

**Real Time Video Segmentation For Recognising Paint Marks On Bad
Wooden Railway Sleepers**

ASIF UR RAHMAN SHAIK

Master Thesis

2008

Nr: E3653D



DEGREE PROJECT
COMPUTER ENGINEERING

Program Master of Science in Computer Engineering	Reg. Number E3653D	Extent 30 ECTS
Name of Student ASIF UR RAHMAN SHAIK	YEAR-MONTH-DATE 2008-12-17	
Supervisor SIRIL YELLA	Examiner MARK DOUGHERTY	
Company / Department RAILDOC	Supervisor at Company/Department JOHAN BLOMKVIST	
Title Real Time Video Segmentation for Recognising Paint Marks on Bad Wooden Railway Sleepers		
Keywords Condition Monitoring, Intelligent Vehicle, Videos, Color Segmentation, Spots(objects), Regions		

ABSTRACT

Wooden railway sleeper inspections in Sweden are currently performed manually by a human operator; such inspections are based on visual analysis. Machine vision based approach has been done to emulate the visual abilities of human operator to enable automation of the process. Through this process bad sleepers are identified, and a spot is marked on it with specific color (blue in the current case) on the rail so that the maintenance operators are able to identify the spot and replace the sleeper. The motive of this thesis is to help the operators to identify those sleepers which are marked by color (spots), using an "Intelligent Vehicle" which is capable of running on the track. Capturing video while running on the track and segmenting the object of interest (spot) through this vehicle; we can automate this work and minimize the human intuitions.

The video acquisition process depends on camera position and source light to obtain fine brightness in acquisition, we have tested 4 different types of combinations (camera position and source light) here to record the video and test the validity of proposed method. A sequence of real time rail frames are extracted from these videos and further processing (depending upon the data acquisition process) is done to identify the spots. After identification of spot each frame is divided in to 9 regions to know the particular region where the spot lies to avoid overlapping with noise, and so on. The proposed method will generate the information regarding in which region the spot lies, based on nine regions in each frame. From the generated results we have made some classification regarding data collection techniques, efficiency, time and speed.

In this report, extensive experiments using image sequences from particular camera are reported and the experiments were done using intelligent vehicle as well as test vehicle and the results shows that we have achieved 95% success in identifying the spots when we use video as it is, in other method were we can skip some frames in pre-processing to increase the speed of video but the segmentation results we reduced to 85% and the time was very less compared to previous one. This shows the validity of proposed method in identification of spots lying on wooden railway sleepers where we can compromise between time and efficiency to get the desired result.

Acknowledgements

So finally I've done it. I've managed to write my master thesis. But, of course, for being able to arrive to this point, I have to thank lot of people.

First, I would like to deeply thank Dr. Siril Yella, my supervisor of this research work, for his guidance, supervision and help throughout the work. Without his motivation and continuous encouragement the work wouldn't have been completed. He helped me lot in methodology development, data collection procedure etc. He also spends his valuable time leading this research work to success.

Second, I would like to thank all the people working in RailDoc, for their constant support and help through out the work, starting from data acquisition procedure to generation of results.

Third, I would like to thank Dr.Hasan Fleyeh. During my studies he taught me knowledge related to Computer Graphics and Image Processing, enabling me to stand on a solid background for this research work.

I would also like to thank Professor Mark Dougherty, Dr .Pascal Rebreyend and others for reviewing report and kindly serving on the committee.

Last but not the least; I would like to thank all of my friends for helping me to complete my work. Finally, I would like to thank the Swedish Government for giving me an opportunity to study in Sweden and Department of Högskolan Dalarna for providing all the resources to complete my work.

 Table of Contents

1. Introduction	09
1.1 Problem Description.....	10
1.2 Proposed Solution	12
1.3 Previous Work.....	13
2. Background.....	15
2.1 Human Vision.....	15
2.2 Computer Vision.....	15
2.2.1 Digital Image.....	17
2.2.2 Image Data.....	18
2.2.3 Image Acquisition.....	19
2.2.4 Image Enhancement	19
2.2.5 Image Restoration.....	19
2.2.6 Color Image Processing.....	20
2.2.7 Segmentation.....	20
2.2.8 Representation.....	21
2.3 Color Information.....	22
2.3.1 Colrimetry.....	22
2.3.2 Color.....	22
2.3.3 Hue.....	23
2.3.4 Saturation	23
2.3.5 Value.....	23
2.4 Color Space.....	24
2.4.1 RGB Color Space.....	25
2.4.2 HSV Color Space.....	26
3. Algorithm and Design	27
3.1 Potential Difficulties.....	27
3.2 Proposed Methodology.....	29
3.3 Steps in Methodology	30
3.3.1 Video Acquisition.....	31
3.3.1.1 TestData1	33

3.3.1.2 TestData2.....	34
3.3.1.3 TestData3.....	34
3.3.1.4 TestData4.....	36
3.3.2 Pre processing.....	36
3.3.2.1 Gaussian Blur.....	37
3.3.2.2 RGB2HSV.....	37
3.3.3 Segmentation.....	39
3.3.4 Object Separation	40
Reflections.....	40
Vegetation.....	41
Sun Light.....	41
Noise Reduction Technique.....	42
3.3.4 Object representation.....	43
Circle Drawing.....	44
Regions.....	44
3.3.5 Result Generation	45
4. Experiments and Analysis.....	45
4.1 Introduction.....	45
4.2 Test Data 1.....	46
4.3 Test Data 2.....	49
4.4 Test Data 3.....	52
4.5 Test Data 4.....	55
Discussion.....	58
Conclusion and Future Work.....	59
Appendix.....	60
6.1 Appendix A.....	60
6.2 Appendix B.....	60
7. References.....	61

List of Figures

Figure 1 Intelligent Vehicle.....	11
Figure 2 Test Vehicle.....	11
Figure 3 RGB Cube.....	25
Figure 4 Hexagon for HSV view.....	26
Figure 5 Vegetation Problem.....	27
Figure 6 Sample Images of Potential Difficulties.....	28
Figure 7 Structural Design of Methodology.....	30
Figure 8 Sample Images of TestData1.....	33
Figure 9 Camera Position in this Procedure.....	34
Figure 10 Sample Images of TestData2.....	34
Figure 11 Sample Images of TestData3.....	35
Figure 12 Camera position in this procedure.....	36
Figure 13 Sample Images of TestData4.....	36
Figure 14 Reflection Noise in Segmentation	40
Figure 15.1 Noise Created by Sun light.....	42
Figure 15.2 Spot Eliminated by Sun light.....	42
Figure 16 Noise Created after Segmentation.....	43
Figure 18 Representations of Different Data.....	45
Figure 19 Segmented Images of TestData1.....	48
Figure 20 Generated Results of TestData1.....	48
Figure 21 Generated Results of TestData1 after acceleration	48
Figure 22 Segmented Images of TestData2.....	51
Figure 23 Generated Results of TestData2.....	51
Figure 24 Generated Results of TestData2 after acceleration.....	51
Figure 25 Segmented Images of TestData3.....	54
Figure 26 Generated Results of TestData3.....	54
Figure 27 Generated Results of TestData3 after acceleration.....	54
Figure 28 Segmented Images of TestData4.....	56
Figure 29 Generated Results of TestData4.....	56

Figure 30 Generated Results of TestData4 after acceleration.....56

List of Tables

Table 1 Methodology Description	31
Table 2 Results of TestData1.....	46
Table 3 Results of TestData2.....	49
Table 4 Results of TestData3.....	52
Table 5 Results of TestData4.....	55

1. INTRODUCTION

Condition monitoring is the process of monitoring a parameter of condition in any domain, such that a significant change is indicative of a developing failure. It is a major component of predictive maintenance. Condition monitoring application in the transportation domain have great importance in ensuring safe operations, numerous different condition monitoring solutions are available in the railway sector to assess maintenance needs and to support maintenance decision-making. The use of condition monitoring allows maintenance to be scheduled, or other actions to be taken to avoid the consequence of failure, before the failure occurs. In transportation domain a failure can lead to accidents and other relevant incidents which happen every year in the world and cause serious destruction of property and injury or death of passengers and crew members. To redress the situation, condition monitoring routines are conducted periodically to ensure proper condition of the structure or the material that is being inspected. It is much more cost effective than allowing it to fail.

The transportation domain in railways uses several kinds of sleepers such as wood, concrete and steel. Wood and concrete are the most common types; however, steel and plastic are also being used. Many sections of railways in Sweden are constructed with wooden sleepers which have been in position for many years and require regular inspection to monitor their condition. Determining the most economic course of maintenance action is difficult. One course of action is to replace those sleepers in very poor condition with new wooden sleepers. This will give some more years of life to the railway, but re-inspection and further replacements will usually be needed before long. Another course of action is to completely replace all of the sleepers on a section, possibly with concrete sleepers. This is obviously a much bigger capital investment. The Inlandsbanan (Inland railway in Sweden) is a railway which has this problem. It was constructed many years ago and uses wooden sleepers throughout. Since it only carries a fairly low level of traffic, keeping maintenance costs to a minimum is essential for the railway to be viable. With the above-mentioned problems in view, it is quite obvious that an efficient and accurate way of measuring the condition of wooden railway sleepers would be advantageous [1]. Wooden railway sleeper inspection in Sweden are generally carried out manually, a railway worker who is expert of this work walks along the railway track visually

examining each sleeper. Such a process of manually inspecting each sleeper is slow and time consuming and it requires skilled persons of that field to know the condition of sleepers. As the manual inspection is done the quality standard will differ from person to person because of the human error. To avoid those errors condition monitoring solutions provide opportunities to gain control of failure modes that may be difficult to identify by manual visual inspections. The condition monitoring techniques are usually deployed using NDT (Non Destructive Testing) methodologies. NDT methods and its procedures concerned with examining all aspects of uniformity, quality and serviceability of material and structures, without causing damage to the material that is being inspected. The automation of condition monitoring work has been proposed in [1]. Where we can avoid the human intuition by the above methods and the automation can be achieved by mounting human behavior on an “intelligent vehicle” which is capable of running on the track. With this we can greatly minimize the problem arising from issues such as human intuition. It has been done by NDT methods like Machine Vision and Impact acoustic signals where machine vision proves to be the better than the other counterpart. These methods have to be work quick and accurate for long periods so that it can be used in future. With these methods bad sleepers are classified and spot is marked with a specific color (blue in this case) so that later we re-identify that spot and further replacement work can be done by maintenance operators if necessary.

1.1 Problem Description:

Through the methods described in [1] we can identify the bad sleepers, but the condition monitoring of railway sleepers and its replacement work will not be implemented at same time. Because the maintenance operators have to produce a report, containing details of each sleeper and its area. The operators have to give every details of each sleeper and present condition of it, can it stand long or beside sleeper can support this one etc. They send this report of higher authority and members responsible for maintenance will decide to proceed further with replacement or wait for another inspection. To proceed further with replacement or another inspection we have to identify that particular sleeper which is already being inspected. To re-identify those sleepers we will mark a spot on it, when we do first inspection of condition monitoring. So that later we can check only the sleepers which are marked by

color or any other material, maintenance operators will proceed with their work what ever is ordered.

The spot is marked, but the problem is same as in [1]. To mark the spot and identify it manually it takes more time as railway track consists thousands of sleepers, and it takes lot of time if it is done manually. To avoid this we have to automate this work as well, as done in condition monitoring [1]. In this thesis the current work aims the automation of manual re-inspection procedure and it has to be done on intelligent vehicle (Figure 1). We have tested different procedure to automate this work on “intelligent vehicle” and “Test vehicle” as well.(Figure 2) With these vehicles we have tested few videos to automate the manual sleeper re-inspection procedure with aim of achieving more reliable and robust results with increased speed and accuracy.



Figure 1: Intelligent Vehicle



Figure: 2 Test Vehicle (Full and Close View)

1.3 Proposed Solution

The proposed solution here is developed a prototype which can identify all the spots that are marked by a specific color, and generate the results of each concerning spot and its region where it is lying. Computer vision algorithms had to assess the running video and segment the object of that particular color individually, and it has to be capable of discriminating between noise and other undesired material, mainly composed of vegetation and reflections. The aim of image segmentation is to divide an image into parts that have a strong correlation with objects or areas of the real world contained in the image [12]. Segmentation consists in determining which regions of the image correspond to background and which represent the objects of interest. We opted for a pixel-oriented segmentation algorithm, because these algorithms are normally faster than other approaches (region-oriented algorithms, etc.). The objects here are blue in color and they are fixed to a background, if the background is moving automatically the object is motion, we just highlight those pixels which are blue from the background. With this we can see only the required object from the whole image. Once the background had been removed, each connected region was labeled as a possible object of interest (under normal circumstances, it should be a spot or some other material). In the same operation, the program estimated the spot and a circle is drawn around it to make sure that, it is a spot not the noise. The unwanted material such as noise is removed by preprocessing of each frame. Finally, the objects (spots) are segmented with out its background on the new screen. It is done continuously for each frame of a video in real time to display the spot and generate the results in which region it lays. Real-time image processing is essential to achieve the required throughput. Intuitive and fast segmentation techniques had to be compared in order to decide which one should be implemented in the real-time. The optimal solution had to be the one that consumed less computing time and was easier to operate by a non-experienced user, without compromising quality standards. Moreover, the solution had to be economically feasible and easy to maintain. On-line video acquisition and its analysis software were programmed in visual C++ language. All source codes were specifically written for this application with the help of free source commercial library know as OpenCV, in order to ensure the control of the operations and real-time responses.

1.3 Previous Work

Previous work [4] demonstrates efficient use of intelligent vehicle. Due to the huge amount of literature available, we will focus on the most promising approaches. Road sign recognition approaches are used to improve safety and efficiency on the road, a computer vision system embedded in a car is and capable of identifying and locating in real time road signs is certainly most challenging one. The aim of Road Sign Recognition is to provide drivers with the ability to understand its neighborhood environment and permit advanced driver support such as collision prediction and avoidance which can be seen in [5 6]. Such road sign computer vision system could be divided into three main modules: detection, identification, and location. But here we will focus on detection because it can rely on color or shape to extract the sign from image. This can be done on an intelligent vehicle.

Moving object segmentation in videos is a key process in a plethora of multimedia and computer vision applications such as content-based retrieval, 3D scene reconstruction, video surveillance, and so on are given in [7]. Most of the difficulties associated to this task are due to the presence of a relative motion of the observer with respect to the motion of objects and background. When the background is stationary and videos are acquired by fixed cameras, moving object detection is a relatively “easy” problem: well-known methods such as background suppression and frame differencing were implemented with success in many applications. Instead, the segmentation of moving objects becomes more critical when the video is acquired by a moving camera with an unconstrained and a priori unknown motion. More in general, we consider the framework of videos with moving foreground objects on a moving background. In this case, segmentation cannot be accomplished computing visual motion only, but other features must be exploited such as color, shape, texture, and so on. In addition, our aim is to develop a system able to compute the dominant motion in real time, since the final goal is to provide a fast segmentation that could be adopted in an on-line video segmentation process. Many works propose the integration of multiple features for detecting and tracking moving objects on video with a moving background. Among them, a previous work of Gelgon and Bouthemy [8] proposes an approach based on a color segmentation followed by a motion-based region merging. Isaac and Gerard [9] proposes an approach based

on graph representation of moving objects which allows to derive and maintain a dynamic template of each moving object by enforcing their temporal coherence. Yokoyama and Poggio [10] propose a fast and robust approach to the detection and tracking of moving objects. Their method is based on using lines computed by a gradient-based optical flow and an edge detector. The extracted edges by using optical flow and the edge detector are restored as lines, and background lines of the previous frame are subtracted. Contours of objects are obtained by using snakes to clustered lines.

Segmentation attempts to identify homogeneous regions within an image and classify them as segments. Numerous intra-frame image segmentations techniques have been developed over the past two decades to segment an image's content. Color segmentation process is part of image preprocessing module, where the accuracy of this process reflects the success of the whole vision system. Inaccurate segmentation may result in inaccurate recognition of the objects [11]. To develop a robust intelligent vehicle vision system for object recognition in railway transportation, algorithms must be developed to extract useful information in the presence of noise associated with unstructured lighting conditions. Color information has more certainty power than intensity values. Therefore, obtaining image features using color information has some specific robustness properties. Since real-time requirement is of important here, colors that are known a priori are chosen to ease this task. This kind of color segmentation is known as supervised color segmentation [11].

A fundamental requirement of reliable vision systems is the ability to extract from digital images visual cues relevant to the imaged scene. To handle the big amount of data contained in video information, segmentation as well as classification/recognition and parameterization are some important steps which produce a data representation in structural form. Most of earlier works in the field of data classification involved techniques mainly based on Markov Random Fields (MRF-based energy minimization), multi resolution scheme, statistics or Genetic Algorithms (GA) which are not yet compatible with a real-time video control [12]. For applications involving robots, image segmentation as well as classification and recognition must also be fully automated and when one has to deal with color images, it is

suitable to take advantage of the multispectral contents of the video information. With color images, a pixel is a mixture of the three fundamental colors: red, green and blue (RGB) and there exist many representations to encode this information. The RGB space maybe the most well-known color representation since it is useful for data storage. However, some color image processing such as enhancement and restoration require that only the luminance component (as the amount of visible light) to be processed whereas some other applications require color (hue and saturation) components to be preserved or modified [12].

The rest of this thesis is organized as follows. Section 1, a brief introduction to the wood inspection procedure and techniques used to classify it. Section 2, describes the techniques used in our approach. Section 3, describes the algorithm design. Section 4, describes the experiment and analysis. Section 5, presents conclusion and future work.

2. Background

2.1 Human vision

The basis of learning in humans is the sensory of touch, smell, vision, hearing and taste. Out of these, vision and hearing are considered to be a complex process [15]. From the beginning of time humans have tried to explain the complexity process of vision. Images and colors are constantly updated as you turn your head and redirect your attention [16]. The seamless quality in the images that we see is possible because human vision updates image, including the details of motion and color, on a time scale so rapid that break on the action is almost never perceived. The range of color, the perception of seamless motion, the contrast and the quality, along with the minute details, that most people can perceive make real-life images clearer and more detailed than any thing seen on a television or computer screen. The efficiency and completeness of our eyes and brain is unparalleled in comparison with any piece of apparatus or instrumentation ever invented [15].

2.2. Computer Vision:

Computer vision has evolved greatly during the last decade, being able to control visual process in a lot of factories, improve the security systems in a lot of facilities, and even enter the video gaming market. We have a lot to thank to hardware developers, since it is the

increase in hardware performance and memory availability that has opened the doors for a lot of complex algorithms [2]. Tracking non-rigid motion from image sequences has been of great interest to the computer vision community. One of the important reasons is because it is very difficult. The problem here is object is still and the camera is moving but the data acquisition process is like camera is still and the object is moving. The detection of moving object in the video sequence is the first relevant step in the extraction of information in many computer vision applications including, for example, vehicle classification, vehicle counting, traffic monitoring and people tracking [3]. The quality of the results obtained by applying this stage is very important. The more reliable the shape and the position on the moving object, the more reliable their identification is. The crucial issue related to automatic video segmentation are to separate the moving object from the background [3]. So we can see only the desired object with out any background in the resulting data.

One of the monolithic goals of computer vision is to automatically interpret general digital images of arbitrary scenes. This goal has produced a vast array of research over the last 35 years, yet a solution to this general problem still remains out of reach. A reason for this is that the problem of visual perception is typically under-constrained. Information like absolute scale and depth is lost when the scene is projected onto an image plane. In fact, there are an infinite number of scenes that can produce the exact same image, which makes direct computation of scene geometry from a single image impossible. The difficulty of this "traditional goal" of computer vision has caused the field to focus on smaller, more constrained pieces of the problem. The hope is that when the pieces are put back together, a successful scene interpreter will have been created [17]. The analysis of image content and conversion into meaningful description is termed as computer vision. In other words, computer vision is a branch of artificial intelligence and image processing concerned with computer processing of images from real world. Computer vision research also depends on the techniques from wide range of other fields such as computer graphics and human computer interaction (HCI). Computer vision uses statistical methods to disentangle data using models constructed with the aid of geometry, physics and learning theory [18].

Employing computer vision technology into smart design is a call for consideration. Firstly, vision subsystem incorporated into the driver support system may exploit all the information processed by human drivers with out any requirements for new traffic infrastructure devices (a very hard expansive task). Smart cars equipped with vision based systems will be able to adapt themselves to operate in different countries (with often quite dissimilar traffic devices). As the integration of various technologies in the field of traffic engineering has been introduce (ITS) the convenience of computer vision usage has become more obvious. More than 50% of papers are focused on Image Processing and Computer Vision methods [6].

Obviously, there exist even disadvantages of the vision-based approach. Smart vehicles will operate in real time traffic conditions on the road. So, the algorithms must be robust enough to give good results even under adverse illumination and weather conditions. Although this system property may seem to be solved easily it is the real challenge for the algorithm developers. There cannot be assured absolute system reliability and the system will not be "fail-safe" because of the definition of individual transportation system. The aim is to provide a level of safety similar to or higher than that of human drivers. From experiments follows, that 60 percent of crashes at intersections and 30 percent of head-on collisions could be avoided if the driver had an additional half-second to react. About 75 percent of vehicular crashes are caused by inattentive drivers [5].

Computer Vision is the science and technology of machines that sees. As a Scientific discipline, computer vision concerned with the theory of building artificial systems that obtain information from images [14]. The image data can take many forms such as.

2.2.1. Digital Image

A Digital Image is described as a $[m, n]$ in a 2D discrete space, and is derived from an analog image $a(x, y)$ in a 2D continuous space through a sampling process that is frequently referred to as digitization. For now we will look at some basic definitions associated with the digital image. Image Processing can be defined as the "act of examining images for the purpose of identifying objects and judging their significance" Image analysis study the remotely sensed data and attempt through logical process in detecting, identifying, classifying,

measuring and evaluating the significance of physical and cultural objects, their patterns and spatial relationship. Digital Image Processing has become increasingly important in many areas, such as remote sensing, robotics, graphic printing, digital telecommunication and medical imaging. Images are often deteriorated by noise due to various sources of interference and other phenomena that affect the measurement process in imaging and data acquisition system. Proper image processing can improve image contrast, reduce noise, sharpen edges, remove artifacts, and recognize image patterns [19]. In order to convert image into meaningful description in image itself, image processing carries these fundamental steps.

2.2.2 Image Data:

Image data is, conceptually, a three-dimensional array of pixels, as shown in figure. Each of the three arrays in the example is called a *band*. The number of rows specifies the image height of a band, and the number of columns specifies the image width of a band. Monochrome images, such as a grayscale image, have only one band. Color images have three or more bands, although a band does not necessarily have to represent color. For example, satellite images of the earth may be acquired in several different spectral bands, such as red, green, blue, and infrared.

In a color image, each band stores the red, green, and blue (RGB) components of an additive image, or the cyan, magenta, and yellow (CMY) components of a three-color subtractive image, or the cyan, magenta, yellow, and black (CMYK) components of a four-color subtractive image. Each pixel of an image is composed of a set of *samples*. For an RGB pixel, there are three samples; one each for red, green, and blue.

An image is sampled into a rectangular array of pixels. Each pixel has an (x,y) coordinate that corresponds to its location within the image. The x coordinate is the pixel's horizontal location; the y coordinate is the pixel's vertical location. Within programming, the pixel at location $(0,0)$ is in the upper left corner of the image, with the x coordinates increasing in value to the right and y coordinates increasing in value downward. Sometimes the x coordinate is referred to as the pixel number and the y coordinate as the line number.

2.2.3 Image acquisition:

The first stage of any vision system is the image acquisition stage. After the image has been obtained, various methods of processing can be applied to the image to perform the many different vision tasks required today. However, if the image has not been acquired satisfactorily then the intended tasks may not be achievable, even with the aid of some form of image enhancement.

2.2.4 Image enhancement:

The aim of image enhancement is to improve the interpretability or perception of information in images for human viewers, or to provide 'better' input for other automated image processing techniques. Or we can say Image enhancement is among the simplest and most appealing areas of digital image processing. Basically, the idea behind enhancement technique is to bring out detail that is obscured, or simply to highlight certain features of interest in an image.

Image enhancement techniques can be divided into two broad categories:

1. Spatial domain methods, which operate directly on pixels, and
2. Frequency domain methods, which operate on the Fourier transform of an image.

Unfortunately, there is no general theory for determining what 'good' image enhancement is when it comes to human perception. If it looks good, it is good! However, when image enhancement techniques are used as pre-processing tools for other image processing techniques, then quantitative measures can determine which techniques are most appropriate.

2.2.5 Image restoration:

Is an area that also deals with improving the appearance of an image? The purpose of image restoration is to "compensate for" or "undo" defects which degrade an image. The image restoration is objective, in the sense that restoration techniques tend to be based on mathematical or probabilistic models of image degradation. Degradation comes in many forms such as motion blur, noise, and camera miss-focus. In cases like motion blur, it is possible to come up with a very good estimate of the actual blurring function and "undo" the

blur to restore the original image. In cases where the image is corrupted by noise, the best we may hope to do is to compensate for the degradation it caused.

2.2.6 Color Image Processing

Is an area that has been gaining importance because of the significant increase in the use of digital images over the Internet? Humans have always seen the world in color but only recently have we been able to generate vast quantities of color images with such ease. In the last three decades, we have seen a rapid and enormous transition from grayscale images to color ones [20]. Today, we are exposed to color images on a daily basis in print, photographs, television, computer displays, and cinema movies, where color now plays a vital role in the advertising and dissemination of information throughout the world. Color monitors, printers, and copiers now dominate the office and home environments, with color becoming increasingly cheaper and easier to generate and reproduce. Color demands have soared in the marketplace and are projected to do so for years to come. With this rapid progression, color and multispectral properties of images are becoming increasingly crucial to the field of image processing, often extending and/or replacing previously known grayscale techniques. We have seen the birth of color algorithms that range from direct extensions of grayscale ones, where images are treated as three monochrome separations, to more sophisticated approaches that exploit the correlations among the color bands, yielding more accurate results. Hence, it is becoming increasingly necessary for the image processing community to understand the fundamental differences between color and grayscale imaging. There are more than a few extensions of concepts and perceptions that must be understood in order to produce successful research and products in the color world

2.2.7 Segmentation:

Procedures partition an image into its constituent parts or objects; 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 problem that requires objects to be identified individually. In general, the more accurate the segmentation, the more likely recognition is to succeed. Segmentation refers to identifying

groups or regions of connected pixels with similar properties. These regions are important for the interpretation of an image because they may correspond to object in a scene. An image could contain several objects and each object may have several regions corresponding to different parts of the object. For an image to be interpreted accurately, it must be portioned into regions that corresponds to objects or parts of and object [16]. Image segmentation is a must when dealing with digital image especially for object recognition. This process is a crucial step to prepare the image for further processing [11]. Image segmentation is the process of isolating objects of interest from the rest of the scene. Or we can say that image segmentation is the process of partitioning an image into non-intersecting regions such as that each region is homogeneous and the union of two non-adjacent regions is homogeneous. The level to which the subdivision is carried out depends on the problem being solved. That is, segmentation should stop when the object of interest in an application have been isolated.

Segmentation of non-trivial images is one of the most difficult tasks in image processing. It is difficult for many reasons which are as follows.

1. Non-uniform illumination.
2. No control of the environment.
3. Inadequate model of the object of interest.
4. Presence of noise.

Since most color used for segmentation in transportation domain are primary colors (red, green and blue), the source code here will work for all the three colors but the main focus will be on the blue in documentation. Because we use this color in our data acquisition process so we work on this color. The main constraints is real-time segmentation of this color, it sounds normally to use the RGB color space to represent the data because this information is directly supplied by the camera with out any transformation [21].

2.2.8 Representation: The representation of image will always follows the output of a segmentation stage, which usually is raw pixel data, constituting either the bound array of the region (set of pixels separation one image region from another) or all the points in the region itself.

From the above fundamental steps some methods are important where input and output varies depending upon the requirement. Some of these methods like Image Enhancement, Image restoration, Color Image Processing are methods whose input and output are Images. Where as Segmentation, Representation and Object Recognition comes under other category where input are images but output are attributes extracted from those images.

2.3 Color Information

The term color is used with different meanings in different technologies. To lamp engineers, color refers to a property of light sources. To graphics art engineers, color is a property of an object's surface (under a given illumination). In each case, color must be physically measured in order to record it and reproduce the same color [22]. The perception of color is a psychophysical phenomenon, and the measurement of color must be defined in such a way that the results correlate accurately with what the visual sensation of color is to a normal human observer.

2.3.1 Colorimetry

Is the science and technology used to quantify and describe physically the human color perception. The basis for colorimetry was established by CIE (Commission Internationale de l'clairage) in 1931 based on visual experiments. Even though limitations are well recognized, the CIE system of colorimetry remains the only internationally agreed metric for color measurement [22]. All the official color-related international standards and specifications use the CIE System. The CIE system works well in most cases, but one should know the assumptions and limitations in visual conditions where the CIE system is defined [23, 24, 25]

The international Commission of Illumination (CIE) (Commission Internationale de l'Eclairage) defined the following terms:

2.3.2 Color:

Color is the perceptual result of light in the visible region of the spectrum, having the wavelength in the region of 400-700 nm, incident up on retina.

The characteristics generally used to distinguish one color from another are hue, saturation and value.

2.3.3 Hue:

It is the attribute of visual sensation according to which an area appears to be similar to one of the perceived colors, red, yellow, green and blue, or a combination of two of them, we are specifying its hue. Hue is more specifically described by the dominant wave length in models such as the CIE system. It represents dominant color as perceived by an observer. Hue is also a term which describes a dimension of color we readily experience when we look at color. It will be the first of three dimensions we use to describe color.

2.3.4 Saturation:

It refers to the relative purity or the amount of white light mixed with hue. Or it refers to dominance of hue in the color. On the outer edges of the hue wheel are 'pure' hues. Moving towards the center of the wheel, the hues are used to describe decreasing color dominance. The pure spectrum colors are fully saturated. Colors such as pink (red and white) and lavender (violet and white) are less saturated, with degree of saturation being inversely proportional to the amount of white light added.

Hue and Saturation taken together are called chromaticity, and therefore, a color may be characterized by its brightness and chromaticity. The amount of red, green, and blue needed to form any particular color are called tristimulus values.

2.3.5 Value:

It represents the brightness of the color. Or it is a measure where a particular color lies along the lightness and darkness axis. It ranges from 0 to 255, with 0 being completely dark and 255 being fully bright. The dominant description for black and white is the term, value. The hue and saturation level do not make a difference when value is at max or min intensity level.

2.4 Color Space

What is Color Space

A color space is a method by which we can specify, create and visualize color. As humans, we may define a color by its attributes of brightness, hue and colorfulness. A computer will describe a color stimulus in terms of the excitations of red, green and blue phosphors on the CRT faceplate. A printing press describes a color stimulus in terms of the reflectance and absorbance of cyan, magenta, yellow and black inks on the paper. Such a color is usually specified by using three coordinates, or attributes, which represent its position within a specific color space. These coordinates do not tell us what the color looks like, only where the color is located within a particular color space [26].

The proper use and understanding of color spaces is necessary for the development of color image processing methods that are optimal for the human visual system. Many algorithms have been developed that process in an RGB color space without ever defining this space in terms of the CIE color matching functions, or even in terms of the spectral responses of R, G, and B. Such algorithms are nothing more than multichannel image processing techniques applied to a three-band image, since there is no accounting for the perceptual aspect of the problem. To obtain some relationship with the human visual system, many color image processing algorithms operate on data in hue, saturation, lightness (HSL) spaces. Commonly, these spaces are transformations of the aforementioned RGB color space and hence have no visual meaning until a relationship is established back to a CIE color space. To further confuse the issue, there are many variants of these color spaces, including hue saturation value (HSV), hue saturation intensity (HSI), and hue chroma intensity (HCI), some of which have multiple definitions in terms of transforming from RGB. Since color spaces are of such importance and a subject of confusion, we will discuss them in detail. There are two primary aspects of a color space that make it more desirable and attractive for use in color devices: 1) its computational expediency in transforming a given set of data to the specific color space and 2) conformity of distances of color vectors in the space to that observed perceptually by a human subject, i.e., if two colors are far apart in the color space, they look significantly different to an observer with normal color vision. Unfortunately, these two criteria are antagonistic. The color spaces that

are most suited for measuring perceptual differences require complex computation, and vice versa.

2.4.1 The RGB Color Space

This color space is used for computer graphics. It is the best known and most widely used color space, each color in this system is represented by three values referred as RGB. It is built in the form of a cube in the Cartesian coordinate system in which the x, y and z axis are represented by R, G and B respectively. The range of RGB values are [0,1] or [0,255]; which leads to black, which is located at the center of the coordinates, being given as (0,0,0) and white as (1,1,1) for the float images where for byte images (0,0,0) for black and (255,255,255) for white. These two colors represent the opposite corners of the RGB space cube. The RGB color scheme is an additive model. Intensities of the primary colors are added to produce other colors. Each color point within the bounds of the cube can be represented as the triple RGB.

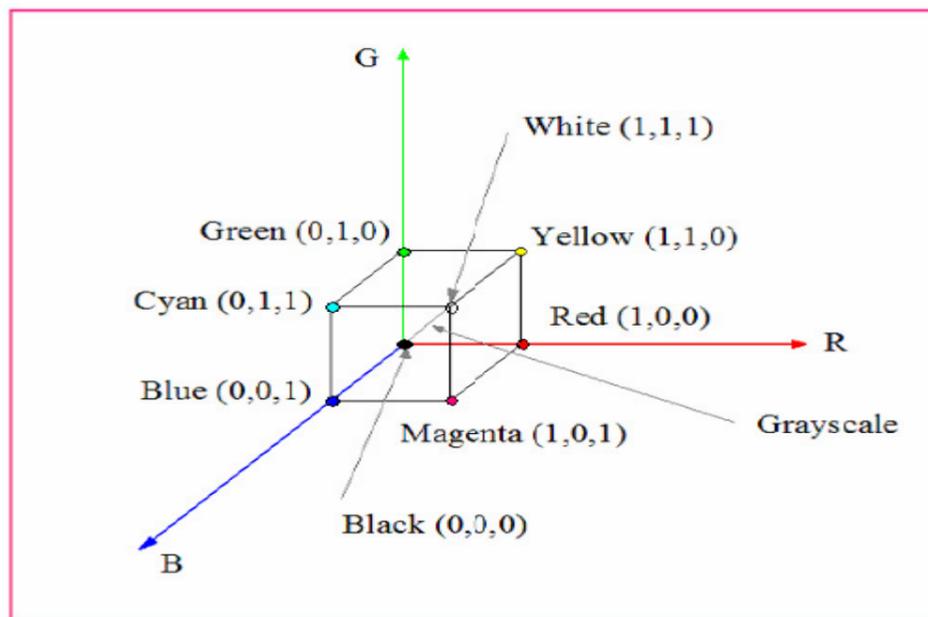


Figure 3: RGB Cube [35]

2.4.2 HSV Color Space

One of the big advantages of using the RGB color space is its difficulty to separate the color information from the brightness one. Instead of using the color primaries, the HSV uses color descriptions which have a more intuitive appeal to users. To give a color specification, a user selects a spectral color and the amounts of white or black which is to be added to obtain different shades, tints, and tones. The 3D representation of the HSV is derived from the RGB cube. If the RGB cube is viewed along the diagonal from the white vertex to the origin (black), the outline of the cube, which has a hexagon shape. The boundary of the hexagon represents the various hues; the saturation is measured along a horizontal axis, the value along a vertical axis through the center of the hexagon.

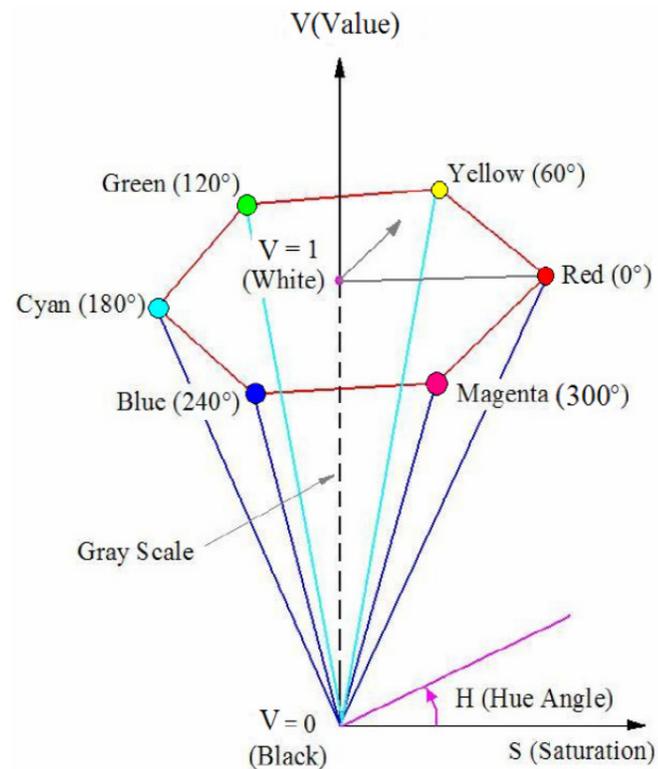


Figure: 4 Hexagon for HSV vales [35]

3. Algorithm Design

3.1 Potential Difficulties

The potential difficulties we come across after starting the work, due to some complex environment in railway transportation domain and nature of the railway track. We started the work with static images and our method was similar to the work done in condition monitoring approach [1] as well as in road side recognition approach [4 5 6]. Working on static images and segmenting those images continuously is a tough task because of its behavior in different environmental conditions and procedure for collecting it. Because the camera is moving at some speed and capturing pictures continuously, there may be chances of getting noise in it such as (blurring, etc). Vegetation and reflection are other type of noise caused by the environment near railway track. There is a problem of light as well, if we have more light possibility of reflections, more intensity etc. These noises can be removed by preprocessing but problem comes with the vegetation, if we unable to see the object what is the use of removing the noise (see Figure 5). There are other difficulties as well with color and position of the spot see (Figure 6). The first proposed technique worked good for static images. In this work the images are taken by the camera continuously while moving on an intelligent vehicle and those images are transferred to the system where we can segment them continuously to know the results of each image. The results can be seen in (figure 6). The position of the spot also creates some problem; the spot on the sleeper will not be seen in winter, or any stone covering it by default. The spot on the rail makes ambiguities if train passes when it is wet. The spot on the side of the rail gives a solid spot segmentation with out any problem except for vegetation; if vegetation is there this spot can't be seen at all.



Figure: 5 Vegetation Problem

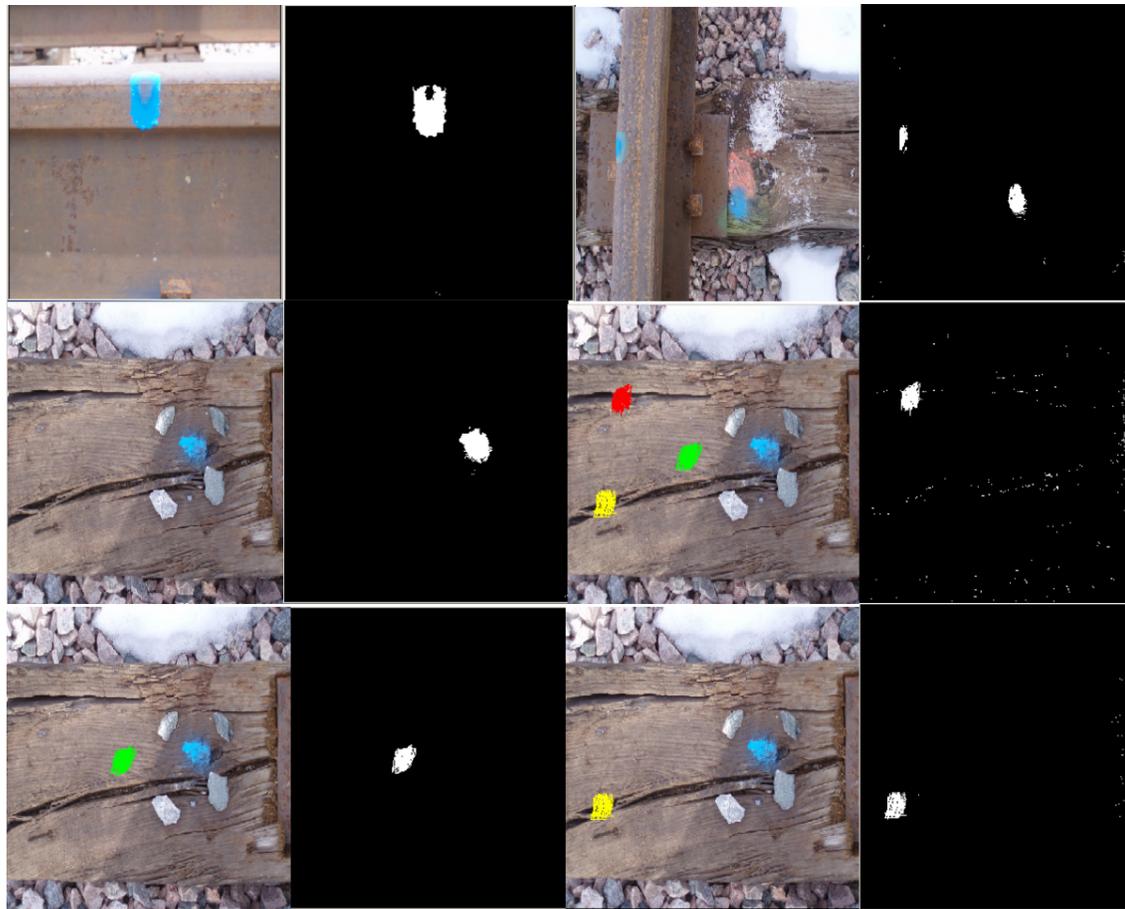


Figure 6: Sample Images of Potential Difficulties

From [L-R](a) Original Image for spot on rail; (b) Binary Image after blue color segmentation; (c) original images taken from top; (d) Binary Image after blue color segmentation; (e) Original Image spot on sleeper with some noise; (f) Binary Image after blue color segmentation; (g) Test sample image for different colors; (h) Binary Image after red color segmentation; (k) Binary Image after green color segmentation; (l) Binary Image after yellow color segmentation;

This method was tested with different positions of the spots such as, on top of the rail, on the sleeper; on side of the rail, to know which will be perfect position for segmentation. As the above results shows that all position will work but in winter the spot on sleeper will not be seen, the spot on the rail makes ambiguities if train passes when it is wet. It has one advantage

it can be seen in any season. The last spot on the side of the rail works well, there is no problem with any of the above but there is noise of vegetation in it. Because all the above three positions of spot we have to use three different angles of camera positions. There may be possibilities of getting noise by the angle of camera it self. If we avoid the vegetation and ambiguity from the above the both position (top and side of rail) will work perfectly. But the main problem here is speed, how much time it will take for segmentation, how much time it will take for capturing an image; we have to take all into consideration because we have to test thousands of sleepers a day so it will take more time if the process is slow. By this method we can remove the noise by pre processing but we can't improve the speed of vehicle. If we improve the speed, the data will be bad and we need more processing and it takes more time and quality will be reduced as well. How can these kind of problems be solved, so we decided to change our method instead of working with still images we decided to go with moving images(i.e. frames) which is video. By moving form still images to video we can avoid the problem of vegetation, seasons around 80% and with improved speed as well. In this method we have tested different colors as well, where we find blue color is perfect for this kind of data. The practical test was not done here, but a few sample tests were done by coloring the object of interest. The red color produce noise because of the rust on the rail, green color will mix up with the vegetation and there will be some segmentation problem, yellow color also produces some noise due to rust on the rail. So we proceed with the blue color in current case.

3.2. Methodology Development

The structural design in this Methodology starts with the video acquisition process, where a raw data is acquired through camera and send to the processing unit. In the figure 7 you can see a structural design of real time object identification process based on series of computer vision technologies. The main objective here is to detect the object and represent it in a better way. In the detection step, acquired image frame are pre-processed, enhanced and segmented according to object properties such as color and shape. Only the regions of interest are extracted from the complex background and represent it in another frame. In the output image which is a binary mask, interested regions are represented with white color and the rest

parts are filled with black color in contrary to represent only the object of interest. The speed and efficiency depends up on time taken in preprocessing since, object detection in this domain is live and straight forward. The time taken to respond the data will be more if it takes more time in pre processing. Meanwhile, we divided the output frame in to nine regions to know particular region of the object detected. In the final step, recognition and generation of results depends upon objected detected and its region where it is lying. The generation of results is done, because it provide information which region object belongs to as well as prior information for analyzing all the objects are detected or we have any noise in it as well. By combing the methods mentioned earlier we can generate reliable and robust results by which we can decide individual objects (spots) and its constraints.

3.3 Steps in Methodology Development

In order to convert series of images into meaningful descriptions, our approach carries out the following fundamentals steps shown in (figure 7).

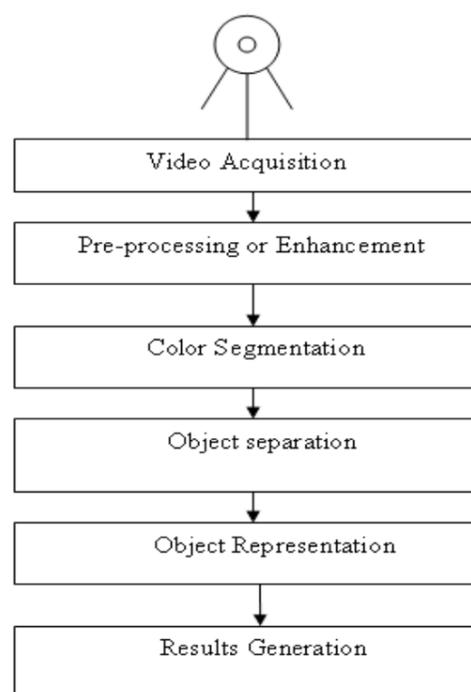


Figure 7 Structure Design of Methodology

Each of the above illustrated steps is an active area of research in computer vision. At each stage, input is received from the previous step and output is fed to the next step. Table1 shows the important research sub-areas at each step.

Steps	Sub-Area
Acquisition	Through camera
Per- Processing	Extracting frames form the video, Filtering
Segmentation	Color Segmentation, Noise Reduction
Representation	Showing only objects in another video
Result Generation	Generating results of individual Spots

Table 1: Methodology Description

For each of the sub area, there has been an extensive past and ongoing research, numerous algorithm have been developed for each of them. For example, there is ongoing research for performing accurate background subtraction and many more in the field of computer vision. All the techniques mention above are important to get the desire results; the explanation of each technique is given below.

3.3.1. Video Acquisitions

The video acquisition plays major part in this work, as the data has to be acquired very clearly so that segmentation becomes very easy. It depends up on how good the camera calibration is. Camera calibration is used to get information of the real world or we can say as primarily, finding the quantities internal to the camera that affect imaging process. Camera calibration has always been an essential component of photogrammetric measurement, with self-calibration nowadays being an integral and routinely applied operation within photogrammetric triangulation, especially in high-accuracy close-range measurement [28]. Accurate camera calibration and orientation procedures are a necessary prerequisite for the extraction of precise and reliable 3D metric information from images. Camera calibration continues to be an area of active research within the CV community, with a perhaps unfortunate characteristic of much of the work being that it pays too little heed to previous findings from photogrammetry [28].

Camera calibration is the process of finding the true parameters of the camera that produced a given photograph or video. Some of these parameters are focal length, format size, principal point, and lens distortion. But in this thesis we have worked only on format size, to reduce the size of each frame in a video. Camera calibration is often used as an early stage in computer vision.

The first step in any computer vision application is to input a digital video of the problem domain. This is normally done using digital camera connected to a computer. Unlike film-based cameras, digital cameras have an image sensor that converts light into electrical charges [29]. The image sensor employed by most digital cameras and web cams is a charged coupled device (CCD). Some low-end cameras use complementary metal oxide semiconductor (CMOS) technology. CMOS technology improves the image quality however it is comparatively slower than CCD cameras. Two important attributes of a camera with respect to computer vision applications are resolution and color. The amount of detail that the camera can capture is called the resolution, and is measured in pixels. The more pixels your camera has, the more details it can capture. High resolution video frames can enhance the performance of pre-processing and segmentation algorithms in computer vision applications. On the other hand, higher resolution can lead to increased processing power and time. Image sensors use filter to look at the incoming light in its three primary colors red, green and blue. A computer vision system either uses a single camera or multiple cameras based on the problem domain. Either way, an accurate camera calibration is always a requirement.

In this thesis we have worked with (Mini cute web camera) which cost around 100 sek, which is connected to vehicle at different angle and with different source of light. With this vehicle we have worked on 4 different types of data, which are different to one another in angle position as well as in source light conditions. In only first data procedure we use artificial light in the remaining data acquisition procedure no artificial light source were used. Here we have 4 different type of data acquired through “Intelligent vehicle” and “Test vehicle” as well. The different data acquisition procedure and frames acquired are given below.

TestData 1

The data (video) acquisition process here is done using an “Intelligent vehicle” on a real track in Sweden (figure 1), ahead of vehicle we have rectangular box in which we hooked a camera perpendicular to the rail and light adjacent to it. The topmost of the box is fixed in front of the vehicle and the bottom is on the rail, so that it can move smoothly on the rail when the vehicle is in motion. The box is enclosed to divert the natural light and eliminate reflections and noises inside it. In this data acquisition we use only one (halogen) light of less frequency and it focus on the rail to see the spot more clearly and precisely. In this acquisition we can prevent the noise caused by vegetation, snow, etc. But, can't prevent the noise caused by reflections inside the box. The unnecessary data is not recoded in this type of acquisition, camera will focus on the rail and other parts of track are covered by the box. Few sample images of data acquisition are given below.

Few sample images of Test data1



Figure 8: Sample Images of TestData1

Test Data 2

The data (video) acquisition procedure here is totally different from what has been done above in TestData1; here the data is recorded on test track using a test vehicle due to unavailability of the original track. This data was recorded to avoid the noises what we get from (TestData1) and to know what happens if we remove the box. In this acquisition we use test vehicle, the camera is fixed to one side of the vehicle with certain height to the rail and no artificial lights were used. But at certain height of camera or top of test vehicle we used a cardboard and covered with cloth, to block the sunlight from one side (Figure 9). With direct sun light the

spots can't be seen due to more intensity. The data is taken by pushing the test vehicle at certain speed. In this data the spot is easily visible and the segmentation will be more perfect than the TestData1, but in this case there is a problem of ambiguities. The capture area here is more comparing to Data1; here we can see rail, plates, fastenings and sleeper etc as well. In this data possibilities of finding noise increases. These noises can mix up with spot as well.



Figure: 9 Camera Position in this procedure

Few sample images of Test data 2



Figure: 10 Sample Images Data Type 2

Test Data 3

The data (video) acquisition process here is done on a test track using a "Test vehicle" as done in TestData2. The data acquisition procedure here is same as in previous data (TestData2), but we change the angle of the camera to 0 degree from the rail (Figure 11) or Parallel to the rail. The main reason here is to overcome the defects caused in TestData2 by avoiding the noise and dummy spots. Every thing is same in this acquisition procedure except the camera angle,

which is 0 from the rail. The camera is attached to a corner of a test vehicle and it will be position parallel to the rail so that we can see one side of the rail, fastenings and plates as well. Here we can reduce the direct noise what we get in the previous case, but we can't reduce indirect noise in this case. As we reduce the camera angle the focus was more towards the light and it causes more noise. In this procedure the possibility of noise increases due to sun light as seen in the below figure, the sun rays can brighten the data at top of the object (spot), and segmentation of those spot will be difficult. We tried our best to stop the light but at some point the sun light pass through it and it will effect the segmentation of spots as you can see in the sample images given below.



Figure: 11 Sample Images DataType3

Test Data 4

The data (video) acquisition process here is done on a test track using a "Test vehicle" as done in previous data. The data acquisition procedure here is same as in previous data (TestData3), but we still lower the angel of the camera as compared to Testdata3 so that we can avoid the noise of sun light what we get in the previous one. In this data collection procedure we have removed every thing from the test vehicle (cloth and cardboard) so that we can get clear data in a normal day light. The camera has to keep at certain distance high so that it doesn't hit with any sleepers or stones etc. This data works very well, but as sunlight increases there will possibility of more brightness in it, (figure 13) where two images are in good condition, remaining are with more brightness. The images with good illumination segment the spot

easily; the other counter part with bad illumination will not segment at all (See Results). Below are few sample images of this type of data collection procedure.



Figure: 12 Camera Position for TestData4

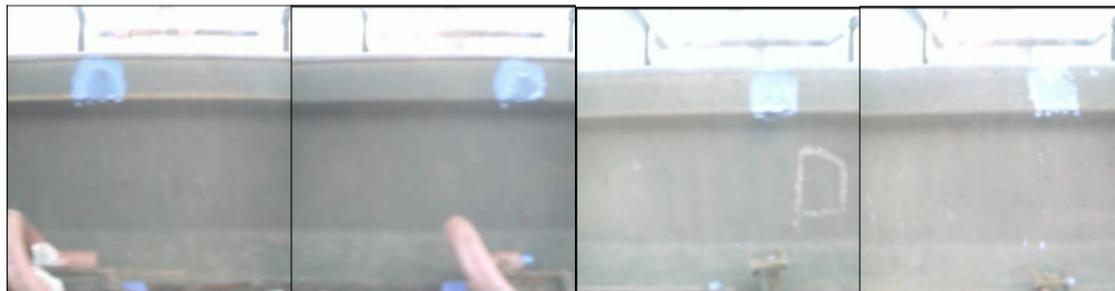


Figure: 13 Sample Images of TestData4

3.3.2 Pre processing

The video format acceptable in OpenCV library is (.avi), before using the video in our algorithm we convert videos to avi format. All the videos acquired are in RGB color format as we are working on color segmentation process. We changed the dimension of captured frame to 300 x 300 pixels, for easy processing with out in loss in resolution. Now we used this video in proposed method. But, before going to segmenting we pre-process the image because when ever an image is acquired by a camera, often the vision system for which it is intended is unable to use it directly in segmentation. Random variations in intensity, variations in illumination or poor contrast could corrupt the image. This must be dealt with in early stages

of vision processing [16]. This is true in public space as the video capturing device could be exposed to various levels of temperature and change, which could result noise in the image. We are working on a real time video, which runs along railway track capturing all the related information for segmentation, and by using this video we can get the desired results. Frames which are extracted from the video are not similar because the light source at the starting point will not be same at the ending point, we have to work on each frame at a time and do the further processing to get the desired result.

The Pre-processing steps before segmentation are given below.

3.3.2.1 Gaussian Blur

Gaussian blur is also known as Gaussian smooth which is used as the per-processing stage in this thesis, in order to enhance image structure at different scales. Smoothing filters are used for blurring and noise reduction. Smoothing is one of the most fundamental and widely studied problems in low-level image processing. The main purpose of image smoothing is to reduce undesirable distortions and noise while preserving important features such as discontinuities, edges, corners and textures [30]. Gaussian blue is used to remove small details from an image prior to large object extraction and bridging of small gaps in lines or curves. Noise reduction can be accomplished by blurring with a linear filter and also by non-linear filtering [31].

The output of a smoothing, linear spatial filter is simply the average of the pixel contained in the neighborhood of the filter mask. These filters some times called as averaging filters. The mask what we have used hear are 3x3, 5x5 and 7x7 to blur the required image. Gaussian blur is the basic form of all kinds of blur methods, which means each of other kinds of blur methods can be simulated by a respective Gaussian Blur [32].

3.3.2.2 RGB to HSV

The conversion of RGB format images to HSV format is important for segmentation problem, because hue feature is invariant to highlight as mention above. In this thesis we are using

some libraries such as “OpenCv”, it has some inbuilt function which can convert the image from RGB to HSV and vice versa. Or we can transform from the source RGB space to HSV space by already established formulas which are given below.

To convert form RGB to HSV, first begin with normalized RGB values:

The Value is given by

$$V = \text{Max}(R, G, B)$$

The Saturation Component is calculated by:

$$S = \frac{\text{Max}(R, G, B) - \text{Min}(R, G, B)}{\text{Min}(R, G, B)} \quad \text{if Max}(R, G, B) \neq 0$$

$$S = 0 \quad \text{if Max}(R, G, B) = 0$$

The H value is given by

H is undefined is S = 0

$$H = \frac{G - B}{\text{Max}(R, G, B) - \text{Min}(R, G, B)} \quad \text{if } R = \text{Max}(R, G, B)$$

$$H = 2.0 + \frac{B - R}{\text{Max}(R, G, B) - \text{Min}(R, G, B)} \quad \text{if } G = \text{Max}(R, G, B)$$

$$H = 4.0 + \frac{R - G}{\text{Max}(R, G, B) - \text{Min}(R, G, B)} \quad \text{if } B = \text{Max}(R, G, B)$$

$$H = 60 * H$$

If $H < 0$ then $H = H + 360$

These predefine formulas can also be used to convert form RGB to HSV [35]. But, in our proposed method we used an inbuilt function which can do this process.

`cvCvtColor (const CvArr* source, CvArr* destination, CV_BGR2HSV)`

This function converts the source image (frame) from RGB to HSV and send to Destination image (frame). This process has to be done after removing the noise to get a clear image after segmentation.

3.3.3 Image Segmentation

This technique is used to identify the moving objects in an image sequence. Or we can say that, its purpose is to identify the object and discard remaining pixels for the image sequence. After video acquisition, each frame is pre-processed to clean the noise and to get more details from it. The pre-processing depends upon the data; if we have noisy data and bad illumination, we use pre-processing techniques before converting it into HSV. Each RGB frame is converted to HSV plane, from the HSV plane we take the HSV values of each pixel and normalize those values to be in the range between [0 and 1]. In this thesis, we are using a byte image, a byte image is assumed to have values in the range [0...255] [33] and the each pixel value of the HSV plane frames are normalized as mentioned in the above equation. We have given some cutoff values for each color, if these normalized values come in the range of given cutoffs for a given color, then these values of each pixel are displayed in another binary image. In this binary image, we can see only the area which is to be segmented by the above technique.

Algorithm

- Step 1: Read Height and Width of Image
- Step 2: For each Height (Image)
 - For each Width (Image)
 - Calculate HSV values
- Step 3: Take the values which come under a particular predefined range
- Step 4: Show only these pixel values which come under this range
- Step 5: Repeat until no frames left.

The above algorithm takes an HSV frame and gives the output frame as a binary image, by highlighting only the area that we wanted to highlight as color segmentation. Before

segmenting in another video we do some noise reduction technique to avoid the ambiguity of spots in identification.

3.3.4 Object Separation

To separate the object after segmentation we have to remove these kinds of noises which are given below. The noises can't be removed in pre processing, it can be avoided by altering the data acquisition procedure or it has to be removed after segmentation. Some noises can be removed either by increasing the threshold value of segmentation as well.

1. Reflections
2. Vegetation & Snow
3. Sun Light

Reflection

The problem caused due to reflection are, ambiguity of the spots, breaking the object in to two by focusing in the middle as in DataType1 where we used an artificial light which focus in the middle of the track. The objects (spots) are marked by spraying, due to force in center the color will spread to corner of the spot. In center no color will be left and it also causes breaking the spot into two. There is small gap beside the rail and the box through with reflection causes a blue line (see figure 12) so all this noises have to eliminate before going for representation. These kinds of noises are more in first procedure which is (DataType1), it can be avoided by improving the acquisition procedure, other noises such as vegetation and direct sun rays are already avoided here because of the box.

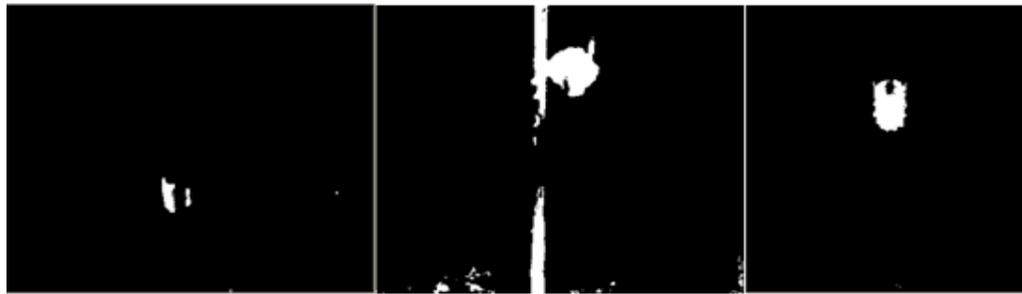


Figure: 14 Reflection noise in Segmentation

Vegetation & Snow

This problem is very common in Sweden where we have more snow in winter and more photosynthesis in summer. It is difficult to work in winter, so don't care about the snow. In summer we have the problem of vegetation near the railway track (Figure 5). These noise doesn't affect the first procedure but in the second where capture angle of camera is increased and we capture every thing beside the rail. The possibility of these noises may occurs or corrupt our data after segmentation. Where we can see multiple spots after segmentation, this noise will affect all the other data acquisition procedures except the first one.

Sun Light

This noise is common in every procedure, but we tried every possibility to avoid it in our acquisition procedures. The TestData3 and TestData4 are affected with this noise. This noise is very problematic it creates dummy spots in (Data3) and it eliminates the spots in (Data4). These kinds of noise have to be taken case very effectively before going for separation. See the (figure 15).

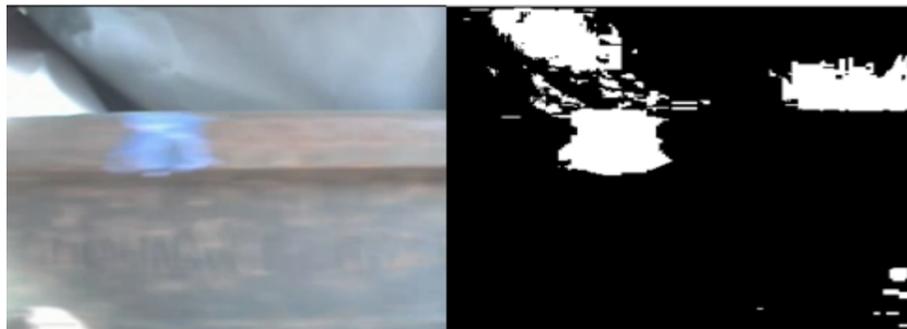


Figure: 15.1 Dummy Spots created by Sun Light



Figure: 15.2 Spot Eliminated by Sun Light

Different data acquisition procedures have different noises, some noises can be removed as mention above, but in some case the noise can't be removed totally still we have multiple spots. Theses spots have to remove as well before representing single desired spot on another frame. See the below figure (16).

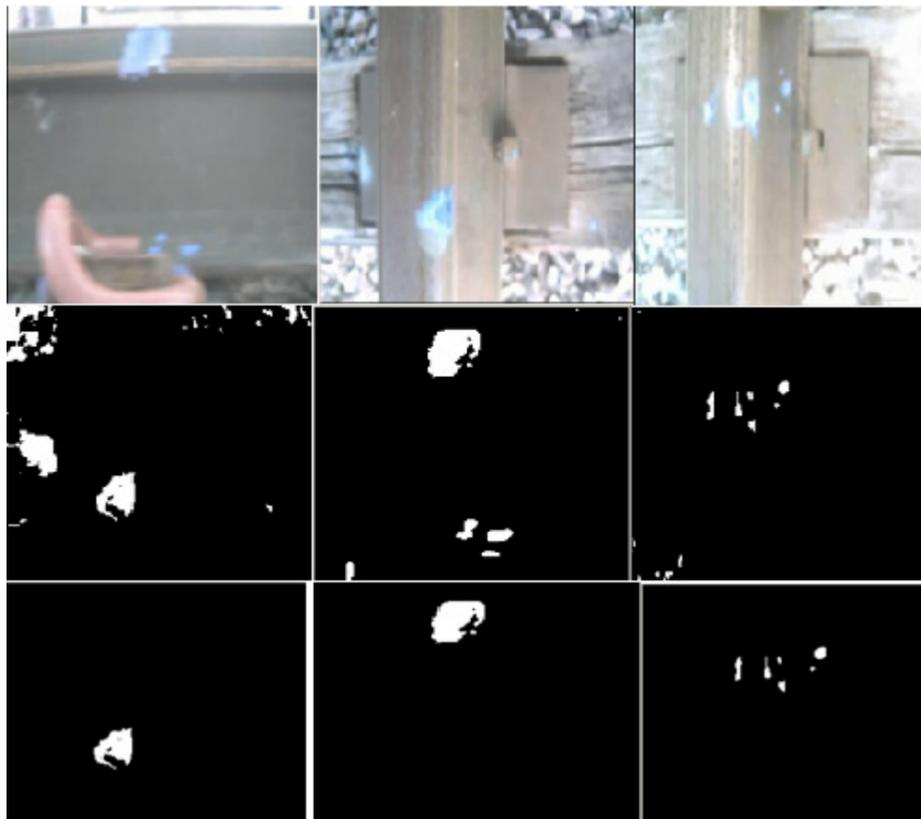


Figure 16: Different Noise caused by angle, reflection etc.

Noise Reduction Technique:

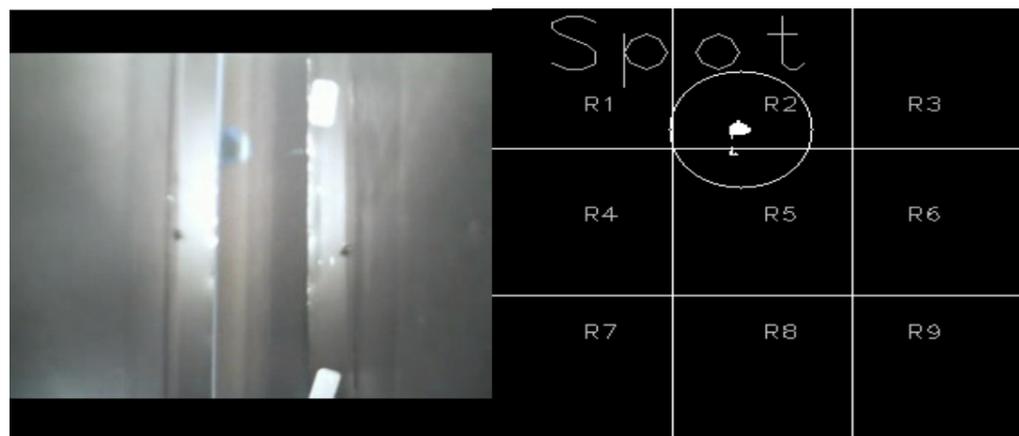
This technique is used very effectively in proposed Method; the reason behind this is to remove the noises acquired above. After segmentation of object using certain threshold value we get the some results, for that results this technique is used to remove the unwanted noise so that we can segment object of interest very easily. The above noise has to be removed to get desired results; here we can't differentiate between noise and object (spot) above Figures (16). They both look the same unless the maintenance operator check the original video and make

conformation. There will be problem in generation of results as well it generate object in multiple regions and it causes lot of confusion for the readers of the report, to avoid that we use this above technique.

The Data acquisition procedure follows a methodology that we focus on spot, such that it will be the in the center in every frame, either in horizontal or vertical the object (spot) will not be in corners any way (exception in TestData4 where it is almost top). The noise above is not with the object but beside the object so by removing the noise from the corners we can segment the desire object as given above. We know in every acquisition the spot is on the rail, keep the rail data and remove the other data beside the rail for vertical and below for horizontal as well. This technique works well for this data, but may be create some problems if the object is not in center. The technique can be avoided if we improve the data acquisition procedure, where just tune the threshold value and get the results. Because these noise caused by angle of capture, sunlight, reflection and it can be avoided with improved equipment where we avoid all this things.

3.3.5 Object Representation

In this method we represent an object in another video, where we can see only the object in a black background and there will be no ambiguities with the noise. The object is covered with a circle to know particular region of interest where the object is lying. The other representation here is we divided each frame in 9 regions so that to know particular region of the object (spot). The figure (18) shows representation of different data types tested in Methodology Development.



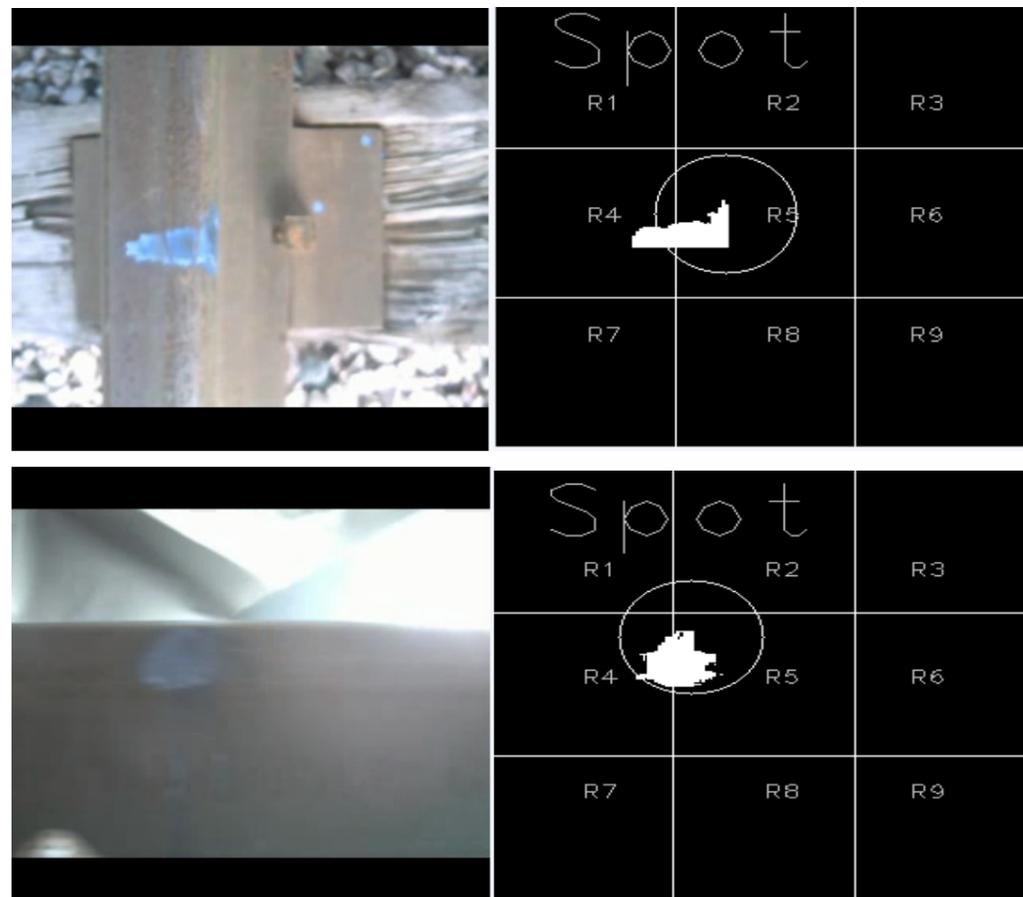


Figure: 18 Representations of Different Data

Circle

A circle is drawn around the spot, so that we can easily differentiate between noise and the required object of interest (spot). It will also help the maintenance operators to justify their decision in identification of spot. The spot is drawn by counting the number of pixels in a segmented object; certain threshold value is given, if the no of pixels are greater than threshold draw the circle. For some objects which are brighter the circle is not drawn but the spot is counted in result generation.

Regions

Each frame of segmented video is divided into 9 regions; the 9 region frame will be fixed on the output video. The segmented object (spots) keeps on moving under this frame so that we

can identify in which region the spots are moving. The regions were divided equally into 9 regions in one frame, size of 300 x 300 pixels. This method gives a unique way to see the continuous spots (All Data) and sunken plates (in TestData3 and TestData4). As mention earlier the methodology to capture data, where the object (spot) is center of attraction and one follows the other in the same direction. If the spots are moving in the vertical direction, they should be in regions (R1,R4,R7) or (R2,R5,R8) or (R3,R6,R9). They may be exception in some video where a spot passes through border of regions and also no region at all due to removing corner data. To check the sunken plates, the spot has to be travel in horizontal direction. If the first on found to be in regions(R1,R2,R3) and the following object to be found in regions(R1/R4,R2/R5,R3/R6) border of both regions. For this we can say the second one has the sunken plates and result generation will give more information regarding this which is given below.

3.3.6 Result Generation

This is very important because it gives backup for the work has been done so far, if the maintenance operators miss some thing while monitoring the sleepers they can check the results and carry on their work. As we know that each frame is divided into 9 regions and the generation of results will be on these regions like (eg spot no x has been found in region no y). We know predefined region in each frame, we just check how many pixels are passing in each region if they cross certain threshold we print the spot number and the region where it found. Spots are counted by a counter, the counter will not increments unless the spot disappears from the frame. The generation of results also helps the maintenance operators to know more critical conditions of the bad sleepers caused by sunken plates as mention earlier.

4. Experimental Results and Analysis

4.1 Introduction

Discussions in previous sections have given a comprehensive statement about the all subroutine of entire algorithm procedure. However, in order to test and evaluate the efficiency our algorithm, 4 different types of real time videos are considered. They represent all general difficulties due to environment and condition monitoring of railway sleeper in Sweden. The

object shape is ignored in the current experiment since it requires particular attention to be paid while marking. It can be implemented in the future experiments. In most cases, it is not easy to segment the object as we have tested different types of video acquisition in this research experiment. The data representing single objects from different angles have been tested; analysis and results are given below.

In this work, the main objective of this work is to segment the object (spot) very precisely and generate the result, it has to be done automatically so that automation can be achieved as required in condition monitoring of wooden railway sleepers problem. We tested different videos for automation of this work and results are given below.

4.2 Test Data 1

Results & Analysis:

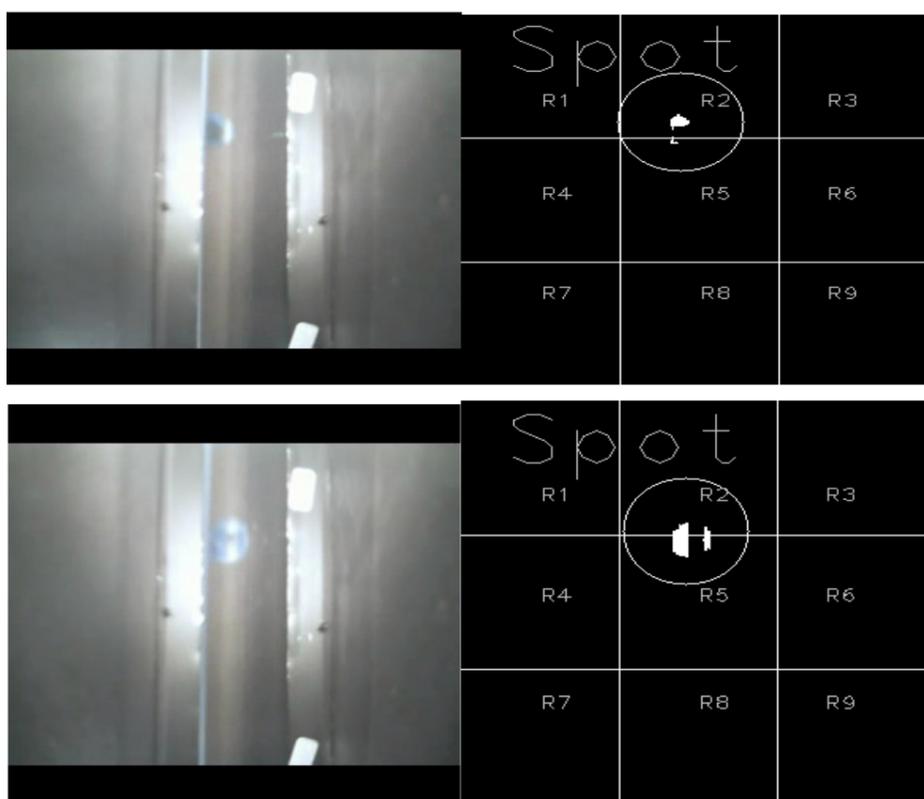
Data	No of Spots	Segmented	Time Taken /Actual Time	No frames/sec	No of frames	Speed of Video
1	29	29	206 / 130 sec	15.21	3135	8 to 10 Km/h
1	29	28	53 / 130 sec	59.15	3135	8 to 10 Km/h
2	47	45	225 / 143 sec	15.34	3453	8 to 10 Km/h
2	47	42	58 / 143 sec	59.53	3453	8 to 10 Km/h
3	29	13	120 / 64 sec	12.98	1558	13 to 15 Km/h
3	29	11	26 / 64 sec	59.9	1558	13 to 15 Km/h

Table 2: Results of TestData1

The TestData1 as mention above was tested in our approach, the experiment results shows that the procedure works well for average speed of vehicle around 10km/h and identification of spots will be in between 44% to 95 %. The results are not stable, since the speed will not be same for every video nor reflections either. The (data3) in above table gives less than 50% because of its speed around 15km/h, and in (data1) gives almost 100% for the speed of around 10km/h. This is the reason for variation of results in range of 50% to 95%. For the same speed video we experiment it by accelerating the frames a little bit and the results were reduced in between 40% to 90%. The time taken in this experiment for each video after accelerating is $\frac{1}{4}$ of the time taken before accelerating. It is obvious that video used as it is will be more in

identifying the spots compared to accelerating it. By improving the efficiency we may lose some quality. The better way is to record the data slowly (above Data 1 and 2) around 10 to 12 km/h and improve the speed by accelerating through programming the identification will be slightly less compared to other. But it can be avoided by altering the data acquisition procedure and using some quality equipment.

The results can be improved even more if we slightly alter data collection procedure, by equal distribution of light inside the box. The single light will focus on one end and it may cause some reflections near the spots or beside the rail as mention above. The segmentation will be easier if we have good illumination inside the box, more illumination will brighten the object and segmentation will take more time, because pre-processing has to be done before segmentation. The results mentioned above are from different video taken by same procedure, the result of data 1 from which above table is defined is given below. It shows one frame of a video, but the main aim of this thesis is to work on a sequence of frames. If we run the video it will look like as shown below.



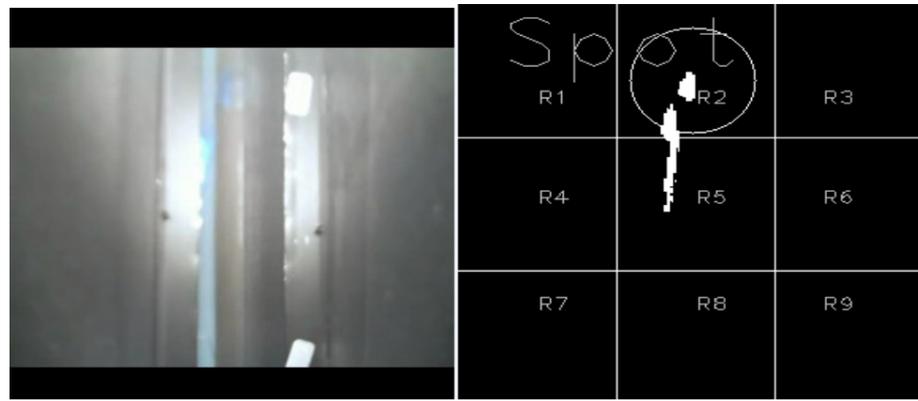


Figure: 19 Segmented Images of TestData1

```
E:\asif data\VideoThesis\Debug>VideoThesis.exe TestData1.avi
width: 300, height: 300, chanel: 3, origin: 1
Spot->1 in R5 Spot->2 in R5 Spot->2 in R2 Spot->3 in R5 Spot->3 in R2 Spot->4 in R5 Spot->4 in R2 Spot->5 in R5 Spot->5 in R2 Spot->6 in R5 Spot->6 in R2 Spot->7 in R5 Spot->7 in R2 Spot->8 in R5 Spot->8 in R2 Spot->9 in R5
Spot->10 in R5 Spot->10 in R2 Spot->11 in R5 Spot->11 in R2 Spot->12 in R5 Spot->12 in R2 Spot->13 in R5 Spot->13 in R2 Spot->14 in R5 Spot->14 in R2
Spot->15 in R5 Spot->15 in R2 Spot->16 in R5 Spot->16 in R2 Spot->17 in R5 Spot->17 in R2 Spot->18 in R5 Spot->18 in R2 Spot->19 in R5 Spot->19 in R2
Spot->20 in R5 Spot->20 in R2 Spot->21 in R5 Spot->21 in R2 Spot->22 in R5 Spot->22 in R2 Spot->23 in R5 Spot->23 in R2 Spot->24 in R5 Spot->24 in R2 Spot->25 in R5
Spot->25 in R2 Spot->26 in R5 Spot->26 in R2 Spot->27 in R5 Spot->27 in R2 Spot->28 in R2 Spot->29 in R5 Spot->29 in R2
Total Time Taken In Sec = 201
No of Frames = 3135
No of Frames per second = 15.597
No of Spots: 29
```

Figure 20: Generated Results of TestData1

```
E:\asif data\VideoThesis\Debug>VideoThesis.exe TestData1.avi
width: 300, height: 300, chanel: 3, origin: 1
Spot->1 in R5 Spot->1 in R2 Spot->2 in R5 Spot->2 in R2 Spot->3 in R2 Spot->3 in R5 Spot->4 in R5 Spot->4 in R2 Spot->5 in R5 Spot->5 in R2 Spot->6 in R5 Spot->6 in R2 Spot->7 in R5 Spot->7 in R2 Spot->8 in R5 Spot->9 in R5
Spot->9 in R2 Spot->10 in R5 Spot->10 in R2 Spot->11 in R5 Spot->11 in R2 Spot->12 in R5 Spot->12 in R2 Spot->13 in R5 Spot->13 in R2 Spot->14 in R5 Spot->14 in R2
Spot->15 in R5 Spot->15 in R2 Spot->16 in R5 Spot->16 in R2 Spot->17 in R5 Spot->17 in R2 Spot->18 in R5 Spot->18 in R2 Spot->19 in R5 Spot->19 in R2
Spot->20 in R5 Spot->20 in R2 Spot->21 in R5 Spot->21 in R2 Spot->22 in R5 Spot->22 in R2 Spot->23 in R5 Spot->23 in R2 Spot->24 in R5 Spot->24 in R2 Spot->25 in R5
Spot->25 in R2 Spot->26 in R5 Spot->26 in R2 Spot->27 in R2 Spot->28 in R5 Spot->28 in R2
Total Time Taken In Sec = 53
No of Frames = 3135
No of Frames per second = 59.1509
No of Spots: 28
```

Figure 21: Generated Results of TestData1 after Acceleration

The above result generation is important in this work, because there may be chance of missing some information by maintenance operators while working with live data. It is better if we have any backup to cover few mistakes. Each frame was split into 9 equal regions to know where the object is lying on the rail; in this data the object (spot) is moving in vertical

direction so the possibilities of finding the spot will in regions (R2, R5,R8) as rail is in center point in video acquisition. If spot identified in (R2, R4, R8) we can say that it passes through border at one point or if spot identified in (R1, R4, R7) while others are in (R2, R5, R8) then we can say that it is noise not the spot. This procedure can followed for all the data test here and it can be analyzed by seeing the generated report.

4.3. Test Data 2

Results & Analysis

Data	No of Spots	Segmented	Time Taken / Actual Time	No frames/sec	No of frames	Speed of Video
1	25	27	76 / 47 sec	15.01	1141	6 to 8 Km/h
1	25	25	20 / 47 sec	57.05	1141	6 to 8 Km/h

Table 3: Results of TestData2

The TestData2 was tested with proposed algorithm; the results are given in (table). In this procedure we worked with test vehicle (See Figure) to record the data. The data recorded by pushing the vehicle around 6 to 8 km/h on the test track (See Figure). The experiment result shows that the captured video works very well for the given speed of vehicle. The identification of spots for videos captured in this procedure is more than 100%. If we accelerate the video even more the results were reduced to 100%. The acceleration is done to get results in quick time, as it takes very less time of the original one. We may loss some segmentation efficiency by accelerating videos around 5% to 10% depending up on video quality and vehicle speed. The reason for more identification here in this procedure than actual is, captured area through camera (See Figure 10) where we can see more noise with spots. The speed of this video is very slow compare to previous data. If we able to avoid the noise, this procedure will be prefect where we can accelerate and get the results in few secs. As the objects (spots) are clearly visible in the video, by accelerating it doesn't affect the quality of results as seen above.

The results here are different as compared to TestData1; here the accelerated video gives better results compared to normal video acquired. The main problem here is capture of similar noise with spot; the captured angle is more so we can see other noises beside the spot

as well. This may increase the ambiguity of spots while segmenting. In this procedure segmentation will be good because data acquired in good illumination (natural day light) with out any sun rays falling on the rail. The results mentioned above are from a video taken by above procedure, the result of data 1 from which above table is defined is given below. It shows one frame of a video, but the main aim of this thesis is to work on a sequence of frames. If we run the video it will look like as shown below.

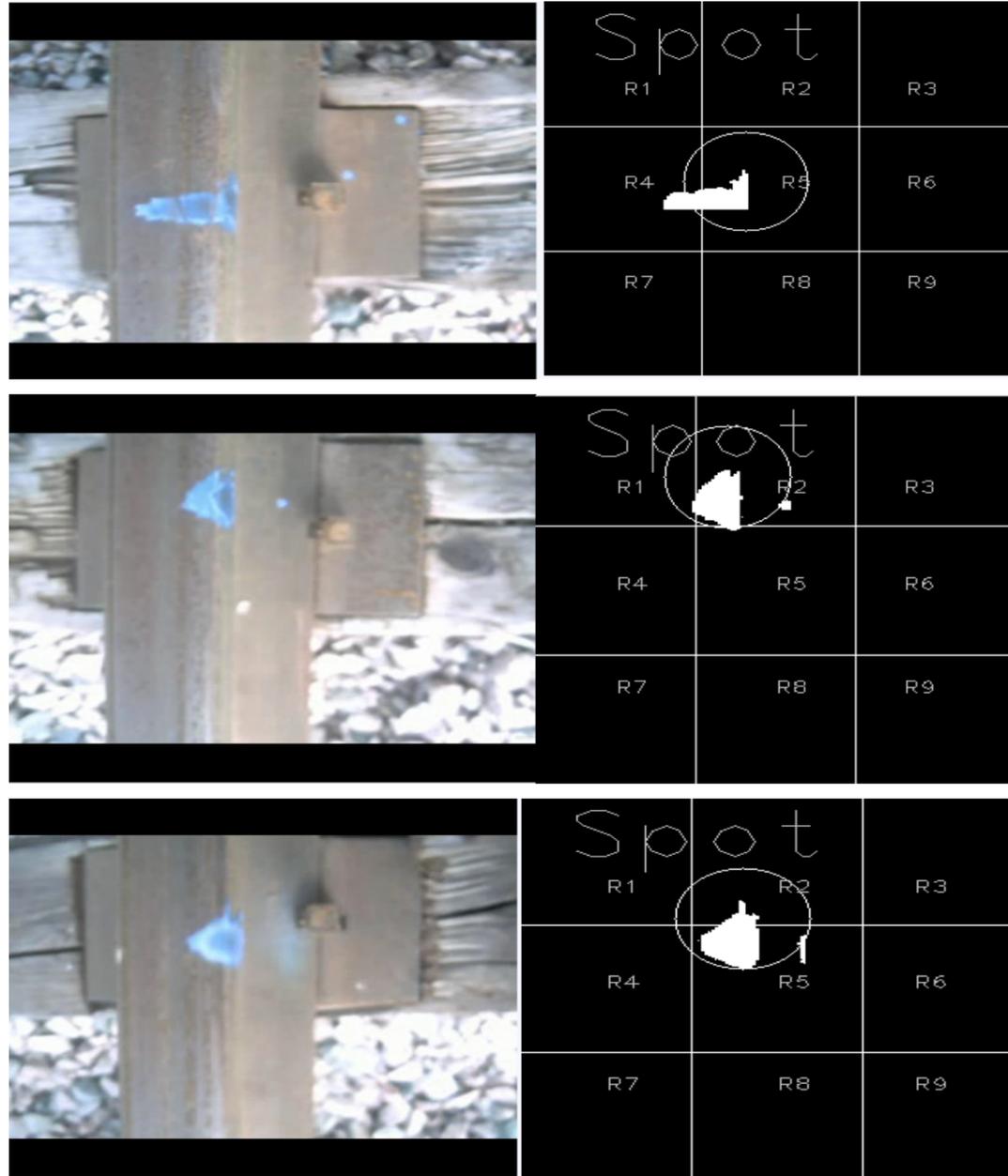


Figure 22: Sample Images of TestData2

```

E:\asif data\VideoThesis\Debug>VideoThesis.exe TestData2.avi
width: 300, height: 300, chanel: 3, origin: 1
Spot->1 in R4 Spot->1 in R5 Spot->1 in R2 Spot->1 in R1 Spot->2 in R5 Spo
t->2 in R2 Spot->3 in R5 Spot->3 in R4 Spot->3 in R2 Spot->3 in R1 Spot->4
in R4 Spot->4 in R5 Spot->4 in R2 Spot->4 in R1 Spot->5 in R4 Spot->5 in R5
Spot->5 in R1 Spot->5 in R2 Spot->6 in R4 Spot->6 in R5 Spot->6 in R1 Spo
t->6 in R2 Spot->7 in R5 Spot->7 in R2 Spot->8 in R5 Spot->8 in R2 Spot->9
in R5 Spot->9 in R2 Spot->10 in R5 Spot->10 in R2 Spot->10 in R4 Spot->10 i
n R1 Spot->11 in R5 Spot->11 in R1 Spot->11 in R2 Spot->12 in R2 Spot->12 i
n R5 Spot->13 in R5 Spot->13 in R2 Spot->14 in R4 Spot->14 in R5 Spot->14 i
n R1 Spot->14 in R2 Spot->15 in R5 Spot->15 in R4 Spot->15 in R2 Spot->15 i
n R1 Spot->16 in R4 Spot->16 in R5 Spot->16 in R1 Spot->16 in R2 Spot->17 i
n R5 Spot->17 in R4 Spot->17 in R2 Spot->17 in R1 Spot->18 in R5 Spot->18 i
n R2 Spot->19 in R5 Spot->19 in R2 Spot->20 in R4 Spot->20 in R1 Spot->21 i
n R4 Spot->21 in R5 Spot->21 in R1 Spot->21 in R2 Spot->22 in R5 Spot->22 i
n R2 Spot->23 in R5 Spot->23 in R1 Spot->23 in R2 Spot->23 in R4 Spot->24 i
n R4 Spot->24 in R5 Spot->24 in R1 Spot->24 in R2 Spot->25 in R4 Spot->25 i
n R5 Spot->25 in R1 Spot->25 in R2 Spot->26 in R4 Spot->26 in R5 Spot->26 i
n R1 Spot->26 in R2 Spot->27 in R1
Total Time Taken In Sec = 76
No of Frames = 1141
No of Frames per second = 15.0132
No of Spots: 27

```

Figure 23: Generated Results of TestData2

```

E:\asif data\VideoThesis\Debug>VideoThesis.exe TestData2.avi
width: 300, height: 300, chanel: 3, origin: 1
Spot->1 in R4 Spot->1 in R5 Spot->1 in R2 Spot->1 in R1 Spot->2 in R5 Spo
t->2 in R2 Spot->3 in R4 Spot->3 in R5 Spot->3 in R1 Spot->3 in R2 Spot->4
in R4 Spot->4 in R5 Spot->4 in R1 Spot->4 in R2 Spot->5 in R1 Spot->5 in R2
Spot->5 in R5 Spot->6 in R4 Spot->6 in R5 Spot->6 in R1 Spot->6 in R2 Spo
t->7 in R2 Spot->7 in R5 Spot->8 in R5 Spot->8 in R2 Spot->9 in R5 Spot->9
in R2 Spot->10 in R2 Spot->10 in R4 Spot->10 in R5 Spot->10 in R1 Spot->11
in R5 Spot->11 in R1 Spot->11 in R2 Spot->12 in R2 Spot->12 in R5 Spot->13
in R2 Spot->14 in R4 Spot->14 in R5 Spot->14 in R1 Spot->14 in R2 Spot->15
in R4 Spot->15 in R5 Spot->15 in R1 Spot->15 in R2 Spot->16 in R4 Spot->16
in R5 Spot->16 in R1 Spot->16 in R2 Spot->17 in R4 Spot->17 in R5 Spot->17
in R2 Spot->17 in R1 Spot->18 in R5 Spot->18 in R2 Spot->19 in R5 Spot->19
in R2 Spot->20 in R4 Spot->20 in R1 Spot->21 in R4 Spot->21 in R5 Spot->21
in R1 Spot->21 in R2 Spot->22 in R5 Spot->22 in R2 Spot->23 in R5 Spot->23
in R1 Spot->23 in R2 Spot->23 in R4 Spot->24 in R4 Spot->24 in R5 Spot->24
in R1 Spot->24 in R2 Spot->24 in R1 Spot->24 in R2 Spot->25 in R1 Spot->25
in R2 Spot->25 in R4 Spot->25 in R5
Total Time Taken In Sec = 20
No of Frames = 1141
No of Frames per second = 57.05
No of Spots: 26

```

Figure 24: Generated Results of TestData2 after Acceleration

The above result generation is important in this work as mentioned earlier; it works fine for clear track rather than full of vegetation and snow. It is very difficult to divert the sun rays if this procedure is implemented on a real track. This procedure will generate the results same as it is in TestData1 where the objects (spots) are moving in vertical direction. As you can see in the results the spot is moving from region 8 to region 2 or vice versa. The generated results are given above.

4.4 Test Data 3**Results & Analysis**

Data	No of Spots	Segmented	Time Taken /Actual Time	No frames/sec	No of frames	Speed of Video
1	24	26	68 / 43 sec	15.32	1042	4 to 6 Km/h
1	24	25	18 / 43 sec	57.88	1042	4 to 6 Km/h
2	26	24	62 / 40 sec	15.51	962	4 to 6 Km/h
2	26	22	17 / 40 sec	56.58	962	4 to 6 Mm/h

Table 4: Results of TestData3

The TestData3 was tested with proposed algorithm, the results shows that this procedure works well for average speed of vehicle around 6km/h and identification of spots were more for some data and less for another as seen above. This procedure is totally different from the above (TestData2) approach, this procedure was initiated to overcome the defects occurred in previous data (TestData2). In this procedure the capture angel was reduced and we concentrate only on the rail from one side. The noise caused by the data collection procedure was reduced, such as (Sleeper, plates and fastenings) where similar colors can be found on them due to some circumstances. However, the results give more than 100% for (data 1) and 92% for other (data 2). We experiment these videos by accelerating it, results were reduced even more and the time taken is very less compared to previous one. The images of this procedure are given below. In this type of data collection procedure the video is running in a horizontal way and the spot is moving from region 6 to region 4 and vise versa.

The procedure has similar problem, as seen in TestData2 where multiple spots were segmented due to noise. Here we change the procedure but still we have the ambiguity of spots, this time it not caused by the acquisition procedure but by the nature i.e. sunlight. If we avoid the sunlight in this procedure this method works very good as you can see for sample data2 in above table where the results were almost near 100% with out any ambiguities. The results of this video frame by frame are given below.

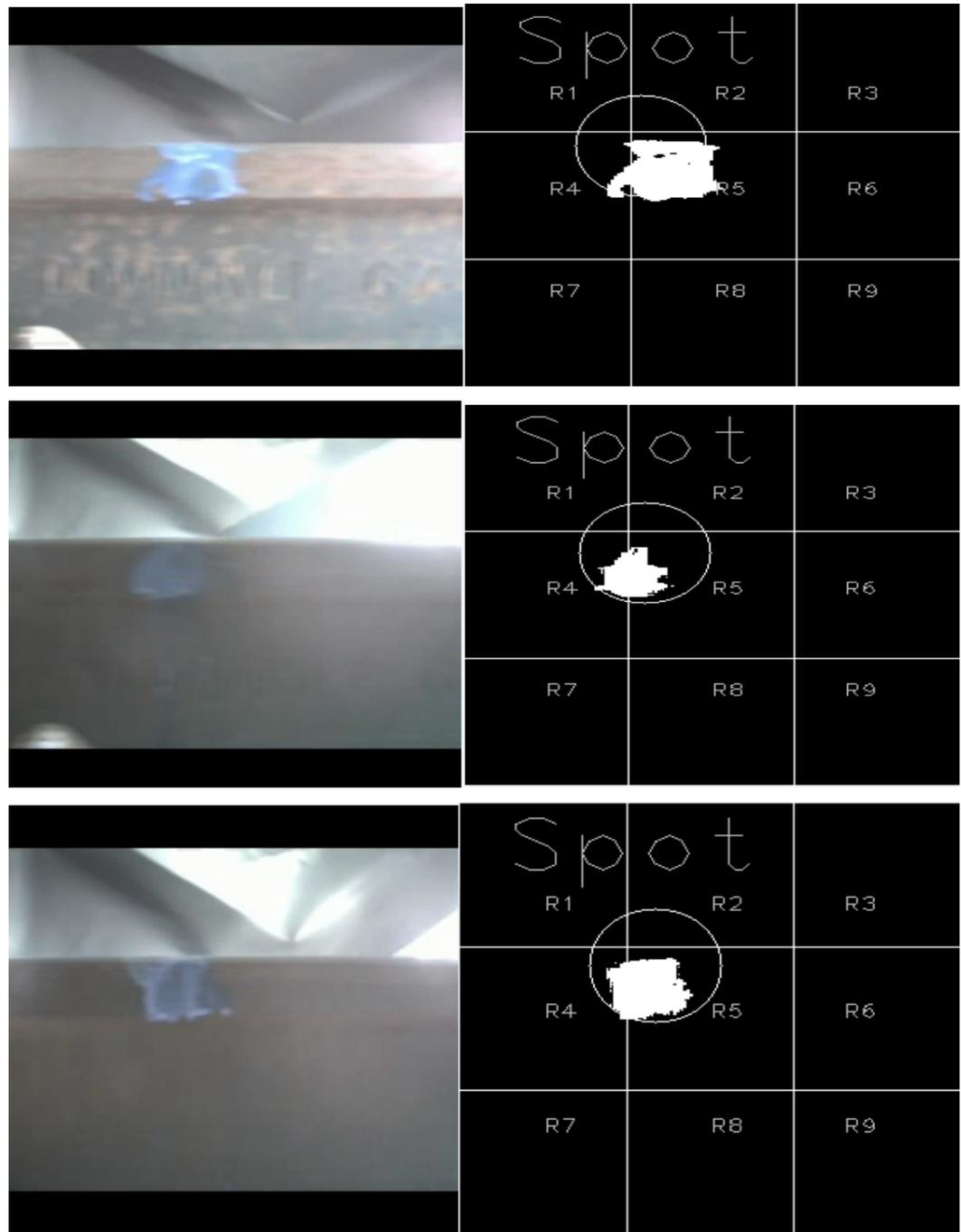


Figure 25: Sample Images of TestData3

```

E:\asif data\VideoThesis\Debug>VideoThesis.exe TestData3.avi
width: 300, height: 300, chanel: 3, origin: 1
Spot->1 in R5 Spot->1 in R4 Spot->2 in R5 Spot->2 in R4 Spot->3 in R5 Spot->3 in R4
Spot->4 in R5 Spot->4 in R4 Spot->5 in R5 Spot->5 in R4 Spot->6 in R5 Spot->6 in R4
Spot->7 in R5 Spot->7 in R4 Spot->8 in R5 Spot->8 in R4
Spot->9 in R4 Spot->10 in R5 Spot->10 in R4 Spot->11 in R5 Spot->11 in R4
Spot->12 in R5 Spot->12 in R4 Spot->13 in R5 Spot->13 in R4 Spot->14 in R5
Spot->14 in R4 Spot->15 in R4 Spot->15 in R5 Spot->16 in R5 Spot->16 in R4
Spot->17 in R5 Spot->17 in R4 Spot->18 in R5 Spot->19 in R5 Spot->20 in R5
Spot->20 in R4 Spot->21 in R4 Spot->22 in R1 Spot->23 in R5 Spot->23 in R4
Spot->23 in R4 Spot->24 in R5 Spot->24 in R4 Spot->25 in R1 Spot->26 in R1
Spot->26 in R2 Spot->26 in R2
Total Time Taken In Sec = 68
No of Frames = 1042
No of Frames per second = 15.3235
No of Spots: 26

```

Figure 26: Generated Results of TestData3

```

E:\asif data\VideoThesis\Debug>VideoThesis.exe TestData3.avi
width: 300, height: 300, chanel: 3, origin: 1
Spot->1 in R5 Spot->1 in R4 Spot->2 in R5 Spot->2 in R4 Spot->3 in R5 Spot->3 in R4
Spot->4 in R5 Spot->4 in R4 Spot->5 in R5 Spot->5 in R4 Spot->6 in R5 Spot->6 in R4
Spot->7 in R5 Spot->8 in R5 Spot->8 in R4 Spot->9 in R4 Spot->10 in R5
Spot->11 in R5 Spot->12 in R4 Spot->12 in R5 Spot->13 in R4 Spot->13 in R5
Spot->14 in R5 Spot->15 in R5 Spot->16 in R5 Spot->16 in R4 Spot->17 in R5
Spot->18 in R5 Spot->19 in R5 Spot->19 in R4 Spot->20 in R4 Spot->21 in R5
Spot->22 in R5 Spot->22 in R4 Spot->22 in R4 Spot->23 in R5 Spot->23 in R4
Spot->24 in R1 Spot->25 in R1 Spot->25 in R2
Total Time Taken In Sec = 18
No of Frames = 1042
No of Frames per second = 57.8889
No of Spots: 25

```

Figure 27: Results after acceleration for TestData3

The above result generation is important in this work as mentioned earlier; it works fine for clear track rather than full of vegetation and snow (see Figure). It is very difficult to divert the sun rays in this procedure. If we avoid the top half of the image in the preprocessing the segmentation will be improved, and we can avoid ambiguities as well from it. This procedure will generate the results same as it is in previous test samples, but here the objects (spots) are moving in horizontal direction. As you can see in the results the spot is moving from region 4 to region 6 or vice versa.

4.5 Test Data 4**Results & analysis**

Data	No of Spots	Segmented	Time Taken /Actual Time	No frames/sec	No of frames	Speed of Video
1	34	33	64 / 34 sec	13.04	835	4 to 6 Km/h
1	34	29	14 / 34 sec	59.64	835	4 to 6 Km/h
2	33	19	64 / 40 sec	15.32	981	4 to 6 Km/h
2	33	16	17 /40 sec	57.70	981	4 to 6 Mm/h

Table 5: Results of TestData4

The TestData4 is also tested with proposed method; results shows that the procedure works well for average speed of vehicle around 6km/h and identification of spots were good for some data and less for another as seen above. We tested this procedure to over the difficulties occurred in the previous procedure, to get the desired through put. The top of the image cause noise in the previous data was removed in this approach, to avoid noise caused by sun light as well. In this approach the ambiguities were not created and identification for (data 1) was near 97% and other (data2) was almost 50% due to intensities of light in both the data acquisition procedures. In this procedure the vehicle was not covered with any other material to avoid the sun light. If we have more light the results were reduced as in data2, if we have less intensity we may increase the identification of spots as in Data1. Due to the noise as you can see in above (figure 13) which causes more brightness through with spots were not recognized. For the same speed video we experiment it by accelerating the frames a little bit and the results we reduced and the time taken is less compared to other video. The images of this procedure are given below. In this type of data collection procedure the video is running in a horizontal direction and the spot are moving from region 6 to region 4 and vise versa

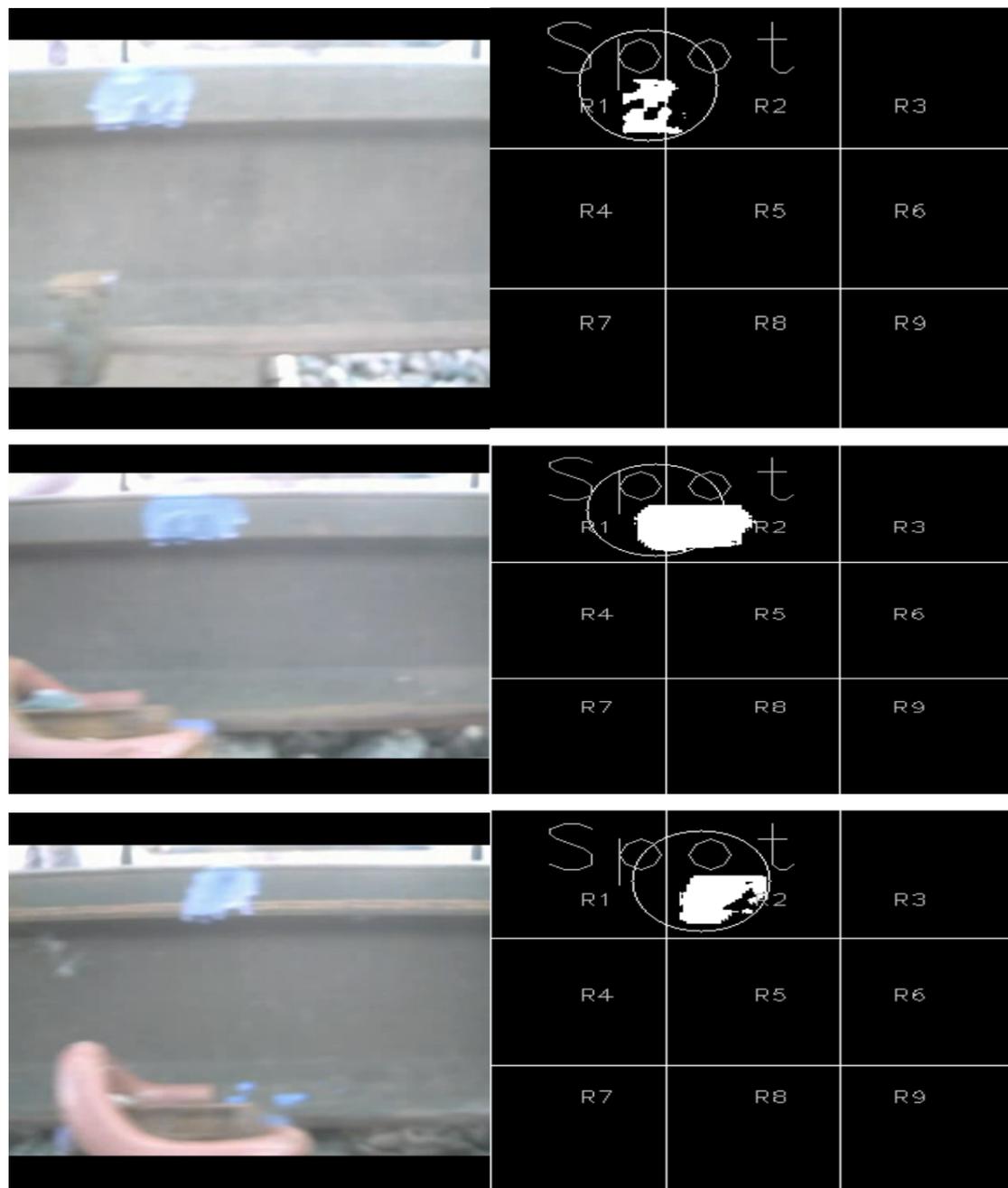


Figure 28: Sample images of TestData4

```

E:\asif data\VideoThesis\Debug>VideoThesis.exe TestData4.avi
width: 300, height: 300, channels: 3, origin: 1
Spot->1 in R2 Spot->2 in R2 Spot->3 in R2 Spot->3 in R1 Spot->4 in R2 Spo
t->4 in R1 Spot->5 in R2 Spot->5 in R1 Spot->6 in R2 Spot->6 in R1 Spot->7
in R2 Spot->7 in R1 Spot->8 in R2 Spot->9 in R2 Spot->9 in R1 Spot->10 in R
2 Spot->10 in R1 Spot->11 in R1 Spot->12 in R2 Spot->12 in R1 Spot->13 in R
2 Spot->13 in R1 Spot->14 in R2 Spot->14 in R1 Spot->15 in R2 Spot->15 in R
1 Spot->16 in R2 Spot->16 in R1 Spot->17 in R1 Spot->18 in R2 Spot->19 in R
2 Spot->19 in R1 Spot->20 in R2 Spot->20 in R1 Spot->21 in R2 Spot->21 in R
1 Spot->22 in R2 Spot->23 in R1 Spot->23 in R2 Spot->24 in R2 Spot->25 in R
1 Spot->25 in R2 Spot->26 in R2 Spot->26 in R1 Spot->27 in R1 Spot->27 in R
2 Spot->28 in R2 Spot->28 in R1 Spot->29 in R1 Spot->29 in R2 Spot->30 in R
2 Spot->30 in R1 Spot->31 in R1 Spot->31 in R2 Spot->32 in R2 Spot->33 in R
2 Spot->33 in R1
Total Time Taken In Sec = 64
No of Frames = 835
No of Frames per second = 13.0469
No of Spots: 33

```

Figure 29: Generated Results of TestData4

```

E:\asif data\VideoThesis\Debug>
E:\asif data\VideoThesis\Debug>VideoThesis.exe TestData4.avi
width: 300, height: 300, channels: 3, origin: 1
Spot->1 in R2 Spot->2 in R1 Spot->2 in R2 Spot->3 in R2 Spot->4 in R1 Spo
t->4 in R2 Spot->5 in R1 Spot->6 in R2 Spot->7 in R2 Spot->8 in R2 Spot->8
in R1 Spot->9 in R1 Spot->10 in R2 Spot->11 in R1 Spot->12 in R1 Spot->13 i
n R1 Spot->13 in R2 Spot->14 in R2 Spot->15 in R1 Spot->16 in R2 Spot->17 i
n R2 Spot->18 in R1 Spot->19 in R2 Spot->20 in R2 Spot->21 in R2 Spot->22 i
n R1 Spot->22 in R2 Spot->23 in R2 Spot->24 in R1 Spot->24 in R2 Spot->25 i
n R1 Spot->25 in R2 Spot->26 in R1 Spot->26 in R2 Spot->27 in R1 Spot->28 i
n R1 Spot->28 in R2 Spot->29 in R2
Total Time Taken In Sec = 14
No of Frames = 835
No of Frames per second = 59.6429
No of Spots: 29

```

Figure 30: Generated Results after acceleration of TestData4

The above result generation is important in this work as mentioned earlier; it works fine for clear track rather than full of vegetation and snow. In this procedure we have removed every thing from the vehicle which causes sun rays to pass, due to this noise one of the above data doesn't work properly where the segmentation is 50% and for other it was around 97%. This shows problems arising in real time for this kind of work. This procedure will generate the results same as it is in previous test samples, but here the objects (spots) are moving in horizontal direction. As you can see in the results the spot is moving from region 4 to region 6 or vice versa. The generated results are given above

Discussion

The critical discussion of this research work, where the segmentation is done to identify the spots lying on the wooden railway sleeper, to overcome the difficulties occurred by human operators and completes the task in a limited amount of time as compared to humans.

Automation of this work can be achieved with speed and efficiency, the time difference between acceleration and normal one is almost near to quarter of the normal one. The segmentation results are around 85% to 95%. It varies for different data acquisition procedures.

The results achieved in this research are in worst case because in the whole process we have used less quality equipment and many objectives such as (camera, more noise, reflections etc). The motive behind this was, to know real time difficulties can be expected in future and how we can avoid those things now. If we get good results in this case then we can improve even more with latest versions.

The generation of result gives an statistical report of individual objects, where maintenance operators can use it as a backup to what has been done till now.

The program will work only for this particular data, might work for different data also but results will not be the same. Due to threshold value used in segmentation of the object works for this particular data. By tuning this value we can use this procedure with different data as well in the same domain.

It can be transferable to any domain in image processing where the segmentation of the object is done, while certain tuning in program is required to get output.

Conclusion and Future Work

The main objective of this project is to help maintenance operators in identifying bad sleepers which are marked, automatically through an intelligent vehicle. For automating the re-identification process in condition monitoring of wooden railway sleeper inspection problem, a real time color segmentation approach has been examined in this work. The use of image processing techniques as a mean of segmenting the objects lying on wooden railway sleepers was discussed. Working with real time video were consider suitable since the problem with static images were more and speed is not sufficient for this type problem. The unexpected environment near wooden railway sleepers where images or videos can't be taken without any noise in it, but with our methodology we segment the object and improve the speed more effectively compare to previous one. The different data were analyzed and results show that we have achieved near 95% for video used as it is, of speed around (10 to 12 Km/h) with acceleration we improved the time but loss some efficiency and segmentation results were reduced to 85%. As these results are not constant, it depends on data acquisition procedures.

The research work here is done to improve the speed of maintenance operators while inspecting bad sleepers. The above results show that if 15 fps (frames per second) gives 90% to 95% results and 58 fps gives near 85% to 90% depending up data used. If we consider the time we may compromise with accuracy, if we want accuracy compromise with time and vise versa. We can improve the results even more, with 58 fps by improving the data acquisition procedure and using some new equipment. Like camera, equal distribution of light, avoiding sum light etc. Through this work we can save lot of time for maintenance operators.

The future work depends on the manual tuning used in this work, like thresholding used for object segmentation. If we used automatic thresholding it can be implemented with any set of data. To avoid the noise what we get in this acquisitions, use the box of TestData1 and camera angle of TestData3 and TestData4 with equal distribution of light inside the box. With this the noises caused previously can be avoided.

Appendix

Appendix A: Opencv Program List

Function Name	Functionality
cvLoadImage()	To read an image from a file
cvCaptureFromAvi()	Initializing capture from a .avi file
cvCaptureFromCam()	Initializing capture from a camera
cvCreateImage()	To allocate an image
cvReleaseImage()	To release an image
cvCloneImage()	To make a copy of an image
cvQueryFrame()	To get correct capture properties and each video file is divided into series of frames
cvZero()	To make an temporary image with all zeroes
cvCvtColor()	To convert from one form to another
CV_BGR2HSV	To convert from RGB image to HSV image
cvGetAt()	To access the pixel values of an image
cvSet2D()	To set the pixel values of an image
cvNamedWindow()	To create an window with the name
cvShowImage()	To display an image on particular window
cvCircle()	To draw an circle on image
cvLine()	To draw an line on image

Appendix B. Main Program Usage

The main program needs the input any video file or live video, if it is video file it must end with .avi extension and an output file can also be created with the same extension where we can see only the required object of interest. The final result would shown in the window immediately after the processing the input, it will be saved automatically.

In this program few parameters have to be tuned manually for the improvement of results.

Variables	Default Value	Comments
Threshold value	0,18(range between 0 and 1)	The default value used to segmenting an object for a required color
Drawing Circle	24 Pixels	At least 24 pixels spot have to displayed, to draw a circle
Frame Resize	300 x 300	Size of the each frame
Regions	9	Default regions of each frame

References

- [1]. Siril Yella, "Pattern Recognition For Automating Condition Monitoring Of Wooden Railway Sleepers" PhD Thesis, Napier University 2008.
- [2] David GAVILAN. *Mobile Image Categorization and Retrieval using Color Blobs*, Department of Computer Science Tokyo Institute of Technology Tokyo, February 2005.
- [3]. Enrique J. Carmona, Javier Martinez-Cantos, Jose Mira "A new video segmentation method of moving object based on blob-level knowledge" Department of Artificial Intelligence, ETSI Informatic, UNED, Juan del Rosal 16, 280400, Madrid, Spain, October 2007.
- [4]. P.Pačlik, The automatical classification of road signs , MSc. thesis, Faculty of Transportation Sciences, Czech Technical University, Prague, 1998.
- [5] The Intelligent Vehicle Initiative - Advancing Human Centered Smart Vehicles, Cheryl Little, on-line:<http://www.tfrc.gov/pubrds/pr97-10/p18.htm>
- [6] *Image Processing in ITS*, Masayoki Aoki, in proceedings of IEEE International Conference on Intelligent Vehicles, 1998
<http://euler.fd.cvut.cz/research/rs2/files/skoda-rs-survey.html>
- [7] Rita Cucchiara , Andrea Prati, Roberto Vezzani, Real Time Motion Segmentation from Moving Cameras, University of Modena and Reggio Emilia, Modena, Italy, March 2004.
- [8] Marc Gelgon, Patrick Bouthemy, A Region-Label Motion Based Graph Representation and Labeling for Tracking a Spatial Image Partition, IRISA /INIRA, Campus Universitaire de beaulieu, France , Nokia Research Center, Tampere, Finland, March 99.

- [9] Isaac Cohen, Gerard Medioni, University of Southern California, Institute for Robotics and Intelligent Systems, Los Angeles, June 99.
- [10] Masayuki Yokoyama, Tomaso Poggio, A contour-Based Moving object Detection and Tracking, Advance LSI Design & Development Division, Tokyo, Japan, Computer Science and Artificial Intelligence Lab, Massachusetts Institute of Technology, Cambridge, 2005.
- [11]. Malik Arman Morshidi¹, Mohammad Hamiruce Marhaban², Adznan Jantan¹, Color Segmentation using Multi layer NN and HSV color Space, 1.Department of computer and Communication Engineering, 2.Department of Electronic Engineering, Kula Lumpur Malaysia.
- [12] C. Doignon, P. Graebling, M. Real-time segmentation of surgical instruments inside the abdominal cavity using a joint hue saturation color feature. de Mathelin Laboratoire des Sciences de l'Image, de l'Informatique et de la Télé-détection (UMR ULP-CNRS 7005), Control, Vision and Robotics Group, University of Strasbourg, Bd. Brant, 67400 Illkirch, France. August 2005.
- [13] Liu J, Yang Y-H. Multi resolution color image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 1994; 16(7):689-700.
- [14] Computer Vision, Wikipedia.
- [15] D. Szaflarski. How We See: The First Steps of Human Vision.
http://www.accessexcellence.org/AE/AEC/CC/vision_background.html.
- [16]. K. Vaghani and R. Green. Real-time Motion Capture in Public Spaces Using Stereo vision. Image and vision computing, New Zealand (IVCNZ). 21-23 November, Akaroa 2004.
- [17] Christopher O. Jayne's, Seeing is Believing: Computer Vision and Artificial intelligence.
- [18]. D. Forysth and J.Ponce. Computer Vision, A Modern Approach, 2003.
- [19]. Ledley R, Chen G, Pollock D, Rosenfeld A. Pictorial Pattern Recognition. Washington DC: Thompson Book Company; 1968.
- [20] H. Joel Trussell, Eli Saber, and Michael Vrhel Color Image Processing, Basic and Special Issue Overview.
- [21]. Mohammed Benallal, Graduate school of Ecole des Mines de Paris, Jean Meunier Department of computer science, University of Montreal, Real Time Color Segmentation of Road Signs.

- [22] Yoshi Ohno, Optical Technology Division National Institute of Standards and Technology, Gaithersburg, MD 20899 USA.
- [23] ISO/CIE10526-1991, CIE standard colorimetric illuminants (1991).
- [24] ISO/CIE10527-1991, CIE standard colorimetric observers (1991).
- [25] For example, G. Wyszecki, and W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*, John Wiley and Sons, Inc., New York (1982).
- [26] Adrian Ford http://www5.informatik.tu-muenchen.de/lehre/vorlesungen/graphik/info/csc/COL_1.htm#topic0
- [27]. R. Jain R. Kasturi and B. Schunck. *Machine Vision*, pages: 112-138, 1995.
- [28] 1.Fabio Remondino, 2.Clive Fraser, *Digital Camera Calibration Methods: Considerations And Comparisons*, (1)Institute of Geodesy and Photogrammetry, ETH Zurich, Switzerland, (2)Department of Geomatics, University of Melbourne, Australia.
- [29] How Digital Cameras Work. <http://electronics.howstuffworks.com/digitalcamera.html>
- [30]. F. Zhang, Edwin R. Hancock. Graph Spectral Image smoothing using the heat kernel. Department of computer science, University of York, York YO10 5DD, UK
- [31]. Rafael C. Gonzalez, Richard E. Woods, *Digital image processing*, Second Edition.
- [32]. Cheng Jin, Chun Chen, Jiajun Bu, A new approach of blind image restoration. College of Computer Science, Zhejiang University Hangzhou, P.R.China.
- [33]. Gady Agam, *Introduction to Programming with OpenCV*, Department of Computer Science.
- [35] Hasan Fleyeh, *Computer Vision Lecture Notes 2008*.