# Identifying anomalous places and routes by GPS feature: a system for child monitoring

Giacomo Abbattista [1], Donato Impedovo [1], Giuseppe Pirlo [1], Lucia Sarcinella [1], and Nicola Stigliano [1]

[1] *University of Bari, Dep. Of Computer Science, Via Orabona 4, Bari, Italy*

**Abstract**

The phenomenon of bullying and cyberbullying is a constant thorn for today's kids.
Some of these phenomena take place on the way from home to school (and back), it can therefore materialize in anomalies through deviations from the standard route, or through pauses / interruptions. These anomalies can be detected through the use of a GPS sensor already available on all smartphones. In this work it is presented a system that through the acquisition of the GPS parameters of the mobile phone is able to recognize abnormal path compared to standard ones and to report the event to parents in order to take appropriate precautions. In addition, parents can visualize paths and events by using a simple web platform.
The system is a preliminary version and has been tested on a sample of 9 users, demonstrating excellent accuracy of the results and a wide acceptance by the selected users.

**Keywords**

Bullying, Cyberbullyng, GPS, DBScan, FoliumMap, Android

## 1. Introduction

With more than four billion Internet users across the globe [1], the online world is now part of everyday life, and it plays a vital role in society. This rapid growth in technology is not coming only with advantages but has surfaced many problems out of which cyberbullying is one of the primary concerns. The internet has turned to be a double-edged sword which has brought unmatched ease in our daily life. On the other hand, the internet has also created grounds for numerous unwanted behaviors, like cyberbullying, a bullying type articulated via electronic means [2].

The bullying actions include physical assault, verbal assault and by spreading fake news, harsh words/comments, rumors, gossips, threats, exclusion from social circle etc. The technological advancement has transformed traditional bullying into cyberbullying [3] which is "the use of information and communication technologies to support deliberate, repeated, and hostile behavior by an individual or group that is intended to harm or defame others [4]", in simple words cyberbullying is "an electronic form of peer harassment [5]". Cyberbullying is considered as more dangerous in comparison to traditional bullying because cyberbullying has the potential to protect the bully due to anonymity. This is the biggest difference as technology, and the internet gives extra mile protection to the perpetrator.

A cyberbully can bully from any part of the world, and all s/he needs is a relevant technology or medium that is readily available in almost all parts of the world. Cyberbullying can be quickly done 24 hours a day and 365 days a year, unlike physical bullying. Cyberbullying can occur at any time of life irrespective of age group [6] and it increases as a person grow [7], [8].

The work reported in this paper is part of an Italian project aimed at creating an app able to record a wide series of events can be referred to a bulling or cyberbulling action, and therefore exploiting the same technologies that created the problem [9]. In particular, in this work we will discuss a functionality

(ITASEC) Italian Conference on Cybersecurity, April 7-9, 2021, Italy
EMAIL: giacomo.abbattista@uniba.it (A. 1); donato.impedovo@uniba.it (A. 2); giuseppe.pirlo@uniba.it (A. 3); lucia.sarcinella@uniba.it (A. 4); n.stigliano@studenti.uniba.it (A. 5)
ORCID: 0000-0003-0850-728X (A. 1); 0000-0002-9285-2555 (A. 2); 0000-0002-7305-2210 (A. 3); 0000-0002-8550-8588 (A. 4)

implemented through a system that acquires by consensus the GPS parameters (latitude and longitude) of the smartphone on which the application is installed and by analyzing these parameters, it is able to recognize the places most frequented by the user and the routes usually used to move, but also and above all, the unusual places and routes taken by the user. The parent then, through a special web platform can view all this information graphically and visually.

The following paper is organized as follows: in section two the methods and technologies used will be presented; in section 3 experiments will be presented; in section 4 results are presented. Section 5 concludes the work.

## 2. Methods

In this section we will examine the different techniques and technologies used within the work. During the project development many solution have been investigated, however only the main and final ones will be described below, thus excluding those discarded.

### 2.1. Second level heading

Fundamental and essential parameters of the work are the GPS parameters related to the users' smartphones. These parameters are part of those acquired by the app (named ShieldApp) we are developing. To date, the application is only available for devices with an Android Operating System having an SDK no older than the 24. Android is the most widespread operating system in the world: it is certified that 62.94% of mobile devices, including car radios, smartwatches, televisions and IoT products, use Android as an operating system, or alternatively an operating system based on Android, each of which has a dedicated graphic interface to make the user experience highly performing.

ShieldApp as soon as the installation is completed shows the user a security policy that asks for consent to acquire his personal data relating to GPS movements and to use them for scientific research purposes, ensuring not outside, all according to the protection regulations of European data (GDPR). The security policies shown to the user, in particular, state that in accordance with the GDPR, the acquired data are used for research purposes and solely and exclusively for the detection of bullying and cyberbullying. Furthermore, all the results obtained from the processing will be visible only to parents.

As soon as the security policy is accepted, whenever the user has turned on the GPS, the application acquires this value and stores it on a mySQL database. In particular, in addition to the GPS parameters (latitude and longitude), for each recorded value, the corresponding ID of the device, the type of data (in this case of the "Sensor" type), the timestamp and the acquisition time (in the format YYYY-MM-DD HH: MM: SS) are also stored. In fig. 1 an example of the acquired data.

| device_id | log_obj | log_content | timestamp_val | date_val |
|---|---|---|---|---|
| c72mYvW1iWw | SENSOR | 25.056027306708685 | 1586945846 | 2020-04-15 12:17:26 |
| c72mYvW1iWw | SENSOR | 27.462614413593027 | 1586945846 | 2020-04-15 12:17:26 |
| c72mYvW1iWw | SENSOR | 25.7573753090741 | 1586945846 | 2020-04-15 12:17:26 |
| c72mYvW1iWw | SENSOR | 26.359483828317135 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 32.30701794042782 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 31.172784487045448 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 26.02055250820073 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 28.363498322833433 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 39.08376920954528 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 41.02900525799934 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 42.61592072475673 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 40.51770988658077 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 34.315383317494245 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 26.54673015337176 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 37.725191364346486 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 41.07833153000573 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 39.14508925389381 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 36.73523233793241 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 27.161806510432186 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 25.944064493095045 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 37.393963037898075 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 43.92129829975359 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 44.574500652261236 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 42.548868962641066 | 1586948057 | 2020-04-15 12:54:17 |
| c72mYvW1iWw | SENSOR | 35.71815192522039 | 1586948058 | 2020-04-15 12:54:18 |
| c72mYvW1iWw | SENSOR | 24.791655040919892 | 1586948058 | 2020-04-15 12:54:18 |

**Figure 1**: Example of stored data

## 2.2. Data Clustering

Data are transferred from the mobile device to a server periodically. A series of processing steps are performed on the server, in this case the user routing behavior is inspected by adopting an unsupervised clustering algorithm: elements of a cluster will be the usual places and paths, while all the outliers will be the anomaly ones. In particular, the choice of the unsupervised is mandatory since during the test phase the user was not asked to report anything or to explicitly interact with the app, therefore labels are not available. This also occurs in a real scenario in which the user normally will not voluntarily tag his/her movements [10], [11], [12], [13].

Cluster analysis groups data objects based only on information found in the data that describes the objects and their relationships. The goal is that objects within one group are similar (or related) to each other and different (or unrelated) from objects in other groups. The greater the similarity (or homogeneity) within a group and the greater the difference between the groups, the better or more distinct the grouping. In this preliminary experiment, DBSCAN has been considered [14] based on the consideration that it has been already adopted in several works dealing with geospatial data for position prediction [15] [16]. The DBSCAN algorithm uses two parameters:

- minPts: the minimum number of points (a threshold) grouped together for a region to be considered dense.
- eps (ε): a distance measure that will be used to locate points in the vicinity of any point.

These parameters can be understood if exploring Density Reachability and Density Connectivity. Reachability in terms of density establishes a point reachable by another if it is within a particular distance (eps) from it. Connectivity, on the other hand, involves a transitivity-based chaining approach to determine whether points are in a particular cluster.

DBSCAN does not require to specify the number of clusters a priori, unlike many other widely used algorithms such as k-means. This is of vital importance since each cluster is equivalent to a place heavily frequented by the user, such as his home, school, workplace, etc., but there is no fixed number of these places that applies to all users, nor is there a fixed number of these places for the same user over time. Also, DBSCAN can find clusters of arbitrary shape. It can even find a cluster surrounded (but not connected) by a different cluster. Due to the MinPts parameter, the so-called single-link effect (several clusters connected by a thin line of dots) is reduced. In addition, it requires only two parameters and is mostly insensitive to the ordering of points in the database.

Unfortunately, there are not only advantages, but DBSCAN also has disadvantages. In this case the main disadvantage is that the quality of DBSCAN depends on the distance measure used in the regionQuery function (P, ε). The most common distance metric used is the Euclidean distance. Especially for high-dimensional data, this metric can be rendered almost useless due to the so-called "Curse of Dimensionality", making it difficult to find an appropriate value for ε. This effect, however, is also present in any other Euclidean distance algorithm.

As for the implementation of dbscan, it is clear that the greatest difficulty lies in deciding the values that eps and minPts will have to assume. An additional difficulty lies in the fact that depending on the device, configurations and possible data distributions will change from time to time. Since it is not possible to predict what kind of data we will have available, it was decided to test a range of values each time. This range for minPts goes from 5 to 400, while for eps, which will be tested on the basis of the best minpt, a value between 0.01 and 0.09 will be chosen if we have less than 51 for minPts, otherwise a value between 0 , 1 and 2. These choices are supported by the silhouette coefficient [17]: an important metric that is calculated using the mean intra-cluster distance (a) and the nearest mean distance (b) for each sample. The silhouette coefficient for a sample is (b - a) / max (a, b), where b is nothing more than the distance between a sample and the nearest cluster of which the sample is not part. The best value is 1 and the worst value is -1. Values close to 0 indicate overlapping clusters. Negative values generally indicate that a sample was assigned to the wrong cluster, as a different cluster is more similar. In the end, therefore, dbscan will be set with the best values based on the silhouette coefficient, using the Euclidean metric, most used metric, such as in [18] where is used as fitness function to control

the process of parameters determination by optimization, or in [19] used to classify the type of text contained in the Al-Quran, or as confirmed by [20] and [21]. Obviously, all these operations were performed on normalized data.

Based on some tests carried out, it was decided to work week by week on the data. This choice derives from the fact that the tests carried out have shown that less than 5 days produce inaccurate results, probably due to the scarcity of available data. The best results are obtained when we have more and more data available, but unfortunately the time required for execution would increase significantly. For this reason, the best result-time compromise was reached with 7 days (Silhouette Coefficient> 0.90).

## 2.3. Data Visualization

Once the classification of the available GPS data is done, it is necessary to visually visualize these results. Powerful visualization tools and libraries are available nowadays. In this work the Folium library has been adopted. It is a powerful data visualization library in Python created primarily to help people visualize geospatial data. Maps are interactive so that zoom in and out are available. Folium will be used to create an interactive map that shows Cluster and Outlier in the most understandable way for the user (in this case the parent). More precisely, four different maps will be created:

1. Interactive map containing all the clusters of the week: in this map it will be possible to view all the clusters in the form of a heatmap, with the addition of a marker on the positions where there are more concentrations of data. This marker will be clickable and will give the user the possibility to see the city, the postcode, the street and possibly also the name of the place where it is located. We also tried to give continuity to the user's movement: the various points of the map were connected based on the time that elapses between them. It was decided to combine the points, based on the recording of which they took place with a time frame of less than 10 minutes (obviously other times were also tested before arriving at this choice). This choice is due to the fact that it is possible for a person to stop at a traffic light, encounter a traffic accident or simply make a stop, without changing your final destination.

2. Interactive map containing all the clusters of the week, sorted by time: Within the map the points are always displayed in the form of a heat map. It also contains a slider where the user (the parent) can move according to the time and view the precise instant in time with the precise point where the position was recorded. The parent will also be given the opportunity to decide whether to view a specific day on which to view the data.

3. Interactive map containing all outliers for the week: identical to the first map, but with outliers instead of clusters.

4 Interactive maps containing all outliers for the week, sorted by time: identical to the second map, but with outliers instead of clusters.

As for the pop-up in which the data relating to the point produced by the coordinates are displayed, a reverse geocoding technique has been implemented. For this technique, the 10 most recurring points were taken (counting the occurrences present in the Data frame) and finally they were fed to Nominatim, a function present in the geopy.geocoders library that returns all the details of the position.

This process was repeated for all four maps. Another action carried out was to create a dictionary, sorted by key (date and time) sent to the "heatmapwithtime" folium to create maps 2 and 4 (Heatmap with time by cluster and outlier). As for this last feature, the user will be given the opportunity to view a certain day. This was possible simply by creating an ad hoc Data Frame, selecting only the values that contain the date entered by the user.

It is important to underline that the user is also given the possibility to disable the heat maps, in case they prevent them from being correctly displayed.

An example of a user display is shown in fig. 2 and 3, where in figure 2, the 2 main clusters identified by a user and the relative usual paths performed by him can be observed, while in figure 3, a path carried out by the user can be observed, identified as anomalous, in particular it is a deviation commute from home to work.

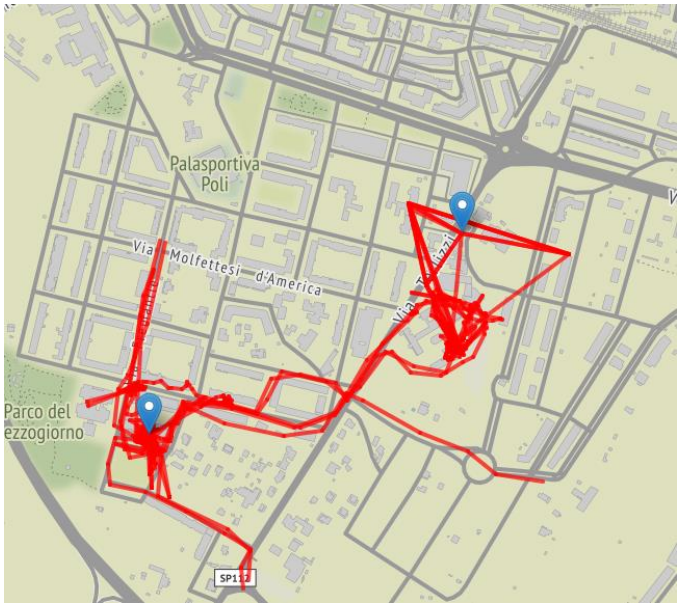


**Figure 2:** Graphic representation of clusters

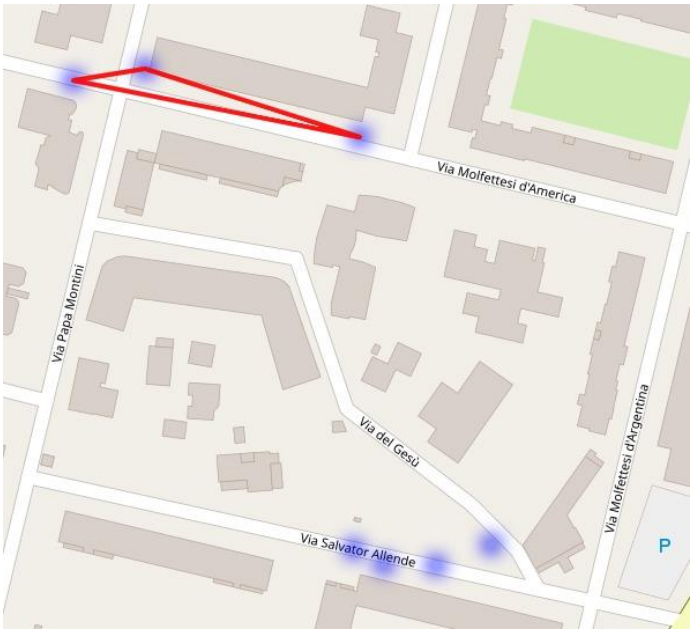


**Figure 3:** Graphic representation of an anomalous path identified

## 3. Experiments

Two methodologies were considered to ascertain the accuracy of the system: the first relies on clustering evaluation metrics, while the second is based on questionnaires administered to users. Two methodologies were used because with simple clustering assessment metrics, it cannot be said for sure whether the locations and paths identified as frequent or unusual by a user have been correctly classified. An answer that only the user can provide.

Regarding clustering evaluation metrics, three metrics were chosen [17]:

- Davies-Bouldin score: The score is defined as the average similarity measure of each cluster with its most similar cluster, where similarity is the ratio of within-cluster distances to between-cluster distances. Thus, clusters which are farther apart and less dispersed will

result in a better score. The minimum score is zero, with lower values indicating better clustering.

- Calinski and Harabasz score: It is also known as the Variance Ratio Criterion. The score is defined as ratio between the within-cluster dispersion and the between-cluster dispersion.
- Silhouette Coefficient: it is calculated using the mean intra-cluster distance (a) and the mean nearest-cluster distance (b) for each sample. The Silhouette Coefficient for a sample is (b - a) / max (a, b). To clarify, b is the distance between a sample and the nearest cluster that the sample is not a part of. Note that Silhouette Coefficient is only defined if number of labels is $2 <= n\_labels <= n\_samples$. The best value is 1 and the worst value is -1. Values near 0 indicate overlapping clusters. Negative values generally indicate that a sample has been assigned to the wrong cluster, as a different cluster is more similar.

Nine users were involved in the testing phase. The monitoring period ranges between 10 and 14 days. At the end of this period, a 9-question questionnaire was administered to the users. A sub-set of questions related to the veracity of the displayed data of the paths and places identified as normal and anomaly. A 5-value Likert Scale (from Strongly Disagree to Strongly Agree) was adopted, each answer was then associated with a value (0, 25, 50, 75, 100). The average was adopted to estimate the degree of accuracy [22].

## 4. Results

The silhouette score was calculated for each user obtaining an average value of 91%, with a minimum value of 73% and a maximum value of 98%. (table 1).

**Table 1**

Silhouette results

| User | Silhouette % |
|---|---|
| 1 | 98% |
| 2 | 88% |
| 3 | 93% |
| 4 | 88% |
| 5 | 88% |
| 6 | 95% |
| 7 | 96% |
| 8 | 73% |
| 9 | 98% |
| **Tot.** | **91%** |

The accuracy related to the evaluation of anomalies reported by the system and evaluated by users reached a value of 87.5%. This result is similar to the silhouette score and therefore confirms the previous classification data.

## 5. Conclusion and future development

In order to monitor the movements of children from infancy to adolescence, an android app and a web platform have been developed. The Andorid app was used with the aim of acquiring the gps parameters related to the children's smartphones, while the web platform was used to visually show the user the data acquired by the app and the results of the analyzes relating to the identification of any anomaly places or routes. The difference between this solution and those already on the market lies precisely in the fact that current solutions usually allow simple real-time monitoring or movement history, while our solution automatically identifies any anomalous routes, supporting parental control. For this purpose, DBScan was used as a clustering algorithm, and FoliumMap and Flask for the creation

of the web platform. The overall system was tested on 9 users, demonstrating an accuracy of 87.5%, confirming its possible use in real contexts.

Of course, in the next studies it is of primary importance to considerably extend the test sample to validate the results currently obtained and, to extend the web platform implemented by integrating with works that use other reference data other than GPS parameters, such as other sensors such as the accelerometer. In fact, alone, the results obtained from the GPS parameters are not always sufficient to affirm a phenomenon of bullying or cyberbullying, for this reason, please note that the following work is only part of a larger project in progress, where with the use of multiple sensors, apps and other technologies, it will be possible to identify these phenomena.

## 6. Acknowledgements

This work is supported by the Italian Ministry of Education, University and Research within the PRIN2017 - BullyBuster project - A framework for bullying and cyberbullying action detection by computer vision and artificial intelligence methods and algorithms. CUP: H94I19000230006.

## 7. References

[1] M. Group, «Internet Top 20 Countries-Internet Users 2020.,» 30 June 2019. [Online]. Available: https://www.internetworldstats.com/top20.htm.

[2] Q.Li., «Cyberbullying in schools: A research of gender differences,» *School Psychol*, vol. 27, n. 2, pp. 157-170, 2006.

[3] M. Dadvar e F. De Jong, «Cyberbullying detection: A step toward a,» *21st Int. Conf. Companion World Wide Web ( WWB )*, pp. 121-125, 2012.

[4] L. Robinson, « Bullying and Cyberbullying,» 5 March 2020. [Online]. Available: https://www.helpguide.org/articles/abuse/bullying-and-cyberbullying.htm.

[5] P. S. Storm e R. D. Storm, «Cyberbullying by adolescents:A preliminary assestment,» *Educ. Forum*, vol. 70, n. 1, pp. 21-36, 2006.

[6] L. Betts, T. Baguley e S. Gardner, «Examining adults' participant roles in cyberbullying,» *J. Social Pers. Relationships*, vol. 36, n. 11-12, pp. 3362-3370, 2019.

[7] R. Ortega, P. Elipe, J. Mora-Merchán, J. Calmaestra e E. Vega, «The emotional impact on victims of traditional bullying and cyberbullying:A study of Spanish adolescents,» *Zeitschrift Psychologie/J.Psychol.*, vol. 217, n. 4, pp. 197-204, 2009.

[8] F. Shaikh, . M. Rehman e A. Amin, «Cyberbullying: A Systematic Literature Review to Identify the Factors Impelling University Students Towards Cyberbullying,» 21 Aug. 2020. [Online].

[9] N. Covertini, N. Logrillo, F. Manca e T. Palmisano, «Recommendation System using Hybrid Fuzzy Association Rules for Human Smart Cities,» *2018 AEIT International Annual Conference, Bari, Italy*, 2018.

[10] D. Impedovo, F. Balducci, V. Dentamaro e G. Pirlo, «Vehicular Traffic Congestion Classification by Visual Features and Deep Learning Approaches: A Comparison,» *Sensors*, vol. 19, n. 5213, 2019.

[11] D. Impedovo, V. Dentamaro, G. Pirlo e L. Sarcinella, «TrafficWave: Generative Deep Learning Architecture for Vehicular Traffic Flow Prediction,» *Sensors*, vol. 9, n. 5504, 2019.

[12] N. Convertini, N. Dentamaro, D. Impedovo, G. Pirlo e L. Sarcinella, «A Controlled Benchmark of Video Violence Detection Techniques,» *MDPI information,* vol. 11, n. 321, 2020.

[13] V. Dentamaro, D. Impedovo e G. Pirlo, «Gait Analysis for Early Neurodegenerative Diseases Classification Through the Kinematic Theory of Rapid Human Movements,» *IEEE Access,* vol. vol. 8, pp. 193966-193980, 2020.

[14] K. S. do Prado, «How DBSCAN works and why should we use it?,» 2 Apr 2017. [Online]. Available: https://towardsdatascience.com/how-dbscan-works-and-why-should-i-use-it-443b4a191c80.

[15] M. S. Suchithra e M. L. Pai, «Data Mining based Geospatial Clustering for Suitable Recommendation system,» *2020 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India,* pp. pp. 132-139, 2020.

[16] M. Perumal e B. Velumani, «Design and development of a Spatial DBSCAN Clustering framework for location prediction- An optimization approach,» *2018 3rd International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India,* pp. pp. 942-947, 2018.

[17] Mohantysandip, «A Step by Step approach to Solve DBSCAN Algorithms by tuning its hyper parameters,» 12 May 2020. [Online]. Available: https://medium.com/@mohantysandip/a-step-by-step-approach-to-solve-dbscan-algorithms-by-tuning-its-hyper-parameters-93e693a91289.

[18] M. Li, X. Bi, L. Wang e X. Han, «A method of two-stage clustering learning based on improved DBSCAN and density peak algorithm,» *Computer Communications,* vol. 167, pp. pp. 75-84, 2021.

[19] M. A. Ahmed, H. Baharin e P. N. E. Nohuddin, «Analysis of K-means, DBSCAN and OPTICS Cluster algorithms on Al-Quran verses,» *International Journal of Advanced Computer Science and Applications,* vol. vol. 11, n. n. 8, pp. pp. 248-254, 2020.

[20] G. Huang, W. B. Qu e H. Y. Xu, «Traffic Accident Location Clustering Based on Improved DBSCAN Algorithm,» *Jiaotong Yunshu Xitong Gongcheng Yu Xinxi/Journal of Transportation Systems Engineering and Information Technology,* vol. vol. 20, n. n. 5, pp. pp. 169-176, 2020.

[21] U. Pandya, V. Mistry, A. Rathwa, H. Kachroo e A. Jivani, «2DBSCAN with Local Outlier Detection,» *International Conference on Recent Advancement in Computer, Communication and Computational Sciences, RACCCS 2019, Ajmer, India,* vol. vol. 1097, pp. pp. 255-263, 17 August 2019.

[22] E. R. S. H. Saputra, E. Utami e A. Nasiri, «Implementation of Location Based Service on Monitoring System of Visually Impaired Position with A-GPS Method,» *2018 3rd International Conference on Information Technology, Information System and Electrical Engineering (ICITISEE),* pp. pp. 271-275, 14 November 2018.