The fourth Chilean Workshop on Pattern Recognition (CWPR'2012), 12-16th November 2012, Valparaiso, Chile.

Confidence Based Active Learning for Vehicle Classification in Urban Traffic

Zezhi Chen, Tim Ellis, *Member, IEEE*, and Sergio A Velastin, *Senior Member, IEEE*

Abstract— **This paper presents a framework for confidence based active learning for vehicle classification in an urban traffic environment. Vehicles are automatically detected using an improved background subtraction algorithm using a Gaussian mixture model. A vehicle observation vector is constructed from measurement-based features and an intensity-based pyramid HOG. The output scores of a linear SVM classifier are accurately calibrated to probabilities using an interpolated dynamic bin width histogram. The confidence value of each sample is measured by its probabilities. Thus, only a small number of low confidence samples need to be identified and annotated according to their confidence. Compared to passive learning, the number of annotated samples needed for the training dataset can be reduced significantly, yielding a high accuracy classifier with low computational complexity and high efficiency. The detected vehicles are classified into four main categories: car, van, bus and motorcycle. Experimental results demonstrate the effectiveness and efficiency of our approach. The method is general enough so that it can be used in other classification problems and domains, e.g. pedestrian detection.**

I. Introduction

Traffic monitoring is an important tool in the development of intelligent transport systems (ITS) involving the detection and categorisation of road vehicles. Such monitoring can support the assessment of a range of needs: traffic volume and speed estimation, flow and congestion control, incident detection, usage type, queue lengths, illegal manoeuvres etc. Applying image processing technologies to vehicle detection and classification has been a hot focus of research in ITS over the last decade. Urban traffic flow analysis is a challenging problem under high vehicle densities which can result in frequent occlusion. Several problems have to be solved, ranging from low and middle level vision tasks, such as the detection and tracking of multiple moving objects in a scene, to high level analyses, like vehicle classification. Classification of road user type is essential for some tasks, and beneficial for others.

Moreover, data collection and annotation is a crucial part of vehicle classification system development because it determines the success of later stages. Data collection and annotation is surprisingly time consuming and costly. The widely used approach for data collection and annotation is called passive learning (PL), where samples are randomly and independently selected from the underlying distributions; human assessors then manually annotate these samples. Considering the time and cost associated with this process, it is often the case that there are insufficient training samples to assure a certain level of performance after training. Active learning (AL) may be a suitable approach to minimize the effort of annotation [1]. In active learning, the learning process repeatedly queries unlabeled samples to select the most informative samples to annotate and update its learned rules. Therefore, unnecessary and redundant annotation is avoided, greatly reducing the annotation cost and time. Active learning is also helpful in reducing computational complexity. Active learning has been shown to be more powerful than learning from random examples [2][3].

This paper uses a improved confidence-based active learning (CBAL) to train an SVM to classify vehicles into four dominant categories: car, van (van, minivan, minibus and limousine), bus (single and double decked) and motorcycle (motorcycle and bicycle). The approach takes advantage of current well-developed classifiers' probability preserving and ordering properties [4], calibrating the output scores to the class-conditional error. Thus, it can estimate the uncertainty level of each sample according to the output score of a classifier and select only the most informative samples for annotation whose output scores are in the uncertain range. From SVM theory we know that only support vectors play a role in SVM learning and the removal of non-support vectors does not change the training results. Support vectors should be located close to the boundary between the two classes. This means that if we train the classifier using only low confidence data, we will obtain similar training results. The approach is general enough to be used with other classifiers.

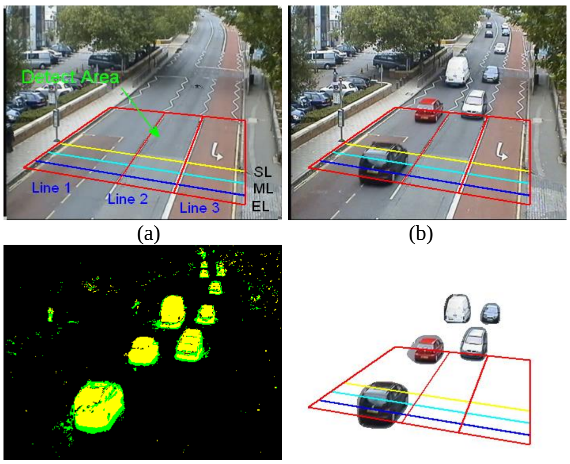
The novelty of the paper includes: a) a robust automatic vehicle detection strategy; b) a CBAL that has been extended from binary classification to multiple classes classification (and originally applied to vehicle classification). The main contributions of the paper are: 1) high accuracy of calibrated probability obtained using smooth interpolated dynamic bin width, that can reflect the underlying probabilities; 2) effective selection of the most informative samples from an unlabeled training data set; 3) dramatic reduction in the number of training samples needing annotation; 4) an approach that minimizes complexity and training times.

II. Vehicle Detection

There are several key considerations when implementing a vehicle detection algorithm, and they vary depending on the specific task. For traffic flow statistics, it is essential to count each vehicle only once. To ensure that vehicles will only be counted as they appear in the detection zone, a virtual loop detector is applied. The virtual loop is comprised of three detect lines, StartLine (SL), MiddleLine (ML) and EndLine (EL). These line detectors are sensitive to miss-detection as a consequence of the ragged edge of a vehicle boundary. To minimize this effect the detectors have a finite width to ensure

Digital Imaging Research Centre, School of Computing and Information Systems, Kingston University, Kingston-upon-Thames, UK. (e-mail: {z.chen, t.ellis@kingston.ac.uk}, sergio.velastin@ieee.org).

a stable detection of the vehicle when it intersects the line (a width of 5 pixels was used in the experiments described later). The separation between detector lines depends on average traffic speed (higher speed, require a larger threshold), and was set to 30 pixels in our experiments. The traffic speed limitation is 30 miles per hour. The detector is configured to operate in both directions, to accommodate the two directions of traffic flow, and should be placed at a location where vehicles are clearly visible with minimal occlusion, i.e. usually closest to the camera. A detector is allocated to each lane to handle the measurements for each traffic stream.



(a) (b) (c) (d)

Fig. 1. Vehicle detection: (a) background GMM and virtual loop detector; (b) current input image with detection lines. (c) background subtraction results: modeled background (black), foreground object (yellow), shadow (green) or reflection highlights (red); (d) foreground image created by extracting the pixels from the original frame using the final foreground object mask.

An Automatic Vehicle Detection System (AutoVDS) is constructed from background subtraction using a Gaussian mixture model. Fig.1 illustrates the object detection procedure. Shadow, road reflection and reflection highlights pixels are minimised, followed by a post-processing binary morphological opening to remove noise and small area objects [5]. To ensure that vehicles are only counted once the detector considers a vehicle to be "present" only when both SL and ML are occupied and EL is unoccupied (for traffic moving towards the camera, i.e. lane 2 and 3). A vehicle is said to be "leaving" when ML and EL are occupied and SL unoccupied. A vehicle is counted only when it changes from the "present" state to the "leaving" state. This is reasonable in congested situations and even stationary traffic. In this way, the detector will not over-count in either case. If the proportion of pixels intersecting the detection line is above a threshold (30% of the lane width), the line is considered occupied, otherwise it is unoccupied. This threshold is chosen as a tradeoff between detecting small vehicles (such as bicycles and motorbikes) but being insensitive to small blobs associated with noise. It is only necessary to swap SL and EL to account for vehicles in the traffic stream moving away from the camera (e.g. lane 1 in Fig.1(a)).

III. FEATURE EXTRACTION

i) Measurement-based feature A set of 13 measurement-based features (MBF) that are cheap to compute and store are used to build a vehicle feature database. The feature vector is comprised of measures of size and shape from the binary silhouette and encompassing bounding box (width, height, and area), circularity (dispersedness, equivdiameter), ellipticity (length of major and minor axis, eccentricity), and shape-filling measure (filled area, convex area, extent, solidity) [6].

ii) Intensity based feature To improve the accuracy of classification, we investigate a potentially effective feature based on a pyramid of histogram of gradient orientations (PHOG). PHOG was first proposed by Bosch et al. [7] and has been successfully applied to object recognition, human expression recognition and image classification [8]. As a spatial shape descriptor, it can represent the statistical information of global shape and local shape (in a sub-region), which is effective for object recognition. The local shape is captured by the distribution over edge orientations within a region, and the spatial layout by tiling the image into regions at multiple resolutions. The descriptor consists of a histogram of orientation gradients over each image sub-region at each resolution level of the bounding box detection. For vehicle classification, we set 3 levels and 9 orientation bins, evenly spaced over 0° - 360°, extract the PHOG features and normalise them for each level. The number of levels and orientations are the optimal numbers for our experiments. The dimension of the resulting vector is constructed by concatenating these into an $9+4\times9+4\times4\times9=189$ element vector. For local shape representation there are two kinds of PHOG available. One is the edge-based PHOG (EPHOG), which is represented by a histogram of edge orientations within an image region and its sub-region. The other is the intensity-based PHOG (IPHOG), which is represented by the distribution of local intensity gradients, without precise knowledge of the corresponding edge point. Extensive experimental results have previously shown [9] MBF+IPHOG to be the best feature combination for vehicle classification.

IV. CONVERTING SVM SCORES INTO PROBABILITIES

The output of a classifier should be a calibrated posterior probability to enable active learning. Standard SVMs do not provide such probabilities. The SVM output score is not a probability but a distance from the separating hyperplane. The sign of the score indicates if the example is classified as positive or negative. The magnitude of the score can be taken as a measure of confidence in the prediction, since examples far from the separating hyperplane are presumably more likely to be classified correctly. Thus, we need a way to transform SVM output scores to probabilities for CBAL. Platt [10] presented an algorithm to train an SVM, then train the parameters of an additional sigmoid function to map the SVM outputs into probabilities.

Drish [11] has proposed a binning method for a probability estimation problem in one dimension. For the fixed-bin-width allocation method, this involves sorting the training examples according to their scores, and then dividing them into b equal sized bins, each having an upper and lower bound. Given a test example x, it is placed in a bin according to its score. The corresponding probability $P(j=1|x)$ is the fraction of positive training examples that fall within the bin. A difficulty of the binning method is that the number of bins has to be chosen by cross-validation. A large bin width will produce a smooth histogram with too little detail; on the other hand, a very small bin width will result in a jagged histogram and a small number of samples in each bin will make too large a contribution. Ideally, the width of bins is chosen so that the estimated probability reflects the true underlying probability distributions without giving too much credence to the dataset at hand. Instead of equal bin widths, Li and Sethi [2] presented a method using an equal number of samples in each bin, called dynamic bin-width (DBW) allocation. This gives a smooth histogram where conditional probabilities are small and it will also give more detail where conditional probabilities are large. In other words, it adapts to the underlying probability distribution.

In order to further improve the accuracy of converting SVM scores to probabilities using DBW, an smooth interpolated DBW histogram (SIDBW) is proposed in this paper. The experimental results demonstrate the high accuracy of the new algorithm.

V. EXPERIMENTS

The CBAL used here is based on the algorithm proposed by Li and Sethi [2]. The high accuracy SIDBW allocation strategy is used to convert SVM scores to probabilities.

A. Covert SVM scores to probabilities

Two algorithms used to calibrate probabilities from linear SVM scores are been compared here: the DBW histogram algorithm and the SIDBW histogram algorithm. Mean square error (MSE) is used to calculate the accuracy of the conversion methods. The squared error (SE) is defined as

$$SE = \sum_j \left(t(j \mid x) - p(j \mid x)\right)^2 \tag{1}$$

where $p(j|x)$ is the probability estimated by the method for example x and class j, and $t(j|x)$ is the true probability of class j for x. For data sets where true labels are known and the probabilities are unknown, $t(j|x)$ is defined to be 1 if the label of x is j and 0 otherwise.

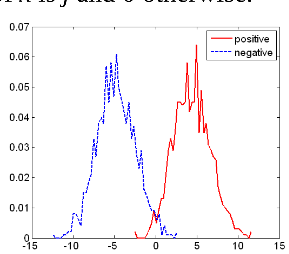


Fig. 2. The histogram for $p(f|y=\pm 1)$ for SVM on synthetic data.

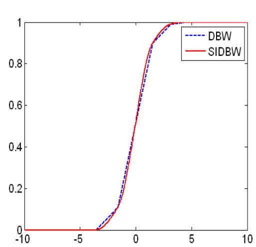


Fig. 3. Histograms of probabilities from DBW and SIDBW.

An experiment uses synthetic examples with a Gaussian distribution (GD) to compare the accuracy of the algorithms. 2000 2D points are created with two Gaussian distributions, $x \in N(-0.3, 0.3)$ and $y \in N[-0.5, 0.5]$. Fig.2 shows the data distribution. The solid red line is $p(f|y=+1)$, while the dashed blue line is $p(f|y=-1)$. A 10-fold cross validation strategy has been used to estimate the best parameters for a linear SVM classifier. The entire data have been used to train the SVM with the best parameters. The classification accuracy is 97.80%. The SVM scores were converted to probabilities. Fig.3 shows the histograms of probability estimation from DBW and SIDBW. The cubic interpolation algorithm is used in the SIDBW implementation. The MSE of probability estimation using DBW and SIDBW from SVM scores of training are 0.0198 and 0.0036, respectively. Obviously, SIDBW improves the accuracy of probability estimation.

B. CBAL with known probabilities

Using the classifier's output calibrated probabilities as a confidence measure, the query function is:

$$Q(x) = \begin{cases} 1 & T_1 < f(x) < T_2 \\ 0 & otherwise \end{cases} \tag{2}$$

where $f(x)$ is the output probabilities from the classifier. The thresholds T_1=0.2 and T_2=0.95 are chosen for the experiment, with 109 low confidence samples (5.45% of training data). 83.78% support vectors were included in the set of low confidence samples. Only the low confidence (the most informative) samples are used to train a classifier, and the classifier is then used to classify the entire training data. Comparison of accuracy, recall, precision, and F1 for passive learning and active leaning are presented in Table I.

TABLE I. THE COMPARISON RESULTS OF PL AND AL

	accuracy	recall	precision	F1
PL	0.9780	0.9800	0.9761	0.9780
AL	0.9785	0.9800	0.9771	0.9785

C. CBAL with unknown probabilities

The same synthetic dataset is use to evaluate the proposed CBAL, but we assume the dataset is unlabeled. 50 samples are randomly chosen as an initial training set to train the linear SVM classifier. The accuracy of the entire training data set is 96.40%. Obviously, this is not as good as the accuracy from all the 2000 training samples (PL). After converting the SVM scores to probabilities, 43 low confidence (most informative) samples are selected from the remaining 1950 unlabeled samples. In the next round, 93 samples (the 50 initial samples plus the 43 newly selected samples) are used to form the training set. The accuracy increases to 98.25% compared to the passive learning accuracy of 97.80%. This means that only 93 samples are used to achieve the same (or better) training results as those from all 2000 samples. The proportion of the training samples is only 4.65%. Fig.4 illustrates the processing of active learning for the synthetic dataset. From the plot it can be seen that the active learning algorithm can incrementally choose the informative samples and correct the training error from the previous incomplete training.

Foreground blobs of vehicles obtained using AutoVDS are used to evaluate the active learning algorithm. For multi-class passive learning, the entire data set has been used to train the SVM system, taking 20.43 hours on a 2.39GHz Pentium laptop. The number of support vectors is 441. The ratio of the number of support vectors over the sample size is 22.05%. This ratio reflects the classification complexity of the training data. The ACC is 97.51%.

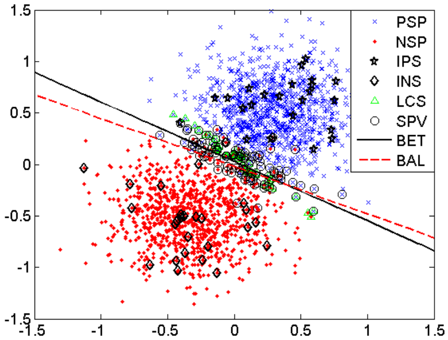


Fig. 4. AL and PL processing for GD. PSP: positive samples; NSP: negative samples; LCS: selected low confidence samples; IPS: Initial random positive samples; INS: initial negative samples; SPV: support vectors from entire training dataset; BET: classification boundary from entire training dataset; BAL: final classification boundary of active learning.

For multi-class active learning, firstly three binary classifiers (motorcycle vs car, car vs van and van vs bus) are trained separately to calibrate their SVM scores to probabilities. For each classifier, 50 samples are randomly selected to form the initial training set. In each round of query, a maximum 25 additional most informative samples are selected from the unlabeled sample pool according to the probability (if more than 25 most informative samples are obtained from the query function (2), 25 samples are randomly selected from them), and added into the training set. Using these most informative samples, selection training is repeated until reasonable classification accuracy is achieved, or for a maximum of 10 rounds. Then a multiple classifier is trained using the selected samples sequentially. In order to test the stability of AL for real data, the program was run 10 times. The variation of mean and *std* of ACC, and the corresponding average number of training samples of each round is illustrated in Fig.5. The figure shows that when the number of training samples increases, the mean of ACC increases and *std* of ACC decreases gradually, which means that the stability of AL increases accordingly. In round 10, 440 training samples (290 of the most informative samples plus 150 randomly selected initial samples) are selected to train the classifier. The classification model is used to classify the whole real dataset, and the mean of ACC is 97.57%. The confidence-based active learning procedure is terminated since the ACC is higher than that for passive learning which is 97.51%. The mean training time is just 31 minutes, 40 times faster than passive learning. In addition, only 20.66% of the real data needs to be annotated for training the classifier.

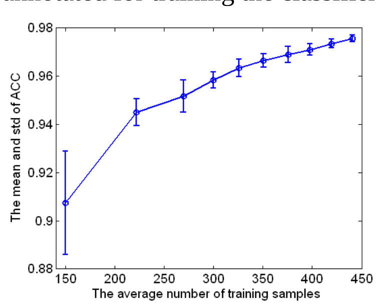


Fig. 5. Variation of classification accuracy.

VI. CONCLUSIONS

This paper has proposed a confidence-based active learning approach for vehicle classification in urban traffic. High accuracy probability estimation is obtained from linear SVM scores using the smooth interpolated dynamic bin-width histogram. Only low confidence samples are used to train the classifier. This can dramatically reduce the number of annotated samples required for training, as well as reducing the overall training time and classification complexity. Experiments on synthetic and real data demonstrated the effectiveness of the approach. Compared with passive learning, active learning required only 4.65% of the training samples to achieve the comparable or improved classification accuracy for a synthetic dataset. For a multi-class classification task with a real, high-dimensional observation dataset, only 20.66% annotated samples were used to achieve superior classification results, with a computational improvement of some 40 times faster than that of using the entire dataset to train the classifier.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the Royal Borough of Kingston (UK) for providing access to the video data.

REFERENCES

[1]. D.D. Lewis and J. Catlett. Heterogeneous uncertainty sampling for supervised Learning. *Proc. 11th International Conference on Machine Learning*, pp. 148-156, 1994.

[2]. M. Li and I. K. Sethi. Confidence-based active learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 8, pp. 1251–1261, 2006.

[3]. S. Sivaraman. and M. Manubhai. A general active-learning framework for on-road vehicle recognition and tracking. *IEEE Transaction on Intelligent Transportation Systems*, 11(2), pp. 267-276, 2010.

[4]. B. Zadrozny and C. Elkan. Transforming classifier scores into accurate multiclass probability estimates. *Proceedings of the eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002.

[5]. Z. Chen and T. Ellis. Self-adaptive Gaussian mixture model for urban traffic monitoring system. *IEEE International Conference on Computer Vision Workshop*, Barcelona, Spain, pp. 1769-1776, 2011.

[6]. Z. Chen, T. Ellis and S. A. Velastin. Vehicle type categorization: A comparison of classification schemes. *14th IEEE Annual Conference on Intelligent Transportation Systems*, the George Washington University, Washington, DC, USA. pp. 74-79, Oct. 5-7, 2011.

[7]. A. Bosch, A. Zisserman, X. Munoz. Representing shape with a spatial pyramid kernel. In: *Proceedings of the ACM International Conference on Image and Video Retrieval*, 2007.

[8]. X. Han, Y. Chen, X. Ruan. Image recognition by learned linear subspace of combined bag-of-features and low-level features. In: *Proceedings of IEEE International Conference on Image Processing*, 1049-1052, 2010.

[9]. Z. Chen, T. Ellis. Multi-shape descriptor vehicle classification for urban traffic. In: *International Conference on Digital Image Computing: Techniques and Applications*, pp. 456-461, Noosa, Australia, Dec 6, 2011.

[10]. J.C. Platt. Probability outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, MIT Press, 61-74, 1999.

[11]. J. Drish. Obtaining calibrated probability estimates from support vector machines. *Technique Report*, Department of Computer Science and Engineering, University of California, San Diego, CA, 2001.