An Adaptive Approach for Real-Time Road Traffic Congestion Detection Using Adaptive Background Extraction

Nijad Al-Najdawi[1], Asma Abu-Roman[1], Sara Tedmori[2], and Mohammad Al-Najdawi[1]
[1]Department of Comuter Science, Al-Balqa Applied University, Jordan
[2]Department of Computer Science, Princess Sumaya University for Technology, Jordan

Abstract: *Traffic congestion is a situation on road networks that occurs as road use increases. When traffic demand increase, the interaction between vehicles slows the speed of the traffic stream and congestion occurs. As demand approaches the capacity of a road, extreme traffic congestion sets in. Current techniques for road-traffic monitoring rely on sensors which have limited capabilities, inflexibility, and are often costly and disruptive to install. The use of video cameras coupled with computer vision techniques offers an attractive alternative to the current sensors. Vision based sensors have the potential to measure a far greater variety of traffic parameters compared to conventional sensors. This work presents an approach for traffic congestion detection based an adaptive background extraction and edge detection techniques using rang filtering. The proposed work uses a special shadow detection algorithm that reduces the chances of misclassification and enhances the segmentation process. An adaptive background extraction technique is used for better object segmentation. In addition, this approach provides real-time statistical information for traffic surveillance on highways such as, the total number of vehicles on the road and the average speed of those vehicles. The proposed system is capable of detecting cars and vans simultaneously.*

Keywords: *Congestion detection, video surveillance, shadow detection, background updating.*

Received June 6, 2013; accepted October 26, 2014; published online June 11, 2015

1. Introduction

Traffic congestion is a situation on road networks that arises as use increases. It is distinguished by slower speeds, longer journey times and increased vehicular queuing. Extreme traffic congestion happens as demand exceeds the capacity of a road. When vehicles are stopped for periods of time, traffic jam occurs. There are number of situations which cause or magnify congestion; most of them reduce the capacity of a road at a given point or over a certain length or increase the number of vehicles required for a given volume of people or goods. According to a study of US vehicles, around half of traffic congestion is attributed to sheer weight of traffic; while most of the rest is attributed to traffic incidents, road work and weather events.

Traffic research still cannot fully predict under which conditions a “traffic jam” may suddenly occur. It has been found that individual incidents may cause wave effects which then spread out and create a sustained traffic jam [22].

Road traffic monitoring involves the collection of data describing the characteristics of vehicles and their movement through road networks. Vehicle count, vehicle speed, vehicle path flow rates, vehicle density, vehicle length, weight, class (car, van, bus) and vehicle identities via the number plate are all examples of useful data. Such data can be very useful for traffic surveillance applications. Since, the construction of paved roads, researchers have struggled with ways to record, understand and investigate vehicular movement. Not only this information is required for proper design of roadways, but also by the new Intelligent Transportation Systems (ITS) which require an understanding and investigation of the transportation networks operation in order to be effective. Moreover, to protect a country’s aging infrastructure reducing traffic congestion and accidents, detailed information about the operation of a given transportation environment is required.

Currently, traffic management and information systems rely on a set of sensors for estimating traffic parameters. Tubes, microwaves and loop detectors are used to monitor and count vehicles. However, all these sensors suffer from a number of drawbacks such as difficult maintenance, high installation costs and poor accuracy under different conditions. Traffic flow monitoring based on computer vision has been considered as an attractive alternative to extracting sufficient information about the status of road networks, such as occlusions and accidents. In addition to the regular statistical numbers such as the number of vehicles and their speed, it can also give information such as vehicle classification type and size. Video cameras are capable of providing a digitized live video that is fed into computers for further processing.

Computer vision algorithms have been applied to traffic scenes for a variety of purposes, including: queue detection, incident detection, vehicle classification and vehicle counting. Referring to vision, video camera is potentially more powerful than any other sensor currently available. The installation of video cameras for monitoring road networks is cheaper, less disruptive, and more flexible than installing other sensors. In most countries, a large number of surveillance cameras is already installed on road networks and used only for surveillance purposes, and where video streams are visually inspected for incidences and recorded with no further processing. Vision based systems have the potential of extracting a variety of information, such as precise vehicle path, vehicle shape, dimensions and other specifications. With proper positioning of the camera, a vision system is capable of tracking vehicles as they manoeuvre through complex road junctions or relatively long stretches of road.

This paper proposes an algorithm that detects traffic congestion in several real-time environments. In order to increase the effectiveness of the proposed work, a dynamic background subtraction, and shadow detection algorithms are used to enhance the segmentation process by reducing the possibility of miss-classifications and increase the robustness of the proposed work through increasing the resistance to environmental changes respectively. The experimental results obtained from the proposed algorithm demonstrate the robustness of the congestion detection in real time applications. For clarity of presentation, the paper is divided into seven sections. Apart from this section which provided an introduction to the application domain, section 2 provides a literature review for the state-of-the-art algorithms in this domain. Section 3 provides details of the proposed work and illustrates the shadow detection algorithm, the proposed segmentation method and the automatic background updating algorithm. Section 4 concludes this work with an insight to future work.

2. Related Works

Traffic flow monitoring based on computer vision has been considered as an attractive alternative to extracting sufficient information about the status of a road networks. Many researchers have investigated several algorithms for this purpose. Wang and Yang [23] proposed a system that extracts traffic parameters, such as: Vehicle number, vehicle velocity and vehicle type, in real-time using multi-camera systems. This system fuses images from different cameras for a same region and project captured images onto a same world coordinator. Deng and Li [11] proposed a method for extracting traffic parameters based on Epi-polar plane image. Their method detects a mean velocity, length of the vehicle and traffic flow.

Chellappa and Rajagopalan [7] developed a tracking algorithm based on high order statistical information. In their work, they derived statistical information about the vehicle class by using a training set of vehicle image patterns. Next, they used higher order statistical information that they derived from sample images to approximately model the unknown distribution of the image patterns of vehicles. Moving objects are segmented by applied thresholding and morphological operations on the stabilized frames. Statistical properties of the background scene are learnt from the given test video sequence to reduce false alarms. Al-Garni and Abdennour [1] proposed a background subtraction and edge detection technique for vehicle detection. This technique consists of two procedures: automatic background extraction, which extracts the background automatically from the successive frames, and vehicles detection procedure, which depends on edge detection and background subtraction. Xu *et al*. [24] used the vehicles and shadows statistical characteristics for detecting the vehicle in the image sequences. The background model was given and then the background subtraction method was used to obtain foreground formation.

Jun *et al*. [16] developed a method that is capable of separating occluded vehicles by tracking the movements of feature points. Foreground vehicles were separated from the background by using the adaptive mixture of Gaussian background model. The system then detects blobs that are suspected of having more than one vehicle in them by using the morphological characteristics of blobs: Solidity, eccentricity and orientation. Next, motion vectors associated with each feature point are obtained by tracking feature points included in the blob and then grouping those motion vectors to construct clusters of feature points by a hierarchical clustering algorithm. Finally, the suspected blobs were segmented into many small pieces by using watershed transform. One of the disadvantages of this technique is that the algorithm produces pretty small patches, which makes it harder to decide where the patch belongs, as the small patch does not contain much information. Han and Zhang [14] developed a method to detect vehicles using background frame differencing, and edge detection. In their approach, a noise variation image is obtained that identifies the moving objects in the scene by subtracting the background image from the Current Image (CI).

Jinglei and Zhengguang [15] used a background subtraction and region growing approach. For illumination disturbance they applied updated background algorithm which uses temporal differencing between three frames. The authors then calculated the ratio of the changing pixels according to the whole difference image. In their approach, if the ratio is lower than a threshold value, the original image is set as the background. The original background is

upgraded periodically in order to filter disturbance by the change of illumination. If the ratio is higher than the threshold, a vehicle motion is detected. In addition, the self-adaptive max-variance threshold algorithm was adopted to determine the threshold. After threshold separation, a fast region growing algorithm was employed to border vehicles in rectangles. Finally, the authors calculated the algorithms traffic parameters, including traffic flux, average speed and duty ratio of road.

Zhi-Fang and Zhi-Sheng [25] used an adaptive background updating method to determine the location of the vehicle. At the beginning, the authors used an adaptive background update and extraction. However, in order to deal with lighting and weather conditions, they updated the background by taking a weighted average of the Current Background (CB) and the current frame of the video sequence, and updated the threshold by using the histogram of the difference image. Next, to accurately locate moving vehicle region, they used hybrid location method that consists of two processes, a course and a refined search. Kanhere *et al.* [18] developed a technique for detecting and tracking vehicles, by matching detections in the new frame with the existing vehicles base fronts. In their approach, the background image is subtracted from the input frame and after suppressing shadows, thresholding and morphological operations are performed to compute foreground mask. Vehicle Base Fronts (VBF) are then detected in the image. The distance between an existing VBF and a new detection of VBF is computed.

Tsai *et al.* [21] developed a system to detect vehicles using edges and vehicle colors. They proposed a color transformation to project all colors of input pixels on a color space, so that vehicle pixels can be easily identified from backgrounds. A Bayesian classifier was used for this identification. Then, each detected vehicle pixel will correspond to a possible vehicle. Since, vehicles have different sizes and orientations, different vehicle hypotheses are generated from each detected vehicle. Chen *et al*. [8] in their approach separated moving objects from the captured image sequences using change detection and background updating. A static background was derived to be the reference frame and then frame-difference technique was used for changing detection. The changing detection was used to analyze temporal information between successive frames. In order to recognize different objects, they used some criteria as dispersants, aspect ratio and area ratio for a moving object. Another approach that uses cellular networks handover count is proposed by Demissie *et al.* [10] explored a complementary method to gauge the status of road traffic conditions. In addition to road monitoring, computer vision has been used to detect and recognise traffic signs and other objects on highways.

Bruno *et al.* [6] proposed a system that can detect and recognize traffic signs automatically using image analysis technique. Ruta *et al.* [19] developed a method for real-time traffic sign recognition with three phases that include: Detecting the area of interest and extracting the colour information, tracking the traffic sign to predict the position and scale over time in order to decrease processing time and finally classifying or recognizing the sign by comparing with ideal sign images that were learned off-line.

3. Proposed Work

Different approaches were proposed in literature, for traffic monitoring and road congestion detection. However, most of the proposed approaches suffer from a number of issues that might affect the robustness of those algorithms, such as: False detections, occlusion, miss-classifications and vulnerability to environmental changes that include scene lighting. In this work, an efficient shadow detection and removal algorithm is used in order in order to reduce the chance of false detections and miss-classifications. In addition, a dynamic background subtraction algorithm is included in order to increase the resistance against environmental changes and to enhance the effectiveness of the proposed work.

3.1. Shadow Detection

Misclassifying shadow areas as foreground leads to inaccurate segmentation and extraction of moving objects. The fact that shadows have similar motion to the objects casting them and are detected as an element of the object, has lead researchers to investigate techniques for effective shadow detection and removal [4]. Generally, shadows are classified as either self or cast shadows. Object tracking algorithms are mostly concerned with cast shadows, i.e., shadows cast by an illuminated object onto other objects. Although, many shadow detection and removal algorithms have been proposed in literature, most of the proposed methods that claim to be object and environment independent include some minor assumptions about the scene geometry or spectral distributions of the light sources. In this paper, the shadow detection method in [2] is adopted. This shadow detection method is based on a physically-derived hypothesis for shadow identification. The algorithm is proven to be fast, reliable and can be applied to real-time applications. It is shown that the algorithm effectively remove shadows umbra and penumbra under various lighting and environmental conditions. The technique can be described as follows:

- *Shadow Condition*: Let $q \in R^3$ is a point on the surface of an object in an illuminated three-dimensional scene and let n_q be a neighbourhood of q in the surface.

Using the simple geometric representation of light rays and a simple reflection model, it is possible to show that the light energy received at points $r \in n_q$ in the absence of an object casting a shadow over n_q is affinity related, to a high degree of approximation, to the energy received when a shadow is cast over n_q by an object. The same affine parameters being applicable to the whole neighbourhood n_q. It is of course clear that when a shadow is cast over a neighbourhood, less light is received there as compared to the fully illuminated state and that this condition should also be included in a shadow model. It follows that reflected energies behave similarly and hence: The luminance function $L: n_q \rightarrow R$ when no shadow is cast over n_q is affinely related to the luminance function $L^*: n_q \rightarrow R$ when a shadow is cast; i.e., for n_q to be in shadow we have $L^*(r) \approx \lambda L(r) + \mu$ and $L^*(r) < L(r)$, for some constants λ and μ, for all $r \in n_q$. These neighbourhood relationships are fundamental to the remainder of the shadow detection technique and constitute the basis of the shadow detection algorithms.

- *Determination of the Affine Parameter*: If J_p denotes the matrix $(J_p)_{i,\ j}=1,\ \forall 0 \leq i,\ j \leq k\text{-}1$ then the neighbourhood luminance relation $L^*=\lambda L+\mu$ translates directly to the relation $P^*=\lambda P+\mu J_p$ for pixel blocks P^* and P at identical positions in the object and background frames F^* and F respectively [3]. Appropriate affine parameters (λ, μ) may be computed, for the block-pair in a number of ways. For example, we could compute them from the relations $\bar{P}^* = \lambda \bar{P} + \mu$, $\sigma(P^*)=\lambda\sigma(p)$ from which we obtain: $\lambda=\sigma(P^*)/\sigma(P)$ and $\mu = \bar{P}* - \lambda \bar{P}$. If λ, μ are the affine parameters for a block pair (P, P^*), then the conditions for P^* to be a shadow block is:

$$\bar{P}^* < \bar{P} \ \ and \ \ \frac{\| P^* - (\lambda P + \mu J_p \|^2}{\| P \|_2} \approx 0 \qquad (1)$$

  I.e., the mean luminance of P^* is lower than that of P and the affine condition hold for the pair (P, P^*). A quantitative evaluation of the proposed shadow detection algorithm and some more applications of the proposed algorithm can be found on [2].

3.2. Object Segmentation

After applying the proposed shadow detection algorithm, the frames are ready for the next step that involves segmenting vehicles from the scene. In order to segment vehicles from the background, the segmentation method needs to accurately separate vehicles from the background. For real-time applications, the algorithm must be fast enough, insensitive to lighting and weather conditions and should require minimal amount of initialization [20]. To detect vehicles, the extracted background must be subtracted from the *CI* through the following steps:

- *Step* 1: Filtering the frames using the median filter in order to remove noise, and smoothing each of the *CB* frame and *CI* frame.
- *Step* 2: Sharpening of frame details by subtracting a blurred (un-sharp) version of the frame from itself. Segmentation with edge sharpening is better than thresholding.
- *Step* 3: Extracting relevant information for both *CB* and *CI* frames with texture analysis. By using range filter, illumination changes are less important in this algorithm and randomly choosing background appearance is avoided.
- *Step* 4: After determining the highlight edge in both *CB* and *CI*, the background subtraction technique is used for subtracting the edge of the background frame from the current frame in order to get the edge of moving object [9]. Background subtraction is a simple and effective technique for extracting the edge of the moving objects from a scene. The frame-difference $D_i(x,\ y)=C_i(x,\ y)\text{-}B_i(x,\ y)$ where $D_i(x, y)$ is the difference frame, $C_i(x, y)$ is the current frame, $B_i(x,\ y)$ is the background frame, I denotes the frame index.
- *Step* 5: Transforming the difference frame to binary image depending on the threshold. If the value of the pixels is lower than a present threshold, then the original image is set as the background. If the ratio is higher than the threshold, then the original image is set as the motion vehicle as in Equation 2

$$R_i(x, y) = \begin{Bmatrix} 0 & D_i(x, y) \leq Th \\ 1 & Otherwise \end{Bmatrix} \qquad (2)$$

  Where $R_i(x,\ y)$ is a binary image, Th is a threshold, and $D_i(x,\ y)$ is the difference image. "1" denotes foreground and "0" denotes background.

- *Step* 6: Morphological cleaning is performed on the binary image to remove remaining noise.
- *Step* 7: Morphological closing is performed on the binary image for connected component of edges using structure element line.
- *Step* 8: Filling of image regions is performed to provide a complete mask of the moving object in order to achieve better extraction later.
- *Step* 9: This step involves generating several holes in the binary mask of the moving object. In order to solve the problem of hollow phenomenon, open morphological operation is used. The algorithm produces a noisy image which retains most of the true moving pixels together with false signals due to noise and uninteresting moving objects, such as hedges and trees. These signals must be removed and the shape of moving objects of interest must be "extracted". Removing these signals has been often called in the image processing community, the False Positive Reduction (FPR) step. Here, it has been accomplished by using open morphological operations. Then, connected components are

labelled in binary image. The segmentation algorithm is illustrated Figure 1.

- *Step* 10: This step involves determining the vehicle type (car or van) based on the ratio between "the major axis length" and "the minor axis length" of all the labeled regions. When a vehicle is moving, these features are extracted, but this feature may be changeable at different extraction time.

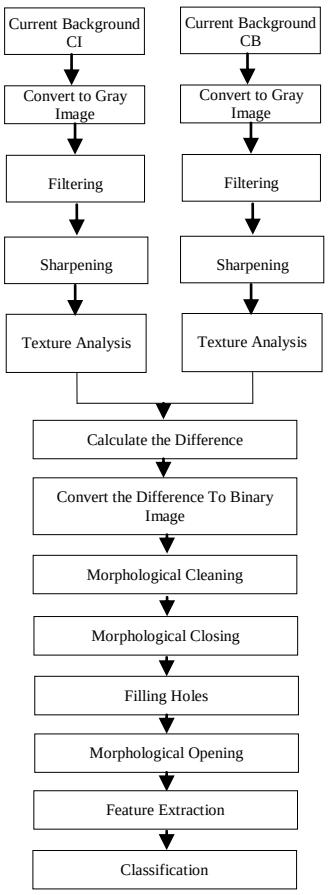


Figure 1. Flow chart of the segmentation algorithm.

To reduce the problem of changeable features extracted from a moving object, the aspect ratio is computed, as follows: *Aspect Ration=Height/Width*. The major axis length is equal to length (height) of the major axis of the ellipse that has the same normalized second central moments as the region.

The minor axis length is equal to length (width) of the minor axis of the ellipse that has the same normalized second central moments as the region. As shown in Figure 2, the left side of the figure shows an image region and its corresponding ellipse. The right side shows the same ellipse, with features indicated graphically and the solid blue lines are the axes. The result value (aspect ratio) is then compared with threshold values. If the ratio is between Th_a and Th_b then the labeled image is set as the vehicle, by drawing a red line around this region. If the ratio is lower than Th_b, it is considered that as background noise is: $Th_a \geq AR_i \geq Th_b$, where $AR(i)$ is aspect ratio, Th_a and Th_b upper and lower threshold and i denotes the frame index. However, two or more vehicles might appear as a single blob in the foreground mask as a result of partial occlusions. In such situations $AR(i)$ is greater than Th_a, detecting the front side of object's base, which it refer to a VBF, helps to detect occlusion.

Figure 2. Image region and its corresponding ellipse.

First, the front side of a vehicle's base is computed, then the connected components labelling algorithm is used to get number of front base in the image. If count of front is greater than 2, occlusion exists and the green line is drawn around this region to indicate that exists occlusion. Base front of a vehicle is easily found using a difference operator in the vertical direction [17]. The foreground pixels are labelled with a positive value, while the background pixels are labelled with the value of zero as in Equation 3:

$$B_i(x,y) = \begin{cases} 1 & F_i(x,y) - F_i(x,y+1) \geq 0 \\ 0 & Otherwise \end{cases} \quad (3)$$

Where $B_i(x, y)$ is the result image with front base, F_i is the *CI* and *i* denote the frame index.

3.3. Automatic Background Extraction

Traditional background updating methods create Instantaneous Background (IB) and take a weighted average of the *CB* and the current frame of the video sequence [13]. Those traditional methods require manual intervention and can be summarized in Figure 3. However, in video sequences of highway traffic, it might be impossible to acquire a background image. Therefore, the choice of an automatic extract of the background image would be more feasible. Automatic background extraction gradually builds up the required background image over time. In order to obtain the required background image, the first few successive frames are used for this process. In this work, the automatic background extraction starts by processing the first three successive frames as follows (the flowchart of the proposed work is illustrated in Figure 4.

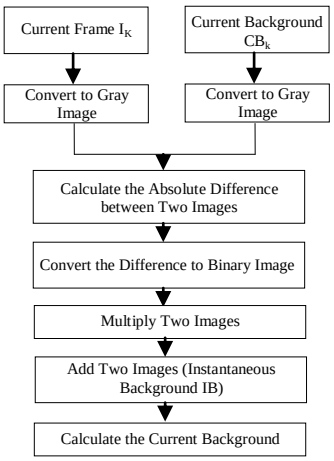


Figure 3. Traditional background updating method.

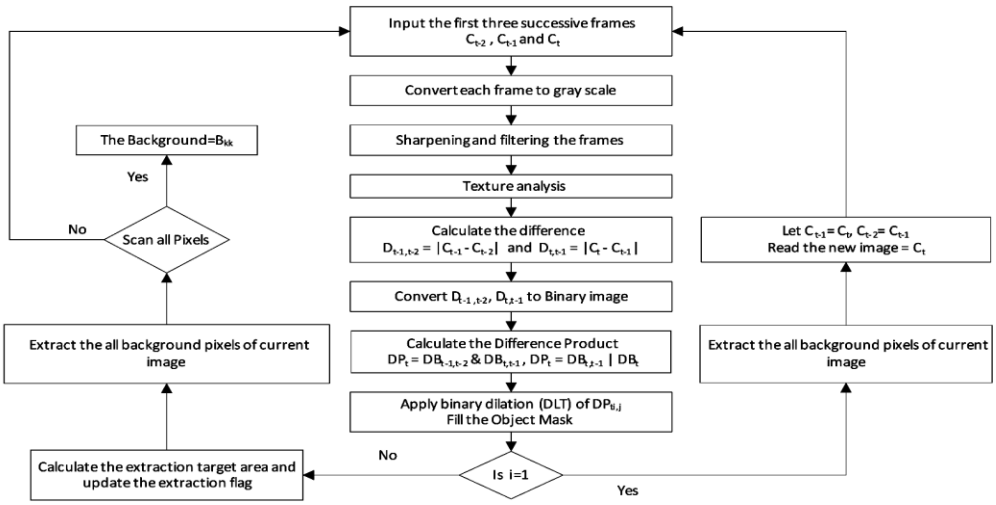
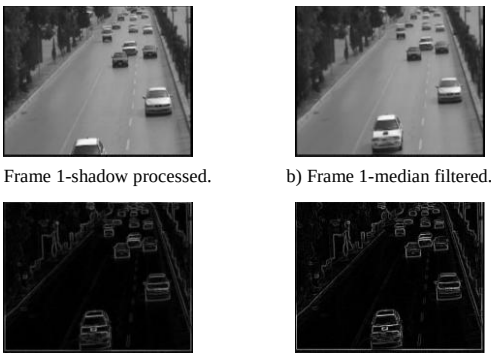


Figure 4. The flowchart of the proposed automatic background extraction algorithm.

- *Step* 1: In live video stream, edges and surface contours are found in frames using range filter. Range filter is used to avoid randomly choosing background appearance [5]. In this step, frames are converted to gray scale and then each frame is filtered using median filter to remove the noise and to smooth the image. Frames are then sharpened using the un-sharp filter to strengthen the edge of detection region. Results of this step can be seen in Figure 5.



a) Frame 1-shadow processed. b) Frame 1-median filtered.

c) Frame 1-edge detection. d) Frame 1-sharpened.

Figure 5. Results of step-1 of the proposed automatic background subtraction method.

- *Step* 2: The first three successive frames C_{t-2}, C_{t-1}, C_t are used to calculate the frame differences $D_{t-1,\,t-2}$ and $D_{t,\,t-1}$ as in Equations 4 and 5 respectively (results of this step can be seen in Figures 6-a and b:

$$D_{t-1,t-2} = \left|C_{t-1} - C_{t-2}\right| \tag{4}$$

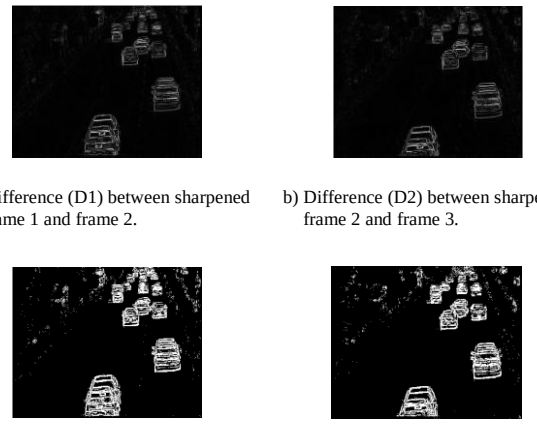
$$D_{t,t-1} = \left|C_t - C_{t-1}\right| \tag{5}$$

- *Step* 3: The frame difference results are transformed to binary frames based on the following Equations 6 and 7:

$$DB_{t-1,t-2}(x,y) = \begin{Bmatrix} 1 & D_{t-1,t-2}(x,y) \geq Th \\ 0 & Otherwise \end{Bmatrix} \tag{6}$$

$$DB_{t,t-1}(x,y) = \begin{Bmatrix} 1 & D_{t,t-1}(x,y) \geq Th \\ 0 & Otherwise \end{Bmatrix} \tag{7}$$

Where *DB* is a binary difference frame, *Th* is a threshold value that represents the mean gray-scale value in each frame. The result of thresholding the binary image reveals that foreground pixels that represents the objects would have the value of 1 and the background pixels would have the value of 0. Results of this step can be seen in Figures 6-c and d.



a) Difference (D1) between sharpened frame 1 and frame 2. b) Difference (D2) between sharpened frame 2 and frame 3.

c) The binary images of a. d) The binary images of b.

Figure 6. The automatic background subtraction algorithm.

- *Step* 4: In this stage, in order to facilitate extracting the moving objects, the Difference Product (DP) is calculated using the bitwise logical AND operation for the obtained binary difference images $DB_{t-1,\,t-2}$ and $DB_{t,\,t-1}$ and the bitwise logical OR operator is applied between P_t, $DB_{t,\,t-1}(x, y)$ in order to provide a moving object, where: $DP_t = DB_t | DB_{t,\,t-1}$ and $DP_t = DB_{t-1,\,t-2}$ and $DB_{t,\,t-1}$.
- *Step* 5: To calculate Moving Object Region (MOR), binary dilation and morphological closing and fill operators are applied. Dilation operation is applied on *DB* frame in order to fill discontinuous regions. Morphological closing operator is applied after the dilation to get the MOR. Finally, the fill operation is applied. Dilation and closing operators are used to growing the object area and removing unwanted fragments in the result respectively. Fill operation is used in order to provide a complete mask of moving

object for achieving better extraction in the next step [12].

- *Step* 6: The initial background B_i is estimated using MOR_i and C_t and the result is stored at the Extraction Flag (EF) where the initial value of $EF_i=MOR_i$ is in Equation 8:

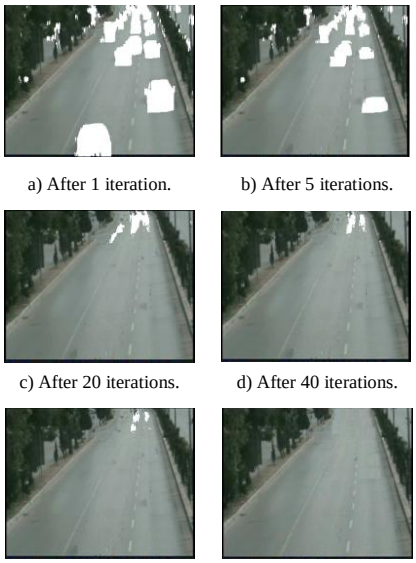
$$B_i(x,y)=\begin{cases}255 & MOR_i(x,y)=1\\ Ct(x,y) & Otherwise\end{cases} \quad (8)$$

Where B_i is background image, C_t is first frame and MOR_i is the *MOR* in the first three frames.

- *Step* 7: The Extracted Target Area (ETA) is calculated using the *EF* and the *MOR*, where $ETA_i=EF_{i-1}$ and MOR_i. The *EF* is updated such that: $EF_i = EF_{i-1} \oplus ETA_i$ where $\oplus$ is the bitwise logical XOR operator, EF_{i-1} is region information of stage i-1.
- *Step* 8: The background pixel of current frame is extracted using the background pixel of stage *i*-1, the *CI*, and the extraction target area of stage *i* is in Equation 9:

$$B_i(x,y)=\begin{cases}Ct(x,y) & ETA_i=1\\ B_{i-1}(x,y) & Otherwise\end{cases} \quad (9)$$

This procedure is repeated until all background pixels are extracted. Figure 7 illustrates the proposed algorithm results after 1, 5, 20, 40, 50 and 60 iterations.



a) After 1 iteration. b) After 5 iterations.

c) After 20 iterations. d) After 40 iterations.

e) After 50 iterations. f) After 60 iterations.

Figure 7. This figure illustrates the output of the proposed automatic background extraction algorithm, results demonstrate the run of 1, 5, 20, 40, 50 and 60 iterations.

3.4. Experimental Results and Traffic Parameters Extraction

In the previous steps, moving vehicles have been successfully extracted from the video scene. Each object centroid is calculated in order to start the tracking process. For testing purposes a live video camera with a frame rate of 10 fps is placed over a pedestrian bridge facing a motorway. The camera feeds frame directly into the systems and is calibrated so that the detection zone that has 100 pixels height (Z_h=100) represents 10 meters in reality.

The vehicle' centroids are used to calculate their location displacements between successive frames based on the following geometric parameters: The object displacement in pixels is measured using the Euclidean distance measure between the vehicle previous and current centroids as shown in Equation 10:

$$d_p=\sqrt{(x_2-x_1)^2+(y_2-y_1)^2} \quad (10)$$

Therefore, the car displacement in meters can be calculated as follows: $d_m=d_p\times 10/Z_h$. The vehicle speed and the average of the vehicle velocities $Tv=v/n$ is calculated, where v is the sum of vehicle speeds and n is the number of detected vehicles on road. As stated earlier the camera is calibrated so that the detection zone has 100 pixels height that represents 10 meters in reality. The congestion ratio is calculated based on the ratio between the allowed road speed and the calculated vehicle speed. An example can be illustrated as shown in Figure 8, where the system monitors three cars.

The speed limit on this road is 60km. the obtained average velocity is equal to 65.8 Km/h and the ratio of congestion is equal to 0.0911% that is calculated as follows: Vehicle 1 centroid in frame number 1 is located at (213, 168). In the next frame, this vehicle is moved to pixel location (209,190). Regarding vehicle number 2 the pixel location for the 1st and 2nd centroids are (308, 162) and (309, 172) respectively. Therefore for the first vehicle, the Distance in pixels=22.3607, which represent a displacement of 2.23 meters. The frame rate is 10 fps, therefore the time between each consecutive two frames is 0.1 seconds, and therefore the first vehicle speed is calculated as 72.4 Km/h. As for the second vehicle, the calculations give 59.2 Km/h. Therefore, the average vehicles velocity is 65.8 Km/h. Figures 8, 9 and 10 represents screenshots of the proposed algorithm's interface. This interface is directly connected to the live video camera, the blue lines in the scene indicates the detection zone, red contour indicates that the calculations is over the, green contour indicates that processing this object has not been completed yet. Video frames appear clear from shadow, the contour around objects indicates successful object segmentation.

Figure 8. Traffic monitoring system detection of two objects.

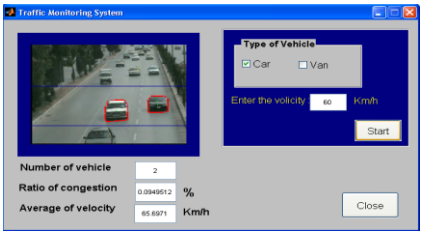


Figure 9. Traffic monitoring system detection of two objects.

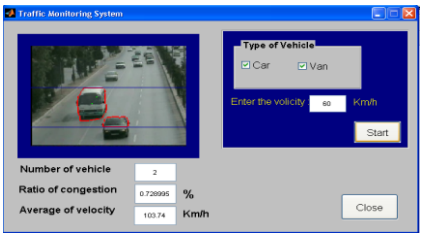


Figure 10. Traffic monitoring system detects two objects (a car and a bus).

4. Conclusions

Congestion often reduces regional economic health; it causes delays, which may result in late arrival for education, meetings and employment, ultimately resulting in lost business, disciplinary action or other personal losses. Inability to forecast travel time accurately, leads to more time spent on travel, and less time on productive activities. Road congestion has a great impact on wasting fuel which increases air pollution and carbon dioxide emissions. This work aims to develop a system of traffic congestion detection for an early and fast authority action. The proposed system aims to reduce the congestion problem by providing useful analysed information for traffic surveillance officers, by means of automatic detection for traffic congestion based on vehicle numbers and velocities on roads. A shadow detection algorithm in addition to an adaptive background extraction algorithm is proposed to assist the system performance. In this work, moving objects are segmented from the captured frame sequence using change detection and background updating. The change detection is used to analyze temporal information between successive frames. The combination of frame difference mask and background subtraction mask is used to acquire the initial object mask. Moreover, the edge detection by rang filtering is introduced to reduce the shadow influence and residual background problem. In addition, traffic parameters are calculated such as: the average speed of vehicles, the congestion ratio of road. Experimental results using real-time videos indicate the proper detection of congestions on roads.

References

[1] Al-Garni S. and Abdennour A., "Moving Vehicles Detection using Automatic Background Extraction," *in Proceedings of World Academy of Science, Engineering and Technology*, pp. 180-184, 2006.

[2] Al-Najdawi N., Bez H., Singhai J., and Edirisinghe E., "A Survey of Cast Shadow Detection Algorithms," *Pattern Recognition Letters*, vol. 33, no. 1, pp. 752-764, 2012.

[3] Al-Najdawi N., Bez H., and Edirisinghe E., "A Novel Approach for Cast Shadow Modelling and Detection," *in Proceedings of IET International Conference on VIE 2006*, Bangalore, India, pp. 553-558, 2006.

[4] Al-Najdawi N., Tedmori S., Bez H., and Edirisinghe E., "An Automated Real-Time People Tracking System Based on KLT Features Detection," *the International Arab Journal of Information Technology*, vol. 9, no. 1, pp. 100-107, 2012.

[5] Bailey D. and Hodgson R., "Range Filters: Local Intensity Subrange Filters and Their Properties," *Image and Vision Computing*, vol. 3, no. 3, pp. 99-100, 1985.

[6] Bruno L., Parla G., and Celauro C., "Improved Traffic Signal Detection and Classification via Image Processing Algorithms," *Procedia-Social and Behavioral Sciences*, vol. 53, pp. 811-821, 2012.

[7] Chellappa R. and Rajagopala A., "Vehicle Detection and Tracking in Video," *in Proceedings of the International Conference on Image Processing*, pp. 351-354, 2000.

[8] Chen T., Lin Y., and Chen T., "Intelligent Vehicle Counting Method Based on Blob Analysis in Traffic Surveillance," *in proceedings of the 2nd International Conference on Innovative Computing, Information and Control*, Kumamoto, pp. 174-179, 2007.

[9] Cui B., Sun S., and Duan Y., "Vehicle Detection Based on Adaptive Background," *in proceedings of the 2nd International Workshop on Knowledge Discovery and Data Mining*, Moscow, pp. 821-824, 2009.

[10] Demissie M., Correia1 G., and Bento C., "Intelligent Road Traffic Status Detection System through Cellular Networks Handover Information: An Exploratory Study," *Elsevier Transportation Research Part C: Emerging Technologies*, vol. 32, pp. 76-88, 2013.

[11] Deng T. and Li B., "A Detection Method of Traffic Parameters Based on EPI," *Procedia Engineering*, vol. 29, pp. 3054-3059, 2012.

[12] Goonzalez C., Woods E., and Eddins L., *Digital Image Processing Using MATLAB. Upper Saddle River*, Prentice Hall, 2004.

[13] Gupte S., Masoud O., Martin R., and Papanikolopoulos N., "Detection and Classification of Vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, no. 1, pp. 37-47, 2002.

[14] Han C. and Zhang Q., "Real-Time Detection of Vehicles for Advanced Traffic Signal Control,"

in Proceedings of International Conference on Computer and Electrical Engineering, Phuket, pp. 245-249, 2008.

[15] Jinglei Z. and Zhengguang L., "A Vision-Based Road Surveillance System Using Improved Background Subtraction and Region Growing Approach," *in Proceedings of the 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, Qingdao, pp. 819-822, 2007.

[16] Jun G., Aggarwal K., and Gokmen M., "Tracking and Segmentation of Highway Vehicles in Cluttered and Crowded Scenes," *in Proceedings of IEEE Workshop on Applications of Computer Vision*, Copper Mountain, pp. 1-6, 2008.

[17] Kamijo S., Ikeuchi K., and Sakauch M., "Vehicle Tracking in Low-Angle and Front View Images based on Spatio-Temporal Markov Random fields," *in Proceedings of the 8th World Congress on Intelligent Transportation Systems*, pp. 1-12, 2001.

[18] Kanhere N., Birchfield S., Sarasua W., and Whitney T., "Real-Time Detection and Tracking of Vehicle Base Fronts for Measuring Traffic Counts And Speeds on Highways," *Journal of the Transportation Research Board*, vol. 1993, no. 1, pp. 155-164, 2007.

[19] Ruta A., Li Y., and Liu X., "Real-Time Traffic Sign Recognition from Video by Class-Specific Discriminative Features," *Pattern Recognition*, vol. 43, no. 1, pp. 416-430, 2010.

[20] Shi J. and Malik J., "Normalized Cuts and Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888-905, 2000.

[21] Tsai L., Hsieh J., and Fan K., "Vehicle Detection using Normalized Color and Edge Map," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 850-864, 2007.

[22] Tsekerisa T. and Geroliminisb N., "City Size, Network Structure and Traffic Congestion," *Journal of Urban Economics*, vol. 76, pp. 1-14, 2013.

[23] Wang J. and Yang D., "A Traffic Parameters Extraction Method using Time-Spatial Image Based on Multicameras," *the International Journal of Distributed Sensor Networks*, vol. 2013, pp. 1-17, 2013.

[24] Xu S., Zhao Y., Yu C., and Shen L., "Vehicle Detection Algorithm Based on Shadow Feature," *in Proceedings of International Colloquium on Computing, Communication, Control, and Management*, Guangzhou, pp. 105-109, 2008.

[25] Zhi-fang L. and Zhisheng Y., "A Real-Time Vision-based Vehicle Tracking and Traffic Surveillance," *in Proceedings of the 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, Qingdao, pp. 174-179, 2007.

Nijad Al-Najdawi received his BSc degree in Computer Science from Mu'tah University, Jordan in 1999. He obtained his MSc degree in Multimedia and Internet Computing in 2003 and a PhD degree in Machine Vision and Autonomous Systems in 2006, from Loughborough University, UK. After which he joined Loughborough University as a research Associate (Post-Doc position) in the Electronic and Electrical Engineering department. Currently, he is appointed as an Associate Professor at Prince Abdullah Bin Ghazi Faculty of Information Technology and is acting as the Head of Computer Science Department at Al-Balqa Applied University, Jordan. His research interests include: Image processing, video coding, objects tracking and recognition.

Asma Abu Roman in 2010, she received her Master degree in computer science from A-Balqa Applied University- department of computer science, her dissertation involved traffic monitoring research, and currently her research interests involve: Image processing and video coding.

Sara Tedmori in 2001, she received her BSc degree in Computer Science from the American University of Beirut, Lebanon. In 2003, she obtained her MSc degree in Multimedia and Internet Computing from Loughborough University. In 2008, she received her Engineering Doctorate in Computer Science from Loughborough University, UK. Currently she is appointed as an assistant Professor in the Computer Science Department at Princess Sumaya University of Technology, Jordan. Her research interests include: Object tracking, image processing, expertise locator, knowledge extraction, knowledge sharing and privacy.

Mohammad Al-Najdawi SEO expert received his BSc degree in computer science from A-Balqa Applied University in 2006, he obtained his master degree in computer science from the same university in 2012, where he was appointed as a part-time instructor, his research interests involve: Image processing, watermarking and security systems.