VividhaVahana: Smartphone Based Vehicle Classification And Its Applications in Developing Regions

Shilpa Garg, Pushpendra Singh, Parameswaran Ramanathan, Rijurekha Sen
shilpa1249@iiitd.ac.in, psingh@iiitd.ac.in, parmesh@ece.wisc.edu, rijurekha@mpi-sws.org

Abstract

Developing region road traffic has a unique characteristic of high heterogeneity in vehicle types. In this paper, we describe VividhaVahana[1]*, a smartphone sensor based system to categorize road vehicles into four predominant categories: two-wheeler bikes, three-wheeler auto-rickshaws, four-wheeler cars and public transport like buses. Using a variety of sensor based features, our system is able to achieve above 90% classification accuracy, evaluated over 1500+ Km of driving data, on two urban road stretches in the Indian city of Delhi.*

We also apply VividhaVahana to empirically examine four representative transport applications, namely travel time estimation, driving behavior detection, traffic state classification and road surface monitoring. We show how each of these applications would benefit from a vehicle class specific analysis, compared to the vehicle agnostic analysis as has been done in the past. Our work gives useful insights on how such applications can be re-designed, to better fit developing region traffic characteristics and requirements.

1 Introduction

Road traffic problems are prevalent in all parts of the world. The issues are exacerbated in developing regions and growing economies like India, where infrastructure growth rate does not match the growth rate in urban population size and number of vehicles. Traffic congestion regularly increases exposure to pollution and fuel consumption and causes unpredictability in travel times. Poor conditions of road surfaces and irregular driving cause accidents. Intelligent Transport Systems (ITS), are systems to monitor traffic and road conditions, and disseminate useful information to citizens, in an attempt to alleviate some of these issues.

Though a number of ITS solutions have been built for lane-based traffic in developed countries [1, 2, 3, 4, 5], and increasingly more for non-laned traffic in developing regions [6, 7, 8, 9, 10, 11], an important characteristic of developing country traffic has been overlooked till date. This is the *heterogeneity* of vehicles, where two-wheeler bikes and scooters, three-wheeler auto-rickshaws, four-wheeler cars and larger public transport like buses, ply on the same road. Fig. 1 and Fig. 2 give a visual comparison of the lane based homogenous traffic in developed countries, vs. the non-laned heterogeneous traffic in developing regions.

Vehicle heterogeneity is probably a combined artifact of absence of proper public transport and high income disparity among people in developing regions. The former motivates people to use some form of personalized transportation, and the latter regulates the purchasing power for the same. Thus cheap two-wheelers and enormously expensive sports cars ply the same road. In the absence of personal vehicles – buses, auto-rickshaws and taxicabs are used by people with increasing purchasing power in that order. Fig. 3 shows an example of the high proportion of two and three-wheelers, which is a common sight on Indian roads.

This phenomenon of vehicle heterogeneity has some direct consequences for ITS solutions. The different vehicles have very different physical and mechanical characteristics, and this causes them to behave differently in similar traffic situations. Two-wheelers can maneuver much more freely in congestion, compared to bigger vehicles. Auto-rickshaws can attain much less maximum speed on an empty road, compared to cars. Thus traffic monitoring systems should consider how to incorporate this non-uniformity, in deciding thresholds for congestion detection or irregular driving detection. Travel time and route prediction softwares should consider the vehicle specific characteristics, possibly giving different predictions for different vehicle classes. The current ITS systems being vehicle class agnostic, these intuitively important aspects have not been handled so far, as we discuss in more detail in Section. 2.

In this paper we present *VividhaVahana*, a system to classify different vehicle categories, based on smartphone sensor data. We examine a wide range of sensor features and classification algorithms in this context, to classify vehicles into four categories - a) two-wheeler bikes, b) three-wheeler auto-rickshaws, c) cars and d) buses. Our heuristics achieve over 90% accuracy on real road data, collected over 1500+ Km of driving, on two different road stretches, using multiple smartphone models.

We also apply *VividhaVahana* to identify four representative ITS applications, that would benefit from vehicle class

[1]The name is a direct *Hindi* translation for the two key words heterogeneous and vehicles - heterogeneous, translated as *Vividha* and vehicles, translated as *Vahana*.

Figure 1. Lane based homogenous traffic

Figure 2. Non-laned heterogeneous traffic

Figure 3. High proportions of bikes (2-wheelers) and autos (3-wheelers)

specific analysis. The first is travel time estimates, where we empirically show different vehicle categories take different times to travel from the same source to the same destination, following the same route under the same traffic conditions. This is a necessary motivation for vehicle specific travel time maps, or route prediction softwares. The second is driving pattern detection, where we do a micro-level analysis of how vehicles classes behave differently in similar traffic situations. This might motivate much more nuanced modeling of traffic flow to build more realistic models and predictors of traffic signal clearance cycles, or better road designs.

A third ITS application examined is detection of traffic states like congestion vs. free-flowing, where crowdsourcing different vehicle classes need the information to be assimilated in a more intelligent way than uniform sampling and averaging. Training the traffic classification thresholds on slower vehicles like public transport, will cause the system to have high false negatives when the test data comes from faster moving two-wheelers. More false positives will occur in the converse use of train and test data. Finally, we also examine the different signatures that different vehicles generate on same road surfaces, which motivates vehicle specific road surface monitoring applications to be designed.

The rest of the paper is organized as follows. Section 2 discusses the related work in this context, identifying important differences between existing literature and this work. Section 3 describes the sensor data and ground truth collection details, followed by the overall system architecture in Section 4. The algorithmic details and evaluation of vehicle classification are given in Section 5. Section 6 describes the four representative ITS applications, that we empirically show to depend on vehicle categories. We discuss some related issues and future work in Section 7, and finally conclude in Section 8.

2 Related Work

ITS solutions are an active area of research, as well as commercial product development and deployment. The four classes of ITS applications examined in this paper, all have several precedents in literature. The first application of travel time estimation has been extensively examined with GPS and cellular data [1, 2, 3]. The second application involves micro-analysis of driving patterns and its possible use in building traffic models. Driving behaviors have been examined using smartphone sensors [10, 12, 13] and traffic models to relate the fundamental transportation parameters like density, speed and flux [14] have also been studied [11, 15, 16, 17, 18].

The third application of traffic state classification as congested vs. free-flowing, has been examined by researchers [4, 6, 7, 8, 9], as well as by commercial products [19, 20]. These have been based on different inputs either from (a) static road-side sensors – video or images from cameras [6], RF signals from wireless radios [7], audio signals from microphone [8] or (b) from probe sensors on vehicles – GPS [4] or smartphone sensors [9]. Finally smartphone accelerometer based road surface monitoring has also been examined [5, 21].

However, though several solutions exist for each of the four application classes, they are all vehicle category agnostic. This is understandable in the subset of solutions designed for laned homogenous traffic in developed countries as in [1, 2, 3, 13, 4, 5]. There, the absence of this specific traffic characteristic of vehicle heterogeneity, retains the utility of the solutions. However, the solutions developed for non-laned heterogeneous traffic of developing regions like [10, 11, 15, 16, 17, 6, 7, 8, 9, 21], will definitely improve by factoring in the vehicle non-uniformity artifacts. We will empirically show the dependency of the applications on vehicle category, to elaborate on this in Section 6.

Vehicle classification has been done in a context different from ITS applications, in the field of human context monitoring, activity recognition and human mobility modeling [22, 23, 24, 25]. Though we examine all the smartphone sensor features discussed in these works during our design of vehicle classification heuristics in Section 5, we differ from these works in two important ways. Firstly, these works originating from developed countries, have much fewer and more easily distinguishable vehicle categories like train, tram, bus and metro [22], where the vehicle forms and travel zones or tracks are quite different. Applying the feature sets to classify vehicles that ply on the exact same road stretches, and are yet different from each other, is a contribution of this paper. Secondly, as the application domains of those works were different, the link between vehicle classification and ITS solutions, that we explore in Section 6, is missing. We seek to bridge this gap between vehicle classification literature and ITS literature.

Finally, there is one system [26], which does vehicle classification for developing regions, and therefore handles the same set of diverse vehicles as we do. However, this solution

uses video image processing on data from road side cameras. Thus though vehicle classification works, the ITS applications [26] can handle, are a subset of what *VividhaVahana* is capable of, as travel time estimation and driving behavior micro-analysis, are only feasible with in-vehicle probe sensing. Our smartphone based system has this important advantage over any static sensor based solution. Secondly, our sensors based schemes are much less computation intensive than video or image processing, thereby resulting in a simpler solution for the vehicle classification problem.

3 Data Collection

We collected smartphone sensor data from two driving stretches in the Indian city of Delhi. The first is a 30+ Km driving stretch along Ring Road, from Punjabi Bagh to Lajpat Nagar (route shown in Fig. 4). The second is a 10+ Km stretch inside an educational institute campus. The first road, being one of the main arterial roads of Delhi, provided an uncontrolled experimental environment, where traffic situations and presence of other vehicles were natural and realistic urban phenomena. The second stretch being a road inside an educational campus, gave us more experimental control. We will refer to these two road stretches as *city-road* and *campus-road* henceforth.

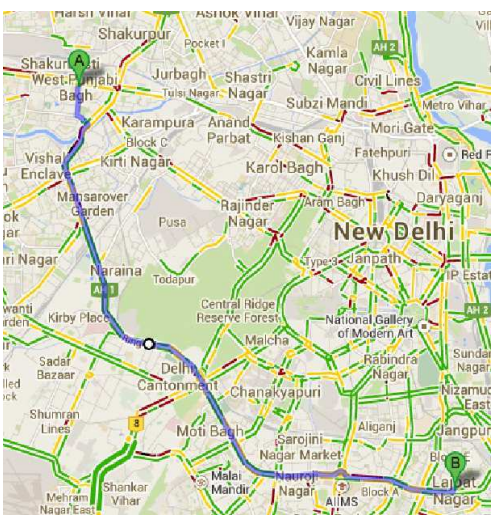

Figure 4. The 30 Km long Ring Road route from (A) Punjabi Bagh to (B) Lajpat Nagar

On *campus-road*, data was collected using the following three vehicle types: car (Tata Indica), 2-wheeler bike (Hero Honda CBZ extreme/ Honda Activa) and three-wheeler auto-rickshaw (standard model driven in Delhi). Each of the three vehicle types were driven for 10 days with 10+ km. each day, generating 300+ Km data in all. To differentiate the vehicles, while the road and traffic conditions remained same, these vehicles were started at the same time from the same source, and they were driven towards the same destination, along the same route. Data was collected following the same methodology on the *city-road*, from the above three vehicle types, and additionally from buses. The four vehicle types were driven over 10 days with 30+ km each day, generating 1200+ Km data overall. The details of the 1500+ Km of sensor data collected overall are summarized in Table 1.

Road	Vehicle classes	Km per day per class	Days	Total Distance (Km)
Ring Road (Punjabi Bagh to Lajpat Nagar)	bike, auto, car, bus	30+	10	1200+
Educational campus	bike, auto, car	10+	10	300+

Table 1. Details of driving experiments

The smartphones were carried by different people in the four vehicle categories, to prevent user related bias in data collection. Also other than the start time, source, destination and route, nothing else was specified to the smartphone carriers. To capture their natural behavior, they were urged to either drive or travel in public transport, as they would have normally done in a non-experimental trip scenario.

Accelerometer, magnetometer, GPS, gyroscope, orientation, light and microphone sensors were sampled using LG Optimus 4x, Google Nexus 4 and Samsung Galaxy Ace phones. The phones were placed in the front pocket of the trouser in each case. The intuitions behind why these particular sensors were sampled, and what features were extracted from each type of sensor data, are detailed in Section 5. The sensors were sampled in the *UI* mode, with average sampling frequencies of 18 Hz for each sensor. Higher sampling rates could be obtained using the *Game* and the *Fastest* modes. But we chose not to use them for reasons of saving smartphone battery, and as our features gave good classification accuracies even at this reduced sampling rate. The sensors, phone models and sampling frequencies are summarized in Table 2. The sampled data was continuously sent to a backend server over a data connection, and logged in the server for offline analysis.

Sensors	Phone models	Sampling frequencies
accelerometer, gyroscope, orientation, GPS, magnetometer, light, microphone	LG Optimus 4x Google Nexus 4 Samsung Galaxy Ace	UI mode 18 Hz

Table 2. Details of sampled sensors and phone models

Ground truth about traffic situations and road conditions were manually noted and later verified using the Google traffic data. Locations of traffic signals were noted from Google Maps. Special cases of traffic situations resulting from one-off events like wedding processions, political rallies and accidents were separately noted down. Though the vehicle

classification evaluation depends only on the vehicle type ground truth, these additional ground truth information on traffic and road conditions have been used in Section 6, to understand how the different vehicle categories reacted to the different traffic and road conditions.

4 VividhaVahana Architecture

Fig. 5 shows the envisioned architecture for *VividhaVahana*. It resembles most existing ITS system architectures with smartphone clients on the roads and one or more ITS application servers residing on the cloud. The smartphones send sensor data to the cloud (shown by solid arrow in Fig. 5), where the ITS application servers process the individual client data, aggregate data from multiple clients and generate useful services and information. These outputs are either sent back to the client devices (shown by dotted arrow in Fig. 5) or utilized by the metropolitan traffic control authorities for traffic management or urban planning (not shown in the figure).

This paper introduces a new server in the cloud, termed as the *VividhaVahana* server, which also receives the same sensor data from the client smartphones, and processes it to generate vehicle category information. This output is fed to the traditional ITS application server. The application server uses this additional category information in its computing, to generate more nuanced ITS information.

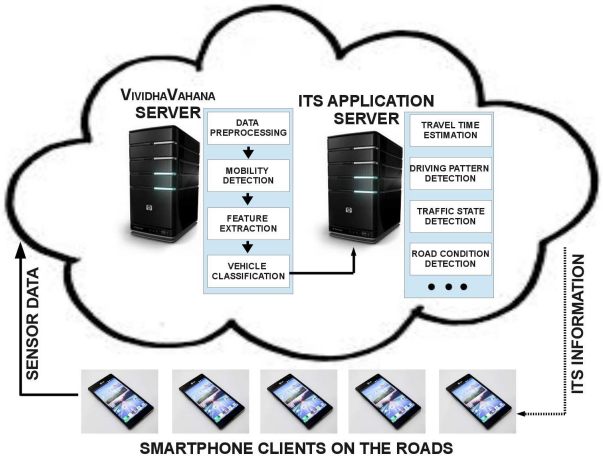

Figure 5. VividhaVahana envisioned architecture

The workflow inside the *VividhaVahana* server, is also outlined in Fig. 5. First the sensor data is pre-processed to clean missing or additional samples due to sampling frequency quirks, which sometimes happen on android phones. Then vehicle mobility is detected, as vehicle classification can happen only for moving vehicles. We detect the standard deviation of acceleration magnitude, to exceed some threshold for consecutive time windows, to detect mobility. This is a standard method of mobile vs. stationary disambiguation in literature [22].

The third step in the *VividhaVahana* workflow is the extraction of suitable features for vehicle classification, from appropriate time windows of data. The fourth and final step is running the vehicle classification algorithm on these extracted features. An analysis of what features and what algorithms give what kind of classification accuracy will be detailed in Section 5.

The four classes of ITS applications, that have been shown to depend on vehicle categories, are outlined for the ITS application server in Fig. 5. These will be described in detail in Section 6. There can be many more ITS applications running on such ITS application servers, which may or may not be dependent on vehicle category. The final services or information sent back to the mobile clients, have not been implemented in this paper, and hence that arrow has been shown as a dotted line in Fig. 5.

5 Vehicle Classification

We describe our vehicle classification system in details in the section. We begin by giving an intuitive idea about why we choose certain sensors to be sampled from the smartphones on the vehicles, reasoning about the potential of those sensors for vehicle classification. We then present the concrete set of features extracted from those sensor streams, the algorithms applied on the features for vehicle classification, and finally the evaluation to compare different features and algorithms.

5.1 Sensor Features

Among the vehicles which ply on the same road stretches, we choose (a) two-wheelers or bikes, (b) three-wheelers or auto-rickshaws, (c) cars and (d) buses, as the four target vehicle categories to disambiguate. These are the four predominant vehicle classes, that carry people on roads of developing regions.[2] Due to the experimental overhead of collecting adequate data for each category, we restrict ourselves to these four broad and predominant categories. More categories or sub-categories might be explored further in future, using the methodology described in this paper.

Having defined the set of target labels, we next explore why these categories should be differentiable in an automated way, and what sensors might help in the disambiguation task. The four vehicle categories, are visibly different in their make and form factors. While bikes and auto-rickshaws are smaller and have less cushioned design, cars and buses are typically larger with much better shock-absorbing facilities. Thus when these vehicles travel on a road stretch, they might exhibit different mobility characteristics, because of these design differences.

Fig. 8 shows the average speed along y-axis vs. time along x-axis, for a free-flow drive by the four vehicle types, on a smooth road. The speed is computed from the accelerometer magnitude ($\sqrt{{a_x}^2 + {a_y}^2 + {a_z}^2}$), where a_x, a_y

[2]There might be more vehicle categories other than these four, that ply the same road stretch, like goods trucks and non-motorized vehicles like bicycles and cycle-rickshaws. There might also be sub-categories within each broad category, like motorbike vs. ladies' scooters in the two-wheeler category or high end vs. low end cars.

Figure 6. Spectrogram of bus engine noise

Figure 7. Spectrogram of auto engine noise

and a_z are the accelerations measured in the different axes of a 3-axis accelerometer. We plot speed instead of acceleration to present a smoother curve for aid of visual comparison. As can be seen, the bike and auto curves have much more fluctuations, than the smoother car and the bus curves. This shows that different amount of jerks are experienced by these vehicle classes, because of their design differences. It is important to note that we take the worst case of free-flow traffic and smooth road in this example. Heavy traffic might cause the smaller bikes and autos to maneuver more erratically or rough roads can cause them to vibrate more, thereby enhancing these characteristic signatures further.

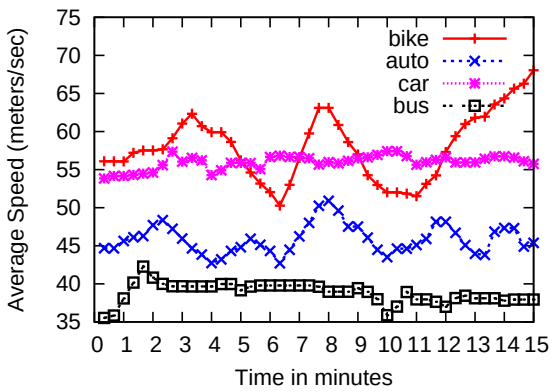

Figure 8. Average speeds on a smooth road in free-flow traffic for four vehicle categories

Secondly, to differentiate between vehicles with similar jerk signatures, the absolute value of speed might help. Bikes have much higher speeds than autos, and cars are much faster than buses. This is also intuitive, as personal vehicles like bikes and cars typically have more expensive engines and are better maintained than public transport like buses and autos. Also the smaller sizes of bikes, compared to autos and of cars, compared to buses, might cause the smaller vehicles to be faster. A third factor affecting speed might be intermittent stop and go behavior to pick up and drop passengers for public transports, while the same behavior is absent for personal vehicles like bikes and cars. This third factor is absent in this particular example shown in Fig. 8, as the 15 minutes plotted belonged to free-flow driving without stops.

Inertial sensors like accelerometer, gyroscope and orientation sensors on the smartphones, might help in monitoring these mobility related features. Though speed can be measured with accelerometer, as has been done in Fig. 8, GPS sensing will give directly measured less noisy speed estimates. The concrete features extracted from these four sensors, to capture the mobility related vehicle characteristics, are summarized in Table. 3.

Sensor	Features
Accelerometer	mean, median, min, max, linear speed, variance, energy, FFT coefficients
Gyroscope	rotational speed
Orientation	orientation
GPS	linear speed

Table 3. Smartphone sensors for mobility features

We also explore a second category of features for vehicle classification, which are more dependent on individual vehicle environment. These include (a) the magnetic field which might depend on the vehicle size, (b) the ambient light which might distinguish public transport at night from other vehicle categories, as the internals of buses are typically brightly lit and (c) the ambient noise, as different vehicles might have very different sound signatures. Fig. 6 and Fig. 7 show the spectrograms for bus and auto engine noises, and the frequencies shown as numbers on the left are visibly different. Table 4 summarizes these environment related features, and the corresponding smartphone sensors which provide the relevant information.[3]

Sensor	Features
Magnetometer	magnetic field
Light sensor	ambient light
Microphone	ambient noise

Table 4. Smartphone sensors for environmental features

It is intuitive that the environment related features will be affected by multiple external factors. Heavy traffic would cause magnetic field or sound from multiple surrounding vehicles, to add noise to the characteristic magnetic and acoustic signatures of a particular vehicle. Putting the phone in pocket would muffle noise and cut off the light sensor. Thus the mobility related features are expected to be more robust than environment related features, but we include the latter in our analysis for comprehensiveness.

[3]The smartphone models, used in our experiments, only had internal temperature sensors to detect phone heating. But recent smartphone models like the Samsung Galaxy S4 have external temperature sensors included. So ambient temperature might be added to this list of environment related features. Buses and cars might be air-conditioned, while bikes and autos have open structures and exhibit atmospheric temperature, thereby showing some distinctive patterns for classification.

5.2 Classification Algorithms

We examine several standard ML classifiers for our four way classification task. These include (a) non-linear classifiers like Decision Tree (C4.5 DT), K Nearest Neighbor (KNN) and Hidden Markov Model (HMM) and (b) linear classifiers like Support Vector Machine (SVM) and Naive Bayes classifier. For the linear classifiers, we use the **one-vs-all** variants, where the classifier is trained to differentiate between one class to be detected as positive and all other three classes to be detected as negative. Thus n classifiers need to be trained for n categories. This has less training overhead than training ${}^nC_2$ classifiers, to differentiate each class against each of the three remaining classes, which is necessary in the **one-vs-one** variants of the same algorithms.

One minute of sensor data is buffered to compute the features listed in Table 3 and Table 4. As mentioned earlier, the sampling frequency is about 18 Hz for each sensor. Thus $18 * 60$ i.e. approximately 1000 samples are used to compute each feature. These features are passed to the ML classifiers, to classify the one minute window into one of the four vehicle classes. The windows are slid by 20 seconds, so that each minute effectively gets 3 labels. 15 such labels are accumulated over 5 minutes, and then the overall 5 minute window is classified into one of the four vehicle categories, according to majority voting. This simple bagging method reduces spurious misclassification errors for each individual one minute classification window.

Thus our minimum classification latency is 5 minutes, or in other words, vehicles have to move for minimum 5 minutes for *VividhaVahana* to output a vehicle category label. Typical trip times for vehicles on Indian roads exceed several tens of minutes. So this 5 minute latency is suitable for most use cases, though there might be some scope of improvement in future.

5.3 Evaluation

To evaluate and compare the different features and classification algorithms, we create a dataset of 2150 window instances, each with 5 minutes of continuous vehicle motion, as labeled by our mobile vs. stationary detection heuristic. These windows are extracted from the overall 1500+ Km. of driving data described in Section 3, and therefore cover different traffic situations and road conditions.

The instances comprise of 678 ground truth labels of 2-wheelers, 715 3-wheelers, 448 cars and 314 bus instances. 4-fold cross validation is run on this dataset, using the standard ML library WEKA, for each of the algorithms and features described above.

We use some standard metrics for evaluation, as summarized in Table 5. T and N are the number of ground truth labels for the positive class (one vehicle category) and the negative class (the other three categories), for a particular classification task. TP denotes the true positives, or the number of positive instances correctly classified as positive, TN denotes the true negatives, or the number of negative instances correctly classified as negative. FP denotes the number of negative instances wrongly classified as positive and FN the converse. The overall accuracy metric values, for the different classifiers, are given in Table 6.

Metric	Definition
Accuracy(Acc)	$\#(TP+TN)/\#(P+N)$
Precision/Positive predicted value(PPV)	$\#TP/\#(TP+FP)$
Recall/Sensitivity(Se)	$\#TP/\#(TP+FN)$
Negative predicted Value(NPV)	$\#TN/\#(TN+FN)$
True Negative Rate/Specificity(Sp)	$\#TN/\#(FP+TN)$

Table 5. Metrics used to evaluate vehicle classification

Algorithms	bike	auto	car	bus
C4.5 DT	**92.69**	**90.41**	**93.76**	**94.65**
KNN	73.87	77.45	79.23	75.23
HMM	82.67	79.25	81.22	78.98
SVM	76.50	74.32	75.23	78.23
Naive Bayes	70.86	75.67	71.34	73.24

Table 6. Classification accuracy for different algorithms

As can be seen from the accuracy values, the C4.5 Decision Tree outperforms the other linear and non-linear classification algorithms by a large margin. The confusion matrix for the four vehicle categories for the C4.5 DT is given in Table 7. The high values along the diagonal of the matrix, show the instances correctly classified by the algorithm, and validate the high accuracy values.

Actual/ Predicted	bike	auto	car	bus
bike	**603**	43	20	12
auto	43	**600**	23	49
car	36	26	**376**	5
public transport	3	22	24	**265**

Table 7. Confusion matrix for C4.5 DT

In many classification tasks, the positive class is more interesting than the negative class. For example in case of detecting an event like traffic congestion, the positive instances of congestion are more important to be detected, than the negative class of free-flow traffic, as some corrective action might need to be taken for the congestion instances. Thus evaluation of the positive instance classification with metrics like precision (how many of the positive classifications are correct) and recall (how many of the positive instances have been correctly classified), are more common to consider.

Vehicle class	PPV	Se	NPV	Sp
bike	88.93	88.02	94.42	94.88
auto	83.91	86.83	93.65	98.89
car	84.87	84.87	96.07	96.07
bus	85.48	80.06	96.4	97.52

Table 8. Metric values for C4.5 DT

In our vehicle classification task, however, the positive and negative classes are equivalent, as both represent some

vehicle category. Thus in addition to precision and recall, we also measure similar metrics for negative instance classification, namely negative predicted value (how many of the negative classifications are correct) and specificity (how many of the negative instances have been correctly classified). These metrics have been summarized in Table 5 and the metric values for the C4.5 DT are given in Table 8. As can be seen, both positive and negative instance classifications are fairly accurate, thereby making C4.5 DT a good choice for our vehicle classification problem.

To explore which sensors are more suitable for the vehicle classification task, we run the C4.5 DT on features extracted from each individual sensor and also some combinations. Fig. 9 shows the accuracy values along y-axis and the sensor names along x-axis. The mobility related features from accel, gyro, orientation and GPS sensors consistently perform better than the environment related features from magnetometer, light sensor and microphone. This observation is in accordance with our earlier intuition, that external factors would make the environment related features noisy.

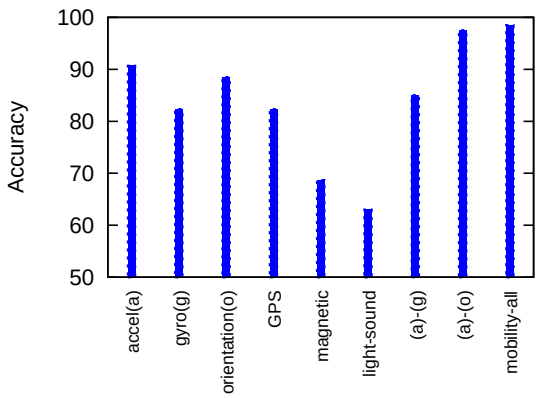

Figure 9. Accuracy with different sensor combinations

Among the mobility related features, combining accel with orientation gives very high accuracy. Adding GPS or gyro to the accel-orientation combination, does not increase accuracy much, as can be seen from the bar labeled "mobility-all". This is important if we consider the sensing related battery drain on the participatory smartphones. As inertial sensors like accelerometer and orientation sensor, are much less energy consuming than GPS [27, 28], sampling GPS or the environment related sensors might be turned off to conserve battery, without any visible reduction in the vehicle classification accuracy.

6 Applications Dependent on Vehicle Class

As discussed in the previous section, *VividhaVahana* can accurately detect four vehicle classes at minimum latencies of 5 minutes, using mobility related features extracted from accelerometer and orientation sensors on smartphones. To understand the practical importance of such an automated vehicle classification scheme, we next explore some well-known ITS applications. These applications are currently vehicle class agnostic, and we show empirically from our experimental data, why adding the vehicle category information can make these applications more accurate and better suited for developing region traffic.

The scope of this section is the analysis of four ITS applications, to show their dependence on vehicle category. The re-design and actual implementation of the applications, incorporating vehicle category related changes, are an avenue of future work.[4]

6.1 Travel time estimation

The first application that we consider is travel time estimation, which is one of the most popular ITS applications around the world. As described in Section 3, the data from the four vehicle types in our experiments, were collected with the vehicles starting from the same source at the same time, and driven towards the same destination, along the same route. Fig. 10 shows the average speeds on the y-axis vs. time in minutes along the x-axis for one such experimental trip, during peak hours. The curves end when each individual vehicle arrives at the destination, which is at a driving distance of about 6 Km from the source.

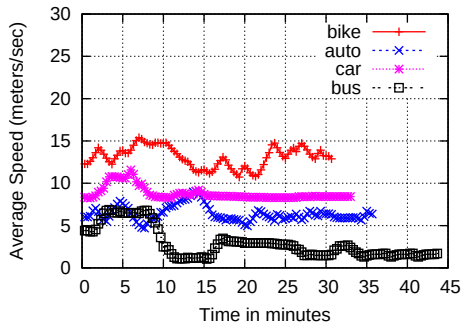

Figure 10. Variable travel times of different vehicles

As can be seen from the figure, the travel times of the different vehicle classes vary, by upto 15 minutes between bikes (travel time - 30 mins) and buses (travel time 45 mins). This variation is for a short driving stretch of 6 Km. Thus longer routes would have more divergent travel times. A travel time prediction service, which takes only the source and the destination as inputs, and does not consider the vehicle category, might therefore be erroneous by an order of magnitude.

Fig. 11 shows an example where vehicles with shorter travel times, gain over the ones with comparatively longer travel times. The y-axis of the plot shows the distance from a particular traffic signal. This distance gradually decreases

[4]We envision that in future, standard ITS apps for smartphones, which either collect participatory data for traffic applications from the phones, or provide traffic related information and services to the phones, would come integrated with a vehicle classification software module. The re-designed apps would use the category information to process the participatory data in more intelligent ways, or provide more streamlined services. A person carrying the same smartphone and traveling in different vehicles like buses, autos or cars, can seamlessly use the ITS applications across different vehicle categories, without manual intervention.

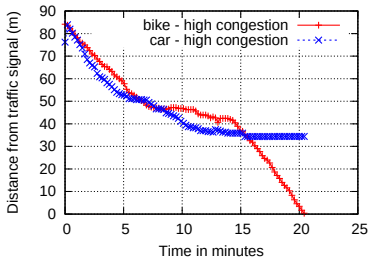

Figure 11. Different speed characteristics of bike and car at a traffic signal

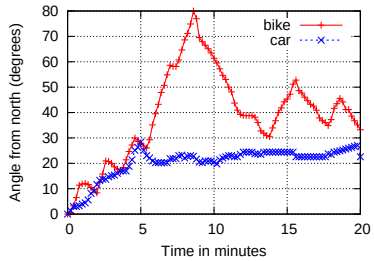

Figure 12. Sharp angular changes due to erratic bike driving in congestion

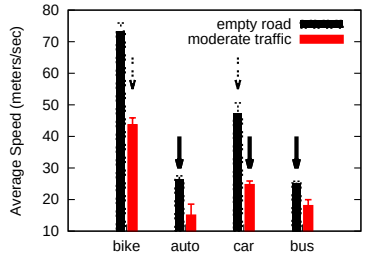

Figure 13. Speed anomalies among different vehicles in characteristic traffic situations

as the vehicles move towards the signal, and time passes in minutes along the x-axis. As we can see, the bike is able to reach the signal in 20 minutes, whereas the car is stationary at a distance of 40 m from the signal from the 10th minute onwards. If the signal now turns green, the bike will quickly attain maximum speed, being at the head of the traffic queue, while the car will accelerate slowly with many vehicles in front, and might get caught in another red cycle. Such small gains in every traffic situation, cumulatively create a significant travel time difference.

6.2 Driving pattern detection

The difference in travel times between vehicles, especially in micro instances like near a traffic signal as discussed above, motivates the analysis of driving pattern for each vehicle type. This is to better understand how the faster vehicles actually achieve their lower travel times. Fig. 12 shows the angle measured from the north by the smartphone orientation sensor, for the same 20 minutes as discussed in Fig. 11. This visually explains the erratic driving of the bike, with sharp changes in direction, as it maneuvers making way through bigger vehicles standing at the signal. The physical characteristics of the bike and the non-laned driving prevalent in developing regions, jointly make this feasible. The car being much larger, cannot mimic this behavior, and has to wait patiently behind other vehicles.

Traffic signal and road design use vehicle mobility models. Such models often make simplistic assumptions even for developing region traffic, like uniform speed for all vehicles at a given road stretch [11]. As is evident from the above empirical examples, heterogeneous vehicles have much more nuanced dynamics even in similar traffic situations, which are significantly more complex than *uniform speed for all*. Incorporating these vehicle category specific information while model building, might thus make them better capture real road scenarios.

6.3 Traffic state detection

Traffic state detection for specific road stretches is another widely used ITS application [19, 20]. Here the road network of a city is visualized in different colors, according to the degree of congestion in different road segments. For example in [19], empty roads are color coded green, segments with moderate and heavy traffic are color coded yellow and red respectively, and zones of anomalous traffic events like accidents are colored black. These traffic states are inferred from participatory speed estimates, for example from GoogleMap users and Android phone users for [19], or the Traffline app users and some GPS enabled buses for [20].

Fig. 13 shows the importance of considering vehicle category information in training such traffic state classification models in developing regions, and also in classifying the traffic states based on the trained models. The y-axis shows the speeds on a particular road stretch, averaged over 10 days of data collection, during traffic states of empty road (green in [19]) and moderate traffic (yellow in [19]). The shaded arrows show the speed similarities between bikes in moderate traffic and cars in empty road, while the solid arrows show the speed similarities between cars in moderate traffic and autos and buses on empty road.

Due to variation in (a) physical properties like vehicle size, (b) mechanical characteristics like engine quality, (c) level of maintenance for private vs. public vehicles, (d) the driving patterns of stop and go for passengers in public transport vs. continuous mobility for personal vehicles, the average vehicle speeds vary under similar traffic situations. Also, speeds for different vehicles are similar in different traffic situations (as shown by the arrows in Fig. 13). Thus speed samples annotated with vehicle type information, can enhance the train and test accuracies of the traffic state classification models, reducing possible errors in the confusion matrix.

Fig. 14 and Fig. 15 highlight another situation, where traffic state detection might benefit from vehicle category information. In normal high congestion like near a traffic signal, different vehicles have different speed characteristics, as seen previously in Fig. 11 and Fig. 12. However, in cases of serious or anomalous events like road accidents, almost everything comes to a standstill, to prevent exacerbating the situation or impossibility of any movement caused by road blocks and road rages. Fig. 15 shows how speeds for both cars and bikes drop to zero for nearly 30 minutes, on a road stretch following an accident (Fig. 14).

Thus disambiguating the normal high congestion (red in [19]) from serious incidents (black in [19]), and quickly detecting the latter for appropriate action, might benefit from vehicle category information. This is analogous to a situation where a human population has significant variance in immu-

Figure 14. Accident image

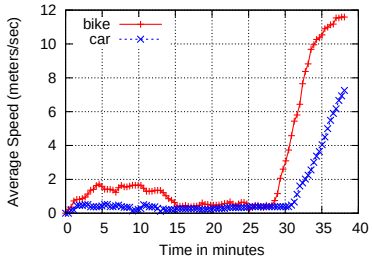

Figure 15. Similar slow driving of different vehicles after road accident

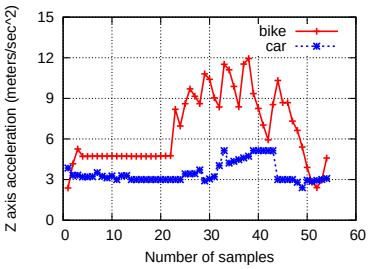

Figure 16. Different acceleration signatures on a speed-breaker

nity levels, and the seriousness of an infection is gauged by its effect on the different immunity classes. When the most immune show signs of succumbing, it is similar to even bikes showing zero speeds for considerable time periods. The seriousness of the traffic situation can thus be better assessed if the speed samples are annotated with category information.

6.4 Road condition monitoring

The final application that we discuss in this context is road condition monitoring. Similar to crowd-sourced traffic maps discussed in the previous section, this application uses crowd-sourced sensor data from smartphones and detects road surface anomalies like potholes, bumps, speed-breakers etc., based on the sensor signatures [5, 21].

Fig. 16 shows the difference in acceleration along z-axis, faced by a bike and a car, while crossing the same road bump or speed-breaker. The sensor signature is visibly different between the two vehicles. This can be intuitively explained by the different form factors and mechanical characteristics like shock absorbers for the two vehicles. The larger size and more cushioned design of cars give a much less pronounced signature than the smaller, bare-bone bike. Thus similar to traffic state classification models, road anomaly detection models might also benefit from vehicle category information, to better detect and classify road surface characteristics.

7 Discussion and Future Work

ITS being an area of active research, the associated services and applications are ever-growing in multitude. In that respect, this paper explores only a representative subset. There might be many more interesting applications, which would benefit from a vehicle category specific analysis. [29] measures fuel consumption and carbon emission of vehicles, and if combined with vehicle type detection, this might help in building statistical emission datasets for different vehicle categories. Similarly, [30] which aids in automobile autotuning in different traffic situations, or [31] which helps in detecting empty parking spaces, will intuitively benefit from vehicle type information.

The ITS applications discussed so far are such that even a small fraction of participating vehicles on the road, can make a marked difference in the related service. These services like travel time estimation or road surface monitoring, can deal with the non-deterministic sparsity of participatory sensing. But there might be another category of services, for whom dense data, or sparse data which is at least deterministically sparse, is necessary. An example is road infrastructure planning as commonly done in India. Decision to add any new transportation resource like traffic signal or flyover is driven by PCU or passenger count units, faced by the road location under consideration. Passenger count units are vehicle dependent as different vehicles can carry different number of passengers.

However, in such cases, to get a statistical measure of what different categories of vehicles ply the road segments in what proportion, sparse participatory data from smartphones as *VividhaVahana* provides, might be too unreliable for appropriate decisions. Static sensors like [26], which process the videos of the entire road segment under consideration for vehicle categorization, might be more suitable. Or appropriate incentive mechanisms need to be studied and deployed [32], to increase levels of participatory sensing. Such incentive studies are especially important in the context of developing countries, where smartphone penetration itself might be low. Thus the choice between competing sensing modes – participatory vs. static, should be driven by the application requirements.

The current work can also be enhanced in several system level aspects. Automatic detection of smartphone orientation [9, 21], with respect to the direction of vehicle motion, will be a useful addition. This will remove the necessity of keeping the phone in a pre-determined location and orientation like in the front pocket of the trouser, as has been done during experimental data collection for this work. Secondly, appropriate filters to remove pedestrian data or spurious motion of the phones using activity recognition techniques [33, 34], will increase the robustness of vehicle classification. Thirdly, with the recent high proliferation of wearable devices [35, 36], the scope of using wearables instead of smartphones might be explored. Finally, the ITS applications have only been analyzed to show their dependency on vehicle category information. Actually incorporating the vehicle class information and providing working systems for each application, would require significant additional engineering efforts in the future.

8 Conclusion

In this paper, we explore a unique characteristic of developing region traffic, in the form of high heterogeneity of vehicle classes. We present *VividhaVahana*, a smartphone sensing based system to classify vehicle categories. *VividhaVahana* achieves above 90% accuracy for four vehicle classes on 1500+ Km of driving data. We also analyze four representative ITS applications to empirically show their dependence on vehicle categories. This is an effort to highlight the importance of incorporating vehicle class information in existing ITS applications, to make such services better suited for developing countries.

References

[1] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson. Vtrack: Accurate, energy-aware road traffic delay estimation using mobile phones. In *Sensys*, 2009.

[2] A. Thiagarajan, L. Ravindranath, H. Balakrishnan, S. Madden, and L. Girod. Accurate, low-energy trajectory mapping for mobile devices. In *NSDI*, 2011.

[3] J. Biagioni, T. Gerlich, T. Merrifield, and J. Eriksson. Easytracker: automatic transit tracking, mapping, and arrival time prediction using smartphones. In *SenSys*, 2011.

[4] J. Yoon, B. Noble, and M. Liu. Surface street traffic estimation. In *MobiSys*, 2007.

[5] Eriksson J., Girod L., Hull B., Newton R., Madden S., and Balakrishnan H. The pothole patrol: Using a mobile sensor network for road surface monitoring. In *MobiSys*, 2008.

[6] V. Jain, A. Sharma, and L. Subramanian. Road traffic congestion in the developing world. In *The 2nd Annual Symposium on Computing for Development (DEV '12)*.

[7] Sen R., Maurya A., Raman B., Mehta R., Kalyanaraman R., Vankadhara N., Roy S., and Sharma P. Kyun queue: A sensor network system to monitor road traffic queues. SenSys, 2012.

[8] Sen R., Raman B., and Sharma P. Horn-ok-please. In *Mobisys*, June 2010.

[9] Mohan P., Padmanabhan V. N., and Ramjee R. Nericell: Rich monitoring of road and traffic conditions using mobile smartphones. In *SenSys*, 2008.

[10] P. Singh, N. Juneja, and S. Kapoor. Using mobile phone sensors to detect driving behavior. In *ACM DEV*, 2013.

[11] Sen R., Cross A., Vashishtha A., Padmanabhan V. N., Cuttrell E., and Thies W. Accurate speed and density measurement for road traffic in india. In *ACM DEV*, 2013.

[12] K. Li, M. Lu, F. Lu, Q. Lv, L. Shang, and D. Maksimovic. Personalized driving behavior monitoring and analysis for emerging hybrid vehicles. 2012.

[13] C. You, N. D. Lane, F. Chen, R. Wang, Z. Chen, T. J. Bao, M. Montes-de Oca, Y. Cheng, M. Lin, L. Torresani, and A. T. Campbell. Carsafe app: Alerting drowsy and distracted drivers using dual cameras on smartphones. In *MobiSys*, 2013.

[14] `http://en.wikipedia.org/wiki/Fundamental_diagram_of_traffic_flow`.

[15] A. Salim, L. Vanajakshi, and S.C. Subramanian. Estimation of average space headway under heterogeneous traffic conditions. *International Journal of Recent Trends in Engineering and Technology*, 2010.

[16] C. Mallikarjuna and K. Ramachandra Rao. Heterogenous traffic flow modeling: a complete methodology. *Transportmetrica*, 2011.

[17] C. Mallikarjuna, K. Ramachandra Rao, and N.V.S.K Seethepalli. Analysis of microscopic data under heterogenous traffic conditions. *Transport*, 2010.

[18] Wang H., Li J., Chen Q., and Ni D. Representing the fundamental diagram: The pursuit of mathematical elegance and empirical accuracy. In *Transportation Research Board 89th Annual Meeting*, 2010.

[19] `http://en.wikipedia.org/wiki/Google_Traffic`.

[20] `http://www.traffline.com/default.aspx`.

[21] R. Bhoraskar, N. Vankadhara, B. Raman, and P. Kulkarni. Wolverine: Traffic and road condition estimation using smartphone sensors. 2012.

[22] S. Hemminki, P. Nurmi, and S. Tarkoma. Accelerometer-based transportation mode detection on smartphones. In *SenSys*, 2013.

[23] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava. Using mobile phones to determine transportation modes. *ACM TOSN*, 2010.

[24] Piggott D. Inferring transportation mode using smartphone sensor data. *Dissertation submitted for Computer Science Tripos, Fitzwilliam College*.

[25] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W. Ma. Understanding mobility based on gps data. In *UbiComp*, 2008.

[26] C. Mallikarjuna, A. Phanindra, and K. Ramachandra Rao. Traffic data collection under mixed traffic conditions using video image processing. *Journal Of Transportation Engineering*, 2009.

[27] R. Jurdak, P. Corke, D. Dharman, and G. Salagnac. Adaptive gps duty cycling and radio ranging for energy-efficient localization. In *SenSys*, 2010.

[28] K. Lin, A. Kansal, D. Lymberopoulos, and F. Zhao. Energy-accuracy trade-off for continuous mobile device location. In *Mobisys*, 2010.

[29] R. K. Ganti, N. Pham, H. Ahmadi, S. Nangia, and T. F. Abdelzaher. Greengps: a participatory sensing fuel-efficient maps application. In *Mobisys*, 2010.

[30] T. Flach, N. Mishra, L. Pedrosa, C. Riesz, and R. Govindan. Carma: towards personalized automotive tuning. In *SenSys*, 2011.

[31] S. Mathur, T. Jin, N. Kasturirangan, J. Chandrasekaran, W. Xue, M. Gruteser, and W. Trappe. Parknet: drive-by sensing of road-side parking statistics. In *Mobisys*, 2010.

[32] Rula J., Navda V., Bustamante F., Bhagwan R., and Guha S. No one-size fits all: Towards a principled approach for incentives in mobile crowdsourcing. In *HotMobile*, 2014.

[33] M. Keally, G. Zhou, G. Xing, J. Wu, and A. Pyles. Pbn: Towards practical activity recognition using smartphone-based body sensor networks. In *SenSys*, 2011.

[34] A. Rai, K. Chintalapudi, V. N. Padmanabhan, and R. Sen. Zee: Zero-effort crowdsourcing for indoor localization. In *Mobicom*, 2012.

[35] Rallapalli S., Ganesan A., Chintalapudi K., Padmanabhan V. N., and Qiu L. Enabling physical analytics in retail stores using smart glasses. In *Mobicom*, 2014.

[36] Ha K., Chen Z., Hu W., Richter W., Pillai P., and Satyanarayanan M. Towards wearable cognitive assistance. In *Mobisys*, 2014.