Automated Toll Collection System based on Vehicle Type Classification using Sparse-Filtered Convolutional Neural Networks with Layer-Skipping Strategy (SF-CNNLS)

Suryanti Awang¹ and Nik Mohamad Aizuddin Nik Azmi¹
¹Soft Computing & Intelligent System Research Group (SPINT), Faculty of Computer Systems & Software Engineering, Universiti Malaysia Pahang, Lebuhraya Tun Razak, 26300, Pahang, Malaysia
*suryanti@ump.edu.my

Abstract. Automated Toll Collection System (ATCS) is one of the technologies to fulfill the Intelligent Transportation System’s (ITS) aim in providing an efficient road and transportation infrastructure at the expressway. This paper is aimed to provide an accurate and efficient ATCS based on a vehicle type classification method rather than the current implementation of toll collection that rely on sensor-based and human observation. To fulfill the aim, we proposed to implement SF-CNNLS framework to extract vehicle’s features and classify it into class 1 (passenger vehicle), class 2 (lorry) and class 4 (taxi). This ATCS is aimed to enhance the efficiency of the toll collection in Malaysia. The biggest challenge in this research is how to discriminate features of class 4 as a different class of class 1 since both classes have almost identical features. However, with our proposed method, we are able to classify the vehicle with an average accuracy of 90.83 %. We tested our method using a frontal view of a vehicle from the self-obtained database (SPINT) taken using mounted-camera at the toll booth and compare the classification performance with a benchmark database named BIT.

1. Introduction
Intelligent Transportation System (ITS) corresponds to the technology applied in road and transportation infrastructure to improve safety and efficiency in related applications, for instance, expressway toll system, traffic census, traffic surveillance, etc. This paper focuses on the expressway toll system especially in Malaysia, conventional methods, for instance, manually observed by human, sensor-based and electronic tools based is implemented. Basically, the expressway toll collection system will charge the vehicle based on the classes; class 1 for passenger vehicle (sedan car, MPV, SUV, etc), class 2 is for lorry, class 3 is for truck, class 4 is for taxi and class 5 is for bus. One of the issues with the current implementation of the toll collection system is human error may occur during the observation. Furthermore, as for the sensor-based and electronic tools based implementation, they are high in maintenance and need some configuration tools as in some of the existing works [1,2]. Therefore, we proposed to overcome the issues by using ATCS that is camera-based (computer vision and combined with the advantages of machine learning). The advantages of this approach is low in maintenance and easy to be configured since only camera and a system are needed to classify the vehicle.

There are a few works in ATCS that implemented computer vision to classify the vehicle class. For example, [3] detects the vehicle for toll collection using computer vision with embedded Linux.
vehicle is classified into light and heavy vehicle based on the vehicle image captured using a mounted-camera at the toll booth. Different from work done by [4], the authors recognized license plate of the vehicle that use an expressway at the toll booth in classifying the class of vehicle by using a segmentation approach and template matching in the image processing.

Since there is very limited of the existing ATCS work that implement the computer vision and machine learning, we considered the works in vehicle counting system (VCS) that are much related to this research. For example, a surveillance camera and traffic camera from top view are used to track and count vehicles in works done by [5,6,7,8]. The vehicles are counted as general (vehicle or non-vehicle) without any specific classification. These approaches are able to count the vehicles, however, unable to give an accurate outcome as expected in the traffic census. [9] also counts the vehicle without classifying its class using a motion estimation with Taylor series approximation. Other than that, there is Principle Component Pursuit (PCP) is used to extract features of vehicles from a satellite image, and vehicles are counted based on that features and does not consider the class of vehicle [10].

There is a VCS that is based on vehicle types proposed by [11]. They count the vehicles as small car, van, and motorcycle using Gaussian Mixture Model (GMM) to detect the vehicle. The drawback of this approach is the feature that extracted to be used in classifying the vehicle is too limited which is only rely on the size of blob. VCS also can be developed by using a combination of techniques in computer vision and machine learning. [12] proposed to implement this combination in VCS for traffic control analysis. They applied Fast Region-based Convolutional Neural Network (FR-CNN) to detect the moving vehicles. However, this approach only detects and count the moving vehicles without able to do the counting based on vehicle types.

Based on the existing works, we found that the ATCS can be improved with the implementation of intelligent techniques in machine learning to classify the vehicle type. Therefore, we proposed to implement SF-CNNLS based on deep learning method. This framework is able to learn vehicle features in detail and the ATCS is classified based on the classes that defined by the expressway concession. However, we notice that the main challenge in this approach is how to differentiate class 1 and class 4. This is because the features of both class is almost similar especially in a grayscale image. Thus, our aim is to overcome that challenge. The explanation on the methodology of ATCS with SF-CNNLS is in the next section, followed by experiment and result section and end with conclusion section.

2. Methodology
There are two stages of methodology in this paper; ATCS and vehicle type classification. We will explain the ATCS process flow first, and later is the explanation on vehicle type classification based on SF-CNNLS framework.

2.1. Automated Toll Collection System (ATCS)
We captured the vehicle image from a mounted camera video to be classified based on the vehicle class and toll rate. For instance, of toll rate with a flat rate for each class taken from the specific toll plaza, whereby, the expressway concession divided the passenger vehicle as class 1, class 2 for lorry, class 3 for truck or heavy vehicle, class 4 for taxi and class 5.

To detect the vehicle in the video, we use region in detecting the region of interest for moving vehicle. The region is allocated in a specific in the video to capture a moving vehicle one by one. Figure 1 shows the process flow of ATCS. For the first phase, we consider to classify the vehicle from class 1, class 2 and class 4. From the video, when a vehicle is detected, vehicle type classification will classify it using SF-CNNLS. The ATCS will display the toll rate according to the class.
2.2. Vehicle Type Classification – SF-CNNLS

This section is about how SF-CNNLS is implemented in classifying the vehicle class. There are two processes, namely, image acquisition to acquire vehicle images and pre-processing to prepare the images prior the implementation of SF-CNNLS. In that processes, the training and testing images that contain frontal-view of the vehicle are recorded using a surveillance camera. The pre-processing technique that has been used is a combination of existing works done by [11,12]. There are five pre-processing methods used, which are converting from RGB color space to grayscale, histogram equalization (HE), resizing with maintained aspect ratio, normalizing to zero mean and unit variance, and local contrast normalization (LCN) consequently. The SF-CNNLS is used as a feature extraction technique to extract vehicle features to be classified in the classification process. The SF-CNNLS is a sparse-filtered of the convolutional neural networks with layer-skipping strategy that has capability to extract both local and global features of a vehicle. Figure 2 shows the overview framework of SF-CNNLS implementation.

![Figure 1. ATCS process flow.](image1)

![Figure 2. SF-CNNLS Overview Framework](image2)

There are 2 major phases in the SF-CNNLS, namely, training and testing phase. In the CNNLS process, there are two stages of hidden layer, one post hidden layer and fully connected that will implement the feature extraction of the pre-processed images. Note that there are two types of training phase, namely, unsupervised and supervised. The purpose of the unsupervised training is to generate two stages of optimized Sparse Filters, whereas the supervised training is to generate parameter weights and biases that will be used by the classifier in the classification process.
During the unsupervised training, a set of pre-processed images are firstly delivered into the Sparse Filtering function to generate a set of optimized stage 1 sparse filters. Later, the set of pre-processed input images are delivered into CNNLS stage 1 hidden layer and convolved with the optimized stage 1 sparse filters. The output from the CNNLS stage 1 hidden layer is used as an input for the Sparse Filtering function to generate a set of optimized stage 2 sparse filters. Thus, the outcome of the unsupervised training is a set of optimized stage 1 sparse filters and a set of optimized stage 2 sparse filters.

For the supervised training, the set of pre-processed images are delivered into CNNLS stage 1 hidden layer and will be convolved with the stage 1 sparse filters. The extracted features from this stage will be an input to the stage 2 hidden layer and the post hidden layer. In the stage 2 hidden layer, the input is convolved with the stage 2 sparse filters and the same layers are deployed to obtain the extracted features. This extracted features will be an input to the post hidden layer. The output from both stages which are the extracted features will be concatenated at the fully connected process into a single vector. This single vector feature is used to train the Softmax Regression classifier. Note that the testing set in testing phase has similar process with the supervised training except at the classifier where the trained weights and biases are used to calculate classification probabilistic in obtaining a classification result.

2.2.1. Feature Extraction: SF-CNNLS and Classification using Softmax Regression. There are five components called layers in each of the hidden layer. The components are convolutional layer, absolute value rectification (AVR) layer, local contrast normalization (LCN) layer, average pooling layer and subsampling layer without zero padding [13]. The output from each layer is the extracted features that will be an input features to another layer accordingly.

In the convolutional layer, the pre-processed image will be convolved with the optimized Sparse Filters. The purpose of convoluting the image is to extract the image features. From here, the convolved images are produced and a sigmoid activation function is applied on each convolved image. The next component is AVR layer. AVR is inspired from biological system that human eyes do not perceived images in negative values. This layer applies absolute value operation on the extracted features from the previous layer and the output will have absolute value elements. In the next component, the LCN that applied during CNNLS hidden layers is different compared to the LCN applied during the pre-processing process. Both LCNs has similar subtractive and divisive operations except that the input is a set of extracted features from the convolutional layer and different maximum value. The purpose of applying an average pooling after the LCN layer on the extracted features is to ensure that the extracted features are robust to geometric distortion. Thus, the features become less sensitive to a variation in angle and size of a vehicle. The last layer in the hidden layer is the subsampling layer without zero padding. The procedure in this layer is similar to the procedure of resize with the maintained aspect ratio in the pre-processing process.

The components that involved in the post hidden layer are AVR and subsampling layer with zero padding. The same process of AVR in the hidden layer is applied in this layer. The subsampling layer with zero padding is applied on the extracted features to ensure every feature has the same size and has the aspect ratio of 1:1 before being delivered into Softmax Regression classifier. The last process of feature extraction in CNNLS is the fully connected. A fully connected vector is a vector that its elements are in 1-dimensional.

Classifying vehicles is performed by executing a Softmax Regression hypothesis function. The hypothesis is calculated for each vehicle classes to find the amount of probabilities that the extracted features belong to which class. There are two phases of the Softmax Regression implementation; supervised training and testing. The Softmax Regression for the supervised training is trained to produce optimum weights and biases that will be used in the testing phase where it will be loaded and the hypothesis is calculated instead of minimizing the negative log-likelihood. The testing is performed on each vehicle class dataset.
3. Result and Discussion
We test our ATCS with the video captured from a mounted-camera with a frontal view of vehicle in a single lane and during daylight. The distance between the detected vehicle image in the video and the toll booth is 10 meter. Details of database gathering is explained in [13]. The processing time is recorded in classifying the vehicles to observe the efficiency of ATCS. This database is known as SPINT vehicle database. Note that the total samples for each vehicle class in the training dataset is 100 images, thus overall total is 300 images. Table 1. Training and Testing Samples Distribution for Performance Comparison

During our video recording for this experiment, we able to capture 65 vehicles for class 1, 19 vehicles for class 2 and 20 vehicles for class 4 for the testing purpose. Based on that, we produce a confusion matrix to the performance of classification for he vehicle type in the ATCS. We also did a performance comparison in terms of the vehicle type classification using a benchmark data from BIT vehicle database [12]. Note that for SPINT dataset, we used the same testing setup as the ATCS experiment. We summarize both of the experimental setup in table 1 below. The table shows the distribution of dataset in the training and testing for each vehicle class for each database. It shown that the taxi class which is class 4 has fewer samples in the BIT dataset, whereas, the truck which is class 2 has fewer samples in the SPINT dataset. For condition of the vehicle images, BIT provides captured images from frontal angle of surveillance camera during daylight similar to SPINT, and with slight illumination and shadow.

Table 1. Training and Testing Samples Distribution for Performance Comparison

| Dataset | Class 1 | Class 2 | Class 4 |
|---------|---------|---------|---------|
| BIT     | Training 100 | 100 | 100 |
|         | Testing 67 | 21 | 17 |
| SPINT   | Training 100 | 100 | 100 |
|         | Testing 65 | 19 | 20 |

Table 2 shows the confusion matrix result of the average correctly classified percentages and misclassified percentages for both experiments. Looking at the confusion matrix, for ACTS result based on SPINT dataset, class 2 has the highest correctly classified percentage with the average of 99%, followed by class 2 with 98.5% and class 4 with 75%. Thus, the average of correctly classified is 90.83%. Class 2 has outperformed other classes where it has been misclassified as class 1 only with 1.05% and 0% for class 4. For the processing time, we found that the average time to classify one vehicle is 5.8 second. It is reasonably efficient in terms of the classification and display the toll rate process when we consider the distance of camera and toll booth.

Table 2. Confusion Matrix of Vehicle Type Classification

| Actual | Predicted | Class 1 | Class 2 | Class 4 |
|--------|-----------|---------|---------|---------|
|        | BIT       | SPINT   | BIT     | SPINT   | BIT     | SPINT   |
| Class 1| 92.5%     | 98.5%   | 4%      | 6%      | 3.5%    | 0%      |
| Class 2| 4%        | 1.05%   | 90%     | 99%     | 6%      | 0%      |
| Class 4| 11%       | 19%     | 0%      | 0%      | 89%     | 75%     |

Furthermore, using BIT dataset, our approach is able to correctly classify the vehicle class respectively. The most important is class 4 is able to be classified as different class from class 1 in both datasets with 89% and 75%, respectively. However, 11% and 19% of the vehicle in class 4 is misclassified as class 1 in which the highest misclassified rate compared to other classes. Overall, SPINT dataset produced a better result compared to BIT due to the illumination in SPINT dataset is less than BIT dataset.
4. Conclusion

We proposed the implementation of Automated Toll Collection System (ATCS) based on vehicle type classification using deep learning method which is SF-CNNLS to classify the vehicle class in determining the toll rate. The aim is to provide an accurate classification result for ATCS. It also needs to overcome the existing issues of the toll collection system as mentioned earlier. Based on our approach, we were able to automate the classification and toll collection process by classifying the vehicle automatically into class 1, class 2 and class 4. We proved that class 4 can be classified as different class from class 1 which is none of the existing works have done this. As mentioned in section 3, class 1 being misclassified as class 4 and vice versa is significantly high, this is due to lack of unique features to differentiate the class. Thus, the approach need to be improved especially in extracting the features to significantly differentiate the class. We also compare the vehicle type classification performance with the benchmark database. Based on that result, self-obtained database (SPINT) produce a better result compared to the benchmark database (BIT). In future, the processing time need to be increased in order to enhance the efficiency of the ATCS due to some camera is installed in shorter distance to the toll booth, for example, 5 meter. Thus, with our average processing time for one vehicle, this approach is required for an improvement especially in the feature extraction.

Reference

[1] Nagothu S K 2016 Automated toll collection system using GPS and GPRS. International Conference on Communication and Signal Processing (ICCSP) Melmaruvathur pp. 0651-0653.

[2] Tan J Y, Ker P J, Mani D and Arumugam P 2016 Development of a GPS-based highway toll collection system 6th IEEE International Conference on Control, Computing and Engineering (ICCSCE) Batu Ferringhi pp. 125-128.

[3] Suryatali A and Dharmadhikari V B 2015 Computer vision based vehicle detection for toll collection system using embedded Linux International Conference on Circuits, Power and Computing Technologies Nagercoil, pp. 1-7.

[4] Soomro S R, Javed M A and Memon F A 2012 Vehicle Number Recognition system for automatic toll tax collection International Conference of Robotics and Artificial Intelligence Rawalpindi pp. 125-129.

[5] Subaweh M B and Wibowo E P 2016 Implementation of Pixel Based Adaptive Segmenter method for tracking and counting vehicles in visual surveillance International Conference on Informatics and Computing (ICIC) Mataram, Indonesia, pp. 1-5.

[6] Kamkar S and Saifabakhsh R 2016 Vehicle detection, counting and classification in various conditions IET Intelligent Transport Systems, vol. 10, no. 6, pp. 406-413.

[7] Bautista, C. M., Dy, C. A., Manalac, M. I., Orbe, R. A., and Cordel, M. 2016. Convolutional neural network for vehicle detection in low resolution traffic videos. In 2016 IEEE Region 10 Symposium (TENSYPYMP), pages 277–281.

[8] Valiere, P., Khoudour, L., Crouzil, A., and Cong, D. N. T. 2015. Robust vehicle counting with severe shadows and occlusions. In International Conference on Image Processing Theory, Tools and Applications (IPTA), Orleans, pp. 99-104.

[9] Prommool, P., Auephanwiriyakul, S., and Theera-Umpon, N. 2016. Vision-based automatic vehicle counting system using motion estimation with Taylor series approximation. In 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), Penang, Malaysia, pp. 485-489.

[10] Quesada, J., and Rodriguez, P. 2016. Automatic vehicle counting method based on principal compo-nent pursuit background modeling. In IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, pp. 3822-3826.

[11] Zhang, Z., Liu, K., Gao, F., Li, X., and Wang, G. 2016. Vision-based vehicle detecting and counting for traffic flow analysis. In International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, pp. 2267-2273.

[12] Dong, Z., Pei, M., He, Y., Liu, T., Dong, Y., and Jia, Y. 2014. Vehicle type classification using unsupervised convolutional neural network. In 22nd IEEE International Conference on Pattern Recognition(ICPR), pages172–177.

[13] Awang S., and Azmi N.M.A.N. 2018. Vehicle Counting System Based on Vehicle Type Classification Using Deep Learning Method. In: Kim K., Kim H., Baek N. (eds) IT Convergence and Security 2017. ICITS 2017.Lecture Notes in Electrical Engineering, vol 449. Springer, Singapore.