An Innovative Vehicle Detection Approach Based on Background Subtraction and Morphological Binary operations Methods

[1,2]Raad Ahmed Hadi, [1]Ghazali Sulong, [3]Loay Edwar George

[1]Faculty of Computing, UniversitiTeknologi Malaysia, Skudai, 81310 Johor, Malaysia
[2]Faculty of Software and Networking Engineering, College of Engineering, Iraqi University, Baghdad, Iraq
[3]Faculty of Computer Science, College of Science, Baghdad University, Baghdad, Iraq
raadwap@yahoo.com

Abstract: This paper proposes a new method of detecting moving vehicles in traffic videos using the background subtraction method, morphological binary operations, and new detection zone technique. Firstly, this method extracts a background image from the video frames using the mode statistical method, wherein the background will be subtracted from subsequent frames to distinguish the foreground objects by using the background subtraction method. Secondly, the morphological binary dilation and erosion operations are used to refine the boundaries and the regions of the detected moving vehicles (foregrounds), and unwanted small objects will be removed from the background respectively. Finally, we adopt the concept of a switch electric circuit design SPST (Single-Pole Single-Throw) as a new method to detect and count the moving vehicles. Performance evaluation of the experimental results is encouraging in that it shows that the proposed detection method has an average precision of more than 0.92, an average recall of more than 0.97, an average f-measure of more than 0.94 and average accuracy of more than 0.99.

[Hadi R A, Sulong G, George L E. **An Innovative Vehicle Detection Approach Based on Background Subtraction and Morphological Binary Methods.** *Life Sci J* 2014;11(8):230-238]. (ISSN:1097-8135). http://www.lifesciencesite.com. 31

Keywords: Vehicle detection; Background subtraction; Morphological binary operations; Detection zone.

1. Introduction

One of the significant applications of video-based supervision systems is traffic surveillance. For many years researchers have investigated the Vision-Based Intelligent Transportation System (ITS), transportation planning and traffic engineering applications to extract useful and precise traffic information for traffic image analysis and traffic flow control such as; vehicle count, vehicle trajectory, vehicle tracking, vehicle flow, vehicle classification, traffic density, vehicle velocity, traffic lane changes, license plate recognition, etc. (Chung-Lin and Wen-Chieh, 2004; Hadi, 2014; Neeraj K. Kanhere, 2008; N. K. Kanhere and Birchfield, 2008; Wei *et al.*, 2008). However, the on-road vehicle detection applications may be in decline and not well recognised due to the vehicles being obstructed by other vehicles or other visual obstacles such as road signs, trees, weather conditions and so on, making the performance of these applications dependent on a good traffic image analysis approach to detect, track and classify the vehicles.

Vehicle appearance may differ in size, shape and colour. In addition, this appearance relies on its position, which is influenced by close objects, cluttered background and difficult outdoor situations (e.g. lighting conditions) (Zehang *et al.*, 2006). For a long time, various methods for vehicle detection based on background subtraction, frame (temporal) differencing and feature-based methods have been used in digital-video-based traffic surveillance; however, these methods were significantly affected by variations in vehicle appearances that tended to reduce the efficiency of the detection process (De Silva, 2001; Gupte *et al.*, 2002; Hsieh *et al.*, 2006).

2. Related Work

Detection of moving vehicles in traffic surveillance video streams has been a wide domain in computer vision researchers. There are various methods based on image and video processing which have been proposed to detect moving objects (e.g. vehicles) that use different methodologies such as thresholding, edge detection, background subtraction, frame (temporal) differencing and multi-resolution processing (Kastrinaki *et al.*, 2003; Mandellos *et al.*, 2011).

Thresholding was used in the 1970s and the 1980s which was presented as a part of the very first automatic observation systems, but it had low accuracy (Mahmassani *et al.*, 1999).

An edge of vehicle detection approach, based on sobel operator edge detection was proposed by (Weihua, 2009). This approach was used for vehicle type recognition through video sequences for a short period of time. It employed the vehicle region vectors as features extracted which were used for detection and similarity process. The main advantage of the edge-based method is that the extracted features from shapes are invariant and to scale, but they suffer from noise and inaccurate estimation of image gradient. In addition, it is difficult to extrapolate the vehicle shapes in congested scenes (Dieter Koller *et al.*, 1994; Weihua, 2009).

The process of extracting moving foreground objects (input image) from a stored background image (static image) or generated background frame from an image series (video) is called background subtraction. This method is one of the most widely used in vehicle regions detection and represents a very well-suited method of detecting foreground objects, but it needs the reference image (empty background image) which is used for the subtraction process (Huwer and Niemann, 2000). Also, the non-adaptive characteristic in the background subtraction method is a drawback which is raised due to the changes in the lighting and the climate situations which necessitates continuous updating (Vasu, 2010). So, several researchers have worked to resolve this drawback by proposed methods in this field (Huang *et al.*, 2012; Iwasaki and Itoyama, 2006; Mandellos, *et al.*, 2011).

Also, the motion trait is another fundamental step in detecting a vehicle in a series of images which is done by isolating the moving objects (blobs) through analysis and assigning sets of pixels to different classes of objects which are based on the orientations and speed of their movements from the background of the motion scene image sequence. This process is called frame differencing (Han *et al.*, 2007; D. Koller *et al.*, 1994; Vasu, 2010; Zhang *et al.*, 2010). Temporal difference is also used to detect the region of interest objects well and accurately without need for a background image. Despite some improvement techniques (Wei, *et al.*, 2008; Yen-Lin *et al.*, 2009; Zhang, *et al.*, 2010) this approach cannot suitably handle realistic traffic situations when there is no vehicle movement in the image for a period of time.

Multi-resolution is a power tool used in image processing which is based on scale space theory. It uses the fine and coarse level colour pixel knowledge for clustering and separating the objects from a background image (Lindeberg, 1996).

Lane marks detection methods are also used for vehicle detection. They analyse and calibrate the road region coordinates based on extracting the road features, and use lane geometrical models to detect vehicles. These methods have highly complex computations and are also affected by shadows and high contrast variations (Lim *et al.*, 2009).

In our paper, we have proposed an innovative approach based on a set of steps: the background reconstruction is used to mention dramatic changes in background due to outdoor variations such as lighting conditions and to generate a background image. In addition, use of background subtraction (image differencing) and thresholding operations are to distinguish the moving vehicles (foreground objects) from the background. Also, refining these foreground objects' regions and borders by using dilation process and removing the unwanted small objects. Finally, a new technique which is called detection zone, will be used to detect and count the moving vehicles.

The rest of the paper is organised as follows: related work is described in section 2. The moving vehicle detection approach is illustrated in section 3. In section 4, experimental results are presented. Finally, conclusion and future work are summarised in section 5.

3. Moving Vehicle Detection Approach

The proposed approach consists of background reconstruction module; foreground detection module and detection zone (Figure 1). This approach will be tested on baseline highway traffic video frames of the change detection data set.

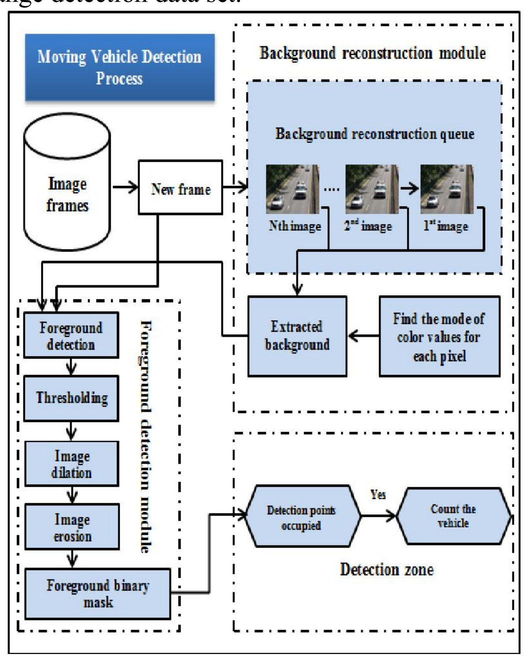

Figure 1: Flowchart of proposed approach

3.1. Background Subtraction

Background subtraction is a simple and effective computer vision computational procedure which is used to extract the foreground objects from a certain view. The extracted important information (foreground object) will be described as an interesting object which helps in decreasing the quantity of data to be managed for specific task consideration. Background subtraction encompasses two substance processes: the background reconstruction module and the foreground detection/extraction module.

3.1.1. Background Reconstruction Module

In the previous flowchart, the video frames passed to the background reconstruction module to extract an empty (reference / background image) by using the statistical mode method. This module manipulates 150 frames which take approximately 5 seconds in processing time to generate the background image (Figure 2).

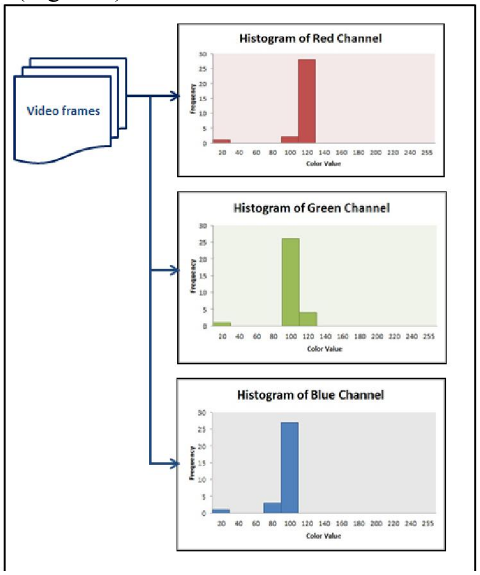

Figure 2: Histogram of mode method of video frames

As mentioned before, to get a clear background image, it uses the mode statistical method (to extract the background pixels/ to eliminate the moving vehicles and illumination changes effects) which is adapted from (Zheng *et al.*, 2006). In each traffic image frame, the pixel colour values at specific positions and times are referred to as a I(*x*,*y*,*t*) where x and y are the traffic image frame coordinates and t is the time interval. The colour space will be used for pixels' colour values, which are Red, Green, Blue (RGB) colour space for chromatic images and grey-scale for monochrome images.

For a given traffic image frames is represented as a matrix of pixel values Mn(x,y) at different times (ti, ti-1 … ti-n) as mentioned in equation (1) below:

Mn={(IR(x,y,ti),IG(x,y,ti),IB(x,y,ti)), (IR(x,y,ti -1),IG(x,y,ti -1),IB(x,y,ti -1))……(IR(x,y,ti – n),IG(x,y,ti -n),IB(x,y,ti -n))} or

Mn={IGr(x,y,ti),IGr(x,y,ti -1)……IGr(x,y,ti -n)} (1)

Where (M) is the matrix of pixels' colour values, (n) is the number of traffic image frames, (I) is the image frame, (R) red, (G) green, (B) blue and (Gr) grey colour values respectively.

However, the mode statistical operation method will compute each pixel at a definite prior location for a set of traffic image frames at different times and produces the most frequent pixel colour value, which is considered as a background pixel colour value for those frames mentioned before (Figure 3).

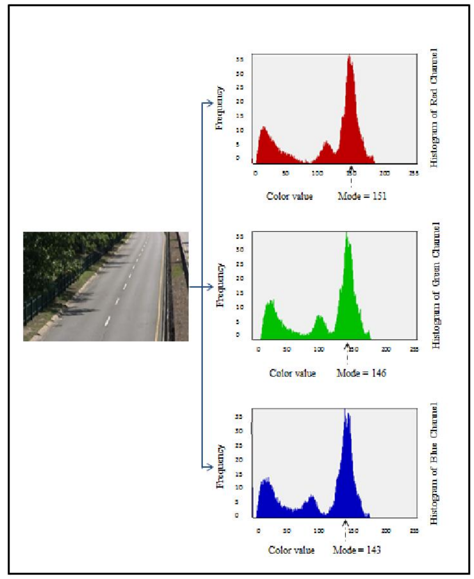

Figure 3: Background reconstruction using mode statistical method

So, let (*S*) be the accumulative ascending or descending sorted matrix of Mn of pixel values of image frames, so, the equations (2) and (3) are examined the mode statistical method below:

$$S = \sum_{i=0}^{n} x_i y_i \qquad (2)$$

$(x = 0, 1, 2\ldots k)$ $(y = 0, 1, 2\ldots k)$

$(k \subseteq \mathbb{Z})$ $\mathbb{Z}$: Integer numbers

And the mode of *S* is:

$$mfv = \lfloor f \rfloor,\ f \in S \qquad (3)$$

Here, *mfv* is the most frequent colour value of pixel at a specific location and time, and (*f*) is the frequency of colour value (*x*,*y*) in the traffic image series *M*n(*x*,*y*). Algorithm 1 depicts the previous background reconstruction method.

Algorithm 1 To extract the empty background from bundle of frame images

Input: Load frame images in list of images $LOI[n]$

Output: $R[q]$, $G[w]$ and $B[e]$ are matrices of RGB three channels of frame images

```
img ⟵ the first image
for y ⟵ 0 to img.height do
  for x ⟵ 0 to img.width do
    q = 0;
    w = 0;
    e = 0;
    for i ⟵ 1 to LOI.length do
      currentpixel = image[i].getpixel(x, y);
      R[q] = currentpixel.Redchannelvalue;
      G[w] = currentpixel.Greenchannelvalue;
      B[e] = currentpixel.Bluechannelvalue;
      q = q + 1;
      w = w + 1;
      e = e + 1;
      i ⟵ i + 1
    end for
    x ⟵ x + 1
  end for
  y ⟵ y + 1
end for
modeR = mode(R[q])
modeG = mode(G[w])
modeB = mode(B[e])
```

3.1.2. Foreground Detection Module

After the initialisation of the background model using the statistical method in the previous section, the next stage is to employ the new foreground detection module for foreground/background pixel classification by using the subtraction process between the current image frame and the background model.

The vital objective of background subtraction is to "subtract" the background pixels in a scene and leave the pixels of foreground objects of interest. Afterwards, the difference in results will be compared with a pre-defined threshold (Th) based on the histogram characteristic of a produced subtracted image. If the difference is larger than (Th), then this pixel will be classified as foreground; otherwise, it will claim that it is background. At run time, the background model image will be used as a reference image $R_{img}(x,y)$ and subtracted from the frame image being processed from video that it is denoted as $F_{img}(x,y)$. The absolute difference values between both previous images at time t are defined as follows in equation (4):

$$D_{img}(x,y) = |IF_t(x,y) - IB_t(x,y)| \quad (4)$$

Where $IF_t(x,y)$ and $IB_t(x,y)$ are the intensities of pixels (x,y) in the $F_{img}(x,y)$ and $R_{img}(x,y)$ respectively. $D_{img}(x,y)$ is the difference image (Figure 4).

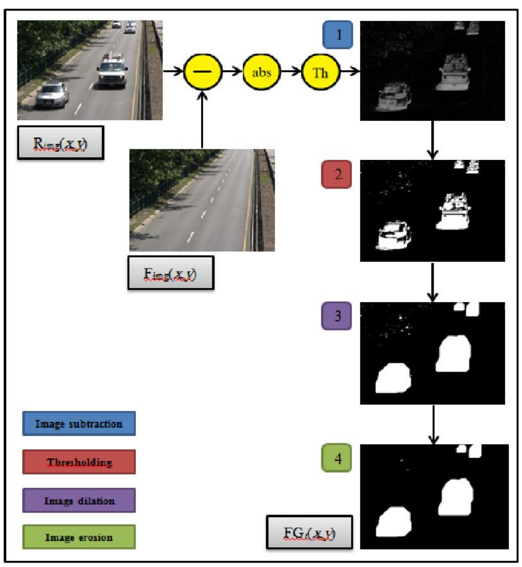

Figure 4: Illustration of foreground detection module

Next, the absolute difference values will be compared with a pre-defined threshold Th, if larger than Th; the pixels value set to 1 which are considered as having important motion, otherwise it is set to 0 which are considered as "stable" pixels of background (as shown in algorithm 2). In relation to threshold value preparing, the Th is determined by analysing the image (chromatic/grey-scale) histogram. In addition, Th is also tested empirically to fit the motion in the frame (here it is set to 50) and gets a good foreground binary mask $FG_t(x,y)$ as mentioned in equation (5):

$$FG_t(x,y) = \begin{cases} 1 & \text{if } FG_t(x,y) > Th \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Algorithm 2 Image subtraction and binarization processes

Input: Load reference/background image R_{img}, frame image F_{img} and Th is the threshold value as it follow: $Th = histogram(D_{img})$

Output: FG is a foreground binary mask image

```
for y ⟵ 0 to R_img.height do
  for x ⟵ 0 to R_img.width do
    if | F_img − R_img | > Th then
      FG = 1;
    else
      FG = 0;
    end if
    x ⟵ x + 1
  end for
  y ⟵ y + 1
end for
```

Meanwhile, some linear morphological binary image operations will be applied to regions of foreground pixels to highlight those objects and to refine the background from some spots or undesirable objects which have an influence on vehicle detection and segmentation processes (described in algorithm 3 and 4). Here, two morphological binary image operations (dilation and erosion) will be used in three directions: horizontally, vertically and through 45 degrees (diagonally) and these direction measures represented in eight pixel connectivity, which is called the (*structure element*).

Algorithm 3 Dilation process

Input: Load the foreground binary mask image FG

Output: FG is a foreground binary mask image

```
for j ⟵ 0 to R_img.height do
  for i ⟵ 0 to R_img.width do
    c = 0;
    if FG = 1 in one of eight directions (Structure Element) then
      c = c + 1;
      if c >= 4 then
        FG = 1;
      end if
    end if
    i ⟵ i + 1
  end for
  j ⟵ j + 1
end for
```

Algorithm 4 Erosion process

Input: Load the foreground binary mask image FG

Output: FG is a foreground binary mask image

```
for j ⟵ 0 to R_img.height do
  for i ⟵ 0 to R_img.width do
    min = 1;
    temp = FG;
    if temp < min in eight directions (Structure Element) then
      min = temp;
    end if
    i ⟵ i + 1
  end for
  FG = min;
  j ⟵ j + 1
end for
```

After the binarisation (thresholding) process, the image will contain unwanted small objects, noise objects and the vehicles' parts will appear as disconnected (with a set of background pixels which are called *holes*) or broken objects due to previous processing steps, and these problems lead to false detection. Therefore, to have solid objects the dilation process is used to fill the holes and these unwanted small objects will be removed by an erosion process. At this point, we will get a refined image and the vehicle objects are prominent without unwanted small objects or noise and ready for the detection process.

3.1.3 Detection Zone

For a more effective performance in detection of the vehicles in traffic, the vision-based traffic surveillance systems require reliable and adequate information (e.g. vehicle counting, speed, etc.) which is extracted from defined regions which are called detection zones (DZ).

In this paper, a newly proposed two-dimensional (2D) detection zone model was constructed and used to detect and count vehicles in the traffic. The DZ's idea was inspired by the SPST (Single-Pole Single-Throw) switch electric circuit design (Figure 5). This design uses the concept of on-off control logic for the electric circuit. A toggle switch is used as a rocking mechanism, handle, or mechanical lever to (connect\close) or (disconnect\open) one electric circuit between two terminals for flowing the electric current (Dickon Ross, 2010).

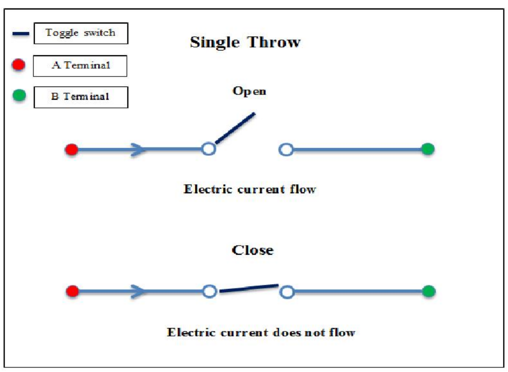

Figure 5: SPST circuit design

As mentioned before, the DZ represents as an electric circuit; a 2D rectangle model will be drawn on the road with adequate coordinates to detect the moving vehicles and put in a desirable location on the road to facilitate a good detection process. Also, there are determined points which are used as a detection sensor or entrance gate in the DZ called the detection or toggle points. These points send a signal or command to the DZ to detect the moving objects, when all the points observe a change in colour values (white or black) at motion state in traffic scene sequences (Figure 6).

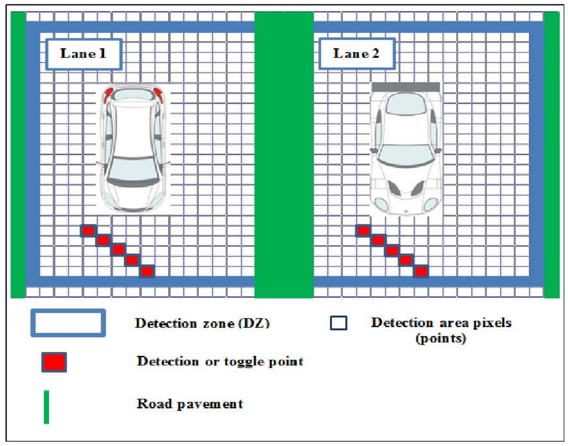

Figure 6: Detection zone design

For above discussed 2D detection model, it comprises of a set of pixels (detection area) which is specified by the width w and height h dimensions of the DZ (parallel to X and Y axis of traffic image frame respectively).

Detection area points P_{ab} are arranged in rows $R_y = h - 1$ and columns $Cx = w - 1$: $a = 0, 1\ldots Cx$, $b = 0, 1\ldots R_y$. The values of R_y and Cx are determined by the size (w * h) of the detection zone and its location in the traffic image frame. In addition, the key idea of toggle points (TPs) are simulated as an electric circuit switch toggle handle, when the vehicles hit all detection points, the command has already been sent to the DZ to discover the moving objects inside it. These points are modelled as a set of five points (xtp_c,ytp_c), where xtp_c and ytp_c are coordinates of the detection points and c $\in$ {1, 2, 3, 4, 5} and it will be placed into DZ on the road with ladder style towards the vehicles' movements. The traffic road may have one or more lanes (li), i $\in$ {1, 2, 3 … n} which contain a set of TPs for each lane.

The detection zone depends on the toggle points to detect the vehicles. When the vehicles pass over these points, they will trigger the DZ to discover the whole moving object. To discuss how the mechanism of this technique works, the probability of passing one or multi-vehicles in the traffic is unknown previously, so for more effective discovering and numbering of the vehicles, the TPs will play an important role in the detection process by observing the motion state of vehicles (Ψ) through changing all the pixels to a white colour value (Figure 7).

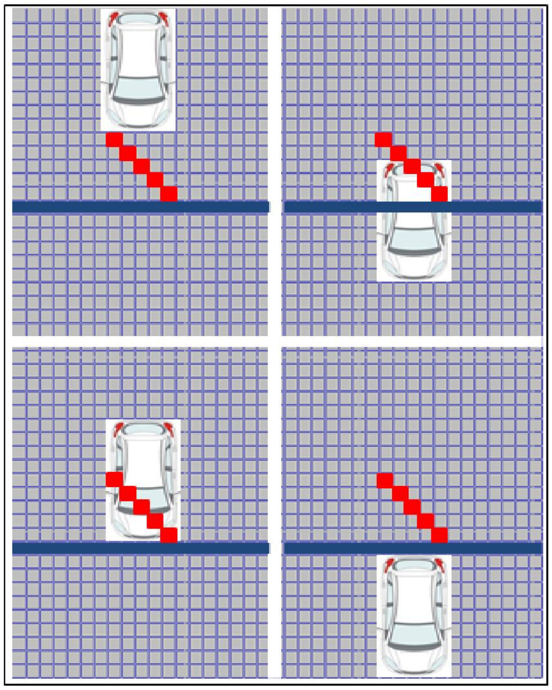

Figure 7: Vehicle detection steps through DZ

When the colour values of points are one (white), it means that this vehicle is detected newly and will be numbered and still in the detection process until the TPs' colour values change to zero (black). The procedure is in equation (6) presented below:

$\forall (xtpc, ytpc) \in$ TP: TP=$\{0, 1\}$, c = $\{1, 2, 3, 4, 5\}$

$$\Psi(xtpc, ytpc) = \begin{cases} \text{Motion} & \text{if } (TPs = 1) \\ \text{Motionless} & \text{if } (TPs = 0) \end{cases} \quad (6)$$

Meanwhile, as every vehicle enters the detection zone, the region growing algorithm will be used to segment and cluster the white pixel areas of vehicles (blobs) from black pixel areas of background image. Region growing is represented as a heuristic method to cluster segmented regions based on their locations. When the vehicle hits the five toggle points (TPs), it will prompt these points to give a command to segment this region.

So, if the moving vehicle is still going over these points, the detection and segmentation processes will stay in idle state at a dedicated lane and will not send a prompt to detect and segment until the TPs are clear of motion of the vehicle. Algorithm 5 describes the detection zone technique process.

The detection zone forms the area of counting the vehicles. Afterwards, every new blob will be numbered and registered in counter buffer if the TPs are checked in state 0, which means a vehicle encountered it, or else it will not be considered new and is dealt with as part of a previously existing vehicle and the presence of the blob is neglected; which means the state of TPs is still 1.

Algorithm 5 Detection Zone (DZ) technique algorithm

Input: FG is a foreground binary mask image

Output: Vehicles sizes and counting

```
Determine how many lane in the street for example (two lanes)
STV1 ⟵ a trigger used to mention the motion state of vehicle
STV2 ⟵ a trigger used to mention the motion state of vehicle
V_counter ⟵ a counter buffer

Lane One:
if TP_s = 1 then
  Region growing segmentation RG();
  if STV1 = False then
    STV1 = True;
    V_counter = V_counter + 1;
    Assign the coordinates of vehicle size in (x1, y1, xx1, and yy1)
    parameters
  end if
else if TP_s = 0 then
  STV1 = False;
end if

Lane Two:
if TP_s = 1 then
  Region growing segmentation RG();
  if STV2 = False then
    STV2 = True;
    V_counter = V_counter + 1;
    Assign the coordinates of vehicle size in (x2, y2, xx2, and yy2)
    parameters
  end if
else if TP_s = 0 then
  STV2 = False;
end if
```

4. Experimental Results and Discussions

We tested our approach on a baseline highway traffic video of the standard change detection data set. This video was captured using a digital camera in RGB colour space. The video consists of 1,700 frames and the video frame size is 320 X 240 pixels of resolution. This approach was implemented on a Laptop Core i3 (2.13 GHz) processor with (4 GB) of memory using Microsoft Visual C#.

In Figure 8, the results present some of the frames in which the vehicle detection and counting is applied on 1,230 frames of abovementioned dataset. We will use four measures for our data test: true positive in the dataset which means the correct number of pixel detected of vehicles (TP), false positive means the number of pixel detected as background that should have been represented as detected foreground of pixels (FP), true negative means the number of pixel detected as a background (TN) and false negative means the number of pixel detected as foreground that should have been represented as a detected background of pixels (FN).

Table 1: Our measurements result

Measures	TP	FP	FN	TN
No. of pixels	5238391	434572	133410	88734427

The previous measures used to calculate the following metrics as defined in (Goyette *et al.*, 2012):

$$\text{Precision (Pr)} = \frac{TP}{TP + FP} \quad (7)$$

$$\text{Recall (Re)} = \frac{TP}{TP + FN} \quad (8)$$

$$\text{F-measure} = \frac{2 * Precsion * Recall}{Precsion + Recall} \quad (9)$$

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \quad (10)$$

Reference frame Ground truth Our result Overlapped result

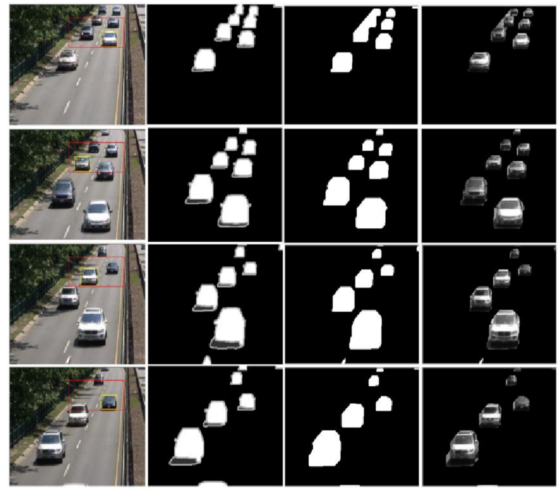

Figure 8: Results of proposed approach on baseline highway traffic dataset

Table 2: Metrics of our approach on baseline highway traffic data set

Methods	**Pr**	**Re**	**F-measure**	**Accuracy**
Our method	0.923396	0.975165	0.948575	0.993992
SC-SOBS	0.934713	0.956528	0.945495	0.993465
CwisarD	0.905885	0.93112	0.918329	0.990187
Local-self similarity	0.863162	0.981306	0.91845	0.989674
GMM|KaewTraKulPong	0.908275	0.649624	0.757478	0.975352

As shown in table (2) and Figure 9, the proposed approach presents promising results of performance metrics (Precision, Recall, F-measure and accuracy) compared to other methods, which are applied on the same video to detect changes in frames due to moving of vehicles.

Here, the shadow removal and overlap of vehicles' situations are not considered in this proposed solution. Therefore, we will make enhancement about the shadow and overlapping of vehicles and this should be worked on in the future.

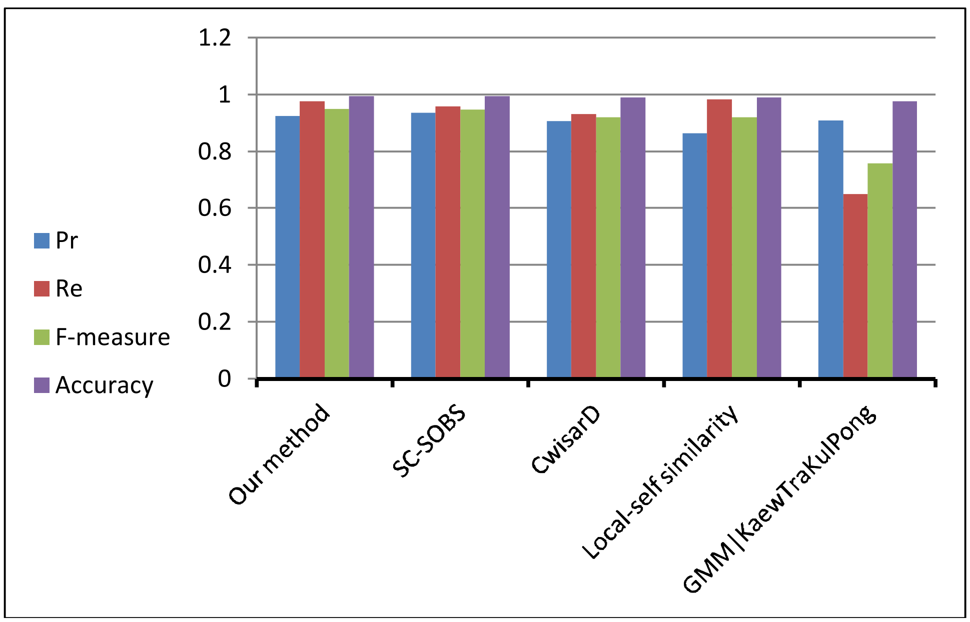

Figure 9: Performance evaluation of our approach with other methods

5. Conclusion and Future Work

A newly proposed approach based on the background subtraction method and morphological binary methods is suggested as a way to distinguish and number the moving foreground objects (vehicles) from a stationary background image. We have constructed a new model inspired by an electronic circuit switch design (SPST), which will be used to classify the pixels patches and determine whether they belong to the foreground or the background. In addition to abovementioned new model, the background reconstruction and foreground detection modules are used together with the morphological binary operations (dilation and erosion) in the proposed approach. The results show a worthy detection process with encouraging performance metrics values in this paper. As a future direction for this work, we will be working on making this approach more resilient to be applied and solve the issues of overlapping detecting and shadow removal.

References

1. Chung-Lin, H. and Wen-Chieh, L. (2004). A vision-based vehicle identification system. Proceedings of the 2004 *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*. 23-26 Aug. 2004. 364-367 Vol.364.
2. De Silva, G. C. (2001). *Automation of Traffic Flow Measurement Using Video Images*. Master of Engineering, University of Moratuwa, Sri Lanka.
3. Dickon Ross, C. S. a. G. M. (2010). *Electronics For Dummies*. England: John Wiley & Sons.
4. Goyette, N., Jodoin, P., Porikli, F., Konrad, J. and Ishwar, P. (2012). Changedetection.net: A new change detection benchmark dataset. Proceedings of the 2012 *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*. 16-21 June 2012. 1-8.
5. Gupte, S., Masoud, O., Martin, R. F. K. and Papanikolopoulos, N. P. (2002). Detection and classification of vehicles. *Intelligent Transportation Systems, IEEE Transactions on*, 3(1), 37-47. doi: 10.1109/6979.994794
6. Hadi, R. A., Sulong, G., & George, L. E. . (2014). Vehicle Detection and Tracking Techniques: A Concise Review. *Signal & Image Processing: An International Journal (SIPIJ)*, 5(1), 1.
7. Han, B., Roberts, W., Wu, D. and Li, J. (2007). Motion-segmentation-based change detection. *SPIE Defence & Security Symposium 2007*, 65680Q-65680Q. doi: 10.1117/12.720308
8. Hsieh, J.-W., Shih-Hao, Y., Yung-Sheng, C. and Wen-Fong, H. (2006). Automatic traffic surveillance system for vehicle tracking and classification. *Intelligent Transportation Systems, IEEE Transactions on*, 7(2), 175-187. doi: 10.1109/tits.2006.874722
9. Huang, D.-Y., Chen, C.-H., Hu, W.-C. and Su, S.-S. (2012). Reliable moving vehicle detection based on the filtering of swinging tree leaves and raindrops. *J. Vis. Comun. Image Represent.*, 23(4), 648-664. doi: 10.1016/j.jvcir.2012.03.002
10. Huwer, S. and Niemann, H. (2000). Adaptive change detection for real-time surveillance applications. Proceedings of the 2000 *Visual Surveillance, 2000. Proceedings. Third IEEE International Workshop on*. 2000. 37-46.
11. Iwasaki, Y. and Itoyama, H. (2006). Real-time Vehicle Detection Using Information of Shadows Underneath Vehicles. In K. Elleithy, T. Sobh, A. Mahmood, M. Iskander & M. Karim (Eds.), *Advances in Computer, Information, and Systems Sciences, and Engineering* (pp. 94-98): Springer Netherlands.
12. Kanhere, N. K. (2008). Vision-based detection, tracking and classification of vehicles using stable features with automatic camera calibration (pp. 105).
13. Kanhere, N. K. and Birchfield, S. T. (2008). Real-Time Incremental Segmentation and Tracking of Vehicles at Low Camera Angles Using Stable Features. *Intelligent Transportation Systems, IEEE Transactions on*, 9(1), 148-160. doi: 10.1109/tits.2007.911357
14. Kastrinaki, V., Zervakis, M. and Kalaitzakis, K. (2003). A survey of video processing techniques for traffic applications. *Image and Vision Computing*, 21(4), 359-381. doi: http://dx.doi.org/10.1016/S0262-8856(03)00004-0
15. Koller, D., Weber, J., Huang, T., Malik, J., Ogasawara, G., Rao, B., et al. (1994). Towards robust automatic traffic scene analysis in real-time. Proceedings of the 1994 *Decision and Control, 1994., Proceedings of the 33rd IEEE Conference on*. 14-16 Dec 1994. 3776-3781 vol.3774.
16. Koller, D., Weber, J. and Malik, J. (1994). Robust multiple car tracking with occlusion reasoning. In J.-O. Eklundh (Ed.), *Computer Vision — ECCV '94* (Vol. 800, pp. 189-196): Springer Berlin Heidelberg.
17. Lim, K. H., Ang, L., Seng, K. P. and & Chin, S. W. (2009). Lane-Vehicle Detection and Tracking. *Proceedings of the International Multi-Conference of Engineers and Computer Scientists (IMECS 2009)*, 2, 5–10.

18. Lindeberg, T. (1996). Scale-space: A framework for handling image structures at multiple scales. *CERN School of Computing*, 695–702.
19. Mahmassani, H. S., Haas, C., Zhou, S. and Peterman, J. (1999). *Evaluation of Incident Detection Methodologies*: Center for Transportation Research, Bureau of Engineering Research, the University of Texas at Austin.
20. Mandellos, N. A., Keramitsoglou, I. and Kiranoudis, C. T. (2011). A background subtraction algorithm for detecting and tracking vehicles. *Expert Systems with Applications*, 38(3), 1619-1631. doi: http://dx.doi.org/10.1016/j.eswa.2010.07.083
21. Vasu, L. (2010). *An effective step to real-time implementation of accident detection system using image processing.* Master of Science, Oklahoma State University, USA. Retrieved from https://vpn.utm.my/docview/857659176?accountid=41678
22. Wei, Z., Wu, Q. M. J., Xiaokang, Y. and Xiangzhong, F. (2008). Multilevel Framework to Detect and Handle Vehicle Occlusion. *Intelligent Transportation Systems, IEEE Transactions on*, 9(1), 161-174. doi: 10.1109/tits.2008.915647
23. Weihua, W. (2009). Reach on Sobel Operator for Vehicle Recognition. Proceedings of the 2009 *Artificial Intelligence, 2009. JCAI '09. International Joint Conference on*. 25-26 April 2009. 448-451.
24. Yen-Lin, C., Bing-Fei, W. and Chung-Jui, F. (2009). Real-time vision-based multiple vehicle detection and tracking for nighttime traffic surveillance. Proceedings of the 2009 *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*. 11-14 Oct. 2009. 3352-3358.
25. Zehang, S., Bebis, G. and Miller, R. (2006). On-road vehicle detection: a review. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(5), 694-711. doi: 10.1109/tpami.2006.104
26. Zhang, W., Wu, Q. M. J. and Yin, H. b. (2010). Moving vehicles detection based on adaptive motion histogram. *Digit. Signal Process.*, 20(3), 793-805. doi: 10.1016/j.dsp.2009.10.006
27. Zheng, J., Wang, Y., Nihan, N. L. and and Hallenbeck, M. E. (2006). Extracting Roadway Background Image: a Mode-Based Approach. *Transportation Research Record: Journal of the Transportation Research Board*, 1944, 82-88.

4/26/2014