Extracting information from driving data using k-means clustering [*]

Nour Chetouane,[‡] Lorenz Klampfl and Franz Wotawa
Christian-Doppler Laboratory for Quality Assurance Methodologies for Autonomous
Cyber-Physical Systems, Institute for Software Technology, Graz University of Technology
Inffeldgasse 16b/2, Graz, Austria
{nour.chetouane, lklampfl, wotawa}@ist.tugraz.at

Abstract

There is an increasing availability of data, but for making decisions and other tasks we need information. Hence, we require to analyze the data and extract parts or come up with relations between different pieces. In this paper, we focus on information extraction within the automotive industry. In particular, we report on applying k-means clustering for identifying episodes in vehicle data. An episode is considered to be a time interval where a vehicle is performing an activity worth being distinguished. The underlying idea is to cluster the data such that we are able to extract such similar situations like breaking before a crossing only considering vehicle data. We discuss a method that allows extracting such episodes capturing actuator and sensor readings over time. Besides introducing the underlying method, we present obtained empirical results making use of a freely available dataset showing that the extracted episodes have indeed a meaningful interpretation.

1 Introduction

We live in a world of increasing availability of data. However, for obtaining information, i.e., data with uncertainty of interpretation removed, required to fulfill certain tasks, we need to analyze available data and set it in relation to a context. This may also lead to removing redundancies as well as coming up with relations between different pieces of data worth being considered in a given application context. In this paper, we focus on the automotive domain. Current vehicles produce a lot of data obtained during driving. Data include vehicle speed, breaking pressure, or the position of the steering wheel, and can be obtained when monitoring the respective bus systems of a car. When driving, however, we see a limited amount of situations worth being distinguished. This includes braking before a crossing or accelerating after stopping. The question now is, whether we are able to ”see” such distinguishable situations also in vehicle data.

In order to answer the question, we propose an approach utilizing clustering for obtaining time intervals we call episodes, and to evaluate whether those episodes can be assigned a meaningful interpretation. The underlying idea behind the approach can be summarized using the overall considered data analysis process depicted in Figure 1. We start with time series data and apply clustering. Ideally, the clusters comprise data points that are falling within a certain time interval. In a second step, we are considering time episodes for clusters and select one of these as representative.

In order to show that the approach really work in practice, we carried out an experimental evaluation relying on the freely available dataset from Audi [2]. This dataset comprises vehicle data but also images from attached cameras allowing us to interpret obtained episodes. Besides a detailed description of the evaluation, we discuss the obtained results.

Applications of our approach in the automotive industry include extracting episodes for testing and in particular test case generation. We can use the episodes in two different ways. First, we make use of episodes for concretizing abstract test cases. An abstract test case state a sequence of actions like accelerating, braking, turning left or right, or driving constant speed. The episodes themselves allow to concretize those abstract actions considering the concrete values for acceleration, braking, etc. Second, the episodes provide means for basic behavior that shall be considered in testing. The extracted episodes in a more abstract meaning provide situations that occur during driving. Hence, we may use these episodes as basic actions for generating arbitrary sequences of actions to be executed for testing.

[*] The financial support by the Austrian Federal Ministry for Digital and Economic Affairs, the National Foundation for Research, Technology and Development and the Christian Doppler Research Association is gratefully acknowledged.

[†] Authors are listed in alphabetical order.

[‡] DOI reference number: 10.18293/SEKE2021-118

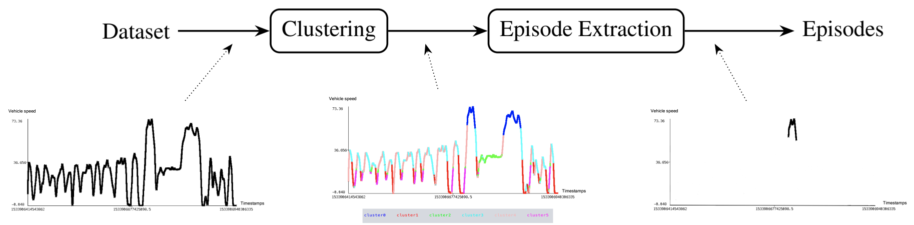

Figure 1: Underlying architecture showing the overall process from the initial dataset to episodes extraction.

The paper is organized as follows. In the next section, we discuss and formalize our episode extraction approach in detail. Afterwards, we introduce the processes carried out for evaluating the approach and present the obtained evaluation results. Finally, we conclude the paper and discuss future research.

2 Information extraction using k-means

In the following, we outline our clustering and episode extraction algorithm in detail. We start discussing the data and requirements on them. For the approach, we assume a set of data $\mathcal{D}$ over time provided for given attributes $a_1, \ldots, a_n$. We assume $\mathcal{D}$ is of the form $\{v_{t_0}, \ldots, v_{t_m}\}$ where v_t is a tuple $(v_1, \ldots, v_n)$ at time point t storing values of their corresponding attributes. We, furthermore, assume that the given dataset is already cleansed and attribute values are available for all points in time. In addition, and for simplification, we assume that the delta time between t_i and t_{i+1} is $\Delta t > 0$ for all i from 0 to $m-1$.

It is worth noting that in practice data may not fulfill these requirements and need to be cleansed and modified. For example, vehicle data often does not follow the requirements regarding time. There maybe no centralized clock, which would be necessary to assure that values are captured at specific points in time. Hence, we need procedures for mapping the original data to the form that is required. This may include making approximations or assumptions, e.g., using splines for interpolating values or assuming that values do not change unless otherwise stated. For vehicle data these modifications seem not to be a problem, because of the frequency used to obtained sensor data.

In this study we propose an approach mainly based on clustering analysis, the general principle behind clustering is to maximize the similarity between elements of the same cluster and to also maximize the dissimilarity of elements from different clusters. The main advantage of clustering analysis is procured especially when labeled data is unavailable. Clustering application has been revealed useful in various domains, such as text mining, information retrieval and data annotation. We can find a large survey about applications of clustering analysis in [6]. In the automotive domain, clustering analysis has been largely applied to explore different datasets. Here, we state some of the studies like [3] where the authors introduced a density-based clustering algorithm to cluster vehicle trajectories. In [5] a framework was proposed to automatically label congestion patterns using hierarchical clustering. Also, in [1], the authors investigated how clustering can be used to extract real-world manoeuvers for autonomous vehicle validation and compared it to other machine learning techniques. In the first step of the approach, we apply clustering and in particular k-means clustering, which is a well known machine learning algorithm [4] that is simple, easy to use and has been shown effective for serving several machine learning and data mining purposes. It mainly consists in splitting a set of unlabeled data into a fixed number k of clusters. K-means clustering works as follows: First, k initial centroids are randomly chosen from the original dataset. At each iteration, the algorithm goes through the data points and computes the distance between each point and the k centroids. The distance can be computed using for example Euclidian distance for numerical data or other types of distance measures depending on the type of data to be clustered. Each data point is then assigned to the cluster having the nearest centroid. After partitioning all data points, each centroid is re-calculated as the mathematical mean of each cluster, i.e. the sum of all the data points belonging to that cluster divided by the number of elements in the group. The process of data points partitioning and centroid adjustment is repeated until each centroid value is stable giving the final clustering of the input data.

Using k-means and given a certain value for the number of clusters k to be computed, we obtain the clusters $C_1, \ldots, C_k$ where $i \in \{1, \ldots, k\} : C_i \subseteq \mathcal{D}$. Note that all clusters are distinct, i.e., for all $i, j \in \{1, \ldots, k\}$ where $i \neq j$: $C_i \cap C_j = \emptyset$. It is worth noting that we are not considering time as an attribute when clustering. Furthermore, clusters may provide partitions over time that are not connected. We assume to points of data v_t and $v_{t'}$ from $\mathcal{D}$ to be connected if and only if $|t - t'| = \Delta t$. We call these two points approximately connected if $|t - t'| \leq m \cdot \Delta t$ for any integer value m. A subset of a dataset is said to be (approximately) connected if all data points in there are (approximately) connected. In the second step we select a cluster C_i and extract a connected subset. A connected subset of a cluster C_i can be formally defined as follows: $C_i^c \subseteq C_i$ such that $\forall v_t \in C_i^c \rightarrow \exists v_{t'} \in C_i^c : |t - t'| \leq m \cdot \Delta t$

Note that such a subset C_i^c may not comprise all data points of the original set $\mathcal{D}$ between the minimum time and the maximum time of C_i^c. Hence, we need to complete such a set using missing data points from $\mathcal{D}$ resulting in an *episode* of Cluster C_i. An *episode* of an approximately connected subset is a set comprising all elements of the subset and all elements of the original dataset $\mathcal{D}$ that fall within the time interval of the subset but have not been considered. Formally we define a function $\mathbf{E}$ on approximately connected subset returning an episode as follows: $\mathbf{E}(C_i^c) = C_i^c \cup \{v_t | \exists v'_{t'}, v''_{t''} \in C_i^c, t' < t < t'' : v_t \in \mathcal{D} \setminus C_i^c\}$

In this case, we also write E_i for referring to an episode that belongs to the connected subset C_i^c, i.e., $E_i = \mathbf{E}(C_i^c)$.

The algorithm EE summarizes the discussion on how episodes for a given dataset are computed:

Algorithm 1 $\mathsf{EE}(\mathcal{D}_I, m, k)$

```
Input: An initial dataset $\mathcal{D}_I$, the value $m$ used for computing approximately connected subsets, and the number of clusters $k$
Output: a set of k episodes.
Let $Sols$ be $\{\}$
Let $\mathcal{D}$ be the cleansed and modified set of data originating from $\mathcal{D}_I$.
Let $C_1$ to $C_k$ be the $k$ clusters obtained calling k-means.
for $i = 0$ to $k$ do
    Let $C_i^c$ be one approximately connected subset of cluster $C_i$ considering the parameter $m$.
    Let $E_i$ be $\mathbf{E}(C_i^c)$.
    Add $E_i$ to $Sols$.
end for
return $Sols$
```

Algorithm EE obviously terminates. Its computational complexity is determined by k-means clustering. Hence, in the worst case the runtime is exponential.

3 Experimental evaluation

The **objective** behind the experimental evaluation outlined in this section is to show whether k-means clustering works on real world driving data and allows deriving distinguished driving scenarios having a meaningful interpretation, like braking before stopping in front of a crossing. In the following, we discuss the setup of the evaluation and results obtained.

Setup: In order to carry out the experiments, we make use of the public available Audi Autonomous Driving Dataset (A2D2) [2]. It includes images and 3D point clouds, semantic segmentation, instance segmentation, plus automotive bus data. In this study, we focus on the vehicle bus data which corresponds to three different driving scenarios recorded in three cities in Germany: Gaimersheim, Ingolstadt and Munich. The data comprises 22 attributes with corresponding timestamps and units. Several sensors are used to measure for example; acceleration pedal, (angular) velocity, GPS coordinates, brake pressure, pitch and roll angles, steering angle, vehicle speed, etc. A2D2 dataset also includes sequential camera images corresponding to each city, we have made use of the camera front images in a second step of our experiment in order to map clustered episodes to sequences of videos and check whether the clustering is capable of finding similar scenarios.

The approach is implemented in Python 3 and for running the k-means clustering algorithm, we make use of python-weka-wrapper3 [1] package which runs different machine learning algorithms from the open source library WEKA [2]. For carrying out the experimental evaluation we use a MacBook Pro (2017) with a 2.8 GHz Intel Core i7 processor running under Mac OS High Sierra Version 10.13.

Before conducting the clustering, we first perform a data pre-processing step. As in the original dataset, each attribute a (sensor) values were recorded in a different time axis. To carry out clustering on data points, we performed data interpolation using same time axis for all attributes. Therefore, we looked for the minimum and maximum recorded timestamps for all sensors, then, created a common time line for all attributes by setting t_0 as the minimum recorded timestamp and continue to add the smallest time difference $|t - t'|$ between all recorded timestamps of all the data sensors, until reaching the maximum timestamp recorded in the data. To make the data interpolation, we used a Cubic Spline function which calculates an interpolating polynomial that has small error. The interpolation simulates each function corresponding to an attribute a with the original values recorded at an initial different Δt, to be used afterwards to compute new data points given as input the new created timeline for all the data attributes. For mapping bus signals to corresponding camera images, we have also performed an interpolation on images timestamps to synchronize them with the bus signals. Further on, in order to achieve clearer interpretation and obtain more precise results, we have carried out data cleansing where all values of the brake pressure attribute which are $<= 0.2$ were set to 0.

Results: During experiments, we focused on four attributes: acceleration pedal $[\%]$, brake pressure $[bar]$, steering angle $[°]$, and vehicle speed $[km/h]$. We run experiments with different values of k. After several trials, we noticed that clustering with $k = 6$ yielded to a better separation of clusters. We have also made our choice by evaluating the similarity between the obtained episodes in every cluster. For this, we computed the Pearson correlation co-

[1] see https://pypi.org/project/python-weka-wrapper3/

[2] see https://www.cs.waikato.ac.nz/ml/weka/

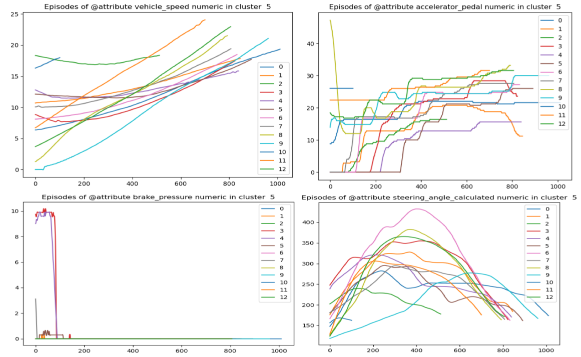

Figure 2: Extracted episodes for attributes: vehicle speed, acceleration pedal, braking pressure, and steering angle in cluster number 5 in the Gaimersheim example.

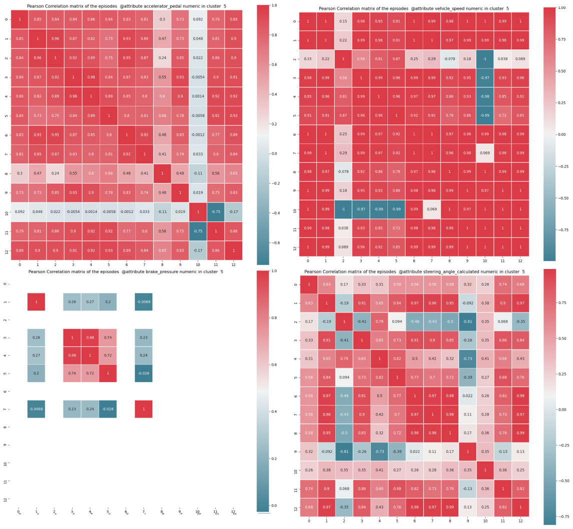

Figure 3: Heatmaps representing Pearson Coefficient between episodes for attributes: acceleration pedal, vehicle speed, braking pressure, and steering angle, in cluster number 5 in the Gaimersheim example.

efficient to measure the strength of the linear relationship between each pair of episodes. In the following, we report the results for an example of the obtained clusters when performing k-means with $k = 6$ on the Gaimersheim dataset. Figure 2 represents the obtained driving episodes in cluster 5 when performing k-means with $k = 6$. It shows, for each of the four selected attributes, graphs of the data points which get clustered in cluster 5. This cluster includes highly similar episodes and represents a turning maneuver. As we can notice in Figure 2, most episodes are showing pedal acceleration values and vehicle speed progressively increasing, basically no braking is occurring, and a parabolic curve corresponding to the steering angle showing episodes values increasing from $100°$ to approximately $350°$ and decreasing back to around $150°$ which highly indicates a turning maneuver.

We computed Pearson coefficient between each pair of episodes as the covariance of the two episodes values divided by the product of the standard deviation of each episode. A score close to 1 shows a large positive correlation, whereas a score close to -1, indicates a large negative correlation and equal to 0 means no significant correlation exists between the two variables. Figure 3 presents four heatmaps, each corresponds to one attribute. The color red indicates a high Pearson correlation coefficient between majority of the episodes, for the attributes: acceleration, steering angle and vehicle speed. For the braking pressure attribute, some of the coefficient values obtained show high correlation, noting that several correlation scores were not computed since the Pearson coefficient cannot be measured if one of the variables is 0 which is the case for braking pressure attribute as all values $<= 0.2$ were set to 0 in the data cleansing step. When performing clustering using the Gaimersheim dataset example, we mainly noticed high correlation for the vehicle speed attribute mostly for all clusters which results of the fact that k-means is mostly using vehicle speed as the dominant attribute in clustering the data. This also can be explained by the fact that there is more variation in vehicle speed values in the used dataset than the other selected attributes values (see Figure 4). We have also measured the probability density distribution for episodes in each cluster. Figure 5 shows four histogram plots corresponding to each attribute in cluster 5 for $k = 6$. We can see that the majority of episodes pedal acceleration values are more or less similarly distributed, as they are mostly arranged between 0% and 30%. Regarding the speed, the majority of episodes have a maximum value reaching $20km/h$. Vehicle speed values in this cluster are slowly elevating from $5km/h$ to around $20km/h$. Similarly with the steering angle, as the values are equally spread as they are increasing from $100°$ and the majority of episodes values exceed $250°$, and some of them even reach $350°$ as shown in Figure 2. For the brake pressure, it is 0 for all episodes.

In Table 1 we report and interpret information extracted from each cluster for the three dataset examples. Based on the changes in the graphs of episodes for every attribute, we could see that k-means clustering could actually separate, to a certain level, different driving situations in different clusters and we were able to observe and interpret similar driving scenarios represented by the episodes in the same cluster. It is worth noting that we further verified our interpretations by mapping episodes to sequences of videos created using corresponding front camera images. When observing clustered episodes, we could distinguish several driving scenarios, for example, an increase of vehicle speed from 0 to

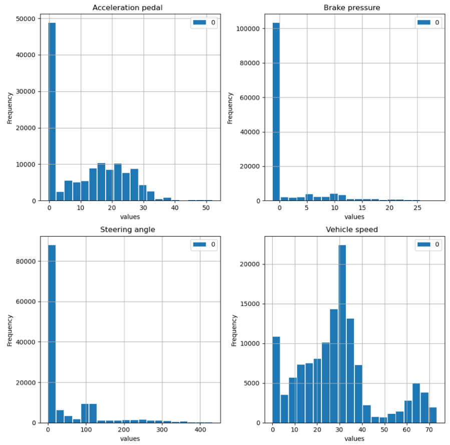

Figure 4: Frequency distribution of attributes: acceleration pedal, vehicle speed, braking pressure, and steering angle in the original dataset in the Gaimersheim example.

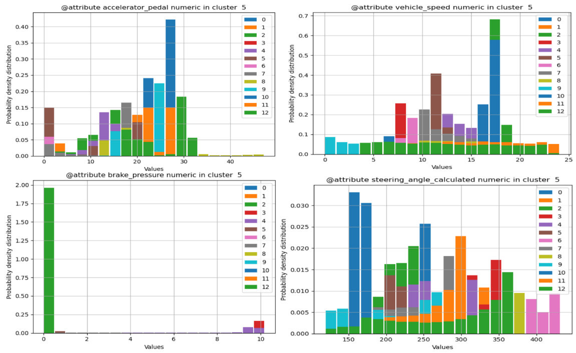

Figure 5: Probability Density Distribution of episodes for attributes: acceleration pedal, vehicle speed, braking pressure, and steering angle, in cluster number 5 in the Gaimersheim example.

a certain value indicates that the vehicle was stopping and starting back again, a high decrease in vehicle speed that reaches 0 along with a brake pressure occurring shows that the car is braking before stopping in a crossroad or a traffic light, also a progressive decrease of vehicle speed and acceleration pedal indicates that the vehicle is approaching an obstacle or a crossroad. A high decreasing or increasing in steering angle refers to the vehicle turning.

For the Munich and Ingolstadt dataset examples, we have noticed some clusters describing similar driving scenarios as these two datasets were recorded in a high traffic environment, for example the vehicle braking when approaching an obstacle like a pedestrian crossing the street or another car during traffic jam. We also found new driving situations for instance a car completely stopping at a red traffic light (cluster 0 in Munich dataset) which didn't occur in the first Gaimersheim example as this one was recorded in low traffic, noting that Gaimersheim example does not include crossroads with traffic lights.

Nevertheless, some similarities were seen between cluster 0 and cluster 2 in the Ingolstadt example which led to few misclassifications. Cluster 1 represents the vehicle driving after making a turn in a crossroad as it shows a sudden decrease in steering angle values while in cluster 2, episodes show the vehicle after crossing or stopping in a traffic light with no turning occurring however we could see two episodes from cluster 2 showing the car driving after making a turn. Also, another limit, is that we couldn't differentiate for instance in cluster 1 in Munich example and cluster 2 in Ingolstadt example, whether the car was stopping in a traffic light or in a traffic jam or because of an obstacle. Yet, we plan to tackle this issue in future work by adding additional information to the clustering algorithm with the use of an object detector.

In summary, we state the k-means clustering was to a certain level able to group together similar driving scenarios. Some clusters included highly correlated episodes such as the ones representing turning, stopping and braking before stopping. We were also able to analyze the clusters results based on the changes in the values of each attribute and come up with reasonable interpretations using the available camera images.

Threads of validity: Regarding internal threads we have to say that the analysis regarding the interpretation of clusters was done manually. Hence, the reported results are to an extent subjective. However, two of the authors carried out the analysis separately to mitigate this thread. Furthermore, we did data cleansing and modifications before carrying out the study to assure that the data meets the requirements. We believe that these changes as described in this paper are reasonable and should not influence the outcome of the evaluation. External threads include the use of a particular dataset and, hence, generalizability of conclusions may be in question. However, the dataset comprises at least different driving routes and situations. Nevertheless, further studies also including different application areas are required.

4 Conclusions

In this paper we introduced an approach for extracting information from real world driving data based on k-means clustering. We mainly try to answer the question whether k-means clustering is able to partition similar time episodes of driving into same groups, each describing a distinct driving scenario. We also tried to investigate whether it is possible to deduce meaningful scenarios interpretations based on the clustered episodes and verified them using camera

Table 1: Number of episodes ($\#$) in each cluster (C) and description of corresponding driving scenarios using the attributes: acceleration pedal [$\%$], braking pressure [bar], steering angle [$^\circ$] and vehicle speed [km/h] for every dataset example.

Gaimersheim		**Munich**		**Ingolstadt**	
C/$\#$	**Scenario description**	**C/$\#$**	**Scenario description**	**C/$\#$**	**Scenario description**
C0/2	**Driving straight on a state highway** Accelerator pedal (avg=17.245) No braking pressure (avg=0.006) Steering angle (avg=2.998) Vehicle speed (avg=64.34)	C0/8	**Car stopping** No acceleration pedal occurring (avg=0) High braking pressure (avg=32.288) Steering angle (avg=13.209) Vehicle speed (avg=0.134)	C0/38	**After crossing or after stopping at red traffic light** Acceleration pedal (avg=7.804) No braking occurring (avg=0) Steering angle decreasing ($\sim$150$\rightarrow$$\sim$0, avg=13.344) Vehicle speed progressively increasing ($\sim$5$\rightarrow$$\sim$25, avg=18.848)
C1/18	**Braking before a crossroad** Very low acceleration pedal (avg=0.008) Brake pressure (avg=10.218) Steering angle (avg=35.172) Vehicle speed decreased ($\sim$30$\rightarrow$$\sim$0, avg=7.842)	C1/29	**After crossing or after stopping at red traffic light or at traffic jam** Acceleration pedal (avg=15.988) No braking occuring (avg=0) Steering angle decreasing ($\sim$200$\rightarrow$$\sim$0, avg=63.297) Vehicle speed increasing progressively ($\sim$5$\rightarrow$$\sim$25, avg=14.063)	C1/9	**Turning** Acceleration pedal (avg=4.603) Brake pressure (avg=1.545) Parabolic curve in steering angle($\sim$150$\rightarrow$$\sim$450 & $\sim$450$\rightarrow$$\sim$150, avg=318.256) Vehicle speed (avg=10.235)
C2/10	**Turn in a roundabout** Acceleration pedal (avg=15.214) Braking pressure (avg =0.0011) Steering angle (avg=107.8886) Vehicle speed(avg=28.120)	C2/38	**Approaching a crossroad or an obstacle (another car, pedestrian, traffic jam)** Acceleration pedal is low (avg=2.268) Slow Braking pressure (avg=1.465) Steering angle (avg=23.687) Vehicle speed progressively decreasing ($\sim$35$\rightarrow$$\sim$15, avg=23.905)	C2/17	**After stopping at a red traffic light or at traffic jam / after surpassing an obstacle (e.g, pedestrian crossing the street)** Acceleration pedal (avg=18.117) Very low braking pressure (avg=0.0001) Steering angle (avg=26.130) Vehicle speed increasing (avg=11.737)
C3/30	**Approaching a crossroad** Low acceleration pedal (avg=2.968) Slow braking occurring (avg=0.569) Steering angle (avg=7.966) Vehicle speed progressively decreasing ($\sim$40$\rightarrow$$\sim$20, avg=29.413)	C3/32	**Car driving straight in a clear road (no stopping before)** Acceleration pedal (avg=15.262) No brake pressure (avg=0) Steering angle (avg=11.334) Vehicle speed increasing ($\sim$20$\rightarrow$$\sim$37.5, avg=28.642)	C3/17	**Car driving straight after crossing (no stopping before)** Acceleration pedal (avg=14.904) Brake pressure (avg=0.012) Steering angle (avg=9.223) Vehicle speed is progressively increasing ($\sim$20$\rightarrow$$\sim$35,avg=29.503)
C4/27	**After turning** Pedal acceleration increasing(avg=23.551) No braking occurring (avg=0) Steering angle decreased($\sim$150$\rightarrow$$\sim$0, avg= 11.8949) Vehicle speed increasing ($\sim$10$\rightarrow$$\sim$40, avg=29.351)	C4/8	**Turning** Acceleration pedal (avg=3.174) Slow brake pressure (avg=3.558) Parabolic curve of steering angle ($\sim$150$\rightarrow$$\sim$400 & $\sim$400$\rightarrow$$\sim$200, avg=303.403) Vehicle speed (avg=5.674)	C4/43	**Approaching a crossroad or an obstacle (another car, pedestrian)** Acceleration pedal is low (avg=0.447) Slow Brake pressure (avg=1.675) Steering angle (avg=17.999) Vehicle speed progressively decreasing ($\sim$35$\rightarrow$$\sim$12, avg=20.391)
C5/12	**Turning** Pedal acceleration increasing progressively (avg=18.554) Brake pressure (avg=0.162) Parabolic curve in Steering angle ($\sim$100$\rightarrow$$\sim$350 & $\sim$350$\rightarrow$$\sim$100, avg=260.541) Vehicle speed (avg=12.757)	C5/22	**Braking before stopping at a traffic light or a crossroad** Very low acceleration pedal (avg=0.160) Braking occurring (avg=8.994) Steering angle (avg=20.861) Vehicle speed decreased ($\sim$14$\rightarrow$$\sim$0, avg=2.215)	C5/18	**Braking before stopping at a traffic light or a crossroad** Very low acceleration pedal (avg=0.023) Braking occurring (avg=12.328) Steering angle (avg=13.051) Vehicle speed decreased ($\sim$20$\rightarrow$$\sim$0, avg=1.672)

images that we mapped to each driving scenario. In order to evaluate the similarity between extracted episodes we measured the Pearson correlation and their probability distribution. We conducted an empirical evaluation using vehicle bus signals of mainly four vehicle sensors measuring the acceleration pedal, braking pressure, steering angle and the vehicle speed. For future work, we intend to improve this approach by considering object detection using artificial neural networks to provide additional inputs to the clustering and be able to come up with more detailed interpretations. We also plan to try other clustering algorithms and to compare the obtained outcome.

References

[1] Ahmetcan Erdogan, Burak Ugranli, Erkan Adali, Ali Sentas, Eren Mungan, Emre Kaplan, and Andrea Leitner. Real-world maneuver extraction for autonomous vehicle validation: A comparative study. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 267–272. IEEE, 2019.

[2] Jakob Geyer, Yohannes Kassahun, Mentar Mahmudi, Xavier Ricou, Rupesh Durgesh, Andrew S. Chung, Lorenz Hauswald, Viet Hoang Pham, Maximilian Mühlegg, Sebastian Dorn, Tiffany Fernandez, Martin Jänicke, Sudesh Mirashi, Chiragkumar Savani, Martin Sturm, Oleksandr Vorobiov, Martin Oelker, Sebastian Garreis, and Peter Schuberth. A2D2: Audi Autonomous Driving Dataset, 2020.

[3] Jiwon Kim and Hani S. Mahmassani. Trajectory clustering for discovering spatial traffic flow patterns in road networks. In *TRB 94th Annual Meeting Compendium of Papers*, Washington DC, United States, 2015.

[4] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 5.1, pages 281–297. Oakland, CA, USA, 1967.

[5] Tin T Nguyen, Panchamy Krishnakumari, Simeon C Calvert, Hai L Vu, and Hans Van Lint. Feature extraction and clustering analysis of highway congestion. *Transportation Research Part C: Emerging Technologies*, 100:238–258, 2019.

[6] Dongkuan Xu and Yingjie Tian. A comprehensive survey of clustering algorithms. *Annals of Data Science*, 2(2):165–193, 2015.