Anticipating Driver Actions via Deep Neural Networks and New Driver Personalization Technique through Transfer Learning

Sahim Kourkouss, Hideto Motomura, Koichi Emura and Eriko Ohdachi
Automotive & Industrial Systems Company, Panasonic Corporation, Osaka, Japan

Keywords: Autonomous Driving, Self-Driving Cars, Driver Monitoring, Driver Behaviour, Deep Learning, Transfer Learning.

Abstract: Anticipating driving behaviours is a promising technology for novel advanced driver assistance systems. In recent years, predicting a driver's future action became an important element to preventive safety technologies and has been advancing greatly contributing to a reduction in road accidents. In this paper, we propose a deep learning network that anticipates driving actions based on information of subject vehicle as well as surrounding vehicles and environment. By re-using a network trained on a great number of various drivers' data with different driving behaviours and linking it to a particular driver with particular taste we propose a method that enables the anticipation of driving behaviours that can be tailored to each driver individually, leading to improved user experiences. We experimentally test our method for acceleration, deceleration and brake profile anticipation task using actual driving data. Our results demonstrate the effectiveness of our approach, achieving a great improvement when anticipating for individuals.

1 INTRODUCTION

For the past hundred years, innovation within the automotive sector has brought major technological advances, leading to safer, cleaner, and more affordable vehicles. In recent years, the industry appears to be on the verge of a revolutionary change engendered by the advent of autonomous or "self-driving" cars.

While recent generations of cars have already driver-assist systems that offer, for example, greater vehicle autonomy at lower speeds as well as reduce the incidence of low impact crashes, it is expected that by 2020, most cars will be able to self-perform multiple tasks such as acceleration, steering and braking simultaneously. Realizing such technology is a challenging task and many problems have been reported (Cabinet Office Japan, 2016; Inagaki, 2015). One of the most important tasks faced is the ability to anticipate future events. Humans use the art of anticipation in every interaction, every movement and every thought without realizing it. If human drivers did not have the ability to anticipate events, we would frustrate or embarrass those we interact with and be in many more car accidents. One other important task is the ability to accommodate the way the car drives itself to every driver's taste, especially for levels 2 to 3 autonomous driving where the driver is still involved in the vehicle's control. Even if perfect self-driving were to be accomplished, that would only be a "one-size fits all" kind of self-driving, which can result in the driver getting bored and intervening with the driving operation. Therefore, the self-driving function would end up being be a useless option.

In this paper, we present a deep learning model that, by learning the driver's behaviour patterns, can anticipate the next driving action based on the driver's likings.

The remainder of the paper is structured as follows. First, in section 2 we give a brief review of previous works on driving behavior anticipation. We then focus in Section 3 on the usage of deep learning algorithms for predicting driving action behaviours, and test it using a simple lane change anticipation problem. Section 4 describes and formalises our method for accommodating driving action anticipation to individual driver's likings, which we then experimentally test on actual driving data in Section 5. Finally, Sect.6 draws up conclusions and suggests possible directions for further research on this topic.

Kourkouss, S., Motomura, H., Emura, K. and Ohdachi, E.
Anticipating Driver Actions via Deep Neural Networks and New Driver Personalization Technique through Transfer Learning.
DOI: 10.5220/0006669002690276
In *Proceedings of the 4th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2018)*, pages 269-276
ISBN: 978-989-758-293-6

2 RELATED WORK

Nowadays, most of the cars available in market come equipped with a variety of cameras and sensors to monitor the surrounding environment and driver status. Through multi-sensory fusion, they provide many assistive features like Lane Keeping Assistance (LKA), Automatic Emergency Braking (AEB), and Adaptive Cruise Control (ACC) etc.

These systems warn drivers when they perform a potentially dangerous manoeuvre (Shia et al., 2014; Vasudevan et al., 2012).

Driver status monitoring for distraction and drowsiness as well as anticipation of driving behaviour, have also been thoroughly researched (Fletcher et al., 2005; Rezaei and Klette, 2014; Herrmann, 2012), and many works have been extensively studied. For example, Volkswagen's Bayesian network anticipates from the vehicle speed and the driver's face direction, whether or not to turn right at a general road intersection while on manual mode driving, and the accuracy is reported to be at 98%.

In addition, BMW's Bayesian network (Liebner, 2013) can anticipate right turn, left turn or straight forward at a general road intersection using driving operations, lane information, GPS etc. Here an accuracy of 98%, 88%, 86% for straight forward, right turn and left turn respectively, has been reported.

However, all the above researches are fitted for "Average-Driver behaviour" and do not respond to each driver's likings.

In this paper, we present a deep learning model that, by learning the driver's behaviour patterns, can anticipate the next driving action based on the driver's likings. Our work complements existing ADAS and driver monitoring techniques by anticipating manoeuvres several seconds before they occur.

3 DRIVING ACTION PREDICTION

3.1 Situation Definition Parameters

In this paper, we define a driving action as one of the following driving operations: lane keeping, acceleration, deceleration and lane change.

In addition, in order to anticipate driving behaviour, we need to define proper parameters to describe driving situations. We examine the parameters that might affect driving behaviours in each driving scene in reference to the seven scenes mainly encountered on a highway as defined by NHTSA (lane keeping / lane change / interchange / branching / junction / lane decrease / emergency vehicle) (NHTSA, 2014).

As a result, we narrow the parameters down to the ones that have the most impact on a driving action and these are:

Subject vehicle information: Car speed, brake, steering wheel information, etc.

Surroundings information: Inter-vehicle distance, angle, relative speed, surrounding vehicles type, status etc.

Road information: Lane width, number of road lanes. (Figure 1).

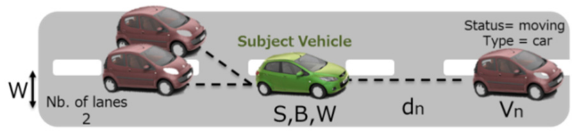

Figure 1: Driving Situation parameters. Subject vehicle speed S, brake status B and wheel information W, the inter-vehicle distance d_n, the relative speed V_n, type and status of the surrounding vehicles, lane width W and number of lanes.

3.2 Stacked Auto-encoder Network

In this paper, we use a deep neural network model for the driving behaviour anticipation task.

Neural networks are a set of algorithms, modelled loosely after the human brain, that are designed to recognize patterns. They interpret sensory data through a kind of machine perception, labelling or clustering raw input.

Deep Learning is a type of Neural Network Algorithm that takes metadata as an input and processes it through a number of layers of a non-linear transformation of the input data to calculate the output. This algorithm has a unique feature which is automatic feature extraction. This means that deep learning algorithms automatically grasp the relevant features required for the solution of the problem. This reduces the burden on the programmer to select the features explicitly. This can be used to solve supervised, unsupervised or semi-supervised type of problems. Therefore, by assuming that driving cases that occurred in the past can and will occur in the future for resembling conditions, deep learning can extract anticipation rules by analysing sets of past driving cases for said conditions and then predict driving behaviour for same conditions a few seconds before they occur.

Neural networks exist in all shapes and sizes, and are often characterized by input and output data type.

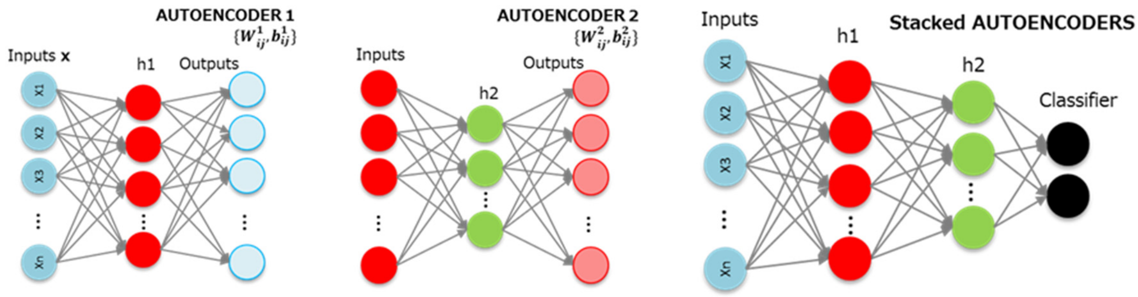

Figure 2: Network for predicting future actions. First, train the first autoencoder individually using backpropagation. Then use the first autoencoder's hidden layer as input of a second autoencoder and train. We repeat this procedure for all the network's layers. Finally, we add a softmax classifier that can classify future driving behaviour to the last layer and train using backpropagation.

An autoencoder neural network is an unsupervised learning neural network algorithm that applies backpropagation, setting the output values to be equal to the inputs. They work by projecting the input into a latent-space representation, and then reconstructing the output from this representation.

By placing constraints on the network, such as limiting the number of hidden units or adding noise to input and train to reconstruct the input from a corrupted version of itself (Denoising Autoencoder), interesting structure about the data can be discovered.

It is difficult for humans to understand all the principles and aspects of driving behaviours, and autoencoder neural network, can be considered as an effective means for our task. By training and "stacking" such autoencoders in a greedy layer-wise fashion for pre-training, we can initialize a regular neural network and train it in a supervised manner.

Here in this paper, we train such Stacked Auto-encoder network using the information of subject vehicle and surrounding vehicles mentioned in 3.1.

3.3 Training

As for the learning procedure, let $\{W^n_{\mathrm{ij}}, b^n_{\mathrm{ij}}\}$ denote the parameters $(\mathbf{W}, \mathbf{b})^n$ of the nth layer of our network where i and j are the number of inputs and outputs respectively at the nth layer. First we perform an unsupervised training on a denoising auto encoder and obtain the first learning parameters $\{W^1_{\mathrm{ij}}, b^1_{\mathrm{ij}}\}$, where the hidden layer h1 is connected to the input $\mathbf{x}$ by a weight matrix $\mathbf{W^1}$ forming the encoding step. The hidden layer then outputs to a reconstruction vector $\tilde{\mathbf{x}}$, using a tied weight matrix $(\mathbf{W^1})^T$ to form the decoder,

$$\mathbf{z} = \mathrm{f}(\mathbf{W^1}\mathbf{x} + \mathbf{b^1}) \tag{1}$$

$$\tilde{\mathbf{x}} = \mathrm{f}((\mathbf{W^1})^T\mathbf{z} + \mathbf{b'^1}) \tag{2}$$

The activation function is f and b is the bias term.

We use the mini-batch stochastic gradient descent (SGD) for the training procedure. Learning occurs via backpropagation using the following error function,

$$\mathrm{E} = \sum_{j=1}^{k}(x_i \log(\tilde{x}_i) + (1 - x_i)\log(1 - \tilde{x}_i)) \tag{3}$$

Next, we input the above parameters to the second layer of the auto encoder and perform an unsupervised training. The second learning parameters $\{W^2_{\mathrm{ij}}, b^2_{\mathrm{ij}}\}$ are then obtained. In the same way, we repeat this learning process for the every layer by using the parameters from the intermediate layer of the previous auto encoder as an input.

After completion of the above learning phase (i.e. pre-training phase) , all the trained layers are stacked on each other, and the learning parameters of each layer obtained are set as initial values of a new neural network. Then, by adding a softmax classifier that can classify future driving behaviour, it is possible to obtain a multi-layered neural network.

Finally, by performing a fine tuning phase, we update the parameters of the entire network with supervised learning. We illustrate this network in Figure 2.

3.4 Evaluating Our Model

Before moving on to our proposed method, which is to accommodate driving action prediction to the driver's likings, we try to evaluate our model's prediction performances for average driver anticipation. Here, we use the lane-change anticipation task as an experimental ground to evaluate the performance of our model.

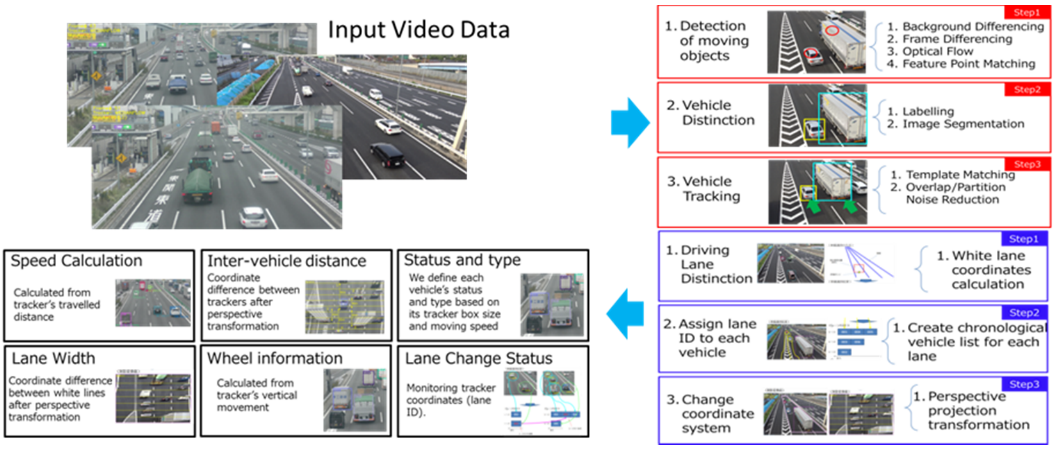

Figure 3: Training data parameters extraction process. We use conventional image processing techniques to track each vehicle appearing in the video data captured from a fix bird eye's view camera. Then we calculate the vehicle information necessary for our model as mentioned in 3.1.

Data Collection: In order to perform this evaluation, instead of installing a variety of multiple sensors on cars and hiring actual drivers to collect the data, which can be a very time-consuming and costly process, we opted to use video data of a different roads traffic that we captured by a fix bird's eye view high resolution camera.

And by analysing and processing these images, we extract the desired training and testing information data mentioned in 3.1. The extraction method is performed using various conventional image processing methods, but we won't go into deeper explanations considering that it is not the main purpose of this paper. A simple diagram of the data extraction process is shown in Figure 3.

We use 500 lane change cases for training and 150 cases for test. Our model has 200 units in the input layer, 2 hidden layers of 100 units each, and 2 units in output layer.

Experiment Results: We evaluate this model based on its correctness in predicting future lane changes. The anticipation is performed offline for each frame at 30fps where the algorithm processes the recent 2 seconds (60frames) context and assigns a probability to each of the two actions (lane change/ lane keep) happening 2 seconds (60frames) later.

We show the prediction results in Figure 4. Of the 150 cases where lane change occurred, 131 were successfully anticipated and the anticipation rate was 87.3%. On the other hand, 134 lane keep cases were anticipated out of 150, and the anticipation rate was 89.3%.

Using stacked autoencoders seems to perform well for the lane change anticipation, when compared to other methods (Li et al., 2015; Hou et al., 2013), even in this case where training data amount is too few.

Figure 4: Classification performance for lane change anticipation. Lane change is anticipated 2 seconds before it occurs. Anticipation rate is 88%.

4 ACCOMMODATING ANTICIPATION FOR INDIVIDUALS

As mentioned before, one of the most important tasks that self-driving faces is the ability to accommodate the way the car drives itself to every driver's taste. Here, we explain how to predict each driver's next action based on his own likings.

4.1 Transfer Learning

Traditional data mining and machine learning algorithms make predictions on the future data using statistical models that are trained on previously collected labelled or unlabelled training data (Yin et al., 2006; Baralis et al., 2008). Nevertheless, most of these assume that the distributions of this labelled and unlabelled data are the same.

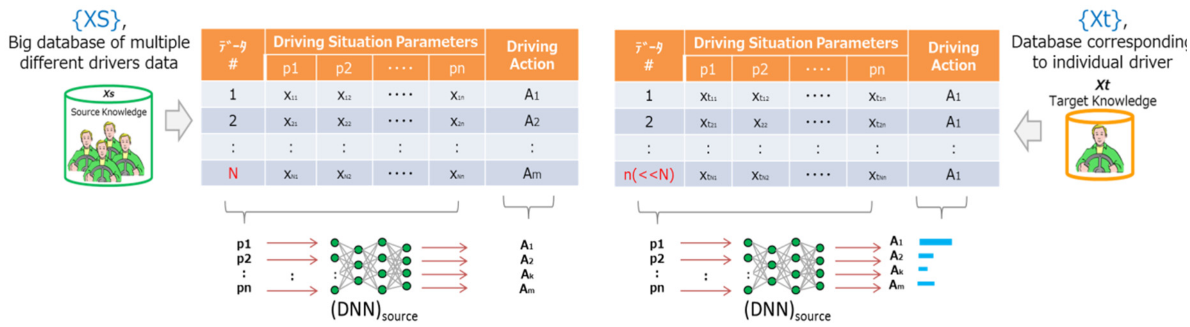

Figure 5: Method Diagram: First we train a stacked autoencoder NN on a big database of multiple different drivers {Xs}. This network predicts future driving actions and performs well for an "average" driver. Next we input individual data {Xt} into the network and compute output histograms (which is a representation of individual's driving characteristics). Finally, we re-train the network with said individual data to tune the parameters and adapt the model to said individual.

Transfer learning, in contrast, allows the domains, tasks, and distributions used in training and testing to be different.

Many examples of transfer learning have already been reported. For example Oquab et al. (Oquab et al. 2014) trained convolutional neural network (CNN) with the ImageNet (Deng et al., 2009) as the source knowledge. After training the CNN, they re-use the parameters from the input layer on the mid-level hidden layer. Then, they add a new layer and tune the parameters using the target knowledge. Also, in the medical domain, medical image datasets such as X-ray CT image datasets are hard to collect and do not have enough data for training the deep neural networks mainly because of privacy problems. Therefore, different datasets are used as source knowledge in order to solve a certain different target task (Sawada et al. 2015). The study of Transfer learning is motivated by the fact that people can intelligently apply knowledge learned previously to solve new problems faster or with better solutions.

4.2 Proposed Method

In this section, we propose a method that re-uses the network trained on a great number of various drivers data with different driving behaviors (henceforth: source knowledge) to improve driving behavior anticipation performance for every particular driver even in the case if we have only few information on said particular driver (target knowledge).

First, we train a stacked autoencoder neural network $(DNN)_{source}$ for anticipating driving behaviors using the source knowledge (i.e. a great number of various drivers data with different driving behaviors), as mentioned in the previous chapter. We note the parameters trained on the source knowledge as $(\mathbf{W}, \mathbf{b})_{source}$.

Secondly, we evaluate the relation between the source knowledge {Xs} and the target knowledge {Xt} corresponding to each individual driver data. To evaluate the relation between source and target, we input the target knowledge {Xt} into the deep neural network $(DNN)_{source}$ trained on the source knowledge. Then, we compute the histograms based on the response of the output layer. After computing the histograms, we select the variables of the output layer corresponding to the target domain. And finally, we tune the parameters $(\mathbf{W}, \mathbf{b})_{source}$ in such a way that the selected variables respond as the outputs of the target knowledge.

It should be noted that the tuning of $\mathbf{W}_S$ corresponds to the re-training of the deep neural network $(DNN)_{SRC}$ using the parameters $\mathbf{W}_S$ as initial parameters and {Xt} as training data. We show our method diagram in Figure 5.

5 EXPERIMENTAL RESULTS

In this section, for the purpose of theoretical confirmation of our method, we perform two different experiments using real vehicle driving data.

In the first experiment, we try to anticipate each driver's acceleration/deceleration behavior a few seconds before they occur. While on the second experiment in hope of getting more individual variability, we set our target to anticipating the braking profile of a driver. Below are the full details.

5.1 Car Speed Anticipation

Acceleration/deceleration (A/D) behaviour of vehicles is important for various applications like the determination of yellow light length at inter-section, ramp design etc. But it is also a very important aspect that can define a pleasant drive and that varies from driver to driver. Here, we try to anticipate each driver's A/D behaviour based on his likings.

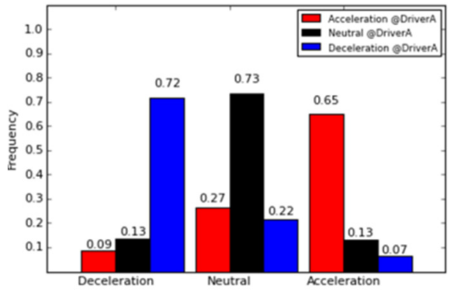

Figure 6: Histogram of relation between multiple various driver data (source knowledge) and individual Driver A data (target knowledge).

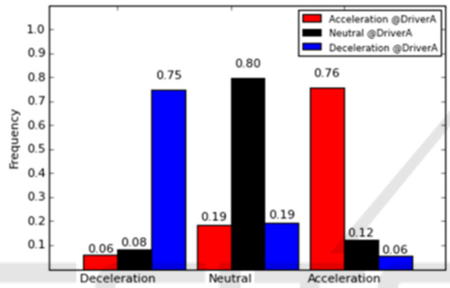

Figure 7: Experimental results shows improvement in anticipation accuracy for driver A by 3% for deceleration, 11% for acceleration and 7% for neutral status.

5.1.1 Experiment Overview

We use 150 hours of driving data from 30 different drivers as our source knowledge X_S. We use a separate 5 hours of driving data of a different 31st driver ("driver A") as the target knowledge Xt_A.

We define an acceleration/deceleration (A/D) behavior as an increase/decrease in speed by 3km/h within a 5seconds time period respectively, while a change of speed of less than 3km/h is counted as a neutral status.

In this experiment, our prediction model has 440 units in the input layer, 1000 units in the first and second hidden layer, and 3 units in output layer. We evaluate this model based on its correctness in predicting future (A/D) actions. A separate driving data of driver A which is not included in Xt_A is used for test. We anticipate actions every 0.5 seconds where the algorithm analyzes the recent driving context and outputs a probability to each of the three driving behaviors: acceleration, deceleration and neutral status that will occur 2 seconds in the future.

Figure 6 shows the computed histograms of the relation between source knowledge (i.e. A/D behavior based on multiple drivers' data trained network) and target knowledge (A/D behavior of driver A). The red, black and blue bars represent the frequency of acceleration, neutral status and deceleration respectively, for said particular driver A. As it is shown here, if we take for example the acceleration behavior, 36% of driverA's acceleration maneuvers were anticipated as deceleration or neutral status. In other words, a model trained by the source knowledge contains information from multiple various drivers and can be used to anticipate actions for an "average" driver, which does not perform so well for said particular driver A.

Next, we select the appropriate variables of the output layers that relate to the source knowledge X_S trained model (henceforth called average driver model), then we tune the network parameters by re-training the model using driver A data while keeping the average model initial parameters.

Figure 7 shows the histograms of the new relation between "driverA-accommodated" source knowledge and target knowledge. Experimental results show that our proposed method improves anticipation accuracy for said driver A by an average of 7%.

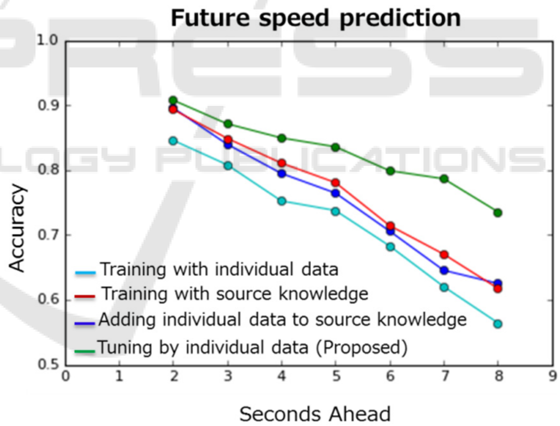

Figure 8: Proposed method performance compared to conventional methods. Our method shows better results for predictions at near and distant times in the future.

5.1.2 Performance Comparison

We compare our method to the following 3 cases:
(a) Using small amount of individual data for training (10 times fewer than source knowledge).
(b) Using source knowledge for training (i.e. average driver model)
(c) Adding individual data to source knowledge and performing training (non-transfer).

In the experiment above, we tried to anticipate driver A/D actions by predicting the car's speed behavior 2 seconds before occurring. Here, for the sake of completeness, we also evaluated our method for longer and shorter future time periods. Figure 8 shows the evaluation results.

Our method shows better results for predictions at near and distant times in the future. It is also worth pointing out that the closer the future we are trying to anticipate, the less likely it is for different drivers to take different actions, while on the other hand the further we go in the future driving behaviors become more likely to vary depending on the person driving. This explains the reason why our method performs better for distant future predictions.

5.2 Brake Profile Anticipation

The application of the brakes is one of, if not the most affecting, driving action that separates a pleasurable drive from an average or unpleasant one.

While some drivers prefer to brake long and slow, depending on the driving situation a fair share of drivers also enjoy a faster and more aggressive braking.

Predicting the way a car brakes is an important task in order to accommodate self-driving cars to the driver's taste.

In this experiment, we propose a model that calculates a braking profile depending on the surrounding situation and that can be tailored to each driver's liking.

5.2.1 Experiment Overview

We use the same driving data mentioned in 5.1.1, a total of 150 hours of driving performed evenly by 30 different drivers, in addition to another 5 hours from a different driver A. But in order for our model to get a better capture of braking features, we limit our data to brake-scenes only, and then extract the braking profiles to be used as training data.

We conduct our experiment on predicting the car's deceleration profile at the event of when brake pedal is hit. It is also important to mention that we consider the distance to complete stop (i.e. distance to stop line or front car) as a known parameter. Thus, by calculating the time to complete stop, we anticipate the car's speed at n different intervals in the future. In this experiment we set n to 10. We illustrate the definition of a brake profile in Figure 9.

In this experiment, our model has 450 units in the input layer, 1000 units in the first and second hidden layer, 500 units in the third layer and 10 units in output layer. We use {Xs} = 10000 and n = 10, and {Xt} = 1000.

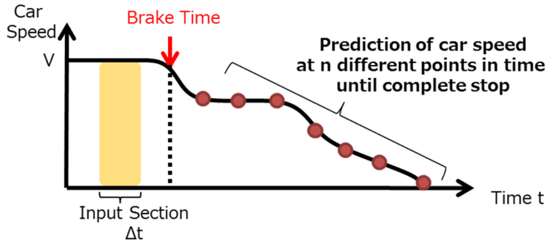

Figure 9: Brake Profile: At the event of a brake, our model anticipates car speed at n (=10 in this experiment) different points in time until complete stop of the vehicle.

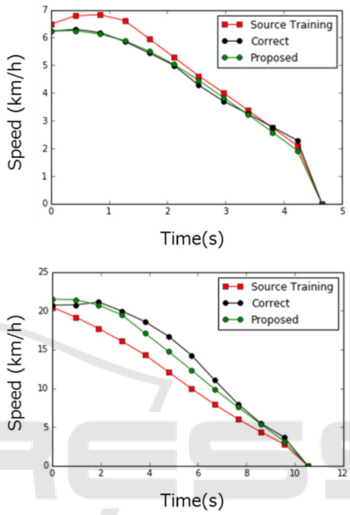

Figure 10: Samples of braking profile prediction: We compare our proposed model (green line) to average driver model (Red). The black line shows the actual brake profile. Our proposed method shows better results for short and long time brakes.

We train the deep neural network using the following objective function,

$$E = \frac{\sqrt{\frac{1}{n}\sum_{j=1}^{n}(y_i - \tilde{y}_i)^2}}{\hat{y}} + \sqrt{\frac{\sum_{j=1}^{n}(grd(y_i) - grd(\tilde{y}_i))^2}{n}} \quad (4)$$

where y is the true value and ỹ the predicted value. $grd(y_i)$ is the differential between observation i and i-1.

Table 1: Brake profile prediction model comparison.

Model	$RMSE_{value}$+$RMSE_{shape}$
{Xs} Trained	0.259
{Xs+Xt} Trained	0.240
Proposed	0.226

5.2.2 Experiment Results

Here, we consider two different aspects for evaluating our braking profile anticipation performance. We calculate the Root Mean Square Error ("RMSE") of the predicted values to measure our prediction in term of real values, and also we calculate the RMSE of the differential between observations to evaluate how well the prediction fits the "shape" of the brake profile.

We use a separate driving data of driver A as test data. The prediction accuracies of the average model driver trained with {Xs}, the average model driver trained with {Xs+Xt} and the driver A tuned model using our method mentioned in 4.2 are shown in Table 1. Figure 10 shows two examples of brake profile predicted with our method. Our method improves prediction accuracy by 12.5%.

6 CONCLUSIONS

In this paper, we considered the problem of anticipating driving actions a few seconds before they are performed. Our work also enables greater comfort and satisfaction by crafting user experiences sensitive to individual driver preferences.

We proposed a deep learning network that anticipates driving behavior estimation based on information of subject vehicle as well as surrounding vehicles and environment. We use the lane change anticipation task as an experiment ground to confirm the theory of our anticipation model, and we accomplished an accuracy of 88%.

We proposed a method which enables the anticipation of driving behaviors that can be tailored to each driver, leading to improved user experiences. Our method re-uses a network trained on a great number of various drivers' data with different driving behaviors and links it to a particular driver with particular taste to train a new model fitted to said driver.

We confirm our theory by predicting individual driver acceleration/deceleration behaviors as well as braking profiles a few seconds before occuring. Our method shows better results compared to conventional methods where individual data quantity is too few (around 1/10 of the source knowledge).

Furthermore, by applying this technology, we believe that estimating other than driving actions is also possible. For example, by analyzing driving behavior history or monitaring the driver's state and condition, it is possible to predict dangerous driving operations. We also think that building an ideal personalized driver model by using the driving behavior history of the model driver, can realize safe and comfortable driving support.

REFERENCES

HMI-TF briefing report, 2016, *SIP Autonomous driving promotion committee 22-3-1-1*, Cabinet Office, Government of Japan.

Inagaki, T., 2015, *Human-Machine Collaboration in Automated Driving*, IA TSS Review Vol.40, No.2.

Shia, V., Gao, Y., 2014, *Semiautonomous vehicular control using driver modelling*. IEEE Transactions on Intelligent Transportation Systems, 15(6).

Vasudevan, R., Shia, V., 2012, *Safe semi-autonomous control with enhanced driver modeling*. In ACC.

Fletcher, L., Loy, G., 2005, *Correlating driver gaze with the road scene for driver assistance systems*. Robotics and Autonomous Systems, 52(1).

Rezaei, M., Klette, R., 2014, *Look at the driver, look at the road: No distraction! no accident!* IEEE CVPR.

Herrmann, (Volkswagen), 2012, *Situation analysis for driver assistance systems at urban intersections, Vehicular Electronics and Safety (ICVES)*, IEEE International Conference on ,P.151.

Liebner, (BMW) , 2013, *Generic driver intent inference based on parametric models*, 16th International IEEE Conference on Intelligent Transportation Systems (ITSC) ,P.268.

NHTSA, 2014, *Human Factors Evaluation of Level2 and Level3 Automated Driving Concepts*, DOT HS 812044.

Hou, Y., Edara, P., 2013, *Modeling Mandatory Lane Changing Using Bayes Classifier and Decision Trees* IEEE Transactions on Intelligent Transportation Systems (Volume: 15, Issue: 2, April 2014).

Li, L., Zhang, M., 2015, *The application of Bayesian filter and neural networks in lane changing predictions,* 5th international Conference on Civil Engineering and Transportation , ICCET2015.

Yin, X., Han, J., 2006, *Efficient classification across multiple database relations: A crossmine approach*, IEEE Transactions on Knowledge and Data Engineering, vol. 18, no. 6, pp. 770–783, 2006.

Baralis, E., Chiusano, S., 2008, *A lazy approach to associative classification*, IEEE Transactions on Knowledge and Data Engineering,vol. 20, no. 2, pp. 156–171.

Oquab, M., Bottou, L., 2014, *Learning and transferring mid-level image representations using convolutional neural networks,*". CVPR 2014. Proceedings of the 2014 IEEE Computer Society Conference on, pp.18.

Deng, J., Dong, W., 2009, *Imagenet: A large scale hierarchical image database 2009*. CVPR 2009. IEEE Conference on IEEE, pp.248.

Sawada, Y., Kozuka, K., 2015, *Transfer Learning Method using Multi-Prediction Deep Boltzmann Machines*, Machine Vision Applications (MVA), 2015 14th IAPR International Conference.