

# Vehicle License Plate Detection using Vertical Edge Detection

K.Thulasimani  
Assistant Professor

Department Of CSE  
Government College of Engineering  
Tirunelveli, India

T. S. Rajashree  
ME, CSE

Government College of Engineering  
Tirunelveli, India

T. K. Renugha  
ME, CSE

Government College of Engineering  
Tirunelveli, India

**Abstract**— The main aim of this paper is to propose a fast method for Vehicle License Plate Detection (VLPD) using Vertical Edge Detection Algorithm (VEDA). This paper proposes a method for Vehicle license plate detection (VLPD) which have three main contributions. The vertical edge detection algorithm (VEDA), that makes use of contrast between the grey scale values, to enhance the speed of VLPD method. The input image is binarized by adaptive thresholding. Morphological operations are performed to enhance the image and then VEDA is applied. The second contribution is that our proposed VLPD method is able to process very low-resolution images taken by a web-camera. After the vertical edges have been detected by VEDA, the most like plate details based on colour information are highlighted. Then the candidate region based on statistical and logical operations will be extracted. The third is comparison of VEDA with sobel operator in terms of accuracy and performance.

**Keywords**— *adaptive thresholding, vehicle license plate detection, sobel operator, vertical edge detection*

## I. INTRODUCTION

The main aim of this project is to propose a fast method for Vehicle License Plate Detection (VLPD) using Vertical Edge Detection Algorithm (VEDA). Many of the existing Vehicle License Plate Detection (VLPD) method make use of Sobel edge detection which is based on convolving the image with a small, separable, and integer valued filter both in horizontal and in vertical direction and is therefore computationally expensive. Under noisy conditions this algorithm provides poor performance. It is best suitable for highly contrasted noiseless images and is restricted towards ambient lighting conditions, interference characters, and other problems such as fixed backgrounds, and known colour. In this proposed system VEDA is used which is based on the contrast between the grey scale values, which in turn enhances the speed of VLPD method. Performance of Vertical Edge Detection Algorithm (VEDA) is faster than Sobel's edge detection. Proposed algorithm gives better result in terms of computation time and the detection rate. Morphological operations are used for extracting the plate from complex background.

## II. OVERVIEW OF THE PROJECT

Vehicle license plate recognition system is an image processing technology used for identifying vehicles by capturing their Vehicle License Plates (VLPs). The other names of vehicle license plate recognition technology are automatic number-plate recognition, automatic vehicle identification, vehicle license plate recognition or optical character recognition for vehicles. Vehicle License Plate Detection and Recognition System (VLPDRS) plays an important role in research areas due to its wide variety of applications such as the payment of parking fee, traffic data collection, highway toll fee, and crime prevention. This Car License Plate Detection and Recognition System mainly consists of three parts: License Plate Detection (LPD), character segmentation, and character recognition. Of these, LPD is the most important part in the system since it affects the system's accuracy. To create successful and fast Vehicle License Plate Detection System (VLPDS), we have to resolve many issues such as poor image quality, plate sizes and designs, processing time, and background details and complexity. The main reason for identifying a vehicle is to prevent crime, to access border control and vehicle control. The features for identifying a vehicle are model, color, and license plate.

The police cars get installed with a camera to track vehicles in vehicle tracking system. High quality cameras are used in many tracking system to capture best images and is equally expensive both in terms of hardware and software. The Vehicle License Plate Detection System usually process images with 640 x 480 resolution. The practical usability of VLPD method is due to its reduction in time and computational complexity. This paper proposed a method for vehicle license plate detection in which a web-camera with 352x288 resolution is used instead of a more sophisticated one

In VLPDRS vertical edge detection and extraction steps affects the system's accuracy. Computation time can be considerably reduced by Vertical Edge Detection Algorithm.

### III. BLOCK DIAGRAM

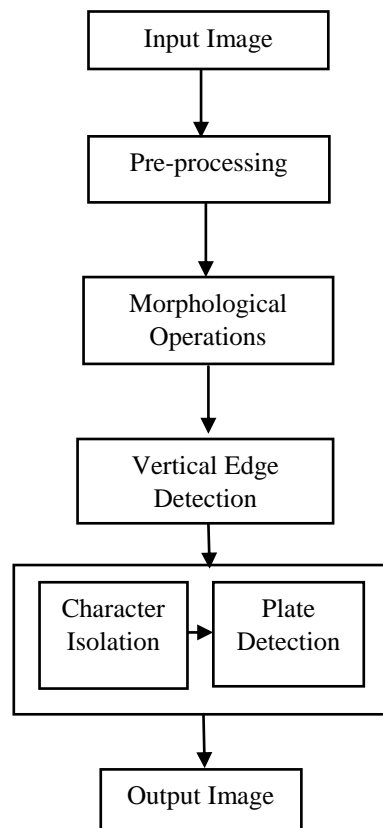


Fig1: Flow Diagram

### IV. SYSTEM ARCHITECTURE

#### A. Pre-processing

The image to be processed is given as input. And the given input image is resized to 400 NAN. RGB colour image get converted in to grey scale image which involves grey shades to indicate RGB colour. In grey scale transformation each pixel contains the intensity information and it eliminates hue and saturation to retain luminance. The total absence of light is indicated by dark black and the lightest possible colour is white. Intermediate shades of grey are represented by equal brightness levels of the three primary colours or transmitted light, or equal amount of three primary pigments CMY (cyan, magenta and yellow) for reflected light.

#### B. Morphological Operations

This operation is to extract image components like shape, skeleton, convex hull. The fundamental morphological operations are erosion and dilation. In the case of erosion operation the image details that are smaller than the structuring element are removed from the image. Usually binary images contain numerous imperfections. In particular, the binary regions produced by simple thresholding are distorted by noise and texture.

Morphological image processing removes imperfections. The erosion of A by b denoted by

$$A \ominus B = \{z/ (B)_z \text{ subset of } A\} \quad (1)$$

This denoted the erosion of A by b is the set of all points z such that B, translated by z, is contained in A. Set B is a structuring element .The statement that indicates B has to be contained in A is expressed as follows

$$A \ominus B = \{z/ (B)_z \text{ subset of } A^c = \emptyset\} \quad (2)$$

$A^c$  is the complement of A and  $\emptyset$  is the empty set. Morphological operations can also be applied to grayscale images such that their light transfer functions are unknown. Morphological techniques probe an image with a small shape or template called a structuring element. The structuring element is positioned at all possible locations in the image and it is compared with the corresponding neighbourhood of pixels. Some operations test whether the element fit within the neighbourhood, while others test whether it hit or intersects with the neighbourhood.

Dilation operation is used to grow or thicken objects in a binary image. This specific manner and extent of this thickening is controlled by the shape of the structuring element used. With A and B as sets in  $Z^2$ , the dilation of A by B denoted by  $A+B$  and is defined as

$$A+B = \{Z/ (B)_z \cap A \neq \emptyset\} \quad (3)$$

This equation is based on reflection of B about its origin, and shifting this reflection by z. The dilation of A by B then is the set of all displacements, Z, such that B and A overlap by at least one element. Based on this interpretation dilation can be denoted as,

$$A+B = \{Z/ [(B)_z \cap A] \leq A\} \quad (4)$$

Here B is the structuring element and A is the set of image objects .The structuring element viewed as a convolution mask. The basic process of flipping(rotating)about its origin and then successively displacing it so that it slides over set a is analogous to spatial convolutions .The dilation is based on set operations and therefore is a nonlinear operation, whereas the convolution is a linear operation.

#### C. Vertical Edge Detection Algorithm

Vertical edges can be detected by using a horizontal gradient operator followed by a morphological operation to detect the extreme values of the gradient. The gradient produces a doublet of extremes, positive-negative or negative-positive, depending on the direction of the transition. Detection of edges are done by sobel operator. But in the case of VEDA only the horizontal gradient values are calculated and removed those horizontal value thin lines to get vertical edges clearly. Use of odd number

of pixels in gradient calculation prevents a shift in locations.

$$B(j,k)=A(j,k+1)-A(j,k-1) \quad (5)$$

If A has grey values in the range 0 to 255, for example, then B may have values in the range -255 to 255. This can be done by the replacement

$$(B(j,k)-B_{\min}/B_{\max}-B_{\min}) \rightarrow B(j,k) \quad (6)$$

#### D. Character Isolation

When the images are monochrome, region analysis must be carried out with a set of descriptors based on intensity levels and spatial properties such as moment or texture.

Descriptors can also yield misleading results if connectivity properties are not used in the region growing process. Grouping pixels with the same intensity level to form a region without paying attention to connectivity would yield a segmentation result. Region growth should stop when no more pixels satisfy the criteria for inclusion in that region. Criteria's are intensity values, texture, colour are local in nature and do not take into account the history of region growth and the shape of the region being grown.

#### E. Plate detection

Plate detection explains the selection process of the LP region from the input image. The plate region can be checked pixel by pixel whether it belongs to the LP region or not. A mathematical formulation is proposed for this purpose and once this formulation is applied on each pixel, the probability of being that pixel an element of the LP can be decided and the candidate regions of each column will be checked one by one. If the column blackness ratio exceeds 50%, then the current column belongs to LP region.

The blkPix represents the total number of black pixels per each column in the current candidate region and the colmnHght represents the column height of the of the candidate region. This condition with a fixed value (0.5) is used with non-blurry images. However, some pixels of the candidate regions will not be detected if the ratio of blackness to the total length (height) of the candidate region is greater than 50%. Therefore, the conditions changed to be less than 50% according to the ratio of the blurry level or the deformation of the LP. The value is reduced when the blurry level is high in order to highlight more important details, and it is increased when the blurry level is low. Therefore, each group consists of ten rows. Due to the How Many Lines[a] values, some rows have number of drawn horizontal lines, and this makes some groups have horizontal lines. The step here is to store the total number of horizontal lines for each group.

In this step, a matrix is created to store the total number of drawn lines for each ten rows (group). So, it is meaningful to use a median filter in order to eliminate

those unsatisfied groups and keep the satisfied ones in which the LP details exist in. Thus, the total number of groups including the parts of LP regions will be counted and stored. The remaining groups after filtering step should have the LP details. Therefore, their locations are stored. Finally extract both upper and lower boundaries of each satisfied group by using its own index.

### V. MODULES

#### A. Pre-processing

In pre-processing, the image to be processed is given as input and resizing the image for keeping the aspect ratio same. The resized image is usually of 400 NAN. Each of the image in the RGB scale converted into grey scale image.

##### a) RGB To Grayscale Conversion

An RGB colour space is any additive colour space based on the RGB colour model. A particular RGB colour space is defined by the three chromaticity of the red, green and blue additive primaries, and can produce any chromaticity that is the triangle defined by those primary colors white point chromaticity and a gamma correction curve. RGB is the most commonly used RGB color space. RGB is an abbreviation for red-green-blue.

A grayscale digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information. Images of this sort, also known as black-and-white, are composed exclusively of shades of gray, varying from black at the weakest intensity to white at the strongest. The intensity of a pixel is expressed within a given range between a minimum and a maximum, inclusive. This range is represented in an abstract way as a range from 0 (total absence, black) and 1 (total presence, white), with any fractional values in between. This notation is used in academic papers, but this does not define what "black" or "white" is in terms of colorimetric.

Another convention is to employ percentages, so the scale is then from 0% to 100%. This is used for a more intuitive approach, but if only integer values are used, the range encompasses a total of only 101 intensities, which are insufficient to represent a broad gradient of grays. Also, the percentile notation is used in printing to denote how much ink is employed in half toning, but then the scale is reversed, being 0% the paper white (no ink) and 100% a solid black (full ink).

Grayscale images are distinct from one-bit bi-tonal black-and-white images, which in the context of computer imaging are images with only the two colors, black, and white (also called bi-level or binary images).



Fig2 (a-f): Sample license plates

Grayscale images have many shades of grey in between. Grayscale images are also called monochromatic, denoting the presence of only one (mono) color (chromaticity).

### B. Morphological operation

Morphological image is a collection of non-linear operations related to the shape or morphology of features in an image.

Morphological operations rely only on the relative ordering of pixel values, not on their numerical values, and therefore are especially suited to the processing of binary images. Morphological operations can also be applied to grayscale images such that their light transfer functions are unknown and therefore their absolute pixel values are of no or minor interest.

Morphological techniques probe an image with a small shape or template called a structuring element. The structuring element is positioned at all possible locations in the image and it is compared with the corresponding neighbourhood of pixels. Some operations test whether the element "fits" within the neighbourhood, while others test whether it "hits" or intersects the neighbourhood.

The structuring element is a small binary image, i.e. a small matrix of pixels, each with a value of zero or one:

- The matrix dimensions specify the size of the structuring element.
- The pattern of ones and zeros specifies the shape of the structuring element.

- An origin of the structuring element is usually one of its pixels, although generally the origin can be outside the structuring element.

The morphological operations are follows

#### a) Erosion

This operation is act as a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries, skeletons, and the convex hull. Erosion and Dilation operations are the fundamental to morphological processing. In erosion the image details smaller than the structuring element are filtered (removed) from the image. Binary images may contain numerous imperfections. In particular, the binary regions produced by simple thresholding are distorted by noise and texture. Morphological image processing pursues the goals of removing these imperfections by accounting for the form and structure of the image. These techniques can be extended to greyscale images. With A and B as sets in  $Z^2$ , the erosion of A by b denoted.

$$A \ominus B = \{z / (B)_z \text{ subset of } A\} \quad (7)$$

This denoted the erosion of A by B is the set of all points z such that B, translated by z, is contained in A. Here set B is assumed to be a structuring element. Because the statement that B has to be contained in A is can express in the following

$$A \ominus B = \{z / (B)_z \text{ subset of } A^c = \emptyset\} \quad (8)$$

$A^c$  the complement of A and  $\emptyset$  is the empty set. Morphological operations can also be applied to grayscale images such that their light transfer functions are unknown and therefore their absolute pixel values are of no or minor interest. Morphological techniques probe an image with a small shape or template called a structuring element. The structuring element is positioned at all possible locations in the image and it is compared with the corresponding neighbourhood of pixels. Some operations test whether the element "fits" within the neighbourhood, while others test whether it "hits" or intersects the neighbourhood.

The erosion of a binary image  $f$  by a structuring element  $s$  (denoted  $f \ominus s$ ) produces a new binary image  $g = f \ominus s$  with ones in all locations  $(x,y)$  of a structuring element's origin at which that structuring element  $s$  fits the input image  $f$ , i.e.  $g(x,y) = 1$  is  $s$  fits  $f$  and 0 otherwise, repeating for all pixel coordinates  $(x,y)$ . Erosion with small (e.g.  $2 \times 2$  -  $5 \times 5$ ) square structuring elements shrinks an image by stripping away a layer of pixels from both the inner and outer boundaries of regions. The holes and gaps between different regions become larger, and small details are eliminated. Erosion removes small-scale details from a binary image but simultaneously reduces the size of regions of interest, too. By subtracting



the eroded image from the original image, boundaries of each region can be found:  $b = f - (f \ominus s)$  where  $f$  is an image of the regions,  $s$  is a  $3 \times 3$  structuring element, and  $b$  is an image of the region boundaries.

#### b) Dilation

Dilation operation is used to grow or thicken objects in a binary image. This specific manner and extent of this thickening is controlled by the shape of the structuring element used. With  $A$  and  $B$  as sets in  $z^2$ , the dilation of  $A$  by  $B$ , denoted  $A+B$ , is defined as

$$A+B = \{Z / (B)_z \cap A \neq \emptyset\} \quad (9)$$

This equation is based on reflecting  $B$  about its origin, and shifting this reflection by  $z$ . The dilation of  $A$  by  $B$  is the set of all displacements,  $Z$ , such that  $B$  and  $A$  overlap by at least one element. This interpretation can be denoted as,

$$A+B = \{Z / [(B)_z \cap A] \neq \emptyset\} \quad (10)$$

Here  $B$  is the structuring element and  $A$  is the set of image objects. The structuring element viewed as a convolution mask. The basic process of flipping or rotating about its origin and then successively displacing it so that it slides over a set of analogous to spatial convolutions. The dilation is based on set operations and therefore is a nonlinear operation, whereas the convolution is a linear operation.

The dilation of an image  $f$  by a structuring element  $s$  (denoted  $f \oplus s$ ) produces a new binary image  $g = f \oplus s$  with ones in all locations  $(x,y)$  of a structuring element's origin at which that structuring element  $s$  hits the input image  $f$ , i.e.  $g(x,y) = 1$  if  $s$  hits  $f$  and 0 otherwise, repeating for all pixel coordinates  $(x,y)$ . Dilation has the opposite effect to erosion. It adds a layer of pixels to both the inner and outer boundaries of regions. The holes enclosed by a single region and gaps between different regions become smaller, and small intrusions into boundaries of a region are filled in.

#### c) Opening

Opening generally smoothen the contour of an object, breaks narrow isthmuses, and eliminates thin protrusions. Closing also tends to smooth sections of contours but, as opposed to opening, it generally fuses narrow breaks and long thin gulfs, eliminates holes, and fills gaps in the contour.

The opening of set  $A$  by structuring element  $B$ , denoted by  $A \circ B$  and is defined as

$$A \circ B = (A \ominus B) \oplus B \quad (11)$$

Thus the opening  $A$  by  $B$  is the erosion of  $A$  by  $B$ , followed by a dilation of the result by  $B$ . The closing of set  $A$  by structuring element  $B$ , denoted  $A \circ B$ , is defined as

$$A \circ B = (A + B) \ominus B \quad (12)$$

It says that the closing of  $A$  by  $B$  is simply the dilation of  $A$  by  $B$ , followed by the erosion of the result by  $B$ . The opening operation has a simple geometric interpretation. The structuring element  $B$  as a (flat) "rolling ball". The boundary of  $A \circ B$  is then established by the points in  $B$  that reach the farthest into the boundary of  $A$  as  $B$  is rolled around the inside of this boundary. This geometric fitting property of the opening operation leads to a set theoretic formulation, which states that the opening of  $A$  by  $B$  is obtained by taking the union of all translates of  $B$  that fit into  $A$ . That is opening can be expressed as a fitting process such that

$$A \circ B = \bigcup \{ (B)_z \mid (B)_z \text{ subset of } A \} \quad (13)$$

Where  $\bigcup \{ \}$  denotes the union of all the sets inside the braces. As in the case with dilation and erosion, opening and closing are duals of each other with respect to set complementation and reflection. That is

$$(A \circ B)^c = (A^c \oplus B) \quad (14)$$

$$(A \circ B)^c = (A^c \circ B) \quad (15)$$

The opening operation satisfies the following properties such as

- i.  $A \circ B$  is a subset (sub image) of  $A$ .
- ii. If  $C$  is a subset of  $D$ , then  $C \circ B$  is a subset of  $D \circ B$ .
- iii.  $(A \circ B) \circ B = A \circ B$ .

In third case, the multiple openings of a set have a no effect after the operator has been applied once. The background noise was completely eliminated in the erosion stage of opening because in this case all noise components are smaller than the structuring element.

#### d) Closing

Closing also tends to smooth sections of contours but fusing narrow breaks and long, thin gulfs and eliminating small holes and filling gaps in the contour. Similarly, the closing operation satisfies the following properties:

1.  $A$  is a subset (sub image) of  $A \circ B$
2. If  $C$  is a subset of  $D$ , then  $C \circ B$  is a subset of  $D \circ B$
3.  $(A \circ B) \circ B = A \circ B$

The size of the noise elements (dark spots) contained within the image (like finger print) actually

increased in size. The reason is that these elements are inner boundaries that increase in size object is eroded.

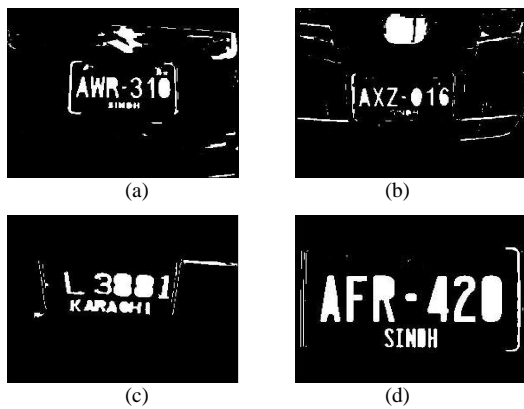


Fig3 (a-d): Results for Erosion and Dilation

### C. Vertical Edge Detection Algorithm

Edge detection is a type of image segmentation techniques which determines the presence of an edge or line in an image and outlines them in an appropriate way. The main purpose of edge detection is to simplify the image data in order to minimize the amount of data to be processed. Generally an edge is defined as a boundary pixels that connect two separate regions with changing image amplitude attributes such as different constant luminance and stimulus values in an image. The detection operation begins with a examination of the local discontinuity at each pixel element in an image.

Amplitude, orientation, and location of a particular subarea in the image that is of interest are essentially important characteristics of possible edges. Based on these characteristics the detector has to decide whether each of the examine pixels is an edge or not.

#### a) Simple Gradient Calculation

Let us represent an image by an array A, in which each element of the array corresponds to the grey level of an image. If the grey levels are in pixel counts, then the numbers might range from 0 to 255 for an eight-bit per pixel image. The gradient is the change in grey level with direction. This can be calculated by taking the difference in value of neighbouring pixels. Let us construct a new array B that contains the values of the gradient from A. The horizontal gradient is formed by taking the differences between column values.

$$B(j,k)=A(j,k+1)-A(j,k) \quad (16)$$

This can be represented by a filter array as shown below:

-1	1
----	---

A problem with this filter is that the location of the gradient in the array B is shifted somewhat to the left. With an even number of pixels in the computation it is impossible to locate the result in the center of the cells used to produce it. It is therefore most common to use an odd number of cells. This can be accomplished by doing the calculation over cells that are separated by step.

$$B(j,k)=A(j,k+1)-A(j,k-1) \quad (17)$$

This can be represented by the array shown below

	0	1
-1		

Note that the result pixel is centered between the left and right pixels used to calculate the gradient, so there is no shift in the location of the gradient result. Horizontal edges would be detected by calculating the vertical gradient. The equation for the separated vertical difference is

$$B(j,k)=A(j+1,K)-A(j-1,k) \quad (18)$$

For an image in which the row coordinates are counted from the bottom edge upward, the corresponding filter array is

1
0
-1

#### b) Vertical Edges

Vertical edges can be detected by using a horizontal gradient operator followed by a threshold operation to detect the extreme values of the gradient. The gradient produces a doublet of extremes, positive-negative or negative-positive, depending on the direction of the transition. The horizontal gradient was calculated by taking differences in the image values between columns. Note that the column before and the column after  $k$  was used. Use of an odd number of pixels in a gradient calculation prevents a shift in location.

$$B(j,k)=A(j,K+1)-A(j,k-1) \quad (19)$$

If A has grey values in the range 0 to 255, for example, then B may have values in the range -255 to 255. The values of B were renormalized to the range 0 to 255 by shifting and scaling. This can be done by the replacement

$$\left[ \frac{B(j,k) - B_{min}}{B_{max} - B_{min}} \times 255 \right] \rightarrow B(j,k) \quad (20)$$

where the brackets  $[]$  indicate rounding to the nearest integer.

#### c) Horizontal Edges

Horizontal edges produce a vertical gradient in the image, and can be enhanced with a vertical gradient detector. A vertical gradient filter can be defined by

$$B(j,k)=A(j+1,K)-A(j-1,k) \quad (21)$$

The gradient values were shifted and normalized by

$$\left[ \frac{B(j,k) - B_{min}}{B_{max} - B_{min}} \times 255 \right] \rightarrow B(j,k) \quad (22)$$

#### d) Diagonal Gradient Detection

A diagonal edge is neither horizontal nor vertical. It will cause a partial response to both the horizontal and vertical edge detectors. An image that is a combination of the two processes can be created by combining the results of each gradient calculation. The image that is so created could be called a gradient image, combining the horizontal gradient and vertical gradient images. The magnitude form is given by

$$B(j,k) = |B_h(j,k)| + |B_v(j,k)| \quad (23)$$

Where  $B_h$  and  $B_v$  are the horizontal and vertical gradient values, respectively. Another way to combine the horizontal and vertical gradients to get an edge gradient is by using

$$B(j,k) = \sqrt{[B_h(j,k)]^2 + [B_v(j,k)]^2} \quad (24)$$

The above equations provide methods to determine a gradient magnitude. The gradient direction can be estimated by using the trigonometric relationship.

The trigonometric relationship is as follows

$$B(j,k) = B_k(j,k) \cos \phi + B_v(j,k) \sin \phi \quad (25)$$

$$\phi = \tan^{-1} \left( \frac{B_v(j,k)}{B_h(j,k)} \right)$$

#### e) Gradient Averaging To Reduce Noise

The use of pixels on either side of  $p$  to calculate the gradient at  $p$  produces a gradient that is properly centered. Detection of the extreme values of the gradient then provides edge detection. However, this method is sensitive

to noise and small fluctuations in image luminance. The effect of noise can be reduced by averaging the gradient calculations over the orthogonal direction. The horizontal gradient computation by the mask

-1	0	1
----	---	---

Vertical averaging can be obtained by adding rows to the mask.

-1	0	1
-1	0	1
-1	0	1

Similarly, the vertical mask can be extended to provide horizontal averaging.

1	1	1
0	0	0
-1	-1	-1

The result produced by using the weights in the mask is placed in the location that is indexed by the centre cell.

## VI. ANALYSIS

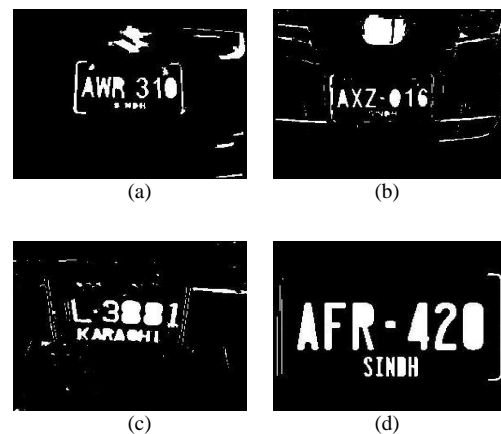


Fig4 (a-d): Results for VEDA

Here we are considering 100 input images of which 92 license plates are detected correctly and so the detection rate get increased to 92% with computation time 7.18ms in comparison with Sobel operator whose computation time is 15.3ms

Evaluation	No.correctly detected plates	No.wrongly detected plates	Detection rate	Computation Time(ms)
<b>VEDA</b>	<b>92/100</b>	<b>8/100</b>	<b>92%</b>	<b>7.18</b>
<b>Sobel Based</b>	<b>89/100</b>	<b>11/100</b>	<b>89%</b>	<b>15.3</b>

## VII. CONCLUSION

This project proposes a new and fast algorithm for vertical edge detection (VEDA) in which its performance is faster than existing algorithms. VEDA contributes to make the whole proposed VLPD method faster. We have proposed a VLPD method in which dataset (images) was captured by using a web-camera. Images are taken from various scenes and under different conditions were employed. Only one LP is considered in each sample for the whole experiments.

In this experiment, the proposed algorithm correctly detects license plate of the vehicle. Then it successfully eliminates the noise which are present in the license plate and extract the characters that are present in the plate. It then finally displays the extracted characters in notepad.

## REFERENCES

1. H. P. Wu, "License Plate Extraction in Low Resolution Video," in Proceedings of the IEEE 18th International Conference on Pattern Recognition, Hong Kong, 2006, pp. 824-827.
2. J.-W. Hsieh, "Morphology-based License Plate Detection from Complex Scenes," in Proceedings of the 16th International Conference on Pattern Recognition, Canada, 2002, pp. 176-179.
3. D. Zheng, "An Efficient Method of License Plate Location," Pattern Recognition Letters, vol. 26, pp. 2431-2438, 2005.
4. S.Kim, "A Robust License-plate Extraction Method under Complex Image Conditions," in Proceedings of 16th International Conference on Pattern Recognition, Canada, 2002, pp. 216-219.
5. S.Rovetta and R.Zunino, "License-Plate Localization by Using Vector Quantization," in Proceedings of the International Conference on Acoustics, Speech and Signal Processing, USA, 1999, pp. 1113-1116.
6. K. Debi, "Parallelogram and Histogram based Vehicle License Plate Detection," in Proceedings of the IEEE International Conference on Smart Manufacturing Application, Korea, 2008, pp. 349-353.
7. B.-H. Ron and J. Erez. (2002). A Real-time vehicle License Plate Recognition (LPR) System. Available: <http://visl.technion.ac.il/projects/2003w24/>
8. Matas and K. Zimmermann, "Unconstrained License Plate and Text Localization and Recognition," in Proceedings of IEEE International Conference on Intelligent Transportation Systems, Austria, 2005, pp. 572-577.
9. S.L.Chang, "Automatic License Plate Recognition," IEEE Transactions on Intelligent Transportation Systems, vol. 5, pp. 42-53, 2004.
10. Jia.W, "Region-Based License Plate Detection," Journal of Network and Computer Applications, vol. 30, pp. 1324-1333,
11. S. Thanongsak and C. Kosin, "Extracting of Car License Plate Using Motor Vehicle Regulation and Character Pattern Recognition," in Proceedings of the 1998 IEEE Asia-Pacific Conference on Circuit and Systems, 1998, pp. 559-562.
12. V.Abolghasemi and A. Ahmadyfard, "Improved Image Enhancement Method for License Plate Detection," in Proceedings of the 15th International Conference on Digital Signal Processing (DSP), Iran, 2007, pp. 435-438.