Journal of Universal Computer Science, vol. 15, no. 5 (2009), 1119-1142
submitted: 1/8/08, accepted: 28/2/09, appeared: 1/3/09 © J.UCS

A Neural Network Based Vehicle Classification System for Pervasive Smart Road Security

Naixue Xiong, Jing He[1,2]
([1]Dept. of Computer Science, Georgia State University, Atlanta, Georgia, USA
nxiong@cs.gsu.edu, jinghe@cc.usu.edu)

Jong Hyuk Park
(Department of Computer Science and Engineering, Kyungnam University, Korea
jhpark1@kyungnam.ac.kr)

Donald Cooley
([2]Dept. of Computer Science, Utah State University, Logan, Utah, USA
cooley@don.cs.usu.edu)

Yingshu Li
(Dept. of Computer Science, Georgia State University, Atlanta, Georgia, USA
yli@cs.gsu.edu)

Abstract: Pervasive smart computing environments make people get accustomed to convenient and secure services. The overall goal of this research is to classify vehicles along the I215 freeway in Salt Lake City, USA. This information will be used to predict future roadway needs and the expected life of a roadway. The classification of vehicles will be performed by a synthesis of multiple sets of features. All feature sets have not yet been determined; however, one such set will be the reduced wavelet transform of the image of a vehicle. In order to use such a feature, it is necessary that the image be normalized with respect to size, position, and so on. For example, a car in the right most lane in an image will appear smaller than one in the left most lane, because the right most lane is closest to the camera. Likewise, a vehicle's size will vary depending on where in a lane its image is captured. In our case, the image capture area for each lane is approximately 100 feet of roadway. A goal of this paper is to normalize the image of a vehicle so that regardless of its lane or position in a lane, the features will be approximately the same. The wavelet transform itself will not be used directly for recognition. Instead, it will be input to a neural network and the output of the neural network will be one element of the feature set used for recognition.

Keywords: Neural Network, Wavelet, Wavelet transform, Vehicle classification, Recognition, Normalization, image processing
Categories: H.5.1, I.2.0, I.2.11, I.2.8, I.2.6, L.2.0

1 Introduction

Pervasive smart computing environments make people get accustomed to convenient and secure services. Traffic management and smart information systems, for pervasive smart road security, depend on a lot of sensors to estimate traffic parameters. Recently, magnetic loop detectors are usually used to compute the number, which is for vehicles passing over them. Vision-based video monitoring systems offer a number of advantages. Besides vehicle counts, a lot of traffic parameters (such as vehicle classifications, lane changes, and so on) can be measured, and cameras are much less disadvantaged to install than magnetic detectors.

It is important for vehicle detection and classification in the computation of the percentages of vehicle classes, which are used in streets and highways. The current situation is described by outdated data, and it is human operator to manually count vehicles at a specific street. Thus, one uses an automated system to obtain perfect design of pavements (e.g., the decision about thickness and width) with obvious results in cost and quality. Even in metro areas, there is a requirement for data about vehicle classes, which use a particular street. A classification system, like the above one, can provide important data for a particular design scenario [Gupte, 00].

In this paper, we use a single camera system mounted on a pole to look down on the traffic scene, and detecting and classifying vehicles in multiple lanes are for any direction of traffic flow. Except for the camera parameters and direction of traffic, it requires no other initialization.

The rest of this paper is organized as follows. Section 2 gives an overview of related work, In Section 3, we talk about the relevant theory on Neural Network. Section 4 gives our theory analysis for the classification system based on wavelet and wavelet transform. In Section 5, we present our algorithm description for the implement of the vehicle classification system. Finally, we give the results and relevant discussion on our vehicle classification system in Section 6, and present the conclusion in Section 7.

2 Related Work

There has been a lot of literature on vehicle detection and classification, and vehicle classification is an inherently hard problem.

Neural networks have gotten successful applications in various fields from pattern recognition to road security diagnosis due to their strong capacity to handle formidable problems and to improve system performance, including vehicle detection and classification. A lot of papers used Neural Networks to solve relevant problems or optimize relevant performance [Chen, 2008; Zuo, 2008; Guarneri, 2008; De, 2008]. In [Chen, 2008], two types of uncertain delays were considered because the asymptotic stability of neural networks is uncertain delays. By combining the discretized procedure on Lyapunov–Krasovskii functional method and the free-weighting matrix technique, the authors developed a new discretized method for analyzing stability of delayed neural networks. The integrated scheme leads to the establishment of novel delay-dependent sufficient conditions in form of linear matrix inequalities for asymptotic stability of delayed neural networks [Chen, 2008], but the results in recent papers [Park, 2006; Ensari, 2005a; Ensari, 2005b] are not applicable.

In [Zuo, 2008], the authors proposed an adaptive Fourier neural network control scheme for controlling a class of uncertain nonlinear systems. Combining Fourier analysis and neural network theory, this scheme used orthogonal complex Fourier exponentials as the activation functions. One obvious feature in this approach is that all the nonlinearities and uncertainties of the dynamical system are focused on and compensated online by this scheme. So it could be applied to uncertain nonlinear systems without any a priori information about the system dynamics, and actually, it is a frequency domain scheme, which can assure asymptotic stability of the closed-loop system. This scheme should improve the robustness because a network with orthogonal activation functions sometimes is more sensitive to weight disturbances than conventional neural networks does.

Many vehicle classification schemes for traffic are presented recently [Ma, 05; Chen, 07; Mohottala, 03; Yoshida, 02; Urazghildiiev, 07; Morris, 06; Duarte, 04]. In [Ma, 05], the authors presented an approach on vehicle classification under a midfield surveillance framework for traffic, explored a repeatable and discriminative feature using edge points, and improved Scale Invariant Feature Transform descriptors. After that, a rich representation for vehicle classification was introduced. While vehicle classification with view changes and occlusion is still to be investigated, more experiments on more vehicle types and vehicle identity recognition should be done. This kind work has been done in our work in next sections. In [Chen, 07], the computer vision is explored based on vehicle classification at a nice granularity, and then the authors gave a framework, which included various aspects of an Intelligent Transportation System and vehicle classification. Furthermore, they implemented an intelligent scheme to identify the features of each vehicle, and used One-class SVM to categorize every vehicle into a certain class.

This paper [Lipton, 98] presented a vehicle tracking and classification system, and it could categorize moving objects between vehicles and humans. However, how to further classify the vehicles into various classes? This paper did not solve it. In [Koller, 93], an object classification scheme, using parameterized 3-D models, was presented. By a 3-D polyhedral model, the system could classify vehicles in a traffic sequence. While they assume that cars are more common than other types of vehicles in typical traffic scenes. The extensive work in classification of the tracked vehicles based on 3-D model matching methods was done by the University of Reading. Baker and Sullivan [Baker, 92] and Sullivan [Sullivan, 92] used the knowledge of the camera calibration and the information that vehicles move on a plane in their 3-D model-based tracking, then they developed three-dimensional wireframe models of various types of vehicles, such as sedans, hatchbacks, wagons, and so on. In [Sullivan, 95], the above scheme was optimized so that the image features act as the forces on the model. This decreased the number of iterations and optimized performance. Later, Sullivan et al. [Sullivan, 97] developed a simplified version of the model-based tracking scheme by orthographic approximations to get real-time performance [Gupte, 00].

In this paper, we used NeuralSIM to train the neural network whose input is the output of the wavelet transform. NeuralSIM is a neural network training package designed by aspen software. It allows for rapid testing of neural networks without having to actually write code for individual neutrons, etc.

3 Neural Network

3.1 Introduction

Neural networks consist of a number of processing elements called neutrons. It is a complicated, non-linear, dynamic system that the neutrons connect each other in some topological structure. It is drawn from research on organization structure and behavior features of the human brain and mimicking the simplified information processing capabilities of biological neural systems. Depending on their structure, neural networks can be categorized into multi-layer feed forward (MLF) networks, self-organization feature map (SOFM), and adaptive resonance theory (ART). The MLF network is the most commonly used network. In this paper, a three layer MLF with sigmoid transfer functions was used.

3.2 Network Architecture

In this project we build a multilayer feed forward neural network model with one hidden layer (Figure 1). There are 256, 19, and 5 neutrons in the input, hidden, and output layer respectively. The number of processing elements in the input layer corresponds to the number of features obtained in each traffic image. The output nodes represent the vehicle types or categories to recognize, i.e., there were 5 categories or classes. To take into account interrelations among the processing elements, a full connection architecture is used. One bias unit, whose input value is 1, is included in each layer in the network. The transfer function for all elements was the sigmoid function.

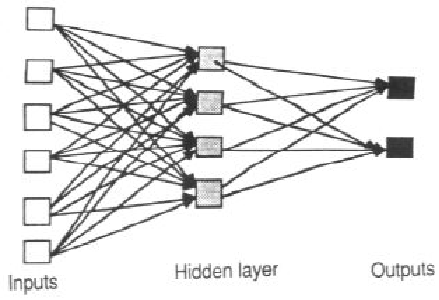


Figure 1: An example of a simple feedforward network

3.3 Backpropagation Learning Rule

The neural network learning rule used in this experiment was backpropagation [Martin, 95]. The following notation is used to represent formulations of the backpropagation learning processes (bolded symbols represent vectors):

$\boldsymbol{P}$ = input vector
$\boldsymbol{t}$ = target output
$\boldsymbol{a}^1$ = actual output of input layer
$\boldsymbol{a}^2$ = actual output of hidden layer
$\boldsymbol{a}^3$ = actual output of output layer
$\boldsymbol{b}^1$ = bias of input layer
$\boldsymbol{b}^2$ = bias of hidden layer
$\boldsymbol{b}^3$ = bias of output layer
$\boldsymbol{W}^1$ = weight vector of input layer
$\boldsymbol{W}^2$ = weight vector of hidden layer
$\boldsymbol{W}^3$ = weight vector of output layer
α = learning rate

As noted in 1.3.1, the transfer function is a sigmoid function, with backpropagation-based learning conducted via the following steps:

1. Compute the actual output of the network:

$$a^1 = \frac{1}{1+e^{-(W^1 P + b^1)}} \text{ (sigmoid function)} \tag{3.1}$$

$$a^2 = \frac{1}{1+e^{-(W^2 a^1 + b^2)}} \tag{3.2}$$

$$a^3 = \frac{1}{1+e^{-(W^3 a^2 + b^3)}} \tag{3.3}$$

2. Compute the error of the output:

$$\delta = t - a^3 \tag{3.4}$$

3. Compute the weight adjustment ΔW^m (m=1, 2, 3) according to the backpropagation rule.
4. Modify the weights:

$$\boldsymbol{W}^m(n+1) = \boldsymbol{W}^m(n) + \Delta \boldsymbol{W}^m + \alpha\,[\boldsymbol{W}^m(n) - \boldsymbol{W}^m(n-1)] \tag{3.5}$$

5. If all training samples have not been inputted repeat steps 1 through 4.
6. Compute the value of the error function ε.

$$\varepsilon = \sqrt{\frac{\sum_j (t\text{-}a^3)^2}{j}} \text{ (RMS of individual errors)} \tag{3.6}$$

7. Repeat steps 1 through 6 until ε (the error value) satisfies some desired condition.

4 Wavelet and Wavelet Transform

4.1 Comparison of Fourier Transform and Wavelet Transform

A wave is usually defined as an oscillating function of time and space, such as a sinusoid. Fourier analysis is wave analysis. It expands signals or functions in terms of sinusoids (or, equivalently, complex exponentials), which has proven to be extremely valuable in mathematics, science, and engineering, especially for periodic, time-invariant, or stationary phenomena. A wavelet is a "small wave" which has its energy concentrated in time. It is a tool for the analysis of transient, non-stationary, or time-varying phenomena. It still has an oscillating, wave-like characteristic but also has the ability to allow simultaneous time and frequency analysis with a flexible mathematical foundation.

Fourier analysis has been used for frequency domain representation of signals for many years. Wavelet theory is a general mathematical tool primarily used for representing signals including image and video data more efficiently. The difference between Fourier transforms and wavelet transforms is the selection of the basis functions. The basis functions of a Fourier transform are sinusoidal waves. For wavelet transforms, wavelets are the basis functions. Figure 2 shows the basis functions of Fourier and wavelet transforms. In the Fourier analysis of a set of data points, each Fourier coefficient is computed by using the entire data set with exponential weighting. Thus, Fourier analysis provides "global" properties of the data. The wavelet transform has a wavelet (also called a mother wavelet) replacing the exponential. Scaling and translation replace frequency shifting. A two-dimensional surface of wavelet coefficients replaces the one dimensional Fourier coefficients. The "local" properties of the data can be obtained by a proper choice of the mother wavelet [Chen, 00].

Wavelet theory is much more general than Fourier theory and it provides a better representation of the data. For image data, wavelet analysis can decompose the image into different resolution levels and at each level different frequency sub-bands can provide different texture and edge information making it very convenient for image compression and feature extraction.

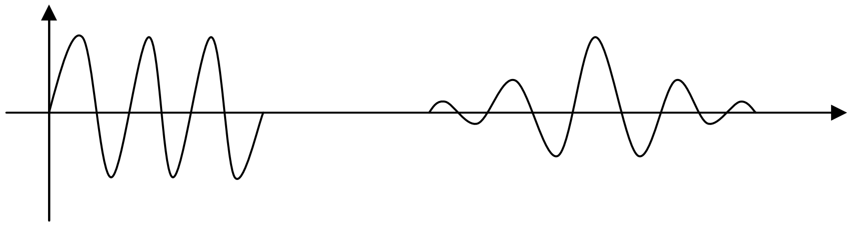


Figure 2: Typical basic function of Fourier transform and wavelet

There are two ways to implement wavelet transforms: the fast wavelet transform and the lifting scheme [Chen, 00]. The fast wavelet approach is a convolution approach as it performs convolution between the original signal and the mother wavelet. The lifting scheme is an interpolation approach that performs sub-sampling and interpolation.

4.2 Wavelets and Wavelet Expansion Systems

4.2.1 What is a Wavelet Expansion or a Wavelet Transform?

A signal or function f (t) can often be better analyzed, described, or processed if expressed as a linear decomposition by

$$f(t) = \sum_l a_l \varphi_l(t) \tag{4.1}$$

where l is an integer index for the finite sum, a_l are the real-valued expansion coefficients, and $\varphi_l(t)$ are a set of real-valued functions of t called the expansion set. If the expansion (4.1) is unique, the set is called a basis for the class of functions that can be so expressed. If the basis is orthogonal, meaning

$$\langle \varphi_k(t), \varphi_l(t) \rangle = \int \varphi_k(t)\varphi_l(t)dt = 0, k \neq l, \tag{4.2}$$

then the coefficients can be calculated by the inner product

$$a_k = \langle f(t), \varphi_k(t) \rangle = \int f(t)\varphi_k(t)dt. \tag{4.3}$$

One can see that substituting (4.1) into (4.3) and using (4.2) gives the single a_k coefficient. If the basis set is not orthogonal, then a dual basis set $\bar{\varphi}_k(t)$ exists such that using (4.3) with the dual basis gives the desired coefficients.

For a Fourier series, the orthogonal basis functions $\varphi_k(t)$ are $sin(k\,\omega_0\, t)$ and $cos(k\,\omega_0\, t)$ with frequencies of $k\,\omega_0$. For a Taylor's series, the nonorthogonal basis functions are simple monomials t^k, and for many other expansions they are various polynomials. There are expansions that use spines and even fractals.

For the wavelet expansion, a two-parameter system is constructed such that (4.1) becomes

$$f(t) = \sum_k \sum_j a_{j,k}\varphi_{j,k}(t) \tag{4.4}$$

where both j and k are integer indices and the $\varphi_{j,k}(t)$ are the wavelet expansion functions that usually form an orthogonal basis.

The set of expansion coefficients $a_{j,k}$ are called the discrete wavelet transform (DWT) of $f(t)$ and (4.4) is the inverse transform.

4.2.2 What is Wavelet System?

The wavelet expansion set is not unique. There are many different wavelet systems that can be used effectively. They have the following three general characteristics.

- A wavelet system is a set of building blocks to construct or represent a signal or function. It is a two-dimensional expansion set (usually a basis) for some class of one- (or higher) dimensional signals. In other words, if the wavelet set is given by $\varphi_{j,k}(t)$ for indices of $j, k = 1, 2, \ldots$, a linear expansion would be $f(t) = \sum_k \sum_j a_{j,k}\varphi_{j,k}(t)$ for some set of coefficients $a_{j,k}$.
- The wavelet expansion gives a time-frequency localization of the signal. This means most of the energy of the signal is well represented by a few expansion coefficients, $a_{j,k}$.
- The calculation of the coefficients from the signal can be done efficiently. It turns out that many wavelet transforms (the set of expansion coefficients) can be calculated with $O(N)$ operations. This means the number of floating-point multiplications and additions increase linearly with the number of samples of the signal. More general wavelet transforms require $O(N\log(N))$ operations, the same as for the fast Fourier transform (FFT).

Virtually all wavelet systems have these general characteristics. Where the Fourier series maps a one-dimensional function of a continuous variable into a one-dimensional sequence of coefficients, the wavelet expansion maps it into a two-dimensional array of coefficients. It is this two-dimensional representation that allows localizing the signal in both time and frequency. A Fourier series expansion localizes in frequency in that if a Fourier series expansion of a signal has only one large coefficient, then the signal is essentially a single sinusoid at the frequency determined by the index of the coefficients. The simple time-domain representation of the signal itself gives the localization in time. If the signal is a simple pulse, the location of that pulse is the localization in time. A wavelet representation will give the location in both time and frequency simultaneously. Indeed, a wavelet representation is much like a musical score where the location of the notes tells when the tones occur and what their frequencies are.

4.2.3 Characteristics of Wavelet Systems

There are three additional characteristics that are more specific to wavelet expansions.

- All so-called first-generation wavelet systems are generated from a single scaling function or wavelet by simple scaling and translation. The two-dimensional parameterization is achieved from the function (sometimes called the generating wavelet or mother wavelet) $\varphi(t)$ by

$$\varphi_{j,k}(t) = 2^{j/2}\varphi(2^j t - k) \quad j, k \in Z \tag{4.5}$$

- where Z is the set of all integers and the factor $2^{j/2}$ maintains a constant norm independent of scale j. This parameterization of the time or space location by k and the frequency or scale (actually the logarithm of scale) by j turns out to be extraordinarily effective.
- Almost all useful wavelet systems also satisfy the multiresolution conditions. This means that if a set of signals can be represented by a weighted sum of $\varphi(t\text{-}k)$, then a larger set (including the original) can be represented by a weighted sum of $\varphi(2t\text{-}k)$. In other words, if the basic expansion signals are made half as wide and translated in steps half as wide, they will represent a larger class of signals exactly or give a better approximation of any signal.
- The lower resolution coefficients can be calculated from the higher resolution coefficients by a tree-structured algorithm called a filter bank [Burus, 98]. This allows a very efficient calculation of the expansion coefficients (also know as the discrete wavelet transform) and relates wavelet transforms to an older area in digital signal processing.

The operations of translation and scaling seem to be basic to many practical signals and signal-generating processes, and their use is one of the reasons that wavelets are efficient expansion functions. Figure 3 is a pictorial representation of the translation and scaling of a single mother wavelet described in (4.5). As the index k changes, the location of the wavelet moves along the horizontal axis. This allows the expansion to explicitly represent the location of events in time or space. As the index j changes, the shape of the wavelet changes in scale. This allows a representation of detail or resolution. Note that as the scale becomes finer (j larger), the steps in time become smaller. It is both the narrower wavelet and the smaller steps that allow representation of greater detail or higher resolution. For clarity, every fourth term in the translation ($k = 1, 5, 9, 13, \ldots$) is shown. What is not illustrated here, but is important, is that the shape of the basic mother wavelet can also be changed. That is done during the design of the wavelet system and allows one set to represent a class of signals.

For the Fourier series and transform and for most signal expansion systems, the expansion functions (bases) are chosen, then the properties of the resulting transform are derived and analyzed. For the wavelet system, these properties or characteristics are mathematically required, and then the resulting basis functions are derived.

Because these functions do not use all the degrees of freedom, other properties can be required to customize the wavelet system for a particular application. Once you decide on a Fourier series, the sinusoidal basis functions are completely set. That is not true for the wavelet. There is infinity of very different wavelets that all satisfy the above properties.

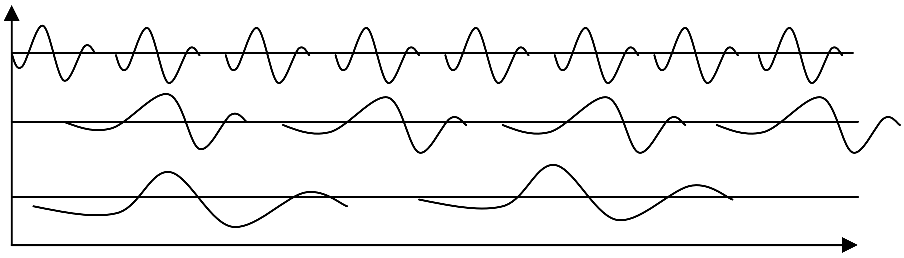


Figure 3: Translation (every fourth k) and Scaling of a Wavelet φ_{D4}

Wavelet analysis is well suited to transient signals. Fourier analysis is appropriate for periodic signals or for signals whose statistical characteristics do not change with time. It is the localizing property of wavelets that allow a wavelet expansion of a transient event to be modelled with a small number of coefficients. This turns out to be very useful in applications [Burus, 98].

4.3 The Haar Wavelet

The Haar wavelet is probably the simplest of all wavelets, and is also the oldest. It has been used in image analysis for many years as the Haar transform. We used this transform in this work. The Haar wavelet is a step function taking the values +1 and – 1. Specifically:

$$\varphi_H(t) = \begin{cases} 1, 0 < t < 1/2 \\ -1, 1/2 \le t < 1 \\ 0, Otherwise \end{cases} \tag{4.6}$$

A convenient (but not the only) way to scale these wavelets is by power of two. For example, the function φ_j (t) = φ_H $(2^j t)$ is scaled versions of φ_H (t). A wavelet is simply a function that, unlike the Fourier transform, not only has a frequency associated with it, but also a scale. Scaling is performed by dividing the argument by a scaling factor. The wavelet $g(\frac{x}{s})$ is simply $g(x)$ scaled by a factor s. In the above example the scale factor $s = \frac{1}{2^j}$. Translation is done by shifting the functions along the axis. A scaled and translated Haar wavelet can be described as

$$\varphi_{j,k}(\text{t}) = \varphi_H(2^j t - k) \tag{4.7}$$

A sample collection of these wavelets can be seen as graphs in Figure 4. They have the appearance of square waves, scaled and translated over a small range. A continuous function can be approximated by these Haar functions in a way similar to the use of sine and cosine functions in the Fourier series approximations. A linear combination of Haar functions is constructed, whose sum approximates the required function:

$$f(t) = \sum_{j=-\infty}^{\infty} A_j \sum_{k=-\infty}^{\infty} B_{j,k} \varphi_H (2^j t - k) \tag{4.8}$$

For any problem that is to be solved using a computer, the bounds of the sums are finite, which is acceptable for an approximation as long as the result has a small specified error bound. Indeed, for image processing and vision purposes, the function f actually consists of regularly sampled values as defined by the sampling grid.

What is called a wavelet transform with respect to the Haar basis is really the calculation of the values for A_j and $B_{j,k}$. Thus the formula for the wavelet transform can be found by [Rafael, 92]:

$$A_j B_{j,k} = 2_j \int_{2^{(-j)}k}^{2^{-j}(k+1)} f(t) \varphi_{j,k}(t) dt \tag{4.9}$$

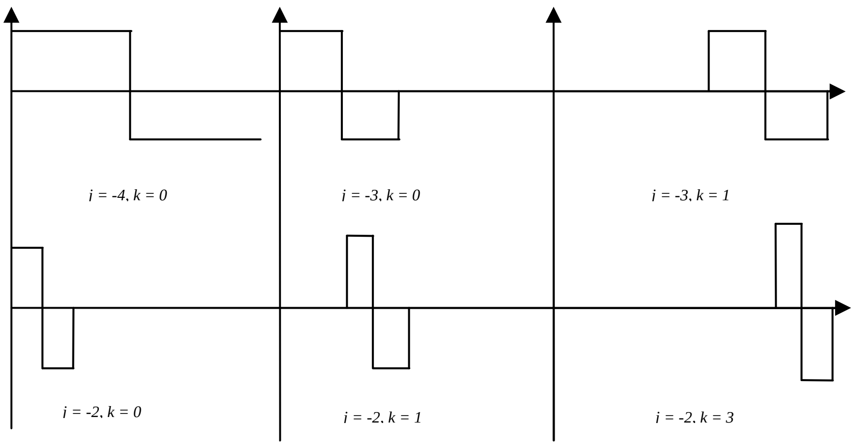


Figure 4: Haar wavelets of various scales and translations

4.4 Why We Used the Wavelet Transform to Extract Vehicle Features

4.4.1 Use wavelet transformation as image compression

The wavelet transformation can be used in image compression, so we can use the smallest size wavelet that maintains all necessary features (shape, length, height, and width) of each vehicle. The wavelet transform approach to image compression is better than the traditional Fourier based approach using the discrete cosine transforms. In practice, a discrete cosine transform (DCT) is generated based on small blocks of image pixels. Often the image block size of the DCT is 8X8 or 8X16. This blocking operation can degrade the image quality with blocking artifacts. Due to the quantization process that discards the high frequency components, the distortions in highly textured areas are much more significant than in homogeneous areas.

The wavelet transform can operate on a whole image with linearly increasing computation which means the computation time is linearly related to the image size. Without breaking the image into several blocks, wavelet-based approaches do not suffer from blocking artifacts. Many research works have shown that the wavelet transform has a better-reconstructed image quality than the DCT. The difference between the wavelet transform and the DCT is especially noticeable as the compression ratio increases [Chen, 00].

4.4.2 NeuralSIM to train the neural network whose input is the output of the wavelet transform.

NeuralSIM is a neural network training package designed by Aspen software. It allows for rapid testing of neural networks without having to actually write code for individual neutrons, etc.

5 Algorithm Description

In this chapter we describe the algorithm (the general process is shown in Figure 6) used to extract a vehicle image and ultimately give it a classification. The process begins by generating a background image (the process is shown in Figure 7). Subsequent captured frames are then compared to the background image (Figure 5). The difference between the captured image (Figure 8) and the background (if it is above a threshold) is a vehicle (Figure 9). Next we normalize the extracted vehicle to a given standard view (Figure 11) and then perform a wavelet transform on the gray-scale equivalent of the normalized vehicle (Figure 12 (b)). The output of the wavelet transform is the input to the neural network. The output of the neural network is the category of the vehicle.

5.1 Generating the background image

The process of actually generating a background image was a three-step process defined as follows:

(1) For each captured frame the pixels are grouped by color. Two colors are considered to be the same when they are in the same group although the intensities of these two colors maybe different. In this project, 32 color groups were used.

(2) Capture 100 images at a preset frequency. For this work, the capture rate was three images per second.

(3) The background image is generated based on the individual pixel values in these 100 images. For each pixel p with coordinate (x, y) find the color group that has the greatest number of occurrences. The color of the pixel p is the color of this group. A color group represents a range of colors. In our case the middle color in the range was used.

5.2 Extract vehicles from the images

A simple approach for detection of changes between two image frames $f(x, y, t_i)$ and $f(x, y, t_j)$ taken at times t_i and t_j, respectively, is to compare the two images pixel by pixel. One procedure for doing this is to form a difference image, i.e., an “image” in which each pixel value is the difference between corresponding pixels in frames t_i and t_j. Suppose that we have a reference image containing only stationary components (background image generated in step 2.1). Subtracting this image from a subsequent image that has a moving object results in a difference (pixels) image in which stationary components (pixels) are cancelled leaving only nonzero entries corresponding to the nonstationary image components.

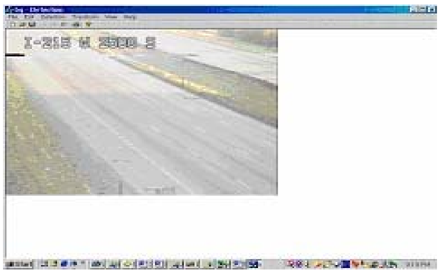


Figure 5: Background Image developed from 100 captured frames

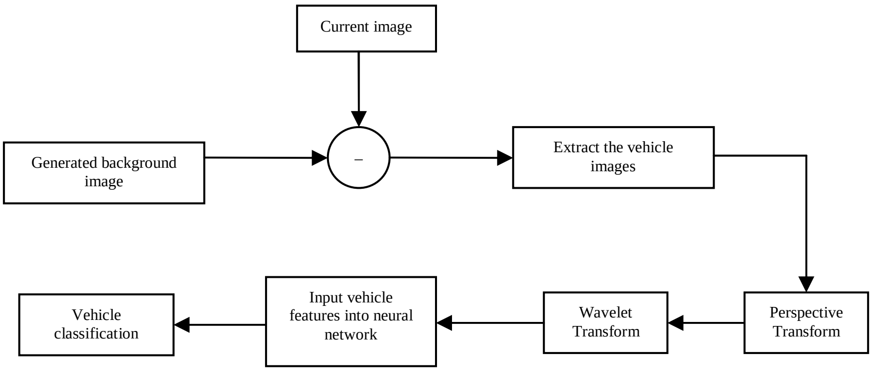


Figure 6: General process for vehicle classification

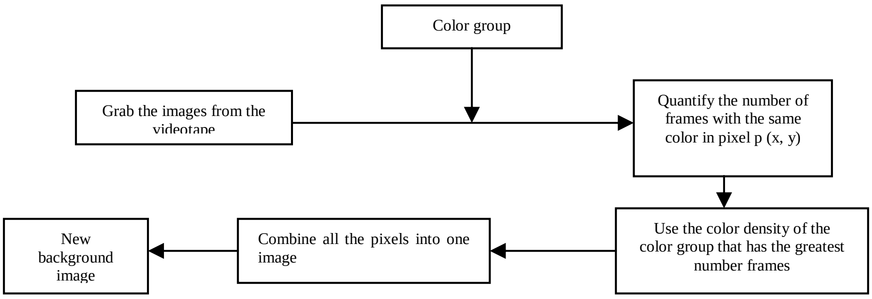


Figure 7: Generate the background image

A difference between two images taken at times t_i and t_j may be defined as

$$d_{i,j}(x,y) = \begin{cases} 1, if \mid f(x,y,t_i) - f(x,y,t_j) \mid > \theta \\ 0, Otherwise \end{cases} \quad (5.1)$$

where θ is a threshold.

Figure 8 is an image of a moving vehicle. Figure 9 is the difference between images 5 (background image) and 8.

Figure 8: Original Image (Moving Vehicle)

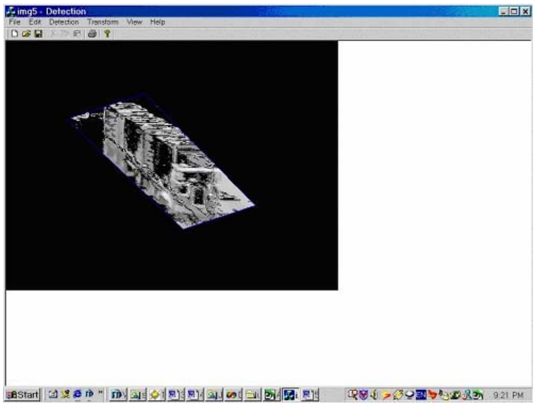


Figure 9: Vehicle extracted (difference image) from the original image

5.3 Perspective Transform

In order to normalize for features such as size, it was necessary to perform a perspective transformation on the vehicle image. This transformation generated a new "view" of the vehicle so that it was oriented and located at a pre-set location. In this way, the view is normalized (Figure 5).

A perspective transformation projects 3-D points onto a plane. Figure 10 shows a model of the image formation process. The camera coordinate system (*x, y*) has the image plane coincident with the *xy* plane and the optical axis (established by the center of the lens) along the *z*-axis. Thus the center of the image plane is at the origin, and the center of the lens is at coordinates ($0, 0, \lambda$). If the camera is in focus for distant objects, λ is the focal length of the lens. Here the assumption is that the camera coordinate system is aligned with the world coordinates system (*X, Y, Z*) [Sullivan, 95].

If we know an image point (x_0, y_0), then according the following formula, we can determine its world coordinates. (Assuming Y - the height, is a constant.)

$$\begin{cases} X = \dfrac{x_0 * (\lambda - Z)}{\lambda} \\ Y = \dfrac{y_0 * (\lambda - Z)}{\lambda} \end{cases} \Longrightarrow \begin{cases} X = \dfrac{x_0 * Y}{y_0} \\ Z = \lambda - \dfrac{\lambda * y}{y_0} \end{cases} \tag{5.2}$$

If we know the distance we want to move in the image (ΔX, ΔZ), according to the *X* and *Z*-axes, then we know the real world coordinate will be changed to the following point:

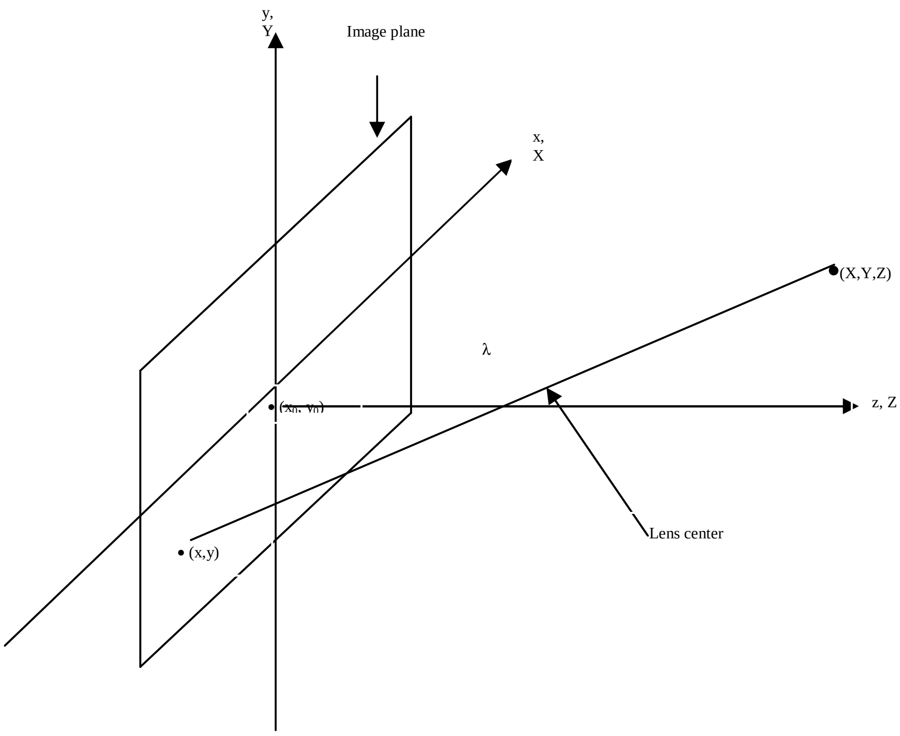


Figure 10: Basic model of the imaging process. The camera coordinate system (x,y) is aligned with the world coordinate system (X,Y,Z)

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} \Delta X \\ 0 \\ \Delta Z \end{bmatrix} = \begin{bmatrix} \dfrac{x_0 * Y}{y_0} + \Delta X \\ Y \\ \lambda - \dfrac{\lambda * Y}{y_0} + \Delta Z \end{bmatrix} \quad (5.3)$$

According to (5.2) and (5.3), we can determine the new image point x_0' and y_0'as:

$$\begin{cases} x_0' = \dfrac{\lambda * X'}{\lambda - Z'} = \dfrac{\lambda * x_0 * Y + \lambda * \Delta X * y_0}{\lambda * Y - y_0 * \Delta Z} \\ y_0' = \dfrac{\lambda * Y}{\lambda - Z'} = \dfrac{\lambda * y_0 * Y}{\lambda * Y - y_0 * \Delta Z} \end{cases} \quad (5.4)$$

Thus we can “move” an extracted vehicle image to any standard position and its size will be normalized.

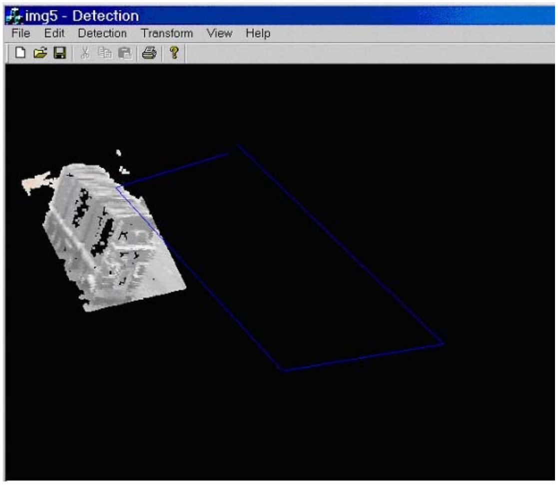


Figure 11: Extracted Vehicle normalized to standard view

5.4 Perform wavelet transform

After computing the normalized color image, we translate it to gray-scale using the simplest formula:

$G_{i,j}$ = gray scale intensity of pixel (i, j)

r,g,b = color intensities of pixel (i, j)

$$G_{i,j} = \frac{\sqrt{r^2 + g^2 + b^2}}{\sqrt{255^2 + 255^2 + 255^2}} \quad (5.5)$$

Using the gray-scale normalized image, we next perform a Haar wavelet transform. We positioned all vehicle images (128 X 128 pixels) in the left-bottom corner of the wavelet matrix. The wavelet matrix was also reduced to a manageable size of 16 X 16 pixels. This was the smallest size that maintained the features (shape, length, height and width) of each vehicle when we performed a reverse-wavelet transform. This reduction was done by taking only the upper left 16 X 16 coefficients of the 128 X 128 matrix. Figure 12 (a) is a duplicate of Figure 5. Figure 12 (b) is the gray-scale equivalent of Figure 12 (a). Figure 12 (c) is the wavelet to inverse wavelet transform using 16 X 16 coefficients.

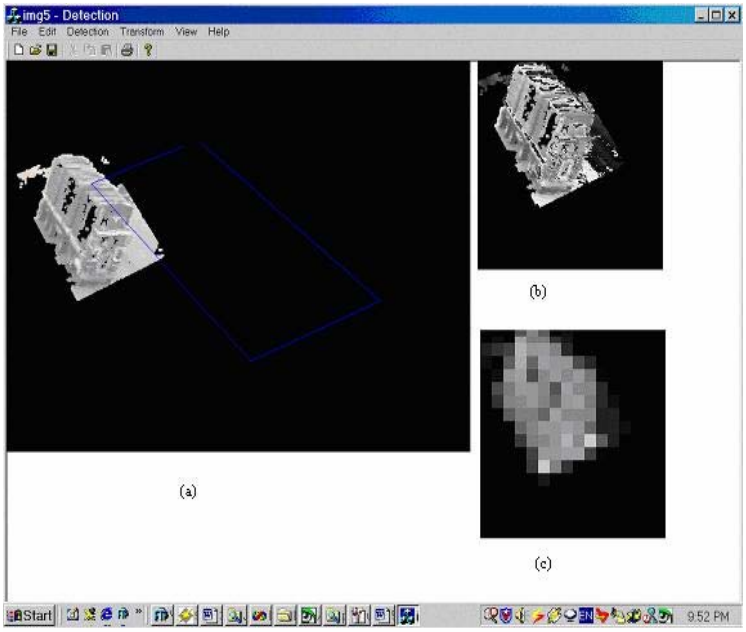


Figure 12: (a) Extracted vehicle normalized for standard view
(b) Gray-scale equivalent of (a) (128 X 128)
(c) Inverse wavelet transform of (b) (16 X 16)

5.5 Input of the wavelet transform to a neural network

Because the output of the wavelet transform is M real values, the input to the neural network is M values. The output of the wavelet transform for a NXN image is a NXN matrix of real values. This matrix is reduced by eliminating high frequency components so that the size of the feature set is more manageable, i.e., in our case 256 real values. Using the NeuralSIM neural network training software, we use a number of training images to set the weights of the network. The output of this neural network as to the category, the vehicle represents, was its decision.

6 Results and Discussion

The data used for this project were collected in Salt Lake City. A section of the four lane I215 freeway at 2500S was videoed with an uncalibrated standard NTSC camera. Data were collected for several days and a subset of this data was further processed and manually verified, i.e., vehicles identified manually. The data set used for this project consisted of approximately 2000 vehicles in moderate flow traffic. Since the majority of traffic was composed of passenger cars, this data set was further reduced to contain approximately 350 vehicles from different vehicle classes.

At first, a training data set composed of 50 vehicles was used for training a classification mode neural network. We used the following vehicle categorizes:

1. Motorcycle (2 data in training set)
2. Car and car with 1-axle trailer, van/pick up, van/pick up with 1-axle trailer (20 data in training set)
3. Bus, 3-axle truck and limousine (8 data in training set)
4. 2 or 3 axle semi with 1 or 2 axle trailer (9 data in training set)
5. >5-axle semi with single or multiple trailers (11 data in training set)

The vehicles were assigned to these categories based on the features that were viewable by the stationary camera. So we need re-category it.

The ultimate or overall goal of this project is to categorize vehicles into one of FHWA thirteen-category taxonomy. Appendix A presents sample images for these 13 categorizes. For this portion of the project, the goal was to improve on the work of Surendra Gupte, Osama Masoud, Robert F. K. Martin, and Nikolaos P. Papanikolopoulos [Gupte, 00]. In their work, they use vehicle dimensions to classify vehicles into two categories: cars (which constitute the majority of vehicles) and noncars (vans, SUVs, pickup trucks, tractor-trailers, semis and buses). They tested the system on image sequences of highway scenes. In a 20-minute sequence of freeway traffic, 90% of the vehicles were correctly detected and tracked. Of these correctly tracked vehicles, 70% of the vehicles were correctly classified.

We then tested the trained neural network on a set of 150 vehicles. The overall classification rate was 83%. Almost all of the misclassifications were classified class 3 as class 4. A more detailed analysis of the results shows that of the 20% errors for class 4, most (18% of the 20%) were class 4 being classified as class 5. Of the misclassifications for class 5 all were class 4. (The detailed results are listed in Table 2.)

After analyzing the errors from the above testing, we merged classes 3 and 4 into a single class. Because of the misclassification rates for class 4 and class 5, we enlarged the image size to 256 X 256 pixels. Class 5 vehicles include large semis. We then re-categorized as follows:

1. Motorcycle (2 data in training set)
2. Car and car with 1-axle trailer, van/pick up, van/pick up with 1-axle trailer (113 data in training set)
3. Bus and 2 or 3 axle truck (33 data in training set)
4. 2 or 3 axle semi with 1 or 2 axle trailer, 6 or 7 axle semi with single trailer (33 data in training set)
5. >5-axle semi with multiple trailers (19 in training set)

	Overall	**Class 1**	**Class 2**	**Class 3**	**Class 4**	**Class 5**
Classification rate	83%	100%	91%	67%	80%	86%
Total number of vehicles	150	3	32	15	50	50
Number of misclassified	25	0	3	5	10	7

Table 1: First Test Results

After the above modification, in order to improve the overall classification rate, we also acquired more data. We trained on a test set composed of 200 vehicles, and then ran the network to classify 341 vehicles. The results are as follows:

	Overall	**Class 1**	**Class 2**	**Class 3**	**Class 4**	**Class 5**
Classification rate	90%	75%	98%	74%	88%	83%
Total number of vehicles	341	4	192	58	57	30
Number of misclassified	33	1	3	15	9	5

Table 2: Final Results

The overall classification rate was 90%. All of the misclassifications were assigned to an adjacent category. The misclassifications of class 1 most likely occurred because we lacked sufficient motorcycle data. The misclassifications of the class 2 are because we classified the car with a trailer as class 3. Most errors occurred when a class 4 vehicle was classified as a class 3 or class 5 vehicles.

7 Conclusion

Because pervasive smart computing environments make people get accustomed to convenient and secure services. Vehicle classification can be defined as observation of highway vehicles and the subsequent sorting of the resulting data into a fixed set of categories. In practice, vehicle-classification data are extremely important because they are involved in most aspect of transportation and traffic engineering, such as pavement design, pavement-maintenance scheduling, commodity flow analysis, highway-capacity analysis, weight enforcement, and environmental analysis [Yuan, 94]. It is difficult to compare the method mentioned in this paper to previous research for several reasons. First, the most recent classification scheme used by previous researchers [Gupte, 00] differed significantly because it used only two different classes instead of the five used in this research. Second, different schemes used different surveillance technologies, such as inductive loops. Finally, other schemes have utilized calibrated cameras placed most advantageously for the process. Such as in the paper [Gupte, 00], the camera is placed on the side of freeway. In this project, the cameras were un-calibrated, and their placement could not be controlled.

The accuracy of this recognizer is encouraging. In addition to the overall classification rates of 83% and 90% for the two test datasets, the classification rates for individual vehicle classes were also consistent. In other words, this system was not biased toward the class with the most samples, namely passenger cars [Sun, 00]. One advantage of this method is the simplicity in the implementation and training of the neural networks. We used NeuralSIM software to implement and train the neural networks.

The classification of vehicles should be performed by a synthesis of multiple sets of features. All feature sets have not yet been determined. In this work, we used a set of features --- the reduced wavelet transform of a normalized image of a vehicle. We are also working to extract a specific set of physical features such as length, width, and height. Each of these sets of features (wavelet, physical characteristics, etc.) can be used to train a separate neural network. The final decision as to vehicle classification will then be based on the output of these two neural networks.

References

[Aono, 88] Aono, S. “Next Step in Automotive Electronic Control”, International Conference on Transportation Electron ProConvergence 88, IEEE, New York, 1988, pp. 83-89.

[Baker, 92] K. D. Baker and G. D. Sullivan, “Performance assessment of model-based tracking”, in Proc. IEEE Workshop Applications of Computer Vision, Palm Springs, CA, 1992, pp. 28-35.

[Burrus, 98] C. Sidney Burrus, Ramesh A. Gopinath, and Haitao Guo, Introduction to Wavelets and Wavelet Transforms, Prentice Hall, 1998.

[Chen, 00] Chen, C. H.; Cheng, Tzu-Hung; Wu, Char-Ur, “Wavelet-based video compression and analysis system for AITS”, ITS America. Meeting (10th: 2000: Washington, D.C.). revolutionary thinking, real results : ITS 2000 : conference proceedings, 15 p. 2000.

[Chen, 07] Xin Chen and Chengcui Zhang, Vehicle Classification from Traffic Surveillance Videos at a Finer Granularity, Vol. 4351, pp. 772-781, the 13th International multimedia modeling conference, MMM 2007, Singapore, January 9-12, 2007.

[Chen, 08] W.-H. Chen,; W. X. Zheng, “Improved Delay-Dependent Asymptotic Stability Criteria for Delayed Neural Networks”, IEEE Transactions on Neural Networks, pp. 1-8, Vol. 19, No. 11, 2008.

[De, 08] Shuzhi Sam Ge; Chenguang Yang; Tong Heng Lee, “Adaptive Predictive Control Using Neural Network for a Class of Pure-Feedback Systems in Discrete Time”, IEEE Transactions on Neural Networks, Vol. 19, No. 9, pp. 1599-1614, 2008.

[Duarte, 04] Marco F. Duarte and Yu Hen Hu, Vehicle classification in distributed sensor networks, Journal of Parallel and Distributed Computing, no. 64, pp. 826-838, 2004.

[Ensari, 05a] T. Ensari and S. Arik, “Global stability of class of neural networks with time varying delays”, IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 52, no. 3, pp. 126–130, Mar. 2005.

[Ensari, 05b] T. Ensari and S. Arik, "Global stability analysis of neural networks with multiple time varying delays", IEEE Trans. Autom. Control, vol. 50, no. 11, pp. 1781–1785, Nov. 2005.

[Faghri, 92] Faghri, A., and J. Hua. "Evaluation of Artificial Neural Network Applications in Transportation Engineering". In Transportation Research Record 1358, TRB, National Research Council, Washington, D. C., 1992, pp. 71-80.

[Faghri, 94] Faghri, A., and J. Hua, "Applications of ANNS to IVHS", In Intelligent Transportation Systems: evaluation, driver behavior, and artificial intelligence: IMPRINT: Washington D. C.: National Academy Press, 1994.

[Guarneri, 2008] Guarneri, P.; Rocca, G.; Gobbi, M., "A Neural-Network-Based Model for the Dynamic Simulation of the Tire/Suspension System While Traversing Road Irregularities", IEEE Transactions on Neural Networks, Vol. 19, No. 9, pp. 1549-1563, 2008.

[Gupte, 00] Gupte, Surendra; Masoud, Osama; Papanikolopoulos, Nikolaos P. "Detection and classification of vehicles", in ITS America. Meeting (10th: 2000: Washington, D.C..). revolutionary thinking, real results : ITS 2000 : conference proceedings, 2000.

[Haykin, 91] Haykin, S., W. Stehwien, C. Deng, P. Weber, and R. Mann. "Classification of Radar Clutter in an Air Traffic Control Environment". Proc. IEEE, vol. 79, no. 6, June 1991, pp. 742-772.

[Hertz, 91] Hertz, J., A. Krogh, and G. R. Palmer. Introduction to the Theory of Neural Computing, Addison-Wesley, Redwood City, Calif. 1991.

[Hua, 93] Hua, J., and A. Faghri. "Dynamic Traffic Pattern Classification." In Transportation Research Record 1399, TRB, National Research Council, Washington, D. C., 1993, pp. 14-19.

[Jiang, 96] Jiang, Zifeng; Jing, Bianshun, "Automatic incident detection using combination approach of neural networks and wavelet transforms", World Congress on Intelligent Transport Systems (3rd: 1996: Orlando, Fla.). Intelligent transportation: realizing the future: abstracts of the Third World Congress on Intelligent Transport Systems, p. 251

[Kenneth, 96] Kenneth R. Castleman, Digital Image Processing, Prentice Hall, 1996.

[Koller, 93] D. Koller, "Moving object recognition and classification based on recursive shape parameter estimation", in Proc. 12th Israel Conf. Artificial Intelligence, Computer Vision, Dec. 27-28, 1993.

[Lipton, 98] A. J. Lipton, H. Fujiyoshi, and R. S. Patil, "Moving target classification and tracking from real-time video," in Proc. IEEE Workshop Applications of Computer Vision, 1998, pp. 8-14.

[Lubin, 92] Lubin, L. M., E. C. Huber, S. A. Gilbert, and A. L. Kornhauser. "Analysis of a Neural Network Controller for an Automated Road Vehicle". IVHS --- Issues and Technology, SAE Special Publications 928. SAE, Warrendale, Pa., 1992, pp. 23-44.

[Martin, 95] Martin T. Hagan, Howard B. Demuth, Mark Beale, Neural Network Design, PWS Publishing Company, 1995.

[Morris, 06] Brendan Morris and Mohan Trivedi, Improved Vehicle Classification in Long Traffic Video by Cooperating Tracker and Classifier Modules, IEEE International conference on Advanced Video and Signal Based Surveillance, Nov. 2006.

[Nakatsuji, 90] Nakatsuji, T., and T. Kaku. "Application of Neural Network Models to Traffic Engineering Problems". Proc., 11th International Symposium on Transportation and Traffic Theory, Yokohama, Japan, 1990, pp. 291-306.

[Nakatsuji, 91] Nakatsuji, T., and T. Kaku. "Development of Self-Organizing Traffic Control System Using Neural Network Models". In Transportation Research Recoed 1324, TRB, National Research Council, Washington, D. C., 1991, pp. 137-145.

[Park, 2006] J. H. Park, "A new stability analysis of delayed cellular neural networks", Appl. Math. Comput., vol. 181, no. 1, pp. 200–205, 2006.

[Rafael, 92] Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing, Addison-Wesley Publishing Company, 1992.

[Rourke, 92] Rourke, A., Bell, Michael G. H., "Image Processing Vehicle Classification System Feasibility Study", In Contractor report (Transport and Road Research Laboratory), 1992, p. 32 [7].

[Sullivan, 92] G. D. Sullivan, "Model-based vision for traffic scenes using the ground-plane constraint", Phil. Trans. Roy. Soc. (B), vol. 337, pp. 361-370, 1992.

[Sullivan, 95] G. D. Sullivan, A. D. Worrall, and J. M. Ferryman, "Visual object recognition using deformable models of vehicles", in Proc. Workshop on Context-Based Vision, Cambridge, MA, June 1995, pp. 75-86.

[Sullivan, 97] G. D. Sullivan, K. D. Baker, A. D. Worrall, C. I. Attwood, and P. M. Remagnino, "Model-based vehicle detection and classification using orthographic approximations", Image Vis. Comput., vol. 15, no. 8, pp. 649-654, Aug. 1997.

[Sun, 00] Carlos Sun, Stephen G. Ritchie, Seri OH, "Inductive Classifying Artificial Network for Vehicle Type Categorization", December 2000.

[Umbaugh, 98] Scott E Umbaugh, Computer Vision and Image Processing: A Practical Approach Using CVIPtools, Prentice Hall, 1998.

[Urazghildiiev, 07] Ildar Urazghildiiev, Rolf Ragnarsson, Pierre Ridderström, Anders Rydberg, Eric Öjefors, Kjell Wallin, Per Enochsson, Magnus Ericson, and Göran Löfqvist, "Vehicle Classification Based on the Radar Measurement of Height Profiles", IEEE Transactions on intelligent transportation systems, vol. 8, no. 2, pp. 245-253, June 2007.

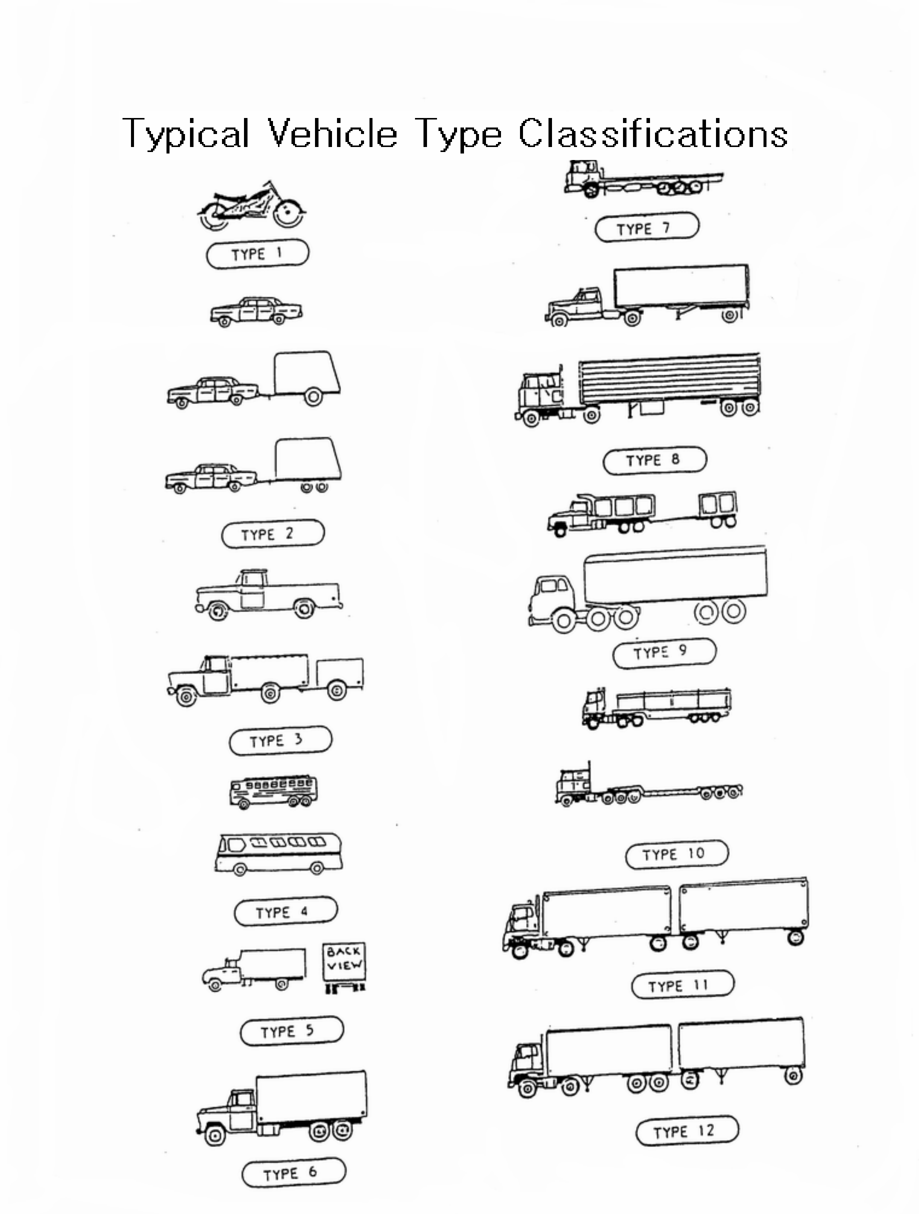
[Wei, 96] Wei, C. et al, "Vehicle Classification Using Advanced Technologies", In Transportation research record, Vol. 1551, Nov. 1996, p. 45-50.

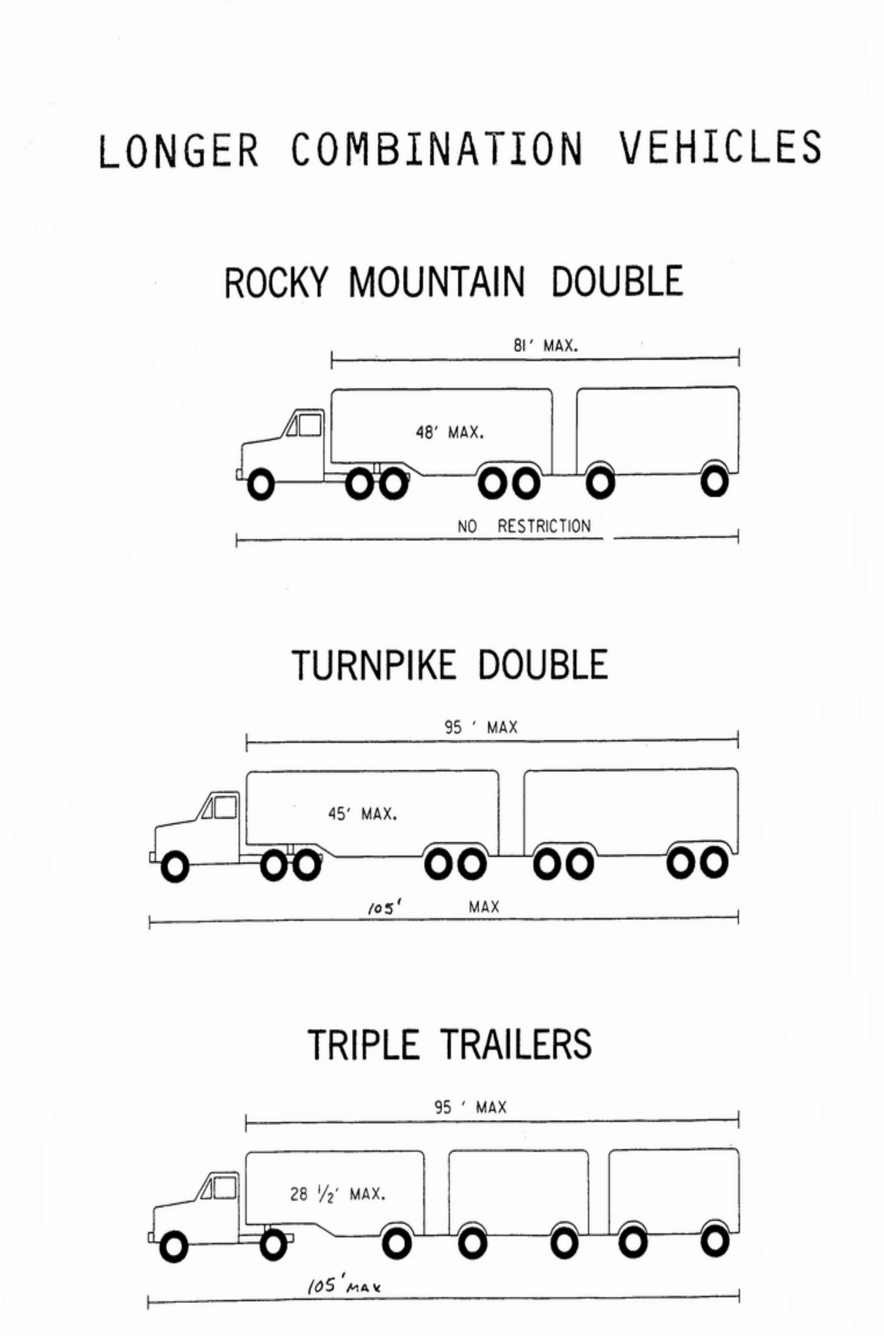
[Yuan, 94] Yuan, Xidong; Lu, Yean-Jye; Sarraf, Semaan, "Computer vision system for automatic vehicle classification", Journal of transportation engineering, Vol. 120, no. 6, November-December 1994, p. 861-876.

[Yoshida, 02] T. Yoshida, S. Mohottala, M. Kagesawa, K. Ikeuchi, "Vehicle Classification System Using Local-Feature Based Algorithm", IEICE TRANS. INF.& SYST.,Vol.E85-D, No11 Nov 2002.

[Zuo, 2008] Wei Zuo; Lilong Cai, "Adaptive-Fourier-Neural-Network-Based Control for a Class of Uncertain Nonlinear Systems", IEEE Transactions on Neural Networks, Vol. 19, No. 10, pp. 1689-1701, 2008.

Appendix A: FHWA thirteen-category taxonomy




LONGER COMBINATION VEHICLES
ROCKY MOUNTAIN DOUBLE
81' MAX.
48' MAX.
NO RESTRICTION
TURNPIKE DOUBLE
95 ' MAX
45' MAX.
105' MAX
TRIPLE TRAILERS
95 ' MAX
28 1/2' MAX.
105' MAX

FHWA TYPE VEHICLE CLASSIFICATION SCHEME

Type	Description	1-2 Axles	2-3 Axles	3-4 Axles	4-5 Axles	5-6 Axles	6-7 Axles
1	Motorcycle	0.1-6.0					
2	Car	6.1-10.2					
2	Car w/1 Axle Trlr	6.1-10.2	6.0-18.0				
2	Car w/2 Axle Trlr	6.1-10.2	6.0-18.0	0.1-6.0			
3	Pickup/Van	10.3-13.0					
3	Pickup/Van w/1A Trlr	10.3-13.0	6.0-18.0				
3	Pickup/Van w/2A Trlr	10.3-13.0	6.0-18.0	0.1-6.0			
4	Bus	20.0-40.0					
4	Bus	20.0-40.0	0.1-6.0				
5	2 Axle- Six Tire	13.1-20.0					
6	3 Axle- Single Unit	6.1-23.0	0.1-6.0				
7	4 Axle- Single Unit	6.1-23.0	0.1-9.0	0.1-9.0			
8	2S1	6.1-17.0	14.0-40.0				
8	3S1	6.1-20.0	0.1-6.0	6.1-40.0			
8	2S2	6.1-17.0	14.0-40.0	0.1-6.1			
9	3S2	6.1-22.0	0.1-6.0	6.1-40.0	0.1-9.0		
9	3 Axle w/Trlr	6.1-22.0	0.1-6.0	6.1-23.0	1.1-23.0		
10	6 Axle- Single Trlr	6.1-22.0	0.1-6.0	0.1-40.0	0.1-11.0	0.1-11.0	
10	7 Axle- Single Trlr	6.1-22.0	0.1-6.0	0.1-40.0	0.1-13.0	0.1-13.0	0.1-13.0
11	5 Axle- Multi Trlr	6.1-17.0	11.1-25.0	6.1-18.0	11.1-25.0		
12	6 Axle- Multi Trlr	6.1-22.0	0.1-6.0	1.1-25.0	6.1-18.0	11.1-25.0	
13	7 Axle- Multi Trlr	0.1-40.0	0.1-40.0	0.1-40.0	0.1-40.0	0.1-40.0	0.1-40.0
15	Unclassified Vehicles						