UNIVERSITY OF BELGRADE

SCHOOL OF ELECTRICAL ENGINEERING

Ahmed Mohamed Goma

The algorithm of a road region extraction from digital image based on usage of texture descriptors

Doctoral Dissertation

Belgrade, 2013

УНИВЕРЗИТЕТ У БЕОГРАДУ

Електротехнички факултет

Ахмед Мохамед Џума

Алгоритам издвајање региона пута из дигиталне ТВ слике коришћењем дескриптора текстуре

Докторска дисертација

Београд, 2013

Abstract

Over the last decades many techniques for the analysis of textured images have been proposed in the literature for the texture classification and segmentation, but the approach in this dissertation challenges most of obstacles in this field. The main goal for this approach was to implement a fast and active texture classification approach based on texture features to have the ability to extract road boarders from the contents of an image produced by camera is mounted on a moving vehicle. This approach analyzes the properties of individual texture elements, using statistical and spatial structural features obtained from the gray level values of each particular area in the input image. Thus, in the field of outdoor usage of wheeled robotics and autonomous vehicles, it's important to extract the road boarders to could guide them in right direction and moreover, enabling further calculations of distances from the borders as well as road directions, etc.
Explores of this approach, thus, the texture of a road is different enough in comparison to the textures characterizing the surrounding environment, allowing the separation of the overall image into few distinguishable regions. Being oriented toward autonomous car guidance in real time, the time consumption for the algorithm implementation was the important factor. That was the reason why the number of classes is limited to three (road, near background, far background) in this algorithm. To accomplish the improvements in the extracting of road class

from the non-road, the segmentation algorithm combines the texture descriptors of statistical nature and the ones based on gray level co-occurrence matrix as features classifications tools. This algorithm is selecting the best descriptor candidates from each group by calculating the seed square size 64*64 for each class selected by the operator in representative positions. In order to decrease the computation time, and to improve the quality from statistical point of view, the algorithm included the acquisition of 1024 values by moving the square around the seed points and making the calculations only for selected best descriptors. The classification thresholds both in two-dimensional case as well as in one-dimensional case, are found based on statistical distributions approximated as Gaussian ones (mean, standard deviation) for each class. Thresholding is done either by using minimum Euclidean distance or by appropriate adaptation of this threshold. The overall input image is scanned horizontally by the square of size 64*64 pixels and segmented with specified texture descriptors into three classes. The most important class for our algorithm is the road class. The segmented result is crude image that is refined in its shape by using splitting and merging process only around the road border with other two segmentation passes with down square sizes 32*32 and 16*16 pixels. After all three segmentation passes, the result of road border as a staircase line. The algorithm will refine this staircase border by linear fitting approximation in order to obtain the straight line borders. Finally, check of the robustness of this algorithm, and challenges introduced by some obstacles (naturally or artificially) on the road have been verified by applying this algorithm on different image examples.

Абстракт:

У последњим деценијама је у литератури предложено више техника за анализу текстурисаних слика и то за класификацију текстура, сегментацију слике на регионе и сл., а у приступу који је изложен у овој дисертацији сусреће се са већином изазова и препрека типичних за ову област. Основни циљ изложеног прилаза састоји се у дефинисању и примени релативно брзе процедуре за издвајање региона пута из слике прибављене из камере монтиране на покретном возилу, а на бази класификације текстура на основу одговарајућих дескриптора. У питању је анализа особина појединачних елемената текстуре у различитим деловима улазне слике и то по основу статистичких показатеља везаних за хистограм нијанси сивог и просторних структурних показатеља везаних за матрице заједничког појављивања нивоа сивог. У области аутоматског управљања објеката као што су моторизовани роботи са точковима или аутоматизована возила на путу, у условима отвореног простора, важно је издвојити границе пута по коме се објекат креће да би се прилагодиле управљачке акције, прорачунала растојања до граница пута, одредило усмерење пута итд.

У основи прилаза у овој дисертацији претпоставња се да је текстура лика пута довољно различита односу на текстуре које карактеришу околину, тако да је могуће сегментисати слику на више карактеристичних региона. Како је крајњи циљ да се дође до алгоритма примењивог у реалном времену где је

брзина обављања рачунских операција критична, предложено је да се број класа текстура ограничи на три (пут, блиска околина и далека околина). Да би се усавршио поступак издвајања пута од онога што није пут, алгоритам сегментације комбинује дескрипторе који су статистичке пpирпде и оне који су стуктурне, тако да је класификација дводимензионог каркатера. Као два основна показатеља селектују се најбољи кандидати из обе групе, тако што се сви потенцијални дескриптори иницијално израчунају за почетне прозоре димензија 64*64 пиксела, постављене у области за које се априорно претпоставља да су репрезенти предвиђених класа. Да би се прибавила валидна статистичка информација, за најбоље дескрипторе по групама, врши се аквизиција у 1024 прозора у околини иницијалног. Прагови за класификацију се по обе димензије рачунају на бази претпоставке да је расподела вредности дескриптора у популацији од 1024 прозора Гаусовска, са датим средњим вредностима и стандардним девијацијама. Сама сегментација се врши или на бази минималног Еуклидског растојања од прототипских вредности класа или на бази оригинално модификованог прага на основу односа стандардних девијација вредности дескриптора по класама. Након почетне сегментације у три класе која се постиже скенирањем слике са прозорима димензија 64*64 пиксела, следи поправка квалитета сегментације дуж границе раздвајања региона пута од околине која има смисао сегментације поделом и растом региона. Нова два скенирања се врше само у граничној области и то са прозорима димензија 32*32 пиксела и 16*16 пиксела. Након сва три корака сегментације примењује се процедура морфолошког филтрирања које има смисао попуњавања празнина и повезивања целина. Коначно, граница пута се издваја као степенаста линија, тако да је завршни корак одговарајућа апроксимација правом линијом.

Испитивање робусности алгоритма је извршено за низ карактеристичних слика које су укључиле проблеме са сенкама и одсјајима на путу, слабо дефинисаним сеоским путевима, „превише“ дефинисаним аутострадама са низом већ постојећи демаркационих елемената, примерима са више потенцијалних класа у околини, али и на самом путу, са природним или вештачким препрекама које нарушавају претпоставку да се региони могу класификовати на бази јединствене текстуре итд. Такође је у завршном делу рада анализирана и могућност да се као „трећа димензија“ у класификацији разматра и садржај плаве компоненте у колор слици.

Keywords :

Autonomous vehicles, Digital image processing, Image segmentation, Texture, Texture classification.

Кључне речи:

Аутономна возила, Процесирање дигиталне слике, Сегментација слике, Текстура, Класификација текстура.

Acknowledgment:

Foremost, I would like to express my sincere gratitude to my advisors Prof. Dr Miodrag Popović, and Prof. Dr Stevica Graovac for the continuous support of my Ph.D study and research, especially, Prof. Graovac for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me during the all time of my research and writing of this thesis.

My sincere thanks to Faculty dean Prof. Dr Branko Kovačević and I will also thank chief of office of student affairs Miss Dragana Trenevski. Besides my advisor, I would like to thank my thesis committee.

My sincere thanks also go to all Innovation center people, Director of the Innovation center Prof. Dr Branimir Reljin, former vice director Prof. Dr Dušan Drajić, and research assistant M.Sc. Ilija Radovanović.

I thank my fellow Libyan and Serbian students for the stimulating discussions and their co-operations.

Last but not the least; I would like to thank my family: my wife and Mother, for giving birth to me at the first place and supporting me spiritually throughout my life.

Contents:

1. Introduction:

In the last two decades, in the field of computer vision plays an important role in low level image analysis and understanding. The computer vision tackles the problem of engineering artificial visual systems capable of somehow comprehending and interpreting our real, three dimensional world. There are many applications in this field includes object recognition, motion interpretation (tracking), autonomous navigation, and the robotic manipulation. Since computer vision has its roots in artificial intelligence, many methods were originally developed to either tackle or represent a problem in computer vision. The fields still have much in common today, especially using texture classification to perform some computer vision applications. Its potential applications range has been shown in various areas such as autonomous guidance of ground vehicles (wheeled robots, cars), remote sensing images, and industrial automation, biomedical image processing etc. The possible applications span from the cases where the vehicle is to move autonomously along very well arranged highways with a number of distinguishable landmarks around, up to the cases where the object is moving through the completely unknown environment, along the road/path poorly separable from the local background. As a reasonable step toward fully autonomous guidance, a number of applications that have the nature of "driving assistant" are envisaged. A number of heterogeneous sensors and systems are used in the context of "intelligent transportation systems". The forward looking TV camera is one of the most popular and useful ones. The digitized image coming from camera is a source of valuable information related to the task of recognition of a road/lane and its borders, detection of other subjects/obstacles on the road (other vehicles, pedestrians), detection and recognition of traffic signs, lights, etc. We have limited our attention here toward the task of separation of the road region, or more precisely, extraction of the border between the road and the surrounding background. This part of image is important to be known in order to calculate vehicle's distance from the border, to estimate the radius of a road curvature, as well as to estimate the heading and slope of a road (in cooperation with some other inertial or magnetic instruments, or a priory known data).[1]

The idea behind this algorithm extracts the road borders from image has two or more regions in an image representing the scene in front of the vehicle, (the road being one of them and the main task) there are two basic approaches: 1-Borders between regions are characterized by the appreciable light intensity variations and in order to extract these pixels some previous high pass filtration is preferable, after which the appropriate segmentation procedure is going to separate the points where the gradient values are above the specified level (threshold). 2-The region of road can be characterized using the range of values of some descriptor(s), such that the other regions around, are well distinguishable from, and the road border is extractable this way. These descriptors can be related to the light intensity, color, or different descriptions of a texture.

The usage of texture descriptors is suggested in this dissertation for defined type of task. One type of these descriptors (statistical ones) are related to the light intensity content typical for some parts of the road image, and generally is based on statistical moments calculated for the population of pixels belonging to it. The other type that was relevant for this approach is based on light intensity co-occurrence matrix and introduces the information about spatial relationships between pixels with different light intensities inside the road area. Moreover, none of these descriptors is universally applicable for all types of roads and local backgrounds. The idea is to make the segmentation algorithm to be semiautomatic in the meaning that the simple initial acquisition of data is made by human operator (driver), while the remaining procedure if fully automatic. As a first part of this procedure, the choice of best candidates among descriptors of both types is done. The threshold values between classes (road, near background, far background) are found by minimization of the erroneous classification. The classification is extended to two dimensions in order to obtain balance between statistical and spatial texture characteristics, due to the fact that none of regions is fully uniform regarding to any descriptor. As a result of intention that the algorithm should finally work in real time, very precise calculation of all points belonging to the border (pixel-based texture classification) is avoided. Instead of that, initial scanning with the window of 64*64 size is followed by the appropriate splitting down to the windows of sizes 32*32 and 16*16, around the rudimentary estimated border. Appropriate morphological filtration based on the assumption that the road region should be connected is the next step, while the whole procedure on one frame is finishing by polynomial fitting through the "stairs-like" border segmented up to this point. Figure 1.1 describes those steps used in this algorithm.

The inspiration for this research work has been found in some of relevant texture based descriptors, and of basic pattern recognition relevant for the image processing applications are given in textbooks [1, 3,5,6]. The scope of methods used in computer vision applied for the control of intelligent road vehicles is given in [4], while the more general survey of usage the vision for autonomous guidance was done in [7], and, specifically, for the purposes of mobile robot navigation, in

[8]. Fundamentals of image segmentation, Articles [9-13] are representatives that are discussing the task of a road border extraction using different methods (optical flow measurements, edge detection followed by Hough transform, segmentation based on color contents, color feature combined with line and road following, texture description alone, and texture combined with color). In [14-16], for the same task, the focus is shifted more toward the proper road modeling (border is modeled as a straight or parabolic line, both for in-door and out-door applications, or, lines are detected in pairs – which is typical for urban traffic environment). In the papers [17,18] the authors go further by including additional devices like odometer or inertial instruments in order to combine the information supplied by them with visual tracking process, and to provide the final information about vehicle's position/velocity on the road. Articles [19-20] are referenced here basically as the examples of road segmentation algorithms oriented toward real-time applications. Description of textures and their classification was of particular interest in this paper. The authors of papers [21-24] have discussed this problem from different aspects relevant for our approach (how to reduce the time for recognizing the texture as one out of predefined set, how to make the acquisition over multiple windows of different size, how to find the subset of relevant features that are sufficient to capture the important properties of the set of classes that are to be distinguished, and how to make so called pixel-based texture classification which determines the class which every pixel of an input image belongs to). Finally, the articles [24-27] are oriented toward combinations of usage of texture descriptors based on co-occurrence matrix and other features like color and variance, as well as to classifications based on combinations of light intensity and color with texture.

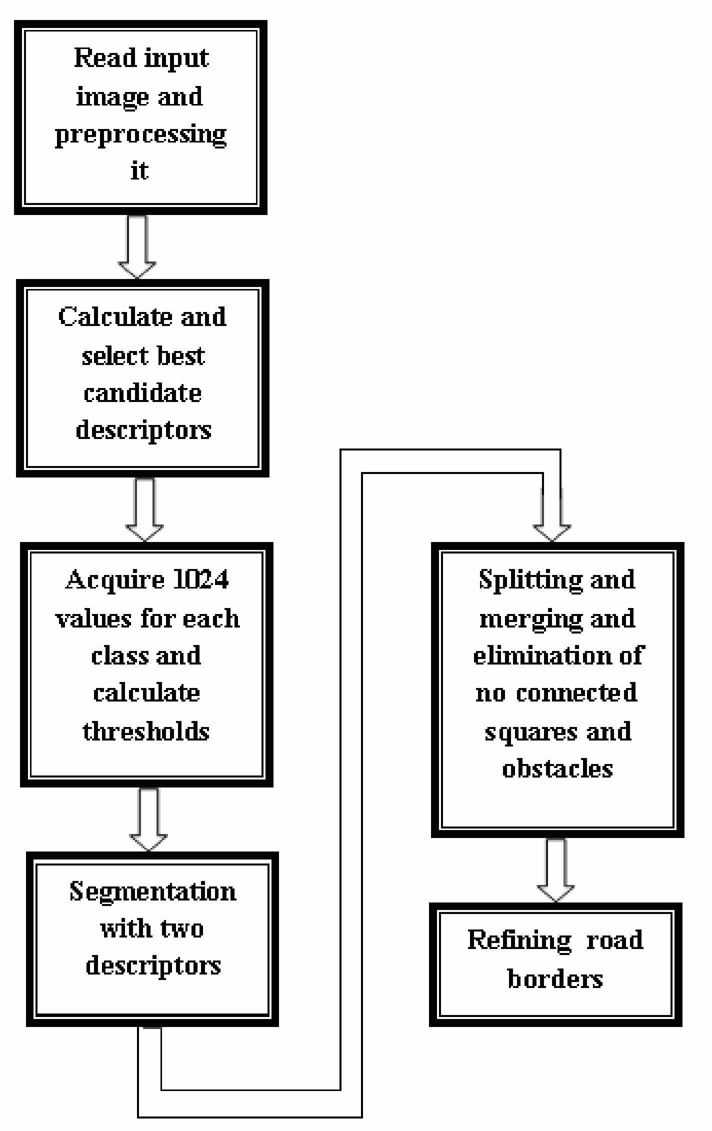


Figure 1.1 Algorithm block diagram

In this dissertation organized as following: chapter 2 will introduce the image, digital image formation, and its types, applications. Chapter 3 shows the digital image enhancement used in this algorithm. Chapter 4 discusses the texture and its classifications and related work available. Chapter 5 represents the algorithm to detect road image borders based on texture classification.

Chapter 6 discusses the algorithm verifications with different input images and texture classifications in two-dimensional descriptors space and combined with blue color.

Objectives of this work have been:

Extracting the road borders used texture features classifications from static image.

1. To step on real time applications, automatic choice of the best candidates for descriptors – the most suitable ones for the actual scene only at begging of the algorithm starting.
2. In order to use Gaussian approximation to calculate the thresholds using the basic training to via acquisition of statistical information related to descriptor values.
3. Reducing the multidimensional classification task to one-dimensional one; in order to calculate the thresholds.
4. To accomplish the road borders extractions from "low-resolution" approach in splitting/merging type of image segmentation used fitting line approximations.
5. The results of this work and its objectives will be closer to real time applications.

2. Digital Image Fundamentals.

2.1 Introduction:

Image processing is a rapidly growing area of computer science. Technological advances in digital imaging, computer processors and mass storage devices have fueled its growth. Fields, which traditionally used analog imaging, and now switching to digital images, for their flexibility and affordability.

2.2 Digital image

In general, the divination of the image is the optical counterpart of an object produced by an optical device (as a lens or mirror) or an electronic device.
In digital fields the digital image may be defined as two-dimensional function *f(x, y)* where *x, y* are special coordinate and *f* is the amplitude at any pair of coordinate *(x, y)* called intensity or gray level for all *x, y*, and *f* are finite and discrete quantities.[1]

2.3 Pixel

The word pixel is an abbreviation of 'picture element'. Indexed as an *(x,y)*location from the upper left of the image, it represents the smallest, constituent element in a digital image and contains a numerical value which is the basic unit of information within the image at a given spatial resolution and quantization level. Commonly, pixels contain the color or intensity response of the image as a small point sample of colored light from the scene. However, not all images necessarily contain strictly visual information. An image is simply a 2-D signal digitized as a grid of pixels, the values of which may relate to other properties other than color or light intensity. The information content of pixels can vary considerably depending on the type of image we are processing.

2.4 Image foundation

Firstly**,** to produce the analog image, this done by reflecting light energy or other energy sources. Light is the predominant energy source for producing images; simply because it is the energy source which human beings can observe directly. We are all familiar with photographs, which are a pictorial record of a visual scene. Many digital images are captured using visible light as the energy source; this has the advantage of being safe, cheap, easily detected and readily processed with suitable hardware. Two very popular methods of producing a digital image are with a digital camera or a flat-bed scanner, etc can obtain the digital image. This image contents are generated by combination of an illumination source and the reflection or absorption of energy from that source by the elements of the scene being imaging. The illumination can be originating from different electromagnetic energy sources, ultrasound or even a computer-generated illumination pattern. Depending on the nature of the source, illumination energy is reflected from, or transmitted through, objects. The internal cameras sensors creates the two dimension function in x and y plane and this function $f(x, y)$ is result from *illumination * reflectance object* [1].

CCD camera. Such a camera has, in place of the usual film, an array of photo sites; these are silicon electronic devices whose voltage output is proportional to the intensity of light falling on them. This is shown schematically in figure 2.1. The output will be an array of values; each representing a sampled point from the original scene. The elements of this array are called picture elements, or more simply pixels.

Scanner works on a principle similar to the CCD camera. Instead of the entire image being captured at once on a large array, a single row of photo sites is moved across the image, capturing it row-by-row as it moves. This is shown schematically in figure 2.2.

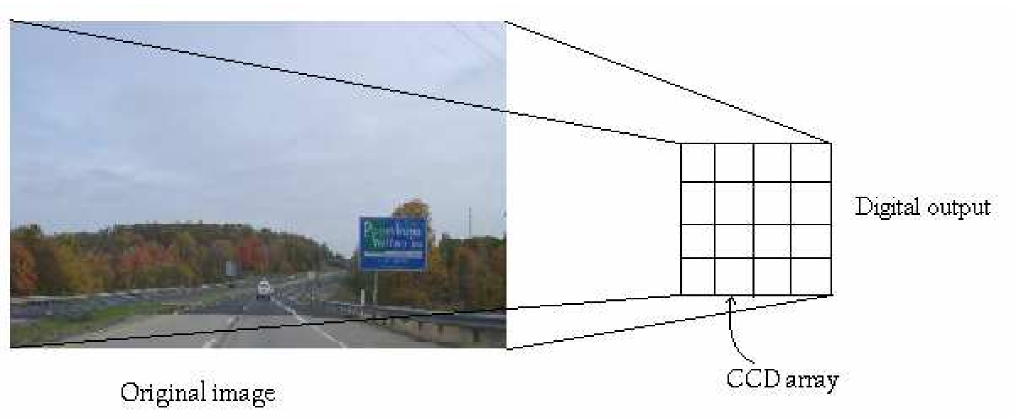


Figure 2.1 capturing an image with CCD camera.

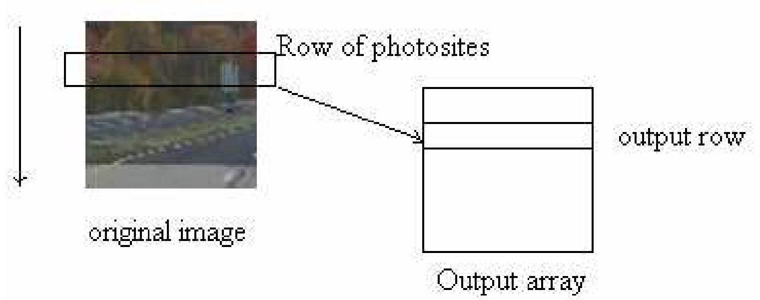


Figure 2.2 capturing an image with CCD scanner.

Although light is popular and easy to use, other energy sources may be used to create a digital image. Visible light is part of the electromagnetic spectrum: radiation in which the energy takes the form of waves of varying wavelength. These range from cosmic rays of very short wavelength, to electric power, which has very long wavelength. Figure 2.3 illustrates electromagnetic wave lengths.

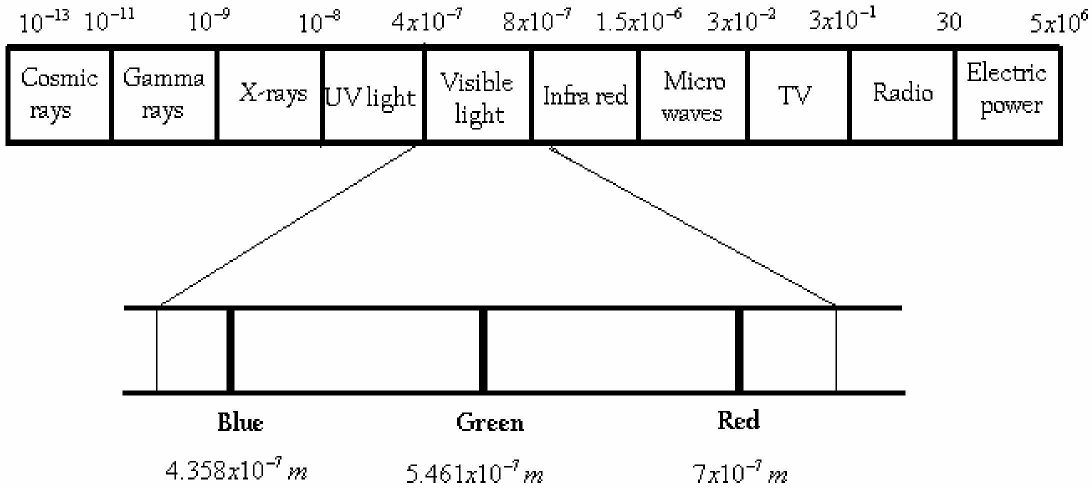


Figure 2.3 the electromagnetic spectrum.

2. 5 Image sampling and quantization

The object of this section how to generate digital image from numerous of ways to acquire input images. The output of most sensors is a continuous voltage waveform whose amplitude and spatial behavior are related to the physical phenomenon being sensed. To create a digital image, we need to convert the continuous sensed data into digital form. This involves two processes: *sampling* and *quantization*[2].

The basic idea behind sampling and quantization is illustrated in figure 2.4 upper left Figure shows a continuous image, $f(x, y)$, that we want to converted to digital form. An image may be continuous with respect to the x and y coordinates, and in amplitude. To convert it to digital form, we have to sample the function in both coordinates and in amplitude. Digitizing the coordinate values is called *sampling*. Digitizing the amplitude values is called *quantization*.

One-dimensional function shown in the upper right figure is a plot of amplitude (gray level) values of the continuous image along the line segment AB in left upper figure .The random variations are due to image noise. To sample this function, we take equally spaced samples along line AB, as shown in left lower figure. The location of each sample is given by a vertical tick mark in the bottom part of the figure. The samples are shown as small white squares superimposed on the function.

The set of these discrete locations gives the sampled function. However, the values of the samples still span (vertically) a continuous range of gray-level values. In order to form a digital function, the gray-level values also must be converted (*quantized*) into discrete quantities. The lower left figure the gray-level scale divided into eight discrete levels, ranging from black to white. The vertical tick marks indicate the specific value assigned to each of the eight gray levels. The continuous gray levels are quantized simply by assigning one of the eight discrete gray levels to each sample. The assignment is made depending on the vertical proximity of a sample to a vertical tick mark. The digital samples resulting from both sampling and quantization are shown in right lower figure. Starting at the top of the image and carrying out this procedure line by line produces a two-dimensional digital image and we should know the quality of digital image depends on the number of samples and discrete gray levels used in sampling and quantization, image contents consideration to select these parameters[1,2], as shown the figure 2.5

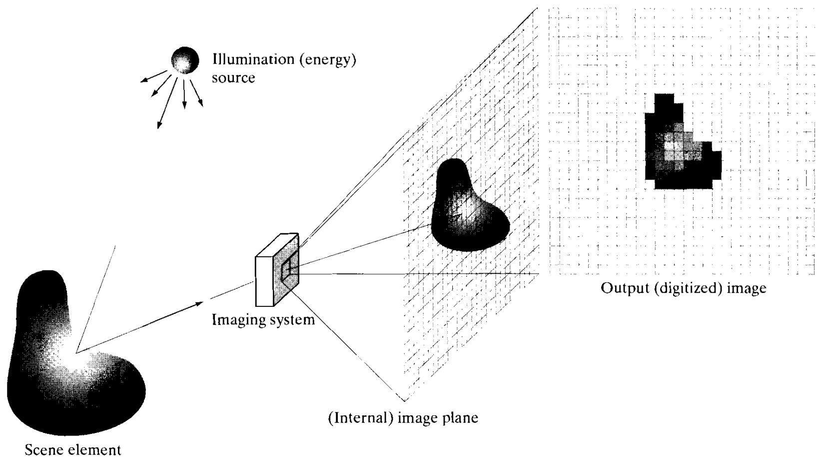


Figure 2.4 an example of the digital image acquisition process

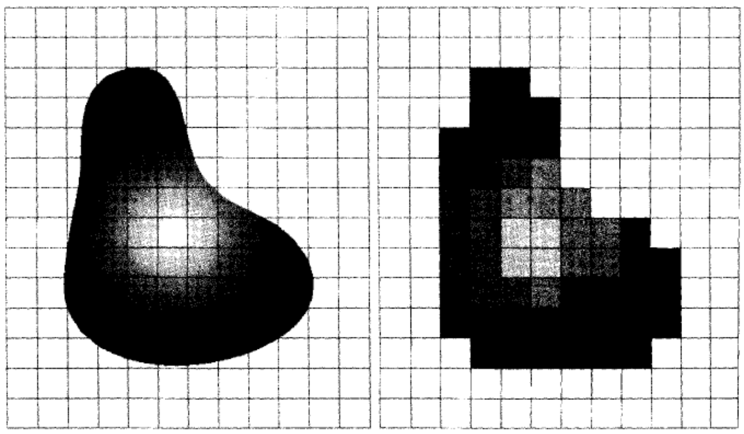


Figure 2.5 Continuous image and its sampling and quantizing result.

2.6 Representing Digital Image

The result of image quantization and sampling is two-dimension matrix of real numbers with *M* rows and *N* columns of function *f(x, y)*. Each point or pixel in this function can be reached through its index value x and y (coordinate). This function can be defined as:

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N_y-1) \\ f(0,0) & f(1,1) & \cdots & f(1,N_y-1) \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ f(M_x-1,0) & f(M_x-1,1) & \cdots & f(M_x-1,N_y-1) \end{bmatrix} \qquad (2.1)$$

2.6.1 Binary image:

This is the simplest type of image with two gray-values, 0 and 1 or black and white and each pixel represented by a single bit. These types of images are useful in computer vision applications where only information about images or outlines are required. It can be created from gray-scale image that uses 0 for pixels with gray levels below the threshold and 1 for other pixel but this way of creation is not useful because most of the information is lost and the image result are smaller[1,3]. In this image in figure 2.6, we have only the two colors: white for the edges, and black for the background.

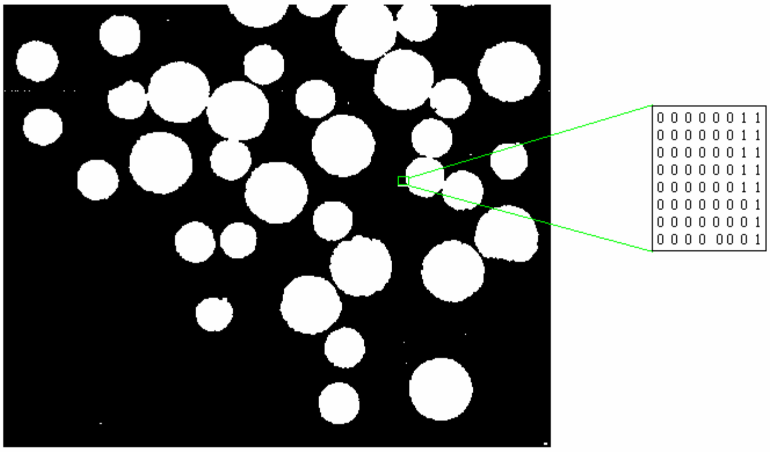


Figure 2.6 a binary image.

2.6.2 Gray-scale image:

These images contain the brightness information. The numbers of bits that are used to represent each pixel are related to the number of different brightness levels available. The typical image contains 8 bits per pixel so there are 256 different possible gray-tones or intensity values from 0 to 255. This range means that each pixel can be represented by eight bits, or exactly one byte. This is a very natural range for image file handling. Other grayscale ranges are used, but generally they are a power of 2. Such images arise in robotics application, images of printed works, and indeed 256 different grey levels are sufficient for the recognition of most natural objects.

Figure 2.7 shows example for grayscale road image.

138	142	131	139	135	131	133	135
139	133	130	132	140	136	143	143
145	130	139	132	150	144	143	141
149	136	144	134	150	143	139	135
148	147	138	140	143	141	149	142
148	158	131	147	140	145	147	141
143	142	155	150	142	150	142	145
146	145	152	150	145	152	140	138

Figure 2.7 Grayscale road image

2.6.3 Color image:

A color image is made up of pixels each of which holds three numbers corresponding to the red, green, and blue levels of the image at a particular location. Red, green, and blue (sometimes referred to as RGB) are the primary colors for mixing light. Color is a vectorial feature assigned to each pixel in a color image. Normally, images are represented as RGB (Red, Green, and Blue) models, and each pixel has 24 bits. The brightness information, color information are coupled and represented in many applications. The use of digital image processing color images is powerful descriptor that often simplified object identification and extraction from image like, in autonomous robot applications [1, 3]. Figure 2.8 shows an example for color image.

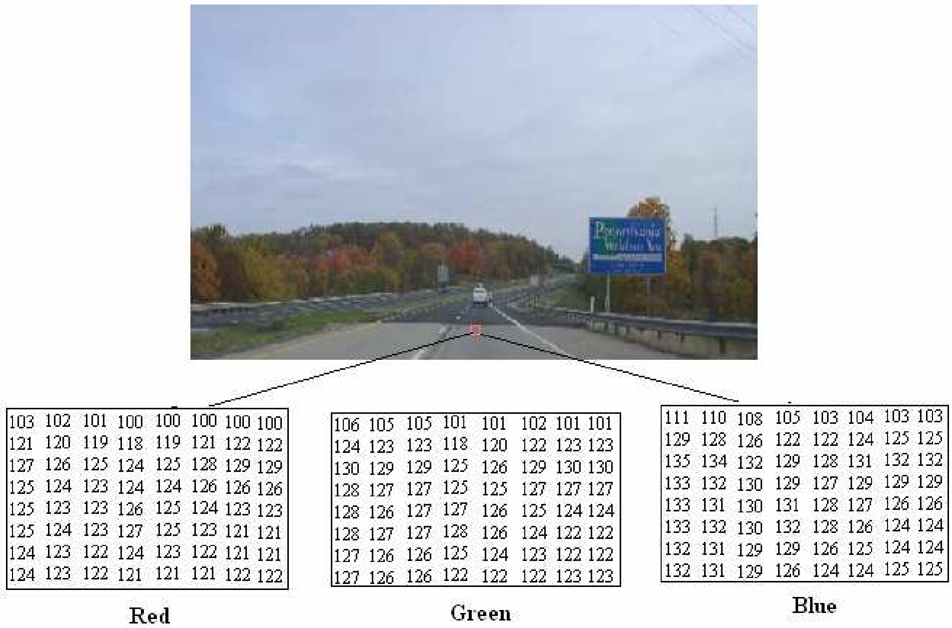


Figure 2.8 Color image.

2.7 Image Analysis

Image analysis considers the image and it is useful in many applications such as computer vision, medical imaging, and pattern recognition. It extracts the useful information from the image. The image analysis involves image segmentation and feature extraction.

Image segmentation: Image segmentation divides the input image into multiple segments or regions, which show objects or meaningful parts of objects. It segments image into homogenous regions thus making it easier to analyze images.

Feature extraction: Large amounts of data are used to represent an image, so the analysis of an image needs a large amount of memory and thus takes more time. In order to reduce the amount of data, an image is represented using a set of features.

Feature extraction is a primitive type of pattern recognition and it is very important for pattern recognition. This step extracts some features such as color, shape, and texture. Features contain the relevant information of an image and will be used in image processing (e.g. searching, retrieval, storing). It decomposes into two parts, feature construction and feature selection.

2.8 Image file format:

There are different formats for storing the pixels of a digital image. Because we are going to modify the contents of digital images to specific file format in our algorithm, it is very important to know which file format to use, since some formats are lossy while some other has less distortion, digital image format has different format like (JPG, GIF, PNG, TIFF, …etc). In this algorithm will use JPG format JPG, or JPEG (for Joint Photographic Experts Group), because it's the most common file format in digital photography. It supports a color depth of 24 bits (3 color channels of 8 bits each). Almost every digital camera is able to write images using this format and it is widely supported in different viewing programs. It produces small file sizes using lossy compression [28].

3. Digital Image Improvements.

3.1 Introduction

The usage of digital processing techniques for image improvements has received much interest with the publicity given to applications in space imagery, autonomous control systems and robotics research, etc. Other applications include image improvement for photographic surveys and industrial radiographic analysis.

One can divide image improvement into two broad categories: the first one where the inputs and outputs are images, and second one where the inputs may be images but whose outputs are attributed from those images. *Image enhancement* entails operations that improve the appearance to a human viewer, or operations to convert an image to a format better suited to machine processing. One of image enhancement procedures processed in spatial domain methods. The term *spatial domain* refers to the image plane itself, and approaches in this category are based on direct manipulation of pixels in an image, like gray level histograms manipulation other some preprocessing approaches. In this work for road boarders detection used some spatial image processing functions and second categories, whose the input is image but the output is attributes from the input image. This type of processing, *Image Segmentation* procedures partition an image into its constituent parts or objects using some threshold to do that, this work used texture descriptors as classification tools to discriminates the road from non road images. In general, autonomous segmentation is one of the most difficult tasks in digital image processing. A rugged segmentation procedure brings the process a long way toward successful solution of imaging problems that require objects to be identified individually. On the other hand, weak or erratic segmentation algorithms almost always guarantee eventual failure. In general, the more accurate the segmentation, the more likely recognition is to succeed. *Morphological functions* its a tools for extracting some image components that are useful in the representation and description, and also some kind of image enhancement but in binary images only. *Representation and description* usually follow the output of a segmentation stage after morphologic functions, which usually are raw pixel data, constituting either the boundary of a region (i.e., the set of pixels separating one image region from another) or all the points in the region itself. In either case, converting the data to a form suitable for computer processing is necessary. The first decision that must be made whether, the data should be represented as a boundary or as a complete region. Boundary representation is appropriate when the focus is on external shape characteristics, such as corners and inflections. Regional representation is appropriate when the focus is on internal properties, such as texture or skeletal shape, which this representation are applied in our algorithm,. In some applications, these representations complement each other. Choosing a representation is only part of the solution for transforming raw data into a form suitable for subsequent computer processing. A method must also be specified for describing the data so that features of interest are highlighted. *Description*, also called *feature selection*, deals with extracting attributes that result in some quantitative information of interest or are basic for differentiating one class of objects from another. *Recognition* is the process that assigns a label (e.g., "vehicle") to an object based on its descriptors. we conclude our coverage of digital image processing with the development of methods for recognition of individual objects[1].

3.2 Histogram Processing:

A histogram plots the relative frequency of each pixel value that occurs in a grey scale image. Now we can define the histogram of a digital image with gray levels in the range $[0\ , L-1]$ is a discrete function $h(r_k) = n_k$, where r_k is the k_{th} gray level and n_k is the number of pixels in the image having gray level r_k. Histogram is the basis for numerous spatial domain-processing techniques, and can be used effectively for image enhancement. The histogram provides a convenient summary of the intensities in an image, but is unable to convey any information regarding spatial relationships between pixels. The information inherent in histograms also is quite useful in other image processing applications as in calculation of statistical texture features. Histograms are simple to calculate in software and lend themselves to economic hardware implementations, thus making them a popular tool for real-time image processing. The following example in figure 3.1 shows histograms for road image (road, background, and sky) sub- images with different contrasts. The horizontal axis of each histogram plot corresponds to gray level values, r_k . The vertical axis corresponds to values of $h(r_k) = n_k$ or $p(r_k) = n_k / n$ if the values are normalized. These histogram plots are simply plots of $h(r_k) = n_k$ versus r_k or $p(r_k) = n_k / n$ versus r_k.

3.3 Image Segmentation

In this type of processing began a transition from image processing methods whose input and output are images, to methods in which inputs are images, but the outputs are attributes extracted from those images.

Segmentation an image consist of subdividing it into a number of non _overlapping, homogeneous regions used some criteria, each of which is spatially connected and differs from neighboring regions in some meaningful properties .Segmentation is a process of partitioning the image into some non-intersecting regions that each region is homogeneous and the union of no two adjacent regions is homogeneous. The segmentation will stop when the objects of interest in an application have been isolated like extractions of road image from non-road images, automated control quality inspection, etc. Segmentation accuracy determines the eventual success or failure of computerized analysis procedures. For this reason, considerable care should be taken to improve the probability of rugged segmentation.

Image segmentation algorithms generally are based on one of two basic properties of intensity values: discontinuity and similarity. In the first category, the approach is to partition an image based on abrupt changes in intensity, such as edges in an image. The principle approaches in the second category are based on partitioning an image into regions that are similar according to a set of predefined criteria like (thresholding, region growing, and region splitting and merging) and this type of segmentation will be used in my algorithm.[1,29]

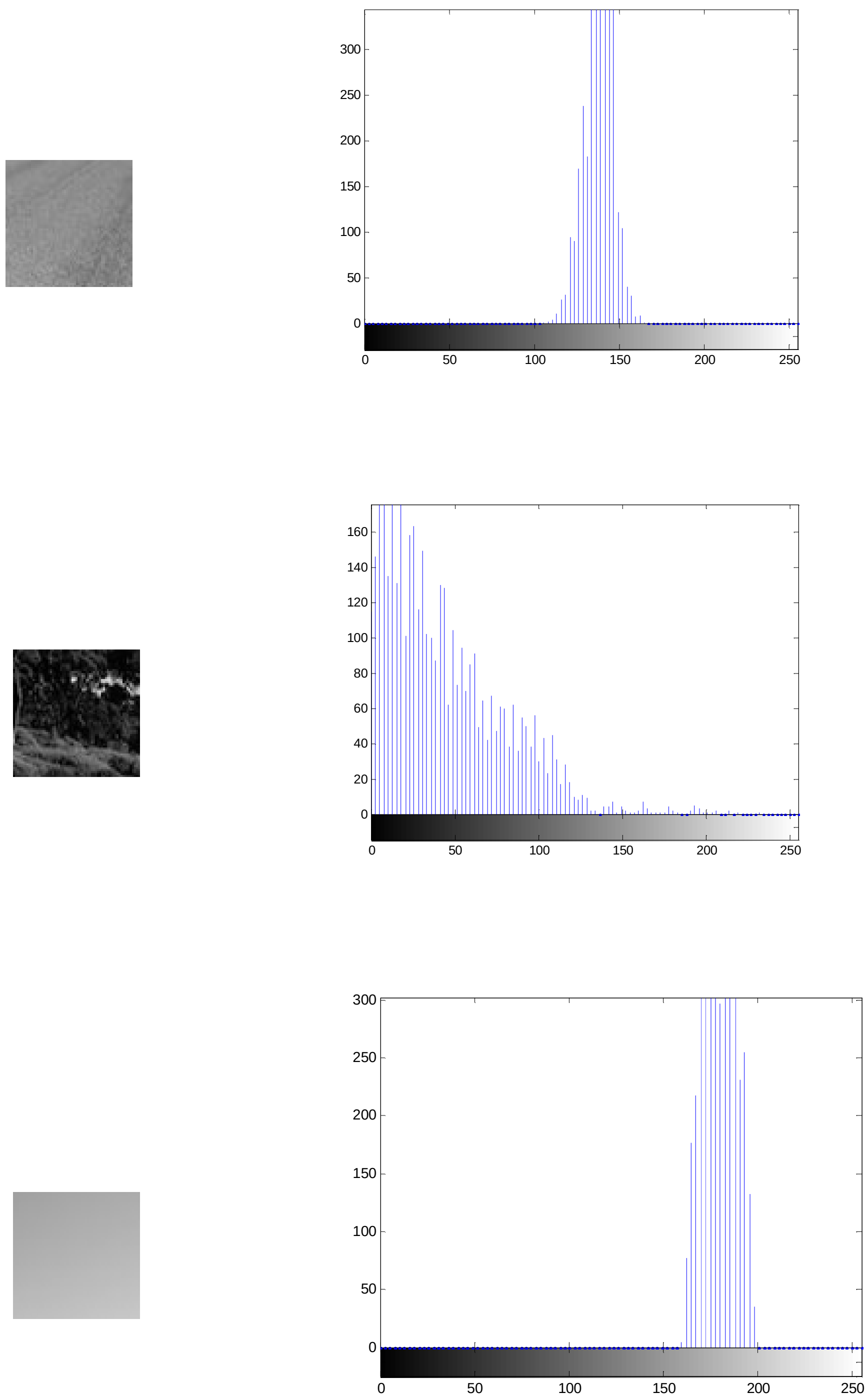


Figure 3.1 Histogram of three different sub- images(low contrast, dark, light)

3.3.1 Detection of Discontinuities

In this section I will present how can detect three basic types of gray-level discontinuities in a digital image (points, lines, and edges). The detecting of the discontinuities, just passing a mask with size n*n through whole the image, where n=3. The idea for detecting isolated point (the point whose gray level is significantly deferent from background) and located in homogenous region will be quite deferent from surroundings and easy to detect by applying this mask.

-1	-1	-1
-1	8	-1
-1	-1	-1

so the area has constant gray level the sum of this mask will equal to zero.

Line detection has a little bit complexity, there are many directions of lines to be detected (Horizontal, $+45^{\circ}$, vertical, -45°). the masks used for detecting lines to those directions defined in [1].

The edge is changes or discontinuities in an image amplitude attribute such as luminance or tri-stimulus value are fundamentally important primitive characteristics of an image because they often provide an indication of the physical extent of objects within the image. Local discontinuities in image luminance from one level to another are called *luminance edges*. Global luminance discontinuities, called *luminance boundary segments*, to detect the edge (discontinuity) in gray level will used second order derivatives. The second derivatives called Laplacian, It is second order derivatives for the 2-D function $f(x, y)$ [1,6] and defined as:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (3.1)$$

Because the un ability of the Laplacian to detect the edges when used in the original form, so will used with Gaussian to do this purpose as:

Laplacian of a Gaussian(LOG)= $\nabla^2 h(r) = -\left[\frac{r^2 - \sigma^2}{\sigma^4}\right] e^{-\frac{r^2}{2\sigma^2}}$ (3.2)

3.3.2 Thresholding

In many vision applications, it is useful to be able to separate out the regions of the image corresponding to objects in which we are interested, from the regions of the image that correspond to background. Thresholding often provides an easy and convenient way to perform this segmentation based on the different intensities or colors in the foreground and background regions of an image. There are many types of thresholds global, local, and adaptive (dynamic) thresholding.

The key issue here is how to choose an optimal threshold value so that the number of misclassified image pixels is kept as low as possible. Theoretically, we can determine the optimal threshold value according to the Byes rule if we know the pixel value distributions of foreground and background classes [5]. However, in practice we know only a mixture of the two distributions, or the pixel value histogram of an image, but not the two separate distributions, so we may have to make some assumption, determine the optimal threshold according to the statistical decision theory. Figure 3.2 shows the histogram of two objects in an image with multiple thresholds. In the next chapter I will discuss threshold in more details.

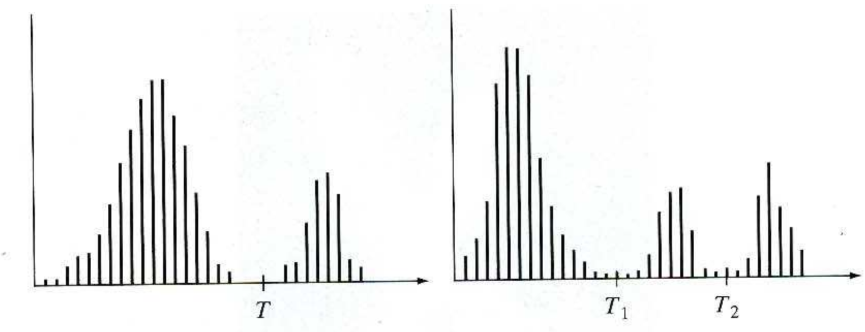


Figure 3.2 Gray-level-histogram can be partitioned by single and multiple thresholds.

3.3.3 Region growing

Region growing is one of the conceptually simplest approaches to image segmentation; neighboring pixels of similar amplitude are grouped together to form a segmented region. This type of segmentation is considered as a process that partitions entire image R into n sub regions, $R_1, R_2, ..., R_n$, such that

$(a)\ \bigcup_{i=1}^{n} R_i = R$

$(b)\ R_i$ is a connected region, $i = 1,2,\dots,n$

$(c)\ R_i \cap R_j = \emptyset$ for all i and j, $i \neq j$

$(d)\ P(R_i) = \text{TRUE}$ for $i = 1,2,\dots,n$

$(e)\ P(R_i \cup R_j) = \text{FALSE}$ for any adjacent regions

Here. $p\ (R_k)$ is a logical predicate defined over the points in set R_k, and ϕ is the null set. Condition (a) indicates that the segmentation must be complete; that is, every pixel must be in a region. Condition (b) requires that points in a region must be connected in some predefined sense. Condition (c) indicates that the regions must be disjoint. Condition (d) deals with the properties that must be satisfied by the pixels in a segmented region-for example $P(R_i)$ = TRUE if all pixels in R_i have the same gray level. Finally, condition(e) indicates that regions R_i and R_j are different in the sense of predicate *P*.[1,3]

points are the white pixels (255)

- region growing criteria are
- gray level difference is less than 65
- pixel has to be 8-connected

3.3.3.1 Region Splitting and Merging

The entire image is subdivided into the set of arbitrary subregions after that splitted or merged in order to satisfy defined criteria specifying the region characteristics, as shown in figure 3.3.

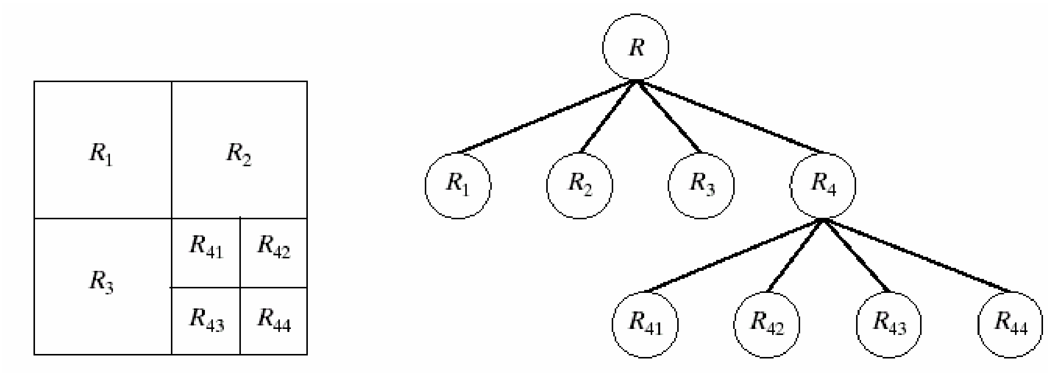


Figure 3.3 partitioned image and its corresponding quad tree.

The following steps can summarize the procedure of splitting:

- split into four disjoint quadrants any region for which $P(R_i)$ = FALSE
- merge any adjacent regions for which $P(R_i \cup R_j) = true$.
- stop when no further merging or splitting is possible

Figure 3.6 shows an input image, which has a criteria $P(R_i) = \text{TRUE}$ and at least of 80% of pixels in *R*i have the property

$$|z_i - m_i| \leq 2\sigma_i$$

where z_i is the gray level of the *jth* pixel in *Ri*, m_i is the mean gray level of that region , and σ_i is the standard deviation of the gray levels in *Ri*

In the following example, splitting and merging to segmented road class from input road image are applied as in figure 3.4. This splitting are made only around the road borders with one or two windows size by 64*64 and merge to main body any new windows are fulfill the thresholds. In our algorithm, this method of splitting repeated two times decreases down scanning window size to 32 and 16 with two passes of segmentation and merging to the segmented main body this implement the half and quadratic splitting and merging method. Figures 3.5,6 show the splitting and merging results in each window size. The result from three passes segmentation procedures are shown in Figure 3.7.

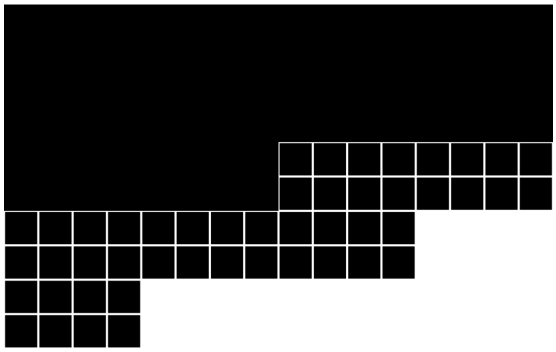


Figure 3.4 segmented road class and splitting with 32*32 around borders.

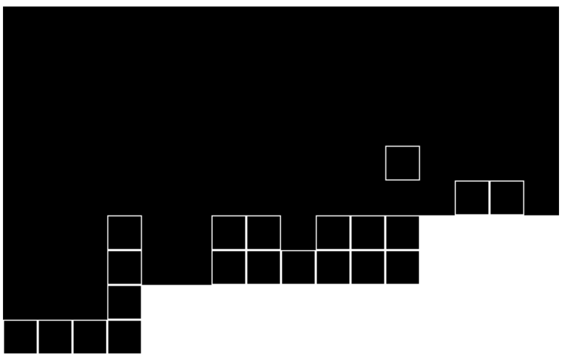


Figure 3.5 merging all splitting windows to the main road class body with 32*32 around borders satisfy criteria from second pass segmentation.

Figure 3.6 merging all splitting windows to the main road class body with 16*16 around borders satisfy criteria from third pass segmentation.

Figure 3.7 result of segmented road class by three passes segmentation applied splitting and merging procedures.

3.4 Morphological Image Processing

The word *morphology* commonly denotes a branch of biology that deals with the form and structure of animals and plants. We use the same word here in the context of *mathematical morphology* as a tool for extracting image components that are useful in the representation and description of region shape, such as road boundaries, skeletons, etc. When dealing with binary images, the principal application of morphological image processing is extracting image components that are useful in the representation and description of shape. These algorithms are used to extract such objects as:

- Boundary Extraction.
- Region Filling.
- Extraction of Connected Components.

The morphological image processing works with **c**oncepts of set theory and the most uses functions in this algorithm are *'Erosion' and 'Dilations'*. The fundamental operations in morphological image processing and all other morphological operations based on these two primitive operations[1].

The Dilation:

Dilation is the set of displacements, *z*, such that reflexion of structuring element, $\hat{B}$, and the set *A* overlap by at least one element

$$A \oplus B = \left\{ z \middle| \left(\hat{B}\right)_z \cap A \neq \emptyset \right\} \quad (3.3)$$

$$A \oplus B = \left\{ z \middle| \left[\left(\hat{B}\right)_z \cap A\right] \subseteq A \right\} \quad (3.4)$$

The Erosion:

For a set A and B in Z^2, The erosion A by B, denoted $A \Theta B$ is defined as

$$A \Theta B = \{z | (B)_z \subseteq A\} \qquad (3.5)$$

In the Figure 3.8 shows an example for the erosion and dilation processes, As we can see , dilation expands an image and erosion shrinks it.

Figure 3.8 left one is the intermediate result for road image, middle erosion image, and right one dilated of the eroded image with structuring element 17.

The next morphological functions are used in this algorithm so called opening and closing, defined as:

opening: $$A \circ B = (A \Theta B) \oplus B \qquad (3.6)$$

closing: $$A \bullet B = (A \oplus B) \Theta B \qquad (3.7)$$

3.5 Representation and Description

After an image has been segmented into regions by one of the segmentation methods, the resulting aggregate of segmented pixels usually is represented and described in a form suitable for further computer processing.

Basically, representing a region involves two choices:

(1) Represent the region in terms of its external characteristics (its boundary).

(2) Represent it in terms of its internal characteristics (the pixels comprising the region).

Choosing a representation scheme, however, is only part of the task of making the data useful to a computer. The next task is to *describe* the region based on the chosen representation. For example, a region may be *represented* by its boundary, and the boundary *described* by features such as its length, the orientation of the straight line joining its extreme points, and the number of concavities in the boundary.

An external representation is chosen when the primary focus is on shape characteristics. An internal representation is selected when the primary focus is on regional properties, such as color and texture. Sometimes it may be necessary to use both types of representation. In either case, the features selected as descriptors should be as insensitive as possible to variations in size, translation, and rotation. For the most part, the descriptors discussed in this chapter satisfy one or more of these properties. The segmentation techniques yield raw data in the form of pixels along a boundary or pixels contained in a region. Although these data sometimes are used directly to obtain descriptors (as in determining the texture of a region), standard practice is to use schemes that compact the data into representations that are considerably more useful in the computation of descriptors. There are many representation approaches can be used to represents the segmented objects [1] as:

3.5.1 Regional Descriptors

It is one of the image description approaches uses both boundary and regional descriptors like (area, topologic, and texture).

3.5.1.1 Texture

Texture descriptors one of an important approach to region description, and will used in this algorithm is to quantify its *texture* contents of the input image in order to extract the interesting road boarders from it. Although no formal definition of texture exists, intuitively this descript or provides measures of properties such as smoothness, coarseness, and regularity, These three principal approaches used in image processing to describe the texture of a region are statistical, structural, and spectral. Statistical approaches yield characterizations of textures as smooth, coarse, grainy.

Structural techniques deal with the arrangement of image primitives, such as the description of texture based on regularly spaced parallel lines. Spectral techniques are based on properties of the Fourier spectrum and are used primarily to detect global periodicity in an image by identifying high energy, narrow peaks in the spectrum, next chapter I will discuss the usage of texture features in this algorithm [1,29, 31].

3.6 Object Recognition

This task of image processing finds a given object in an image or video sequence. Humans recognize a multitude of objects in images with little effort, despite the fact that the image of the objects may vary somewhat in different viewpoints, in many different sizes / scale or even when they are translated or rotated. Objects can even be recognized when they are partially obstructed from view. This task is still challenges computer vision systems.

The approaches are divided into two principal areas: decision -theoretic and structural. The first category deals with patterns described using quantitative descriptors, such as length, area, and texture. The second category deals with patterns best described by qualitative descriptors, such as the relational descriptors [1].

4. Texture analysis.

4.1 Introduction:

The texture has been a topic of great interest in the field of image processing in the last decades. While traditional object recognition and segmentation techniques often rely on areas of constant intensity for object separation and description, texture analysis is an important feature of many image types, which is the pattern of information or arrangement of the structure found in the picture. The usage of texture features one key function in various image-processing applications to capture certain visual property of the interested region such as remote sensing and computer vision for autonomous guided vehicle applications. In image analysis, texture is broadly classified into two main categories, statistical and structural. Textures that are random in nature are well suited for statistical characterization, for example, as realizations of random fields. They do not have easily identifiable primitives (e.g., bark, sand, etc.). Structural textures, on the other hand, are characterized by a set of primitives (*texels*) and placement rules. The placement rules define the spatial relationships between the *texels* and these spatial relationships may be expressed in terms of adjacency, closest distance or periodicities. The *texels* themselves may be defined by their gray level, shape or homogeneity of some local property. Many real-world textures contain the structural characteristic. A large number of woven fabrics and commercial furniture are good examples of purely structural or semi-deterministic textures. Microscopic images of electron beam textures in steel surface and human endothelium also have structural characteristics. Structural texture classification is an important task in pattern recognition applications.

Texture classification approaches can be also organized into two main categories: statistical and structural approaches. Statistical approaches consider textures as complicated pictorial
patterns on which sets of statistics can be defined to characterize these patterns. In the structural methods, the texture is considered to be a cellular and an ordered phenomenon. Hence, the purpose of the first stage of the analysis is to define the *texel*. Since structural methods involve numerous images, pre-processing procedures to extract *texels* so that they are time consuming, statistical approaches are the more efficient approach for texture matching. Thus, most of the former methods used statistical approaches regardless of the class of textures. In the classification of structural textures, Examples can be seen in Fig. (4.1). **[7]**

(a)Textile 1. (b) Textile 2. (c) Textile 3.

Figure 4.1 Examples of structural textures.

Here in this dissertation, the texture features (statistical moments and spatial structure) used to extract the road borders from the digital input image read by camera mounted on autonomous vehicle.

4.2 Texture:

Texture analysis is one of the main topics in the field of digital image processing. Although, there is no formal definition for the texture, but can described as a pattern with some kind of smoothness, coarseness and regularity. There are a number of ways how to characterize a texture. Typical examples are statistical, structural, and spectral. These
properties can be used as an approaches to classify the image contents. Fig.(4.2) shows texture properties.

Fig.(4.2) From left to right shows white square smooth, coarse, and regular textures.

Two of main application domains related to texture-based image analysis are: 1- supervised texture classification (determining the class (texture) to which an image or image region belongs using a set of examples for each class as training set) and 2- unsupervised texture segmentation (partitioning of a given image into disjoint regions of uniform texture). The approach used is in this work a mixture of these two. While the main task consists in partitioning of an image into uniform regions, the starting point in the segmentation algorithm has a nature of a simple training process **[1, 30, 31].**

4.3 Texture Features

Texture features extracted by three methods, main categories, namely: statistical, structural, and spectral which are briefly explained in the following sections.

4.3.1 Statistical Method

Statistical method is first order histogram represents the texture indirectly according to the non-deterministic properties that manage the distributions and relationships between the gray levels of an image. This technique is one of the first methods in machine vision. By computing local features at each point in the image, and deriving a set of statistics from the distributions of the local features, statistical methods can be use to analyze the spatial distribution of gray values. Based on the number of pixels defining the local feature, statistical methods can classified into first-order (one pixel), second-order (two pixels) and higher-order (three or more pixels) statistics. The difference between these classes is that the first-order statistics estimate properties (e.g. average and variance) of individual pixel values by waiving the spatial interaction between image pixels, but in the second-order and higher-order, statistics estimate properties of two or more pixel values occurring at specific locations relative to each other**[1, 32,33].**

The statistical descriptors used to extract the image object as following:

Standard deviation of light intensities:

$$\sigma = \sqrt{\mu_2(z)} = \sqrt{\sum_{i=0}^{L-1} (z_i - m)^2 p(z_i)} \tag{4.1}$$

Third moment:

$$\mu_3(z) = \sum_{i=0}^{L-1} (z_i - m)^3 p(z_i) \tag{4.2}$$

Uniformity:

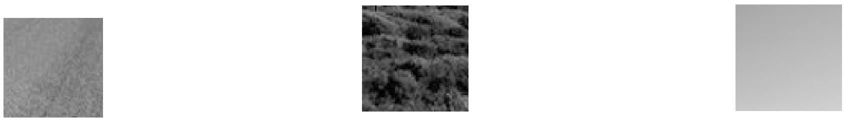
$$u = \sum_{i=0}^{L-1} p^2(z_i) \tag{4.3}$$

Average entropy:

$$e = -\sum_{i=0}^{L-1} p(z_i) \log_2 p(z_i) \quad p \in [0,1] \tag{4.4}$$

Where m the mean value: $m = \sum_{i=0}^{L-1} z_i p(z_i)$

Here, example for statistical descriptors calculations for different input images size 64*64 pixels as shown in Fig (3.4).



Road Ground Sky

Fig (4.3) input images size 64*64.

In the table 4.1 shows the statistical descriptors calculations for three different images has different texture types. It obvious, the sky and road have less deviations than the ground means more regular and smoothness than ground. In third moment descriptor the sky is smoothness than road and ground(less sleekness from the mean), uniformity measure the how the gray level pixels are homogenous to each other, so sky is more homogenous than road and ground. Finally, the average entropy measure coarsens of each class, the ground has more coarsens than road and sky.

Table 4.1 calculations of four statistical descriptors for three classes.

	σ	3^{rd}_mom	Uniformity	Entropy
Road	8.775	275.9111	0.0334	5.1392
Ground	29.4029	5174.128	0.0098	6.8278
Sky	9.4695	73.9777	0.0289	5.2059

4.3.2 Structural method:

Structural method is second order histogram derived from so called "co-occurrence matrix" representing spatial relationships between pixels of different light intensities. The general idea of a co-occurrence matrix is to represent an image's texture characteristics by counting pixel intensity pairs, using a matrix that keeps track of all the pixel-pair counts. The normalized co-occurrence matrix is denoted by $c_{ij}(d,\theta)$ where *d* is the displacement vector, θ *is* the angle, and *i* and *j* represent the gray-levels in the vertical direction (along the rows) and horizontal direction (along the columns), respectively. In order to capture texture patterns to be evaluated in a further step, we calculate any different displacements for eight directions (0°, 45°, 90°, 135) as shown in figure (4.4.a).

As a further illustration, I consider the small image shown in figure (4.4.b). To produce co-occurrence matrix for a given value of *d*, we merely need to calculate the number of cases for which pixels a distance *d* apart have intensity values *i* and *j*. Here, we content ourselves with two cases *d* = (1,0) and *d* =(1, $\pi/2$). **[1,33]**. we thus obtain the matrices shown in figures 4.4.c and 4.4.d. this simple example demonstrate, that the amount of data in the matrices is liable to be many times more than in the original image.

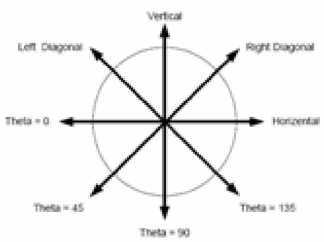


Fig (4.4a) displacement in eight directions.

0	0	0	1
1	1	1	1
2	2	2	3
3	3	4	5

Fig (4.4.b) input image

	0	1	2	3	4	5
0	2	1	0	0	0	0
1	1	3	0	0	0	0
2	0	0	2	1	0	0
3	0	0	1	1	1	0
4	0	0	0	1	0	1
5	0	0	0	0	1	0

Fig (4.4.c) shows the co-occurrence matrix for d = (1,0)

	0	1	2	3	4	5
0	0	3	0	0	0	0
1	3	1	3	1	0	0
2	0	3	0	2	1	0
3	0	1	2	0	0	1
4	0	0	1	0	0	0
5	0	0	0	1	0	0

Fig(4.4.d) shows the co-occurrence matrix for d =(1, $\pi/2$)

Another example illustrates the function of the co-occurrence matrix to extract some information about the texture in the input image. As shown in Fig (4.5.a) this is the input image of sky which has more homogenous texture and more brightness , so the co-occurrence matrix of this image as shown in fig(4.5.b) its diagonal values and more brightness image(this image has more smoothness texture). In fig (4.6.a), it is the sub image from concrete road image not homogenous enough like the first one and darker than the first, the occurrence matrix image shown in figure 4.6.b. The co-occurrence matrix output is explains these differences compared to the first image. Its brightness in middle, not so homogenous but regular because its symmetrical about the diagonal of the result matrix, but the fig(4.7.a) contrast of the previous examples, thus its more coarseness and less homogenous represented the ground image texture, like(grass and forest). The co-occurrence matrix output as shown in figure 4.7.b is representing those texture features (not homogenous, not smooth and darker image)[1].

The structural descriptors defined as:

Contrast:

$$C=\sum_i\sum_j\left(\frac{i-j}{N}\right)^2 c_{ij}\Big/\sum_i\sum_j c_{ij} \qquad (4.5)$$

Inverse moment of first order:

$$IM=\sum_i\sum_j c_{ij}/(i\text{-}j)^k \quad i\neq j \ \ , \ k=1 \qquad (4.6)$$

Uniformity:

$$U=\sum_i\sum_j c^2{}_{ij} \qquad (4.7)$$

Entropy:

$$E=-\sum_i\sum_j c_{ij}\log_2 c_{ij} \qquad (4.8)$$

Fig (4.5.a) Sky image

Fig (4.5.b) Co-occurrence matrix shown as image

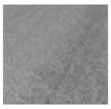
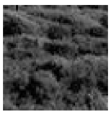


Fig (4.6.a) Road image.

Fig (4.6.b) Co-occurrence matrix shown as image.



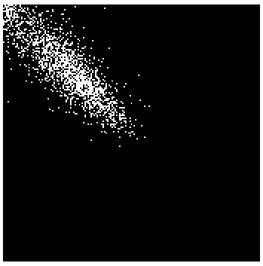


Fig (4.7.a) Ground image.

Fig (4.7.b) Co-occurrence matrix shown as image.

Although none of these descriptors can be universally applicable for specified task and none of the image regions can be considered as completely homogenous relative to anyone of them, this approach tends to find which one among them (in both categories) can be adopted as the best representative in the particular case. Simultaneous consideration of both statistical and structural properties have been used as a measure of redundancy, trying to take into account not just global statistical measures of coarseness, but some structural regularities of the road region distinguishing it from other parts of image[1].

4.3.3 Spectral method:

4.3.3.1 Collapsed Frequency Domains:

A third way to analyze texture is in the frequency domain to test the input textures are periodic patterns. The entire frequency domain is, however, as much information as the image itself. We can condense the information by collapsing a particular frequency across all orientations (by integrating around circles of fixed distance from the frequency origin) or by collapsing all frequencies in a particular orientation (by integrating along each of a unique orientation though the origin). If we express the frequency-domain coordinates in polar coordinates as in equations 4.9,10, these are:

$$S(r) = \sum_{\theta=0}^{\pi} S(r, \theta) \qquad (4.9)$$

and

$$S(\theta) = \sum_{\theta=0}^{N/2} S(r,\theta) \qquad (4.10)$$

$S(r)$ tells us the distribution of high and low frequencies across all angles. $S(\theta)$ tells us the distribution of frequency content in specific directions, As shown in Fig. (4.8). these two one-dimensional descriptors can be useful for discriminating textures [1].

4.3.3.2 Local Frequency Content:

As we discussed earlier, the frequency domain contains information from all parts of the image. This makes the previous method useful for global texture analysis, but not local. If a region has already been segmented, you could pad the region with its average intensity to create a rectangular image. This does not, however, provide a useful way of using texture to do the segmentation. We can define local frequency content by using some form of co-joint spatial-frequency representation. As we discussed, though, this only partially localizes the position or frequency of the information—you can't do both perfectly.

A simple way to do this would be to examine the $N \times N$ neighborhood around a point and to compute the Fourier Transform of that $N \times N$ sub image. The frequency content of the window changes as you move from one textured region to another. Differences in the frequency content of each window could then be use as a means of segmentation. Of course, one still needs to distill descriptors from the frequency content of each window. One such descriptor (Coggons, 1985) is to compute the total energy (squared frequency content) of the window. If you exclude the zero frequency term, this is invariant to the average intensity. If you normalize by the zero-frequency term, it is invariant to intensity gain as well. There are, of course, other descriptors you could use as well.

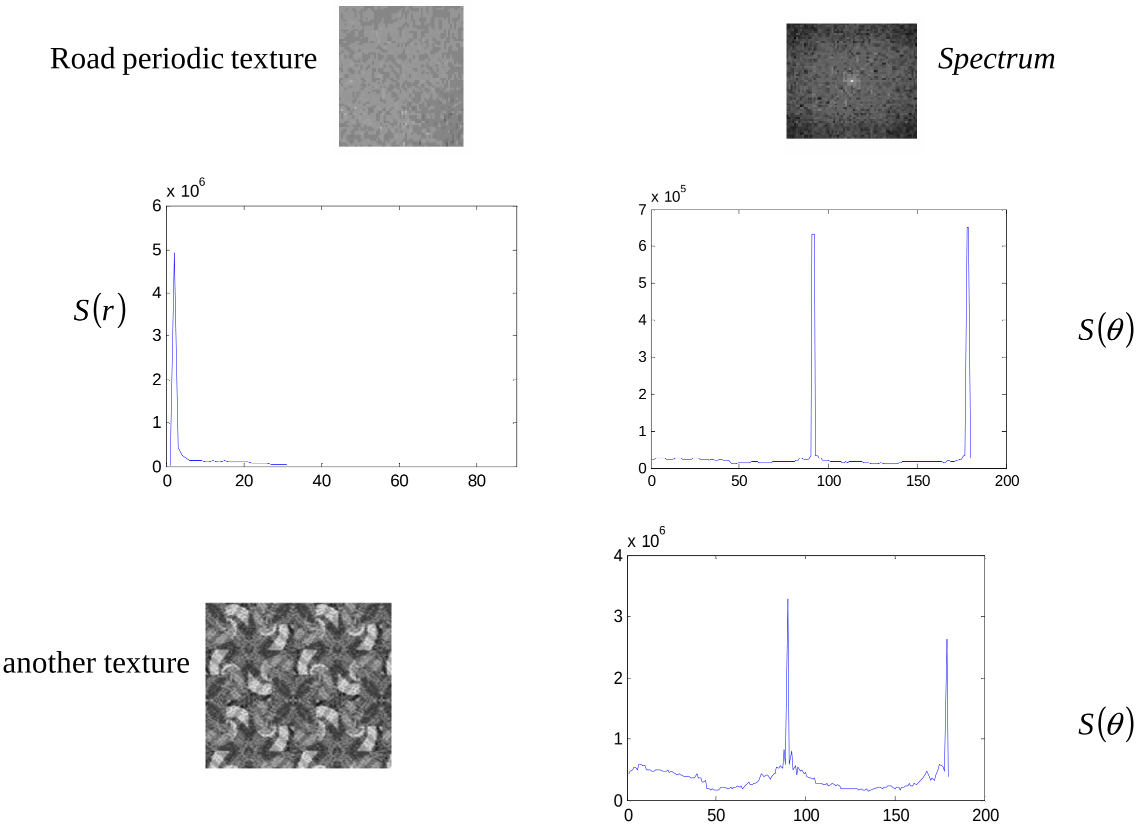


Figure (4.8) shows different types of periodic texture and its spectrum.

4.4 Texture Analysis:

Texture analysis is one of the fundamental aspects of human vision by which we discriminate between surfaces and objects. In a similar manner, computer vision can take advantage of the cues provided by surface texture to distinguish and recognize objects. In computer vision, texture analysis may be used alone or in combination with other sensed features (e.g. color, shape, or motion) to perform the task of recognition. Either way, it is a feature of paramount importance and boasts a tremendous body of work in terms of both research and applications. Currently, the main approaches to texture analysis must be sought out through a variety of research papers. This collection of chapters brings together in one handy volume the major topics of importance, and categorizes the various techniques into comprehensible concepts. The methods covered will not only be relevant to those working in computer vision, but will also be of benefit to the computer graphics, psychophysics, and pattern recognition communities, academic or industrial.

Texture analysis is an essential issue in computer vision and image processing, such as in remote sensing, content based image retrieval, and so on **[31, 33].** Texture analysis has two steps:

4.4.1 Texture extraction:

For extracting texture feature information, there are two primary methodologies. The first class of methods applies a linear transform, filter, or filter bank globally to the image. The local energy of the filter responses represents the local texture feature values. Generally, these methods have high computational complexity. The second class of methods divides the whole image into many small non-overlapping pixel blocks (k*k), and then applies some extractions features selected, to each block to get the local information. These methods extract texture features for a block of pixels. Both methodologies have the problem of generating texture information for each individual pixel. To extract texture features for each pixel, we apply a window of some pre-determined size, ***k*k***, to each pixel, as illustrated in Fig (4.9). The center of the window slides over every pixel and perform any textural extractions methods as statistical or structural at each location to determine each pixel's texture feature.

In the first step of texture analysis, some basic information on the image that is useful for other steps is calculated. This information defines the homogeneity or similarity between different regions of an image.**[34]**

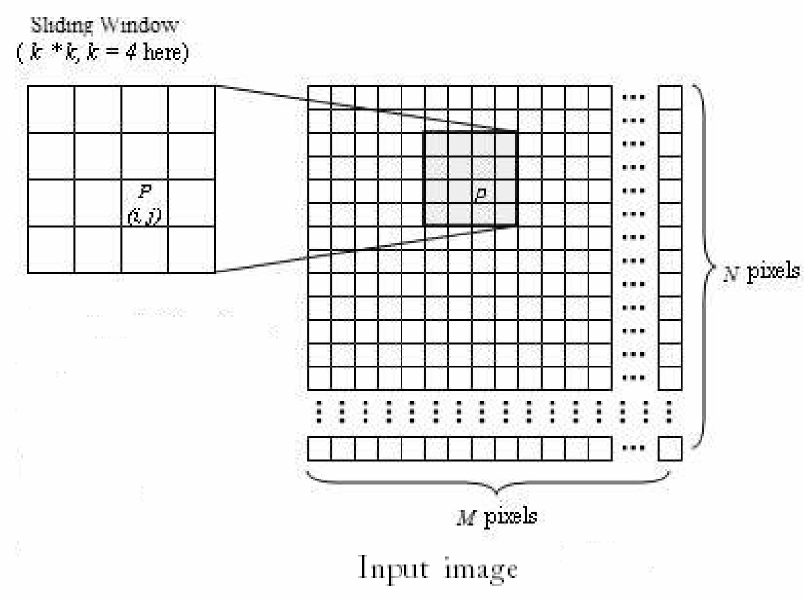


Fig (4.9): Texture Feature Extraction for pixel *p(i, j)* (window size is 4*4)

4.4.2 Texture classification:

Texture classification assigns a given texture to some texture classes. There are two main classification methods *supervised* and *unsupervised* classification. Supervised classification provided examples of each texture class as a training set. A supervised *classifier* is trained using the set to learn a characterization for each texture class. Unsupervised classification does not require prior knowledge, which is able to automatically discovers different classes from input textures. Another class is *semi-supervised* with only partial prior knowledge being available such class are used in our algorithm.

The majority of classification methods involve a two-stage process. The first stage is feature extraction, which yields a characterization of each texture class in terms of feature measures. It is important to identify and select distinguishing features that are invariant to irrelevant transformation of the image, such as translation, rotation, and scaling. Ideally, the quantitative measures of selected features should be very close for similar textures. In this algorithm the operator must be click three representative selected classes positions to define to the system in prior three kinds of texture types. Then, after some calculation select best texture extractor among texture features to decrease number of computational cost and time. However, it is a difficult problem to design a universally applicable feature extractor, and most present ones are problem dependent and require more or less domain knowledge. The second stage is classification, in which classifiers trained to determine the classification for each input texture based on obtained measures of selected features. In this case, a classifier is a function, which takes the selected features as inputs and texture classes as outputs**[22,34,35].** The threshold values must be calculated and defined, the simplest idea is to specify them in the middle of distance between prototype values of descriptors (either directly from inspect windows, or based on calculated mean values of 1024 acquisition values). However, it was realized in practice that due to non-homogeneous nature of all three regions, the information about standard deviation of particular descriptor should be used also. As a result of this, threshold values are set on the points where the probability of the membership to both neighboring classes is equal (at the intersections of particular probability density functions). Assuming that the partial probability density functions are of Gaussian type (PDF), the overall PDF can be represented as equation 4.11:

$$p(z) = \frac{P_1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(z-\mu_1)^2}{2\sigma_1^2}} + \frac{P_2}{\sqrt{2\pi}\sigma_2} e^{-\frac{(z-\mu_2)^2}{2\sigma_2^2}} \tag{4.11}$$

To determine the optimal threshold according to the statistical decision theory, even so , threshold selection can still be complicated mathematically since may involve nonlinear equations. The method used to estimate thresholds to produce minimum average segmentation error. If we consider an image contains only two principal gray-level regions. Let *z* denote gray- level values. we can view these values as random quantities, and their histograms may be considered an estimate of their probability density function(PDF), *p(z)*.This overall density function is the sum or mixture of two densities, one for the light and the other for the dark regions in the image. Furthermore, the mixture parameters are proportional to the relative areas of the dark and light regions. If we know the values of the densities or even assumed, it is possible to determine an optimal threshold (in terms of minimum error) for segmenting the image into the two distinct regions. Figure 4.10 shows two probability density functions**[1,5].**

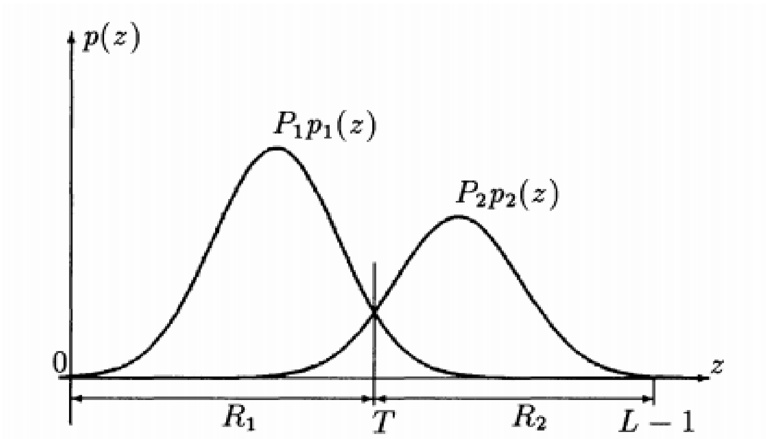


Figure (4.10) Gray-level probability density functions of two regions in an image.

$$g(x,y)=\begin{cases}1 & if \quad f(x,y)\le T\\ 0 & if \quad f(x,y)>T\end{cases}$$

The two thresholds are needed where one of the classes is decreased slowly and the second decreased rapidly as z decreased to zero, in this case $R1 \in [T1+1, T2]$ and $R2 \in [0, T1] \cup [T2+1, L-1)$.

and the thresholded image *g(x, y)* for input image *f(x, y)* are defined as:

$$g(x,y)=\begin{cases}1 & if \quad T1< f(x,y)\le T2\\ 0 & if \quad f(x,y)\le T1 \; or f(x,y)>T2\end{cases}$$

Figure 4.11 illustrates this case.

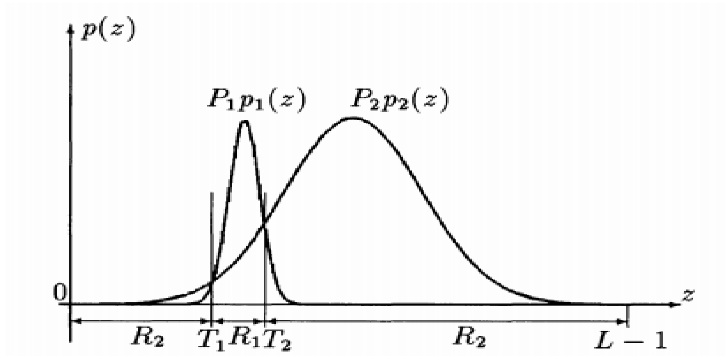


Figure (4.11) Two class problems required two thresholds T1,T2

Assume that the larger of the two probability density functions corresponds to the background levels while the smaller One describes the gray levels of the objects in the image. The mixture probability density function describing the overall gray-level variation in the image as:

$$p(z) = P_1 p_1(z) + P_2 p_2(z) \tag{4.12}$$

Where, $P1, P2$ are the probabilities of occurrence of the two classes of pixels, and the sum of these two probabilities are equal one.

$$P_1 + P_2 = 1$$

To find the optimal values for thresholds will solve the equation for T, by assumption $P1 = P2$, then the optimal threshold is where the intersection of the curves $p_1(z)$ and $p_2(z)$. If we consider the approximation of both PDF are as Gaussian density functions, which is completely characterized by two parameters : the *mean and variance*. In this case the overall mixture PDF as in equation (4.11).

Where μ_1 and σ_1 are the mean and variance of the Gaussian density of one class pixels (objects) and μ_2 and σ_2 are the mean and variance of the Gaussian density of the other class.

After differentiation the solutions of T is:

$$AT^2 + BT + C = 0 \tag{4.13}$$

$$A = \sigma_1^2 - \sigma_2^2 \tag{4.14}$$

$$B = 2(\mu_1\sigma_2^2 - \mu_2\sigma_1^2) \tag{4.15}$$

$$C = \sigma_1^2\mu_2^2 - \sigma_2^2\mu_1^2 + 2\sigma_1^2\sigma_2^2 \ln(\sigma_2 P_1 / \sigma_1 P_2) \tag{4.16}$$

Since a quadratic equation has two possible solutions, two threshold values*(T1, T2)* may be required to obtain the optimal solution. Depending on the decision regions R1 and R2 can be determine as following cases [1,5]:

Case 1: If square root are not real values, so *C* is very large and *P2* is very small as illustrated in figure 4.12.

Case 2: $T1 \in [0, L-1]$, and $T2 \notin [0, L-1] \Rightarrow R1 = [0, T1]$, and $R2 = [T1+1, L-1]$.

Case 3: $T1 \notin [0, L-1]$, and $T2 \in [0, L-1] \Rightarrow R1 = [0, T2]$, and $R2 = [T2+1, L-1]$.

Case 4: $T1 \in [0, L-1]$, and $T2 \in [0, L-1]$

Bothe *T1 and T2* are useful solutions

$$\Rightarrow\ R1 = [T1+1, T2], \quad \text{and } R2 = [0, T1] \cup [T2+1, l-1], T1 < m1.$$

$$\Rightarrow\ R1 = [0, T1-1] \cup [T2+1, L-1], \text{ and } R2 = [T1+1, T2], T1 > m1.$$

As illustrated in the figure (4.10).

Case 5: $T1 = T2 \in [0, L-1]$, $\sigma_1 = \sigma_2 = \sigma$ and *A=0.*

$$\sigma^2 = \sigma_1^2 = \sigma_2^2 \Rightarrow T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln\left(\frac{P_2}{P_1}\right) = -\frac{C}{B} = \frac{\mu_1 + \mu_2}{2} \qquad (4.17)$$

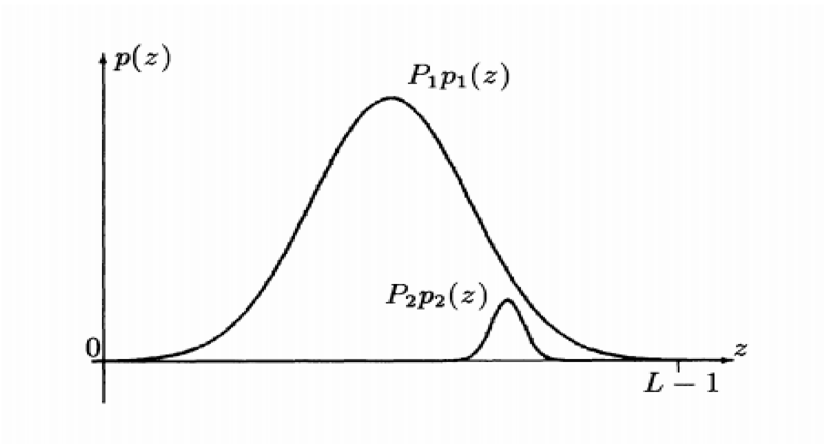


Figure (4.12) Two class problems there are no real soluttions for t1,t2

4.5 Related work:

Fundamentals of image segmentation, texture based descriptors, and basic pattern recognition relevant for the image processing applications has given in textbooks

[31,36]. Wide varieties of texture analysis and texture methods extractions and classifications have been proposed **.** This proposal attempt to find new feature selection algorithm by use a set of multiple texture methods from different families, automatically determines a reduced subset of methods whose integration yields better classification results with in quality, quantity and lower computational cost**[21]**.

This paper introduces a new method by integrating a multiple texture features extraction from different families. Each method evaluated over multiple window size and identify the texture pattern to which every image pixel belongs. However, the particular window size to allow optimal discrimination of arbitrary texture **[22]**.

This method aims to use texture features to use as texture classified for unknown input application will be processed. Here proposed some methodology automatically determines a reduced subset of features whose integration produces good classification results independently of the texture classification problem to be solved **[23]**.

Here offer anew technique for supervised pixel based classification determine a subset of prototypes in order to characterize each texture class based on the outcome of a multi-channel Gapor filter bank. Then a simple minimum distance classifier used to classify every image pixel into one of the given texture classes. Multi-size evolution windows following a top-down are used in order to improve accuracy near frontier of regions of different texture **[24]**.

Introduce in this paper presentation of single and multi-channel co-occurrence matrices for color texture and gray scale texture analysis and put intensity independent color textures in perspective **[30]**.

This approach introduces one method to describe texture by using the orientation and the local variance to get better noise removal and closing of structures. To decrease the computational complexity used some unsupervised methods like the separability of features in order to reduce the redundancy **[25]**.

This paper described the comparison of texture classifications algorithms like Gaussian classifier, Gabor energy features and Fractal dimension; also introduce a framework for comparing classifications of texture variations **[35, 37]**.

5. Road borders extraction algorithm.

5.1 Introduction:

The concept of this approach developed to use texture features classifications to segment a road borders from the contents of an image produced by forward looking TV camera mounted on a moving vehicle. The extraction of a road boundary is an important step in the context of autonomous vehicle guidance and navigation systems, enabling further calculations of a distance from the border, direction of a road, etc. The main idea of using texture features as segmentation tool in this algorithm because; the texture of a road is different enough in comparison to the textures characterizing the surrounding environment, allowing the separation of the overall image into few distinguishable regions. The segmentation algorithm will use the texture descriptors defined in the previous chapter as a method for extract the road borders as (statistical nature and spatial structure). The algorithm will process each descriptor group alone and next will use both descriptors in two-dimensional space. The algorithm is written by MATLAB code and some typical images are used in order to verify the algorithm as well as to show the real limits of its applications as shown in next chapter.

Invention of this algorithm presented here mainly consists in the following aspects:

1. The automatic choice of the best candidates for descriptors – the most suitable ones for the actual scene;

2. Easy way of introduction of basic training via acquisition of statistical information related to descriptor values;
3. Fast threshold specification via reducing the multidimensional classification task to one-dimensional one;
4. "Low-resolution" approach in splitting/merging type of image segmentation.
5. Using some morphological image processing functions to remove disconnected windows and filling some holes in side main body created by nature or artificial obstacles. .
6. Refining extracted road borders by line fitting approximations.

5.2 Input Image

In this approach the input image is a static image, one frame only from the road scene. Preprocessing of it means: changing to grayscale image and resizing to standard algorithm size. Reading of dynamic frames means the reading of dynamic image from real live video
camera. The read image could be color or monochromic image read directly from TV camera which is mounted on autonomous vehicle. If the input image is read from camera as color image, it needs some preprocessing work to change into monochromic (multiple gray level) image. Finally, the input image is resized by standard algorithm image size (512* 320).

5.3 Read samples

The objective of this algorithm to work in Real-Time-applications, In order to do that, the algorithm will decrease number of classes to three. These three main classes are nature classes (road, near ground, far background) in the input road image. The operator have to click only at three representative positions on three classes, and the system will read three squares, each one from specific class. The spite for this approach let's only the operator (driver) to make this intervention at starting of the algorithm, to choose the most representative positions for so called seed-squares. The operator also will choose the proper square size max 64*64 and min 32*32 depends on the first image classes regions. For example if he cant fit the square size 64x 64 pixels inside one of the three classes(road, ground, and sky), so he can changes square size into 32x32, by the way this done only one time (at beginning of the algorithm start phase), and some kind of training will be done later to acquire 1024 values from each seed square. According to previous steps the algorithm will work as semi automatic algorithm means between two texture categories (supervised and unsupervised texture segmentation). After the driver choose the three seed-squares by specific size, the algorithm starts the segmentation of road borders. The segmentation of the input image uses large window size, splitting only at segmented road body thus, to decrease computation time in order to be applicable in real-time applications in future. Figure 5.1 shows how to select the input seed squares by size 64x64.

Figure 5.1 Input image with input seed-squares.

As mentioned before, this algorithm will use texture features as classification tool. These texture features are statistical (standard deviation, third moment, uniformity, average entropy) and co-occurrence matrix descriptors (contrast, inverse moment, uniformity, average entropy).

All eight descriptors mentioned above are calculated for the populations of these pixels inside these inspect windows. It is reasonable assumption that the texture of the near background is generally less homogeneous (coarser) than the one representing the road, while the texture of the far background (especially because of the sky) is more homogeneous (smoother) than the road class and this not important to road segmentation because always, the road in the lower half of the input image.

Gray level histograms, and co-occurrence matrices(represented as images) for all three inspect windows size 64*64 are shown on figures 5.2,3,4.

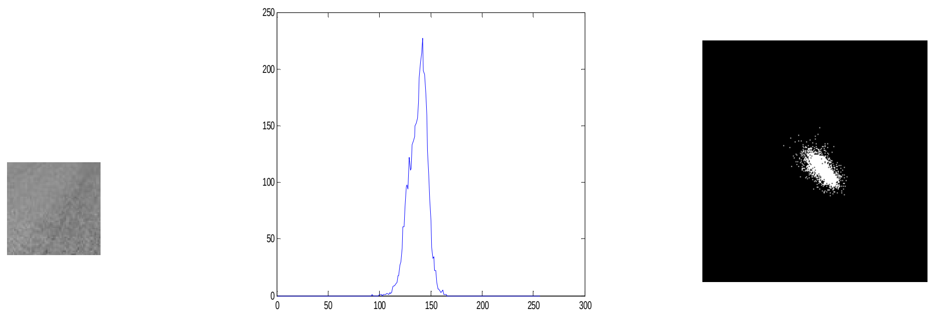


Figure 5.2 Road seed square, gray level histogram, and co-occurrence matrix.

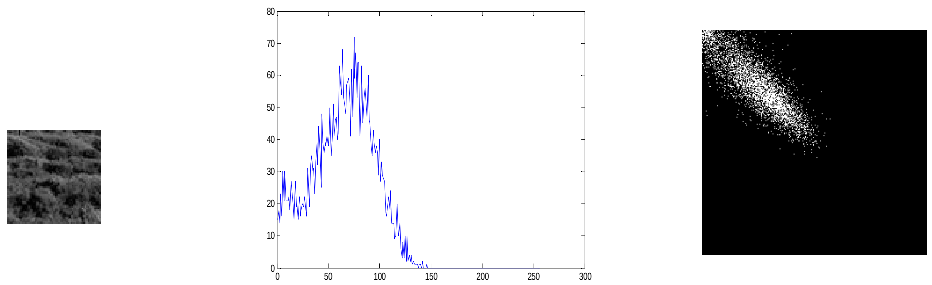


Figure 5.3 Ground seed square, gray level histogram, and co-occurrence matrix.

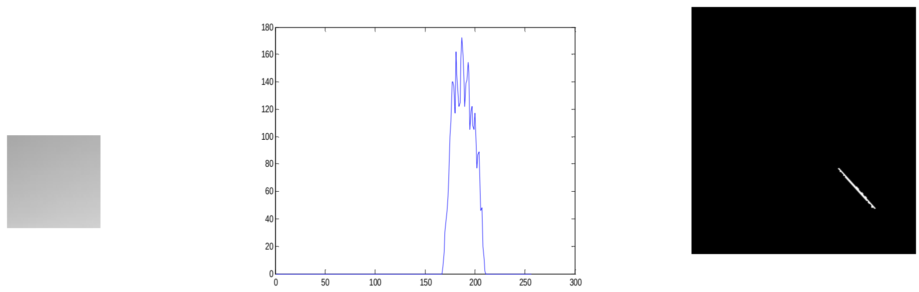


Figure 5.4 Sky seed square, gray level histogram, and co-occurrence matrix.

There are big differences between three classes of histograms shown in above figures as well as in the case of co-occurrence matrices. Thus, the statistical moment's descriptor like (third moment) generally measures the skewness to the left or right and determining the degree of symmetry of histograms around its mean value, and this information's gives a rough idea of weather the gray levels are biased toward dark or light side of the main. In this approach, based texture descriptors the variation (standard deviations) are very large between three seed squares calculations.

Evidently, for the road class (regular) texture, it's histogram has less gray level intensity than sky class and it is more symmetrical around the mean value, having the lower deviation from the main values compared to the ground class. For the ground class (coarse) texture, it's histogram has more variability of gray levels, higher value of standard deviation and it is non symmetrical around the main value. But the third class (sky) is smoother than the others and has symmetric histogram and lower deviation from the mean value. From the co-occurrence matrix point of view, this gives information about the positions of pixels with equal or nearly equal intensity values. The sky class has smoother texture than the road and ground, that is why it is concentrated at the diagonal of the matrix (symmetric) and toward more brightness. The road class has less homogenous texture, it is darker than the sky, and also symmetrical around the diagonal. The background is not homogenous at all (coarser) and it is darker than other two classes.

Table 5.1 shows all statistical descriptors calculated for three seed squares and from this calculations one may notice that there is a contrast between descriptors. For example, third moment calculated for the road class has four times greater value than for sky an eighteen times lower than for background. Compared from standard deviations point of view, the sky and road are separated enough from background but they are very close to each other.

Table 5.1 Calculations of statistical moments' for three seed squares.

	St. deviation	**Third mom.**	**Uniformity**	**Entropy**
Road	8.775	275.9111	0.0334	5.1392
Ground	29.4029	5174.128	0.0098	6.8278
Sky	9.4695	73.9777	0.0289	5.2059

Table 5.2 shows parameters calculated based on co-occurrence matrix for three seed squares and obviously, most of descriptors have closer values between classes except contrast descriptor that is separated enough between classes. For instant, road class has 2.5 time's lower value than background and one hundred times greater than sky. The contrast descriptor is opposite to the uniformity descriptor and any texture that has lower smoothing have high contrast value, and vise versa.

Table 5.2 co-occurrence matrix calculations for three seed squares.

	Contrast	**Inverse mom**	**Uniformity**	**Entropy**
Road	0.0016	0.1443	0.0014	9.8273
Ground	0.0042	0.0977	0.0004	11.3734
Sky	0.0000163	0.2427	0.0144	6.8078

5.4 Selection of descriptors

From the previous chapter we have seen that there are two texture feature groups (statistical and spatial structure depends on so-called co-occurrence matrix) where the first group depends on the statistical moments of gray level histogram and the second gives spatial relationship between neighboring pixels. The algorithm will use one descriptor from each group alone; because each group has four descriptors as in, statistical group (standard deviation, third moment, and uniformity, average entropy) and spatial structure group (contrast, inverse moment, and uniformity, average entropy). This method will be used to select the best descriptor (candidate) among descriptors from each group according to relative ratios. This method divide the maximum value among three descriptor calculations for three classes by the medium value and then divide the medium value by the smallest value, then take the average ration between result divisions of maximum/medium and medium/minimum as in equation 5.1. The best candidate descriptor for each group should have the highest average division ratios value in comparison to other descriptors in the same group. The minimum value (at least 1.5 times) must exist for any of two relative ratios. This value is expected to be enough to use for texture classification in order to extract the interesting object from the input image.

Best candidate descriptor=average[(max/med)+(med/min)] >1.5 (5.1)

Calculations for how to select the best candidate from each group (statistical and co-occurrence matrix) illustrated in tables 5.3,4 as an examples from seed squares selected in the input image in figure 5.1.

The algorithm will calculate these relative ratios and will select the highest average result from each group as best candidate. From both tables the highest average relative ratios are third moment from table 5.3, and contrast from table 5.4.

Table 5.3 Statistical features calculations and their relative ratio

	σ	third-mom	u	e
road	8.775	275.9111	0.0334	5.1392
ground	29.4029	5174.128	0.0098	6.8278
sky	9.4695	73.9777	0.0289	5.2059
max/med	3.105	18.7529	1.1562	1.3116
med/min	1.0791	3.7297	2.9569	1.013
average	2.0921	11.2413	2.0565	1.1623

Table 5.4 Spatial structure features calculations and their relative ratio.

	C	IM	U	e
road	0.0016	0.1443	0.0014	9.8273
ground	0.0042	0.0977	0.0004	11.3734
sky	0.0163	0.2427	0.0144	6.8078
max/med	2.6905	1.6813	9.9848	1.1573
med/min	95.8896	1.4779	3.3314	1.4435
average	49.29	1.5796	6.6581	1.3004

5.5 Acquisition of relevant data

In spite of the fact that the inspect windows are located onto the representative positions, the only one value of descriptor per class cannot be good representation of the whole class which may carried no information's about the whole class, especially no body guarantee for next neighbor square texture type for this class and by the way this is still the biggest challenger for computer vision applications. The second spite, threshold calculations which done by intersecting of a Gaussian approximations curves for seed squares 64*64 has lower populations , and this is hasn't statistical main, because from statistical point of view it's function of population number. In order to acquire some statistically relevant information, and avoid complete training procedure to store all possible texture types, the next step consists in calculation of the values of selected three descriptors over the set of windows around the initial one. The initial square is shifted in all four main directions for 16 pixels, pixel by pixel, producing the populations of 1024 values of selected best descriptor per each class. This step has a nature of simple training procedure – the set of 1024 values carries the information how the particular descriptor is distributed over all three areas. The histograms of statistical descriptor candidate calculations by 1024 values are shown in figures 5.5,6,7. Thus, the acquisition phase could be time consuming, the mechanization of appropriate calculations of statistical moments and co-occurrence matrices given in section 5.4 are made as iterative ones. In case of co-occurrence matrix acquisition, shift the square 16 pixels in four main direction and any new translated position of window, the result is obtained for previous one just corrected by the amounts related to disappearing and newly added rows/columns, and at same time subtracting number of translation from other side from rows/columns depends on the translation up/down or write/ left. Figures 5.8,9,10 show the shifting of seed square by 16 in each direction to read 1024 values by contrast calculations with square sizes 64*64, 32*32, 16*16, and their histograms for three classes with each square size. All histograms for both descriptors show how it distributed over all classes and shows the kind of distributions symmetrical or bimodal.

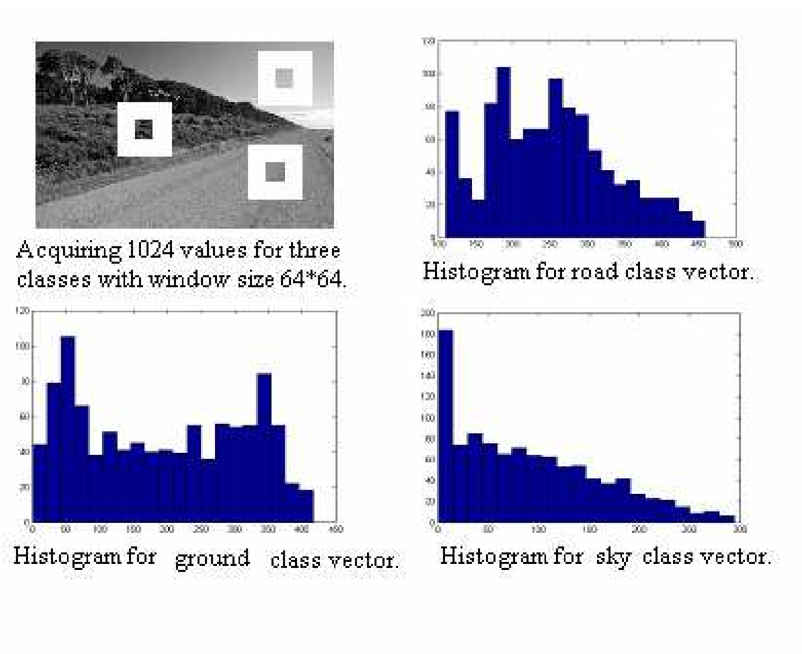


Figure 5.5 Statistical third moment reading 1024 values by three windows size 64*64 and their histograms.

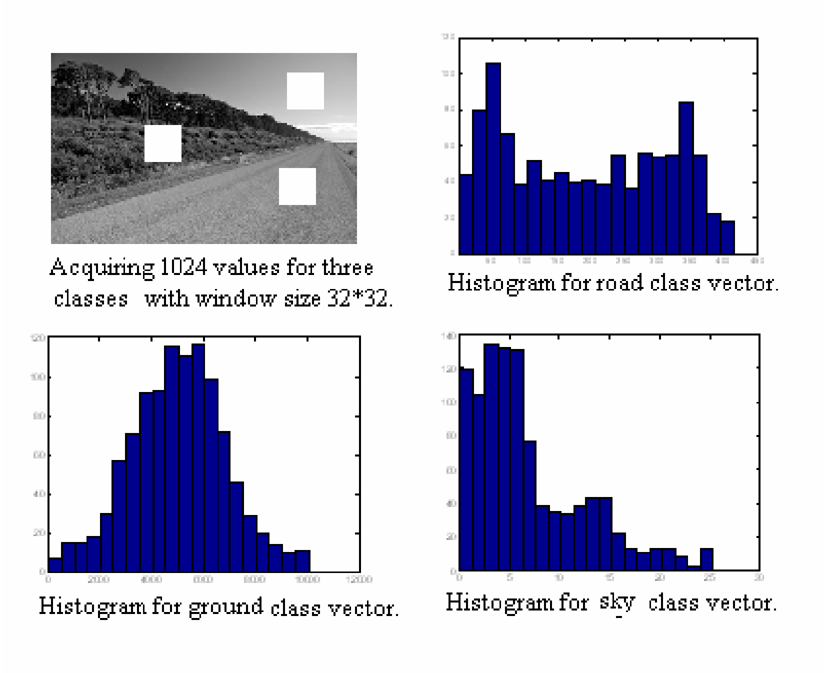


Figure 5.6 Statistical third moment reading 1024 values by three windows size 32*32 and their histograms.

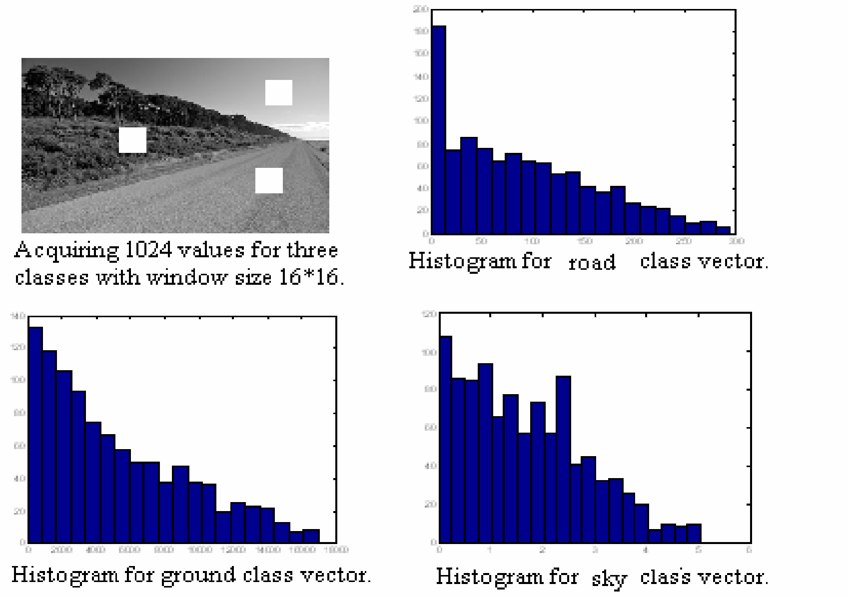


Figure 5.7 Statistical third moment reading 1024 values by three windows size 16*16 and their histograms.

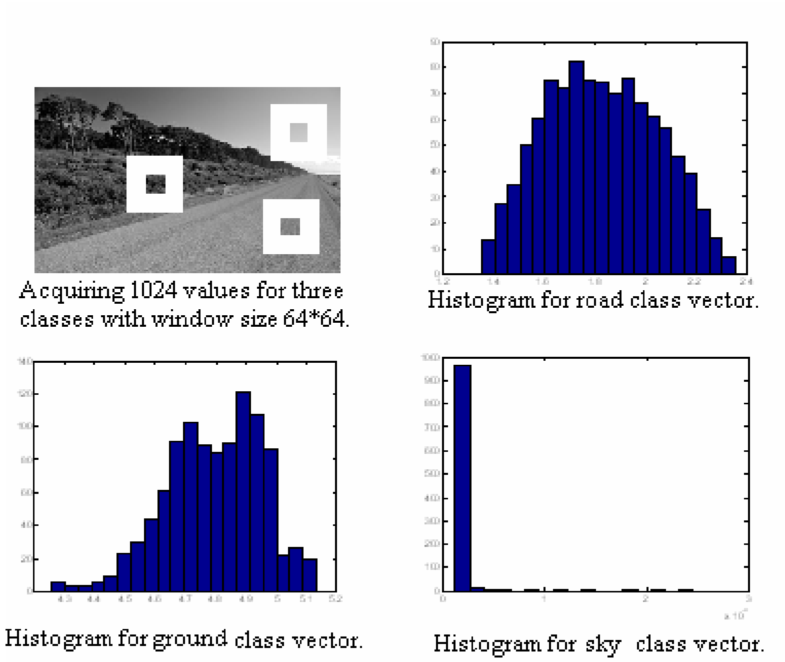


Figure 5.8 Co-occurrence descriptor (contrast) reading 1024 values by three windows size 64*64 and their histograms.

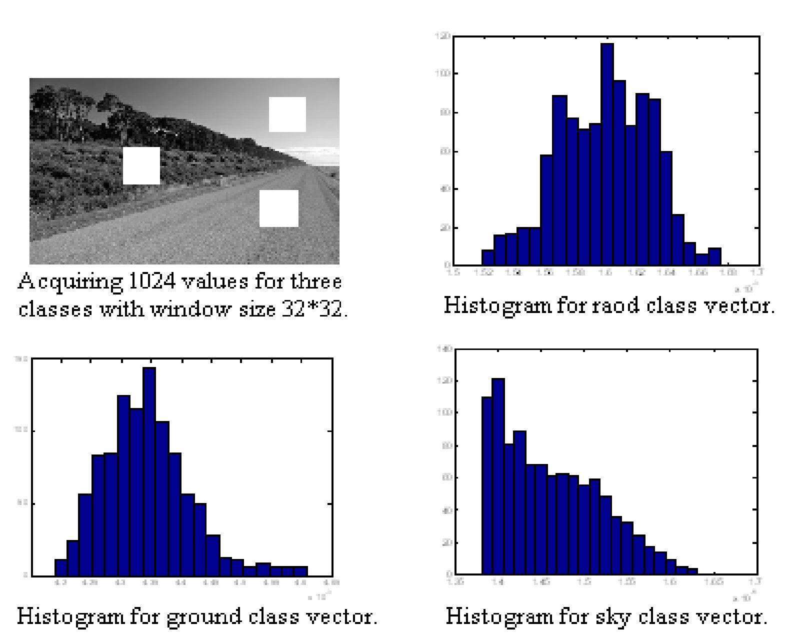


Figure 5.9 Co-occurrence descriptor (contrast) reading 1024 values by three windows size 32*32 and their histograms.

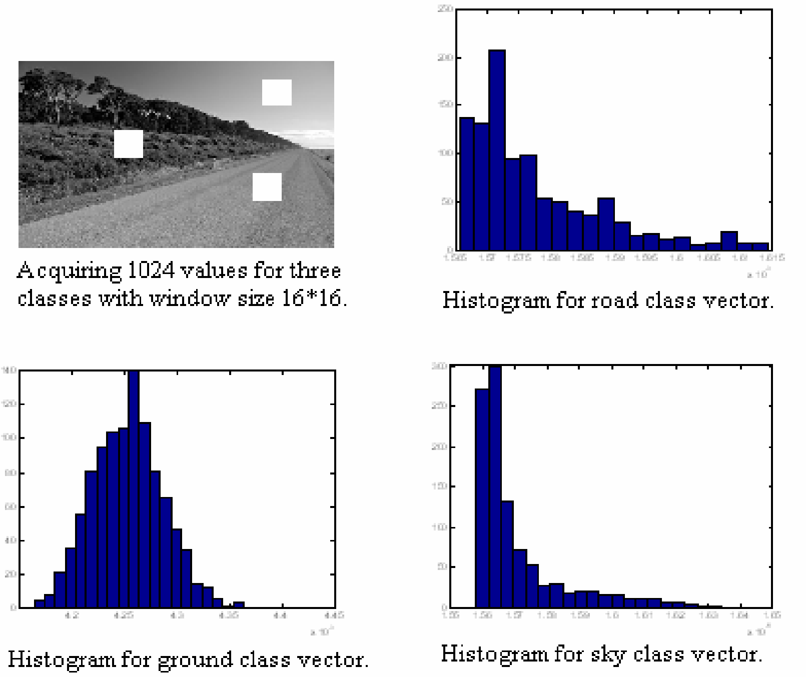


Figure 5.10 Co-occurrence descriptor (contrast) reading 1024 values by three windows size 16*16 and their histograms.

Actual distributions of descriptor values inside the areas encompassed by these 1024 windows can be different. Sometimes it is not symmetrical around the mean value, or it could be even bimodal if the whole area around inspect window encompasses two textures "different enough". The last case is the most frequent for near background class, although it can be obtained for the sky (clear/cloudy) and for non-uniform road texture also. After the mean values and standard deviations are calculated for both descriptors and all three classes, the assumption that the values of descriptors are normally distributed is adopted then. A number of experiments have verified that this approximation used in order to specify the threshold value (or separation line between classes) was acceptable even if the actual distributions have not been Gaussian ones like as shown from previous figures; some of histograms are not symmetrical at all. Segmentation based on the identification of particular probability density functions would be more adequate way, but both the identification phase itself as well as the more complex way of calculation of separating points/lines would result in a meaningful consumption of computing time. The following tables 5.5,6 illustrate results of (mean values and standard deviations) obtained for statistical descriptor (third moment) calculations with 1024 values and spatial structure (co-occurrence matrix) descriptors (contrast) calculations with 1024 values, for three different window sizes (64*64, 32*32, and 16*16) to the seed-squares shown in figure 5.1.

Table 5.5 Statistical parameters of distribution of third moment values (mean, σ) for different window sizes.

	m_{64x64}	σ_{64x64}	m_{32x32}	σ_{32x32}	m_{16x16}	σ_{16x16}
road	250.0199	82.7304	191.5258	120.9077	91.4344	72.3055
gnd	4149.8901	1948.0874	5048.0863	1810.8277	5173.0226	4138.7577
sky	95.2672	57.0091	6.9028	5.6123	1.6761	1.1831

Table 5.6 Co-occurrence matrix parameters of distribution of contrast (mean, σ) values for different window sizes.

	m_{64x64}	σ_{64x64}	m_{32x32}	σ_{32x32}	m_{16x16}	σ_{16x16}
road	0.0016	0.0016	0.0015	0.0016	0.0016	72.3055
gnd	0.0038	0.004	0.0039	0.004	0.004	0.0001
sky	0.0000198	0.0000314	0.0000142	0.000000692	0.0000155	0.000000216

5.6 Thresholds setting

The setting of thresholds is the proper numeric values can put to separate the image classes between each other with low distortion. In image segmentation based on texture features very difficult to find exact threshold because each image region has different gray level values even inside same class as in the road called non-uniform road textures, especially when there are some natural or artificial obstacles on or beside the road, light reflections, shadows, etc.

When to define the separating line between classes, the simplest idea is to specify them in the middle of distance between prototype values of descriptors (either directly from inspect windows, or based on calculated mean values) for example in one dimensional descriptor space, if there are three defined classes in the road image, the extracted class for our case is a middle values of t1 and t2 as:

$$t1=m1+(m2-m1)/2,\ t2=m2+(m3-m2)/2 \qquad (5.2)$$

$$d1=m2-ct1;\ d2=m3-ct2; \qquad (5.3)$$

Where m1, m2, m3 are the mean values for each class in tables 5.5,6, or could be the proto-type values of windows for each class from tables 5.1,2, t1, t2 are threshold values to separate the three distributions, and *ct1, ct2* midpoint thresholds.

The threshold values are set on the points where the probability of the membership to both neighboring classes is equal (at the intersections of particular probability density functions). Assuming that the partial probability density functions are of Gaussian type, the overall probability density function can be represented as their mixture as shown in equation(4.4). To find the threshold values by solving binomial equation (4.5).

The boarder value, which is lies in between of two mean values, is chosen between two solutions of equation (4.5). The formal application of this approach is sometimes not guaranteeing acceptable results. Therefore, we introduced the following exceptions:

- if both standard deviations are much less than the half-distance between two mean values – the threshold is adopted in the middle between two mean values as in equation 5.2.;
- if just one among standard deviations is much less than the half-distance while the other one is comparable with, the threshold value is shifted toward the class characterized by small value of standard deviation (in the amount of one quarter of distance between mean values, relative to mid-point);
- if at least one standard deviation is higher than the distance between mean values – it is an implicit sign that the distribution(s) are not Gaussians and the threshold is adopted at the middle point again; (Otherwise, for example, the solutions of eq.(4.5) could be completely outside the range defined between two mean values).
- Only if both standard deviations are comparable to the half-distance between mean values, the solution of eq. (4.5) is used as a threshold value. The calculation of the threshold value is a function of its distribution of Gaussian approximation for that class value only if they are really normal ones. Figures 5.8,9show distributions of both best candidate descriptors (third mom. and contrast) overall 1024 value, and all Gaussian are symmetrical around the mean values *m1, m2, m3.* These figures also shows the standard deviation from the

mean values and its clear that the sky and road classes are less than background deviation which it has large sigma. All mean values are sorted in ascending order m1 represent sky mean, m2 represent road mean, and m3 represent background mean and also for standard deviations($\sigma_1,\sigma_2,\sigma_3$) with same miner, figures 5. 11,12,13 and figures 5.14,15,16 shows mean and standard deviations for three classes with both best candidates by three different square sizes(64,32,16) respectively. The thresholds are shown on these figures t1,t2 are middle points between distributions with out any intervention.

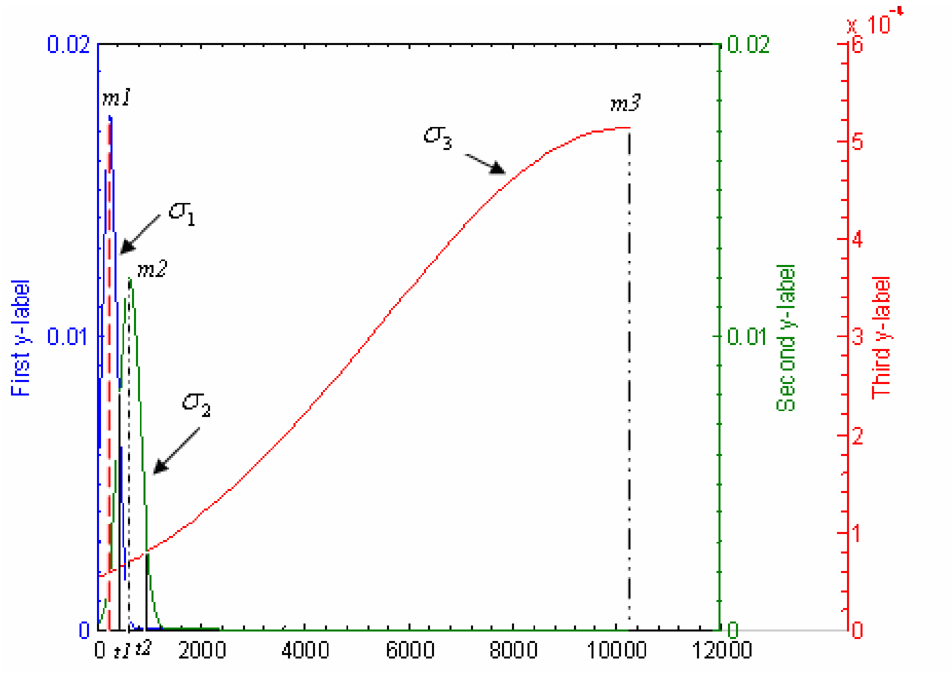


Figure 5.11 Normal distribution functions of three classes with third mom. at square size 64*64.

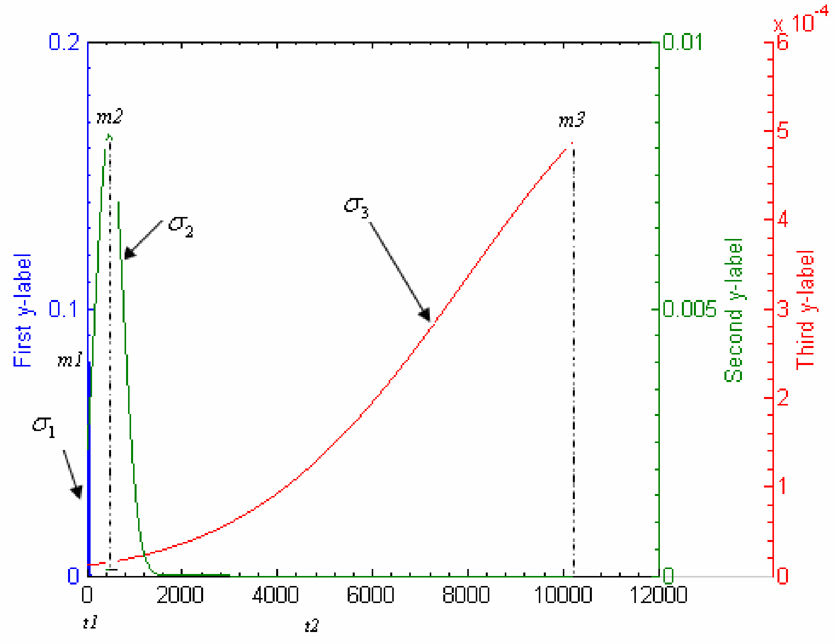


Figure 5.12 Normal distribution functions of three classes with third mom. at square size 32*32.

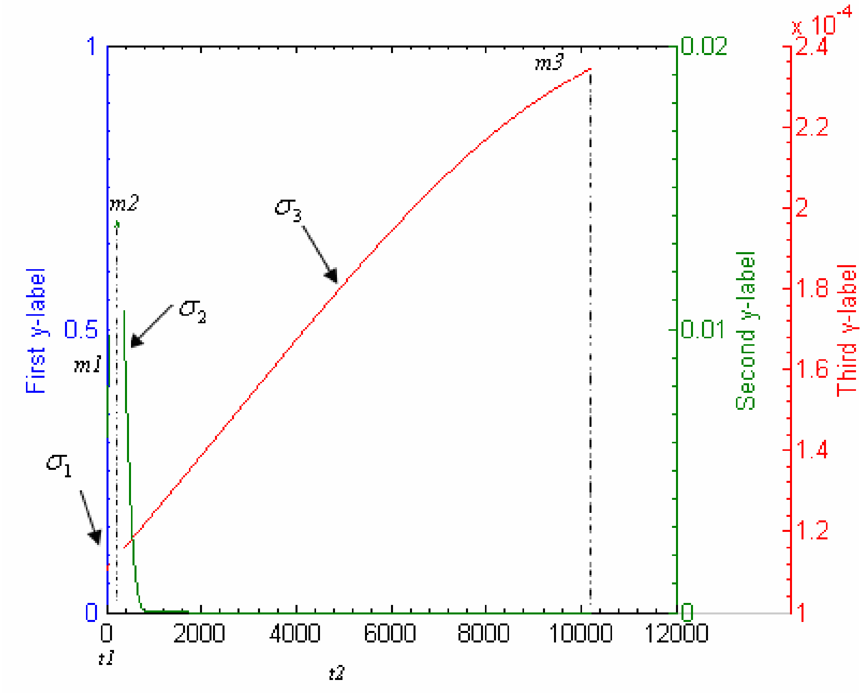


Figure 5.13 Normal distribution functions of three classes with third mom. at square size 16*16.

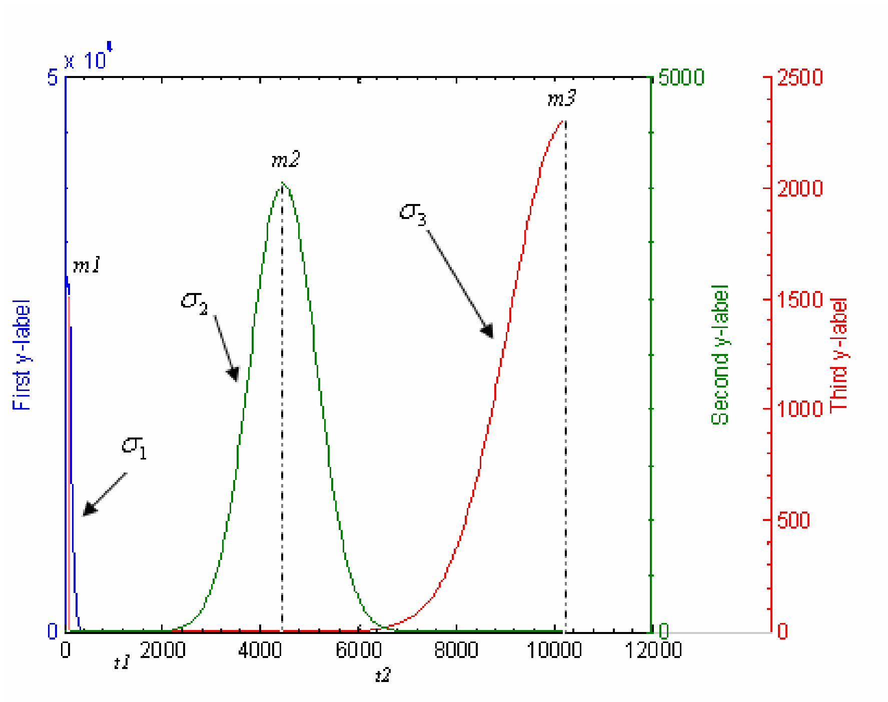


Figure 5.14 Normal distribution functions of three classes with contrast at square size 64*64.

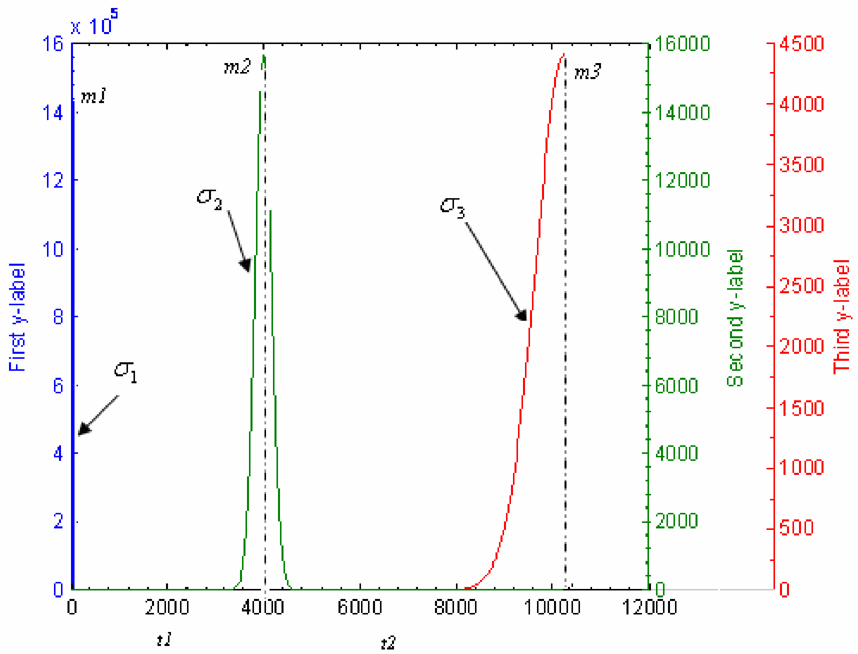


Figure 5.15 Normal distribution functions of three classes with contrast at square size 32*32.

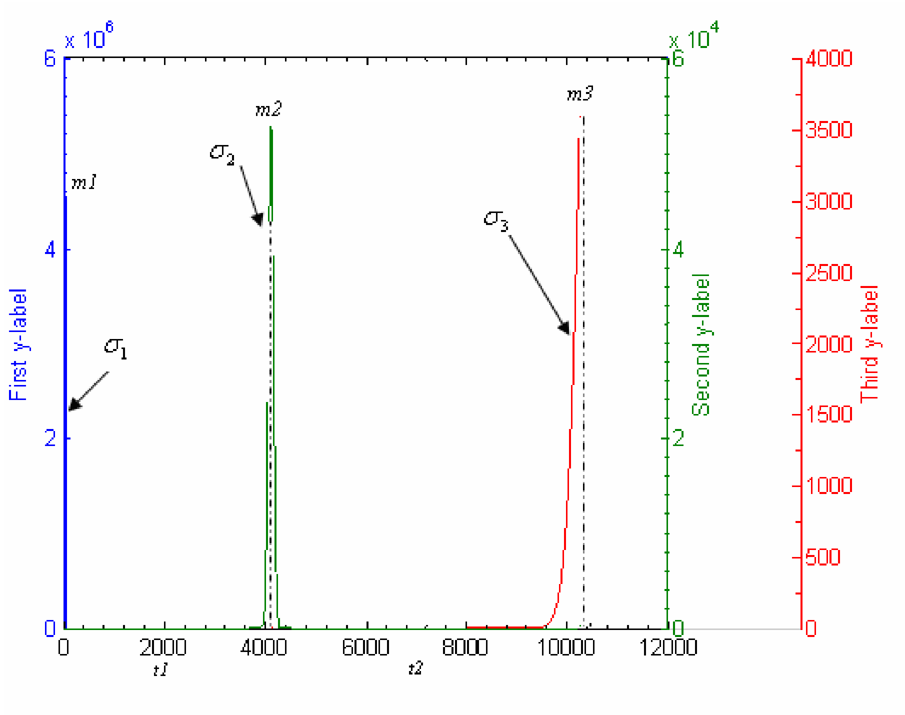


Figure 5.16 Normal distribution functions of three classes with contrast at square size 16*16.

If the classes are very well separated, the Euclidean distance is more applicable also. The flowchart shown on Figure 5.17 is an illustration of the above mentioned specification of the threshold value (*T*) discriminating two classes represented via their mean values (M1, M2) an (M2, M3) on the distance *d1 and d*2 respectively. This done for all three square sizes 64,32,16, and the threshold values T1, T2 are calculated with third mom. and contrast descriptors as in tables 5.8,9. For both thresholds according to previous steps are made some adaptations or considered as mid values between the main values. The algorithm has capability to works for either statistical texture descriptors or spatial structure (co-occurrence) group as one dimension as shown in above figures 5.11-16 or two-dimension descriptors space used both best descriptors from both groups.

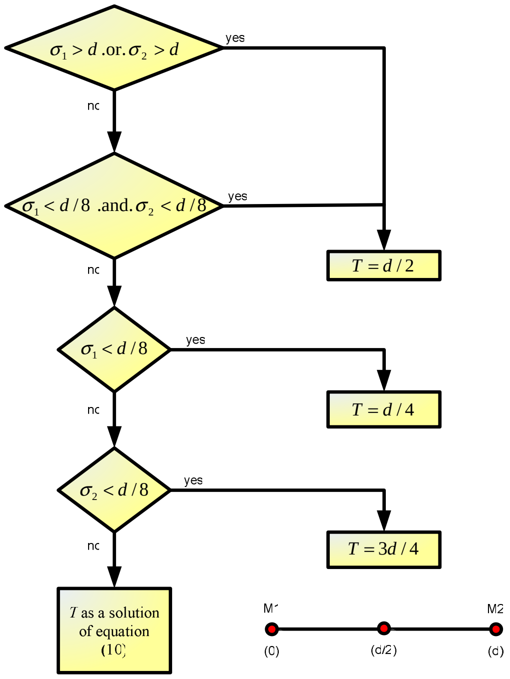


Figure 5.17 Flowchart for specification of the threshold.

The algorithm works under the driver control for selecting three representative proto-type values with square size 64or32, selecting the best candidate descriptor, and calculating the mean values via 1024 acquisition around the proto-type positions with three different square sizes(64, 32, and 16).

Table 5.8 Third mom threshold values for different square sizes

Window size	T1	T2
64*64	172.64	2199.95
32*32	99.21	2619.80
16*16	5.35	726.63

Table 5.9 Contrast threshold values for different square sizes

Window size	**T1**	**T2**
64*64	0.00092	0.0033
32*32	0.00080	0.0029
16*16	0.0008	0.0029

The system can start segmentation process to the input image by a specific window size (64 or 32) to separates interesting classes. In previous sections, I used the best statistical descriptors as well as best special descriptors, but there is no unique descriptor among them to be extracting all class interest from the input image. Attempting to find solution, use both best descriptors from both classes, but also no body can guarantee for using more than one descriptor to get better result compared to one-dimension descriptor result. When the classification used both descriptors, it is called *tow-dimensional-descriptors space*. The idea of using two-dimensional descriptor space, separating curves can be defined as the straight lines (based on Euclidean distance from the prototype points–defined as the appropriate pair of mean values) as shown in figure 5.18 used best descriptors among two groups were used (third-moment and contrast). The idea here calculate the Euclidean distance between the each new calculation and proto-type values, take the minimum distance and considered belongs to the nearest class. Equation 5.3 shows how to calculate this minimum distance.

$$\begin{aligned}\min[\, & sqrt((mean_{R\_stat} - new\_value_{stat})^2 + (mean_{R\_coc} - new\_value_{coc})^2), \\ & sqrt((mean_{G\_stat} - new\_value_{stat})^2 + (mean_{G\_coc} - new\_value_{coc})^2), \\ & sqrt((mean_{S\_stat} - new\_value_{stat})^2 + (mean_{S\_coc} - new\_value_{coc})^2)] \quad (5.3)\end{aligned}$$

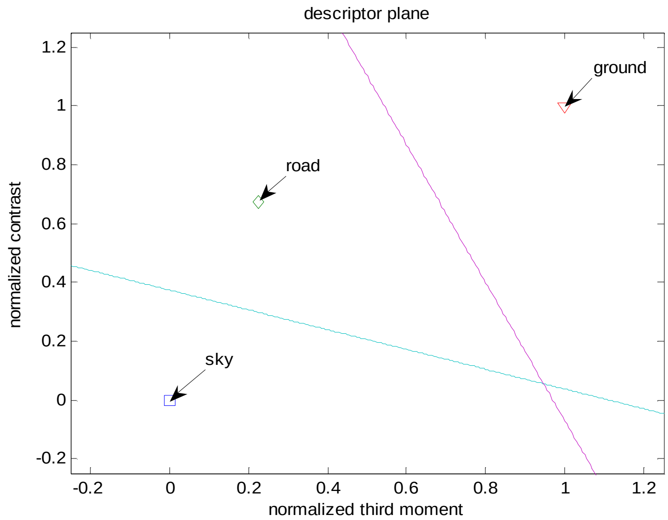


Figure 5.18 The classification used min Euclidean distance.

The second, using projections of the curves obtained at the intersections of two dimensional probability density functions (the typical 3D representation of distributions of values of two descriptors for three classes is represented on figure 5.19). In order to overcome numerical problems arising from the fact that ranges of candidate descriptors could be extremely different, the normalization procedure has been performed in the way that all absolute values of one type of descriptor (including mean values and standard deviations) are divided by the maximal mean value of particular descriptor (among three classes). The descriptor plane is now approximately reduced to unity square this way (still allowing some values outside 0-1 range). Two-dimensional distributions around normalized mean values have generally elliptic profiles. The same reasoning in threshold specification described above in one-dimensional case is now applied along the line connecting two particular normalized mean values (*Road-Sky and Road-Background*). Equivalent standard deviations ($\sigma_{RS}, \sigma_{SR}, \sigma_{RB}, \sigma_{BR}$) are calculated for the particular direction based on standard deviations known for both descriptors partially. The point defined this way determines the crossing of the discriminating straight line perpendicular to the line connecting mean values as shown in figure 5.20.

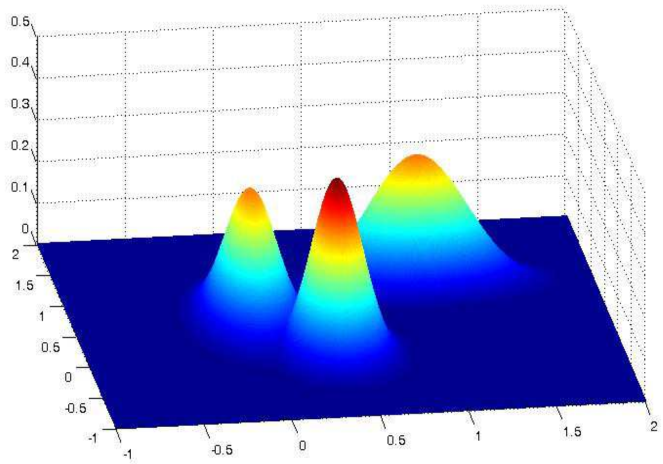


Figure 5.19 Typical 3D distributions of two normalized values of texture descriptors for three specified classes.

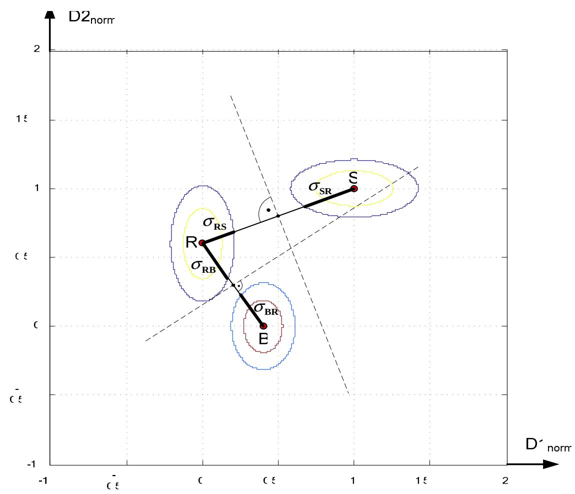


Figure 5.20 The example of discriminating lines extracting the road class in two- dimensional case.

5.7 Segmentation process

The algorithm starts segmentation from high left corner of the input image by specified square size and scan row by row until the end. According to the threshold values, the algorithm can discriminate the classes and store them in separate images (same size of input image) with '0' and '1' values. The output image for any class has approximate shape for this class in the input image, thus scanning by large square size (64, 32), but has benefits for decreasing the computation time. The following figure 5.21 shows the segmentation results used one dimensional descriptor, the best statistical descriptor (***third moment),*** and figure 5.22 shows the segmentation results used one dimensional descriptor, the best co-occurrence matrix descriptors(***contrast***).

From the output, images of deferent classes there are some images with disconnected windows. The algorithm will remove disconnected windows like in the segmented road in figures 5.21, 5.22 by applying some morphologic functions. Also will fill some holes inside the segmented class caused by some obstacles beside the road image or inside the road image for instance cars on the road or tree shadows, clouded sky, and reflections, etc. These functions are written a cording to assumption, the road class always in the input images at lower half of the image, apposite the sky (far background) always at upper half image. The road image segmentation result omits any disconnect window at the upper half of its output

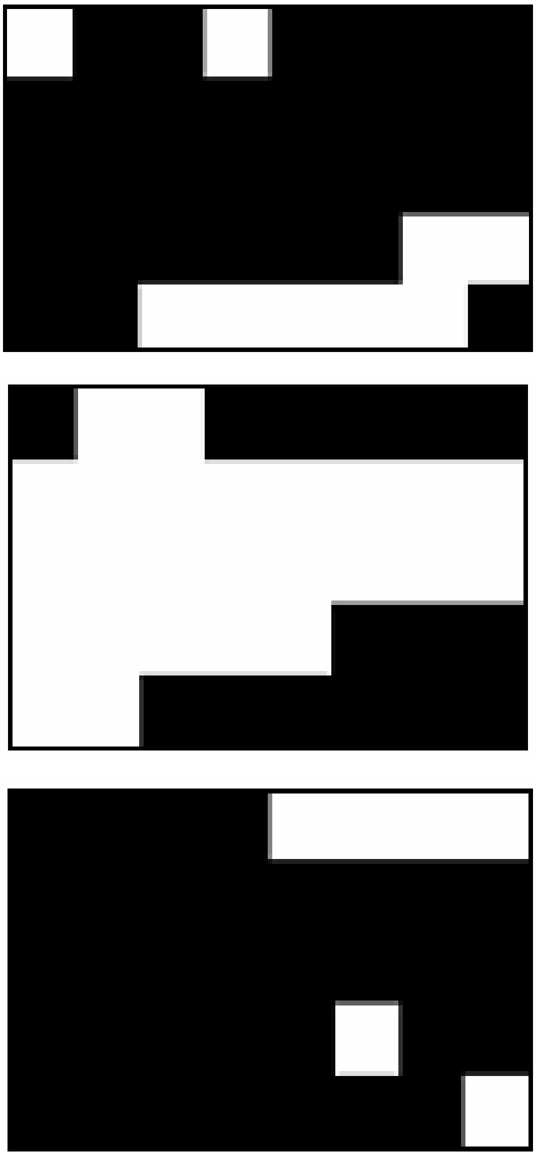


Figure 5.21 Segmentation results used one dimensional descriptor third mom for (road, ground, and sky) respectively.

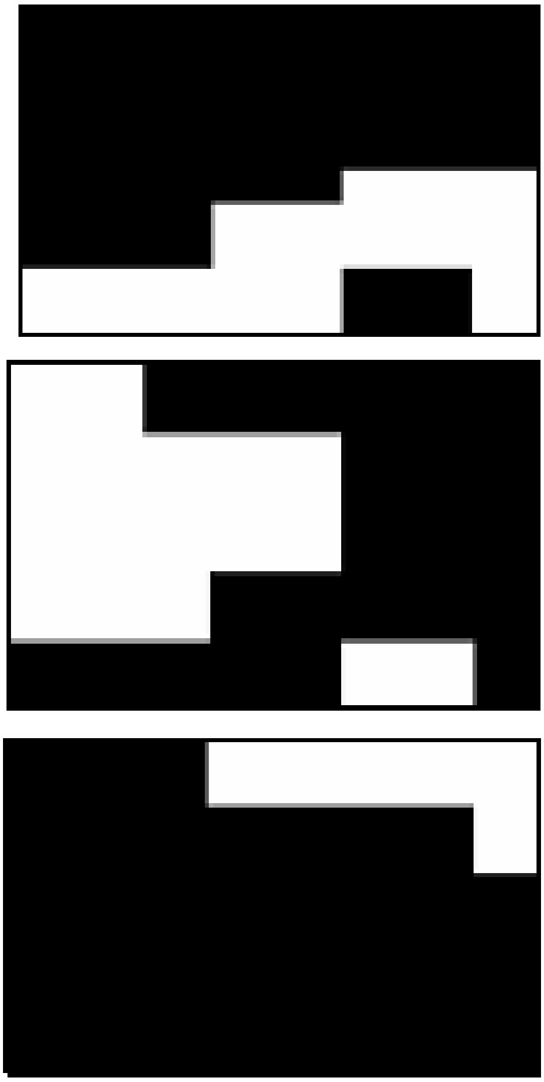


Figure 5.22 Segmentation results used one dimensional descriptor contrast for(road, ground, and sky) respectively.

image and will fill any connected windows available at the lower half of the output image using the pixel populations inside each window size with mean body. The following figure 5.23 shows the results functions for filling holes and removing disconnected windows inside extracted road class segmented by different descriptors. All these functions are built in the algorithm by MATLAB code.

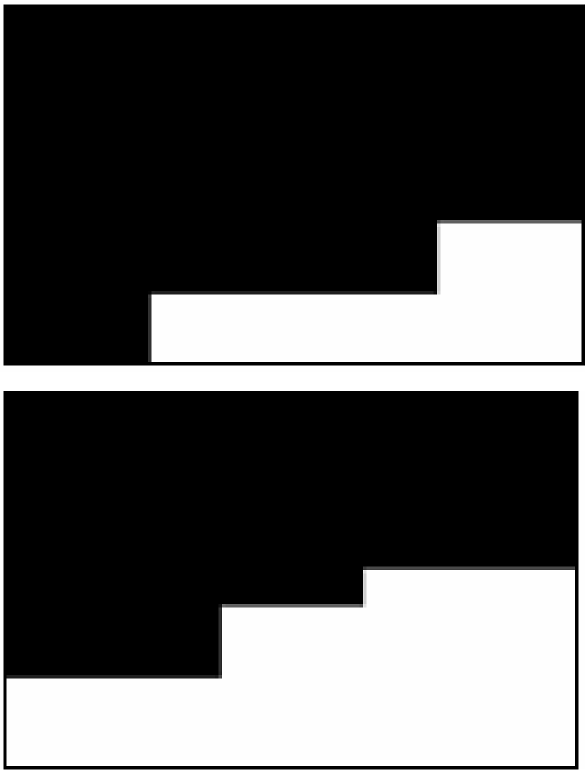


Figure 5.23 Filling the holes and removing disconnected windows from road class images that are segmented by third mom. and contrast respectively.

5.8 Merging/Splitting procedure

The objective of this algorithm toward a real-time applications, the scanning of the input image done by large window size (64x64). The output image for interesting class as we saw in previous figure has poor resolution – just rudimentary showing the areas belonging to road classes. The next step after segmentation so called Merging and spitting means segment with new window size and check by the specified threshold if it's merge it with old segmented mass othrer wise omitt it from the figure. The algorithm will do this step of segmetation only arround the border of the segmented class by one or more windows size upper and lower the class border. The procedure will resegment this spicfied area only by decreasing window size down into 32x32 and 16x16 and using the new threshod values for using one dimenssinal descriptor statistical or structural (t3, t4), (t5, t6) for square sizes 32, 16 respectively, but at two-dimenssinal descriptors space used minimuim Euclidean distance, but with new protype values(mean values) for windows size 32*32 and 16*16 .

If I consider the first segmentation is first pass by square size 64x64, so the next step called second pass segmentation(the result image from first pass) by square size 32x32, and the last step called third pass segmentation(segment only the squares are not satisfied the thresholds in second pass. Third pass segmentation start segment those squares with down square size 16x16, all values are fulfilled the criteria will add to the main class body, and omit squares are not. Moreover, further splitting of window would lead to refined border, but from the other aspect, further reducing of the size of window would deteriorate the statistical validity of data. Finally, the other extreme consisting in scanning with 64*64 window (statistically preferable) over the whole image, pixel-by-pixel, produces typical "over-segmentation" results – the border is "too precisely" extracted (too much dependent on local variations around the border) and there is a need for a lot of morphological processing after.

From first pass segmentation result, as we saw some morphological filtration is needed and producing segmented image with reduced spatial resolution. Here also after each splitting steps (second and third) needs to eliminate all portions of the image classified as road, in fact not connected to the real road body and some times belongs to the other classes, filling holes produced in side the mean body by (either due to shadows/reflections or due to existence of other vehicles). The following figure 5.24 shows intermediate result from second pass segmentation for road image.

Figure 5.24 Intermediate result for road image segmented by third mom.

5.9 Morphological functions

This type of image enhancement done after each segmentation result as we have seen from previous section, by removing non-connected windows or filling holes by some criteria. Another morphologic image processing function done at the end of splitting the segmentation result by square size 16*16 (refining the road border) by so called opening (erosion followed by dilation) function with structural element 17*17. finally, can extract the road border by erosion the result image with structural element 3*3 and subtracted from the original result as we can see from following figures 5.25,26.

Figure 5.25 Extraction of road borders used third mom.

Figure 5.26 Extraction of road borders used contrast.

5.10 Line fitting

This is the last step in the road border extraction to refine the staircase as shown from previous figure to approximate as straight line.

From the purpose of estimation of the distance from the border and road orientation (finding the vanishing point based on left and right borders), it is appropriate to fit the line through the set(s).

from extracted road borders by statistical third mom. descriptor in figure 5.25, there are three types of points could fit a line through them as shown in figure 5.27, the circle points is called leftmost points, stars points is called rightmost points, and diamond points is called mid points(average of leftmost and rightmost). Here I used leftmost points as for both figures in 5.25,26. According to the type of morphological processing was used, the set of most informative points along the stairs was consisting from:

- the "leftmost on the each stair on ascending part" and
- the "rightmost on the each stair on descending part".

Figures 5.28, 29 illustrate the results for line fitting into third mom and contrast results for figure 5.21,22 respectively, using leftmost points.

If one is interested in more sophisticated results (i.e. positioning of the vehicle on the map, vehicle's speed control according to the actual road curvature, etc.) some type of polynomial approximation through the complete set of points is needed (most frequently, binomial).

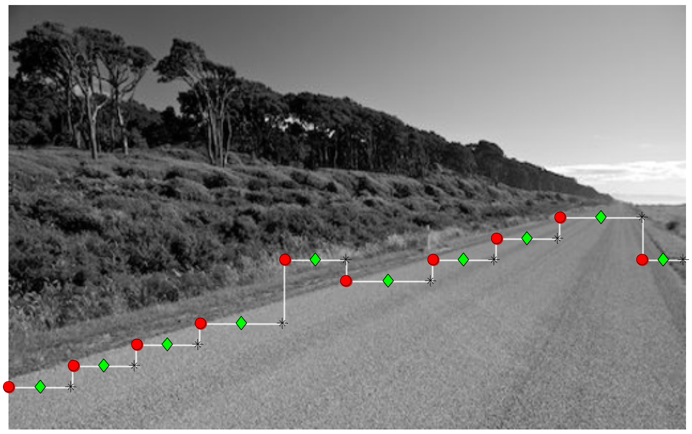


Figure 5.27 All possible points of line fitting

Figure 5.28 Line fitting to the staircase in figure 5.21

Figure 5.29 Line fitting to the staircase in figure 5.22

In the following an example using two dimensional descriptor space to extract the road borders. These descriptors are calculated from above which the best candidates among their descriptors statistical and spatial structure groups. The input image for this example same as in figure 5.1 with same seed squares. The calculations of best candidates among both descriptors group are *third mom.* and *contrast*. In two dimensional descriptors space the thresholds will calculated same idea in one dimensional descriptor space, but with minimum Euclidian distance, or by determines the crossing of the discriminating straight line perpendicular to the line connecting mean values. In this example thresholds used the minimum Euclidean distances to segment the road class from other image classes based texture descriptors. The segmentation results for each class are shown below in figure 5.30.

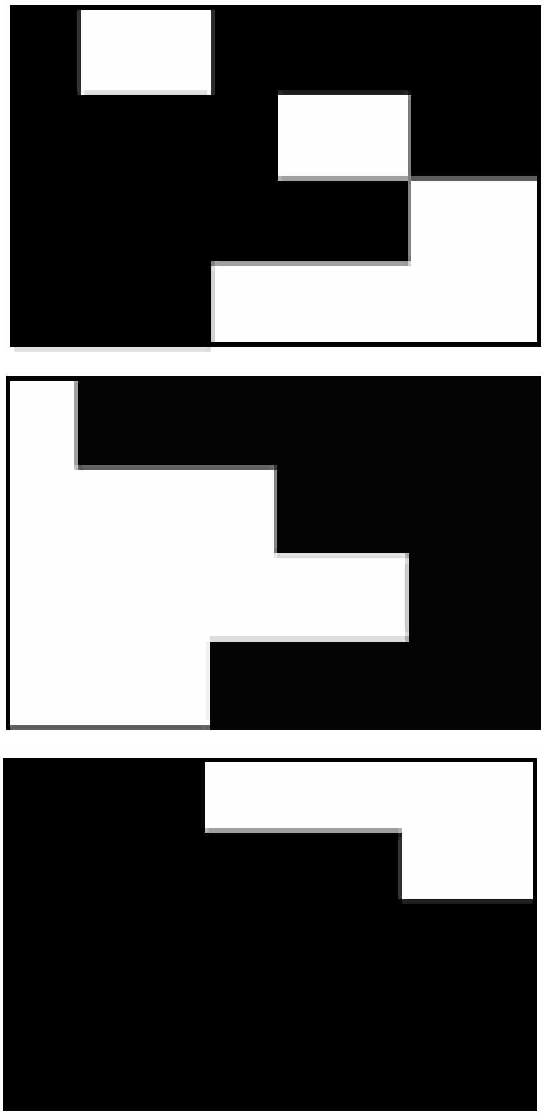


Figure 5.30 Segmentation results used two-dimensional descriptors for (road, ground, and sky) respectively.

The next step of the algorithm, process the segmented result by removing non connected windows and filling holes in the main body. This step of process as one of morphological image processing which are built in the algorithm code to fill holes inside the main body of the segmented class or removing unconnected windows created by obstacles like some reflection on the concrete, shadows, etc. Figure 5.31 shows the filtering result for segmented road class.

Figure 5.31 filling holes and removing disconnected windows from road class images segmented by two dimensional descriptor space.

Splitting procedure and merging are done only on the road class borders to refine the crude image and produce stair case image. This new segmentations done by using new sizes square windows 32*32 and 16*16. The thresholds also used a new calculated threshold by new main values or seed squares window sizes. Figure 5.32 show the intermediate result after splitting and merging with window size 16*16.

The result from this procedure are staircase image because the algorithm will remove all disconnected windows and filling holes in the main body by using morphological image processing for all classifications methods one, or two dimensions descriptors has same number of functions(number of erosion and dilation), figure 5.33 shown extracted road image with staircase borders.

Figure 5.32 Intermediate result used window size 16*16.

Figure 5.33 Extracted road image with staircase borders.

Finally, the algorithm will refine this staircase borders by fitting straight line approximations through most left points as shown in figure 5.34 to show more acceptance extracted road borders results .

From figure 5.35 shows the final extracting road borders from input image using best candidate's texture descriptors from both groups in two-dimensional descriptors space. The texture classifications used here minimum Euclidian distance. It is obvious from the figure, these road borders extraction are superimposed on the real road borders which are more acceptable result compared to the previous results used one dimensional descriptor either by statistical or co-occurrence matrix feature, but doesn't mean this valid for all road

Figure 5.34 Leftmost points used to fit line on staircase.

frames on the same class region, because some times vise versa. Moreover, benefits of using two-dimensional descriptors is in more information, and better results thus, each descriptor from texture point of view has own classification description of texture features like statistical groups describe texture features by gray levels histograms and its deviations from the mean on particular area, but spatial structures descriptors describe texture by the positions of pixels has same gray level values, which may describe the different texture region in same class by same previous seed squares.

Figure 5.35 Extraction of road borders used two-dimensional descriptors space.

6. Algorithm verifications.

6.1 Introduction:

In this algorithm for detection of the road image borders based on texture classification has a number of principal artificial and naturally problems to be mentioned as algorithm robustness and limitation, in order to discuss what other criteria and reasoning should be included as a kind of algorithm upgrade depending on particular obstacle. In computer vision field used texture features in segmentation considers a big challengers, because no one could detect in prior, the next region or image frame of the processed scene which kind of texture belongs. In the case of extracting road borders, some strategy scenario should be taken in mind, like suddenly, discontinuities of the road borders as in (duplication of faster car, opposite direction car, and some obstacles beside road. The algorithm should follow in these cases old extraction road borders results for some future frames. Thus, this algorithm is real time oriented in future, so the limited number of classes, restricted number of training is done only at begging of algorithm starting by the operator. These prototype squares windows sized by 64*64, but in some scene images the operator can't fit this size into representative place of particular class. The Figure 6.1 illustrates typical problems related to these facts: The area of sky is not wide enough to provide placing of a set of 64*64 windows; There is a slope of the road visible in the upper part of image, and the further part of road is undistinguishable from the background, because the scanning by large window size even by 32*32. The result given on Figure 6.2 is obtained after initial placing of

32*32 windows and subsequent splitting down to 16*16 and 8*8 pixel size. The main conclusions that could be derived from this example would be:

1)Obvious, from this figure the road class surrounded completely by the background so, it is not relevant to have more than two classes – classification to "road" – "non-road" regions can be acceptable;

2) It is true especially if the attention should be paid to lower part of the image dominantly (there is no contact between the road and further background).

3) Such image has more than three dominant classes (concrete, grass, and mounting, sky).

Figure 6.1 Shows images have restriction for windows size.

Figure 6.2 Extracted borders before slope point.

In the next example in figure 6.3, where the road area includes visible lane borders and more dominant surrounding between road and non-road classes. Comparing to using this algorithm in such cases, it's more expensive and combinations time consuming if some kind of gradient algorithm have been used. The detection of a lane border would be easier in these cases. On the contrary, if the region texture is used for description, the existence of visible lane borders makes this task to be a little bit harder – declaring by texture that the white lane borders are different from the road itself. The result shown on Figure 6.4. is illustrating this type of road environment. One can conclude that the border lines detected by this algorithm would tend to be a little bit inside the road region relative to the actual ones.

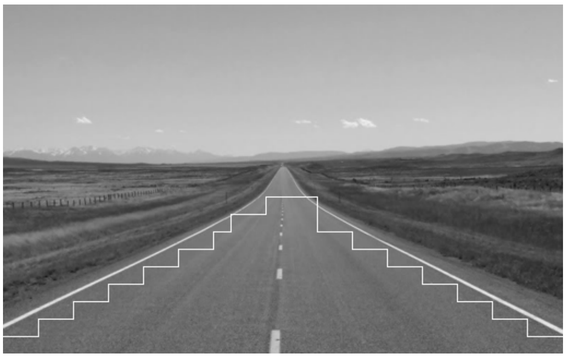


Figure 6.3 Extraction of the road borders only inside road lanes.

Figure 6.4 Extraction, where the lane borders are visible and final borders fitting.

The existence of visible white lines separating the tracks in typical highway environment can generally produce the problems because of the texture that is different enough in comparison to the typical road texture. Thus, the algorithm is sensitive to these cases and robust enough, and in some images, the elements of a dashed line would produce “the holes on the road segmentation result” that can be efficiently eliminated by morphological processing during or before starting splitting procedures. As in the case of figure 6.5, in this case of image, the area of doubled medial white line has been recognized by the algorithm as a non-road class. Thus, this area has more white intensity levels, allowing the separation of physical one road into “two roads”. This case can be considered as a particular one, but it is obviously allowing some benefits, and the algorithm should be able to recognize the possibility of more precise extraction of “road tracks” instead of the “road” as the single image area as shown in figure 6.6.

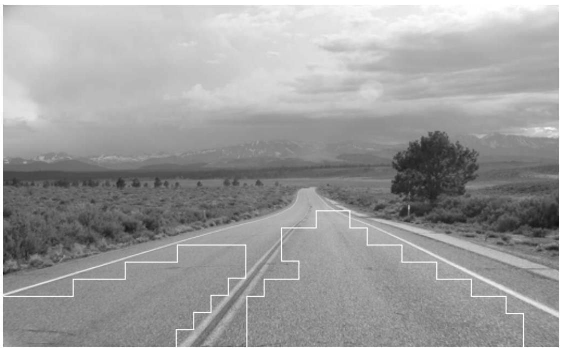


Figure 6.5 Highway scene where two road tracks have been extracted.

Figure 6.6 The Extracted borders are intensified in black.

Taking in count, the natural obstacles on the road, the algorithm should robust enough to manipulate and overcomes most of those obstacles, such obstacles sunlight reflections, raining, and shadowing. Speaking from general "texture point of view", it is obvious that the most serious problems could arise in the cases of reflections/shadows existing over the road image – making its texture to be non-unique for the whole road area. Like this problem, sunlight reflection as shown in the test image on figure 6.7. is a typical example of such an environment. In spite of the fact that this image was not produced by camera mounted on board a road vehicle, it was used here to illustrate the typical problems that could be encountered.

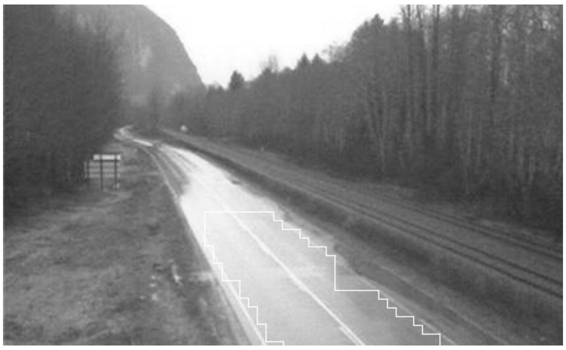


Figure 6.7 Segmentation of road class with reflection.

Figure 6.8 Extracted borders are intensified in black.

Two main conclusions can be derived from this example.

1. Reflections/shadows would make that the road area can be estimated as narrower than it is as shown in figure 6.8 – leading toward more conservative conclusions/actions.

2. The further parts of the road are not distinguishable because it's texture are more closer to sky class than road class descriptor texture – the algorithm can produce reliable results in some local area where the road texture is approximately uniform relative to chosen descriptors. This is a general conclusion arising from the previous examples also.

Moreover, the reflections from wet concrete can produce their own typical problems, while, the situation characterizing country roads with no concrete cover of the road can produce the problems related to the poor distinguishing between the road and local background and this is another challenging to this algorithm, although our algorithm must applied only on will concrete roads. Typical example of this type is shown on figure 6.9**.** Thus, some parts of road side banks can be added to the road area, while actually, there are no big qualitative differences between them.

Relevant points used for the line fitting are specified in Section 3.6 as the "leftmost on the each stair on strictly ascending part" and the "rightmost on the each stair on strictly descending part". They are illustrated on figure 6.9 by superimposed white circles. The road borders are extracted as shown in figure 6.10**.**

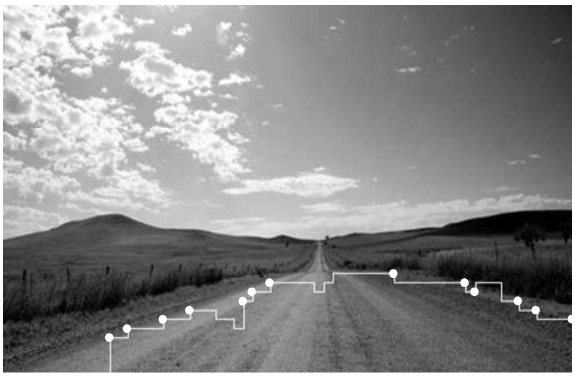


Figure 6.9 extracted country road borders as staircase scene .

Figure 6.10 Extracted country road borders.

Finally, we shall discuss here the abilities of the algorithm to extract the road region in a typical urban environment as it is shown on figure 6.11, which are contains on deferent
kinds of objects beside the road, means that deferent kind of textures. Urban road usually not as highway road, it have a number of particular problems (shadows produced by trees, typical signs marking the road tracks, and buildings, so the result poor separability of further part of a road,...) are present here. While both one-dimensional approaches in classification can give acceptable results as figures 6.11 shows statistical descriptors classifications has limited classification because, some shadows so the road not completely segmented, and figure 6.12 shows special structure descriptors classifications which has improvement segmentation than statistical but, still some road portions is segmented as not road and vise versa, their combination used in two-dimensional classification is showing visible improvement figure 6.13 comparing to one-dimensional results, figure 6.14 shows the result of final border fitting of two-dimensional classifications.

Figure 6.11 Result obtained using statistical descriptor only.

Figure 6.12 Result obtained by structural descriptor only.

Figure 6.13 Result obtained via 2D classification.

Figure 6.14 Final results obtained after border fitting into 2D classification.

6.2 Combining color and texture

Although, increasing number of computations and calculations and with out any guarantee to improve segmentation results, one solution introduced through this dissertation to integrative blue color with two texture descriptors and this work will be as three- dimensional descriptors. Beside texture, color is an issue not only in human vision but in digital processing where its impact is still rising. In contrast to intensity, coded as scalar gray values, color is a vectorial feature assigned to each pixel in a color image.

The parallel concept for color texture analysis separates the processing of both phenomena. Color is measured globally according to the histogram ignoring local neighboring pixels. Texture is characterized by the relationship of the intensities of neighboring pixels ignoring their color. The results of both analyses are combined subsequently to a feature vector. The parallel approach is advantageous, because the known methods of gray-scale texture analysis as well as pixel based color analysis can be applied directly. Up to now, parallel algorithms are most commonly used for image retrieval applications with color and texture as separated discriminative features. However, the view on texture as a pure intensity based structure is very simplified and disregards, for instance, colored texture primitives with constant intensity [29].

While this dissertation work focus only on extracting road borders with texture descriptors, but the extra work may apply the blue color in combination with texture descriptors. This would lead to three dimensional space, in order to overcome and get some segmentation improvements in some natural and artificial obstacles road images.

Figure 6.1 has image with slop and narrow road, and this is some algorithm limitations. Because difficult to segment the narrow road portion by selected minimum scanned window size 16*16, but with using third integrative blue color there is possibility to segment this narrow road portion with slop much greater than figure 6.1. The problems faced here how to fit straight line through extracted staircase with slop and narrow road. Figure 6.15 show the improved segmented road compared to figure 6.1. and figure 6.16 show fitting approximation line through staircase points.

Figure 6.15 Extractions of road borders obtained by three dimensional texture space.

Figure 6.16 Line fitting approximation to the previous result.

From figure 6.17 there are improvement to the extracted road borders compared to figure 6.10, Although, sloped and narrowed country road is there but, the algorithm extracted these portions by combinations with color. Line fitting is straight line that does not follow the road slope as obvious from figure 6.17.

Figure 6.17 Three-dimensional country road borders extraction

Conclusion

In this dissertation, the objectives have been oriented toward the algorithm of road boundaries extraction in the field of automatic ground vehicle guidance and driver safety assistance systems. The texture of the road region and other image regions is analyzed as a source of information. Two types of descriptors derived from the texture statistical and structural properties are calculated and the choice of the best candidates from both categories is made in an automatic manner. The threshold values discriminating the classes are obtained by an automatic procedure of acquiring of statistical information regarding chosen descriptors.

The algorithm is generally oriented toward real-time applications. This is the main reason why the number of texture classes is limited to three (road, near and far background), the refining by window splitting is done just around the rudimentary extracted border, the splitting is restricted down to the 16*16 window size, and morphological processing steps are based on simple connectivity criteria. Two-dimensional classification was chosen in order to balance statistical and structural texture features, increase the improvement of the segmentation and this approach has better extraction results in comparison to one-dimensional one in both descriptor types, and even with color. Some typical problems related to the irrelevance of third class, and to the non-uniqueness of the road region texture have been analyzed. In spite of the fact that there are a number of factors affecting the performances of the algorithm, it was shown that by the careful selection of algorithm steps and particular parameters, it was possible to obtain promising results. Conclusions derived from these examples have led toward the possible initial choices like: 1) region of interest is just the lower half of an image; 2) the number of classes is just two (road and non-road); 3) it is possible to obtain more than one road region; 4)it is enough to extract just one of two borders (if the other one is, for example, too short).

While the question related to the number of relevant classes is practically answered (the introduction of new classes representing more than one road region and/or more than one background region would be leading toward more complex calculations without general guarantees that the results are going to be improved), the number of descriptors can be still a subject of research. It seems that the adding of color as the third independent feature can provide more flexible approach in segmentation based on chromatic contrast between road and off-road (or obstacles).

In this dissertation many different aspects of using texture classifications were discussed and all results that have been shown are related just to the processing of static images that can be considered as an initial procedure. Basically, the algorithm should be in future functional in a dynamic scenario – which is the main objective of my actual and further work in this area, but the study duration and factor of time are limits make me to stop in this step. Dynamic nature (processing of the sequence of images) is creating some new problems, but it is also introducing some beneficial features. The variable existence of obstacles all over the road region and especially over the road border do introduces new problems. On the other hand, once extracted border in combination with rather robust scanning/splitting procedure around it and supported by proper prediction based on results obtained in previous frames, makes the main segmentation strategy promising acceptable results again.

[1] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison- Wesley, 2nd edition, 1992.

[2] Robert M. Gray and David L. Neuhoff. Quantization. *IEEE Transactions on Information Theory*, 44(6):1.63, October 1998.

[3] Digital Imaging Processing and Analysis: Human and Computer Vision application with CVIP tools.Аутор: Scott E Umbaugh

[4] L. Li**,** F. Y**.** Wang (Eds.)**,** Intelligent Vehicle Vision Systems , in *Advanced Motion Control and Sensing for Intelligent Vehicles, Springer*, 2007. (323-399).

[4] Nikhil R. Pal and Sankar K. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26(9):1277.1294, 1993.

[5] Chi_Yan_Pham. Fuzzy Algorithms: with Application to Image Processing and Pattern Recognition.

[6] Digital Image Processing. Third Edition. William K. Pratt.

[7] M. Bertozzi, A. Broggi, A. Fascioli: „Vision-based intelligent vehicles: State of the art and perspectives", *Robotics and Autonomous Systems* 32 (2000) 1–16

[8] G. N. DeSouza, A. C. Kak: "Vision for mobile robot navigation: A survey", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24, (2), pp. 237-267, (2002)

[9] D. Kuan, G. Phipps, A. C. Hsueh: "Autonomous Robotic Vehicle Road Following",*IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. Io, No. 5, September 1988, 648-658.

[10] K. H. Chen, W. H. Tsai: "Vision-Based Autonomous Land Vehicle Guidance in Outdoor Road Environments Using Combined Line and Road Following Techniques", *Journal of Robotic Systems*, Volume 14, Issue 10, pages 711–728, October 1997.

[11] J. Zhang; H.-H. Nagel: "Texture Based Segmentation of Road Images", *Proc. Symposium of Intelligent Vehicles*, pages 260 – 265, Paris, France, October 1994

[12]C. Rasmussen: "Combining Laser Range, Color, and Texture Cues for Autonomous Road Following", *Proc. IEEE Inter. Conf. on Robotics & Automation*, Washington, DC, May, 2002

[13]C. Rasmussen: “Grouping Dominant Orientations for Ill-Structured Road Following”, *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (CVPR’04).

[14]M. Oussalah, A. Zaatri, H. Van Brussel: “Kalman Filter Approach for Lane Extraction and Following”, *Journal of Intelligent and Robotic Systems* Vol. 34**,** 2002, pp. 195–218.

[15]Y. Wanga, E. K. Teoha, D. Shen: “Lane detection and tracking using B-Snake”, *Image and Vision Computing,* 22 (2004), pp. 269–280.

[16]J. Lu, M. Yang, H. Wang, B. Zhang: “Vision-Based Real-Time Road Detection in Urban Traffic”, *Proc. SPIE* Vol. 4666, pp. 75-82, Real-Time Imaging VI, 2002.

[17]T. K. Xia, M. Yang, R.Q. Yang: “Vision Based Global Localization for Intelligent Vehicles”, *Proc. of Intelligent Vehicles Symposium* 2006, (pp. 571-576). June 13-15, 2006, Tokyo, Japan.

[18]Z. Jiaa,_A. Balasuriyaa, S. Challab: “Sensor Fusion-Based Visual Target Tracking for Autonomous Vehicles with the out-of-Sequence Measurements Solution”, *Robotics and Autonomous Systems* 56 (2008) pp. 157–176.

[19]A. Broggi, S. Berte: “Vision-Based Road Detection in Automotive Systems: A Real-Time Expectation-Driven Approach”, *Journal of Artificial Intelligence Research* 3 (1995), pp. 325-348.

[20]R.Marzotto, P. Zoratti, D. Bagni, A. Colombari, V. Murino: “A Real-Time Versatile Roadway Path Extraction and Tracking on an FPGA Platform”, *Computer Vision and Image Understanding* 114 (2010), pp. 1164–1179.

[21]D. Puig, M. A. Garcia: “Automatic Texture Feature Selection for Image Pixel Classification”, *Pattern Recognition* 39 (2006), pp. 1996 – 2009.

[22]M. A. Garcia, D. Puig: “Supervised Texture Classification by Integration of Multiple Texture Methods and Evaluation Windows”, *Image and Vision Computing* 25 (2007), pp. 1091–1106.

[23]D.Puig, M. A. Garcia, J. Melendez: “Application-Independent Feature Selection for Texture Classification”, *Pattern Recognition* 43 (2010), pp. 3282–3297.

[24]D. Puig, M. A. Garcia, J. Melendez: “Multi-Level Pixel-Based Texture Classification through Efficient Prototype Selection via Normalized Cut”, *Pattern Recognition* 43 (2010), pp. 4113–4123.

[25]Q. Chen, J. Luo, P. A. Heng, D. Xia: "Fast and Active Texture Segmentation Based on Orientation and Local Variance", *Journal of Visual Communication & Image Representation* 18 (2007), pp. 119–129.

[26]M. Ekinci, F. W. J. Gibs, B/ T. Thomas: „Knowledge-based Navigation for Autonomous Road Vehicles", *Turkish Journal of Electrical Engineering*, Vol. 8, N^o 1, 2000.

[27]I. Tang, T. P. Breckon: "Automatic Road Environment Classification", *IEEE Transactions On Intelligent Transportation Systems*, Vol 12., Issue 2., pp. 476 – 484, June 2011.

[28] Sebastian Montabone, *Beginning Digital Image Processing,Using Free Tools for Photographers, 2010.*

[29] C. Palm, *Color Texture Classification by Integrative Co-Occurrence Matrices*, Pattern Recognition 37 (2004) 965 – 976.

[30] Mahshid Roumi, MSc THESIS,Implementing Texture Feature Extraction Algorithms on FPGA, Delft University of Technology , CE-MS-2009-25.

[31] M. Mirmehdi, X. Xie, J. Suri. 'Hand book of texture analysis '.2008.

[32] A. Materka, M. Strzelecki, *Texture Analysis Methods – A Review*, Technical. University of Lodz, Institute of Electronics, COST B11 report, Brussels 1998.

[33] Hui Zhang*, Jason E. Fritts, Sally A. Goldman.*' A Fast Texture Feature Extraction Method for Region-based Image Segmentation'*. Dept. of Computer Science and Engineering, Washington University, One Brookings Drive. St. Louis, MO, USA, 63130.

[34] Bongkyu Lee,*' A New Method for Classification of Structural Textures'*, International Journal of Control, Automation, and Systems Vol. 2, No. 1, March 2004.

[35] R. M. Haralick, K. Shanmugam, and I. Dinstein. Texture features for image classification. *IEEE Trans. Systems, Mans and Cybernetics*, SMC-3:610.621, 1973.

[36] E. R. Davies, *Machine Vision – Theory, Algorithms, Practicalities*, Third Edition, Elsevier, 2005.

[37] Guy Smith, Ian Burns, Measuring texture classification algorithms, *School of Information Technology, The UniÍersity of Queensland, Brisbane, QLD 4072.*

Biography of the Author:

Ahmed M. Goma was born on 20 September, 1965 in K. Khair, Libya. He completed elementary school in Aluse, and high school in Tripoli. Faculty of Nuclear and electronic Engineering, University of Tripoli. Enrolled in June,1989. 01.09.1989. employed as electronic engineer in Instrumentation workshop, Raslanauf of Oil Company. 01.01.1991. He moved to Al-Fattah Research center, Tripoli as researcher(1990-1995).

He graduated in 30 September 1997, from Department of Electronics, Faculty of Electrical Engineering, Belgrade University. Master's thesis titled " Microcontroller based Servo-Loop for Canards System".

01.11.1997. to 30.09.2004 employed as a researcher in Al-Fattah Research center Tripoli, Libya.

He move to Faculty of Electrical Engineering, Al-Mergab University, Al-Khoms, Libya as university member ship from (01.10.2004 to 31-12-2008)

In the year 2009 he applied as Ph.D candidate in Faculty of Electrical Engineering, Belgrade University, Signals and systems department, in the field of Digital image processing .

As co-author, has had one scientific paper published in **international journal of advanced robotics systems, December 2012:**

1. S. Graovac**, A. Goma:** " (Detection of the road image borders based on the texture classification) " , Vol. 9, 242:2012, ISSN: 1729-8806, DOI: 10.5772/54359, Decembar 2012.

He was a participant in the 3 international scientific conferences first author and co-author.

1. **A. Goma,** S. Graovac**:** "The extraction of road borders based on texture classification", 56 konferencija ETRAN, Zlatibor, Juni 2012.
2. S. Graovac**, A. Goma**: "One Approach to Road Border Segmentation Based on Texture Descriptors", Turkish Automatic Control Conference, TOK 2011, Izmir, Turkey, Septembar 2011.
3. S. Graovac**, A. Juma**, M. Branković, A. Ristivojević: "Izdvajanje regiona puta na bazi obeležja teksture", 55 konferencija ETRAN, Teslić, Republika Srpska, Juni, 2011.

Изјава о ауторству

Потписани: Ахмед Мохамед Џума

Изјављујем

да је докторска дисертација под насловом:

„The algorithm of a road region extraction from digital image based on usage of texture descriptors“

у преводу:

„Алгоритам издвајање региона пута из дигиталне ТВ слике коришћењем дескриптора текстуре“

- резултат сопственог истраживачког рада,
- да предложена дисертација у целини и у деловима није била предложеназа добијање било које дипломе према студијским програмима других високошколских установа,
- да су резултати коректно наведени и
- да нисам кршио ауторска права и користио интелектуалну својину других лица

У Београду, 19. септембра 2013. г.

Потпис докторанда

____

Изјава о истоветности штампане и електронске верзије докторског рада

Име и презиме аутора: Ахмед Мохамед Џума

Наслов рада: „The algorithm of a road region extraction from digital image based on usage of texture descriptors"

у преводу: „Алгоритам издвајање региона пута из дигиталне ТВ слике коришћењем дескриптора текстуре"

Ментор: Проф. др Миодраг Поповић

Потписани: Ахмед Мохамед Џума

изјављујем да је штампана верзија мог докторског рада истоветна електронској верзији коју сам предао за објављивање на порталу **Дигиталног репозиторијума Универзитета у Београду.**

Дозвољавам да се објаве моји лични подаци везани за добијање академског звања доктор наука, као што су име и презиме, година и место рођења и датум одбране рада.

Ови лични подаци могу се објавити на мрежним страницама дигиталне библиотеке, у електронском каталогу и у публикацијама Универзитета у Београду.

У Београду, 19. септембра 2013. г.

Потпис докторанда

Изјава о коришћењу

Овлашћујем Универзитетску библиотеку „Светозар Марковић“ да у Дигитални репозиторијум Универзитета у Београду унесе моју докторску дисертацију под насловом:

„The algorithm of a road region extraction from digital image based on usage of texture descriptors“

у преводу:

„Алгоритам издвајање региона пута из дигиталне ТВ слике коришћењем дескриптора текстуре“

која је моје ауторско дело.

Дисертацију са свим прилозима предао сам у електронском формату погодном за трајно архивирање.

Моју докторску дисертацију похрањену у Дигитални репозиторијум Универзитета у Београду могу да користе сви који поштују одредбе садржане у одабраном типу лиценце Креативне заједнице (Creative Commons) за коју сам се одлучио

1. Ауторство
2. Ауторство – некомерцијално
3. Ауторство – некомерцијално – без прераде
4. Ауторство – некомерцијално – делити под истим условима
5. Ауторство – без прераде
6. Ауторство – делити под истим условима

У Београду, 19. септембра 2013. г.

Аутор:

Ахмед Мохамед Џума