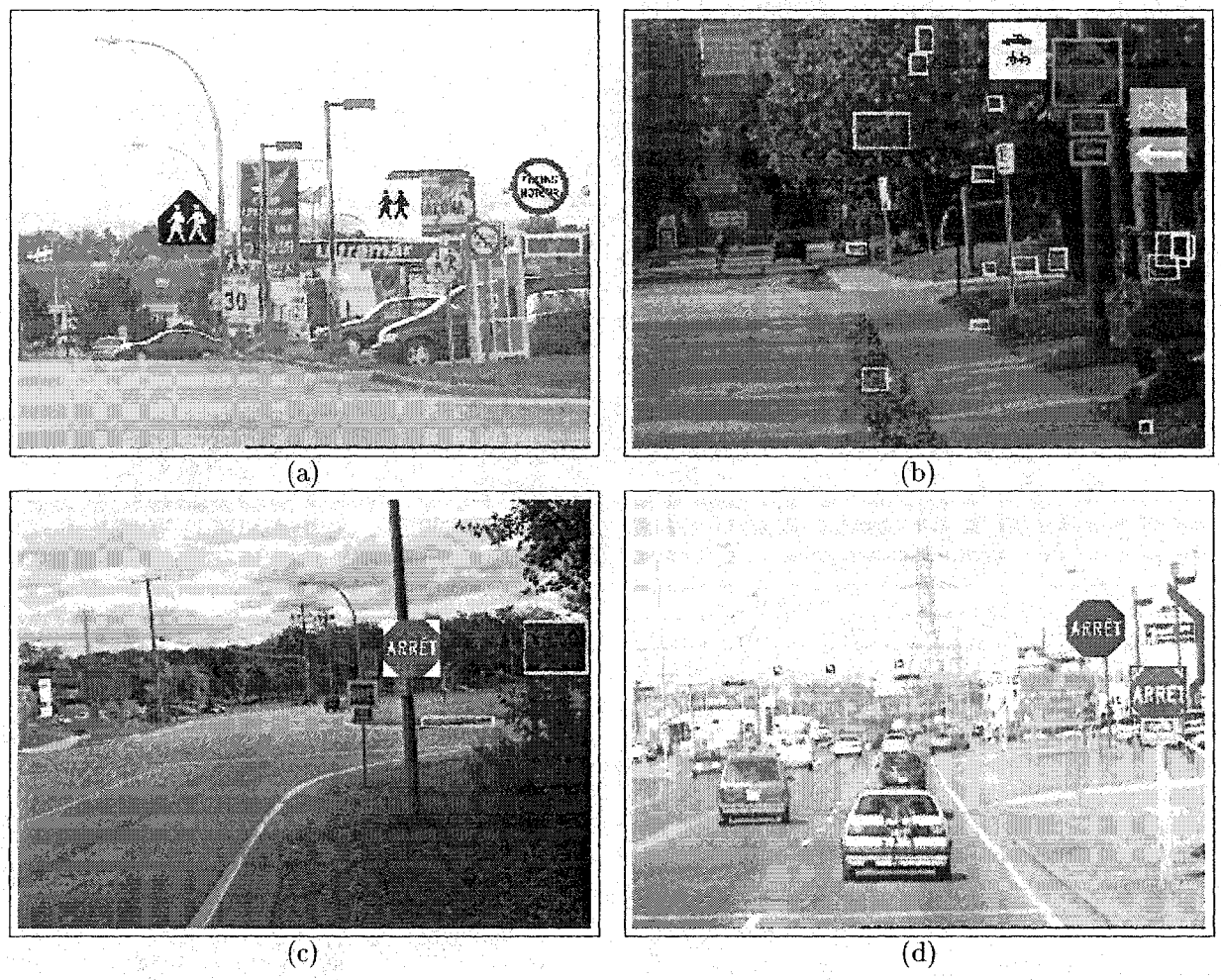



FIGURE 5.31. Images taken in urban scenes. In (a), notice the 30km/h black and white speed limit RS underneath the blue schoolkids RS which is not detected. As said previously, these are not currently processed by the system. (b) Image already considered in Fig. 5.11, but complete recognition results were then not given. (c) Dark image of a stop sign. (d) Image presented earlier in Fig. 3.18.

while the beacon at the center is being considered by the system as it becomes of sufficiently large dimensions.

## 5.7 Conclusion and Further Work

A modular RSR system relying on modelling for both detection and recognition was presented. Color detection is carried using a newly developped physics-based technique that accounts for daylight illumination variations. For recognition, a contribution resides in the

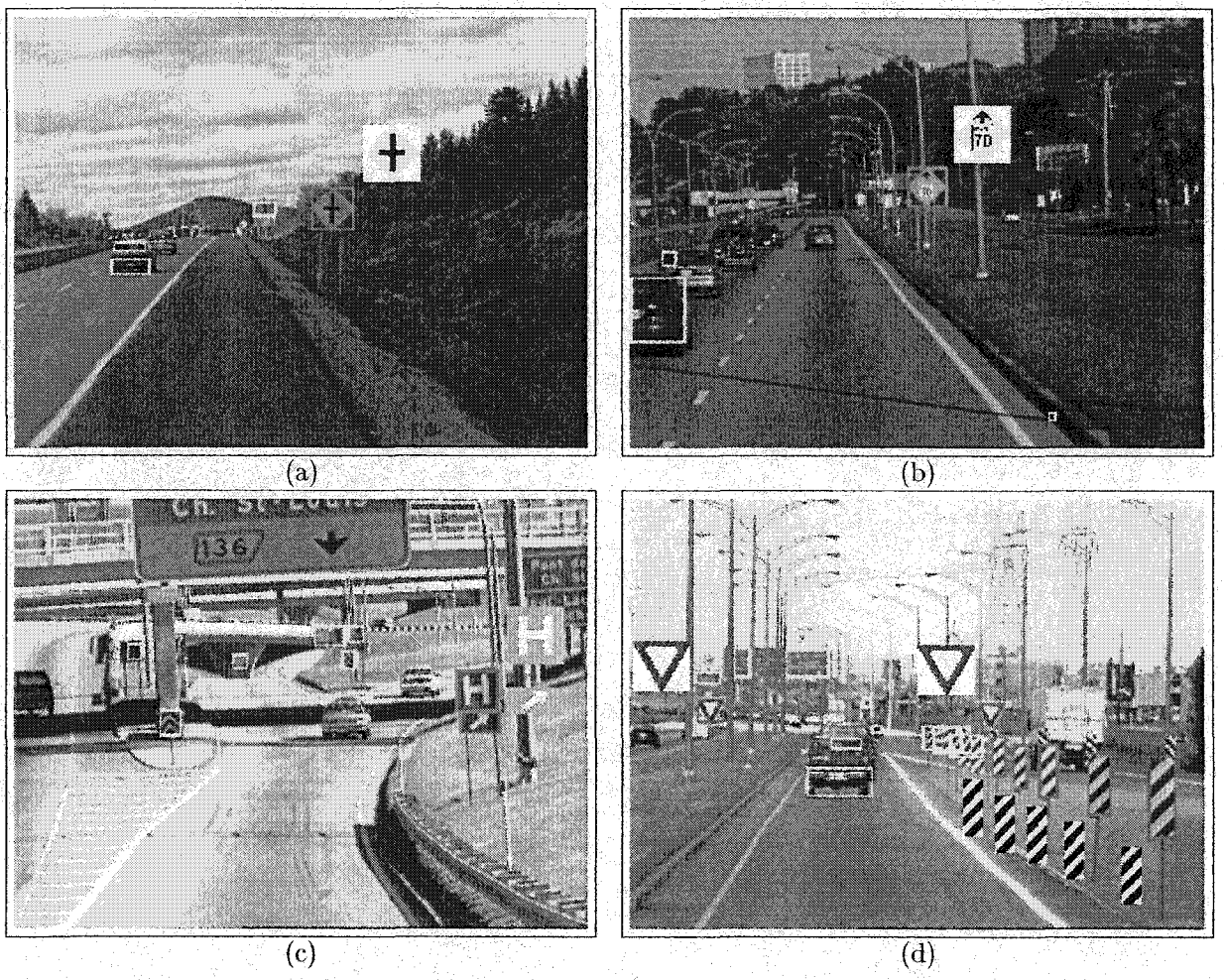



FIGURE 5.32. Images taken along highways. (a) Image captured under cloudy conditions. (b) Image acquired while approaching the end of a highway. (c) Image captured while exiting a highway. (d) Image taken just before lanes will merge.

application of a novel template matching technique that accounts for object color content by way of an orthogonal color encoding, and provides for invariance to geometrical transformations. This technique plays a central role in the recognition procedure. The content of the recognition database, organized in a tree-like structure, is generated automatically without supervision using as input a single model image for each RS and few geometrical parameters. This offers great flexibility in terms of the geometrical imaging conditions and the associated parameter values that can be chosen at will and makes the database easily scalable and adaptable to different road signposting regulations.

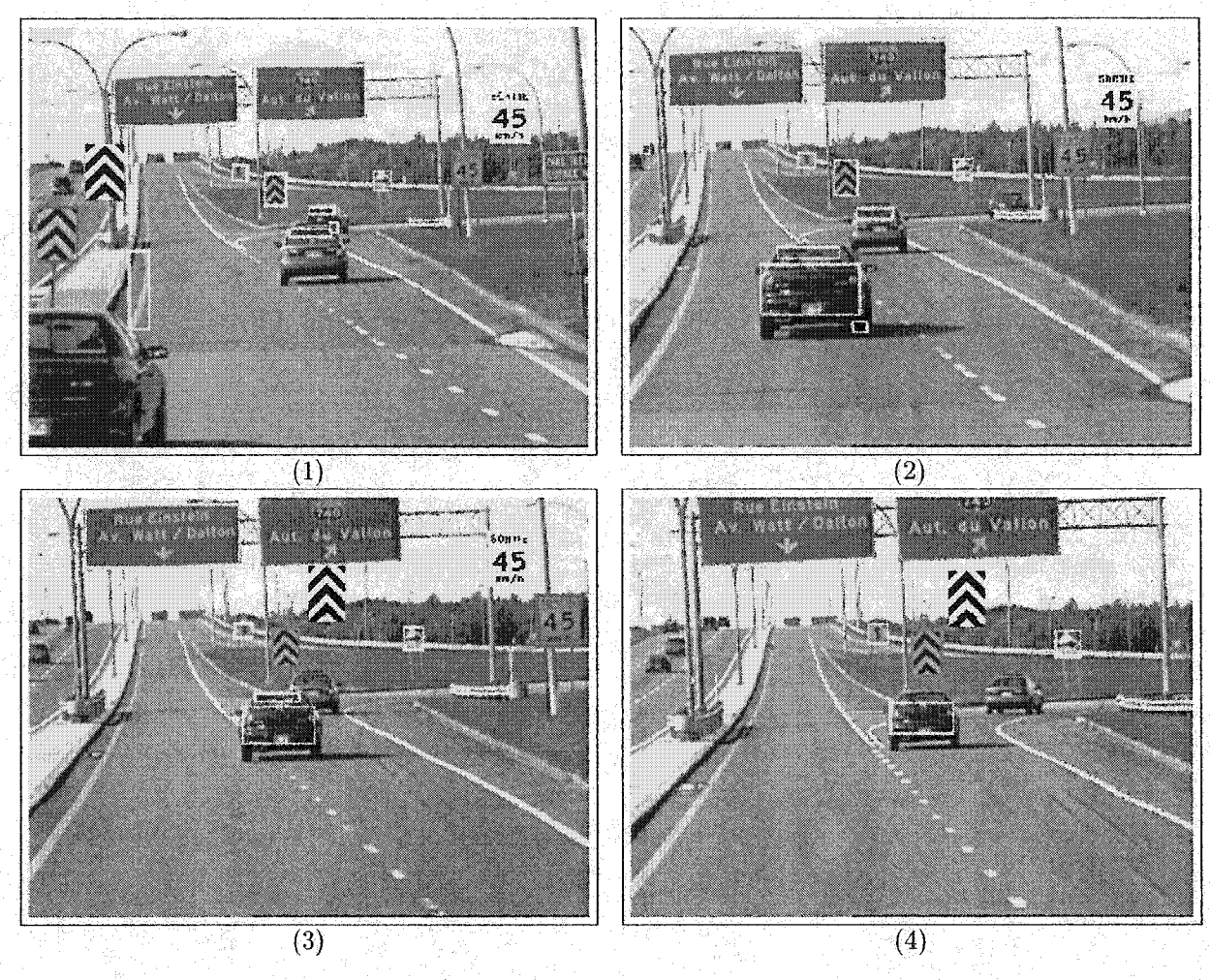



FIGURE 5.33. Recognition results for a sequence of images. In frame (1), the chevron at the center is not sufficiently large to be recognized, but in frame (3) it does become large enough to be recognized. The RS's recognized in frame (1) ultimately run out of sight as the truck progresses.

As regards the recognition algorithm, given an object to be recognized, multiple possibilities are considered at the class recognition level when several classes have correlation scores close to that of the class with highest score. In such situations, this avoids to consider only a single branch of the tree recognition structure, which is the main pitfall when using such a structure.

With these system's foundations laid, further issues have been investigated to improve performances. These had to do with ROI grouping, sub-image re-labelling and background cleaning, and ROI filtering. To perform grouping and re-labelling, techniques exploiting

the information fed back by a partial class recognition were developed. The philosophy behind those techniques is to progressively improve the segmentation of an RS as more information can progressively be gained about it. Finally, global aspects regarding RS scene interpretation needed to be considered to exploit constraints that some RS classes are subjected to.

The system gives high recognition performance figures considering that images are being processed independently, and that a large database of RS's is considered. Tracking in image sequences should bring further improvements, as it will allow to accumulate evidence over frames (temporal integration of information) and exploit the consistency between the lists of possibilities described in Section 5.3.2. The problem of detecting and recognizing black and white RS's has not been addressed in this thesis. At the recognition level, all is there as the database already contains the models for these RS's. It is at the detection level that work needs to be done. This should shortly be in progress as it is of importance.

Until now, the efforts have been put on the quality of the results, rather than on processing speed. The point has now been reached where work on improving this aspect can be undertaken. Some of the structures used in the actual C++ implementation of the system are general-purpose, and were developed with no significant efforts put on optimizing the processing speed as regards the specific application to RSR. Nevertheless, already with the present implementation an average processing rate of more than one image per second is achieved. As computing power continues to increase and optimization issues are investigated, the author is confident that a processing rate of around 3 images per second (twice the actual rate) will be attained in a nearby future.

On a longer time frame, there are plans to incorporate an OCR back-end to the system for interpreting guidance RS's that are not pictographic in nature.

# CHAPTER 6

# Conclusions

## 6.1 Summary

In this thesis, a road sign recognition system has been developed for the processing of single images. This required a certain number of interesting sub-problems to be solved, and the early chapters were devoted to the more fundamental ones (*i.e.*independent of the specific application) having to do with color detection under daylight and its variations, and template matching taking into account labelling information (here color), geometrical deformations, and classes of similar objects. Building on these techniques as basic tools, a road sign recognition system was developed.

The system divides into three main stages of processing. In the first, the color detection technique for daylight illumination is exploited to label the input image, and find regions of interest having road sign colors. Such regions determine where road signs may potentially be located. Sub-images are extracted from these regions to subsequently be more finely re-labelled to account for the color combinations of which road signs are composed. Top-down mechanisms that resort to a partial recognition are applied 1) to recover fragmented regions of interest corresponding to road signs that are not detected as a connected region of identically labelled pixels (this occurs either by accident because of a failure of the system, or because the road sign is inherently composed of multiple connected components), and 2) to re-label the pixels in a region of interest consistently with the partial recognition result. This last step serves to improve the robustness of the full recognition of a road sign carried at the next stage.

The second stage of processing is devoted to recognizing road signs in the regions of interest, if any are present, and to filter out irrelevant regions. Recognition involves a database containing more than 400 road signs arranged in a tree structure. The information contained in this database, and used by the recognition algorithm, is generated automatically by way of geometrical modelling of the image formation process. This algorithm is able to cope with the main pitfall of using a tree recognition database, which is to consider a wrong branch at a given level of the tree, by allowing for multiple possibilities to be considered at each stage during the course of recognition. For each region of interest, the final result of the recognition process is a list containing the identity of possible road signs (hypotheses) arranged in decreasing order of saliency, the first element being the identity of the most probable road sign that the region of interest contains, and with the list being empty when the inner content of the region of interest does not correlate sufficiently strongly with any of the elements in the database.

At the third stage of processing, a scene understanding module exploits constraints on the position of road signs in the image along with the relationships they must have to other road signs in the image in some cases, such as proximity to other specific road signs. This allows to further filter out false positives and avoid unrealistic situations, such as postulating the presence of a road sign at a position where it cannot possibly be located.

The performance of the system has been evaluated and proves to be highly competitive compared to that reported in other work, with the crucial point being that the results given here were attained with the largest database reported thus far in the literature.

The approach exposed here was a comprehensive and rigorous approach, where physics and geometry were used as much as possible to model the situations.

## 6.2 Contributions

The research expounded in this thesis impacts on several aspects of road sign recognition in the field of computer vision. The contributions can be considered to reside at two levels. First are those that relate to basic problems that needed to be solved, which are not necessarily specific to road sign recognition, and which could find use in other applications. Second are the contributions that are more specific to the road sign recognition system as such (although in the latter case, the general principles of the techniques could also be

applied to other problems; as mentioned in the introduction, road sign recognition is a nice example of a typical computer vision problem).

These contributions have been raised in the introductory chapter, but can now be better appreciated, and find further justification in view of the body of work that was presented. The union of all of these contributions has allowed to solve a difficult problem: that of recognizing road signs.

As regards more generic contributions, these are as follows:

- **A novel approach for determining color space regions associated with specific colors under various daylight illumination conditions is presented.** It is based on a physical model of color formation combined with a semi-empirical model of the spectral distribution of the different illumination conditions found outdoors during the day. The approach requires the spectral characteristics of the color camera used to be precisely determined, as well as the spectral reflectance functions of the materials to be detected. This information allows to theoretically predict regions in color space associated with specific objects colors under daylight. These color space regions are then used to automatically find colored examples of the objects of interest in real images from which color data are subsequently extracted. These data are in turn exploited to re-determine the theoretical color space regions originally used, so that the new "calibrated" regions be to a further degree faithfully representative of the color variations actually seen in real images compared to what can be achieved with the theoretical regions.
- **A novel method for template matching, taking labelling as well as different sources of spatial information into account, is introduced.** It is based on a new orthogonal bitwise encoding of the labels, and relies on logical operations performed on the encoding for both embodying spatial information in different forms in a template (geometrical deformations and representing classes of similar objects), and for performing the matching computation. It has the advantage of using all the information carried by the labelling without loss, and treats all labels on equal grounds. It involves simple and natural operations which make it computationally very efficient.

More specifically concerning road sign recognition:

- **The overall road sign recognition system relies heavily on modelling.** This is significantly different from other approaches, which do not consider modelling at all for color detection, and perhaps only to a small extent for recognition (but then in a much different manner compared to here). Overall, the methodology constitutes a comprehensive approach to road sign recognition. More specifically:
  - This approach offers more flexibility, as more can be done without having to go on the roads, and with less human intervention, *e.g.*for collecting and extracting examples needed in color and shape learning, which is a main drawback in learning approaches.
  - As regards road sign colors, the color constancy problem is directly addressed, and their associated color space regions are not determined in an ad-hoc manner in contradistinction with other work, but rather from a model. The color detection approach developed in this work brings something new to color image understanding, since it provides one of the few examples of a fundamentally physics-based technique applied in an outdoors real-world application.
  - For recognition, models built using CAD software are used to generate in an automated and deterministic manner through geometrical modelling the information (templates and features) contained in the recognition database. This represents a definite advantage when it comes to modifying or expanding the database, or eventually considering road signs of a different regulation.
- **The system is endowed with top-down mechanisms exploiting partial recognitions to increase robustness in detection steps carried prior to full recognition.** In these mechanisms, a partial recognition of the class of a road sign provides further information to the task being carried. In this work, the tasks considered were: grouping fragmented road signs, and relabelling colors consistently with the class to which a road sign belongs.

- **The recognition algorithm readily gives an application of the novel template matching approach.** This template matching technique plays a central role in the recognition module as the correlation scores obtained from it are used extensively to search the recognition database.
- **The recognition algorithm is able to account for multiple "nearby" possibilities at the different levels during the search through the database.** At the class recognition level, this allows to avoid the main difficulty with the tree-like recognition structure on which the search is made, which is to take a wrong branch in ambiguous situations. At the individual level, the lists of possibilities generated open the way for exploiting consistency between those lists from subsequent frames in a future tracking module.
- **The road sign recognition system is the first to incorporate a scene understanding module.** This module exploits natural constraints to which road signs are subjected, not artificial ones imposed for reasons of processing reliability or speed.
- **The system is the most complete presented thus far as regards the number and variety of road signs considered.** This is also the first large-scale recognition system as regards North American road signs.

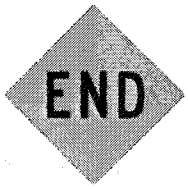
## 6.3 Future Work

The work on color in this thesis has shown on a specific example that a physics-based approach can serve as a useful starting point in a practical application. As mentioned in [**83**], it is usually for such applications that physics-based approaches are criticized due to the limiting assumptions that must be made on the investigated scenes and objects to prevent the number of parameters from becoming prohibitive. Hence, a somewhat contradictory situation is reached where the entities that need to be studied cannot be completely modelled, as would be the ultimate goal. In the present work, the initial physics-based approach to color detection was supplemented by a calibration step to account for model impredictabilities. In a general fashion, the interplay between models and how to use real data to cope with their limitations is a question that should be further investigated, perhaps first in the context of specific applications, and then from a more general standpoint once

enough experience is gained into such matters. This is in line with the general methodology of the present work discussed in the introductory chapter.

From a more practical point of view as regards the present road sign recognition system, some issues which were left aside or barely touched upon readily require further considerations. First, the detection of black and white road signs must obviously be tackled in further work related to this thesis. As was already mentioned, as far as the recognition structure is concerned, all is there in the present system since the database contains models for those road signs, but it is at the detection level that work must be carried out. Also, tracking road signs in image sequences is yet to be performed. As regards processing speed, this is also an area for further work. There is still room for improvements in the data structures and algorithms used in the current implementation. Of course, the ever increasing computing power will also play a role in these matters. Had computing technology remained at the same level as it was when this project was started, the present system would probably not be practical, unless significant efforts would be put in developing dedicated hardware and optimizing the algorithms. This being a long term project, a choice was made right at the beginning on putting the efforts on the quality of the recognition results, with a certain hope that computers would get sufficiently fast. Needless to say that this was a major factor in what is now a viable system.

Lying further on the horizon is the integration of an OCR module for the interpretation of non-pictographic guidance road signs (so-called variable text road signs). Other questions that could be investigated relate to more sophisticated mechanisms that could be used at the different levels of decision in the system. For instance in color detection, instead of the hard boundaries for the color space regions used in the detection scheme developed here, a Bayesian approach could be investigated. Also, at the recognition level as regards the algorithm, heuristic arguments based on thresholds are used. Although these are efficient, there would be a need for more seriously founding these heuristics, or use others, based again on a Bayesian approach, or other paradigms of decision theory.



# APPENDIX A

---

# Color Camera Characterization

## A.1 Introduction

In this appendix, a technique for measuring the spectral sensitivity curves for each channel of a color camera will be presented. These curves are useful in applications requiring precise color measurements, such as in some quality control and detection tasks. The aim is to discuss the basic issues involved in doing such measurements. Other groups [**131, 132, 133**] have also considered color camera characterization using different approaches than the one presented here.

Before describing how the spectral sensitivity curves of a given camera can be obtained, some important facts about color cameras must be discussed. This is the content of Section A.2, which is adapted from work by Novak *et al.* [**99**]. The motivation is that real cameras deviate from idealized camera behavior, and this must be taken into account while obtaining the spectral sensitivity curves. More generally, these facts should be known to anyone making use of a color camera for computer vision purposes, since they have consequences on the quality of color images, and also influence the efficiency of algorithms that rely on ideal camera behavior. Of major interest in this respect is the linearity of the response of a camera to the amount of light that impinges on it. A method for characterizing this linearity is explained and illustrated as well. In Section A.3, a measurement technique of the spectral sensitivity curves is discussed in detail with emphasis on important issues related to the optics required to make the measurements. Finally, some concluding remarks are provided in Section A.4.

## A.2 Facts about Color Cameras

The problems that can occur in real color images caused by non-ideal behavior of color cameras have been thoroughly investigated by Novak, Shafer, and Willson [**99**]. This will be reviewed in some detail, as it is of importance in the sequel.

**A.2.1 Linearity.** Of major importance in applications requiring accurate color measurements is to insure that the camera has a linear response, and if not, to make the appropriate corrections, because the equations modelling color formation in cameras rely on this linearity. Explicitly, these equations have the form

$$C_i = \int E(\lambda) s_i(\lambda) d\lambda, \quad i = R, G, B, \tag{A.2.1}$$

with $C_i$ being the response in channel $i$, $E$ the spectral distribution of irradiance reaching the sensor, and $s_i$ the spectral sensitivity of the combination imaging-optics/color-filter/sensor/-framegrabber in channel $i$ (these equations have already been presented in Section 2.4 of Chapter 2 where $E(\lambda)$ was expressed as the spectral distribution of the illumination times the spectral reflectance of an object). Typically in $RGB$ cameras, the sensitivities are distributed so that one channel detects long visible wavelengths, one medium wavelengths, and one short wavelengths, with some degree of overlap. These are commonly called the red, green, and blue channels respectively. When use of those equations is made, it is thus assumed that as more light enters the camera, the resulting camera measurement will grow proportionately at a given pixel. However, many cameras do not behave in such a linear way. The non-linearities occuring in cameras are described by,

$$C_{i,meas} = C_i^{1/\gamma_i}, \tag{A.2.2}$$

with $C_i$ being the value that would be expected from Eq. (A.2.1), and $C_{i,meas}$ the value that is actually output. To correct for this, $C_{i,meas}$ has to be raised to the power $\gamma_i$, thus requiring $\gamma_i$ to be known.

The response of a camera with respect to the entering light intensity can be measured with gray calibration standards consisting of several gray samples made from materials of nearly constant reflectances over the visible spectrum. Ideally, when no non-linearity is present, values measured by the camera viewing these gray patches under evenly distributed

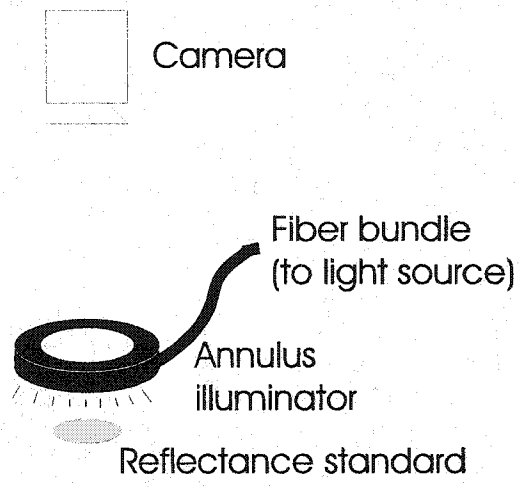



FIGURE A.1. Schematic of the set-up for characterizing the linearity of the camera. The annular light source is a fiber optic light source giving very uniform lighting (Banner, Minneapolis, U.S.A.).

illumination should lie on a straight line when plotted against percentage reflectance. If it is not the case, then $\gamma_i$ can be estimated by using a log-log plot. An experimental set-up for performing such measurements is depicted in Fig. A.1. In such measurements, great care must be used in evenly illuminating the calibration standards. It must also be ensured that no light other than that reflected by the calibration standard reaches the camera.

Results of measurements performed to check the linearity of a Panasonic GP-US502 3-CCD micro head color video camera will now be described[1]. Reflectance calibration standards from Labsphere Inc. with nominal percent reflectances of 2%, 10%, 20%, 40%, 60%, 80%, and 99% were employed. With the standards evenly illuminated, the camera responses are easily related to the percent reflectance. The light reaching the sensor is first expressed as the light from the source which is reflected by the standard

$$E(\lambda) = R(\lambda)L(\lambda), \tag{A.2.3}$$

with $L(\lambda)$ being the light from the source and $R(\lambda)$ the spectral reflectance of the standard. Combining this with Eqs. A.2.1 and A.2.2 gives

$$C_{i,meas} = \left( \int R(\lambda)L(\lambda)s_i(\lambda)d\lambda \right)^{1/\gamma_i}, \quad i = R, G, B. \tag{A.2.4}$$

[1]This camera is used throughout in the following.

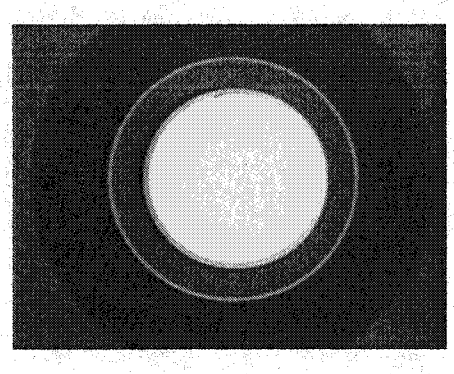


FIGURE A.2. Image of the reflectance standard (white disk at the center) having an 80% reflectance as seen by the red channel of the camera. The data point corresponding to this reflectance standard used in constructing the curve of Fig. A.3 (a) is obtained by averaging the values of all the pixels located on the standard in the image. The darker annulus around the standard is the light source seen from the back, see Fig. A.1.

These equations are the color formation equations with gamma correction included. Taking into account that the spectral reflectance of a standard is constant over the visible spectrum, it can be pulled out of the integral, giving rise to

$$C_{i,meas} = a_i \cdot (\%reflectance)^{1/\gamma_i}, \tag{A.2.5}$$

with $a_i = \left(\int L(\lambda) s_i(\lambda) d\lambda\right)^{1/\gamma_i}$. Obviously, the light source must be set to a constant value throughout the measurements on the different reflectance standards. Taking the logarithm on both sides results in

$$\log C_{i,meas} = \frac{1}{\gamma_i} \log(\%reflectance) + \log a_i. \tag{A.2.6}$$

The slopes and the ordinates at the origin for the different channels were estimated by least-square fitting a line to the data in the log-log plot. For a given reflectance standard, the data point has been obtained by averaging the values of the pixels over the region where the standard is located in the image, see Fig. A.2. The following values were obtained: $\gamma_R = 2.09$, $\gamma_G = 2.06$, and $\gamma_B = 2.03$. Fig. A.3 illustrates the data points and the fitted curves.

**A.2.2 Sensitivity to Non-Visible Wavelengths.** Another problem that may occur is that the camera is sensitive to wavelengths out of the visible range. In the ultraviolet (UV), this is not a problem because silicon is negligibly sensitive. Moreover, glass, out of which most lenses are made, is not transparent to such wavelengths. However, silicon is

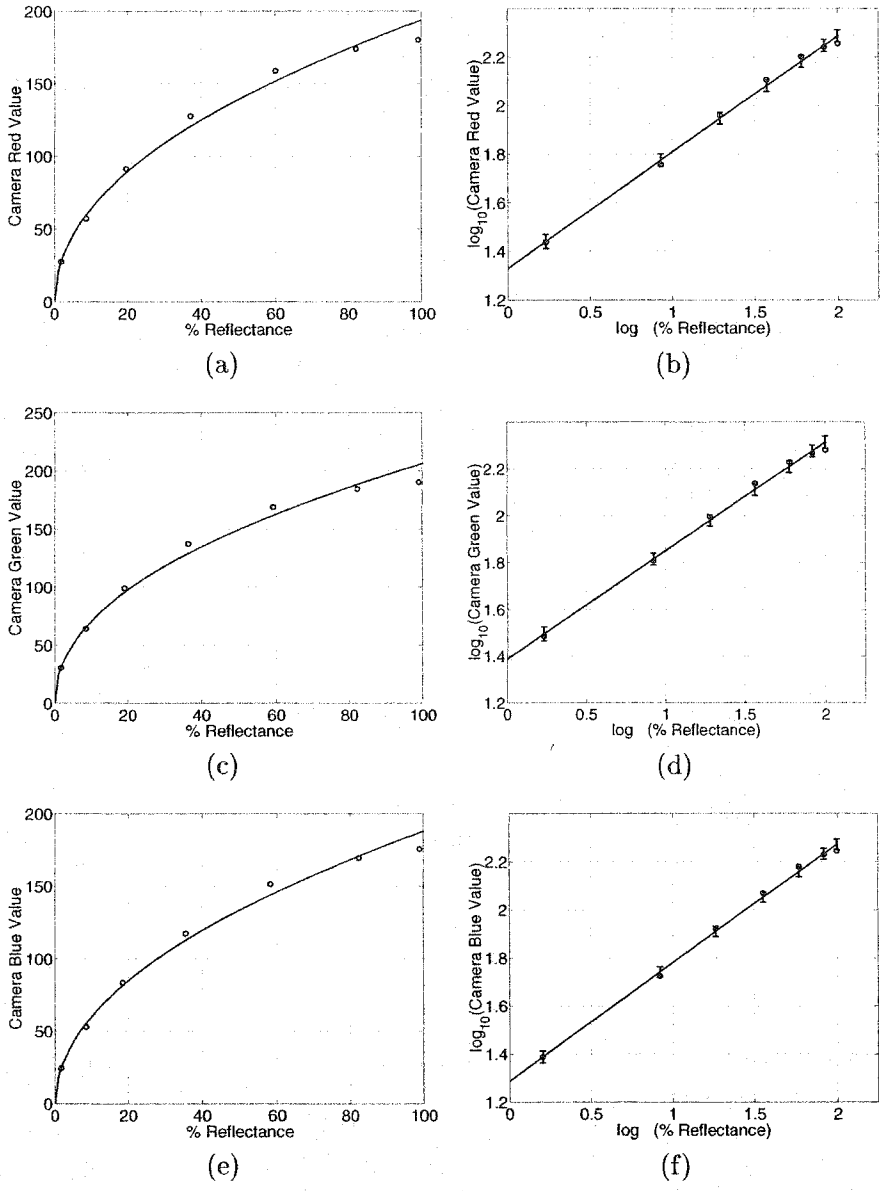



FIGURE A.3. Plots for obtaining the gamma parameters of the camera. Circles in (a), (c), and (e) show the measured data points for the red, green, and blue channels respectively. The same points are shown in (b), (d), and (f) on a log-log plot, where the lines are least-square fits to the data points, *c.f.* Eq. (A.2.6), with error bars showing the estimated errors of the fit at the data points. The data points are within the error bounds. Using the parameters $a_i$ and $\gamma_i$ in Eq. (A.2.5) obtained from the fit, the curves approximating the data points in (a), (c), and (e) are plotted on these same graphs (continuous curves).

sensitive to infrared (IR) radiation. The peak sensitivity of silicon detectors is typically reached at around 750 $nm$. Thus, depending on the color filters used in a given color camera, it may be necessary to use an IR blocking filter. In the case of the camera used here, the responses of the channels to IR radiation was found to be negligible, see Fig. A.7.

**A.2.3 Color Imbalance.** A further problem is that of color imbalance in which colors are shifted towards certain portions of color space. The cause can be that the camera is more sensitive to certain ranges of wavelengths, and/or the light source is emitting more strongly in certain ranges of wavelengths. In both cases, the solution is to change the sensitivity of the camera to the different portions of the spectrum. This is usually done by adjusting the gains of the color channels of the camera using the knobs on the camera unit. In doing so, however, it should be kept in mind that while the signal is boosted, the noise is also amplified, resulting in no improvement in the signal-to-noise ratio (SNR)[2]. This may be important in certain applications, but with the quality of cameras available today it should not be of concern in usual applications. Fig. A.7 (b) of the next section shows the spectral sensitivities of the camera for the three color bands, when the factory settings are used. The maximum sensitivity of the blue band is significantly higher than that of the other two. In those settings, the manufacturer has probably compensated for the fact that the incoming light is generally weaker in the blue end of the spectrum for usual indoor artificial light sources. However, depending on the illuminant, those settings might not be appropriate since whites may not be well centered in color space, and colors may tend to be displaced towards blue. Fig. A.4 shows an instance of this for daylight (the colors are illustrated in the $(p, q)$ coordinate system defined in Section 2.4.4), which may have a strong component in the blue end of the spectrum depending on the conditions, see Fig. 2.5. Such drifting due to imbalance can lead to difficulties when colors must be discriminated, because colors then tend to get more closely packed in certain portions of color space. In a given application, the colors of interest should be as uniformly distributed as possible in the color space considered.

[2]See [**83, 99**] for an interesting, but perhaps somewhat outdated approach for dealing with color imbalance, since a monochrome camera with a color filter wheel is used. Moreover, this method is impractical in view of real-time applications

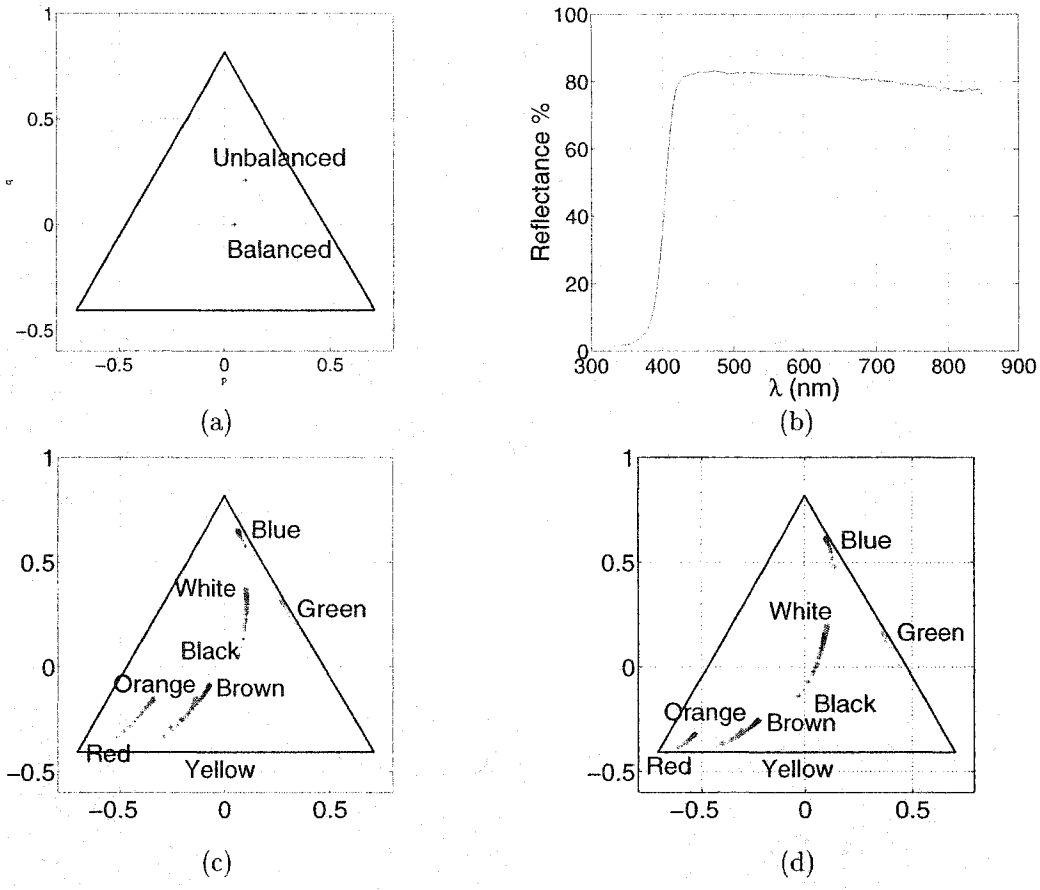



FIGURE A.4. Color balancing versus no balancing. When balancing is performed, whites should be close to the center. In the coordinates used here, a perfect white, (*i.e.* with a perfectly flat spectral reflectance), should ideally be centered at $(0,0)$. In (a), a white having the spectral reflectance plotted in (b) is shown with $D_{65}$ (daylight with correlated color temperature of $6500K$) as the illuminant and with a camera having the balanced and unbalanced (factory settings) spectral sensitivities illustrated in Fig. A.7. In (c) and (d), the effect of color balancing is illustrated; (c) shows the distribution of a few colored samples illuminated by different phases of daylight when no balancing is done, the colors are shifted towards the blue corner of the color triangle. In (d) balancing has been performed.

**A.2.4 Clipping, Blooming, and Chromatic Aberration.** In [**99**] other problems are mentioned: clipping, blooming, and chromatic aberration. Clipping is related to the finite dynamic range of real cameras, and the problem can be avoided by adjusting the aperture, so that the camera does not saturate. Other than that, not much can be done. Blooming, as well, can be prevented in this manner. Concerning chromatic aberration, in common applications it is generally not critical, and can be brought to an acceptable level

by resorting to good quality optics. When critical, the method of [**99**], which consists of correcting chromatic aberration by mechanically adjusting the lens geometry and camera position seems somewhat outdated, and can definitely not be used in real-time applications. The only solution is to invest in high quality compensated optics.

## A.3 Spectral Sensitivity Curves

In this section, a method for determining the spectral sensitivity curves of a color camera will be presented.

**A.3.1 Method.** To obtain the spectral sensitivity curves, monochromatic light has to be presented to the camera. This can be done with a monochromator, an instrument that disperses an incident flux of radiation into its spectrum, and from which any narrow band of wavelengths can be isolated with a slit aperture. Details about monochromators are given in [**77**]. The width of the band of wavelengths (or spectral bandwidth), which is a measure of the monochromaticity of the light exiting the monochromator, depends on the focal length $L$ of the monochromator, the width $w$ of the entrance and exit slits[3], and the pitch $p$ of the diffraction grating (in $lines/mm$), in the following manner,

$$\Delta\lambda = \frac{w}{p \cdot L}. \tag{A.3.1}$$

Because of this finite width, it will thus be more accurate to talk in terms of quasi-monochromatic signals in the sequel. For the monochromator used here, the values are $L = 500\,mm$, $p = 1200\ lines/mm$. In the measurements to be presented later, the width of the slits was set to $3\,mm$, resulting in a bandwidth of $5\,nm$. This is not narrow for an optical bandwidth, and much better can be achieved (on the order of tenths of nanometers), but this is sufficient for the purpose of characterizing a camera for most computer vision applications. Moreover, there is a trade-off between monochromaticity and light intensity. Ideally one would like to have a bandwidth $\Delta\lambda$ as small as possible, for greater spectral purity, but then less light is available at the exit of the monochromator, because the slits are narrower. In the present case, enough light intensity was required to exploit the dynamic range of the camera as much as possible, and so the maximum slit width was used.

[3]Those widths should always be the same, because the optics within the monochromator images the entrance slit onto the exit slit.

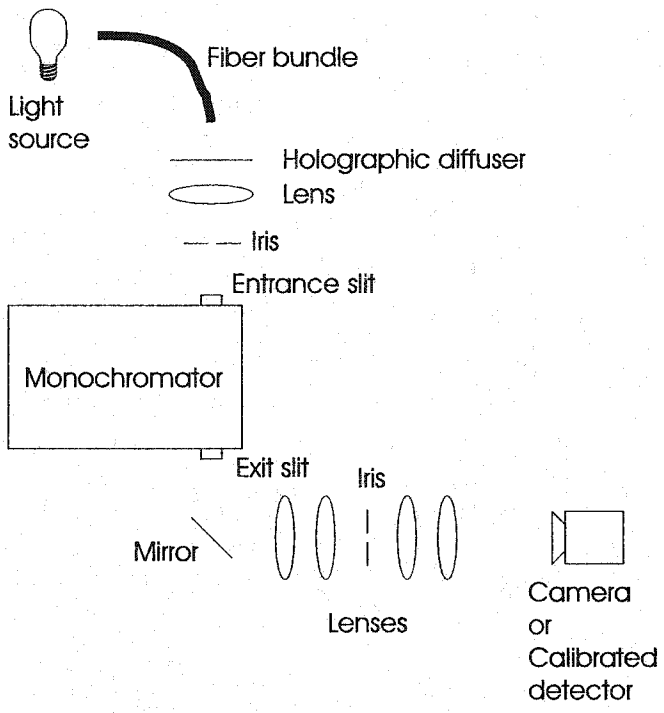



FIGURE A.5. Schematic of the set-up for measuring the spectral sensitivities of the camera using the substitution method. Light from the source is coupled through an incoherent fibre bundle and a holographic diffuser in order that the light be as uniform as possible, see text. In the exit arm of the set-up, lenses are used in combination to compensate for chromatic aberration. Had the direct method been used, a beam splitter would have been placed in front of the camera, and the calibrated optical power detector would be located in the other path of light generated by the beam splitter.

To properly image the light emerging from the exit slit onto the detector, and to measure in which amount light reaches it, an optical set-up must be used. This is schematically illustrated in Fig. A.5.

For each narrow band of wavelengths, an image of the slit is grabbed (see Fig. A.6 (a) for an example of a stimulus centered at 700 $nm$), and the amount of power in the stimulus reaching the camera is measured using an optical power meter. This last measurement is necessary in order to refer the $R, G, B$ responses of the camera to the amount of power contained in the stimulus which is not the same for all wavelengths. The measurement of the power can be done while the image of the slit is being grabbed, in which case a beam splitter must be used to send part of the beam to a calibrated optical power detector, and the other part to the camera. This is an example of a so-called direct method. It has the

drawback that not all the light exiting the monochromator is sent to the camera. Since this amount is already small for a reasonable light source (200 Watts), the full dynamic range of the camera cannot be covered in the measurements, leading to SNRs that can definitely be improved. To alleviate this, one must use more poweful light sources, which require substantial cooling, and tend to wear out rapidly. Moreover, such light sources tend to have large temporal fluctuations, making the measurements less accurate. The fact that only small amounts of light are actually available at the exit of the monochromator is also due to the need of having a very uniform patch of light at its output as explained below. If no care is used, the problem encountered is that the filament of the lamp is imaged onto the camera, making the patch seen by the camera highly non-uniform. One way to avoid this is to ensure that the light fed to the monochromator is uniform. This can be done with the use of an incoherent illumination fibre bundle and a diffuser for coupling the light from the source to the monochromator. The price paid is a reduced amount of source light that can be converted into quasi-monochromatic light.

Instead of a direct method, substitution can be used. Here the quasi-monochromatic stimulus is sent onto the camera, and an image is acquired. The camera is then replaced by a calibrated optical detector to measure the power contained in the stimulus. This requires the source to be stable in time since the image of the stimulus and the measurement of its power are not made a the same time. This is easily achieved for not too intense sources powered with a stabilized current source. The advantage of this method is that all the light available at the output of the monochromator is sent onto the sensor that must be characterized. The disadvantage is that the source may have fluctuated between the image acquisition and the measurement of the power contained into the patch. Averaging can be used to minimize such fluctuations. In what follows, the substitution method with averaging is the one that has been employed.

Once the image of the patch at a given wavelength has been captured, and the power measured, the data must be processed to extract the sensitivities at that wavelength. Since the patch covers a region of pixels, a spatial average is first computed. The region over which this average is computed is chosen to be the same *for all* wavelengths. In this manner, it is always the same pixels that are considered. It is thus the average sensitivities over these pixels that will be obtained in the end. To determine this region, three binary masks are

generated by thresholding the blue, green and red images of stimuli centered at 480, 565, and 700 $nm$ respectively, wavelengths at which each channel is the most sensitive for the camera considered here[4]. The masks are ANDed to get the final mask. Refer to Fig. A.6 for images of the stimuli and masks. For the measurements to be meaningful, great care must be taken in making the intensity of the patch of quasi-monochromatic light as uniform as possible. How to do this was discussed above. This prevents the responses of the pixels within the patch from being too scattered. Looking at the masks for the red, green, and blue channels in Fig. A.6, it is seen that not all pixels in the patch are present. Some were eliminated because they had values too far from those of the rest of the pixels. This further reduces the scattering of the responses, and is easily determined by using a histogram.

Let $C_{avg,i}(\lambda_0)$ denote the spatial average in channel $i$ at wavelength $\lambda_0$. In terms of spectral integration, this is equal to

$$C_{avg,i}(\lambda_0) = \left( \int L_{\lambda_0}(\lambda) s_i(\lambda) d\lambda \right)^{1/\gamma_i}. \tag{A.3.2}$$

Here, $s_i(\lambda)$ must be considered as the average of the individual spectral sensitivities of the pixels partaking in the spatial average. Assuming the CCD arrays and color filters used in the manufacturing of color cameras to be uniform, which is reasonable with today's manufacturing capabilities, these averages should be an excellent estimate of the spectral sensitivities for each pixel. $L_{\lambda_0}(\lambda)$ represents the quasi-monochromatic stimulus centered at $\lambda_0$. In first approximation it can be set equal to a delta function with a multiplicative factor proportional to the measured power contained in the stimulus,

$$L_{\lambda_0}(\lambda) = b \ power(\lambda_0) \ \delta(\lambda - \lambda_0). \tag{A.3.3}$$

The delta function approximation gets better as the width of the quasi-monochromatic signal gets narrower. Combining the previous equations gives

$$b\, s_i(\lambda_0) = \frac{C_{avg,i}^{\gamma_i}}{power(\lambda_0)}. \tag{A.3.4}$$

The spectral sensitivities are thus obtained within a constant factor which is the same for the three channels and for all wavelengths. In other words relative values are obtained.

[4]For other types of cameras these wavelengths may be different, and must be determined.

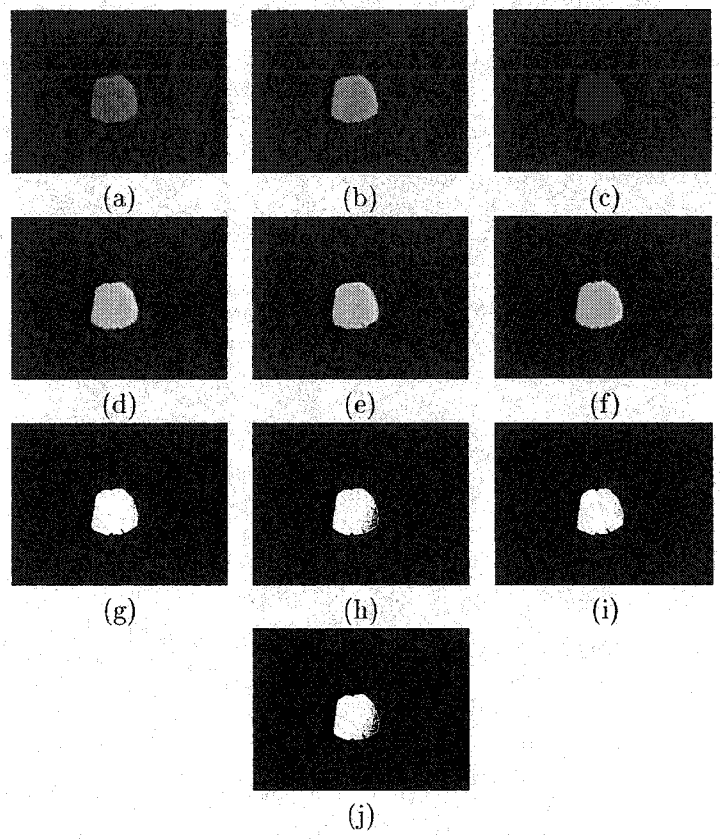



FIGURE A.6. Illustrating the image of the exit slit onto the detector and the binary masks generated to measure the spectral sensitivity curves of the camera. (a), (b), and (c) show quasi-monochromatic stimuli centered at 700 $nm$, 565 $nm$, and 480 $nm$ respectively. (d), (e), and (f) are respectively: the image of the red channel of (a), the image of the green channel of (b), and the image of the blue channel of (c). These images are thresholded to generate the binary masks in (g), (h), and (i). The three masks are ANDed to generate the final mask in (j).

**A.3.2 Results.** Using the method just described, the spectral sensitivity curves of the camera were obtained. Fig. A.7 depicts the results for the camera equipped with an objective. It is important to characterize the camera with the objective that will be used in the application, as well as for the channel gains at which it will be set. These are some of the reasons why it is useful to do the measurements oneself, because manufacturers cannot provide the spectral sensitivity curves for every camera objective on the market, and for everyone's settings. All measurements should be made in complete darkness, and with the automatic gain control (AGC) of the camera disabled, in order for the camera not to adjust the gains itself. Of course, when the camera is actually used, the AGC must also be off

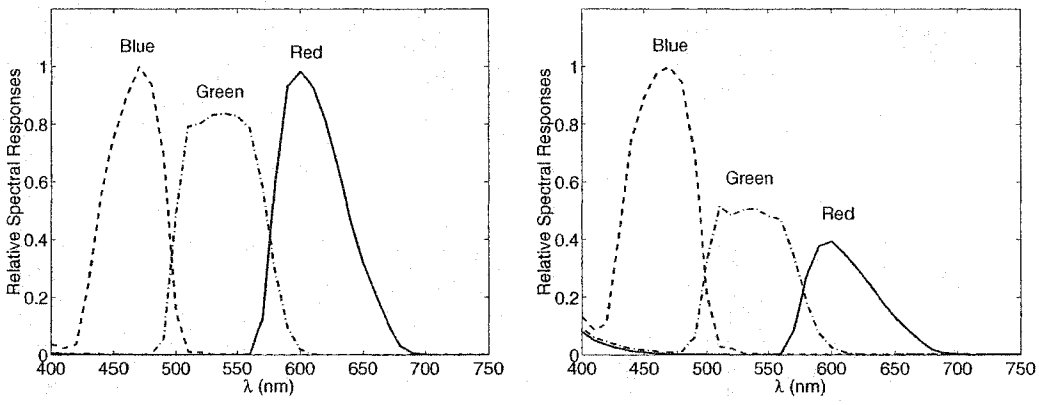



FIGURE A.7. Spectral sensitivity curves for the camera characterized with an objective. In (a) color balancing has been performed, whereas this is not the case in (b) where the curves are those of the factory settings.

for the same reasons. In reality, what is so characterized is the combined effect of sensor, filters, gain adjustments, optics and frame grabber. For computer vision applications, this is what is necessary.

**A.3.3 Summary of the Method.** In summary, when using the substitution method, the sequence of steps to obtain the spectral sensitivity curves is as follows (in the direct method, steps 2 and 3 reduce to a single one):

(i) Put the camera in place, and determine the binary mask over which spatial averaging takes place.

(ii) Scan the visible spectrum from say 330 to $800\,nm$ in steps of $10\,nm$, with the camera in place.

(iii) Take out the camera, replace it by the calibrated optical detector, and scan the visible spectrum again to obtain the power measurements.

(iv) For each wavelength, perform the spatial averaging using the masks previously determined, and use Eq. A.3.4 to obtain the sensitivities.

Steps 2 and 3 can be repeated and the results averaged to smooth out fluctuations, as mentioned before.

## A.4 Conclusion

A method for measuring the spectral sensitivity curves of a color camera was presented. Practical aspects were discussed, with some emphasis on fundamental issues related to non-ideal camera behavior, and the properties of the quasi-monochromatic light that should be sent to the camera.

# APPENDIX B

## Daylight Model Details

This Appendix presents some further details concerning the daylight model of Judd, MacAdam, and Wyszecki, and the correlated color temperature.

### B.1 A Brief History of the CIE Daylight Model

The sun is the most important source of radiant energy on earth. Considerable efforts have been devoted to characterize daylight illumination theoretically [**91**], and several spectroradiometric measurements of daylight have been reported [**87, 92, 89**].

As well as being of importance in color science, spectroradiometric data about daylight is used in astrophysics, materials research, architecture, photobiology, illumination engineering, solar energy, space research, remote sensing, and many other fields. For instance, one of the goals of illumination engineering is to develop sources of light having relative spectral power distributions as close as possible to that of natural daylight such as found under cloudy conditions. In photography such data is important for judging the color rendering quality of photographs.

The body of work on daylight has led the Commission Internationale de l'Éclairage (CIE) to recommend a method for calculating relative spectral radiant power distributions that are representative of the typical phases of daylight at the earth's surface. The CIE method is based on the model of Judd, MacAdam and Wyszecki [**79**] of which the relevant facts to the present work were presented in Chapter 2. From the work of Judd *et al.*, illumination standards, the so-called D-illuminants $D_{55}$, $D_{65}$ and $D_{75}$, have also been developed.

To build their model, Judd *et al.* [**79**] have developed a semi-empirical method for reconstituting spectral power distributions representative of the different phases of daylight. In 1964, it was recommended by the CIE, and has become to be known as the CIE Method of Calculating D-Illuminants. It is based on 622 spectral radiant power distributions measured by Budde [**86**], Henderson and Hodgkiss [**89**], and Condit and Grum [**87**]. Judd *et al.* have subjected those experimental measurements to a characteristic vector analysis (principal components analysis), and it was found that the measured relative spectral power distributions could be very well reproduced by appropriate linear combinations of the calculated mean spectral distribution $S_0(\lambda)$ and spectral distributions $S_1(\lambda)$ and $S_2(\lambda)$ of the two most important eigenvectors of the set of the 622 measured distributions. These vectors were shown in Fig. 2.4, and their numerical values are given in Table B.1.

## B.2 Connection Between the Model and Real Illumination Conditions

In the model of Judd *et al.*, the notion of correlated color temperature plays an important role. It is defined as follows: The correlated color temperature $T_c$ of a source, or selective radiator, is defined as the temperature of the blackbody radiator whose perceived color most closely resembles that of the given selective radiator at the same brightness, and under specific viewing conditions[1]. To give an idea of the correlated color temperatures involved with daylight, here are some broad relationships (taken from [**77**] and [**92**]):

(a) Daylight from the sun plus the total sky incident on a horizontal surface ranges in correlated color temperature from about $5000K$ to $7000K$ regardless of cloud coverage, *i.e.* clear to overcast sky.

(b) Daylight from the north sky (that is the part of the sky in the north direction where the sun never appears), but with the sun occluded, has correlated color temperatures above $7000K$ . The amount of cloud coverage determines the correlated color temperature most applicable. For a clear north sky, very high correlated color temperatures of $40000K$ and higher can be reached. As cloud coverage increases, the correlated color temperature diminishes, to the point that the overcast sky falls into catergory (a).

[1]Here by "perceived" is meant as would be perceived by the CIE 1931 standard observer.

| $\lambda$ (nm) | $S_0$ | $S_1$ | $S_2$ | $\lambda$ (nm) | $S_0$ | $S_1$ | $S_2$ |
|---|---|---|---|---|---|---|---|
| 300 | 0.04 | 0.02 | 0.0 | 570 | 96.0 | -1.6 | 0.2 |
| 310 | 6.0 | 4.5 | 2.0 | 580 | 95.1 | -3.5 | 0.5 |
| 320 | 29.6 | 22.4 | 4.0 | 590 | 89.1 | -3.5 | 2.1 |
| 330 | 55.3 | 42.0 | 8.5 | 600 | 90.5 | -5.8 | 3.2 |
| 340 | 57.3 | 40.6 | 7.8 | 610 | 90.3 | -7.2 | 4.1 |
| 350 | 61.8 | 41.6 | 6.7 | 620 | 88.4 | -8.6 | 4.7 |
| 360 | 61.5 | 38.0 | 5.3 | 630 | 84.0 | -9.5 | 5.1 |
| 370 | 68.8 | 42.4 | 6.1 | 640 | 85.1 | -10.9 | 6.7 |
| 380 | 63.4 | 38.5 | 3.0 | 650 | 81.9 | -10.7 | 7.3 |
| 390 | 65.8 | 35.0 | 1.2 | 660 | 82.6 | -12.0 | 8.6 |
| 400 | 94.8 | 43.4 | -1.1 | 670 | 84.9 | -14.0 | 9.8 |
| 410 | 104.8 | 46.3 | -0.5 | 680 | 81.3 | -13.6 | 10.2 |
| 420 | 105.9 | 43.9 | -0.7 | 690 | 71.9 | -12.0 | 8.3 |
| 430 | 96.8 | 37.1 | -1.2 | 700 | 74.3 | -13.3 | 9.6 |
| 440 | 113.9 | 36.7 | -2.6 | 710 | 76.4 | -12.9 | 8.5 |
| 450 | 125.6 | 35.9 | -2.9 | 720 | 63.3 | -10.6 | 7.0 |
| 460 | 125.5 | 32.6 | -2.8 | 730 | 71.7 | -11.6 | 7.6 |
| 470 | 121.3 | 27.9 | -2.6 | 740 | 77.0 | -12.2 | 8.0 |
| 480 | 121.3 | 24.3 | -2.6 | 750 | 65.2 | -10.2 | 6.7 |
| 490 | 113.5 | 20.1 | -1.8 | 760 | 47.7 | -7.8 | 5.2 |
| 500 | 113.1 | 16.2 | -1.5 | 770 | 68.6 | -11.2 | 7.4 |
| 510 | 110.8 | 13.2 | -1.3 | 780 | 65.0 | -10.4 | 6.8 |
| 520 | 106.5 | 8.6 | -1.2 | 790 | 66.0 | -10.6 | 7.0 |
| 530 | 108.8 | 6.1 | -1.0 | 800 | 61.0 | -9.7 | 6.4 |
| 540 | 105.3 | 4.2 | -0.5 | 810 | 53.3 | -8.3 | 5.5 |
| 550 | 104.4 | 1.9 | -0.3 | 820 | 58.9 | -9.3 | 6.1 |
| 560 | 100.0 | 0.0 | 0.0 | 830 | 61.9 | -9.8 | 6.5 |

TABLE B.1. Mean and characteristic vectors.

(c) Daylight from the sun disk alone or daylight from the sun disk at low altitudes plus sky generally result in spectral power distributions corresponding to correlated color temperatures below $5000K$.

(d) The correlated color temperature of the clear sky ranges from approximately $100000K$ and over in the morning, to approximately $10000K$ just before sunset. Considerably less variation is observed for the overcast sky, for which correlated color temperatures range from approximately $6000K$ to $7000K$ almost throughout the day: an increase to about $8500K$ is observed in the evening.

(e) The overcast sky is associated with a fairly constant correlated color temperature (i.e. there are not much changes in the spectral distribution) and large changes in illuminance. The reverse is observed for the clear sky, i.e. for the clear sky, it is mainly chromatic changes that are observed, and not illuminance.

Note that the last item is a further justification for the use of normalized coordinates in color processing under daylight, since they allow illuminace changes to be factored out.

In the ultraviolet (300 to 400 $nm$) and infrared (above 700 $nm$) regions of the spectrum, seasonal and time-of-day variations are known to occur. Thus, in these spectral regions, the CIE distributions may, at times, deviate significantly from the actual distribution. This, however, need not be of concern here, as we are interested in the visible range, *i.e.* from about 400 $nm$ to 700 $nm$, and the camera used is only sensitive to light in that range.

A few comments about the model of Judd *et al.*. From Table B.1 and Fig. 2.4, it will be noticed that at $\lambda = 560nm$, the so-called *normalizing wavelength*, the mean spectral power distribution is arbitrarily set to 100, and the values of the characteristic vectors are all zero. Thus, from Eq. 2.4.4, and as shown in Fig. 2.5, all reconstituted spectral power distributions of daylight will be normalized to a value of 100 at the normalizing wavelength. In Fig. 2.4 it is also seen that the values of the first characteristic vector $S_1$ are positive for wavelengths less than 560 $nm$ and negative for wavelengths greater than 560 $nm$. As this vector is the one of greatest importance for describing the variations of daylight, it follows that the most common variation between two measured curves of daylight spectral distribution consists chiefly in a change in correlated color temperature. This corresponds to yellow-blue variations[2], and is caused by the presence or absence of clouds in the sky, and by the inclusion or exclusion of direct sunlight. As light clouds develop, the blue component increases slightly, and with heavy overcast the compostion of daylight is only slightly less blue than for clear sky conditions, and actually exceeds it in the ultraviolet region. Note also that the second characteristic vector has low values from 380 to 560 $nm$ and higher values in the red portion of the visible spectrum. Thus, this second most common variation corresponds to a pink-green variation as might be caused by the presence of little or much water in the form of vapor and haze.

[2]Recall that the correlated color temperature corresponds to the temperature of a blackbody.

In summary, given the correlated color temperature of a phase of daylight, a representative of its relative spectral power distribution can be obtained with the help of the CIE method of calculating daylight illuminants. Studies since 1964, when the CIE issued its recommendation for calculating daylight illuminants, have on the whole confirmed the CIE spectral distributions. However, deviations have been observed and attributed to the location at which the measurements were taken as well as to seasonal parameters, see the references in [**77**], p.11. Also since then, it is reasonable to expect that due to atmospherical factors, mainly pollution, daylight has changed, and so newer data would be welcome. Nevertheless, the method and general conclusions of Judd *et al.* should remain applicable to new data.

# APPENDIX C

## Some Radiometric Terminology and a Derivation of the Color Formation Equations

In this Appendix, a model for describing each monochromatic component of light reaching a sensor after having suffered reflection from objects will be exposed. Then it will be seen how a camera integrates these components to generate a color signal. From this, the so-called *color formation equations* will be obtained.

### C.1 Reflectance Model

In an effort to specify the geometrical reflectance properties of surfaces in a unified way, Nicodemus *et al.* [**134**] have introduced the notion of bidirectional reflectance-distribution function (BRDF). Moreover, in view of standardization, these authors have also proposed a radiometric nomenclature, comprising definitions of concepts, terms, symbols, and units, for categorizing and specifying reflectance quantities for a variety of incident and reflected beam configurations.

In a clear exposition based on these concepts, Horn and Sjoberg [**135**] have described the imaging process of a scene in precise terms, and we shall borrow from it. We first begin by stating relevant concepts defined in the nomenclature proposed by Nicodemus *et al.*[1].

Radiant flux $\Phi$ is the power propagated as optical radiation and is measured in Watts ($W$). Radiant intensity $I$ of a source is the exitant flux per unit solid angle, and is measured in watts per steradian ($W \cdot sr^{-1}$). The total flux emitted by a source is the radiant intensity

[1]In all that follows, we shall however slightly depart from the notation in [**134**] and [**135**], in that we shall write explicit dependence on wavelength, since this dependence is important when dealing with color.

integrated over the full sphere of possible directions ($4\pi\ sr$). Irradiance $E$ on a surface is the incident flux density, while radiant exitance $M$ from a surface is the exitant flux density from that surface; both are measured in Watts per square meter of surface ($W \cdot m^{-2}$). Total radiant exitance is equal to total irradiance if the surface reflects all incident light, and no absorption or transmission takes place. The radiance $L$ is the flux emitted per unit foreshortened surface area per unit solid angle. Radiance is measured in Watts per square meter per steradian ($W \cdot m^{-2} \cdot sr^{-1}$). It can equivalently be defined as the flux emitted per unit surface area per unit projected solid angle. Radiance is an important concept because the apparent brightness of a surface patch is related to its radiance.

The BRDF, denoted by $f_r$, embodies the information about how bright a surface appears when viewed from a given direction, as it is illuminated from another direction. More precisely, $f_r$ is the ratio of reflected radiance $dL_r$ in the direction towards the viewer to the irradiance $dE_i$ in the direction towards an element of the source. Explicitly,

$$f_r(\lambda, \theta_i, \phi_i, \theta_r, \phi_r) = dL_r(\lambda, \theta_i, \phi_i, \theta_r, \phi_r; E_i)/dE_i(\lambda, \theta_i, \phi_i). \tag{C.1.1}$$

The geometry is illustrated in Fig. C.1, where $\theta_i$ is the angle of incidence between the illumination direction $\mathbf{L}$ and the surface normal $\mathbf{N}$ (taken to be in the $z$ direction), $\phi_i$ the azimuth of the illumination direction, $\theta_r$ the angle of exitance between $\mathbf{N}$ and the viewing direction $\mathbf{V}$, $\phi_r$ the azimuth of the viewing direction [**134**]. The BRDF allows one to obtain reflectance for any incident and reflected ray geometry. This is, in fact, its most useful property, since it allows a unified description of reflectance.

Gray levels in a digital image are quantized measurements of the scene radiance reaching the camera (color images can be thought of as three gray level images). Horn and Sjoberg have shown that image irradiance, $E_p$, is proportional to scene radiance, $L_r$. Mathematically their result is expressed as

$$E_p = L_r(\pi/4)(d/f_p)^2 \cos^4 \alpha, \tag{C.1.2}$$

where the relevant quantities are illustrated in Fig. C.2 (do not confuse $f_p$ with $f_r$, $f_p$ being the distance from the image plane to the lens). The reader is referred to [**135**] for the details of the derivation. The factor $\cos^4 \alpha$ represents the effect of vignetting, and depends on image position through the angle $\alpha$. Underlying the last equation is the assumption that

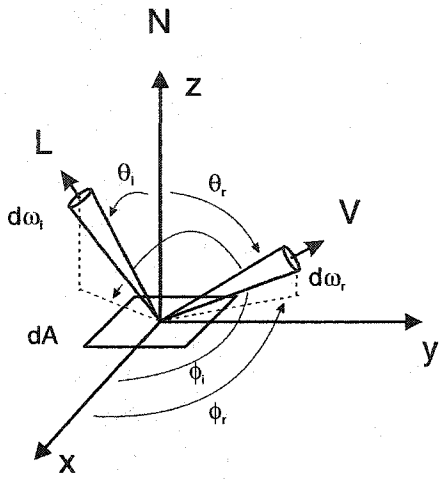



FIGURE C.1. Local geometry of incident and reflected rays. Adapted from [**134**] and [**135**].

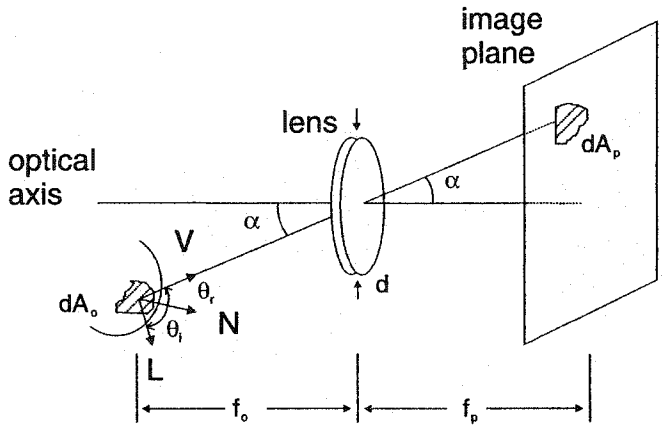



FIGURE C.2. Simple image forming configuration. Light collected by the lens from the surface patch of area $dA_o$ is imaged as the patch of area $dA_p$. Adapted from [**135**].

the dimensions of the lens are small compared to its distance from the object, and so for a point $p$ in the scene, $\theta_r$ and $\phi_r$ can be considered as fixed. Now scene radiance, $L_r$, at $p$ depends on properties of the surface, via the BRDF, and the distribution of light sources,

$$L_r(\lambda;\theta_r,\phi_r) = \int_{\omega_i} f_r(\lambda;\theta_i,\phi_i,\theta_r,\phi_r) L_i(\lambda;\theta_i,\phi_i)\cos\theta_i d\omega_i. \tag{C.1.3}$$

Here $\cos\theta_i$ is the geometrical foreshortening factor, $\omega_i$ is the solid angle subtended by all sources, and explicit dependence of $L_i$ on the angles $\theta_i$ and $\phi_i$ expresses the fact that, in general, the incident flux may arrive from many portions of an extended source and that

this source may not necessarily be uniform. The term $L_i(\lambda;\theta_i,\phi_i)\cos\theta_i d\omega_i$ is the irradiance $dE_p(\lambda;\theta_i,\phi_i)$ corresponding to the incident light flux arriving from direction $(\theta_i,\phi_i)$. As object surface element centered coordinates are used here, all angles vary with the point $p$ on the surface. Notice that, so written, Eq. (C.1.3) is general, and hence also valid in the case of many light sources. It is the combination of the last two equations that constitutes our model for the reflection on a surface and collection by a camera of each spectral component of light, explicitly

$$E_p(\lambda) = (\pi/4)(d/f_p)^2\cos^4\alpha\int_{\omega_i} f_r(\lambda;\theta_i,\phi_i,\theta_r,\phi_r)L_i(\lambda;\theta_i,\phi_i)\cos\theta_i d\omega_i. \tag{C.1.4}$$

This equation describes one of the basic ingredients taking part in the process of color image formation. What has been done here is a more rigourous and detailed presentation of what is to be found in [**59**].

C.1.0.1 *Spectral Integration.* So far the description has been phrased in terms of a continuous representation of the light spectrum: its spectral power distribution. Sensing devices, whether it be cameras or the human eye, use a finite set of measurements to represent the spectrum. For color cameras and normal trichromats, there are three such measurements obtained by filtering the incoming light spectrum and integrating over the filtered spectrum. It is these measurements that are used to describe what we call color. The filtering action is done by placing color filters in the path of the incoming light on its way to the sensor. Mathematically, this amounts to a projection of the infinite dimensional space of all spectral distributions onto a three dimensional space: the color space. To each filter (typically red, green and blue) is associated a transmittance function $\tau_i(\lambda)$ ($i = R, G$ and $B$) specifying the fraction of light transmitted at each wavelength[2]. The spectrum of an incoming light beam at a position $(x, y)$ on the sensor is therefore represented by a triple $\mathbf{C}(x,y) = (C_R, C_G, C_B)$[3], the color triple, or in short the color, given by

$$C_i = \int_{visible} E_p(\lambda)\tau_i(\lambda)s(\lambda)d\lambda \quad i = R, G, B, \tag{C.1.5}$$

[2]We will also assume that the spectral effects of the imaging optics, if any, is included in these transmittance functions.

[3]These quantities should not be confused with the $R, G, B$ values defined in colorimetry and related to a standard mean observer [**77**]. This is a confusion, which unfortunately occurs too frequently. This is why we resort to the notation $(C_R, C_G, C_B)$ instead of $(R, G, B)$.

with $s(\lambda)$ being the spectral responsivity of the sensor. For convenience and following the notation in [82] p.11, we now introduce the following quantities

$$s_i(\lambda) = \tau_i(\lambda)s(\lambda), \tag{C.1.6}$$

which represent the combined filtering effect of filter/lens/sensor. The measurement of these quantities is discussed in Appendix A. So the final expression modelling the responses of a camera to a light signal is

$$\mathbf{C} = \begin{pmatrix} \int_{visible} E_p(\lambda)s_R(\lambda)d\lambda \\ \int_{visible} E_p(\lambda)s_G(\lambda)d\lambda \\ \int_{visible} E_p(\lambda)s_B(\lambda)d\lambda \end{pmatrix}. \tag{C.1.7}$$

## C.2 Applying the Reflectance Model

Having described a general reflectance model, and how colors are formed, these ideas will now be applied to the problem of imaging under daylight. In applying the general reflectance model to each spectral component of light collected by a camera, Eq. (C.1.3) will be the starting point. It will then be combined with Eq. (C.1.2) to compute the image irradiance taking part in spectral integration, and which describes the contribution of the light coming from each point in a scene, *i.e.* Eq. (C.1.4) will be rewritten for the present specific case.

Eq. (C.1.3) embodies both the reflectance properties of a surface, and how it is illuminated. In Appendix B about daylight, it is seen that there are two contributions partaking in the spectral distribution of daylight: that of the sun, and that of the sky. These contributions may be direct or not, depending on the amount of overcast. In the measurements about daylight, these contributions are averaged together and described by a unique parameter: the correlated color temperature. To fix ideas, consider the example of a sunny day. Eq. (C.1.3), which is reproduced here,

$$L_r(\lambda;\theta_r,\phi_r) = \int_{\omega_i} f_r(\lambda;\theta_i,\phi_i,\theta_r,\phi_r)L_i(\lambda;\theta_i,\phi_i)\cos\theta_i d\omega_i, \tag{C.2.1}$$

describes the situation. In this example, $L_i$ is clearly dependent on $\theta_i$ because for angles pointing to the sun, the spectral distribution is different from that for angles pointing to

the sky. We just considered a specific example, that of a sunny day, but Eq. (C.2.1) is valid for any illumination conditions.

Now with regard to the reflectance properties of the materials, the focus will be put on the particular colors of objects, and only body reflectance will be considered (although, as was seen in Section 2.4.1, some degree of specularity must usually be taken into consideration, which would amount to add a specular term in Eq. C.2.2 below to account for this).

For this type of reflectance, the BRDF can be assumed to be independent of the incidence and reflection angles, and given by

$$f_r(\lambda) = \frac{R(\lambda)}{\pi}, \tag{C.2.2}$$

where $R(\lambda) \in [0, 1]$ is the diffuse (or body) reflectance factor which is measured as described in Section 2.4.1. Strictly speaking, body reflection is not Lambertian, because not all incident light is reflected back. This is why the reflectance factor $R$ appears in Eq. (C.2.2)[4]. However, when only body reflection is observed, it is true that the object looks the same color with the same brightness irrespective of the viewing direction. Combining the last equation with Eq. (C.2.1), we obtain

$$L_r(\lambda) = R(\lambda) \left[ \frac{1}{\pi} \int_{\omega_i} L_i(\lambda; \theta_i, \phi_i) \cos\theta_i d\omega_i \right]. \tag{C.2.3}$$

It is seen that the term in brackets is an average of the illumination over all incident directions, and so it must correspond, to within a constant factor, to the *relative* spectral power distributions, as presented in Section 2.4.3. This average will be denoted $\overline{L_i}(\lambda)$, and Eq. (C.2.3) becomes

$$L_r(\lambda) = R(\lambda)\overline{L_i}(\lambda). \tag{C.2.4}$$

Combining Eqs. (C.2.4) and (C.1.2), Eq. (C.1.4) can be rewritten as

$$E_p = R_p(\lambda)\overline{L_{i,p}}(\lambda)(\pi/4)(d/f_p)^2 \cos^4\alpha. \tag{C.2.5}$$

In the last equation there is a dependence on the point $p$ on the object in the scene. Making abstraction of pixelization in the image, there is a one-to-one correspondence between

[4]If the reflection was Lambertian at all wavelengths, $R(\lambda)$ would be constant and equal to one.

the viewed points in the scene and the points in the image. For an object of small extent compared to its distance to the light source, this dependence on $p$ can be neglected[5]. Finally, lumping together all geometrical factors and the arbitrary factor related to the measurements of daylight[6] into a constant, we get that for all points on an object

$$E(\lambda) = cst \cdot R(\lambda) S_D(\lambda; T_c). \tag{C.2.6}$$

## C.3 About Shape from Shading

Before closing on image formation, we would like to add some comments on Eq. (C.1.3) in relation to shape from shading. In this equation, $L_i(\lambda; \theta_i, \phi_i)$ can be considered as a directional quantity specified by the vector $\mathbf{L}_i(\lambda; \theta_i, \phi_i)$ pointing towards an element of the light source, and the length of which gives the strength of that source element. We will write $\mathbf{L}_i$ as $L_i \mathbf{u}_i$, $L_i$ being the length, and $\mathbf{u}$ a unit vector in the direction of $\mathbf{L}$. With this, Eq. (C.1.3) can be rewritten as

$$L_r(\lambda; \theta_r, \phi_r) = \int_{\omega_i} f_r(\lambda; \theta_i, \phi_i, \theta_r, \phi_r) \mathbf{L}_i(\lambda; \theta_i, \phi_i) \cdot \mathbf{N} d\omega_i, \tag{C.3.1}$$

with $\mathbf{N}$ being the normal to the surface at $p$, and the dot product embodying the geometrical foreshortening factor.

In shape from shading, the light rays coming from the source are usually assumed to be well collimated (i.e. point source at infinity), and thus

$$\mathbf{L}_i(\lambda; \theta_i, \phi_i) = E_0(\lambda) \frac{\delta(\theta_i - \theta_0)\delta(\phi_i - \phi_0)}{\sin\theta_0}, \tag{C.3.2}$$

hence

$$L_r(\lambda; \theta_r, \phi_r) = f_r(\lambda; \theta_0, \phi_0, \theta_r, \phi_r) E_0(\lambda) \mathbf{u}_i(\theta_0, \phi_0) \cdot \mathbf{N}. \tag{C.3.3}$$

Further, it is also supposed that the surface is Lambertian, in which case $f_r = R(\lambda)/\pi$ (see previous section). With those reflection assumptions, Eq. (C.1.3) reduces to

$$L_r(\lambda) = \frac{R(\lambda)}{\pi} E_0(\lambda) \mathbf{u}_i(\theta_0, \phi_0) \cdot \mathbf{N}. \tag{C.3.4}$$

[5]This is the case here since planar objects (be it road signs or the small diamond shaped color samples) and the sky are considered.
[6]We only have relative, not absolute, spectral distributions.

For a black and white camera, the gray level $G$ at a pixel $p$ is obtained through spectral integration[7]

$$G_p = k \int E_p(\lambda) s(\lambda) d\lambda, \tag{C.3.5}$$

where $k$ is the electronic transduction gain in the camera and $s$ the spectral sensitivity of the camera. Using the last equation in conjunction with Equations (C.1.2) of Section C.1, and (C.3.4), we find

$$G_p = k' \mathbf{u}_i(\theta_0, \phi_0) \cdot \mathbf{N} \int R(\lambda) E_0(\lambda) s(\lambda) d\lambda, \tag{C.3.6}$$

where constant factors have been lumped into the constant $k'$. Now, in the shape from shading literature, using the present notation, one usually sees the following equation

$$G_p = a \rho L \, \mathbf{N} \cdot \mathbf{u}, \tag{C.3.7}$$

$a$ being a proportionality factor. This is not fully correct, because in writing $\rho L$ in the last equation, spectral integration, which also takes place in black and white cameras, is neglected. However, regarding shading effects, those are correctly described by the $\mathbf{N} \cdot \mathbf{u}$ factor. Eq. (C.3.7) would be correct when lighting is monochromatic, but Eq. (C.3.6) is the one to be used for the broadband case (usually in shape from shading white sources of light are used).

[7]This is the same as Eq. (C.1.5), but restricted to a single channel with no color filter.

# APPENDIX D

---

# Road Sign Database

## D.1 Description

This appendix serves to illustrate the variety of road signs considered in the system described in this thesis. It is not the purpose here to give all the road signs in the database, but rather to show a few examples for each class to give the reader an idea of the diversity of the road signs. The reader will find images of the road signs in the figures of the following pages.It will be noticed that more examples are given for some classes than for others; the number of examples shown somewhat represent how large is a class, with the class *LosJn3* definitively being the largest, see Fig. D.15. Moreover, some classes contain only one road sign. The number in parentheses that appears at the end of the figure caption for each class gives the number of road signs contained in the class. In total, the database contains 509 road signs distributed over 48 road sign classes.

A few remarks are in order. Road sign colors have specific meanings, and these are specified for the eight following colors [**136, 137**]:

- Red - stop, prohibition, and prescription.
- Yellow - general warning, and danger.
- Green - indicated movements permitted, direction guidance.
- Orange - construction and maintenance warning.
- Blue - motorist services guidance.
- Brown - recreational and cultural interest guidance.
- Black and white - regulation, and prescription.

Lately, a new color has been added to the road sign coloring system which is called yellow-green. It has a fluorescent yellow appearance, and is dedicated to signal school areas and school kids. Also, the orange color dedicated to construction and maintenance warning has been replaced by a fluorescent orange for higher visibility. These colors are not considered in the present system, but should be in future versions. Finally, although black and white road signs are not processed as mentioned in Chapter 5, the database nevertheless contains the models for those for future purposes, and these are depicted here.

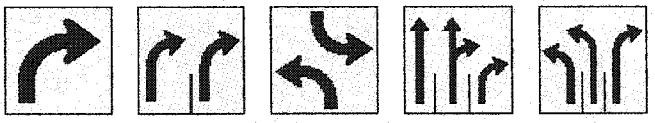


FIGURE D.1. Road signs in the class *CarBn* (13).

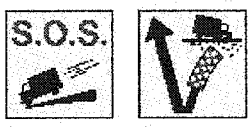



FIGURE D.2. Road signs in the class *CarBnr* (2).

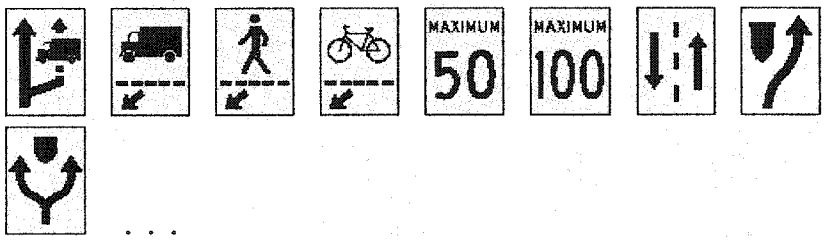



FIGURE D.3. Road signs in the class *RectAlngBn* (33).

FIGURE D.4. Road sign in the class *RectAlngBnr* (1).


FIGURE D.5. Road signs in the class *NosAut* (10).

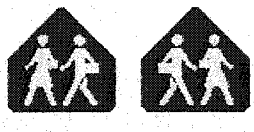


FIGURE D.6. Road signs in the class *PentIrr* (2).

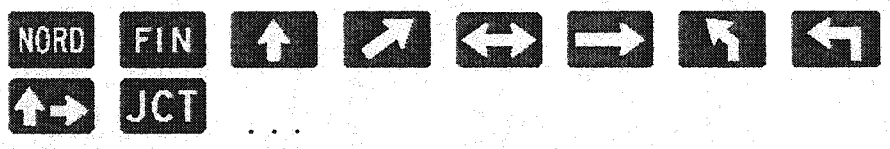



FIGURE D.7. Road signs in the class *RectBeba* (26).

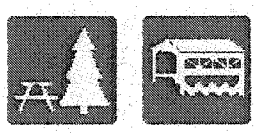


FIGURE D.8. Road signs in the class *CarBr* (2).

FIGURE D.9. Road signs in the class *RectAlngHautBr* (1).

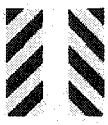


FIGURE D.10. Road signs in the class *Bal1CotJn* (2).

FIGURE D.11. Road sign in the class *Bal2CotJn* (1).

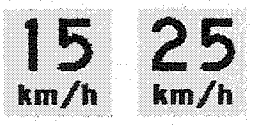



FIGURE D.12. Road signs in the class *CarLimVitJn* (8).

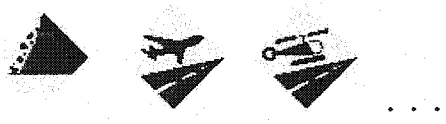


FIGURE D.13. Road signs in the class *LosJn1* (6).

FIGURE D.14. Road signs in the class *LosJn2* (5).

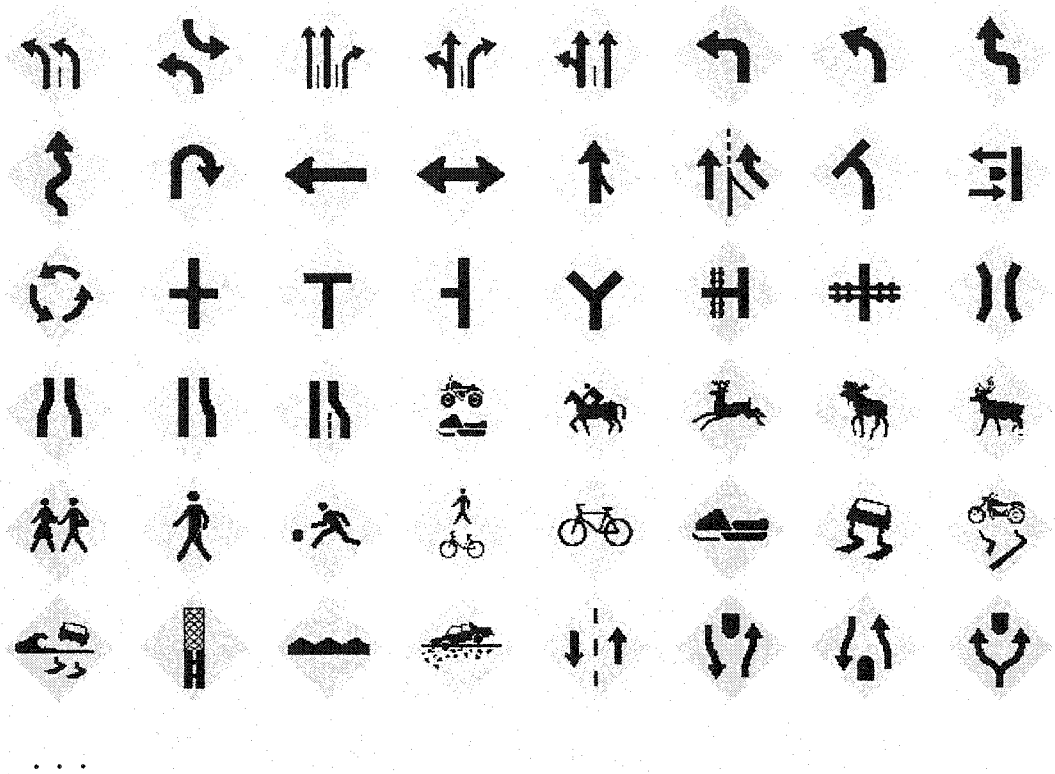


FIGURE D.15. Road signs in the class *LosJn3* (142).

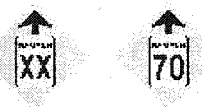



FIGURE D.16. Road signs in the class *LosJnb1* (2).

FIGURE D.17. Road signs in the class *LosJnb2* (1).

FIGURE D.18. Road sign in the class *LosJnr* (1).

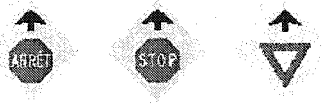



FIGURE D.19. Road signs in the class *LosJnrbA* (3).

FIGURE D.20. Road sign in the class *LosJnrbB* (1).

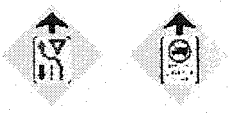


FIGURE D.21. Road signs in the class *LosJnrbC* (2).

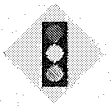


FIGURE D.22. Road sign in the class *LosJnrjv* (1).

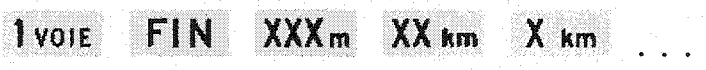



FIGURE D.23. Road signs in the class *RectAnn* (8).

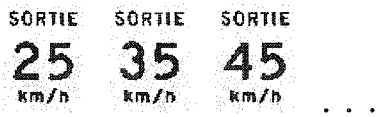



FIGURE D.24. Road signs in the class *RectLimVitJn* (8).


FIGURE D.25. Road signs in the class *SortHaut* (1).


FIGURE D.26. Road signs in the class *SortLar* (1).

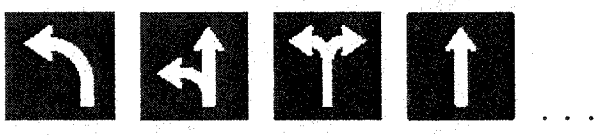


FIGURE D.27. Road signs in the class *CarNb* (6).

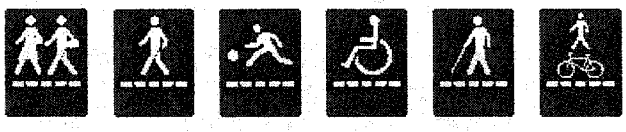


FIGURE D.28. Road signs in the class *RectNb* (6).

FIGURE D.29. Road signs in the class *SensUn* (2).

FIGURE D.30. Road sign in the class *BalO* (1).

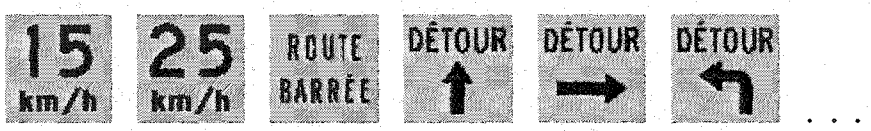



FIGURE D.31. Road signs in the class *CarOn* (16).

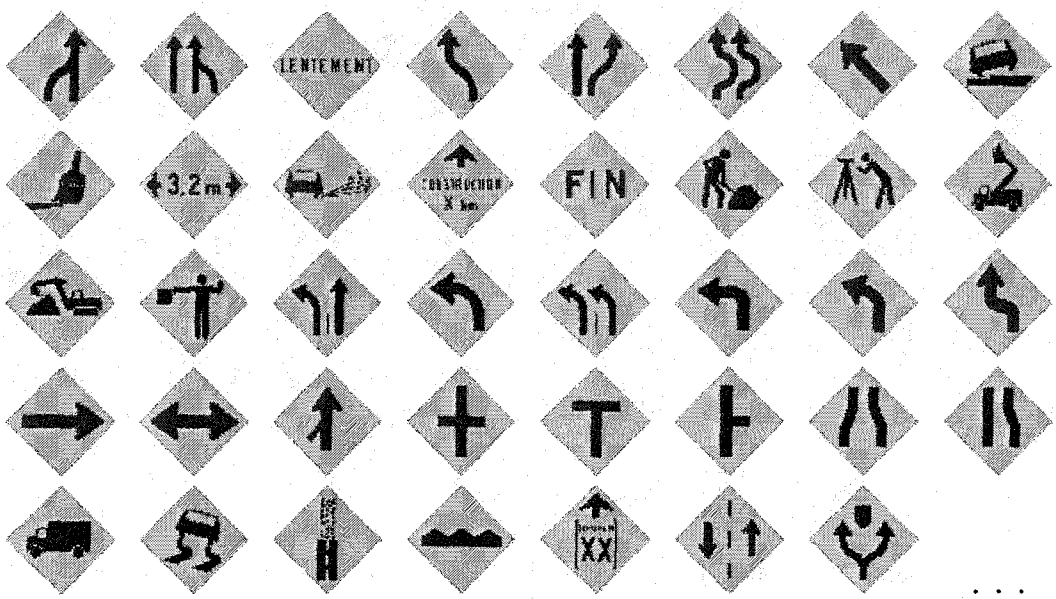



FIGURE D.32. Road signs in the class *LosOn* (70).


FIGURE D.33. Road signs in the class *LosOnrb* (1).

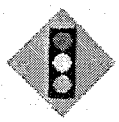
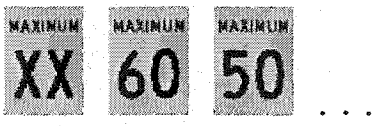


FIGURE D.34. Road sign in the class *LosOnrjv* (1).


FIGURE D.35. Road signs in the class *PetRectOn* (4).




. . .

FIGURE D.36. Road signs in the class *RectLimVitOn* (5).


FIGURE D.37. Road signs in the class *Arr* (2).

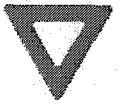


FIGURE D.38. Road sign in the class *Ced* (1).

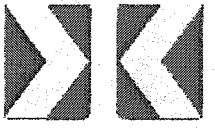


FIGURE D.39. Road signs in the class *Chev* (2).

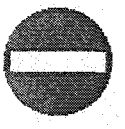
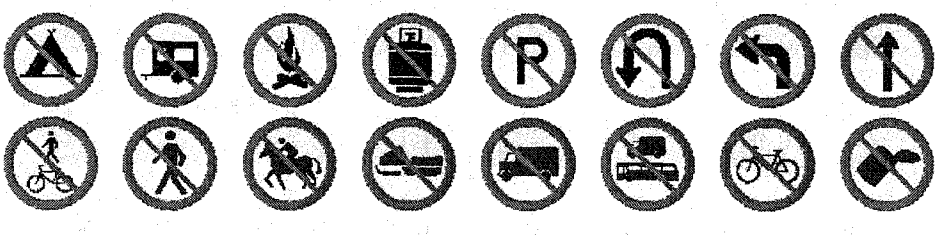


FIGURE D.40. Road signs in the class *EntInt* (1).




. . .

FIGURE D.41. Road signs in the class *IntRbn* (26).

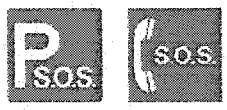



FIGURE D.42. Road signs in the class *SOS* (2).

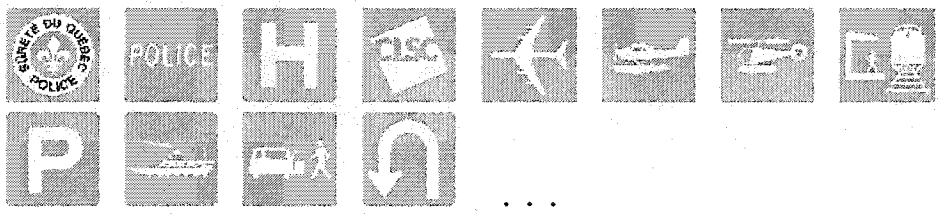



FIGURE D.43. Road signs in the class *CarVb* (15).


FIGURE D.44. Road signs in the class *NosRout* (3).

FIGURE D.45. Road signs in the class *OblVbn* (14).

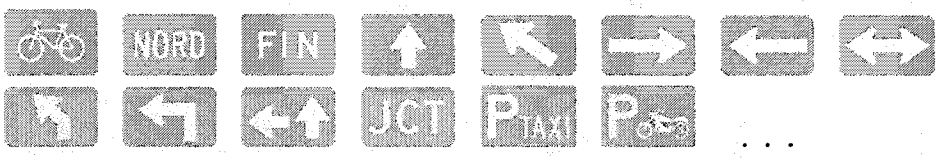



FIGURE D.46. Road signs in the class *RectPet1* (28).

FIGURE D.47. Road signs in the class *RectPet2* (19).

FIGURE D.48. Road sign in the class *Trans* (1).

# REFERENCES


[1] S. Estable, J. Schick, F. Stein, R. Janssen, R. Ott, W. Ritter, and Y.-J. Zheng, "A Real-Time Traffic Sign Recognition System," in *Proc. Intelligent Vehicles '94 Symposium*, pp. 213–218, 1994.

[2] L. Guibert and M. Attia, "Optical Processing at the Driver's Service," in *Towards an Intelligent Transport System. Proceedings of the First World Congress on Applications of Transport Telematics and Intelligent Vehicle-Highway Systems*, vol. 4, pp. 2134–2140, 1995.

[3] B. Ulmer, "VITA II - Active Collision Avoidance in Real Traffic," in *Proceedings of the Intelligent Vehicles '94 Symposium*, pp. 1–6, 1994.

[4] Ministère des Transports du Québec, *Personal communication*, October 1997.

[5] L. Pacheco, J. Batlle, and X. Cufi, "A New Approach to Real Time Traffic Sign Recognition Based on Colour Information," in *Proceedings of the Intelligent Vehicles '94 Symposium*, pp. 339–344, 1994.

[6] H. Akatsuka and S. Imai, "Road Signposts Recognition System," in *Proceedings SAE Vehicle Highway Infrastructure: Safety Compatibility*, pp. 189–196, 1987.

[7] J. J. Lumia, "A Mobile System for Measuring Retroreflectance of Traffic Signs," in *SPIE Proceedings: Optics, Illumination, and Image Sensing for Machine Vision V*, vol. 1385, pp. 15–26, 1990.

[8] M. de Saint Blancard, "Road Sign Recognition: A Study of Vision-Based Decision Making for Road Environment Recognition," in *Vision-Based Vehicle Guidance* (I. Masaki, ed.), Springer Series in Perception Engineering, pp. 162–173, Springer Verlag, New York, 1992.

[9] D. Krumbiegel, K. F. Kraiss, and S. Schreiber, "A Connectionist Traffic Sign Recognition System for Onboard Driver Information," in *5th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design and Evaluation of Man-Machine Systems 1992*, pp. 201–206, 1993.

[10] M. Betke and N. C. Makris, "Fast Object Recognition in Noisy Images Using Simulated Annealing," Tech. Rep. A.I. Memo No. 1510, C.B.C.L. Memo No. 109, Artificial Intelligence Laboratory and Center for Biological and Computational Learning Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, December 1994.

[11] D. L. Kellmeyer and H. T. Zwahlen, "Detection of Highway Warning Signs in Natural Video Images using Color Image Processing and Neural Networks," in *IEEE Proceedings of the International Conference on Neural Networks* (K. R. Steven, ed.), vol. 7, pp. 4226–4231, 1994.

[12] M. Lalonde and Y. Li, "Detection of Road Signs using Color Indexing," Tech. Rep. CRIM-IIT-95/12-49, Centre de Recherche Informatique de Montréal (CRIM), 1995.

[13] M. Swain and D. Ballard, "Color Indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[14] S. Azami and M. Aoki, "Route Guidance Sign Recognition," in *Proceedings of the Intelligent Vehicles '95 Symposium*, pp. 338–343, 1995.

[15] S. Azami, S. Katahara, and M. Aoki, "Route Guidance Sign Identification Using 2-D Sructural Description," in *Proceedings of the 1996 IEEE Intelligent Vehicles Symposium*, pp. 153–158, 1996.

[16] P. Arnoul, M. Viala, J. Guerin, and M. Mergy, "Traffic Signs Localisation for Highways Inventory from a Video Camera on Board a Moving Collection Van," in *Proceedings of the 1996 IEEE Intelligent Vehicles Symposium*, pp. 141–146, 1996.

[17] M. Zadeh, T. Kasvand, and C. Suen, "Localization and Recognition of Traffic Signs for Automated Vehicle Control Systems," in *SPIE Proceedings: Intelligent System and Automated Manufacturing*, vol. 3207, pp. 272–282, 1997.

[18] N. Kehtarnavaz, N. C. Griswold, and D. S. Kang, "Stop-Sign Recognition Based on Color-Shape Processing," *Machine Vision and Applications*, vol. 6, pp. 206–208, 1993.

[19] N. Kehtarnavaz and A. Ahmad, "Traffic Sign Recognition in Noisy Outdoor Scenes," in *Proceedings of the Intelligent Vehicles '95 Symposium*, pp. 460–465, 1995.

[20] L. Estevez and N. Kehtarnavaz, "A Real-Time Histographic Approach to Road Sign Recognition," in *Proceedings of the IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 95–100, 1996.

[21] S. K. Kim and D. A. Forsyth, "A New Approach for Road Sign Detection and Recognition Algorithm," in *30th Int. Symposium on Automative Technology and Automation (ISATA 97), Florence, Italy, June*, pp. 171–178, June 1997.

[22] S. K. Kim, "An Efficient Roadsign Detection and Recognition Algorithm," in *SPIE Proceedings: Conference on Enhanced and Synthetic Vision*, vol. 3364, pp. 336–347, 1999.

[23] H. Murase and S. K. Nayar, "Visual Learning and Recognition of 3D Objects from Appearance," *International Journal of Computer Vision*, pp. 5–24, 1995.

[24] S. K. Nayar, H. Murase, and S. Nene, "Real-Time 100 Object Recognition System," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 3237–3244, 1994.

[25] A. de la Escalera, L. E. Moreno, M. Salichs, and J. Armingol, "Road Traffic Sign Detection and Classification," *IEEE Transactions on Industrial Electronics*, vol. 44, no. 6, pp. 848–859, 1997.

[26] G. Nicchiotti, E. Ottaviani, P. Castello, and G. Piccioli, "Automatic Road Sign Detection and Classification from Color Image Sequences," in *Proceedings 7th International Conference on Image Analysis and Processing* (S. Impedovo, ed.), pp. 623–626, World Scientific, 1994.

[27] G. Piccioli, E. De Micheli, P. Parodi, and M. Campani, "Robust Road Sign Detection and Recognition from Image Sequences," in *Proceedings of the Intelligent Vehicles '94 Symposium*, pp. 278–283, 1994.

[28] E. De Micheli, R. Prevete, G. Piccioli, and M. Campani, "Color Cues for Traffic Scene Analysis," in *Proceedings of the Intelligent Vehicles '95 Symposium*, pp. 466–471, 1995.

[29] P. Parodi and G. Piccioli, "A Feature-Based Recognition Scheme for Traffic Scenes," in *Proceedings of the Intelligent Vehicles '95 Symposium*, pp. 229–234, 1995.

[30] G. Piccioli, E. De Micheli, P. Parodi, and M. Campani, "Robust Method for Road Sign Detection and Recognition," *Image and Vision Computing*, vol. 14, pp. 209–223, 1996.

[31] L. Priese, J. Klieber, R. Lakmann, V. Rehrmann, and R. Schian, "New Results on Traffic Sign Recognition," in *Proc. Intelligent Vehicles '94 Symposium*, pp. 249–253, 1994.

[32] L. Priese, V. Rehrmann, R. Schian, and R. Lakmann, "Traffic Sign Recognition Based on Color Image Evaluation," in *Proc. Intelligent Vehicles '93 Symposium*, pp. 95–100, 1993.

[33] L. Priese and V. Rehrmann, "A Fast Hybrid Color Segmentation Method," *Fachberichte Informatik, Universität Koblenz-Landau*, 1992.

[34] L. Priese, R. Lakmann, and V. Rehrmann, "Ideogram Identification in a Realtime Traffic Sign Recognition System," in *Proc. Intelligent Vehicles '95 Symposium*, pp. 310–314, 1995.

[35] W. Ritter, "Traffic Sign Recognition in Color Image Sequences," in *Proc. Intelligent Vehicles '92 Symposium*, pp. 12–17, 1992.

[36] R. Janssen, W. Ritter, F. Stein, and S. Ott, "Hybrid Approach for Traffic Sign Recognition," in *Proc. Intelligent Vehicles '93 Symposium*, pp. 390–395, 1993.

[37] W. Ritter, F. Stein, and R. Janssen, "Traffic Sign Recognition Using Colour Information," *Math. Comput. Modelling*, vol. 22, pp. 149–161, 1995.

[38] J. K. Tsotsos, "The Complexity of Perceptual Search Tasks," in *Proc. of the 8th IJCAI, Detroit*, vol. 2, pp. 1571–1577, 1989.

[39] B. Besserer, S. Estable, and B. Ulmer, "Multiple Knowledge Sources and Evidential Reasoning for Shape Recognition," in *Proc. of the 4th International Conference on Computer Vision*, (Berlin), pp. 624–631, 1993.

[40] B. Mertsching, H. Austermeier, U. Büker, and G. Hartmann, "Interpretation of Traffic Scenes Using a Hierarchical Data Structure," in *Proceedings of the Intelligent Vehicles '93 Symposium*, pp. 163–168, 1993.

[41] N. Bartneck and W. Ritter, "Colour Segmentation with Polynomial Classification," in *11th Int. Conf. on Patt. Recog. (ICPR)*, (The Hague), pp. 635–638, 1992.

[42] J. Schürmann, *Polynomklassifikatoren für die Zeichenerkennung*. Oldenbourg Verlag, 1977.

[43] E. Mandler and M. Oberländer, "One-Pass Encoding of Connected Components in Multi-Valued Images," in *10th International Conference on Pattern Recognition (ICPR)*, pp. 64–69, 1990.

[44] Y.-J. Zheng, W. Ritter, and R. Janssen, "An Adaptive System for Traffic Sign Recognition," in *Proc. Intelligent Vehicles '94 Symposium*, pp. 165–170, 1994.

[45] "Computer im Auto erkennt Verkehrszeichen," *Laser und Optoelektronik, AT-Fachverlag GmbH Stuttgart*, Februar 1997.

[46] D. M. Gavrila, "Multi-Feature Hierarchical Template Matching Using Distance Transforms," in *Proceedings of the International Conference on Pattern Recognition*, (Brisbane, Australia), pp. 439–444, 16-20 August 1998.

[47] D. M. Gavrila and V. Philomin, "Real-Time Object Detection Using Distance Transforms," in *Proc. of the International Conference on Intelligent Vehicles*, (Stuttgart, Germany), 28-30 October 1998.

[48] D. M. Gavrila, "Traffic Sign Recognition Revisited," in *Proceedings of DAGM Symposium für Mustererkennung*, (Bonn, Germany), 15-17 September 1999.

[49] D. M. Gavrila and V. Philomin, "Real-Time Object Detection for "Smart" Vehicles," in *Proc. of the International Conference on Computer Vision*, (Corfu, Greece), pp. 87–93, 20-25 Sept. 1999.

[50] S. Prince and R. Bergevin, "Road Sign Detection and Recognition Using Perceptual Grouping," in *30th Int. Symposium on Automative Technology and Automation (ISATA 97), Florence, Italy, June*, pp. 163–170, June 1997.

[51] C.-L. Huang and S.-H. Hsu, "Road Sign Detection and Recognition Using Matching Pursuit Method," in *Proceedings of the 4th Asian Conference on Computer Vision, Taipei, Taiwan*, pp. 681–686, January 2000.

[52] S.-H. Hsu and C.-L. Huang, "Road Sign Detection and Recognition Using Matching Pursuit Method," *Image and Vision Computing*, vol. 1, pp. 119–129, 2001.

[53] L. Guibert, G. Keryer, A. Servel, M. Attia, H. S. MacKenzie, P. Pellat-Finet, and J.-L. de Bougrenet de la Tocnaye, "On-Board Optical Joint Transform Correlator for Real-Time Road Sign Recognition," *Optical Engineering*, vol. 34, pp. 135–143, January 1995.

[54] L. Guibert, Y. Pétillot, and J.-L. de Bougrenet de la Tocnaye, "Real-Time Demonstration of an On-Board Nonlinear Joint Transform Correlator System," *Optical Engineering*, vol. 36, pp. 820–824, March 1997.

[55] K. Matsuoka, M. Taniguchi, and Y. Mokuno, "Application of Optical Multiple-Correlation to Recognition of Road Signs: the Ability of Multiple-Correlation," in *Inst. Phys. Conf. Ser. No.139: Part III: Opt. Comput. Int. Conf. Edinburgh 24-25 August 1994*, pp. 305–308, IOP Publishing Ltd, 1994.

[56] K. Matsuoka, M. Taniguchi, and Y. Mokuno, "Detection of Road Signs by using a Multiple Correlator and Partial Invariant Filters," in *SPIE Proceedings: Optics in Computing '98*, vol. 3207, pp. 178–181, 1993.

[57] E. Perez and B. Javidi, "Scale and Illumination-Invariant Road Sign Detection," in *Conference Proceedings of the 13th Annual Meeting of the IEEE Lasers and Electro-Optics Society (LEOS 2000)*, vol. 2, pp. 748–749, 2000.

[58] E. Perez and B. Javidi, "Composite filter bank for road sign recognition," in *Conference Proceedings of the 13th Annual Meeting of the IEEE Lasers and Electro-Optics Society (LEOS 2000)*, vol. 2, pp. 171–178, 2000.

[59] S. A. Shafer, "Using Color to Separate Reflection Components," *Color Research and Application*, vol. 10, no. 4, pp. 210–218, 1985.

[60] H.-C. Lee, "Method for Computing the Scene-Illuminant Chromaticity from Specular Highlights," *Journal of the Optical Society of America A*, vol. 3, no. 10, pp. 1694–1699, 1986.

[61] C. L. Novak and S. A. Shafer, "Supervised Color Constancy for Machine Vision," *Human Vision, Visual Processing, and Digital Display II, SPIE Proceedings*, vol. 1453, pp. 353–368, 1991.

[62] H. L. F. von Helmholtz, *Handbuch der Physiologischen Optik*. Leipzig: Voss, 1867. Also translated: J.P.C. Southall, *Handbook of Physiological Optics*, Dover, New York, 1962.

[63] E. H. Land, "Color Vision and the Natural Image. Part I," *Proceedings of the National Academy of Sciences, U.S.A.*, vol. 45, pp. 115–129, 1959.

[64] E. H. Land, "Color Vision and the Natural Image. Part II," *Proceedings of the National Academy of Sciences, U.S.A.*, vol. 45, pp. 636–644, 1959.

[65] E. H. Land, "Experiments in Color Vision," *Scientific American*, pp. 84–96, May 1959.

[66] E. H. Land and J. J. McCann, "Lightness and Retinex Theory," *Journal of the Optical Society of America*, vol. 61, pp. 1–11, January 1971.

[67] E. H. Land, "The Retinex Theory of Color Vision," *Scientific American*, vol. 237, no. 3, pp. 108–128, 1977.

[68] B. K. P. Horn, "Determining Lightness from an Image," *Computer Graphics and Image Processing*, vol. 3, pp. 277–299, 1974.

[69] A. Blake, "Boundary Conditions for Lightness Computation in Mondrian World," *Computer Vision, Graphics, and Image Processing*, vol. 32, pp. 314–327, 1985.

[70] D. H. Brainard and B. A. Wandell, "Analysis of the Retinex Theory of Color Vision," *Journal of the Optical Society of America A*, vol. 3, no. 10, pp. 1651–1661, 1986.

[71] P. Sällström, "Colour and Physics: Some Remarks Concerning the Physical Aspects of Human Colour Vision," Tech. Rep. 73-09, Institute of Physics, University of Stockholm, Stockholm, 1973.

[72] G. Buchsbaum, "A Spatial Processor Model for Object Colour Perception," *Journal of the Franklin Institute*, vol. 310, pp. 1–26, July 1980.

[73] M. H. Brill, "A Device Performing Illuminant-Invariant Assessment of Chromatic Relations," *Journal of Theoretical Biology*, vol. 71, pp. 473–478, 1978.

[74] L. T. Maloney and B. A. Wandell, "Color Constancy: A Method for Recovering Surface Spectral Reflectance," *Journal of the Optical Society of America A*, vol. 3, pp. 29–33, January 1986.

[75] D. A. Forsyth, "A Novel Algorithm for Colour Constancy," *International Journal of Computer Vision*, vol. 5, no. 1, pp. 5–36, 1990.

[76] J. von Kries, "Die Gesichtsempfindungen," *Nagel's Handbuch der Physiologie des Menschen*, vol. 3, p. 211, 1904.

[77] G. Wyszecki and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*. New York: John Wiley and Sons, 2 ed., 1982.

[78] G. Finlayson, "Color Constancy in Diagonal Chromaticity Space," in *5th Int. Conf. on Computer Vision (ICCV)*, pp. 218–223, 1995.

[79] D. B. Judd, D. L. MacAdam, and G. Wyszecki, "Spectral Distribution of Typical Daylight as a Function of Correlated Color Temperature," *Journal of the Optical Society of America*, vol. 54, pp. 1031–1040, August 1964.

[80] Y. Ohta and Y. Hayashi, "Recovery of Illuminant and Surface Colors from Images Based on the CIE Daylight," in *Lecture Notes in Computer Science, Vol. 801: Proceedings of the European Conference on Computer Vision* (J.-O. Eklundh, ed.), pp. 235–246, Springer-Verlag, 1994.

[81] G. Finlayson, B. Funt, and K. Barnard, "Color Constancy under Varying Illumination," in *5th Int. Conf. on Computer Vision (ICCV)*, pp. 720–725, 1995.

[82] G. A. Healy, S. A. Shafer, and L. B. Wolff, eds., *Physics-Based Vision, Principles and Practice in 3 Volumes: Color, Radiometry, and Shape Recovery*. Boston and London: Jones and Bartlett Publishers, 1992.

[83] G. J. Klinker, *A Physical Approach to Color Image Understanding*. Wellesley, Massachusetts: A. K. Peters, 1993.

[84] Y. Bérubé Lauzière, D. Gingras, and F. P. Ferrie, "Autonomous Physics-Based Colour Learning under Daylight," *SPIE Proceedings of the EOS/SPIE Conference on Polarisation and Colour Techniques in Industrial Inspection, Munich*, vol. 3826, pp. 86–100, June 1999.

[85] G. Finlayson, "Color in Perspective," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 1034–1038, 1996.

[86] W. Budde, "Spectral Energy Distributions of Natural Daylight." Unpublished, 1963.

[87] H. R. Condit and F. Grum, "Spectral Energy Distribution of Daylight," *Journal of the Optical Society of America*, vol. 54, pp. 937–944, July 1964.

[88] E. R. Dixon, "Spectral Distribution of Daylight," *Journal of the Optical Society of America*, vol. 68, pp. 437–450, April 1978.

[89] S. T. Henderson and D. Hodgkiss, "The Spectral Energy Distribution of Daylight," *British Journal of Applied Physics*, vol. 14, pp. 125–131, 1963.

[90] C. J. Kok, "Spectral Irradiance of Daylight at Pretoria," *Journal of Physics D: Applied Physics*, vol. 5, p. 1513, 1972.

[91] W. E. K. Middleton, "The Color of the Overcast Sky," *Journal of the Optical Society of America*, vol. 44, pp. 793–798, October 1964.

[92] Y. Nayatani and G. Wyszecki, "Color of Daylight from North Sky," *Journal of the Optical Society of America*, vol. 53, pp. 626–629, May 1963.

[93] K. E. Torrance and E. Sparrow, "Theory for Off-Specular Reflection from Roughened Surfaces," *Journal of the Optical Society of America*, vol. 57, no. 9, pp. 1105–1114, 1967.

[94] P. Beckmann and A. Spizzichino, *The Scattering of Electromagnetic Waves from Rough Surfaces*. New York: MacMillan, 1963.

[95] S. K. Nayar, K. Ikeuchi, and T. Kanade, "Surface Reflection: Physical and Geometrical Perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 7, pp. 611–634, 1991.

[96] R. L. Cook and K. E. Torrance, "A reflectance model for computer graphics," *Computer Graphics*, vol. 15, no. 3, pp. 307–316, 1981.

[97] G. A. Healy, "Using Color for Geometry-Insensitive Segmentation," *Journal of the Optical Society of America A*, vol. 6, no. 6, pp. 920–937, 1989.

[98] M. Born and E. Wolf, *Principles of Optics, 6th Ed.* Pergamon Press, 1980.

[99] C. L. Novak, S. A. Shafer, and R. G. Willson, "Obtaining Accurate Color Images for Machine Vision Research," *Perceiving, Measuring, and Using Color, SPIE Proceedings*, vol. 1250, pp. 54–68, 1990.

[100] Y. Bérubé Lauzière, D. Gingras, and F. P. Ferrie, "Color Camera Characterization with an Application to Detection under Daylight," *Proceedings of the Vision Interface 1999 Conference, Trois-Rivières, Québec, Canada*, pp. 280–287, May 1999.

[101] H. Niedrig, ed., *Lehrbuch der Experimentalphysik, Band 3: Optik*. Berlin-New York: Walter de Gruyter, 9 ed., 1993.

[102] N. Ohta and G. Wyszecki, "Color Changes Caused by Specified Changes in the Illuminant," *Color Research and Application*, vol. 1, no. 1, pp. 17–21, 1976.

[103] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. John Wiley and Sons, 1973.

[104] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 8, pp. 62–66, 1978.

[105] Y. Bérubé Lauzière, D. Gingras, and F. P. Ferrie, "Bit Encoded Labelling and Quasi-Invariant Spatial Pattern Matching," *Proceedings of the 4th Asian Conference on Computer Vision, Taipei, Taiwan*, pp. 759–764, January 2000.

[106] R. Gnanadesikan and J. Kettenring, "Robust Estimates, Residuals, and Outlier Detection with Multiresponse Data," *Biometrics*, vol. 28, pp. 81–124, March 1972.

[107] N. Campbell, "Robust Procedures in Multivariate Analysis I: Robust Covariance Estimation," *Applied Statistics*, vol. 29, no. 3, pp. 231–237, 1980.

[108] S. Devlin, R. Gnanadesikan, and J. Kettenring, "Robust Estimation of Dispersion Matrices and Principal Components," *Journal of the American Statistical Association*, vol. 76, pp. 354–362, June 1981.

[109] Y. Gofman, "Outline of a Set of Points," *Pattern Recognition Letters*, vol. 14, pp. 31–38, 1993.

[110] A. R. Chaudhuri, B. B. Chaudhuri, and S. K. Parui, "A Novel Approach to Computation of the Shape of a Dot Pattern and Extraction of its Perceptual Border," *Computer Vision and Image Understanding*, vol. 68, pp. 257–275, December 1997.

[111] G. T. Toussaint, "A Graph-Theoretical Primal Sketch," in *Computational Morphology* (G. T. Toussaint, ed.), pp. 229–260, North Holland, Amsterdam, 1988.

[112] H. Edelsbrunner, D. G. Kirkpatrick, and R. Seidel, "On the Shape of a Set of Points in the Plane," *IEEE Transactions on Information Theory*, vol. 29, pp. 551–559, July 1983.

[113] H. H. Arsenault, "Distortion-Invariant Pattern Recognition Using Circular Harmonic Matched Filters," in *Optical Processing and Computing* (H. H. Arsenault, T. Szoplik, and B. Macukow, eds.), ch. 10, pp. 315–342, Academic Press, 1989.

[114] S. Chang, S. Boothroyd, P. Palacharla, and S. Pachanathan, "Rotation-Invariant Pattern Recognition Using a Joint Transform Correlator," *Optics Communications*, vol. 127, pp. 107–116, 1996.

[115] A. Rosenfeld, *Picture Processing by Computer*. Academic Press, 1969.

[116] P. Aschwanden and W. Guggenbühl, "Experimental Results from a Comparative Study on Correlation-Type Registration Algorithms," in *Robust Computer Vision: Quality of Vision Algorithms* (Forstaer and Ruwiedel, eds.), pp. 268–289, Winchmann, 1992.

[117] R. Brunelli and T. Poggio, "Template matching: Matched spatial filters and beyond," *Pattern Recognition*, vol. 30, pp. 751–768, 1997.

[118] J. W. Goodman, *Introduction to Fourier Optics*. McGraw-Hill, 1996.

[119] J. Wood, "Invariant Pattern Recognition: A Review," *Pattern Recognition*, vol. 29, no. 1, pp. 1–17, 1996.

[120] D. Casasent and D. Psaltis, "Scale Invariant Optical Correlation Using Mellin Transforms," *Optics Communications*, vol. 17, no. 1, pp. 59–63, 1976.

[121] D. Casasent and D. Psaltis, "Scale Invariant Optical Transform," *Optical Engineering*, vol. 15, no. 3, pp. 258–261, 1976.

[122] D. Casasent and D. Psaltis, "Position, Rotation, and Scale Invariant Optical Correlation," *Applied Optics*, vol. 15, no. 7, pp. 1795–1799, 1976.

[123] M. Ferraro and T. M. Caelli, "Relationship between Integral Transform Invariances and Lie Group Theory," *Journal of the Optical Society of America A*, vol. 5, pp. 738–742, 1988.

[124] Y. Sheng and H. H. Arsenault, "Experiments on Pattern Recognition Using Invariant Fourier-Mellin Descriptors," *Journal of the Optical Society of America A*, vol. 3, no. 6, pp. 771–776, 1986.

[125] D. Mendlovic, Z. Zalevsky, I. Kiryuschev, and G. Lebreton, "Composite Harmonic Filters for Scale-, Projection-, and Shift-Invariant Pattern Recognition," *Applied Optics*, vol. 34, no. 2, pp. 310–316, 1995.

[126] B. V. Kumar, "Tutorial Survey of Composite Filter Designs for Optical Correlators," *Applied Optics*, vol. 31, pp. 4773–4801, August 1992.

[127] B. W. Kernighan and D. M. Ritchie, *The C Programming Language*. Prentice-Hall, 2 ed., 1988.

[128] B. Jähne, *Image Processing for Scientific Applications*. CRC Press, 1997.

[129] Y. Bérubé Lauzière, D. Gingras, and F. P. Ferrie, "A Model-Based Road Sign Identification System," *Accepted for presentation at CVPR 2001, Dec. 2001, Hawaii, U.S.A.*, 2001.

[130] O. Faugeras, *Three-Dimensional Computer Vision*. MIT Press, 1993.

[131] N. Strachan, P. Nesvabda, and A. Allen, "Calibration of a Video Camera Digitising System in the CIE $L^*u^*v^*$ Colour Space," *Pattern Recognition Letters*, vol. 11, pp. 771–777, November 1990.

[132] P. L. Vora, J. E. Farrell, J. D. Tietz, and D. H. Brainard, "Digital Color Cameras-2-Spectral Response," Tech. Rep. HPL-97-54, Hewlett-Packard Laboratory, 1997.

[133] K. Barnard and B. V. Funt, "Camera Calibration for Color Research," *Electronic Imaging IV, SPIE Proceedings*, vol. 3644, 1999.

[134] F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg, and T. Limperis, "Geometrical Considerations and Nomenclature for Reflectance," *National Bureau of Standards Monograph 160*, 1977.

[135] B. K. P. Horn and R. W. Sjoberg, "Calculating the Reflectance Map," *Applied Optics*, vol. 18, pp. 1770–1779, June 1979.

[136] Ministère des Transports du Québec, *Le répertoire de la signalisation routière au Québec*, 1996.

[137] U.S Department of Transportation, Federal Highway Administration, Washington D.C., *Manual on Uniform Traffic Control Devices (MUTCD)*, 1988.