Identify Hard stop and momentary stop using vehicle trajectory dataset

Hetav Shah, Parshwa Shah, Ved Savalia, Aarsha Shah and Dev Gorakhiya

*Department of Computer Science*   
*Ahmedabad University*Ahmedabad, India  
Email: {hetav.s1, parshwa.s1, ved.s, aarsha.s, dev.g} @ahduni.edu.in

*Abstract*— The goal of this study is to employ velocity analysis to identify hard stops and temporary stops in vehicle trajectory data. Preprocessing of the dataset involved performing interpolation for missing values and filtering track IDs with more than 100 frames. Average and instantaneous velocities were computed to study the motion of the vehicle. Velocity variations were analyzed to identify stop patterns by applying DBSCAN clustering. The differentiation between kinds of stops was eased by velocity-time graphs. The proposed method supports intelligent transportation systems by providing knowledge about traffic behavior.

Keywords- Vehicle Trajectory Analysis, Velocity Computation, DBSCAN Clustering, Stop Detection, Intelligent Transportation Systems (ITS), Traffic Behavior, Hard Stops, Momentary Stops, Data Preprocessing, Machine Learning in Traffic Analysis, Urban Mobility, Autonomous Vehicle Navigation.

# Introduction

Development of intelligent transportation systems (ITS), improvement of road safety, and maximizing of urban mobility all depend on traffic analysis. Maximizing traffic flow, lowering accidents, and allowing autonomous vehicles’ decision-making depends on awareness of vehicle’s stopping behavior. This work introduces a vehicle trajectory dataset including velocity (v), temporal (frameNo), and spatial (x, y) information. In this, first track IDs with more than 100 frames were filtered from the dataset and then missing frames were interpolated. Later instantaneous and average velocities were computed in order to define each vehicle’s motion. Different kinds of stops were categorized based on velocity transitions using DBSCAN unsupervised clustering algorithm, thereby facilitating traffic behavior research. The system given here supports urban traffic management and self-driving vehicle navigation using machine learning approaches. The work presented here includes experimental results and possible real-world applications after discussing dataset processing, speed estimates, and stop categorization technique.

# DATA PREPROCESSING AND FEATURE EXTRACTION

Raw dataset contained vehicle trajectory data with columns such as trackID (track identifier, unique vehicle), frameNo (time index), x and y (x and y coordinates of position), and v (velocity). To maintain consistency and accuracy in data, the following preprocessing was performed:

1. Finding Center X and Y using given data:

We were given top, left, width and height of each frame. We then found X and Y for each frame.

Center X = Left + (Width / 2)

[1]

Center Y = Top + (Height / 2)

[2]

1. Handling Missing Data with Interpolation:

Some of the trackIDs had missing frames, which were interpolated linearly to ensure trajectory continuity. This is important for ensuring smooth calculation of velocity and avoiding data gaps from influencing the analysis.

1. Velocity Computation

Instantaneous velocity was computed as follows:

[3]

Average velocity for a trackID was computed as:

[4]

1. Splitting Data by trackID with sufficient frames:

Those vehicles having less than 100 frames were eliminated in order to concentrate on significant motion patterns. The division allows per-vehicle visualization and analysis.

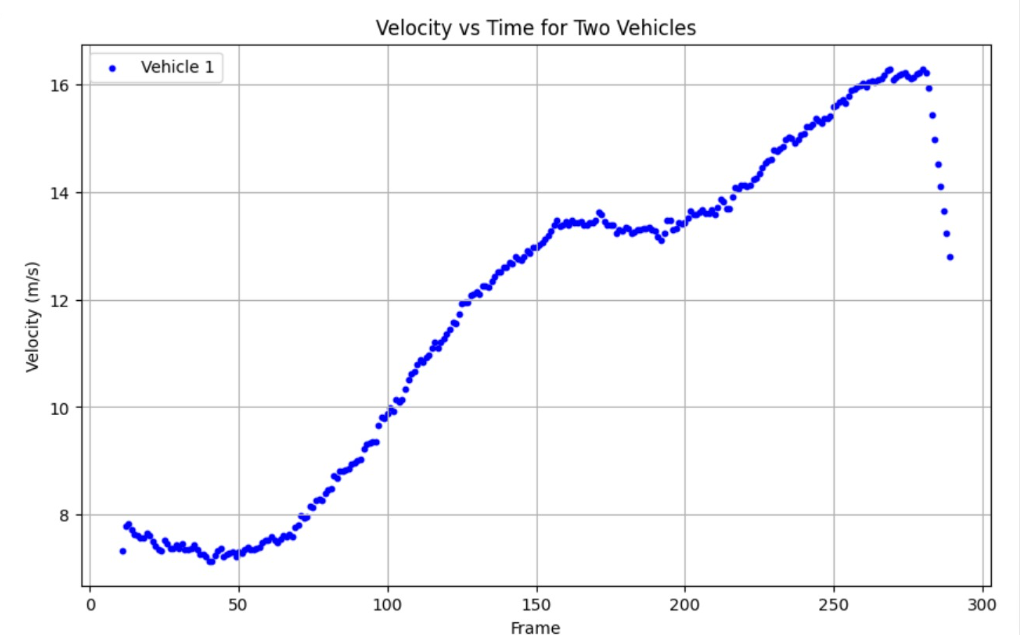
With these preprocessing methods, the dataset was formatted for subsequent analysis, such as stop detection and clustering.

# Detecting Vehicle Stops Using Velocity Scatter Plots and DBSCAN

We present a method whereby numerical values from the dataset can be graphically shown to efficiently study vehicle movement patterns and detect various halting behaviors. We especially want to create a scatter map depending on instantaneous velocity over time for every vehicle that has been found by their unique Track ID. Visualizing the velocity variances in a scatter graph helps us to uncover significant patterns using the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) approach and apply them to locate major stops.

We first found out the instantaneous speed of every vehicle in the dataset. This calls for repeatedly determining the X and Y coordinates of the vehicle. These coordinates were generated from the given mathematical formulas within the dataset. After the coordinates were acquired, we applied the conventional velocity formula which incorporates change in position over a certain time interval to get the instantaneous velocity.

Once the instantaneous velocity values have been determined, we represented the data using a scatter plot. The plot's x-axis represented frame numbers; the y-axis of the graph included instantaneous velocity values. Similarly, every vehicle can have a scatter plot which could help in finding variations in speed with respect to time for each vehicle. This graphical display is quite important since it lets us visualize where the car abruptly slows down or stops.



Next, the approach was focused on DBSCAN cluster-based analysis of the scatter graphs. Quite effective in differentiating several types of stopping behaviors, DBSCAN is a density-based strong clustering method. The DBSCAN technique would identify clusters inside each of these instantaneous velocity v/s time scatter plot and will make decision based upon that.

We shall carefully choose DBSCAN's parameters: minimal points (MinPts), which determines the least number of points required to form a dense cluster; epsilon (ε), which provides the maximum distance between points for them to be treated as part of the same cluster *(DBSCAN Clustering Algorithm Based on Density, 2020).* These parameters should be adjusted in such a way that the differentiation between two kinds of stops were accurate:

1. Hard stops are those in which a car abruptly stops and remains motionless for a considerable length of time.
2. Momentary stops that are, brief pauses in movement such as stopping at a traffic signal or momentarily slowing down in reaction to congestion.

DBSCAN would search the scatter graphs for clusters matching certain stopping patterns. Hard stops will be those spots forming dense clusters near the zero-velocity zone for an extended period; transitory stops will be those clusters displaying brief pauses with velocity fluctuations.

Once the clustering process finishes, depending on the discovered groups our model will classify the stopping behavior of every car. Traffic analysis, accident avoidance, and anomaly detection in transportation systems are among the further applications for this categorization. We automate the identification of several stopping behaviors using DBSCAN, therefore providing a logical and efficient approach to investigate vehicle motion patterns in useful environments.

##### Acknowledgments

We express our sincere gratitude to Prof. Mehul Raval and Mr. Yagnik Bhavsar from Ahmedabad University for their invaluable guidance and support throughout this project. We appreciate their efforts in providing us with the dataset, which served as the foundation for our analysis. Their insights and expertise have been instrumental in shaping our approach and understanding of the problem.

##### References

1. *DBSCAN Clustering Algorithm based on density*. (2020, September 1). IEEE Conference Publication | IEEE Xplore. <https://ieeexplore.ieee.org/abstract/document/9356727?casa_token=bE-GpoKD4UYAAAAA:MWfySFyC8222IMJ2ghlPFd7P2z6MMkdreLU_k6Zcylzf8msJu8nXTHoyF25vetmheHvKEbxbKQWG>
2. *Clustering of vehicle trajectories*. (2010, September 1). IEEE Journals & Magazine | IEEE Xplore. <https://ieeexplore.ieee.org/abstract/document/5462900/authors#authors>
3. Stats Wire. (2021, July 16). *DBSCAN Clustering | Python | Clustering* [Video]. YouTube. <https://www.youtube.com/watch?v=2eDFjw456AM>
4. StatQuest with Josh Starmer. (2022, January 10). *Clustering with DBSCAN, Clearly Explained!!!* [Video]. YouTube. <https://www.youtube.com/watch?v=RDZUdRSDOok>
5. Birant, D., & Kut, A. (2006). ST-DBSCAN: An algorithm for clustering spatial–temporal data. *Data & Knowledge Engineering*, *60*(1), 208–221. <https://doi.org/10.1016/j.datak.2006.01.013>