ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

ADAPTIVE TRAFFIC CONTROL SYSTEM USING HEURISTIC APPROACH

Sharanya C. and Muthiah M. A.
Department of Electronics and Communication Engineering, Sathyabama University, Chennai, India

ABSTRACT

Traffic which is the main backbone of a city plays an important role in social stability and development. The traffic signals in our country are standalone systems where the duration of the signal is determined by a traffic officer or a constant value of signal duration is pre-programmed. Sometimes this timing is too short in high traffic density and too long for lesser number of vehicles. These lead to economic losses and unnecessary delays. Therefore we go for Adaptive Traffic Control (ATC) systems which change the signal duration continuously based on the changing arrival pattern of the vehicles. In this paper we design an ATC using heuristic approach where the system is trained to many images with various possibilities of vehicles using Probabilistic Neural Network (PNN). Before training each image is processed using background subtraction with morphological filtering and connected component analysis to detect the number of vehicles. The real time input image from the camera is forwarded to the system which uses heuristic technique to match the current image with one of the trained images and displays the duration of the traffic signal based on the counted number of vehicles thereby regulating traffic easily without any congestion.

Keywords: adaptive traffic control, heuristic approach, probabilistic neural network, morphological filtering, connected component analysis.

1. INTRODUCTION

Adaptive traffic control (ATC) systems adjust signal timings according to the actual traffic demand. The use of ATCs improves travel time reliability, reduces congestion and prolongs the effectiveness of traffic signal timing. They are operationally demanding and need to be controlled and supervised by skilled engineering staff. ATCs reduce labour needed for developing the signal duration thereby maximizing throughput and delivering network control easily. They are also easy to expand, maintain, upgrade and coordinate with other transportation systems. There are many other Adaptive Traffic Control systems such as OPAC (Optimized Policies for Adaptive Control) [1], SCOOT (Split Cycle Offset Optimization Technique) [2], SCATS (Sydney Coordinated Adaptive Traffic System) [3] and RHODES (Real time Hierarchical Optimizing Distributed Effective System) [4] used by several countries. But they are quite complex and expensive. Hence our aim is to design an ATC which is easy to operate and cost effective with good efficiency. The ATCs can be accomplished by the combined use of hardware and software. The ATC which we use in this paper consists of two main parts. The camera which is the input sensor acts as “eyes” of the system and the processor which determines the number of vehicles acts as the “brain” of the system.

2. PROPOSED SYSTEM

The schematic representation of the proposed system is shown in Figure-1. The camera gives the interconnected video image of the vehicles standing in the traffic signal. The input from the camera is given to the system which is already trained with many sample images having various possibilities of vehicles in the corresponding lane.

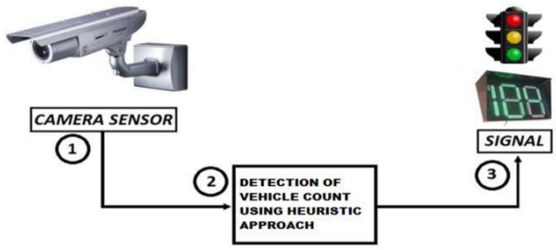

Figure-1. Schematic diagram of the Adaptive Traffic Signal Control System.

This training is based on heuristic approach using probabilistic neural network (PNN). And the images used for training are preprocessed using filtering and connected component analysis. The system takes the current image as its input and decides the number of vehicles in it based upon the training. Thus the timing duration of green signal is allotted according to the detected number of vehicles thereby reducing congestion and other delays.

3. SOFTWARE IMPLEMENTATION AND DESCRIPTION

A. Image smoothing

Smoothing is often used to reduce noise in an image or to make a less pixilated image. Smoothing of an image takes place when the disparity between pixel values is decreased by averaging the neighborhood pixels. The use of low pass filter retains the low frequency information of an image and reduces the high frequency information.

VOL. 10, NO. 9, MAY 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

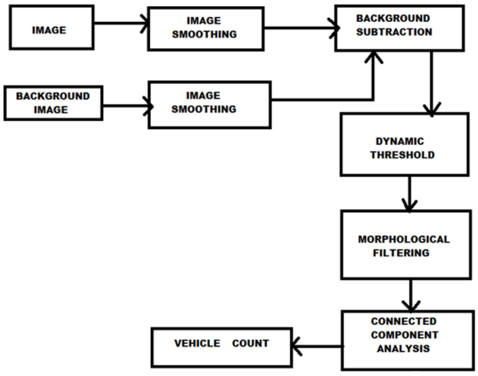

Figure-2. Block diagram involving image processing.

B. Background subtraction

Background subtraction is a technique in image processing where an image's foreground is extracted for further processing. It is mostly used for detecting moving objects in videos from fixed cameras. Background subtraction applies the logic of detection of the moving objects by determining the difference of the current frame from the reference frame (background image). It is based on static background hypothesis [5].

C. Connected component analysis

Connected Component Analysis is a application where subsets of connected components are uniquely labelled based on a given Heuristic. After detecting the region boundaries in the image, we extract the regions which are not seperated by boundaries. The set of pixels which are not separated by a boundary are said to be connected [6, 7]. Each maximal region of connected pixels is called a connected component. For a connected set: A region $R \subset S$ is said to be connected under c (s) if for all s, $r \in R$ there exists a sequence of M pixels, s1, s2, s3 •••, sM such that $s1 \in c(s)$, $s2 \in c(s1)$,•••, $sM \in c(sM -1)$, $r \in c(sM)$ i.e. there is a connected path from s to r. Connected component labeling functions by scanning the image, pixel-by-pixel (from left to right and top to bottom) to identify connected pixel regions, i.e. pixels which have the same set of Intensity Values (IV). Connected component labeling can be implemented on binary or graylevel images (For a binary image IV={1}; however, in a graylevel image IV takes a range of values). Though different measures of connectivity are possible, in this paper we consider 8-connectivity.

D. Morphological filtering

Morphological image processing is a gather of non-linear operations related to the shape or features in an image. Relative ordering of pixel values are of main concern in morphological operators. They are independent of their numerical values and are therefore specially suited for processing of binary images [8,9]. Morphological techniques examine the image with a structuring element which can be a small shape or a template. The structuring element is arranged at many possible locations of the image and compared with the pixels of its neighbourhood. The new binary image has a successful test if the pixel has a non zero value at the location in the input image.

E. Dynamic thresholding

In dynamic thresholding, threshold value for binarizing image is not constant but is dynamic. It is calculated to be independent from changes in foreground darkness and background lightness and in the illumination level. By automatically adjusting contrast levels, Dynamic Thresholding improves the accuracy of low-contrast objects [10].

4. ALGORITHM/TECHNIQUE

A. Heuristic approach

Heuristic refers to experience based technique for problem solving, learning and discovery to find a solution which may not guarantee to be the best, but satisfactory or good enough for a given set of goals. Heuristic approaches are used to fasten up the method of finding a satisfactory solution through mental shortcuts to ease the load of making a decision. Heuristic approaches can be implemented in neural or fuzzy networks. In this paper we execute heuristic technique using Probabilistic Neural Network (PNN).

1). Probabilistic neural network (PNN)

A PNN is predominantly a classifier which maps any input pattern to a number of classifications and can be forced into a more general function approximator. The architecture of PNN is as shown shown in Figure 3. It is standarized into a multilayered feedforward network with fourlayers i.e. Input layer, Hidden layer, Summation/Pattern layer and Output layer. The training set must be thoroughly a representative of the actual population for effective classification. As the training set increases in size, the PNN deeply converges to the Bayes optimal classifier. Adding and removing training samples will simply involve adding or removing neurons in the pattern layer [11, 12, 13]. The architecture of PNN is as shown:

a) **Input layer:** This layer consists of one neuron for each predictor variable. If there are categorical variables, M-1 neurons are used where M denotes the number of categories. The range of the values are normalized by the input neurons by deducting the median and then dividing it by the interquartile range. It then progress these values to the neurons in the hidden layer.

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

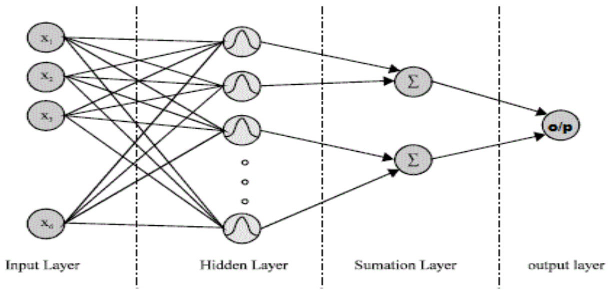

Figure-3. Architecture of Probabilistic Neural Network (PNN).

b) **Hidden layer:** This layer has one neuron for every case in the training data set. Each neuron stores the predictor variable values along with the target value. When the input layer presents the x vector of input values, the hidden neuron calculates the Euclidean distance of the sample case from the centre point of the neuron. Then the RBF kernel function is implemented using the sigma value(s). The emerging value is passed to the pattern layer neurons.

c) **Summation layer/pattern layer:** For each category of target variable there is one pattern neuron corresponding to it in this layer. Each hidden neuron stores the target category of each training case. And the weighted value which comes out of a hidden neuron is fed to the pattern neuron which corresponds to the hidden neuron's category only. The represented values of the classes are added by the pattern neurons and therefore it gives a weighted vote for that particular category.

d) **Decision layer:** For each target category, the decision layer compares the weighted votes that is accumulated in the pattern layer. It then uses the largest vote to predict the final target category.

5. HARDWARE IMPLEMENTATION AND DESCRIPTION

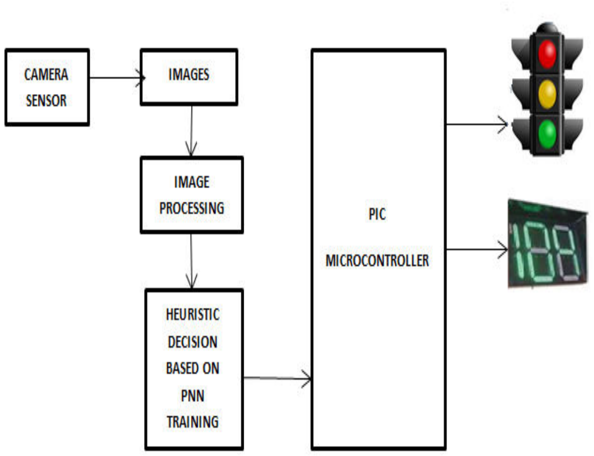

Figure-4. Block diagram of hardware implementation.

The image from the camera sensor is processed and compared with the trained images based on which the number of vehicles are estimated. Depending upon the vehicle count, the timing duration is calculated. This calculated duration is sent to the PIC controller and displayed in the LCD/LED. Initially the red light glows until the image is processed. After the signal duration is determined by the system, PIC is programmed in such a way that the yellow light glows for five seconds before the green light starts. The green light is then switched on with the time (in seconds) being displayed in the LED. Then the timing duration decreases to zero after counting the estimated seconds until which the green light is on. After reaching zero, the system resets and gets the next input image. This block diagram of this hardware implementation is shown in Figure-4.

A. Controller board

PIC microcontrollers have a 8-bit data memory bus and a 12, 14 or 16 bit program memory buslength relying on their family. They have different combinations of on-chip peripherals like A/D converters, Comparators, weak pull-ups, PWM modules, UARTs, Timers, SPI, I2C, USB, LCD, CAN etc. PIC microcontrollers come in many sizes varying from 6-pin (world's smallest microcontroller) to the high pin counts. But from designers perspective the 16F series of PICs are found to be the most ideal. PIC contollers are also foundto be faster and cost effective with good performance. The PIC specification used in this paper is PIC 16F877A which is a 40 pin, 4MHz, 8KB microcontroller with 1000,000 erase/write cycle enhanced flash program, 3 timers/counters, 35 single word instructions and a wide operating range from 2.0 to 5.5V.

B. Traffic signal board

The red, yellow and green LEDs act as the signal lights while the timing duration is displayed in the LCD.

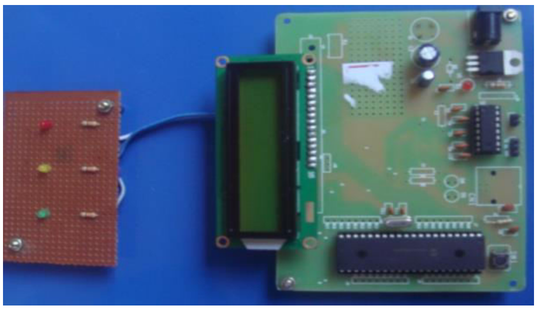

Figure-5. Controller and Traffic signal board.

The snapshot of the hardware used is shown in Figure-5. This system was implemented for various lanes and the counting accuracy was found to be considerably good.

6. RESULTS AND DISCUSSIONS

The designed system was implemented for images in Figures 6 and 7. The number of vehicles in the

ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

image were detected based on the trained samples with accuracy levels as shown in Figure-8. The accuracy level is also dependent upon climate and camera position.

Figure-6. Captured image of image1.avi and image2.avi.

Figure-7. Captured image of image3.avi and image4.avi.

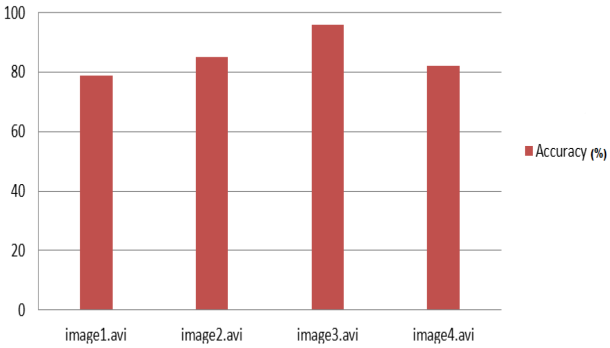

Figure-8. Graph indicating the vehicle counting accuracy.

7. CONCLUSIONS

In this paper, congestion issues are addressed by identifying and deploying innovative technology that reduces delay to the motoring public which result in improved fuel efficiency and lower vehicle emissions which have positive impact on the environment. The designed ATC uses heuristic approach with PNN training to make decisions on deciding the number of vehicles in the lane and calculating the timing duration based on it. This design is found to maximise throughput and deliver network control effortlessly. It is also easy to maintain, expand, upgrade and coordinate with other transportations in their respective regions. Heuristic approach and its training used in this method gives an accuracy of about 95% under normal traffic conditions. It can be improved by increasing the number of training samples. However the accuracy also depends on the weather conditions and location of the camera.

REFERENCES

[1] Nathan H. Gartner, Farhad J. Pooran and Christina M. Andrews, "Implementation of the OPAC adaptive control strategy in a traffic signal network," IEEE intelligent Transportation Systems Conference Proceedings - Oakland (CA), USA - August 25-29, 2001.

[2] Bretherton , M. Bodger and N. Baber, "SCOOT - the future [urban traffic control]," Road Transport Information and Control, 2004.

[3] The SCATS Adaptive Traffic System. [Online]. Available: http://ocw.nctu.edu.tw/course/sc011/2012-08-21.pdf

[4] Pitu Mirchandani and Fei-YueWang, "RHODES to Intelligent Transportation Systems," Intelligent Systems, IEEE, 2005.

[5] T. Intachak and W. Kaewapichai, "Real time illumination feedback system for adaptive background subtraction working in traffic video monitoring," Intelligent Signal Processing And Communication System (ISPACS), 2011.

[6] Mozef, S. Weber, J. Jaber and E. Tisserand, "Parallel architecture dedicated to connected component analysis," International Conference on Pattern Recognition, 1996.

[7] S. Palaniappan and S. Natarajan, " Parallel realization of single pass connacted component analysis on a multicore architecture," International Conference in Communication And Signal Processing (ICCSP), 2014.

[8] J. Crespo and V. Maoio, "Shape preservation in morphological filtering and segmentation," Proceedings XII Brazilian synposium on Computer Graphics and Image processing, 1999.

[9] Guan Ming Shan and Ren Honge, "Multiscale morphological filtering for preserving the details of images," Asia Pacific Conference on Computational Intelligence and Industrial Applications, 2009.

[10] Jian Chao Cao, Qiu Hua Lin and Meng Yin Yu, "Extraction of single-pixel-width single edge using dynamic thresholding for lane detection in structured roads," IEEE China summit, 2014.

[11] Srinivasan, Xin Jin and R. L. Chen, " Evaluation of adaptive neural network models for freeway incident

detection," IEEE transactions on Intelligent Transportation Systems, 2004.

[12] Runyuan Sun and Bo Yang, "Traffic classification using Probabilistic Neural Networks," Sixth International Conference on Natural Computation (ICNC), 2010.

[13] Abdulhai, H. Porwal, and W. W. Recker, "Short-term traffic flow prediction using neuro-genetic algorithms," ITS J., vol. 7, no. 1, pp. 3-41, January 2002.

[14] Wisnu Jatmiko, *et.al*, "Detection and Counting of Vehicles Based on Video Processing In Distributed Traffic System", International Conference on Advance Computer Science and Information System 2010.

[15] W. D. Cottrell, "Estimating the probability of freeway congestion recurrence," in Proc. Transp. Res. Rec., Washington, DC, 1998, vol. 1634, pp. 19-27, Transp. Res. Board, Nat. Res. Council.

[16] S. Ishak and C. Alecsandru, "Optimizing traffic prediction performance of neural networks under various topological, input, and traffic condition setting," J. Transp. Eng., vol. 130, no. 4, pp. 452-465, July/August 2004.

[17] R. Kuhne, R. Mahnke, I. Lubashevsky, and J. Kaupus, "Probabilistic description of traffic breakdowns," J. Phys. Rev. E, Stat., Nonlin., Soft Matter Phys., vol. 65, no. 6, pp. 066125-1–066125-13, June 2002.

[18] L. Shen and M. Hadi, "Freeway travel time prediction with dynamic neural networks," presented at the Proc. Transp. Res. Board Annu. Meeting, Washington, DC, 2010, Paper 10-3105.

[19] P. Mirchandani and L. Head, "A real-time traffic signal control system: Architecture, algorithms, and analysis," Transport. Res. C, vol. 9, no. 6, pp. 415–432, 2001.

[20] Bingham, "Reinforcement learning in neurofuzzy traffic signal control,"Eur. J. Oper. Res., vol. 131, pp. 232-241, 2002.

[21] L. Smith and M. J. Demetsky, "Short-term traffic flow prediction: Neural network approach," in Proc. Transp. Res. Rec., Transp. Res. Board, 1994, vol. 1453, pp. 98-104.

[22] Srinivasan, D., Choy, M. C., and Cheu, R. L, " Neural Networks for Real- Time Traffic Signal Control," IEEE Transactions On Intelligent Transportation Systems Vol. 7, No.3 261-272. 2006.

[23] Traffic Engineering Handbook, 5th ed. Washington, D.C.: Inst. Transp. Eng., 1999.

[24] I Vlahogianni, J. C. Golias, and M. G. Karlaftis, "Short-term traffic forecasting: Overview of objectives and methods," Transp. Rev., vol. 24, no. 5, pp. 533-557, September 2004.

[25] R. Cucchiara, M. Piccardi, and P. Mello, "Image analysis and rule-based reasoning for a traffic monitoring system," IEEE Trans. Intell. Transp. Syst., vol. 1, no. 2, pp. 119-130, Jun. 2000.