
This item was submitted to [Loughborough's Research Repository](#) by the author.
Items in Figshare are protected by copyright, with all rights reserved, unless otherwise indicated.

Object detection, recognition and classification using computer vision and artificial intelligence approaches

PLEASE CITE THE PUBLISHED VERSION

PUBLISHER

© Gharib Ismail Al Matroushi

PUBLISHER STATEMENT

This work is made available according to the conditions of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) licence. Full details of this licence are available at: <https://creativecommons.org/licenses/by-nc-nd/4.0/>

LICENCE

CC BY-NC-ND 4.0

REPOSITORY RECORD

Matroushi, Gharib I. Al. 2019. "Object Detection, Recognition and Classification Using Computer Vision and Artificial Intelligence Approaches". figshare. <https://hdl.handle.net/2134/33725>.

Object Detection, Recognition and Classification using
Computer Vision and Artificial Intelligence Approaches

By

Gharib Ismail Al Matroushi

A Doctoral Thesis

Submitted in partial fulfilment

of the requirements for the award of

Doctor of Philosophy

of

Loughborough University

March 2018

© by Gharib Ismail Al Matroushi

Abstract

Object detection and recognition has been used extensively in recent years to solve numerous challenges in different fields. Due to the vital roles they play, object detection and recognition has enabled quantum leaps in many industry fields by helping to overcome some serious challenges and obstacles. For example, worldwide security concerns have drawn the attention and stimulated the use of highly intelligent computer vision technology to provide security in different environments and in diverse terrains. In addition, some wildlife is at present exposed to danger and extinction worldwide. Therefore, early detection and recognition of potential threats to wildlife have become essential and timely. The extent of using computer vision and artificial intelligence to convert the seemingly insecure world to a more secure one has been widely accepted. Such technologies are used in monitoring, tracking, organising, analysing objects in a scene and for a number of other countless purposes.

With the goal of further expanding the application domains of computer vision and artificial intelligence, a novel algorithm to classify mixed industrial airborne waste particles is developed under the research context of this thesis. Initially, a dataset of industrial airborne particles was collected and imaged for this purpose. All images were captured using an advanced microscopic camera. The particles are segmented from the background image using K-means clustering and Hue, Saturation and Value (HSV) colour space features. Subsequently, the Hue, Saturation and Value (HSV) colour histogram data and Gray Level Co-occurrence Matrix (GLCM) texture features are combined into one feature vector of size 278 features, for use in dust sample classification. A feature selection algorithm is employed to minimise the number of features to ten, without compromising the accuracy that can be obtained when using all features. This significant minimisation of features used in classification would improve the processing time.

A further computer vision system proposed in this thesis is for automated road surface type recognition aimed at use within autonomous vehicles. Traditional feature driven machine learning based algorithm and a deep learning based approach are proposed to recognise 5 different road surfaces namely: asphalt, concrete, rough, blocks and sand. Firstly, a novel algorithm is developed to exclude irrelevant objects and to create the region of interest within a snapshot image of the road surface. This segmentation algorithm eliminates unnecessary objects being considered for processing as the road in which the vehicle is being driven, by dividing the input image into top and bottom parts, to attempt a fast, region-dependent segmentation. Subsequently, the algorithm uses region-growing to create region of interests. For the recognition task, a feature based algorithm that uses a combination of colour histogram, Gray Level Co-occurrence Matrix (GLCM), Local Binary Pattern

(LBP) and Gabor feature is proposed. The experiments conducted demonstrate the challenges faced by this algorithm in its practical use to classify the road surfaces. Resolving the shortcoming of this approach a state-of-the-art algorithm is subsequently proposed, as an alternative approach, to recognise road surfaces. In this approach, the Convolutional Neural Network AlexNet is used as a pre-trained network. This algorithm shows significant positive results as compared to other benchmark algorithms that are able to classify almost 90% of surfaces examined.

Finally a detection and recognition algorithms for Camel heads is proposed in this thesis, via feature based and deep learning based approaches. A well-known algorithm, Viola Jones, is fine-tuned and used as the feature based algorithm to detect the heads of camels. The detection was based on Haar-like, Histogram of Oriented Gradients (HOG) and Local Binary Pattern (LBP) features. The results show that Haar features obtained better classification accuracy than other features. However, HOG features are shown to provide better results when training time is considered. In the second approach, a unique deep learning based algorithm is proposed in this thesis for the detection of camel heads. For training purposes, a new Convolutional Neural Network (CNN) is created and trained from scratch. The algorithm uses Convolutional Neural Network (CNN) to extract image features and create a Region Proposal Network (RPN) with many proposed anchors. A None-Max Suppression (NMS) is applied to eliminate unnecessary anchors within the proposed regions. Then, Region Of Interest (ROI) pooling technique is used to create fixed size feature maps for each proposal region, regardless of the size of input. Due to the need to train all system components in one run, end-to-end training is adopted for better results. The experimental results prove that the proposed method was able to detect most of the camels' heads in the test dataset with a recall percentage of over 92% compared with the 65% recall percentage obtained for a similar benchmark algorithm.

Acknowledgment

In the name of God, the Most Gracious, the Most Merciful.

Right from the start, I would like to begin my chain of gratitude to the God Almighty. I praise him for all the blessings that have always been with me and for given me heath, strength, tolerant and being able to complete this work within the limited time frame.

I dedicated this work to my dear parents who deserve special thanks and mention for all sacrifices they have made. My mother, Aisha Al-Ameri, is the one who sincerely raised me up with extra care and love since I was a child and to date. My father, Ismail Al-Matroushi, was the one who draw the learning road map for me and was always proud of me and my achievements at different stages of my life, may God Almighty rest his soul in peace.

My heartiest thanks go to my beloved wife, Fatma, who has always encouraged me and prayed for my success. Beside her study and busy time, she fulfilled her other family duties with uninterrupted patience and exceptional stamina during these years. My heartiest thanks also go to my sons, Almohannad and Alyamman. They were born during my PhD studies and made this period a wonderful and unforgettable time. My special thanks to my brothers, sisters, father-in-law, mother-in-law, sisters-in-law, brothers-in-law and all members of my immediate and extended family. Thank you for your boundless support during my stressful time.

I shall express my deepest appreciation to my supervisor, Prof. Eran Edirisinghe for his outstanding guidance and endless support. His inspiration, valuable comments, editorial advice, unquestioning faith in my capabilities and patience during my study journey have helped me realise my ultimate research goals and has shaped this thesis into its final form.

I would like to thank my annual reviewer, Dr. Qinggang Meng, for his valuable recommendations on my annual progress reports. To all staff, my PHD colleagues and friends in computer science department at Loughborough University, thank you for your kind help, advice and support.

It is my privilege to extend my gratitude to all of the Omani Student Society members of Loughborough University for placing their trust and confidence in my ability to lead the society during the last three years of my study. Their co-operation was the key behind the great success of the society.

I would also like to thank my sponsor, Ministry of Manpower – Sultanate of Oman, for giving me the opportunity to acquire new knowledges and address new global challenges through pursuing a PHD in one of the UK's leading universities, Loughborough University.

Gharib Ismail Al Matroushi

10th March, 2018

Table of Contents

Abstract	II
Acknowledgment	IV
Table of Figures	IX
List of Tables	XI
List of Abbreviations	XII
Chapter 1	1
Introduction.....	1
1.1 Research Motivation	3
1.2 Aim and Objectives.....	6
1.3 Contributions of Research.....	7
1.4 Thesis Organisation.....	8
Chapter 2	9
Literature Review.....	9
2.1 Introduction.....	9
2.2 Airborne Dust Particle Classification Algorithms	9
2.2.1 Colour Based Image Segmentation Approaches.....	9
2.2.2 Shape Based dust particles Segmentation	12
2.2.3 Size based dust particles classification	13
2.3 Road Surface Recognition.....	16
2.4 Animal Detection and Recognition.....	18
2.4.1 Animal Object Segmentation	18
2.4.2 Feature Extraction	21
2.4.3 Animal Object Classification	23
Chapter 3	24
Research Background	24
3.1 K-Means Clustering	24
3.2 Colour Spaces	25
3.2.1 Converting from the RGB colour space to other colour spaces	25
3.3 Feature Descriptors	29
3.3.1 HOG Features	29
3.3.2 LBP Features.....	30
3.4 Machine Learning	31
3.4.1 Support Vector Machine	31
3.4.2 Boosted Cascade Object Detection using Viola-Jones algorithm	33
3.5 Deep Learning and Convolutional Neural Network	37
3.5.1 AlexNet	38

3.5.2	ROI Pooling Layer	39
3.5.3	Max Pooling	40
3.5.4	Anchors	41
Chapter 4	43
Airborne Industrial Dust Particle Classification.	43
4.1	Introduction.....	43
4.2	Background	44
4.3	The Proposed System.....	46
4.3.1	Dust sample collection and image acquisition	47
4.3.2	Segmentation.....	47
4.3.3	Post Processing	57
4.3.4	Feature Extraction	60
4.3.5	Feature Selection.....	61
4.4	Experimental Setup.....	61
4.5	Experimental Result and Analysis	63
4.6	Chapter Conclusion.....	71
Chapter 5	72
Road Surface Type Recognition for Autonomous Vehicles		72
5.1	Introduction.....	72
5.2	Method-1: Colour and Texture Feature Based Classification Approach	73
5.2.1	The Test Dataset.....	73
5.2.2	Road Surface Segmentation	74
5.2.3	Feature Extraction	78
5.2.4	Result and Analysis.....	79
5.2.5	Conclusions.....	81
5.3	Method 2: Deep Learning Based Approach.....	83
5.3.1	Methodology	83
5.3.2	Road Surface Images.....	84
5.3.3	Road Segmentation Process	84
5.3.4	Pre-Processing.....	84
5.3.5	Feature Extraction Process.....	84
5.3.6	Analysis of Results.....	86
5.3.7	Conclusion	88
5.4	Chapter Conclusion.....	89
Chapter 6	91
Camel Head Detection and Recognition		91
6.1	Introduction.....	91

6.2	The Proposed Method	93
6.2.1	Feature-Based Approach	93
6.2.2	Deep Learning Approach	105
6.3	Chapter Conclusion.....	115
Chapter 7	116
Conclusion and Future Work	116
7.1	Conclusion	117
7.2	Future Work	119
References	121
Appendix A	133
Appendix B	134

Table of Figures

Figure 2-1 Comparison of proposed method with different colour space [8].	11
Figure 2-2 Flowchart of the seed selection [9].	11
Figure 2-3 Particle dimension measurement using Feret's diameter and pixel-march methods [13].	13
Figure 2-4 Process flow in the developed machine vision plugin [13].	14
Figure 2-5 Non-touching and touching particles [15].	15
Figure 3-1 The CIE LAB colour space [18].	26
Figure 3-2 Single-hexcone model of HSV colour space [103].	26
Figure 3-3 Gradient direction.	29
Figure 3-4 Example of gradient calculation.	29
Figure 3-5 Example of 3X3 pixel block for 8 pixel neighbourhoods.	30
Figure 3-6 Optimal classification of 2 classes.	32
Figure 3-7 SVM Multi-class classification.	32
Figure 3-8 Computation of rectangular features.	33
Figure 3-9 Example of Haar-like rectangular areas to detect human body [108].	34
Figure 3-10 Stages of the cascade classifier.	35
Figure 3-11 Viola-Jones detector summery [110].	36
Figure 3-12 Deep learning pipeline [116].	37
Figure 3-13 AlexNet architecture [112].	39
Figure 3-14 ROI Pooling Illustration.	39
Figure 3-15 Single ROI with size of 4x4.	40
Figure 3-16 Fixed size ROI 2x2.	40
Figure 3-17 Example of Max pooling operation [114].	41
Figure 3-18 Anchor sizes and ratios [115].	42
Figure 4-1 The proposed system.	46
Figure 4-2 Histogram for low density image.	48
Figure 4-3 Histogram for medium density image.	49
Figure 4-4 Histogram for high density image.	49
Figure 4-5 image segmentation steps using K-means clustering.	51
Figure 4-6 RGB input image.	51
Figure 4-7 Black particles. Figure 4-8 Light brown particles. Figure 4-9 Brown particles.	52
Figure 4-10 Low density original image. Figure 4-11 Low density segmented image - black objects.	53
Figure 4-12 Low density segmented image - brown objects. Figure 4-13 Low density segmented image – others.	53
Figure 4-14 Medium density original image. Figure 4-15 Medium density segmented image - black objects.	54
Figure 4-16 Medium density segmented image-brown objects. Figure 4-17 Medium density segmented image - others.	54
Figure 4-18 High density original image. Figure 4-19 High density Segmented Image - black objects.	55
Figure 4-20 High density segmented image - brown objects. Figure 4-21 High density segmented image - others.	55
Figure 4-22 High density original image. Figure 4-23 High density Extra Colour A.	56
Figure 4-24 High density original image B. Figure 4-25 High density Extra Colour C.	57
Figure 4-26 High Density - black object.	58
Figure 4-27 Medium Density - black object.	58
Figure 4-28 Low density - black object.	59

Figure 4-29 Example of image category.....	59
Figure 4-30 Example of image patches.....	60
Figure 4-31 Classifiers / feature selection algorithms, comparison model.	62
Figure 4-32 Accuracy summary for all features.	65
Figure 4-33 ROC curve for best classifiers using all features.....	67
Figure 4-34 ROC curve for classifiers using selected features.	69
Figure 4-35 AUC percentage per class.	70
Figure 5-1 Method 1 – the feature based approach for road surface type recognition.....	73
Figure 5-2 Input images splitting process.	75
Figure 5-3 Region growing segmentation.....	76
Figure 5-4 Segmented image and result of morphological operations.....	77
Figure 5-5 Example of selected region of interest.	77
Figure 5-6 Comparison of different algorithms using colour and GLCM dataset.	80
Figure 5-7 Best accuracy obtained from different feature set combinations.	80
Figure 5-8 The proposed, deep learning based approach to road surface type recognition.	83
Figure 5-9 AlexNet Architecture.	85
Figure 5-10 Concreate road images.	87
Figure 5-11 Comparison of accuracy obtained by different method.	88
Figure 6-1 Overview of the proposed method.	93
Figure 6-2 Feature based approach of the proposed method.	94
Figure 6-3 Stages of training the cascade object detector.....	96
Figure 6-4 Misclassifications due to ambiguity in texture.....	97
Figure 6-5 Misclassifications due to ambiguity of shape.....	99
Figure 6-6 Misclassifications due to head occlusions.....	99
Figure 6-7 Frontal head detection at stage 20 for different features.	100
Figure 6-8 Examples of camel head counting.....	101
Figure 6-9 Haar-like features detection rates.	101
Figure 6-10 HOG features based detection rates.	102
Figure 6-11 LBP features based detection rates.....	102
Figure 6-12 Detection rate comparison between Haar-like, HOG and LBP.....	103
Figure 6-13 Overview of Faster R-CNN architecture.....	105
Figure 6-14 Convolutional Neural Network (CNN) Layers.	106
Figure 6-15 Example of initial features extracted from first convolutional layer.....	106
Figure 6-16 Region Proposal Network (RPN) pipeline.	107
Figure 6-17 Cross boundary anchor.....	108
Figure 6-18 Example of IoU.	108
Figure 6-19 Example of None-Max Suppression.....	109
Figure 6-20 Fast R-CNN Detector architecture.	110
Figure 6-21 Detection camel head using Faster R-CNN (the proposed method).	111
Figure 6-22 Detection camel head using Fast R-CNN.....	111
Figure 6-23 Proposed method detection rate.	112
Figure 6-24 Fast R-CNN Detection Rate.	113
Figure 6-25 Performance analysis plot on camel detection.	114
Figure 95 Sample of Rod Surfaces.	134
Figure 96 Example of Camel Database.....	135

List of Tables

Table 3-1 Colours Conversion Values from RGB to HSV.	28
Table 4-1 Model classification accuracy,.....	62
Table 4-2 RandomForest classifier parameters.	63
Table 4-3 Texture based dataset accuracy.	64
Table 4-4 Texture based dataset detailed accuracy for LogitBoost classifier.	64
Table 4-5 Colour based dataset accuracy.	64
Table 4-6 Colour and textur dataset accuracy for all features.	65
Table 4-7 Colour and texture dataset, detailed accuracy for RandomForest classifier using all features.	66
Table 4-8 RandomForest confusion matrix using all features.	66
Table 4-9 Colour and texture dataset accuracy for selected features.	67
Table 4-10 CFS selected Features.	68
Table 4-11 Wrapper approach selected Features.	69
Table 4-12 Classifiers AUC percentage.	70
Table 5-1 Best classifier performance per feature dataset.	81
Table 5-2 Confusion Matrex of unsegmented images.	86
Table 5-3 Accuracy obtained using unsegmented images dataset.	86
Table 5-4 Confusion matrix of the proposed method.	87
Table 5-5 Accuracy obtained using the proposed method.	87
Table 6-1 Computation time at stage 10 for camel head detection using Haar, HOG and LBP features.	104
Table 6-2 TPR, FNR, Precision, Recall and F.Measure performance values.	113

List of Abbreviations

2D - Two Dimensional
3D – Tree Dimensional
ADAS - Advanced Driver Assistance Systems
AUC - Area Under Curve
CFS - Correlation-based Feature Selection
CNN - Conventional Neural Network
CVAAS - Camel-Vehicle Accident Avoidance System
END - Ensemble of Nested Dichotomies
ERC - Environmental Research Centre
FC – Fully Connected
FN – False Negative
FNR - False Negative Rate
FP – False Positive
Ghz – Gigahertz
GLCM - Gray Level Co-occurrence Matrix
GPRS - General Packet Radio Service
GPS - Global Positioning System
GPU - Graphics Processing Unit
GT – Ground Truth
HD - High Definition
HOG - Histogram of Oriented Gradients
HSI - Hue Saturation Intensity
HSV - Hue, Saturation and Value
IoU - Intersection over Union
KHz - Kilohertz
KNN - K-Nearest Neighbour
LBP - Local Binary Pattern
MECA - Ministry of Environment and Climate Affairs
MoU - Memorandum of Understanding

NMS - None-Max Suppression
OAPS - Oman's Air Pollution Standards
PCA - Principal Component Analysis
PM₁₀ - 10 micrometers in diameter
RAM – Random Access Memory
R-CNN – Regional Convolutional Neural Network
ReLU – Rectified Linear Unit
RGB – Red, Green, Blue of colour channels
ROC - Receiver Operating Characteristics
ROI – Region Of Interest
RPN - Region Proposal Network
SMO - Sequential Minimal Optimization
SMS - Short Message Service
SVM - Support Vector Machine
TN – True Negative
TP – True Positive
TPR - True Positive Rate
TRC - The Research Council of Oman
WHO - World Health Organization

Chapter 1

Introduction

Object detection and recognition has been used to solve different problems in many disciplines and has hence become a vital research area in recent years. A number of technologies, i.e. pattern recognition, computer vision and machine learning, have been integrated to detect, recognise and classify objects. Such technologies have been deployed widely in many application domains, i.e. Surveillance and security, healthcare, automated unmanned vehicles, robotics, smart phones, transportation and in many manufacturing industries, to name a few. Generally, these intelligent technologies can be implemented, trained and tuned to have the capability to automate various tasks, with minimum or no human involvement, in most sectors. However, this automation will open the door for numerous challenges that need to be addressed to meet high level of expected accuracy while pushing the boundaries towards human like capabilities.

Usually objects of interest can be identified in a digital image or a video by enabling machine learning algorithms to analyse different object features, i.e. colour, texture and shape, with the intention to compare a new image with existing ones based on the same features. There are a number of key steps that are carried-out in the process of object recognition, which are image/video capturing, pre-processing, object detection and segmentation, and feature extraction, feature selection/reduction and finally feature based object recognition. The quality of input digital images play an important role in the overall accuracy obtained within the recognition steps. Interestingly, high quality images have become a de-facto assumption in most of new low-cost digital cameras as they support high resolution and have the capability to record High Definition (HD) images and videos. Once a good image is captured, an image processing algorithms such as de-blurring and morphological algorithms could be applied as pre-processing steps to remove any noise and enhance image presentation and quality. Next, computer vision algorithms can be used to detect motion, objects or/and to extract foreground from the background. In the final step, the machine learning algorithms can be used to extract various object related features. To speed up the processing time, feature selection algorithms should be used to eliminate the redundant features and select the most significant ones. The selected features will be used to train a machine learning algorithm such as Support Vector Machine (SVM), to identify the object type based on previous learning.

Despite the extensive implementation of computer vision and artificial intelligence algorithms to solve multifaceted issues in many areas, the literature review conducted showed that existing

algorithms still have some limitations to address. These limitations can be covered under further research to develop new algorithms or improvement of existing algorithms.

Machine Learning using Deep Convolutional Neural Networks (CNNs) named more often as ‘Deep Learning’ is another recent approach that has received significant attention of the computer science research community in recent years. The reason behind this is, its capability to achieve significant object detection and recognition accuracy results that were not possible before. Deep Learning pushes the boundaries of object detection and recognition to reach the human capability limits. Numerous deep learning algorithms have already been proposed, developed, implemented and embedded in many applications. For instance, in driverless vehicles an application would be to detect lanes, road signs, pedestrians etc. Deep learning based algorithms have the ability to learn and recognise the objects from raw data through series of layers of neural processing. In the first convolutional layer, the basic object features in the image or video i.e. edges, are extracted and used to train the layer whereas complex features are extracted in subsequent layers. The processing values of previous layers are shared with the next layers of the network until the system recognises the object. The accuracy of this approach depends on the amount of data that has been used to train the network, which needs a computer with Graphics Processing Unit (GPU) processing capabilities to speed up the processing. Fortunately, many large databases i.e. ImageNet, LSUN, MS COCO and Google’s Open Images, have been made publicly available to the researchers for recognition, segmentation, detection and other tasks.

Regardless of the availability of large number of databases of images, some limitations exist about their choice as each database has been made for a specific purpose. Lack of a large number of object types, for instance, in these databases, makes recognising different objects a challenging task. However Deep Learning based approaches have relatively recently been introduced and have already been heavily used to solve numerous challenges in different application domains such as autonomous vehicles applications and animal detection and recognition.

1.1 Research Motivation

Air pollution can occur due to various types of industry generated dust and remains a major societal problem in many countries where environmental regulations are not well defined and closely monitored. For example during a peaceful protest of more than 200 citizens in Sohar city, Oman, in 2012, it was claimed that “industry air pollution” has taken a severe toll on the health of residents of the city evidenced by an increased occurrence of cancer and other respiratory and skin diseases. However no scientific evidence gathering, investigation or research has been undertaken to classify the airborne dust types and subsequently to trace them back to their sources. If this was not the case the government and regulatory bodies would have been in a position to take corrective measures.

Computer vision and machine learning algorithms have been adapted in many application areas such as medical, agriculture, mining, security etc. However, the literature review conducted within the research context of this thesis indicated that sufficient effort has not been put into addressing the problem of airborne dust particle classification using computer vision based approaches. Recent research in [5], shows that SO₂ emission levels in Sohar industrial area in Oman has been investigated using digital spectrometry and the result has been compared with the regulatory limits set by the Ministry of Environment and Claimant Affairs in Oman. However airborne dust particle sizes have not yet been checked to confirm, whether or not, the regulatory limit has been exceeded by the industry operational in the area known to be causing air pollution due to industry generated dust. Although some work has been reported in literature on particle dust based air pollution monitoring [6, 7 and 8] the image capture mechanisms and processing systems used are not suitable for a detailed analysis of airborne dust particle classification. To capture the airborne dust particles not visible to the naked human eye, a state-of-the-art microscope is needed as the particles are too small to be captured by a standard camera. Moreover, the particle/object segmentation methods used in other research fields are not adequate to segment and identify airborne dust particles. Thus, a novel technique is to be developed to overcome this limitation. In the current state-of-the-art work the research focuses only on measuring the particle size and size distribution. Additionally, the dust samples considered in previous research were of single pollutant type, from visible (to the naked eye), already “known” sources, e.g. wood, mining, mineral processing, etc. However, industrial dust from mixed industrial sources, containing different particle pollutants, often not visible/noticeable to the naked eye, has not been investigated with a view of segmentation and classification to different dust types. Segmenting airborne dusts particles of mixed type based on computer vision needs a microscopic resolution camera and image capture and shape, colour, texture and size based particle type classification.

Autonomous vehicles' related applications are a more recent application domain that has recently attracted the attention of computer vision researchers. The vehicle manufacturers are attempting to make the vehicles intelligent enough to detect and sense hazards as early as possible. Thus, they have fitted the latest generation vehicles with numerous different technologies such as low energy radar, Global Positioning System (GPS), computer vision, laser lights, vehicle-to-vehicle communication etc., in their attempt to improve a vehicle's mobility, stability, safety and customer stratification. All these technologies are typically integrated together and are capable to share and process captured data to avoid obstacles, increase traffic flow and reduce road collisions. Car companies such as Mercedes, Tesla, BMW, Volvo, Audi and etc., are investing heavily in research and development to develop their driverless cars. However, departments regulating transportation and traffic safety in different countries have set out standards, regulations and legislation, in order to allow fully automated cars to be used in their respective countries with the view of ensuring the safety of passengers and the roadside infrastructure can be met efficiently. Thus, Advance Driver Assistant Systems (ADAS) have gone through extensive upgrades due to many recent new developments. The literature review conducted within the research context of this thesis revealed the need to further improve existing ADAS systems to include the capability to recognise different road surfaces, as this issue is not yet been practically and fully addressed. How a vehicle is being driven should largely depend on the road surface it is moving on and thus developing algorithm for automated road type recognition should help reduce road collisions and increase stability and safety in autonomous vehicles.

Protecting animal lives and welfare is essential and has become a priority for many countries in recent years. The Camel is one of the most expensive animals in Arab Peninsula where camel racing is of part of Arabs cultures. For instance, prize money for winning camel can be millions of dollars in a race. Therefore, stealing, killing and harming camels, especially good racing camels, have become a series problem for camels' owners. Additionally, camel-vehicle collisions are another problem faced by the owners of camels, road users and transportation and traffic safety concerned government departments in the Gulf region. Although, traditionally different sensor/tracker based technologies have been employed to track animals, camels have less chance for the adaptation of this adaptation due to difficult environments that they live-in, such as the desert environment. Vehicle manufacturers (e.g. Volvo) have recently introduced computer vision and machine learning based system to detect animals and avoid animal-vehicle collisions. However, such systems have been developed to detect few commonly found domestic and farm animals in Europe and cannot detect camels only found in the Gulf region and might have more severe impact during an accident, being a heavier animal

compared to common farm and domestic animals. Given the above this thesis focuses on the use of traditional machine learning and deep learning based approaches for camel detection.

Given all the above reasons, the research conducted within the context of this thesis has received motivation from the research gaps highlighted above leading to the following aim and objectives of the research.

1.2 Aim and Objectives

This research aims to develop robust and efficient algorithms that have the capability to classify mixed industrial airborne particles, recognise different road surfaces for autonomous vehicles at day-time and detect and recognise camel heads in different environments including desert.

Objectives

The below objectives are to be undertaken to achieve the above aim.

1. Carry out a detailed literature review in the area of airborne dust particle classification, road surface recognition and detection and recognition of camels.
2. Create test image databases for airborne dust particle classification, road surface recognition and the detection of camels.
3. Design and implement an efficient algorithm to classify mixed-type industrial airborne particles based on traditional machine learning approaches.
4. Investigate the use of differed pre-trained Convolutional Neural Network models and transfer learning approaches in object detection and recognition.
5. Investigate the possibility of designing deep learning networks and training them for specific object detection and classification tasks.
6. Design and implement feature based and deep learning based approaches to recognise five different road surfaces, namely, rough, asphalt, sandy, concrete and block paved.
7. Design and implement feature based and deep learning based approaches to detect and recognise camels.
8. Evaluate all the proposed algorithms, identify the limitations and recommend further enhancement and extension possibilities.

1.3 Contributions of Research

The research presented in this thesis, to classify airborne dust particles, road surfaces and to detect and recognise camels has resulted in a number of original contributions in object detection and classification. The main contributions are outlined below:

1. Capturing and creating a new experimental dataset for invisible, mixed type airborne dust particles. The images have been captured as RGB images using a state-of-the-art microscopic camera and using industrial airborne dust samples collected from the industrial zone of the Sohar region on Oman (see chapter 4). This dataset has been used for dust classification experiments and will be made publicly available for the benefit of the wider research community.
2. A novel algorithm to segment and classify mixed industrial airborne dust particles by combining features of colour histogram bins and the Gray Level Co-Occurrence Matrix (GLCM). Dust particles have been first segmented using K-means clustering and HSV colour space. The experiments conducted proved that accurate dust type classification can only be achieved by merging colour and texture features (see section 4.5). It is shown that after Applying feature selection (10 features), the algorithm had the capability to maintain same accuracy, as compared to the accuracy obtained by using all features (278 features).
3. Capturing and creating a new database for road surface type recognition. The captured dataset contained five different road surfaces namely, Asphalt, Block Paved, Concrete, Rough and Sandy (see chapter 5). The dataset was captured using a camera fitted into dashboard of vehicle. It has been collected from different cities in Oman and UK at day-time and under different illumination conditions: sunny, partly cloudy and cloudy. This dataset will be made publicly available for the benefit of the wider research community.
4. Design and implementation of a state-of-the-art algorithm to segment road surfaces from the frontal views of a vehicle automatically and creating a Region of Interest (ROI) for road surface analysis and type classification. This algorithm was designed based on region growing principles (see section 5.2.2.2).
5. Comprehensively evaluate the ability of traditional feature based machine algorithms to classify the road surfaces to an acceptable accuracy (see section 5.2.3). This investigation involved the use of many features such as, GLCM, Colour, LBP and Gabor, widely used for image classification in literature. Although a wide range of feature types were used to classify the road surfaces by using a support vector machine based classifier, the results demonstrated that neither individual features nor combined features can adequately classify different road

surfaces. It was revealed that this was mainly due to the significant intra-class variations observed.

6. Designing and implementing a robust and efficient algorithm for the classification of road surface types. This algorithm was developed based on transfer learning using deep learning (see section 5.3). Alex Network [112] was used as the pre-trained model and was subsequently fine-tuned to classify road surfaces to the best level of accuracy.
7. Investigate the use of the well-known cascaded classifier in the detection and recognition of Camels to a satisfactory level of accuracy (see section 6.2.1).
8. A novel algorithm to detect and recognise camels from raw data without any segmentation or pre-processing to the input images. This algorithm is based on deep learning where convolutional neural networks were used (see section 6.2.2). Despite the limited data used in this research, a new network was designed and trained with end-to-end learning capability.

The above research has led to the preparation of a number of papers that will be submitted according to the publication plan in Appendix A.

1.4 Thesis Organisation

For clarity of presentation, this thesis is organised and structured as follows: chapter 2, discusses the previous research work on particle detection and recognition in general, road surface type recognition and camel detection and recognition. Chapter 3 provides the reader with a background of algorithms and mathematical foundations of systems and processes used within the contributory chapters of this thesis. These include colour spaces, thresholding, K-means clustering and details of many different commonly used feature descriptors. It also includes a detailed explanation of Convolutional Neural Network (CNN), deep learning and some concepts that are used in deep learning and CNN. The chapter further presents the conceptual background of ROI pooling, Max Pooling and anchors. Chapter 4 presents a novel approach to segment and classify mixed, industrial airborne dust particles. Chapter 5 represents two novel approaches for road surface segmentation and recognition, one based on traditional machine learning and the other based on deep learning. Chapter 6 provides two novel approaches to detecting and recognising camels, one based on traditional machine learning and the other based on deep learning. Chapter 7 contains the thesis conclusions and suggested future work related to the proposed algorithms.

Chapter 2

Literature Review

2.1 Introduction

This chapter initially reviews existing research on industrial dust particle classification that uses machine vision, describing the algorithms proposed and analysing the experimental results presented by the authors. The chapter subsequently provides the reader with a comprehensive study on current approaches to road surface detection, segmentation and classification. Finally, the chapter proposes existing research on the detection and recognition of animals, in view of the design and implementation of the novel algorithms for the detection and recognition of camels, presented in chapter 6 of this thesis.

2.2 Airborne Dust Particle Classification Algorithms

A detailed review of existing research on airborne dust particle segmentation and classification was conducted as a part of this review of literature. This review concluded that in existing research the most common approaches used were only taking the ‘size’ of the particles into account when classification of different kinds of dust particles. Further study and investigations revealed that this is due to the fact that air pollution legislations worldwide only sets a safe legal dust based air pollution limit based on dust particle size. Colour and shape information however can provide additional queues that could enable the classification of dust particles to different dust types, allowing the authorities to trace the particles back to their sources, by further investigations. As this was the intention of the novel research to be presented in this thesis, the following sections also include general research literature on colour and shape based image segmentation algorithms proposed in the past.

2.2.1 Colour Based Image Segmentation Approaches

The study in [6], discussed an approach of colour image segmentation. In this study, the HSV colour space, K-means clustering and Otsu’s thresholding algorithms were used in the segmentation of objects. As far as input image is concerned, the most commonly captured RGB colour space images were used. Seven steps were proposed for segmenting a given image. As the HSV colour space is similar to the way the human eye perceives colour, the RGB input image was first converted into the HSV (Hue, Saturation and Value) colour space and the V (Value) channel was extracted and used in the segmentation. To achieve best image thresholding, Otsu’s multi-thresholding approach [117] was

applied on extracted images of the V channel. Since when Otsu's multi-thresholding algorithm is used, an over region segmentation may occur, the K-means clustering algorithm was used to merge the over segmented regions. Subsequently background subtraction was used followed by a final stage of morphological operations to refine the segmented regions. Authors used the Berkley segmentation image database for testing and evaluation purposes. The proposed method is compared with three different types of colour based segmentation algorithms that ensure accuracy and quality of different types of colour images. The Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE) matrices were used to obtain the experimental results. The result proved that the proposed algorithm produces better results compared to other algorithms. However as only the V channel information was used in the segmentation, the system proposed ignores an important and vital H (Hue) channel that refers to 'chrominance', which is a vital shortcoming in particular when objects cannot be separated only based on luminance.

In [7] an integration of K-Mean and Watershed algorithms were used by Bora & Gupta as a new approach to segmenting colour images. The main objective behind the integration of the two algorithms was to improve the accuracy of the results obtained by using only one algorithm, similar to the approach used in [6]. As input, a RGB image was used and converted into HSV colour space. The resulting image was used as input to a K-mean algorithm. Sobel filter was applied for the segmented image before analysis by watershed algorithm. For post processing, a median filter was used to remove any noise that could occur because of segmentation processes. The segmentation result of the proposed system was found to be satisfactory.

In recent years, there has been an increasing amount of literature on using K-Means clustering to segment the objects in colour images as in [8]. In this study, different types of RGB images were used as input images, in the experiments. Input images were first converted into different colour spaces such as L^*A^*B , RGB and HSV, before assigning as inputs to K-means clustering algorithms to segregate the colours. The proposed algorithm was tested with different colour spaces as indicated by Figure 2-1. The results showed that the L^*A^*B and YCBCR colour spaces provide the best segmentation results amongst other colour spaces evaluated in the experiments. The table in Figure 2-1, summarises the results obtained indicating an observation of the types of images for which each colour space could best be utilised for segmentation purposes.

Color Space	Execution Time	Image Segmentation	Remark
RGB	Highest execution time	Not Satisfactory	Not suitable for images with more color variants.
L*a*b	Least execution time	Best	Suitable for all kind of images,
YCbCr	Less execution time	Better	Suitable for images having distinct color components.
HSV	High execution time	Good	Suitable for images having distinct color components.

Figure 2-1 Comparison of proposed method with different colour space [8].

The authors of [9] presented a new method to segment images based on region growing and ant colony clustering. As far as the colour space is concerned, the original RGB images were first converted to the Hue Saturation Intensity (HSI) colour space. Seeds were selected automatically by using edge information and intensity similarity among pixels in the HSI colour space. Pixels with similar characteristics were rounded up to grow regions. The Figure 2-2 below illustrates the process of seed selection and region growing.

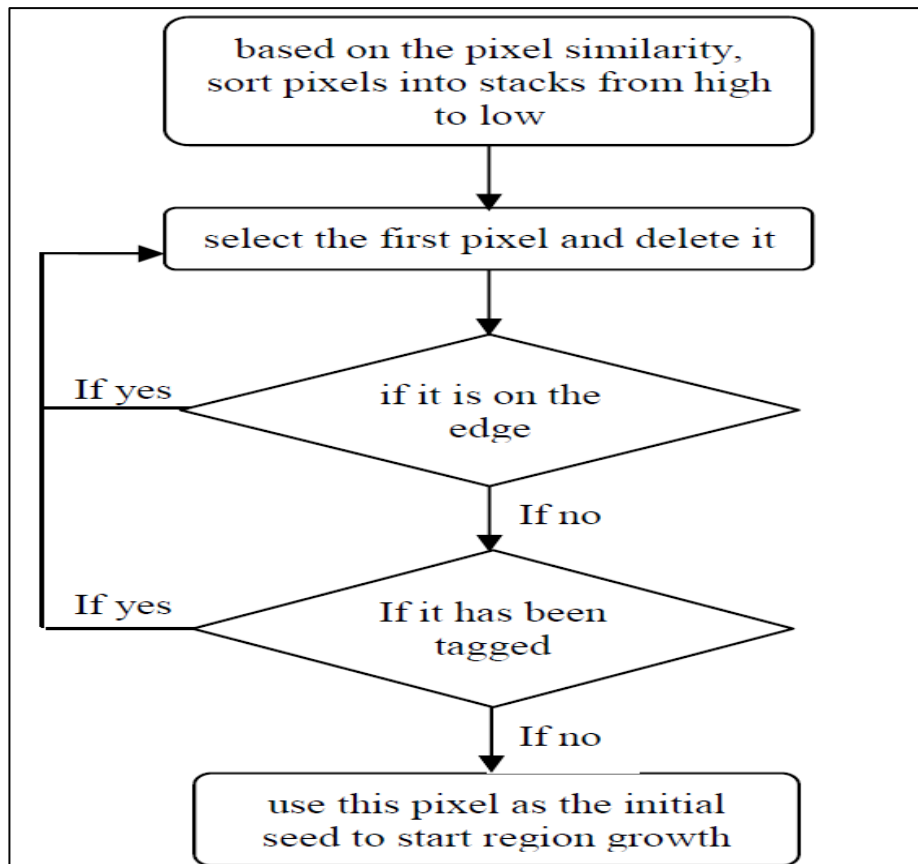


Figure 2-2 Flowchart of the seed selection [9].

The results of region growing were used as input to an ant colony clustering method [118]. Subsequently, the ant colony clustering method was applied to merge different regions of homogeneity. The experimental results showed that the proposed method is effective for colour image segmentation. This method also eliminated large quantities of cycles and calculations caused by traditional ant colony algorithms.

2.2.2 Shape Based dust particles Segmentation

The authors in [10] introduced a shape-based algorithm to classify environmental microorganisms. The proposed approach has three main phases, which are, semi-automatic segmentation, shape description, and classification. The Sobel edge detector was initially used for image segmentation as it is less sensitive to the noise, especially in low-contrast, microscope captured, images. Due to the nature of environmental microorganisms, i.e. they are colourless and transparent, which mean nearly impossible to extract their colour and texture features, the authors decided to obtain the shape feature based on optic boundary between light and shade of different objects. Four shape fetures were used: Edge Histogram Descriptor, Geometrical Feature including perimeter, area, etc, Fourier Descriptor, and Structure Histogram. High-dimensional features led the authors to use a Support Vector Machine classifer in the classification phase. For supervised training, the samples were segmented manually and semi-automatically. The results of the experiments show that the proposed method was effective in classifying environmental microorganisms.

Another shape-based approach proposed in [11] to automatically segment the nuclei in 3D microscopoc images. The authors integrated 3D watershed segmentation and region grouping methods to segment individual nuclei. The Euclidean Distance metric was used to identify the distance transform of the foreground pixels in the mask. A 'Leica' confocal microscope was used to acquire the image with 63x/1.4 oil lens. The results of the proposed method was compared with other methods such as traditional intensity-based segmentation approach. The result shows that proposed method was able to segment much better than other methods used for comparison. The average accuracy which was achieved by the proposed method was 93.43% for the tested images.

Joint, colour and shape-based segmentation methods were proposed in [12] to segment cell nuclei. The aim of comparing colour with nuclei appearance information was to achieve a good result of segmentation for nuclei detection and to overcome microscope image problems. The CIE Lab colour space was used in the experiment. Normalized cross correlation of the greyscale image and mean nuclei were applied to calculate the 'circularity' of the area. Colour details and area circularity

information were used in a fuzzy logic engine with rules to distinguish between nuclei and background areas. A Gaussian Mixture Model was applied on the results obtained and the final results were compared with the results of the other non-fuzzy methods. The result of the proposed method was much better as compared with the results obtained from other methods.

2.2.3 Size based dust particles classification

Several attempts have been made to classify dusts based on size. In [13], the study focused on measuring the size and size distribution of airborne dust samples. The researchers used high resolution images and developed ImageJ code to measure the length and width of the particles. The same source materials were used as test materials, which are, ground pine tree bark pellets and soft pine wood sawdust pellets. The original dust particles were divided into subsamples using a 230 mesh (63 μm) sieve. Strong and thin cardboard was used in the sampling procedure, which indicates that particles can be seen by the human eye. For measurement purpose, Feret's diameter in ImageJ was used to measure the length of particle and "pixel-march" techniques for width, which is similar to the one used in [23]. The figure below shows dimension measurements used.

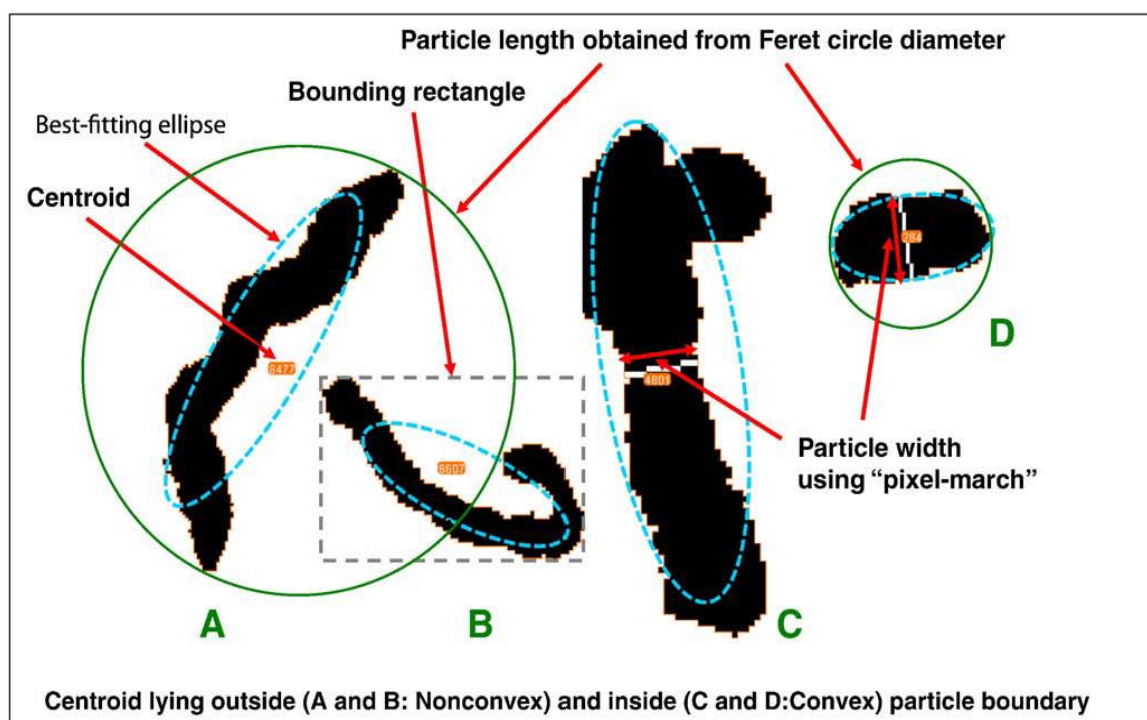


Figure 2-3 Particle dimension measurement using Feret's diameter and pixel-march methods [13].

For the ease of analysis, the particles were spread to avoid any touch or overlap with each other, before scanning the image. This approach leads to a simpler scanned image where particle overlap does not challenge segmentation and subsequent measurements. An automatic-thresholding algorithm was used, when converting an image into a binary image, separating the foreground objects

from the background. Particles were grouped together based on their lengths. Other parameters such as skewness, minimum and maximum dimensions; and arithmetic means were also used to evaluate length and width. The below figure illustrates the overall system flow.

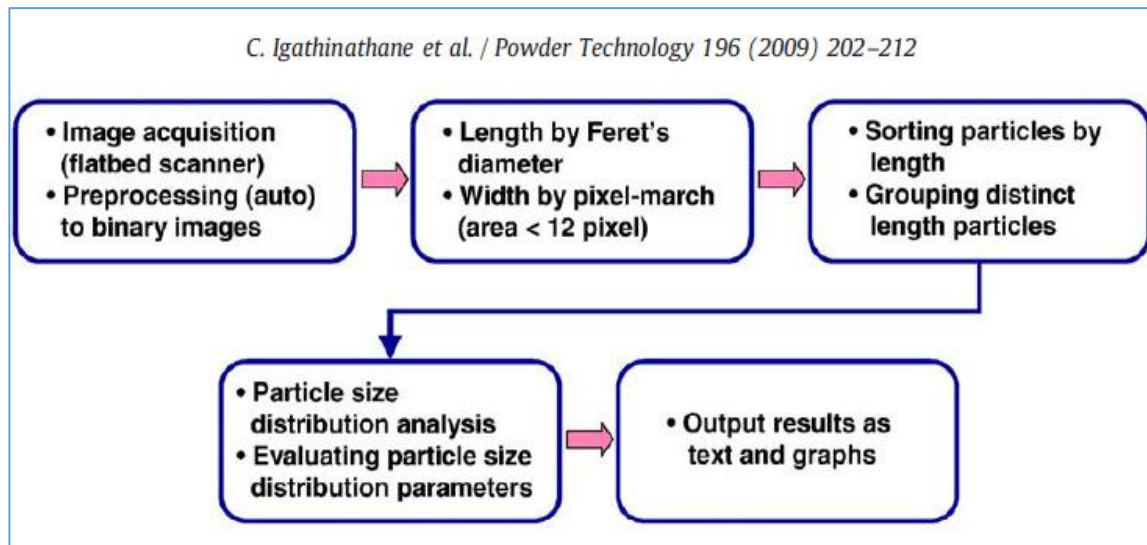


Figure 2-4 Process flow in the developed machine vision plugin [13].

As far as the result is concerned, the implemented algorithm was able to analyse eight images per second with high accuracy, exceeding 98%. Arithmetic mean of wood length was 0.1138 ± 0.0123 mm and bark pellets length was 0.1181 ± 0.0149 mm. However a drawback on the approach is the need for the scanning to be done after ensuring that particles do not overlap. This is a practical challenge that is difficult to meet.

In [15] proposed a novel technique to analyse the particle size distribution. Their research compared two methods for calculating size distribution of touching and non-touching particles. A set of different types of gravel with different shape and size were used as sample material. The images were taken using a high speed digital camera with 16 mega pixel resolution; though for the experiments conducted the resolution was set at 2 MP. The images were converted into grey-level images and a median filter was used to remove noise. An edge detection method was also used during the segmentation process of particles to detect edges. There were two diameter methods used: equivalent area diameter and Feret's diameter. For experimental purposes, two different images were used. One has touching particles and the other has non-touching particles.

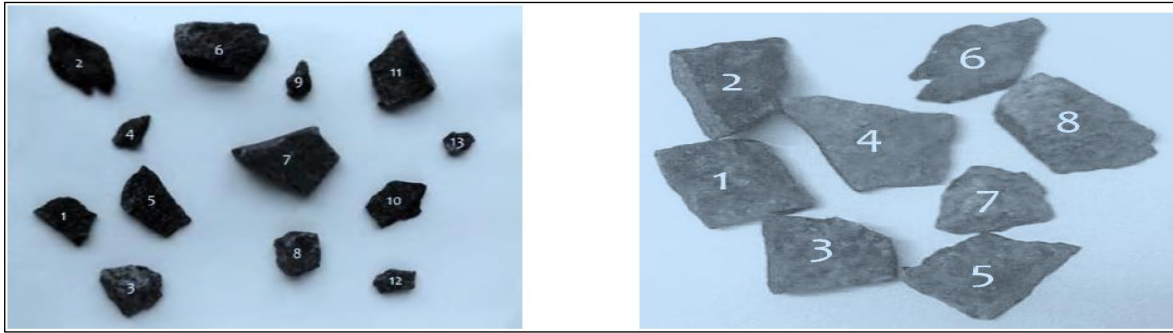


Figure 2-5 Non-touching and touching particles [15].

The results show that the segmentation accuracy of non-touching particles is higher than that of the touching particles. Additionally, the analysis using Feret's diameter based method achieved higher accuracy than equivalent area diameter based approach.

The authors in [16] used a method to estimate particle size distribution by using the weight of limestone during the loading of a ship. In this paper the authors considered the distribution of size as a function of the number of particles. Initially, they focused on showing that a significant weight-transformation error can occur when transforming from one type of size distribution to another. The system used a camera mounted at 24 degrees and a laser with line-generation optics to capture images before they were converted to a 3D surface data, for further analysis. Particles were separated manually and labeled. Then a classifier was trained to separate the particles according to the size. The result concluded that, the good results from size based distribution can be obtained by weight.

An initial inspection of the collected dust particles to be used in the research presented in this thesis revealed that particle size cannot be used in the classification of dust types being investigated. This is due to the common occurrence of dust particles of the same size dimensions, but belonging to different types of dust.

2.3 Road Surface Recognition

Generally, a road image can be classified into a structured (e.g., a road in urban area) or unstructured one (e.g., a road in rural area). For structured roads, the localization of road borders or road markings is one of the most commonly used approaches. Colour cues [4], [5], [6], Hough transform [7], [8], steerable filters [9], [10], and Spline models [11], [12], [13] etc. have been utilized to find the road boundaries and markings. The drawback of these methods is that they only consistently work for structured roads which have noticeable markings or borders. Methods based on segmenting the road using the colour cues have also been proposed, but they do not work well for general road images, especially when the roads have little difference in colours between their surfaces and the environment. In addition, Laser [14], Radar [15] and stereovision [16] techniques have also been used for structured-road detection.

For unstructured roads or structured roads without clear boundaries and markings, the investigators in [17] have combined the Adaboost-based region segmentation and the boundary detection constrained by geometric projection to find the “drivable” road area. However, it needs many different types of road images to train a region classifier, which might be onerous. Reverse optical flow technique [18] provides an adaptive segmentation of the road area, but the method does not work well on chaotic roads when the camera is unstable and the estimation of the optical flow is not robust enough. Stereo cameras [19], [20] are also used to determine terrain traversability. When there is little difference in colour between the road and off-road areas, it is hard to find a strong intensity change to delimit them. The one characteristic that seems to define the road in such situations is texture. The associated approaches [21], [22], [23] have attempted to define the forward “drivable” image region by utilizing the texture cue. They compute the texture orientation for each pixel, then seek the vanishing point of the road by a voting scheme, and finally localize the road boundary using the colour cue. The approach belongs to this line of research. Although multiple-sensor method [24] can handle an unstructured road case, it was considered beyond the scope of the paper, which only uses visual information.

Existing literature in road surface recognition proposes a number of different techniques and approaches. Most of these approaches and techniques are used in structured roads where lane and borders are clear. The authors in [53] used Local Adaptive Soft Voting (LASV) algorithm to estimate the road vanishing points. The detection process of the road can be divided into two steps. Initially, the author associated the estimated vanishing point with the main part of the road before segmenting the road area based on detecting vanishing point. Gabor filter and edge detection techniques were used to detect road boundaries and improve the accuracy.

A Wavelet Packet Transform technique has been implemented by the investigators in [54] to detect road surface conditions. The authors considered four different pattern states of road surface namely dry, wet, snow and icy. A polarization filter was attached to the camera to acquire vertical and horizontal images. Wet roads were separated from other roads by considering a threshold ratio of 1.3 and more. Implementation of wavelet algorithm indicated that images with snow had low frequency, dry road had high frequency and ice composition had frequency components between that of dry and snow roads. For training and classification purposes, a Support Vector Machine (SVM) was used and over 87% accuracy was obtained.

In [55], a novel approach has been proposed and developed to detect road surface. An on-board camera was used to detect the free road surface captured from a vehicle. The authors showed that the developed algorithm is robust to shadow and camera dependent. Thus, intrinsic parameters of the camera had to be utilised in the design. It combines a shadow invariant feature space with road class-likelihood. Images are segmented manually to generate ground truth to validate the algorithm proposed. The algorithm works accurately in still images and does not depend on shape of the road or any temporal restrictions.

The literature review above on road surface type detection revealed that most of the research conducted in this area either focuses on detecting the most likely area of the road, hence separating it from the other objects of the scene or on recognising few types of road surface types. Therefore within the research context of this thesis, we investigate only road surface type and assume that the road region has already been detected. Further we investigate the recognition of five different road surface types.

2.4 Animal Detection and Recognition

Many algorithms have been proposed over the past years to detect objects and recognised the detected objects as animal using different attributes of an animal appearance such as shape, colour and etc. The processes of detecting and recognising animal are similar to recognising any other object. These processes can be summaries in three main steps: segmenting object from the background, extracting the features' of the extracted object and classification step in which the object type can be identified. Each of the above steps has different challenges that need to be addressed before moving the next step. For instance, animal images captured in wildlife field makes the classification of the animals very challenging tasks due to the fact that animals are often surrounded by plants or trees and shadow in the background in such environment [74]. Additionally, animals' appearance in different pose, cluttered background of the image, different illuminations, different view point and occlusions makes the segmentation step a challenging task and the result might lead to miss-classification of the animal type.

2.4.1 Animal Object Segmentation

A popular object detection framework was proposed in [92] by Viola and Jones. It was primarily developed for real-time human face detection before being used to detect animal faces lately. The Viola-Jones algorithm can be used to train and detect a variety of different object classes. The main advantages of this algorithm are: the high detection rate (high true positive and low false negative) and the superior performance in real-time detection. The algorithm is developed for face detection and not for face classification as its purpose is to distinguish between face and non-face regions. The algorithm has four stages: feature selection, Integral image creation and representation, AdaBoost feature classifier (Weak classifiers) and Cascade classifiers (Strong classifiers). The idea is to combine face alignment with face detection in order to observe the aligned face based on their shapes and thus to provide better features for face classification [63]. In order to make the combination more effective, the authors made the approach learn the two tasks jointly in the same cascaded framework. This is achieved by exploiting recent advances in shape based face alignment. Extensive results provided by the authors demonstrate that the approach achieves good accuracy on challenging datasets. In [127], the authors used Sobel edge detector to detect the bird body. The authors in [73], proposed a thresholding segmentation method to detect animals in order to extract the target from the background. However, when using this technique, thresholding value should be selected carefully. It is noted that in order to obtain more accurate segmentation using thresholding based approaches, different background of objects will require different threshold values and this could limit the

algorithm's practical use. Canny Edge detector is used in [88] to identify lines in images, leading to the possibility of segmentation.

In [75], a stereo Vision based algorithm was proposed detect and identify the animals. The researchers tried background segmentation where the background pixels were identified. The authors compared each pixel's value with values of background model using Euclidean distance. If the result of the comparison is greater than the threshold then it will be considered as a foreground and a candidate has to be defined. The authors claimed that the threshold's value depends on the global illumination of the scene, and the computation of threshold is obtained from the average of the pixel intensity, besides the darker and lighter pixels' differences.

The investigator in [90], stated that there are mainly two ways for segmentation. First one is a complete segmentation of the animal by using the shape and size as features. The second one is a partial segmentation, where only a part or parts of the animal is segmented. The researcher reported that the pixel based features for partial segmentation is preferred. A Histogram of Oriented Gradients (HOG) was proposed in [87] for detecting animals in still image and video. In some cases, the animal might be divided into two parts, the reason is that some pixels of the animal's body were filtered by the threshold. Therefore, it is suggested that using texture and colour space segmentation will be more effective [65]. A coarse segmentation is used in [65] to remove unnecessary regions of most parts of the background by applying a pixel velocity threshold. Based on the segmented regions, there is a need to filter out potential background pixels. Additionally, morphologic operations are required to remove speckle noise and fill holes. A study of [74], proposed a method where the input animal image is segmented by a graph-cut based technique. The researchers divided the segmented animal image into a number of blocks. They also studied the performance by considering the entire image as a single block, and also by splitting the image into 4, 16 and 64 blocks. The authors in [81], presented a method for automatic reconstruction of permanent structures of an indoor scene. The main idea is to formulate a graph cut to label the space portioning. The authors state that by formulating the graph cut the final surface is extracted perfectly. A new segmentation technique to detect micro calcifications in mammogram images by modifying GrabCut Segmentation technique [123]. The new technique uses k-Means algorithm instead of Gaussian Mixture Model (GMM) to compute the clusters. The texture component is integrated into GrabCut framework to detect the texture-based boundaries that enables a highly accurate and efficient segmentation proposed in [82]. On the other hand, the study conducted in [77] presented a new scheme for image segmentation based on saliency detection. The authors state that GrabCut is a useful technique to segment an image with satisfactory results when an informative input are given. By modelling colour data with GMM. The researchers

suggested to use another image representation to improve the efficiency of image segmentation. In addition, full-automatic SpatioTemporal GrabCut human segmentation methodology proposed in [67]. They tried to detect subjects using HOG-based cascade classifiers. The researchers initialize GrabCut algorithm by defining a set of seeds from face detection and skin colour model. They also used GrabCut segmentation algorithm to smooth the face fitting in the video sequences.

The authors in [80], presented the requirements for the segmentation of an image into piglets (foreground) and background pixels in order to distinguish the low-contrast, often motionless, outlines of the piglets from the profusion of edges in the scene. The first stage of image segmentation was image differencing, and the difference image was thresholded, where pixels belonging to the regions of piglets are defined as those above the threshold. The second stage of segmentation used was a Laplacian operator to improve the separation of piglets. The third stage of segmentation was to group the segmented pixels. Chain coding was also used to group the pixels into blobs. The blobs that are too small to be piglets were ignored as noise. The investigators in [93], proposed using scale-invariance, noise tolerance, variation tolerance and rotation invariance to segment animals in a desert. The study in [91], used graph-cut by imposing an additional connectivity prior. Then the authors formulated several versions of the connectivity constraint and showed that the corresponding optimization problems are all NP-hard. They proposed a practical heuristic technique named Dijkstra GC optimization algorithm, and a slow method based on problem decomposition, which provides a lower bound for the problem.

Segmentation algorithms have been used in application areas beyond animal detection and recognition. For example, in [69] an accurate automated 3D liver segmentation algorithm was developed to measure liver volumes in MRI images using active contours. Active contour segmentation was used to refine the initial surface to precisely determine the liver boundaries. In [96], a new infinite active contour model using hybrid region information of the image was proposed for blood vessel segmentation problem in retinal images. Further, the authors in [71], developed a method for lung nodule detection. The authors segmented the lung area by using an active contour model.

2.4.2 Feature Extraction

Feature extraction is essential step for object classification. Some animals possess highly distinctive shapes, colour, texture patterns, or they can be characterized by a combination of these properties.

The authors in [62], proposed a method for animal classification that uses facial features. The study conducted in [76], proposed a method to differentiate the individual zebra using their coat markings. The investigators in [59], presented a model for elephant identification based on the elephant's neck shape. The authors in [86], built a 2D articulated models and this model used to detect the animal in the videos using texture features. The study in [72], stated that animals can be classified using unsupervised learning. According to [58], developing an animal classification system can be performed using joint textural information. The study in [95], focused on automated detection of elephants in wildlife using video to record the elephant which uses colour models. A method for automated identification of animal species using camera trap images was proposed in [94]. They used the dense SIFT descriptor and cell-structured LBP (CLBP). The authors in [93], used a feature-based learning approach using an LBP-like operator.

According to [90] using pixel based features for partial animal segmentation is the preferred way as there will be no information about the entire animal body. In addition, the author pointed out that there are seven features that can be effectively used to recognize an animal, and they are: "hue, saturation, horizontal Sobel, vertical Sobel, mean, variance and Canny Edge Detection" [90]. The hue and saturation can be easily extracted by converting the RGB colour image to HSV. Due to the likely variation in light intensity, the 'value' component will often be useless. Further, the horizontal and vertical Sobel edge components can be computed by the Sobel kernels application. The authors emphasized that the mean and variance of the image are also used as features. The average Gray scale value mainly describes the mean of the image and the variance defines whether there is a vast difference between the lowest and highest pixel values. Finally, the Canny Edge Detector is used to find all edges in the image in order to represent the animals' texture. Briefly, it is functioning by using Sobel Kernels to find the maximum gradients. The Author in [98], used three descriptors which are: Haar-like features, Hog (Histogram of Oriented Gradients) features and LBP (Local-binary pattern) features. Animals can be detected using their colours, beside texture and shape features. Colour extraction can be achieved using segmentation methods which allows ignoring unnecessary regions by intensifying certain colours. The selection of a suitable colour space such as RGB, Luv, HSI or HSV is the head of this step [70].

Features such as Haar-like feature, Local Binary Pattern (LBP) features, and Principal Component Analysis (PCA) are widely used in object detection [92]. Another algorithm example that uses Haar-features is detecting and tracking Lions' faces [61].

For instance, in [95] started by pre-processing the input images to reduce the processed amount of data. They proposed using LUV colour space, and applying mean-shift clustering algorithm on the pre-processed images in order to perform colour segmentation. Their approach is learning the colour model of elephants by training the images. Unfortunately, this method is useful only on daytime conditions, and it seems it is not applicable to work in night-time conditions. Additionally, a colour-based detection depends on the background subtraction method determined in [85]. This method can be divided into two steps, background extraction and colour- based moving object detection. Yet this method focuses on a stable background and moving target only.

For example Scale-invariant Feature Transform (SIFT) or Histogram of Oriented Gradient (HOG), use an object's edge and contour to define and locate the boundary of the object in image or frame [68]. The investigator in [94], applied a new Contour-Based HOG algorithm (CNT-HOG) to detect the deer from the thermal images. This device has an excellent performance in night-time deer detection. Colour texture moments (CTM) features are extracted from each block after splitting them [74]. The researcher used feature extraction using HSV colour space because of its perceptual uniformity. The HSV colour space is widely used in the field of computer vision where it is a non-linear transform of the RGB space.

2.4.3 Animal Object Classification

After feature extraction, deciding on the appropriate classifier is a crucial stage. The authors in [84], proposed a model using template matching for animal detection. The student conducted by [95], used a Support Vector Machine (SVM) classifier by using colour features for animal classification. the authors in [86], investigated the use of different classifiers such as K-way logistic regression, SVM, and K-Nearest Neighbours. The experiments conducted in [58], used three different classifiers namely, a Single Histogram, SVM and Joint Probability model. Both [94] and [93], used linear SVM classifier for animal classification. The study in [90], adapted KNN as a classification method (supervised classifier) in all the experiments. According to [87], Cascading is a concatenation of various classifiers (group based learning). The key advantages of boosted cascade classifiers over monolithic classifiers are that it is a fast learner and requires low computation time. Cascading also eliminates candidates (false positives) early on, so later stages don't bother about them. AdaBoost and SVM classifiers were used in [98]. An improved version of Haar features and AdaBoost classification algorithm has been investigated. But they focus on animal faces instead of the animal bodies [62]. The extracted features are queried to neural network and K-nearest neighbours, which can be used to know the class label of unknown animals [74].

Chapter 3

Research Background

The contributory work presented in chapters 4-6 of this thesis is based on a number of fundamental concepts and background theory that are presented in this chapter. The chapter first presents the popular K-means clustering technique. Additionally, it presents the theoretical, mathematical and technical concepts of different colour spaces, and different feature descriptors, including HOG features and LBP features, which are used to describe the local structure of images. Moreover, detailed descriptions of machine learning algorithms, which are used for classification, are also presented in this chapter. Furthermore, it also introduces the reader to a detailed explanation of Convolutional Neural Networks (CNN), deep learning and some concepts that are used in within them such as ROI pooling, Anchors, Anchor Filters and Max pooling operations used to improve the results and speed up processing time.

3.1 K-Means Clustering

According to [99], K-Means clustering is an algorithm that was developed with the aim to group n objects into set of K clusters in which each object fits to a cluster with the nearest mean. There are two types of clustering algorithms: supervised and unsupervised. K-means clustering algorithm is considered as an unsupervised learning algorithm. Given $X = \{x_1, x_2, x_3, \dots, x_n\}$ where X is set of feature vectors [20] the goal of K-means clustering is to separate similar pixels into one cluster. A K-Means clustering algorithm operates in five steps. First step: Define an initial (random) solution as vectors of means. It can be defined as $M(t=0) = [m_1, m_2, m_3 \dots m_k]^t$ where M is a vector of means, K is the number of clusters and t is the iteration being considered. Second Step: classify each input data according to the randomly defined means $m(t)$. Third step: classification obtained in second step is to be used to re-compute the vectors of means $m(t+1)$. Fourth step: update the iteration $t = t + 1$. Fifth step: compare and check the new vector of means with old one ($m(t) - m(t-1)$) and re-compute the input data.

3.2 Colour Spaces

Generally, the colour images use the RGB colour space, which has three channels (red, green, and blue) per pixel. Each pixel of a RGB image is allocated a range of 0-255 RGB values. All the RGB images fall under the RGB colour Model, where an image of size $H \times W$ is represented as an $H \times W \times 3$ array of colour pixels. Although this colour space is suitable for display systems, it is not recommended for image analysis and segmentation as the R, G and B colour components of a pixel is highly co-related. The direct use of these components cannot achieve desirable effects, which makes it necessary to transform a RGB image into another colour characteristic space when solving real problems [17].

3.2.1 Converting from the RGB colour space to other colour spaces

The RGB colour space is the colour model that is used widely in cameras, computer monitors etc. This is due to the fact that imaging sensors of a camera captures an image in the RGB format. However, there are numerous other colour models available for colour representation and conversion. The RGB has three colour components to represents all the colour in the image. Hence, there might be a need to convert the RGB colour components into two, three or four dimensional colour space to address different issues. Following are some of the colour spaces that can be used for image processing, manipulation or for the purpose of enhancing overall image quality.

- 1- Hue, Saturation and Value (HSV).
- 2- CIE L^*A^*B
- 3- Hue, Saturation and Intensity (HIS).
- 4- HSL
- 5- CMYK

3.2.1.1 CIE L^*A^*B Colour Space

CIE L^*A^*B , or LAB for short, is a 3D colour space that uses lightness (L) as a luminance channel and A and B as colour channels. This colour space is closer to the human eye's perceptual capability as compared to the RGB colour space. The lightness (L) increases from the bottom to the top of the three-dimensional model. The lightness falls between 0% -100%, where 100% represents White, 50% represents Mid-Gray and 0% represents Black.

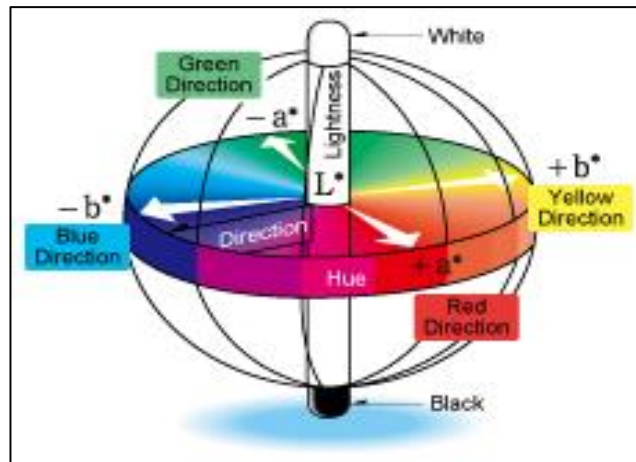


Figure 3-1 The CIE LAB colour space [18].

The A axis extends from green ($-a$) to red ($+a$) and the b axis from blue ($-b$) to yellow ($+b$). This colour space is better suited to many digital image manipulations than the RGB colour space, which is typically used in image editing programs. The LAB model is non-linear unlike RGB. Thus, the colour differences, which humans perceive, correspond to distances when colour is metrically measured. Additionally, it is known as device independent, i.e. it can be used to communicate different colours through different devices [100 & 101].

3.2.1.2 HSV Colour Space

The Hue Saturation Value (HSV), is another widely used colour model. It was designed in such a way to represent the way human eyes observe the colours [102]. The figure below shows the HSV model.

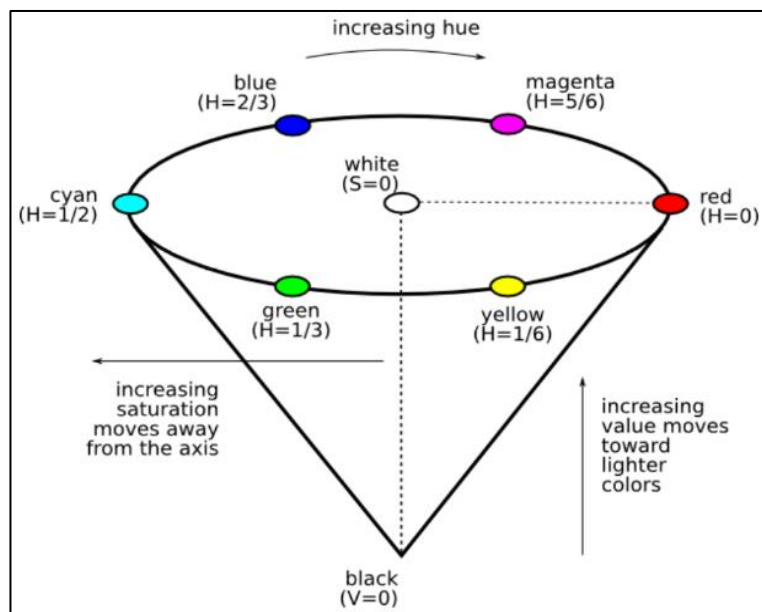


Figure 3-2 Single-hexcone model of HSV colour space [103].

The Hue (H) is the dominant component, which contains the colours that are perceived by the observers. All pure colours can be observed in this channel, with an H value that ranges between 0 and 1. For instance, the hue (H) attribute of Red colour is 0, whereas, for Green it is $1/3$ and $2/3$ for the Blue colour. The Saturation (S), describes the whiteness level of the colours in Hue. The saturation attributes fall between 0 and 1. The White colour has a 0 saturation attribute and 1 for pure Red colour. All other whiter red colours, have saturation less than 1. The Value (V) describes the intensity or the darkness of the colours. The Black colour has value of 0 while, White colour has value of 1.

An RGB image can be converted to obtain the HSV components by applying the formula below. The RGB model uses a range between 0 and 255, unlike HSV, which uses values between 0 and 1. Thus, one should obtain Δ delta in order to convert the range by dividing RGB value by 255 [104].

$$R' = R/255$$

$$G' = G/255$$

$$B' = B/255$$

$$C_{\max} = \max(R', G', B')$$

$$C_{\min} = \min(R', G', B')$$

$$\Delta = C_{\max} - C_{\min}$$

Hue Calculation formula:

$$H = \begin{cases} 0^\circ & \Delta = 0 \\ 60^\circ \times \left(\frac{G' - B'}{\Delta} \bmod 6 \right) & , C_{max} = R' \\ 60^\circ \times \left(\frac{B' - R'}{\Delta} + 2 \right) & , C_{max} = G' \\ 60^\circ \times \left(\frac{R' - G'}{\Delta} + 4 \right) & , C_{max} = B' \end{cases}$$

Equation 3-1

Saturation Calculation formula:

$$S = \begin{cases} 0 & , C_{max} = 0 \\ \frac{\Delta}{C_{max}} & , C_{max} \neq 0 \end{cases}$$

Equation 3-2

Value or Brightness Calculation formula:

$$V = C_{max}$$

Equation 3-3

The table below shows the new colour attributes after converting from the RGB colour space to HSV by apply the above formulas.

Colour Name	RGB			HSV		
	R	G	B	H	S	V
Black	0	0	0	0°	0%	0%
White	255	255	255	0°	0%	100%
Red	255	0	0	0°	100%	100%
Blue	0	0	255	240°	100%	100%
Yellow	255	255	0	60°	100%	100%
Gray	128	128	128	0°	0%	50%
Green	0	128	0	120°	100%	50%
Navy	0	0	128	240°	100%	50%

Table 3-1 Colours Conversion Values from RGB to HSV.

3.3 Feature Descriptors

In computer vision a feature descriptor is defined as a vector that describes an image or a part of it based on its characteristics features. Often a feature descriptor combines/concatenates a set of features belonging to an object of interest. The numerical details of a feature vector can be used to distinguish the object being described, from other objects. There are numerous feature extraction algorithms: SURF, HOG, LBP, HAAR, and GLCM, which can be used to extract different features and information from an image. In image recognition such features are usually grouped together to form a more substantial feature descriptor with the hope that such a descriptor will be able to describe an image more accurately for the purpose of object detection or recognition.

3.3.1 HOG Features

Histograms of Oriented Gradients (HOG) is one of the feature descriptor algorithms that is used widely in object recognition. According to [105], this algorithm focuses on the appearance and shape of the object through intensity gradients. It uses a sliding window technique where a small cell is moved over the input image and accumulates the results of each sliding window, a 1-D histogram, of gradient direction. This algorithm computes the gradient vectors of each pixel in an image by measuring the left and right or top and bottom, pixel value differences to obtain the pixel value gradient. The example below shows the gradient vector calculation.



Figure 3-3 Gradient direction.

Horizontal value difference or x direction gradient can be computed as:

$$XD = \text{right pixel} - \text{left pixel}$$

Equation 3-4

Vertical value difference or y direction gradient can be calculated as:

$$YD = \text{top pixel} - \text{bottom pixel}$$

Equation 3-5

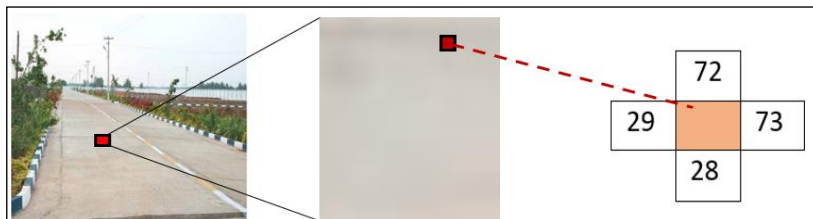


Figure 3-4 Example of gradient calculation.

The computations for the above example are:

$$XD = 73 - 29 = 44$$

Equation 3-6

$$YD = 72 - 28 = 44$$

Equation 3-7

Using both, XD and YD, and calculating the ratio (YD/XD) and obtaining the inverse tangent gives the pixel value gradient at the point. A gradient vector can be calculated by putting together the gradients at each point of the images, scanned in raster scanned order. Once the gradient vector is obtained the gradient values are put into a 9 bin histogram that has range between 0 and 180 degrees. The next step is normalising gradient vectors by applying histogram normalisation, which make the vector invariant to the illumination, and block normalisation.

3.3.2 LBP Features

The Local Binary Patterns (LBP) is one of the techniques used to describe the texture details of an image. The idea is to label each pixel of the image with a decimal number so that the local structure of the image is fully encoded. It uses 3X3 pixel blocks to transfer image pixels into decimal numbers [106].

p8	p1	p2
p7	pc	p3
p6	p5	p4

Figure 3-5 Example of 3X3 pixel block for 8 pixel neighbourhoods.

The 3X3 equation can be described as follows:

$$\begin{pmatrix} p8 & p1 & p2 \\ p7 & pc & p3 \\ p6 & p5 & p4 \end{pmatrix}$$

Equation 3-1

Once the 3X3 block is obtained, each pixel of the 8 neighbours' pixels is thresholded (T) with the value of centre pixel. In this operation, each pixel is subtracted from centre pixel value (pc).

$$\begin{pmatrix} T(p8 - pc) & T(p1 - pc) & T(p2 - pc) \\ T(p7 - pc) & & T(p3 - pc) \\ T(p6 - pc) & T(p5 - pc) & T(p4 - pc) \end{pmatrix} = \begin{pmatrix} T(x8) & T(x1) & T(x2) \\ T(x7) & & T(x3) \\ T(x6) & T(x5) & T(x4) \end{pmatrix}$$

Equation 3-2

The result of each threshold $T(x)$ operation is either encoded as 1, for a positive, or 0, for a negative value.

$$T(x) = \begin{cases} 1 & \text{when } x \geq 0 \\ 0 & \text{when } x < 0 \end{cases}$$

Equation 3-3

To label the centre pixel, each encoded result from the thresholding operation is multiplied by the power of 2 before it is summed with other encoded values. The calculation below is used to obtain the label of centre pixel in specific radius R of pixel neighbourhoods.

$$LBP_{P,R} = \sum_{v=0}^{P-1} T(p_v - p_c) 2^v$$

Equation 3-4

Where P is sampling point, which starts from top left and is decreased by 1 until the last pixel of the 3X3 block, before the centre pixel, and v is the encoded pixel value.

3.4 Machine Learning

Machine learning is an application side of Artificial Intelligence (AI) that provides the computers the ability to learn without being explicitly programmed [22]. It is a computer's way of learning from data. There are many machine learning algorithms used for different purposes of a computer's learning processes.

3.4.1 Support Vector Machine

The Support Vector Machine (SVM) is a technique that is used in classification and regression to analyse and classify data into different classes. It is a supervised learning algorithm that needs labelled data in order to separate the data. Based on data, SVM can be used for either binary or multiclass classification.

3.4.1.1 Binary classification

In binary classification, the SVM uses a hyperplane to separate two groups of data with the maximum margin between data in n-dimensional spaces. The typical classification of two classes using this

approach is achieved by maintaining maximum width or range between the closest points of support vectors of both classes [107].

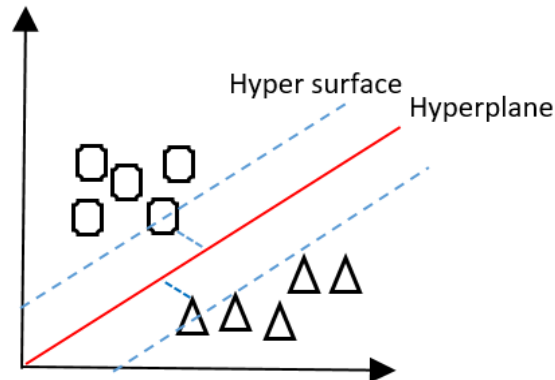


Figure 3-6 Optimal classification of 2 classes.

In the above example, the hyper surfaces have maintained maximum and same width between the closest objects to hyperplane, called support vectors, to distinguish between the classes. This example uses a linear classification to classify the data. However, in some conditions, the distribution of the classes make the classification difficult and data cannot be separated linearly. In such a case, a non-linear transformation can be applied where the data is mapped in 3 dimensional, rather than 2-dimensional space in order to classify the data. The SVM is fundamentally a binary classifier that has been developed for two class separation i.e., SVM, basic Logistics classifier and perceptron classifier. However, further enhancements have been made for its use in multiclass classification. The figure below illustrates multi categories classification.

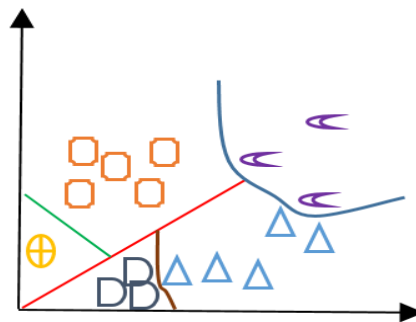


Figure 3-7 SVM Multi-class classification.

3.4.1.2 Multiclass classification

The multiclass SVM is designed to overcome the limitation of the original SVM algorithm, which was developed to classify two classes only. It is widely used in solving real time problems as multiple categories are involved. The Naive Bayes (NB), K-Nearest Neighbours (KNN), Decision Tree (DT) and Logistic are examples of other multiclass classifiers. To solve multi class problems, a

number of SVMs can be used in the training stage. Each binary SVM can separate not more than 2 classes from the dataset. There are some fundamental principles or assumptions in multiclass classification. For instance, the algorithms assume that all the classes are not overlapped. In other words, it assumes that each instance belongs to one class only.

3.4.2 Boosted Cascade Object Detection using Viola-Jones algorithm

Object detection is a process to discover and identify the existence of objects of a certain class. One of the commonly used methods to detect objects from images is using features. However, by working only with image intensities, the RGB pixel values in each and every single pixel in an image makes the feature calculation rather computationally expensive and as a result it performs very slow on most platforms. This problem is addressed and solved by Haar-like features which were developed by Viola and Jones on the basis of the proposal presented in [83] in 1998.

3.4.2.1 Haar-like feature computation

A Haar-like feature considers neighbouring rectangular regions at a specific location in a detection window, sums up the pixel intensities in each region and calculates the difference between these sums. This difference is then used to categorise subsections of an image.

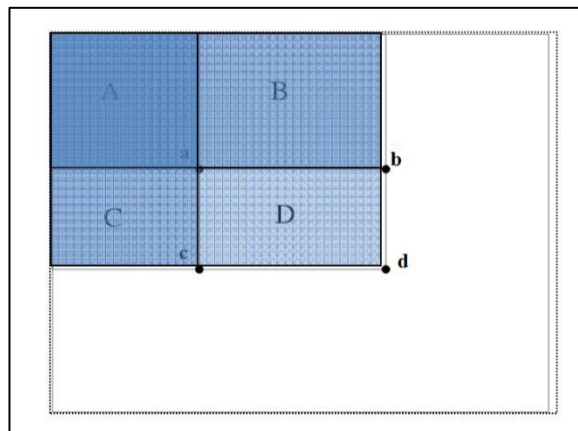


Figure 3-8 Computation of rectangular features.

Assume that:

$$a = A$$

$$b = A + B$$

$$c = A + C$$

$$d = A + B + C + D$$

The integral sum inside rectangle D can be computed as:

$$D = d + a - b - c$$

Equation 3-5

An example of this, is the detection of human faces. Commonly, the areas around the eyes are darker than the areas on the cheeks. One common example of Haar-like features for face detection is a set of two neighbouring rectangular areas above the eye and cheek regions. Another example of using Haar feature in detection is to detect full human body. The below figure shows an example of human body detection.

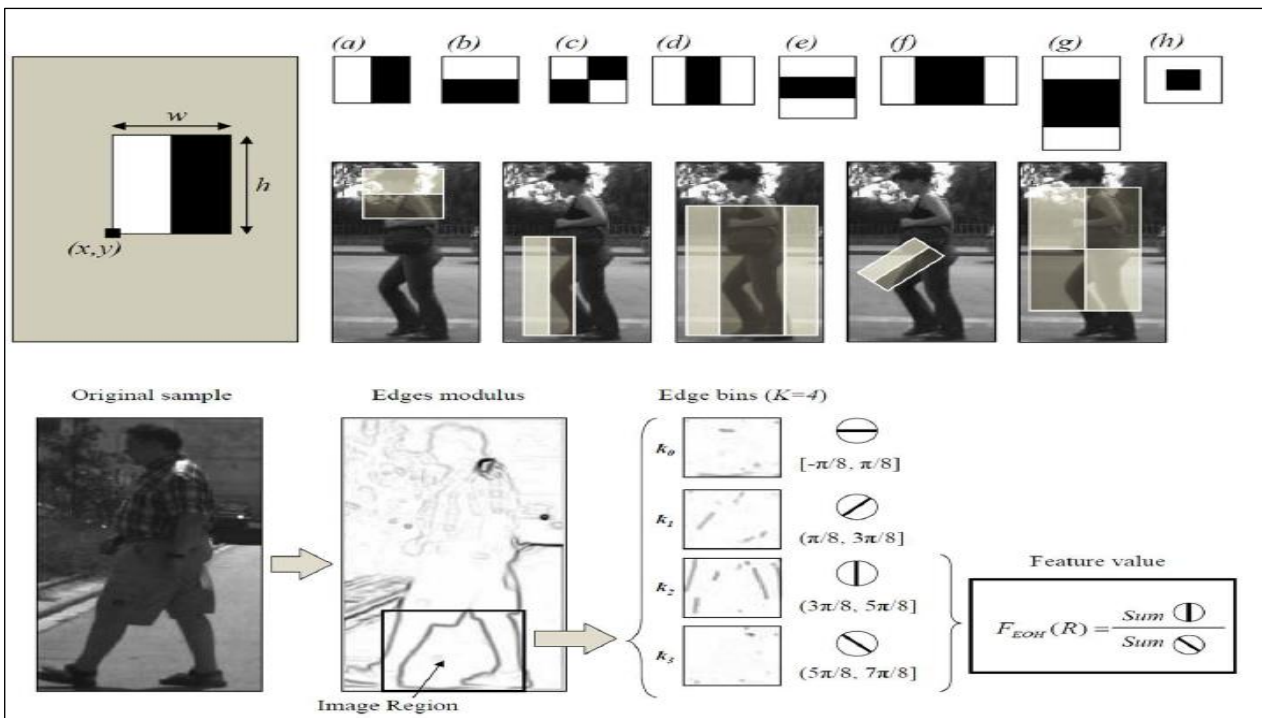


Figure 3-9 Example of Haar-like rectangular areas to detect human body [108].

3.4.2.2 Adaptive Boost (AdaBoost) algorithm for feature selection

AdaBoost algorithm is used to select a subset of relevant features which are informative from Haar-like features to model a human face, for example. It constructs a strong classifier as a linear combination of weighted simple weak classifiers. Which means, features are considered as weak classifiers.

$$F(x) = \alpha_1 f_1(x) + \alpha_2 f_2(x) + \alpha_3 f_3(x) + \dots$$

Equation 3-6

Where F is strong classifier, x represent the image, α_1 is weight and f_1 is the week classifier

3.4.2.3 Cascade classifier

Cascade classifier consists of a list of stages, where each stage consists of a list of weak learners. The system detects objects of interest by moving a window over the image. At each stage, the classifier labels the specific region defined by the current location of the window as either positive or negative; positive means an object was found, on the other hand, negative means that the object of interest was not found in the image [109].

In case the labelling yields a negative result, then the classification of this specific region is hereby complete and the location of the window is moved to the next location in a sequent order. In case the labelling yields a positive result, then the region moves to the next stage of classification. The classifier yields a final verdict of positive, when all the stages, including the last one, yield a result, considering that the object is found in the image.

A true positive means that the object of interest is truly in the image and the classifier labels it as a positive result. A false positive means that the labelling process falsely determines that the object is located in the image, although it is not. A false negative occurs when the classifier is unable to detect the actual object from the image and a true negative means that a non-object was correctly classified as not being the object of interest. Each stage of the cascade must have a low false negative rate, because if the actual object is classified as a non-object, then the classification of that branch stops, with no way to correct the mistake made. However, each stage can have a relatively high false positive rate, because even if the N stage classifies the non-object as actually being the object, then this mistake can be fixed in N+1 and subsequent stages of the classifier. Below figure illustrates the stages of cascade classifier.

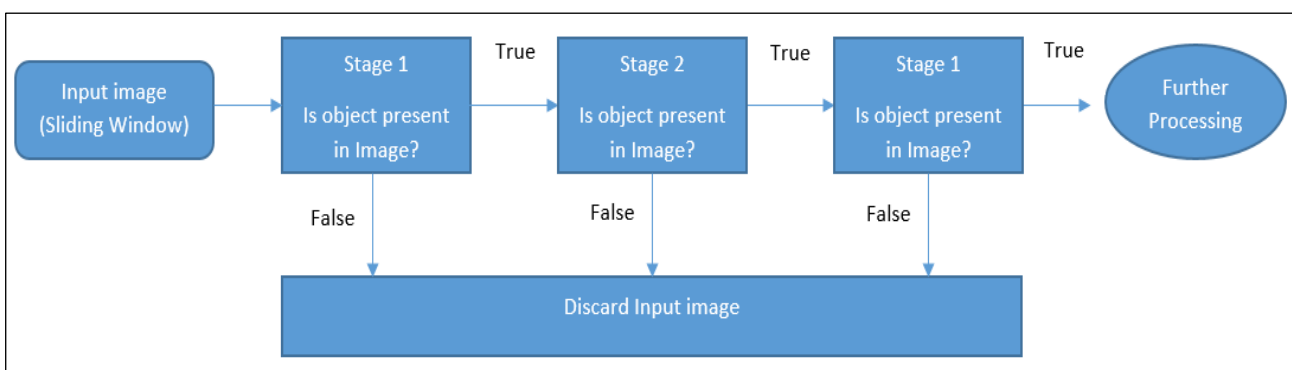


Figure 3-10 Stages of the cascade classifier.

In order to train a cascade classifier, it is suggested to set feature type such as Haar, LBP, or HoG, and other function parameters. Additionally, it requires a set of positive samples (i.e. images contain the targeted objects such as face), and a set of negative images (i.e. images with non-faces) or images without the object/s of interest. The user has to provide a set of positive images with regions of interest (ROIs) specified in the positive samples. It is also required to supply a set of negative images from which the function automatically generates negative that do not contain objects of interest. Before training a new stage, the detector runs using the function that is already trained on the negative samples. Hence, if any object is detected from the negative images it will be considered as a false positive, which is then used as negative samples. Following the process means that by training each new stage of cascade the low chance of mistakes, or in other words, each stage is to correct the mistakes made in the previous stages. Therefore, the detector's overall false positive rate decreases with increasing stage number because it contains a greater number of weak learners. On the other hand, increasing number of stages increases the false negative rate, which results in increasing the chance to discard and reject the positive sample that is made by mistake.

In terms of positive samples, the function of training the cascade object detector automatically determines the number of positive samples to train at each stage. This is based on the total number of positive samples provided by the user and values are determined by the user for true positive rate and the number of cascade stages parameters.

In terms of negative samples, the function calculates the negative samples used at each stage by multiplying the number of positive samples used at each stage by the negative samples factor parameter value.

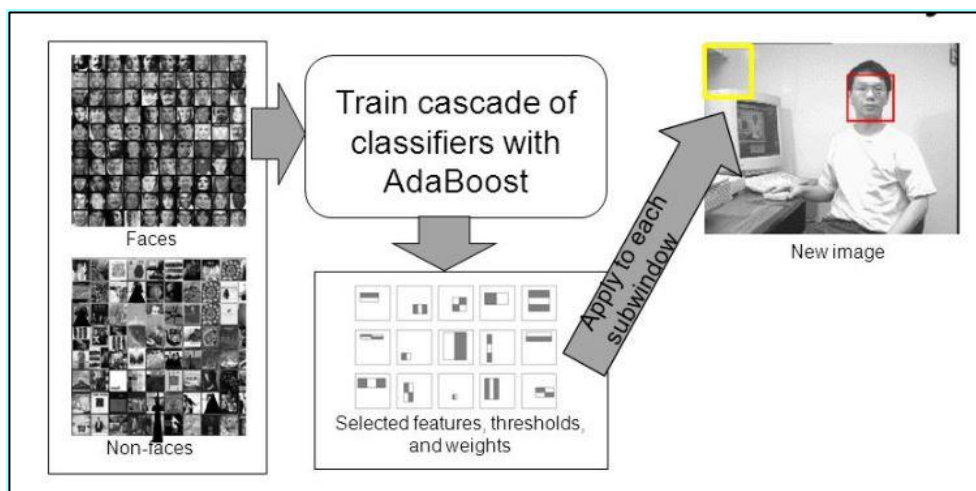


Figure 3-11 Viola-Jones detector summary [110].

3.5 Deep Learning and Convolutional Neural Network

According to [111], deep learning is a part of machine learning that uses a set of layers to classify images and objects. The word “Deep” refers to the number of layers that are employed to extract and learn image features. These layers have the ability to extract different features of the input image and classify the image. Deep learning is based on Convolutional Neural Network (CNN). It uses different layers for different purposes in order to achieve a better result. A description of some layers are described as follow:

- Input: input layer that accepts the RGB image of n size.
- Conv: Convolutional layer that filters or computes the output of neurons that are connected to local region in the input. The features of image or objects are extracted by this layer. It has a so called slider and padding values. The slider value is used to slide the filter to produce smaller output volumes. The padding value is used to pad the input volume a value (usually 0) around the boarder. This allows to control the size of output volumes.
- RELU: is used to apply an elementwise activation function (threshold to zero) to leave the size of volume unchanged.
- Norm: cross channel normalisation.
- POOL: used to perform down sampling operations.
- FC: Is the Fully Connected layer to compute the class score. It has full connection to all activations in the previous layer.
- Softmax: the output is classified by this layer which is always attached to the end of the network.

Below figure show the basic deep learning pipeline.

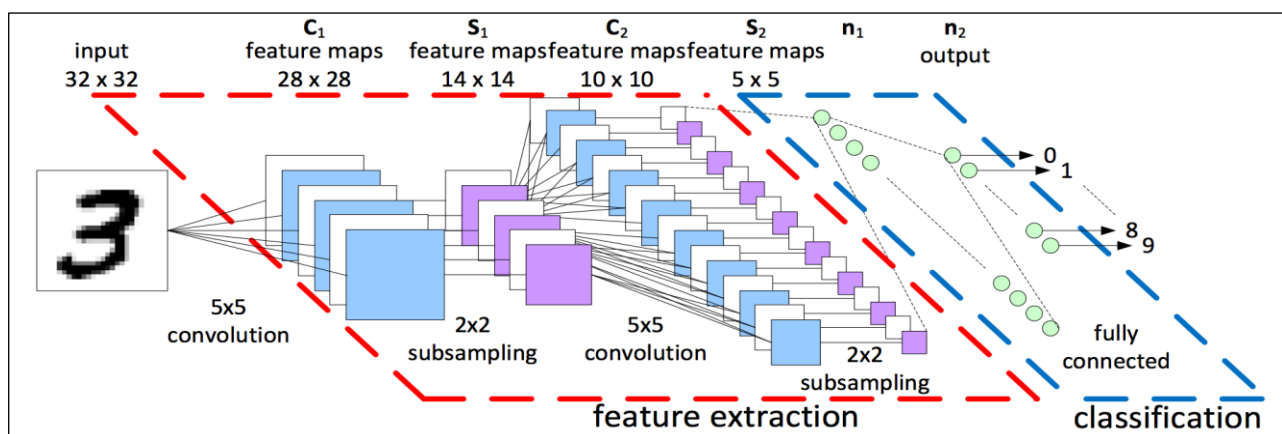


Figure 3-12 Deep learning pipeline [116].

In a deep learning approach, the input image features are extracted in different convolutional layers and fed into next layer. Each of these layers learn from the previous image features.

Deep learning is being used widely in many application domains such as healthcare, autonomous vehicles, surveillance, voice recognition, robotics and manufacturing industries, to name a few. In autonomous vehicles, for instance, deep learning is employed to empower the capability of the vehicles to detect and recognise different objects at the same time, to avoid any obstacles on the road. One of the most critical parts of the deep learning approach is training the network with an extensive and enormous training dataset. Training a network with such massive data is time consuming and require a computer with GPU computing capability. Therefore, various networks: AlexNet and Overfeat, were created, which have been pre-trained with a large database of images such as ImageNet, and have been made publically available. The ImageNet, for instance, has more than 14 million images recently with the aim to increase the figure to 1000 images for each of 100000 different objects [125].

3.5.1 AlexNet

Alex Net is a deep Convolutional Neural Network that has been developed with an aim to recognise different objects within an image/scene. Originally, the network was created to support different research projects within the deep learning research community for better image classification. According to [112], the network was presented in the ImageNet Large Scale Visual Recognition Challenge (LSVRC) - 2010. The aim of the contest was to classify 1.2 million HD images into 1000 different classes. The network achieved 37.5% in top-1 and 17% for top-5 of error rate in test data that was used in the contest [112]. This result is better than previous and other state-of-the-art methods. Since then, AlexNet become widely used as a pre-trained model for image classification. Architecture of AlexNet is illustrated in below figure.

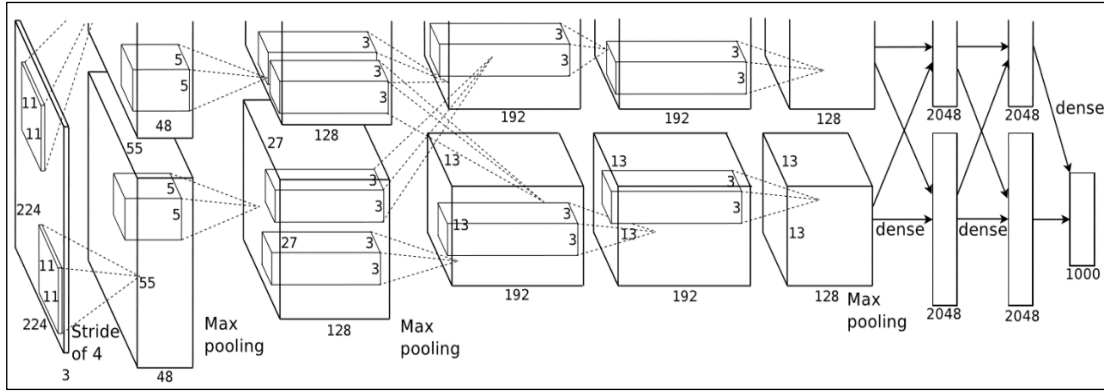


Figure 3-13 AlexNet architecture [112].

The AlexNet was built using 8 layers with weights. The input layer accepts a RGB image with size of 224x224. The last layer is a fully connected layer and can categorise up to 1000 classes.

3.5.2 ROI Pooling Layer

Region of Interest (ROI) pooling layer is a neural network layer that is used in an object detection processes [113]. The fundamental of ROI pooling is transforming the feature map that is obtained from CNN into a fixed-size feature maps for better processing time and detection accuracy. It uses Max Pooling operation in order to achieve the objective. The figures below illustrate the ROI pooling processes.

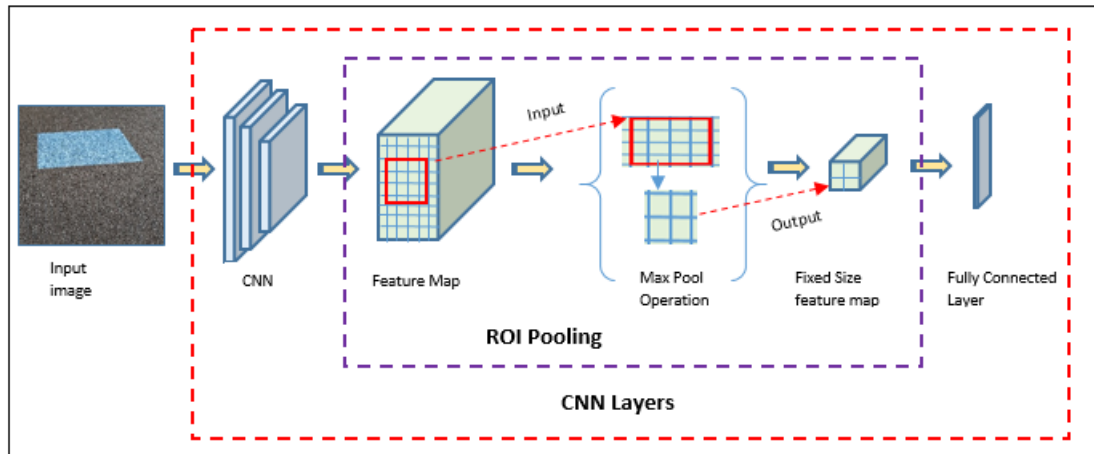


Figure 3-14 ROI Pooling Illustration.

In detecting objects using deep learning, the input image is fed into a set of CNN layers, which produce an arbitrary feature map that contain the regions of interest (ROIs). Each ROI might have different sizes which consumes time during training and makes training difficult especially for real-time application. An overlapping between regions is another challenging part that needs to be addressed to improve training detection accuracy. Therefore, each ROI needs to be down sampled by applying the Max pooling operation as below.

40	25	8	13	30	47	22	9
10	14	21	8	39	9	28	37
7	3	17	33	9	2	33	35
38	8	5	18	2	5	21	11
22	11	13	17	6	19	10	29
21	13	9	21	17	31	32	7
8	23	31	6	23	11	39	4
7	22	45	4	31	18	41	23

Figure 3-15 Single ROI with size of 4x4.

The above 4x4 ROI can be down sampled into a 2x2 ROI in order to speed up the training time.

33	39
18	19

Figure 3-16 Fixed size ROI 2x2.

Once the fixed size ROI with size of 2x2 is obtained, it is fed to the fully connected layer for classification. The same technique can be applied to other ROIs in same feature map, which makes feature map reusable. The ROI pooling technique allows the system to have end-to-end training which improves the accuracy. It also improves training and testing time.

3.5.3 Max Pooling

Max pooling is an essential part of deep learning which controls the overfitting of the network. It uses in Convolutional neural network to down sample the input image. It is commonly implemented between convolutional layers by using filter size and stride. An example of Max pooling operation is demonstrated in below figure.

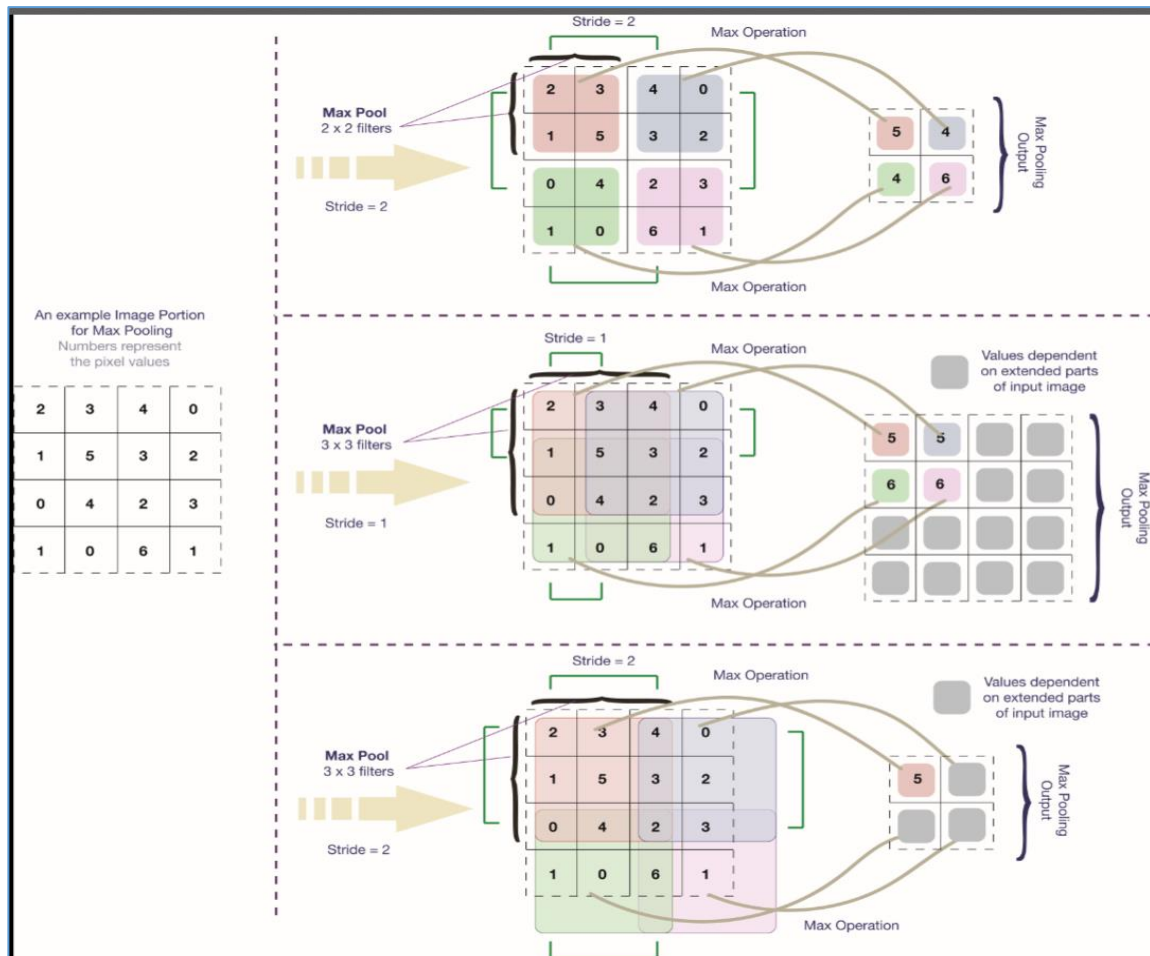


Figure 3-17 Example of Max pooling operation [114].

In the above example, the max pooling is applied in 3 different scenarios. The first scenario is commonly used in deep learning where the filter size is 2x2 and stride is 2. In this case, 4 samples of size 2x2 are created and the maximum number is selected from each sample to create the 2x2 pooling sample. This reduces the samples by 75% of the activation. There are other pooling operations, such as the mean and average, which can be used for down sampling.

3.5.4 Anchors

Anchors are the proposed boxes for a proposed region of interest. It is a technique used in object detection using deep learning to address the variations of object sizes and ratios. In Faster-RCNN, the input images are passed through a convolutional network up until an intermediate layer, ending up with a set of convolutional feature maps on the last convolutional layer. The feature map is divided into multiple squared tiles. A sliding window is run over the feature map to generate region proposals. For each of these region proposals, 9 potential bounding boxes, also called anchors, are generated. The anchors have 4 offset values to represent the box size which are W , H , X and Y .

Where W = width, H = height, (X, Y) = centre.

For instance, a feature map with size of 60x60 can give 3600 possible regions. For each region, 9 anchors are generated, which gives a total of 32400 anchors. The Anchor calculation can be defined as follows:

$$A = (H \cdot W) \cdot 9$$

Equation 3-7

Where A is the total number of anchors, H and W is height and width of the feature map. The figure below shows the different anchor sizes.

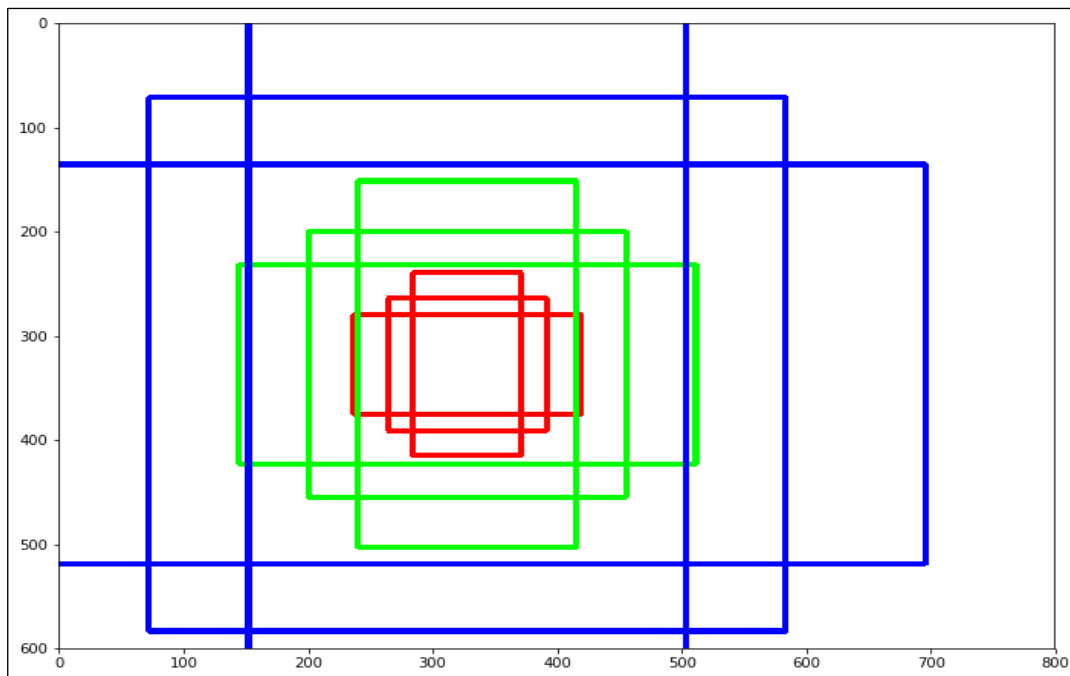


Figure 3-18 Anchor sizes and ratios [115].

It is noticed that all anchors are created with different sizes and ratios. The anchors are generated in three different box scales; 128, 256 and 512 to fit the object inside the box. Each of these scales has three different height and width ratios; 1:1, 1:2 and 2:1.

The variety of anchor sizes and ratios helps to fit different objects within the same image. However, having 9 different anchors for each region reduces accuracy as it generates height false flags.

Chapter 4

Airborne Industrial Dust Particle Classification.

4.1 Introduction

Air pollution is a serious issue which has significant effects on people, the environment, the economy and the planet. It contains harmful materials such as gases, dust, fumes and odor, which suspend in the atmosphere. Airborne dust is one type of air pollution composition which contains very small size of particles in the atmosphere. The airborne particles cause damage or an undesirable impact on living organisms such as food crops, or to the natural, built environment or may cause death or disease on humans and animals. In medical research, airborne dust particles are classified as particles that are less than 10 micrometers in diameter (PM_{10}). Airborne dust particles sizes, types and the hazard caused are clearly defined by World Health Organization (WHO) in [1]. Due to the huge increase in industry, businesses and human activities in recent years, there has been a need to monitor the air quality especially in industry and surrounded areas. Many countries have established regulations and standards for the control of airborne dust particle levels for this reason. Additionally, technology has been deployed widely in many industry areas in order to enforce meeting these regulations and adhering to the standards.

Given the above observations, a need to develop a novel automated/semi-automated computer vision based system to classify airborne industry dust particles is felt significantly and in a timely manner. Such a system should be capable of analysing dust particles present in air and should be able to recognise the type of dust thus making it possible for the authorities to predict/estimate the source. For example such a system can help the Ministry of Environment and Climate Affairs within Oman for more accurately monitoring the air quality in industrial and surrounding areas, such that legal action can be taken against the industry that is causing excessive levels of air pollution.

In this chapter a novel, automated, computer vision based, dust particle classification system that is based on colour and texture features is proposed. The system is designed, implemented and finally tested on real samples of dust collected within the Sohar city industrial region of Oman.

For clarity, this chapter is divided into several sections. Section 4.2 provides the practical background of this project and explains the potential real impact it could have on environmental pollution analysis. Section 4.3 introduces the reader to the proposed system's design and implementation. Section 4.4 presents the dust sample collection and image acquisition system used. Section 4.5

discusses the experimental results and carries out a comprehensive analysis of the performance of the proposed system. Section 4.6 finally summarises and concludes the chapter.

4.2 Background

The Sultanate of Oman is one of the Arab countries located in southwest Asia. According to the World Bank, 2015 [2] Oman's economy is considered as a high income economy based on oil and petro-chemical industry. The government of Oman has been aiming to divert the economy from an oil-dependent economy to an oil independent economy in order to have a sustainable economy [3] for the future. To achieve this goal, one of the initiatives launched by the government of Oman is a strategic plan, namely "Vision 2020" [4]. The development of the Sohar port is one of the government projects initiated to achieve this economic diversification goal. It is aimed to be developed to be the main industrial and commercial port of the Sultanate of Oman. Many industrial production plants have been established in the Sohar region in recent years such as oil refineries and petrochemical, plastic, rubber, ceramic, chemical, iron and steel processing plants, to name a few. Moreover, the Sohar city area has become a logistic hub of Oman and the adjoining Gulf countries which has made the government to invest heavily in providing different facilities such as buildings for business and dwelling, entertainments facilities and infrastructure for public transport such as highways and railway lines in order to attract investors and people who will be willing to work for industry. Since the initiation of the above project, there has been a significant increase in industrial/business activities and commercial/public transportation in the Sohar region. Additionally, the Sohar city region is surrounded by farms where people increasingly burn agricultural waste and other waste in open areas without any concern to the environment or subject to monitoring and control of the government authorities. Moreover, poor management of the drylands of Sohar has been causing periodic dust storms in the Sohar area, which has aggravated the situation as this causes breathing difficulty and effects on visibility, causing a saviour impact on the environment and the region's economy.

All above factors have caused significant levels of air pollution in the Sohar region and has resulted in public protests by the city's population against the sources of pollution demanding immediate action by the government and its regional authorities. As a result, a royal directive has been issued to the concerned authorities to relocate the people who have been severely affected by air pollution in the Sohar economic/port area. Additionally, the Ministry of Environment and Climate Affairs (MECA) of Oman in collaboration with The Research Council of Oman (TRC) and Sohar University has established the Environmental Research Centre (ERC) at Sohar University aimed at conducting fundamental and applied research into air pollution. The air quality management has become one of

the main objectives of the ERC of Oman. In 2015 Loughborough University signed a Memorandum of Understanding (MoU) with Sohar University to develop novel technologies to investigate and analyse the dust collected from Sohar industrial region, which resulted in the initiation of this research.

4.3 The Proposed System

Air pollution can be caused by various types of dust particles that originate from both natural and industrial processes. Identification of the level of concentration of the dust and the recognition of the type of dust will enable the subsequent determination of the dangers posed by the said particle in air pollution and the monitoring and control of their sources. The particles are usually of different size, shape, colour and texture. These properties can be used to attempt to separate the particles into recognisable groups. If known samples of the particles are available as training data or ground truth, one can use this information to classify the dust leading to the recognition of the type of dust.

Machine vision-based systems can be developed to analyse the industry dust based on colour, shape, size and texture. The proposed technique consist of four main stages namely particle segmentation, feature extraction, feature selection, and classification process. In the segmentation stage, the dust particles are segmented based on colour and texture. A novel algorithm is developed, by merging different segmentation techniques, in order to obtain accurate results and overcome segmentation limitations and complications of this type of dust. The block diagram below illustrates the logical flow of the proposed system.

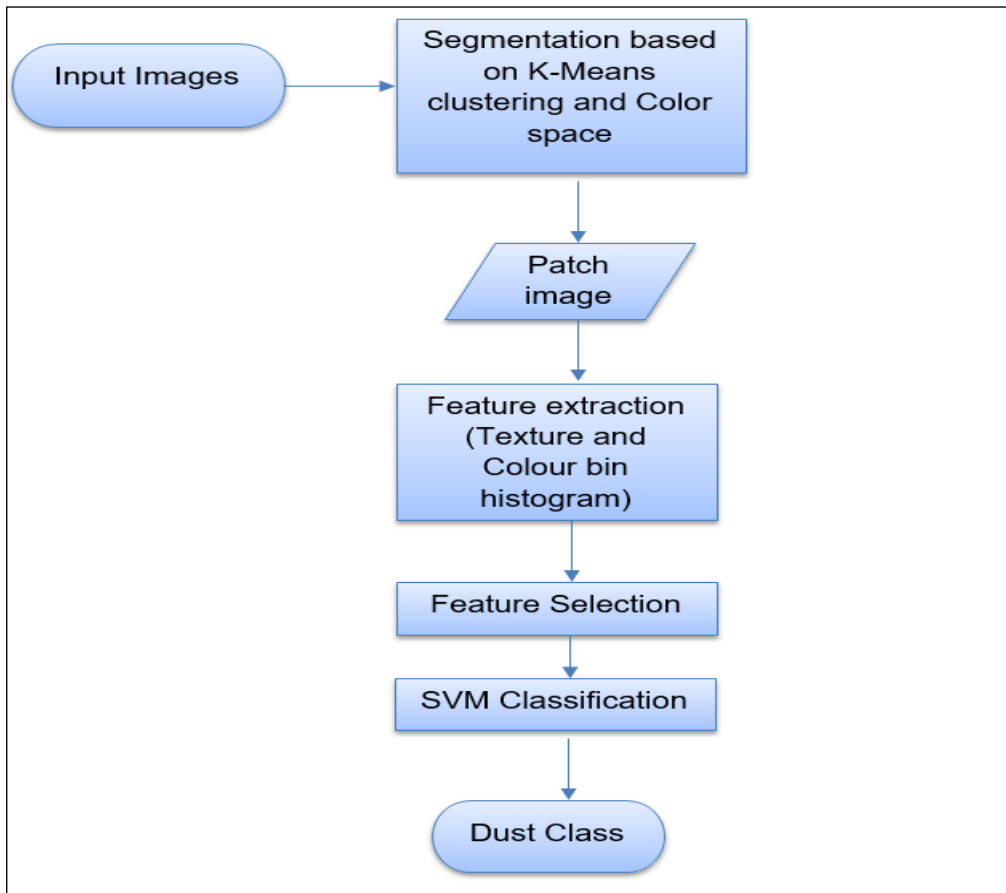


Figure 4-1 The proposed system.

4.3.1 Dust sample collection and image acquisition

The experiments were conducted for the analysis of airborne dust samples which were collected from the Sohar industrial area of Oman. The dust samples were collected using three different systems.

- 1- Collecting dust on a clean upright container. This is the most basic system that requires the collected dust to be brought to the laboratory and manually investigated under a microscope.
- 2- Collecting dust on a sticky take that is wrapped around a cylindrical drum. These sticky tapes subsequently have to be viewed under a microscope for manual inspection.
- 3- Collecting dust on circular filter paper through which a known quantity of air is passed under pressure. It is noted that this is an industry standard approach used in many countries.

In the research conducted within this thesis, all samples obtained for experiments were captured based on the first approach. Insufficient samples obtainable using the second and third approaches led to the exclusion of these two capture approaches.

Once the dust samples are collected, a digital microscopic camera is used to take an image of the dust sample, at three different spread densities (on a white sheet of paper used as the background). The dimensions of captured microscopic images were, 1280X960 pixels. All dust samples were collected from Sohar industry area in AlBatinah Region, Oman. Images were captured in JPEG format. For clarity representation, the image sample captured are categorised into three different categories (high, medium and low density particle spread) based on their visible density of particles, due to the practical image acquisition carried out.

4.3.2 Segmentation

Initially, the sample images are segmented to obtain the foreground objects, i.e. the dust particles which can be subsequently analysed to determine their type, hence resulting in their classification. Two different methods were considered to determine the best algorithm to be used for dust particle segmentation. The first experiment was based on RGB colour histogram based segmentation. The second experiment was conducted to separate the particles based on particles' colours using different colour space namely RGB, L^*A^*B and HSV and K-means Clustering. [Note: Once the segmentation task is completed the dust samples can be classified into different types, 8 classes in this case, code named as A, B, C, D, E, F, G and H (see section 4.3.3)].

4.3.2.1 *Histogram-based segmentation*

A colour histogram is a very important method that can be used to represent or visualise large image data sets graphically. Additionally, they help to indicate where the major colour values of an image fall in a measurement scale and within a variation level. The Figures 4-2 - Figure4-4 below show how the colours of dust particles are typically being distributed among three different RGB colour channels, for three different densities of dust particles, that results from three different capture attempts.

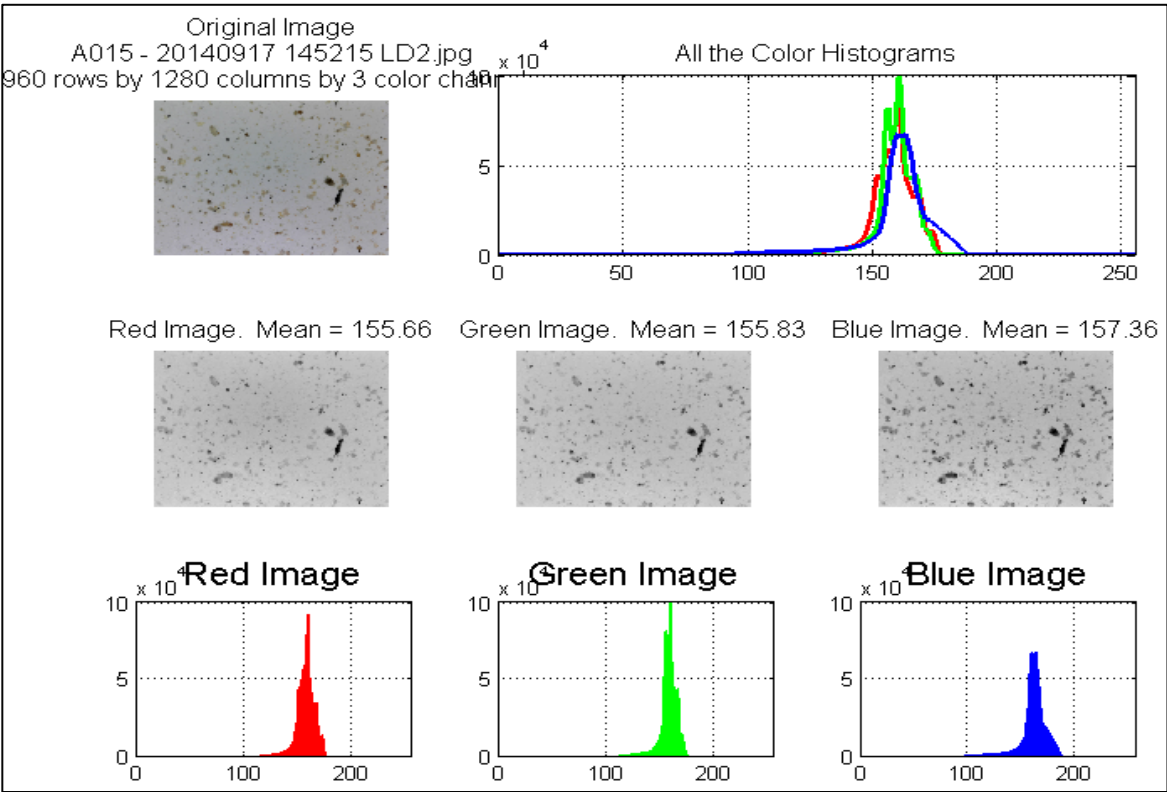


Figure 4-2 Histogram for low density image.

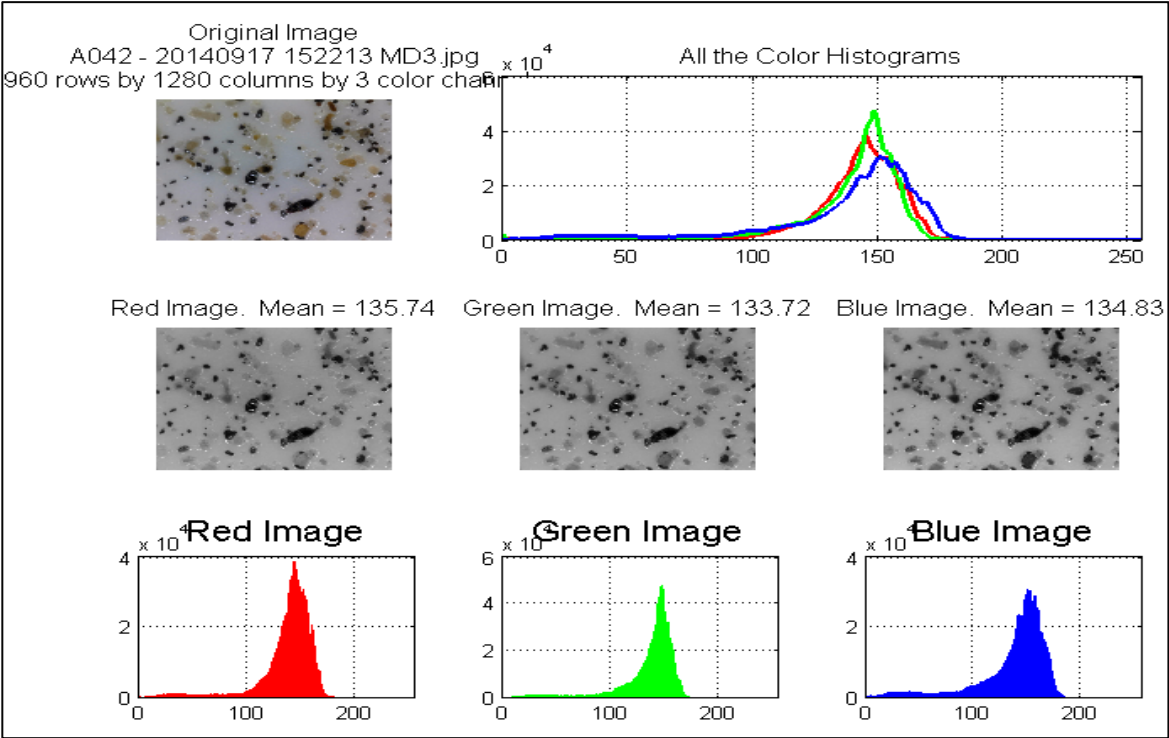


Figure 4-3 Histogram for medium density image.

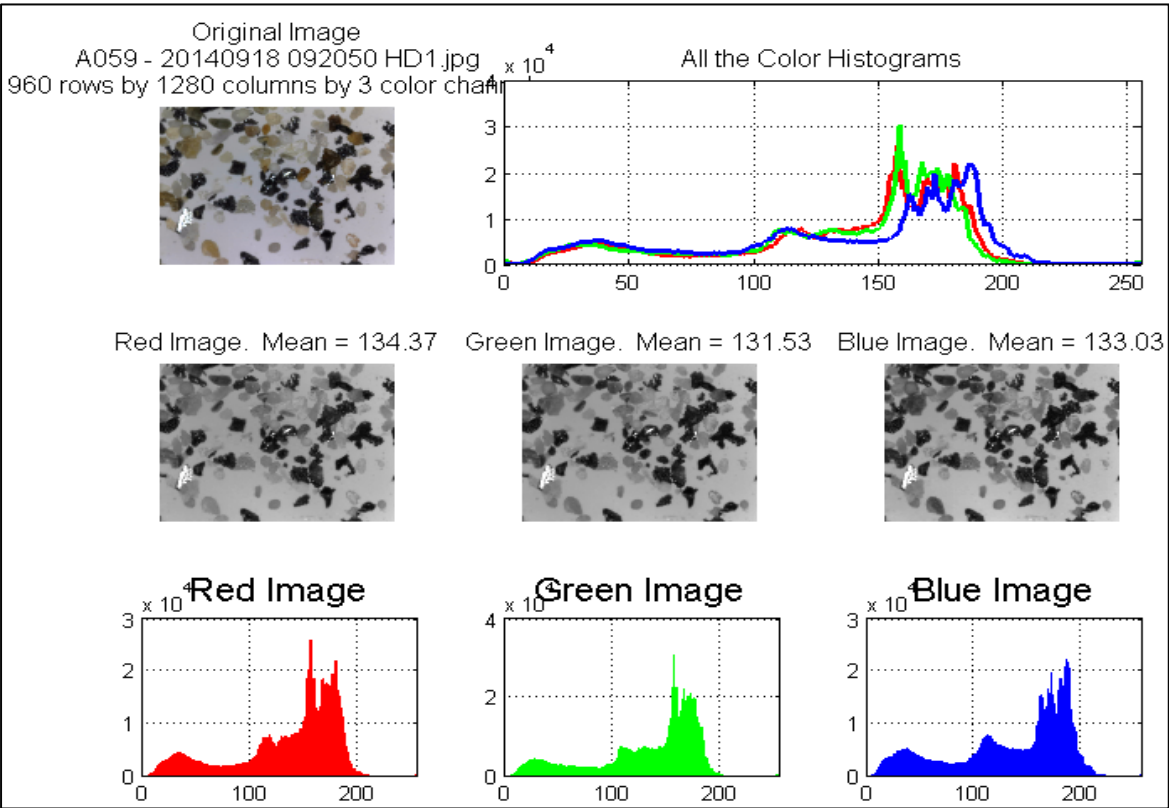


Figure 4-4 Histogram for high density image.

4.3.2.1.1 Experimental Analysis:

A close analysis of the individual R, G, B histograms of Figures 4-2, Figure 4-3 and Figure 4-4 reveal that the capture density of particles have a significant impact on the ability to use the histograms in any form of subsequent classification. Whilst at low particle density capture, the peaks are more distinguishable at the highest density the graphs are spread with multiple local peaks. At all densities the peaks are at similar values, making it difficult to use R, G, and B colour channel values for subsequent classification.

A conclusion of this experiment is that R, G, B colour space is not suitable for dust sample classification. Further the use of colour histogram based thresholds for the individual R, G, B values will also not be a viable option for classification as separating out the individual colour thresholds will be difficult due to their similarity in value. Different dust samples having different colour spreads can only make these challenges more demanding to meet.

Therefore a decision was made to consider the use of other colour spaces which are known in many other application domains to have distinguishable features that can be used in colour based image classification. Further the use of K-means clustering approach as a means of classification was to be considered.

4.3.2.2 Colour Segmentation using K-Means Clustering and Different Colour Space

K-means clustering is considered as a segregation method of data-clustering. In other word, it is a method that can be used to separate the objects based on multiple, different features such as for e.g., colour, object area, etc. The main objective of this experiment is to segregate dust particles into separate type of clusters based on colour. To compare the results, the use of K-means clustering in three different colour spaces has been investigated.

The approach for dust particle segmentation involves six steps. They are: reading image; colour space conversion; classifying the dust particles using colour space features and K-means clustering; labelling every pixel in the image; segment and display the dust particles. The tasks that are carried out within these steps can be elaborated as follows:

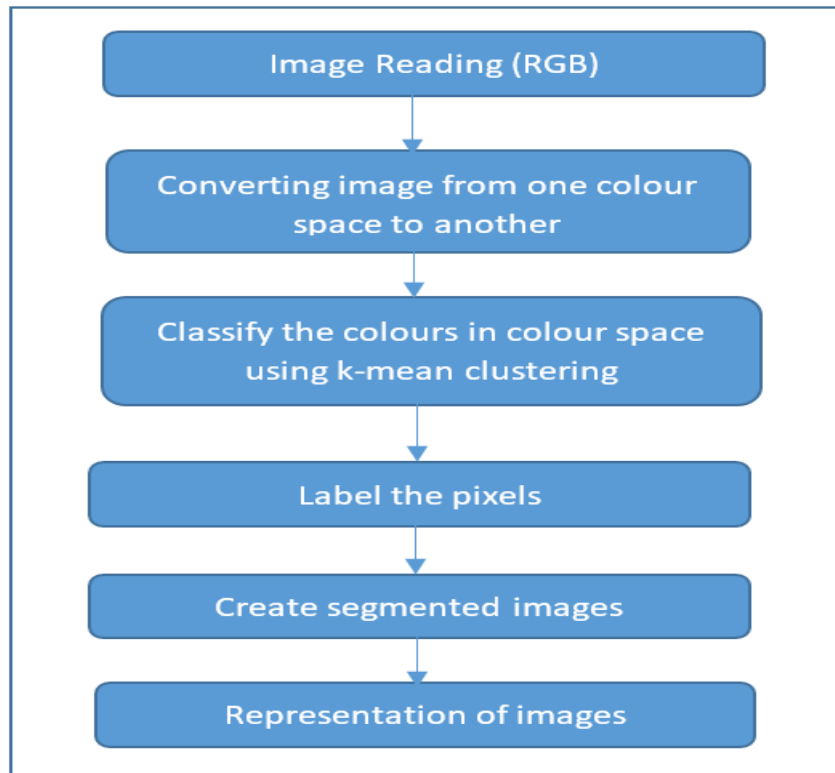


Figure 4-5 image segmentation steps using K-means clustering.

Step 1: Image Reading: In this step image of the dust sample (with a white background) is decoded and read so that the RGB values for each pixel location will be available for the subsequent step.



Figure 4-6 RGB input image.

Step 2: The input image originally read as an RGB image will be converted to the colour space of which the colour components are to be used in clustering. Standard colour space conversion equations will be used for this purpose.

Step 3: In this step, K-means clustering is used to classify each type of dust based on colour. It is noted that the number of clusters to be partitioned (i.e. K) needs to be specified.

Step 4: Every pixel in input image will be labelled with the relevant cluster index which results from K-means clustering.

Step 5: Based on pixel labelling done in the step-4, sub images are created depicting different types of dust.

Step 6: In this step the segmented images will be displayed along with input image.

The result of the above steps will be analysed in detail before applying pixel value based thresholding for the removal of insignificant objects and any morphological operations to remove noise.

4.3.2.2.1 Colour segmentation using K-Means Clustering and RGB Colour Space

In this experiment the RGB colour space is used with K-means clustering to segment the dust samples. RGB is the colour representation of the originally captured images that are read as input and hence no colour conversion is needed.

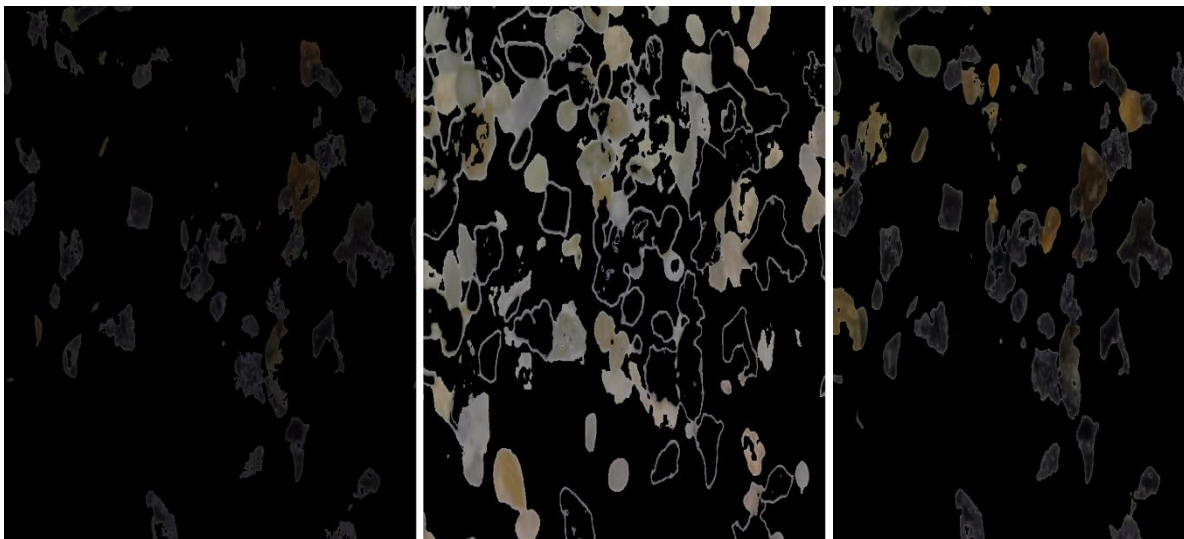


Figure 4-7 Black particles.

Figure 4-8 Light brown particles.

Figure 4-9 Brown particles.

4.3.2.2.2 Experimental Analysis

The results of this experiment show that in attempting to detect, segment, label and hence cluster particles of differing colours, colours are often mixed together. For instance, both Figure 4-7 and Figure 4-9 contain both black and brown objects. Therefore the RGB colour space based approach has failed to separate and cluster dust particles based on the particle's colour. The result clearly indicates that RGB colour space is not a suitable colour space to be used for this purpose and proves the previously published views of Mathur and Purohit in [8] in the area of dust particle clustering. Therefore, this approach was not considered to be taken to the next processing level, which is thresholding and morphological operations (see section 4.3.2.2).

4.3.2.2.3 Colour Segmentation Using K-Means Clustering and LAB Colour Space

In this experiment, input image which is in RGB colour space is first converted into LAB colour space. This colour space has three layers. It uses L for luminosity or lightness and A and B for chromaticity layers. All the colours falling between red to green axis will be shown in A chromaticity while B chromaticity layer contains colours falling between blue to yellow axes. The experiment was conducted for three different dust particle densities (low, medium and high). The result of using LAB colour space for dust particle segmentation using K-Means clustering is illustrated below in Figures 4-10 – Figure 4-21. The attempt made was to separate particles into three clusters, black, brown and all other coloured particles different to the image background.

A. Low Density.

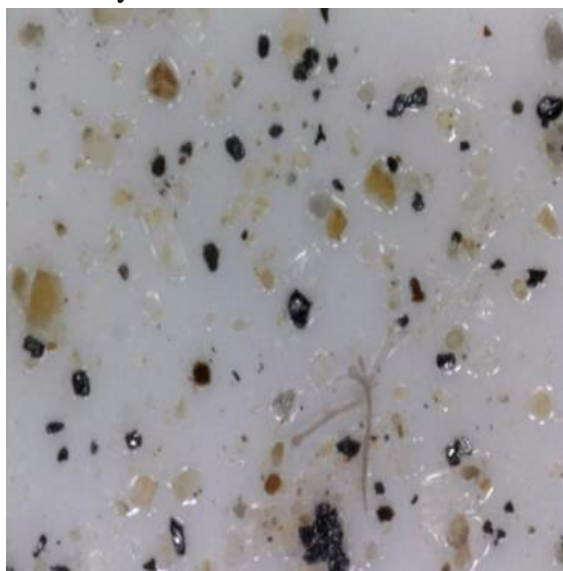


Figure 4-10 Low density original image.



Figure 4-11 Low density segmented image - black objects.

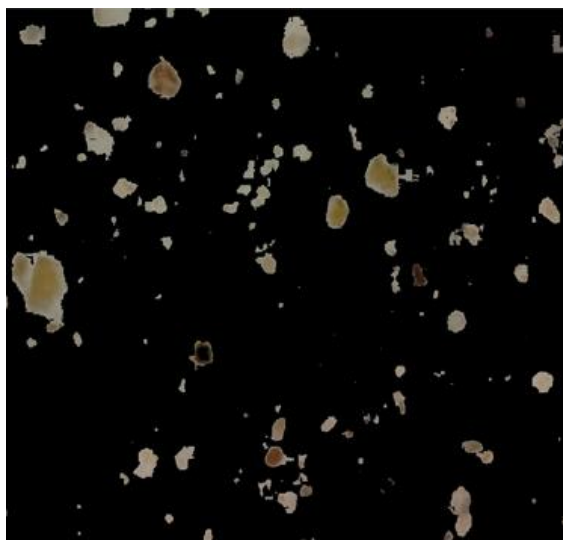


Figure 4-12 Low density segmented image - brown objects.

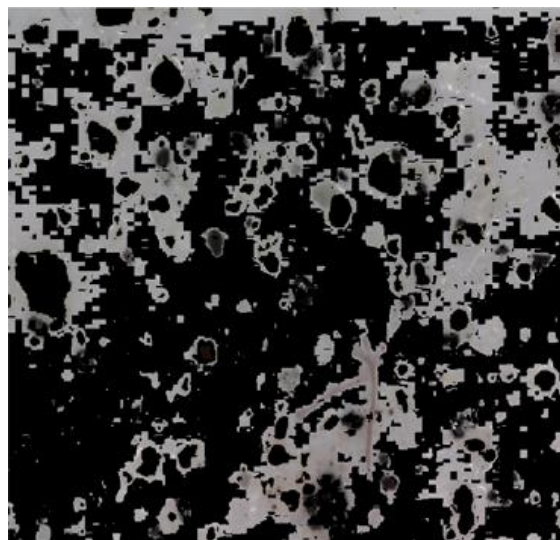


Figure 4-13 Low density segmented image – others.

Medium Density.

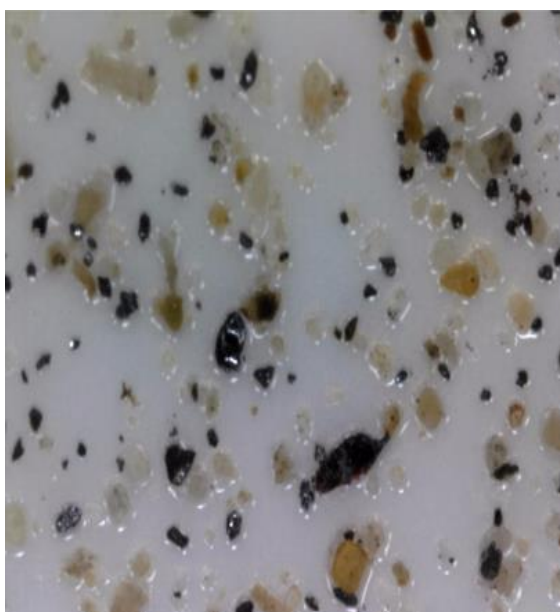


Figure 4-14 Medium density original image.

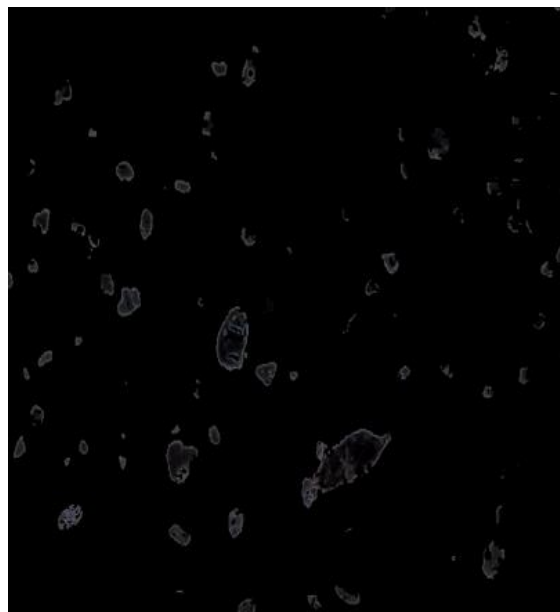


Figure 4-15 Medium density segmented image - black objects.

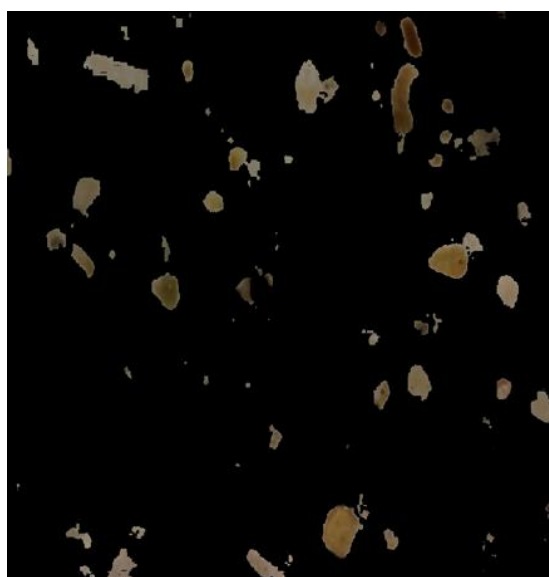


Figure 4-16 Medium density segmented image - brown objects.



Figure 4-17 Medium density segmented image - others.

B. High Density.



Figure 4-18 High density original image.

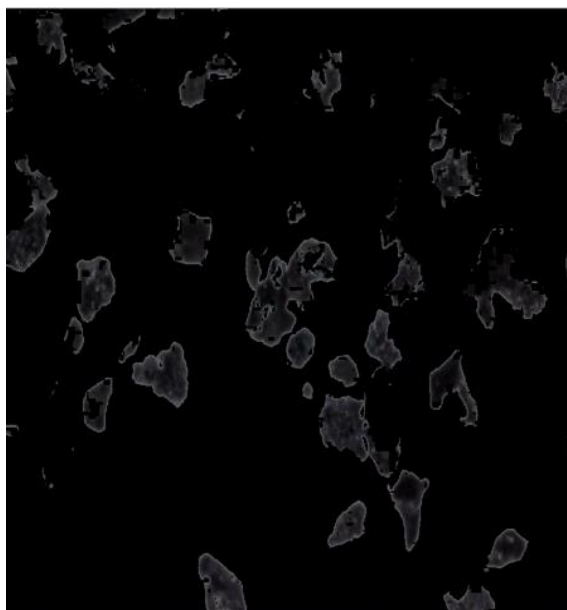


Figure 4-19 High density Segmented Image - black objects.



Figure 4-20 High density segmented image - brown objects.

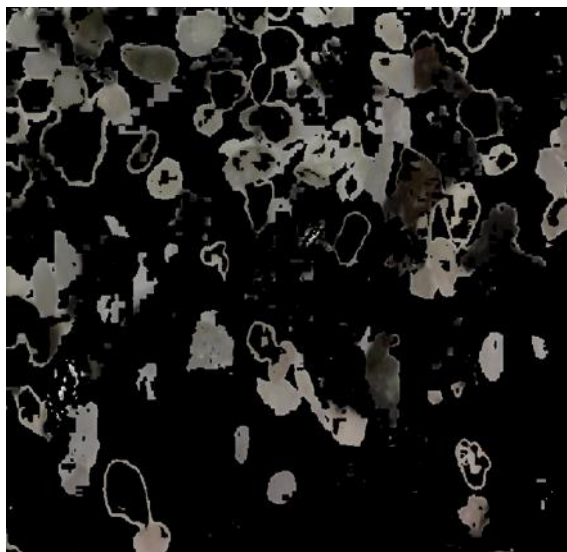


Figure 4-21 High density segmented image - others.

4.3.2.2.4 Experimental Analysis:

A close analysis of the results above illustrate that the use of the LAB colour space makes it possible to separate dust particles into three categories based on their colour, namely black, brown and others. Unlike RGB colour space, black objects were filtered without being mixed with any other colour. Additionally, brown objects contain light and dark brown but do not contain particles of other colours, which is what was intended. However the LAB the approach failed to represent the third segmented colour in a better way as it does for black and brown colour s. This can be clearly seen in Figures 4-13, Figure 4-17 and Figure 4-21 although it still shows a better result compared with results of using the RGB colour space.

4.3.2.2.5 Colour Segmentation using K-Means Clustering and HSV Colour Space

In this experiment, input RGB image is converted into Hue, Saturation and Value (HSV) colour space. This colour space categorises the colours based on their shade and brightness. Hue is expressed as a number between 0 to 360 degrees. This channel represents colour/hue of red, yellow, green, cyan, blue and magenta colours. The Hue fall values range between 0 to 360 degrees. The level of Gray is represented within the Saturation channel, whereas brightness or intensity of the colour is represented within the Value channel. The experiment was conducted only for high level of dust particle densities as this is where the technical challenge is highest and particles of different colours are more likely to be present. The results of using the HSV colour space show that extra colours were able to be identified and extracted as compared to the number of colours extracted when using the LAB colour space (section 4.3.2.2.1 and section 4.3.2.2.3). Figures 4-22 to Figure 4-25 illustrates the segmentation results achieved.

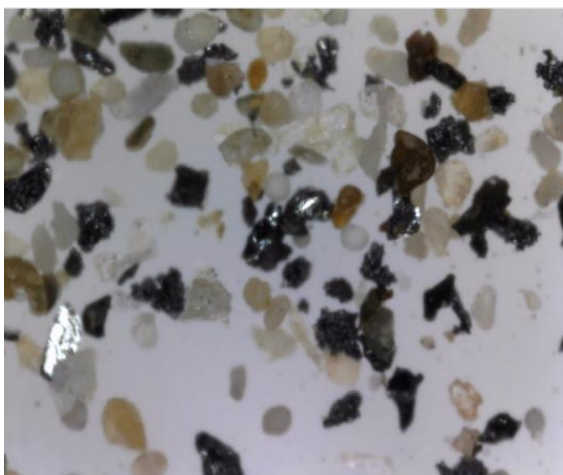


Figure 4-22 High density original image.

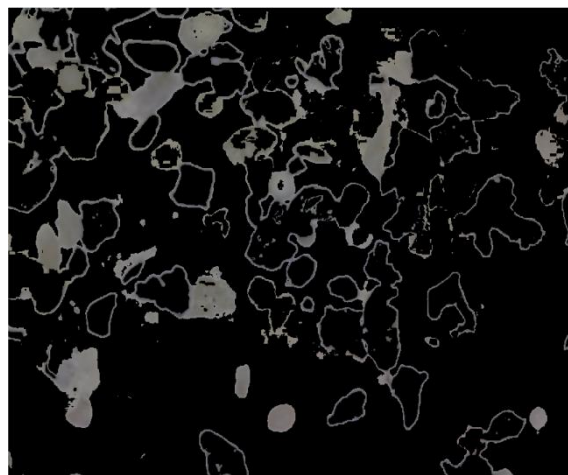


Figure 4-23 High density Extra Colour A.

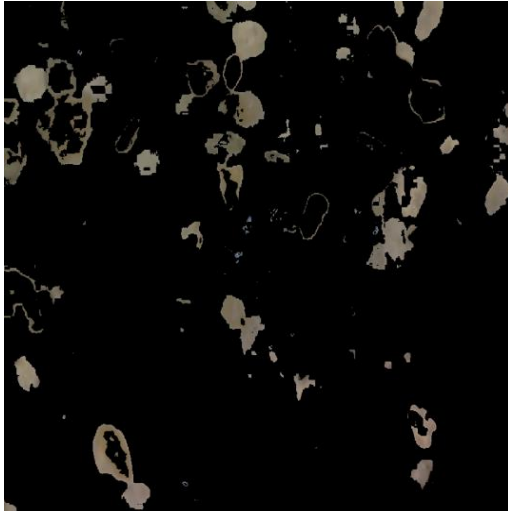


Figure 4-24 High density original image B.



Figure 4-25 High density Extra Colour C.

4.3.2.2.6 Experimental Analysis:

The above experimental results show that the use of HSV colour space has resulted in better dust particles segmentation as compared to using the LAB and RGB colour spaces. The HSV based approach was able to separate the objects of a larger amount of colours (i.e. types of dust), more accurately. Based on this result, it was decided to use the HSV colour space in the further refinement of the segmentation process and further experiments presented in section 4.3.3 and beyond in this chapter.

4.3.3 Post Processing

It is also observed that as a result of K-means clustering, the output contains small holes and the appearance of the particle boundaries needs to be smoothed to remove impact of noise that has affected the colour based segmentation process. Therefore, the thresholding and morphological operations are applied as post-processing stages to the colour based segmentation. It is noted that the Huang method [119] is used as the thresholding method. The Figures 4-26 – Figure 4-28 illustrate the outcome of applying these post processing stages to the segmented regions of ‘black’ particles detected at three different particle densities.

A. High Density:



Figure 4-26 High Density - black object.

B. Medium Density:

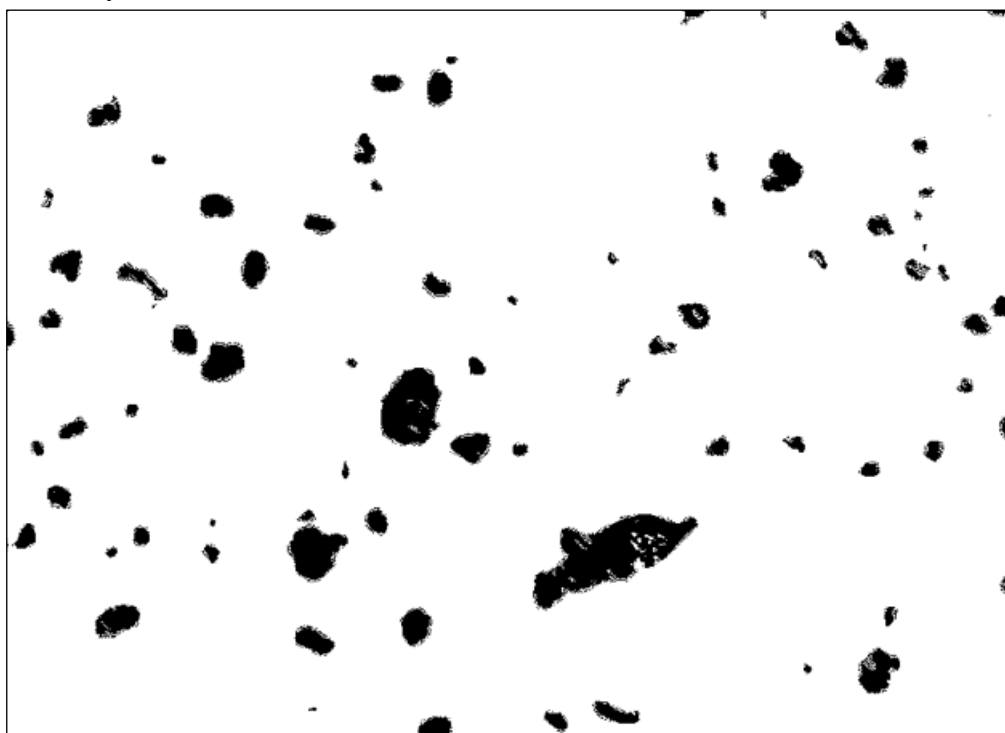


Figure 4-27 Medium Density - black object.

C. Low Density:



Figure 4-28 Low density - black object.

As a result of all segmentation processes, the images have been categorized into 8 classes based on colour and texture. Due to unavailability of ground truth as it needs chemical composition analysis for these images, we have labeled them as class A, class B, class C, class D, class E, class F, class G and class H and used as ground truth in all the experiments.

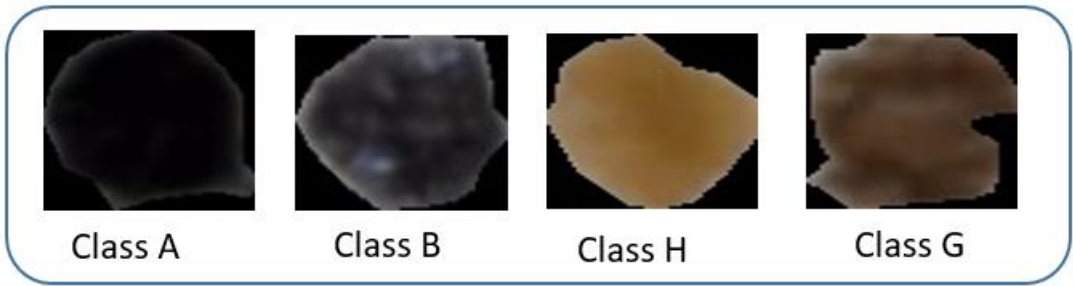


Figure 4-29 Example of image category.



Figure 4-30 Example of image patches.

4.3.4 Feature Extraction

A substantial number of texture and colour features were extracted from all images. These features are used in machine learning for training and testing purposes. The features are extracted as follows:

- 1- Hue Colour histogram: there are 256 bins which are obtained from the hue channel of HSV colour space.
- 2- Gray Level Co-occurrence Matrix (GLCM) texture Features: initially, a GLCM has been created to each patch by applying “graycomatrix” function in MATLAB in order to examine texture. Twenty two features of statistical textures were obtained from GLCM for each image.

These features are listed follow:

- Entropy, sum of entropy, Sum of average, Sum of squares, Sum of variance, contrast, Difference of entropy, Correlation, Information measure of correlation, Energy, Contrast, Information measure of correlation 2 as in [38];
- Entropy, Cluster prominence, Homogeneity, Autocorrelation, Cluster shade, Dissimilarity, Max probability as defined in [39].
- Inverse difference normalised, Inverse difference moment normalised as in [40].
- Homogeneity, Correlation as in [41].

All the features which were extracted from colour and texture were combined together for the purpose of appearance based categorisation of the images. This gives us as a total of 278 features to recognise the particles. A close visual inspection of images shown in Figure 4-29 and Figure 4-30 indicates the similarity in colour but different in patterns between class A and class B, and class H

and class G. Therefore, the combined features of colour and texture should provide a reasonable accuracy for all the classes.

4.3.5 Feature Selection

Feature selection has become an essential data processing step prior to applying the learning algorithm. Combination of colour and texture features that were presented in section 4.3.4 had 278 features to be used within one feature vector. This large number of features has an immediate effect on the processing time and recognition accuracy. It has been noted that removal of irrelevant and redundant information using feature selection method can help to improve machine learning accuracy [32]. Therefore, only the more discriminative features have been selected from feature vector to improve the recognition accuracy and processing time. To select the most effective features, two feature selection approaches can be utilised, namely; filter based and wrapper based approaches [43]. Based on the investigation carried out within this research, the wrapper based approach has performed better than the filter based approach. Thus, the wrapper based approach has been selected as feature selection algorithm for all the experiments in this chapter. The wrapper based approach uses a subset evaluator to create all possible subsets from the features vector. Then, it induces a classifier to each feature subset before it considers the subset with which classification algorithm performs best.

4.4 Experimental Setup

Three different software packages have been used in the proposed experiments, namely, MATLAB, GIMP image manipulation and Weka. The PC configuration used for all experiments is an Intel i5 dual core processor 3.2 GHz, with 8 GB RAM memory.

An experimentation model was designed to evaluate the performance of different classifiers as illustrated in the figure below.

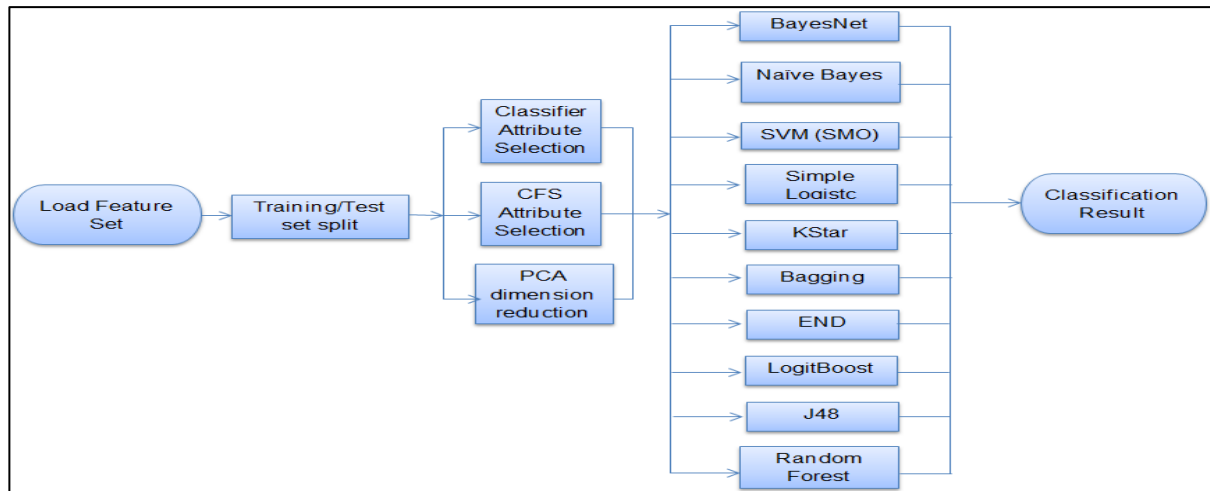


Figure 4-31 Classifiers / feature selection algorithms, comparison model.

The model has means to investigate three different feature elimination techniques; Principal Component Analysis (PCA) [120], Correlation-based Feature Selection (CFS) [121], and Classifier based attribute selection. The experiment setup can also investigate ten classifiers namely; BayesNet, NaiveBayes, SMO, SimpleLogistic, KStar, Bagging, End, LogitBoost, J48 and RandomForest [122]. The table below shows the results obtained for the feature selection experiment.

Classifier Name	Wrapper Based	Filter Based	
	Classifier Attribute Selection	CFS Attribute Selection	PCA
BayesNet	84.14%	89.02%	76.82%
NaiveBase	90.24%	89.02%	60.97%
SMO	92.68%	86.58%	60.97%
SimpleLogistic	90.24%	90.24%	91.46%
KStar	53.65%	81.70%	68.29%
Bagging	82.92%	82.92%	68.29%
END	84.14%	86.58%	81.70%
LogitBoost	90.24%	90.24%	79.26%
J48	78.04%	84.14%	68.29%
RandomForest	92.68%	86.58%	86.58%

Table 4-1 Model classification accuracy,

It can be clearly seen from the above table that the RandomForest classifier with the wrapper based attribute selection approach performs better than any other classifier-attribute selection algorithm

combination. Therefore the RandomForest classifier and classifier based attribute selection method have been adopted for all other experiments in this chapter.

The specific parameters used for RandomForest and Classifier based feature selection algorithms are:

A) Classifier based selection:

- Attribute evaluator: ClassifierSubSetEval.
- Search Method: Bestfirst –D 1 –N5.

b) RandomForest classifier:

Parameter Name	Parameter Value
Debug	False
maxDepth	0
numFeatures	0
numTrees	100
Seed	1

Table 4-2 RandomForest classifier parameters.

The dataset used in all experiments contained 240 images in total representing 8 different classes (30 images for each class). The dataset has been separated into 34 % for training, and remaining 66% for the testing purposes.

4.5 Experimental Result and Analysis

Numerous experiments were conducted in order to analyse the performance of different classifiers. Accuracy is the main measurement which was used to measure the performance of different classification algorithms. It is based on rate of True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) of analysed data. The accuracy is defined as:

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \cdot 100 \%$$

Equation 4-8

Initially, a dataset of 22 statistical texture features (i.e. only texture features) was used to classify the 8 different dust particles classes. All classifiers (see section 4.4) and texture features were used in this experiment to compare the performance of each classification algorithm. Table 4-3 illustrate the result obtained from texture based dataset for each classifier and Table 4-4 shows the detailed accuracy for best algorithm performance, i.e. LogitBoost.

Bays Net	Naive Base	Simple Logistic	SMO	Kstar	Bagging	END	Logit Boost	J48	Random Forest
63%	56%	67%	64%	54%	57%	63%	70%	63%	65%

Table 4-3 Texture based dataset accuracy.

Whole Features						
TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.667	0.014	0.889	0.667	0.762	0.904	A
0.818	0.028	0.818	0.818	0.818	0.951	B
0.714	0.053	0.556	0.714	0.625	0.97	C
0.333	0.075	0.5	0.333	0.4	0.827	D
0.9	0	1	0.9	0.947	0.989	E
1	0.013	0.875	1	0.933	0.989	F
0.818	0.099	0.563	0.818	0.667	0.918	G
0.667	0.055	0.6	0.667	0.632	0.945	H
0.707	0.044	0.717	0.707	0.702	0.926	Average

Table 4-4 Texture based dataset detailed accuracy for LogitBoost classifier.

The above results indicate the low classification accuracy obtained for all classification approaches when only texture features are used as the highest accuracy result achieved is only 70%. Although, feature selection can improve the accuracy marginally, it failed to improve the overall accuracy to satisfactory level. The best result obtained was 73% accuracy, when 8 most discriminant features were separated using a wrapper based feature selection approach. The accuracy achieved using filter based approaches to feature selection was not better than 70%.

A further experiment was conducted using only 256 colour features within a feature vector. Both wrapper and filter approaches were used for feature selection investigations. The result obtained are tabulated below:

Colour based dataset accuracy		
Wrapper approach	Filter approach	
	CFS	PCA
75 % (17 features)	79 % (24 features)	69% (47 features)

Table 4-5 Colour based dataset accuracy.

The best accuracy recorded for classifier based feature selection was 75% using 17 selected features. As far as filter based feature selection approaches are concerned, CFS had best result which was 79% accuracy using 24 selected features. Although when using colour features instead of texture features the classification results show some performance improvement, an acceptable level of accuracy has not been achieved.

Even though increasing the data in training and testing set could improve the accuracy, a significant improvement cannot be expected using texture or colour features individually.

The third experiment conducted concatenated and applied both colour and texture features. This combination gave a total of 278 features to be used in one feature vector to recognise the particles. The experiment uses the same setup explained in section 4.4. The figure below summarises the accuracy results obtained when only colour, only texture and combination of texture and colour features were used, respectively.

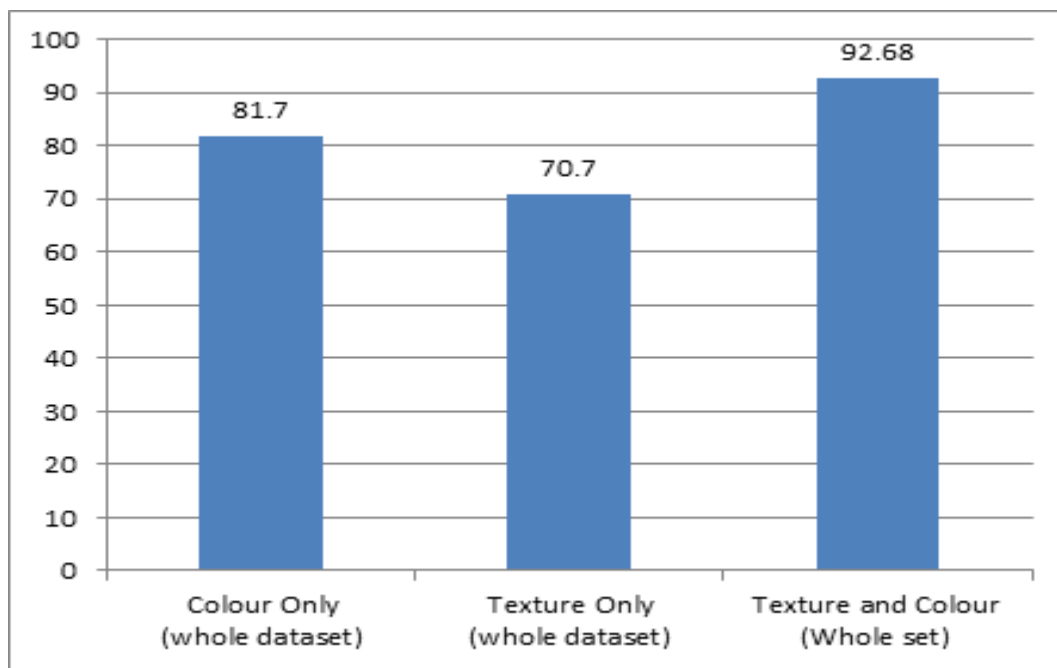


Figure 4-32 Accuracy summary for all features.

The results indicate a significant improvement compared with previous experiments' results. As far as the classification algorithms are concerned, the RandomForest classifier obtained the best percentage accuracy and performed marginally better than BayesNet, END and Sequential Minimal Optimization (SMO) classifiers. Table 4-6 shows the accuracy obtained for all of the different classifiers, whereas Table 4-7 illustrates the detailed accuracy obtained for the best classifier.

Bays Net	Naive Base	Simple Logistic	SMO	Kstar	Bagging	END	Logit Boost	J48	Random Forest
91.46 %	84.14 %	84.14 %	91.46 %	80.48 %	86.58 %	91.46 %	89.24 %	82.92 %	92.68 %

Table 4-6 Colour and texture dataset accuracy for all features.

Whole Features						
TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.833	0.019	0.862	0.833	0.847	0.989	A
0.833	0.014	0.893	0.833	0.862	0.993	B
1	0	1	1	1	1	C
0.967	0.014	0.906	0.967	0.935	0.995	D
0.967	0	1	0.967	0.983	1	E
0.9	0.01	0.931	0.9	0.915	0.994	F
1	0.005	0.968	1	0.984	1	G
0.933	0.019	0.875	0.933	0.903	0.989	H
0.929	0.01	0.929	0.929	0.929	0.995	Average

Table 4-7 Colour and texture dataset, detailed accuracy for RandomForest classifier using all features.

The above detailed accuracy table for RandomForest algorithm indicates recognition weakness for some classes as compared to the others. For example classes C, E and G appear to be recognised more accurately as compared to the others. For further detailed analysis, the confusion matrix can be used (see Table 4-8).

Whole Features								
a	b	c	d	e	f	g	h	Classified as
25 (83.3%)	3 (10%)	0	2 (6.66%)	0	0	0	0	a=class A
4 (13.3%)	25 (83.3%)	0	0	0	0	1 (3.3%)	0	b=class B
0	0	30 (100%)	0	0	0	0	0	c=class C
0	0	0	29 (96.66%)	0	0	0	1 (3.3%)	d=class D
0	0	0	0	29 (96.66%)	1 (3.3%)	0	0	e=class E
0	0	0	0	0	27 (90%)	0	3 (3.3%)	f=class F
0	0	0	0	0	0	30 (100%)	0	g=class G
0	0	0	1 (3.3%)	0	1 (3.3%)	0	28 (93.3%)	h=class H

Table 4-8 RandomForest confusion matrix using all features.

Results for both class A and B indicate higher level of confusion as compared to the other classes. A closer analysis of the confusion matrix indicates that of the 30 class A samples, only 25/30 particles were accurately predicted as class A, 3 of the particles were misclassified as class B (10%) and 2 particles misclassified as D class. Same numbers of particles were misclassified in class B making 13.3% classified as A and 3.3% misclassified as class G.

Receiver Operating Characteristic (ROC) curves for five selected best performing classifiers have been plotted in Figure 4-33.

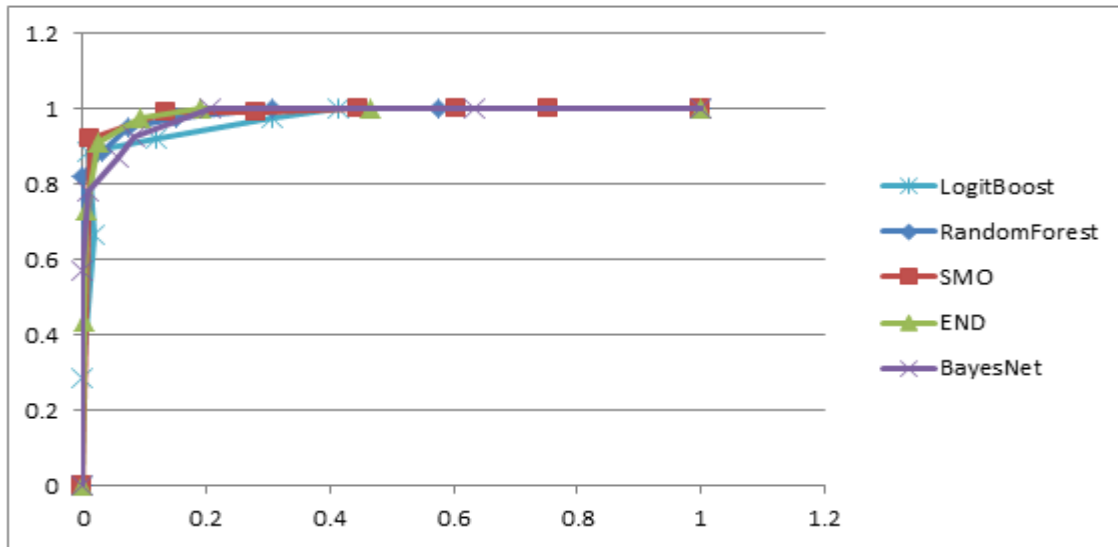


Figure 4-33 ROC curve for best classifiers using all features.

Attribute section Method	Bays Net	Naive Base	Simple-Logistic	SMO	Kstar	Bagging	END	Logit-Boost	J48	Random-Forest	Average
CFS (Filter)	89%	89%	90.2%	86.5 %	81%	82%	86%	90.2%	84%	86%	86.4 %
27 Features											
PAC (Filter)	76.8 2%	60.97 %	91.46%	60.97 %	68.2 9%	68.29%	81.7 0%	79.26 %	68.2 9%	86.58%	74.2 %
48 Features											
Classifier based (Wrapper)	84.1 4%	90.24 %	90.24%	92.68 %	53.6 5%	82.92%	84.1 4%	90.24 %	78.0 4%	92.68%	83.9 %
	10	17	9	23	3	10	6	12	14	10	
Features											
Average	83.3 %	80%	90.6%	80%	67.6 %	77.7%	83.9 %	86.6%	76.8 %	88.4%	

Table 4-9 Colour and texture dataset accuracy for selected features.

To optimise the performance and speed up processing time, feature selection techniques can be applied to the collective dataset. Therefore further experiments were conducted to analyse the use of reduced, combined colour and texture features in predicting the classes. Wrapper and filter based feature selection approaches were used with each classifier to compare the performance of all classifiers under feature selection (see section 4.4). The result of this experiment is tabulated in Table 4-9.

Two methods of filter based feature selection approaches were tested, namely CFS and PAC. Although, PAC ends up using more features for classification, than CFS, the accuracy rates obtainable with CFS is better for all, but the RandomForest classifier. The best result recorded using CFS method was 90.2% using 27 features of colours and texture for SimpleLogistic and LogitBoost algorithms. These features are given in Table 4-10. Around 91.5% was recorded as best result for PAC method. Unlike the filter based approaches, the number of features utilised were differed for each classifier in the wrapper based feature selection approach used in this experiment. Classifier based method selects the most relevant optimum number of features for each classifier. As far as accuracy is concerned, no improvement has been noticed between using all features or selected features. However, the proposed technique has maintained the same percentage accuracy values with less number of features. It can be concluded that wrapper based feature selection approach is more efficient than filter based feature selection approaches as with the reduced feature sets the latter results in a reduction of accuracy as compared to when using all features for all tested classifiers.

Feature Name	Feature Type
Autocorrelation	Texture
Contrast	
Correlation	
Correlation [1,2]	
Cluster Prominence	
Cluster Shade	
Entropy	
Sum of squares	
Sum average	
Sum variance	
Information measure of correlation1	
bin6	Colour
bin12	
bin13	
bin14	
bin17	
bin18	
bin22	
bin29	
bin36	
bin47	
bin160	
bin163	
bin179	
bin180	
bin188	
bin235	

Table 4-10 CFS selected Features.

Results in Table 4-9 illustrate that both SMO and RandomForest results in a very good accuracy rate of 92.68%. However the RandomForest classifier uses less than half the number of features selected by the SMO classifier. This makes RandomForest the best classifier as the reduced set of features will make training faster and computational cost low. The wrapper method when using the RandomForest classifier selected two texture features and eight colour bins giving a total of 10 selected features. The table below shows the features select and considered to be the most relevant features for classification with RandomForest classifier.

Feature Name	Feature Type
Sum variance	Texture
Information measure of correlation 2	
Bin4	
Bin7	Colour
bin13	
Bin36	
Bin40	
Bin94	
Bin188	
Bin235	

Table 4-11 Wrapper approach selected Features.

For further analysis, ROC curves have been plotted in Figure 4-34 for the best performing classifiers under feature selection.

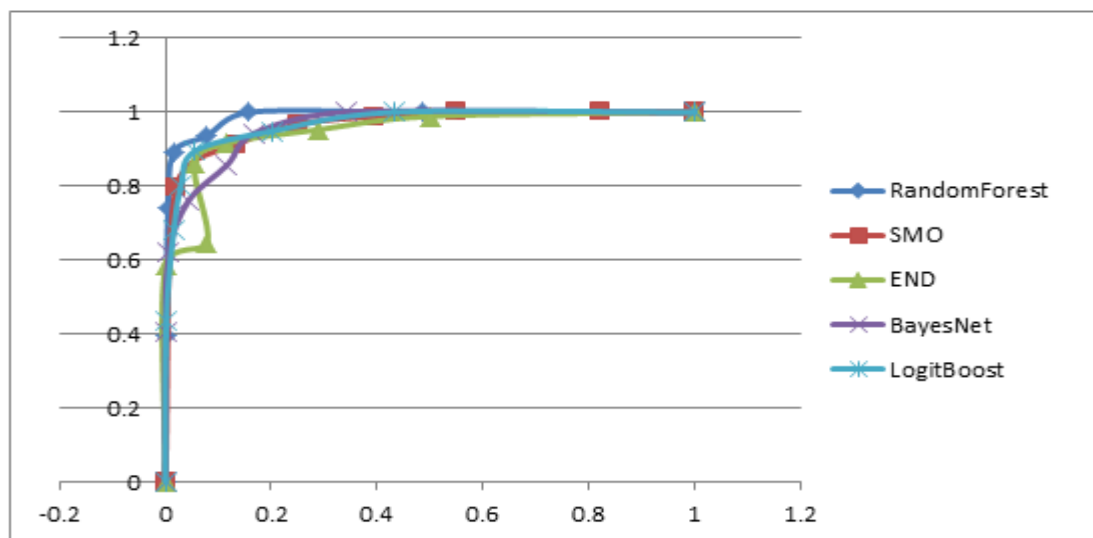


Figure 4-34 ROC curve for classifiers using selected features.

It was discussed before that both SMO and RandomForest achieved accuracy values of 92.68%. However, ROC curves show that RandomForest performs better than the SMO and all other algorithms used in this experiment. Another interesting metric that can be used to evaluate the performance of algorithms is the Area under the Curve (AUC). Table below demonstrates the percentage of AUC for all algorithms used whereas Figure 4-35 shows the percentage of AUC for each class per classifier.

Bays Net	Naive Base	Simple-Logistic	SMO	Kstar	Bagging	END	Logit-Boost	J48	Random-Forest
96.8%	98.1%	97.2%	96.8%	96.8%	98.6%	96.4%	97.2%	92.5%	98.7%

Table 4-12 Classifiers AUC percentage.

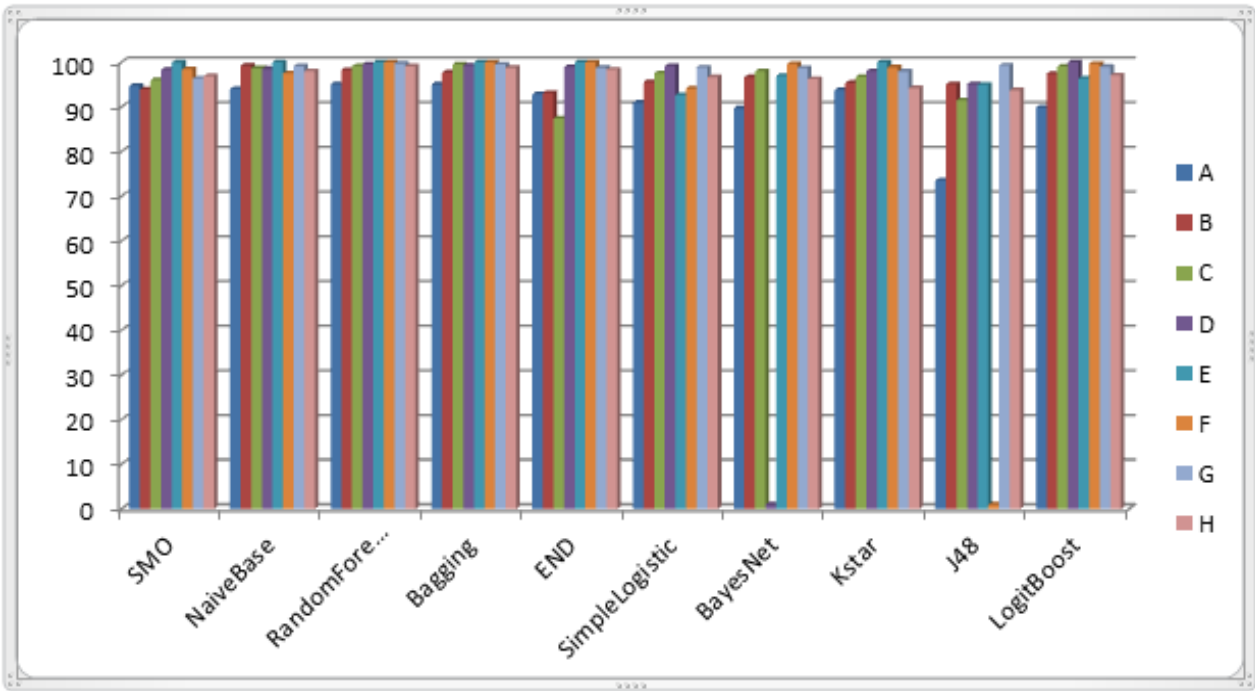


Figure 4-35 AUC percentage per class.

The RandomForest classifier once again maintained the highest percentage accuracy. This was proven by both prediction accuracy values and the AUC calculated. Therefore, RandomForest with Classifier base feature selection approach was selected as the best approach for dust particle classification. In the above experiments, it noted that cross-validation was not carried out. It is possible to use cross-validation in order to improve the reliability of the validation results.

4.6 Chapter Conclusion

Air pollution is a serious issue which has a significant negative impact on people, the environment, and the world economy. Air pollutants contain harmful material such as harmful gases, dust, fumes and odor, which suspend in the atmosphere. The airborne particles could cause damage to living organisms such as food crops, or to the natural, built environment or may cause death or disease to humans and animals. The huge increase in industry, businesses and heightened human activities in recent years, force many countries to monitor the air quality.

Machine vision techniques have been deployed in many fields and recently used in the air pollution field. However, the background research conducted within the context of this thesis revealed that there is a gap in the existing methods that can be used to classify industrial airborne dust.

The experiments were conducted for airborne dust samples which have been collected from Sohar industrial area of Oman. A state-of-art digital microscope was used to take images of the dust samples. The dimension of captured images were 1280X960 pixels in horizontal and vertical directions, respectively.

In colour-based experiment, K-means clustering algorithm is used to cluster the colour of the dust. In this experiment, the K-means clustering was used with RGB, L*A*B and HSV colour spaces. The results of the three algorithms were evaluated and compared. With RGB colour space, black and brown colours were mixed together and the algorithm failed to separate them. The result clearly indicates that RGB colour space is not a suitable colour space to be used for this segmentation. The HSV colours space were perform better than RGB and L*A*B. a total of 240 images were used for segmentation and as a result 8 classed were created.

More than 20 texture and 256 colour features were tested and evaluate separately. Colour and texture features, separately were unable to maintain a good accuracy level. To overcome this limitation, colour and texture features were combined together in one dataset giving a total of 278 features.

Several experiments were conducted to analyse the accuracy and the performance of classification algorithms. Two feature selection methods were adopted. The result shows that wrapper method for feature selection was better in comparison with filter method. It was also shown that colour features play a vital role in the classification process. When using the Random Forest classifier the proposed system was able to maintain a good accuracy with only 10 features consisting of two texture features and eight colour features. The results of accuracy were also compared against using the AUC of the ROC to ensure that result were consistent.

Chapter 5

Road Surface Type Recognition for Autonomous Vehicles

5.1 Introduction

A substantial aspect of vision-based road detection is to first classify image pixels as belonging or not to the road surface. However identifying road pixels is a substantial practical challenge due to the intra-class variability caused by lighting conditions [53]. The identification of the road type is very important for an autonomous vehicle as key decisions on the vehicle's drivability depends on the road type.

This chapter focuses on the development of two novel approaches to recognise five different road surface types, namely, Asphalt, Blocks, Concrete, Rough and Sandy at day-time. Two methods were implemented: a feature-based method based on traditional machine learning approaches and a deep learning based method.

For clarity of presentation, this chapter is divided into subsections. Section 5.2 discusses the first method (the feature-based method) proposed to recognise the road surface type. A detailed discussion on the deep learning based approach is presented in section 5.3; and finally, conclusions and future work are discussed in section 5.4.

5.2 Method-1: Colour and Texture Feature Based Classification Approach

This section introduces the reader to the first approach proposed in this chapter for road surface recognition. It presents in detail the functionality of each stage of the proposed system under three main topics: road surface segmentation, feature extraction and road surface classification. The figure below illustrates the block diagram of the colour and texture feature based approach.

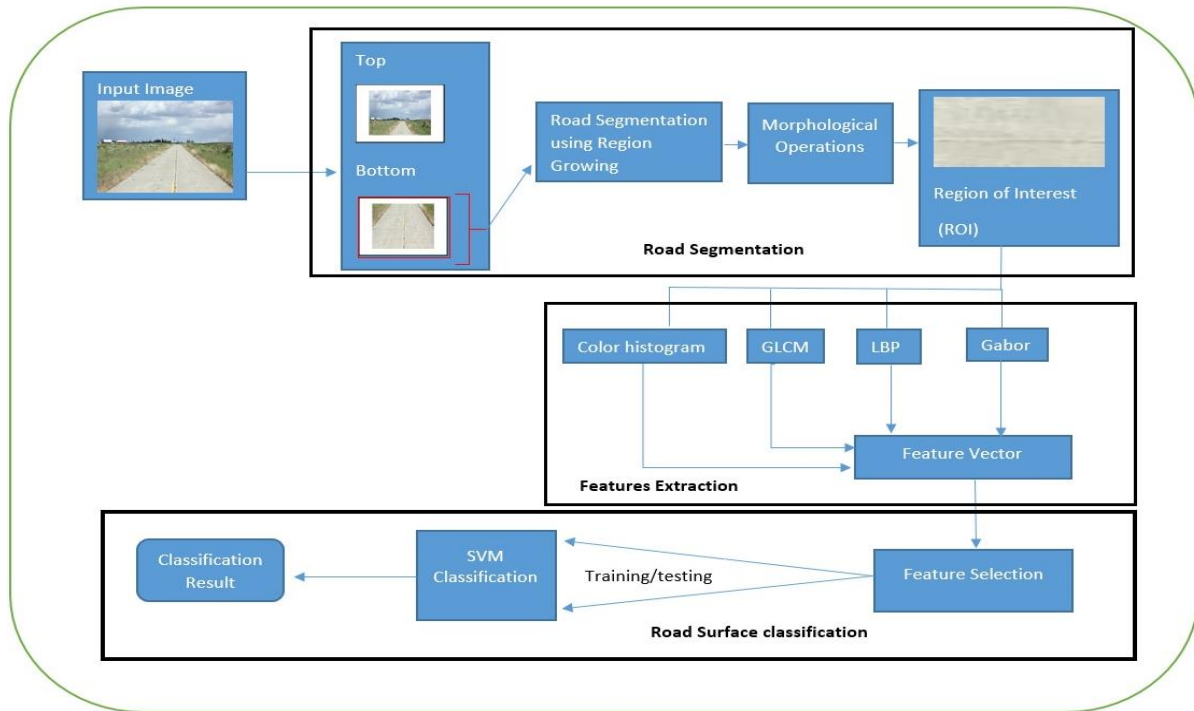


Figure 5-1 Method 1 – the feature based approach for road surface type recognition.

The following sections present in detail the operational aspects and blocks of this approach.

5.2.1 The Test Dataset

The test dataset used in the experiments has been created from different image data resources, but mostly captured as a part of the research conducted in this thesis. The dataset consists of 150 images of a typical road surface captured on board a vehicle and classified into five different road surface types namely asphalt, blocks, concrete, rough and sandy. Each category has a total of 30 images captured by a camera fitted onto the dashboard of a vehicle. Majority of the test images have been acquired using different cameras, resolutions and qualities as a part of the research conducted. These images were also gathered from different cities in UK and Oman under different illumination conditions: sunny, partly cloudy and cloudy at day-time. In addition to these images, 50 licence free images were downloaded from google images for different pavement types.

5.2.2 Road Surface Segmentation

The road surface segmentation is the first stage of the proposed system. It is an essential stage to subtract the road surface from input image so that further analysis of feature capture and surface type recognition can be carried out within the segmented regions, excluding the background that does not depend on the type of the road surface. Several steps are involved in this segmentation namely:

5.2.2.1 *Pre-Processing*

The input images were first split into two parts: top and bottom halves; for fast segmentation. It is worth noted that splitting the image reduces unwanted area in the image. In this case, the top part of input image is the unnecessary part, assuming that the camera on the dashboard of the vehicle is appropriately and intentionally installed to capture the road surface. The top half of the image contains unrelated objects from the scene's background. If this half is considered in road surface segmentation it will consume additional, unwanted time and compute power during the segmentation process, unlike the bottom half of the image which is the most important part for autonomous vehicle to recognise pavement type. In contrast the bottom half of the image contains a wide area of clear road surface and texture. This part of the image provides the desired details of road type for further analysis, speed of processing and proper execution. The figures below show the image splitting process to the top and bottom half of the input image.

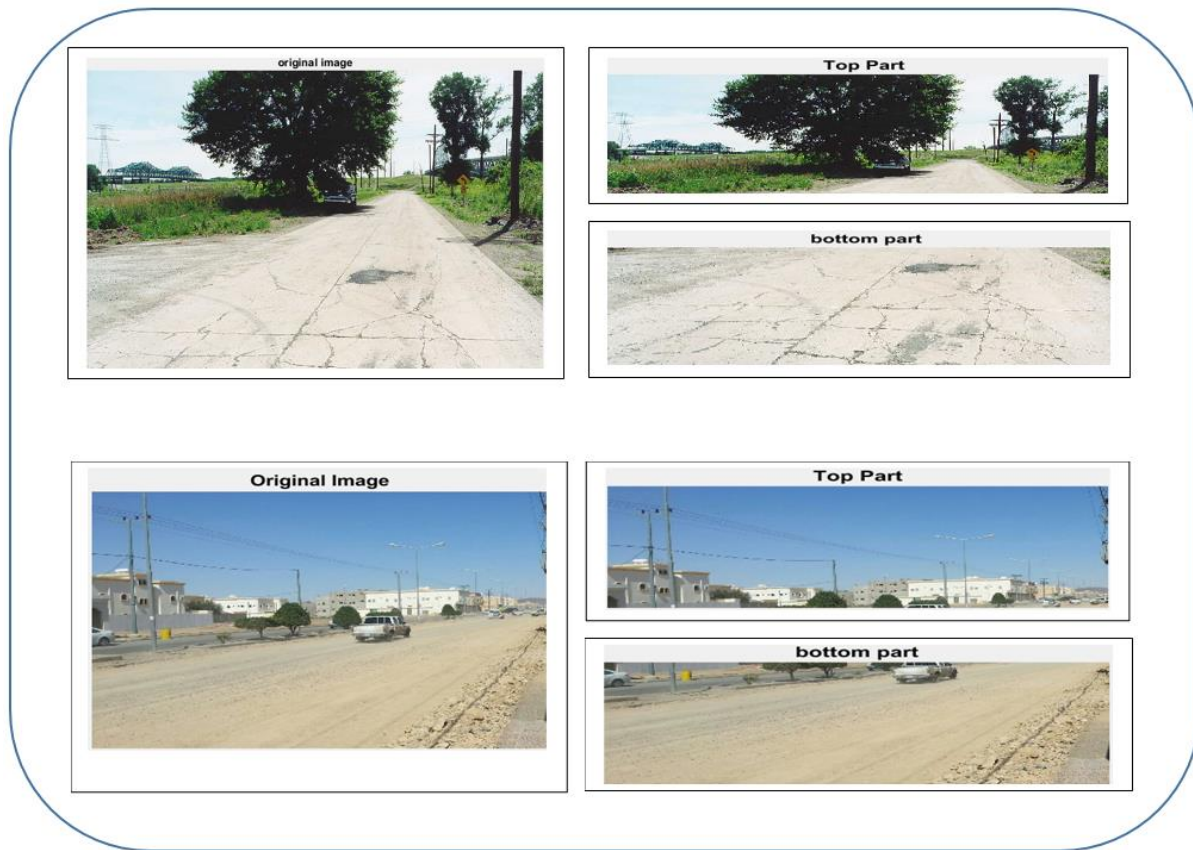


Figure 5-2 Input images splitting process.

5.2.2.2 Region Growing

Region growing [124] is a pixel-based segmentation technique. This algorithm was applied on the bottom part of input image obtained via the process presented in section [5.2.2.1](#). The implemented algorithm segments the road surface irrespective of the road marking and parts of any background. It segments the surface from side to side with the ability to distinguish irrelevant surface areas or objects on the road surface. Figure 5-3 illustrates the segmentation process that results when region growing algorithm is applied.

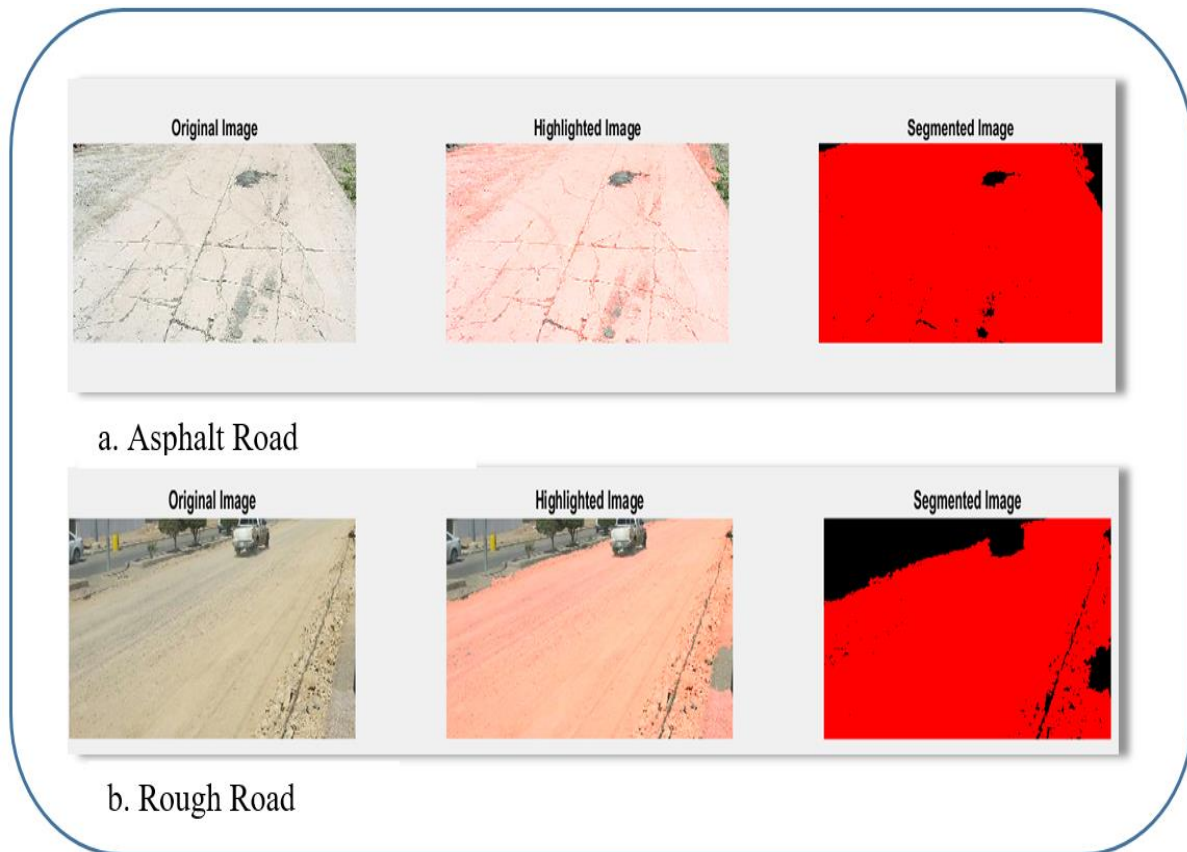


Figure 5-3 Region growing segmentation.

It can be clearly seen from the above figure that the region growing algorithm has segmented entire asphalt area and it excludes the concrete material which is used to fill the holes on the asphalt surface. Similarly, it eliminates irrelevant objects and surface finally selected is the rough road surface.

The segmented surface regions need further refinement to represent the true shape of the region. Therefore, several morphological operations, including desk structure, dilation and filling, were used to smoothen and improve the segmented area. The improvements of each process can be noticed in the experimental results presented in Figure 5-4.

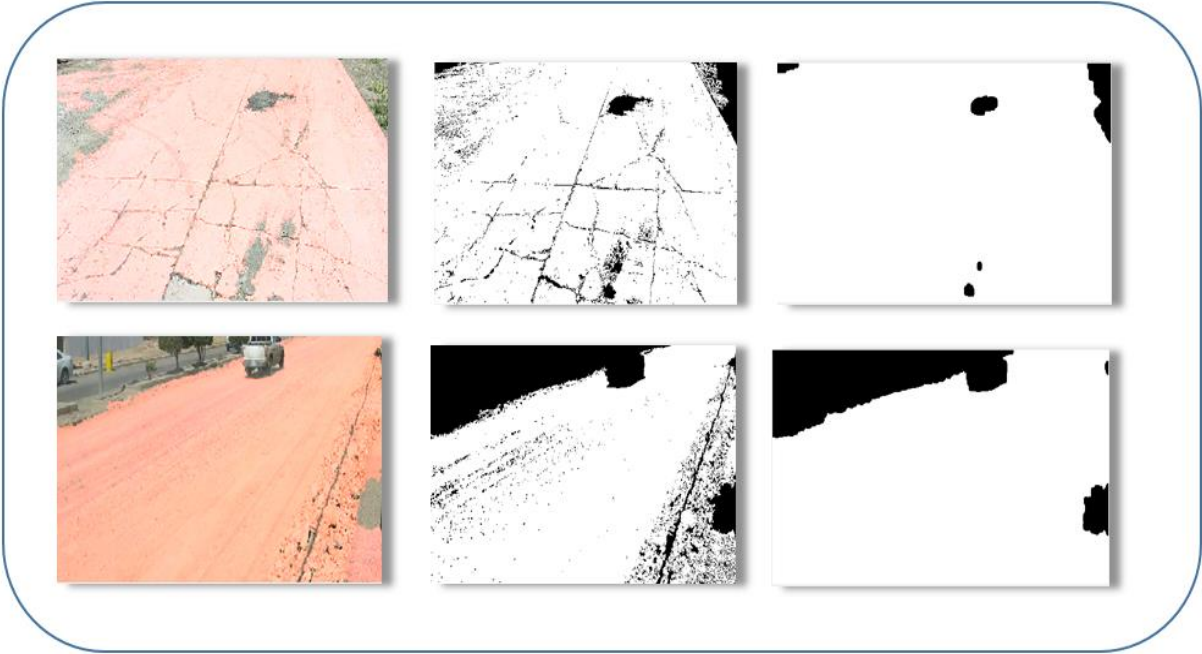


Figure 5-4 Segmented image and result of morphological operations.

After the extraction of the foreground road surface objects, regions of interest (ROI) were selected. They were resized to a size of 100×100 pixel blocks for further processing. In this process, all regions of interest (ROI) do not contain any background pixels or irrelevant objects. This can be clearly noticed in Figure 5-5.



Figure 5-5 Example of selected region of interest.

Although the algorithm has successfully segmented the surfaces in this dataset, the result could be effected in case of insufficient visibility of road surface in image; such as during traffic jam where vehicle are close to each other and road surface is not visible/clear enough to be detected. Hence, maintaining a safety distance

between the vehicles could help to overcome the limitation of this algorithm. Additionally, in practical application, the decision of segmentation is based on the differences between many frames.

5.2.3 Feature Extraction

Feature extraction can be used to concisely represent an image and hence when applied to the final output from section 5.2.2 can be used to recognise the road surface type using a classifier. In the proposed approach a substantial number of features were extracted from the regions of interest images extracted in section 5.2.2.2. For clarity and ease of discussion, the features extracted are presented in detail in the following sub-sections.

5.2.3.1 Colour Features

Extracting efficient image features is one of the most important parts in image classification, retrieval and indexing. The colour features along with shape and texture features are the most widely used visual features. Distribution of image pixels can be clearly recognised in an image histogram. Thus, the Hue colour histogram was used in the proposed approach to represent a colour of the road surface patches. Initially, all the images were converted into HSV colour space to obtain and subsequently separate the Hue features. 256 different colour bins were obtained from the hue channel of the HSV colour space.

5.2.3.2 Gray Level Correlation Matrix (GLCM)

Gray Level Matrix Correlation (GLCM) is a texture based feature extraction process. Initially, GLCM features were extracted by applying “graycomatrix” function in MATLAB to the road surface patches in order to subsequently examine the texture. Twenty two features of statistical textures were obtained from GLCM for each image. These features are listed follow:

- Entropy, sum of entropy, Sum of average, Sum of squares, Sum of variance, contrast, Difference of entropy, Correlation, Information measure of correlation, Energy, Contrast, Information measure of correlation 2 as in [38];
- Entropy, Cluster prominence, Homogeneity, Autocorrelation, Cluster shade, Dissimilarity, Max probability as defined in [39].
- Inverse difference normalised, Inverse difference moment normalised as in [40].
- Homogeneity, Correlation as in [41].

The GLCM features extracted above forms a part of the full feature description of an image patch and are concatenated with the other types of features extracted.

5.2.3.3 Local Binary Patterns (LBP)

The Local Binary Patterns (LBP) are one of the visual descriptor types that are popularly used for image classification. 59 bins of LBP features are extracted from each image patch and these are subsequently added onto the full feature representation.

5.2.3.4 Gabor Filter

A Gabor filtering technique presented in [56] was used to extract the Gabor features from input image patches. A total of 6760 features were extracted using the Gabor filter and appended into the full feature representation.

5.2.3.5 Feature Combination

In order to recognise and classify different types of road surface, a number of appearance based and texture based features were extracted from the image patches representing the road surface areas. Colour histogram, GLCM, LBP and Gabor Filtered features were extracted from the foreground image patches and concatenated into one feature vector to be used in the classification experiments that follow. The full feature vector consist of 256 colour features, 22 GLCM features, 59 LBP features and 6760 Gabor features to recognise asphalt, blocks, concrete, rough and sandy road surface types.

5.2.4 Result and Analysis

The dataset used in the experiments consists of features extracted from ROIs of 150 different road surface images captured on-board a vehicle. As explained above, the features of the dataset; includes colour features, GLCM features, LBP features and Gabor features; combined together in one feature vector. Numerous experiments were conducted in order to analyse the performance of different classifiers. Initially, a dataset of colour and GLCM features were used to recognise the five different road surface types. The dataset was split into training and testing sets.

The figure below illustrates the accuracy details obtained when using different classification algorithms, but using only colour and GLCM features.

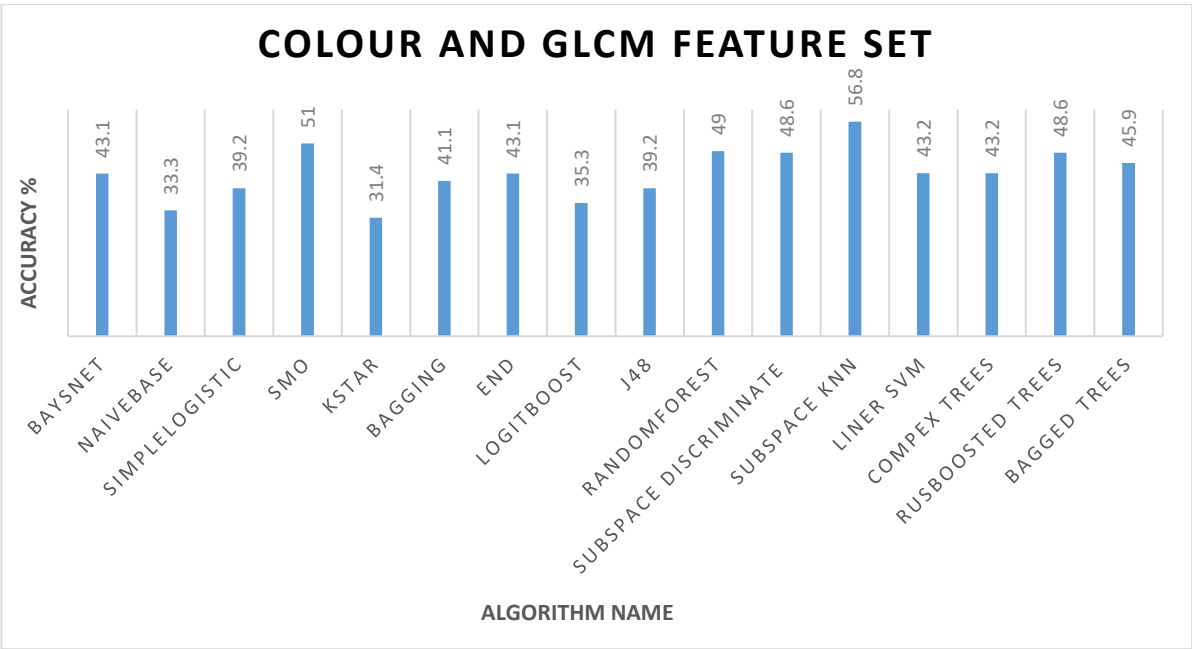


Figure 5-6 Comparison of different algorithms using colour and GLCM dataset.

It can be clearly seen from the above figure that accuracy obtained is below 60% for all classification algorithms investigated. Although, the Subspace KNN algorithm has achieved marginally better results, it is still considered below a practically acceptable limit and hence needs to be improved. Thus, other extracted features should be included with the intention to improve the accuracy of predicting the road surface type. Therefore, LBP and Gabor features were added to the colour histogram and GLCM features creating a substantially large feature vector.

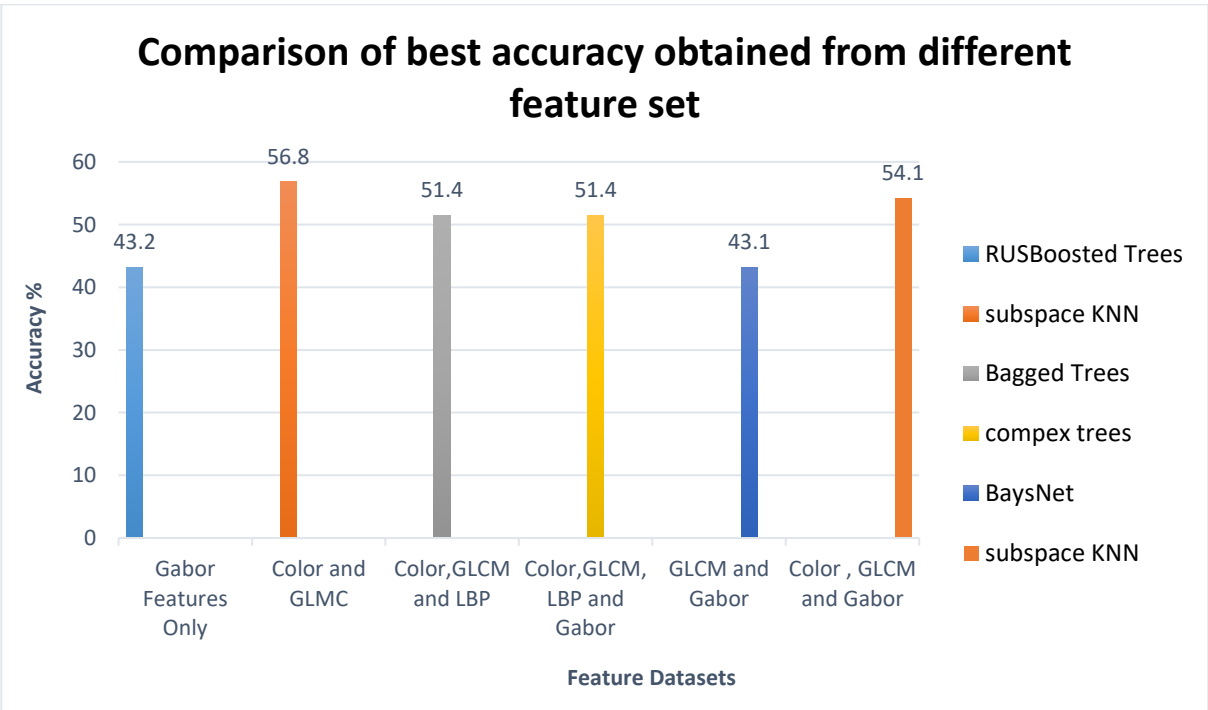


Figure 5-7 Best accuracy obtained from different feature set combinations.

Table 5-1 illustrates another form of percentage per classifier using different feature datasets.

Feature Datasets	Gabor Features Only	Color and GLMC	Color, GLCM and LBP	Color, GLCM, LBP and Gabor	GLCM and Gabor	Color, GLCM and Gabor
Classifier Name	RUSBoosted Trees	Subspace KNN	Bagged Trees	Complex Trees	BaysNet	Subspace KNN
Accuracy obtained	43.2%	56.8%	51.4%	51.4%	43.1%	54.1%

Table 5-1 Best classifier performance per feature dataset.

A numerous experiments were conducted with different combinations of features extracted in section 5.2.3 in order to investigate the potential improvement of the accuracy of classification. Unfortunately the results of these experiments did not show any significant increase in accuracy via different combinations of features. The results indicate overall low classification accuracy for all feature combinations. The highest accuracy result obtained was only 56.8%. It is worth noting that combining different feature datasets failed to improve the overall accuracy to a satisfactory level.

5.2.5 Conclusions

A substantial number of features were extracted from the region of interest images. The features were obtained from the colour histogram, GLCM, LBP and Gabor filtered output and grouped into one vector. Numerous experiments were conducted in order to analyse the performance of different datasets against 16 different classifiers.

Initially, a dataset of colour and GLCM features was created and used to recognise the 5 different road surfaces. The dataset was split into training and testing sets. The best rate of accuracy obtained was 56.8% with the subspace KNN classifier. A possible explanation for this low accuracy might be that colour and GLCM features are not the optimal feature combination. It was concluded that other features should be included in the feature vector in order to improve the accuracy of predicting the road surface type. Therefore, LBP and Gabor features; along with colour and GLCM features; were included in the same feature vector with the intention of improving the prediction accuracy. However contrary to the expectations, this approach did not result in a significant improvement of accuracy or

performance, when using different feature combination. Even the best results obtained by the experiments conducted above and represented in Figure 5-8 indicate the difficulty of classifying road surface using traditional features based approaches. Although, using feature selection techniques could improve results, the accuracy level is not expected to be improved to a satisfactory level, as the accuracy obtained without feature selection remains substantially low. This finding was unexpected and suggests that solving road surface classification issue requires another approach in order to achieve a good prediction accuracy.

5.3 Method 2: Deep Learning Based Approach

This section presents the details of a further novel approach to road surface type recognition. The method is based on deep learning, where a convolutional neural network is used to distinguish between road surfaces types.

5.3.1 Methodology

The main objective of this approach is to develop a fully automated machine learning based system to recognize different road surfaces regardless of road edges and marks. The system should be robust to be able to classify the road surface type irrespective to road edges and markings.

A detailed block diagram of the proposed deep learning based method for road surface type recognition is illustrated in Figure 5-8 below. In the stage 1, road surfaces were segmented and regions of interest (ROI) were selected following an approach identical to that presented in section 5.2. All the ROIs were resized to the same size of first layer of the CNN. In stage 2, the features are extracted from the ROIs using Convolutional Neural Network (CNN). The training and prediction processes implemented in stage 3.

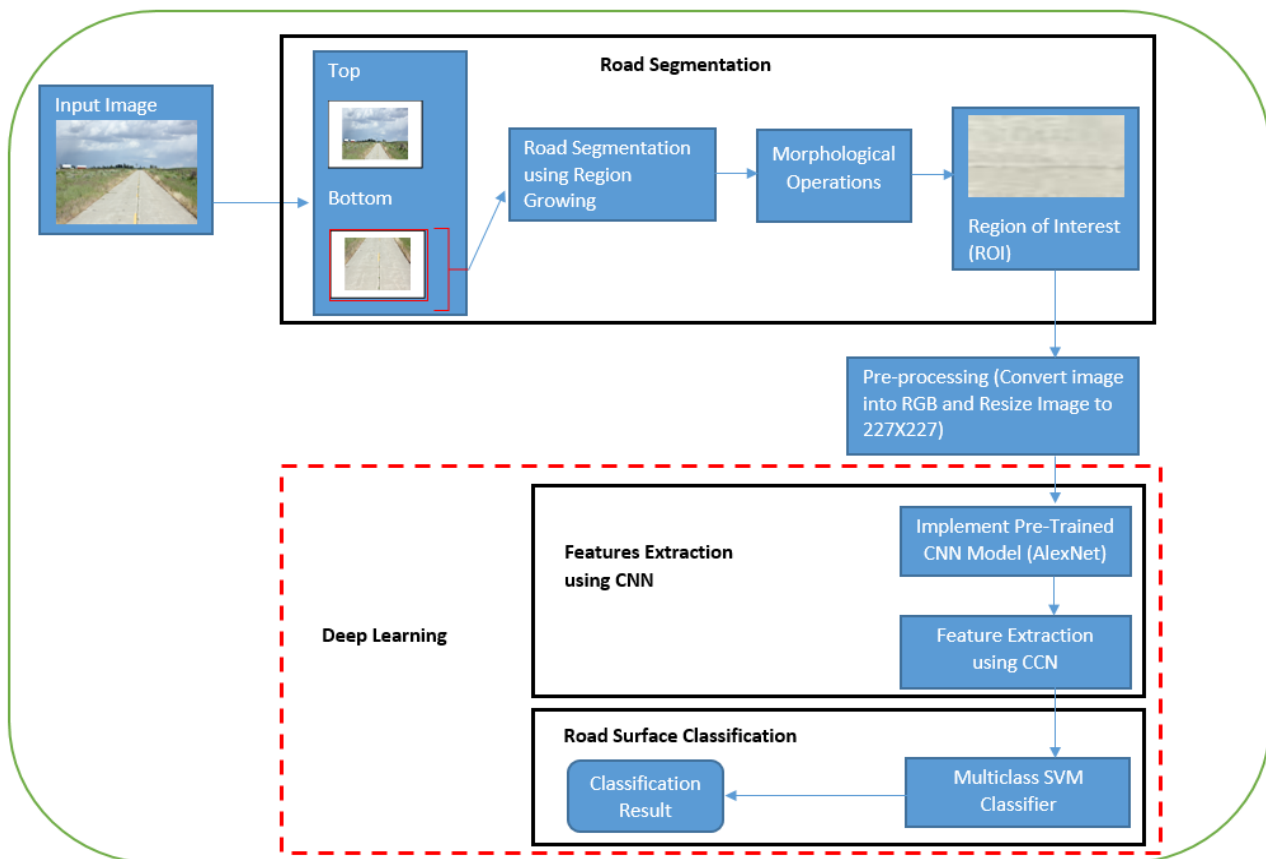


Figure 5-8 The proposed, deep learning based approach to road surface type recognition.

The operational details of each of the stages can be described as follow:

5.3.2 Road Surface Images

A total of 150 images of different road surfaces were used. The images were gathered from the internet and captured from different cities in Oman and UK as a part of the research conducted. The images were captured using different camera settings, environmental conditions and quality.

5.3.3 Road Segmentation Process

In this stage, the road surface is segmented regardless of the road marks and edges. A detailed discussion of a road surface segmentation approach used was presented in section 5.2.2. The same approach is used here.

5.3.4 Pre-Processing

Extracting features manually is different than when using CNN. Each CNN based method has its own requirements based on structure of used algorithm/network. In the proposed approach, the images were resized to the same size of first convolutional layer (input layer) of AlexNet. The structure of the used network was built to hold an image of width 227, height 227 and with 3 colour channel R, G and B.

5.3.5 Feature Extraction Process

The features of an image contains of comprehensive details of all objects, colours, illumination, background and foreground data of an image. Process of extracting feature is a critical step. Irrelevant or extra features can have direct negative impact on the prediction accuracy. In this proposed approach, features are extracted using a Conventional Neural Network (CNN). For clarity of presentation, feature extraction steps are presented in the subsections below.

5.3.5.1 Implementation of Pre-trained CNN Model

Extracting image features using deep learning requires a neural network. Additionally, the network has to be trained from scratch on a related large scale problem. Thus, a dataset of millions of different road surface images is needed to train the network. Alternatively, pre-trained conventional networks can be used to train small or sufficiently large datasets and to classify the images. In the proposed method, transfer learning technique was adopted. AlexNet model described in section [3.5.1](#) was used to train the dataset. The figure below shows the layers used of implemented model.

1	'input'	Image Input	227x227x3 images with 'zerocenter' normalization
2	'conv1'	Convolution	96 11x11x3 convolutions with stride [4 4] and padding [0 0]
3	'relu1'	ReLU	ReLU
4	'norm1'	Cross Channel Normalization	cross channel normalization with 5 channels per element
5	'pool1'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0]
6	'conv2'	Convolution	256 5x5x48 convolutions with stride [1 1] and padding [2 2]
7	'relu2'	ReLU	ReLU
8	'norm2'	Cross Channel Normalization	cross channel normalization with 5 channels per element
9	'pool2'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0]
10	'conv3'	Convolution	384 3x3x256 convolutions with stride [1 1] and padding [1 1]
11	'relu3'	ReLU	ReLU
12	'conv4'	Convolution	384 3x3x192 convolutions with stride [1 1] and padding [1 1]
13	'relu4'	ReLU	ReLU
14	'conv5'	Convolution	256 3x3x192 convolutions with stride [1 1] and padding [1 1]
15	'relu5'	ReLU	ReLU
16	'pool5'	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0]
17	'fc6'	Fully Connected	4096 fully connected layer
18	'relu6'	ReLU	ReLU
19	'fc7'	Fully Connected	4096 fully connected layer
20	'relu7'	ReLU	ReLU
21	'fc8'	Fully Connected	1000 fully connected layer
22	'prob'	Softmax	softmax
23	'classificationLayer'	Classification Output	cross-entropy with 'n01440764', 'n01443537', and 998 other classes

Figure 5-9 AlexNet Architecture.

The proposed pre-trained network has 7 convolutional layers, 3 fully connected layers and 15 other layers for normalisation and down sampling. It receives an input using the input layer and transforms it through a series of hidden layers. The details of some layers are described as follow:

- Input: input layer that accepts an RGB image of size 227 x227.
- Conv: Convolutional layer that filters or computes the output of neurons that are connected to local region in the input. It has slider and padding values. The slider value is used to slide the filter to produce smaller output volumes. The padding used to pad the input volume a value (usually 0) around the boarder. This allow to control the size of output volumes.
- RELU: used to apply elementwise activation function (threshold to zero) to leave the size of volume unchanged.
- Norm: cross channel normalisation.
- POOL: used to perform a down sampling operation
- FC: fully connected layer to compute the class score. It has full connection to all activations in the previous layer.

5.3.5.2 Feature Extraction

Initially, a sample of input image was taken using the input layer and transformed through a series of hidden layers. After extensive computations, features were extracted from the fully connected layer. A ConvNet was used as the fixed feature extractor. This means, the last fully connected layer was removed (output are 1000 class scores) and features were taken from last fully connected layer that has a 4096-D vector. The AlexNet computed a 4096-D vector of each image. All the features are automatically combined in one feature vector. The vector is divided into training and testing datasets.

5.3.6 Analysis of Results

The dataset used in our experiments consists of features of 150 different images of road surfaces. The proposed method extracted 4096 dimensions of features for each image. Numerous experiments were conducted to evaluate the performance of the proposed method for road surface classification. Initially, a dataset with unsegmented images was used as an input image to AlexNet. The Accuracy results and confusion matrix are shown in Table 5-2 and Table 5-3 respectively.

Output Classes	Asphalt	88.89 %	11.11 %	0 %	0 %	0 %
	Blocks	0 %	88.89 %	0 %	0 %	11.11 %
	Concrete	22.22 %	0 %	55.56 %	11.11 %	11.11 %
	Rough	0 %	0 %	0 %	88.89 %	11.11 %
	Sandy	0 %	0 %	0 %	11.11 %	88.89 %
		Asphalt	Blocks	Concrete	Rough	Sandy
Target classes						

Table 5-2 Confusion Matrex of unsegmented images.

Accuracy
82.22 %

Table 5-3 Accuracy obtained using unsegmented images dataset.

The results in the above confusion matrix table indicate that the concrete road surfaces were poorly predicted. More than 22 % of concrete surfaces were classified as asphalt. Moreover, more than 11% each were classified as rough and sandy. This result may be explained by the fact that both concrete and asphalt surfaces have some similarities in shape, colour, texture and road marking; taking into account that no segmentation was applied and no regions of interest (ROI) were selected. Another explanation for this is that some images of concrete surface have small amounts of Asphalt that is used to fill the ‘holes’ on the road surface as illustrated in the figure below.



Figure 5-10 Concrete road images.

The most interesting finding was that 88.89% of the remaining road surfaces were correctly predicted. Another experiment was conducted using the proposed method. A detailed result is shown in the table below.

Output Classes	Asphalt	100 %	0 %	0 %	0 %	0 %
	Blocks	0 %	100 %	0 %	0 %	0 %
	Concrete	0 %	0 %	88.9 %	0 %	11.1 %
	Rough	0 %	0 %	0 %	77.8 %	22.2 %
	Sandy	11.1 %	0 %	0 %	11.1 %	77.8 %
Target classes		Asphalt	Blocks	Concrete	Rough	Sandy

Table 5-4 Confusion matrix of the proposed method.

Accuracy
88.9 %

Table 5-5 Accuracy obtained using the proposed method.

The table above clarifies that asphalt and blocks were classified perfectly. Concrete class was classified as 88.9% correctly. Whereas, 11.1% of concrete surfaces were classified as sandy. In addition, more than 77.8% surfaces were classified as rough and the remaining classified as sandy. Moreover, sandy classification distributed between sandy, for 77.8%, rough and asphalt. Both rough and sandy have common features. This justifies the low prediction accuracy for both classes.

It is worth noted that accuracy obtained using the proposed method is higher than using unsegmented dataset.

For comparison and benchmarking purposes, the proposed method was evaluated against another CNN model, namely OverFeat network [99]. Figure 5-11 shows the accuracy obtained by different methods.

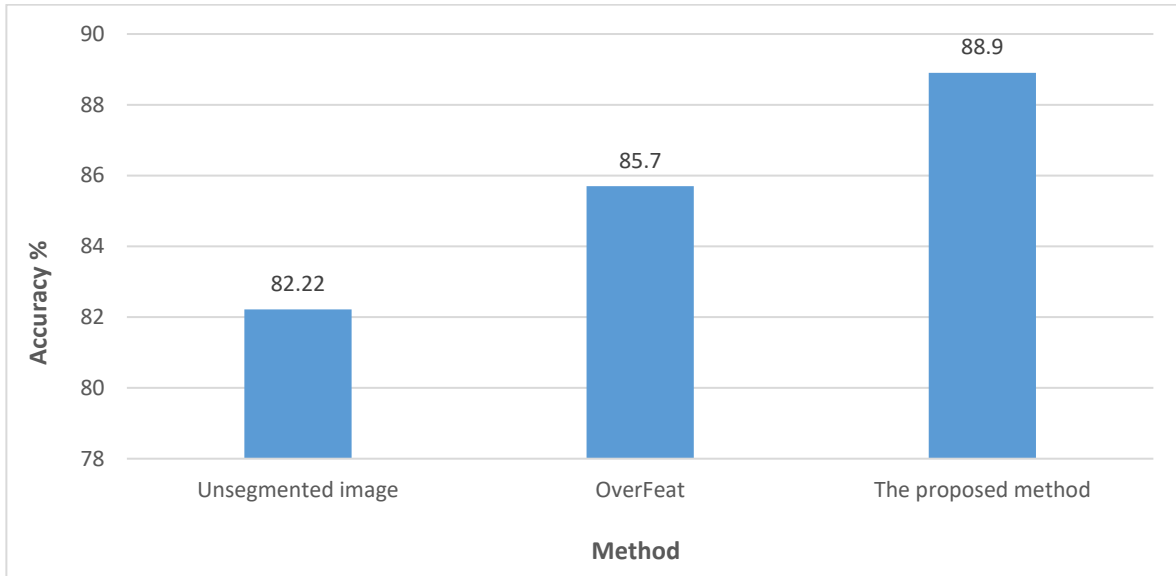


Figure 5-11 Comparison of accuracy obtained by different method.

The figure compares prediction accuracy of three different methods. The most remarkable observation to emerge from the data comparison is that the proposed method achieved the highest prediction accuracy among benchmarked methods. It is worth taking into consideration that similarity of visual features of rough and sandy roads has decreased the accuracy to the current level.

5.3.7 Conclusion

In the proposed approach, the images were resized to a size of 227 x 227 pixels. AlexNet model of Pre-trained conventional network was used to extract image features. A ConvNet was used as the fixed feature extractor. This means that the last fully connected layer was removed and a vector of 4096-D for each image was generated. All feature vectors were concatenated in one feature vector and the data thus obtained was divided into training and testing data sets. Numerous experiments were conducted to evaluate the prediction accuracy of the proposed method against other methods. It has been shown that the proposed method obtains the highest prediction accuracy among benchmarked methods.

5.4 Chapter Conclusion

This chapter proposed 2 approaches to classify 5 different road surface namely: Asphalt Road, Blocks Road, Concrete Road, Rough Road and Sandy Road. The images were gathered from the internet and captured from different cities in Oman and the UK. A total of 150 road surface images were collected for 5 different road surface types above. The images were captured using different camera settings and quality. This dataset used in both proposed approaches. In the research presented so far in literature the datasets used are largely limited in that they don't vary significantly in illumination. Time-of-day, quality, resolution, etc., and hence the database collected and used in this research is comprehensive and novel. In due course an attempt will be made to make this data publically available for the benefit of the wider research community.

In the first approach proposed for road surface type recognition, traditional machine learning approaches were used to recognise different types of road surfaces. The input images were first split into two parts: top and bottom; to accelerate the segmentation process. Typically, the top part of input image can be excluded from detailed analysis as it is less likely to hold significant area of the road ahead of the vehicle. Considering this half of the image for road surface type recognition consumes time and power during the segmentation and analysis processes. In contrast it was shown that the bottom half of the image is the most relevant in the recognition of road surface type as a significant proportion is covered by the road ahead. Thus, bottom part of image was used in segmentation process. A Region Growing segmentation technique was implemented to initially segment the surface road and remove background from being considered by the sections that follow. Subsequently a substantial number of features were extracted from the region of interest images. The features were obtained from the colour histogram, GLCM, LBP and Gabor filter and combined into one vector. Numerous experiments were conducted in order to analyse the performance of different combinations of feature datasets obtained above against 16 different classifiers. Unfortunately using this approach the best possible accuracy results obtained was below 60%. This indicates the difficulty of classifying road surfaces using traditional feature based approaches. Although, using feature selection techniques it was shown that accuracy results can be improved, the improvement obtainable were shown to be marginal. It is noted that in the existing research few approaches presented have reported higher overall accuracy levels. However closer investigation of the datasets used by the authors revealed that these reported results were biased by the fact that only few road surface types were detected, using only training and test dataset with no variability of data due to illumination or quality of resolution of the images.

In the second approach proposed in this thesis a deep learning based approach was proposed for road surface type recognition. The segmentation process applied in the first approach was used for the extraction of the most relevant ROI that contains the road surface. The AlexNet model of a pre-trained conventional network was subsequently used to extract image features from the segmented ROIs. A vector of 4096-D was generated for each image and combined in one feature vector. The overall results of conducted experiments of this proposed method shows that significantly better prediction accuracy is achievable when comparing with the benchmarked methods in literature and the traditional machine learning based approach proposed in this chapter. It is worth noted that asphalt and block road surface areas were recognised with 100% accuracy. The Concreate road surface areas were recognised with an accuracy of 88.9%, with the remaining percentage of concrete surfaced roads being misclassified as Sandy roads. In addition, 77.8% or the roads with Rough surfaces were accurately classified as rough and the remaining percentage was misclassified as Sandy road surfaces. Detailed analysis of the results when experimenting with Concreate, Rough and Sandy road surfaces revealed that the inaccuracies of classification was due to the similarities between these three types of roads and the presence of mixed roads such as concrete roads with Sandy or Rough patches due to the collection of dust on the road or repairs being done to the Concreate road surfaces.

Chapter 6

Camel Head Detection and Recognition

6.1 Introduction

Machine learning has been widely used to solve many different practical problems of different complexity over the past. In particular detecting, monitoring, tracking and recognising objects of interest is in widespread use within industry, medical science, business and human activity monitoring applications. Despite the use of machine learning in several fields, protecting human beings and animals from danger has been one of the key priority applications. In particular animals are exposed to increasing amount of danger due to hunting and poaching worldwide and hence the focus of many AI/Computer Vision researches has been shifted more recently towards this important and timely application area animal detection and recognition, extending in some cases to animal behaviour analysis.

For centuries in history camels have been used for transportation purposes within the Middle Eastern and African countries and their milk has been used as a source of nutrients and as medication for a number of diseases (Mabood, et al., 2017). The number of living camels in the Arabian Peninsula (Dromedary) has been estimated to be around 1.6 million (FAO statistics, 2011) and (Abdallah & Faye, 2012). The reason behind the need for safeguarding camels in the Arabian Peninsula is due to their high value which might reach millions of dollars per individual dromedary in some cases. For example as camel racing has become a part of Arab culture, prize money for a winning camel can be millions of dollars per race. Therefore, theft and/or harming / killing camels used for racing purposes have become a series problem for camels' owners in the recent past. Moreover, camels prefer to live/to-be-kept in open areas especially in desert environments and generally walks tens of kilometres a day. Sometimes a camel can move more than 100 kilometres from the base. During summer, camels leave their shelters for food and usually shelter in the shade under trees during daytime. All of the above mentioned behaviour of camels makes it difficult to quickly locate missing camels. Furthermore, camel farmers are often interested in taking a daily count of their livestock and for this purpose they are increasingly using low flying drones. Therefore camel detection and counting is a vital computer based approach that can help farmers carry out this task within minimal human intervention.

Further to the above, camels found dangerously close to roads and the highways can create significant hazards to the road users. Hence their movements need to be monitored regularly. The number of collisions between camels and vehicles in Saudi Arabia for example, were estimated to reach more than a hundred each year (M.S.Zahrani, et al., 2011). The researchers implemented Camel-Vehicle Accident Avoidance System (CVAAS) using two technologies GPS and GPRS to detect camel's position and transmit that position to the CVAAS server consequently. The CVAAS server checks the camel position and its orientation in order to decide if there is a need to warn the drivers through activating the warning system, especially if the camel is within a danger zone. It is also mentioned that the cost of deploying CVAAS on a great scale is too much. More importantly, the system suffers from many false negatives as a result of the dependency on many parameters such as a width of the dangerous zone, variation in camel speed and delay in receiving SMS messages.

It is worth noting that, there is no research that has been undertaken to detect and track camels in desert environment using computer vision and machine learning approaches. However a machine vision-based system can be developed to detect camels based on their heads' features from different angles and orientations. With the aim of putting a boundary around the scope of the project, the research conducted in this thesis has been focused only on 'camel head' detection, rather than detecting full camels, including their bodies. The detection of the head only serves many of the applications discussed above and is hence a feasible solution to the problem to be solved.

The aim of this project is to develop a novel automated machine learning based system to detect and classify camels' head in different environments, including a desert. The system should accept the image data, processes the input and return the image with detections. The performance of this system will be evaluated and benchmarked with other techniques and algorithms.

For clarity of presentation, this chapter is structured into several sub-sections. Section 6.1 presents an introduction to the problem of camel detection. In section 6.2, an overview of proposed methods, namely the feature based approach and the deep learning based approach, are presented. Section 6.2.1, includes a detailed discussion and experimental results for feature based approach. The details and results of the deep learning based approach are presented in section 6.2.2. Finally section 6.3 concludes the chapter.

6.2 The Proposed Method

Two approaches are presented for the detection and recognition of camels. The first approach is a feature based approach where traditional machine learning algorithms are utilised. In this approach, Haar-like features [see section 3.4.2.1] were first extracted and used to detect the camels' heads. For comparison purposes, the use of HOG and LBP features were also investigated. In the second approach focused on deep learning, a Region-based CNN was implemented to detect camel heads. Two models, Fast R-CNN and Faster R-CNN, were used and the detection accuracies obtained were compared. Figure 6-1 gives a block diagram that provides an overview of the two proposed methods.

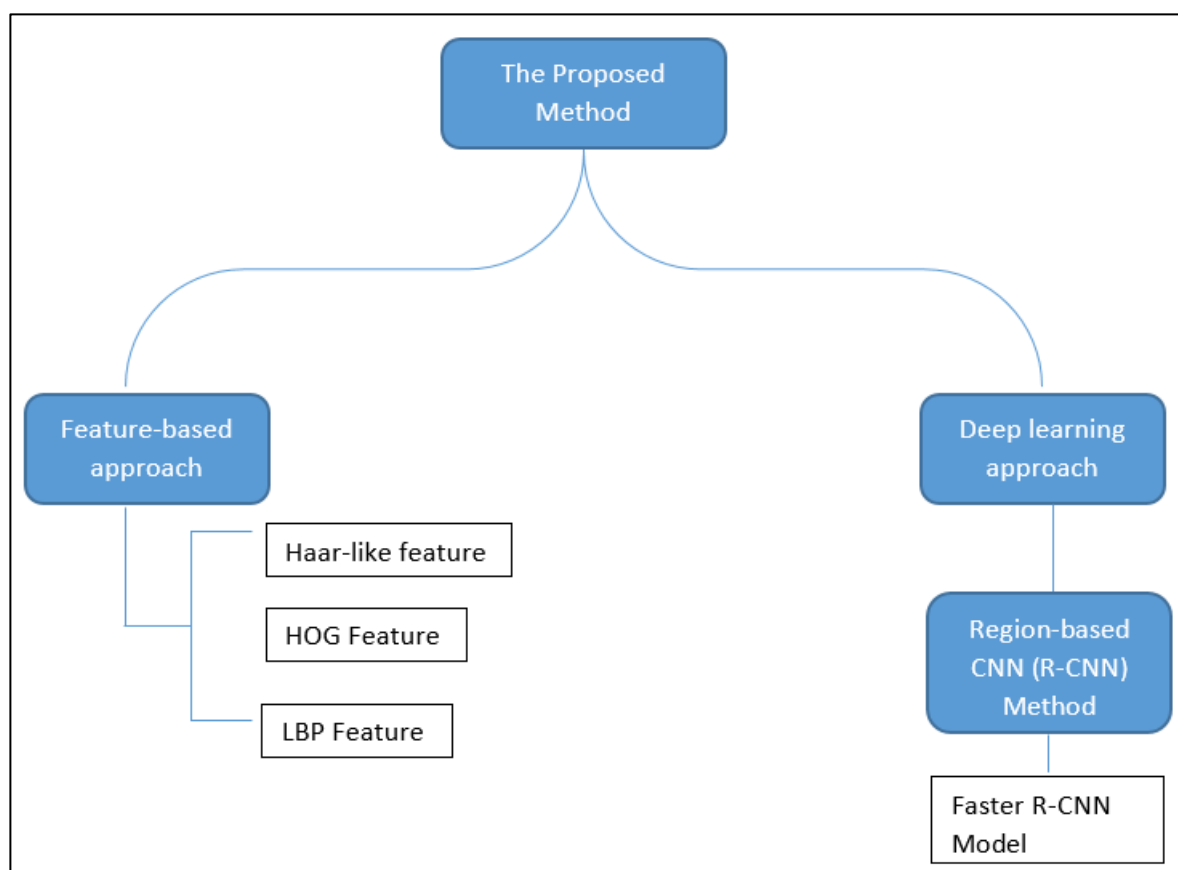


Figure 6-1 Overview of the proposed method.

6.2.1 Feature-Based Approach

This section introduces the reader to the first approach proposed for camel detection and recognition. It presents in detail the functionality of the techniques used within the proposed system under three sub-stages: training a cascade object detector by labelling the images for training a classifier, cascade

object detection to detect and classify the camels’ heads, and counting the total number of detected heads in an image. The Figure 6-2 below illustrates the process of camel head detection and recognition using the above stages.

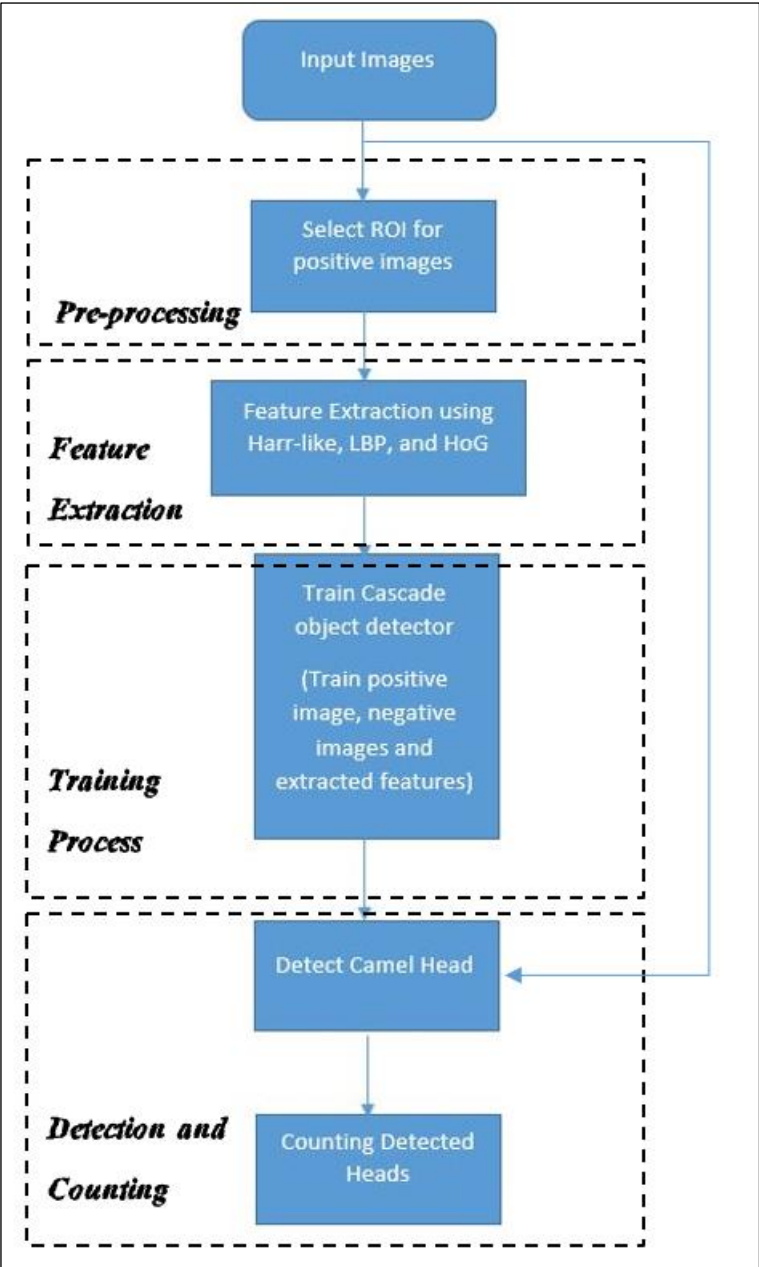


Figure 6-2 Feature based approach of the proposed method.

6.2.1.1 Dataset

The dataset used in this research has a total of 918 images. It has been categorised into three different sets namely: the training set, the testing set and the negative image set. A total of 418 images were used as camel heads, for training, 450 images of non-camel-head images as negative images and 50 images as testing images containing camel/s.

The images were gathered from different locations in Oman. Additionally, 532 licence free images were downloaded from google images. All images were in 'jpeg' format. The images have been acquired using different cameras, different resolutions and quality, different sizes and backgrounds, and from different angles and orientations. Appendix B, illustrates some example images used in the training process.

6.2.1.2 Pre-Processing

This process is used to select the Region of Interest (ROI) for all positive images in the training set. Training Image Labeler application from MathWorks was used to specify the ROI and create the ground truth. Multiple ROIs were also selected in some images.

6.2.1.3 Feature Extraction

The next step after ROI selection is feature extraction. Initially, Haar-like features were extracted from the images to be used as features in camel face detection. The extracted features were assigned into a feature vector for training purposes. For comparison, HOG and LBP features were also extracted and used in training and testing.

6.2.1.4 Training

For the purpose of camel face detection, the well-established Viola-Jones face detection algorithm [3.4.2] was used to detect a camel's head. The aim is to have a cascade object detector with high correct camel face detection rate and low false detection rate. There are three main steps in training cascade detector. The first step is to train a binary classifier, starting with training images that contain camels' heads as positive images and the non-camel-head images as negative images. The next step is using the extracted features to represent the features of a camel's head. The final step was focused on the classification of the camel's head by using a cascade classifier. Figure 6-3 below illustrates the various steps of training the cascade object detector:



Figure 6-3 Stages of training the cascade object detector.

6.2.1.5 Experiment Setup

In the experiment, the object training size has been set as 'auto' to determine the object size automatically based on the median width-to-height ratio of the positive instances. Additionally, the number of cascade stages were set as 10 among 50 stages. Moreover, the acceptable false alarm rate was determined to be approximately 0.5 and 0.99 for true positive rate out of 1 in order to achieve fewer false detections and a greater number of correct detections. The number of negative samples factor to be used at each stage was selected to be 1.

6.2.1.6 Experimental Results and Analysis

A number of experiments were conducted using Haar, LBP and HOG features to detect camel heads. All experiments that had less than 10 cascade stages were shown to obtain very high false detections for all of the different feature sets investigated. Acceptable results were only achieved with 10 stages. Hence, the minimum number of cascade stages to be used in the experiment was set to 10 stages.

For clarity of presentation and analysis of results, the results are discussed based on misclassifications that results from ambiguity of texture, shape and overlap/occlusion of heads, respectively in Figure 6-4, Figure 6-5 and Figure 6-6 respectively.

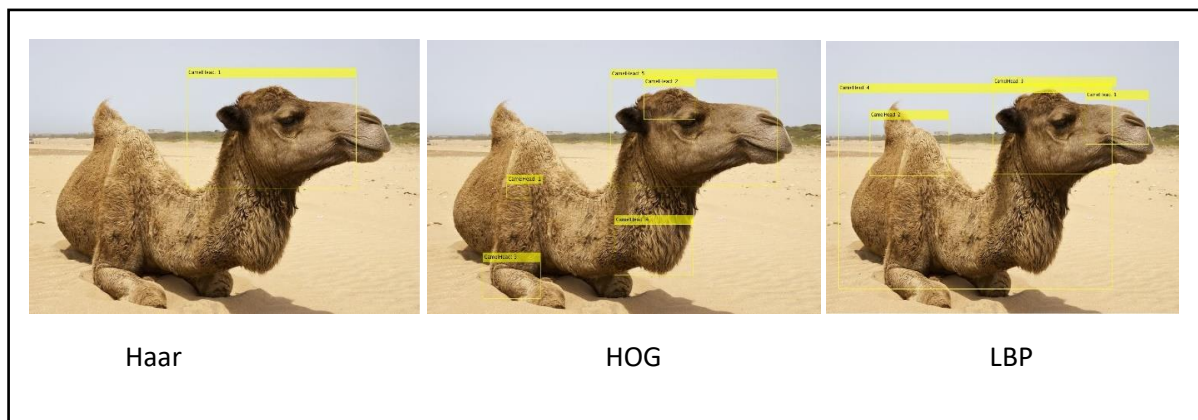


Figure 6-4 Misclassifications due to ambiguity in texture.

The texture of camel head can be clearly noticed in the images illustrated in Figure 6-4. The results show that that Haar feature based detector performs better for camel head detection than HOG and LBP features. Haar like features has detected the camel head accurately with no false detections. LBP and HOG features performed worse with few non-head areas being detected as heads. A reason for this could be that Haar like feature set is the only feature set that describes the head based of camel's head shape and contrast variations, as these can be represented well with Haar like features. The

HOG is based on colour histograms and LBP on the luminance value variations leading to changes in binary patterns. It appears that some background regions have variations in colour and binary patterns very similar to that of a camel head.

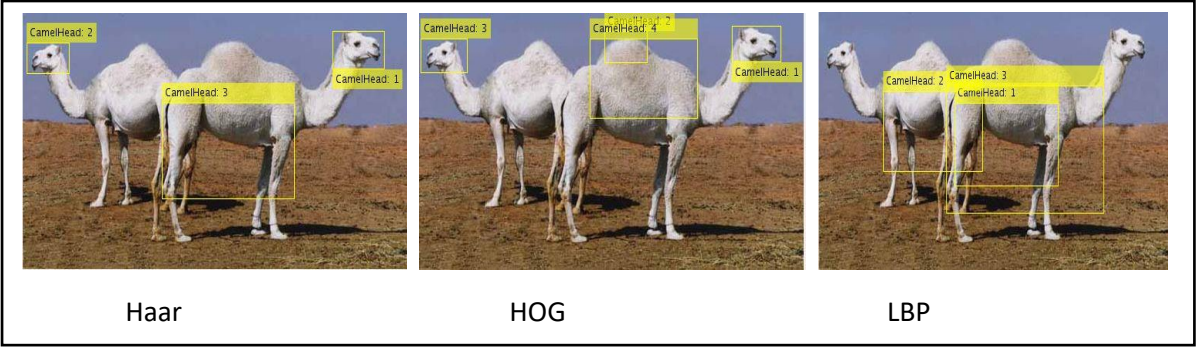


Figure 6-5 Misclassifications due to ambiguity of shape.

In Figure 6-5 although the head texture is not clear in above figure, Haar based detector has detected the heads of both camels’ perfectly. However the Head-3 refers to a misclassification, where the Haar features have picked features similar to that present on heads. The performance of HOG was similar although indicating two misclassifications. On the other hand, all the detections in LBP feature based approach were false detections as none of the camel faces were detected. It can be noticed that false detections in HOG is based on shape ambiguities. A closer analysis revealed that the false detections in HOG detected camel humps, which has a shape similar to a camel’s head.

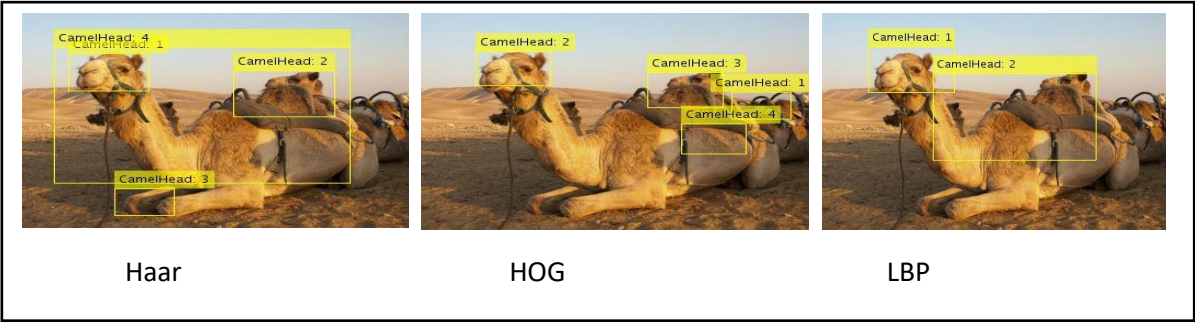


Figure 6-6 Misclassifications due to head occlusions.

Figure 6-6. illustrates misclassifications that result from occlusion of camel heads in test images. It shows that all classifiers perform differently with results obtained from Haar and HOG features being able to detect more accurate boundaries around the two camel heads, but leading to also higher number of false detections as compared to LBP.

Another experiment that was conducted is using 20 cascade stages instead of 10 used in the above experiments. The results obtained using different features are illustrated in Figure 6-7 below.



Figure 6-7 Frontal head detection at stage 20 for different features.

From Figure 6-7, it can be observed that Haar-like feature based detector detects the head very accurately. However, both HOG and LBP features based detectors failed to detect the head with 20 stages of the cascade classifier. Despite that the training was to use 20, the training halts and returns the cascade detector with 16 stages, only due to insufficient training images. It means the function cannot generate enough negative samples, and therefore the training terminates early. Hence, the comparison of detection result was made based on the results obtained at stage 10. All the detected heads were counted as seen in Figure 6-8.

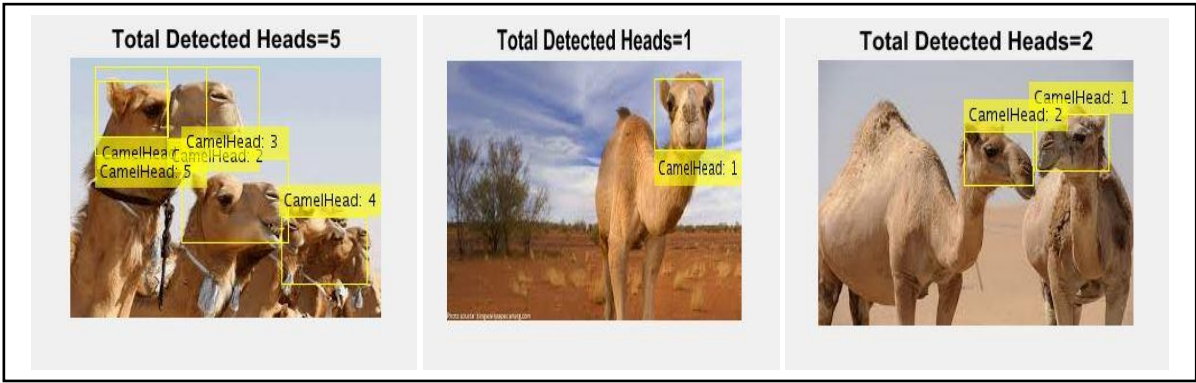


Figure 6-8 Examples of camel head counting.

A counter was set to count all the detected camel heads in a given image. The total number of detected heads are based on the detected bounding boxes. The total of counted heads will be inaccurate if there is a false detection (i.e. detection in random locations other than camel head/s) or a detection is missed.

6.2.1.7 Performance Evaluation

A number of experiments were conducted to evaluate the detection performance of the proposed method. Figure 6-9, Figure 6-10 and Figure 6-11 illustrate the detection rates of camel head detection using Haar, HOG and LBP features respectively.

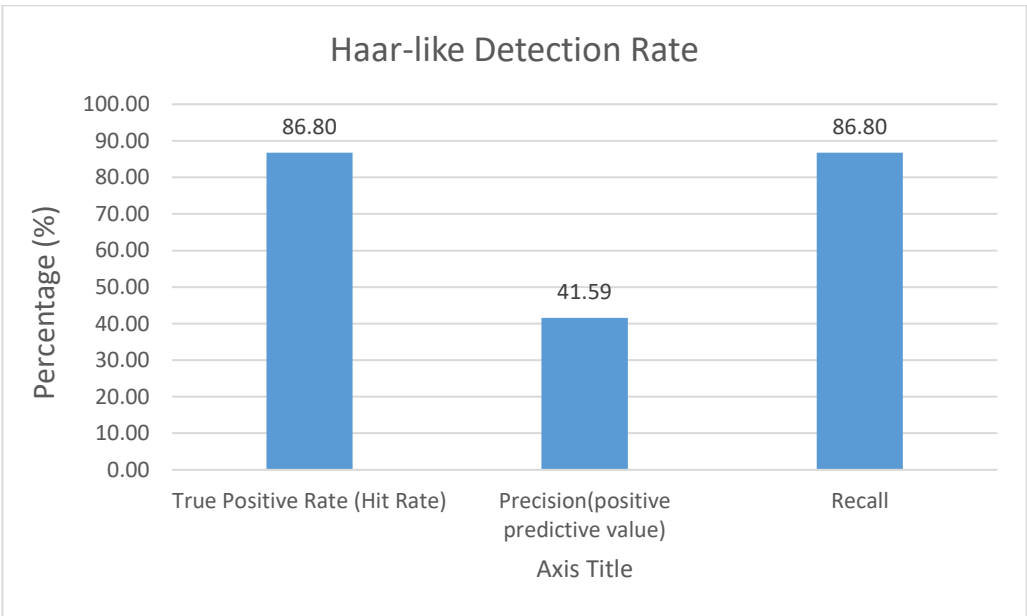


Figure 6-9 Haar-like features detection rates.

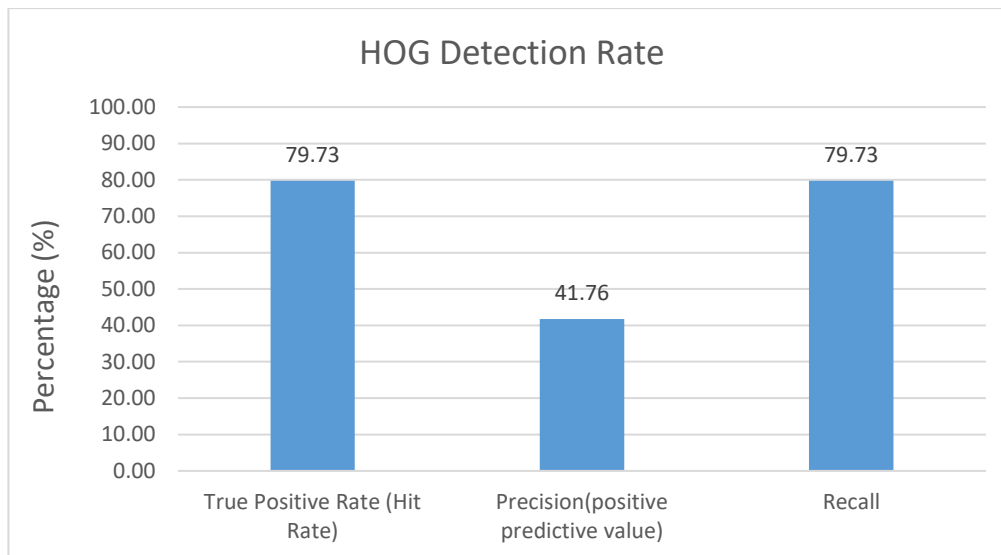


Figure 6-10 HOG features based detection rates.

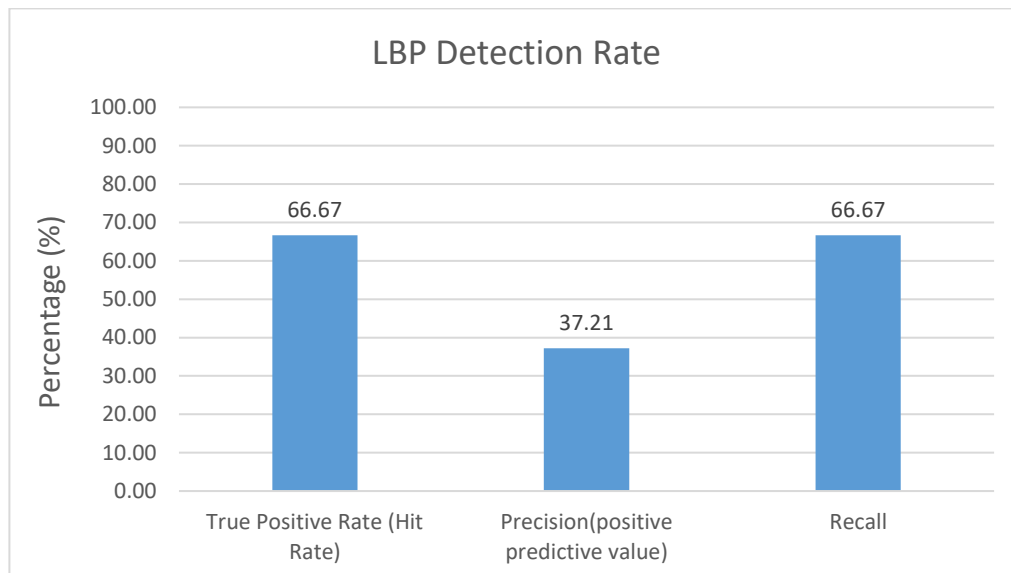


Figure 6-11 LBP features based detection rates.

Below equations were used to evaluate detection rate achieved by different features.

$$\text{True Positive Rate (TPR)} = \frac{TP}{TP + FN} \cdot 100$$

Equation 6-9

$$\text{Precision} = \frac{TP}{TP + FP} \cdot 100$$

Equation 6-10

$$Recall = \frac{TP}{TP + FN} \cdot 100$$

Equation 6-11

$$F. Measure = \frac{2 \cdot (Precision \cdot Recall)}{Precision + Recall}$$

Equation 6-12

Where TP refers to true positives, FN refers to false negatives, and FP is the false positive. The True Positive Rate (TPR) or Hit rate and recall are the probabilities of detection that are correctly identified. Based on the above analysis, Haar-like features detect the camel head more accurately, the true positive rate is 86.80% (Figure 6-9). HOG features occupied the second place for 79.73% (Figure 6-10). Only 66.67% is the hit rate for LBP features, (Figure 6-11). Precision measures the random error (i.e. the wrong detection in random locations in the image). More precisely, precision can be defined as the repeatability, or reproducibility of the measurement. From the above figures, 41.76%, 41.59%, and 37.21% are the precision rate for HOG, Haar-like, and LBP respectively. By balancing the precision and the recall rates; it is agreed that Haar-like features is more accurate than HOG and LBP features for camel head detection.

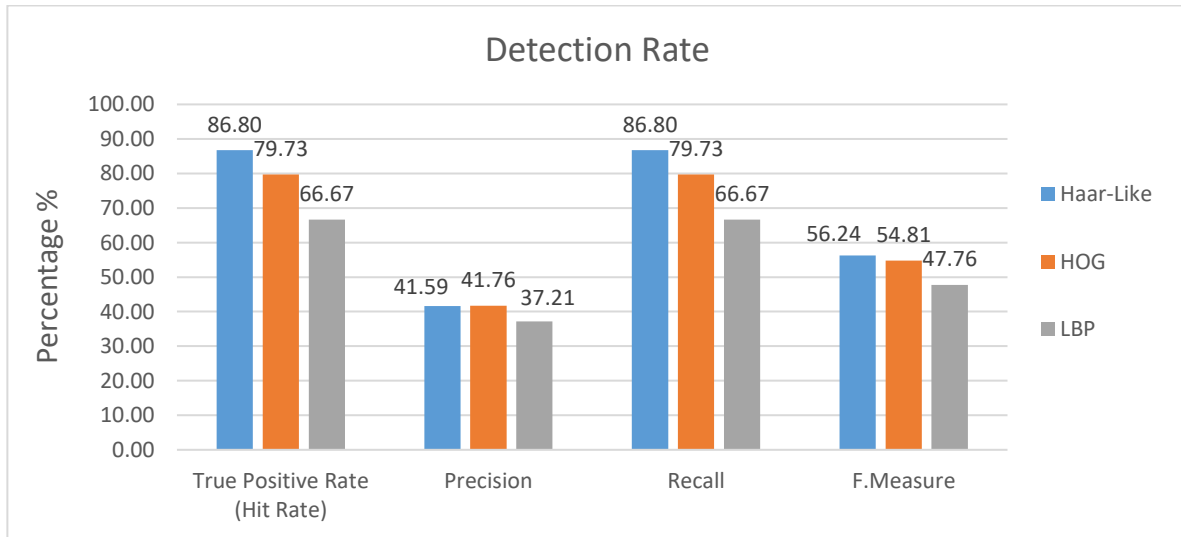


Figure 6-12 Detection rate comparison between Haar-like, HOG and LBP.

Despite the fact that Haar feature based approach achieves best detection rate among others, the detection rate is considered as significantly low. The evidence for that can be clearly seen in the F-measure in Figure 6-12, where the best rate achieved is only 56.24%. Moreover, F-measure shows that Haar and HOG rates are close to each other and the difference is only 1.43%. Additionally, the

slight difference in precision rate between Haar and HOG can be measured as inconsiderable versus true positive rate or recall rate. Precision rate for Haar feature based detector is higher than the expected because the detection suffers from false detection in some cases (i.e. there is/are detection/s in other locations other than camel head/s).

As far as training time and speed are concerned, Haar takes a long time to train the data. Unlike Haar, both LBP and HoG feature based approaches train and perform reasonably faster with less time consumption. Haar might take several hours or days to complete the training. The training time and speed depends mostly on the number of given training dataset besides the number of stages given for training.

Below is the computation time comparison between Haar-like feature, HOG feature, and LBP feature at training the last stage 10:

	Haar-Like	HOG	LBP
Training time (in seconds)	1450	119	35

Table 6-1 Computation time at stage 10 for camel head detection using Haar, HOG and LBP features.

The above table shows the significant differences between the times consumed to train the data using Haar, HOG and LBP. Taking into account the training time and the detection rate achieved, HOG features are better in detecting the camel head. It is also worth pointing out that the time consumption is different at each training stage. The above times are examples of the time consumed by the above features during the experiments for 10 cascade stages.

6.2.2 Deep Learning Approach

6.2.2.1 Region-Based CNN Method

The Region-based CNN or Regional Convolutional Neural Network (RCNN) consists of proposed Regions (R) and a Convolutional Neural Network (CNN) to correctly identify the objects in the image. It creates a bunch of bounding boxes or region proposals in the input image using a selective search technique to generate regions and check if any of them matches an object. Unlike Fast R-CNN, the proposed method uses Region Proposal Network (RPN) to generate regions (see section 6.2.2.1.1). Faster R-CNN is the model that will be used by the proposed approach to detect camel heads. RCNN and fast R-CNN models are considered as the most representative region proposal based models. The block diagram in Figure 6-13 shows the processes of detecting camels' heads using faster R-CNN.

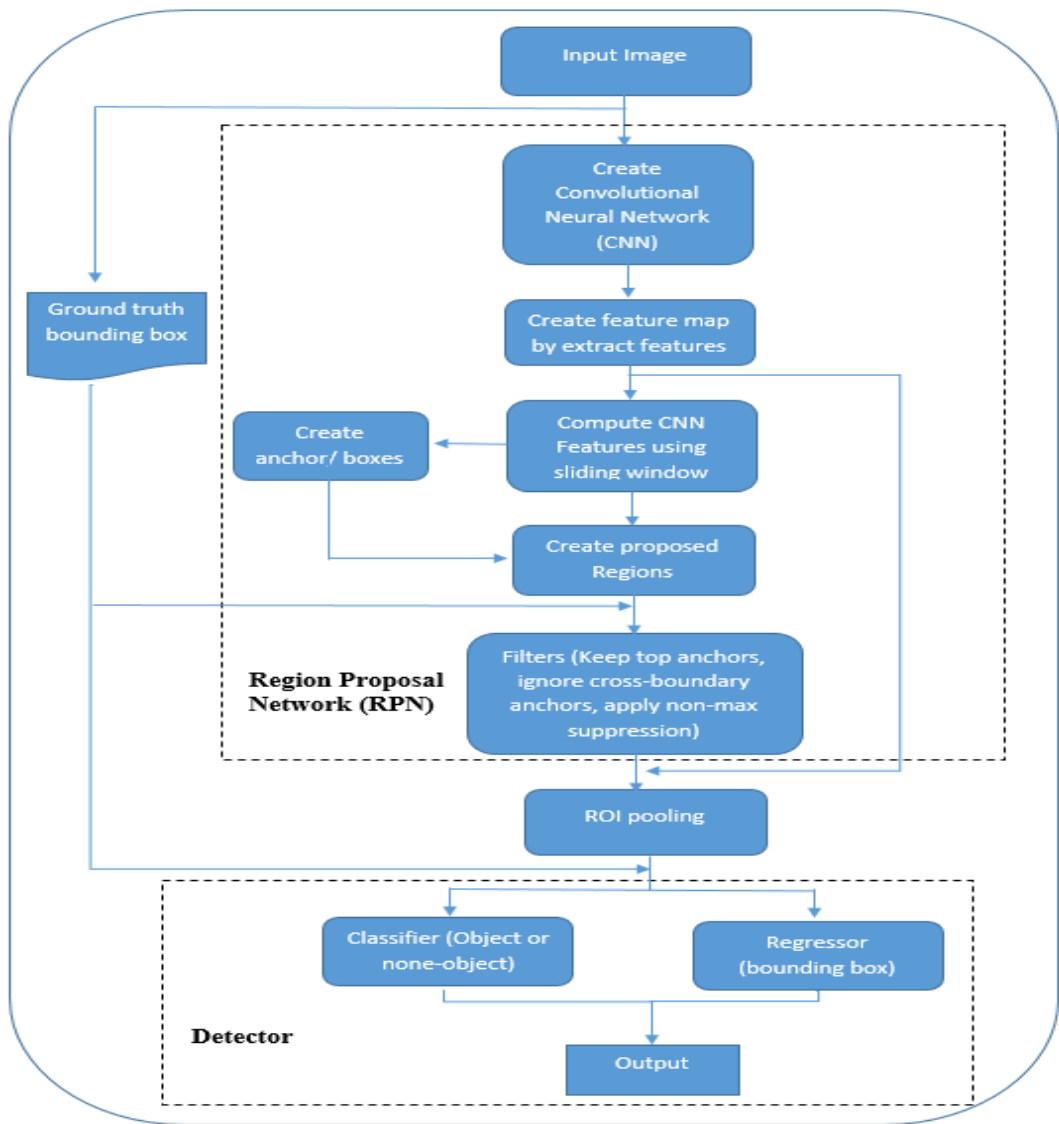


Figure 6-13 Overview of Faster R-CNN architecture.

Based on the above figure, the camel head detection system is composed of two stages. The first stage is the Region Proposal Network (RPN) that proposes regions, which is based on Convolutional Neural Network (CNN). The second part is the Fast R-CNN detector which uses the proposed regions from first stage.

6.2.2.1.1 Region Proposal Network (RPN)

The Region Proposal Network (RPN) is created by using several processes. In the first stage, a Convolutional Neural Network (CNN) is created. Figure 6-14 below shows the CNN structure used.

layers =			
15x1 Layer array with layers:			
1	''	Image Input	32x32x3 images with 'zerocenter' normalization
2	''	Convolution	32 5x5x3 convolutions with stride [1 1] and padding [2 2]
3	''	ReLU	ReLU
4	''	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0]
5	''	Convolution	32 5x5 convolutions with stride [1 1] and padding [2 2]
6	''	ReLU	ReLU
7	''	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0]
8	''	Convolution	64 5x5 convolutions with stride [1 1] and padding [2 2]
9	''	ReLU	ReLU
10	''	Max Pooling	3x3 max pooling with stride [2 2] and padding [0 0]
11	''	Fully Connected	64 fully connected layer
12	''	ReLU	ReLU
13	''	Fully Connected	2 fully connected layer
14	''	Softmax	softmax
15	''	Classification Output	crossentropyex

Figure 6-14 Convolutional Neural Network (CNN) Layers.

This network has a total of 5 layers; 3 convolutional layers and 2 fully connected layers. The CNN is used to extract the features of the input images to create feature maps on the last convolutional layer. An example of extracted features from first convolutional layer at initial stage is shown the Figure 6-15.

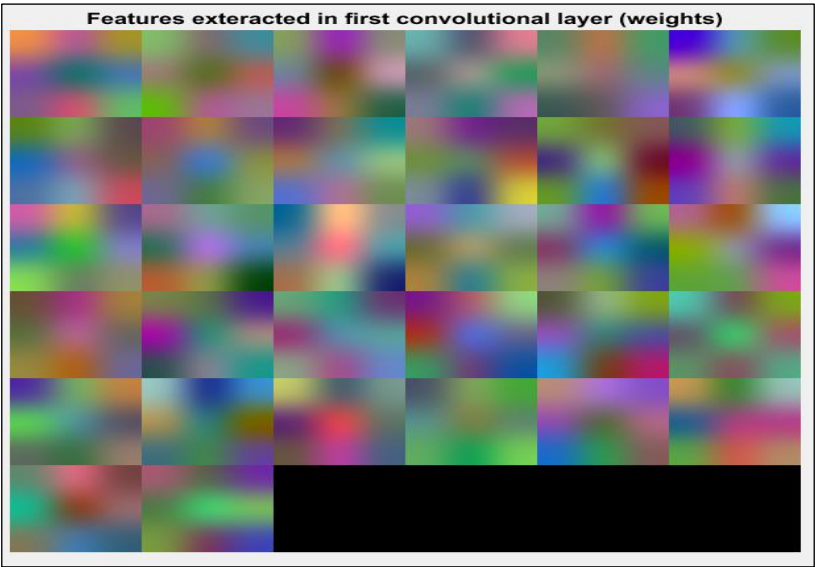


Figure 6-155 Example of initial features extracted from first convolutional layer.

Figure 6-16 illustrates the process of creating Region Proposal Network (RPN).

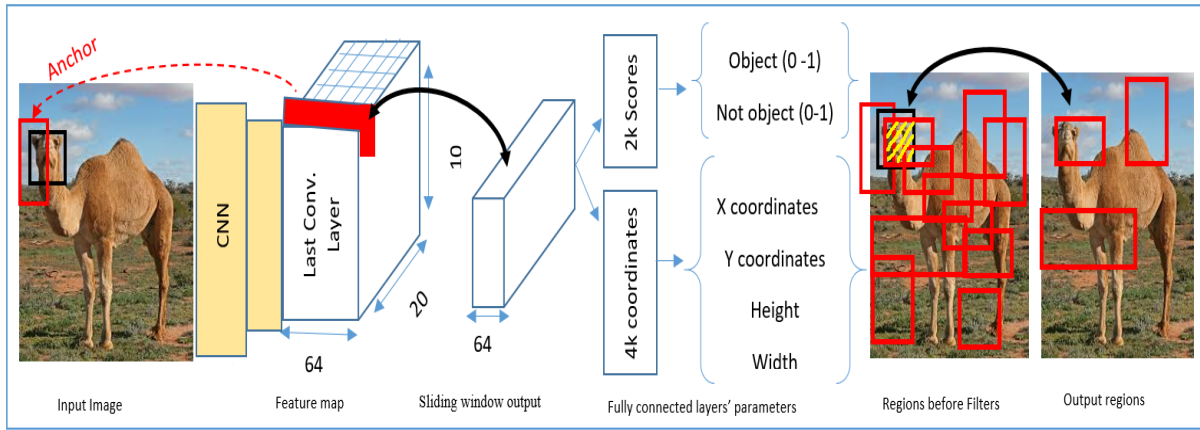


Figure 6-166 Region Proposal Network (RPN) pipeline.

The input images are passed through the convolutional network up until an intermediate layer, ending up with a set of convolutional feature maps on the last convolutional layer. The feature map is divided into multiple squared tiles. A sliding window runs over the feature maps to generate region proposals. The size of the sliding window is 3x3 and the output of each sliding window gives a set of 9 potential bounding boxes called anchors and object confidence scores. The scores represent the probability of objectness (contains object or not) whereas, coordinates represent 4 offset values of the anchor box (w, h, x, y).

Where w = width, h = height and (x, y) = centre co-ordinates.

The anchors are generated in three different box scales; 128, 256 and 512 to fit the object inside the box. Each of these scales has three different height and width ratios; 1:1, 1:2 and 2:1. Hence, each possible anchor generated in RPN is represented in 9 different sizes (see 3.5.4). This has resulted in many anchors with no object and some anchors are outside the image boundary. Anchor filters method (see 6.2.2.1.2) is used to eliminate unnecessary anchors and to create region proposals. In this method, all anchors that cross the image boundary were ignored. Additionally, all remaining anchors were categorised into foreground or background objects. An object was considered a foreground object if the anchors overlap a ground truth object with an Intersection over Union (IoU) bigger than 0.5 and those with less IoU threshold were considered background. Then, the Non-Max Suppression (NMS) algorithm is applied to sort the proposals based on score.

6.2.2.1.2 Filters

During generation of regional proposal network in faster R-CNN, there is a high possibility to have wrong proposed regions. Additionally, the probability of having overlap anchors is also high.

Moreover, many anchors might be either have no objects or some of them may be outside the image boundary. Anchor filters are a way of eliminating unnecessary anchors. There are two methods: ignoring cross boundary and Non-Max suppression, that can be used to reduce redundant and needless anchors. In first method, any anchor that exceeds the boundary of the image is ignored.

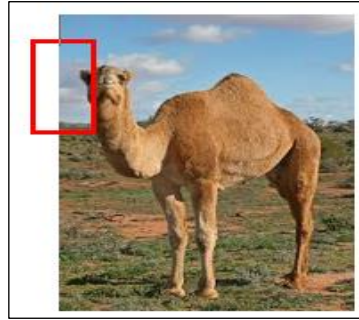


Figure 6-177 Cross boundary anchor.

Once all cross boundary anchors are eliminated, a Non-Max Suppression (NMS) is applied on the remaining anchors. The NMS uses the measure Intersection over Union (IoU) to exclude anchors. The ROI box is used as the ground truth in this calculation. In this method, the anchor is ignored if the anchors overlap a ground truth with an Intersection over Union (IoU) less than 0.5.

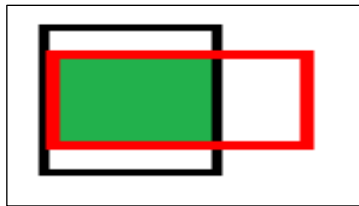


Figure 6-188 Example of IoU.

The calculation of IoU is defined as follow:

$$IoU = \frac{A \cap GT}{A \cup GT}$$

Equation 6-13

Where A is proposed anchor and GT is ground truth box.

Those with less IoU threshold were considered as belonging to the background and were discarded. Then, the higher score of anchor is used to apply None-Max Suppression (NMS) algorithm where one box is selected for each proposed region. Figure 6-19 illustrates example of applying NMS.

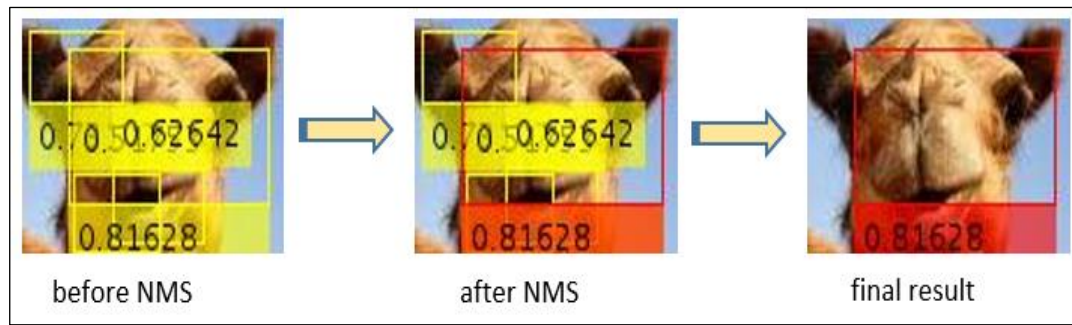


Figure 6-199 Example of Non-Max Suppression.

6.2.2.1.3 Region of Interest (ROI) Pooling

Last fully connected layer of CNN is the layer used for classification. It has fixed size and can only accept same size features maps. The output of the above processes generate regions with different size. In other word, the output of the proposed regions require different size of CNN for the last fully connected layer. This will led to performance related problem, overfitting and making real time object detection difficult to implement. Additionally, it is impossible to have end-to-end training which is used to train all system components in one run to achieve better results. Hence, Region of Interest (ROI) pooling layer is used (see 3.5.2) to overcome these limitations. Both RPN and feature map that are created by CNN are used as input to the ROI pooling layer. This layer applies max pooling operation (as describe in 3.5.3) on all regions. The output of ROI pooling is a fixed size feature map for each proposed region regardless the size of input.

6.2.2.1.4 Detector

After obtaining fixed sized feature maps, by applying ROI polling, the feature maps were used for detection and classification. Fast R-CNN is used as a detector to detect camel head. It uses last fully connected layer of CNN to classify the regions, either as camel head or background, and to better adjust the bounding box for the camel head.

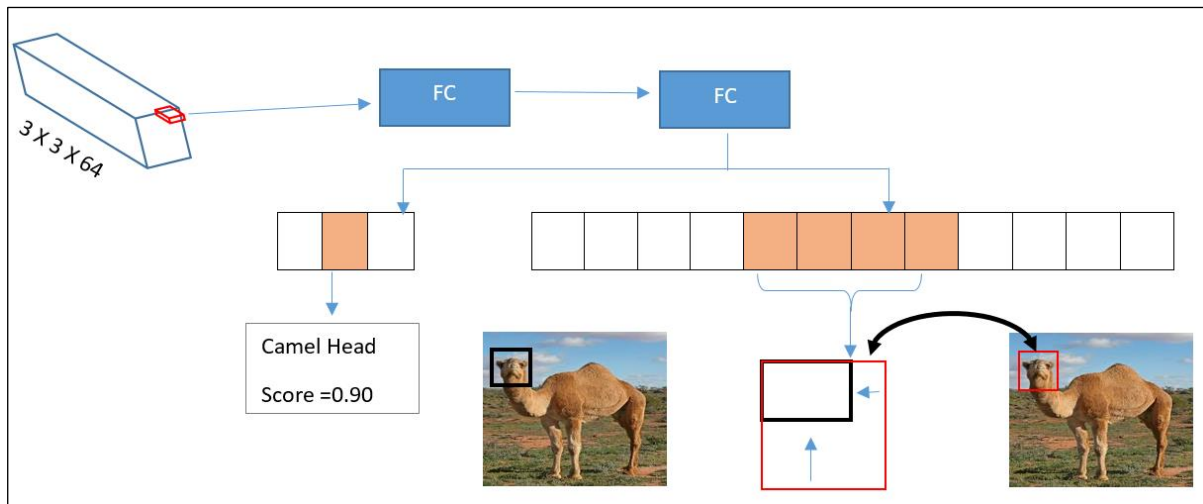


Figure 6-20 Fast R-CNN Detector architecture.

The detector takes the feature map of each proposal from the fixed size feature map and uses them as input to fully connected layers. The first fully connected layer has $N + 1$ units where N is the number of classes and the extra one for background. The second fully connected layer is used for regression prediction. It has 4 units to predict the bounding box (X, Y, Height and Width) for each class (N). Then, ground truth boxes were used against proposals to calculate the Intersection over Union (IoU). Proposals that have IoU threshold greater than 0.5 with any ground truth box get assigned to that ground truth. Those with lower threshold get labelled as background. There are 4 steps to train the entire network.

- 1- Train a Region Proposal Network (RPN) using the training dataset.
- 2- Train Fast R-CNN detector using proposals generated from RPN for all training images.
- 3- Re-train RPN using weight sharing with Fast R-CNN.
- 4- Re-train Fast R-CNN using Updated RPN.

6.2.2.1.5 Experiments and Results

Turing now to the experimental results of proposed system, a number of experiments were conducted using Camel dataset given in section 6.2.1.1 to detect camel heads. The dataset has a total number of 418 images and split into training and testing sets. The training set contains 292 images and the remaining 126 images were used for testing set. The experiments runs on GPU with 1582000 clock rate (KHz) and 1.1811e+10 memory capacity. The initial learning rate has sets at 0.001 and a Momentum with fraction value 0.9 is used for the Stochastic Gradient Descent (SGD) optimizer. Initially, a number of experiments were conducted using different configurations. The above configuration provides best result using the proposed method. Then, the results were compared

against another R-CNN (Fast R-CNN) model results. The figures below show detection results of proposed method and Fast R-CNN respectively.

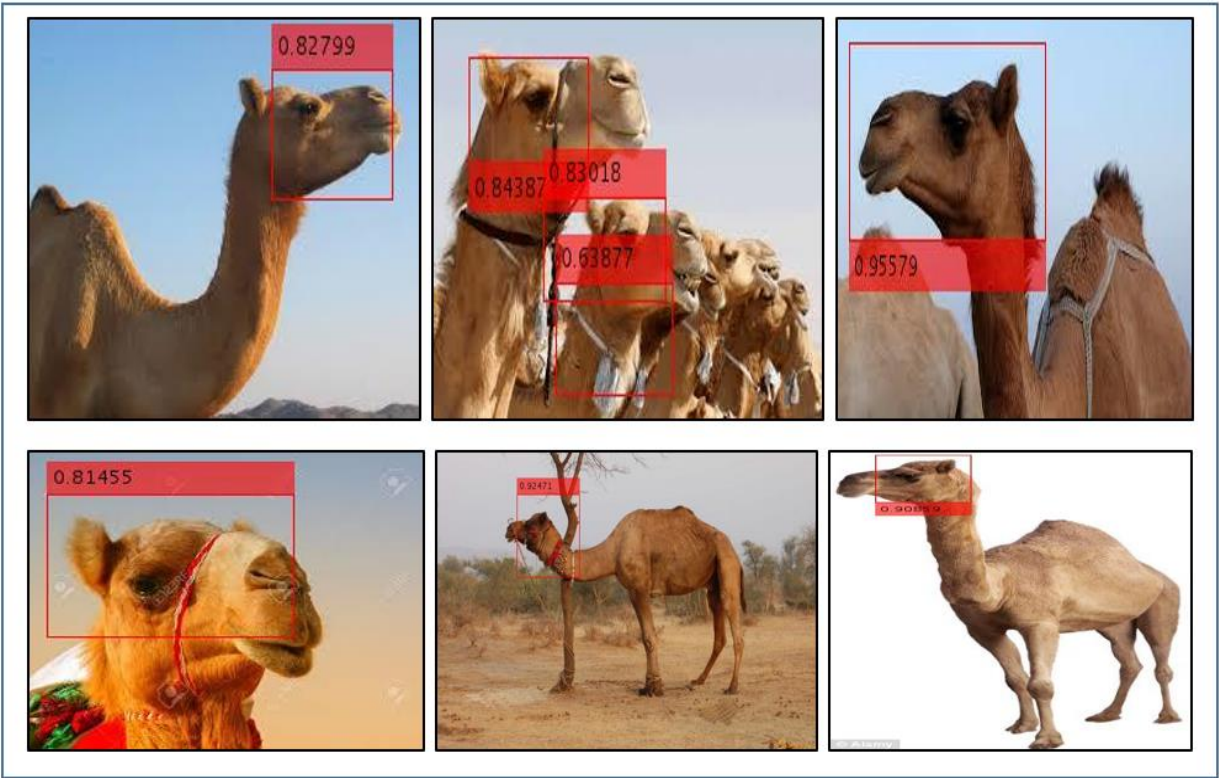


Figure 6-21 Detection camel head using Faster R-CNN (the proposed method).

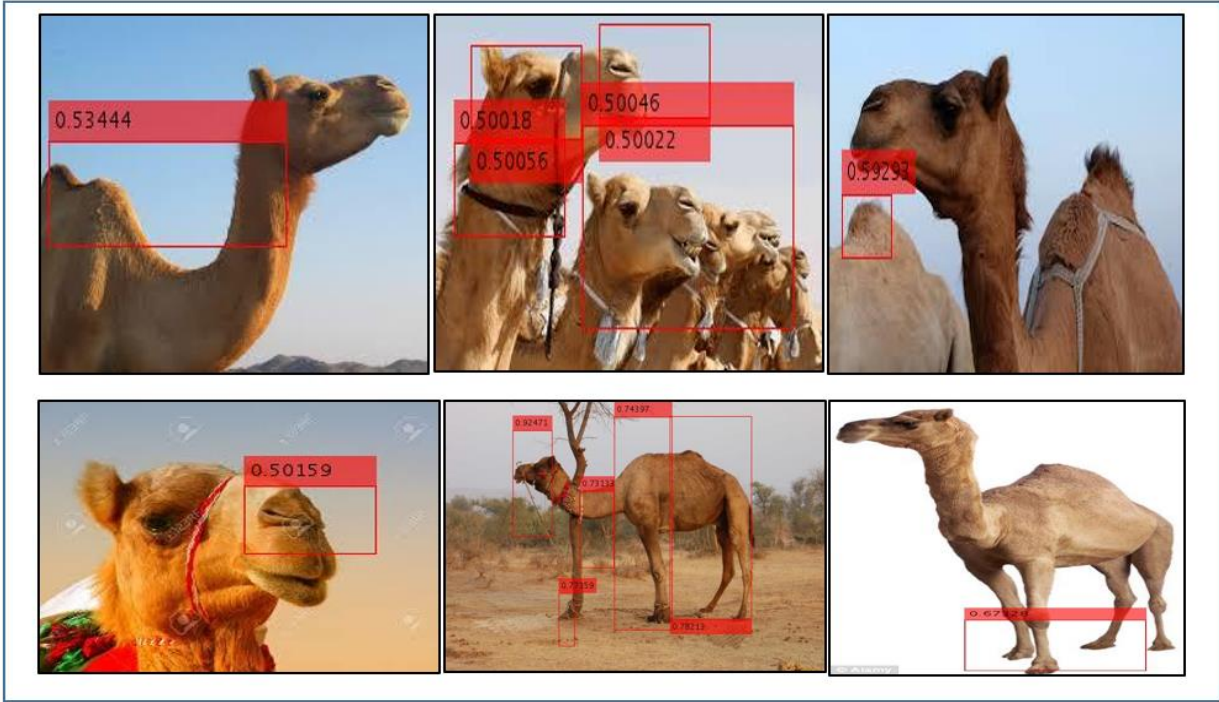


Figure 6-22 Detection camel head using Fast R-CNN.

Comparing the two results, it can be observed that Fast R-CNN failed to detect some of camel heads. Moreover, Fast R-CNN is affected by noise and flagged few false alarms comparing to proposed method. On the other hand, Faster R-CNN performs slightly better in detecting multiple heads in one image.

After conformational analysis of the proposed method, it was necessary to use further performance related matrices to evaluate the performance of proposed method against Fast R-CNN. Figures 6.23 provide detection rates for the proposed method and fast R-CNN method respectively. The hit rate or True Positive Rate (TPR), miss rate or False Negative Rate (FNR), precision, recall, and F.Measure were used to evaluate the performance of each model.

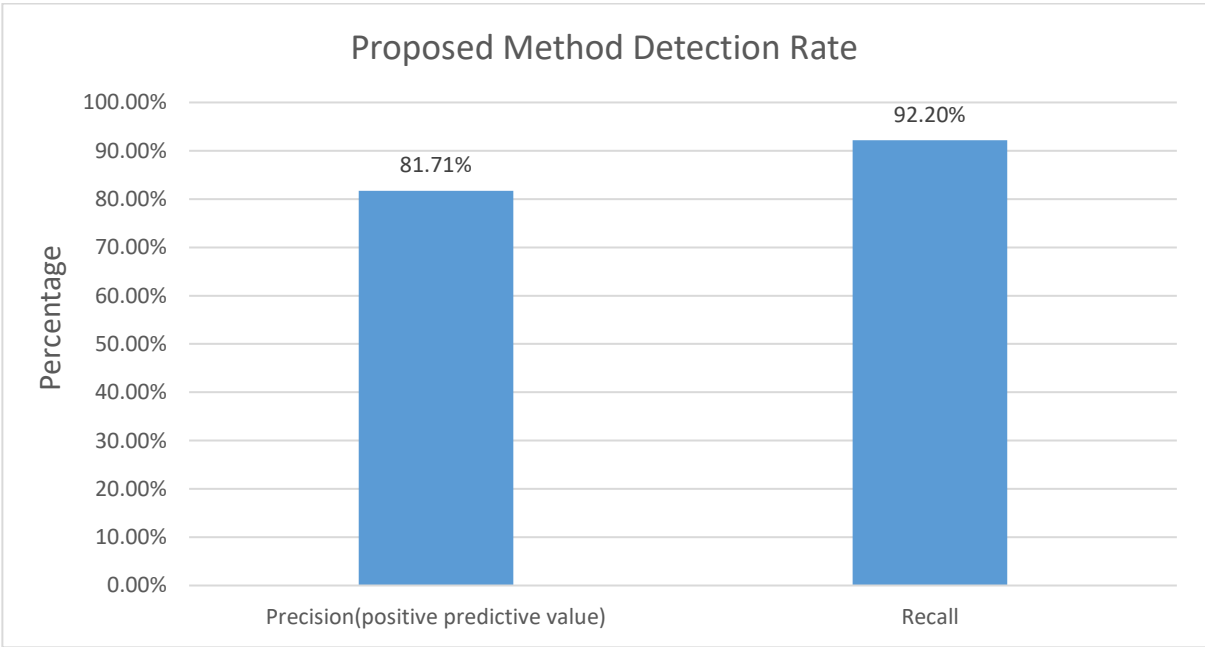


Figure 6-23 Proposed method detection rate.

It is apparent from the above figure that the proposed method was able to detect over 92% of the heads. Only a minority of camel heads (7.8%) were missed and not detected. Another interesting result is that the proposed method achieves almost 82% in precision measurement. This means less wrong detection of camel heads. Although the proposed method (Faster R-CNN) has been used in different application domains such as stop sign detection (with pre-trained network), we propose to use it in animal detection to detect camel head without using a pre-trained network.

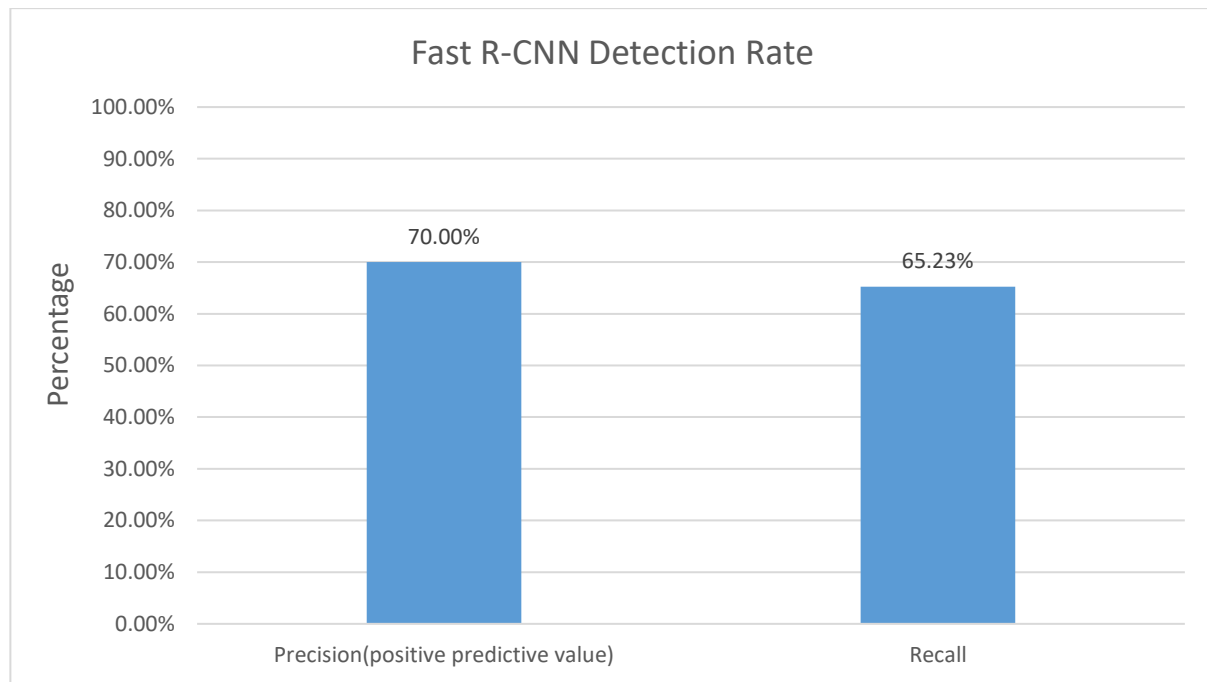


Figure 6-24 Fast R-CNN Detection Rate.

In contrast to proposed method, Fast R-CNN has achieved low detection rate in both measures. Only 65% of heads were detected. It is worth noting that, false head detection rate is higher in this model. It is worth noted that both models (the proposed method and Fast R-CNN) have used the same CNN with same configurations in this experiment. A summary of detection rates are set out in Table 6-2 and in Figure 6-25.

	True Positive Rate	False Negative Rate	Precision	Recall	F.Measure
Proposed Method	92.2%	7.8%	81.7%	92.2%	86.6%
Fast R-CNN	65.2%	34.8%	70%	65.2%	67.5%

Table 6-2 TPR, FNR, Precision, Recall and F.Measure performance values.

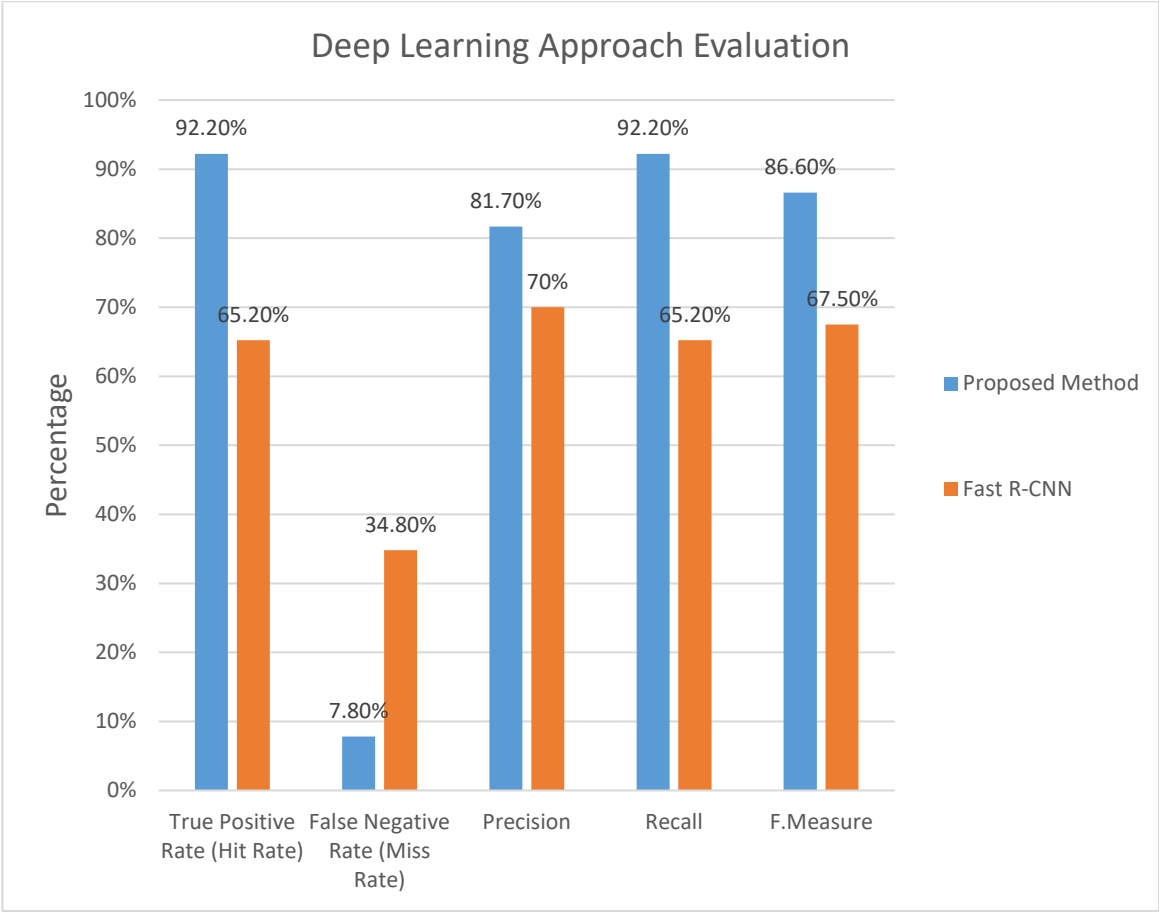


Figure 6-25 Performance analysis plot on camel detection.

What is interesting in this data is the significant difference in recall measurement between proposed method and Fast R-CNN. The proposed method was able to detect most of the camels' heads in testing dataset with a recall percentage over 92% compared with 65% for Fast R-CNN.

Interestingly, there was also a difference in the precision result between the two model results. The proposed method has obtained almost 82% in this measurement whereas Fast R-CNN has more noise which drop the rate to 70%. Similarly, F-Measurement for proposed method is 86.6% versus 67.5% for Fast R-CNN. The justification of this result could be due to the different structure between the two methods. Fast RCNN uses selective search technique [126], to generate the possible object location (the proposed regions); in which many false regions could be included. Whereas, Faster RCNN uses CNN to generate the proposed regions (see section 6.2.2.1.1) where all the proposed regions are checked against the ground truth, before training the detector, to eliminate unnecessary bounding boxes. Another justification for the above result could be the training steps that involved in training the detector. Unlike Fast R-CNN which uses one step to train the detector, the proposed method uses four steps: training Region Proposal Network (RPN), training detector using RPN from

first step, re-training RPN using weight sharing with the detector and finally re-training the detector with the updated RPN.

6.3 Chapter Conclusion

This chapter presented two approaches to Camel Head detection. In the first approach a well-established Viola-Jones face detection algorithm is adopted and used with HOG features. The HOG results were benchmarked with Haar and LBP features. A deep learning approach was used as the second approach to detect the camel heads. This approach used state of art CNN models to detect the head of camel. The result of Faster RCNN is compared with Fast R-CNN.

The outcomes of first approach showed that the detection rates are low. Furthermore, HOG performs better at describing the shape of the object. Whereas, Haar works better at describing object shading and texture. The hit detection rate indicates that the Haar features are better than HOG and LBP features. On the other hand, F.Measure proves that the difference between HOG and Haar is only 1.43%. Another noticeable result is that the Haar features have taken very long time to train the classifier. The time consumed by Haar features was 1450 seconds comparing with 119 seconds and 35 seconds for HOG and LBP respectively. Overall, HOG performs better when training time is considered.

As far as the deep learning approach is concerned, the proposed method uses CNN and trained the network from scratch rather than using pre-trained model. Despite the fact that training a network from scratch requires enormous amount of database for good detection rate, a small database is used for network training. The most interesting finding was that more than 92% of camels' heads were detected using proposed method. Another significant achievement for proposed method is that the false negative rate is less than 8% compared with almost 35% for other model. It is worth noting that there were significant differences in all other measurements (i.e. recall, precision and F.Measure) in favour of the proposed method. In summary, these results show that proposed method outperformed Fast RCNN by recording better detection rate of all measurements used to evaluate the performance.

The result of this chapter indicates that deep learning approach is better than feature based approach for camels' heads detection.

Chapter 7

Conclusion and Future Work

This thesis presents a number of novel approaches in chapters 4, 5 and 6 for object detection, recognition and classification. This is the first time that computer vision and artificial intelligence approaches have been used together in a comprehensive manner to classify mixed industrial dust samples and road surfaces; and to detect and recognise camel heads in different environments, including desert.

The first contribution made in this thesis was a novel algorithm for industrial dust type classification. The system helps classifying different dust types generated in multi-sector industrial environment that includes; chemical, petrochemical, logistics, metals, steel, iron, rubber, plastics, automotive, food processing, power stations, and other materials production facilities. Such a system can help to give an indication about the dust types that is produced more and hence exceeds safety regulations in such environments. Further, the results can be used for alerting or reporting an increase of air pollution in such environments to enable early warning generation for the safety of citizens. Such warnings will also help the government, industrial area management authorities or concerned public action groups to take proper action to control the pollution at early stage and save human, organisms and environment.

The second contribution made in this thesis was a novel approach for road surface type recognition for use in autonomous vehicle systems. Different road surfaces were classified using state of art techniques and a number of novel approaches. Stability of any vehicle depends on the road surface the vehicle is moving on. Therefore, identifying the type of surface is essential and extremely important, especially for autonomous vehicle that needs to be controlled and navigated accordingly. Such systems can be adopted in any vehicle regardless of whether they are fully-autonomous and will have a high impact on improving the safety of passengers and vehicle stability, especially in self-driven vehicles. Moreover, the system can be integrated with other safety systems, i.e. Advanced Driver Assistance Systems (ADAS), to reduce car accidents and improve the safety of autonomous vehicles. This study has demonstrated, for the first time, that the road surface can be recognised with significant level of accuracy using traditional feature based machine learning algorithms.

The third and final contribution was focused on camel detection and recognition in different environments including a desert environment. A novel algorithm was developed to detect and

recognise camel heads irrespective of the environment and angle of orientation of the camel's head. Such systems can help in alerting and locating missing camels. Additionally, it can be used to save the lives of camels from dangers and theft. Detecting and recognising camels near/on roads is crucial. It helps in preventing animal-vehicle collisions and save lives of camels and passengers.

7.1 Conclusion

A novel technique was proposed for industrial dust classification in Chapter 4 of this thesis. Four state-of-the-art algorithms were tested in the segmentation of the foreground dust particles from the background of the images captured by a microscopic camera. However, three of the algorithms tested were challenged in their ability to segment all different categories of dust types investigated. Colour features were initially extracted from all images of all 8 dust type categories. It was observed that colour features alone were unable to classify all classes to a satisfactory level based on experimental results obtained. For better result, colour and texture features were combined together in one feature vector. This combination managed to differentiate between all dust classes effectively. The results proved that, better dust classification can be achieved by merging colour and texture features together. It was also shown that the importance of feature selection in maintaining same accuracy with a minimum number of features without compromising the performance. The evidence of that can be clearly seen in the classification accuracy results obtained by using all 278 features and only 10 selected features, which were very similar. It was proven that feature selection reduces the complexity of dust classification process and hence will significantly reduce the processing time in testing and validation. This gives a clear conclusion that proposed system is fast in processing time and can be used for real-time dust classification. It was proven that the Random Forest classified was the best to use to classify the dust types using the abovementioned feature extraction and reduction approaches.

Two novel techniques were proposed and implemented in chapter 5 to recognise five most commonly used road surfaces. It was observed within the literature review that previous research did not investigate the classification of many different types of road surfaces under varying environmental conditions. Hence the existing datasets were enhanced further by capturing new data within the context of this research. An innovative algorithm was developed to segment the road surface from irrelevant objects in the image, and to select Region of Interest (ROI) defining the road area that will be used for classification. The road type recognition was carried out using two novel approaches. In the first technique, we integrated colour, GLCM, LBP and Gabor features of ROI to classify the road surfaces. To validate the performance of the proposed technique, the use of each feature type has been tested individually before combining all the datasets to investigate the impact of feature

combination. Although a feature selection algorithm was adopted to improve classification performance of this integration, the results showed that neither individual features nor combined features can adequately classify different road surfaces within a traditional feature based machine learning approach. In the second technique, a state of art model was adopted to classify road surfaces using deep learning. It uses the same ROIs generated from road segmentation algorithms used in the first approach. Due to the limited data available for experiments and in order to speed up the process and reduce the training time, a well-known pre-trained neural network model was adopted with transfer learning in this approach. Each ROI goes through a series of layers for feature extraction and feature computation. The last layer of the network uses all computed features from previous layers to classify the ROI into one of the road surface types. The experimental results of this technique was benchmarked with other transfer learning models and with the same pre-trained model used on images with no ROI segmentation. The proposed technique shows a significant improvement in performance against other algorithms. The proposed system is implementable in real-time, especially in autonomous vehicles, wherein human safety cannot be compromised. This research will serve as a base for future studies in road surface classification.

Chapter 6 of this thesis, proposed two approaches to detect and recognise camel heads. First approach is based on camel head features detection where well-known feature extraction algorithms were adopted for object detection and subsequent recognition. For comparison purposes, Haar-like, HOG and LBP features were extracted and used to train cascade detector. The results of this investigation show that an acceptable hit rate was achieved. However, the high false rate was shown to reduce the overall accuracy rate. It was also shown that the processing time had a high impact on the overall performance of the detection approach. The most obvious research outcome to emerge from this study is that feature based detection requires more data to train, more cascade stages if one is to improve the detection rate. However the accuracy rates obtainable is rather limited.

In order to overcome the limitation of the first approach the research conducted in this thesis developed a novel algorithm to detect and recognise camel heads. This algorithm is based on deep learning where a convolutional neural network is used. It has the ability to accept any image size to extract different image features and create feature maps. Likewise, it can detect the heads regardless head size or number of heads to be detected in a single image. Despite the limited data used in this research, a new network was created and trained from the scratch. The investigation of proposed method has shown that respectable detection accuracy was obtained. It was also shown that a low false rate was achieved. In summary, these results show that proposed method outperformed other similar method by recording better detection rate of all measurement metrics used to evaluate the

performance. One of the more significant findings to emerge from this study is the capability of the system to perform end-to-end training and overcome the common bottlenecks of other similar systems. The proposed system can be used in real-time detection especially in animal tracking in different environments to save animals and human lives and to reduce accident that caused by animals.

7.2 Future Work

Object detection, recognition and classification are a wide, complicated and sometimes heavily involved multidisciplinary fields of research. Form the scope of this research, it is clear that not all parts of this task could be incorporated in this study. The outcome of this investigation has led to a number of original contributions advances the existing state-of-art technology in different areas of object detection, recognition and classification. Aside from the contributions made in this thesis, there are further possibilities for future extension and improvement of the proposed algorithms.

For the dust sample classification, a number of extension possibilities can be included to the proposed method. The current study was unable to segment and classify the dust based on shape. A shape based segmentation algorithm can be developed and integrated with proposed algorithm. This integration might improve segmentation process. Experiments can be conducted form the same dust samples to analyse the differences in the result and to study the potentiality of merging the two algorithms to improve overall system performance. Another opportunity to extend the functionality of proposed method is that identification of particle size. Particle size can be used to compare with World Health Organisation (WHO) and Oman's Air Pollution Standards (OAPS) or any other organisation to help the government to take an appropriate action. Further this study is limited by the lack of information on the chemical composition of dust particles due to restricted access to resources and due to financial reasons. Such information will help to identify the name of each dust particles so that we can have better evaluation process, ensuring the classifications result in dust particle type recognition.

In chapter 5 of this thesis, the proposed algorithm identified the road surfaces type at day-time, but under varying levels of illumination. This work can be extended so that road surfaces are not identified at day-time only but rather also at night, using night vision cameras. Night-time road surface classification could open the door for other challenges in this domain. Due to the importance of road surface recognition to the autonomous vehicles, future research should therefore concentrate on the investigation of road surface detection under different weather conditions i.e. rainy and heavy rainy weather. The proposed system has only examined and evaluated the algorithms under limited

illumination conditions/variations: sunny, partly cloudy and cloudy at day-time. Consequently, further studies, which take these variables into account, will need to be undertaken. It is also possible to include more data and new type of road surfaces to improve the robustness of proposed system. On the contrary, the proposed road surface recognition system can be deployed and integrated with existing advanced driver assistance systems to provide added safety to the vehicle and passengers.

To address the problem of hidden or overlapping camel heads, the proposed camel head detection system can be further improved to enable the detection and recognition of camel body. The uniqueness of camel body shape feature could play a significant role in improving the detection accuracy. The proposed algorithm was not specifically intended to evaluate aspects related to weather or illumination conditions. Such evaluations can be covered in future studies. It is also possible to include more camel and none camel images to the database to improve training and hence the overall detection performance. Another enhancement that can be applied to the existing system is using real-time videos, as the system is currently demonstrated and evaluated based on still images. In future investigations, it might be possible to fine-tune the proposed method to detect a camel from a top/elevation view, in particular having applications in drone imaging. Supplementary experiments to evaluate the performance of proposed deep learning approach under pre-trained models had been originally planned to be investigated but was not possible due to limited time. Likewise, new features of camel head could not be tested on first approach for the same reason.

References

- [1] Hazard Prevention and Control in the Work Environment: Airborne Dust. [online]. [viewed 05/10/2014]. Available from:
www.who.int/occupational_health/publications/en/oehairbornedust.pdf
- [2] The World Bank,R,2015. Country and Lending Groups. [online]. [viewed 02/10/2014]. Available from:
<http://data.worldbank.org/about/country-and-lending-groups>
- [3] Antoine Mansour,2013. PLANNING FOR ECONOMIC DIVERSIFICATION IN OMAN.[online].[viewed 02/10/2014]. Available from:
<http://www.oea-oman.org/03-Mansouier.pdf>
- [4] Central Bank of Oman, R.,2010. The Vision for Oman's Economy: Oman 2020.[online].[viewed 02/10/2014].Available from:
http://shodhganga.inflibnet.ac.in:8080/jspui/bitstream/10603/3836/11/11_chapter%206.pdf
- [5] Abdul-Wahab, S. A. et al., March 2011. Evaluating the performance of an integrated CALPUFF-MM5 modeling system for predicting SO₂ emission from a refinery. *Springer*, 10(1007), pp. 841-854.
- [6] Jumb, V., Sohani, M. & Shrivas, A., February 2014. Colour Image Segmentation Using K-Means Clustering and Otsu's Adaptive Thresholding. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, 3(9), pp. 72-76.
- [7] Bora, D. . J. & Gupta, A. K., December 2014. A New Approach towards Clustering based Colour Image Segmentation. *International Journal of Computer Applications*, 107(12), pp. 23-30.
- [8] Mathur, G. & Purohit, H., Dec. 2014. Performance Analysis of Colour Image Segmentation using K-Means Clustering Algorithm in Different Colour Spaces. *IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)*, 4(6,), pp. 01-04.
- [9] Xinyan, M., Ying, Z., Yanxiao, H. & Binjie, S., May 2009 . Colour Image Segmentation Method Based on Region Growing and Ant Colony Clustering. *IEEE*, Volume 1, pp. 173 - 177 .

- [10] Yang, C. et al., Aug. 2014. Shape-Based Classification of Environmental Microorganisms. *IEEE*, pp. 3374 - 3379.
- [11] Long, F., Peng, H. & Myers, E., 2007. AUTOMATIC SEGMENTATION OF NUCLEI IN 3D MICROSCOPY IMAGES OF C.ELEGANS. *IEEE*, pp. 536 - 539.
- [12] Begelman, G. et al., 2004. CELL NUCLEI SEGMENTATION USING FUZZY LOGIC ENGINE. *IEEE*, Volume 5, pp. 2937 - 2940.
- [13] C. Igathinathane, et al., December 2009. Machine vision based particle size and size distribution determination of airborne dust particles of wood and bark pellets. *Elsevier*, 196(2), p. 202–212.
- [14] C. Igathinathane, L.O. Pordesimo, W.D. Batchelor, Major orthogonal dimensions measurement of food grains by machine vision using ImageJ, *Food Research International* 42 (2009) 76–84.
- [15] C. Shanthi, Porpatham, R. K. & N. Pappa, 2014. Image Analysis for Particle Size Distribution. *International Journal of Engineering and Technology (IJET)*, 6(3), pp. 1340-1345.
- [16] Andersson, T., Thurley, M. J. & Carlson, J. E., January 2012. A machine vision system for estimation of size distributions by weight of limestone particles. *Minerals Engineering*, 25 (1), p. 38–46.
- [17] R.C. Gonzalez, *Digital image processing (MATLAB edition)*, Electronic Industry Press, Beijing, 2005, 9.
- [18] What is UCS(Uniform Colour Space)?. [online].[viewed 12/02/2015].
Available from:
https://www.nippondenshoku.co.jp/web/english/colour_story/07_what_is_ucs.htm
- [19] Image Segmentation 2012.[R].[online].[viewed 03/07/2014].
Available from:
https://homes.di.unimi.it/ferrari/ElabImm2011_12/EI2011_12_16_segmentation_double.pdf
- [20] Y. Li, “Review of Colour Image Segmentation Technologies”, *Scientific and technological information development and economy*, China, 2008, 18(10), pp. 1005-6033.

- [21] Smola, A. & Vishwanathan, S., 2008. *INTRODUCTION TO MACHINE LEARNING*. 1 ed. Cambridge,: the press syndicate of the university of cambridge.
- [22] Ejiri, M., 1990. Machine Vision Technology: Past, Present, and Future. *IEEE*, XXIX - XXXX vol.1(185), p. 12.
- [23] C. Igathinathane, U. Ulusoy & L.O. Pordesimo, January 2012. Comparison of particle size distribution of celestite mineral by machine vision Σ volume approach and mechanica sieving. *Elsevier B.V.*, Volume 215–216, p. 137–146.
- [24] Chen, S., Cremers, D. & J. Radke, R., December 2012. Image segmentation with one shape prior A template-based formulation. *Image and Vision Computing*, 30(12), p. 1032–1042.
- [25] Julia, C., Moreno, R., Puig, D. & Garica, M. A., 2011. Shape-based image segmentation through photometric stereo. *Computer Vision and Image Understanding*, 115(1), pp. 91-104.
- [26] Mogireddy, K. K. R., May 2011. *Physical Characterization of Particulate Matter Employing Support Vector Machine Aided Image Processing*. s.l.:The University of Toledo.
- [27] ZHANG, Z., YANG, J., SU, X. & DING, L., 2013. Analysis of large [article sizes using Machine Vision System. *Physicochemical Problems of Mineral Processing* , 49(2), pp. 397-405.
- [28] Haralick, R. M., Shanmugam, K. & IT's Has Dinstein, 1973. Texture features for image classification. *Systems, Man and Cybernetics. IEEE*, 3(6), pp. 610 - 621.
- [29] L.K.Soh & C.Tsatsoulis, 1999. Texture analysis of sar sea ice imagery using gray level co-occurrence matrices.. *Geoscience and Remote Sensing, IEEE*, 37(2), pp. 780-795.
- [30] David, A. C., 2002. An analysis of co-occurrence texture statistics as a function of grey level quantization.. *Canadian Journal of remote Sensing*, 28(1), pp. 45-62.
- [31] Avinash Uppuluri. GLCM,.[R].[online].[viewed 12/11/2015]. Available from:
<https://www.mathworks.com/matlabcentral/fileexchange/22354-glcmm-features4-m--vectorized-version-of-glcmm-features1-m--with-code-changes->
- [32] Martins E. Irhebhude, Object detection, recognition and re-identification in video footage. PHD thesis. Loughborough University, 20015.
- [33] Mark A Hall. *Correlation-based feature selection for machine learning*. PHD thesis, The University of Waikato, 1999.

- [34] J. C. McCall and M. M. Trivedi, "Video based lane estimation and tracking for driver assistance: Survey, system, and evaluation," *IEEE Trans. on Intelligent Transportation Systems*, pp. 20–37, 2006. 1
- [35] D. A. R. P. A. (DARPA), "Darpa grand challenge," Online source: <http://www.darpa.mil/grandchallenge>. 1
- [36] H. Kong, J.-Y. Audibert, and J. Ponce, "Vanishing point detection for road detection," *CVPR*, 2009. 1, 5, 9
- [37] S.-J. T. Tsung-Ying Sun and V. Chan, "Hsi colour model based lanemarking detection," *IEEE Intelligent Transportation Systems Conference*, pp. 1168–1172, 2006. 1
- [38] K.-Y. Chiu and S.-F. Lin, "Lane detection using colour -based segmentation," *IEEE Intelligent Vehicles Symposium*, 2005. 1
- [39] Y. He, H. Wang, and B. Zhang, "Colour -based road detection in urban traffic scenes," *IEEE Trans. on Intelligent Transportation Systems*, 2004. 1
- [40] J. B. Southhall and C. Taylor, "Stochastic road shape estimation," *ICCV*, pp. 205–212, 2001. 1
- [41] B. Yu and A. K. Jain, "Lane boundary detection using a multiresolution hough transform," *ICIP*, vol. 2, pp. 748–751, 1997. 1
- [42] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *PAMI*, vol. 13, no. 9, pp. 891–906, 1991. 1
- [43] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation," *IEEE Transactions on Intelligent Transportation Systems*, 2006. 1
- [44] A. Kaske, R. Husson, and D. Wolf, "Chi-square fitting of deformable templates for lane boundary detection," *IAR Annual Meeting*, 1995. 1
- [45] C. R. Jung and C. R. Kelber, "A robust linear-parabolic model for lane following," *17th Brazilian Symposium on Computer Graphics and Image Processing*, 2004. 1
- [46] Y. Wang, E. K. Teoh, and D. Shen, "Lane detection and tracking using b-snake," *Image and Vision Computing*, pp. 269–280, 2004. 1
- [47] J. Sparbert, K. Dietmayer, and D. Streller, "Lane detection and street type classification using laser range images," *IEEE Proceedings in Intelligent transportation Systems*, pp. 456–464, 2001. 1

- [48] B. Ma, S. Lakshmanan, and A. O. Hero, "Simultaneous detection of lane and pavement boundaries using model-based multisensor fusion," *IEEE Trans. on Intelligent Transportation Systems*, 2000. 1
- [49] M. Bertozzi and A. Broggi, "Gold: A parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Trans. on Image Processing*, vol. 7, no. 1, pp. 62–81, 1998. 1
- [50] Y. Alon, A. Ferencz, and A. Shashua, "Off-road path following using region classification and geometric projection constraints," *CVPR*, 2006. 2
- [51] A. Lookingbill, J. Rogers, D. Lieb, J. Curry, and S. Thrun, "Reverse optical flow for self-supervised adaptive autonomous robot navigation," *IJCV*, vol. 74, no. 3, pp. 287–302, 2007. 2
- [52] A. Broggi, C. Caraffi, R. I. Fedriga, and P. Grisleri, "Obstacle detection with stereo vision for off-road vehicle navigation," *IEEE International Workshop on Machine Vision for Intelligent Vehicles*, 2005. 2s
- [53] Kong, H., Audibert, J.-Y. & Ponce, J., 2010. General road detection from a single image. *IEEE TRANSACTIONS ON IMAGE PROCESSING*, 19(8), pp. 2211-2220.
- [54] Yang, H.-J., Jang, H. & Jeong, D.-s., 2013. Detection algorithm for road surface condition using wavelet packet transform and SVM. *The 19th Korea-Japan Joint Workshop on Frontiers of Computer Vision*, pp. 323-326.
- [55] Álvarez, J. M. & López, A. M., 2011. Road Detection Based on Illuminant Invariance. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, 12(1), pp. 184-193.
- [56] Haghighat, M., Zonouz, S. & M. A.-M., 2015. CloudID: Trustworthy cloud-based and cross-enterprise biometric identification. *Expert Systems with Applications*, 42(21), pp. 7905-7916.
- [57] Abdallah, H. R. & Faye, B., 2012. Phenotypic classification of Saudi Arabian camel (*Camelus dromedarius*). *ANIMAL SCIENCE*, 24(3), p. 272.
- [58] Afkham, H. M., Targhi, A. T., Eklundh, J.-O. & Pronobis, A., 2008. Joint Visual Vocabulary For Animal Classification. *Pattern Recognition*.
- [59] Ardovalini, A., Cinque, L. & Sangineto, E., 2008. Pattern Recognition. *Identifying elephant photos by multi-curve matching*, 41(6), p. 1867—1877.
- [60] Boykov, Y. & Kolmogorov, V., 2004. In *IEEE Transactions on PAMI. An Experimental Comparison of Min-Cut/Max-Flow Algorithms*, 26(9), pp. 5-8.

- [61] Burghardt, T. & Calic, J., 2006. 8th Seminar on Neural Network Applications in Electrical Engineering. *Real time face detection and tracking of animals.*, pp. 29-30.
- [62] Burghardt, T. & Calic, . J., 2006. In Neural Network Applications in Electrical Engineering, NEUREL.. *Real-time face detection and tracking of animals.*, Volume 115, pp. 27-32.
- [63] Chen, D. et al., 2014. European Conference on Computer Vision. *Joint Cascade Face Detection and Alignment*, pp. 109-122.
- [64] Comaniciu, D., 2002. IEEE Trans. Pattern Anal. Machine Intel. *Mean shift: A robust approach toward feature space analysis.* , Volume 24, pp. 603-619.
- [65] Fang, Y. et al., 2016. *Motion Based Animal Detection in Aerial Videos*. Bhubaneswar, Odisha, India , 2nd International Conference on Intelligent Computing, Communication & Convergence .
- [66] Getreuer, P., 2012. *Chan–Vese Segmentation*, s.l.: Image Processing On Line .
- [67] Hern´andez, A., Reyes, M., Escalera, S. & Radeva, P., 2010. Computer Vision and Pattern Recognition Workshops (CVPRW). *Spatio-Temporal GrabCut Human Segmentation for Face and Pose Recovery*.
- [68] Huang, L.-L., Shimizu, A., Hagihara, Y. & Kobatake, H., 2003. Pattern Recognition. *Gradient feature extraction for classification-based face detection*, 36(11).
- [69] Huynh, T. H., Karademir, . I., Oto, A. & Suzuki, K., 2014. *Computerized Liver Volumetry on MRI by Using 3D Geodesic Active Contour Segmentation*.
- [70] Joules, R. et al., 2016. USING A MULTI-MODALITY AUTOMATED SEGMENTATION METHOD. *QUANTIFICATION OF WHITE MATTER HYPERINTENSITIES:*, 12(7), p. 63.
- [71] Keshani, M., Azimifar, Z., Tajeripour, F. & Boostani, R., 2013. A complete intelligent system. *Lung nodule segmentation and recognition using SVM classifier and active contour modeling*, 43(4).
- [72] Khorrami, P., Wang, J. & Huang, T., 2012. *Multiple Animal Species Detection Using Robust Principal Component Analysis and Large Displacement Optical Flow*.
- [73] Koik, B. T. & Haidi, I., 2012. International Journal of Future Computer and Communication. *A Literature Survey on Animal Detection Methods in Digital Images*, Volume 1, p. 25.

- [74] Kumar, Y. H. S., Manohar, N. & Chethan, . H. K., 2015. *Animal Classification System: A Block Based Approach*. International Conference on Advanced Computing Technologies and Applications (ICACTA-2015). Volume 45, p. 337.
- [75] L'évesque, O. & Robert, B., 2010. Detection and Identification of Animals Using Stereo Vision.
- [76] Lahiri, M. et al., 2011. Biometric Animal Databases from Field Photographs. *Identification of Individual Zebra in the Wild*, Volume 6.
- [77] Li, Q., Zhou, Y. & Yang, . J., 2011. Multimedia Technology (ICMT). *Saliency based Image Segmentation*.
- [78] M.S.Zahrani, Ragab, K. & Haque, A., 2011. *Design of GPS-based system to avoid camel-vehicle collisions: A review*, 4(4), p. 362–377.
- [79] Mabood, F. et al., 2017. Development of new NIR-spectroscopy method combined with multivariate analysis for detection of adulteration in camel milk with goat milk. *Food Chemistry*, Volume 221, p. 746.
- [80] McFarlane, N. & Schofield, C., 1995. Machine Vision and Applications. *Segmentation and tracking of piglets in images*, pp. 188-189.
- [81] Oesau, S., Lafarge, . F. & Alliez, P., 2014. ISPRS Journal of Photogrammetry and Remote Sensing. *Indoor Scene Reconstruction using Feature Sensitive Primitive Extraction and Graph-cut*, Volume 90.
- [82] P., S. & Thenmozhi, G., 2012. International Conference on Emerging Trends in Science, Engineering and Technology. *Detection of Microcalcification in Mammogram Images using Semi-Automated Texture based GrabCut Segmentation* .
- [83] Papageorgiou, C., Oren, M. & Poggio, T., 1998. International Conference on Computer Vision. *A General Framework for Object Detection*.
- [84] PARIKH, M., PATEL, M. & PROF. BHATT, D., 2013. International Journal of Research in Modern. *Animal Detection Using Template Matching Algorithm*, 1(3), p. 26—32.
- [85] Peijiang, C., 2009. International Symposium on Computer Network and Multimedia Technology. *Moving Object Detection Based on Background* , pp. 1-4.

- [86] Ramanan, D., Forsyth, D. A. & Barnard, K., 2006. *Building models of animals from video*. IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE. 28(8), pp. 1319--1334.
- [87] SHARMA, S. U. & SHAH J., D., 2016. *A Practical Animal Detection and Collision Avoidance System Using Computer Vision Technique*. SPECIAL SECTION ON INNOVATIONS IN ELECTRICAL AND COMPUTER ENGINEERING EDUCATION. Volume 5, pp. 350-351.
- [88] Tack, J. L. P. et al., 2016. AnimalFinder: A semi-automated system for animal detection in time-lapse camera trap images. *Ecological Informatics*, Volume 36.
- [89] Taniai, T., Matsushita, Y. & Naemura, T., 2014. *Graph Cut based Continuous Stereo Matching using Locally Shared Labels*. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1613-1620.
- [90] Valsby-Koch, D., 2014. *Recognizing Animals on the Savannah in Aalborg Zoo*. Interactive Computer Vision System. pp. 3-5.
- [91] Vicente, S., Kolmogorov, V. & Rother, C., 2008. *Graph cut based image segmentation with connectivity priors*, s.l.: IEEE Conference on Computer Vision and Pattern Recognition.
- [92] Viola, P. & Jones, M., 2001. *Rapid object detection using a boosted cascade of simple features*. In Computer Vision and Pattern Recognition, CVPR. Volume 1, pp. 5008-511.
- [93] Wilber, M. J. et al., 2013. *Animal Recognition in the Mojave Desert: VisionTools for Field Biologists*. Army Engineering Research and Development Center. pp. 3-5.
- [94] Yu, X. et al., 2013. *Automated identification of animal species in camera trap images*. EURASIP Journal on Image and Video Processing.
- [95] Zeppelzauer, M., 2013. *Automated detection of elephants in wildlife video*. EURASIP Journal on Image and Video Processing. Volume 46, pp. 12-17.
- [96] Zhao, Y. et al., 2015. *Automated Vessel Segmentation Using Infinite Perimeter Active Contour Model with Hybrid Region Information with Application to Retinal Images*. IEEE TRANSACTIONS ON MEDICAL IMAGING. 34(9).

- [97] Zhou, D., 2012. *Infrared Thermal Camera-Based Real-Time Identification and Tracking of Large Animals to Prevent Animal-Vehicle Collisions (AVCs) on Roadways*, Minnesota: University of Minnesota Duluth.
- [98] Zhou, D., 2014. *Real-time Animal Detection System for Intelligent Vehicles*, Ottawa, Canada: University of Ottawa.
- [99] Francis, B., 2014. *K-Means, EM, Gaussian Mixture and Graph Theor*, Lecture3, Paris, France: École Normale Supérieure (ENS) University.
- [100] Bora, D. . J., Gupta, A. K. & Khan, F. A., 2015. Comparing the Performance of L*A*B* and HSV Colour Spaces with Respect to Colour Image Segmentation. *International Journal of Emerging Technology and Advanced Engineering*, 5(2), pp. 192-203.
- [101] Lab Colour Space. [Online]. [Viewed 03/03/2018]. Available from: https://ipfs.io/ipfs/QmXoypizjW3WknFiJnKLwHCnL72vedxjQkDDP1mXWo6uco/wiki/Lab_colour_space.html
- [102] Realtime 3D Computer Graphics / Virtual Reality – WS 2005/2006, [R]. [Online]. [Viewed 03/03/2018]. Available from: [https://www.techfak.uni-bielefeld.de/ags/wbski/lehre/.../8a.RT3DCGVR-colour .pdf](https://www.techfak.uni-bielefeld.de/ags/wbski/lehre/.../8a.RT3DCGVR-colour.pdf)
- [103] The hue-saturation-value (HSV) colour model. [Online]. [Viewed 03/03/2018]. Available from: <http://infohost.nmt.edu/tcc/help/pubs/colour theory/web/hsv.html>.
- [104] RGB to HSV colour conversion. [Online]. [Viewed 03/03/2018]. Available from: <https://www.rapidtables.com/convert/colour /rgb-to-hsv.html>
- [105] Dalal , N. & Triggs, B., 2005. Histograms of oriented gradients for human detection. *In Computer Vision and Pattern Recognition, CVPR 2005. IEEE Computer Society Conference on*, Volume 1, p. 886–893.
- [106] Pietikäinen, M., Hadid, A., Zhao, G. & Ahonen, T., 2011. *Computer Vision Using Local Binary Patterns*. 1 ed. London: Springer-Verlag London.
- [107] Sonka, M., Hlavac, V. & Boyle, R., 2007. *Image Processing, Analysis, and Machine Vision*. Third ed. Delhi: Cengage Learning.

- [108] D. Vázquez, A. M. López, D. Ponsa and D. Geronimo, "Interactive training of human detectors," in *Multimodal Interaction in Image and Video Applications Intelligent Systems Reference Library*, Springer, 2013, pp. 4-16.
- [109] Real Time Robust Embedded Face Detection Using High Level Description. [Online]. [Viewed 10/01/2018]. Available from: <https://www.intechopen.com/books/new-approaches-to-characterization-and-recognition-of-faces/real-time-robust-embedded-face-detection-using-high-level-description>
- [110] Viola-Jones Face Detector. [Online]. [Viewed 10/01/2018]. Available from: <https://www.slideshare.net/zukun/aaai08-tutorial-visual-object-recognition>
- [111] LeCun, Y., Bengio, Y. & Hinton, G., 2015. Deep learning. *Nature, International Journal of Science*, Volume 521, pp. 436 - 444.
- [112] Krizhevsky , A., Sutskever , I. & Hinton , . G. E., 2012. ImageNet Classification with Deep Convolutional Neural Networks. *NIPS'12 Proceedings of the 25th International Conference on Neural Information Processing Systems*, Volume 1, pp. 1097-1105.
- [113] Girshick, R., 2015. *Fast R-CNN*. Santiago, Chile, Computer Science - Computer Vision and Pattern Recognition, 2015 IEEE International Conference on Computer Vision (ICCV).
- [114] Max Pooling in in convolutional neural networks. [Online]. [Viewed 20/01/2018]. Available from: <https://www.quora.com/What-is-max-pooling-in-convolutional-neural-networks>
- [115] Anchor sizes and ratios. [Online]. [Viewed 15/01.2018]. Available from: <https://medium.com/@smallfishbigsea/faster-r-cnn-explained-864d4fb7e3f8>
- [116] Building CNNs with Keras. [Online]. [Viewed 22/06.2018]. Available from: https://www.kernix.com/blog/a-toy-convolutional-neural-network-for-image-classification-with-keras_p14
- [117] N. Otsu , "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62 - 66 , Jan. 1979.
- [118] B.-J. Zhao , "An Ant Colony Clustering Algorithm," in *2007 International Conference on Machine Learning and Cybernetics, IEEE*, Hong Kong, China, 2007.
- [119] L.-K. Huang and M.-J. Wang, "Image thresholding by minimizing the measure of fuzziness." *Pattern Recognition Society, Elsevier Science Ltd*, vol. 28, no. 1, pp. 41-51, 1995.

- [120] H. Abdi and L. J. Williams, "Principal Component Analysis," *John Wiley & Sons, Inc*, vol. 2, pp. 433 - 459, 2010.
- [121] Mark A. Hall, "Correlation-based Feature Selection for Machine Learning". The University of Waikato, Hamilton, NewZealand [Thesis]. [Online]. [Viewed 18/01.2015]. Available from: <https://www.cs.waikato.ac.nz/~mhall/thesis.pdf>
- [122] Niklas Donges, "The Random Forest Algorithm". [Online]. [Viewed 20/01.2015]. Available from: <https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd>
- [123] C. Rother, V. Kolmogorov and A. Blake, "'GrabCut' — Interactive Foreground Extraction using Iterated Graph Cuts," in *ACM Transactions on Graphics (SIGGRAPH)*, Cambridge, UK, 2004.
- [124] M. M. S. J. Preetha, L. P. Suresh and M. J. Bosco, "Image segmentation using seeded region growing," in *2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*, *IEEE*, Kumaracoil, India, 2012.
- [125] ImageNet Organisation. [Online]. [Viewed 22/06.2018]. Available from: <http://www.image-net.org/about-overview>
- [126] J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers² and A.W.M. Smeulders, "Selective Search for Object Recognition," *International Journal of Computer Vision (IJCV)*, Springer, 2012.

Appendix A

Submitted to conference:

- 1- Gharib I. Al Matroushi, Fatma M. Al Jarwani, Eran A. Edirisinghe, Mohammad Athar Ali. “ Camel Detection using Faster RCNN”, in: The British Machine Vision Conference (BMVC), September 2018 - Newcastle, UK.

To be submitted:

- 1- Gharib I. Al Matroushi, Eran A. Edirisinghe. “Road Surface Recognition using Deep Learning ”, In: International Conference on Machine Learning and Applications (ICMLA 2018), December 2018 - Florida, United States.
- 2- Gharib I. Al Matroushi, Eran A. Edirisinghe. “Indestrial Dust Classification using K-means and HSV colour space”, In: International Journal of Computer Vision (IJCV), July 2018 .

Appendix B

A- Sample of Road Surfaces:


























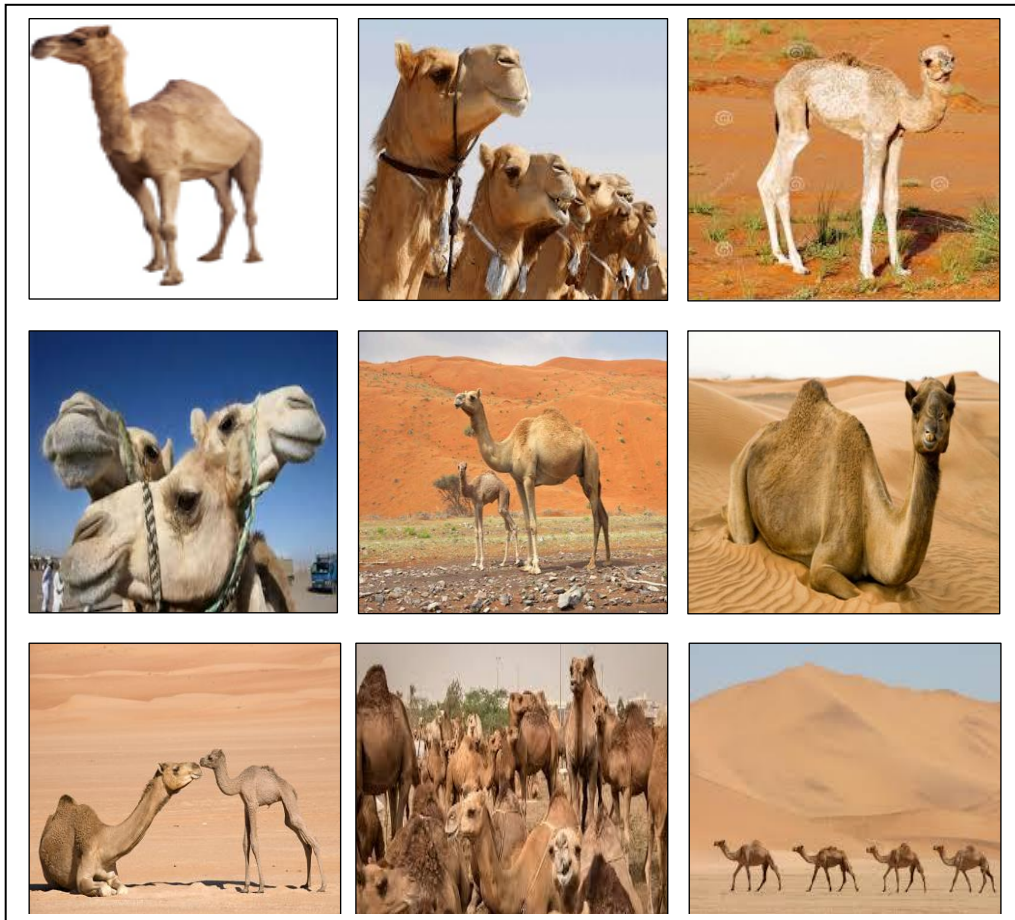
Asphalt	Concrete	Blocks	Rough	Sand
				
				
				
				
				

Figure 26 Sample of Rod Surfaces.

B- Sample of Camel database:

*Figure 96 Example of Camel Database.*