# Moving Object Detection and Classification Using Neuro-Fuzzy Approach

M. A. Rashidan[1*], Y. M. Mustafah[1], A. A. Shafie[1], N. A. Zainuddin[1],
N. N. A. Aziz[1], and A. W. Azman[2]
*[1]Department of Mechatronics*
*[2]Department of Electrical and Computer Engineering*
*Faculty of Engineering, International Islamic University Malaysia (IIUM)*
*Jalan Gombak, 53100 Kuala Lumpur, Malaysia.*
*ariffrashidan@gmail.com, yasir@iium.edu.my, aashafie@iium.edu.my,*
*fiqahzainuddin@gmail.com, nornadirahaziz89@gmail.com, amy@iium.edu.my*

## *Abstract*

*Public surveillance monitoring is rapidly finding its way into Intelligent Surveillance System. Street crime is increasing in recent years, which has demanded more reliable and intelligent public surveillance system. In this paper, the ability and the accuracy of an Adaptive Neuro-Fuzzy Inference System (ANFIS) was investigated for the classification of moving objects for street scene applications. The goal of this paper is to classify the moving objects prior to its communal attributes that emphasize on three major processes which are object detection, discriminative feature extraction, and classification of the target. The intended surveillance application would focus on street scene, therefore the target classes of interest are pedestrian, motorcyclist, and car. The adaptive network based on Neuro-fuzzy was independently developed for three output parameters, each of which constitute of three inputs and 27 Sugeno-rules. Extensive experimentation on significant features has been performed and the evaluation performance analysis has been quantitatively conducted on three street scene dataset, which differ in terms of background complexity. Experimental results over a public dataset and our own dataset demonstrate that the proposed technique achieves the performance of 93.1% correct classification for street scene with moving objects, with compared to the solely approaches of neural network or fuzzy.*

***Keywords:*** *Moving object detection, neural fuzzy systems, object classification, street crime, visual surveillance*

## 1. Introduction

Closed-circuit television (CCTV) is popular among law-abiding citizens who perceive it as a preventive measure to feel much safer through public surveillance. Thus, it will be more reliable if it can fit the purpose to prevent, rather than only can detect crime, and as such implementation it would be very useful. At present, it has been served as resourceful assistance in crime investigation; however CCTV is used mostly as post investigation tool, rather than real-time preventive measures tool. Thus, the acquisition visual sensor such CCTV should be truly intelligent and able to capture scenes at where it is installed, as this could facilitate the process of decision making. Inspired by this idea, the development of research in manipulating CCTV data into an automatic decision making system has received much attention in the past decade. One of important measures in developing such system is recognition of moving objects. Among many approaches, Neuro-fuzzy modeling is one of the most recent techniques that able to achieve high accuracy in moving object classification and recognition. The Neuro-fuzzy modeling incorporates the advantages of fuzzy logic and neuro-learning model in order to make the system justify the action based on the object classification decision.

In this paper, a moving object detection and classification system which comprised of three major processes is presented. First, the adaptive background subtraction with a mixture of Gaussians is developed in order to capture the moving foreground objects. The moving object features are then extracted in terms of spatial and temporal attributes. Finally, the object classification based on its shared-attributed features is implemented using fuzzy neural classifier.

The rest of the paper is organized as follows: Section 2 discusses the common approaches used in moving object detection algorithm based on gray-level resolution, as well as the classification of moving objects that is related to the field of smart surveillance system. The proposed technique is presented in Section 3, while the experimental results in terms of both quantitative and qualitative perspectives are discussed in Section 4. Finally, Section 5 provides some concluding remarks.

## 2. Related Works

Over the last decades, video analysis and understanding has been one of the main active fields of computer vision and image analysis, where applications relying on this field are various, like video surveillance, traffic monitoring, and object tracking [1] [2]. In this section, we will review papers that directly related to our research scope, i.e. on the development of the recognition of moving objects in video for smart surveillance system applications.

In smart surveillance system, the moving object recognition must possess three reasonable competences in providing wide visual security against the street crime. The first is to segment the foreground of moving objects from the background image; the second is to extract the discriminating features of the target in order to provide valuable feature vector; and the third is to classify the object to its eligible classes. Hence, it should be able to prioritize the high tendency classes contributed to any misbehaviour.

Moving object detection techniques usually manipulate the colour intensity [3] [4] [5], motion [6] [7] [8], and shape [2] [9] of an object as the feature vector. Mixture of Gaussians (MoG), histogram of oriented gradient (HOG), Temporal/Frame Differencing, optical flow, and statistical filter are among the primary approaches to detect the existence of moving targets in image sequences. Some researchers used the MoG to obtain the distortion of brightness and colours as feature vectors [3] [5]. The colour cue is vastly used in detection; however different forms of extraction method were applied. For example, [4] extract the illumination feature using HOG, and then calculate the brightness relation variation ratio between foreground and background image. In [10], discrete wavelet transformation was performed on images at pixelwise level to obtain the wavelet coefficient of entropy variations as the input feature vector. Another study used optical flow to extract the features of speed and direction as significant features [6]. In [2], the calculation of Signal-to-Noise ratio (SNR) from region-of-interest area was used to classify different types of targets such as vehicle, animal, and human. In [2], they used geometric properties of fitted ellipse and the star skeleton with respect to the combined features of shape and motion. While [9] and [11] benefit the advantage of geometric shape by constructing vertical projection histogram of its binary silhouette. The detection performance is improved by constructing a good background model. They also used parametric probability density functions in the form of weighted sum of Gaussian models in order to segment the foreground pixels from the background.

When the discriminant features have been obtained, the requirement to recognize the target is demanded. Recently, neural network [12] [13] and fuzzy logic [14] classifier has received much attention in the development of the object's classification task in surveillance system [15]. In [16], the classification of vehicle frontal-view images use a combination of low level local features and high level global features through two stages of convolutional neural network (CNN). However, this approach is sensitive to noise since Laplacian filtering was used. The increasing noise in an image will degrade the magnitude of detected edges. In addition, its operation gets diffracted by some of the existing edges in the noisy image. In [14], fuzzy rule classification based on HSV colour scheme is used to classify the pixel of moving pedestrian and background. In [17] performs rigid object classification by the integration of Neural Network (High-Order NN) and Fuzzy (Choquet). It is used together with Biomimetic Pattern Recognition (BPR) classifier to form feature input vectors of moments, area, and velocity of an object. In this paper, a new approach that is based on fuzzy neural network (FNN) is implemented in order to classify common moving objects in street scene that includes motorcycle, pedestrian, and car. From the related works, better accuracy of moving object classification can be obtained by combining the approaches of fuzzy logic and neural network as it unites the advantages and excludes the disadvantages between each other.

## 3. Proposed Algorithm

A smart surveillance system that is able to classify moving objects in street scenes is developed. It consists of three main processes which are moving object detection, feature extraction of moving object region and classification of objects using fuzzy neural network, as shown in Figure 1. A brief description about the object localization and object region extraction are discussed in the forthcoming sections, followed by the neuro-fuzzy neural networks which perform the classification process.

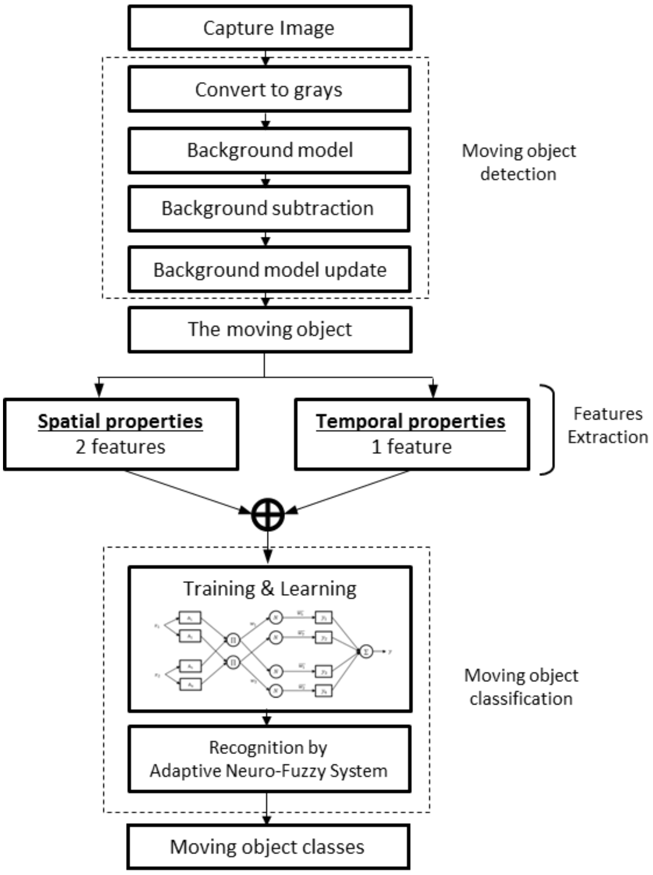


Figure 1. Proposed system diagram

### 3.1. Moving Object Detection

The detection of moving regions is the initial process in object recognition. The aim of moving object detection is extracting moving objects in image sequences which mostly interlace with the background pixel. The proposed detection algorithm implemented in this paper consists of three stages: i) background model generation; ii) background subtraction; and iii) background model update. Details of these stages are described in the followings subsections.

#### 3.1.1. Background model generation

Being motivated by the work of [18], the background is modelled as independent statistical approach at pixelwise level on each image sequences. The brightness light distribution are varies for each pixel, thus the density function of this distribution is reinstated by their matching colour regions and form a region-map of the image. The brightness probability density function is estimated by counting the occurrence rate of each brightness level in the region to generate spatial distribution. It is further normalized to obtain the total area under the curve equal to 1.

However, in the real world scene, the background varies depending on the scene, and it remains unknown to the system. Therefore, at first, a background model is generated through a training stage, which requires several initial image sequences with no moving object. This stage will initialize the parameters of mean, $\mu$, variances, $\sigma^2$, and weight, $\omega$. Each pixel is modelled as a distribution of Gaussian mixture models.

The probability value for each pixel can be written as:

$$Pr(X_t) = \sum_{i=1}^{K} \omega_{i,t} G_i\left(X_t, \mu_{i,t}, \Sigma_{i,t}\right) \tag{1}$$

where K is the number of distribution that determined by the available computational memory which default K=3, $\omega_{i,t}$ is the weight parameter of the K$^{th}$ Gaussian model, G is a probability density function, $\mu_{i,t}$ is the mean that denote the highest intensity value, $\Sigma_{i,t}$ is the covariance matrix of the K$^{th}$ component.

Gaussian function can be expressed as:

$$G_i\left(X_t, \mu_{i,t}, \Sigma_{i,t}\right) = \frac{1}{(2\pi)^{\frac{D}{2}} \left|\Sigma_{i,t}\right|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t - \mu_{i,t})^T \Sigma_{i,t}^{-1} (X_t - \mu_{i,t})} \tag{2}$$

The approach is based on the assumption that the values of RGB colour channel are independent and possesses the same variances. The normal distribution is parametrized in terms of mean and the variance, denoted by $\mu_{i,t}$ and $\sigma_i^2$ respectively.

$$\text{Mean, } \mu_{i,t} = \frac{\sum_{i=1}^{t} X_i}{t} \tag{3}$$

$$\text{Variance, } \sigma_i^2 = \frac{\Sigma_{i,t}}{I} = \frac{1}{t} \sum_{k-1}^{t} \left(X_k - \mu_{i,t}\right)^2 \tag{4}$$

$$\text{Weight, } \omega_{i,t} = (1 - \alpha),\ \omega_{i,t-1} + \alpha\left(M_{k,t}\right) \tag{5}$$

### 3.1.2. Background subtraction

After the parameter's initialization for background model has been made, the constructed model is used to calculate the difference with tested frame in which it will correspond to the moving region of interest. The background pixel visibly appears more frequent than foreground pixels. If the current pixel matches with any distribution model and satisfies Equation 6, it will be classified as background pixel and the parameters of $\mu_{i,t}$, $\sigma_i^2$, and $\omega_{i,t}$ will get updated. Otherwise, if the pixel did not matched with any distribution, the rank mode will be replaced with a new one with $X_t = \mu_{i,t}$ and it will be set as foreground pixel. Each pixel was compared against the mixture of Gaussian models based on the equation below:

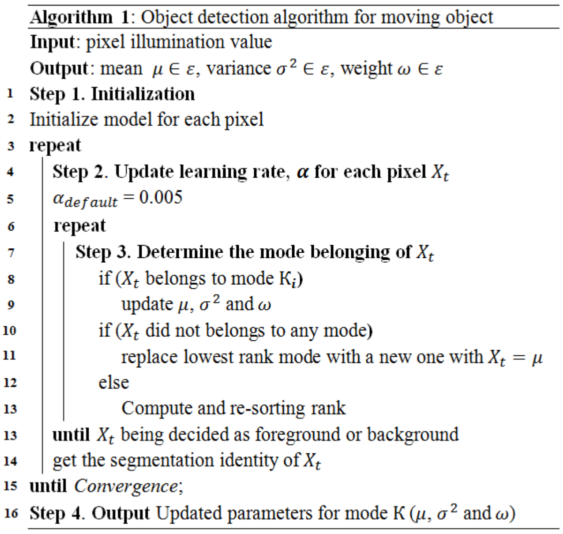
$$B = arg\left[min_b\left(\sum_{i=1}^{b}\omega_{i,t} > T\right)\right] \qquad (6)$$

where $\omega_{i,t}$ is the mixture weight, and $T$ is the threshold value.

As the background model generation is computed at pixelwise level, the thresholding in foreground segmentation stage will benefit the system in terms of computational cost, which is essential in order to achieve real time performance.

### 3.1.3. Background model update

The iterative update of background model parameters was carried out as in Algorithm 1.



```
Algorithm 1: Object detection algorithm for moving object
Input: pixel illumination value
Output: mean $\mu \in \varepsilon$, variance $\sigma^2 \in \varepsilon$, weight $\omega \in \varepsilon$
Step 1. Initialization
Initialize model for each pixel
repeat
    Step 2. Update learning rate, $\alpha$ for each pixel $X_t$
    $\alpha_{default} = 0.005$
    repeat
        Step 3. Determine the mode belonging of $X_t$
            if ($X_t$ belongs to mode $\mathrm{K}_i$)
                update $\mu$, $\sigma^2$ and $\omega$
            if ($X_t$ did not belongs to any mode)
                replace lowest rank mode with a new one with $X_t = \mu$
            else
                Compute and re-sorting rank
    until $X_t$ being decided as foreground or background
    get the segmentation identity of $X_t$
until Convergence;
Step 4. Output Updated parameters for mode K ($\mu$, $\sigma^2$ and $\omega$)
```

### 3.2. Features Extraction

After performing detection, it is crucial to extract the important feature vector from the moving objects. This will be used as a core for classification stage. Extensive experimentation has shown that using temporal feature only is insufficient for object classification. Thus the approach in this paper infuses properties from both spatial and temporal attributes of an object. The system accumulates this information for construction of classification vector.

### 3.2.1. Spatial properties

Instead of pixel based, shape based approach is used to extract the spatial properties, as it is prone to be less sensitive to adverse environment changes. Thus, it provides effective and low computation burden to the whole system. The consideration is made on the appearance attributes of moving target in each image frame. The whole shape of the object could be represented by using compactness and height-to-width ratio. The compactness is calculated as in Equation 7:

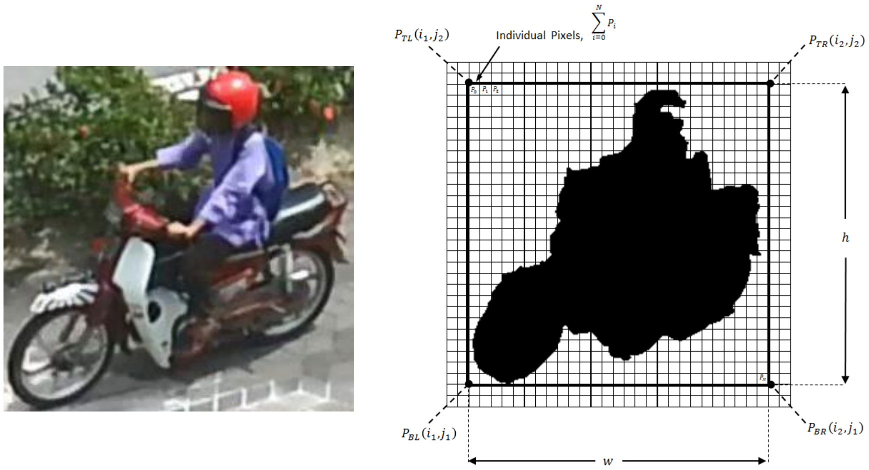
a) Compactness

$$C = \frac{Area}{Perimeter^2} \tag{7}$$

b) Height-to-width ratio (HWR)

Meanwhile, for the height-to-width ratio, the vertices of ROI are denoted by four points which are $P_{TL}$ as top-left point, $P_{BL}$ as bottom-left point, $P_{TR}$ as top-right point, and $P_{BR}$ as bottom-right point, as described in Equation 8. The illustration of the coordinates $P_{TL}$, $P_{BL}$, $P_{TR}$, and $P_{BR}$ are shown in **Error! Reference source not found.**.

$$P_{TL} = (i_1, j_2),\ P_{BL} = (i_1, j_1), P_{TR} = \ (i_2, j_2), P_{BR} = (i_2, j_1) \tag{8}$$



(a) (b)

Figure 2. The ROI properties.
(a) The extracted moving object (b) The geometrical calculation from the foreground image

The width, $w$ and height, $h$ of the moving object can be determined from,

$$w = \ i_2 - \ i_1 \ \text{ and } \ h = \ j_2 - \ j_1 \tag{9}$$

Consequently, the height-to-width ratio (HWR) was derived as,

$$HWR = \frac{h\ (pixel)}{w\ (pixel)} \tag{10}$$

### 3.2.2. Temporal properties

For the temporal-based properties, the approach of optical flow is adopted to obtain primary motion information. ROI obtained in Section 3.1, the position of center point is extracted. In order to provide inexpensive computation, only the speed of the center point pixel would be utilized to form the feature vectors. The center point $(x_c, y_c)$ was calculated for each bounding box found in Section 3.1 by:

$$x_c = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad and \qquad y_c = \frac{1}{N}\sum_{i=1}^{N} y_i \tag{11}$$

The fundamental assumptions made are the movement of brightness pattern of any pixel is constant over time, and each pixel over the whole image moves in a similar manner in term of velocity smoothness. The pixel position was represented by two-dimensional coordinate, $(x, y)$ where $x$ and $y$ denote the horizontal and vertical positions, respectively. A Taylor series approximation is applied in this approach at local based. The local operation involved is the differential between spatial and temporal coordinate [19].

In Equation 12, assume that $P_c(x, y, t)$ is the center pixel in $m \times m$ neighbourhood, $\delta x, \delta y$ are the slight movement in time, $\delta t$.

$$P_c(x, y, t) = P_{c+1}(x + \delta x, y + \delta y, t + \delta t) \tag{12}$$

In Equation 12, both sides of the equation contain identical pixel that described on the adjacent images, thus it can be expressed as Equation 13.

$$P_x V_x + P_y V_y = -P_t \tag{13}$$

where $P_x, P_y, P_t$ are pixel's intensity derivatives in $x, y, t$ respectively, and $V_x, V_y$ are the velocity components of $P_c(x, y, t)$ that mentioned in Equation 12. It will then yield two dimensional vectors that carry information of velocity magnitude and direction of motion is assigned to each pixel in an image, given by Equation 14.

$$P_{x,y,t} = \begin{pmatrix} Velocity\ magnitude_{x,y,t} \\ Direction_{x,y,t} \end{pmatrix} \tag{14}$$

The apparent velocity estimation of the moving objects is often very valuable feature in classifying between motorcyclist and pedestrian, since the the velocity difference is very large among these two classes. Thus, the feature vector is augmented with normalized velocity of the detected objects.

### 3.3. Moving Object Classification

In the previous Section 3.2, the discriminant vector has obtained; thus, it will be further utilized in a proper defined classifier. The Adaptive Neuro-Fuzzy Inference System (ANFIS) classifier is adopted to eliminate the limitations in the individual implementation of neural network or fuzzy. In the following subsections, the implemented structure, learning process and moving object classification are described.

#### 3.3.1. Network architecture

The learning capability of ANN was fully manipulated for automatic *IF-THEN* rules generation and parameter optimization of fuzzy system, which mostly regards as the common problem of fuzzy system. Therefore, three inference systems were independently developed for the intended moving object classes (pedestrian, motorcycle, and car). The structure of the system for each class is as shown in Figure 3.

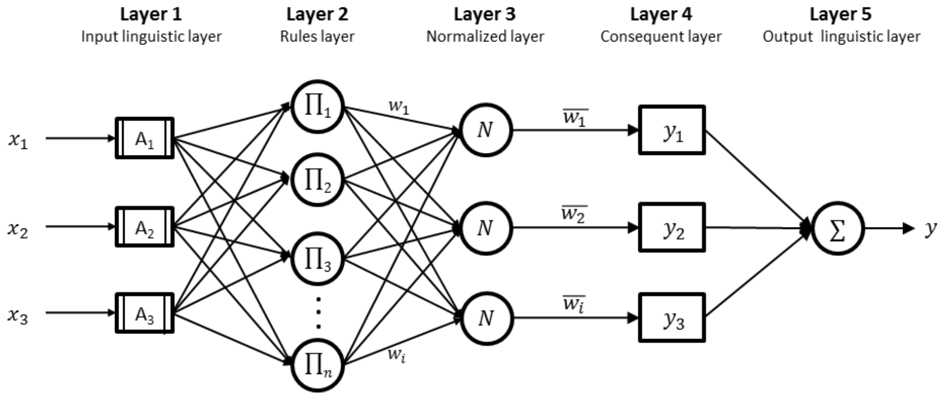


Figure 3. Structure of the ANFIS network

The initial part of ANFIS is based on first-order Takagi-Sugeno-Kang (TSK) method, since the consequence parameter is in linear equation terms. Each of them contained three input nodes of $x_1 - x_3$, 27 rules of TSK, and one output variable, $y$. Each of ANFIS fuzzy rules is in the form of *IF-THEN,* as shown in Equation 15:

$$\text{Rule } N \ : \quad \text{IF } x_i \text{ is } A_i^N \text{ and } x_{i+1} \text{ is } A_{i+1}^N\text{, THEN } y_N = a_0^N + \sum_{j=1}^{n} a_j^N x_j \qquad (15)$$

where $A_i^N$ is the membership functions, $x_i$ are the inputs, and $a_j^N$are the parameters of consequent equation. The reasoning system for TSK model constitutes of five layers, and the function of each layer is described as follows:

**Layer 1**. *Input linguistic layer*. The nodes are in the form of linguistic variables, and functioning as TSK rule bases. Each node $A_i^N$ performs a membership value computation. The Gaussian function is adopted as membership function in calculating the degree of membership value, which could be defined as:

$$\mu_{A_i}(x_i) = exp\left[-\frac{(x_i - c_i)^2}{2{\sigma_i}^2}\right] \qquad (16)$$

where $c_i$ and $\sigma_i$ are respectively the mean (or center) and the variance (or width) of the Gaussian membership function of the $x_i$ input variable.

**Layer 2**. *Rules layer*. This layer is performing the algebraic product operation of all the functions obtained from the previous layers. The outputs, $w_i$ from this particular layer represent the firing strength. It is determined by:

$$w_i = \prod_{j=1}^{n} \mu_{A_j(x_j)} \tag{17}$$

**Layer 3**. *Normalized layer*. This layer constitutes of fixed nodes that calculate the ratio of the $N^{th}$ firing strength, $w_i$ to the sum of all firing strengths. The normalized firing strength is given by,

$$\overline{w_{\mathrm{i}}} = \frac{w_i}{\sum_{k=1}^{r} w_k} \tag{18}$$

where $r$ is total number of rules.

**Layer 4**. *Consequent layer*. Every node in this layer is an adaptive node which calculates the consequence value, $O_i$, given by

$$\overline{w_{\mathrm{i}}} y_N = \overline{w_{\mathrm{i}}}(a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n) \tag{19}$$

where $\overline{w_{\mathrm{i}}}$ is the normalized firing strength from layer 3 and $(a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n)$ is the parameters of these nodes.

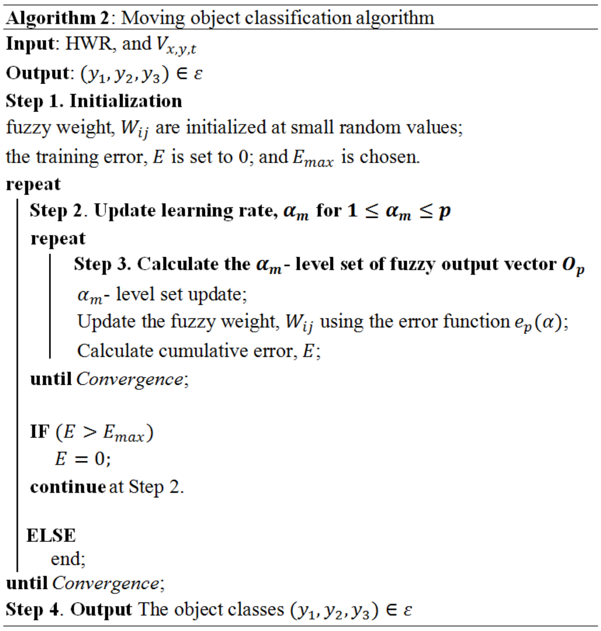
**Layer 5**. *Output linguistic layer*. It includes a fixed node denoted as $\sum$, that functions as a summation of overall ANFIS output network. This implies a defuzzification operation in order to obtain the crisp value. By the Weighted Fuzzy Mean (WFM) method, the overall output, $y$ is obtained by,

$$y = \sum_i \overline{w_{\mathrm{i}}} y_N = \frac{\sum_i w_i y_i}{\sum_i w_i} \tag{20}$$

### 3.3.2. Learning and testing

In the training stage, three FIS were developed, in which each system is specifically for each output class. The network input is a feature vector that composed of two major elements; two spatial features, and one temporal feature. Meanwhile, each network has one output node, which will eventually be adjoined to recognized three different classes of moving objects including pedestrian, motorcycle, and car.

Initially, the ANFIS training was done using our own datasets. All moving objects in the dataset are divided into three categories: pedestrian, motorcycle, and car. 200 positive samples, 1434 negative samples, and 200 test samples were randomly selected for each class in the ANFIS parameter training. The overall classification procedure is summarized in Algorithm 2.



**Algorithm 2**: Moving object classification algorithm

```
Input: HWR, and $V_{x,y,t}$
Output: $(y_1, y_2, y_3) \in \varepsilon$
Step 1. Initialization
fuzzy weight, $W_{ij}$ are initialized at small random values;
the training error, $E$ is set to 0; and $E_{max}$ is chosen.
repeat
    Step 2. Update learning rate, $\alpha_m$ for $1 \leq \alpha_m \leq p$
    repeat
        Step 3. Calculate the $\alpha_m$- level set of fuzzy output vector $O_p$
        $\alpha_m$- level set update;
        Update the fuzzy weight, $W_{ij}$ using the error function $e_p(\alpha)$;
        Calculate cumulative error, $E$;
    until Convergence;

    IF $(E > E_{max})$
        $E = 0$;
    continue at Step 2.

    ELSE
        end;
until Convergence;
Step 4. Output The object classes $(y_1, y_2, y_3) \in \varepsilon$
```

The classification algorithm adjusts the consequent parameters of layer 4 in feed-forward propagation. Whereas, the backward propagation was applied iteratively to minimize the error. The output of the the network is in terms of probability value in the range of $[0, 1] \in \mathbb{Z}^+$. As the desired outputs were to be in finite value, a simple threshold is applied. Therefore the outputs were (1, 0, 0), (0, 1, 0), and (0, 0, 1) for the pedestrian, motorcycle, and car respectively.

During the testing stage, the feature vector of each region-of-interest that detected by segmentation process will be extracted. This discriminant vector will be given to the network as an input. The output of linguistic layer will determined the belonging class of the moving object, which would be in the sequence of $(y_1, y_2, y_3)$.

## 4. Results and Discussion

In this section, several experiments were conducted to evaluate the performance and robustness of the proposed method against different video sequences of street scenes.

The optimized classifier was obtained from Section 3.3.2, and was used to test on different videos as shown in Table 2. Each test dataset varies in sizes of $854 \times 480$, $1024 \times 720$, and $1280 \times 720$ pixels, respectively. These videos were converted to uniform size of image sequences, which is $320 \times 240$ pixels. These images will be processed by the system to detect and classify the intended moving objects. In order to test the classifier accuracy, our proposed method was compared to the notable approaches in this field used by [17] and [20] on the real traffic video datasets maintained by KOGS-IAKS, University of Karlsruhe. The proposed algorithm runs on a workstation with Intel® i7-4700MQ 2.40 GHz CPU, and the software is written on MATLAB platform. The performance of the proposed algorithm is analyzed based on its running time at each process. The result is shown in Table 1, where most of the running time is spend in the feature extraction stage. However, with the average of 31.9 fps for the entire process, the proposed algorithm can perform well in real-time applications as the common rate of CCTV running rate is around 30 fps.

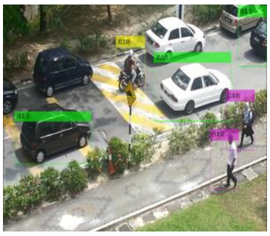
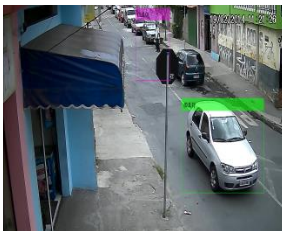
Table 1. Processing rate performance of proposed method

| Function | Moving object detection | Feature Extraction | Moving object classification | Average frame rate |
|---|---|---|---|---|
| Running rate (fps) | 42.6 | 22.8 | 30.4 | 31.9 |

## 4.1. Accuracy analysis of proposed method

Our proposed method was tested on three different image sequences of *Street_1*, *Street_2* and *Street_3* datasets, respectively. All videos were outdoor street scenes that contain non-stationary background object (such as moving leaves, shadow, etc.), and exposed to illumination changes. The confusion matrices are shown in Table 2. The classification for pedestrian, motorcycle, and car are automatically annotated by different colour of magenta, yellow, and green, respectively.

Table 2. Classification accuracy of proposed method using several datasets

|  | (a)<br>*Street_1* | (b)<br>*Street_2* | (c)<br>*Street_3* |
|---|---|---|---|
| Classification results |  | |  |
| Confusion matrices | Ped Mot Car<br>Ped 85.1 8.2 6.7<br>Mot 6.8 82.9 10.3<br>Car 0 8.8 91.2 | Ped Mot Car<br>Ped 81.9 10.1 8.0<br>Mot 7.2 85.5 7.3<br>Car 2.6 9.7 87.7 | Ped Mot Car<br>Ped 93.4 6.6 0<br>Mot 1.6 96.4 2.0<br>Car 0 3.3 96.7 |

Quantitatively, for *Street_1* dataset, our proposed method achieves 85.1% correctly classified pedestrian, 82.9% correctly classified motorcycle and 91.2% correctly classified car. Notable misclassification occurrence for motorcycle and car are 10.3% and 8.8% respectively. A possible reason for this phenomenon might be due to the camera angle that ends up capturing the motion of object in the diagonal direction and the objects have very similar velocity from the observer’s point of view. For example, as the motorcycle moves straight away from the observer, it looks very similar like a car. Nonetheless, pedestrian and motorcycle are classified relatively well in this dataset.

Meanwhile, for *Street_2* dataset, the classification of pedestrian, motorcycle and car achieves 81.9%, 85.5% and 87.7%, respectively. From the observation of accuracy performance of *Street_2* and *Street_3* dataset, the classifier has problem in distinguishing pedestrian from motorcycle, with the misclassification of 10.1% and 6.6% respectively. The underlying reason is due to the generic upper part appearance of these two adjacent classes. In the event that the moving object detection algorithm could only detect the upper part of the motorcycle riders, there are several close affinities between the discriminant vectors of pedestrian and motorcycle in the database. As a result, misclassifications occur, which mostly due to sudden illumination changes. In addition, factors such as side's perspectives of the objects and far distance from camera causes several features become less useful to define the specified classes.

On the other hand, the highest accuracy is obtained in *Street_3* dataset, as the proposed method achieves 93.4% correctly classified pedestrian, 96.4% for motorcycle, and 96.7% for car. Meanwhile, there is a notable small misclassification occur between pedestrian and motorcycle with 6.6% pedestrian was predicted as motorcycle and 1.6% actual motorcycle was misclassified as pedestrian. We can deduce this is happened due to the small variability in the motorcycle class that was in the database in the training stage.

By comparing the performance of all the datasets, we can summarize the correctly-classified classes and misclassification classes as given in Table 3. From the table, it is clear that the highest class that have been correctly classified is car (91.9%), followed by motorcycle (88.3%) and pedestrian (86.8%), which yield 89.0% in average. Meanwhile, the highest misclassification occur between pedestrian and motorcycle which is 8.3%, and 5.2%, vice versa. The average confusion matrix verifies the effectiveness and stable performance of the proposed approach.

Table 3. Average confusion matrix of proposed method on all tested datasets

|  | Ped | Mot | Car |
|---|---|---|---|
| Ped | 86.8 | 8.3 | 4.9 |
| Mot | 5.2 | 88.3 | 6.5 |
| Car | 0.9 | 7.3 | 91.9 |

## 4.2. Comparison result

For the classifier comparison, we compare our proposed method with other well established methods used in this field, as shown in Table 4. For fair comparison basis, all methods are tested by using KOGS-IAKS dataset[1]. This dataset contains real traffic video sequences which are suitable for this research.

Table 4 shows the result with its adopted concepts and features selection for the BPR, BPR-CI and our proposed method. The best accuracy is calculated on the performance of recognition with respect to each class. For BPR classifier method that solely depends on NN, the correct classification accuracy obtained is 88.70%. Meanwhile, there are competitive results illustrated between BPR-CI classifier and our proposed method. Both methods are the infusion of NN and Fuzzy that give better result as the accuracy obtained are 92.9% and 93.1%, respectively. We can see that the difference in the accuracy of both methods is very small. However, our approach yields the best recognition rate for 3 out of 4 classes. Thus, with average accuracy of 93.1%, our proposed method is an effective and efficient method that able to work in real traffic scene.

[1] KOGS-IAKS dataset can be retrieved from http://i21www.ira.uka.de/image_sequences/ which maintained by Institute of Algorithms and Cognitive Systems, University of Karlsruhe, Germany.

Table 4. Classification result of different method on dataset KOGS-IAKS

| Methods | Adopted concept | Features | Ped (%) | Mot (%) | Car (%) | Other (%) | Average (%) |
|---|---|---|---|---|---|---|---|
| BPR [20] | NN | Moment, area, velocity | 82.3 | 88.5 | 91.2 | 90.8 | 88.70 |
| BPR-CI [17] | Two-stages NN-Fuzzy | Moment, area, velocity | **91.3** | 92.5 | 94.2 | 93.7 | 92.9 |
| **Proposed** | **One-stage Neuro-fuzzy** | **shape & motion** | 90.5 | **92.9** | **95.8** | **-** | **93.1** |

## 5. Conclusion

In this paper, we have described a novel approach to detect and classify moving objects in street scenes. For the detection method, we are using background subtraction technique incorporating Gaussian mixture distribution method. It performs with stability in dynamic scenes, and less impacted by illumination changes. The approach used for the classification is based on the infusion of Fuzzy and NN. Based on experimental results, the better result is achieved as the infused concept outweighs both advantages and disadvantages of solely implementation concept. The detected moving objects are classified into three main groups by the method of combining spatial and temporal based features. In addition, the proposed algorithm was tested with three different scenes in which significant classification accuracy has been achieved, and proves the robustness of the proposed method.

## Acknowledgements

The Ministry of Higher Education (MOE) kindly provided funding for the research through the Fundamental Research Grant Scheme (FRGS), project No. FRGS 12-030-0271. This work is a collaboration of IIUM and Royal Malaysia Police (RMP) to reduce street crime rate in the country.  The authors also thank the anonymous reviewers for their valuable comments.

## References

[1] M. T. Razali, "Detection and classification of moving object for smart vision sensor," in *Information and Communication Technologies (ICTTA)*, **2006**.

[2] Y. Bogomolov, D. Gideon, L. Stanislav, E. Rivlin, and M. Rudzsky, "Classification of Moving Targets Based on Motion and Appearance," in *British Machine Vision Conference (BMVC)*, Norwich, **2003**, pp. 1-10.

[3] R. Zhang, S. Zhang, and S. Yu, "Moving Objects Detection Method Based on Brightness Distortion and Chromaticity Distortion," *IEEE Transactions on Consumer Electronics*, **2007**.

[4] F. P. Li, B. Li, Z. Song, M. J. Wu, and C. Shen, "Detecting Shadow of Moving Object based on Phong Illumination Model," in *International Conference on Information Sciences, Machinery, Materials and Energy (ICISMME)*, **2015**.

[5] C. I. Patel and R. Patel, "Illumination Invariant Moving Object Detection," *International Journal of Computer and Electrical Engineering*, vol. 5, no. 1, Feb. **2013**.

[6] C. S. Royden and M. A. Holloway, "Detecting moving objects in an optic flow field using direction- and speed-tuned operators," *Journal of Vision Research*, vol. 98, pp. 14-25, May **2014**.

[7] H. Fradi and J.-L. Dugelay, "Robust foreground segmentation using improved gaussian mixture model and optical flow," in *International Conference on Informatics, Electronics & Vision (ICIEV)*, Dhaka, Bangladesh, **2012**, pp. 248-253.

[8] S. Indu, M. Gupta, and A. Bhattacharyya, "Vehicle tracking and speed estimation using optical flow method," *Int. J. Engineering Science and Technology*, vol. 3, no. 1, pp. 429-434, **2011**.

[9] I. Haritaoglu, D. Harwood, and L. S. Davis, "W4: real-time surveillance of people and their activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. Vol. 22, no. No. 8, pp. 809-830, Aug. **2000**.

[10] W. Cho, S. Kim, and G. Ahn, "Detection and recognition of moving objects using the temporal difference method and the hidden Markov model," in *Computer Science and Automation Engineering (CSAE), IEEE International Conference*, Shanghai, **2011**.

[11] H. Lee, J. Kim, and J. Kim, "Decision fusion of shape and motion information based on bayesian framework for moving object classification in image sequences," *Foundations of Intelligent Systems*, pp. 19-28, **2006**.

[12] P. Sengottuvelan and R. Arulmurugan, "Object classification using substance based neural network," *Mathematical Problems in Engineering*, **2014**.

[13] M. A. Rashidan, Y. M. Mustafah, and S. B. A. Hamid, "Detection of Different Classes Moving Object in Public Surveillance Using Artificial Neural Network (ANN)," in *International Conference In Computer and Communication Engineering (ICCCE)*, **2014**, pp. 240-242.

[14] M. Sivabalakrishnan and D. Manjula, "Fuzzy Rule-based Classification of Human Tracking and Segmentation using Color Space Conversion," *International Journal of Artificial Intelligence & Applications (IJAIA)*, vol. Vol. 1, no. No. 4, Oct. **2010**.

[15] C.-F. Juang and L.-T. Chen, "Moving object recognition by a shape-based neural fuzzy network," *Neurocomputing*, vol. 71, no. 13, pp. 2937-2949, **2008**.

[16] Z. Dong, Y. Wu, and Y. Jia, "Vehicle Type Classification Using a Semisupervised Convolutional Neural Network," *IEEE Transactions on Intelligent Transportation Systems*, no. No. 4, Aug. **2015**.

[17] L. Wang, L. Xu, R. Liu, and H. H. Wang, "An approach for moving object recognition based on BPR and CI," *Information Systems Frontiers*, vol. 12, no. 2, pp. 141-148, **2010**.

[18] Z. Zivkovic, "Improved Adaptive Gaussian Mixture Model for Background Subtraction," in *International Conference Pattern Recognition (ICPR) 2004*, **2004**, pp. 28-31.

[19] H. M. Nasab and S. Aslani, "Optical flow based moving object detection and tracking for traffic surveillance," *International Journal of Electrical, Electronics, Communication, Energy Science and Engineering*, vol. 7, no. 9, pp. 789-793, **2013**.

[20] W. Shoujue, C. Xu, and L. Weijun, "Object-Recognition with oblique observation directions Based on Biomimetic Pattern Recognition," in *International Conference on Neural Networks and Brain (ICNN&B)*, Beijing, **2005**, pp. 1498-1502.