International Journal of Science and Engineering Applications
Volume 8–Issue 08,384-389, 2019, ISSN:-2319–7560

A Vehicle Detecting and Counting Technique Based on Digital Image Processing

Zo Nun Khuma
Department of Information Technology,
West Yangon Technological University, Myanmar

Kyu Kyu Than
Department of Information Technology,
West Yangon Technological University, Myanmar

Dr. Khaing Myat Mon
Department of Information Technology,
West Yangon Technological University, Myanmar

Abstract: Many large cities in Myanmar are suffering from severe traffic problems as their vehicle populations have been being unprecedentedly increase. Nowadays, due to busy schedule of work style, the people are facing heavy traffic problem. This research presents the automatic vehicles counting based on the density of vehicles in a particular road using image processing techniques. In this work, video image processing to extract video frame as input & using detector. The surveillance cameras are fixed at the top view on that road to capture all vehicle motions. Then, frame images are retrieved from the live video and vehicle regions are segmented applying Gaussian Mixture Model and Morphological operations such as opening and closing. After that, motion vehicle regions are appeared as foreground regions and counted. The accuracy of the proposed method will presents with various experiments of different roads.

Keywords: Digital Image Processing, Vehicle Detecting, Vehicle Counting, Gaussian Mixture Model, Morphological operations

1. INTRODUCTION

As digital cameras and computers have become wide-spread, the number of applications using vision techniques has increased significantly. One such application that has received significant attention from the computer vision community is traffic surveillance.

Video camera is a promising traffic sensor because of its low cost and its potential ability to collect a large amount of information (such as the number of vehicles, vehicles speed/acceleration, vehicle class, vehicles track) which can also infer higher-level information (incidents, speeding, origin-destination of vehicles, macroscopic traffic statistics and etc).

The video cameras (CCD or CMOS) are connected to a computer that performs images/video processing, object recognition and object tracking. Numerous research projects aiming to detect and track vehicle from stationary rectilinear cameras have been carried out in terms of measuring traffic performance during the past decades. It is widely recognized that vision-based systems are flexible and versatile in traffic monitoring applications if they can be made sufficiently reliable and robust. Some common and related works of vehicle detection and counting can be summarized as follow.

With respect to vehicle counting, a system of vehicle detection and classification was presented in [1]. The work used a self-adaptive background subtraction technique to separate vehicles from the background. The resulting connected regions are then tracked and counted over a sequence of images using a spatial matching method. The tracked regions are grouped together to form vehicles. Next adaptive background detection method which was also applied to identify vehicles was published in [2]. In this work, the vehicles were detected and counted based on contour extraction. Prewitt filter kernel was used for edge detection. The contour linking method used for connecting separated edge parts of the original object into one closed contour. A contour labelling method is used to mark and calculate the number of vehicles within frames. In [3], feature based vehicle detection and tracking algorithm had been used. Offline camera calibration had been carried out to detect the parameters such as line correspondences for a projective mapping, detection region and multiple fiducially points for camera stabilization. The projective transformation is necessary as the features are tracked in world coordinates to exploit known physical constraints on vehicle motion. The transformation is used to calculate distance based measures such as position, velocity and density. Another system for vehicle detection and counting was described in [4]. In this work, the Block Matching Algorithm (BMA), which is one of the motion estimation algorithm was applied in the MPEG compression standard video processing. BMA partitions the current frame in small, fixed size blocks and matches them in the previous frame in order to estimate displacement of blocks between two successive frames. BMA provided motion vectors, which are then regularised using a Vector Median Filter. After the regularisation step, motion vectors are grouped based on their adjacency and similarity, and a set of vehicles is identified per singular frame. Finally, the algorithm establishes the correspondences between the vehicles detected in each frames. It considered the BMA output as the basic tracking information associated with each block and combine this information with the already available block-level tracking as a grouped output in order to achieve the desired result.

This paper is intended to detect and count vehicles using some video processing and image processing techniques. In this work, surveillance cameras are fixed at the top view on a particular road to capture all vehicle motions. Then, frame images are retrieved from the live video and vehicle regions are segmented by combining Gaussian Mixture Model based foreground detector and some morphological operations. After that, motion vehicle regions are appeared as foreground regions and counted.

The paper is organized as five sections. Introduction and some of previous works are presented in section 1. Background Theory is shown in section 2. In section 3, the proposed methodology is presented. The section 4 shows experimental results and the last section 5 is conclusion.

2. BACKGROUND THEORY

The proposed technique is one of the vehicle detection and counting to determine traffic congestion on a particular road. In this technique, Gaussian Mixture Model, Morphological opening, closing, filling and regional labeling are used to count the number of vehicles on a particular road. Hence, these theories and operations are discussed as theories background in this section as the following.

2.1 Gaussian Mixture Model

A Gaussian Mixture Model (GMM) is a parametric probability density function represented as a weighted sum of Gaussian component densities. GMMs are commonly used as a parametric model of the probability distribution of continuous measurements or features in a biometric system, such as vocaltract related spectral features in a speaker recognition system. GMM parameters are estimated from training data using the iterative Expectation Maximization (EM) algorithm or Maximum A Posteriori (MAP) estimation from a well-trained prior model.

These component functions are combined to provide a multimodal density. They can be employed to model the colours of an object in order to perform tasks such as real-time colour-based tracking and segmentation [5]. These tasks may be made more robust by generating a mixture model corresponding to background colours in addition to a foreground model, and employing Bayes' theorem to perform pixel classification. Mixture models are also amenable to effective methods for on-line adaptation of models to cope with slowly-varying lighting conditions.

Gaussian mixture models can also be viewed as a form of generalized radial basis function network in which each Gaussian component is a basis function or 'hidden' unit. The component priors can be viewed as weights in an output layer. Finite mixture models can be formulated as follow.

Let the conditional density for a pixel ε belonging to a multi-colored Φ be a mixture with M component densities:

$$p(\varepsilon|\emptyset) = \sum_{j=1}^{M} P(\varepsilon|J)P(j) \qquad \text{-------- (1)}$$

Where, a mixing parameter *P(j)* corresponds to the prior probability that pixel ε was generated by component j and where $\sum_{j=1}^{M} P(j) = 1$. Each mixture component is a Gaussian with mean μ and covariance matrix Σ, i.e. in the case of a 2D colour space:

$$p(\varepsilon|j) = \frac{1}{2\pi|\Sigma_j|^{\frac{1}{2}}}\, exp^{-\frac{1}{2}(\varepsilon-\mu_j)\sum_{j=1}^{M}(\varepsilon-\mu_j)} \qquad \text{-------- (2)}$$

The Gaussian Mixture models are a semi-parametric alternative to non-parametric histograms [6] and provide greater flexibility and precision in modeling the underlying statistics of sample data. They are able to smooth over gaps resulting from sparse sample data and provide tighter constraints in assigning object membership to colour-space regions. Such precision is necessary to obtain the best results possible from colour-based pixel classification for qualitative segmentation requirements. The following figure shows the three color segmentation from original multi-colored image using GMM.

Figure 1 Description of GMM color segmentation (a) original image and (b) three colored segmented image [9]

2.2 Morphological Opening, Closing and Filling

Morphology: a branch of biology that deals with the form and structure of animals and plants. Morphological image processing 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. A morphological operation on a binary image creates a new binary image in which the pixel has a non-zero value only if the test is successful at that location in the input image. The fundamental operations of morphological image processing are dilation and erosion. There are morphological operations by combining erosion, dilation, and simple set-theoretic operations such as the complement of a binary image:

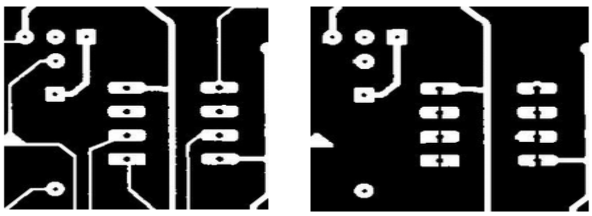
- Opening
- Closing
- Hit-miss transformation
- Thinning
- Regional Filling
- Boundary Extraction and etc.

2.2.1 Opening

In mathematical morphology, opening is the dilation of the erosion of a set A by a structuring element B:

$$A \circ B = (A \ominus B) \oplus B \qquad \text{-------- (3)}$$

Where ⊖ and ⊕ denote erosion and dilation, respectively.



Binary image *f* $f \circ s$ (5×5 square)

Figure 2 Results of Opening with a Square Structuring Element [11]

Opening removes small objects from the foreground (usually taken as the bright pixels) of an image, placing them in the background. Opening smooth the contour of an object, breaks narrow strips and eliminates thin protrusions (bulges). Hence, opening is so called because it can open up a gap between objects connected by a thin bridge of pixels. Any regions that have survived the erosion are restored to their original size by the dilation:

2.2.2 Closing

In mathematical morphology, the closing of a set (binary image) *A* by a structuring element *B* is the erosion of the dilation of that set,

$$A \cdot B = (A \oplus B) \ominus B \qquad \text{---------} \ (4)$$

Where $\ominus$ and $\oplus$ denote erosion and dilation, respectively.

Closing is so called because it can fill holes in the regions while keeping the initial region sizes. Like opening, closing is dual operation of opening. In image processing, closing is, together with opening, the basic workhorse of morphological noise removal. Opening removes small objects, while closing removes small holes. Closing can sometimes be used to selectively fill in particular background regions of an image. Whether or not this can be done depends upon whether a suitable structuring element can be found that fits well inside regions that are to be preserved, but doesn't fit inside regions that are to be removed.

2.2.3 Regional Filling

Region filling is a Morphological algorithm in image processing, which deals with filling the region in the image with some colours. In a binary image, *A* denote the boundary points of a region and *p* is a point inside the boundary. To fill the entire region with 1s,

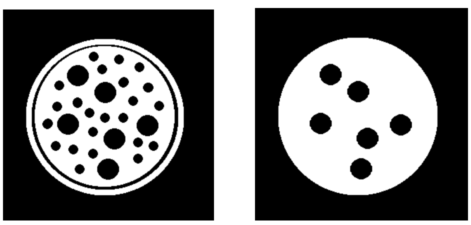
$$X_k = (X_{k-1} \oplus B) \cap A^c \qquad \text{--------} \ (5)$$

Where, $X_0 = p$, B is the structuring element

Algorithm terminates if $X_k = X_{k-1}$ and Union of X_k and A is the region filled.

The image region can be selected in two ways:

- Interior region and
- Boundary region.



Binary image *f* $\quad$ $f \cdot s$ (22 pixel diameter disk)

Figure 3 Results of closing with a disk structuring element with 22 pixel diameter [12]

Interior regions are defined by assigning the same value to all the pixels inside that region. The algorithms used to change the values of all pixels in the interior regions to new values are FLOOD-FILL algorithms.

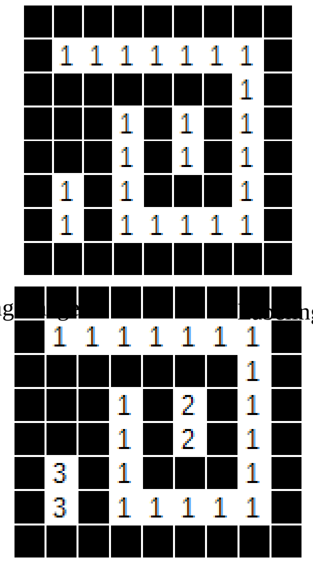
Boundary regions are defined by assigning the same value to all the pixels on the boundary of the region. Boundary region pixels and the interior region pixels should not have the same values. The algorithms used to change the value of all pixels in boundary regions to new value are BOUNDARY-FILL algorithms.

2.2.4 Regional Labeling

Regional is an algorithmic application of graph theory, where subsets of connected components are uniquely labeled based on a given heuristic.

In the labeling process, connected components are scanned in an image and groups its pixels into components based on pixel connectivity, i.e. all pixels in a connected component share similar pixel intensity values and are in some way connected with each other. Once all groups have been determined, each pixel is labeled with a gray level or a color (color labeling) according to the component it was assigned to.

Regional labeling is used in computer vision to detect connected regions in binary digital images, although colour images and data with higher dimensionality can also be processed. When integrated into an image recognition system or human-computer interaction interface, connected component labeling can operate on a variety of information. Blob extraction is generally performed on the resulting binary image from a thresholding step, but it can be applicable to gray-scale and color images as well.



Thresholding $\quad$ ng Image

Figure 4 Results of Regional Labeling Process [13]

The above expressions are the fundamental theories of proposed technique in detecting and counting vehicles on a particular road. Using these theories, the following proposed method is implemented.

3. PROPOSED METHOD

This proposed method is divided into three main portions, image acquisition, pre-processing, vehicle detection and counting. The system flow diagram of the proposed method can be summarized as follow figure 5.

3.1 Frame Image Acquisition

Input video is captured by fixed camera which is about 6.5 m above a particular road. It can capture a video using 30 frame per second. It coverage area is 50 feet of width and 100 feet of length. In this proposed method, not all frame with a second are processed. Only three frames (1st frame, 15th frame and

30th frame) are processed within a second to speed up vehicle detection process. Hence, in every 1/3 second, input frame image is captured by snapshot via camera and it is ready to later process.

3.2 Pre-processing

In detecting motion vehicles on the road, there are many challenges to confuse this detection process. Among them, shadows, light occlusions, dusts and other unwanted motion such as peoples and animals can be confused in detecting actual vehicle motions. To eliminate this unwanted environmental effects, pre-processing is necessary before motion detection.

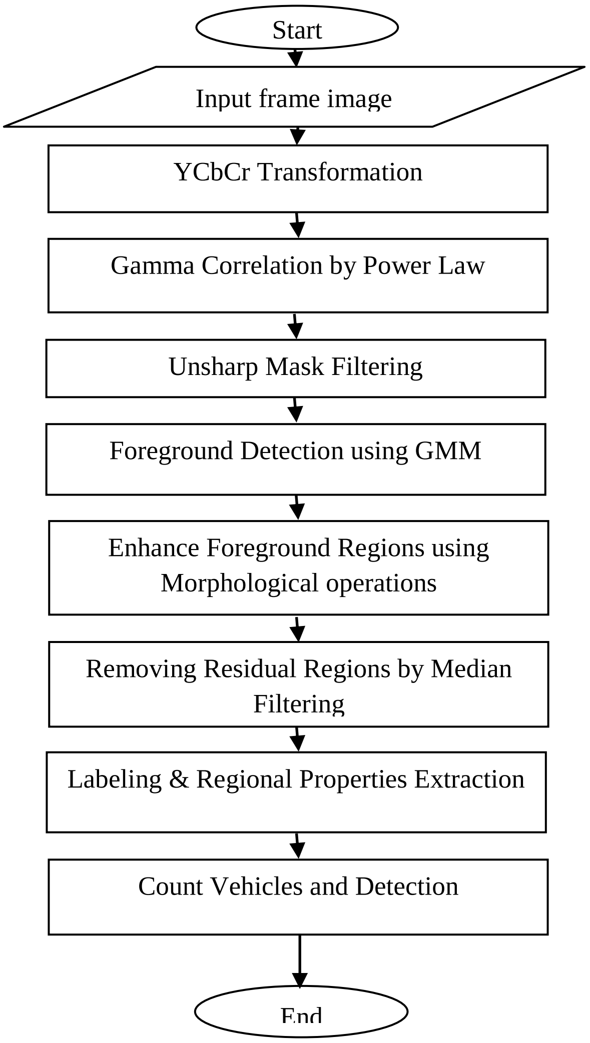


Figure 5 System flow diagram of proposed system

In this proposed system, input color image is converted into luminance Chrominance (YCbCr) color model and only used luminance channel to eliminate unwanted lighting occlusions. This channel image is only presented as intensity image. To eliminate unwanted light shadows and other wrong exposure effect, gamma correlation is adjust by power law transformation. It is very fast and ease technique to remove light shadows and eliminate heavy shadow. After that, to enhance the edges and textures of input image, unsharp mask filtering is applied. This filtering is actually the combination of original image and its edge image. Hence, the filter can give sharper edge image as its output. After finishing these pre-processing operations, the image is ready for later vehicle detection processes.

3.3 Vehicle Detection and Counting

On the pre-processed frame image, foreground objects are detected and trained using Gaussian Mixture Model of Computer Vision Toolbox. The Gaussian Mixture Model (GMM) is already described in background theory about why it is use in this process.

Next, after the foreground objects detection using foreground detector, the gray scale frame image is changed again edge image. In the edge image, there are unconnected or broken edges and isolated edges. To connect broken edges and remove isolated edges, some morphological image processes are needed to perform. Hence, to modified and enhance the edges, morphological image operations such as “closing” and “opening” are applied. The first “closing” function can connect broken edges and the follower “opening” function can remove isolated edges. Thus, there are all connected edges are remained on the edge image.

Among them, some are closed boundary so they are needed to fill as regions. To fill close boundaries, the morphological operation, “image filling” is used. In the edge image, there may be more than one closed boundaries. These closed boundaries can be seen as regions. In these regions, to remove smaller regions and unwanted residual edges, median filter with larger kernel size (17x17) is used. After finishing the above processes, all of the black background objects except motion region can be removed on the frame image. By this ways, the motion regions are appeared as white regions on the black background.

The next process is to labeling the foreground motion region and produce regional properties, area, BoundingBox and etc. In this proposed system, the Matlab command “bwlabel()” is used to label the foreground regions. The operation of this function has already discussed in theories background section. The command is label all connected components of the foreground regions. The Matlab command “regionprops()” is used to measure each regional properties to point out motion object with BoundingBox. By using these two processes, the proposed system can retrieve geometrical features, regional area, BoundingBox of each foreground region from the static background. The numbers of foreground region can describe the numbers of vehicles. The proposed technique counts the numbers of vehicles by depending on the foreground region and its area. Sometime, the foreground region may be too large when two or more vehicles are very close. Hence, the proposed method is defined as minimum (1800 pixels) and maximum (1900 pixels) threshold area to describe whether a foreground region is a vehicle or not. As above expressions, the proposed method can count the numbers of vehicles for a particular lane on road. The accuracy and performance of the proposed technique can be seen with many experimental results.

4. EXPERIMENTAL FRSULT & DISCUSSION

This section is mainly focus on the performance and accuracy measurement of the proposed technique. The accuracy measurement is estimated by the following formula.

$$Accuracy\ (\%) = \frac{numbers\ of\ correct\ tests}{numbers\ of\ total\ tests}\ x\ 100$$

To measure the performance and accuracy of the proposed technique, there are 20 testing in three different places which are main roads of Yangon. The following are some results images of vehicle detecting and counting.

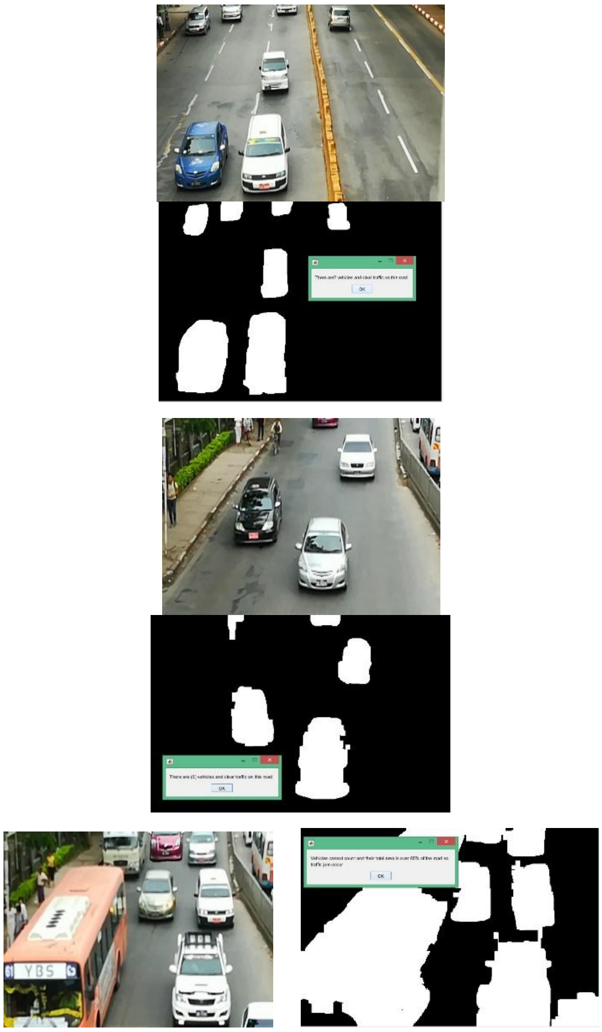


Figure 6 Some Experimental Results of Vehicle Detection and Counting of the Proposed Technique

In the above figure 6, the first two pair of figures are shown count the numbers of vehicle. In the lower traffic jam figure, vehicles cannot count because these vehicles are very close each other.

Table 1 Accuracy Measurement of Vehicle Counting And Traffic Jam Determination

No	**Testing**	**Actual no; of vehicle**	**Vehicle counting by proposed system**	**Accuracy of vehicle counting**
1	Test 1	7	7	100%
2	Test 2	5	5	100%
No	**Experiment**	**Actual no; of vehicle**	**Vehicle counting by proposed system**	**Accuracy of vehicle counting**
3	Test 3	9	7	78%
4	Test 4	6	6	100%
5	Test 5	5	5	100%
6	Test 6	7	6	86%
7	Test 7	3	3	100%
8	Test 8	4	3	75%
9	Test 9	7	7	100%
10	Test 10	9	9	100%
11	Test 11	8	9	88%
12	Test 12	4	4	100%
13	Test 13	6	6	100%
14	Test 14	5	5	100%
15	Test 15	3	3	100%
16	Test 16	7	7	100%
17	Test 17	12	4	33
18	Test 18	16	7	43
19	Test 19	11	7	64
20	Test 20	9	5	65
	Over all accuracy			86.15%

To determine vehicle counting condition, the total lane area is pre-defined as 195086 reference unit (pixels) which is measured from figure. According to the segmented figure, total foreground area is 185328 pixels so it is over 85% of total road area. Hence, the proposed method determine that this situation of the place is occurred traffic jam. The following table show the accuracy of vehicle counting.

According to the above results and accuracy of table 1, the proposed technique achieved 86.15% of vehicle counting accuracy. In vehicle counting, the proposed technique make wrong counts when two or more vehicle are very close each other. At this situation, in the foreground object extraction, the close vehicles are remarked as one region so it gives wrong results. However, vehicles are always moving in every frame image, sequentially so these vehicle can definitely count other frame although current frame give wrong results. By the above 20 testing results, the proposed method give acceptable results and it can be reliable.

5. CONCLUSION

The proposed technique for automatic vehicle detecting and counting has been discussed and presented. According to the experimental results, the proposed technique gives over 86% accuracy in vehicle counting and detection. However, the proposed sometime give wrong results when two or more vehicles are very close together. Furthermore, vehicle counting and detection is based on segmented region so it can confuse and give wrong results whenever camera is shake or sometime light occlusion is occurred. To be more accurate, the vehicle classification method should be used. The research area of Intelligent Traffic Monitoring and Transportation Management is very interesting and currently on demand. Hence, our researches will continue on this area by trying better results and performance.

6. REFERENCES

[1] Gupte, S. “Detection and Classification of Vehicles”, University of Minnesota 2002.

[2] Atkociunas, E. "Image Processing in Road Traffic Analysis", Vilnius University, 2005

[3] Coifman, B. "A Real-Time Computer Vision System for Vehicle Tracking and Traffic Surveillance" University of California, 2005.

[4] Di Stefano, L. "Vehicle Detection and Tracking Using the Block Matching Algorithm", Department of Electronics, Computer Science and Systems (DEIS) University of Bologna, ITALY.

[5] Raja, Y. McKenna, S. and Gong, S. “Segmentation and tracking using colour mixture models,” in Asian Conference on Computer Vision, Hong Kong, January 1998.

[6] Bishop, C. “Neural Networks for Pattern Recognition,” Oxford University Press, 1995.

[7] https://kittipatkampa.wordpress.com/2011/02/17/image-segmentation-using -gaussian-mixture-models/

[8] https://www.mmorph.com/html/morph/mmopen.html/

[9] https://homepages.inf.ed.ac.uk/rbf/HIPR2/close.htm