Road Surface Type classification through Inertial Sensors and Deep Learning

Harishankar M[1]

Abstract— **Environmental perception is an important aspect when it comes to the design of Intelligent Transportation Systems. Both active and passive sensing methodologies are used for this. The use of inertial sensors in passive sensing methodologies to understand the environment is a rewarding field of study due to its cost-effectiveness and safety compared to other methods. In this study, we utilize ML-based methodologies such as Deep Neural Networks to gain an understanding of the road terrain through which the vehicle is moving and classify them. We use CNN-based deep neural networks to classify the road surface into cobblestone, dirt or asphalt-type roads. We train the ML model using time series data of various kinematic aspects of the vehicle, such as linear and angular acceleration and speed. The time series data is divided into windows and passed into the model for training. The computational cost required for this can be high. Hence, we look into correlation-based metrics for the various features which can reduce the input dimensionality to a great extent and validate its use as a suitable way to capture the properties of the multivariate time series data and classify it. Further, we also briefly look into the impact of class imbalance and relative difficulty in classifying between classes on the performance factors of the model.**

I. Introduction

Intelligent Transport Systems are considered the next major step in the transport and mobility industry. Extensive data is acquired through diverse sensors placed within the vehicle, aimed to generate situational awareness and enhance overall performance. Depending on the chosen methodology for data acquisition, the raw environmental and system data can be collected by either active or passive sensing techniques. In active sensing situations such as using radar, ultrasonic sensors and laser scanning, the sensors interact with the environment to produce the data. In the case of passive sensing, there is no active interaction with the environment; the physical data is sampled directly. The use of inertial sensors to get motion data and cameras for visual data are some of the most widely used passive sensing techniques.

Understanding the environment through which the vehicle is moving is crucial to creating intelligent transport systems. The knowledge of the terrain in which the vehicle is moving can help improve the traction and stability control of the vehicle, create adaptive suspension systems, and optimize automatic transmission. The requirement of understanding the terrain(surface) through which the vehicle is moving leads to the challenge of road surface type classification. The road surface can be identified by using data derived from passive sensing techniques using inertial sensors (measuring acceleration and speed). In comparison to active sensing, the passive approach is safer, non-polluting, and less costly. The usage of inertial sensors such as accelerometers and gyroscopes to obtain linear and angular acceleration information can provide great insights into environmental features, driving characteristics, and vehicle-specific properties. The use of machine learning-based methodologies for understanding the sensor data as well as deriving important deductions about the environment and vehicle is an important area of research.

[1]Department of Electrical Engineering, Indian Institute of Technology Hyderabad, India.

II. Literature Review

The use of inertial sensors for developing intelligent transport systems has been studied extensively by Menegazzo and von Wangenheim ([1], [2] [3]). [2] [3] gives an overall review and survey on the existing approaches and methodologies, as well as previous works for the development of intelligent transportation systems using various sensor data. [1] explored the possibility of using machine learning-based approaches centred on Deep Neural Networks to classify the road surface into the categories of asphalt, dirt or cobblestone. They have explored the use of deep neural networks, such as Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), for the road surface classification problem. The multivariate time series obtained from the vehicle were subjected to windowing, and the time windows labelled with the road surface type were inputted for training the machine learning model. We aim to reproduce the study using the same data sets using CNN-based Machine learning models.

As mentioned above, the input to the model is a multivariate time series window whose size can have a significant impact on the complexity of the model and training process. Ferdousi, Cohnstaedt and Scoglio [4] have explored the use of correlation-based metrics of the windowed time series data as features of the data from training and, thereby reducing the input size at the same time while maintaining the prediction accuracy. We extend this idea of using correlation-based metrics for training to our current study and explore it. The major objective is not to extract a model with the highest possible accuracy but to validate the possibility of using correlation as a metric for the classification of multi-featured time series in our current study.

III. Materials and Methods

The number of public data sets available on inertial sensor data of road vehicles is much less. The data set used for the study was obtained from Kaggle and published as part of a study of Inertial sensor data in intelligent transport technologies by Menegazzo and von Wangenheim at Universidade Federal de Santa Catarina, Brazil.

DataSet	Vehicle	Driver	Scenario	Distance
PVS 1	Volkswagen Saveiro	Driver 1	Scenario 1	13.81 km
PVS 2	Volkswagen Saveiro	Driver 1	Scenario 2	11.62 km
PVS 3	Volkswagen Saveiro	Driver 1	Scenario 3	10.72 km
PVS 4	Fiat Bravo	Driver 2	Scenario 1	13.81 km
PVS 5	Fiat Bravo	Driver 2	Scenario 2	11.63 km
PVS 6	Fiat Bravo	Driver 2	Scenario 3	10.73 km
PVS 7	Fiat Palio	Driver 3	Scenario 1	13.78 km
PVS 8	Fiat Palio	Driver 3	Scenario 2	11.63 km
PVS 9	Fiat Palio	Driver 3	Scenario 3	10.74 km

Fig. 1: Collected datasets

Hardware	Sensor	Data	Sampling Rate
HP Webcam HD-4110	Camera	720p Video	30 Hz
Xiaomi Mi 8	GPS	Speed in m/s, latitude, longitude, etc.	1 Hz
MPU-9250	Accelerometer	3-axis acceleration in m/s²	100 Hz
MPU-9250	Gyroscope	3-axis rotation rate in deg/s	100 Hz
MPU-9250	Magnetometer	3-axis ambient geomagnetic field in μT	100 Hz
MPU-9250	Temperature	Sensor temperature in ◦C	100 Hz

Fig. 2: Sensors used with sampling rate

A. Data Description

The data set includes the following data obtained by various sensors:

- Linear acceleration in three directions
- Angular acceleration in three directions
- Speed
- Temperature, GPS data and Magnetometer data

The data set includes data from three different vehicles, each driven by a different driver and driven over three different scenarios in distinct geographical locations(Figure 1). The dataset comprises the time series data of the various sensors, including linear and angular acceleration in all three directions. The sensors were placed at different points, such as the dashboard, below suspension and above suspension, to obtain three sets of acceleration data. The overall speed, as well as the longitude and latitude position of the vehicle, were collected. Since the data points are digitally sampled versions of the physical data, the rate of sampling is also important(This is shown in Figure 2)

At each time point, the road surface in which the vehicle was moving was noted and was found to belong to one of the classes of either dirt(class 0), cobblestone(class 1) or asphalt (class 2)(Figure 3).

B. Methodology

For our study, we utilize the linear and angular acceleration along the three axes collected below the suspension and the vehicle's speed as the features. We supply a sequence of time instants to the machine learning model for classification.

Fig. 3: Types of roads

Each window will have seven features. Each window's label corresponds to the last sample's label in the window. The window is inputted into the model via two approaches, which are described as follows;

- Approach 1: The window with a fixed length and seven features(representing the columns) is inputted as such to the ML model. Thus, the input will be a matrix with dimensions size of window * 7.
- Approach 2: The correlation between the window's column vectors(features) is considered for input. The correlation between different column vectors corresponding to various accelerations is calculated, and a correlation matrix is generated. The correlation metric used is Pearson's correlation, which is calculated as follows

  $$\rho_{xy} = \frac{cov(xy)}{\sigma_x \sigma_y}$$

  Here, x and y are the column vectors corresponding to different accelerations(both linear and angular), cov denotes the covariance and σ indicates the standard deviation of the corresponding column vector. Since the correlation matrix is symmetric, use only the elements above the diagonal of the matrix, which will result in 15-element vectors on flattening. We also input the average speed of the vehicle within the window as another feature(since it remains constant most of the time due to the higher sampling rate of acceleration sensors). Hence, overall, we pass a 16-dimensional vector as the input to the machine-learning model representing each window for classification.

C. Data Preprocessing

The time series from the sensor was divided into windows, each size 300, and each window was assigned the road surface type label corresponding to the last data point in the window. The division into windows is done in two ways. In Case 1, we do not overlap between the windows, whereas in Case 2, we have a 50 per cent overlap between each window.

The linear acceleration and angular acceleration along the three directions, along with the speed, combine to form the seven features used for training. The inertial sensor data from different terrains can have different statistical properties. Hence, data normalization was done within each window using min-max scaling. The model learns data from all vehicles and drivers for some scenarios but not all vehicles with all drivers for all scenarios. The datasets PVS 1,2,4,5,7,8 were used for training and validation(80 per cent for training and 20 per cent for validation). The datasets PVS 3 6 9 were used for testing as they are generated in a different geographical location compared to the training and validation data set. Hence, it will give us the performance of the model in a new environment.

D. Model Used and Training

We use Convolutional Neural Networks(CNN) for the purpose of classification. We have a total of 4 classifiers: two(one for input with non-overlapping windows and one for windows with overlapping windows) for each approach. They are;

- Classifier A: Input with a non-overlapping time series window and inputted as described in approach 1.
- Classifier B: Input with overlapping time series window and inputted as described in approach 1.
- Classifier C: Input with a non-overlapping time series window and inputted as described in approach 2.
- Classifier D: Input with overlapping time series window and inputted as described in approach 2.

Classifiers A and B: The architecture for these models consists of multiple convolutional layers to capture the local patterns and spatial hierarchies in time. They also have hidden dense layers coupled with dropout and batch normalization layers for regularization.

Classifiers C and D: The architecture for these models is comparatively simpler, with a few hidden dense layers and dropout layers (for regularization).

The choice of the loss function and weightage of each class(in the case of imbalanced classes) is also an important factor which determines the accuracy of the model. We experiment with categorical cross entropy as the loss function. First, we use the loss function without specifying any class weights(all classes have the same weightage). Later, we use the inverse class frequencies in the training data as the class weights for the same loss function and compare the results we get with the former loss function without using class weights.

The train flow diagram shown in Figure 4 describes the different models and approaches used :

Fig. 4: Training flow diagram

IV. Experimental Results

A total of 200 epochs were run for each classifier, and the model with the least validation loss was selected as the best model since validation loss is a good metric for the performance of the model for unseen inputs. The evaluation metrics used for assessing each model are f1 score, precision, recall and accuracy.

1) Case 1: classes have equal weightage: The per-class evaluation metrics of various classifiers are shown in the table below.

Classifier	Class	Accuracy	Precision	Recall	F1-Score
Classifier A	0	0.658	0.706	0.658	0.681
	1	0.725	0.700	0.725	0.713
	2	0.984	0.969	0.984	0.976
Classifier B	0	0.726	0.732	0.726	0.729
	1	0.766	0.752	0.766	0.759
	2	0.979	0.985	0.979	0.982
Classifier C	0	0.801	0.671	0.801	0.731
	1	0.632	0.754	0.687	0.632
	2	0.968	0.977	0.973	0.968
Classifier D	0	0.769	0.679	0.769	0.721
	1	0.679	0.730	0.704	0.679
	2	0.957	0.989	0.973	0.957

The confusion matrix for classifiers A and C obtained are shown in Figure 5:

The mean f1 score, precision, recall and accuracy of the various classifiers are summarised as follows:

Classifier	Accuracy	Precision	Recall	F1-Score
Classifier A	82.70	82.52	82.70	82.58
Classifier B	85.33	85.41	85.33	85.37
Classifier C	83.00	83.52	83.00	82.96
Classifier D	83.02	83.58	83.02	83.19

2) Case 2: With class weights as the inverse of class frequency : The per-class evaluation metrics of various classifiers are shown in the table below.

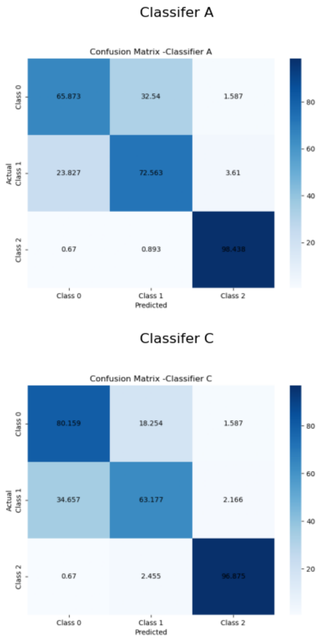

Fig. 5: Confusion matrix for Classifiers A and C

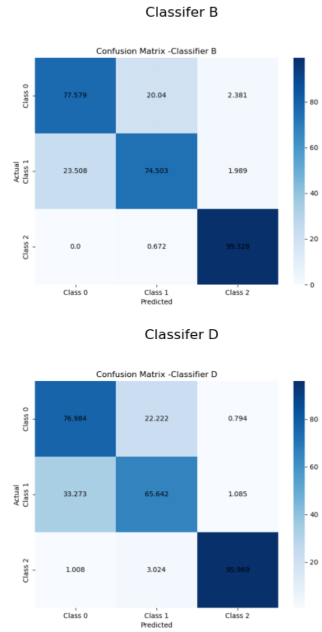

Fig. 6: Confusion matrix for Classifiers B and D

Classifier	Class	Accuracy	Precision	Recall	F1-Score
Classifier A	0	0.706	0.781	0.706	0.741
	1	0.794	0.753	0.794	0.773
	2	0.991	0.971	0.991	0.981
Classifier B	0	0.775	0.750	0.775	0.763
	1	0.745	0.793	0.745	0.768
	2	0.993	0.975	0.993	0.983
Classifier C	0	0.778	0.700	0.778	0.736
	1	0.696	0.745	0.696	0.720
	2	0.971	0.993	0.971	0.982
Classifier D	0	0.769	0.667	0.769	0.715
	1	0.656	0.723	0.656	0.688
	2	0.959	0.988	0.959	0.973

The confusion matrix for classifiers B and D obtained are shown in Figure 6:

The mean f1 score, precision, recall and accuracy of the various classifiers are summarized as follows:

Classifier	Accuracy	Precision	Recall	F1-Score
Classifier A	86.18	86.05	86.18	86.04
Classifier B	86.67	86.54	86.67	86.57
Classifier C	84.34	84.72	84.34	84.45
Classifier D	82.46	83.03	82.46	82.59

V. Inference

Classifiers A and C(with equal as well as unequal class weightage) take time series windows which are not overlapping and input them using approaches 1 and 2, respectively. It was observed that the mean evaluation metrics, such as accuracy, precision, recall and f1-score, lie in a close range between 80-85 per cent, with classifier A slightly better than classifier C for equally weighed classes and vice versa in the case of unequally weighted classes. When it comes to classifiers B and D(with equal as well as unequal class weightage), which take time series windows which are not overlapping and input them using approaches 1 and 2, respectively, it is seen that the classifier B is slightly ahead of classifier D in all aspects of performance metrics being assessed, but the differences are not very high. Upon summarizing, we can say that the correlation metric-based approach gave a similar performance as compared to supplying the time series directly. This result is highly rewarding in the aspect of computational cost as the model required for the correlation-based approach is relatively simpler compared to that used in a regular approach.

Upon looking into the per-class evaluation metrics, such as f1 score, recall and precision of all the classes and the confusion matrix, the evaluation metrics of class 2(95-100 per cent) are very high compared to class 0 and 1(60-80 per cent). The per-class performance was increased for classes 0 and 1 by using inverse class occurrence frequency as class weights. This indicates an imbalance in class distribution in the training set. Even after using unequal class weights, the performance of classes 0 and 1 was far behind class 2. This can be inferred as the model being unable to accurately classify between classes 0 and 1(evident from the confusion matrices also) and easier identification of class 2.

VI. Conclusion

Thus, from the above results, it can be concluded that the correlation-based metric is a good measure of the features of the inertial sensor time series and can be used for classifying the time series window. The good correlation between the various dynamical aspects of the vehicle, such as acceleration and speed, enables us to successfully classify(with a simpler ML model) the road surface based on the multi variate time series of the kinematic characteristics of the vehicle. However, the choice of the correlation metric still needs to be optimized. The Pearson correlation for the above study can be affected by the presence of outliers in the data(an anomaly in the sensor data due to the occurrence of speed bumps, cross junctions, etc.) and can affect the performance of the model. Hence, the use of correlation metrics has to be explored. Although the use of inverse class occurrence frequency reduced the problem of class imbalance, the per-class performance has to be improved for classes 0 and 1. This aspect needs to be explored by using different loss functions, such as focal loss, which uses down-weighting to overly confident classes(in this case, class 2) , thus giving more attention to class 0 and 1.

REFERENCES

[1] J. Menegazzo and A. von Wangenheim, "Multi-contextual and multi-aspect analysis for road surface type classification through inertial sensors and deep learning," in *2020 X Brazilian Symposium on Computing Systems Engineering (SBESC)*, pp. 1–8, 2020.

[2] J. Menegazzo and A. Von Wangenheim, "Vehicular perception and proprioception based on inertial sensing: a systematic review," *Brazilian National Institute for Digital Convergence-Technical Reports*, 2018.

[3] J. Menegazzo and A. von Wangenheim, "Vehicular perception based on inertial sensing: A structured mapping of approaches and methods," *SN Computer Science*, vol. 1, no. 5, p. 255, 2020.

[4] T. Ferdousi, L. W. Cohnstaedt, and C. M. Scoglio, "A windowed correlation-based feature selection method to improve time series prediction of dengue fever cases," *IEEE Access*, vol. 9, pp. 141210–141222, 2021.