九十年國際道路交通
安全與執法研討會
中華民國 90 年 9 月

結合模糊類神經網路與背景相減法應用於車輛偵測之研究

Vehicles Detection Using Background Differencing Method Incorporated with Fuzzy Neural Network

藍武王(Lawrence W. Lan)[1] 郭怡雯(April Y. Kuo)[2] 黃業傑(Y-C. Huang)[3]

摘要

本文結合模糊類神經網路與背景相減法，發展彩色影像車輛偵測模式，並以市區道路與高速公路之車流影像加以驗證，其中模糊類神經網路系統係以四層設計，再以倒傳遞演算法進行網路訓練。為比較不同偵測點數對偵測準確率的影響，本文分別測試三點、五點與七點之虛擬線偵測器，實驗發現五點與七點虛擬偵測器之偵測效果並無明顯差異，惟均較三點為佳。以七點偵測為例，結合模糊類神經網路的彩色影像偵測，無論是市區道路或高速公路，白晝或夜晚，其所得的車輛偵測準確率皆約達 90%或以上。

關鍵詞：彩色影像車輛偵測、模糊類神經網路、背景相減法

ABSTRACT

This paper develops a fuzzy neural network (FNN) color image vehicular detection (CIVD) system. Background differencing method is used to process the color-based traffic flow images. A four-layer fuzzy neural network is constructed and its network parameters are trained by backpropagation algorithm. Analog color traffic scenes from different roads with various lighting conditions are collected by a video camera and then converted into digital images in order to be processed by the computer. Pseudo detectors with three-, five- and seven-points are aligned on the monitor and their detection accuracy rates are compared. The experimental results show that detection "success" rates for this FNN CIVD system with seven detection points under various testing environments can reach 90% or even higher.

Key Words: Color Image Vehicular Detection (CIVD), Fuzzy Neural Network (FNN), Background Differencing Method

1. Introduction

Advanced traffic control and management relies heavily on automatic detection on traffic flow data. In recent years, more and more traffic parameters have been automatically collected by image detectors rather than by conventional techniques such as loop and magnetic detectors. The major drawbacks for the conventional detectors are their limitations on measuring such important parameters as traffic composition, and on assessing traffic conditions accurately. Other drawbacks may include small detection zone and placement without flexibility (Michalopulos, 1991). Besides, data collected by the conventional vehicular detectors cannot be applied in other areas including vehicle tracking, incident detection (Washburn and Nihan, 1999), and monitoring the movement of vehicles within a junction (Fathy and Siyal, 1995).

Traffic detection through image processing may improve the drawbacks mentioned above. Recently, extensive research and development efforts have been devoted to image processing techniques applied in traffic data collection and analysis (Hoose, 1992). More traffic parameters including vehicle classification and tracking and the related applications such as incident detection can be obtained through image processing detection. With the support of such languages as C++, Delphi and Visual Basic, the image processing detection has become more easily and friendly in facilitating the users to change the detection algorithms and interfaces.

[1] 國立交通大學交通運輸研究所教授兼所長(100 台北市忠孝西路一段 114 號 4 樓, Email: lwlan@cc.nctu.edu.tw)
[2] 國立交通大學交通運輸研究所碩士(Email:kuoyiwen.tt88g@nctu.edu.tw)
[3] 國立交通大學交通運輸研究所博士班研究生(Email:u8632501@cc.nctu.edu.tw)

Methods of applying image processing on vehicular detection include blob detection, pattern recognition and background differencing. Blob detection does not work well on rainy conditions because not all the vehicles are brighter or darker than the measured background road surface (Blosseville, *et al.*, 1989). Pattern recognition neither works very well when vehicles in the detection zone are not well suited into the defined templates (Dickson and Wan, 1989). In addition, both blob detection and pattern recognition methods often need more computation time to detect vehicles than the most common and simple approach used in traffic image detection -- background differencing technique (Fathy and Siyal, 1995).

Background differencing technique is based on pixel-by-pixel comparison of a background frame and the instantaneous frame of the traffic scenes (Dickinson and Waterfall, 1984a,b). Because it is sensitive to ambient lighting, this method tends to have successful detection in the daytime with fair weather. It may lose detection accuracy near dusk and dawn or in bad weather conditions. Most previous image processing researches are based on gray-level. A color-level image is consisted of red (R), green (G) and blue (B) pixels; thus it provides more information than a gray-level image. A fuzzy neural network (FNN) is characterized with learning ability and capabilities of dealing with uncertainties (Chiou and Lan, 1997), thus it may help to solve the variations of ambient lightings near dusk and dawn or in bad weather conditions. In other words, a color-based image processing through background differencing method incorporated with FNN seems promising to accommodate more environmental changes than the same color-based image processing without incorporating with FNN. For this reason, the main purpose of this paper is to develop a FNN CIVD detection system. Vehicular detection procedures containing four modules are to be explained. A four-layer fuzzy neural network is then constructed. Traffic scenes under various environments will be taken from the fields and then tested off-line to train the network parameters as well as to validate the detection accuracy.

2. System Design

The FNN CIVD system mainly contains four modules -- image digitalization, pseudo line detectors allocation, fuzzy neural network (FNN), and vehicle detection as depicted in Fig. 1. Analog traffic scenes are first digitalized by the image grabber, a pseudo line detector consist of appropriate number of detection points is then placed across a specific traffic lane on the monitor. The differencing of each pixel value (R, G, B) on each pseudo detection point between instantaneous traffic and background images is calculated through FNN model. If the differencing value is greater than a specific trained threshold value, it will be identified as a vehicle passing. The four modules in this detection system are explained in-depth as follows:

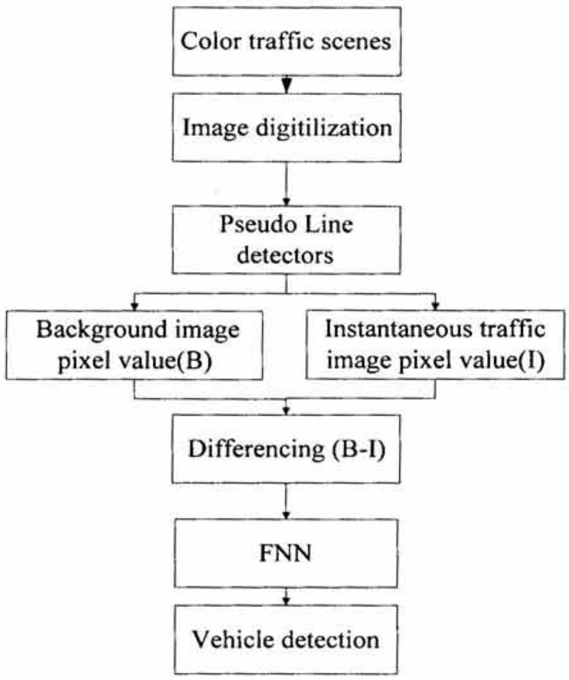

Fig. 1 Vehicular detection procedure

2.1 Image Digitalization Module

The function of this module is to convert analog images by an image grabber into digital images. After digitalization, video images can then be seen on the monitor and processed by the computer.

2.2 Pseudo Line Detector Module

The module is to define the coordinates of both ends of a pseudo line detector on the monitor. A pseudo line is in effect constituted by several detection points, which act as the instrument for vehicular detection. When creating a pseudo line detector, the right and left coordinates are directly input on the interactive window. While traffic flow video images are played, each detection point will automatically read R, G, B pixel values for the background image at the first one-tenth second. After that, each point will then read R, G, B pixel values for the traffic images every one-tenth second. The difference of pixel values for background and traffic images is used as the input data of fuzzy neural network module.

2.3 Fuzzy Neural Network Module

This module is to construct and train a fuzzy neural network. A training set, collected by the pseudo line detector module, is composed of 1000 to 1800 training examples. Each training example consists an input vector (differences of pixel values) and output vector (binary values indicating vehicle passing information). Fig. 2 is the training interface through which one can input desired network parameters. To generate training examples, one has to play the video traffic scene and then press the button "vehicle passing?" at the moment that one see a vehicle entering the pseudo line detector and press the same button again as it leaves the detector. Repeat such actions until a satisfactory number of training examples are obtained. Table 1 illustrates a training set with six training examples. R_i, G_i, B_i (i=1~7) are obtained by subtracting the pixel values of traffic scenes from the background every one-tenth second. Examples 1 through 3 with Veh = 0 represent no vehicle passing; while 4 through 6 with Veh = 1 indicate a vehicle passing.

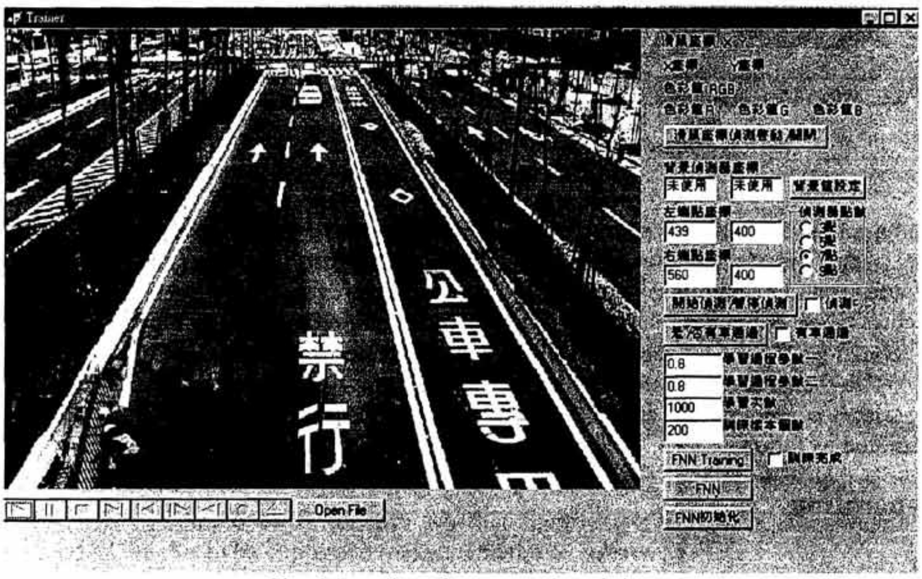

Fig. 2 Interface for network training

Table 1 An illustration of training set

No	R_1	G_1	B_1	R_2	G_2	B_2	R_3	G_3	B_3	R_4	G_4	B_4	R_5	G_5	B_5	R_6	G_6	B_6	R_7	G_7	B_7	Veh
1	0	4	9	0	0	0	0	4	9	0	0	0	0	0	0	0	0	0	0	0	9	**0**
2	0	4	0	0	4	0	0	4	9	0	0	0	0	0	0	0	4	8	0	0	9	**0**
3	0	4	9	0	4	0	0	4	9	0	0	0	0	4	0	0	4	8	0	0	9	**0**
4	140	138	49	148	138	50	140	130	41	148	130	42	132	86	25	8	12	41	8	8	17	**1**
5	25	36	83	16	57	82	15	81	116	8	69	99	33	12	24	8	16	25	16	8	17	**1**
6	140	106	16	148	98	9	124	78	8	41	0	49	16	24	33	17	16	16	16	12	25	**1**

Background pixel values may change over time as well as under different lighting conditions. Each lighting condition requires a specific trainings set; therefore, different fuzzy neural network must be trained under different lighting conditions (daytime or nighttime, rainy or clear, etc). Appropriate network parameters can be obtained through back propagation algorithm, which will be explained in the following section. With such trained parameters, one can further develop the vehicle detection system.

2.4 Vehicular Detection Module

This module is to determine whether there is a vehicle passing the pseudo line detector. Fig. 3 shows an example of vehicular detection interface through which the users can select appropriate environments and lighting conditions. As one moves the cursor to any position within the traffic scene, this module will automatically show the exact X and Y coordinates as well as the corresponding instantaneous traffic image pixel (R, G, B) values.

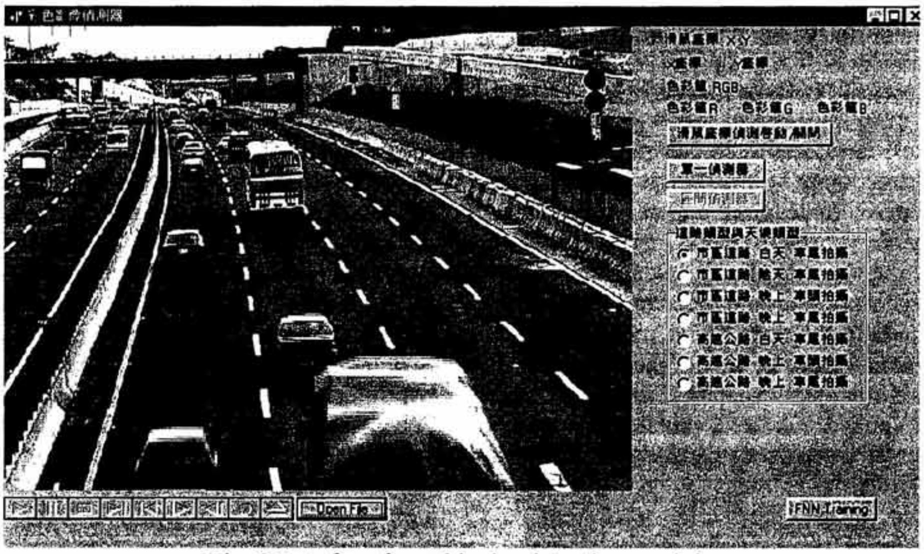

Fig. 3 Interface for vehicular detection module

3. Network Construction

In this study, a four-layers fuzzy neural network with q detecting points is constructed as shown in Fig. 4. The first layer, named input layer, receives the differencing of pixel values from pseudo detection points between instantaneous and background images and transmits these values to the second layer. The second layer, named membership layer, computes membership degree of each input value. The third layer is called rule layer. It connects the related links between the membership and rule nodes and calculates the weighted averages (R, G, B) of rule nodes. The fourth layer is output layer. It performs defuzzification to get numerical output value. Back propagation algorithm is employed to train the network parameters. The layer operation and training algorithm are further elaborated as follows.

3.1 Layers operation

(1) Input layer

$o_i^1 = f(u_i^1) = x_i^1$ for i=1~r (r=3q, where q is the number of pseudo detection points)

Since the weights (w_i^1) at layer one are set equal to unity, both output (o_i^1) and input (x_i^1) values at this layer can be further expressed as:

$$o_i^1 = u_i^1 = x_i^1$$

where

$$u_i^1 = w_i^1 x_i^1$$

o_i^1 = the output value (differencing of pixel values) of ith node at layer one.

x_i^1 = the input value (differencing of pixel values) of ith input value at layer one

w_i^1 = connection weight at layer one

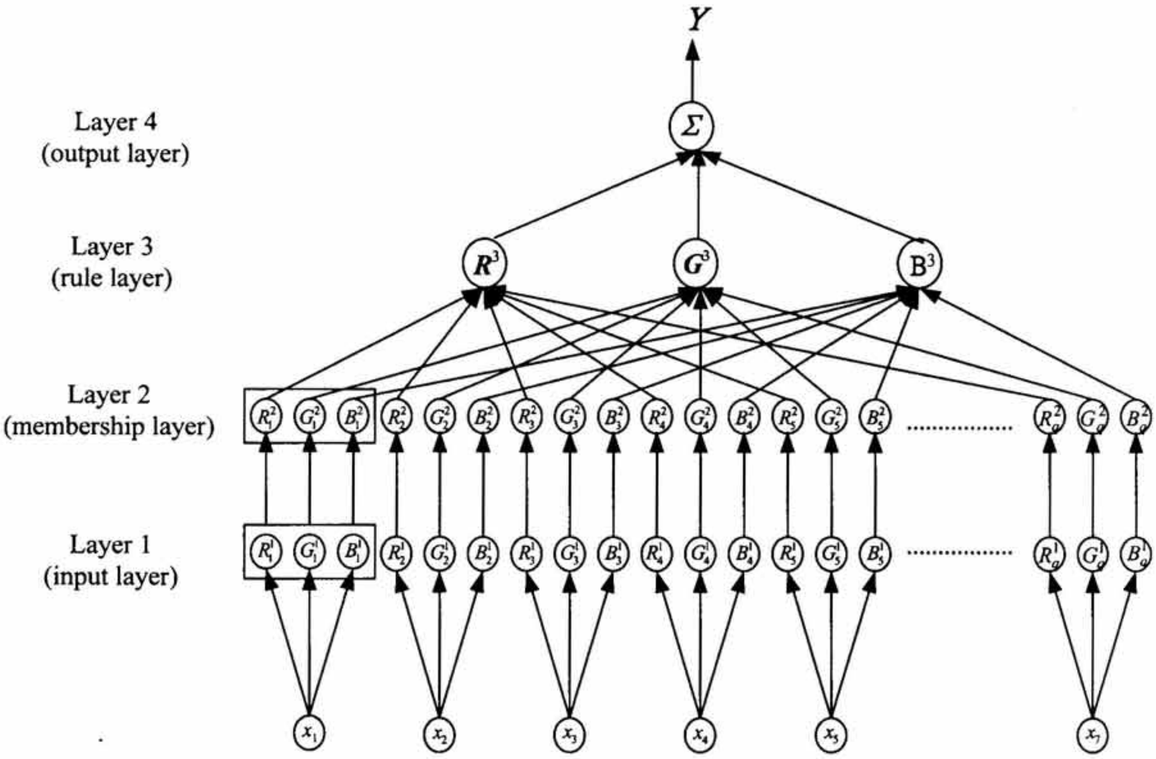

Fig. 4 A fuzzy neural network with q detection points

(2) Membership layer

The second layer in the network is the membership layer. The main function for this layer is to fuzzify the output values from layer one ($o_i^1 = u_i^1 = x_i^1$) by utilizing membership function and to decide membership degrees of input variables. The number of nodes in this layer is the same as that in layer one. Trapezoidal membership function as shown in Fig. 5 is utilized because its shape can correspond to the situations whether there exists a vehicle. A differencing of pixel values (x_j^2) less or equal to the threshold value (a) implies no vehicle passing over the detector. If x_j^2 is greater than the other threshold value (b), a vehicle is judged as passing over the detector. For the case of $a < x_j^2 \le b$, the corresponding membership degree is calculated and then a training algorithm is employed to identify whether there exists a vehicle. The illustrated trapezoidal membership function in Fig. 5 can be expressed mathematically as follows:

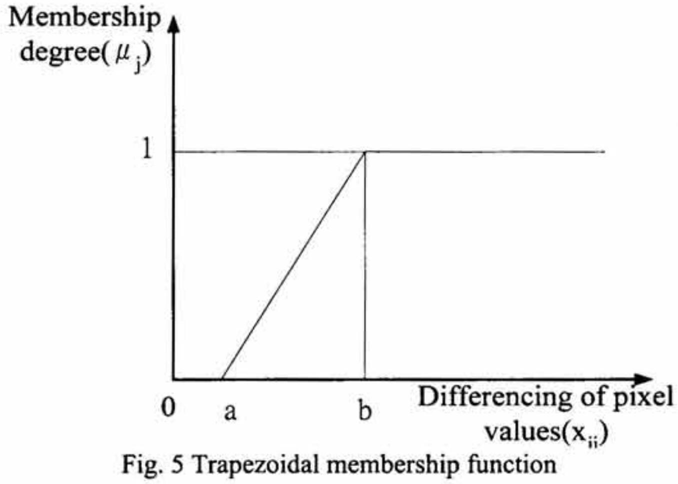

Fig. 5 Trapezoidal membership function

$$o_j^2 = f_j(u_j^2) = \mu_j(x_j^2)$$

$$= \begin{cases} 0 & \text{for } x_j^2 \le a_j^2 \\ \dfrac{x_j^2 - a_j^2}{b_j^2 - a_j^2} & \text{for } a_j^2 < x_j^2 \le \mathrm{b}_\mathrm{j}^2 \\ 1 & \text{for } x_j^2 > \mathrm{b}_\mathrm{j}^2 \end{cases} \qquad \text{for i=1~r, j=1~s}$$

where

o_j^2 = the output value of j^{th} node at layer two

x_j^2 = the input value of j^{th} node at layer two

a_j^2 = parameter of trapezoidal membership function

b_j^2 = parameter of trapezoidal membership function

(3) Rule layer

Rule layer, the third layer, connects the related links between the membership and rule nodes such that antecedent matching is determined. Its primary function is to establish various kinds of fuzzy inference rules to obtain the reasonable output. The nodes at this layer will perform a summation operation as follows.

$$o_k^3 = f(u_k^3) = \sum_{j=1}^{s} w_{jk}^3 \cdot x'_{jk} \quad \text{for} \quad \text{j=1~s, k=1~v}$$

The term (x'_{jk}) can be derived from the rule layer input value (x_{jk}) multiplied by a fixed value for the j^{th} detection node (z_j). W_{jk}^3 represents the connection weight between k^{th} rule node and j^{th} membership node. W_{jk}^3 is set equal to unity for all j and k. o_k^3 represents the output value of k^{th} node at layer three.

Since each detection point consists of R, G, B nodes, a pseudo line detector with q detection points will have 3q nodes. In this paper, q=3, 5, 7 will be experimented on a single traffic lane. To avoid the situation that any lane-changing vehicle taking very small lane space is counted in the designated detection lane, the fixed values on the detection nodes are set unequally in such a way that the values for the middle detection nodes are larger than those for both end detection nodes. Table 2 presents an example of the fixed values distribution over the detection nodes, which are to be used in this study.

Table 2 An example of fixed values distribution

Number of detection points	Number of nodes at layer two	Fixed value (Zj)	Nodes (j)
3	9	0.25	1,2,3,7,8,9
		0.5	4,5,6
5	15	0.1	1,2,3,13,14,15
		0.2	4,5,6,10,11,12
		0.5	7,8,9
7	21	0.05	1,2,3,19,20,21
		0.1	4,5,6,16,17,18
		0.2	7,8,9,13,14,15
		0.3	10.11.12

The case of seven detection points is explained as follows. Since nodes 1, 4, 7, 10, 13, 16, 19 in layer two will be aggregated into node 1 (R^3) of layer three, R^3 can then be expressed as:

$$R^3 = \sum_j w_{ji}^3 \cdot x'_{j1} \quad \text{for} \quad \text{j=1, 4, 7, 10, 13, 16, 19}$$

Similarly, G3 and B3 can be expressed as:

$$G^3 = \sum_j w_{j2}^3 \cdot x'_{j2} \quad \text{for} \quad j=2, 5, 8, 11, 14, 17, 20$$

$$B^3 = \sum_j w_{j3}^3 \cdot x'_{j3} \quad \text{for} \quad j=3, 6, 9, 12, 15, 18, 21$$

(4) Output layer

This layer performs the defuzzification to obtain numerical outputs by utilizing the center average defuzzifier. The connection weight W_{km}^4 between k^{th} rule and m^{th} output node represents the consequence fuzzy singleton. The variables (R^3, G^3, B^3) computed by the network produce a binary output value -- "0" representing no vehicle passing and "1" indicating a vehicle passing. At this layer, the node operation is expressed as follows.

$$o_m^4 = f(u_m^4) = \sum_{k=1}^{v} W_{km}^4 \cdot x_{km}^4 \quad \text{for} \quad m=1$$

The connection weights (W_{km}^4) are to be adjusted by the supervised training algorithm that is explained as follows.

3.2 Training Algorithm

Step (1) Set network parameters(α, a, b)

$$Learning\ \text{rate} = 1 - \frac{\text{n}}{\text{N}}$$

Learning rate (η) would decrease as the number of training cycles (n) increases. N represents the number of current training epochs. Initially, the network parameters, including momentum parameter (α), a_j, b_j are set equal to 0.8, 20, 35, respectively.

Step (2) Input a training example and compute the network output

A training example is composed of an input vector (differences of pixel values) and output vector (binary values indicating vehicle passing information). The output values at each layer are calculated as follows:

Layer1: $o_i^1 = x_i^1 \quad$ for i=1~r

Layer2:

$$o_j^2 = f_j(u_j^2) = \mu_j(x_j^2)$$

$$= \begin{cases} 0 & \text{for } x_j^2 \le a_j^2 \\ \dfrac{x_j^2 - a_j^2}{b_j^2 - a_j^2} & \text{for } a_j^2 < x_j^2 \le \text{b}_\text{j}^2 \\ 1 & \text{for } x_j^2 > \text{b}_\text{j}^2 \end{cases} \quad \text{for i=1~r, j=1~s}$$

Layer3: $o_k^3 = f(u_k^3) = \sum_{j=1}^{s} w_{jk}^3 \cdot x'_{jk} \quad$ for j=1~s, k=1~v

Layer4: $o_m^4 = \sum_{k=1}^{v} w_{km} x_{km} \quad$ for m=1

Step (3) Employ a network output and desired output to get δ_m^4 of output layer

$$\delta_m^4(t) = d_m^4(t) - o_m^4(t)$$

Step (4) Renew connection weight w_{km}^4 between rule layer and output layer

$$w_{km}^4(t+1) = w_{km}^4(t) + \eta \cdot \delta_m^4(t) \cdot x_{km}^4(t) + \alpha \ \Delta w_{km}^4(t)$$

Step (5) Compute propagated error signal δ_k^3 of rule layer

$$\delta_k^3 = \delta_m^4 \cdot w_{km}^4$$

Step (6) Compute propagated error signal δ_j^2 of membership layer

$$\delta_j^2 = \delta_k^3 \cdot z_j$$

Step (7) Renew the adjusted parameters of membership layer

$$a_j^2(t+1) = a_j^2(t) + \eta \cdot \delta_j \cdot \frac{x_j^2 - b_j^2}{(b_j^2 - a_j^2)^2} + \alpha \Delta a_j^2(t)$$

$$b_j^2(t+1) = b_j^2(t) + \eta \cdot \delta_j^2 \cdot \frac{a_j^2 - x_j^2}{(b_j^2 - a_j^2)^2} + \alpha \Delta b_j^2(t)$$

Step (8) Repeat step2 to step7

In this step, a total error square value is calculated. Repeat step 2 to step 7 until all training examples are finished (called an epoch). In each epoch, the energy function (TE) is calculated by

$$TE_n = \frac{1}{2}\sum_{t=1}^{T}(d_m(t) - o_m(t))^2$$

where $d_m(t)$ is the desired output for the t^{th} training example and $o_m(t)$ is the output of t^{th} training example in FNN.

Step (9) Test if the stop condition satisfies

Stop condition can be a predetermined fixed number of training cycles or the energy function converges; i.e. $|TE_n - TE_{n-1}| \le \varepsilon$. If the TE value decreases smoothly, a stop condition is reached. Otherwise, go to step 2.

4. Experiments

4.1 Data collection and evaluation criteria

A digital video camera is set up on a pedestrian or grade overpass to take the roadway upstream and downstream traffic scenes. The camera's field view is set as vertical to the road as possible in order to reduce vehicular occlusion situations. When videotaping traffic scenes in the daytime, a downstream viewing is taken; while in the nighttime, both upstream and downstream viewings are taken so that the effects of vehicle headlights and taillights on the detection accuracy can be compared.

Off-line experimental tests are conducted. Fuzzy neural network parameters are first trained to accommodate various lighting conditions. The trained network is then employed to detect the traffic flows. Pseudo line detectors with three-, five- and seven-detection points are equally spaced across a specific traffic lane on the monitor and their corresponding detection performances are compared.

The detection outcomes are classified into success, missing, and false. A "success" is referred as the situation that a vehicle is detected when that vehicle actually passes over the pseudo line detector. A "missing" is counted when no vehicle is detected but a vehicle in effect passes over the detector. A "false" is identified when a vehicle is detected but in fact no vehicle does exist. The detection accuracy is evaluated by these three criteria. In this paper, traffic on a specific lane is detected. Thus, a lane-changing vehicle that may takes a small portion of the lane width will not be counted as a success outcome. Both freeway mainline and urban street are the experimental fields. Videotaping hours are from 3 pm to 4 pm and from 6 pm to 7 pm. Detection performance in different times and locations are compared in the following subsections.

4.2 Freeway

The study location is at the Hsin-Chu mainline section of Freeway No.1 with three lanes southbound and four lanes northbound. Traffic scenes were videotaped on December 5, 2000 in good weather condition.

Training examples under different lighting conditions associated with various detection points are collected for network training. Table 3 presents the number of training examples and the converged epochs corresponding to the error values for different situations. For example, in nighttime downstream viewing with seven detection points, the number of training examples is 1195 and the energy function value is converged at about 600th epoch with an error value of 119. Fig. 6 shows the evolution of this network-learning process where X-axis represents the number of training epochs and Y-axis is the energy function value, or the error value (TE).

Table 3 Network training on freeway mainline

Videotaping time	Detection points	Training examples	Training epochs	Error value
Daytime (clear)	3	1040	3800	84
	5	1040	700	105
	7	1040	700	38
Nighttime (upstream viewing)	3	1120	2800	30
	5	1120	500	56
	7	1120	400	37
Nighttime (downstream viewing)	3	1195	3200	85
	5	1195	900	111
	7	1195	600	119

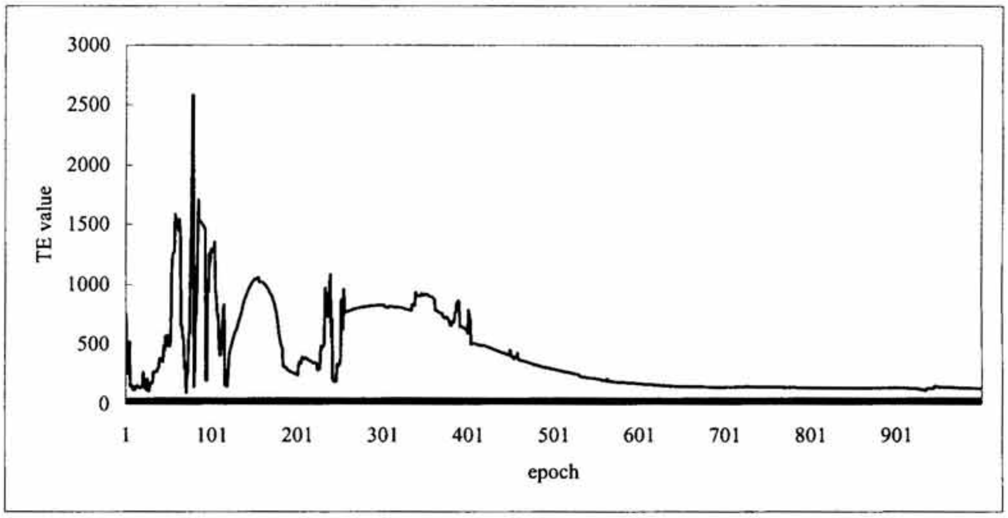

Fig. 6 Training process in freeway (nighttime downstream viewing with seven detection points)

Table 4 presents the detection performance in freeway. Under various lighting conditions combined with different numbers of detection points, in general, the detection performance in the daytime is the best. The performances of five and seven detection points are better than those of three points. Notice that the difference of detection "success" between five and seven detection points is insignificant. For the cases of detection failures, "missing" often occurs in upstream viewing while "false" often happens in downstream viewing. The main reason for detection failure in the daytime is due to the resemblance between the color pixels of some vehicles and the road background. Most of those vehicles are gray or near gray. The second reason is the lane-changing vehicles, which are not counted in this detection system.

The high missing rate in the nighttime upstream viewing is due to too high a threshold value set in the FNN system. In the nighttime, a vehicle is identified when both headlights are simultaneously detected by the pseudo line detector. If only one headlight is detected (lane-changing in most cases), then that vehicle will not be counted in this detection system. In contrast, the high false rate in the nighttime downstream viewing can be ascribed to too low a threshold value being set. Lane changing vehicles are mostly seen in this case.

Table 4 Freeway detection performance

Detection performance		Daytime (clear)	Nighttime (upstream viewing)	Nighttime (downstream viewing)
Number of vehicles passed		220	119	196
Success	3 detection points	198 (86.8%)	106 (84.8%)	186 (85.3%)
	5 detection points	210 (92.5%)	111 (88.8%)	190 (90.0%)
	7 detection points	212 (92.9%)	113 (90.2%)	188 (89.9%)
Missing	3 detection points	22 (9.6%)	13 (10.4%)	10 (4.5%)
	5 detection points	10 (5.2%)	18 (4.5%)	6 (2.8%)
	7 detection points	8 (3.5%)	6 (4.0%)	8 (3.8%)
False	3 detection points	8 (3.6%)	6 (4.8%)	22 (10.9%)
	5 detection points	7 (3.8%)	5 (2.2%)	15 (7.1%)
	7 detection points	8 (3.5%)	4 (1.8%)	13 (6.2%)

4.3 Urban street

The study location is at section 1 of Chunghwa Road, Taipei City, with four lanes northbound and five lanes southbound. Traffic scenes were videotaped on December 7, 2000 with fair weather from clear to cloudy.

Training examples are collected for network training. Table 5 illustrates the training conditions in the daytime and nighttime. For instance, in the daytime (cloudy) with five detection points, the number of training examples is 1280 and the energy function value converges at around 770^{th} epoch with an error value of 85. Fig. 7 shows such a network-learning process.

Table 5 Network training on urban street

Videotaping time	Detection points	Training examples	Training epochs	Error value
Daytime (clear)	3	860	900	54
	5	860	400	53
	7	860	900	58
Daytime (cloudy)	3	1280	3000	59
	5	1280	770	85
	7	1280	400	16
Nighttime (upstream viewing)	3	1600	900	7
	5	1600	750	25
	7	1600	850	17
Nighttime (downstream viewing)	3	1040	2000	83
	5	1040	1520	47
	7	1040	400	38

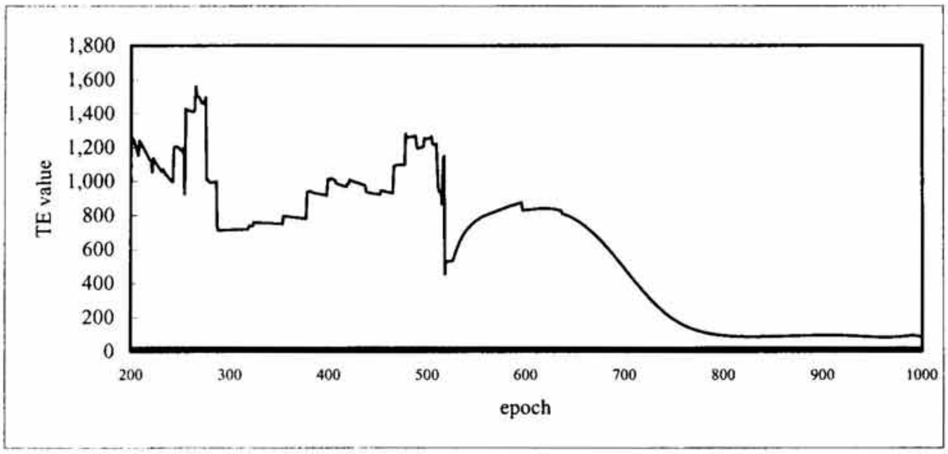

Fig. 7 Training process in urban street (daytime cloudy with five pseudo detection points)

Table 6 summarizes the detection performance. Except for three detection points, the detection performances in the daytime with five and seven detection points are the best under various conditions. Generally, the performance of five- and seven-points is better than that of three-points. Also notice that the difference of detection performance between five- and seven-points is slight. Reasons for the detection "missing" and "false" under different times of days with various weather conditions are the same as those mentioned in the freeway case.

Table 6 Urban street detection performance

Detection performance		Daytime (clear)	Daytime (cloudy)	Nighttime (upstream viewing)	Nighttime (downstream viewing)
Number of vehicles passed		153	114	96	144
Success	3 detection points	135 (83.8%)	100 (85.4%)	84 (82.3%)	130 (83.3%)
	5 detection points	144 (90.5%)	103 (90.0%)	88 (89.7%)	137 (90.8%)
	7 detection points	148 (93.1%)	106 (89.8%)	87 (87.8%)	135 (89.4%)
Missing	3 detection points	18 (11.1%)	14 (11.9%)	12 (11.8%)	14 (8.9%)
	5 detection points	9 (5.6%)	11 (10.0%)	8 (8.2%)	7 (4.6%)
	7 detection points	5 (3.1%)	8 (6.7%)	9 (9.1%)	9 (5.9%)
False	3 detection points	8 (5.1%)	3 (2.7%)	6 (5.9%)	12 (7.8%)
	5 detection points	6 (3.7%)	3 (2.5%)	2 (2.1%)	8 (5.1%)
	7 detection points	6 (3.7%)	4 (3.4%)	3 (3.1%)	7 (4.6%)

5. Concluding Remarks

In this study, a color image vehicular detection system incorporating with fuzzy neural network is developed. Three-, five- and seven-points pseudo line detectors are placed on a single traffic lane to compare the detection accuracy in various conditions. The conclusions of this study are summarized as follows.

In the freeway, the detection performance in the daytime is better than that in the nighttime. While in urban streets, expect for three detection points, daytime detection performance is better than that of nighttime. Generally, the detection performances for five and seven points are better than those for three points in various lighting conditions. The difference of detection accuracy between five and seven detection points is small.

The main reason for detection failure in the daytime is due to resemblance of the color pixels between vehicles and road backgrounds. The second reason is the lane-changing vehicles, which is not counted in this detection system. In the nighttime, "missing" often occurs in upstream viewing while "false" often happens in downstream viewing. The high missing rate in the nighttime upstream viewing is due to high threshold value set in the FNN system. The high false rate in nighttime downstream viewing, on the other hand, is due to low threshold value.

This paper has dealt with single lane traffic detection. Of course, the FNN CIVD system can also detect the traffic scenes by placing a pseudo line detector on the monitor across the whole road section with multiple lanes. For multiple-lanes traffic detection, one must define a rule to assign any lane-changing vehicle to a specific lane to avoid missing or double counting. In future works, more traffic parameters such as vehicle length (classification), headways and speeds can be measured by allocating tandem pseudo line detectors. Last but not the least, this paper has constructed four-layers fuzzy neural network systems and tested three-, five- and seven-points of pseudo line detectors. Different types of FNN structures may further be explored and more detection points can also be experimented.

Acknowledgements

This paper is part of the research granted by the National Science of Council, R.O.C. (NSC-89-2211-E-009-082)

References

1. Blosseville, J.M., Krafft, C., Lenoir, F., Motyka, V. and Beucher, S., "TITAN: a Traffic Measurement System Using Image Processing Techniques," *Second International Conference on Road Traffic Monitoring*, pp.84-88, 1989.

2. Chiou, Y. C. and Lan, L. W., "Development and Application of Learning Algorithm for Fuzzy Neural Networks: A Case of Pavement Diagnosis," *Transportation Planning Journal*, Vol.26, No.2, pp.233-252, June 1997.

3. Dickinson, K. W. and Waterfall, R. C., "Image Processing Applied to Traffic, 1 A General Review," *Traffic Engineering & Control*, Vol. 25, No.1, pp.6-13, Feb. 1984a.

4. Dickinson, K. W. and Waterfall, R. C., "Image Processing Applied to Traffic: 2 Practical Experiences," *Traffic Engineering & Control*, Vol. 25, No.1, pp.60-67, Feb. 1984b.

5. Dickinson, K. W. and Wan, C. L., "Road Traffic Monitoring Using the TRIP II System," *Second International Conference on Road Traffic Monitoring*, pp.56-60, 1989.

6. Fathy, M. and Siyal, Y. M., "An Image Detection Technique Based on Morphological Edge," *Pattern Recognition Letters,* Vol.16, Issue 12, pp.1321-1330, 1995.

7. Hoose, N., "IMPACT: an Image Analysis Tool for Motorway Analysis and Surveillançe," *Traffic Engineering & Control*, Mar., 1992

8. Michalopoulos, P. G., "Vehicle Detection Video Through Image Processing: The Autoscope System," *IEEE Transactions on Vehicular Technology*, Vol.40, No.1, pp. 21-29, Feb. 1991.

9. Washburn, S. S. and Nihan, N. L., "Estimating Link Travel Time with the Mobilizer Video Image Tracking System," *Journal of Transportation Engineering*, pp. 15-20, Jan., 1999.