Vehicle Detection and Classification System for Traffic Video Surveillance

Traiwit Intachak  
Department of Computer Engineering  
Prince of Songkla University  
Songkhla, Thailand  
5310120055@email.psu.ac.th

Wacharin Kaewapichai  
Department of Computer Engineering  
Prince of Songkla University  
Songkhla, Thailand  
watcharin@coe.psu.ac.th

Abstract—In this paper, we introduce a real-time vehicle classification algorithm to solve a classical problem “computation cost and time consumption” for traffic video monitoring during on the day in application of Intelligent Transportation System (ITS). Our algorithm consists of three parts: adaptive background subtraction with sudden illumination change handling, object detection by improved active contour detection (Snake), and vehicle classification by shape matching using shape context. This system achieved a high performance with 100% minivan and 89.21% average classification. In experimental results show efficiency and effectiveness of our algorithm which worked in traffic video monitoring in real-time condition (5 cars per second).

Keyword: image processing; traffic video monitoring; object detection; adaptive background subtraction; active contour detection; vehicle classification;

I. INTRODUCTION

An automatic vehicle classification system is very useful system for highway monitoring to verify types of vehicles and automatic tolls collection. Normally, a lot of workers are applied to intelligent transportation system (ITS) in manual toll correction and manual road charging sections. In this research, we focus on image processing algorithms (object detection and vehicle classification) working in case of sudden illumination change and real-time condition to solve a classical image processing problem: “computation cost and time consumption”.

II. LITERATURE REVIEW

![Sequence System Review](image)

Normally, image processing process can be divided to three main processes: Preprocessing, Feature Extraction, and Classification.

“Preprocessing” is a method to separate background and foreground/object in current image, which is called background subtraction. From previous researches, it can be divided to two main groups: Static and non-Static background subtraction. Static background subtraction, this group is used in controlled and low variant environment, which is applied to many research fields. Gaussian Mixture [1-4], K-Means [2], and Kalman filter [4], [5] algorithms are applied on gray-scale histogram of training image set to recognize and create a static background image. The algorithms in this group work in real-time condition with satisfy speed, however they are not deal with the problems of outdoor environment especially sudden illumination change. Second, non-Static background subtraction, Reference [6], [7] use dynamic threshold parameters calculated from light approximation to classify foreground/object pixels. Reference [8] uses graph of summation pixels to detect object. Reference [9] create the background image from three back frames (two previous and current frames). Reference [10], [11] introduce an algorithm to detect object and shadow removing, which is a technique “intersection of object positions from two current images”. The object position in two current images is extracted by normal background subtraction. However, the problem “sudden illumination change” is not solved by those techniques. A cause of sudden illumination change (outdoor environment) is cloud movement affecting to very fast illumination change. It is a main problem of background subtraction process. In this research, we introduced a method to solve very fast (sudden) illumination change problem explained in III.A “Background Subtraction”.

“Feature Extraction” used to extract object(s)’s shape in current image are edge detection algorithms. They can be classified to two main groups: Formable model and Deformable model. A method to classify feature extraction methods to Formable and Deformable model is what shape of object(s) [12] can or cannot be described with simple mathematical shapes such as triangular, circle, ellipse, and others, respectively. First, Formable model, Hough transform [13] is a popular method used to detect simple shape in current image. It is not work in traffic monitoring system because car shape cannot be described with the basic shapes. Second, Deformable model, Reference [14] implements background subtraction algorithm and integrates corner detection algorithm improved by [15] to extract
possible deformable object in current image. Reference [16], [17] applies “Active contour detection algorithm” called Snake to extract object from current image. They improve performance of Snake by adjust external energy term of the algorithm to handle moving of Snake control points. Reference [18] applies Quadtree algorithm to the external energy term of Snake dealing with moving of Snake control points. In this group, deformable model (Snake) is suitable to detect car in an image because of its shape. However a main problem of Snake is “computational costs”. It is improved in this paper show in III.B “Edge Detection”.

“Classification”, the shape description of objects (cars) from Feature Extraction process is used to separate car types. Basic information such as aspect ratio, shape, size, and radius is popular used as input of classification algorithms to classify. Thresholding of car size and aspect ratio is introduced to classify the car types by [17], [19]. Aspect ratio and size of object are used as information for classification by feeding to Back-propagation Neural Network Model (BPNNM) [20]. From previous works, car aspect ratio and object size are popular used as information for classification which is robustness [21]. However, a lot of data set requirement in training process is a problem. Shape radius, distance between centroid to shape contour in 0-359 degree, is another good feature which is used by shape matching classification. Reference [22] creates a model of shape radius for training and testing set comparing. A minimum difference rate of training and testing data is applied to classify car types in a current image. This method is very high efficiency and effectiveness working in intelligent transportation system (ITS), explained in III.C “Graph Matching”.

III. SYSTEM DESIGN

This research consists of three main steps: Preprocessing “Background Subtraction”, Feature Extraction “Edge Detection”, and Classification “Graph Matching”.

A. Background Subtraction

The first process, we manually initial region of interest (ROI) area in image to reduce computation costs. A Learning process uses the 20 first images (T = 20) in a video scene to create background image model (B). It is based on an assumption “there is no foreground/object in ROI of this image set”, defined mathematically by

\[ B = \frac{\sum_{t=1}^{T} I_t}{T} \]

where \( I_t \) is a color image scene [Red Green Blue]

Estimate process computes illuminance difference \( (L_t) \) between current image \( (I_t) \) and the background image \( (B) \) in ROI, defined mathematically by

\[ L_t = \sum [G_{I_t} - G_B] / R \]

where \( G \) is gray scale image;

\[ G = 0.299(\text{Red}) + 0.587(\text{Green}) + 0.114(\text{Blue}) \]

\( R \) is number of pixels in ROI area.
Normally, during a day and no object in ROI, illuminance changing can be separated into two events: slow changing and sudden changing [23]. Noticeably, from our experimental result, \( L_t \) is can be used to refer illuminance changing behaviors: \( L_t < 2 \) slow changing and \( L_t > 2 \) sudden changing. Thus, two weight parameters \((\alpha, \rho)\) for background updating to response illuminance changing are introduced. These parameters are performed to deal with the slow and sudden illuminance changing problems, show in

\[
\alpha_t = \begin{cases} 
1.90; & \text{if } L_t > 2 \text{ and no object in ROI} \\
1.70; & \text{others}
\end{cases}
\]

where \( \alpha \) is a weight parameter of background image.

\( \rho \) is a weight parameter of current image; \( \rho_t = 2 - \alpha_t \).

Pixels classified as background by Subtraction process will be used to update background image \((B)\) by (5) with the weight from equation (3).

Subtraction process computes the foreground/object pixels in a current image \((l_t)\) which is subtracted from the background image \((B)\). The operation to create a binary image \((Bi)\) is defined mathematically by

\[
Bi_{x,y} = \begin{cases} 
1; & f_{x,y} > THS \text{ (foreground pixel)} \\
0; & \text{other (background pixel)}
\end{cases}
\]

where \( x, y \) are position of pixels in ROI.

\[
f_{x,y} = \sqrt{S_{x,y}^2 + S_{x,y}^2} - B_{x,y}
\]

\[
S_{x,y} = l_{x,y} - B_{x,y}
\]

A threshold parameter \((THS)\) is set to 25 for object pixels and background pixels separation in ROI.

The current binary image \((Bi_t)\) from (4) is feedbacked to update the background model. Binary pixels \((Bi_{x,y} = 0)\) which are justified as background pixels in the current binary image \((Bi_t)\) are used as information for the background model \((B_t)\) updating in this step. The updated background \((B)\) image are used as background image for the next iteration \((t + 1)\), mathematically defined by

\[
B_{t+1} = \left\{ \rho(B_t) + \alpha_t(B_t) \right\} / 2
\]

A sub-process of Subtraction process is shadow removing. Normally, a basic background subtraction process is not provide a satisfy result. A main problem of background subtraction is what a threshold for object and background pixels separation is fixed (in this research \(THS = 25\)). Thus, we apply a basic idea to solve this problem. A color of shadow pixels travels on a diagonal line in Red-Green-Blue color plans shown in fig. 7.

![Figure 7. Shadow in RGB color plan.](image)

The pixels which are classified as object from (4) will be considered as shadow or real object pixels in this step. All object pixels \((Bi_{x,y} = 1)\) is classified as shadow pixels, if distance \((D)\) of color pixels \(x,y\) to the diagonal line is less than 12.5, shown in fig. 7. So, an updated binary image called object image \((O)\) is defined by

\[
O_{x,y} = \begin{cases} 
0; & D_{x,y} < 12.5 \\
Bi_{x,y}; & \text{others}
\end{cases}
\]

An image morphological “Hole Filling” algorithm is applied to object in Morphological process [24]. The output of Background Subtraction process is filled image \((H_t)\), and passed to next process.

![Figure 7. (a) Current Image (b) Result of Background Subtraction process (c) Shadow removing (d) Morphological operation (holes filling).](image)
B. Edge Detection

In this process, edge of object is extracted and smoothed. The extracted object edge is passed to Graph Matching for car classification.

Corner process is a corner detection algorithm (Haris’s operator), which is applied to the current image \( I_c \) and provides possibility corner points \( C_c \) in the current image. Corner points in ROI \( C_r \) are selected by (7). They passed to Initial process.

\[
C_r = C_c \cap H_t
\]  
(7)

The corner points on perimeter of object are selected as initial control points for Snake algorithm working in Snake process.

Classically, Snake is essentially a minimization energy function. The energy function without constraint can be represented by \( E = E_{\text{Internal}} + E_{\text{External}} \). Internal energy \( E_{\text{Internal}} \) is used to represent contour elasticity and smoothness of control points and external energy \( E_{\text{External}} \) is used to perform texture of object. In this research, we solve a classic problem of Snake which is computation time for minimization energy function. Basically, a good option of external energy term for Snake is gradient vector flow (GVF)\(^{8}\) [15], so we reduce time of the minimize function by GVF adjusting \( (GVF_{adj}) \) defined by (9)

\[
GVF_{adj}(x,y)_t = \begin{cases} 
0 & \text{if} -1 < GVF(x,y)_t < 1 \\
GVF(x,y)_t & \text{otherwise}
\end{cases}
\]  
(9)

The output result of Edge Detection process is a list of control points called edge of object \( (E_t) \) in ROI. It will be passed to next process for car classification.

C. Graph Matching

To create Car Models(CM): Motorcycle, MPV, Pickup, Saloon, and Van, a few sample cars of each set is selected as training set for model creation. The edge of object from last process is used to create Car graph \( (M) \), mathematically defined in (10). Average radius of the object edge on its angle \( (R_\theta) \) from the training set is extracted to create the car graph model (10) in training process shown in fig. 14.

\[
M = R_\theta
\]  
(10)

where \( R \) is average radius of the object edge on its angle; \( \theta = 0^\circ - 359^\circ \)

![Figure 8. Edge Detection sequence.](image)

![Figure 9. (a) An input image (b) Corner points in the image (c) Selected corner points in ROI (d) Initial control points of Snake](image)

![Figure 10. (a) Gradient vector image in x-axis, (b) Gradient vector image in y-axis.](image)

![Figure 11. (a) Edge points in Initial method in first iteration (b) Edge points from active contour in Deform method in 25 iterations.](image)

![Figure 12. Graph Matching sequence](image)

![Figure 13. Compute object in an image to graph.](image)
In Matching process, the testing car graph extracted from the current image \( (M) \) is classified to any types of five possible car models \( (CM_y) \), thus \( y \) means car type including Motorcycle, MPV, Pickup, Saloon, and Van.

\[
\text{Car Type} = \min_y \left( \sum_{\theta=0}^{359} |CM_y - M_{\theta}| \right)
\]

IV. Experimental Result

Four video scenes are recorded by a webcam camera for testing. The video scene size is 640 pixels width, 360 pixels height, and fixed angle 45° to the object.

The number of testing cars is 575 extracted from all four video scenes. Table II is a confusion matrix showing performance of our car classification system.

<table>
<thead>
<tr>
<th></th>
<th>Moto</th>
<th>MPV</th>
<th>Pickup</th>
<th>Saloon</th>
<th>Van</th>
<th>False</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moto</td>
<td>75</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>MPV</td>
<td>0</td>
<td>85</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Pickup</td>
<td>0</td>
<td>13</td>
<td>130</td>
<td>6</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>Saloon</td>
<td>0</td>
<td>11</td>
<td>202</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>Van</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>21</td>
<td>0</td>
</tr>
</tbody>
</table>

Table II. Percentage of vehicle classified in four video scenes

<table>
<thead>
<tr>
<th></th>
<th>Moto</th>
<th>MPV</th>
<th>Pickup</th>
<th>Saloon</th>
<th>Van</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real</td>
<td>88</td>
<td>93</td>
<td>155</td>
<td>218</td>
<td>21</td>
</tr>
<tr>
<td>Predict</td>
<td>75</td>
<td>111</td>
<td>133</td>
<td>214</td>
<td>24</td>
</tr>
<tr>
<td>Success</td>
<td>91.46%</td>
<td>94.39%</td>
<td>87.24%</td>
<td>93.41%</td>
<td>100%</td>
</tr>
</tbody>
</table>
In this research, we introduced a real-time vehicle detected and vehicle classification with many concepts and algorithms to solve problem of outdoor traffic monitoring on the day. From our experimental results, the best performance of classification is Van (100%) class and the worst of classification is Pickup (87.24%). Accessories on the testing car, especially on the roof, affect to performance worst of our system. However, the average system efficiency is 89.21% and effectiveness is 5 cars per second (working on Matlab 2012). The output of our system (vehicle class/types) is usefully results for future works such as traffic monitoring and others.

V. CONCLUSION

In this research, we introduced a real-time vehicle detected and vehicle classification with many concepts and algorithms to solve problem of outdoor traffic monitoring on the day. From our experimental results, the best performance of classification is Van (100%) class and the worst of classification is Pickup (87.24%). Accessories on the testing car, especially on the roof, affect to performance worst of our system. However, the average system efficiency is 89.21% and effectiveness is 5 cars per second (working on Matlab 2012). The output of our system (vehicle class/types) is usefully results for future works such as traffic monitoring and others.

REFERENCE
