Deep Transfer Learning Based Intersection Trajectory Movement Classification for Big Connected Vehicle Data

Md Mostafizur Rahman Komol1,2, Mohammed Elhenawy1,2, Mahmoud Masoud1,2, Sebastien Glaser1,2, Andry Rakotonirainy1,2, Merle Wood3 and David Alderson3

1 Queensland University of Technology, Centre for Accident Research and Road Safety-Queensland, Australia
2 Queensland University of Technology (QUT), Institute of Health and Biomedical Innovation (IIBI), Australia
3 Department of Transport and Main Road (Queensland), Australia/Department of Physics, Colorado State University, Fort Collins, CO 80523 USA

Corresponding author: Md Mostafizur Rahman Komol (e-mail: mdmostafizurrahman.komol@hdr.qut.edu.au)

ABSTRACT Trajectory movement labelling is an important pre-stage for predicting connected vehicle (CV) movement at intersections. Drivers’ movement prediction and warning at intersections ensure advanced transportation safety and researchers use machine learning-based data-driven approaches to implement these technologies. However, prediction of drivers’ movements at intersections requires labelling the train and test dataset accurately with different vehicle movements at intersections to evaluate the performance of the prediction model by comparing the actual and predicted intersection movements. Moreover, due to GPS detection error or missing co-operative awareness messages (CAM), the data resides with many abnormal trajectories which are unable to be matched with regular straight or any turning movements. Especially for big data with million trajectories, it is tedious to label the movements manually. To solve this problem, we have created an automated trajectory movement classification technique using a dual approach of map matching technique and deep transfer learning modelling. Data of connected vehicle trajectory information is taken from the Ipswich Connected Vehicle Pilot (ICVP) Project, which is one of the largest connected vehicle pilots within a naturalistic driving environment in Australia. Map matching approach is performed as initial labelling by analysing the origin and destination of the vehicle CAM messages at intersections and then was converted as image datasets of 19202 samples. The map matching error and abnormal trajectories are identified by visual inspection. With properly labelled 9496 training images, 10 transfer learning models are built and tested through the remaining 9706 testing images. The maximum testing accuracy (99.73%) is achieved from the Densenet169 model, and the result shows satisfactory accuracy for individual classes: straight (99.85%), turn left (99.59), turn right (99.25), u-turn (100%), abnormal (98.63%). This model becomes a routine tool that is used daily to automatically classify thousands of trajectory movements of the C-ITS data in the ICVP project.

INDEX TERMS connected vehicle, movement classification, intersection, map matching algorithm, transfer learning.

I. INTRODUCTION

The connected vehicle is an emerging technology of Intelligent Transportation System (ITS) that has potential road safety applications and advanced transportation facilities [1]. It is an advanced transformative solution to traffic collisions, vulnerable road users’ safety, road work zone safety, congestion, and dilemma zone problems [2, 3]. Here, vehicles and system infrastructures are connected wirelessly with a vehicular ad hoc network (VANET) and communicate with each other to properly navigate with upcoming circumstances. The communication transmits bi-directionally from vehicle to vehicle (V2V) and vehicle to infrastructure (V2I) in order to locate vehicle positions, intended movements, destination and produce safety warnings based on circumstances [2]. The multipurpose implementation of this technology and consequent safety warning for drivers improve transportation in many ways [4, 5]. Some most frequent safety applications of connected vehicle technology include back of queue warning [6], road work warning [7], advanced red light warning [8], red-light running prediction [3], emergency vehicle travel time reduction [9]. In Queensland, Australia, vulnerable road users: pedestrians, bicyclists, and motorcyclist fatality is 12.3%, 3% and 13% respectively of overall road death [10].
Connected vehicle technology is effective in producing warnings for vulnerable road users crossing at roads. Apart from safety implications, connected vehicle technology ensure emission reduction and fuel-saving which is environmentally friendly as future mobility [11].

At present, there exist many pilot studies of connected vehicles in developed countries like the USA, Australia, Japan, China, France and so on. Some common and renowned connected vehicle pilots are New York City DOT Pilot [12], Tampa-Hillsborough Expressway Authority Pilot [13], Wyoming DOT Pilot [7], Ipswich Connected vehicle Pilot [14, 15], Safe and Intelligent Mobility Project [16] etc. They are working on implementing connected vehicle technology and applications practically on-road as field operation test (FOT) over large urban areas. Such large-scale pilot studies confront major challenges with accurate data collections, management, processing and analysis.

When evaluating drivers’ behaviour under the influence of C-ITS, identifying and labelling individual trajectory movements of connected vehicles at intersections is essential for accurate analysis and understanding. Labelling can be a labour-intensive task that also introduces some level of error. An automated intersection trajectory movement classification is the desired approach for handling big connected-vehicle datasets. As an alternative, there are other commercial routing tools that extract accurate geo-location and map information during dynamic vehicle movements. However, such commercial tools incur a license fee, and this expense rises immensely with usage over very large datasets. Moreover, the quality of collected data highly relies on the transmission latency of information between vehicles and infrastructures. During the data collection in such extensive field operation test procedure, errors are occurred by communication devices like GPS and cause inaccurate CAM location information, which raises the difficulty in properly classifying vehicle trajectories. Especially at intersections, this inaccurate location information leads to obscurity in labelling the vehicle trajectory movements. An example of an erroneous trajectory found during data analysis is shown in the following Figure 1.

**FIGURE 1.** Example of an Erroneous Trajectory
Identify erroneous abnormal trajectories which have missing CAM information due to GPS error at intersections.

Automatically annotate trajectory movement for downstream statistical analysis or as for any supervised machine learning based prediction analysis like drivers’ intended intersection movement prediction.

Create models suitable for hardware capacity. Moreover, this study compares many pre-trained networks of different sizes to annotate trajectories at intersections so that hardwares with different computational power can use the best performing model based on its capacity.

II. WORKING PRINCIPLE

Our working principle comprises several stages. For our current study, naturalistic driving data of connected vehicles is taken from the ICVP project [14], the largest connected vehicle pilot study in Australia. The naturalistic driving method provides intuition to regular driving behaviour, and it is advantageous in collecting very large datasets in quantitative and qualitative terms from field operational test (FOT) [17-19]. The ICVP project has installed roadside units at 29 signalised intersections and undergoes a field operation test of 351 connected vehicles on Queensland roads.

The map matching approach is performed using road topology information from MAP Extended Messages (MAPEM) and polygon drawing over desired intersection zone using Google Earth Pro software. By analysing the origin and destination of the CAM messages at these polygon regions, vehicle trajectory movements at intersections are labelled initially. Following that, trajectory information is converted as image datasets of 19202 samples using our MATLAB app. The map matching error and abnormal trajectories are identified by visual inspection over the images, and they are labelled into five classes, including straight, turn left, turn right, U-turn and abnormal trajectory class. By the end of the map matching and visual inspection, we take a data set that consists of 19202 images with pure labels. Consequently, 9496 images are used to train 10 pre-trained models through transfer learning, and the rest are used to test the trained models. The flowchart in Figure 2 shows the Automated trajectory movement labelling framework. The detailed schematic procedure of trajectory movement identification is illustrated in the following subsections.

III. MAP MATCHING BASED MANEUVER LABELLING

This section describes the first step in labelling vehicle trajectories at the signalised intersections. This step matches the trajectory points to the surveyed lane centre lines to identify the origin and destination of a trajectory in the intersection segments. The surveyed lane centre lines are extracted from the MAPEM. The steps to do this labelling are as follow:

A. PREPARING SCENARIO

Initially, a set of polygons are constructed for the 29 signalised intersections. Google Earth Pro is used to draw the polygons, which are then saved as a KML file. Finally, the KML is imported into MATLAB and converted to a structure. Figure 3 shows an example of the polygon drawing approach over the intersection.
We import the trajectory data points and use the polygon structure to select all CAMs inside the intersection area. Then we filter out all CAMs which has a speed value of less than 4.32 km/h. We consider any CAM with a speed value of less than 4.32 km/h is broadcasted by a stationary vehicle. The removal of stationary CAMs can minimise duplicate data points to improve data processing efficiency.

B. ORIGIN-DESTINATION (OD) APPROACH

We use the constructed Origin-Destination (OD) matrix of the intersection to estimate the origin and destination of the trajectories. Because of inaccuracies in the location information, we group the lanes of each leg into two groups, namely the ingress group and the egress group, as shown in Figure 3. Each ingress and egress of intersections is assigned with a unique id. Based on a schematic analysis of origin and destination, we can assign the label of trajectory movement. Segmentation of a four-leg intersection into numeric ids based on ingress (labelled with red fonts) and egress (labelled with green fonts) is shown in Figure 4, and trajectory labelling scheme with origin and destination is shown in Table I.

C. MATCHING TRAJECTORIES WITH OD MATRIX

We calculate the distance from each point on the trajectory to the intersection centre point and use these distances to divide the trajectory into two parts ingress trajectory and egress trajectory. Then we match the constructed ingress trajectory CAM points against all possible ingress lane centre lines extracted from the intersection’s KML file. Remove any CAM that is far more than five meters from the nearest centre line as they are deemed to add noise to the data matching. Based on this match, each ingress lane has a score reflecting its probability of being part of the ingress trajectory. We repeat the above procedure using the egress trajectory and the egress centre lanes. After that we use the output lane scores from ingress and egress matching and the approach OD matrix to assign a label (i.e. straight, turning left, turning right and U-turn) to the trajectory. The flowchart in Figure 5 depicts the overall map matching procedure.
FIGURE 5. Flow chart of CAM filtering, trajectory construction and map matching.
Using the map machine based automated labelling, four types of connected vehicle trajectory movements are labelled at a signalised intersection consisting of straight, turn left, turn right and U-turn movements. However, there are no ground truth labels of trajectory movements to compare our map matching based labelling, and so, it is not feasible to measure the accuracy of map matching based labelling. Moreover, our map matching approach is incapable of identifying abnormal and erroneous trajectories that do not match with regular four intersection movements. The trajectories with missing CAM messages are not identifiable using this algorithm as it only analyses the origin and destination of vehicle trajectory points. So, we converted intersection trajectories into images using our designed MATLAB application and inspected their movements visually to compare with map matching based labels. Vehicles positions at trajectory are continuous dots that originate from frequent CAM message-based communication. These continuous dots generate the trajectory line at images which define its movement at intersections with its origin to destination approach. Also, using the vehicle position information at longitude and latitude in the map, the trajectories of each event are drawn into images. The graphical user interface (GUI) of the MATLAB app is shown in Figure 6.

To evaluate the result of map matching based trajectory movement labelling, we visually inspect all 19000 trajectories. Here, many incorrect labelling is detected when visually inspecting the trajectory images, and the estimated accuracy is 90%. Also, the abnormal trajectories were identified during the visual inspection of images which were unable to be detected by the map matching approach. So, only labelling with map matching may produce further mislabelled data of intersection trajectories with a lot of noisy abnormal movement incidents. These abnormal trajectory movements found through GPS tracking are often caused by data incompleteness, double trajectory lines or undefined movement class. It is noted that some trajectory origins have displaced a few steps ahead due to missing a few CAM messages, but they are still clearly identifiable of their movement class. So, instead of considering them as abnormal trajectories, they are still considered under regular movement classes. Also, sometimes vehicles parked or passed through a sideway after crossing the intersection. These incidents still show incomplete trajectories, but their movement is still identifiable. But trajectory movements that are not identifiable to the regular four-movement class are considered abnormal and don't contribute to our research purpose. Instead, these can be considered as an error in data collection or detection equipment (GPS). These abnormal trajectories are needed to be identified such that any downstream analysis of the data becomes more accurate. So, it is crucial to identify five different trajectory movement classes, including abnormal trajectories. Figure 7 and Figure 8 shows output labels and some abnormal trajectories, respectively. The assigned label in Figure 7 shows the mistaken label.

![FIGURE 6. MATLAB application interface developed to visually check and correct any wrong label.](image)

![FIGURE 7. Output Labels for Several Trajectories with An Assigned Misclassified Label.](image)
Some abnormal trajectories of intersection vehicle movements.

To classify trajectory movements with a higher level of accuracy, we used deep transfer learning-based modelling. The detailed schematic procedure of trajectory movement identification is illustrated in the following subsections.

IV. DEEP TRANSFER LEARNING-BASED MANEUVER LABELLING

Transfer learning is a machine learning design methodology where a model developed for a task is reused as the starting point for a model on a second related task. It is a popular methodology in deep learning where it is not feasible to label millions of data points to learn the massive number of parameters in a neural network. Therefore, pre-trained models are used as the starting point on a second related task. In the context of our trajectory classification problem, transfer learning uses knowledge learned from the image classification task for which millions of labelled data is available in identifying the trajectory movement type where only a little labelled data is available. Recall that creating labelled data is time-consuming and expensive, and our goal is to reduce human efforts and cost in trajectory classification.

In general, there are two common transfer learning approaches we can use in our trajectory classification problem, namely: develop a model approach and a pre-trained model approach. In the developed model approach, we start by selecting a source task related to our task. This source task must have massive data to train a deep neural network from scratch. Then, we use the source task data to train the deep neural network and ensure that it learns the source task. Finally, we reuse and finetune this model or parts of it in our trajectory classification task. The pre-trained model approach is different from the above approach in that we directly reuse and finetune one of the models trained on large and challenging datasets and released by many research institutions.

We adopted ten pre-trained deep neural networks of different sizes to classify trajectories. The rationale of using ten models is to identify the best performing models for accurate classification of trajectory movements. Moreover, complex networks are often beyond the limit of hardware capacity and computational power. So, it is crucial to show the prediction performance of different models so that the best performing model based on hardware capacity can be chosen for the users’ benefit. These models were trained using a massive number of very high-quality images to classify images into one of many predefined categories. The selected pretrained model used in this formula are described in following TABLE II.

| No | Pre-Trained Models | Input Size | Total Layers with Learnable Weights | FCN Layers |
|----|--------------------|------------|-------------------------------------|------------|
| 1  | Alexnet [20]       | 256x256    | 8 (5 CNN)                           | 3          |
| 2  | Vgg16 [21]         | 224 x 224  | 16 (13 CNN)                         | 3          |
| 3  | Vgg19 [21]         | 224 x 224  | 19 (16 CNN)                         | 3          |
| 4  | Googlenet [22]     | 224 x 224  | 22 (21 CNN)                         | 1          |
| 5  | Shuffle net v2 [23]| 224 x 224  | 44 (41 CNN)                         | 1          |
| 6  | Resnet18 [24]      | 224 x 224  | 18 (17 CNN)                         | 1          |
| 7  | Squeeze net [25]   | 224 x 224  | 18 (2 CNN, 15 Fire module of squeezed CNN) | 1          |
| 8  | Densenet161 [26]   | 224 x 224  | 161 (1 Convolution, 3 Transition Layer, 156 Dense Block of CNN) | 1          |
| 9  | Densenet121 [26]   | 224 x 224  | 121 (1 Convolution, 3 Transition Layer, 116 Dense Block) | 1          |
| 10 | Densenet169 [26]   | 224 x 224  | 169 (1 Convolution, 3 Transition Layer, 164 Dense Block of CNN) | 1          |

In the transfer learning technique, the pre-trained models discussed in Table II are finetuned. All the layers are kept frozen, and only the fully connected layer is changed based on the number of movement classes of our current study. To create deep transfer learning models, we needed a labelled dataset with a very large number of intersection movements. The ground truth labels are critical to justify the model accuracy and performance. So, at first, we pick a portion of ICVP data with 19202 trajectory movement incidents which is initially labelled using a map matching approach. Then we convert them into image datasets and correct the mislabeled images by inspecting them visually. Also, we create the abnormal trajectory movement class as the fifth class in the image dataset so that we can identify them separately. Considerably, 19202 trajectory movements generated 19202 images of trajectory movements. Among them, a training dataset is created with 9496 images, and 10 pre-trained models were built for classifying the trajectory movements accurately. The remaining 9706 images are considered as the test dataset. Figure 9 shows examples of 5 different class data used for training and evaluation.

FIGURE 8. Some abnormal trajectories of intersection vehicle movements.
First, we proceed with image augmentation with random flip, random rotation (0-360 degree) and input shear from -10 to 10 to increase the variety of datasets. This adds more noises in training data and helps the model understanding better situation assessment. We also up-sample the abnormal trajectory and U-turn images in the training dataset. Gaussian noises are also augmented with U-turn images and up-sampled. After up-sampling, the image dataset for all classes during training and testing is illustrated in the following TABLE III.

![Figure 9](image-url)  
**FIGURE 9.** trajectory images used as training and testing models: (a) straight (b) turn left (c) turn right (d) U-turn (e) abnormal trajectory.

Also, we convert the image size to model specific input size: 224X224 for all pre-trained models except 227X227 for Alexnet. For trajectory movement classification, the training and testing of pre-trained models were performed in python using Pytorch library. It was run on high configured PC with intel i7 10th generation 3.6GHz processor, 16 GB RAM. We use a GPU Nvidia GeForce RTX 3060 Ti which have 8GB GPU memory and 3584 CUDA cores, to help in faster parallelisation of our classification accuracy by classes. We split the training data into 90% for training and 10% for validation. As hyperparameter tuning, we tune the batch size, optimiser and learning rate using Random hyperparameter search tuning with optuna python library which we have found very flexible to integrate with Pytorch library. As an optimiser, we use Adam optimiser for all models except Shufflenet shows better performance in stochastic gradient descent (SGD) with a momentum of 0.9 Learning rate 0.0001 as default works well for all models, and we run the training and validation for around 25 epochs with early stopping criteria on best validation accuracy. After tuning, the selected hyperparameters for trained models are shown in following TABLE IV.

| No. | Model | Batch Size | Optimiser | Learning Rate | Epoch Number |
|-----|-------|------------|-----------|---------------|--------------|
| 1   | Alexnet [15] | 8 | SGD | 0.00015 | 24 |
| 2   | Vgg16 [16] | 16 | Adam | 0.00018 | 12 |
| 3   | Vgg19 [16] | 16 | SGD | 0.00019 | 7 |
| 4   | Googlenet [17] | 8 | Adam | 0.00025 | 14 |
| 5   | Shuffle net v2 [18] | 4 | Adam | 0.00002 | 14 |
| 6   | Resnet18 [19] | 16 | Adam | 0.000157 | 13 |
| 7   | Squeezenet [20] | 8 | SGD | 0.000195 | 10 |
| 8   | Densenet161 [21] | 8 | SGD | 0.000128 | 13 |
| 9   | Densenet121 [21] | 16 | Adam | 0.0001 | 10 |
| 10  | Densenet169 [21] | 16 | Adam | 0.0001 | 15 |

**V. RESULT & DISCUSSION**

After training the pre-trained models with training images datasets, they are tested with the remaining test dataset of 9706 sample images. The overall model accuracy is measured as well as the prediction accuracy of individual movement classes. The accuracy of all ten models are shown in following Table V, including their test accuracy for each individual intersection trajectory movement class.
From the result, it is clearly identifiable that DenseNet169 gives the highest performance. The maximum testing accuracy (99.73%) is achieved from the Densenet169 model, and the result shows satisfactory accuracy for individual classes: straight (99.85%), turn left (99.59), turn right (99.25), u-turn (100%), abnormal (98.63%). Only, Alexnet outperforms DenseNet169 in classifying the Turn Right class, which is slightly higher (0.25%). Also, Googlenet performs best in classifying Straight movements, but the performance difference between Googlenet and Densenet169 in identifying the straight movement is very narrow. Among the other models, Vgg16 shows comparatively superior performance classifying all movement classes.

Figure 10 shows Densenet169 as the best performing model with the highest overall prediction accuracy and reliability for all movement classes. However, comparing models based on accuracy is not sufficient to justify the model performance. For the variation of test dataset size for different classes, confusion metrics, precision, recall, f1-score is required to understand the performance of the model in real-time implementation. The Confusion Metrics of the best four models is illustrated in TABLE VI.
Confusion metrics show a clear view of predicted vs actual sample movements and help compare the predicted output of different classes of different models. Analysing Table VI, DenseNet169, Alexnet and Googlenet models correctly classify the highest number of turn left (724), turn right (792), and straight (8088) movements, respectively. The correct classification rate of the abnormal trajectory (72) and u-turn movement (11) is equivalent to the best four models.

However, overall, DenseNet169 and VGG16 are reliable with the accurate classification of all movements. Alexnet and Googlenet models classify turn right and straight movement in the highest number, but their performance is less reliable for classifying other classes. Instead, DenseNet169 and VGG16 models have correct classification rates for all movements. For further evaluation of model performance, the performance metrics of the best four models are shown in Table VII.

### Table VI

| Models   | Class          | Actual | Predicted |
|----------|----------------|--------|-----------|
| DenseNet169 | Abnormal | 72 | 0 | 0 | 0 | 1 | 73 |
|           | Straight     | 4  | 8087 | 3 | 4 | 0 | 8098 |
|           | Turn Left    | 0  | 2   | 724| 1 | 0 | 727 |
|           | Turn Right   | 6  | 1   | 1  | 788| 0 | 796 |
|           | U Turn       | 0  | 0   | 0  | 0 | 11| 11 |
| VGG16     | Abnormal     | 72 | 0 | 0 | 0 | 1 | 73 |
|           | Straight     | 7  | 8081| 2 | 3 | 0 | 8098 |
|           | Turn Left    | 1  | 2 | 721 | 3 | 0 | 727 |
|           | Turn Right   | 5  | 1   | 0 | 792| 0 | 796 |
|           | U Turn       | 0  | 0 | 0 | 0 | 11 | 11 |
| Alexnet   | Abnormal     | 72 | 0 | 0 | 0 | 1 | 73 |
|           | Straight     | 4  | 8088| 3 | 3 | 0 | 8098 |
|           | Turn Left    | 1  | 2 | 722 | 2 | 0 | 727 |
|           | Turn Right   | 8  | 2   | 1 | 785| 0 | 796 |
|           | U Turn       | 0  | 0 | 0 | 0 | 11 | 11 |
| Googlenet | Abnormal     | 72 | 0 | 0 | 0 | 1 | 73 |
|           | Straight     | 4  | 8088| 3 | 3 | 0 | 8098 |
|           | Turn Left    | 1  | 2 | 722 | 2 | 0 | 727 |
|           | Turn Right   | 8  | 2   | 1 | 785| 0 | 796 |
|           | U Turn       | 0  | 0 | 0 | 0 | 11 | 11 |

### Table VII

| Classes  | Precision | Recall | F1 Score |
|----------|-----------|--------|----------|
| Densenet169 | Abnormal | 98.63 | 87.8 | 92.90 |
|           | Straight  | 99.86 | 99.96 | 99.91 |
|           | Turn Left | 99.59 | 99.45 | 99.52 |
|           | Turn Right| 98.99 | 99.37 | 99.18 |
|           | U Turn    | 100 | 100 | 100 |
| VGG16     | Abnormal  | 98.63 | 80.89 | 89.29 |
|           | Straight  | 99.86 | 99.96 | 99.91 |
|           | Turn Left | 98.76 | 99.72 | 99.24 |
|           | Turn Right| 99.25 | 99.49 | 99.37 |
|           | U Turn    | 100 | 100 | 100 |
| Alexnet   | Abnormal  | 99.63 | 84.7 | 91.56 |
|           | Straight  | 99.80 | 99.96 | 99.88 |
|           | Turn Left | 99.31 | 99.45 | 99.38 |
|           | Turn Right| 99.50 | 98.88 | 99.19 |
|           | U Turn    | 100 | 100 | 100 |
| Googlenet | Abnormal  | 98.63 | 84.70 | 91.14 |
|           | Straight  | 99.88 | 99.95 | 99.91 |
|           | Turn Left | 99.31 | 99.45 | 99.38 |
|           | Turn Right| 98.61 | 99.37 | 98.99 |
|           | U Turn    | 100 | 100 | 100 |
Performance metrics in Table VII shows the precision, recall and f-1 score of the best four models for each individual movement class. All of them shows 100% precision, recall and f-1 score for straight movement classification. For identifying abnormal trajectory, DenseNet169 shows highest precision (98.63%), recall (87.8%) and f-1 (92.9%) score. DenseNet169 and VGG16, both models, perform accurate classifying straight movement, and their precision, recall and f-1 score is 99.86%, 99.96% and 99.91% equivalently. Googlenet model has a higher precision score (99.88%), but the recall score is reduced to 99.95%, and the f1 score remains similar (99.91%) to the f-1 score of DenseNet169 and Vgg16. DenseNet169 also shows best precision (99.59%), recall (99.45%) and f-1 score (99.52%) for turn left classification. For right turn classification, Alexnet shows the highest precision score (99.50%), whereas VGG16 shows the highest recall (99.49%) and f-1 score (99.37%). DenseNet169 also shows better precision (98.99%), recall (99.37%), f-1 score (99.18%) in classifying right turn movement, which is very close to the highest measured precision (99.50%), recall (99.49%) and f-1 score (99.37%). The comparison of performance metrics among the best four models is visualised in the following Figure 11 along with the precision, recall and the f-1 score of models for individual movement classes.

Based on the performance metrics comparison of the best four models, DenseNet169 is considered the best performing model for our current study. Overall, DenseNet169 classifies the trajectory movement for accurate labelling of mislabelled data and identifying abnormal movements.

In this study, a limited number of U-turn samples and abnormal trajectories are tested, evaluating model performance as the dataset was collected from naturalistic driving and field operational tests. Also, the study classifies abnormal trajectories which only found in the field operation test of the ICVP project. So, model performance may vary if abnormal trajectories of different categories are required to be tested except found in the ICVP projects. However, the methodology of this study is beneficial in solving such limitations. If the model is trained with different categories of erroneous or abnormal trajectories found in a particular experiment, it can successfully classify any intersection movements and abnormal trajectories. This model has successfully classified over 1000,000 trajectory movements of the C-ITS intersection data collected by the ICVP project. It removed a substantial amount of manpower that would have been required to classify the movements with a high level of accuracy. Although we are using transfer learning the model, it still needs a significant number (around 19000) of manually labelled trajectories for training the model. Recent advanced AI technologies like using Generative Adversarial Network may help in producing similar trajectory images of multiple classes and reduce the burden of manual data labelling for supervised learning techniques [27]. In our proposed study, the four best models are explained, and they can be used for different applications. For any cases of memory shortage in testing phases, a smaller network that supports the hardware can be chosen. The methodology of this study has future potential for automated data labelling and scenario analysis of various domains using artificial intelligence techniques. Especially, our prediction models can be directly used labelling big data for supervised machine learning techniques and filtering erroneous data in predicting drivers’ turning behaviour. Moreover, the proposed methodology is useful for automated data labelling applications or classifying scenarios of industrial purposes and field operation tests in a large scale.

VI. IMPLICATION

The proposed methodology defines a simple approach of labelling trajectory movements at intersections. It is highly recommendable for connected vehicle pilot studies with its accurate classification performance. The selected model is currently being used in our ICVP project.

Any co-operative transport assistance and prediction at an intersection require accurate ground truth labelling to measure the accuracy between actual vs predicted values. Especially for turning movement prediction at intersections, red-light running behaviour prediction or stop-go prediction at amber light requires accurate labelling of vehicle movement at intersections. In the case of a big, connected vehicle dataset, a small accuracy loss may cause mislabelling of huge vehicle movement events at intersections. Even a good prediction accuracy will not be considered reliable when the ground truth is not properly labelled. This will jeopardise the quality of evaluation for the implementation of connected vehicle technology on roads. Our proposed framework is highly accurate (near 100%) for the practical implementation of connected vehicle pilot studies. In most case scenarios, commercial tool like Google direction API is potentially used to identify the vehicle trajectory movement information through geo-location and maps. However, the ICVP project would like to develop our own intersection trajectory movement labelling tools so that we do not need to rely on any commercial product. Moreover, Google direction API has its expenses in license sharing and peruse on data; and this expense is noticeable for large datasets of connected vehicle pilot studies. Our proposed methodology is free of cost and ensures accurate trajectory movement classification. For research purposes at pre-processing data stage, abnormal trajectories and missing information in data are needed to be disregarded as data cleaning [28]. Identifying abnormal trajectories due to CAM message error is crucial to potential data analysis and understanding. Our proposed classification model also helps to identify abnormal trajectories which are created due to CAM message error and unable to be matched with regular straight or any turning movements.
FIGURE 11. Performance Metrics Comparison of Top 4 Models
VII. CONCLUSION
This research demonstrates a dual methodological approach to classify vehicle trajectory movement at intersections. It is capable of handling big, connected vehicle datasets of a pilot study by proposing an automated approach of trajectory movement labelling using a simple map matching algorithm and deep transfer learning. The proposed methodology is also cost-effective rather than using expensive commercial tools for identifying vehicle geo-location dynamically and trajectory movement labelling. Also, it helps in data cleaning problems and dataset error identification. The accurate prediction rate of our proposed model in this research defines the potential to use this methodology in the real-time application for trajectory movement labelling of big, connected vehicle data.

ACKNOWLEDGMENT
Profound acknowledgement goes to the Department of Transport and Main Roads (Queensland), Australia, for providing the data of our current study. This research is funded by the Department of Transport and Main Roads, the Queensland University of Technology, iMOVE Australia, and supported by the Cooperative Research Centres program, an Australian Government initiative.

REFERENCES
[1] D. Elliott, W. Keen, and L. Miao, “Recent advances in connected and automated vehicles,” Journal of Traffic and Transportation Engineering (English Edition), vol. 6, pp. 109-131, 2019/04/01/ 2019.

[2] S. Minelli, P. Izadpanah, and S. Razavi, “Evaluation of connected vehicle impact on mobility and mode choice,” Journal of Traffic and Transportation Engineering (English Edition), vol. 2, pp. 301-312, 2015/10/01/ 2015.

[3] M. M. R. Komol, M. Elhenawy, S. Yasin, M. Masoud, and A. Rakotomairany, “A Review on Drivers Red Light Running and Turning Behaviour Prediction,” arXiv, cs.AI 2020.

[4] W. Genders and N. Razavi Saiadeh, "Impact of Connected Vehicle on Work Zone Network Safety through Dynamic Route Guidance," Journal of Computing in Civil Engineering, vol. 30, p. 04015020, 2016/03/01/ 2016.

[5] A. Olia, H. Abedgawad, B. Abdullahi, and S. N. Razavi, “Traffic-Flow Characteristics of Cooperative vs. Autonomous Automated Vehicles,” Transportation Research Board, 2014.

[6] S. Khazraei, M. Hadi, and Y. Xiao, “Safety Impacts of Queue Warning in a Connected Vehicle Environment,” Transportation Research Record, vol. 2621, pp. 31-37, 2017/01/01 2017.

[7] O. Raddaoui, M. M. Ahmed, and S. M. Gaweesh, “Assessment of the effectiveness of connected vehicle weather and work zone warnings in improving truck driver safety,” IATSS Research, vol. 44, pp. 230-237, 2020/10/01 2020.

[8] S. Banerjee, M. Jethani, N. K. Khadem, and M. M. Kabir, “Influence of red-light violation warning systems on driver behavior – a driving simulator study,” Traffic Injury Prevention, vol. 21, pp. 265-271, 2020/05/18 2020.

[9] K. Shaaban, M. A. Khan, R. Hamila, and M. Ghanim, “A Strategy for Emergency Vehicle Preemption and Route Selection,” Arabian Journal for Science and Engineering, vol. 44, p. 8905-8913, 2019/10/01 2019.

[10] M. M. R. Komol, M. M. Hasan, M. Elhenawy, S. Yasin, M. Masoud, and A. Rakotomairany, “Crash severity analysis of vulnerable road users using machine learning,” PLOS ONE, vol. 16, p. e0255828, 2021.

[11] S. Chandra and F. Camal, “A Simulation-based Evaluation of Connected Vehicle Technology for Emissions and Fuel Consumption,” Procedia Engineering, vol. 145, pp. 296-303, 2016/01/01/ 2016.

[12] DOT, “NYC Connected Vehicle Project,” US Department of Transport (DOT) Website, 2020.

[13] DOT, “THEA Connected Vehicle Pilot,” Tampa Hillsborough Expressway Authority- US Department of Transport (DOT) Website, 2020.

[14] TMR, “Ipswich Connected Vehicle Pilot,” Queensland Department of Transport and Main Road Website, 2020.

[15] M. Elhenawy, A. Bond, and A. Rakotomairany, “C-ITS Safety Evaluation Methodology based on Cooperative Awareness Messages,” in 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2018, pp. 2471-2477.

[16] SimTD, “Safe and Intelligent Mobility - Test Field Germany,” City of Frankfurt am Main Website, 2013.

[17] J. Balsa-Barreiro, P. M. Valero-Mora, M. Menéndez, and R. Mehmdo, “Extraction of Naturalistic Driving Patterns with Geographic Information Systems,” Mobile Networks and Applications, 2020/10/23 2020.

[18] J. Balsa-Barreiro, P. M. Valero-Mora, J. L. Berné-Valero, and F.-A. Varela-García, “GIS Mapping of Driving Behavior Based on Naturalistic Driving Data,” ISPRS International Journal of Geo-Information, vol. 8, 2019.

[19] I. van Schagen and F. Sagberg, “The Potential Benefits of Naturalistic Driving for Road Safety Research: Theoretical and Empirical Considerations and Challenges for the Future,” Procedia - Social and Behavioral Sciences, vol. 48, pp. 692-701, 2012/01/01/ 2012.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Commun. ACM, vol. 60, pp. 84–90, 2017.

[21] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” 2015.

[22] C. Szegedy, L. Wei, J. Yangqing, P. Sermanet, S. Reed, D. Anguelov, et al., “Going deeper with convolutions,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 1-9.

[23] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, “ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design,” in Computer Vision – ECCV 2018, Cham, 2018, pp. 122-138.

[24] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778.

[25] F. N. Iandola, M. Moskewicz, K. Ashraf, S. Han, W. Dally, and K. Keutzer, “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size,” ArXiv, vol. abs/1602.07360, 2016.

[26] G. Huang, Z. Liu, L. V. D. Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2261-2269.

[27] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, et al., “Generative adversarial nets,” Advances in neural information processing systems, vol. 27, 2014.

[28] M. Shokrolah Shirazi and B. T. Morris, “Trajectory prediction of vehicles turning at intersections using deep neural networks,” Machine Vision and Applications, vol. 30, pp. 1097-1109, 2019/09/01 2019.