

Road Accident Detection And Prevention Using Machine Learning

Submitted in partial fulfillment of the requirements
for the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

KOTHAMASU ADITYA (37110374)

KOTA REVANTH MANAS (37110371)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119**

MARCH – 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with “A” grade by NAAC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600119

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING



BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **KOTHAMASU ADITYA (37110374)** and **KOTA REVANTH MANAS (37110371)** who carried out the project entitled “**Road Accident Detection And Prevention Using Machine Learning**” under my supervision from November 2020 to April 2021.

Internal Guide

Dr. S. Jancy MCA., M.Tech., Ph.D.

Head of the Department

Dr.S. Vigneshwari M.E., Ph.D.,

Dr. L. Lakshmanan M.E., Ph.D.

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **KOTHAMASU ADITYA** hereby declare that the Project Report entitled “**Road Accident Detection And Prevention Using Machine Learning**” done by me under the guidance of **Dr. S. Jancy MCA., M.Tech., Ph.D.** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in 2017-2021.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D., Dean**, School of Computing **Dr. S. Vigneshwari M.E., Ph.D.** and **Dr. L. Lakshmanan M.E., Ph.D.** , Heads of the Department of Computer Science and Engineering for providing us necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and a deep sense of gratitude to my Project Guide **Dr. S. Jancy MCA., M.Tech., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

The management of educational establishments are concerned about the regular attendance of students. As, attendance has become a key factor while evaluating a student, their overall academic performance alone doesn't quite cut it now. Traditionally, we are used to taking attendance either by calling out student's admission numbers or taking their signatures in a log book. They both were more time expending and troublesome. Subsequently, there is a necessity of PC based understudy participation in the executives' framework which will help the workforce for keeping up participation records consequently. In this undertaking, we have actualized the computerized participation framework utilizing **OPEN CV** package in **Python**. We have anticipated our plans to execute "Automated Attendance System Based on Facial Recognition", in which it assimilates huge applications. The application incorporates face distinguishing proof, which spares time and dispenses with odds of intermediary participation in view of the face approval. Thus, this framework can be actualized in a field where participation assumes a significant job.

TABLE OF CONTENTS

Chapter No. TITLE Page No.

ABSTRACT v LIST OF FIGURES viii LIST OF

ABBREVIATIONS ix

1 INTRODUCTION	1
1.1 IMAGE PROCESSING	1
1.2 DIGITAL IMAGE PROCESSIN	2
1.3 BASICS OF IMAGES	5
2 LITERATURE SURVEY	9
2.1. OBJECT-BASED URBANENVIRONMENT MAPPING	9
2.2. A REMOTE SENSING CLASSIFICATION	13
2.3. ROAD EXTRACTION BY DEEP RESIDUAL U-NET	15
3 SOFTWARE DESIGN	21
3.1. SOFTWARE DESIGN INTRODUCTION	21
3.2 HARDWARE AND SOFTWARE SPECIFICATION	
3.2.1HARDWARE REQUIREMENTS	21
3.2.2 SOFTWARE REQUIREMENTS	22
3.3 TECHNOLOGIES USED	22
3.3.1 MATLAB R2013a	22
3.3.1.1INTRODUCTIONTOMATLAB8.3.0.532(R2014a)	23
3.3.1.2WORKINGOFMATLAB®Session	24
4 DESIGN METHODOLOGY	25
4.1NEURAL NETWORK	29
5 RESULTS	31
5.1 CODING	
5.2 CODING STANDARDS	40
5.2.1NAMING CONVENTIONS	40
5.3 TEST PROCEDURE	41
5.4 TEST DATA AND OUTPUT	
5.4.1 UNIT TESTING	42
5.4.2 FUNCTIONAL TESTS	42

5.4.3 PERFORMANCE TEST	42
5.5 TESTING TECHNIQUES / TESTING STRATEGIES	
5.5.1 TESTING	43
5.5.1.1 WHITE BOX TESTING	44
5.5.1.2 BLACK BOX TESTING	45
5.5.2 SOFTWARE TESTING STRATEGIES	45
5.5.2.2 PROGRAM TESTING	45
5.5.2.3 SECURITY TESTING	46
5.5.2.4 VALIDATION TESTING	46
5.5.2.5 USER ACCEPTANCE TESTING	47
6 EXPERIMENTAL FIGURES	
6.1 MODEL SUMMARY	
LIMITATIONS	
REFERENCES	43
APPENDICES	
A. PUBLICATION AND PLAGARISM REPORT	

LIST OF FIGURES

Figure No.	Figure Name	Page No.
------------	-------------	----------

1.1	DIGITAL SIGNAL IMAGE PROCESSING	3
2.1	BINARY IMAGE	6
2.2	GRAYSCALE IMAGE	7
3.1	BLOCK DIAGRAM	27

LIST OF ABBREVIATIONS

ABBREVIATIONS	EXPANSION
CNN	Convolutional Neutral Network
ILSVRC	ImageNet Large Scale Visual Review Challenge
SIANN	Space Invariant Counterfeit Neural Systems
VGG	Visual Geometry Group
SVM	Supervised Machine Learning
LDA	Linear Discriminant Analysis

CHAPTER 1

INTRODUCTION

IMAGE PROCESSING

Image Processing is a technique to enhance raw images received from cameras/sensors placed on satellites, space probes and aircrafts or pictures taken in normal day-today life for various applications. Various techniques have been developed in Image Processing during the last four to five decades. Most of the techniques are developed for enhancing images obtained from unmanned spacecrafts, space probes and military reconnaissance flights. Image Processing systems are becoming popular due to easy availability of powerful personnel computers, large size memory devices, graphics softwares .

Image Processing is used in various applications :

- Remote Sensing
- Medical Imaging
- Non-destructive Evaluation
- Forensic Studies
- Textiles
- Material Science.
- Military
- Film industry
- Document processing
- Graphic arts
- Printing Industry

Image processing :

- Image processing
- Digital image processing

- Digital imaging
- Medical imaging

Image processing :

- In electrical engineering and computer science, image processing is any form of signal processing for which the input is an image, such as photographs or frames of video; the output of image processing can be either an image or a set of characteristics or parameters related to the image.
- Image processing usually refers to digital image processing, but optical and analog image processing are also possible .

Digital image processing :

- Digital image processing is the use of computer algorithms to perform image processing on digital images. As a subfield of digital signal processing, digital image processing has many advantages over analog image processing; it allows a much wider range of algorithms to be applied to the input data, and can avoid problems such as the build-up of noise and signal distortion during processing .

Image Processing Fundamentals :

- Teach the fundamental image processing tools available in machine vision software
- Provide some background into how the algorithms work
- Accelerate your machine vision learning curve

Image Processing for Machine Vision :

- Objective – To extract useful information present in an image, in a limited time

- Secondary – To display an image for users

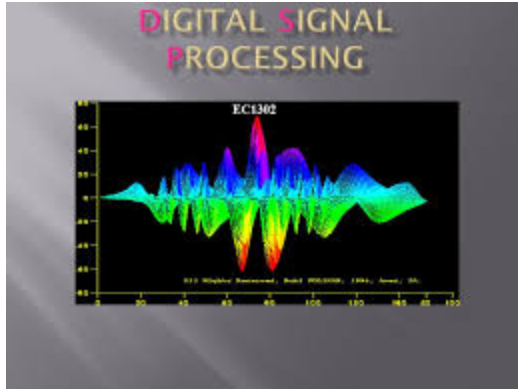
- Not – Improve appearance of image in general
- Used for – Image pre-processing
- Minimize variations of information in the image
- Prepare the image for processing and measurement – Application specific processing
- Use image to count, locate, and measure attributes

Image Types :

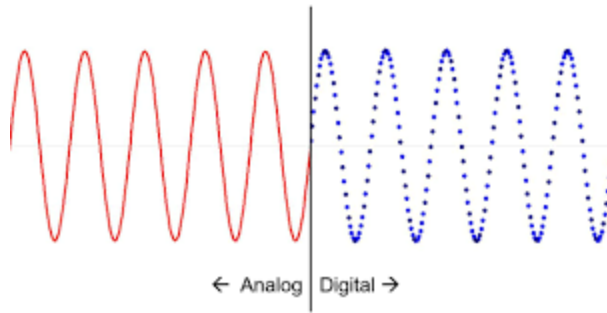
- **Grayscale** – 8 bit: pixel values range from 0 to 255 – 12 bit: pixel values range from 0 to 4095 – 16 bit: pixel values range from 0 to 65535
- **Color** – Composed of 3 grayscale images (RGB)
- **Other types** – Binary: pixel values: 0 and 1
- Commonly used to identify objects of interest in an image
 - Usually the result of image processing step – Floating point: pixel values range from – X.xx to X.xx •

Usually a result of a computation .





DIGITAL SIGNAL IMAGE PROCESSING

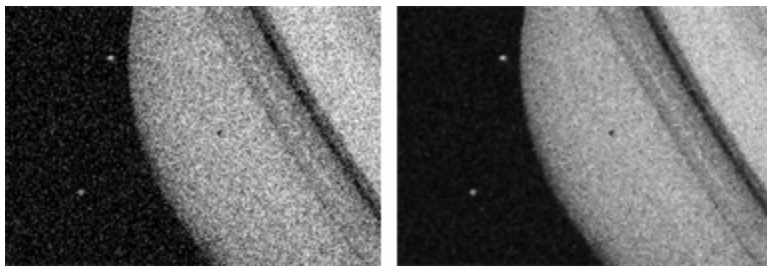


FUNDAMENTAL OF DIGITAL SIGNAL

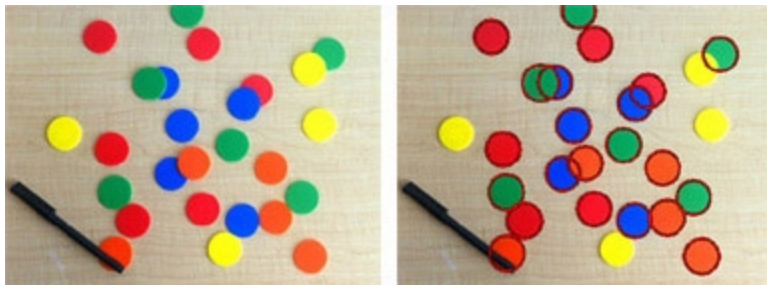
PROCESSING

Process digital images with computer algorithms :

- Digital image processing is the use of computer algorithms to create, process, communicate, and display digital images. Digital image processing algorithms .

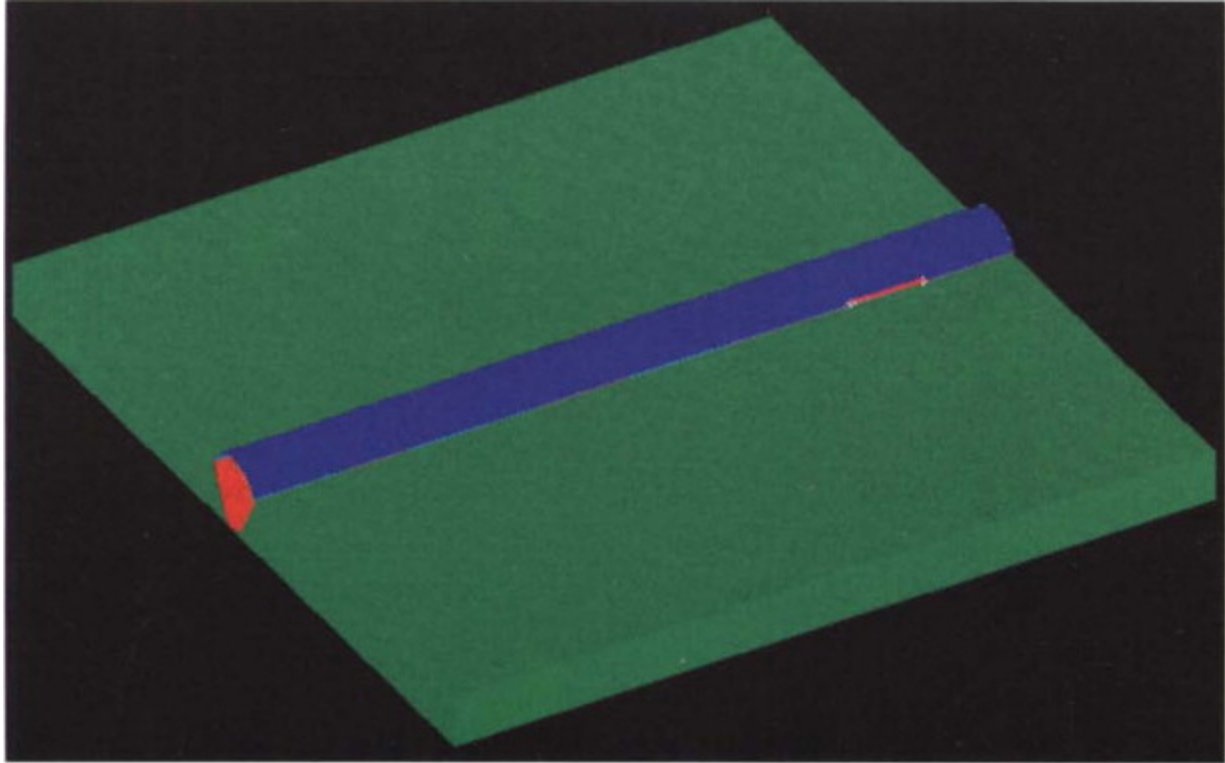


Removing noise using a Wiener Filter.



OBJECT IMAGE

COUNTING CIRCULAR



AUTOCAD VISUALISATION

OpenGL Imaging Support :

- Image processing is an important component of applications used in the publishing, satellite imagery analysis, medical, and seismic imaging fields. Given its importance, image processing functionality has been added to OpenGL in a number of ways.
- A bundle of extensions targeted toward accelerating common image processing operations, referred to as the ARB imaging subset, is defined as part of the OpenGL 1.2 and later specifications .

Basic Types of Images :

There are a total of four basic types of images,

- Binary

- Grayscale
- True color or red-green-blue (RGB)
- Indexed

Binary image :

- A binary image is a digital image that has only two possible values for each pixel.
- The pixel is made up of either in black or white color.
- Binary images are also called bi-level or two levels



BINARY IMAGE

Grayscale image :

- A grayscale digital image is an image in which the value of each pixel is not the same. The only colors are shades of gray .
- Each image pixel is made up of shades of gray from 0 (black) to 255 (white). Each pixel can be represented by 1 byte or 8 bits.



GRayscale IMAGE

True color or RGB image :

- Each pixel will have a particular color that is being described by the amount of red, green, and blue in it. If each of the components has a range of 0-255, this will give a total of 2^{24} different possible colors in an image.
- Pixel will require 24 bits and they are called 24-bit color images.



RGB IMAGE

Image Compression Using MATLAB :

- Image processing with MATLAB is somewhat complex, so we will confine ourselves to simple operations with gray scale images.

```
>> A = imread('filename.ext');
```

```
>> imagesc(A);
```

```
>> colormap(gray);
```

Edge Detection:

- The first parallel implementation of an ant colony optimization (ACO)-based edge detection algorithm on the GPU. The proposed implementation is able to match the quality of edge maps produced by the sequential implementation and executed up to 150× faster.
- This approach yields the best result and speedups of around 150× against an optimized sequential counterpart.

Registration :

- Image registration is the process of transforming different sets of data into one coordinate system. Data may be multiple photographs, data from different sensors, times, depths, or viewpoints .
- Image registration is often used in medical and satellite imagery to align images from different camera sources.



Registering Aerial Photos Using Point Mapping

- A high-performance PSO powered by the GPU for ground control point-based nonlinear registration of airborne push-broom imagery was introduced by Reguera-Salgado and Martin-Herrero.

Watermarking :

- Watermarking is presented when a pattern is inserted in an image, video or audio file, it helps to copyright the information in the files .
- The objective of optimization is to find the best frequency band set to insert the watermark within the image. Different frequency bands are tested through the iterations of the algorithm finding out the best solution.

CHAPTER 2

LITERATURE SURVEY

LITERATURE SURVEY

Project Title: OBJECT-BASED URBAN ENVIRONMENT MAPPING WITH HIGH SPATIAL RESOLUTION IKONOS IMAGERY

Author Name: Ruiliang Pu

Year of Publishing: 2009

Abstract:

Advances in remote sensing such as increasing spatial/spectral resolutions have strengthened its ability of urban environmental analysis. Unfortunately, high spatial resolution imagery also increases internal variability in landcover use unit, which can cause consequent classification result showing a “salt and pepper” effect. To overcome this problem, region-based classification has been used. In such a classification,

image-object (IO) is used rather than pixel as a classification unit. Using IKONOS high spatial resolution imagery, in this study, we propose to test whether the IO technique can

significantly improve classification accuracy when applied to urban environmental mapping with high spatial resolution imagery compared to pixel-based method in Tampa Bay, FL, USA. We further evaluate the performance of artificial neural network (ANN) and Maximum Likelihood Classifier (MLC) in urban environmental classification with high resolution data and test the effect of number of extracted IO features on urban classification accuracy. Experimental results indicate that, in this particular study, a statistically significant difference of classification accuracy is proved between using pixel-based and IO-based data; ANN outperforms MLC when both using 9 features pixel-based data; and using more features (30 vs. 9 features) can increase IO classification accuracy, but seems not statistically significant at the 0.9 confidence level at this study.

Project Title: A Quantitative Method to Test for Consistency and Correctness
in Photointerpretation

Author Name: RUSSELGL . CONGALTON

Abstract:

method has been developed to quantitatively test the degree of similarity between photointerpreters and/or photointerpretation variables, such as film/filter type, season, and scale. This method involves giving each photointerpreter the same set of photos to interpret, and then a similarity matrix is generated for each interpreter by comparing the interpretation results to the actual ground cover. It is assumed that the interpretation has been sampled in such a way as to assure that the resulting similarity matrix is representative of that interpretation. The similarity matrix is analyzed using a computer program called KAPPA, which implements a discrete multivariate analysis technique to determine if there is a significant difference between matrices. This technique allows the comparison of individual interpreters, a test for photointerpreter consistency, and the comparison of photointerpretation variables

Project Title: ASSESSING THE BENEFITS AND COSTS OF THE URBAN FOREST

Author Name: John F. Dwyer, E. Gregory McPherson

Year of Publishing: 1992

Abstract:

With effective planning and management, urban trees and forests will provide a wide range of important benefits to urbanites. These include a more pleasant, healthful, and comfortable environment to live, work, and play in, savings in the costs of providing a wide range of urban services, and substantial improvements in individual and community wellbeing. Urban forestry plans should begin with consideration of the contribution that trees and forests can make to people's needs. Planning and management efforts should focus on how the forest can best meet those needs. Past planning and management efforts have not been as effective as they might have been because planners and managers have underestimated the potential benefits that urban trees and forests can provide, and have not understood the planning and management efforts needed to provide those benefits, particularly the linkages between benefits and characteristics of the urban forest and its management.

Project Title: Exploitation of Very High Resolution Satellite Data for Tree Species Identification

Author Name: A. Carleer and E. Wolff

Year of Publishing: 2004

Abstract:

With the emergence of very high spatial resolution satellite images, the spatial resolution gap which existed between satellite images and aerial photographs has decreased. A study of the potential of these images for tree species in “monoculture stands” identification was conducted. Two Ikonos images were acquired, one in June 2000 and the other in October 2000, for an 11- by 11-km area covering the Sonian Forest in the southeastern part of the Brussels-Capital region (Belgium). The two images were orthorectified using a digital elevation model and 1256 geodetic control points. The identification of the tree species was carried out utilizing a supervised maximum-likelihood classification on a pixel-by-pixel basis. Classifications were

performed on the orthorectified data, NDVI transformed data, and principal components imagery. In order to decrease the intraclass variance, a mean filter was applied to all the

spectral bands and neo-channels used in the classification process. Training and validation areas were selected and digitized using detailed geographical databases of the tree species. The selection of the relevant bands and neo-channels was carried out by successive addition of information in order to improve the classification results. Seven different tree species of one to two different age classes were identified with an overall accuracy of 86 percent. The seven identified tree species or species groups are Oaks (*Quercus* sp.), Beech (*Fagus sylvatica* L.), Purple Beech (*Fagus sylvatica purpurea*), Douglas Fir (*Pseudotsuga menziesii* (Mirb.) Franco), Scots Pine (*Pinus sylvestris* L.), Corsican Pine (*Pinus nigra* Arn. subsp. *laricio* (Poir.) Maire var. *corsican*), and Larch (*Larix decidua* Mill.).

Project Title: Planimetric accuracy of Ikonos 1-m panchromatic image products and their utility for local government GIS basemap applications

Abstract:

A detailed assessment of the planimetric accuracy of high-resolution (1 m) panchromatic Ikonos orthoimage products for three different test sites located in the State of Missouri is presented. The main objective of this study was to evaluate the potential of Ikonos orthoimage products for use as digital image basemaps in local government GIS systems. For maximum utility and adoption in a wide variety of local government planning and management applications, a planimetric accuracy of 3-4 m CE90 is considered nominal. Evaluation of the planimetric accuracies of Ikonos orthoimages acquired under a Space Imaging/NASA Databuy agreement were found to be 1.1 m and 1.7 m CE90 at two different test sites. The terms of the NASA Databuy agreement require a 2 m CE90 planimetric accuracy. Thus, our independent assessment confirms that the Space Imaging products met this NASA Databuy specification. Low precision, low cost georeferenced Ikonos image products (Carterra Geo, 50 m CE90) were orthorectified using third party commercially available software and various custom NAPP DEMs. The planimetric accuracies of the resulting Ikonos orthoimages were found to vary between 2-4 m CE90.

In addition, USGS DEMs were also used to orthorectify georeferenced Ikonos image products. The resulting Ikonos orthoimages were found to have planimetric accuracies

from 2-7 m CE90. Planimetric accuracies of 2-3 m CE90 can be obtained from georeferenced Ikonos using USGS DEMs with RMS vertical accuracies on the order of 2 m. The approach demonstrated here can be used to deliver up-to-date, cost effective orthoimages from Ikonos Carterra Geo products that yield planimetric accuracies suitable for use as digital image basemaps by local governments.

Project Title: A REMOTE SENSING CLASSIFICATION FOR LAND-COVER
CHANGES AND MICRO-CLIMATE IN KUWAIT

Author Name: UDDIN, A.N. AL GHADBAN, A. AL DOUSARI, M.

Year of Publishing: 2010

Abstract:

The paper addresses the issue of micro-climatic variations due to changing land-cover in the study area. Kuwait has undergone a unique environmental catastrophe during 1991 Gulf war in the form of oil lakes and spills on land, which has left permanent scars on the desert surface. The region is experiencing large scale urban expansion with the growing population needs and political stability. New townships planned with expansion of existing ones, rapid development in petroleum sector, all these developments are resulting in Urban Heat Island (UHI) effect both over the city and the hydrocarbon contaminated sites. In this study the Landsat image of 1989, 1991 and 2000 are used to compute surface temperature and land-cover classification to understand the relationship between micro-climate and land-cover. The results have shown increased temperatures over both the built-up areas and hydrocarbon contaminated surface. The long term temperature trend in the area shows an incremental trend since 1980s, which increases sharply in post 1990s, probably due to cumulative effect of hydrocarbon pollution, urban-industrial development and increased carbon emission. These indices are related to the Land Surface Temperature (LST) computed using satellite datasets. Results show a negative

correlation of NDVI, NDWI, and NDBare with surface temperature. NDBI is positively correlated with temperature. Although there are isolated instances where bare

surfaces show increased temperature at places and these temperatures are comparable to built-up area, the most pertinent reason for this temperature hike is obliteration of emissive property of desert soil due to hydrocarbon leftovers and mixing of hydrocarbon leftovers with soil for bioremediation. These remediated soils are used in plantation and re-vegetation giving higher thermal signatures, mainly because of scanty vegetation cover.

Project title: Very Deep Convolutional Networks For Large-Scale Image Recognition

Author Name: Karen Simonyan, Andrew Zisserman

Year Of Publish: 2015

Abstract:

In this work we investigate the effect of the convolutional network depth on its accuracy in the large-scale image recognition setting. Our main contribution is a thorough evaluation of networks of increasing depth using an architecture with very small (3×3) convolution filters, which shows that a significant improvement on the prior-art configurations can be achieved by pushing the depth to 16–19 weight layers. These findings were the basis of our ImageNet Challenge 2014 submission, where our team secured the first and the second places in the localisation and classification tracks respectively. We also show that our representations generalise well to other datasets, where they achieve state-of-the-art results. We have made our two best-performing ConvNet models publicly available to facilitate further research on the use of deep visual representations in computer vision.

Project title: SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation

Author Name: Vijay Badrinarayanan, Alex Kendall

Year Of Publish: 2016

Abstract:

We present a novel and practical deep fully convolutional neural network architecture for semantic pixel-wise segmentation termed SegNet. This core trainable segmentation engine

consists of an encoder network, a corresponding decoder network followed by a pixel-wise classification layer. The architecture of the encoder network is topologically identical to the 13 convolutional layers in the VGG16 network [1]. The role of the decoder network is to map the

low resolution encoder feature maps to full input resolution feature maps for pixel-wise classification. The novelty of SegNet lies in the manner in which the decoder upsamples its lower resolution input feature map(s). Specifically, the decoder uses pooling indices computed in the max-pooling step of the corresponding encoder to perform non-linear upsampling. This eliminates the need for learning to upsample. The upsampled maps are sparse and are then convolved with trainable filters to produce dense feature maps. We compare our proposed architecture with the widely adopted FCN [2] and also with the well known DeepLab-LargeFOV [3], DeconvNet [4] architectures. This comparison reveals the memory versus accuracy trade-off involved in achieving good segmentation performance. SegNet was primarily motivated by scene understanding applications. Hence, it is designed to be efficient both in terms of memory and computational time during inference. It is also significantly smaller in the number of trainable parameters than other competing architectures and can be trained end-to-end using stochastic gradient descent. We also performed a controlled benchmark of SegNet and other architectures on both road scenes and SUN RGB-D indoor scene segmentation tasks. These quantitative assessments show that SegNet provides good performance with competitive inference time and most efficient inference memory-wise as compared to other architectures.

Project title: Road Extraction by Deep Residual U-Net

Author Name: Zhengxin Zhangy, Qingjie Liuy

Year Of Publish: 2017

Abstract:

Road extraction from aerial images has been a hot research topic in the field of remote sensing image analysis. In this letter, a semantic segmentation neural network which combines the strengths of residual learning and U-Net is proposed for road area extraction. The network is built with residual units and has similar architecture to that of U-Net. The benefits of this model is two-fold: first, residual units ease training of deep networks. Second, the rich skip connections within the network could facilitate information propagation, allowing us to design networks with fewer parameters however better performance. We test our network on a public road dataset and compare it with U-Net and other two state of the art deep learning based road extraction

methods. The proposed approach outperforms all the comparing methods, which demonstrates its superiority over recently developed state of the arts.

Project title: Fully Convolutional Network for Automatic Road Extraction from Satellite Imagery

Author Name: Alexander Buslaev, Selim Seferbekov

Year Of Publish: 2017

Abstract:

Analysis of high-resolution satellite images has been an important research topic for traffic management, city planning, and road monitoring. One of the problems here is automatic and precise road extraction. From an original image, it is difficult and computationally expensive to extract roads due to presences of other road-like features with straight edges. In this paper, we propose an approach for automatic road extraction based on a fully convolutional neural network of U-net family. This network consists of ResNet-34 pre-trained on ImageNet and decoder adapted from vanilla U-Net. Based on validation results, leaderboard and our own experience this network shows superior results for the DEEPGLOBE - CVPR 2018 road extraction sub-challenge. Moreover, this network uses moderate memory that allows using just one GTX 1080 or 1080ti video cards to perform whole training and makes pretty fast predictions.

Project title: HD Maps: Fine-grained Road Segmentation by Parsing Ground and Aerial Images

Author Name: Gell'ert M'attyus, Shenlong Wang

Year Of Publish: 2015

Abstract:

In this paper we present an approach to enhance existing maps with fine grained segmentation categories such as parking spots and sidewalk, as well as the number and location of road lanes. Towards this goal, we propose an efficient approach that is able to estimate these fine grained categories by doing joint inference over both, monocular aerial imagery, as well as ground images taken from a stereo camera pair mounted on top of a car. Important to this is reasoning about the alignment between the two types of imagery, as even when the measurements are taken

with sophisticated GPS+IMU systems, this alignment is not sufficiently accurate. We demonstrate the effectiveness of our approach on a new dataset which enhances KITTI [8] with

aerial images taken with a camera mounted on an airplane and flying around the city of Karlsruhe, Germany.

Project title: Road Centerline Extraction via Semi-supervised Segmentation and Multidirection Non-maximum Suppression

Author Name: Guangliang Cheng, Feiyun Zhu

Year Of Publish: 2016

Abstract:

Accurate road centerline extraction from remotely sensed images plays a significant role in road map generation and updating. In the road extraction problem, acquisition of labeled data is time-consuming and costly, thus there are only a small amount of labeled samples in reality. In the existing centerline extraction algorithms, the thinning based algorithms always produce small spurs that reduce the smoothness and accuracy of the road centerline; the regression based algorithms can extract smooth road network, while they are time-consuming. To solve the above problems, we propose a novel road centerline extraction method, which is constructed based on semi-supervised segmentation and multiscale filtering (MF) & multidirection nonmaximum suppression (M-NMS). Specifically, a semi-supervised method, which explores the intrinsic structures between the labeled samples and the unlabeled ones, is introduced to obtain the segmentation result. Then, a novel MF&M-NMS based algorithm is proposed to gain smooth and complete road centerline network. Experimental results on a public dataset demonstrate that the proposed method achieves comparable or better performances by comparing with state-of-the-art methods. In addition, our method is nearly 10 times faster than state-of-the-art methods.

Project title: Road Recognition From Remote Sensing Imagery Using Machine Learning

Author Name: P. Deepan, S. Abinaya

Year Of Publish: 2018

Abstract:

The advance of remote sensing technology has been capable of providing abundant spatial and contextual information for object or area detection in satellite or aerial imagery, which facilitates subsequent automatic analysis and interpretation of the optical remotes sensing images (RSIs). Most existing object or area detection approaches suffer from two limitations. First, the feature

representation used is insufficiently powerful to capture the spatial and structural patterns of objects and the background regions. Second, a large number of training data with manual annotation of object bounding boxes is required in the supervised learning techniques adopted while the annotation process is generally too expensive and sometimes even unreliable. We propose an end-to-end framework for the dense, pixel-wise classification of satellite imagery with convolution neural networks (CNNs). In our framework, CNNs are directly trained to produce classification maps out of the input mages. We first devise a fully convolution architecture and demonstrate its relevance to the dense classification problem. We then address the issue of imperfect training data through a two-step training approach: CNNs are first initialized by using a large amount of possibly inaccurate reference data, and then refined on a small amount of accurately labeled data. To complete our framework, we design this module with the two different stages that alleviates the common trade-off between recognition and precise localization. A series of experiments in MATLAB show that our works consider a large amount of Meta context to provide fine-grained classification maps.

Project title: Learning Aerial Image Segmentation from Online Maps

Author Name: Pascal Kaiser, Jan Dirk Wegner

Year Of Publish: 2016

Abstract:

This study deals with semantic segmentation of high-resolution (aerial) images where a semantic class label is assigned to each pixel via supervised classification as a basis for automatic map generation. Recently, deep convolutional neural networks (CNNs) have shown impressive performance and have quickly become the de-facto standard for semantic segmentation, with the added benefit that task-specific feature design is no longer necessary. However, a major downside of deep learning methods is that they are extremely data-hungry, thus aggravating the perennial bottleneck of supervised classification, to obtain enough annotated training data. On the other hand, it has been observed that they are rather robust against noise in the training labels. This opens up the intriguing possibility to avoid annotating huge amounts of training data, and instead train the classifier from existing legacy data or crowd-sourced maps which can

exhibit high levels of noise. The question addressed in this paper is: can training with large-scale, publicly available labels replace a substantial part of the manual labeling effort and still achieve

sufficient performance? Such data will inevitably contain a significant portion of errors, but in return virtually unlimited quantities of it are available in larger parts of the world. We adapt a state-of-the-art CNN architecture for semantic segmentation of buildings and roads in aerial images, and compare its performance when using different training data sets, ranging from manually labeled, pixel-accurate ground truth of the same city to automatic training data derived from OpenStreetMap data from distant locations. We report our results that indicate that satisfying performance can be obtained with significantly less manual annotation effort, by exploiting noisy large-scale training data.

Project title: Convolutional Neural Networks for Large-Scale Remote Sensing Image Classification

Author Name: Emmanuel Maggiori, Yuliya Tarabalka

Year Of Publish: 2017

Abstract:

We propose an end-to-end framework for the dense, pixelwise classification of satellite imagery with convolutional neural networks (CNNs). In our framework, CNNs are directly trained to produce classification maps out of the input images. We first devise a fully convolutional architecture and demonstrate its relevance to the dense classification problem. We then address the issue of imperfect training data through a two-step training approach: CNNs are first initialized by using a large amount of possibly inaccurate reference data, then refined on a small amount of accurately labeled data. To complete our framework we design a multi-scale neuron module that alleviates the common trade-off between recognition and precise localization. A series of experiments show that our networks take into account a large amount of context to provide fine-grained classification maps.

Project title: Road Extraction from High-Resolution Remote Sensing Imagery Using Deep Learning

Author Name: Yongyang Xu , Zhong Xie

Year Of Publish: 2018

Abstract:

The road network plays an important role in the modern traffic system; as development occurs, the road structure changes frequently. Owing to the advancements in the field of high-resolution remote sensing, and the success of semantic segmentation success using deep learning in computer vision, extracting the road network from high-resolution remote sensing imagery is becoming increasingly popular, and has become a new tool to update the geospatial database. Considering that the training dataset of the deep convolutional neural network will be clipped to a fixed size, which lead to the roads run through each sample, and that different kinds of road types have different widths, this work provides a segmentation model that was designed based on densely connected convolutional networks (DenseNet) and introduces the local and global attention units. The aim of this work is to propose a novel road extraction method that can efficiently extract the road network from remote sensing imagery with local and global information. A dataset from Google Earth was used to validate the method, and experiments showed that the proposed deep convolutional neural network can extract the road network accurately and effectively. This method also achieves a harmonic mean of precision and recall higher than other machine learning and deep learning methods.

Project title : Road Extraction from High-Resolution Remote Sensing Imagery Using Deep Learning

Author Name: Yongyang Xu, Zhong Xie

Year Of Publish: 2018

Abstract:

The road network plays an important role in the modern traffic system; as development occurs, the road structure changes frequently. Owing to the advancements in the field of high-resolution

remote sensing, and the success of semantic segmentation success using deep learning in computer vision, extracting the road network from high-resolution remote sensing imagery is

becoming increasingly popular, and has become a new tool to update the geospatial database. Considering that the training dataset of the deep convolutional neural network will be clipped to a fixed size, which lead to the roads run through each sample, and that different kinds of road types have different widths, this work provides a segmentation model that was designed based on densely connected convolutional networks (DenseNet) and introduces the local and global attention units. The aim of this work is to propose a novel road extraction method that can efficiently extract the road network from remote sensing imagery with local and global information. A dataset from Google Earth was used to validate the method, and experiments showed that the proposed deep convolutional neural network can extract the road network accurately and effectively. This method also achieves a harmonic mean of precision and recall higher than other machine learning and deep learning methods.

CHAPTER 3
SYSTEM DESIGN
SOFTWARE DESIGN : MATLAB 2018A

3.1 INTRODUCTION

The requirements specification is a technical specification of requirements for the software products. It is the first step in the requirements analysis process it lists the requirements of a particular software system including functional, performance and security requirements. The requirements also provide usage scenarios from a user, an operational and an administrative perspective. The purpose of software requirements specification is to provide a detailed overview of the software project, its parameters and goals. This describes the project target audience and its user interface, hardware and software requirements. It defines how the client, team and audience see the project and its functionality.

3.2 HARDWARE AND SOFTWARE SPECIFICATION

3.2.1 HARDWARE REQUIREMENTS

Hard Disk : 40GB and Above
RAM : 512MB and Above

Processor : Pentium III and Above

3.2.2 SOFTWARE REQUIREMENTS

MATLAB R2013a

Windows Operating System XP and Above

3.3 TECHNOLOGIES USED

3.3.1 MATLAB R2013a

The MATLAB ® high-performance language for technical computing integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include

- Math and computation
- Algorithm development
- Data acquisition
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization Scientific and engineering graphics
- Application development, including graphical user interface building

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. It allows you to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar noninteractive language such as C or Fortran.

The name MATLAB stands for matrix laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK

projects. Today, MATLAB engines incorporate the LAPACK and BLAS libraries, embedding the state of the art in software for matrix computation.

MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. In industry, MATLAB is the tool of choice for high-productivity research, development, and analysis.

MATLAB features a family of add-on application-specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow you to learn and apply specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. You can add on toolboxes for signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many other areas.

3.3.1.1 INTRODUCTION TO MATLAB 8.3.0.532 (R2014a)

The MATLAB® System

The MATLAB system consists of these main parts:

Desktop Tools and Development Environment

This is the set of tools and facilities that help you use and become more productive with MATLAB functions and files. Many of these tools are graphical user interfaces. It includes the MATLAB desktop and Command Window, a command history, an editor and debugger, a code analyzer and other reports, and browsers for viewing help, the workspace, files, and the search path.

Mathematical Function Library

This is a vast collection of computational algorithms ranging from elementary functions, like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrix inverse, matrix eigenvalues, Bessel functions, and fast Fourier transforms.

The Language

This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both

"programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create large and complex application programs.

Graphics

MATLAB has extensive facilities for displaying vectors and matrices as graphs, as well as annotating and printing these graphs. It includes high-level functions for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level functions that allow you to fully customize the appearance of graphics as well as to build complete graphical user interfaces on your MATLAB applications.

External Interfaces

This is a library that allows you to write C and Fortran programs that interact with MATLAB. It includes facilities for calling routines from MATLAB (dynamic linking), for calling MATLAB as a computational engine, and for reading and writing MAT-files.

3.3.1.2 WORKING OF MATLAB® Session

Starting a MATLAB® Session

On Microsoft® Windows® platforms, start the MATLAB® program by double-clicking the MATLAB R2014a shortcut on your Windows desktop.

On Apple® Macintosh® platforms, start MATLAB by double-clicking the MATLAB 7.6 icon on your Macintosh desktop.

On The Open Group UNIX® platforms, start MATLAB by typing matlab at the operating system prompt.

There are alternative ways to start MATLAB on Windows and Macintosh platforms, and you can customize MATLAB startup. For example, you can change the directory in which MATLAB starts or automatically execute MATLAB statements upon startup.

For More Information See Startup and Shutdown in the Desktop Tools and Development Environment documentation.

The Desktop

When you start MATLAB, the desktop appears, containing tools (graphical user interfaces) for managing files, variables, and applications associated with MATLAB.

The following illustration shows the default desktop. You can customize the arrangement of tools and documents to suit your needs. For more information about the desktop tools, see Desktop Tools and Development Environment.

CHAPTER 4

DESIGN METHODOLOGY

Abstract:

Road accident plays an important role in many applications, such as intelligent transportation system and urban planning. Currently high resolution visible remote sensing images are widely used due to the fast development of remote sensing technology. These high resolution images present a new challenge for road accident methods, because high resolution images have more details than conventional images and, thus, contain multi scale roads and complex background. The FCN has a few false accidents on complex background and segments most of the wide roads, but it misses lots of the narrow roads and fails to segment the accurate boundaries of the roads. The poor performance of

FCN on narrow roads is due to its lack of road details during the down sampling to up sampling operation. The performance of U-Net lies between

VGG and FCN; it fails to segment the accurate boundaries of the roads and has lots of false accidents on complex background. In this proposed system the method of image contrast enhancement on the basis of the contrast distribution at the boundaries of objects and background on the image is proposed. Besides, deep learning methods especially regression neural network have also been widely used for road recognition and accident.

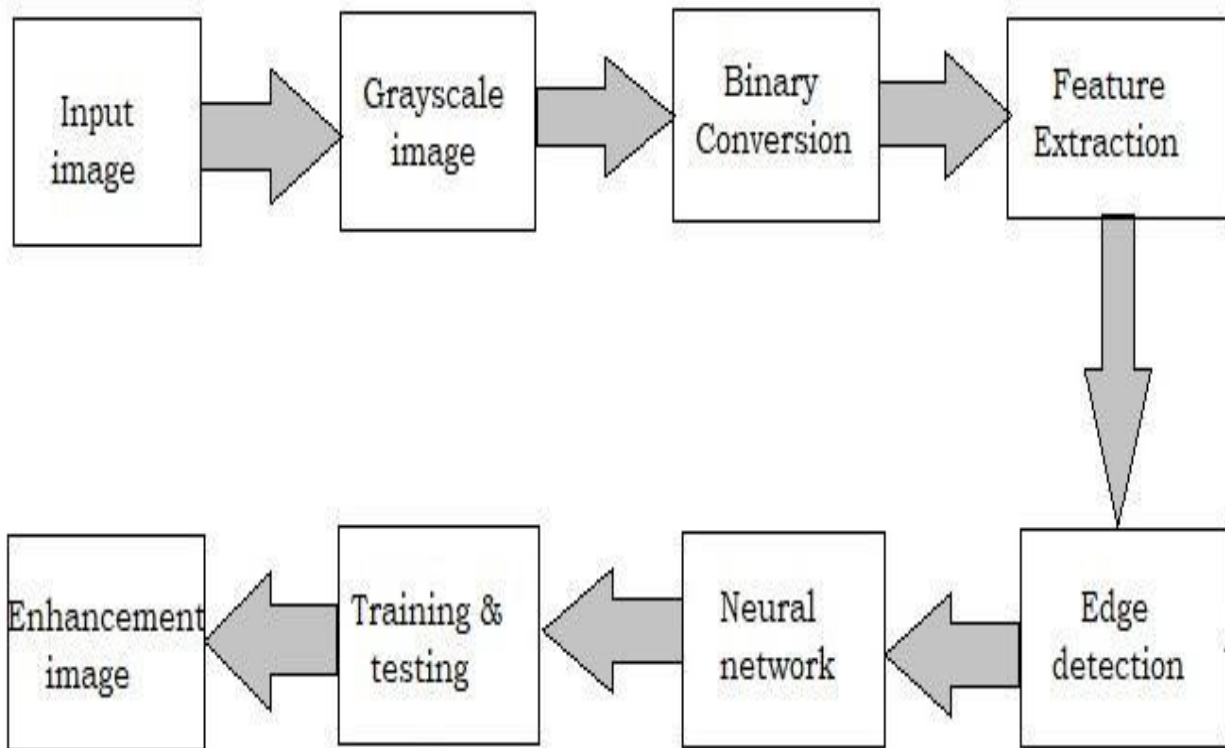
Existing System:

A hybrid convolution network (HCN) to segment roads from high-resolution visible remote sensing images, with the hope to achieve high accuracy for multi scale roads with complex background. The FCN has a few false accidents on complex background and segments most of the wide roads, but it misses lots of the narrow roads and fails to segment the accurate boundaries of the roads. The poor performance of FCN on narrow roads is due to its lack of road details during the down sampling to up sampling operation.

Proposed System:

In this proposed system the method of image contrast enhancement on the basis of the contrast distribution at the boundaries of objects and background on the image is proposed. Besides, deep learning methods especially regression neural network have also been widely used for road recognition and accident.

Block diagram:



Module Description:

Module 1: Preprocessing

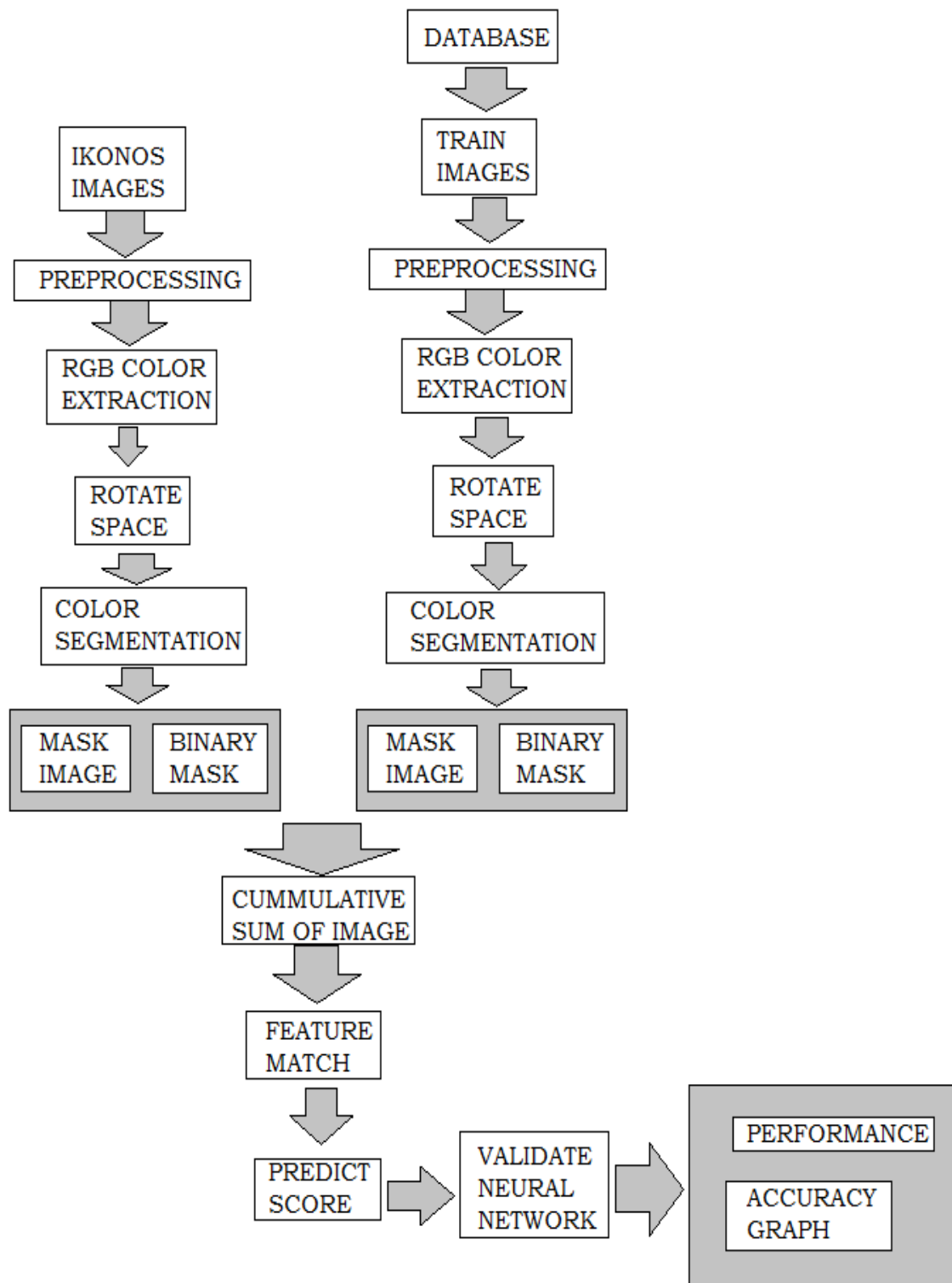
If the input images are color images means then we are convert to gray scale from the color images. Then it can be converted into binary conversion (0 means black and 1 means white).The median filter is applied in order to remove the noise from the image.

Module 2: Edge detection

Edge detection is a technique for finding the boundaries of objects within images. Edge detection includes a variety of mathematical methods that aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities.

Module 3:

In this module neural network classifier is used in order to detect the road recognition.



Design Descriptions

NEURAL NETWORK

Linear Regression Models

Regression models describe the relationship between a *dependent variable*, y , and *independent variable* or variables, X . The dependent variable is also called the *response variable*. Independent variables are also called *explanatory* or *predictor variables*. Continuous predictor variables might be called *covariates*, whereas categorical predictor variables might be also referred to as *factors*. The matrix, X , of observations on predictor variables is usually called the *design matrix*.

A multiple linear regression model is

$$y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_p X_{ip} + \varepsilon_i, \quad i = 1, \dots, n,$$

where

- y_i is the i th response.
- β_k is the k th coefficient, where β_0 is the constant term in the model. Sometimes, design matrices might include information about the constant term. However, `LinearModel.fit` or `LinearModel.stepwise` by default includes a constant term in the model, so you must not enter a column of 1s into your design matrix X .
- X_{ij} is the i th observation on the j th predictor variable, $j = 1, \dots, p$.
- ε_i is the i th noise term, that is, random error.

In general, a linear regression model can be a model of the form

$$y_i = \beta_0 + \sum_{k=1}^K \beta_k f_k(X_{i1}, X_{i2}, \dots, X_{ip}) + \varepsilon_i, \quad i = 1, \dots, n,$$

where $f(\cdot)$ is a scalar-valued function of the independent variables, X_{ij} s. The functions, $f(X)$, might be in any form including nonlinear functions or polynomials. The linearity, in the linear regression models, refers to the linearity of the coefficients β_k . That is, the response variable, y , is a linear function of the coefficients, β_k .

Some examples of linear models are:

$$y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_{3i} + \varepsilon_i$$

$$y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_{1i}^3 + \beta_4 X_{2i}^2 + \varepsilon_i$$

$$y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 X_{1i} X_{2i} + \beta_4 \log X_{3i} + \varepsilon_i$$

The following, however, are not linear models since they are not linear in the unknown coefficients, β_k .

$$\log y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \varepsilon_i$$

$$y_i = \beta_0 + \beta_1 X_{1i} + \frac{1}{\beta_2 X_{2i}} + e^{\beta_3 X_{1i} X_{2i}} + \varepsilon_i$$

The usual assumptions for linear regression models are:

- The noise terms, ε_i , are uncorrelated.
- The noise terms, ε_i , have independent and identical normal distributions with mean zero and constant variance, σ^2 . Thus

$$\begin{aligned} E(y_i) &= E\left(\sum_{k=0}^K \beta_k f_k(X_{i1}, X_{i2}, \dots, X_{ip}) + \varepsilon_i\right) \\ &= \sum_{k=0}^K \beta_k f_k(X_{i1}, X_{i2}, \dots, X_{ip}) + E(\varepsilon_i) \\ &= \sum_{k=0}^K \beta_k f_k(X_{i1}, X_{i2}, \dots, X_{ip}) \end{aligned}$$

and

$$V(y_i) = V\left(\sum_{k=0}^K \beta_k f_k(X_{i1}, X_{i2}, \dots, X_{ip}) + \varepsilon_i\right) = V(\varepsilon_i) = \sigma^2$$

So the variance of y_i is the same for all levels of X_{ij} .

- The responses y_i are uncorrelated.

The fitted linear function is

$$\hat{y}_i = b_0 + \sum_{k=1}^K b_k f_k(X_{i1}, X_{i2}, \dots, X_{ip}), \quad i = 1, \dots, n,$$

where $\hat{y}_i$ is the estimated response and b_k s are the fitted coefficients. The coefficients are estimated so as to minimize the mean squared difference between the prediction vector $\hat{bf}(X)$ and the true response vector y , that is $\hat{y} - y$. This method is called the *method of least squares*. Under the assumptions on the noise terms, these coefficients also maximize the likelihood of the prediction vector.

In a linear regression model of the form $y = \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p$, the coefficient β_k expresses the impact of a one-unit change in predictor variable, X_j , on the mean of the

response, $E(y)$ provided that all other variables are held constant. The sign of the coefficient gives the direction of the effect. For example, if the linear model is $E(y) = 1.8$

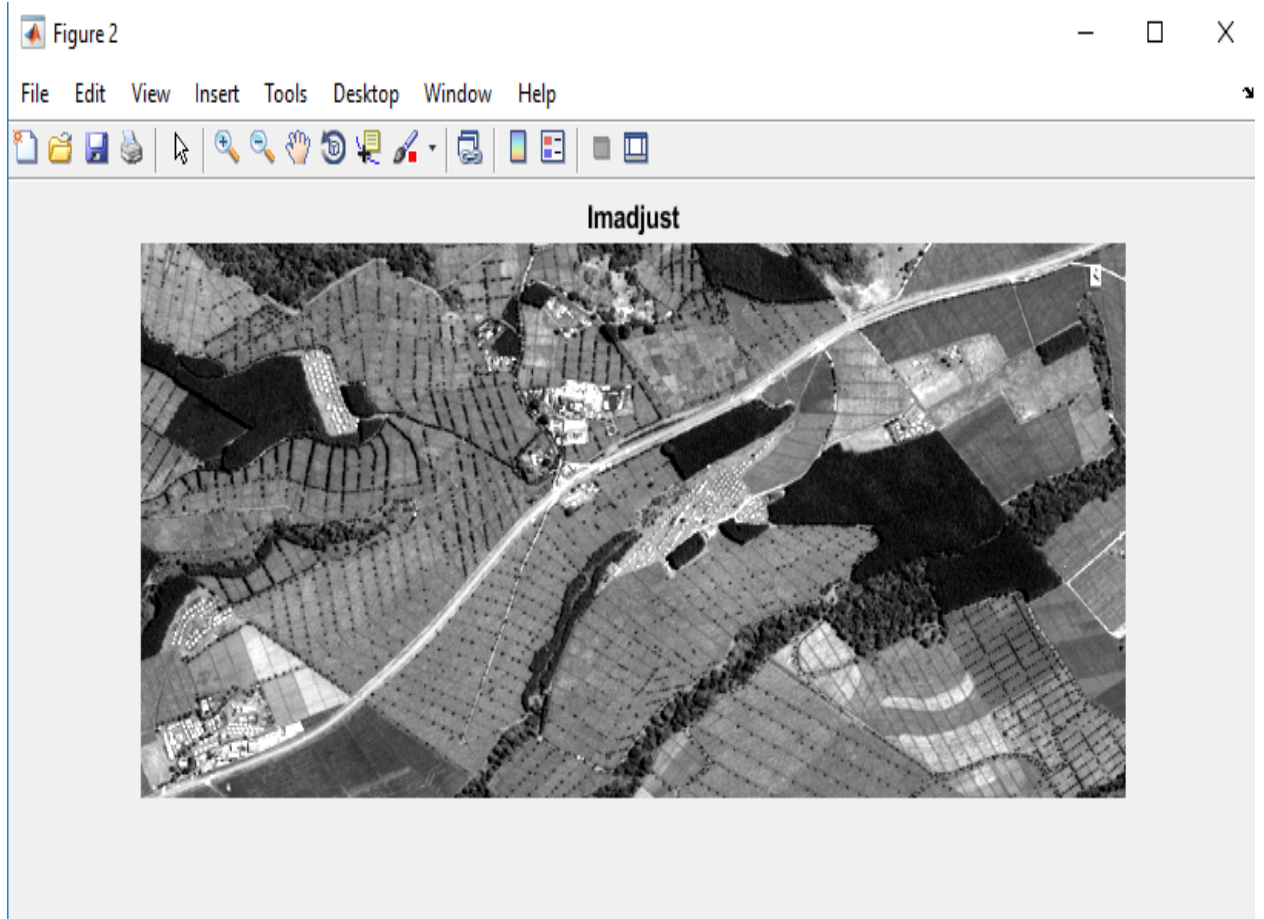
$- 2.35X_1 + X_2$, then -2.35 indicates a 2.35 unit decrease in the mean response with a one-unit increase in X_1 , given X_2 is held constant. If the model is $E(y) = 1.1 + 1.5X_1^2 + X_2$, the coefficient of X_1^2 indicates a 1.5 unit increase in the mean of Y with a one-unit increase in X_1^2 given all else held constant. However, in the case of $E(y) = 1.1 + 2.1X_1 + 1.5X_1^2$, it is difficult to interpret the coefficients similarly, since it is not possible to hold X_1 constant when X_1^2 changes or vice versa.

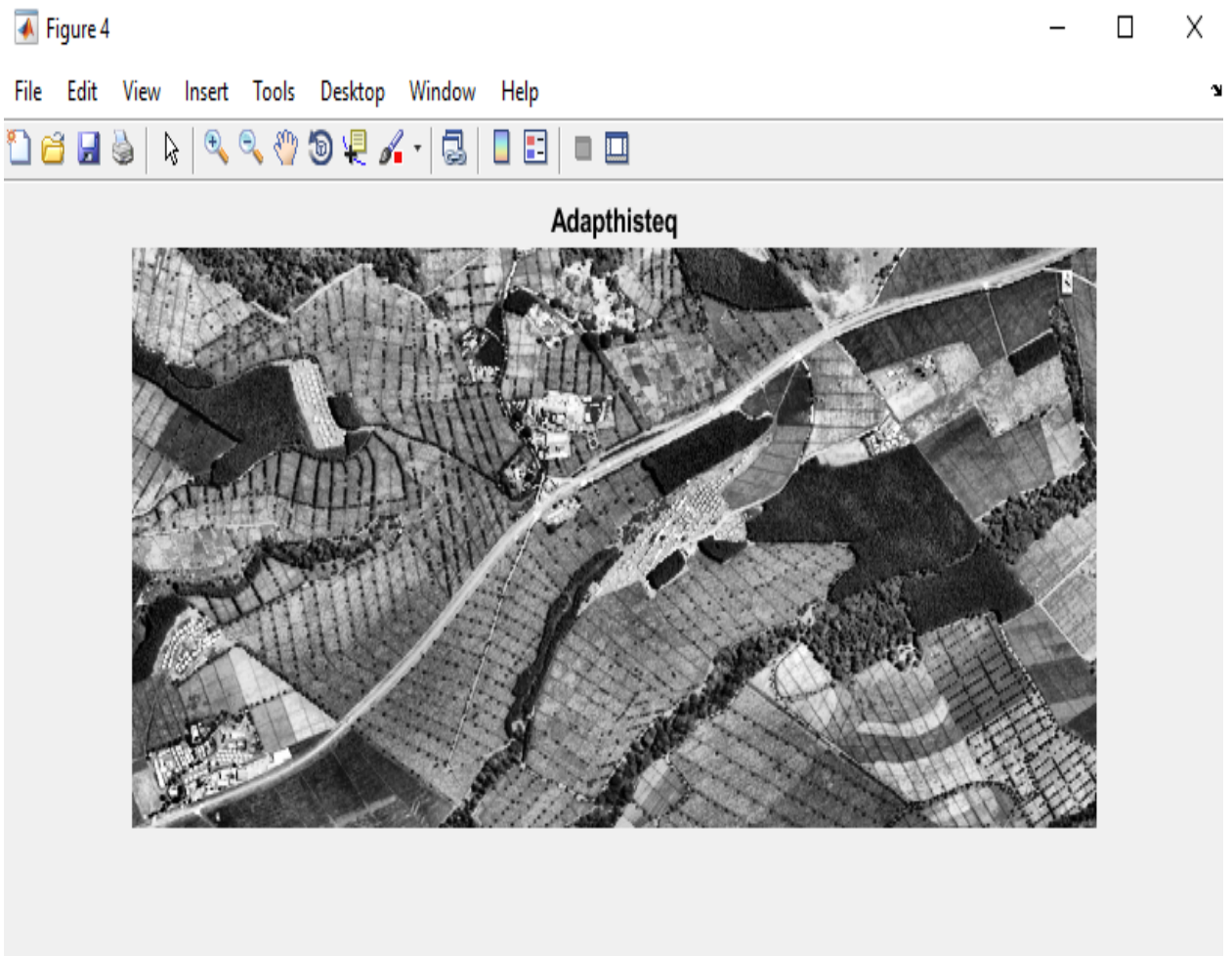
CHAPTER 5

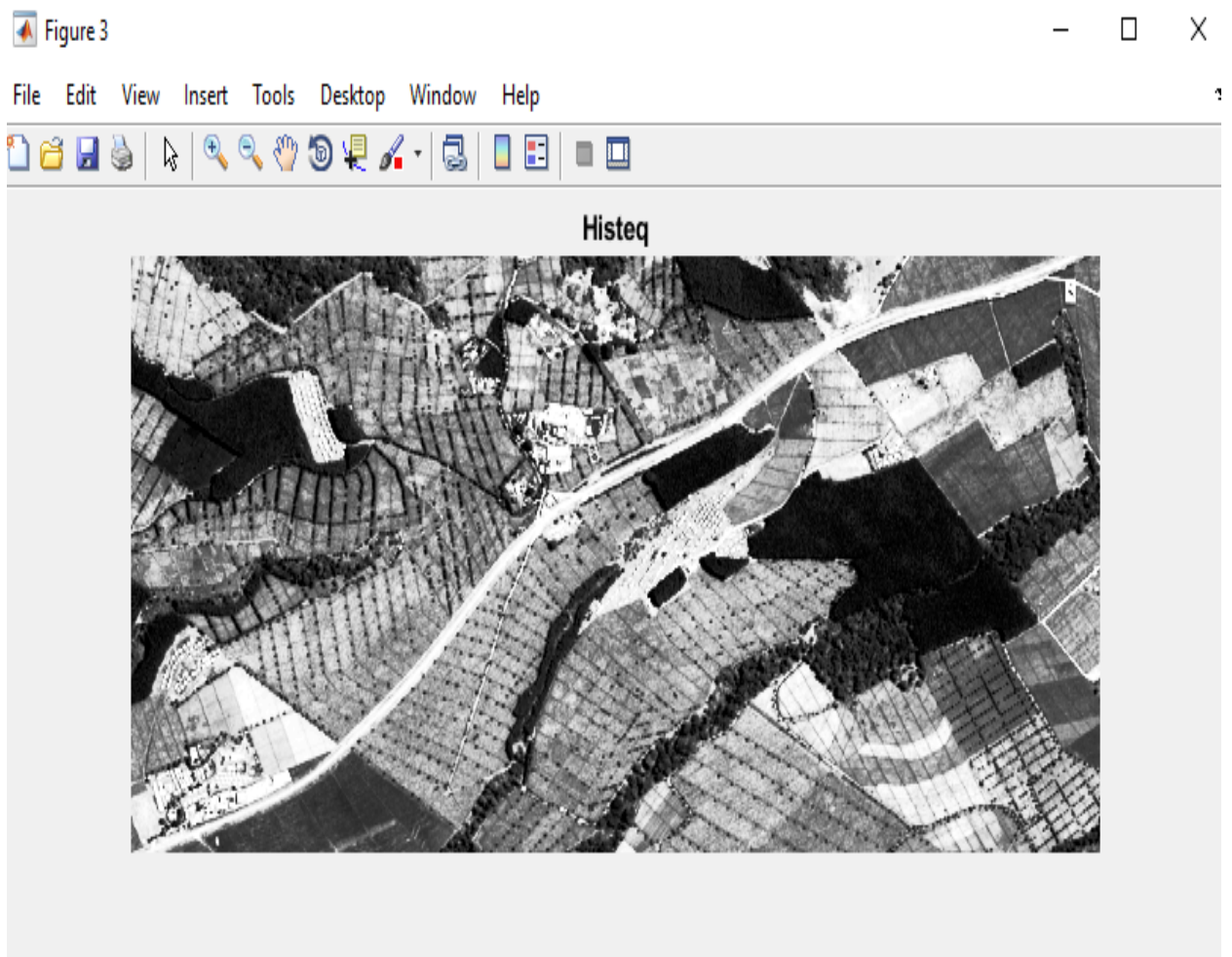
RESULTS

Original



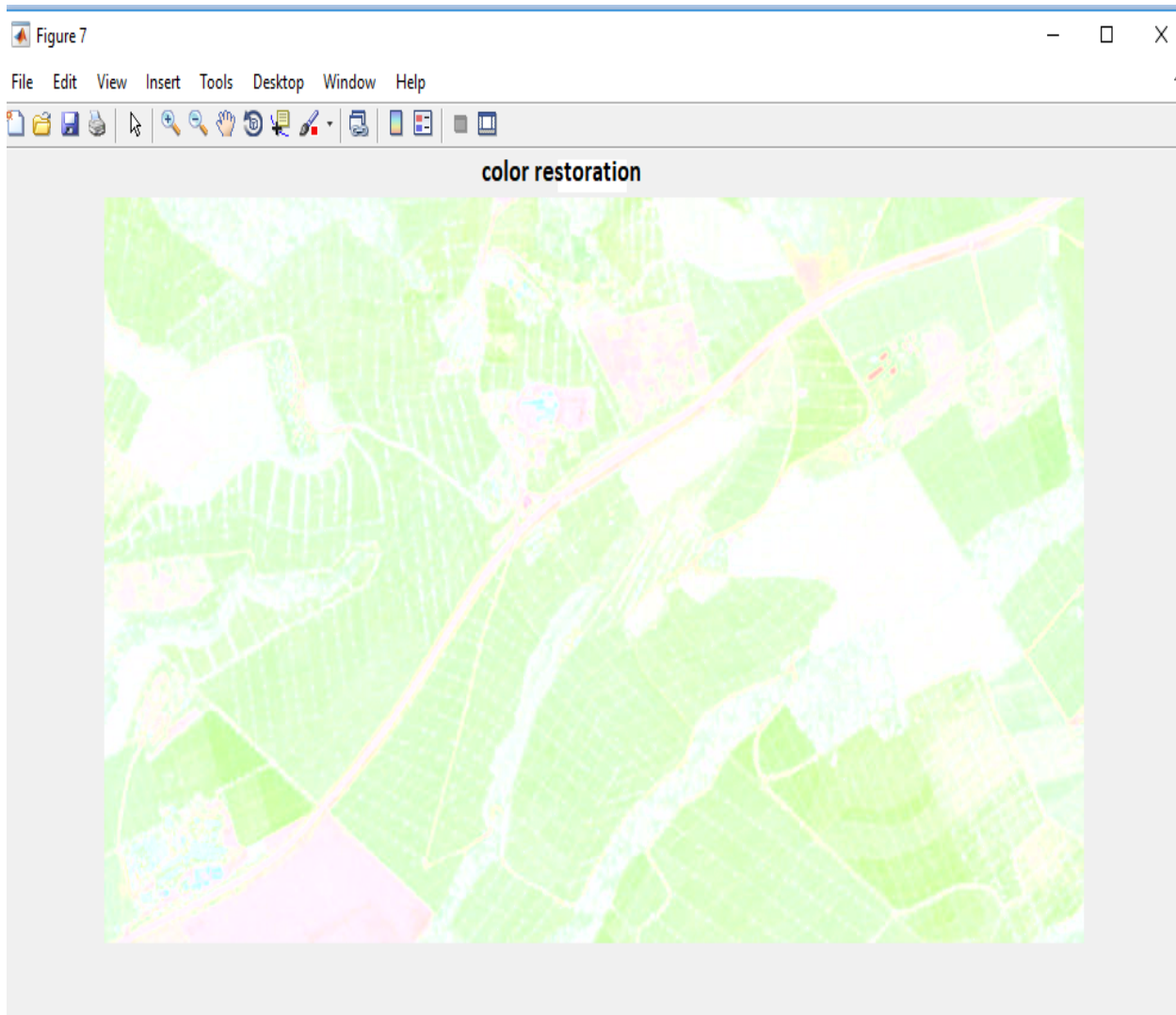


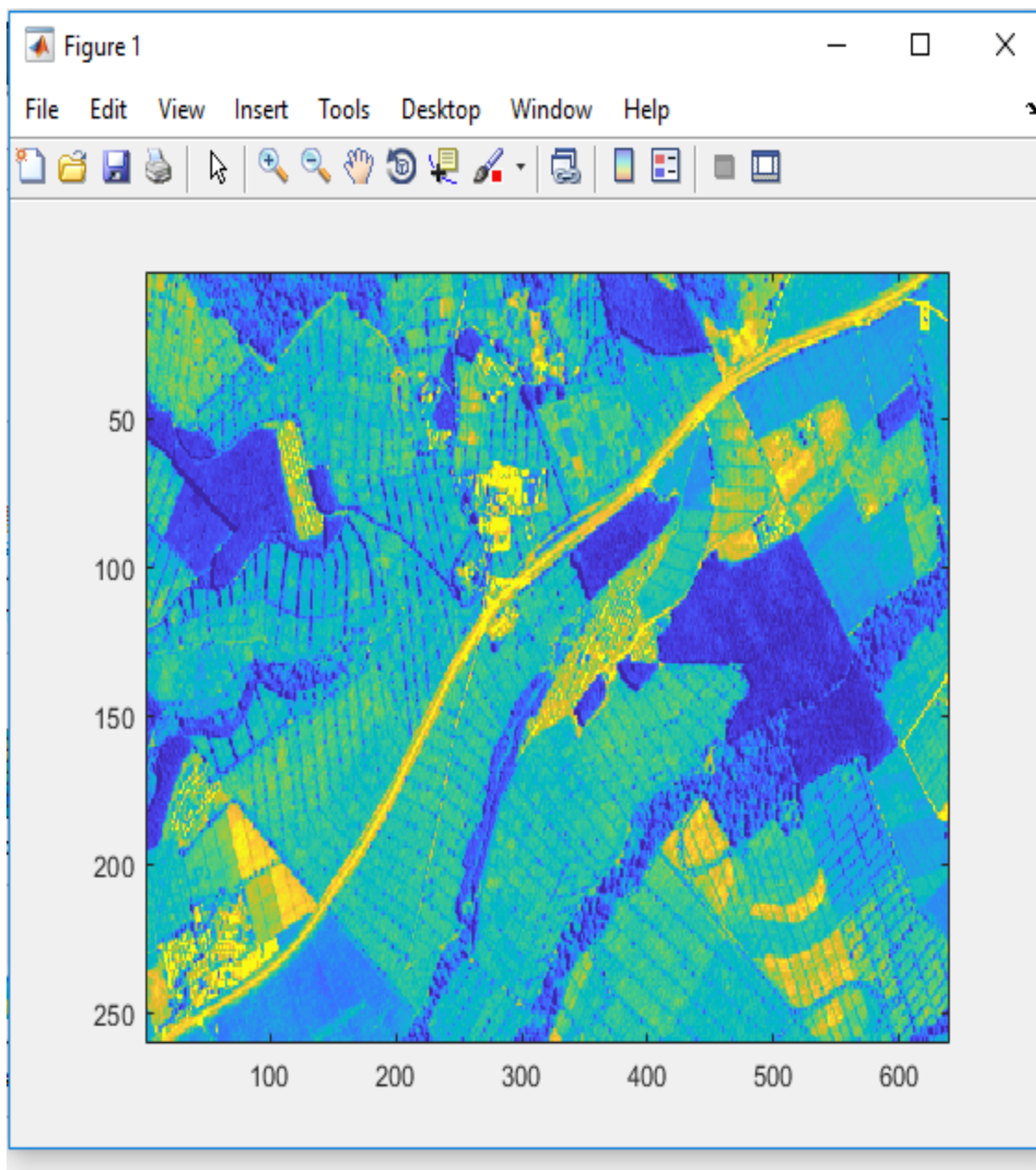




Imadjust









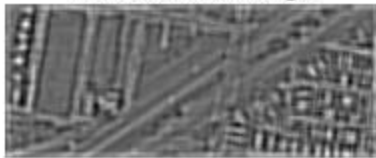
Original Image



grayscale Image



DOG Filtered Image



Binarized



CODING & TESTING

5.1 CODING

Once the design aspect of the system is finalized the system enters into the coding and testing phase. The coding phase brings the actual system into action by converting the design of the system into the code in a given programming language. Therefore, a good coding style has to be taken whenever changes are required it easily screwed into the system.

5.2 CODING STANDARDS

Coding standards are guidelines to programming that focuses on the physical structure and appearance of the program. They make the code easier to read, understand and maintain. This phase of the system actually implements the blueprint developed during the design phase. The coding specification should be in such a way that any programmer must be able to understand the code and can bring about changes whenever felt necessary. Some of the standard needed to achieve the above-mentioned objectives are as follows:

Program should be simple, clear and easy to understand.

Naming conventions

Value conventions

Script and comment procedure

Message box format

Exception and error handling

5.2.1 NAMING CONVENTIONS

Naming conventions of classes, data member, member functions, procedures etc., should be self-descriptive. One should even get the meaning and scope of the variable by its name. The conventions are adopted for easy understanding of the intended message by the user. So it is customary to follow the conventions. These conventions are as follows:

Class names

Class names are problem domain equivalence and begin with capital letter and have mixed cases.

Member Function and Data Member name

Member function and data member name begins with a lowercase letter with each subsequent letters of the new words in uppercase and the rest of letters in lowercase.

5.2.2 VALUE CONVENTIONS

Value conventions ensure values for variable at any point of time. This involves the following:

Proper default values for the variables.

Proper validation of values in the field.

Proper documentation of flag values.

5.2.3 SCRIPT WRITING AND COMMENTING STANDARD

Script writing is an art in which indentation is utmost important. Conditional and looping statements are to be properly aligned to facilitate easy understanding. Comments are included to minimize the number of surprises that could occur when going through the code.

5.2.4 MESSAGE BOX FORMAT

When something has to be prompted to the user, he must be able to understand it properly. To achieve this, a specific format has been adopted in displaying messages to the user. They are as follows:

X – User has performed illegal operation.

! – Information to the user.

5.3 TEST PROCEDURE

SYSTEM TESTING

Testing is performed to identify errors. It is used for quality assurance. Testing is an integral part of the entire development and maintenance process. The goal of the testing during phase is to verify that the specification has been accurately and completely incorporated into the design, as well as to ensure the correctness of the design itself. For example the design must not have any logic faults in the design is detected before coding commences, otherwise the cost of fixing the faults will be considerably

higher as reflected. Detection of design faults can be achieved by means of inspection as well as walkthrough.

Testing is one of the important steps in the software development phase. Testing checks for the errors, as a whole of the project testing involves the following test cases: Static analysis is used to investigate the structural properties of the Source code. Dynamic testing is used to investigate the behavior of the source code by executing the program on the test data.

5.4 TEST DATA AND OUTPUT

5.4.1 UNIT TESTING

Unit testing is conducted to verify the functional performance of each modular component of the software. Unit testing focuses on the smallest unit of the software design (i.e.), the module. The white-box testing techniques were heavily employed for unit testing.

5.4.2 FUNCTIONAL TESTS

Functional test cases involved exercising the code with nominal input values for which the expected results are known, as well as boundary values and special values, such as logically related inputs, files of identical elements, and empty files.

Three types of tests in Functional test:

Performance Test

Stress Test

Structure Test

5.4.3 PERFORMANCE TEST

It determines the amount of execution time spent in various parts of the unit, program throughput, and response time and device utilization by the program unit.

5.4.4 STRESS TEST

Stress Test is those test designed to intentionally break the unit. A Great deal can be learned about the strength and limitations of a program by examining the manner in which a programmer in which a program unit breaks.

5.4.5 STRUCTURED TEST

Structure Tests are concerned with exercising the internal logic of a program and traversing particular execution paths. The way in which White-Box test strategy was employed to ensure that the test cases could Guarantee that all independent paths within a module have been have been exercised at least once.

Exercise all logical decisions on their true or false sides.

Execute all loops at their boundaries and within their operational bounds.

Exercise internal data structures to assure their validity.

Checking attributes for their correctness.

Handling end of file condition, I/O errors, buffer problems and textual errors in output information

5.4.6 INTEGRATION TESTING

Integration testing is a systematic technique for construction the program structure while at the same time conducting tests to uncover errors associated with interfacing. i.e., integration testing is the complete testing of the set of modules which makes up the product. The objective is to take untested modules and build a program structure tester should identify critical modules. Critical modules should be tested as early as possible. One approach is to wait until all the units have passed testing, and then combine them and then tested. This approach is evolved from unstructured testing of small programs. Another strategy is to construct the product in increments of tested units. A small set of modules are integrated together and tested, to which another module is added and tested in combination. And so on. The advantages of this approach are that, interface dispenses can be easily found and corrected.

The major error that was faced during the project is linking error. When all the modules are combined the link is not set properly with all support files. Then we checked out for interconnection and the links. Errors are localized to the new module and its intercommunications. The product development can be staged, and modules integrated in as they complete unit testing. Testing is completed when the last module is integrated and tested.

5.5 TESTING TECHNIQUES / TESTING STRATEGIES

5.5.1 TESTING

Testing is a process of executing a program with the intent of finding an error. A good test case is one that has a high probability of finding an as-yet –undiscovered error. A successful test is one that uncovers an as-yet- undiscovered error. System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as expected before live operation commences. It verifies that the whole set of programs hang together. System testing requires a test consists of several key activities and steps for run program, string, system and is important in adopting a successful new system. This is the last chance to detect and correct errors before the system is installed for user acceptance testing.

The software testing process commences once the program is created and the documentation and related data structures are designed. Software testing is essential for correcting errors. Otherwise the program or the project is not said to be complete. Software testing is the critical element of software quality assurance and represents the ultimate the review of specification design and coding. Testing is the process of executing the program with the intent of finding the error. A good test case design is one that as a probability of finding an yet undiscovered error. A successful test is one that uncovers an yet undiscovered error. Any engineering product can be tested in one of the two ways:

5.5.1.1 WHITE BOX TESTING

This testing is also called as Glass box testing. In this testing, by knowing the specific functions that a product has been design to perform test can be conducted that demonstrate each function is fully operational at the same time searching for errors in each function. It is a test case design method that uses the control structure of the procedural design to derive test cases. Basis path testing is a white box testing.

Basis path testing:

Flow graph notation

Cyclometric complexity

Deriving test cases

Graph matrices Control

5.5.1.2 BLACK BOX TESTING

In this testing by knowing the internal operation of a product, test can be conducted to ensure that “all gears mesh”, that is the internal operation performs according to specification and all internal components have been adequately exercised. It fundamentally focuses on the functional requirements of the software.

5.5.2 SOFTWARE TESTING STRATEGIES:

A software testing strategy provides a road map for the software developer. Testing is a set activity that can be planned in advance and conducted systematically. For this reason a template for software testing a set of steps into which we can place specific test case design methods should be strategy should have the following characteristics:

Testing begins at the module level and works “outward” toward the integration of the entire computer based system.

Different testing techniques are appropriate at different points in time.

The developer of the software and an independent test group conducts testing.

Testing and Debugging are different activities but debugging must be accommodated in any testing strategy.

5.5.2.1 INTEGRATION TESTING:

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with. Individual modules, which are highly prone to interface errors, should not be assumed to work instantly when we put them together. The problem of course, is “putting them together”- interfacing. There may be the chances of data lost across on another’s sub functions, when combined may not produce the desired major function; individually acceptable impression may be magnified to unacceptable levels; global data structures can present problems.

5.5.2.2 PROGRAM TESTING:

The logical and syntax errors have been pointed out by program testing. A syntax error is an error in a program statement that in violates one or more rules of the language in which it is written. An improperly defined field dimension or omitted

keywords are common syntax error. These errors are shown through error messages generated by the computer. A logic error on the other hand deals with the incorrect data

fields, out-of-range items and invalid combinations. Since the compiler s will not deduct logical error, the programmer must examine the output. Condition testing exercises the logical conditions contained in a module. The possible types of elements in a condition include a Boolean operator, Boolean variable, a pair of Boolean parentheses A relational operator or on arithmetic expression. Condition testing method focuses on testing each condition in the program the purpose of condition test is to deduct not only errors in the condition of a program but also other a errors in the program.

5.5.2.3 SECURITY TESTING:

Security testing attempts to verify the protection mechanisms built in to a system well, in fact, protect it from improper penetration. The system security must be tested for invulnerability from frontal attack must also be tested for invulnerability from rear attack. During security, the tester places the role of individual who desires to penetrate system.

5.5.2.4 VALIDATION TESTING

At the culmination of integration testing, software is completely assembled as a package. Interfacing errors have been uncovered and corrected and a final series of software test-validation testing begins. Validation testing can be defined in many ways, but a simple definition is that validation succeeds when the software functions in manner that is reasonably expected by the customer. Software validation is achieved through a series of black box tests that demonstrate conformity with requirement. After validation test has been conducted, one of two conditions exists.

- * The function or performance characteristics confirm to specifications and are accepted.

- * A validation from specification is uncovered and a deficiency created.

Deviation or errors discovered at this step in this project is corrected prior to completion of the project with the help of the user by negotiating to establish a method for resolving deficiencies. Thus the proposed system under consideration has been tested by using validation testing and found to be working satisfactorily. Though there were deficiencies in the system they were not catastrophic.

5.5.2.5 USER ACCEPTANCE TESTING

User acceptance of the system is key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch

with prospective system and user at the time of developing and making changes whenever required. This is done in regarding to the following points.

- Input screen design.
- Output screen design.

Road Accident Detection and Prevention Using Machine Learning

K.Revanthmanas¹,K.Aditya¹, Dr.S.Jancy³

^{1,2}UG Student, Dept. of CSE, Sathyabama Institute of Science and Technology, Chennai, India

^{3, 4,5}Associate professor, Dept. of CSE, Sathyabama Institute of Science and Technology, Chennai, India
Email id: jancyphd16@gmail.com

ABSTRACT

There are numerous investigations on the counteraction or discovery of auto collisions. The vast majority of them include hearing things that could cause mishaps or insights of mishaps. In this examination the danger recognizable proof framework will be considered. The framework will gather the necessary data from close by vehicles and use AI devices to handle data to distinguish expected perils. AI calculations have shown accomplishment in recognizing ordinary conduct and typical conduct. The reason for this investigation is to examine gridlock and to treat vehicles driving uniquely in contrast to as of now. The outcomes show that the combining calculation can successfully recognize hazards.

I. INTRODUCTION

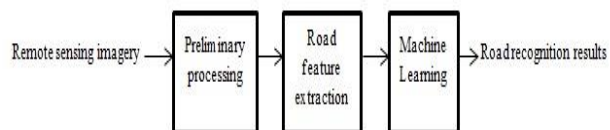
A hybrid convolution network (HCN) used to isolate streets from inaccessible visual sharpness pictures to accomplish significant degrees of mishaps on different streets with complex foundations. FCN has had various bogus positives behind an intricate area, what isolates numerous wide streets, however avoided many limited streets and neglected to recognize appropriate street limits. FCN horrible showing on limited streets is because of an absence of street information during examining down to the presentation of the example.

II. OBJECTIVES

The principle targets of the street mishap determining framework are:

* Analysis of past chances area, this will help us settle on more choices regions inclined to threat and help us

III. SYSTEM ARCHITECTURE



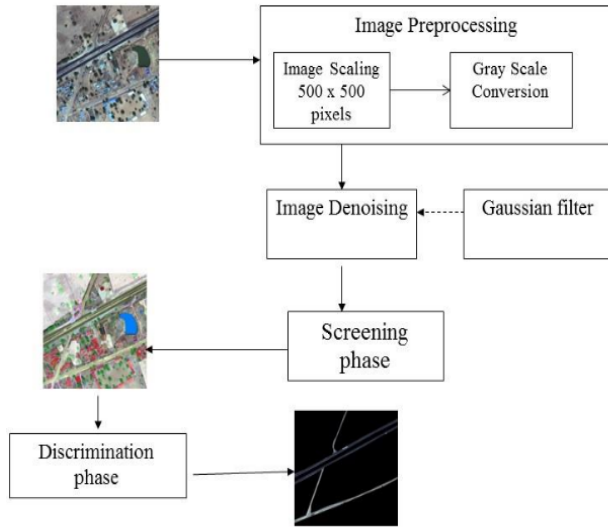


Figure 1.1 System Architecture

IV. LITERATURE SURVEY

Unsupervised feature selection with ensemble learning

In this article, we show that the strategy for interior estimation is utilized to gauge the estimation of fluctuation in irregular woodlands additionally applies to consolidating unattended understanding alternatives. We propose another methodology called the Random Cluster Ensemble (RCE), which estimates the estimation of outer variables in a progression of divisions. Every section is built utilizing an irregular subset of different bootstrap tests and capacities. We present the immovability brings about 19 benchmark informational indexes, showing that RCE (Guyon and Elisseeff, Journal of Machine Learning Research, 3: 1157-1182, 2003) has built up an improved rehashing factor (RFE) framework. very Among those directed and undeniable level unaided abilities, there is a restricted breaking point on the blend of a bunch of highlights. This methodology shows the guarantee of managing extremely huge areas.

A Novel Approach to Automatic Road-Accident Detection using Machine Vision Techniques

This paper proposes another sort of programmed street mishap location technique. This methodology depends on the location of harmed vehicles in pictures got by observation cameras introduced on streets and expressways, demonstrating the event of a street mishap. In the field of mechanical perception, the disclosure of harmed vehicles is currently securing, which has not yet been finished. This paper proposes another administered learning technique, which incorporates three separate segments, gathered into a clear casing, and can successfully distinguish harmed vehicles in standing pictures. These three stages utilize five vector support machines, prepared with histogram angles (HOG) and dark level co-event lattice (GLCM) capacities. As no endeavor has been made to find the harmed vehicles, two vehicle information bases - Damaged Car Data

Set 1 (DCD-1) and Damaged Car Data Set 2 (DCD-2) - have been delivered openly. Tests are performed on DCD-1 and DCD-2, and these tests fluctuate dependent on the distance of the caught picture and the nature of the picture. With great quality DCD-1 introduced at around 2 meters, framework precision is 81.83%, and with great quality DCD-2 introduced at around 20 meters, framework exactness is 64.37%.

V. MODULES

- Preprocessing
- Edge detection
- Accident detection

PREPROCESSING

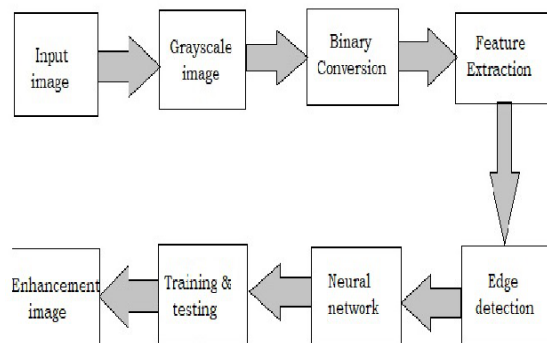


Figure 1.2.Data Preprocessing

On the off chance that the info picture is a shading picture, it implies that we have changed over from a shading picture to grayscale. It would then be able to be changed over to a twofold transformation (0 methods dark, 1 methods white). The center channel is utilized to eliminate commotion in the picture.

EDGE DETECTION

Edge recognition is a strategy for finding the limit of an article in a picture. Edge recognition incorporates an assortment of numerical strategies to distinguish focuses in an advanced picture where the brilliance of the picture changes more or more methodically than when it quits working.

ACCIDENT DETECTION

- In this module neural network classifier is used in order to detect the road recognition.

METHODOLOGY

- Method of approach should be presented clearly with its outline and justification.
- Describe the technology, software tools, analysis techniques, measurement technique used etc.

- Your methodology should also describe your objective and how you are going to achieve your objective.

VI. RESULTS AND DISCUSSION

This movement has utilized high-goal pictures gathered from Google Earth screen captures. Milestone is physically set apart by a reference map, and preset information is openly accessible. There are 224 pictures in the informational collection, and each picture has in any event 600×600 pixels. As a portrayal, the spatial change for every pixel picture a good ways off of 1.2 m. The information covers metropolitan, metropolitan and rustic zones. There are water, mountains and sloping zones covered with vegetation. A large portion of the first photos are found under complex territory, making street development a significant test. The genuine picture of the ground has two kinds of pixels: including street pixels and obscure pixels (mess); including pixels and pixels. The breadth of the genuine picture of the earth is around 12-15 pixels. To evade the utilization of pixels put away for free test informational collection for network readings, all examples and genuine pictures were intermittently partitioned into two sections, 80% used to prepare the GL-Dense-U-Net model, and 20% used to prepare the model. GL-Dense-U-Net. Used to approve a prepared model. The all out set of check information is sliced to 256×256

Distant hearing. 2018, 10, 1461 7 out of 16 examples not surpassing. To improve preparing tests, every unique picture and earth-related pictures are recorded with a 256×256 moment window (with 64 pixels dispersing). To expand informational collection measure and secure cushioning or invalid qualities, all consolidated square examples are turned 90°

VII. ALGORITHM FOR PROPOSED ALGORITHM

```

1.Load: Input Image
2.Stamp:=0
3.F(p)=uigetfile(image)
4.I=imread([p,f])
5.Ioriginal=I
6.Find:RGB Colour Thresholder
7.Get Dimension of the Image
8.Number of colour band =1
9.Number of colour band=size(grayImage_RoadSeg)
10.Apply Gray scale conversion

```

```

11.Determine Gaussian Filter
12.Determine dog Filter
13.Enhanced=histeq(road_seg,dog filter image)
14.Convolution-constant-test=(sum(sum(roadseg_dogfilterImage)))/
100
15.Test_data in= Test_data out
16.If convolution_constsnt_test= =
convolution_constant(loop_index)
17.Stamp= =loop_index
18.If stamp>= Thresholder
19.Result=no defect in the road
Else
20.Result=Defected

```

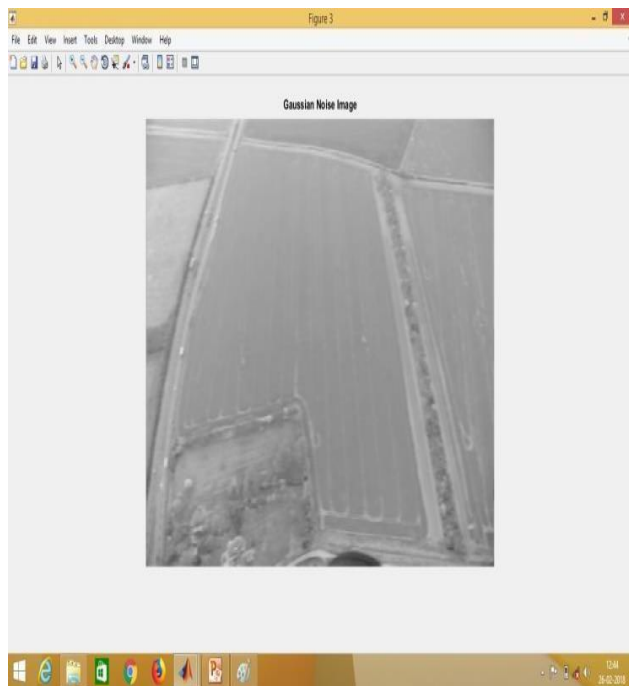


Figure 1.3 Gaussian Noise Image



Figure 1.4 Gaussian Filtering

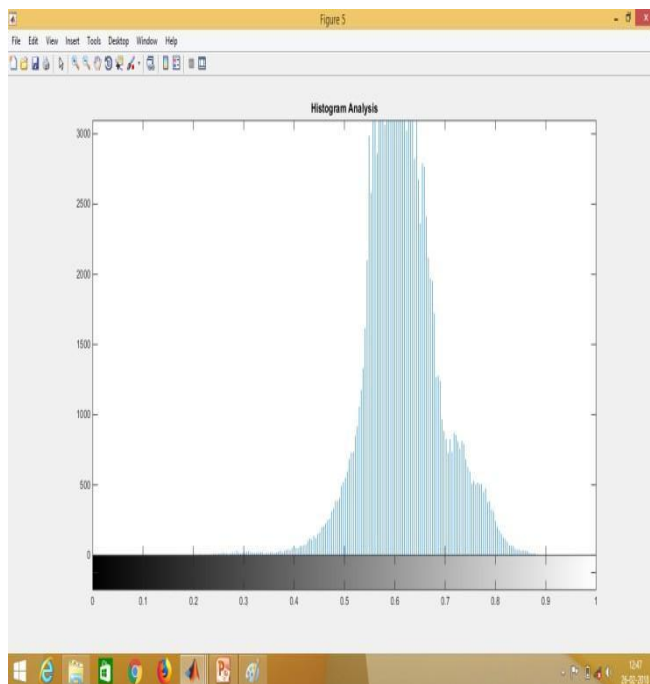


Figure 1.5. Histogram Analysis

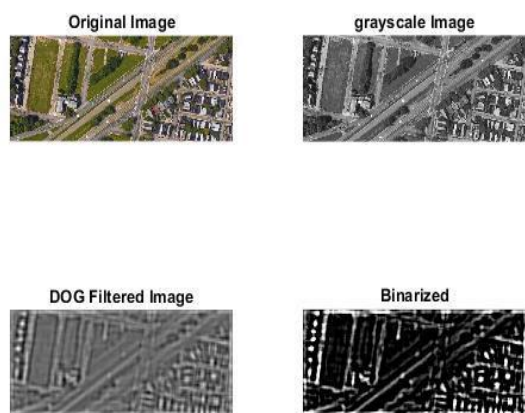


Figure 1.6 Original image converted in to Binaries

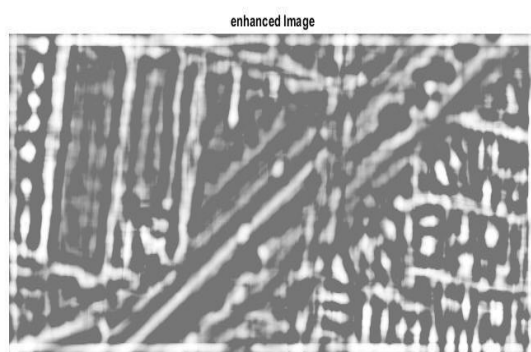


Figure 1.7 Enhanced Image

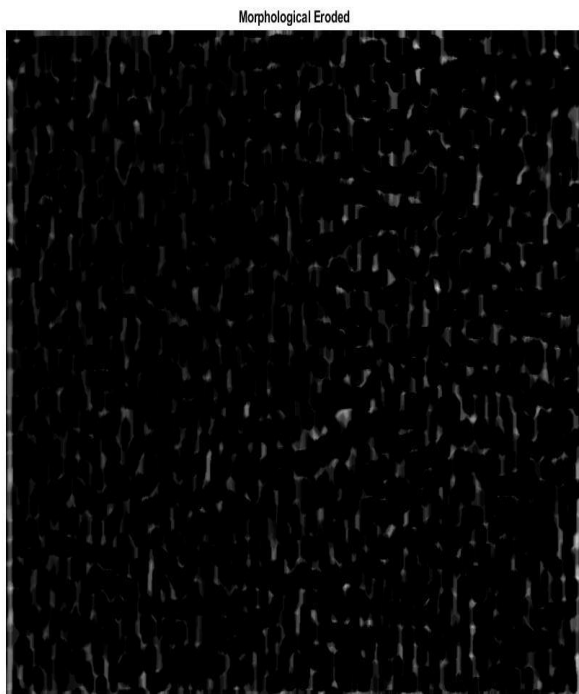


Figure 1.8. Morphological Eroded

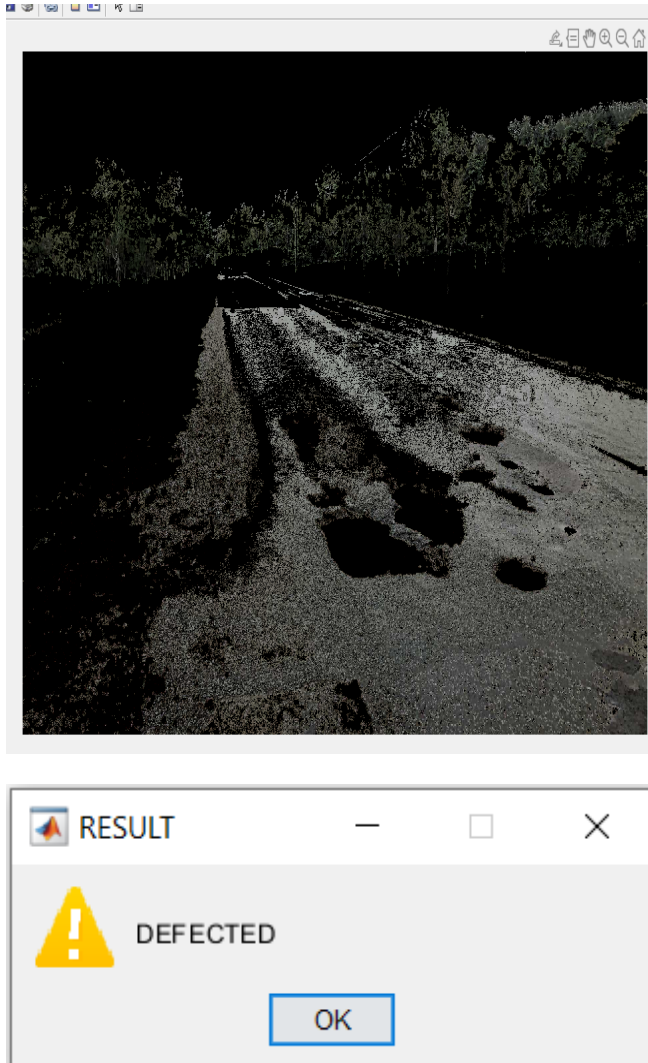


Figure 1.9. Defected Image

VIII. CONCLUSION

Programmed hazard recognition has gotten progressively significant in rush hour gridlock the board frameworks. Mishap discovery will dodge future mishaps and will help the experts in making traffic part of the street accessible once more. In this investigation, we have shown that the area and speed of a vehicle can be utilized to dissect traffic conduct, and that new articles out and about might be viewed as likely risks to drivers close to the location of a mishap.

REFERENCES

1. J Tang,XiaHu,HuijiGao,Huan Liu,” Unsupervised Feature Selection for Multi-View Data in Social Media”,Proceedings of the ACM international conference on Information and Knowledge management (2015),pp.1673-1678.

2. HaythamElzhazel and Alex Auseem, "Unsupervised Feature Selection with ensemble learning ", Machine Learning ,Springer (2013).
3. Mohak Shah, Mario Marchand, and Jacques Corbeil, " Feature Selection with Conjunctions of Decision Stumps and Learning ", Ieee Transactions On Pattern Analysis And Machine Intelligence, VOL. 34, NO. 1,(2012) pp.174-186.
4. Qingzheng XU, Lei WANG, Baomin HE, Na WANG," Modified Opposition-Based Differential Evolution for Function Optimization", Journal of Computational Information Systems 7:5 (2011) pp.1582-1591.
5. HaythamElzhazel and Alex Auseem, "Unsupervised Feature Selection with ensemble learning ", Machine Learning ,Springer (2013).
6. Mohak Shah, Mario Marchand, and Jacques Corbeil, "Feature Selection with Conjunctions of Decision Stumps and Learning ", Ieee Transactions On Pattern Analysis And Machine Intelligence, Vol. 34, No. 1,(2012) pp.174-186.
7. Qingzheng XU, Lei WANG, Baomin HE, Na WANG,"Modified Opposition-Based Differential Evolution for Function Optimization", Journal of Computational Information Systems 7:5 (2011) pp.1582-1591.