Pedestrian Group Detection Methods with K-Means and Object Path Analysis

Hyunjae Cho, M.S., The George Washington University
Nima Zahadat, Ph.D., The George Washington University

ABSTRACT

The edge-computing, a distributed computing system that processes complex computation and brings this computation to the source of the data, is a growing innovation technique in artificial intelligence and IoT (Internet of Things). Using the YOLO on NVIDIA Jetson-Nano, pedestrians in the Washington, DC area are detected, and the methods for pedestrian feature detection proposed based on YOLO and K-means clusters. YOLO model can detect more than 9000 objects with high accuracy score, but group detection remains as a challenging task. This paper will introduce four methods to detect groups, and these analyses will provide insights to city developers to find out business potentials and bring insights into the communities. In the research, the algorithm based on YOLOv4 will detect the existing real-time pedestrian and K-means clustering will be used to measure and evaluate group units.

Keywords: Object Detection, Motion Detection, K-means, DBSCAN, YOLO

INTRODUCTION

Object detection technique has significantly advanced and been used in our daily life. For example, iPhone uses a face detection system to unlock the screen and video surveillance for mask detection has been installed in the building entrances. Object tracking is still a challenging task in the machine learning field because target representation and location are required to extract the object movements from a series of pictures and to estimate the trajectories. Nevertheless, the proliferation of object detection techniques has accelerated with diverse tools and deep learning methods and has been studied in diverse fields.

The project was conducted with Basil Labs, a consumer intelligence startup that helps local businesses expand and optimize marketing strategies by capturing consumer behaviors with artificial intelligence. The project aims to understand 'where the families, couples, are friends go' to find the vibrancy of neighborhoods in cities. This information can be used by policymakers, retailers, urban planners to plan new business and optimize customer satisfaction. Even though the YOLO model has developed and detected numerous objects with high accuracy, detecting the number of groups on the streets has not studied yet and regarded as a convoluted study. For example, to detect family, friends, and couples, their genders, heights, and ages need to be detected while objects are tracked in the camera.

This paper will discuss the methods to find the groups within the crowds. First of all, the edge computing system is required to detect the real-time trackers, as the object detection architecture is based on edge computing systems to achieve distributed and efficient object detection through wireless communication (Ren, 2018). To set up an edge computing environment for high-speed real-time processing, Jetson Nano developer toolkits are used to collect the pedestrians' movements. The hand-sized Jetson Nano device has the advantage of providing a large processing capacity and allows the execution of functions with both CPU and GPU computing paradigm involved in parallel (Liang, 2020). Using collected videos, YOLO models were used to extract pedestrian information such as object id, tracker id, and bounding boxes of the object. Object tracker script, which tracks pedestrian trajectories, coded and uploaded on GitHub. Using the trajectories and bounding boxes information, K-means, DBSCAN (Density-Based Spatial Clustering of Applications with Noise), and K-means based on DTW (Density Time Warping) were used to cluster the group by trajectories. These methods are compared by silhouette scores that measure the density of the separated clusters, ranging from -1 to 1. In the end of the paper, the most efficient ways to group clusters and calculates distances to track real-time object based on the analyses.

LITERATURE REVIEW

YOLO (You Only Look Once), a real-time object detection system, developed by Joseph Redmon, and YOLO algorithm detects objects over 9000 categories with high accuracy. When the YOLO generates the bounding boxes of the object in frame with the ID number, the following tracking is detected by SORT (Simple Online and Realtime Tracking), which uses the Hungarian algorithm and Kalman filter to track objects (Bathija, 2019). Kalman filter is used to find the motion model of the object and estimate the position of the object in the next frame (Zhang, 2020), and the Hungarian algorithm correlates the neighboring frames with the improvement of enabling the multiple to one assignment (Wang, 2019). Deep Sort algorithm, which combined with a detection framework made of YOLOv4 and Retina Net, integrates object appearance information in an association matrix, and it supplements the loss in performance in SORT (Kapania, 2020). In the YOLOv4, the dataset used to train the corresponding models are

Market1501 and MARS. Both datasets were created for re-identification methods and MARS is an extended version of Market1501 by specializing in time series video data. As Deep Sort is mainly used for video data tracking, it results in better performance with scores with the MARS dataset.

Based on the research, four methods to detect groups can be proposed. The first method is to find the groups using K-means. K-means clustering is an unsupervised method and supports high-dimensional data. The number of K-means clusters can be determined using the Elbow method that the volatility within the cluster decreases sharply as additional clusters are increased and finds the optimum of the number of clusters. The process of the K-means cluster is to set the initial clustering and find the nearest nodes using Euclidean distances. When the nodes are assigned to the k cluster, the centroid of each cluster is calculated, and repeats these steps by assigning the closer nodes (Bholowalia, 2014). The iterations of these steps will be ended when the centroids are stabilized, and the final centroids will be used to classify the input data.

The second method is finding K-means of the centroids of moving objects using the extended K-means method. Omnia Ossama, Hoda Mokhtar, and Mohamed El-Sharkawi (2020) proposed an extended K-means algorithm for clustering moving object trajectory data. Unlike, K-means cluster, the extended K-means cluster initializes the centroids based on trajectory dissimilarity to remove the impact of dead centroids. The algorithm clustered the trajectories using Euclidean distance and a direction-based measure. Although the K-means algorithm seeks to minimize the average squared distance between points in the same cluster, the extended K-means algorithm also seeks to group trajectories featuring similar motion patterns (Ossama, 2020).

However, K-means has drawbacks of dependence on the first movement, which means it gathers the data into the best clusters in existing clusters (Zhang, 2017). To solve such a problem, the third method is suggested by grouping the density of trajectories. DBSCAN (Density-Based Spatial Clustering of Application with Noise) is one of the methods to cluster the trajectories with density. DBSCAN does not need to set the number of clusters, and it makes groups by putting an edge between all core points within the neighborhood of each other (Kanagala, 2016). In addition, DBSCAN has the advantages of finding arbitrary shaped clusters, like trajectories, and is robust to outliers and noises. However, the DBSCAN outcome heavily depends on epsilon and a minimum number of points, and the two points need to be adjusted accordingly (Kurumalla, 2016). In order to select the appropriate parameter for DBSCAN, the elbow method can be used to calculate the average distances between the points and neighbors.

Lastly, K-means with Density Time Warping can be used to cluster pedestrians based on the time-varying trajectory changes. One of the drawbacks of K-means with Euclidean distance in the time series data is Euclidean distance calculation ignores time dimension. However, DTW is suitable for tracking speed and the length of the objects, and K-means clustering with DTW can be used for group detection. M. Jang et al. introduced the DTW-based K-means cluster and proved its efficiency. The algorithm of the DTW-based K-means cluster partitions the reference patterns into a small number of clustering to the similarity measured by DTW, and the algorithm extracted the centroids of each cluster. In other words, K-means driven DTW not only collect time series of shape but also clusters centroids with respect to time-varying warping. Based on this method, the expected result would the centroids to have the average shape of the neighbors of the clusters.

RESEARCH METHODOLOGY

GPU, Python 3.6, and Python libraries, especially sklearn, were used to process the real-time multiple object detection and build models for clustering. NVIDIA Jetson Nano device, which has in-built support for CUDA tool kits and OpenCV, was installed in Washington, DC, and Bay Areas, San Francisco, and collected pedestrian movements on the street. The object tracker model developed with the YOLOv4 model and Deep Sort, and the collected information of an object and its location written in CVS format. The first K-means was conducted to cluster the first and the last trajectories points and compared the points to group the pedestrians. The next method is suggested with DBSCAN to supplement the drawback of K-means. Lastly, DTW is used to group the clusters, considering the time frames. To compare these methods, Silhouette coefficients were used to evaluate the clusters.

DATA

GPU and edge computing systems are required to collect the data. With the environment set-up, the video filed the random pedestrians in Washington, DC area. With the YOLO model, track id, frame number, type of object, centroids, and bounding boxes are collected and exported in the CSV format. The original YOLO model object tracker script had modified to identify the trajectories. 7 Videos collected and seven CSV files that contain pedestrians' information had uploaded with codes in GitHub, Zenodo, and GitHub.io.

DATA ANALAYSIS

Figure 1 is screenshots of the videos that show the pedestrians detected in the bounding boxes, and the trajectories are represented with white points. The point of trajectories ranges from 2 to 1258. If the trajectories points are less than 20 points, it is considered as outliers and noises and has been removed. For the data cleaning, MinMaxScale from the sklearn library was used to normalize the data and to find the optimal parameters for each clustering method. In the report, 5 minutes and 42 seconds long video that taken in the National Malls (crowd.csv) will be addressed for the analysis. In this video, 1224 people were detected, and the average length of trajectories is 132.88.

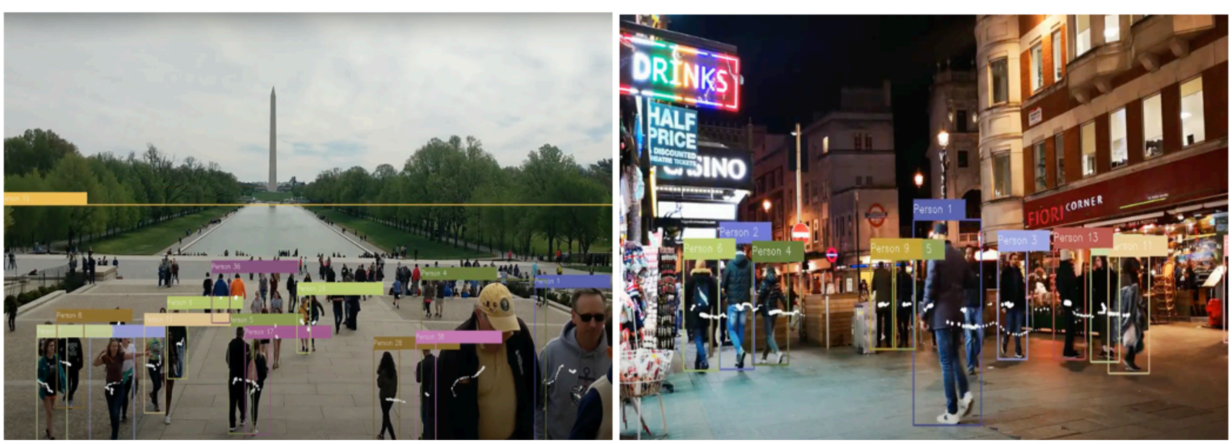
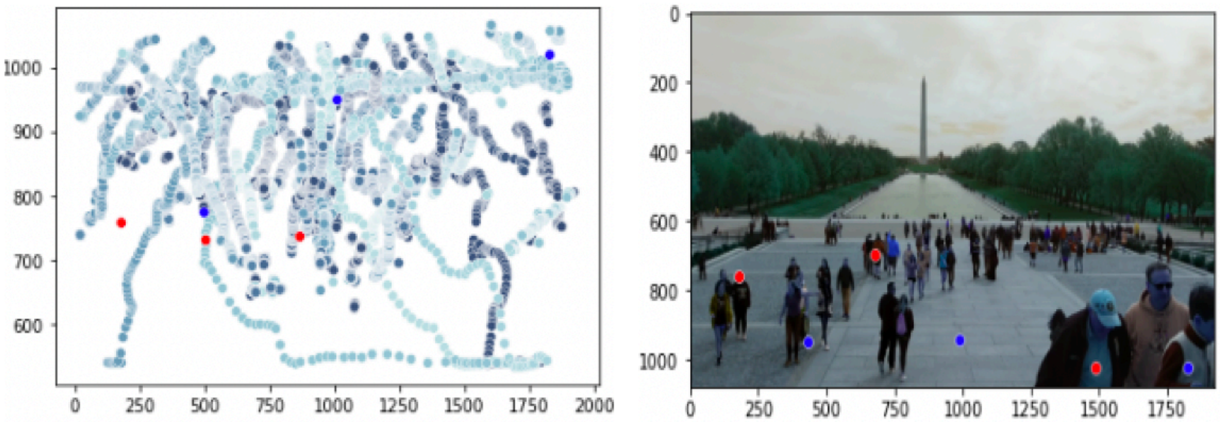


Figure1. The examples of object detection and the trajectories.

KEY FINDINGS

Method 1: K-Means with Elbow Methods and Extended K-Means

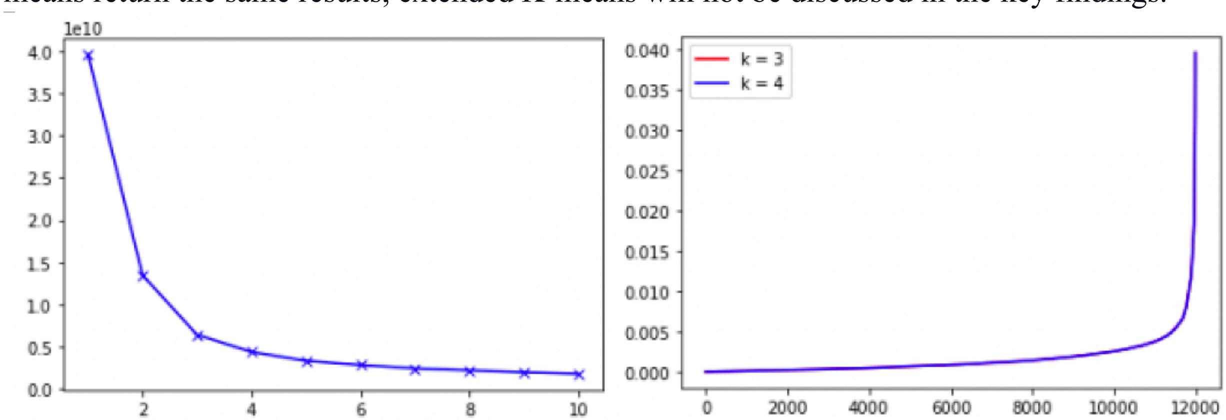


(a) (b)

Figure2. a) Trajectories with K-means clusters (k=3). b) Cluster points on the video moment.

Based on the research, the K-means algorithm was used to cluster pedestrian trajectories. The elbow method, which produces information in determining the best number of clusters by looking at the percentage of the comparison between the number of clusters (Rena, 2019) was implemented to choose the optimal number of clusters. Figure 3-b shows the significant changes after3 or 4 clusters. To evaluate the results of two clusters, a silhouette score was used, and the scores were 0.4897and 0.4571, respectively. Therefore, the cluster at k = 3 was determined for the analysis.

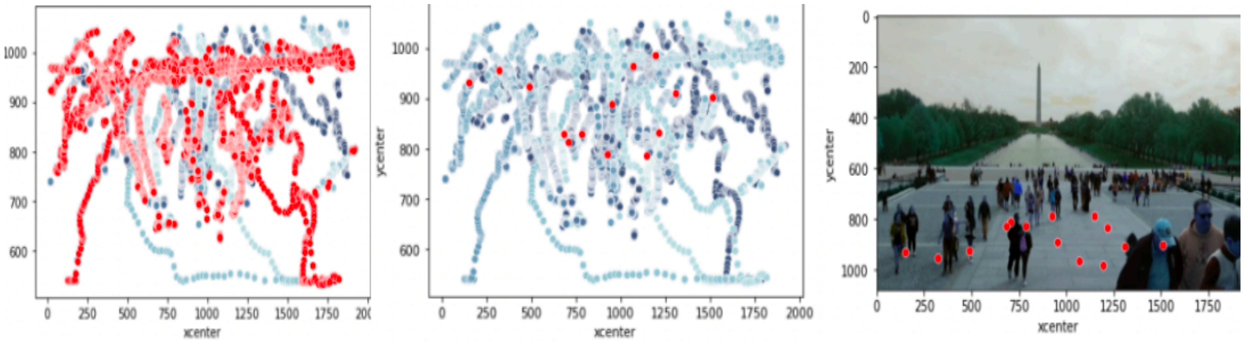
Figure 2 shows that K-means at cluster k =3 clustered the first and last trajectories points. If the starting points and the ending points are the same, pedestrians are likely the groups. The red points are the clusters of the first points and the blue points are the clusters of the last points of trajectories. The National Mall crowds in Figure 2-b shows the overlapping blue and red points in some groups. Since K-means has the drawbacks of first cluster dependency, extended K-means clustering used to cluster the trajectories of moving objects. K-means and extended K-means return the same results, extended K-means will not be discussed in the key findings.



(a) (b)

Figure3. a) K-means elbow plot. b) DBSCAN elbow plot.

Method 2: Density Based Spatial Clustering of Application with Noise



(a) *(b)* *(c)*

Figure4. a) Red trajectories are the results of DBSCAN, and blue trajectories are all pedestrians' trajectories. b) Red dots represent the centroids of DBSCAN trajectories c) The red dots plotted on the video moment.

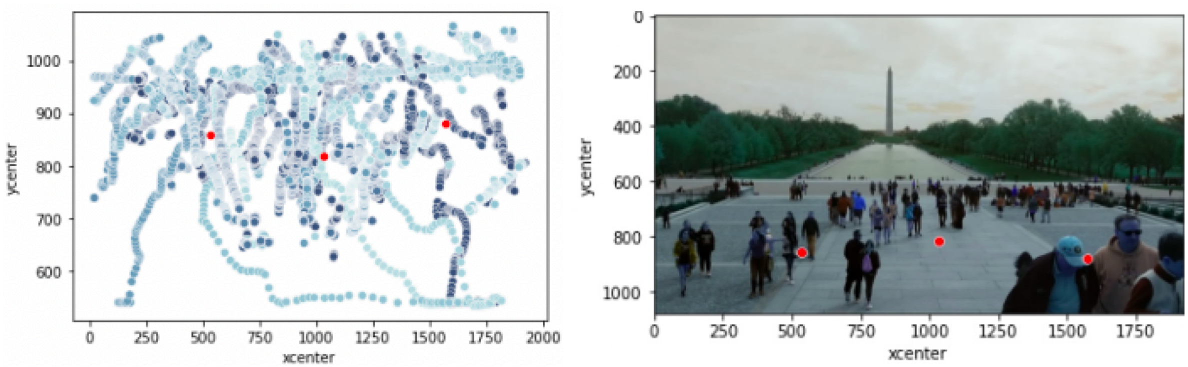
DBSCAN is another efficient method to cluster trajectories when the trajectories have a similar density. As the groups have the properties of having the same trajectories, DBSCAN is expected to cluster pedestrians by the trajectory density. Figure 4 plotted high-density trajectories with epsilon value 0.005 and the minimum point was 2, which resulted in 468 clusters and identified 562 noises. DBSCAN estimated 14 tracker id numbers could be the group and showed their trajectories in Figure 4-a. The centroids of the trajectories are presented in Figure 4-b and the dots are plotted on the video moment in Figure 4-c. Figure 4-3 shows that multiple dots failed to group the pedestrians properly, and the silhouette scored clusters poorly. This might be caused due to data or due to the wrong parameters, and extra analyses conducted for the better cluster scores.

DBSCAN heavily relies on two parameters - epsilons, and minimum points - as the dense regions are calculated and formed by these parameters. The elbow plot is used to find optimal epsilons in Figure 3-b. The points are sorted by the distances of nearest neighbors and showed that the points between 0.005 to 0.010 would be the elbow points. When the minimum points are larger than 5, DBSCAN was unable to cluster the groups and returned errors. The epsilons from 0.005 to 0.010 and minimum points from 2 to 4 were tested and evaluated by the silhouette scores. However, Table 1 shows all scores are negative, which means the data that belongs to the cluster would be incorrect. Therefore, the DBSCAN would be optimal method to cluster trajectories for grouping and it have a poor ability in handling high dimensional data. Another problem with DBSCAN was it is the results are changed drastically depends on the epsilon and the number of minimum points.

min samples	epsilon (eps: 0.005 – 0.010)					
	0.005	0.006	0.007	0.008	0.009	0.010
2	-0.411	-0.412	-0.424	-0.422	-0.596	-0.601
3	-0.342	-0.342	-0.342	-0.341	-0.409	-0.408
4	-0.341	09.342	-0.422	-0.342	-0.342	-0.342

Table1. DBSCAN Silhouette scores

Method 3: Dynamic Time Warping K-Means Clusters



(a) *(b)*

Figure5. a) Trajectories with K-means clusters (k=3). b) Cluster points on the video moment.

Another method to detect groups was analyzing the trajectory movement as the same group trajectories are similar at different time frames. Also, the data is a collection of the observation at the different time intervals, and the last methods proposed with Dynamic Time Warping (DTW). DTW collects time series of similar shape and the centroids are computed in the algorithm like the K-means method. However, Figure 2-b and Figure 5-b results are similar and couldn't find any improvements. To measure the time efficiency, execution time is calculated for each method in Table 2. Compared to K-means, DTW-based K-means was not time efficient to compute large-scale data.

	Parameters	Time	Silhouette Score
K-means	K = 3	1.2 seconds	0.4897
	K= 4	1.4 seconds	0.4571
DBSCAN (min sample: 2)	eps = 0.005	9.34 seconds	-0.411
	eps = 0.010	9.43 seconds	-0.246
DTW	K = 3	58.34 seconds	0.448
	K = 4	82.22 seconds	0.429

Table2. Clustering Methods Evaluation

RECOMMENDATIONS

This paper mainly focused on K-means and DBSCAN clustering methods. Another approach could be DTW-based K-medoids. V. Niennattrakul et al introduced the efficacy of DTW-based K-medoids. Both K-means and K-medoids algorithm are efficient to cluster the arbitrary data points, K-medoids uses the most centrally located object in the cluster and less likely to be affected by outliers than the K-means cluster. In addition, Hausdorff distance is another approach to cluster trajectories and these methods have been applied to cluster satellite trajectories in J. Chen et al. The paper suggested that the clustering trajectories miss common sub-trajectories that contained direction information and the Hausdorff distance method was used to find the similarity between the trajectories. Likewise, the Hausdorff distance method not only contains both position and direction information but also distinguishes the different direction trajectories. This method can be utilized to estimate and predict the pedestrians' next movements.

CONCLUSION

Object detection techniques are widely used in our daily life. Simple object detection methods have significantly improved and scored high accuracy. However, detecting groups of moving objects have not progressed yet, three clustering methods were proposed and tested with the hypotheses in this paper. To evaluate each method, a silhouette score was used, and the scores were evaluated from -0.5 to 0.5 in Table 2. Table 2 shows that K-means at cluster k = 3 resulted in the highest scores. DBSCAN is expected to be the most efficient method for the trajectory density, but DBSCAN performed poorly in the dataset. Table 1 shows the different parameter results of DBSCAN, but it turned out to be the data does not well-fitted to estimate the clusters. K-means was not only a simple algorithm and implementation but also it returned faster and better results than other methods. K-means would be applied Yolo model to group the real-time moving object with fast computation and higher accuracy of the clusters.

BIOGRAPHY

Hyunjae Cho is a graduate student in the Data Science Program at the George Washington University. Her interests of the area are machine learning driven data analysis, modeling and forecasting. She enjoys Pilates, Cookie Run Kingdom game, and reading.

Dr. Nima Zahadat is a professor of information systems and computer science. His interests include data mining, information visualization, digital forensics, mobile security, risk management and information security. Dr. Zahadat has worked extensively in both the public and private sector and has held positions such as Chief Security Officer, Chief Information Officer, Directory of Security, Directory of Training Solutions, etc.

REFERENCES

D. Anh, Le Thanh (2015). An Efficient Implementation of K-Means Clustering for Time Series Data with DTW Distance, International Journal of Business Intelligence and Data Mining, 10(3) pp 213-232, doi:10.1504

Bathija and G. Sharma (2019). Visual Object Detection and Tracking using YOLO and SORT. International Journal of Engineering Research & Technology, Vol.8 Issue 11, 705-708.

J. Chen, R. Wang, L.Liu and J. Song (2011) Clustering of Trajectories Based on Hausdorff Distance, International Conference on Electronics, Communications and Control, pp. 1940-1944, doi: 10.1109

P. Bholowali, A. Kumar (2014) EBK-Means: A Clustering Technique Based on Elbow Method and K-Means in WSN, International Journal of Computer Applications, Vol. 105, No. 9, doi:10.11.7337

M. Jang, M. Han, J. Kim, and H. Yang (2011). Dynamic Time Warping Based K-Means Clustering for Accelerometer-Based Handwiriting Recognition, Studies in Computational Intelligence, vol 363. doi:10.1007

S. Kapania, S.Saina, G. Sachin, and N. Thakur (2020). Multi Object Tracking with UAVs using Deep SORT and YOLOv3 Retina Net Detection Framework, AIMS, No.1 pp1-6, doi:10.1145.

S. Kurumalla, and P.S. Rao (2016) K-Nearest Neighbor Based DBSCAN Clustering Algorithm for Image Segmentation, Journal of Theoretical and Applied Inofmration Technology, Vol. 92, No.2

Artiom Basulto-Lantsova, Jose A.Padila-Medina (2020). Performance comparative of OpenCV Template Matching method on Jetson TX2 and Jetson Nano Developer Kits, IEEE, doi:10.1109, 813- 816.

R. Nainggolan, R. Perangin-angin, E. Simarmata, and A.F. Tarigan (2019) Improved the Performance of the K-Means Cluster Using the Sum of Squared Error (SSE) Optimized by Using the Elbow Method. Journal of Physics: Conference Series.

L. Liang, H. He, J.Zhao, C. Liu, Q. Luo and X. Chi (2020). An Erasure-Coded Storage System for Edage Computing, IEEE, Vol 8, 99271-96286, doi: 10.1109.

V. Niennattrakul and C. A. Ratanamahatana (2007). On Clustering Multimedia Time Series Data Using K-Means and Dynamic Time Wraping, International Conference on Multimedia and Ubiquitous Engineering, pp. 733-738, doi: 10.1109

H. K. Kanagala and V.V. Jaya Rama Krishnaiah (2016), A comparative study of K-Means, DBSCAN and OPTICS, International Conference on Computer Communication and

Informatics pp. 1-6, doi: 10.1109

O. Ossama, H. Mokhtar, M. El-Sharkawi (2011). An Extended K-Means Technique for Clustering Moving Objects. Egyptian Informatics Journal, Vol 12 Issue1, dio.10.1016

J. Ren, Y. Guo, D. Zhang, Q. Liu, and Y. Zhang (2018) Distributed and Efficient Object Detection in Edge Computing: Challenges and Solutions, IEEE Network, Vol. 32, No. 6, pp 137-143, doi: 10.1109

F. Wang, HH. Franco-Penya, J.D. Kelleher, and R. Ross (2017). An Analysis of the Application of Simpilfied Sihouette to the Evaluation of K-Means Clustering Validation. Machnie

Z. Wang, K. Walsh, A Koirala (2019). Mango Fruit Load Estimation Using Video Based MangoYOLO Kalman Filter – Hungarian Algorithm Method. 19, 2942, doi: 10.3390.
Learning and Data Mining Pattern Recognition. Vol. 10358, doi.10.1007.

L.Zhang, S.Deng, and S. Li, (2017) Analysis of Power Consumer Behavior Based on the Complementation of K-Means and DBSCAN, IEEE Conference on Energy Internet and Energy System Integration, pp 1-5, doi: 10.1109

Y.Zhang, Z.Chen, and B.Wei (2020). An Sport Athlete Object Tracking Based on Deep Sort and Yolov4 in Case of Camera Movement, IEEE, p 1312-1316, doi: 10.1109.

Z. Fu, W. Hu, and T. Tan (2005). Similarity Based Vehicle Trajectory Clustering and Anomaly Detection, IEEE International Conference on Image Processing, pp. II-602, doi:10.1109