**1. INTRODUCTION**

**1.1 ABOUT PROJECT**

Nighttime vehicle detection has become an important task in intelligent transportation systems (ITS) in recent decades. It is also one of the key technologies for advanced driver assistance systems(ADAS) and autonomous driving systems (ADS). About 30% of all vehicular accidents are caused by rear-end collisions that are one of the most fatal traffic.

In this paper, we focus on detecting moving vehicles in front of the driver at night to avoid rear-end collisions. Some state-of-the-art object detectors are able to extract features from the original images taken in daytime. However, in night-time images, the contrast between background and object, and the overall brightness are so low that some details of vehicles (e.g., edge, color, and shape features of vehicles) become unclear. As a result, we should enhance the contrast, the brightness and details of the night time images before feature extraction for accurate vehicle detection .

Inspired by the retinal information processing mechanisms of the biological visual system, we propose an effective nighttime image enhancement approach that models several important steps of the retinal information processing mechanism. At night, the moving vehicles often turn on the taillights which are the most salient. Thus the taillights are very useful for extracting accurate regions of interest (ROIs). Generating a set of ROIs such as object proposal methods can improve the performance of current detection methods.

In this paper, we adopt the ROI extraction approach proposed in that combines vehicle taillight detection with Edge Boxes. Detection methods based on single features have been proved to be effective. However, when dealing with more complex scenes, these types of detection methods might lead to misclassiﬁcations. Therefore, we extract not only features from convolution neural network (CNN) but also compute two commonly used effective features: histogram of oriented gradient (HOG) and local binary pattern (LBP), to complement CNN features. Because we utilize multiple features, a key step is to combine them effectively. Score-level feature fusion has been reported to be effective.

We focus on how to make full use of the complementarily of each feature and the different capabilities of the same feature for different classes, and then develop a score-level feature fusion approach that combines the three features with weights learnt from scores using a linear SVM.

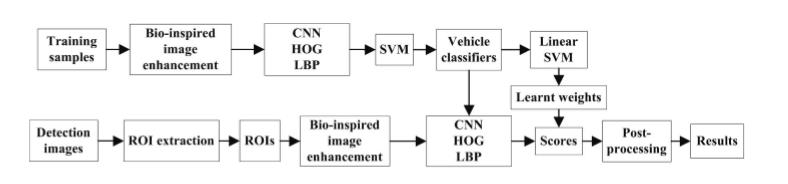


Fig 1.1

The framework of the proposed nighttime vehicle detection approach is shown in Figure1. During the training stage, the original training samples are enhanced by the proposed image enhancement approach and then three complementary features: CNN features, HOG and LBP are extracted from the enhanced images. Three SVM classiﬁers are trained with Lib SVM and ﬁve-fold cross-validation is carried out using each individual feature respectively. The score vectors of each classiﬁer are also computed. The scores are used to learn weights of each feature for each individual class using a linear SVM (Lib Linear). During the detection stage, accurate ROIs are extracted from the input images. Subsequently, the three features are extracted from each enhanced ROI. Next, the trained classiﬁers are used to classify the corresponding features and computes cores which are summed with learnt weights to obtain ﬁnals core vectors for prediction. Finally, post-processing (non maximum suppression) is performed to ensure that each vehicle is surrounded by a single window. This paper makes the following contributions:

A novel and effective bio-inspired nighttime image enhancement method is proposed to enhance the contrast, the brightness and details of nighttime images. Our approach is more effective than state-of-the-art methods. (2) To complement CNN features, we extract HOG and LBP, and then utilize an effective weighted score-level feature fusion based on learnt weights using a linear SVM to combine the three features. The way to learn weights and bias terms is different from state-of-the-art classiﬁer ensemble methods.

**2. LITERATURE SURVEY**

**Authors N. Dalal and B. Triggs,**

Some state-of-the-artobject detectors are able to extract features from the original images taken in daytime. However, in night time images, the contrast between background and object, and the overall brightness are so low that some details of vehicles (e.g., edge, color, and shape features of vehicles) become unclear. As a result ,we should enhance the contrast, the brightness and details of the nighttime images before feature extraction for accurate vehicle detection. Inspired by the retinal information processing mechanisms of the biological visual system, we propose an effective nighttime image enhancement approach that models several important steps of the retinal information processing mechanism.

Personalized recommendation based on reviews and ratings alleviating the sparsity problem of collaborative filtering

**Authors P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Raman**

At night, the moving vehicles often turn on the taillights which are the most salient. Thus the taillights are very useful for extracting accurate regions of interest (ROIs). Generating a set of ROIs such as object proposal methods can improve the performance of current detection methods. In this paper, we adopt the ROI extraction approach proposed in that combines vehicle taillight detection with EdgeBoxes.

**Authors T. Schamm, C. V. Carlowitz, and J. M. Zollner**

In nighttime scenes, the features of vehicle are not salient anymore. Thus, some vehicle detection methods focus on the vehicle lights, which are more salient at night. In these methods, one or a pair of detected vehicle lights represents a vehicle. The highlighted head-lights were detected by a model-based detection method . Light pixels were distinguished from reﬂection pixels by a MarkovRandom Field model that considered the reﬂection intensity and suppressed map, and the image intensity to represent and detect vehicles. In, night vehicle recognition was achieved by searching taillights using image processing techniques. However, these methods are sensitive to scene variations because vehicle lights are difﬁcult to detect accurately in complex scenes.

**3. SYSTEM ANALYSIS**

**3.1 EXISTING SYSTEM**

* Number-plate recognition is a technology that uses optical character recognition on images to read vehicle registration plates.
* Automatic number plate recognition can be used to store the images captured by the cameras as well as the text from the license plate, with some configurable to store a photograph of the driver.
* Systems commonly use infrared lighting to allow the camera to take the picture.
* It first uses a series of image manipulation techniques to detect, normalize and enhance the image of the number plate, and then optical character recognition (OCR) to extract the alphanumeric of the license plate.

**3.1.1 DISADVANTAGES**

* Poor  file resolution, usually because the plate is too far away but sometimes resulting from the use of a low-quality camera.
* Blurry images, particularly motion blur.
* Poor lighting and low contrast due to overexposure, reflection or shadows.
* A different font, popular for vanity plates (some countries do not allow such plates, eliminating the problem).

**3.2 PROPOSED SYSTEM**

* This paper presents an effective night time vehicledetection system that combines a novel bio-inspired image enhancement approach with a weighted feature fusion technique.
* At night, the moving vehicles often turn on the tail-lights which are the most salient.
* Thus the tail lights are very useful for extracting accurate regions of interest (ROIs). Generating a set of ROIs such as object proposal methods can improve the performance of current detection methods.
* In this paper, we adopt the ROI extraction approach proposed in that combines vehicle tail-light detection with Edge Boxes .
* A novel and effective bio-inspired night time image enhancement method is proposed to enhance the contrast, the brightness and details of night time images.

**3.2.1 ADVANTAGES**

* Our proposed method can deal with various scenes including vehicles of different types and sizes and those with occlusions and in blurred zones.
* It can also detect vehicles at various locations and multiple vehicles.

**3.3 SOFTWARE REQUIREMENT ANALYSIS**

**3.3.1 Software Requirement Specification**

The Software Requirement Specification (SRS) will form the basis for the development and testing of the envisaged application. The SRS document is expected to describe all the functional requirements of the system without going into the implementation details. The SRS document will serve as the input document for design phase of the software project.

**3.3.2 Functional requirements**

Functional requirements describe what the system should do, i.e. the services provided for the users and for other systems.

**3.3.2.1 INPUT**

* The admin should register first into the application.
* The admin should login into the system.
* The admin should search the image for the vehicle number.
* The admin will check the user details.
* The admin can cancel the license if the limit is exceeded.
* The admin can verify the status of the user.

**3.3.2.2 OUTPUT**

* Admin can navigate to next page if login is successful.
* User details are tabulated if the admin enters the vehicle number.
* User status is displayed when admin enters his/her vehicle number.
* The image is preprocessed and an output image is generated when the admin runs the file with an input image.

**3.3.3 Non-Functional Requirements**

In non-functional requirements the following are the things that come under .They are as follows:

1. **Portability:** As the application is designed with python and php as programming languages, we know java can be run on any operating system. Hence the application is portable to run on any operating system.
2. **Extensibility:** The application can be extended at any level if the user wish to extend that in future this is done because python and php are open source mediums which doesn’t have any time limits for expiry or renewal.
   * 1. **MODULES**

**3.3.4.1 ADMIN**

In this module, the Admin has to login by using valid user name and password. After login successful he can do some operations such as check user details, check status of the user.

**3.3.4.1.1 Check user details**

In this module, the admin can search for the user giving the vehicle number as the input.

**3.3.4.1.2 Check status**

In this module, the admin can view the license status of the vehicle holder based on the given input.

* + - 1. **SYSTEM**

In this module, the given input image is preprocessed and an enhanced output image is generated to read the vehicle plate.

* + - 1. **USER**

In this module, the particular vehicle holder will receive a message from the admin.

**3.4 SOFTWARE REQUIREMENTS/ENVIRONMENNT**

* Windows 7, 8, 10, Server 2008, Server 2012, 64 bit (PC or Mac computers using Boot Camp).
* Python 3.6.
* Php.
* Database – Oracle.
* Front End - HTML,CSS, Java Script.

**3.5 HARDWARE REQUIREMENTS/ENVIRONMENT**

* Any CPU(Intel i5+/ Xeon recommended).
* Any GPU that is compatible with OpenGL 3.2.(Integrated graphic cards Intel HD 4000 or above).
* Small project (under 100 images at 14 MP): 4 GB RAM, 10 GB HDD Free Space.
* Medium projects(between 100 and 500 images at 14 MP): 8 GB RAM, 20 GB HDD Free Space.
* Large projects (between 500 and 2000 images at 14 MP): 16 GB RAM, 40 GB HDD Free Space.
* Very Large projects (over 2000 images at 14 MP): 16 GB RAM, 80 GB HDD Free Space.

**4. SOFTWARE DESIGN**

In the context of software, design is problem solving process whose objective is to find and describe a way to implement the functional requirements while respecting the constraints imposed by the non-functional requirements and by adhering to general principles of good quality. The goal of the design process is to produce a model or representation of a system which can be used later to build that system and use this model to build the overall system.

**4.1 SYSTEM ARCHITECTURE:**

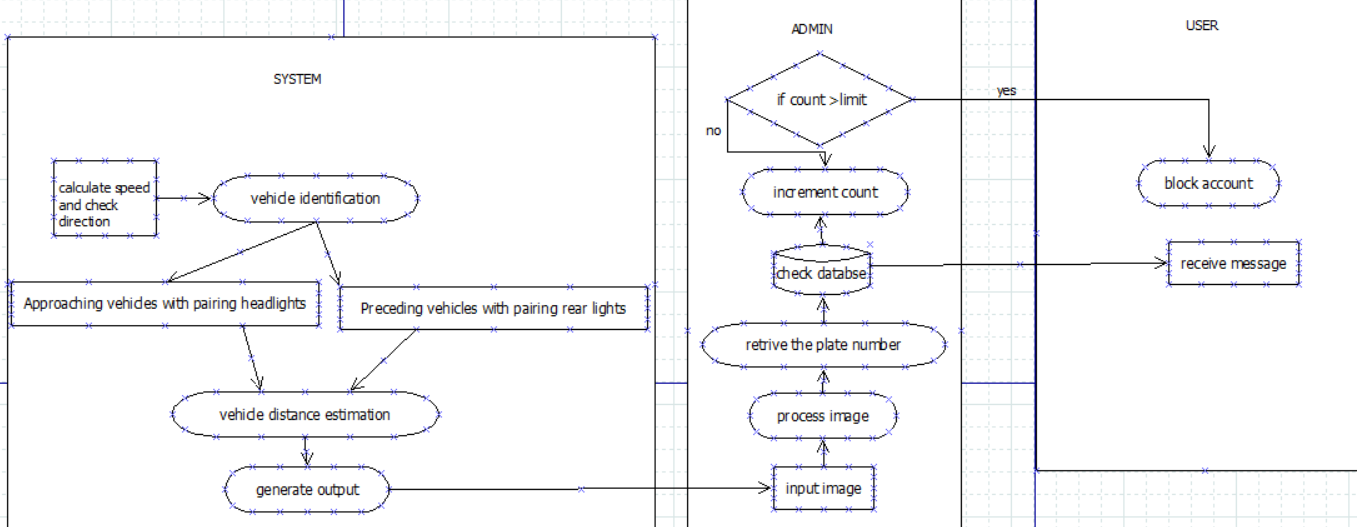


Figure 4.1

**4.2 Data FLOW DIAGRAM**

****

Fig 4.2

**4.3 ER DIAGRAM**

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is a component of data. In other words, ER diagrams illustrate the logical structure of databases.

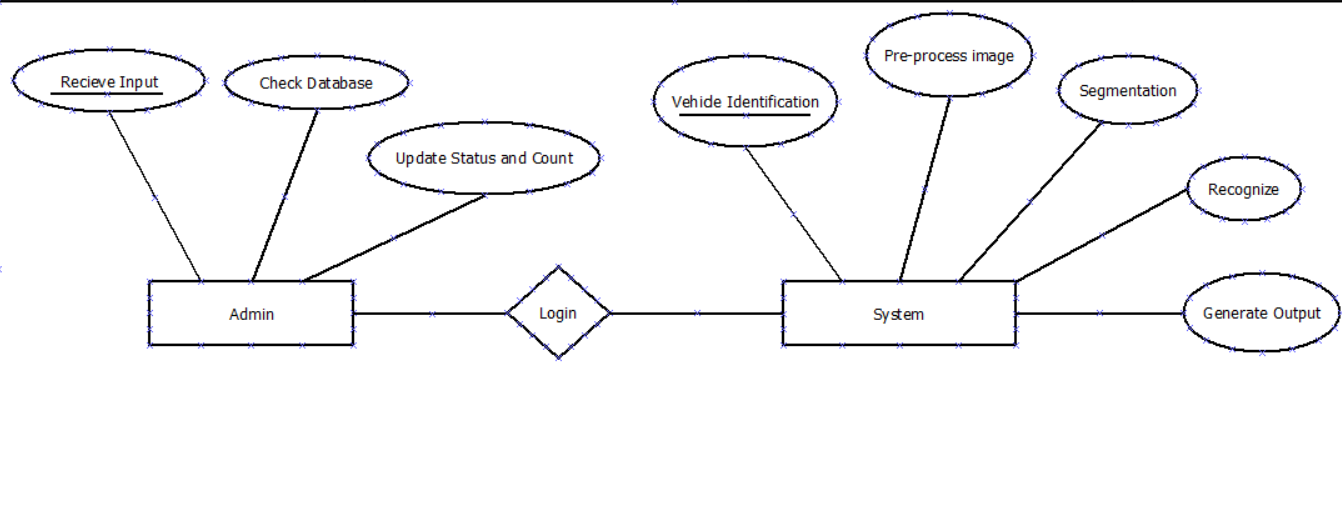
****

Fig 4.3

**4.4 UML Diagrams**

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML. The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business mode and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

**GOALS**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.

**4.4.1 USE CASE DIAGRAM**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

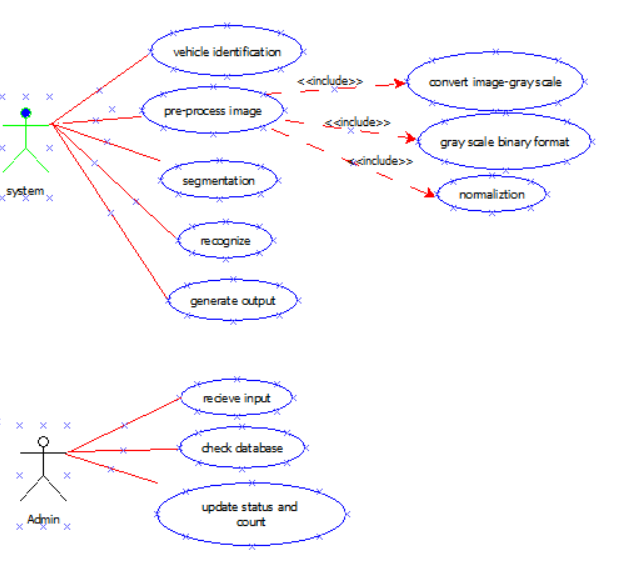


Figure 4.4.1

**4.4.2 CLASS DIAGRAM**

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

The class shape itself consists of a rectangle with three rows. The top row contains the name of the class, the middle row contains the attributes of the class, and the bottom section expresses the methods or operations that the class may use. Classes and subclasses are grouped together to show the static relationship between each object.

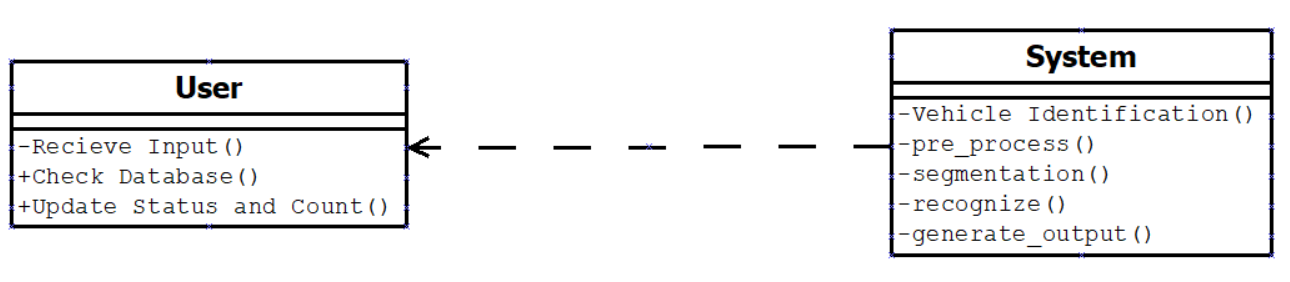


Fig 4.4.2

**4.4.3 ACTIVITY DIAGRAM**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques.

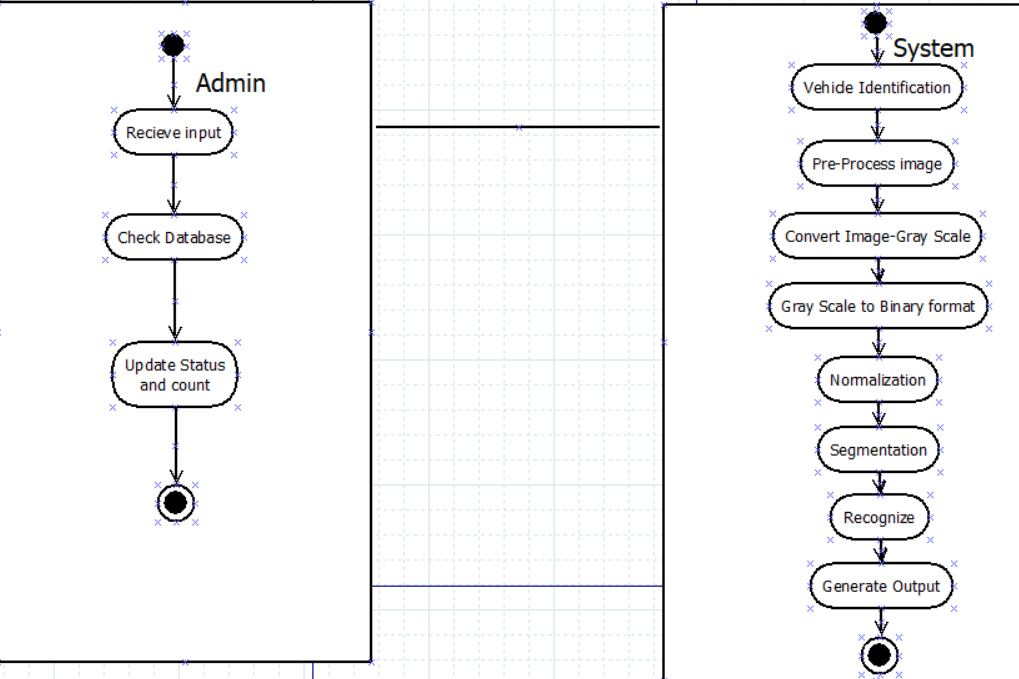


Fig 4.4.3

**4.4.4 SEQUENCE DIAGRAM**

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

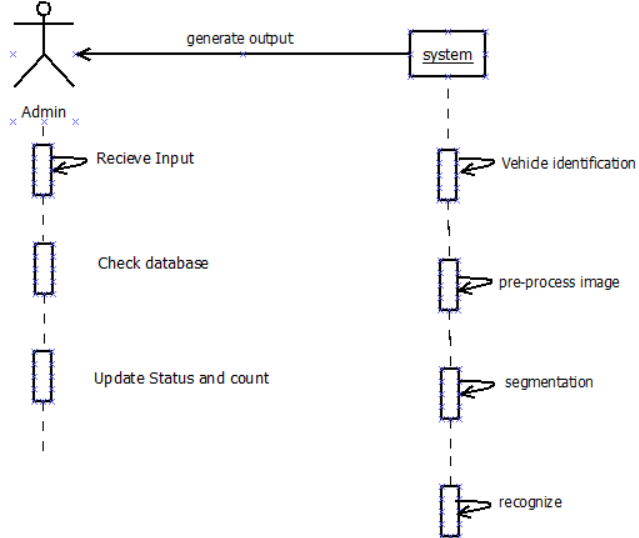


Fig 4.4.4

**4.4.5 COLLABORATION DIAGRAM**

A collaboration diagram resembles a flow chart that portrays the roles, functionality and behavior of individual objects as well as the overall operation of the system in real time. The relationships between the objects are shown as lines connecting the rectangles. The messages between objects are shown as arrows connecting the relevant rectangles along with labels that define the message sequencing.

A collaboration diagram resembles a [flowchart](http://whatis.techtarget.com/definition/flowchart) that portrays the roles, functionality and behavior of individual objects as well as the overall operation of the system in [real time](http://searchcio-midmarket.techtarget.com/definition/real-time). Objects are shown as rectangles with naming labels inside. These labels are preceded by colons and may be underlined. The relationships between the objects are shown as lines connecting the rectangles.

****

Fig 4.4.5

**5.CODING**

**5.1 Home Page (Index.html)**

**<!doctype html>**

**<html lang="en">**

**<head>**

**<meta charset="utf-8">**

**<title>number plate detection</title>**

**<meta http-equiv="x-ua-compatible" content="ie=edge">**

**<meta name="viewport" content="width=device-width, initial-scale=1">**

**<meta name="keywords" content="">**

**<meta name="description" content="">**

**<link rel="stylesheet" href="css/bootstrap.min.css">**

**<link rel="stylesheet" href="css/font-awesome.min.css">**

**<link rel="stylesheet" href="css/animate.min.css">**

**<link rel="stylesheet" href="css/et-line-font.css">**

**<link rel="stylesheet" href="css/nivo-lightbox.css">**

**<link rel="stylesheet" href="css/nivo\_themes/default/default.css">**

**<link rel="stylesheet" href="css/style.css">**

**<link href='https://fonts.googleapis.com/css?family=roboto:400,300,500' rel='stylesheet' type='text/css'>**

**</head>**

**<body data-spy="scroll" data-target=".navbar-collapse" data-offset="50">**

**<!-- preloader section -->**

**<div class="preloader">**

**<div class="sk-spinner sk-spinner-circle">**

**<div class="sk-circle1 sk-circle"></div>**

**<div class="sk-circle2 sk-circle"></div>**

**<div class="sk-circle3 sk-circle"></div>**

**<div class="sk-circle4 sk-circle"></div>**

**<div class="sk-circle5 sk-circle"></div>**

**<div class="sk-circle6 sk-circle"></div>**

**<div class="sk-circle7 sk-circle"></div>**

**<div class="sk-circle8 sk-circle"></div>**

**<div class="sk-circle9 sk-circle"></div>**

**<div class="sk-circle10 sk-circle"></div>**

**<div class="sk-circle11 sk-circle"></div>**

**<div class="sk-circle12 sk-circle"></div>**

**</div>**

**</div>**

**<!-- navigation section -->**

**<section class="navbar navbar-fixed-top custom-navbar" role="navigation">**

**<div class="container">**

**<div class="navbar-header">**

**<button class="navbar-toggle" data-toggle="collapse" data- target=".navbar-collapse">**

**<span class="icon icon-bar"></span>**

**<span class="icon icon-bar"></span>**

**<span class="icon icon-bar"></span>**

**</button>**

**</div>**

**<div class="collapse navbar-collapse">**

**<ul class="nav navbar-nav navbar-right">**

**<li><a href="#home" class="smoothscroll">home</a></li>**

**<li><ahref="signin/login/about.html"class="smoothscroll">about< /a></li>**

**<li><a href="sign-in/index.html" ="smoothscroll">login</a></li>**

**<li><a href="#contact" class="smoothscroll">contact</a></li>**

**</ul>**

**</div>**

**</div>**

**</section>**

**<!-- home section -->**

**<section id="home">**

**<div class="container">**

**<div class="row">**

**<div class="col-md-12 col-sm-12">**

**<h3>nighttime/ bio-inspired/weighted score-level</h3>**

**<h1>vehicle detection</h1>**

**<hr>**

**</div>**

**</div>**

**</div>**

**</section>**

**<!-- about section -->**

**<section id="about">**

**<div class="container">**

**<div class="row">**

**<div class="col-md-12 col-sm-12 text-center">**

**<div class="section-title">**

**<strong>about</strong>**

**<h1 class="heading bold">main idea</h1>**

**<hr>**

**</div>**

**</div>**

**<div class="col-md-6 col-sm-12">**

**<img src="images/about-img.jpg" class="img-responsive" alt="about img">**

**</div>**

**<div class="col-md-6 col-sm-12">**

**<h3 class="bold">vehicle detection</h3>**

**<h1 class="heading bold">nighttime vehicle detection based on bio-inspired image enhancement and weighted score-level feature fusion**

**</h1>**

**<!-- nav tabs -->**

**<ul class="nav nav-tabs" role="tablist">**

**<li class="active"><a href="#design" aria-controls="design" role="tab" data-toggle="tab">detection</a></li>**

**<li><a href="#mobile" aria-controls="mobile" role="tab" data-toggle="tab">background process</a></li>**

**<li><a href="#social" aria-controls="social" role="tab" data-toggle="tab">result</a></li>**

**</ul>**

**<!-- tab panes -->**

**<div class="tab-content">**

**<div role="tabpanel" class="tab-pane active" id="design">**

**<p> Night-time vehicle detection system that combines a novel bioinspired image enhancement approach with a weighted feature fusion technique. inspired by the retinal mechanism in natural visual processing, we develop a nighttime image enhancement method by modeling the adaptive feedback from horizontal cells and the center-surround antagonistic receptive fields of bipolar cells.</p>**

**</div>**

**<div role="tabpanel" class="tab-pane" id="mobile">**

**<p> histogram of oriented gradient, and local binary pattern to train the classifiers with support vector machine. these features are fused by combining the score vectors of each feature with the learnt weights. during detection, we generate accurate regions of interest by combining vehicle tail-light detection with object proposals.</p>**

**</div>**

**<div role="tabpanel" class="tab-pane" id="social">**

**<p>our vehicle detection method demonstrates a 95.95% detection rate at 0.0575 false positives per image and outperforms some state-of-the-art techniques. our proposed method can deal with various scenes including vehicles of different types and sizes and those with occlusions and in blurred zones.</p>**

**</div> </div> </div> </div> </div> </section>**

**<script src="js/jquery.js"></script>**

**<script src="js/bootstrap.min.js"></script>**

**<script src="js/smoothscroll.js"></script>**

**<script src="js/isotope.js"></script>**

**<script src="js/imagesloaded.min.js"></script>**

**<script src="js/nivo-lightbox.min.js"></script>**

**<script src="js/jquery.backstretch.min.js"></script>**

**<script src="js/wow.min.js"></script>**

**<script src="js/custom.js"></script>**

**</body>**

**</html>**

**5.2 LOGIN PAGE**

**<!DOCTYPE html>**

**<html xmlns="http://www.w3.org/1999/xhtml">**

**<head>**

**<meta charset="utf-8" />**

**<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />**

**<meta name="description" content="" />**

**<meta name="author" content="" />**

**<!--[if IE]>**

**<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">**

**<![endif]-->**

**<title> VEHICLE DETECTION </title>**

**<!-- BOOTSTRAP CORE STYLE CSS -->**

**<link href="assets/css/bootstrap.css" rel="stylesheet" />**

**<!-- FONTAWESOME STYLE CSS -->**

**<link href="assets/css/font-awesome.css" rel="stylesheet" />**

**<!-- CUSTOM STYLE CSS -->**

**<link href="assets/css/style.css" rel="stylesheet" />**

**</head> <body >**

**<div class="navbar navbar-inverse navbar-fixed-top " >**

**<div class="container">**

**<div class="navbar-header">**

**<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">**

**<span class="icon-bar"></span>**

**<span class="icon-bar"></span>**

**<span class="icon-bar"></span>**

**</button>**

**<a class="navbar-brand" href="#" ><strong style=""></strong> Vehicle <small>Detection</small></a> </div>**

**<div class="navbar-collapse collapse move-me">**

**<ul class="nav navbar-nav navbar-right set-links">**

**<li><a href="about.html">ABOUT</a></li>**

**<li><a href="search\_image.html">SEARCH IMAGE</a></li>**

**<li><a href="contact.html" class="active-menu-item">CHECK STATUS</a></li>**

**<li><a href="pricing.html">CHECK USER</a></li>**

**<li><a href="logout.php">LOG OUT</a></li>**

**</ul> </div> </div> </div>**

**<!--MENU SECTION END-->**

**<section id="home-sec">**

**<div class="overlay text-center">**

**<h1 >Detection of vehicle based on image</h1>**

**<hr class="hr-set"/>**

**<p>Vehicle Detection Based on Bio-Inspired Image Enhancement </p>**

**</div> </section>**

**<!--HOME SECTION END-->**

**<section id="search-domain" >**

**<div class="container">**

**<div class="row">**

**<div class="col-md-8">**

**<input type="text" class="form-control input-cls" />**

**</div>**

**<div class="col-md-4">**

**<input type="button" class="btn btn-info btn-lg btn-set" value="SEARCH" />**

**</div> </div> </div> </section>**

**<!--SEARCH SECTION END-->**

**<!-- JAVASCRIPT FILES PLACED AT THE BOTTOM TO REDUCE THE LOADING TIME -->**

**<!-- CORE JQUERY -->**

**<script src="assets/js/jquery-1.11.1.js"></script>**

**<!-- BOOTSTRAP SCRIPTS -->**

**<script src="assets/js/bootstrap.js"></script>**

**<!-- CUSTOM SCRIPTS -->**

**<script src="assets/js/custom.js"></script> </body> </html>**

**5.3 ABOUT THE PROJECT PAGE**

**<!DOCTYPE html>**

**<html xmlns="http://www.w3.org/1999/xhtml">**

**<head>**

**<meta charset="utf-8" />**

**<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />**

**<meta name="description" content="" />**

**<meta name="author" content="" />**

**<!--[if IE]>**

**<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">**

**<![endif]-->**

**<title>ABOUT</title>**

**<!-- BOOTSTRAP CORE STYLE CSS -->**

**<link href="assets/css/bootstrap.css" rel="stylesheet" />**

**<!-- FONTAWESOME STYLE CSS -->**

**<link href="assets/css/font-awesome.css" rel="stylesheet" />**

**<!-- CUSTOM STYLE CSS -->**

**<link href="assets/css/style.css" rel="stylesheet" />**

**</head>**

**<body >**

**<!--MENU SECTION END-->**

**<section class="headline-sec">**

**<div class="overlay ">**

**<h3 >ABOUT PROJECT <i class="fa fa-angle-double-right "></i></h3>**

**</div>**

**</section>**

**<!--HOME SECTION END-->**

**<section>**

**<div class="container">**

**<div class="row">**

**<div class="col-md-8" >**

**<h3><strong>Overview</strong></h3>**

**<p>**

**This paper presents an effective night time vehicle detection system that combines a novel bio-inspired image enhancement approach with a weighted feature fusion technique.**

**During detection, we generate accurate regions of interest by combining vehicle tail-light detection.**

**</p>**

**</div>**

**<div class="col-md-4 p-top-row " >**

**<img src="assets/img/about1.jpg" class="img-responsive img-rounded" alt="" />**

**</div>**

**</div>**

**<div class="row p-top-row">**

**<div class="col-md-8" >**

**<h3><strong>Future Scope</strong></h3>**

**<p>**

**The user who frequently disobey the rules will have to get his/her license rejected as a sign of punishment.**

**</p>**

**</div>**

**<div class="col-md-4 p-top-row " >**

**<img src="assets/img/about3.jpg" class="img-responsive img-rounded" alt="" />**

**</div>**

**</div>**

**</div>**

**</section>**

**<section id="footer-sec" >**

**<div class="container">**

**<div class="row">**

**</div>**

**</div>**

**</div>**

**<!-- COPY TEXT SECTION END-->**

**<!-- JAVASCRIPT FILES PLACED AT THE BOTTOM TO REDUCE THE LOADING TIME -->**

**<!-- CORE JQUERY -->**

**<script src="assets/js/jquery-1.11.1.js"></script>**

**<!-- BOOTSTRAP SCRIPTS -->**

**<script src="assets/js/bootstrap.js"></script>**

**<!-- CUSTOM SCRIPTS -->**

**<script src="assets/js/custom.js"></script>**

**</body>**

**</html>**

**5.4 CHECK USER**

**<!DOCTYPE html>**

**<html xmlns="http://www.w3.org/1999/xhtml">**

**<head>**

**<meta charset="utf-8" />**

**<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />**

**<meta name="description" content="" />**

**<meta name="author" content="" />**

**<!--[if IE]>**

**<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">**

**<![endif]-->**

**<title>IT & HOSTING TEMPLATE</title>**

**<!-- BOOTSTRAP CORE STYLE CSS -->**

**<link href="assets/css/bootstrap.css" rel="stylesheet" />**

**<!-- FONTAWESOME STYLE CSS -->**

**<link href="assets/css/font-awesome.css" rel="stylesheet" />**

**<!-- CUSTOM STYLE CSS -->**

**<link href="assets/css/style.css" rel="stylesheet" />**

**</head>**

**<style>**

**#customers {**

**font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;**

**border-collapse: collapse;**

**width: 100%;**

**}**

**#customers td, #customers th {**

**border: 3px solid #ddd;**

**padding: 8px;**

**}**

**#customers tr:nth-child(even){background-color: #f2f2f2;}**

**#customers tr:hover {background-color: #ddd;}**

**#customers th {**

**padding-top: 12px;**

**padding-bottom: 12px;**

**text-align: left;**

**background-color: #4CAF50;**

**color: white;**

**}**

**</style>**

**<body >**

**<div class="navbar navbar-inverse navbar-fixed-top " >**

**<div class="container">**

**<div class="navbar-header">**

**<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">**

**<span class="icon-bar"></span>**

**<span class="icon-bar"></span>**

**<span class="icon-bar"></span>**

**</button>**

**<a class="navbar-brand" href="#" ><strong style=""></strong> STATUS<small >user status</small></a>**

**</div>**

**</div>**

**</div>**

**<!--MENU SECTION END-->**

**<section class="headline-sec">**

**<div class="overlay ">**

**<h3 >BLANK PAGE <i class="fa fa-angle-double-right "></i></h3>**

**</div>**

**</section>**

**<!--HOME SECTION END-->**

**<section>**

**<div class="container">**

**<div class="row">**

**<div class="col-md-6" >**

**<h2><strong> <i>VEHICLE NUMBER</i> </strong></h2>**

**<br />**

**<img src="plate.jpg"></img>**

**<br />**

**<h4><i>Email: </i>info@domain.com</h4>**

**<h4><i>Call:</i> +1-908-78-55-5555</h4>**

**</div>**

**<div id="check\_status">**

**<fieldset style="width:30%"><legend><b>CHECK STATUS</b></legend>**

**<table border="0" id="customers">**

**<tr>**

**<form action="check\_status.php" method="POST" >**

**<td><b>Enter vehicle number:</b></td><td> <input type="text" name="number\_plate" required="required"></td>**

**</tr>**

**<tr>**

**<td><input id="button" type="submit" name="submit" value="Check"></td>**

**</tr>**

**</form>**

**</table>**

**</fieldset>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**</section>**

**<!--FOOTER SECTION END-->**

**<div class="copy-txt">**

**<div class="container">**

**<div class="row">**

**<div class="col-md-12 set-foot" >**

**&copy 2014 your domain | All rights reserved | Design by : <a href="http://www.binarytheme.com"target="\_blank" style="color:#7C7C7C;">binarytheme.com**

**</a>**

**</div>**

**</div>**

**</div>**

**</div>**

**<!-- COPY TEXT SECTION END-->**

**<!-- JAVASCRIPT FILES PLACED AT THE BOTTOM TO REDUCE THE LOADING TIME -->**

**<!-- CORE JQUERY -->**

**<script src="assets/js/jquery-1.11.1.js">**

**</script>**

**<!-- BOOTSTRAP SCRIPTS -->**

**<script src="assets/js/bootstrap.js">**

**</script>**

**<!-- CUSTOM SCRIPTS -->**

**<script src="assets/js/custom.js"></script>**

**</body>**

**</html>**

**5.5 CHECK STATUS**

**<!DOCTYPE html>**

**<html xmlns="http://www.w3.org/1999/xhtml">**

**<head>**

**<meta charset="utf-8" />**

**<meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />**

**<meta name="description" content="" />**

**<meta name="author" content="" />**

**<!--[if IE]>**

**<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">**

**<![endif]-->**

**<title>IT & HOSTING TEMPLATE</title>**

**<!-- BOOTSTRAP CORE STYLE CSS -->**

**<link href="assets/css/bootstrap.css" rel="stylesheet" />**

**<!-- FONTAWESOME STYLE CSS -->**

**<link href="assets/css/font-awesome.css" rel="stylesheet" />**

**<!-- CUSTOM STYLE CSS -->**

**<link href="assets/css/style.css" rel="stylesheet" />**

**</head>**

**<style>**

**#customers {**

**font-family: "Trebuchet MS", Arial, Helvetica, sans-serif;**

**border-collapse: collapse;**

**width: 100%;**

**}**

**+++#customers td, #customers th {**

**border: 3px solid #ddd;**

**padding: 8px;**

**}**

**#customers tr:nth-child(even){background-color: #f2f2f2;}**

**#customers tr:hover {background-color: #ddd;}**

**#customers th {**

**padding-top: 12px;**

**padding-bottom: 12px;**

**text-align: left;**

**background-color: #4CAF50;**

**color: white;**

**}**

**</style>**

**<body >**

**<div class="navbar navbar-inverse navbar-fixed-top " >**

**<div class="container">**

**<div class="navbar-header">**

**<button type="button" class="navbar-toggle" data-toggle="collapse" data-target=".navbar-collapse">**

**<span class="icon-bar"></span>**

**<span class="icon-bar"></span>**

**<span class="icon-bar"></span>**

**</button>**

**<a class="navbar-brand" href="#" ><strong style=""></strong> STATUS<small >user status</small></a>**

**</div>**

**</div>**

**</div>**

**<!--MENU SECTION END-->**

**<section class="headline-sec">**

**<div class="overlay ">**

**<h3 >BLANK PAGE <i class="fa fa-angle-double-right "></i></h3>**

**</div>**

**</section>**

**<!--HOME SECTION END-->**

**<section>**

**<div class="container">**

**<div class="row">**

**<div class="col-md-6" >**

**<h2><strong> <i>VEHICLE NUMBER</i> </strong></h2>**

**<br />**

**<img src="plate.jpg"></img>**

**<br />**

**<h4><i>Email: </i>info@domain.com</h4>**

**<h4><i>Call:</i> +1-908-78-55-5555</h4>**

**</div>**

**<div id="check\_status">**

**<fieldset style="width:30%"><legend><b>SEARCH USER</b></legend>**

**<table border="0" id="customers">**

**<tr>**

**<form action="search\_database.php" method="POST" >**

**<td><b>Enter vehicle number:</b></td><td> <input type="text" name="number\_plate" required="required"></td>**

**</tr>**

**<tr>**

**<td><input id="button" type="submit" name="submit" value="Search"></td>**

**</tr>**

**</form>**

**</table>**

**</fieldset>**

**</div>**

**</div>**

**</div>**

**</div>**

**</div>**

**</section>**

**<!--FOOTER SECTION END-->**

**<div class="copy-txt">**

**<div class="container">**

**<div class="row">**

**<div class="col-md-12 set-foot" >**

**&copy 2014 your domain | All rights reserved | Design by : <a href="http://www.binarytheme.com" target="\_blank" style="color:#7C7C7C;">binarytheme.com</a>**

**</div>**

**</div>**

**</div>**

**</div>**

**<!-- COPY TEXT SECTION END-->**

**<!-- JAVASCRIPT FILES PLACED AT THE BOTTOM TO REDUCE THE LOADING TIME -->**

**<!-- CORE JQUERY -->**

**<script src="assets/js/jquery-1.11.1.js"></script>**

**<!-- BOOTSTRAP SCRIPTS -->**

**<script src="assets/js/bootstrap.js"></script>**

**<!-- CUSTOM SCRIPTS -->**

**<script src="assets/js/custom.js"></script>**

**</body>**

**</html>**

**5.6 PRE-PROCESSING IMAGE**

**import cv2**

**# Importing the Opencv Library**

**import numpy as np**

**# Importing NumPy,which is the fundamental package for scientific computing with Python**

**# Reading Image**

**img = cv2.imread("71.jpg")**

**cv2.namedWindow("Original Image",cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**cv2.imshow("Original Image",img)**

**# Display image**

**# RGB to Gray scale conversion**

**img\_gray = cv2.cvtColor(img,cv2.COLOR\_RGB2GRAY)**

**cv2.namedWindow("Gray Converted Image",cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**cv2.imshow("Gray Converted Image",img\_gray)**

**# Display Image**

**# Noise removal with iterative bilateral filter(removes noise while preserving edges)**

**#noise\_removal = cv2.GaussianBlur(img\_gray,(5,5),0)**

**noise\_removal = cv2.bilateralFilter(img\_gray,9,160,160)**

**cv2.namedWindow("Noise Removed Image",cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**cv2.imshow("Noise Removed Image",noise\_removal)**

**# Display Image**

**# Histogram equalisation for better results**

**clache=cv2.createCLAHE(clipLimit=2.0,tileGridSize=(8,8))**

**equal\_histogram = clache.apply(noise\_removal)**

**#equal\_histogram = cv2.equalizeHist(noise\_removal)**

**#equal\_histogram = np.hstack((noise\_removal,equ))**

**cv2.namedWindow("After Histogram equalisation",cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**cv2.imshow("After Histogram equalisation",equal\_histogram)**

**# Display Image**

**# Morphological opening with a rectangular structure element**

**kernel = cv2.getStructuringElement(cv2.MORPH\_RECT,(5,5))**

**#morph\_image = cv2.morphologyEx(equal\_histogram,cv2.MORPH\_OPEN,kernel,iterations=15)**

**morph\_image = cv2.morphologyEx(equal\_histogram, cv2.MORPH\_OPEN, kernel)**

**#cv2.namedWindow("Morphological opening",cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**#cv2.imshow("Morphological opening",morph\_image)**

**# Display Image**

**# Image subtraction(Subtracting the Morphed image from the histogram equalised Image)**

**sub\_morp\_image = cv2.subtract(equal\_histogram,morph\_image)**

**#cv2.namedWindow("Subtraction image", cv2.WINDOW\_NORMAL)0**

**# Creating a Named window to display image**

**#cv2.imshow("Subtraction image", sub\_morp\_image)**

**# Display Image**

**# Thresholding the image**

**ret,thresh\_image = cv2.threshold(sub\_morp\_image,0,255,cv2.THRESH\_OTSU)**

**#cv2.namedWindow("Image after Thresholding",cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**#cv2.imshow("Image after Thresholding",thresh\_image)**

**# Display Image**

**# Applying Canny Edge detection**

**canny\_image = cv2.Canny(thresh\_image,100,200)**

**#cv2.namedWindow("Image after applying Canny",cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**#cv2.imshow("Image after applying Canny",canny\_image)**

**# Display Image**

**canny\_image = cv2.convertScaleAbs(canny\_image)**

**# dilation to strengthen the edges**

**kernel = np.ones((3,3), np.uint8)**

**# Creating the kernel for dilation**

**dilated\_image = cv2.dilate(canny\_image,kernel,iterations=1)**

**#cv2.namedWindow("Dilation", cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**#cv2.imshow("Dilation", dilated\_image)**

**# Displaying Image**

**# Finding Contours in the image based on edges**

**new,contours, hierarchy = cv2.findContours(dilated\_image, cv2.RETR\_TREE, cv2.CHAIN\_APPROX\_SIMPLE)**

**contours= sorted(contours, key = cv2.contourArea, reverse = True)[:10]**

**# Sort the contours based on area ,so that the number plate will be in top 10 contours**

**screenCnt = None**

**# loop over our contours**

**for c in contours:**

**# approximate the contour**

**peri = cv2.arcLength(c, True)**

**approx = cv2.approxPolyDP(c, 0.06 \* peri, True) # Approximating with 6% error**

**# if our approximated contour has four points, then**

**# we can assume that we have found our screen**

**if len(approx) == 4: # Select the contour with 4 corners**

**screenCnt = approx**

**break**

**final = cv2.drawContours(img, [screenCnt], -1, (0, 255, 0), 3)**

**# Drawing the selected contour on the original image**

**#cv2.namedWindow("Image with Selected Contour",cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**#cv2.imshow("Image with Selected Contour",final)**

**# Masking the part other than the number plate**

**#height,width = img\_gray.shape**

**#x=int(width/2)**

**#y=int(height/2)**

**#cv2.rectangle(img\_gray,(x-180,y-180),(x+180,y+180),(0,0,255),2)**

**#mask = np.zeros(img\_gray.shape,np.uint8)**

**#cv2.circle(mask,(x,y),280,1,thickness=-1)**

**#new\_image = cv2.drawContours(mask,[screenCnt],0,255,-1,)**

**#new\_image = cv2.bitwise\_and(img,img,mask=mask)**

**r = cv2.selectROI(final)**

**new\_image= final[int(r[1]):int(r[1]+r[3]), int(r[0]):int(r[1]+r[2])]**

**cv2.namedWindow("Final\_image",cv2.WINDOW\_NORMAL)**

**cv2.imshow("Final\_image",new\_image)**

**# Histogram equal for enhancing the number plate for further processing**

**y,cr,cb = cv2.split(cv2.cvtColor(new\_image,cv2.COLOR\_RGB2YCrCb))**

**# Converting the image to YCrCb model and splitting the 3 channels**

**y = cv2.equalizeHist(y)**

**# Applying histogram equalisation**

**final\_image = cv2.cvtColor(cv2.merge([y,cr,cb]),cv2.COLOR\_YCrCb2RGB)**

**# Merging the 3 channels**

**cv2.namedWindow("Enhanced Number Plate",cv2.WINDOW\_NORMAL)**

**# Creating a Named window to display image**

**cv2.imshow("Enhanced Number Plate",final\_image)**

**# Display image**

**cv2.waitKey() # Wait for a keystroke from the user**

**5.7 ADMIN TABLE**

**Table 1**

|  |
| --- |
| ****username** **password**** |
| **admin1 Admin$123** |
| **administrator admin456** |
| **navya navya123** |
| **prasanthi prasanthi123** |
| **prashu prashu123** |
| **satyacomp abcd** |

**6.8 USER TABLE**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ****Number\_plate**** | ****User\_name**** | ****Phone\_number**** | ****Door\_no**** | ****street**** | ****city**** | ****pincode**** | ****count**** | ****status**** |
| ****AP123J567**** | **Ram** | **987456321** | **22-01-97** | **kotharoad** | **visakhapatnam** | **530001** | **0** | **Rejected** |
| ****AP187Y65**** | **Arun** | **962357410** | **56-08-64** | **poornamarket** | **visakhapatnam** | **530001** | **0** | **Rejected** |

**Table 2**

**6. TESTING**

### 

### 6.1 TESTING METHODOLOGIES

The following are the Testing Methodologies:

* Unit Testing.
* Integration Testing.
* User Acceptance Testing.
* Output Testing.
* Validation Testing.

**6.1.1 Unit Testing**

Unit testing focuses verification effort on the smallest unit of Software design that is the module. Unit testing exercises specific paths in a module’s control structure to ensure complete coverage and maximum error detection. This test focuses on each module individually, ensuring that it functions properly as a unit. Hence, the naming is Unit Testing. During this testing, each module is tested individually and the module interfaces are verified for the consistency with design specification. All important processing path are tested for the expected results. All error handling paths are also tested.

**6.1.2 Integration Testing**

Integration testing addresses the issues associated with the dual problems of verification and program construction. After the software has been integrated a set of high order tests are conducted. The main objective in this testing process is to take unit tested modules and builds a program structure that has been dictated by design.

**The following are the types of Integration Testing:**

1. **Top Down Integration**

This method is an incremental approach to the construction of program structure. Modules are integrated by moving downward through the control hierarchy, beginning with the main program module. The module subordinates to the main program module are incorporated

Into the structure in either a depth first or breadth first manner. In this method, the software is tested from main module and individual stubs are replaced when the test proceeds downwards.

**2. Bottom-up Integration**

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

* The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.
* A driver (i.e.) the control program for testing is written to coordinate test case input and output.
* The cluster is tested.
* Drivers are removed and clusters are combined moving upward in the program structure

The bottom up approaches tests each module individually and then each module is module is integrated with a main module and tested for functionality.

**6.1.3 User Acceptance Testing**

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

[User acceptance testing](https://bugwolf.com/blog/user-acceptance-testing-not-only-reduces-costs-but-also-captures-indispensable-feedback) is also important because failure to do it places an unnecessary burden on system developers who, while they may be experts in development, are not familiar with the vagaries of running an organisation or the day to day difficulties that the software they are developing must account for.

**6.1.4 Output Testing**

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

**6.1.5 Validation Checking**

Validation checks are performed on the following fields.

**Text Field:**

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

**Numeric Field:**

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error messages. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

A successful test is one that gives out the defects for the inappropriate data and produces and output revealing the errors in the system.

**Preparation of Test Data**

Taking various kinds of test data does the above testing. Preparation of test data plays a vital role in the system testing. After preparing the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

**Using Live Test Data:**

Live test data are those that are actually extracted from organization files. After a system is partially constructed, programmers or analysts often ask users to key in a set of data from their normal activities. Then, the systems person uses this data as a way to partially test the system. In other instances, programmers or analysts extract a set of live data from the files and have them entered themselves.

It is difficult to obtain live data in sufficient amounts to conduct extensive testing. And, although it is realistic data that will show how the system will perform for the typical processing requirement, assuming that the live data entered are in fact typical, such data generally will not test all combinations or formats that can enter the system. This bias toward typical values then does not provide a true systems test and in fact ignores the cases most likely to cause system failure.

**Using Artificial Test Data:**

Artificial test data are created solely for test purposes, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information systems department, make possible the testing of all login and control paths through the program.

The most effective test programs use artificial test data generated by persons other than those who wrote the programs. Often, an independent team of testers formulates a testing plan, using the systems specifications.

The package “Virtual Private Network” has satisfied all the requirements specified as per software requirement specification and was accepted.

**6.1.6 TESTING STRATEGY :**

A strategy for system testing integrates system test cases and design techniques into a well planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

**SYSTEM TESTING:**

Software once validated must be combined with other system elements (e.g. Hardware, people, database). System testing verifies that all the elements are proper and that overall system

function performance is achieved. It also tests to find discrepancies between the system and its original objective, current specifications and system documentation.

**UNIT TESTING:**

In unit testing different are modules are tested against the specifications produced during the design for the modules. Unit testing is essential for verification of the code produced during the coding phase, and hence the goals to test the internal logic of the modules. Using the detailed design description as a guide, important Conrail paths are tested to uncover errors within the boundary of the modules. This testing is carried out during the programming stage itself. In this type of testing step, each module was found to be working satisfactorily as regards to the expected output from the module.

In Due Course, latest technology advancements will be taken into consideration. As part of technical build-up many components of the networking system will be generic in nature so that future projects can either use or interact with this.The future holds a lot to offer to the development and refinement of this project.

**6.2TEST CASES FOR UNIT TESTING AND VALIDATION TESTING**

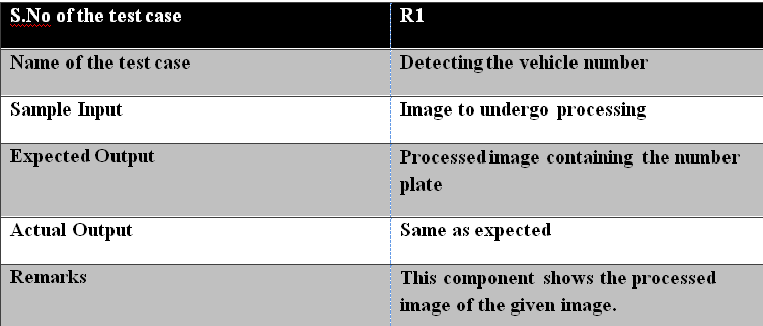


Table 3

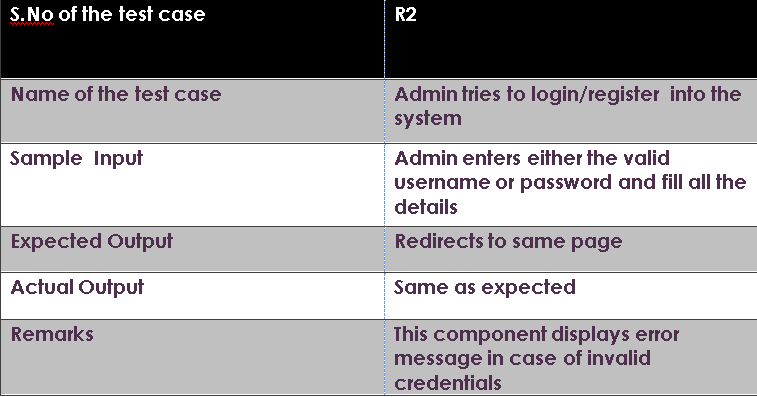


Table 4



Table 5

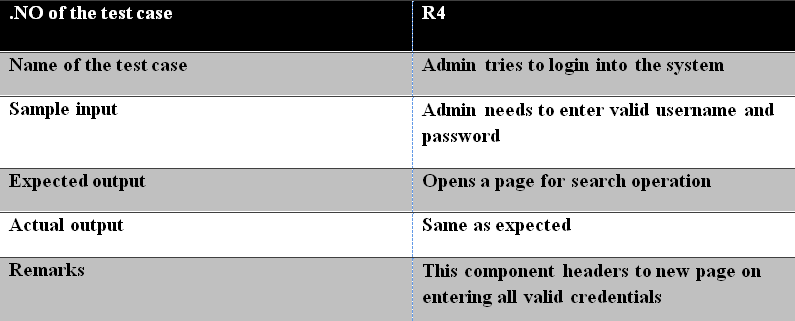


Table 6

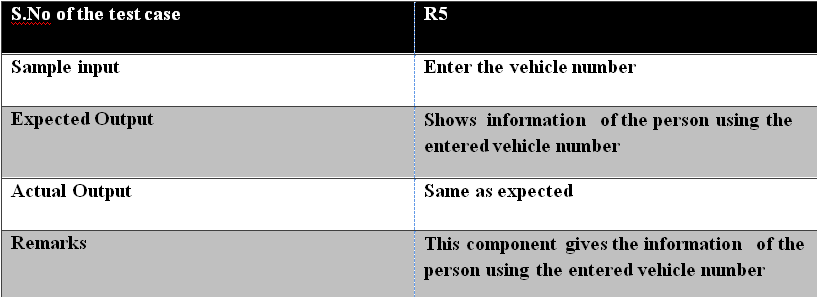


Table 7

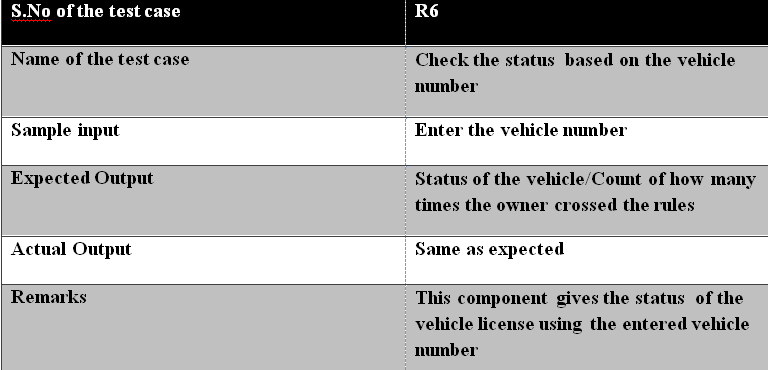


Table 8

**7.OUTPUT SCREENS**

**7.1 Sample Screen for Main Page**

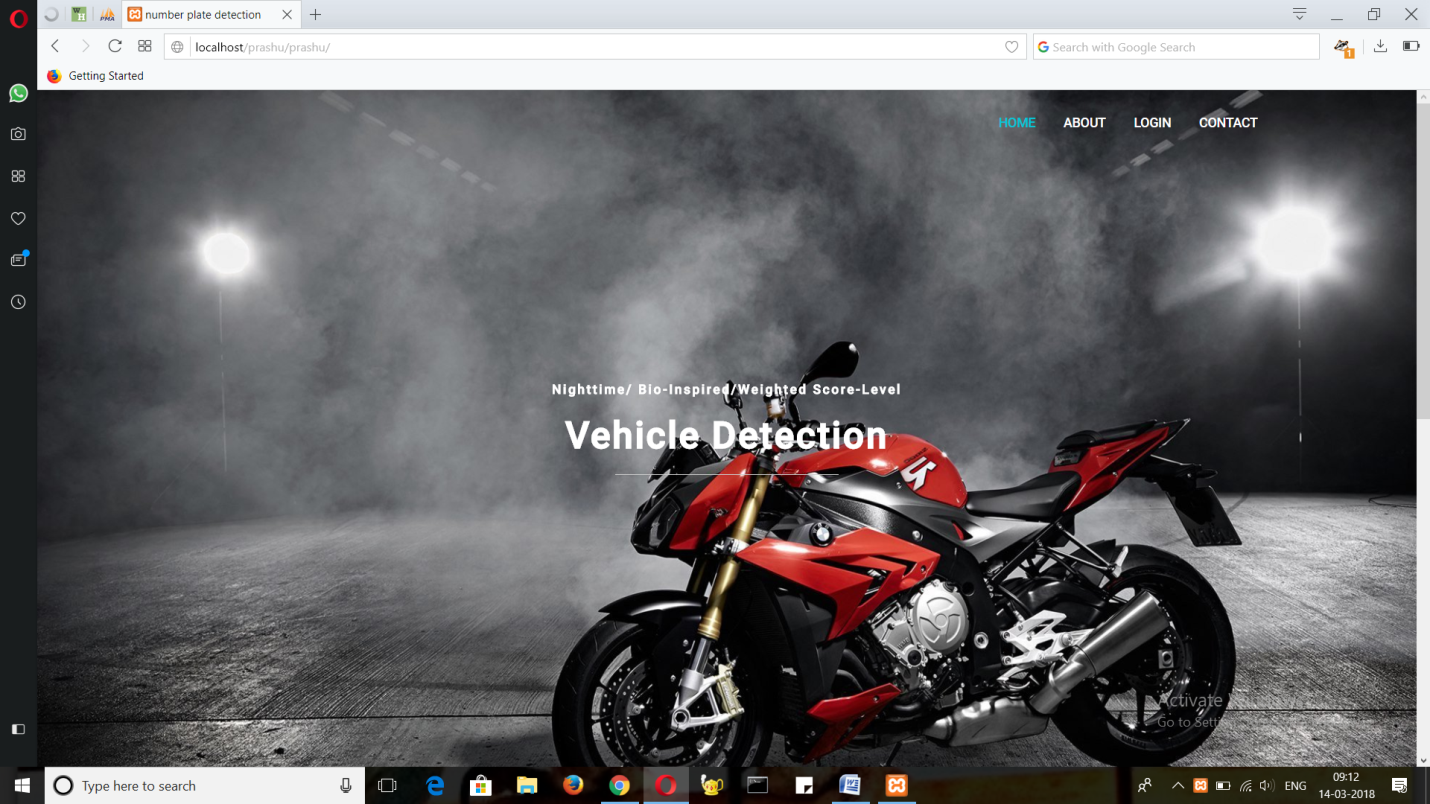
****

Fig 7.1

**Explanation**

In the above window we can clearly find out that main page contains two main modules like about and login module.

**7.2 About Page**

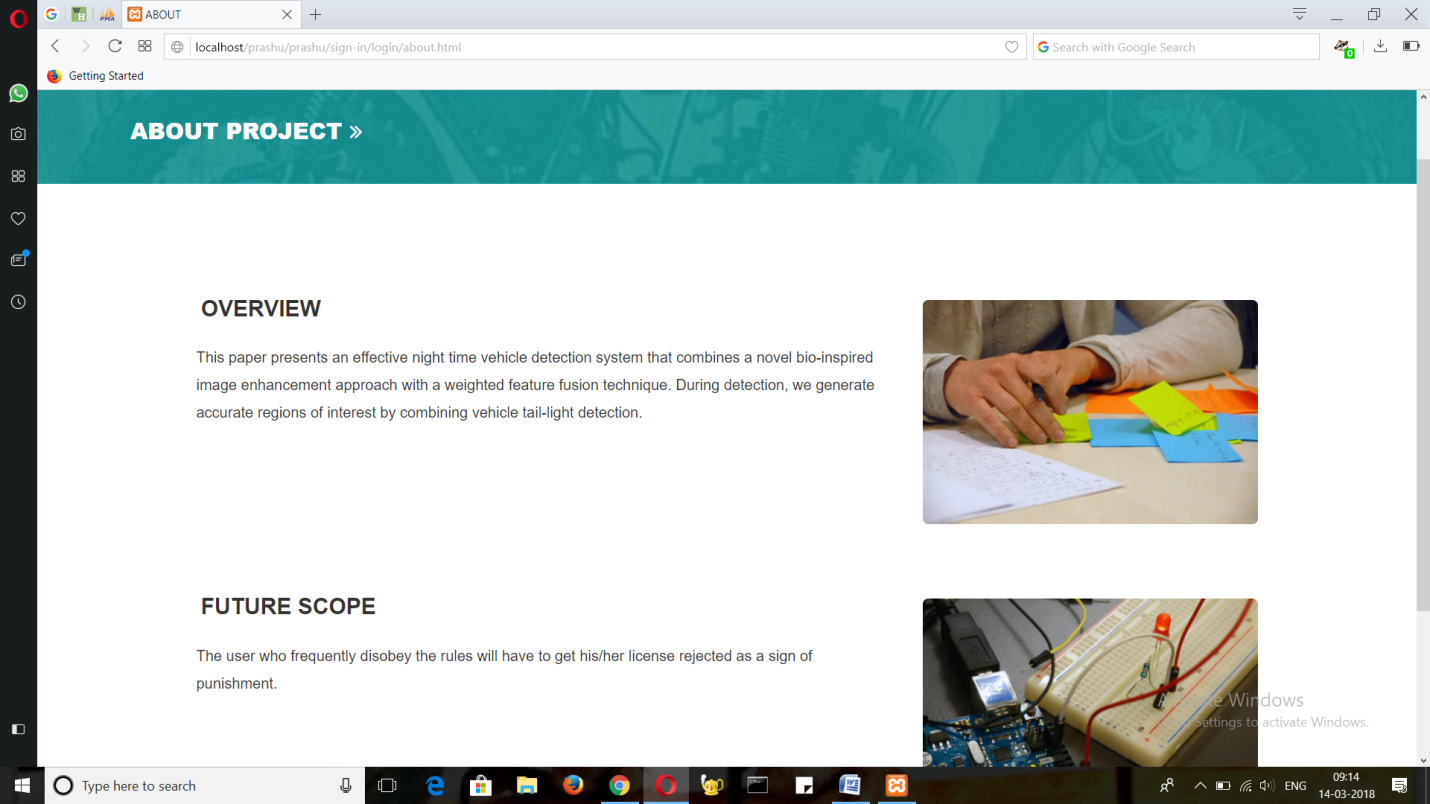
****

Fig 7.2

**Explanation**

From the above window we can identify the details of the project.

**7.3 Sample window for Login Page**

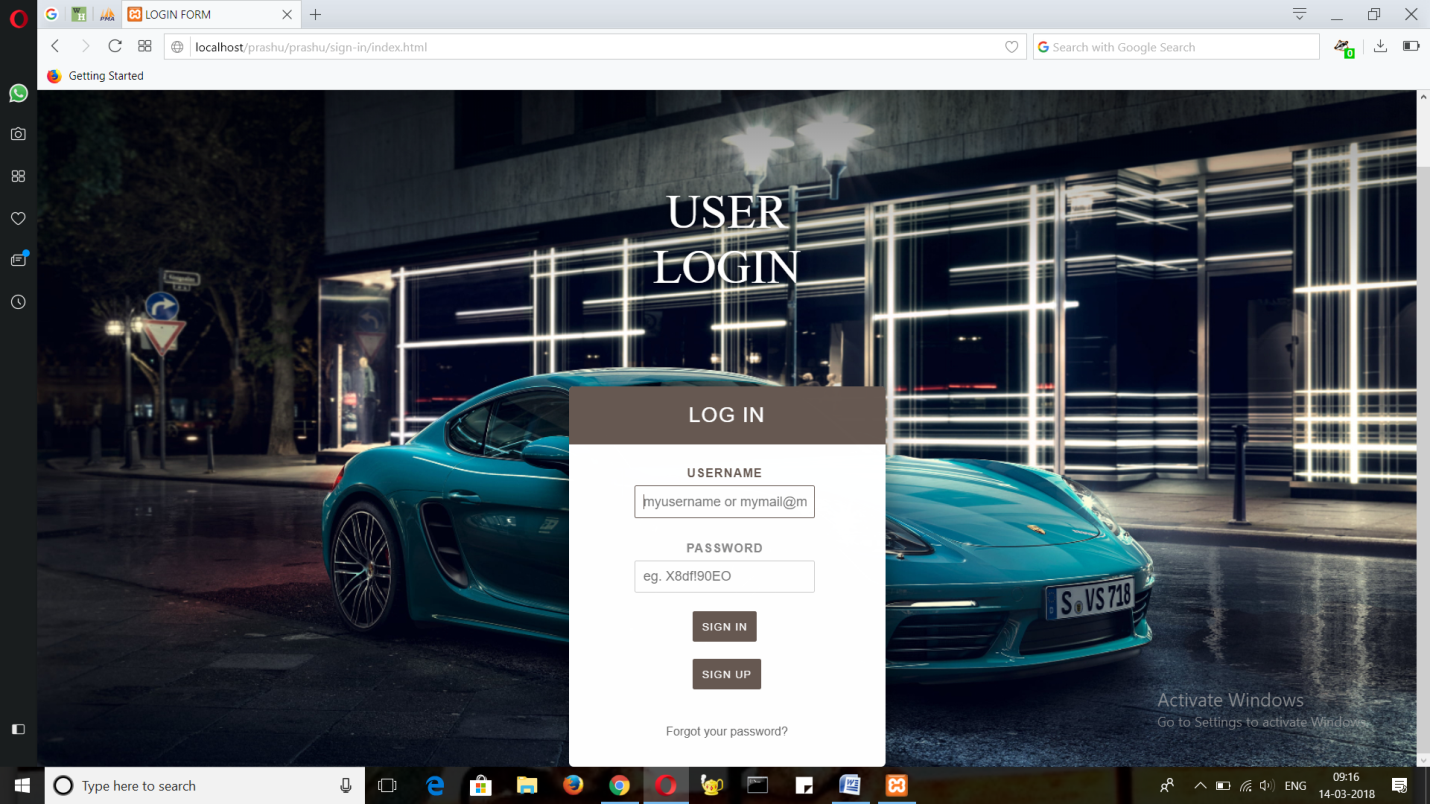


Fig 7.3

**Explanation**

From the above window we can clearly identify that login page contains login form.

* 1. **Registration Page**

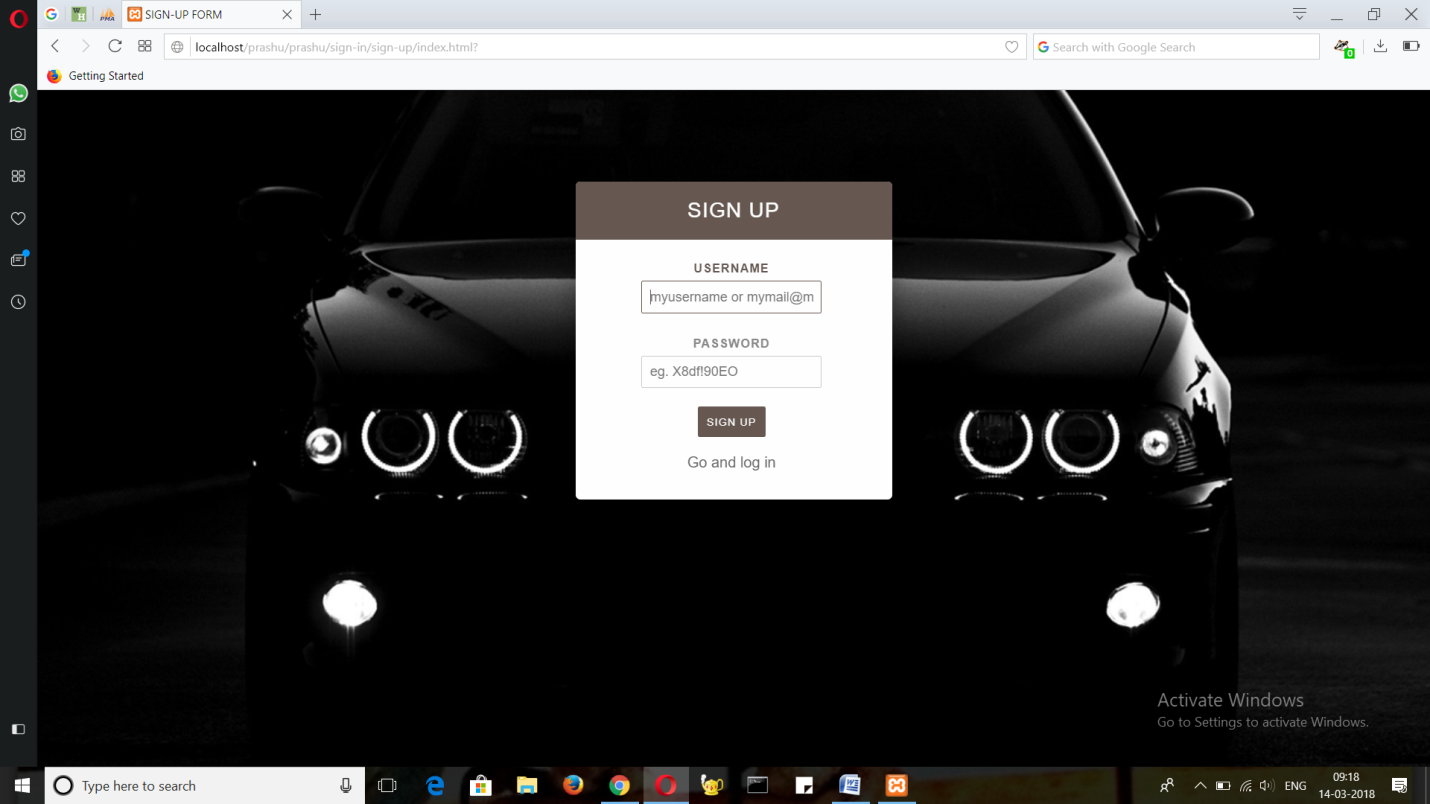


Fig 7.4

**Explanation**

Here we can see the registration page to register to the database.

* 1. **After login**

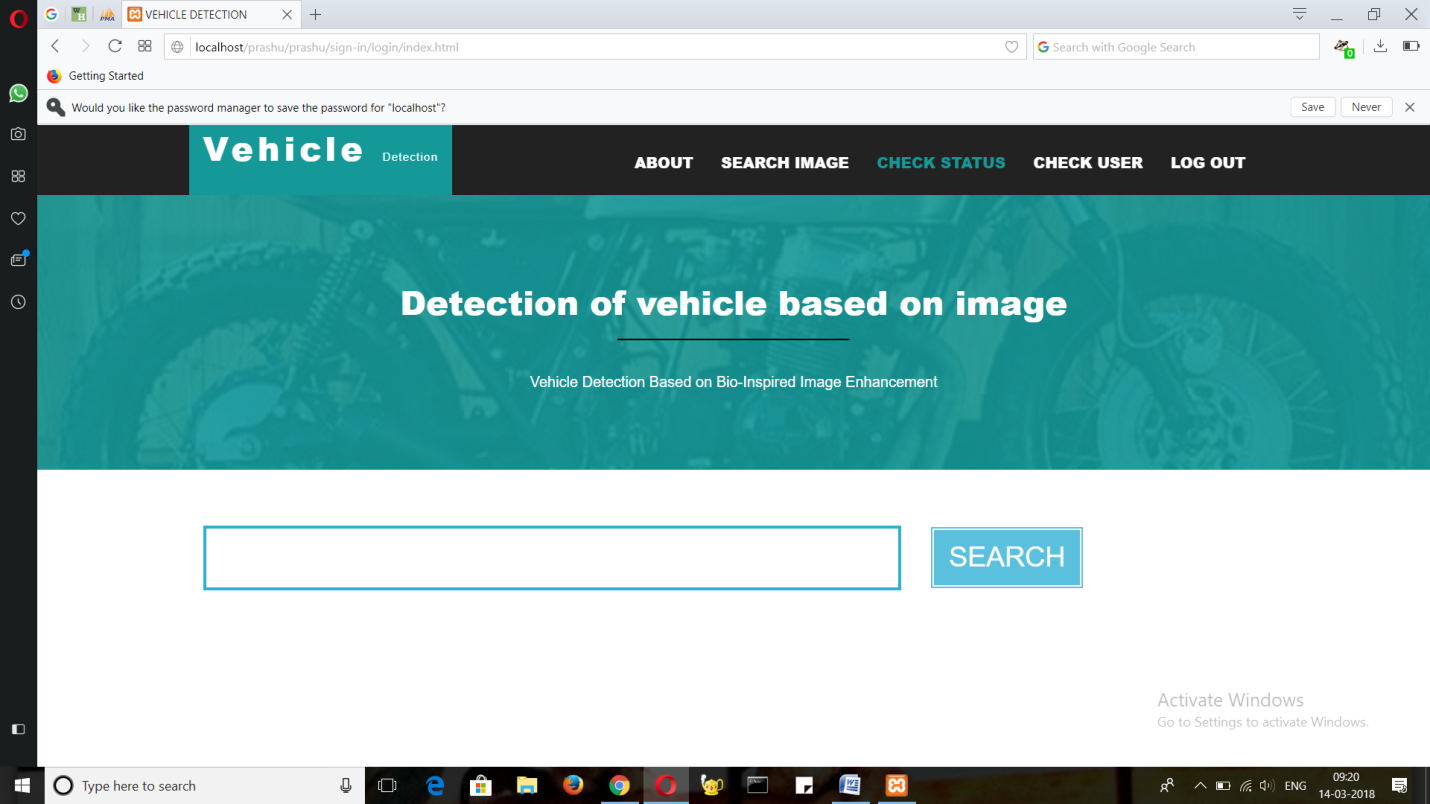
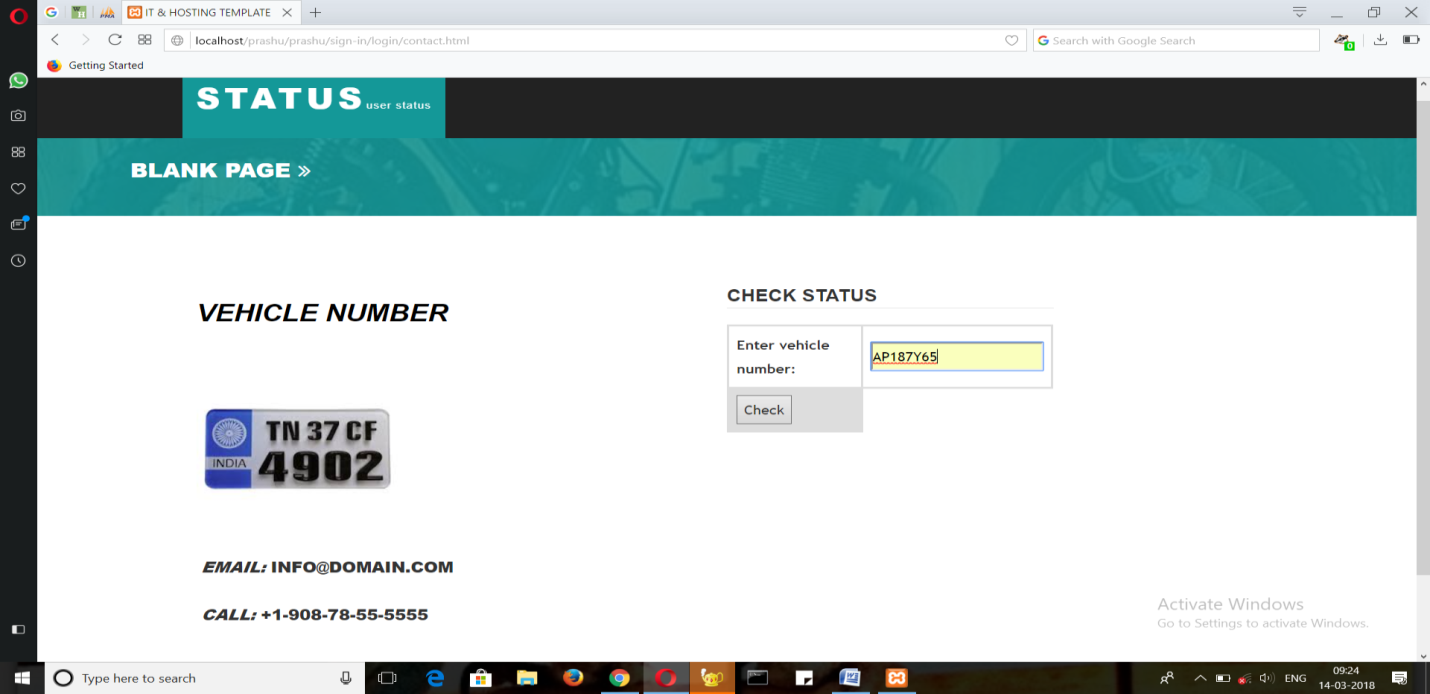


Fig 7.5

**Explanation**

Here the admin can perform different operations like verify user and check status.

* 1. **Check Status**



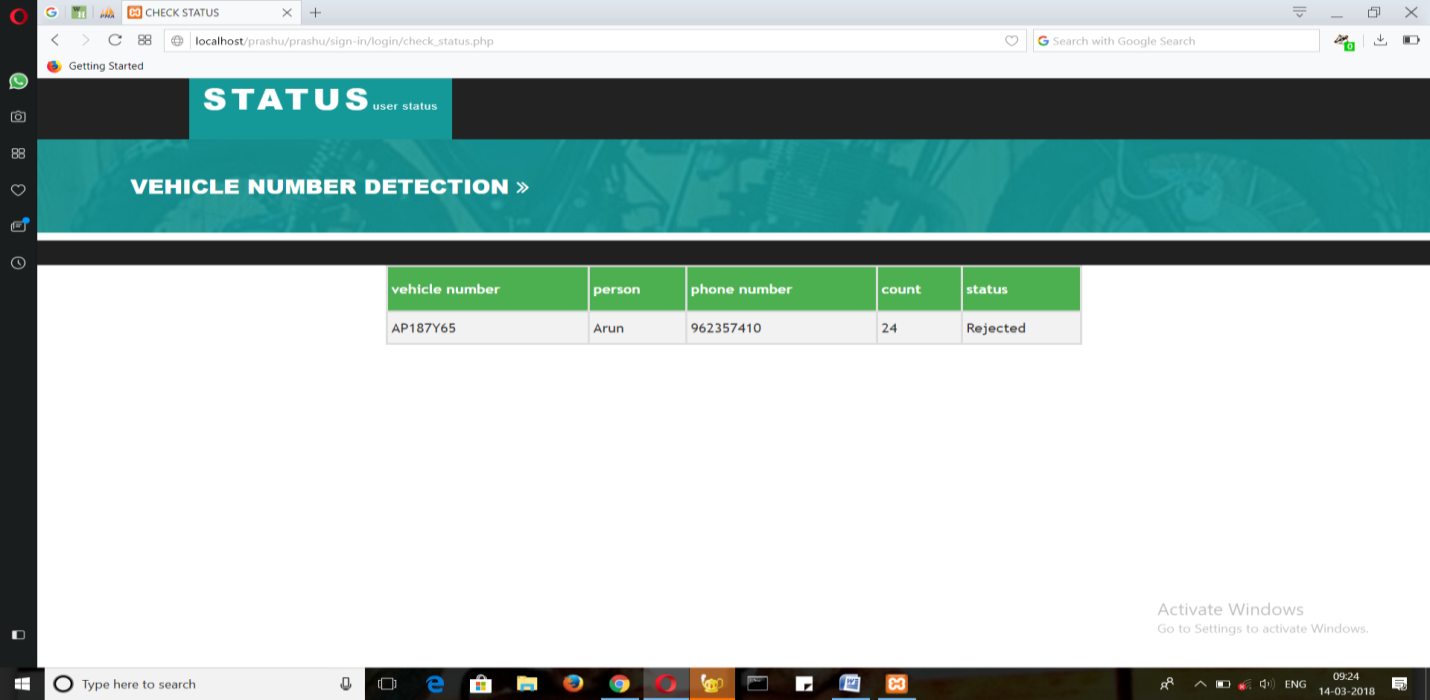
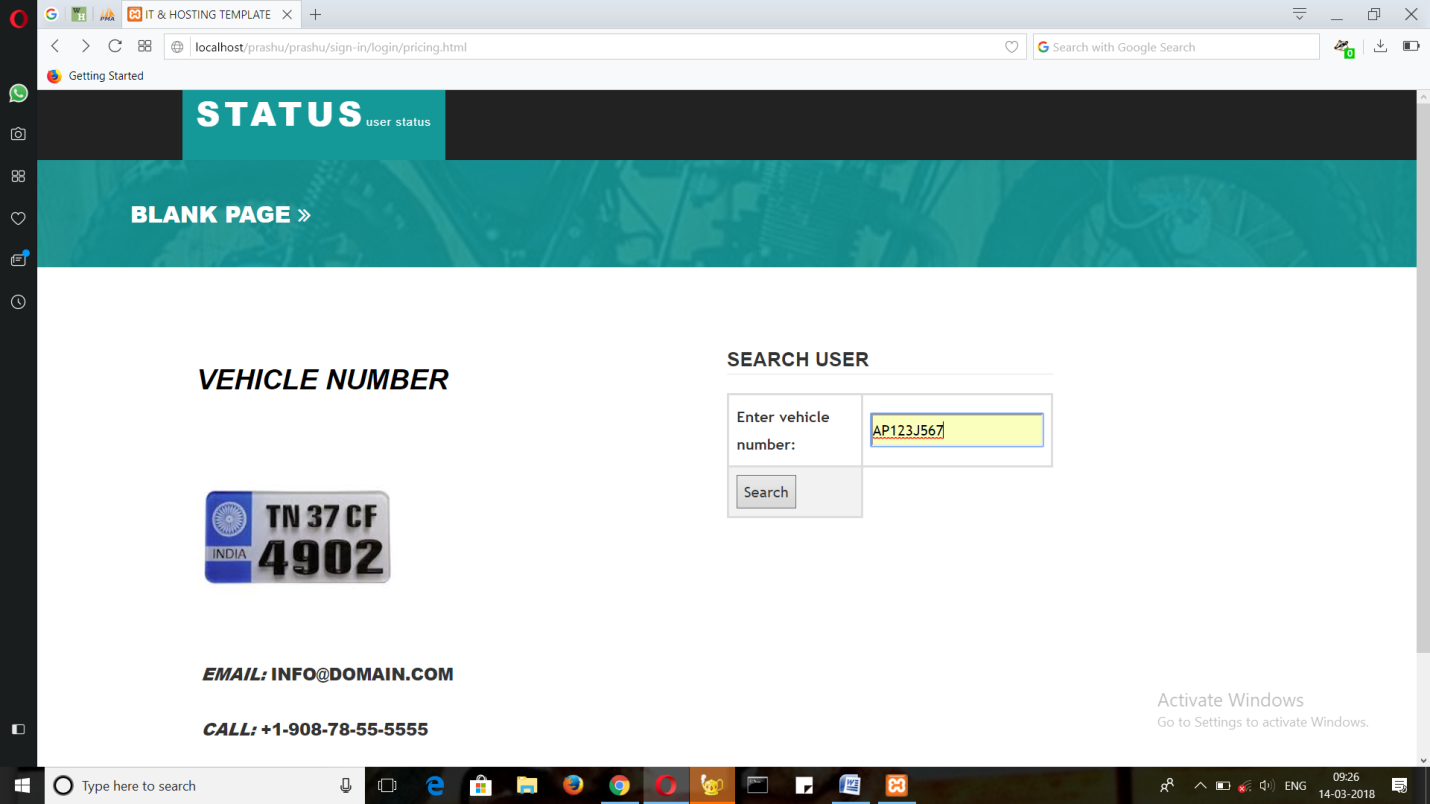


Fig 7.6

**Explanation**

Here the admin enters the vehicle number and license status is displayed.

* 1. **. Check User**



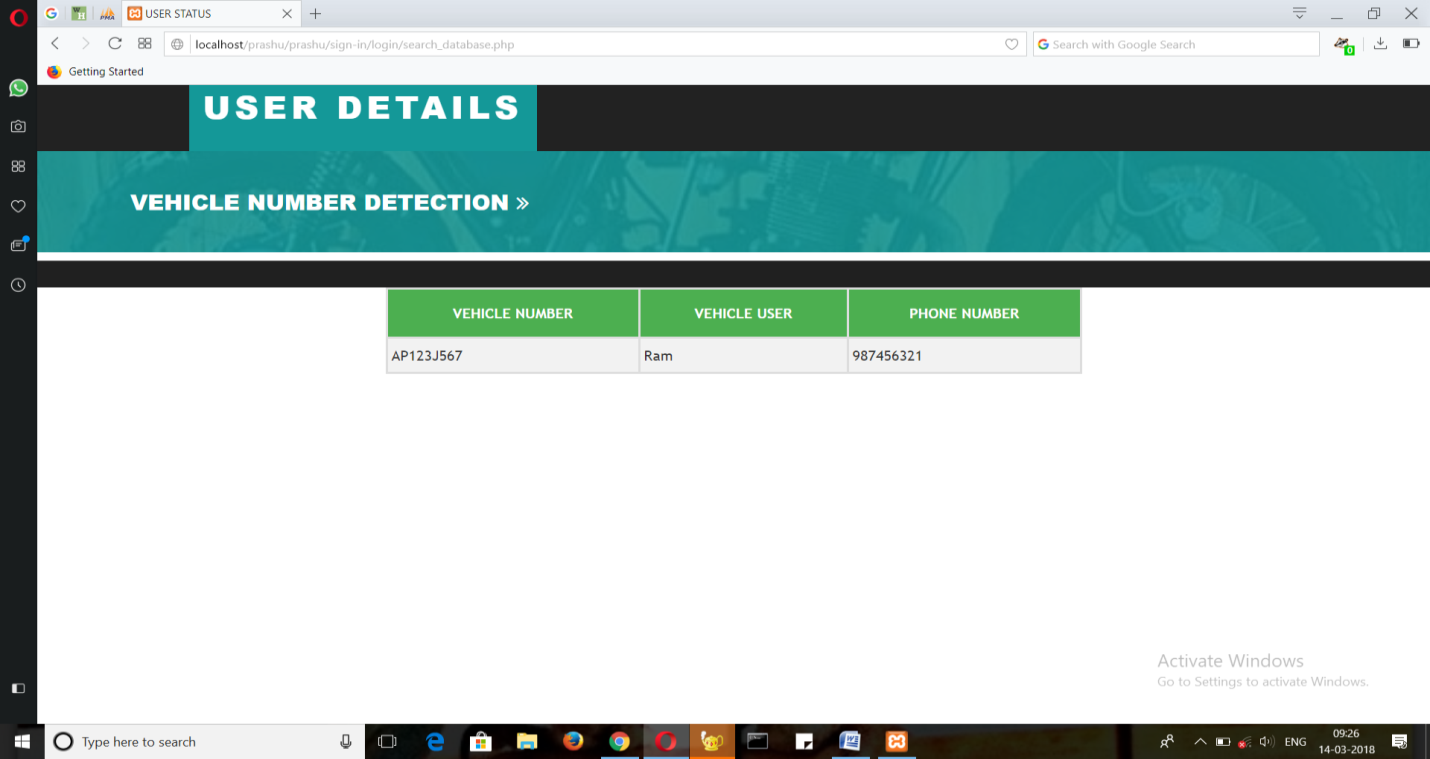


Fig 7.7

**Explanation**

Here the admin enters the vehicle number and the user details are displayed.

**7.8 Pre-processing Image**



Fig 7.8

**Explanation**

Here the image is preprocessed and the number plate is displayed.

**9.CONCLUSION AND FUTURE SCOPE**

This paper proposes a novel night time vehicle detection method combining a novel bio-inspired image enhancement approach and a weighted score-level feature fusion strategy. Experimental results demonstrate that our night time image enhancement method can enhance contrast and brightness well and can also preserve and improve the object details, and is better than some state-of-the-art night time image enhancement techniques in terms of contributing to better vehicle detection. Moreover, complementary features are combined linearly using a new weighted score-level feature fusion approach (DPM”, “CNN+SVM”). The vehicles in different types and sizes in complex scenes can be successfully detected.

**REFERENCES**

[1] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in Proc. IEEE CVPR, 2005, vol. 1, pp. 886–893.

[2] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, “Object detection with discriminatively trained part-based models,”

[3] H. Kuang, L. Chen, F. Gu, J. Chen, L. Chan, and H. Yan, “Combining region-of-interest extraction and image enhancement for nighttime vehicle detection,” IEEE Intell. Syst., vol. 31, no. 3, pp. 57–65, May/Jun. 2016.

[4] B. Alexe, T. Deselaers, and V. Ferrari, “Measuring the objectness of image windows,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 34, no. 11, pp. 2189–2202, Nov. 2012.

[5] C. L. Zitnick and P. Dollár, “Edge boxes: Locating object proposals from edges,” in Proc. ECCV, 2014, pp. 391–405.

[6] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, “CNN features off-the-shelf: An astounding baseline for recognition,” in Proc. IEEE CVPR, 2014, pp. 512–519.

[7] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, and R. Fergus, “Overfeat: Integrated recognition, localization and detection using convolutional networks,” in Proc. ICLR, 2014, pp. 1–16.

[8] T. Ahonen, A. Hadid, and M. Pietikäinen, “Face recognition with local binary patterns,” in Proc. ECCV, 2004, pp. 469–481.

[9] Y. M. Chen and J. H. Chiang, “Face recognition using combined multiple feature extraction based on Fourier-Mellin approach for single example image per person,” Pattern Recognit. Lett., vol. 31, no. 13, pp. 1833–1841, 2010.

[10] G. Han, C. Zhao, H. Zhang, and X. Yuan, “A new feature fusion method at decision level and its application,” Optoelectron. Lett., vol. 6, pp. 129–132, 2010.

[11] Y. Guermeur, “Combining multi-class SVMs with linear ensemble methods that estimate the class posterior probabilities,” Commun. Statist., Theory Methods, vol. 42, no. 16, pp 3030, 2013.

[12] S. Chernbumroong, S. Cang, and H. Yu, “Genetic algorithm-based classifiers fusion for multisensor activity recognition of elderly people,” IEEE J. Biomed. Health Informat., vol. 19, no. 1, pp. 282–289, Jan. 2014.

[13] C. C. Chang and C. J. Lin, “LIBSVM: A library for support vector machines,” ACM Trans. Intell. Syst. Technol., vol. 2, no. 3, pp. 389–396, 2011.

[14] R. E. Fan, K.W. Chang, C. J. Hsieh, X. R.Wang, and C. J. Lin, “Liblinear: A library for large linear classification,” J. Mach. Learn. Res., vol. 9, pp. 1871–1874, 2008.

[15] A. Mukhtar, L. Xia, and T. B. Tang, “Vehicle detection techniques for collision avoidance systems: A review” IEEE Trans. Intell. Transp. Syst., vol. 16, no. 5, pp. 1–21, Oct. 2015.

[16] R. Girshick, F. Iandola, T. Darrell, and J. Malik, “Deformable part models are convolutional neural networks,” in Proc. CVPR, 2014, pp. 437–446.

[17] H. Tehrani, T. Kawano, and S. Mita, “Car detection at night using latent filters,” in Proc. IEEE Intell. Veh. Symp., 2014, pp. 839–844.

[18] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan, “Vehicle detection in satellite images by hybrid deep convolutional neural networks,” IEEE Geosci. Remote Sens. Lett., vol. 11, no. 10, pp. 1797–1801, Oct. 2014.

[19] T. Schamm, C. V. Carlowitz, and J. M. Zollner, “On-road vehicle detection during dusk and at night,” in Proc. IEEE Intell. Veh. Symp., 2010, pp. 418–423.

[20] K. Robert, “Video-based traffic monitoring at day and night vehicle features detection tracking,” in Proc. IEEE Conf. Intell. Transp. Syst., 2009, pp. 1–6.

[21] W. Zhang, Q. M. J. Wu, G. Wang, and X. You, “Tracking and pairing vehicle headlight in night scenes,” IEEE Trans. Intell. Transp. Syst., vol. 13, no. 1, pp. 140–153, Mar. 2010.

[22] K. B. K. Hemanth, A. Naik, and A. Gowda, “Vehicle recognition at night based on tail light detection using image processing,” Int. J. Res. Eng. Sci., vol. 2, no. 5, pp. 68–75, 2014.

[23] M. S. Sayed and J. Delva, “Low complexity contrast enhancement algorithm for nighttime visual surveillance,” in Proc. Int. Conf. Intell. Syst.

[24] Pallavi and R. Sharma, “A novel algorithm for nighttime context enhancement,” Int. J. Emerg. Technol. Adv. Eng., vol. 3, no. 7, pp. 444–447, 2013.

[25] Y. Rao and L. Chen, “A survey of video enhancement techniques,” J. Inf. Hiding Multimedia Signal Process., vol. 3, no. 1, pp. 71–99, 2012.

[26] H. Lin and Z. Shi, “Multi-scale retinex improvement for nighttime image enhancement,” Optik, vol. 125, no. 24, pp. 7143–7148, 2014.

[27] Z. U. Rahman, D. J. Jobson, and G. A. Woodell, “Retinex processing for automatic image enhancement,” J. Electron. Imag., vol. 13, no. 1, pp. 100–110, 2004.

[28] X. Wang, T. X. Han, and S. Yan, “An HOG-LBP human detector with partial occlusion handling,” in Proc. IEEE ICCV, 2009, pp. 32–39.

[29] R. H. Masland, “The neuronal organization of the retina,” Neuron, vol. 76, pp. 266–280, 2012.

[30] C. Joselevitch, “Human retinal circuitry and physiology,” Psychol. Neurosci., vol. 1, pp. 141–165, 2008.

[31] M. Kamermans, I. Fahrenfort, K. Schultz, U. Janssen-Bienhold, T. Sjoerdsma, and R. Weiler, “Hemichannel-mediated inhibition in the outer retina,” Science, vol. 292, pp. 1178–1180, 2001.

[32] D. Xin and S. A. Bloomfield, “Dark- and light-induced changes in coupling between horizontal cells in mammalian retina,” J. Comput. Neurol., vol. 405, pp. 75–87, 1999.

[33] S. A. Bloomfield and B. Völgyi, “The diverse functional roles and regulation

of neuronal gap junctions in the retina,” Nature Rev. Neurosci., vol. 10, pp. 495–506, 2009.

[34] M. Srinivas, M. Costa, Y. Gao, A. Fort, G. I. Fishman, and D. C. Spray, “Voltage dependence of macroscopic and unitary currents of gap junction channels formed by mouse connexin50 expressed in rat neuroblastoma cells,” J. Physiol., vol. 517, pp. 673–689, 1999.

[35] X.-M. Song and C.-Y. Li, “Contrast-dependent and contrast-independent spatial summation of primary visual cortical neurons of the cat,” Cerebral Cortex, vol. 18, pp. 331–336, 2008.

[36] A. Kaneko and M. Tachibana, “Double color-opponent receptive fields of carp bipolar cells,” Vis. Res., vol. 23, pp. 381–388, 1983.

[37] C. Enroth-Cugell and J. G. Robson, “The contrast sensitivity of retinal ganglion cells of the cat,” J. Physiol., vol. 187, pp. 517–552, 1966.

[38] D. Y. Chen, Y. H. Lin, and Y. J. Peng, “Nighttime brake-light detection by Nakagami imaging,” IEEE Trans. Intell. Transp. Syst., vol. 13, no. 4, pp. 1627–1637, Dec. 2012.

[39] R. Girshick, “Fast R-CNN,” in Proc. IEEE ICCV, 2015, pp. 1440–1448.

[40] M. M. Cheng, Z. Zhang,W. Y. Lin, and P. Torr, “BING: Binarized normed gradients for objectness estimation at 300fps,” in Proc. IEEE CVPR, 2014,

pp. 3286–3293.

[41] Z. Wang, E. P. Simoncelli, and A. C. Bovik, “Multi-scale structural similarity for image quality assessment,” in Proc. IEEE Asilomar Conf. Signals, Syst., Comput., Nov. 2003, pp. 1398–1402.

[42] Z.Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600–612, Apr. 2004.

[43] A. Mittal, A. K. Moorthy, and A. C. Bovik, “No-reference image quality assessment in the spatial domain,” IEEE Trans. Image Process., vol. 21, no. 12, pp. 4695–4708, Dec. 2012.

[44] A. Mittal, R. Soundararajan, and A. C. Bovik, “Making a completely blind image quality analyzer,” IEEE Signal Process. Lett., vol. 22, no. 3, pp. 209–212, Mar. 2013.

[45] P. Dollár, Z. Tu, P. Perona, and S. Belongie, “Integral channel features,” in Proc. BMVC, 2009, pp. 91.1–91.11.

[46] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool, “Pedestrian detection at 100 frames per second,” in Proc. IEEE CVPR, 2012, pp. 2903–2910.