iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** V **Month of publication:** May 2020

DOI: http://doi.org/10.22214/ijraset.2020.5433

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 8 Issue V May 2020- Available at www.ijraset.com

Distance and Speed Estimation of Moving Object using Video Processing

S. Jeffy Berna[1], S. Swathi[2], C. Yamuna Devi[3], Varalakshmi. D[4]
[1, 2, 3]*Student,* [4]*Assistant Professor, Electronics & Communication, Engineering, S. A Engineering College, Chennai, Tamil Nadu*

Abstract: ***This paper aims to describe the comprehensive approach in order to localize target of any moving object or vehicles in video under various environmental conditions in each application basis. The fundamental step for automated video analysis, which is based on object detected in many vision application. Using Centroid value of an moving vehicle, distance travelled by the vehicle is calculate this approach is implemented in OPENCV(Python) by using video processing functions. The accuracy of the system proposed in the speed measurement is nearly 90 percent comparable to the actual speed of the moving vehicles, as the error rate is very low.***
Keyword: Centroid, OPENCV, Video processing

I. INTRODUCTION

The main objective of the project is to determine the distance and speed of moving object, especially for vehicles to control the over-speeding vehicles, using OpenCV (python). People nowadays encounter more problems as road traffic becomes congested Congestion is mainly due to level of road capacity and inbuilt infrastructure. To reduce or minimize these problems, so new approach has been developed for finding vehicle speed using centroid method. Video and image processing presented by a system [1] for speed measurement.[2]Presented a system for calculating speed of vehicle using centroid method in matlab. Thus speed calibration is being tested to coordinate in the form of centroid.

II. PURPOSE OF PROJECT

The aim of the project is to estimate the distance and speed of the vehicle. Along with the development of information and communication technology, the world urban people now recognize a new term called Smart city components is smart transportation, known as Intelligent Transportation System. Speed estimation is one of many important parts of intelligent traffic system which can be done by using video processing technique. It acts as a prototype which can be more developed and complex for large system to build a complete ITS system for smart city development. The accuracy of the system proposed in the speed measurement is compared to the actual speed of the moving vehicles. This project uses the depth information of the scene together with few computer vision algorithms in order to detect the speed of an object along an appropriate position to estimate the speed of the object in camera's view. Speed estimation has been done typically using fixed camera to estimate speed of object from a known distance from the camera.

III. PROCESSING STEPS

A. Video Processing

A particular case of signal processing along video processing in electronic engineering which often employs video filters and where the input and output signals are video files or video streams. A stream of processing architecture required from video processing systems, in which continuous stream of video frames are processed one (or more)at a time. This type of processing is critical in systems that have live video or the video data is so large that loading the entire set into workspace is insufficient. Sequence of time varying images is basically is any video signal. Frames are a series of images treated as a video signal. An illusion of continuous video is obtained by changing termed as frame rate. The resultant pixel intensities are quantized.

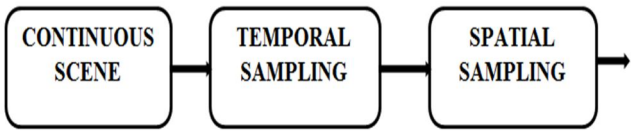


Fig 1.1 Block Diagram of Video Digitalization

©IJRASET: All Rights are Reserved

B. Digital Video Formats

Video frame is prescribed as frame rate that are displaced from the digital video. A frame rate used in NTSC video is 30 fps. For individual frames is defined in terms of the size which is specified for the frame format. The most common used frame formats are The Common Intermediate Format (CIF) has 352×288 pixels, and the Quarte CIF (QCIF) format has 176×144 pixels.

The work that is done in Video format is AVI. Frame constantly varies based on the motion of the video. It is a file format for moving content of images that wraps a video bitstream with other data chunks and payback synchronous to support picture-sound. AVI files consists of one RIFF ”chuck” tagged as AVI and these are divided into “subchunks” ,each is being identified by Microsoft’s FOURCC – four character codes. ”hdrl” is tagged as the first subchunk, providing metadata about the video along with height, width and frame rate. Wide range ofcodesis employ to the actual picture. A third optional subchunk is tagged id×1 and within the files the data of offsets are indexed respectively.

IV. BLOCK DIAGRAM

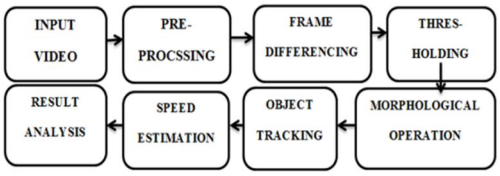


Fig 2.1 Block Diagram for Speed Estimation

A. Video Capturing

Video is captured by webcam in real time. Then the captured video is converted into frames. To capture a video, we need to create a moving object. It is considered to be an argument that can be taken as either the device index or the name of a video file. Device index is just the number to specify which camera to be taken to capture the video clearly. There are respectively two type of camera present in this operation. Normally one camera will be connected. So simply pass 0 (or -1). You can select the second camera by passing 1and so on. After that, you can capture frame-by-frame. But at the end, we need to release the capture.

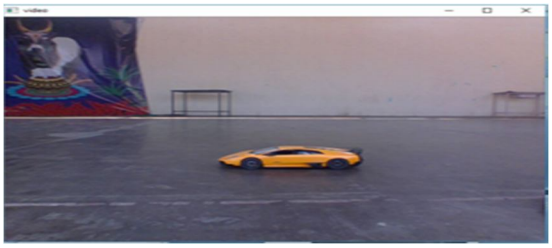


Fig 2.2 Input image (Video is captured using webcam)

B. Preprocessing

The aim of preprocessing is an improvement of the image data that suppresses unwanted distortions or enhances some image features important for further processing.

1) *RGB Video To Gray Video Conversion:* A grayscale image is a single sample in each pixel value representing only an amount of light they are also known as black-and-white or monochrome. We have to convert **RGB** Image to **Gray** Image. Because the reason is Gray scale reduces complexity: from a 3D pixel value (R, G, B) to a 1D value.

We use this function to converts an image into the given colorspace.

RGB to Gray: Y←0.299·R+0.587·G+0.114·B



Fig 2.3 Gray image

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

C. Smoothening

Smoothening(image blurring) is often used to reduce noise within an image or to produce a less pixelated image. Most smoothening methods are based on low pass filters. Image blurring (smoothening) is achieved by convolving the image with a low-pass filter kernel. It is useful for removing noise. The image resulting in edges being blurred when this is filter is applied as it actually removes high frequency content.

OpenCV provides mainly four types of blurring techniques.

1) Averaging
2) Gaussian Filtering
3) Median Filtering
4) Bilateral Filtering

Here we have used median and Gaussian blurring techniques.

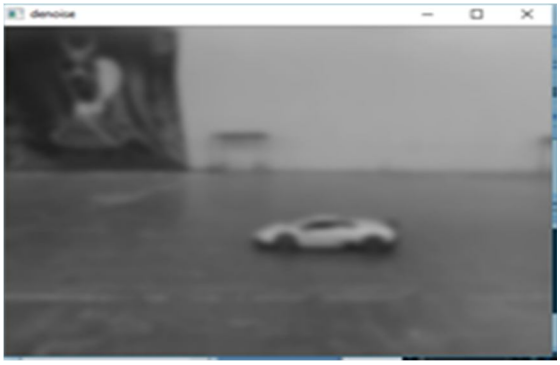


Fig 2.4 image smoothening

V. FILTERS

Filtering is useful for many applications. A filter is considered as a kernel applied to each pixel in small array form and its neighbors within an image. The center of the kernel considered in most application basis as aligned with the current pixel and an odd number of elements are of square in each dimension. The process to apply filters is known for the captured image as convolution and the spatial or frequency domain are applied in either form.

A. Median Filter

The algorithm selects the median average of all the pixels in the support: the central value in an ordered list of the pixels. The function which computes the median of all the pixels under the kernel window and the central pixel is replaced with this median value. This is highly effective in removing salt-and-pepper noise. The central element of filtered value not exists in the original image. Kernel size must be a positive odd integer.

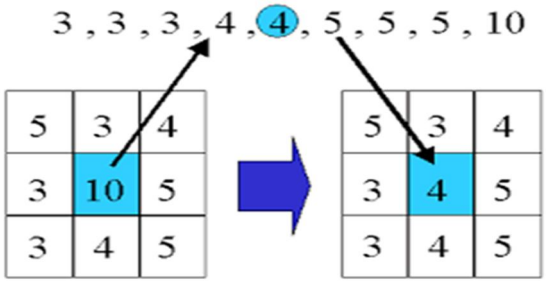


Fig 3.1 Median filtering

B. Gaussian Filtering

Instead of a box filter consisting of equal filter coefficients, in this approach a Gaussian kernel is used. Specify each width and height of kernel. We also should specify the standard deviation in the X and Y directions, sigma X and sigma Y respectively. If both are given as zeros, they are calculated from the kernel size. Gaussian filtering is highly effective only when the Gaussian noise is removed from the image. It is created by using Gaussian kernel for filtering.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

VI. MOVING OBJECT DETECTION

Object detection and tracking includes vision applications like traffic control, video surveillance, and person tracking. For detecting moving objects are considered in a video thus object detection algorithms compares at the pixel level for a static background frame with the current frame. The techniques briefed in this chapter include Frame differencing, Thresholding, Morphological Operations, Contour Extraction, Bounding Rectangle.



Fig 4.1 moving object detection

A. Frame Difference

Detection of a sequence of frames captured in order for a moving object from a static camera is widely performed by frame differencing method. The adjacent frame difference method is known as frame difference. The objective is to detect the moving objects from the difference between the existing frame and the reference frame. The frame difference method adopts pixel-based difference to find the moving object. The pixel difference is greater than the set threshold as the background pixels. The moving object is detected after threshold operation.

The absolute differential image is defined as follows:

$$D_{(x,y)} = 1\ if\left(\left|F_{(x,y)} - B_{(x,y)}\right|\right) > T$$

= 0 others

B. Thresholdin (Segmentation)

Thresholding is a simple method used for image segmentation in digital image processing. Separate out regions of an image corresponding to objects which we desire to analyze. Variation of intensity is between the object pixel and the background pixels to each separation also create binary to grayscale image. Arguments are based on source images separating based on threshold values. There are 5 different types of simple Thresholdingavailable in the OpenCV library, viz THRESHOLD BINARY, THRESHOLD BINARY – INVERTED, TRUNCATE, THRESHOLD TO ZERO and THRESHOLD TO ZERO – INVERTED. Out of them, THRESHOLD BINARY is chosen as the process involves simple background elimination and the output is required to be a binary image.



Fig 4.2 Thresholding

C. Threshold Binary

This Thresholding operation can be expressed as

$$\mathtt{dst}(x,y) = \begin{cases} \mathtt{maxVal} & \text{if } \mathtt{src}(x,y) > \mathtt{thresh} \\ 0 & \text{otherwise} \end{cases}$$

So, if the intensity of the pixel src(x, y) is higher than thresh, then the new pixel intensity is set to a max Val. Otherwise, the pixels are set to zero.

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue V May 2020- Available at www.ijraset.com

D. Morphological Operations

Morphological operations are considered as a set of operations that process images based on shapes. It is performed normally on binary images. The structuring element is applied to an input image and an output image is generated. They have a wide array of uses.

1) Removing noise.
2) Isolation of individual elements and joining disparate elements in an image.
3) Finding of intensity bumps or holes in an image.



Fig 4.3 Morphological opening

E. Basic Operations

1) *Erosion:* Erosion erodes away the boundaries of foreground object (foreground in white). The kernel slides through the image. A pixel in the original image considered 1 only otherwise it is eroded or if all the pixels under the kernel are 1 considered as original image.
a) It is useful for removing small white noises.
b) Detach two connected objects etc.
2) *Dilation:* Dilation is opposite to erosion. In dilation, a pixel element is considered as '1' if at least one pixel under the kernel is '1'. Therefore the size of white region or foreground object increases.
3) *Opening:* Opening is nothing but erosion followed by dilation. In removing noises, opening operation is used frequently. Because, erosion removes white noise, it also shrinks our project. So we dilate it. It is also useful in joining broken parts of the object.
4) *Closing:* Closing is considered as dilation followed by erosion (i.e) the reverse of opening. They are more useful when the noise is gone. Thus the operation of performing closing is considered to be part of reversing the opening.

F. Contour Extraction

Contour tracing is a technique that is applied to digital images in order to extract their boundary, known as border following or boundary following. The Output of the Thresholding operation will be a noisy image with widespread white pixels. Initially, all the contours in the input binary image were detected and subsequently an area. Mode – Contour retrieval mode is mostly used in detecting the moving object and arguments are based on source image.



Fig 4.4 Contour extraction

1) *Centroid:* The centroid or geometric center of a plane figure is the arithmetic mean position of all the points in the shape. The definition extends to any object in n-dimensional space. Its centroid is the mean position of all the points in all of the coordinate directions. Informally, it is the point at which a cutout of the shape could be perfectly balanced on the tip of a pin.

To compute the centroid (i.e., the center *(x, y)*-coordinates of the object) of contours,
Centroid is given by the relations,

$$C_x = M_{10} / M_{00}$$
$$C_y = M_{01} / M_{00}$$

©IJRASET: All Rights are Reserved

G. Bounding Rectangle

In digital image processing, the bounding box/bounding rectangle is merely theminimum Bounding Rectangles which, are frequently used as an indication of the general position of a geographic feature or dataset, for display, first-approximation spatial query, or spatial indexing purposes.



Fig 4.5 Bounding rectangle

VII. SPEED ESTIMATION

The core technique used in finding the speed of object is from the center of bounding rectangle. In this method, center value of object has been calculated when it enters the frame. And after traversing distance, when the object leaves the frame again we find the center value of bounding rectangle. Two center values of object have been calculated at two different positions. From these values, distance has been calculated from which speed is derived in meter per hour.

1) Experiment 1

a) Speed Estimated from Code

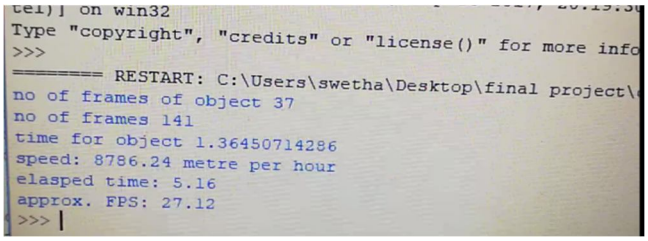


Fig 5.1 Speed estimated

b) Manual Calculation

Distance = 3.3 meters

Time taken by the object to cover that distance =1.36 seconds

Speed = Distance / Time (meter per hour)

Speed = 3.3 / 1.36

= 2.4265 (meter per second)

= 2.4265 * 3600

= 8735.4 (meter per hour)

2) Experiment 2

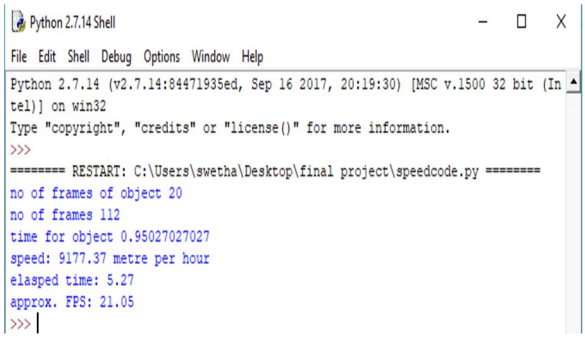


Fig 5.2 Speed estimated

©IJRASET: All Rights are Reserved

a) Manual Calculation

Distance = 2.2 meters
Time taken by the object to cover that distance =0.95 seconds
Speed = Distance / Time (meter per hour)
Speed = 2.2 / 0.95
= 2.3157 (meter per second)
= 2.3157 * 3600
= 8336.8 (meter per hour)

3) Experiment 3

(Object moving diagonally)

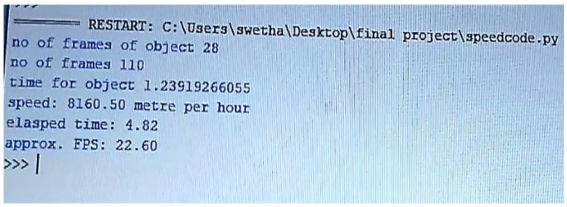


Fig 5.3 Speed estimated

b) Manual Calculation

Distance = 2.75 meters
Time taken by the object to cover that distance =1.23 seconds
Speed = Distance / Time (meter per hour)
Speed = 2.75 / 1.23
= 2.2357 (meter per second)
= 2.2357 * 3600
= 8048.78 (meter per hour)

A. Deque

Deque is implemented on python using the module “collections”. Deque is preferred over where quicker append and pop operations from both the ends of container based on time complexity.

B. Distance

After calculating center value at two position of object, distance traversed by object is calculated by using the distance formula:

$$\boldsymbol{D = \sqrt{(C_{x2} - C_{x1})^2 + (C_{y2} - C_{y1})^2}}$$

C. Pixel To Meter Conversion

In order to find the speed in meter per hour, we convert the pixel values (distance) in to meter.
1 pixel (x) = 0.000264583 meter (m)

D. Time

The time taken by the object is given by the formula,
Time = number of frames covered by the object / frame rate.

1) Frame Rate: Frame rate is expressed in frames per second or fps. It is considered as the frequency (rate) at which consecutive images called frames, which appear on each display. The term applies equally to film and video cameras, computer graphics, and motion capture systems.

E. Speed

Speed is calculated by the mathematical formula as given below,
Speed = Distance / Time (meter per hour)

©IJRASET: All Rights are Reserved

VIII. TABULATION

EXPERIMENT	SPEED OBTAINED FROM CODE (METER PER HOUR)	ACTUAL SPEED (METER PER HOUR)	NO OF FRAMES	ERROR (%)
1	8786.24	8735.4	37	0.58 %
2	9177.37	8336.8	20	7.08 %
3	8160.50	8048.78	28	1.38 %
4	7694.47	8244.27	30	6.66 %

Table 5.1 Results and Evaluation

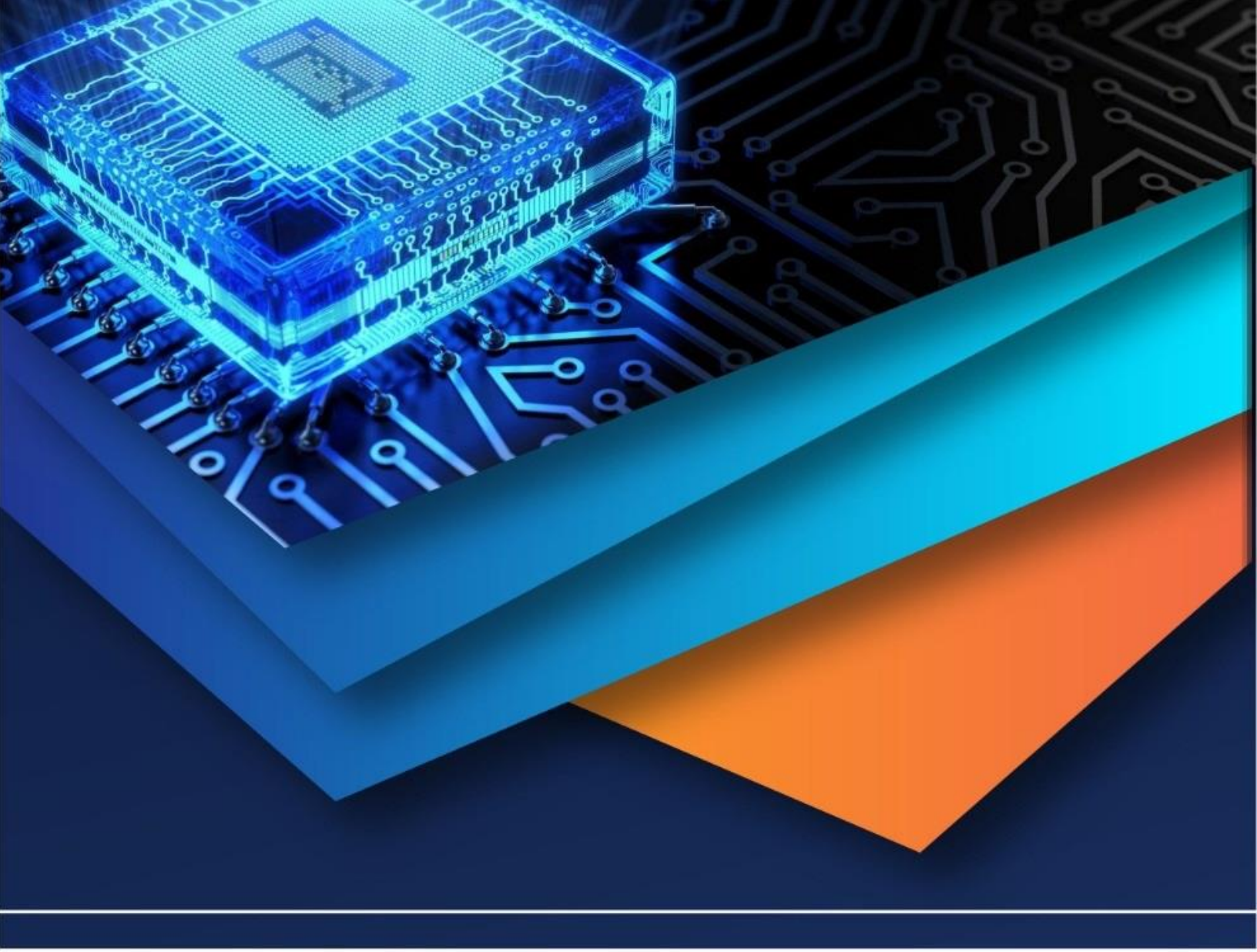
IX. CONCLUSION

An efficient image processing approach has been suggested and analyzed for estimation of speed of vehicle using centroid method, which is good alternative to the traditional radar system.Many of the system developed till date but no one has focused their best suitable based on efficiency, reliability and accuracy. The accuracy of the system proposed in the speed measurement is nearly 90 percent comparable to the actual speed of the moving vehicles, as the error rate is very low. We hope that this system will find extensive application in real time traffic management field and many other purposes.

REFERENCES

[1] Khan, I. Ansari, M. S. Z. Sarker, and S. Rayamajhi, “Speed estimation of vehicle in intelligent traffic surveillance system using video image processing,” International Journal of Scientific & Engineering Research, vol. 5, no. 12, pp. 1384–1390, 2014.
[2] B. C. Putra, “Moving vehicle classification with fuzzy logic based on image processing,” Master’s thesis, SepuluhNopember Institute of Technology, 2016.
[3] B. Setiyono, D. R. Sulistyaningrum, I. Mukhlash, and R. A. J. Firdaus, “A new approach algorithm for counting of vehicles moving based on image processing,” International Journal of Computer Science and Information Security, vol. 14, no. 10, pp. 366–370, 2016.
[4] DanangWahyuWicaksono and Budi Setiyono, “Speed Estimation On Moving Vehicle Based On Digital Image Processing”, International Journal of Computing Science and Applied Mathematics, vol. 3, no. 1, February 2017.
[5] DolleyShukla and Ekta Patel,” Speed Determination of Moving Vehicles using LucasKanade Algorithm”, ShriShankaracharya College of Engg.& Tech. Junwani, Bhilai-490020 Durg,(CG), India, International Journal of Computer Applications Technology and Research Volume 2– Issue 1, 32-36, 2013, ISSN: 2319–8656.
[6] Diogo Carbonera Luvizon, BogdanTomoyukiNassu, and Rodrigo Minetto “A Video-Based System for Vehicle Speed Measurement in Urban Roadways” IEEE transactions on intelligent transportation systems, vol. 18, no. 6, june 2017
[7] HARDY SANTOSA SUNDORO, AGUS HARJOKO,” Vehicle Counting and Vehicle Speed Measurement based on Video Processing”, Department of Computer Science and Electronics UniversitasGadjahMada, Indonesia, Journal of Theoretical and Applied Information Technology 20th February 2016. Vol.84. No.2.
[8] H. Sundoro and A. Harjoko, “Vehicle counting and vehicle speed measurement based on video processing,” Journal of Theoretical and Applied Information Technology, vol. 84, no. 2, pp. 233–241, 2016.
[9] Pratishtha Gupta and G N Purohit and ManishaRathore,” Estimating Speed of Vehicle using Centroid Method in MATLAB”, Banasthali University Jaipur, India, International Journal of Computer Applications (0975 – 8887) Volume 102– No.14, September 2014
[10] SiddharthJhumat,” Vehicle Speed Estimation in Accident Prone Areas using Image Processing”, Department of Computer Science & Engineering, University School of Information & Communication Technology, Guru Gobind Singh Indraprastha University, Delhi, India, International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 5, May 2014
[11] Sumer Jabri, ZoranDuric, Harry Wechsler, AzrielRosenfeld, “Detection and Location of People in VideoImages Using Adaptive Fusion of Color and EdgeInformation,” In Proc. 15th Int″l Conf. on Pattern Reg,2000,vol. 4,pp. 627 – 630.

©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)