SMART AUTOMOBILE PARKING SYSTEM USING BLOB ANALYSIS AND MEASUREMENT TECHNIQUES

A Project

Report

Submitted in the partial fulfillment of the requirements for

the award of the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

K. K. S. KARTHIKEYA (190040197)

K. CHARANKANTH REDDY (190040227)

K. NAGASAI DWARAKA (190040265)

under the supervision of

Dr. S Koteswara Rao

PROFESSOR

Department of Electronics and Communication Engineering

KLEF, Green Fields,

Vaddeswaram- 522502, Guntur(Dist), Andhra Pradesh, India.

November 2022.

Declaration

The Project Report entitled “**Smart Automobile Parking System Using Blob Analysis and Measurement Techniques** “is a record of the bonafide work of K. Kushal Sai Karthikeya (190040197), K. Charankanth Reddy (190040227), K. Nagasai Dwaraka (190040265) submitted in partial fulfillment for the award of B. Tech in Electronics and Communication Engineering to the K L University. The results embodied in this report have not been copied from any other departments/University/Institute.

K. K. S. KARTHIKEYA (190040197)

K. CHARANKANTH REDDY (190040227)

K. NAGASAI DWARAKA (190040265)

Certificate

This is to certify that the Project Report entitled "**Smart Automobile Parking System Using Blob Analysis and Measurement Techniques**" is being submitted by K. Kushal Sai Karthikeya (190040197), K. Charankanth Reddy (190040227), K. Nagasai Dwaraka (190040265) submitted in partial fulfillment for the award of B. Tech in Electronics and Communication Engineering to the K L University is a record of bonafide work carried out under our guidance and supervision. The results embodied in this report have not been copied from any other departments/ University/Institute.

Signature of the Supervisor

Name : Dr.S Koteswara Rao

Designation: PROFESSOR

Signature of the HOD

Signature of the External Examiner

ACKNOWLEDGEMENT

Our sincere thanks to **Dr. S Koteswara Rao**, Professor, ECE Department, in this project for his outstanding support throughout the project for the successful completion of the work.

We express our gratitude to **Dr. Suman Maloji**, Head of the Department for Electronics and Communication Engineering for providing us with adequate facilities, ways, and means by which we have been able to complete the work.

We would like to place on record a deep sense of gratitude to the honorable Vice Chancellor, K L Deemed to be University for providing the necessary facilities to carry out this project.

Last but not the least, we thank all teaching and non-teaching staff of our department for their support in the completion of our project.

ABSTRACT

With the increment in the total number of vehicles due to the rapid expansion of major cities and the construction of huge buildings and parking areas, there is an obligation in improving the present parking system. This paper aims to current a smart parking system solution using image processing techniques. A camera can be used as a sensor that captures different images at different points in time for proper analysis and contains information about the free car's parking space. In the existing system, there are many troubles in finding a space The main problem is finding a faster space in a multistory parking slot. Especially, on weekends or public occasions. Insufficient parking space activity might block the driver. This automatic parking system makes the whole process of parking cars more efficient and less complex for both drivers and administrations.

INDEX

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

One of the worst things which can happen to any individual is the long waiting time for a single parking slot. Nowadays, Parking is a frustrating job that involves an abundance of time, money, and effort spent chasing a free slot. According to a study that was done over a long duration of time by the Centre for Science and Environment, an Indian car is parked for an average of 95% of the time and is driven for less than 5%. It also reveals that one particular car spends over 8,293 hours standing and is only driven for 467 hours in a year. The main problem is the natural increase of vehicle owners globally in a cruise for parking i.e., Vehicles looking for a parking space resulting in longest queues, overcrowding, and pollution which arises due to a lack of enough parking spaces in Indian cities cruising. Low pricing on-street parking results in fewer parking areas for hours and instances have been recorded of cars being parked for weeks. Statistics say that on average over 30% of traffic is caused by drivers wandering around for a parking area. According to recent studies, a commuter in New Delhi spends over 80 hours annually searching for a parking area. The parking issues in India are worsening with every passing year. The situation in India, where the land is limited and at the same time it is expensive, especially in metropolitan cities like Mumbai, New Delhi, Chennai, Kolkata, Hyderabad, etc., rising parking area demand increases immense pressure and advancement in the parking management system is much needed. For instance, in New York, the midtown road area per person stands at about 33.3 sqm while at the same time it is no more than 1.7 sqm in the case of Mumbai's Null Bazaar. Which clearly states that automobiles in Mumbai inflict a cost nearly 20 times as much as one in New York. Now the actual question arises Is increasing parking slots a solution or optimization and automation of parking management a solution? The pace at which the registration of new cars is taking place which always creates a barrier to a meeting of stable demand and supply of parking slots. Thus, Increasing the number of parking slots is never the solution. As we can see in metropolitan cities where the blockage of traffic has become ordinary due to the rapid increment in the total number of vehicles. The reality of cars in our daily life is an undeniable fact. The improper alignment of vehicles results in unoccupied divisions in parking areas even when the total occupancy is high. The present-day parking areas are incapable of directing the customer with proper directions and signs when he enters a parking slot along with additional information such as whether the parking lot is fully occupied, partly occupied, or vacant. This results in a lot of waste of time and fuel as a customer is unaware of how many empty parking areas are present and where to find a parking space. Having pre-defined knowledge about the total no. of vacant parking slots available in the parking area which can be performed using image processing techniques helps us to resolve the problem at a low cost. The system captures images from a camera fixed at top of the sealing and these captured images further undergo filtering and pre-processing techniques such that it removes all the noise and then performs our required image processing technique to count the vacant parking slots. The entire system undergoes different stages to achieve our desired output which involves planning, analyzing, designing, developing, and testing phases. In each phase of the methodology, we use different kinds of image processing techniques. This system is more compactable compared to the existing system with a minimum implementation cost. One of the best things that happened due to the digital era is the rapid increment of digital cameras and image processing techniques. As Image Processing techniques are increased now, we can perform high-quality image processing and manipulate different aspects of the process. Due to the shooting of fast processing speed computers and decreased cost for memory management, the field of image processing become efficient and experienced growth.

1.1 Digital Image Processing

Digital image processing is an advanced technology that enables you to manipulate digital images through computer software. It is a branch of signal processing that largely focuses on visuals. Using a digital image as an input and a variety of algorithms to produce an output is possible with digital image processing. Depending on the final image intended, these algorithms may change from image to image.

One of the main advantages of routinely integrating is that faults are easier to find and can be found more rapidly. Finding the specific modification that caused a defect is frequently accomplished fast because each change introduced is typically modest.

There are three primary phases that constitute image processing Utilizing picture acquisition techniques to import the image, processing, and handling of images and output that allows for the modification of an image or report based on image analysis.

1.2 Classification of Digital Image Processing

The digital image uses the spectral information represented by the digital numbers in one or more spectral bands which classify each individual pixel based on this spectral information. Also known as spectral pattern recognition.

1.2.1 Image Acquisition

In Image processing, Image Acquisition is referred to as the action of obtaining an image from a source, typically a hardware-based source, for processing in image processing. It is the first step in the workflow sequence because processing cannot take place without an image. The captured image is entirely unprocessed.

Now, the combination of input electrical power and sensor material that responds to a specific sort of energy being sensed converts the incoming energy into a voltage. The response of the sensor is represented by the output voltage waveform, and each sensor's response is digitized to produce a digital quantity.

There are 3 different kinds of Image acquisition techniques:

1. Image acquisition using a single sensor

2. Image acquisition using a line sensor

3. Image acquisition using an array sensor

1. Image acquisition using a single sensor: The motion must be in both the x and y directions in order to produce a two-dimensional image using a single sensor. Rotation provides motion in one direction. Linear motion provides motion in the perpendicular direction. The best example of an Image acquisition using a single sensor is a photodiode. This is an inexpensive method, we can get high-resolution images with very precise control. The drawback of this approach is that it is cumbersome.

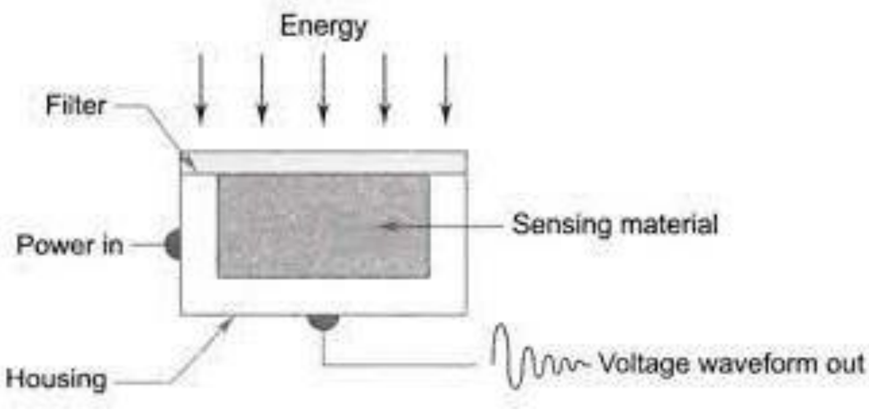

Figure 1. Image Acquisition using a single sensor

Image acquisition using a line sensor: A-line scan sensor captures an image of a moving object in one-pixel "slices" that are later put together into a 2D image. The sensor strip provides imaging in one direction. Motion perpendicular to the strip provides imaging in another direction.

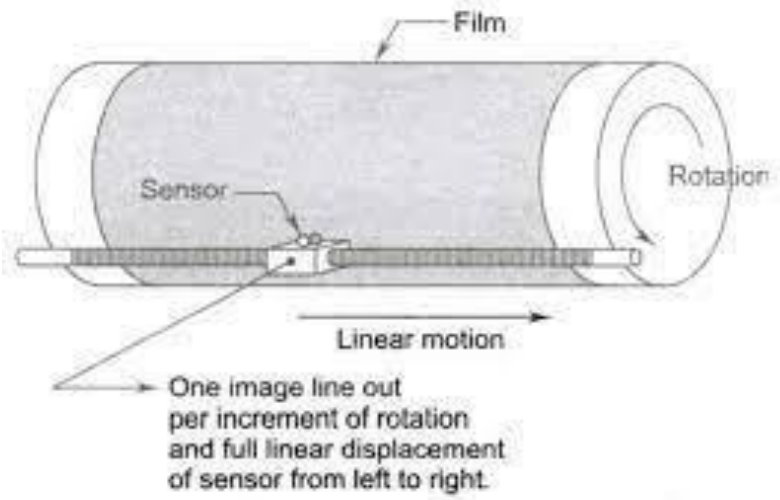

Figure 2. Image acquisition using a line sensor

Image acquisition using an array sensor: Image acquisition using an array of sensors where individual sensors are stacked in a 2-D array. Such a setup is observed in digital cameras as a CCD array. Where CCD stands for the charged coupled device. Each sensor's reaction in this is proportional to the integral of the projected light energy onto the sensor's surface. The sensor is allowed to combine the data to reduce noise. The sensor array gives an output proportional to the integral of light received at each sensor, which is coincident with the focus plane. These outputs are swept by digital and analog circuitry, which transforms them into a video signal that is subsequently processed by another part of the imaging system. Finally, A digital image is obtained after fine processing.

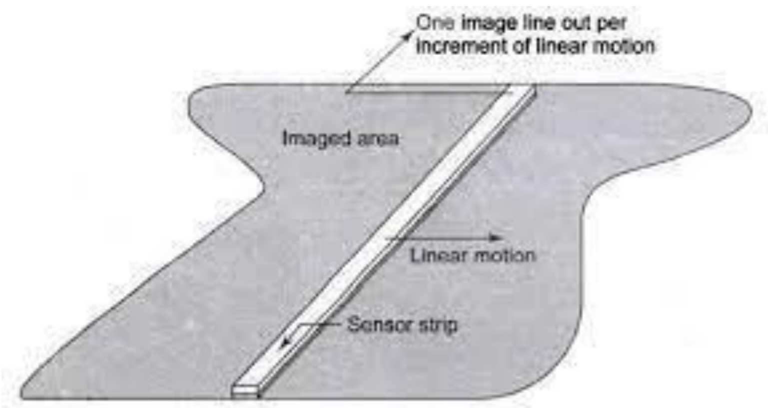

Figure 3. Image acquisition using an array sensor

1.2.2 Thresholding

Thresholding is a type of image segmentation, where we change the pixels of an image to make the image easier to analyze. In thresholding, we convert an image from color or grayscale into a binary image, i.e., one that is simply black and white. The simplest method for separating an image’s foreground from the background is image thresholding. We should utilize a grayscale image while applying thresholding algorithms. That grayscale image will be changed into a binary image during thresholding.

Different types of thresholding include simple thresholding, adaptive thresholding, and automatic image thresholding.

1. Simple thresholding:

Simple thresholding assigns a standard value to all pixel values that are higher than the chosen threshold value. Pixel values are compared to a unique threshold value. We may view the segmented photos based on threshold values after pixel separation. Based on the threshold operator, threshold approaches are usually split into three types global, local and dynamic.

Where the global threshold operator depends on the gray values of the pixel, the local threshold operator depends on the gray values of the pixel and local properties and the dynamic threshold operator depends on the gray values of the pixel and local properties and its position.

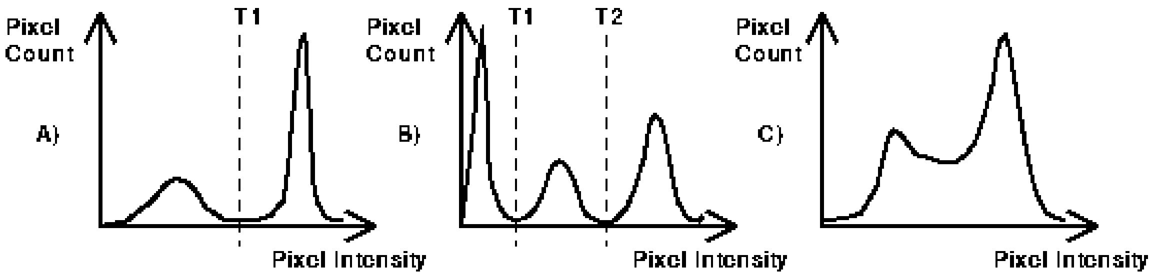

Figure 4. Simple Thresholding

2. Adaptive thresholding:

Like global thresholding, adaptive thresholding is used to separate desirable foreground image objects from the background based on the difference in pixel intensities of each region. In Adaptive thresholding, the input image will be segmented into small areas. These areas are non-overlapping areas. Different threshold values are used for each image segment. Threshold values depend on the locations of pixel values because each image segment has different light areas.

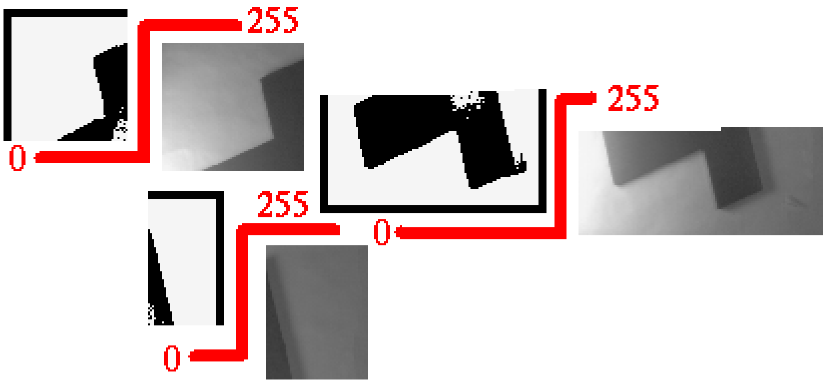

Figure 5. Adaptive Thresholding of an image

3. Automatic Image thresholding:

The automatic optimal Image thresholding (or) OTSU method is a global adaptive binarization threshold image segmentation algorithm. This algorithm takes the maximum inter-class variance between the background and the target image as the threshold selection rule.

This "Automatic Optimal Threshold Detection" method uses the image's histogram to determine the ideal global threshold value. The ideal threshold setting should fall between the two peaks of an image histogram. According to the threshold setting, the image will be divided into the foreground and background. The Otsu technique has two categories, 1-D and 2-D. The 2-D method is more efficient than the 1-D method.

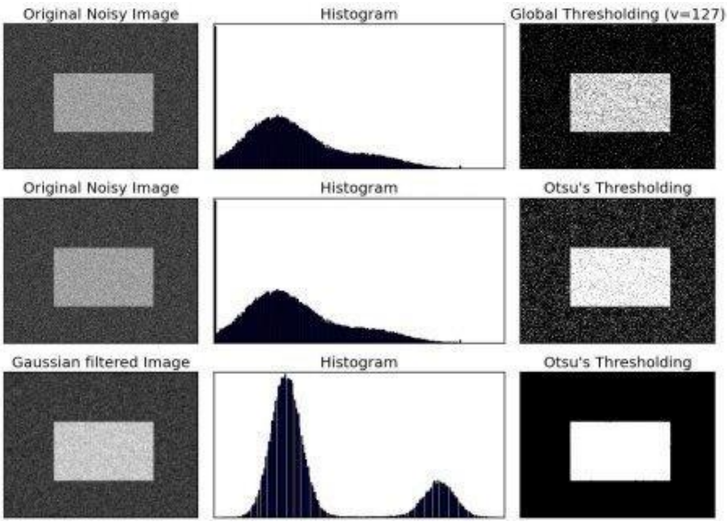

Figure 6. Automatic Image Thresholding (OTSU method).

1.2.3 Image Enhancement

The alteration of an image in order to change its impact on the viewer is known as an enhancement. Enhancement is generally prohibited until the restoration steps are finished since it can damage the original digital data.

There are three different types of Image Enhancement techniques Contrast Enhancement, Density Slicing, and Edge Enhancement.

1. Contrast Enhancement: In order to make things in the scene more visible, contrast enhancement techniques change the relative brightness and darkness of the objects in the scene. By using a grey-level transform to map the image's grey levels to new values, it is possible to alter the image's contrast and tone.

2. Density Slicing: Density slicing is a method for breaking up an image's continuous grey tone into slices with varying densities, each of which corresponds to a different digital range. Each slice is shown in a different color.

Figure 7. Thresholding and Density Slicing

3. Edge Enhancement: Edge enhancement is an image processing filter that enhances the edge contrast of an image to improve its apparent sharpness. The result of this feature is that any edges in the image have subtly bright and dark highlights on either side of them, giving the edge a sharper appearance.

An image's edge is a fundamental component. By identifying sharp discontinuities in pixel values, edges define the boundaries between regions in an image, aiding in segmentation and object identification.

1.2.4 Image Restoration

The process of restoring an image from a degraded copy typically a blurry and noisy image is known as image restoration. An important issue in image processing is picture restoration, which also serves as a test case for more complicated inverse issues. The quality of the restored image, the algorithm's computational effectiveness, and the estimate of important parameters like the point-spread function are the main concerns that need to be addressed (PSF). In order to shed light on the nature of the issue, fundamental picture restoration techniques are reviewed. Additionally, these methods offer solutions for fuzzy image deblurring that are efficient and manageable in terms of computing complexity.

There are numerous techniques for restoring images, including the blind deconvolution method, the Weiner filter, the limited least squares filter, and an inverse filter. Some techniques, both linear and non-linear, aid in removing blur and noise from the image.

1.2.5 Color Image Processing

When color pictures are processed, an abstract mathematical model called color space is used to explain the colors in terms of intensity levels. This color space uses a three-dimensional coordinate system. For varied applications, there are many various color spaces.

Color image processing is the analysis, manipulation, and interpretation of color-presented visual data. It can produce a range of results, such as the grayscale conversion of a black-and-white image or a careful analysis of the information contained in a telescope image. A variety of programs may be employed, in addition to manual chores carried out by programmers, to process digital photos. Research and development activities in this area are now being carried out at both academic institutions and places owned by private businesses.

There are several uses for color image processing. When a picture is actually recorded, saved, or committed to film due to an error during image acquisition, it may be essential to interpret the image in some way in order to extract meaningful information. For instance, a computer must interpret the results of magnetic resonance imaging (MRI) and provide them to the user in a visual format. In order to boost contrast and clarity, the application also offers the capability to color-code particular scan zones.

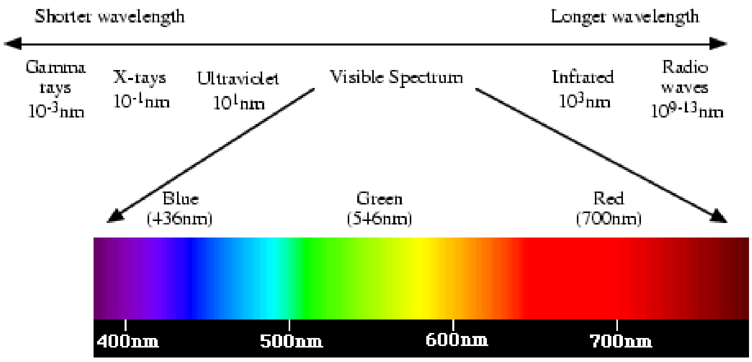

Figure 8. Electroscopic Spectrum

There are two main reasons why color is used in image processing. In the first place, color is a strong identifier that typically makes it simpler to recognize and remove objects from a scene. The two primary subfields of color image processing are full-color and pseudo-color processing. Frequently, a full-color sensor, such as one found in a color TV camera or a color scanner, was used to take the photographs in question in the first set. Choosing which color to use for a particular monochrome intensity or range of intensities is a challenge in the second category. Up until recently, the majority of digital color image processing was done at the pseudo-color level. On the other hand, reasonably affordable color sensors and processing technology have been available over the past ten years. As a result, full color.

1.2.6 Wavelets and Multi-Resolution techniques

An Image signal is broken down into several frequency components using the wavelet transform at various resolution scales as multiresolution scales. This enables the simultaneous disclosure of the spatial and frequency aspects of an image. Additionally, something may be evident at one resolution but obscure at another.

Wavelets are functions that are centered around a certain place in both time and frequency. This transformation technique overcomes the drawbacks of the Fourier method. The Fourier transformation deals with frequencies but does not provide any temporal information. Wavelets are waveforms that have a nonzero norm and a zero average value, effectively reducing their duration. Transients frequently interrupt the piecewise smooth behavior of many fascinating signals and images.

By viewing compressed representations of the original array, the multiresolution approach enables decisions that would otherwise be prevented by the quantity of data available. Multiresolution analysis is a technique that involves expressing a signal in more than one resolution or scale. Features that could go undetected at one resolution may be simple to spot at another. A signal is split into pieces in multiresolution analysis, which when put back together produce the exact same signal. For a signal to be useful for data processing, how it is separated is critical.

1.2.7 Compression

Image compression is a method for reducing the number of bytes in a graphics file without sacrificing the quality of the images. By reducing the file size, more photos can be stored on a given quantity of discs or memory.

The most widely used compression method, JPEG, has the propensity to muddle the appearance of lines, particularly at low resolutions. Lossless image compression techniques, in contrast, maintain the original quality while only partially compressing the image. Examples include the file types TIFF, GIF, and PNG.

Image compression seeks to reduce the superfluous and unnecessary elements of the image data to make it more effective for storage or transmission. The objective is to represent an image using the fewest number of bits possible. There are two types of picture compression: lossy and lossless.

1.2.8 Morphological Processing

Morphology, a broad category of image processing methods, transforms images based on shapes. Morphological processes transform an input image into an output image of the same size by adding a structuring feature. The morphological concepts constitute a useful set of instruments for spotting intriguing details in an image. Implementation benefits from erosion and dilation's simplicity are significant.

A variety of image processing methods called morphological operations are used to process digital images based on their forms. In a morphological process, each image pixel corresponds to the value of other pixels in the surrounding area. Morphological image processing is a collection of non-linear operations that deal with the appearance or morphology of features in an image.

1.2.9 Segmentation

Image segmentation is a technique for breaking up a digital image into smaller groupings called image segments, which reduces the complexity of the image and makes each segment more easily processed or analyzed. Technically, segmentation is the process of giving labels to pixels in an image in order to distinguish between objects, persons, or other significant aspects.

Object detection is a frequent use of image segmentation. It is usual practice to first apply an image segmentation method to discover things of interest in the image before processing the complete image. The object detector can then work with a bounding box that the segmentation algorithm has previously established. By stopping the detector from processing the entire image, accuracy is increased and inference time is decreased. A crucial component of computer vision technologies and algorithms is image segmentation. It is employed in a variety of real-world contexts, including as face identification and recognition in video surveillance, medical image analysis, computer vision for autonomous cars, and satellite image analysis.

There are three different types of Segmentation such as semantic segmentation, Instance segmentation, and panoptic segmentation.

1. Semantic segmentation:

Semantic segmentation involves arranging the pixels in an image based on semantic classes. In this model, every pixel belongs to a specific class, and the segmentation model does not refer to any other context or information.

For instance, doing semantic segmentation on an image containing numerous trees and vehicles will provide a mask that groups all tree kinds into a single-class tree and all vehicle types, such as buses, cars, and bicycles, into single-class vehicles.

2. Instance segmentation:

Classifying pixels based on instances of an item is known as instance segmentation. Instance segmentation methods instead divide comparable or overlapping regions based on the boundaries of objects rather than knowing which class each region belongs to.

Consider how an instance segmentation model might analyze a picture of a busy street. In an ideal scenario, it should count the instances of each object in the image and find them among the crowd of people. The region or item cannot be predicted for every case.

3. Panoptic segmentation:

A more recent type of segmentation called panoptic segmentation frequently combines instance and semantic segmentation. It distinguishes between each occurrence of each thing in the image by predicting the identification of each object. For many goods that need a lot of information to function, panoptic segmentation is helpful. For instance, self-driving automobiles must be able to precisely and swiftly collect and comprehend their environment. They can do this by providing a panoptic segmentation algorithm and a live stream of images.

1.3 Noise in an Image

Image noise is a random variation of brightness or color information in the images captured. It is degradation in image signal caused by external sources. Images containing multiplicative noise have the characteristic that the brighter the area the noisier it. But mostly it is addictive.

$$A(x,y) = H(x,y) + B(x,y)$$

Where, A(x,y)= function of the noisy image, H(x,y)= function of image noise, and B(x,y)= function of the original image. Major Sources of Image noise are included While the image is being sent electronically from one place to another, Sensor heat while clicking an image, and varying ISO Factor which varies with the capacity of the camera to absorb light.

There are different types of image noise. They can typically be divided into 3 types.

1.3.1 Gaussian Noise

Gaussian Noise is a statistical noise having a probability density function equal to a normal distribution, also known as Gaussian Distribution. A random Gaussian function is added to the Image function to generate this noise. It is also called electronic noise because it arises in amplifiers or detectors. Source: thermal vibration of atoms and discrete nature of radiation of warm objects.

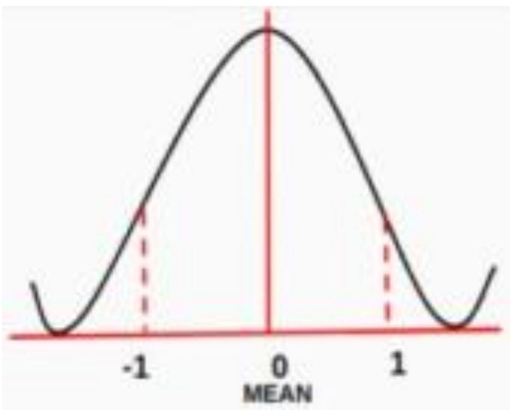

Figure 9. Gaussian distribution of noise in an input image

The side image is a bell-shaped probability distribution function that has a mean of and a standard deviation(sigma) of 1.

1.3.2 Impulse Noise:

In the discrete world impulse function is on a value of 1 at a single location and in the continuous world impulse function is an idealized function having a unit area.

There are three types of impulse noises Salt Noise, Pepper Noise, and Salt and Pepper Noise.

1. Salt Noise: Salt noise is added to an image by the addition of random brightness (with a 255-pixel value) all over the image.

2. Pepper Noise: Salt noise is added to an image by the addition of random dark (with a 0-pixel value) all over the image.

3. Salt and Pepper Noise: Salt and Pepper noise is added to an image by the addition of both random bright (with 255-pixel value) and random dark (with 0-pixel value) all over the image. This model is also known as data drop noise because statistically, it drops the original data values. Source: Malfunctioning of camera's sensor cell.

1.3.3 Poisson Noise:

The appearance of this noise is seen due to the statistical nature of electromagnetic waves such as x-rays, visible lights, and gamma rays. The x-ray and gamma-ray sources emitted a number of photons per unit of time. These rays are injected into the patient's body from their source, in medical x-rays and gamma rays imaging systems. These sources are having random fluctuation of photons. Result gathered image has spatial and temporal randomness. This noise is also called quantum (photon) noise or shot noise.

CHAPTER 2

LITERATURE SURVEY

“2.1 S. Khantasak, N. Jindapetch, P. Hoyingcharoen, K. Chetpattananondh, M. Ikura and S. Chumpol, "Parking Violation Detection System based on Video processing," ***2018 IEEE 5th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA),*** **2018, pp. 1-5, doi: 10.1109/ICSIMA.2018.8688790.”**

This paper presents a parking violation detection system based on video processing. The proposed system captures and processes the images from a parking violation area, and also identifies the case of no vehicles in the parking area. In this paper, a camera is used as a detector to take videos to determine the violation. Moving vehicles are also detected. The process for this is based on the background subtraction method. With the proposed technique in this system module, we can well detect the vehicles which violate the parking zone regulations.

“2.2 J. Liu, M. Mohandes and M. Deriche, "A multi-classifier image based vacant parking detection system," ***2013 IEEE 20th International Conference on Electronics, Circuits, and Systems (ICECS),*** **2013, pp. 933-936, doi: 10.1109/ICECS.2013.6815565.”**

With the rapid expansion of major cities, the tremendous increase in the number of vehicles, and the construction of huge buildings and parking areas, there is a need to develop smart parking substantial research efforts in developed countries. There are mainly four categories of car parking management systems: counter-based, wired-sensor-based, wireless-sensorbased, and image-based. In this paper, we develop, implement, and test an image-based system for the detection of vacant spaces in a parking area.

“2.3 A. Sofwan, M. S. Hariyanto, A. Hidayatno, E. Handoyo, M. Arfan and M. Somantri, "Design of Smart Open Parking Using Background Subtraction in the IoT Architecture," ***2018 2nd International Conference on Electrical Engineering and Informatics (ICon EEI)*****, 2018, pp. 7-11, doi: 10.1109/ICon-EEI.2018.8784334.”**

The Internet of Things (IoT) has evolved and penetrated to our live since the end of the last century. Nowadays, many devices for any purpose are connected through the Internet. A smart node, in smart campus environment, can detect an availability of an open parking space by calculating the vehicle that enters or outs from the space. The node applies a background subtraction method, which is deployed in IoT architecture.

“ 2.4 A. O. Kotb, Y. -C. Shen, X. Zhu and Y. Huang, "iParker—A New Smart Car-Parking System Based on Dynamic Resource Allocation and Pricing," in ***IEEE Transactions on Intelligent Transportation Systems*****, vol. 17, no. 9, pp. 2637-2647, Sept. 2016, doi: 10.1109/TITS.2016.2531636.”**

Parking in major cities, particularly with dense traffic, directly effects the traffic flow and people’s life. In this paper, we introduce a new smart parking system that is based on intelligent resource allocation, reservation, and pricing. The proposed system solves the current parking problems by offering guaranteed parking reservations with the lowest possible cost and searching time for drivers and the highest revenue and resource utilization for parking managers. New fair pricing policies are also proposed that can be implemented in practice.

“ 2.5 S. Tuteja, S. Poddar, D. Agrawal and V. Karar, "PredictV: A Vehicle Prediction Scheme to Circumvent Occluded Frames," in ***IEEE Access*****, vol. 10, pp. 20029-20042, 2022, doi: 10.1109/ACCESS.2022.3151973.”**

Although several methods to overcome traffic congestion on the roads exist, occlusion is still a major bottleneck in processing the traffic images and need researchers attention. In the rural and urban areas, heavy congestion on the roads has become the leading cause of occlusion. The PredictV method works on the prediction principle based on existing values of blob width and height. This scheme uses blob detection for the first frame and predict other vehicles based on percentage increment in the different blob parameters to locate the vehicle in the next frame. With the help of this approach, occlusion is handled easily and it prefers to use predicted points to detect vehicles on the frame. The suggested approach is implemented via a MATLAB simulator.

“ 2.6 M. Dixit, C. Srimathi, R. Doss, S. Loke and M. A. Saleemdurai, "Smart Parking with Computer Vision and IoT Technology," ***2020 43rd International Conference on Telecommunications and Signal Processing (TSP)*****, 2020, pp. 170-174, doi: 10.1109/TSP49548.2020.9163467.”**

Some elemental problems faced by the parking lots in large cities include the difficulty in locating a free parking spot, security of the parked vehicle as well as people parking in a reserved parking spot. In this paper we propose a blended use of the mobile application, computer vision and IoT technologies to counter these problems and if it is implemented, it will surely save some valuable time. We will also be able to guarantee the security of the parked vehicle using automatic security bollards. We will be using Node MCU as a microcontroller and ultrasonic sensors as proximity sensors. We will also be using CCTV camera live footage for verifying readings from the IoT devices to eliminate all the false positives. At all times the system will display the live status of the parking spaces in the parking lots to all the users of the mobile application.

“ 2.7 H. Kong, H. C. Akakin and S. E. Sarma, "A Generalized Laplacian of Gaussian Filter for Blob Detection and Its Applications," in ***IEEE Transactions on Cybernetics*****, vol. 43, no. 6, pp. 1719-1733, Dec. 2013, doi: 10.1109/TSMCB.2012.2228639.”**

In this paper, we propose a generalized Laplacian of Gaussian (LoG) (gLoG) filter for detecting general elliptical blob structures in images. The gLoG filter can not only accurately locate the blob centers but also estimate the scales, shapes, and orientations of the detected blobs. These functions can be realized by generalizing the common 3-D LoG scale-space blob detector to a 5-D gLoG scale-space one, where the five parameters are image-domain coordinates (x, y), scales (σx, σy), and orientation (θ), respectively.

“ 2.8 J. Lin, S. -Y. Chen, C. -Y. Chang and G. Chen, "SPA: Smart Parking Algorithm Based on Driver Behavior and Parking Traffic Predictions," in *IEEE Access*, vol. 7, pp. 34275-34288, 2019, doi: 10.1109/ACCESS.2019.2904972.”

Smart parking problems have received much attention in recent years. In literature, many smart parking allocation algorithms that considered the parking grid reservation and recommendation have been proposed. However, the parking policies for maximizing parking rate and benefits still can be improved. This paper proposes a smart parking allocation algorithm (SPA), which aims to maximize the benefits created by a given parking lot while guaranteeing the quality of parking services. The proposed SPA algorithm predicts the driver behavior and estimated parking traffic in the near future based on the historical parking records. These predictions help SPA to better match the parking demands and the resource of available parking grids and, hence, improve the utilization and the created benefit of each parking grid.

“ 2.9 J. Zuo, Z. Jia, J. Yang and N. Kasabov, "Moving Target Detection Based on Improved Gaussian Mixture Background Subtraction in Video Images," in *IEEE Access*, vol. 7, pp. 152612-152623, 2019, doi: 10.1109/ACCESS.2019.2946230.”

In recent years, background subtraction techniques have been used in vision and image applications for moving target detection. However, most methods cannot provide fine results due to dynamic backgrounds, noise, etc. The Gaussian mixture model (GMM) is a background modeling method commonly used in moving target detection. The traditional GMM method is vulnerable to noise interference, especially from dynamic backgrounds; thus, its detection performance is not good. Because of the influence of background noise and dynamic effects on moving target detection, we propose a method of moving target detection for dynamic backgrounds based on improved GMM background subtraction.

“ 2.10 T. Huynh-The, C. -H. Hua, N. A. Tu and D. -S. Kim, "Locally Statistical Dual-Mode Background Subtraction Approach," in *IEEE Access*, vol. 7, pp. 9769-9782, 2019, doi: 10.1109/ACCESS.2019.2891084.”

Due to the variety of background model in the real world, detecting changes in a video cannot be addressed exhaustively by a simple background subtraction method, especially with several motion detection challenges, such as dynamic background, camera jitter, intermittent object motion, and so on. In this paper, we propose an efficient background subtraction method, namely locally statistical dual-mode (LSD), for detecting moving objects in video-based surveillance systems. The method includes a local intensity pattern comparison algorithm for foreground segmentation by analyzing the homogeneity of intensity patterns of the input frame and the background model, in which the homogeneity is calculated by the mean and standard deviation of pixel intensity. Besides that, a dual-mode scheme is developed to temporally update the background model for the short- and long-term scenarios corresponding to sudden and gradual changes in the background.

CHAPTER 3

THEORETICAL ANALYSIS

The proposed system is initialized with a live stream video of the parking lot from the camera where the system starts with receiving the first frame image and stores it as the fixed background image. Different images are captured and extracted from a live-stream video when any particular car enters the parking lot and the system extracts pixels from the current image as the foreground by the difference between the fixed background image and the current frame A thresholding operation is performed to extract the features from a foreground image, which is also useful for obtaining an updateable background image. These RGB images are converted into binary images where the updateable background image is updated every 50 frames.

The system utilizes the convergence operation of two differential results in order to reduce foreground aperture due to the vehicle's speed and the frame rate. In order to remove noise caused by the camera noise, the reflectance noise, short noise, digital noise, shadow, and the background-colored object noise we apply erosion and dilation operations onto a foreground pixel map. Then the blob analysis technique is employed in accordance with the foreground image and background image. This blob analysis operation obtains a frequency difference between foreground and background images by background subtraction method along with blob analysis. As the proposed methodology is platform dependent the frames are cropped lane-wise and considered sequentially and individually in a loop and implement blob measurement techniques where the total number of vacant slots available with their respective lane is obtained. Now the tokenization system can navigate and provide the best possible parking slot based on the shortest distance and convenience to the customer.

In the existing system, there are many troubles in finding an empty space. Especially, In big malls in metropolitan cities. The main problem is to find a faster empty space in a multistory parking slot on weekends or on public occasions. Exploring space for parking on weekends may take a long time. Shopping malls are even crowded during peak periods, and also finding more trouble in these areas is a matter of concern to customers. Insufficient parking space activity might block the driver. In this proposed system, we have a camera that is fixed at the top of the parking area where all parked cars or vehicles are visible from a top angle. The camera continuously collects images at different points in time which acts as input for our proposed system. These collected images are generally called color images or RGB images.

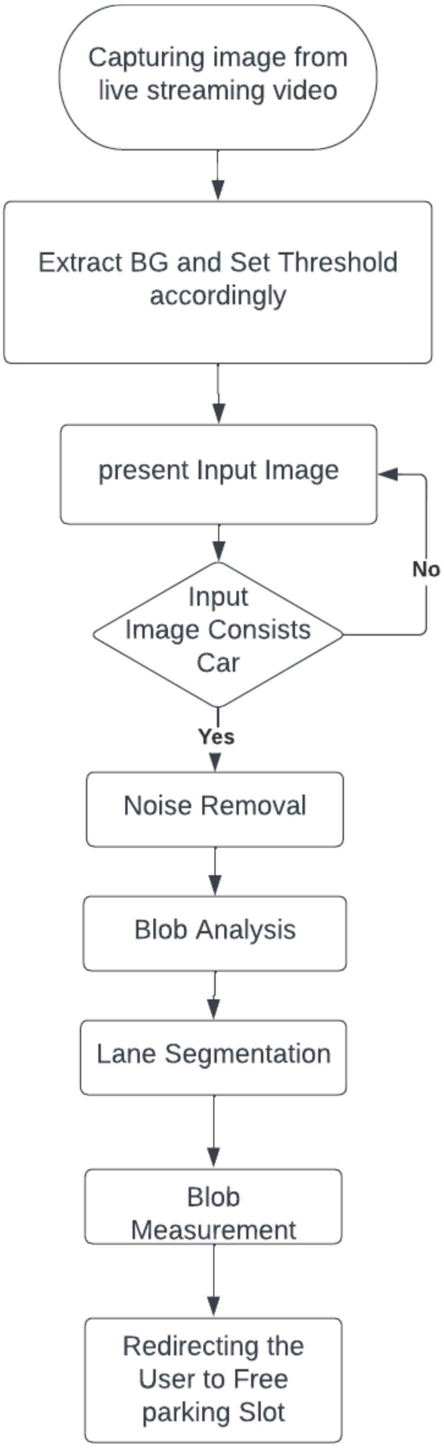

Figure 10. Workflow of the proposed methodology

3.1 Image Acquisition

An RGB image sometimes referred to as a true color image, is stored as an m-by-n-by-3 data array that defines red, green, and blue color components for each individual pixel. The RGB color model is an additive of red, green, and blue light which are added together in varying proportions to produce an extensive range of colors. An RGB image is converted into a grayscale image by eliminating the hue and saturation information while retaining the luminance. Using grayscale images, we can identify each pixel value which is represented only in the intensity information of the light. Such images typically display only the darkest black to the brightest white.

The photometric measurement of the luminous intensity per unit area of light traveling in a given direction is used to convert color-based pixels to grayscale pixels by storing an m-by-n-by-3 data array for each individual pixel which results in a more sophisticated version of the averaging method. It results in a weighted average to account for human perception as it averages the pixel values. Out of the 3 colors in the color model, green is more sensitive to the input image which results in heavier weightage. The luminosity of our color space is

$$Y = 0.296R + 0.587G + 0.112B$$

An empty parking slot image is used as a background image where we perform comparison operations between background and foreground images. Now we perform image thresholding which converts an image from color or grayscale into a binary image. where the binary image is simply black and white. Most frequently, we use thresholding as a way to select areas of interest in an image, while ignoring the parts we are not concerned with. Image thresholding is a simple form of image segmentation. This is typically done in order to separate "object" or foreground pixels from background pixels to aid in image processing.

3.2 Background Subtraction

Background Subtraction helps in finding the average intensity of the background frame which have be initially captured while the initialization of the system thus resulting in a fixed threshold value. This threshold value forms a simple image segmentation and can be applied for background subtraction irrespective of the video and can be adapted for moving object detectors. This process results in the difference in intensity between the current image and the background image. The basic principle is that the first frame image is stored as background image *B(x,y)* and the current image *I(x,y)*. Now, the current image I(x,y) is subtracted from B(x,y). If the difference between the current and background image is higher than the fixed threshold value obtained then it is recognized as a moving object. The accuracy of the Background subtraction technique is more dependent on the accurate threshold value after proper analysis. The false change points result due to low threshold values at the same time high threshold values result in decreasing the scope of change in the movement of the object. As there are many things that impact the resulting value of the threshold such as the wavelength of the colors and changes in light conditions it is very important to calculate the appropriate threshold value such that the system can adapt to different scenarios. The dynamic thresholding concept can be employed for a better system. Where the dynamic thresholding can be represented as

$$D(x,y) = 1 \quad |I(x,y)-B(x,y)| > TH$$

$$D(x,y) = 0 \quad \text{otherwise}$$

Where *D(x,y)* is the output pixel, *I(x,y)* is the pixel intensity of the input image and *B(x,y)* is the pixel intensity of the background image and *TH* is the selected threshold. If *D(x,y) = 1*, these pixels of the current image are classified as the foreground image objects.

3.3 Denoising Using Wiener Filtering

At the time of capturing an image, Gaussian noise occurs as a result of sensor limitations during image acquisition under low-light conditions, which makes it difficult for the visible light sensors to efficiently capture details of the scene. Gaussian Noise is a statistical noise having a probability density function equal to a normal distribution, also known as Gaussian Distribution. A random Gaussian function is added to an Image function to generate this noise. It is also called electronic noise because it arises in amplifiers or detectors.

By using a Weiner filter this noise can be reduced to a very extent. It removes the additive noise and inverts the blurring simultaneously. The Wiener filtering is optimal in terms of the mean square error. In other words, it minimizes the overall mean square error in the process of inverse filtering and noise smoothing. The Wiener filtering is a linear estimation of the original image.

Inverse filtering is a restoration technique for deconvolution, i.e. when the image is blurred by a known lowpass filter, it is possible to recover the image by inverse filtering or generalized inverse filtering. However, the inverse filtering is very sensitive to additive noise. The approach of reducing one degradation at a time allows us to develop a restoration algorithm for each type of degradation and simply combine them. The Wiener filtering executes an optimal tradeoff between inverse filtering and noise smoothing. It removes the additive noise and inverts the blurring simultaneously.

The Wiener filtering is optimal in terms of the mean square error. In other words, it minimizes the overall mean square error in the process of inverse filtering and noise smoothing. The Wiener filtering is a linear estimation of the original image. The approach is based on a stochastic framework. The orthogonality principle implies that the Wiener filter in the Fourier domain can be expressed as follows:

$$W(f_1, f_2) = \frac{H^*(f_1, f_2) S_{xx}(f_1, f_2)}{|H(f_1, f_2)|^2 S_{xx}(f_1, f_2) + S_{\eta\eta}(f_1, f_2)},$$

Where $S_{xx}(f_1, f_2), S_{\eta\eta}(f_1, f_2)$ are respectively power spectra of the original image and the additive noise, and $H(f_1, f_2)$ is the blurring filter. It is easy to see that the Wiener filter has two separate parts, an inverse filtering part and a noise smoothing part. It not only performs the deconvolution by inverse filtering (high pass filtering) but also removes the noise with a compression operation (low pass filtering).

3.3.1 Implementation of wiener filtering

To implement the Wiener filter in practice we have to estimate the power spectra of the original image and the additive noise. For white additive noise, the power spectrum is equal to the variance of the noise. To estimate the power spectrum of the original image many methods can be used. A direct estimate is the periodogram estimate of the power spectrum computed from the observation:

$$S_{yy}^{per} = \frac{1}{N^2}[Y(k,l)Y(k,l)^*]$$

where **Y(k,l)** is the DFT of the observation. The advantage of the estimate is that it can be implemented very easily without worrying about the singularity of the inverse filtering. Another estimate which leads to a cascade implementation of inverse filtering and noise smoothing is

$$S_{xx} = \frac{S_{yy} - S_{\eta\eta}}{|H|^2},$$

which is a straightforward result of the fact: $S_{yy} = S_{\eta\eta} + S_{xx}|H|^2$. The power spectrum S_{yy} can be estimated directly from the observation using the periodogram estimate. This estimate results in a cascade implementation of inverse filtering and noise smoothing:

$$W = \frac{1}{H} \frac{S_{yy}^{per} - S_{\eta\eta}}{S_{yy}^{per}}.$$

The disadvantage of this implementation is that when the inverse filter is singular, we have to use the generalized inverse filtering. People also suggest the power spectrum of the original image can be estimated based on a model such as the $1/f^{\alpha}$ model.

3.4 Binarization Process

Setting the desired threshold value as a standard and converting a Gray image into values of 0 and 1 is called "binarization processing" (usually black is 0 and white is 1). By using this process, it is possible to focus on only the object of inspection and conduct various analyses.

3.4.1 Blob analysis:

The method of analyzing an image that has undergone binarization processing is called "blob analysis". A blob refers to a lump. Blob analysis is image processing's most basic method for analyzing the shape features of an object, such as the presence, number, area, position, length, and direction of lumps.

Blob detection methods are used to spot regions in a digital image that stands out from the background in terms of brightness or color. The Blob Analysis block is used to calculate statistics for labeled regions in a binary image. The block returns a number of things, such as the centroid, bounding box, label matrix, and blob count. The Blob Analysis block can accept and produce signals of different sizes.

3.4.2 Blob Detection:

The main purpose of image processing is to extract various information from an image. It is obvious that this type of information is derived from digital photographs given that they contain a variety of items and information.

To do this, image processing techniques can be used to identify and detect such features and objects. Blob Detection is one of the most promising approaches.

In a sense, a blob is any enormous object or bright object against a dark background. In photographs, we can generalize a blob as a collection of pixel values that resembles a colony or a massive object that stands out from its surroundings. Such blobs in an image can be found via image processing.

3.4.3 Blob Measurement

Blob Measurement is the process of counting the total number of lumps or blobs present in an image once the blob detection process is done. So that, we can count the total number of cars or vehicles parked in the parking area.

As our proposed methodology is platform-dependent during the process of blob analysis we will divide the entire background into different parts and perform different operations by comparing foreground and background images. Analyzing using segmented images results in obtaining the total number of cars present in each individual lane and also makes the process of counting the total number of cars present in the parking a much easier task to perform. Such that, we can also redirect the user to a particular lane based upon cars present in an individual lane.

CHAPTER 4

EXPERIMENTAL INVESTIGATION

4.1 MATLAB Code:

```
image = imread('C:\Users\DELL\OneDrive\Desktop\SMART CAR PARKING
SYSTEM USING FOREGROUND DETECTION AND BLOB MEASUREMENT
TECHNIQUE-DESKTOP-3F465H1-DESKTOP-3F465H1\harifg.jpg'); %To read
Cars parked in parking area.

background = imread('C:\Users\DELL\OneDrive\Desktop\SMART CAR PARKING
SYSTEM USING FOREGROUND DETECTION AND BLOB MEASUREMENT
TECHNIQUE-DESKTOP-3F465H1-DESKTOP-3F465H1\haribg.jpg'); %To read
background image or initial image of parking area

img = double(rgb2gray(image));%convert to gray

bg = double(rgb2gray(background));%convert 2nd image to gray

[height width] = size(img); %image size?

%ITS WORK FOR WHOLE PARKING SYSTEM

totalslot=36;    %Given Total number of slot in the parking area.

 %Foreground Detection

 thresh=11;

 fr_diff = abs(img-bg);

 for j = 1:width

 for k = 1:height

 if (fr_diff(k,j)>thresh)

 fg(k,j)  = img(k,j);

 else

 fg(k,j) = 0;

 end

 end

 end

 park=sprintf('Present Parking Area with total %d slots',totalslot);
```

```
subplot(2,2,1) , imshow(image), title (park);
subplot(2,2,2) , imshow(mat2gray(img)), title ('Gray Scale Input Image');
subplot(2,2,3) , imshow(mat2gray(bg)), title ('Back Ground Image');

sd=imadjust(fg);% adjust the image intensity values to the color map
level=graythresh(sd);
m=imnoise(sd,'gaussian',0,0.025);% apply Gaussian noise
k=wiener2(m,[5,5]);%filtering using Weiner filter
bw=im2bw(k,level);
bw2=imfill(bw,'holes');
bw3 = bwareaopen(bw2,5000);
labeled = bwlabel(bw3,8);
%Blob measurements
blobMeasurements = regionprops(labeled,'all');
numberofcars = size(blobMeasurements, 1);
cars=sprintf('Fore Ground with Total %d spaces available',totalslot-numberofcars);
subplot(2,2,4) , imagesc(labeled), title (cars);
hold off;

%CONDITION TO CHECK THE VACANT SPACE
%IF YES then it divide it itno 6 parts as 6 LANEs are there then for each
%lane image processing is applied as before and lane with vacant space
%comes first with their space.
%LANE number are like
%        LANE 1    LANE 2
%        LANE 3    LANE 4
%        LANE 5    LANE 6

if((totalslot-numberofcars)>0);
```

```
 fprintf('You can enter into the parking area');
fprintf('\n Number of car present');
disp(numberofcars);% display number of cars
fprintf('Number of vacant space present present');
disp(totalslot-numberofcars);
fprintf('PARKING AREA STRUCTURE with LANE:- \n LANE 1\t\t LANE 2 ');
fprintf('\n LANE 3\t\t LANE4 \n LANE 5\t\t LANE 6');
%These code just divide the image of full parking area into 6 parts as 3
%rows and 2 coloums.
r3=int32(height/3);
c2=int32(width/2);
img1=img(1:r3,1:c2);
img2=img(1:r3,c2+1:end);
img3=img(1+r3:2*r3,1:c2);
img4=img(1+r3:2*r3,c2+1:end);
img5=img(2*r3+1:end,1:c2);
img6=img(2*r3+1:end,c2+1:end);
imga={img1 img2 img3 img4 img5 img6}; %An Array iscreated which store 6
image.
bg1=bg(1:r3,1:c2);
bg2=bg(1:r3,c2+1:end);
bg3=bg(1+r3:2*r3,1:c2);
bg4=bg(1+r3:2*r3,c2+1:end);
bg5=bg(2*r3+1:end,1:c2);
bg6=bg(2*r3+1:end,c2+1:end);
bga={bg1 bg2 bg3 bg4 bg5 bg6};

% LOOP is taken from LANE 6 to LANE 1 for better understandibilty.
% And again previos process is taken for each lane for vacant space
% detection.
```

```
for a=6:-1:1
 totalslota=6;     %Also all variables are changed by adding a as suffix.
    imgb=imga{a};


    bgb=bga{a};
    [heighta widtha] = size(imgb);
    thresha=11;
fr_diffa = abs(imgb-bgb);
for j = 1:widtha
for k = 1:heighta
if (fr_diffa(k,j)>thresha)
fga(k,j) = imgb(k,j);
else
fga(k,j) = 0;
end
end
end

sda=imadjust(fga);% adjust the image intensity values to the color map
levela=graythresh(sda);
ma=imnoise(sda,'gaussian',0,0.025);% apply Gaussian noise
ka=wiener2(ma,[5,5]);%filtering using Weiner filter
bwa=im2bw(ka,level);
bw2a=imfill(bwa,'holes');
bw3a = bwareaopen(bw2a,5000);
labeleda = bwlabel(bw3a,8);

blobMeasurementsa = regionprops(labeleda,'all');
numberofcarsa = size(blobMeasurementsa, 1);
```

```
%If lane is available with vacant slot then this statement is true and car will get its
direction.

if((totalslota-numberofcarsa)>0)

    fprintf('\n \nGo to Lane %d',a);

fprintf('\n Number of car present');

   disp(numberofcarsa);% display number of cars

   fprintf('Number of vacant space present present');

   disp(totalslota-numberofcarsa);

   figure, subplot(2,2,1), imshow(image), title('Entire parking area');

   lane=sprintf('Lane %d is availabe with vacant slot',a);

   subplot(2,2,2) , imshow(mat2gray(imgb)), title (lane);

subplot(2,2,3) , imshow(mat2gray(bgb)), title ('Back Ground Frame ');

cars=sprintf('Fore Ground Image with Total space available in lane %d is
%d',a,totalslota-numberofcarsa);

subplot(2,2,4) , imagesc(labeleda), title (cars);

hold off;

break;

end


   end

   else

    fprintf('\n No space available in parking area.\n Exit');

    % If whole parking area is full then this statement will execute.

End
```

CHAPTER 5

EXPERIMENTAL RESULTS

A camera is fixed at the top of a parking area which continuously captures a live video and extracts images time-to-time. This system is capable of capturing 50 frames per second *Figure 11*. Indicates an image of the present-time parking area and the total capacity of the parking area can be calculated. *Figure 11*. Is considered as the present input and *Figure 12*. Is obtained by converting an RGB image to a Gray-Scale image for further image processing. Grayscale images are useful in the simplification of different algorithms which reduces the computational requirement.

Figure 11. Present Parking Area with total 36 slots

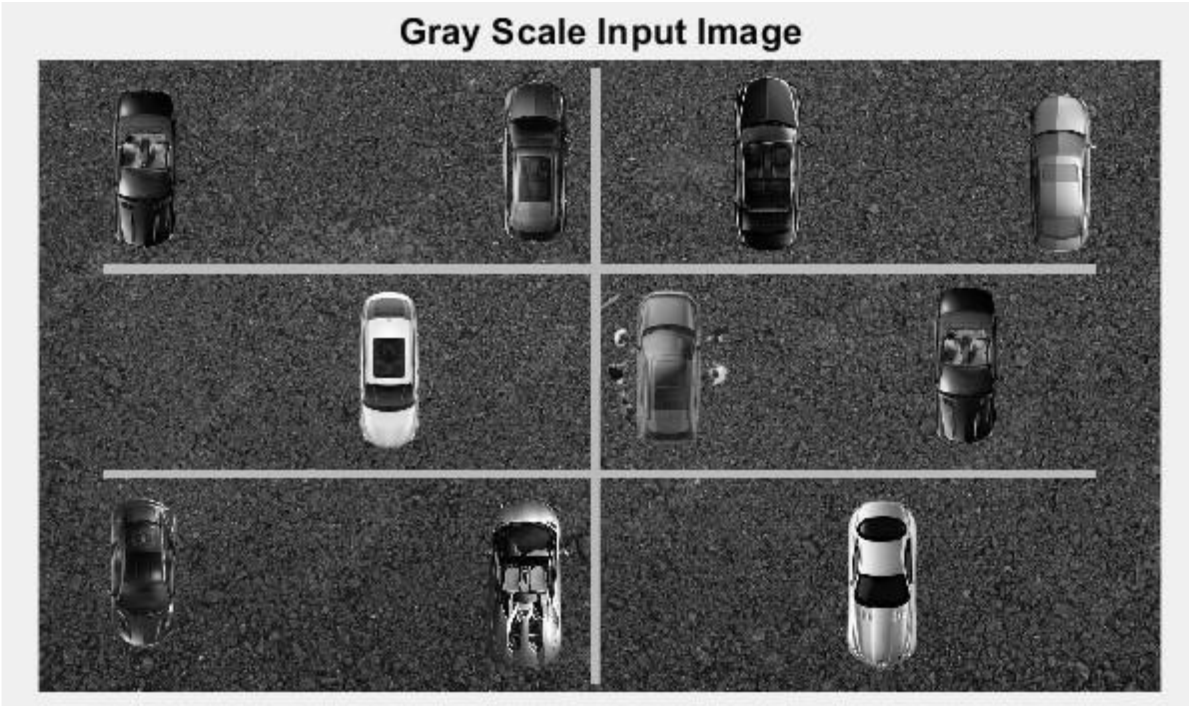

Figure 12. Gray Scale Input Image

During the initialization or initial process of applying our algorithm, *Figure 13.* an empty parking area is captured by a fixed camera which is mostly used in the processes of background subtraction method and extracting different features associated with the image. Once the threshold value is fixed along with the background image we can implement and analyze images by applying the blob analysis method where the change in frequency among foreground and background images is obtained as a blob.

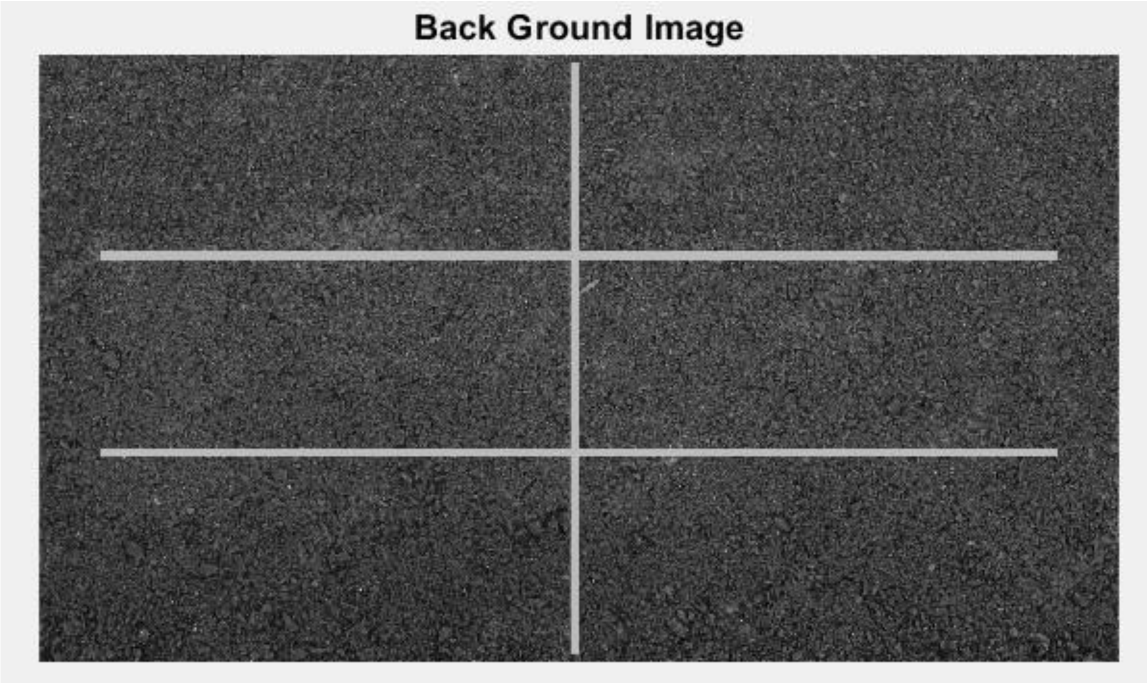

Figure 13. Back Ground Image

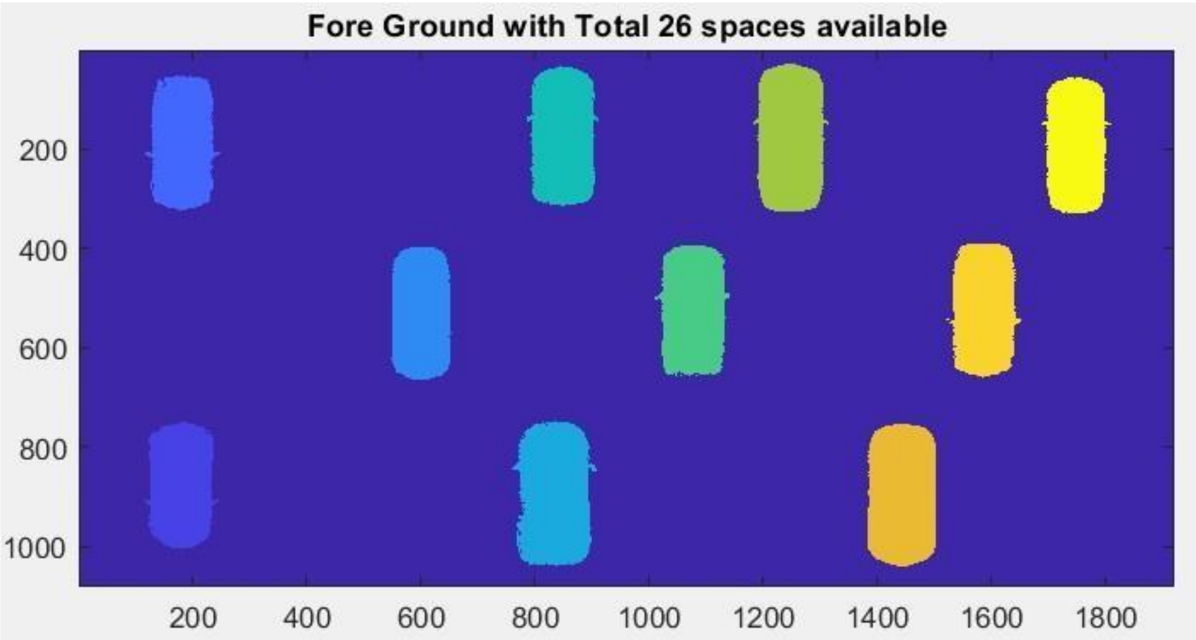

Figure 14. Blob Analysis of Complete Parking area

Once we calculate the total parking slots available in the parking area. A segmentation process is followed by this algorithm where each and every lane in the parking slot undergoes blob analysis followed by a blob measurement. *Figure 15*. Indicates a segmented frame from our present parking area and *Figure 16.* Is a segmented frame extracted from the fixed background image. Blob analysis is applied considering *Figure 15.* And *Figure 16*. Which results in a frequency change among the images *Figure 17.* And then we apply the blob measurement technique where we count the total number of cars in that particular lane. If the lane is full, we further apply our algorithm for the next following lane and if there is any empty slot available in the analyzed lane segment, we redirect the user to that specific lane. The system returns no parking space available if the total count after blob measurement reaches the maximum capacity of the parking area.

Figure 15. Segmented input image for blob analysis

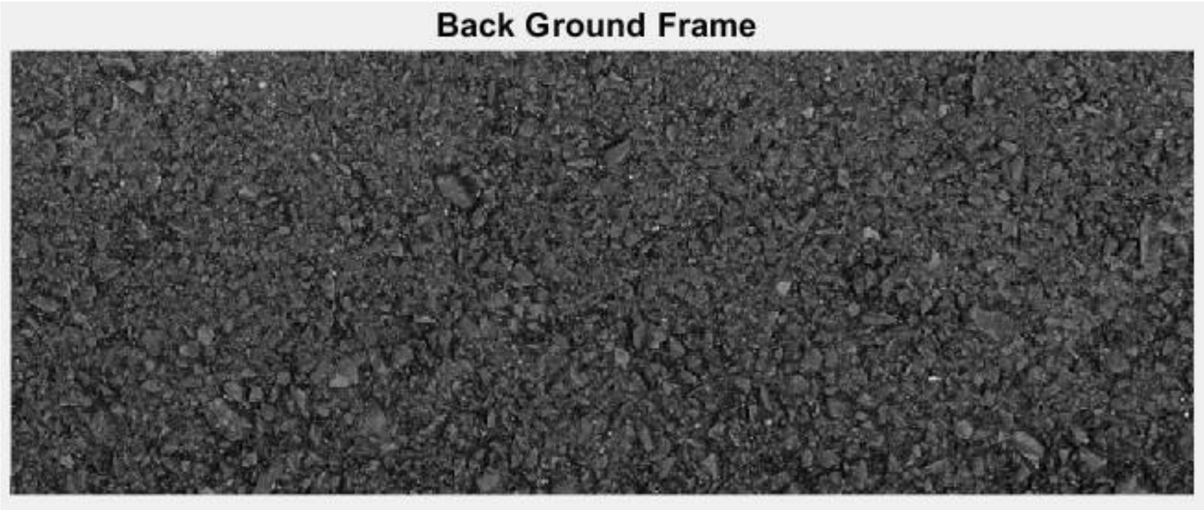

Figure 16. Segmented background image for blob analysis

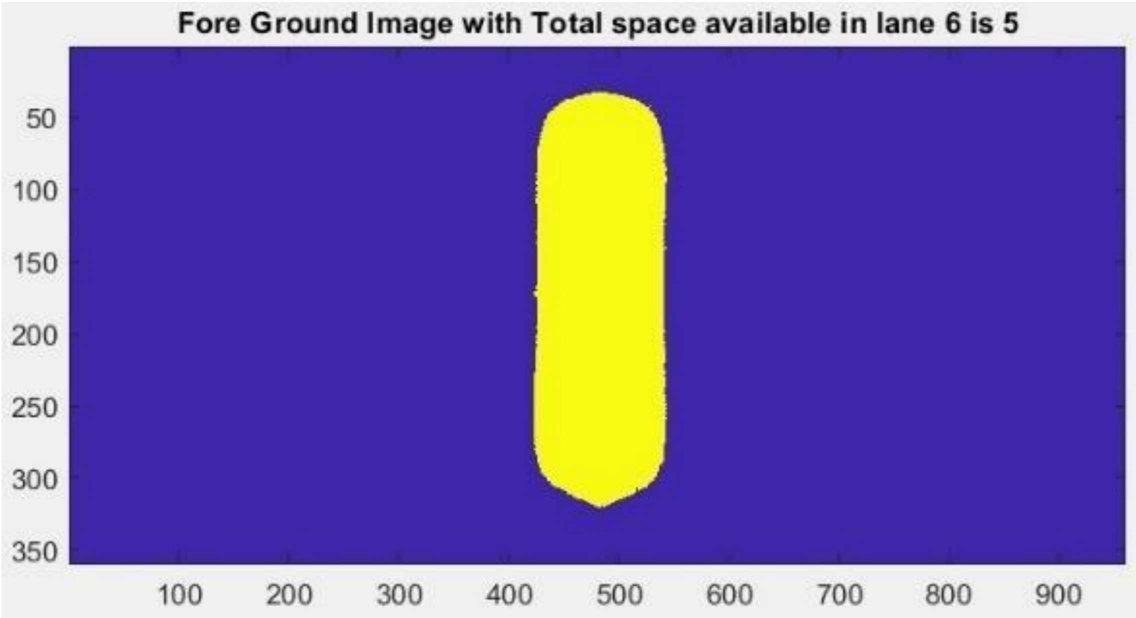

Figure 17. Finding the total number of blobs in a lane of the parking area

CHAPTER 5

CONCLUSION

An advanced parking system management can be implemented and has been presented with the experimental results. Advancement in the parking area tokenization system is obtained by applying blob analysis and blob measurement techniques at different times on different segmented image frames. The advantages of this proposed method for a smart parking management system are there is no time required for searching for a parking slot, the efficiency of the parking management system is increased compared to the existing system, and emissions are greatly brought down and reduced. In the future, the proposed system module will be implemented and developed in FPGA for a high-speed real-time embedded system.

REFERENCES

1. Khantasak, S., Jindapetch, N., Hoyingcharoen, P., Chetpattananondh, K., Ikura, M., & Chumpol, S. (2018, November). Parking violation detection system based on video processing. In *2018 IEEE 5th International Conference on Smart Instrumentation, Measurement and Application (ICSIMA)* (pp. 1-5). IEEE.

2. Liu, J., Mohandes, M., & Deriche, M. (2013, December). A multi-classifier image based vacant parking detection system. In *2013 IEEE 20th International Conference on Electronics, Circuits, and Systems (ICECS)* (pp. 933-936). IEEE.

3. Sofwan, A., Hariyanto, M. S., Hidayatno, A., Handoyo, E., Arfan, M., & Somantri, M. (2018, October). Design of Smart Open Parking Using Background Subtraction in the IoT Architecture. In *2018 2nd International Conference on Electrical Engineering and Informatics (ICon EEI)* (pp. 7-11). IEEE.

4. Kotb, A. O., Shen, Y. C., Zhu, X., & Huang, Y. (2016). iParker—A new smart car-parking system based on dynamic resource allocation and pricing. *IEEE transactions on intelligent transportation systems*, *17*(9), 2637-2647.

5. Tuteja, S., Poddar, S., Agrawal, D., & Karar, V. (2022). PredictV: A Vehicle Prediction Scheme to Circumvent Occluded Frames. *IEEE Access*, *10*, 20029-20042.

6. Dixit, M., Srimathi, C., Doss, R., Loke, S., & Saleemdurai, M. A. (2020, July). Smart parking with computer vision and IoT technology. In *2020 43rd International Conference on Telecommunications and Signal Processing (TSP)* (pp. 170-174). IEEE.

7. G. Wang, C. Lopez-Molina and B. De Baets, "High-ISO Long-Exposure Image Denoising Based on Quantitative Blob Characterization," in *IEEE Transactions on Image Processing*, vol. 29, pp. 5993-6005, 2020, doi: 10.1109/TIP.2020.2986687.

8. H. Kong, H. C. Akakin and S. E. Sarma, "A Generalized Laplacian of Gaussian Filter for Blob Detection and Its Applications," in *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 1719-1733, Dec. 2013, doi: 10.1109/TSMCB.2012.2228639.

9. U. Mittal, T. Shekher, K. Agarwal, S. K. Singh and H. Chaudhary, "Real-Time Smart Parking System," *2021 IEEE 8th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON),* 2021, pp. 1-5, doi: 10.1109/UPCON52273.2021.9667606.

10. K. B. Dsouza, S. Mohammed and Y. Hussain, "Smart parking — An integrated solution for an urban setting," *2017 2nd International Conference for Convergence in Technology (I2CT)*, 2017, pp. 174-177, doi: 10.1109/I2CT.2017.8226115.

11. I. Makhdoom, J. Lipman, M. Abolhasan and D. Challen, "Science and Technology Parks: A Futuristic Approach," in *IEEE Access*, vol. 10, pp. 31981-32021, 2022, doi: 10.1109/ACCESS.2022.3159798.

12. J. Lin, S. -Y. Chen, C. -Y. Chang and G. Chen, "SPA: Smart Parking Algorithm Based on Driver Behavior and Parking Traffic Predictions," in *IEEE Access*, vol. 7, pp. 34275-34288, 2019, doi: 10.1109/ACCESS.2019.2904972.

13. J. Zuo, Z. Jia, J. Yang and N. Kasabov, "Moving Target Detection Based on Improved Gaussian Mixture Background Subtraction in Video Images," in *IEEE Access*, vol. 7, pp. 152612-152623, 2019, doi: 10.1109/ACCESS.2019.2946230.

14. T. Huynh-The, C. -H. Hua, N. A. Tu and D. -S. Kim, "Locally Statistical Dual-Mode Background Subtraction Approach," in *IEEE Access*, vol. 7, pp. 9769-9782, 2019, doi: 10.1109/ACCESS.2019.2891084.

final project report.pdf

ORIGINALITY REPORT

4%

SIMILARITY INDEX

PRIMARY SOURCES

1	Bo Ma, Zheru Chi, Tianwen Zhang. "Motion detection and tracking based on level set algorithm", ICARCV 2004 8th Control, Automation, Robotics and Vision Conference, 2004., 2004 Crossref	15 words – 1%
2	"Computational Vision and Bio Inspired Computing", Springer Science and Business Media LLC, 2018 Crossref	14 words – 1%
3	copp.inforang.com Internet	11 words – < 1%
4	Amir O. Kotb, Yao-chun Shen, Yi Huang. "Smart Parking Guidance, Monitoring and Reservations: A Review", IEEE Intelligent Transportation Systems Magazine, 2017 Crossref	10 words – < 1%
5	Nadir Mohamed Belmessaoud, Youcef Bentoutou, Miloud Chikr El-Mezouar. "FPGA implementation of feature detection and matching using ORB", Microprocessors and Microsystems, 2022 Crossref	10 words – < 1%
6	"Proceedings of the First International Conference on Computational Intelligence and Informatics", Springer Science and Business Media LLC, 2017	

Crossref

8 words – < 1%

7	Mathias Gabriel Diaz Ogás, Ramon Fabregat, Silvana Aciar. "Survey of Smart Parking Systems", Applied Sciences, 2020 Crossref	8 words – < 1%
8	en.wikipedia.org Internet	8 words – < 1%
9	www.ijraset.com Internet	8 words – < 1%
10	"Advances in Multimedia Information Processing – PCM 2018", Springer Science and Business Media LLC, 2018 Crossref	6 words – < 1%

EXCLUDE QUOTES	ON	EXCLUDE SOURCES	OFF
EXCLUDE BIBLIOGRAPHY	ON	EXCLUDE MATCHES	OFF