iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** VIII **Month of publication:** August 2021

DOI: https://doi.org/10.22214/ijraset.2021.37463

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VIII Aug 2021- Available at www.ijraset.com

Road Lane Marking Detection with Deep Learning

Manu S[1], R M Shivadarshana[2], Sahana B[3], Abhijit Das[4], Shreyas Gowd M[5]
[1, 2, 3, 4, 5]*Dept. of Information Science Sai Vidya Institute of Technology Affiliated to VTU, Bengaluru, India*

Abstract: Road Lane detection is an important factor for Advanced Driver Assistant System (ADAS). In this paper, we propose a lane detection technology using deep convolutional neural network to extract lane marking features. Many conventional approaches detect the lane using the information of edge, color, intensity and shape. In addition, lane detection can be viewed as an image segmentation problem. However, most methods are sensitive to weather condition and noises; and thus, many traditional lane detection systems fail when the external environment has significant variation.
Keywords: Road Lane, Deep Learning, Lane Detection, Neural Network, Image Processing, VGG 16

I. INTRODUCTION

In the existing technology, the outcomes have been acquired based on the samples from real-world scenarios of the road environment as of the road testing. The images and vehicle signals have been acquired from a monocular camera installed in a commercial vehicle windshield and instrumentation of the data bus. The use of IPM algorithm allows the range determination, of the tracking for several ROI sizes. The smaller ROI (100 lines) analysed covers a range of 10.4 m ahead of the vehicle; the largest (150 lines) reaches a coverage of 34.5m.

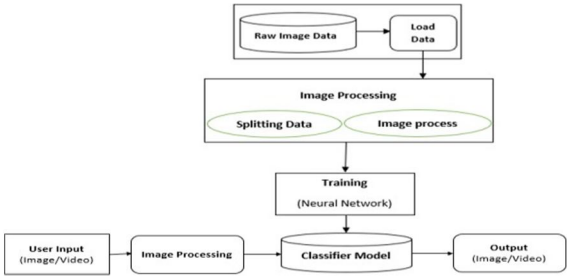


Figure 1. Flowchart of the proposed neural network training model

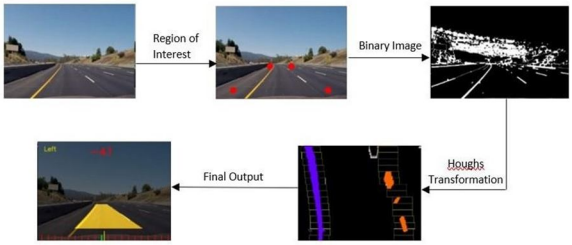


Figure 1. Flowchart of the proposed neural network training model

II. STATE OF THE ART AND EXISTING TECHNOLOGIES

A. Review on Recent Traffic signs in Lane Markings[2018]

India is a developing country and its cities are undergoing quick development an d upgrading as a result there is high growth in the road traffic. Number of0road user s were dyeing every day because of0road accidents. We need to provide road surface marking is any kind of0device or material that is used on a road surface in order to take authorized information. Lane detection is a process that is used to locate the la ne markers on the road, with the help of0this lane markers presents these locations to an intelligent system. This system decreases the road accidents and also helps to I'm proves traffic conditions.

©IJRASET: All Rights are Reserved

B. Road Marking System: A case study of Akolacity[2020]

Road is defined as an open way generally public for travel or transportation and at the same time it is a way to achieve nation's progress. Importance of roads in the world ofnation's infrastructure and socio- economic development is well known. Understanding the importance of roads, present study was carried out in Akola city of Maharashtra state in India. Study was aimed to canvas the condition of the two busiest roads in Akola city and develop solutions and feasible recommendations under the light of IRC 35 – 1997 and IRC 67- 2010. Study found that quality and quantity of road markings is far from satisfaction and needs urgent improvement for efficient traffic management. It is found that lack of maintenance has worsened the traffic management system in the city.

C. Modernisation Of Traffic Sign And Markings (India V/S Ot Her Country) For Effective Traffic Management: State Of A Rt[2018]

Highly populous countries like India are facing problem of increase in demand of transport facilities which has lead to heavy motorization. Exponential increase in number of vehicles compared to snail pace improvement of road causes many problems like traffic congestion, high accident rate and insufficient facilities. Focus in this paper is on increase in number of accidents due to insufficient provision of signs and road markings at any intersection. For safe and efficient traffic management, signs and markings must be designed and implemented in a way that the messages they convey are clear, unambiguous, visible and legible and give sufficient time to respond safely. Also, the significant improvement and maintenance is required for proper utilization of signs and markings by its users. This study will help other researchers to bring change or adopt new improvements in signs and markings while practicing in order to ensure safety and maintain smooth flow of traffic at any intersection.

D. Planning and Designing of Green Road for SingleLane[2020]

As per census of 2011, rural roads accounts of 70% of Indian total population. Rural account of 60% of the total road length in India. It covers 24,50,559 km over the country. In that more 7,81,900 km are single lane road. So we are planned to design a green road for single lane in an economical way. We use prefabricated plastic Blocks for road construction by utilizing waste plastic material. The main objective of our project is rural road development and make the India as developed country. We prepare road only for the movement of tyre. Other spaces kept empty for futuristic basis. The design of the single lane road is completely different from the conventional single lane road. These roads consist of fluorescent marking. The modelling of this single lane road is done using AUTO CADD Architecture.

E. Suitability of Different Materials Used for Road Marking : AReview[2015]

Road surface marking is any kind of0device or material that is used on a road surface in order to convey official information. They can also be applied in other facilities used by vehicles to mark parking spaces or designate areas for other uses. Various types of materials such as Traffic paints, Thermoplastics, Preformed tapes, Epoxy, polyester can be used for pavement marking. The aim of0this work is to study various types of0road markings available and ascertain the extent of0usage of0each type and to compare the economy visàvis suitability of0these materials in terms of0servicelife, skid resistance, day time luminance and night time reflectivity on different types of0 roads. The roads of0Chandigarh region in North India is taken as a test site. It is also necessary to study the performance w.r.t aging for various types of0road marking and examine the adequacy of0the IRC specifications w.r.t. various road types and recommend/ suggest modifications if0any.

F. Lane Detection using NeuralNetworks[2020]

Numerous groups have conducted many studies on lane detection. However, most methods detect lane regions by colour feature or shape models designed by human. In this paper, a traffic lane detection method using fully convolutional neural network is proposed. To extract the suitable lane feature, a small neural network is built to implement feature extraction from large amount of images. The parameters of0lane classification network model are utilized to initialize layers' parameters in lane detection network. We use Canny and Hough algorithm for detecting the lanes accurately In particular, a detection loss function is proposed to train the fully convolutional lane detection network whose output is px wise detection of0lane categories and location. The designed detection loss function consists of0lane classification loss and regression loss. With detected lane pxs, lane marking can be easily realized by random sample consensus rather than complex postprocessing. Experimental results show that the classification accuracy of0the classification network model for each category is larger than 97.5%. And detection accuracy of0the model trained by proposed detection loss function can reach 82.24% in 29 different road scenes.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VIII Aug 2021- Available at www.ijraset.com

G. Study Of Road Safety Audit In Municipalarea[2017]

The numbers of road accidents are increasing at an alarming rate in India. Thus, there is an urgent need for a systematic approach to improve road safety. Road safety audit is formal procedure for assessing accident potential and safety performance in the provision of0new road schemes, the improvement and rehabilitation of existing ro ad & in maintenance of0roads. Road safety improvement program is a systematic approach to reduce the injuries, fatalities, deaths and loss of0public properties because of0 road accidents. In this project analysis of0one of0the major sub arterial street of0PCM C will be undertaken. Intersection near Akurdi railway Station. The roadway carries considerable amount of traffic throughout the day and it has number of0conflict points . The project aims to identify deficiencies, developing mitigating strategies, improving public relations, enhancing credibility ty of0the roads and calculating the crash rate of0intersectionof0roads.

H. Gradient Enhancing Conversion for Illumination Robust Lane Detection[2013]

Lane detection is important in many advanced driver assistance systems (ADAS). Vision based lane detection algorithms are widely used and generally use gradient information as a lane feature. However, gradient values between lanes and roads vary with illumination change, which degrades the performance of0lane detection systems. In this p aper, we propose a gradient enhancing conversion method for illumination robust lane detection. Our proposed gradient enhancing conversion method produces a new gray level image from an RGB color image based on linear discriminant analysis. The con verted images have large gradients at lane boundaries. To deal with illumination changes, the gray level conversion vector is dynamically updated. In addition, we propose a novel lane detection algorithm, which uses the proposed conversion method, adaptive Canny edge detector, Hough transform, and curve model fitting method. We performed several experiments in various illumination environments and confirmed that the gradient is maximized at lane boundaries on the road. The detection rate of0the proposed lane detection algorithm averages 96% and is greater than 93% in very poor environments.

I. Lane Detection of0curving roads for structural highway with Straight Curve ModelonVisionpublished in 2019 by IEEE transaction on Vehicular Technology

Curve is the traffic accident prone area in the traffic system of the structural road. Ho w to effectively detect the lane line and timely give the traffic information ahead for drivers is a difficult point for the assisted safe driving. The traditional lane detection technology is not very applicable in the curved road conditions.

Thus, a curve detection algorithm which is based on straight curve model is proposed in this paper and this method has good applicability for most curve road conditions. First, the method divides t he road image into the region of0interest and the road background region by analysing the basic characteristics of the road image. The region of interest is further divided into the straight region and the curve region.

At the same time, the straight curve mathematical model is established. The mathematical equation of the straight model is obtained by using the improved Hough transform. The polynomial curve model is established ac cording to the continuity of0the road lane line and the tangent relationship between the straight model and the curve model. Then the parameters of the curve model equation are solved by the curve fitting method.

Finally, the detection and identification of the straight and the curve are realized respectively and the road lane line is reconstructed. Experiments show that this method can accurately identify the curve lane line, provide effective traffic information, make early warning and it also has a certain universality.

Due to the high volume of traffic on modern roadways, transportation agencies have proposed high occupancy vehicle (HOV) and high occupancy tolling (HOT) lanes to promote carpooling. Enforcement of the rules of0these lanes is currently performed by roadside enforcement officers using visual observation Officer based enforcement is, however, known to be inefficient, costly, potentially dangerous, and ultimately ineffective

Violationratesupto50%to80% have been reported, whereas manual enforcement rates of less than 10% are typical. Near infrared (NIR) camera systems have been recently proposed to monitor HOV/HOT lanes and enforce the regulations. These camera systems bring an opportunity to automatically determine vehicle occupancy from captured HOV/ HOT NIR images. Due to their ability to see through windshields of0vehicles, these cameras also enable enforcement of0other passenger compartment violations such as seat belt violation and driver cell phone usage, in addition to determining vehicle occupancy.

©IJRASET: All Rights are Reserved

J. Automatic Detection and Classification of Road Lane Markings Using Onboard Vehicular Cameras published in 2015 by IEEE transaction on Intelligent Transport Systems

This paper presents a new approach for road lane classification using an onboard camera. Initially, lane boundaries are detected using a linear–parabolic lane model, and an automatic on-the-fly camera calibration procedure is applied. Then, an adaptive smoothing scheme is applied to reduce noise while keeping close edges separated, and pairs of local maxima–minima of the gradient are used as cues to identify lane markings. Finally, a Bayesian classifier based on mixtures of Gaussians is applied to classify the lane markings present at each frame of a video sequence as dashed, solid, dashed solid, solid dashed, or double solid. Experimental results indicate an overall accuracy of over 96% using a variety of video sequences acquired with different devices and resolutions.

K. Real Time Lane Detection for Autonomous Vehicles published in 2016 atInter national Conference on computer and communication Engineering

An increasing safety and reducing road accidents, thereby saving lives are one of0gr eat interest in the context of0Advanced Driver Assistance Systems. Apparently, among the complex and challenging tasks of future road vehicles is road lane detection or road boundaries detection. It is based on lane detection(which includes the localization of the road, the determination of the relative position between vehicle and road, and the analysis of the vehicle's heading direction). One of the principal approaches to detect road boundaries and lanes using vision system on the vehicle. However, lane detection is a difficult problem because of the varying road conditions that one can encounter while driving. In this paper, a vision based lane detection approach capable of0reaching real time operation with robustness to lighting change and shadows is presented. The system acquires the front view using a camera mounted on the vehicle then applying few processes in order to detect the lanes. Using a pair of hyperbolas which are fitting to the edges of the lane, those lanes are extracted using Hough transform. The proposed lane detection system can be applied on both painted and unpainted road as well as curved and straight road indifferent weather conditions. This approach was tested and the experimental results show that the proposed scheme was robust and fast enough f or real time requirements.

III. PROPOSED TECHNIQUE

We propose a lane detection algorithm for vehicles in complex road conditions and dynamic environments. Firstly, converting the distorted image and using the superposition threshold algorithm based on the Sobel operator and colour space for edge detection, an aerial view of the lane was obtained by using ROI (Region of Interest) extraction and inverse perspective transformation. We use Hough transformation method to extract the features of the images obtained and use the extracted features to correct the distortions in the image. Compared with traditional methods and deep learning- based methodologies, this lane detection algorithm had excellent accuracy and real-time performance, high detection efficiency and strong ant interference ability.

A. Objective

To use Deep Convolutional Neural Network (DCNN) based methods to implement lane mark detection as they outperform the traditional approaches on many applications. It also demonstrates a huge success on image semantic segmentation (filters, gradient information, colour information). We thus use this technique to extract stable lane features.

B. Scope of the Project

This technology of detection of road lanes can be applied in self driven cars and autonomous vehicles in order to detect the continuous path through which the vehicle or driver can move through at any weather conditions or poorly marked road lanes. The system would be trained to identify road lanes at any possible conditions and enable the system or the driver to move smoothly without any distortions.

C. Deep Learning

Deep learning is a subfield of machine learning, which is, in turn, a subfield of artificial intelligence (AI). For a graphical depiction of this relationship, please refer to Figure 2.1. The central goal of AI is to provide a set of algorithms and techniques that can be used to solve problems that humans perform intuitively and near automatically, but are otherwise very challenging for computers. A great example of such a class of AI problems is interpreting and understanding the contents of an image – this task is something that a human can do with little-to-no effort, but it has proven to be extremely difficult for machines to accomplish.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VIII Aug 2021- Available at www.ijraset.com

While AI embodies a large, diverse set of work related to automatic machine reasoning (inference, planning, heuristics, etc.), the machine learning subfield tends to be specifically interested in pattern recognition and learning from data.

Artificial Neural Networks (ANNs) are a class of machine learning algorithms that learn from data and specialize in pattern recognition, inspired by the structure and function of the brain. As we'll find out, deep learning belongs to the family of ANN algorithms, and in most cases, the two terms can be used interchangeably. In fact, you may be surprised to learn that the deep learning field has been around for over 60 years, going by different names and incarnations based on research trends, available hardware and datasets, and popular options of prominent researchers at the time. We'll review a brief history of deep learning, discuss what makes a neural network "deep", and discover the concept of "hierarchical learning" and how it has made deep learning one of the major success stories in modern day machine learning and computer vision.

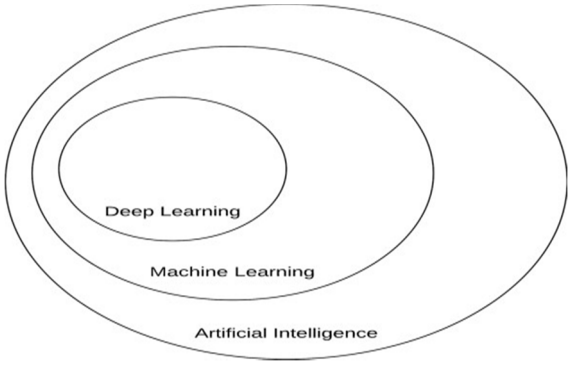


Figure 2.1: A Venn diagram describing deep learning as a subfield of machine learning which a subfield of artificial intelligence is in turn.

D. Modules

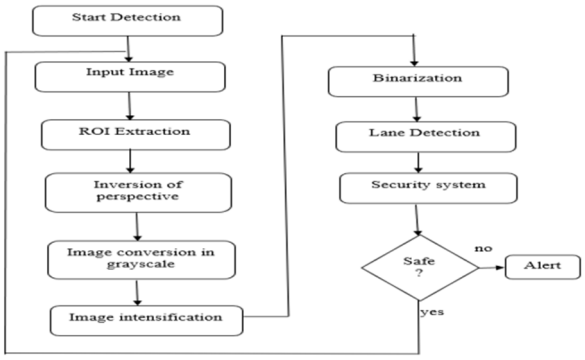
1) Calibrate the camera
2) Threshold the image using gradients and colours
3) Perspective transform
4) Identify the lane lines
5) Calibrate the camera

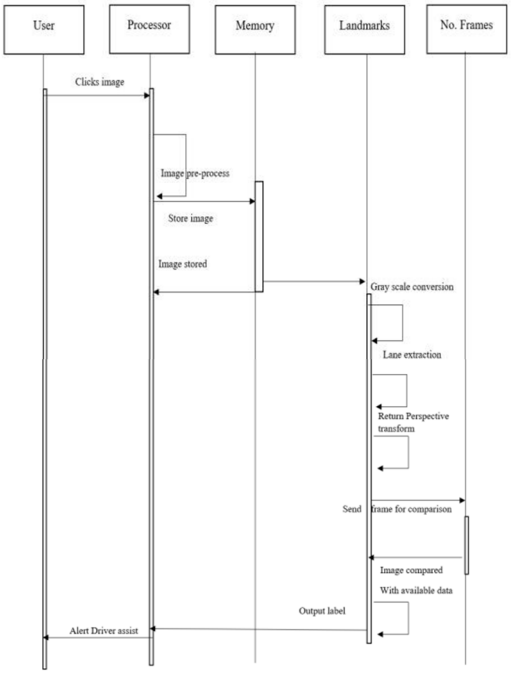
Calibrating the camera really means accounting for the distortion in an image introduced by the camera's lens. This is done using multiple images of checkerboard pattern, which should have straight lines. Examining how the checkerboard patterns are distorted (not straight) allows us to precisely identify how the camera lens is distorting images.

a) Threshold the image using gradients and colours Thresholding is a method of isolating the pixels we are interested in. This can be done using a combination of gradient and colour filters.
b) Perspective transform While undistorting and thresholding help isolate the important information, we can further isolate that information by looking only at the portion of the image we care about—the road. To focus in on the road-portion of the image we shift our perspective to a top-down view of the road in front of the car. While we don't gain any extra information from this step, it's much easier to isolate lane lines and measure things like curvature from this perspective.
c) Identify the lane lines Finally, we take all this information we gathered and draw the results back onto the original image. The colour differentiation allows us to identify the lane. The calculated right/left lane curvature and centre- lane offset are shown in the top-left of the image as well.

It's extremely important that the training set and testing set are independent of each other and do not overlap! If you use your testing set as part of your training data, then your classifier has an unfair advantage since it has already seen the testing examples before and "learned" from them. Instead, you must keep this testing set entirely separate from your training process and use it only to evaluate your network.

©IJRASET: All Rights are Reserved





IV. METHODOLOGIES

A. The Four Steps To Constructing A Deep Learning Model

1) Step #1: Gather Your Dataset

The first component of building a deep learning network is to gather our initial dataset. We need the images themselves as well as the labels associated with each image.

These labels should come from a finite set of categories, such as: categories = dog, cat, panda.

Furthermore, the number of images for each category should be approximately uniform (i.e., the same number of examples per category). If we have twice the number of cat images than dog images, and five times the number of panda images than cat images, then our classifier will become naturally biased to overfitting into these heavily-represented categories.

Class imbalance is a common problem in machine learning and there exist a number of ways to overcome it. We'll discuss some of these methods later, but keep in mind the best method to avoid learning problems due to class imbalance is to simply avoid class imbalance entirely.

2) Step #2: Split Your Dataset

Now that we have our initial dataset, we need to split it into two parts:

a) A training set

b) A testing set

A training set is used by our classifier to "learn" what each category looks like by making predictions on the input data and then correct itself when predictions are wrong. After the classifier has been trained, we can evaluate the performing on a testing set.

©IJRASET: All Rights are Reserved

Common split sizes for training and testing sets include 66:6%33:3%, 75%=25%, and 90%=10%, respectively. (Figure 4.7):

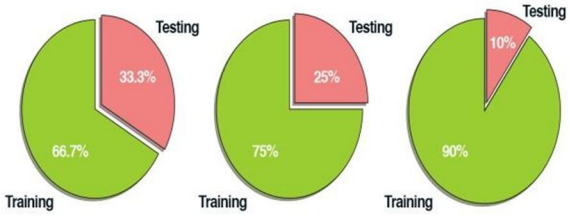


Figure 4.7: Examples of common training and testing data splits.

These data splits make sense, but what if you have parameters to tune? Neural networks have a number of knobs and levers (ex., learning rate, decay, regularization, etc.) that need to be tuned and dialled to obtain optimal performance. We'll call these types of parameters hyperparameters, and it's critical that they get set properly. In practice, we need to test a bunch of these hyperparameters and identify the set of parameters that works the best. You might be tempted to use your testing data to tweak these values, but again, this is a major no-no! The test set is only used in evaluating the performance of your network. Instead, you should create a third data split called the validation set. This set of the data (normally) comes from the training data and is used as "fake test data" so we can tune our hyperparameters. Only after have we determined the hyperparameter values using the validation set do we move on to collecting final accuracy results in the testing data. We normally allocate roughly 10- 20% of the training data for validation. If splitting your data into chunks sounds complicated, it's actually not.

3) Step #3: Train Your Network

Given our training set of images, we can now train our network. The goal here is for our network to learn how to recognize each of the categories in our labelled data. When the model makes a mistake, it learns from this mistake and improves itself. So, how does the actual "learning" work? In general, we apply a form of gradient descent.

4) Step #4: Evaluate

Last, we need to evaluate our trained network. For each of the images in our testing set, we present them to the network and ask it to predict what it thinks the label of the image is. We then tabulate the predictions of the model for an image in the testing set.

Finally, these model predictions are compared to the ground-truth labels from our testing set. The ground-truth labels represent what the image category actually is. From there, we can compute the number of predictions our classifier got correct and compute aggregate reports such as precision, recall, and f-measure, which are used to quantify the performance of our network as a whole.

VGG16 and VGG19

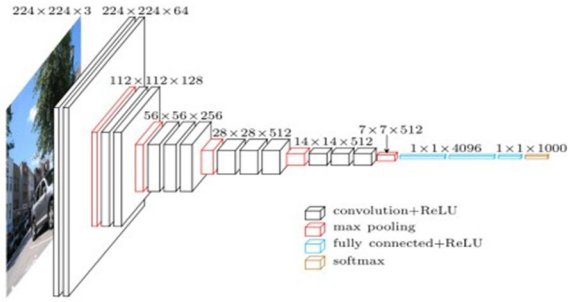


Figure 1: A visualization of the VGG architecture (source).

B. Convolutional Neural Networks

Each layer in a CNN applies a different set of filters, typically hundreds or thousands of them, and combines the results, feeding the output into the next layer in the network. During training, a

CNN automatically learns the values for these filters.

In the context of image classification, our CNN may learn to:

1) Detect edges from raw pixel data in the first layer.

2) Use these edges to detect shapes (i.e., "blobs") in the second layer.

3) Use these shapes to detect higher-level features such as facial structures, parts of a car, etc.in the highest layers of the network.

©IJRASET: All Rights are Reserved

The last layer in a CNN uses these higher-level features to make predictions regarding the Contents of the image.
In terms of deep learning, an (image) convolution is an element-wise multiplication of two matrices followed by a sum.

a) Take two matrices (which both have the same dimensions).
b) Multiply them, element-by-element (i.e., not the dot product, just a simple multiplication).
c) Sum the elements together. Kernels

Again, let's think of an image as a big matrix and a kernel as a tiny matrix (at least in respect to the original "big matrix" image), depicted in Figure 11.1. As the figure demonstrates, we are sliding the kernel (red region) from left-to-right and top-to-bottom along the original image. At each (x;y)-coordinate of the original image, we stop and examine the neighbourhood of pixels located at the centre of the image kernel. We then take this neighbourhood of pixels, convolve them with the kernel, and obtain a single output value. The output value is stored in the output image at the same (x;y)-coordinates as the centre of the kernel. If this sounds confusing, no worries, we'll be reviewing an example in the next section. But before we dive into an example, let's take a look at what a kernel looks like (Figure 11.3):

Figure 11.1: A kernel can be visualized as a small matrix that slides across, from left-to-right and top-to-bottom, of a larger image. At each pixel in the input image, the neighbourhood of the image is convolved with the kernel and the output stored.

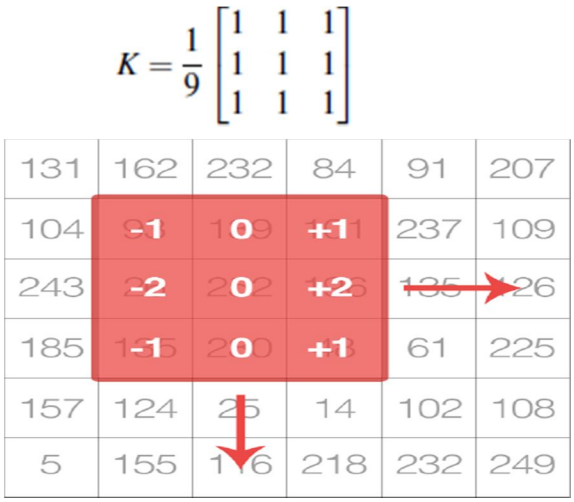


Figure 11.1: A kernel can be visualized as a small matrix that slides across, from left-to-right and top-to-bottom, of a larger image. At each pixel in the input image, the neighbourhood of the image is convolved with the kernel and the output stored.

We use an odd kernel size to ensure there is a valid integer (x;y)-coordinate at the centre of the image (Figure 11.2). On the left, we have a 3_3 matrix. The center of the matrix is located at x = 1;y = 1 where the top-left corner of the matrix is used as the origin and our coordinates are zero- indexed. But on the right, we have a 2_2 matrix. The center of this matrix would be located at x = 0:5;y = 0:5.

But as we know, without applying interpolation, there is no such thing as pixel location (0:5;0:5) – our pixel coordinates must be integers! This reasoning is exactly why we use odd kernel sizes: to always ensure there is a valid (x;y)-coordinate at the center of the kernel.

Example of Convolution: Now that we have discussed the basics of kernels, let's discuss the actual convolution operation and see an example of it actually being applied to help us solidify our knowledge. In image processing, a convolution requires three components:

1) An input image.
2) A kernel matrix that we are going to apply to the input image.

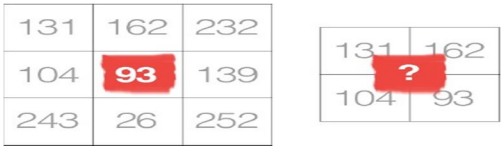


Figure 11.2: Left: The center pixel of a 3_3 kernel is located at coordinate (1;1) (highlighted in red). Right: What is the center coordinate of a kernel of size 2_2?

©IJRASET: All Rights are Reserved

An output image to store the output of the image convolved with the kernel.
Convolution (i.e., cross-correlation) is actually very easy.
All we need to do is:

a) Select an (x;y)-coordinate from the original image.
b) Place the centre of the kernel at this (x;y)-coordinate.
3) Take the element-wise multiplication of the input image region and the kernel, then sum up the values of these multiplication operations into a single value. The sum of these multiplications is called the kernel output.
4) Use the same (x;y)-coordinates from Step #1, but this time, store the kernel output at the same (x;y)-location as the output image.

Below you can find an example of convolving (denoted mathematically as the ? operator) a 3_3 region of an image with a 3_3 kernel used for blurring:

$$O_{i,j} = \frac{1}{9}\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix} \star \begin{bmatrix}93 & 139 & 101\\ 26 & 252 & 196\\ 135 & 230 & 18\end{bmatrix} = \begin{bmatrix}1/9\times 93 & 1/9\times 139 & 1/9\times 101\\ 1/9\times 26 & 1/9\times 252 & 1/9\times 196\\ 1/9\times 135 & 1/9\times 230 & 1/9\times 18\end{bmatrix} \quad (11.4)$$

Therefore,

$$O_{i,j} = \sum \begin{bmatrix}10.3 & 15.4 & 11.2\\ 2.8 & 28.0 & 21.7\\ 15.0 & 25.5 & 2.0\end{bmatrix} \approx 132. \quad (11.5)$$

After applying this convolution, we would set the pixel located at the coordinate (i; j) of the output image O to Oi; j = 132.

That's all there is to it! Convolution is simply the sum of element-wise matrix multiplication between the kernel and neighbourhood that the kernel covers of the input image.

V. TESTING AND RESULTS

A. Testing

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding software bugs (errors or other defects). Software testing involves the execution of a software component or system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test:

1) Meets the requirements that guided its design and development,
2) Responds correctly to all kinds of inputs,
3) Performs its functions within an acceptable time, is sufficiently usable,
4) Can be installed and run in its intended environments, and
5) Achieves the general result its stakeholder's desire.

As the number of possible tests for even simple software components is practically infinite, all software testing uses some strategy to select tests that are feasible for the available time and resources. As a result, software testing typically (but not exclusively) attempts to execute a program or application with the intent of finding software bugs (errors or other defects). The job of testing is an iterative process as when one bug is fixed; it can illuminate other, deeper bugs, or can even create new ones.
Software testing can provide objective, independent information about the quality of software and risk of its failure to users and/or sponsors. Software testing can be conducted as soon as executable software (even if partially complete) exists. The overall approach to software development often determines when and how testing is conducted. For example, in a phased process, most testing occurs after system requirements have been defined and then implemented in testable programs. In contrast, under an Agile approach, requirements, programming, and testing are often done concurrently.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VIII Aug 2021- Available at www.ijraset.com

B. The Main Aim Of Testing

The main aim of testing is to analyze the performance and to evaluate the errors that occur when the program is executed with different input sources and running in different operating environments.
In this project, we have developed an Image Processing code which helps in detection of lane. The main aim of testing this project is to check if the lane is getting detected accurately and check the working performance when different images are given as inputs.

The testing steps are:

1) Unit Testing.
2) Integration Testing.
3) Validation Testing.
4) User Acceptance Testing.
5) Output Testing

C. Unit Testing

Unit testing, also known as component testing refers to tests that verify the functionality of a specific section of code, usually at the function level. In an object-oriented environment, this is usually at the class level, and the minimal unit tests include the constructors and destructors. Unit testing is a software development process that involves synchronized application of a broad spectrum of defect prevention and detection strategies in order to reduce software development risks, time, and costs. The following Unit Testing Table shows the functions that were tested at the time of programming. The first column gives all the modules which were tested, and the second column gives the test results. Test results indicate if the functions, for given inputs are delivering valid outputs. Function Name Tests Results Uploading for uploading test video different types and sizes of images. Detecting lane Tested for different images of lane.

Fig. 1. Function Name	Fig. 2. Tests Results
Fig. 3. Uploading video	Fig. 4. Tested for uploading different types and sizes of video.
Fig. 5. Detecting lane	Fig. 6. Tested for different lane
Fig. 7. Display result	Fig. 8. Lane prediction output(left , right and straight).

D. Integration Testing

Integration testing is any type of software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an iterative way or all together ("big bang"). Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed. Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.

E. Validation Testing

At the culmination of integration testing, software is completed assembled as a package. Interfacing errors have been uncovered and corrected. Validation testing can be defined in many ways; here the testing validates the software function in a manner that is reasonably expected by the customer. In software project management, software testing, and software engineering, verification and validation (V&V) is the process of checking that a software system meets specifications and that it fulfils its intended purpose. It may also be referred to as software quality control. The following table indicates validation tests done for checking functionality of the project after its integration with the front- end

©IJRASET: All Rights are Reserved

F. User Acceptance Testing

Performance of an acceptance test is actually the user's show. User motivation and knowledge are critical for the successful performance of the system. The above tests were conducted on the newly designed system performed to the expectations. All the above testing strategies were done using the following test case designs.

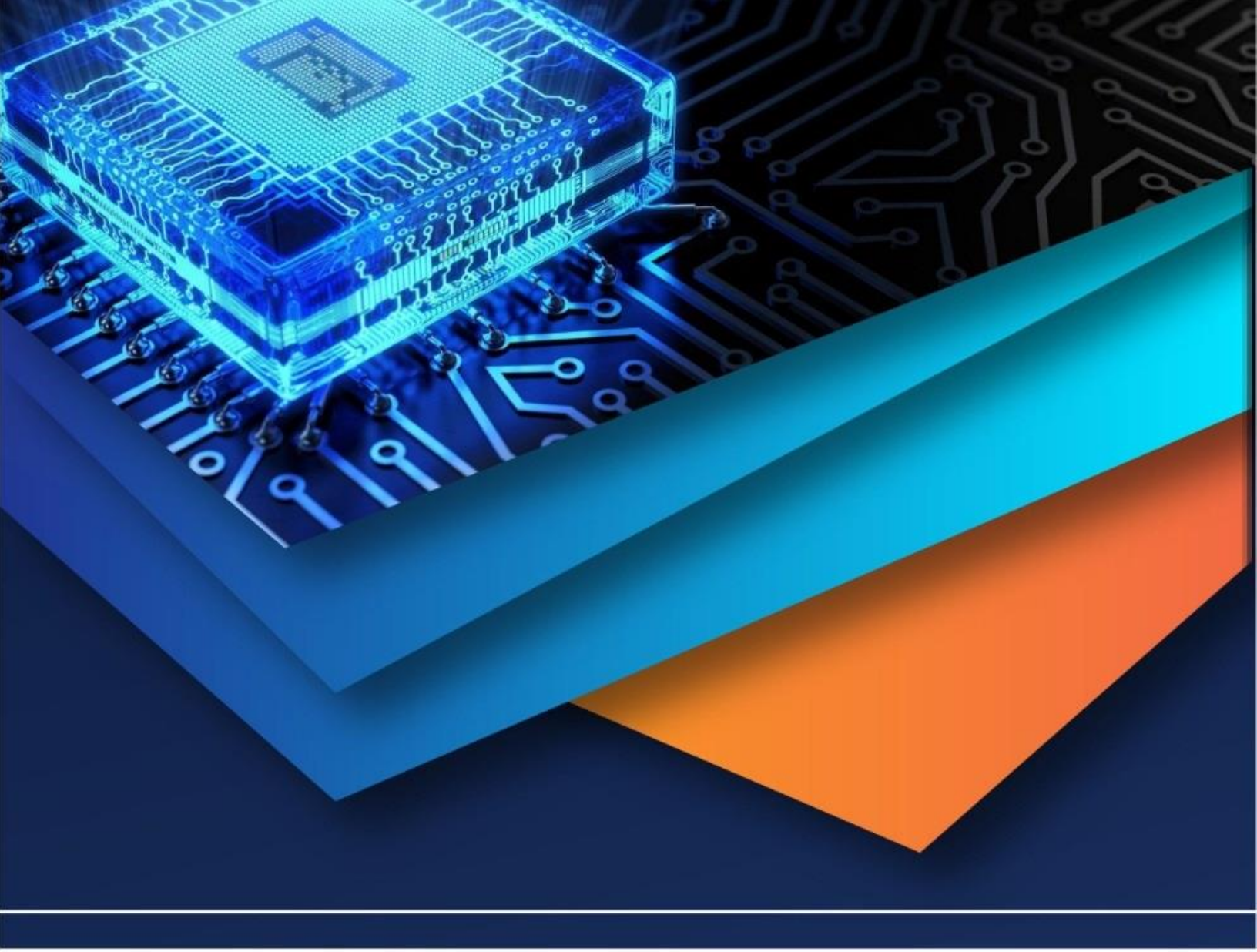
VI. CONCLUSION

In our project we make use of the algorithms that can accurately identify the road lane-line and give the deviated information of vehicle and the direction of the curve. It has great significance to improve the active safety driving and assisted driving of the vehicle which is in the curved road conditions. Even though, lot of progress has been attained in the lane detection and tracking area, further improvements can be made as per the requirements.

REFERENCES

[1] Ramesh Surisetty, B.V. Suresh Kumar "Review on Recent Traffic signs in Lane Markings", International Research Journal of Engineering and Technology (IRJET) Volume: 05 Issue: 12 | Dec 2018
[2] Prof. S.S. Suratkar, Prof. A.S. Gangane "Road Marking System: A case study of Akola city" International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 04 | Apr 2020
[3] Neha D. Solanki, Ankita J. Patel, Vipinkumar G. Yadav "MODERNISATION OF TRAFFIC SIGN AND MARKINGS (INDIA V/S OTHER COUNTRY) FOR EFFECTIVE TRAFFIC MANAGEMENT: STATE OF ART" International Research Journal of Engineering and Technology (IRJET) Volume: 05 Issue: 12 | Dec 2018
[4] T. Aakash raj, M. Susindharan, K. Hariharan, B. Vinodhini "Planning and Designing of Green Road for Single Lane" International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 06 | June 2020
[5] Sheikh Azhar U Rehman, A.K Duggal "Suitability of Different Materials Used for Road Marking: A Review" International Research Journal of Engineering and Technology (IRJET) Volume: 02 Issue: 02 | May-2015
[6] M Juanita Sarah Rachel, S Kalaiselvi, R Salini "Lane Detection using Neural Networks" IRJET Volume: 07 Issue: 03 | Mar 2020
[7] NilamPhadtare, MadhavChinchole "STUDY OF ROAD SAFETY AUDIT IN MUNICIPAL AREA" IRJET Volume: 04 Issue: 08 | Aug -2017
[8] H. Yoo, U. Yang, and K.Sohn, "Gradient-enhancing conversion for illumination-robust lane detection," IEEE Transactions on Intelligent Transportation Systems, 14(3), 1083-1094, 2013.
[9] C. Y. Low, H. Zamzuri, and S. A. Mazlan, "Simple robust road lane detection algorithm," in Intelligent and Advanced Systems (ICIAS), June 2014..

©IJRASET: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)