

# Bus Detection for Intelligent Transport Systems Using Computer Vision

Mijail Gerschuni and Alvaro Pardo

Department of Electrical Engineering, School of Engineering and Technologies,  
Universidad Catolica del Uruguay  
mgerschu@gmail.com, apardo@ucu.edu.uy

**Abstract.** In this work we explore the use of computer vision for bus detection in the context of intelligent transport systems. We propose a simple and efficient method to detect moving objects using a probabilistic modelling of the scene. For classification of the detected moving regions we study the use of eigenfaces.

## 1 Introduction

In recent years there has been an increasing interest on improving public transport services. This is due to both direct and indirect benefits that can be obtained by cities from a logistical, environmental and social point of view. As a direct benefit, a well organized and efficient public transport system allows to reduce travel times, reduce traffic congestion while offering a comfortable alternative to family cars, and therefore also reduces pollution levels. On the other hand, one of the indirect benefits is the impact on the economy of the city. Big cities are increasingly important in the economy of the countries; providing appropriate logistical infrastructure helps to attract more business and investment. Governments recognize this reality and are increasingly investing in infrastructure, highways, subways, etc.

In order to improve public transport systems the main infrastructure investment is in exclusive bus corridors. However, the high costs associated with them make it not feasible to do it in all arteries of the city. Therefore, an alternative solution is the demarcation of preferential lanes for public transport in already existing streets and avenues. To be truly effective this solution requires the classification of vehicles traveling on them to detect buses and give them right of way at traffic lights.

In recent years the use of Computer Vision has expanded its application in Intelligent Transportation Systems, in particular for vehicle classification [2,5,3]

In the case of exclusive corridors, the detection of buses can be easily implemented because there is a physical separation between them and private vehicles. In the case of preferential lanes this is not the case. A computer vision based system has two major advantages, its cost and its scalability. It is of low cost, compared to other solutions, because it enables the use of preferential lanes and reduces the construction cost associated with exclusive lanes. Computer vision

it is also an interesting technology that can be used for other purposes such as vehicle counting, detecting special vehicles, speeding control and surveillance.

In this work we present a system that applying computer vision automatically detects buses preferential lanes to enable the synchronization of traffic lights on a main road in order to minimize travel times.

## 2 Existing Solutions

To solve the problem of vehicle detection a sensor is required. Sensors can be classified into two types: intrusive and non-intrusive.

Intrusive sensors include inductive loops, piezoelectric cables and magnetometers among others. These are installed directly on the floor, either on it or under by pipeline as is shown in Figure 1. The operation thereof is generally simple and well known because they are mature technologies. The major disadvantage is that they require traffic disruptions for installation and maintenance. They also have many flaws associated with pavement condition and life depends a lot on the installation procedures.

On the other hand, non-intrusive sensors may be installed and maintained with minimal traffic distortion (Figure 1). These sensors include computer vision based solutions, microwave radar, laser, infrared detection etc. They allow the supervision of several lanes and are able to provide further information such as the vehicle type detected.



**Fig. 1.** Left: Installation of intrusive loops. Right: Computer Vision sensors.

In the case of exclusive corridors, intrusive sensors can be used for bus detection because there is a physical separation between them and private vehicles. For preferential lanes (most widespread solutions due to its lower costs) these sensors can not ensure the correct detection due to private vehicles. Some recent works that address the problem of sorting vehicles according to their magnetic signature captured with inductive loops was presented in [1].

The advantage of computer vision solution is the high value that can be generated due to its greater scalability. Most large scale traffic control systems use them for a variety of applications. First, they let you have a visual view of the traffic state, particularly useful when analyzing the causes of accidents that may

result in roads. They also allow more elaborated statistical data analysis such as the level of congestion and the use of avenues. This can be obtained by calculating the occupation times and queues lengths. Computer vision solutions also allow to calculate the speed of vehicles that can be used for speeding infractions. It is also possible to do plate recognition, which can be used in shadow tolling, among other things.

In the next section we review the literature on computer vision based solutions for intelligent transport systems.

### 3 Computer Vision for Intelligent Traffic Systems

Although the work presented in [1] it is not based in computer vision it proposes a pattern recognition approach to classify vehicles by their magnetic signature obtained using inductive loops. The magnetic signature is a characteristic of each vehicle which depends on their geometry and distribution of metal parts. One way to obtain the magnetic signature is via the oscillation frequency versus time of an oscillator that uses an inductive loop during the time the vehicle passes over it. The database used is rather small and contains 34 cars, 9 vans and 18 buses. With this database the best classification is obtained with a naive Bayes classifier in the dissimilarity space with 99,3% of the vehicles correctly classified. Due to the small size of the database this can be considered as an upper bound of the classification performance. Nevertheless, the use of inductive loops it is a very robust method for this problem, and although it is an intrusive method, we include these results here to use it as comparison of our proposal.

In [2] a real-time vehicle detection based on background learning of the scene is proposed. The method has two processes running in parallel, one at high level and another one at low level. The low level process is the estimation of the background and runs in real time. The high level one runs at a lower frequency and is responsible of classifying the pixels into categories: lines, pavement or neither of the aforementioned. For this classification, color and shape features are considered. The proposed method only detects vehicles but not classifies them; it obtains a 90% of correct detection rate. We include this work to have a reference on the detection rate.

The work presented in [5] addresses the real-time vehicle classification based on eigenfaces [4]. The method implies taking a set of training images to learn the features of each class of interest: buses, cars, etc. The feature space is calculated using principal components analysis and then a nearest neighbor classifier is applied. The published results show a classification rate of 100% but using as the test set the same set used for the training consisting of 100 images of vehicles fronts. We apply the same methodology but considering complete images of the vehicles as shown in Figure 2. We also train and test the algorithm with independent sets to evaluate different classifiers to understand the potential of the method in a real scenario.

Finally in [3] an algorithm for detection and classification of vehicles is presented. The detection achieves a good detection rate of 90% but the recognition based on structural features only achieves 70% of correct classification.



**Fig. 2.** Models uses for training

## 4 Proposed Method

Using a camera conveniently located we are able to capture traffic images that are processed in real time in order to detect the presence of buses. Before proceeding we shall mention that we assume the following working conditions: good visibility and normal weather conditions.

Our system is composed of the following modules: image acquisition, segmentation, feature extraction and classification. The first step of image acquisition also prepares the image for further processing, for example adding noise filtering operations. In the segmentation step vehicles in motion are extracted from the background. For this step we propose a probabilistic approach that facilitates the typical selection of thresholds for segmentation purposes. Once the regions of interest are obtained from the segmentation step, each region is expanded in the feature space given by the eigenfaces method [4]. Finally, based on the calculated features each moving region is assigned to a given class (car, bus, truck, etc.).

### 4.1 Segmentation

Given two consecutive frames at times  $n$  and  $n+1$ ,  $I_n(x)$  and  $I_{n+1}(x)$  we consider their difference  $d(x) = |I_n(x) - I_{n+1}(x)|$  where  $x$  indicates the pixel. If we assume that most of the pixels belong to the background and that differences among these pixels are assumed to be small, then most of the pixels will exhibit small values in  $d(x)$ . If we look at the histogram of  $d(x)$  we will find that most of the pixels are concentrated at small values. If we view the image intensity differences as a random variable, whose magnitude represents the probability that a pixel

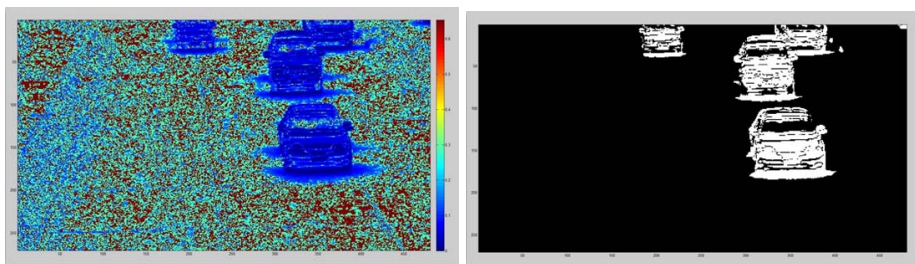
belongs or not to a moving object, we can interpret the histogram of  $d(x)$  as an empirical approximation to the density function,  $f(y)$ . Given a threshold  $\alpha$  the probability that the difference falls in  $[\alpha, 255]$  is:

$$P(\alpha \leq d \leq 255) = 1 - \sum_{y=0}^{\alpha} f(y).$$

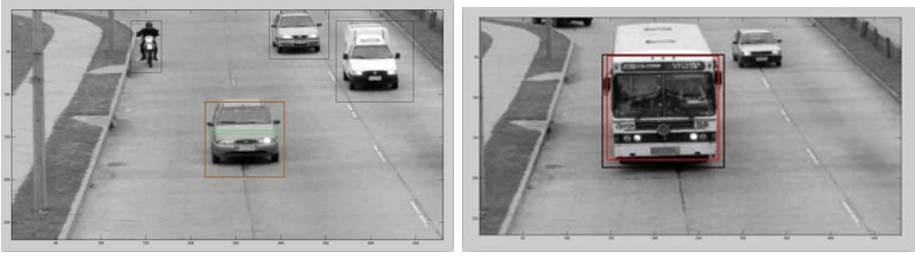
Instead of fixing the threshold  $\alpha$  which can vary due to lightning conditions and shadows, we fix the probability instead. That is, a pixel  $x$  is declared as moving if the probability of its difference  $d(x)$  is below a probability threshold  $P_\alpha$ :  $P(\alpha \leq d(x) \leq 255) \leq P_\alpha$ . We define the following probabilities image:

$$IMP_n(x) = 1 - \sum_{y=0}^{\alpha} f(y),$$

and with it a binary image with objects labeled as one:  $IMS_n(x) = IMP_n(x) \leq P_\alpha$ . In all the experiments of this paper we choose  $P_\alpha = 0.08$ . In Figure 3 the probability image and the final segmentation mask. Once we have the binary image with detected moving objects we apply mathematical morphology to simplify the detected regions followed by a labelling process which labels all connected components and extract their bounding boxes. In Figure 4 we show an image with all detected regions of interest. All bounding boxes are processed with some heuristic to join close bounding boxes and remove the ones that dont fulfill basic requirements of size and shape, see Figure 3. Finally we apply an optimization procedure to shrink the bounding boxes towards the real boundaries of the vehicles. To do this we use the integral image of  $IMP_n(x)$  to move the bounding box inwards to minimize the mean probability inside the bounding box; the bounding box is reduced until no noticeable reduction is observed. The use of the integral image allows a fast implementation. Observe that the same procedure must be applied to each bounding box in the image. See Figure 4 for an example of the output of this procedure.



**Fig. 3.** Left: Probabilities image. Right: Segmentation of moving objects.



**Fig. 4.** Left: Probabilities image. Right: Segmentation of moving objects.

## 4.2 Classification

Once we have the bounding boxes of the regions of interest we use a classifier to decide the class of each of them. Starting from the greyscale values of regions of interest, see Figure 2, we apply the method of eigenfaces to extract the projection coefficients and use them as classification features. The method of eigenfaces [4], originally developed to recognize faces can be applied to our problem of vehicle recognition. In order to make this article self contained we are going to summarize the main concepts behind eigenfaces.

The training set contains  $M$  vehicles images of known class scanned in lexicographical order. We assume that all images are resized to have the same size  $N \times N$  and therefore each sample  $\Gamma_k$  in the training set has  $N^2$  elements;  $\Gamma = \{\Gamma_1, \dots, \Gamma_M\}$ . The eigenfaces method is based on principal component analysis (PCA). The first step for the application is to remove the mean of the samples:

$$\Phi_k = \Gamma_k - \frac{1}{M} \sum_{i=1}^M \Gamma_i$$

Then, the eigenvectors  $u_i$  of the covariance matrix  $C$  of the set  $\{\Phi_1, \dots, \Phi_M\}$  are computed. The matrix  $C$  is calculated as:

$$C = \frac{1}{M} \sum_{i=1}^m \Phi_i \Phi_i^t = \frac{1}{M} A A^t,$$

where  $A$  is a matrix with vectors  $\Phi_k$  as columns.

Since this matrix  $C$  size is  $N^2 \times N^2$  we use the same idea of [4] and compute the eigenvectors of  $A^t A$ . It can be shown that if  $v_i$  is an eigenvector of  $A^t A$  then  $A v_i$  is an eigenvector of  $A A^t$ . In this way we obtain  $M$  eigenvectors and eigenvalues of  $A A^t$ . Once we have the eigenvectors all the data points in the training set are projected into them to build the feature space. That is, the eigenvectors  $u_i$  constitute the base of the feature space and the coordinates on each eigenvector are the features to be used for classification.

During recognition, each region of interest is resized to size  $N \times N$ , the mean of the training set is removed and then projected to  $\{u_1, \dots, u_L\}$  to obtain its coordinates in the feature space. In this space we can apply any supervised

classifier. For this work we tested five standard classifiers to select the one with best performance and lowest computational cost.

The training set was build using samples for each of the target classes as shown in Table 1. In Figure 2 we show some of the image samples. As we can see the sample contain the object of interest but no fine tuning was imposed.

**Table 1.** Number of samples for each class in the testing set

Class	#Samples	Class	#Samples
Bus	76	Truck	46
4x4	54	School Buses	21
Vans	43	Cars	115
Motorbikes	30		
	Total	385	

## 5 Classification Results

Five classifiers were trained following the procedure of previous sections and performance evaluated using 10-fold crossvalidation. Although our main goal is to detect buses we also evaluated the potential of the approach to correctly classify all the classes of vehicles contained in the training set. In Table 2 we show the results for the five selected classifiers.

**Table 2.** Bus classification performance for different classifiers. The last column contains the correct classification for the seven classes.

Classifier	TP Rate	FP Rate	Precision	Recall	Global Correct Class.
Naive Bayes	0,855	0,016	0,929	0,855	73,5%
Neural Network	0,895	0,023	0,907	0,895	74,3%
Random Forest	0,868	0,032	0,868	0,868	68,3%
k-NN (k=1)	0,947	0,006	0,973	0,947	80,3%
k-NN (k=3)	1,000	0,010	0,962	1,000	79,2%

If we concentrate ourselves in the correct recognition of buses, that is we measure the correct classification into two metaclasses buses and non-buses, the performance of the five classifiers increases. Table 3 we show the percentage of correctly classified buses.

As we can see the best classifier is k-NN with  $k = 3$ ; it achieves 100% of true positives and only 3 false positives. Although, the five classifiers obtain good classifier results, it is importante to note that the nearest neighbor classifiers are the ones with best performance in terms of true and false positives. This is important due to the difference between the number of samples in bus and non-bus classes.

**Table 3.** Classification performance for different classifiers into two classes bus and non-bus

Classifier	Bus/Non-Bus Correct Class.	Bus TP Rate	Bus FP Rate
Naive Bayes	95,8%	85,5% (65/76)	1,6% (5/309)
Neural Network	96,1%	89,4% (68/76)	1,9% (7/309)
Random Forest	94,8%	86,8% (66/76)	2,9% (9/309)
k-NN (k=1)	98,4%	94,7% (72/76)	0,6% (2/309)
k-NN (k=3)	99,2%	100% (76/76)	0,9% (3/309)

## 6 Discussion and Conclusions

The complete system was tested with a set of videos where 14 buses were correctly classified among above 100 vehicles and only 1 false positive was produced. We note that a false positive is tolerable while missing a bus is not. From this point of view the proposed system achieves good performance in terms of recall and precision.

If we compare our results with the ones presented in [1] using inductive loops we can see that we obtain 81,2 % in the classification in all classes while in the previous work the author reports 98,4% over a smaller set consisting only in cars, vans and buses. If we consider both systems as bus detectors their performance is equivalent since we reach 100%.

When comparing our results to [5] the first observation is that in this work the authors recognize car categories and therefore a direct comparison is not possible. We showed that using the same basic algorithm, eigenfaces, buses can be correctly recognized among other vehicles. This shows the potential of this solution for real applications in the context of intelligent transport systems.

We also propose a simple and efficient method to detect moving regions based on the probability of a pixel to be part of the static background. This method simplifies the threshold selection and auto-adapts its value based on the probability distribution of the static pixels from the background.

## References

1. Derregibus, A.: Clasificación de vehículos con lazos inductivos. Master's thesis, Universidad Católica del Uruguay (2012)
2. Tan, X.J., Li, J., Liu, C.L.: A video-based real-time vehicle detection method by classified background learning. *World Transactions on Engineering and Technology Education* 6, 189–192 (2007)
3. Mo, S., Liu, Z., Zhang, J., Wu, C.: Real-time vehicle classification method for multi-lanes roads. In: *IEEE Conference on Industrial Electronics and Applications*, pp. 960–964 (2009)
4. Turk, M., Pentland, A.: Eigenfaces for recognition. *Journal of Cognitive Neuroscience* 3(1), 71–86 (1991)
5. Wang, W., Shang, Y., Guo, J., Qian, Z.: Real-time vehicle classification based on eigenface. In: *International Conference on Consumer Electronics, Communications and Networks*, pp. 4292–4295 (2011)