Geographical Self-Organizing Map Clustering in Large-Scale Urban Networks for Perimeter Control

Maha Elouni[1][a], Hesham A. Rakha[2][b], Monica Menendez[3][c] and Hossam M. Abdelghaffar[3,4][d]

[1]*Department of Computer Science, Randolph-Macon College, Ashland, Virginia, U.S.A.*

[2]*Charles E. Via, Jr. Dept. of Civil and Environmental Engineering, Virginia Tech, Blacksburg, Virginia, U.S.A.*

[3]*Division of Engineering, New York University Abu Dhabi, Abu Dhabi, U.A.E.*

[4]*Department of Computer Engineering and Systems, Faculty of Engineering, Mansoura University, Mansoura, Egypt*

Keywords: GeoSOM, Neural Network, Clustering, Traffic Congestion, Perimeter Control.

Abstract: Traffic congestion in urban areas presents a major challenge to efficient transportation systems. Recent advancements in traffic management provide promising solutions, with perimeter control emerging as a technique to tackle network-wide congestion. However, it is crucial to identify geographically connected homogeneously congested areas for effective implementation. This research explores the application of clustering techniques, particularly geographical self-organizing maps (GeoSOM), to identify spatially connected and homogeneously congested areas within transportation networks. While GeoSOM has found applications across various domains, its adaptation to transportation networks for congestion clustering is novel. This study introduces and implements an adaptation of the GeoSOM algorithm tailored for the large-scale urban environment of downtown Los Angeles. Its performance is assessed through a comparative evaluation with two other clustering algorithms, namely DBSCAN and K-means. The results demonstrate that GeoSOM surpasses other clustering algorithms, exhibiting improvements of up to 43% in traffic density variance, up to 61% in the spatial quantization error, and 15% in the quantization error. This finding demonstrates that the proposed clustering algorithm is effective in identifying a spatially homogeneous congested area within a large-scale transportation network.

1 INTRODUCTION

Traffic congestion has become a prevalent issue in many urban areas. Recent advances in traffic management and control techniques have proven effective in addressing this issue and increasing the efficiency of urban transport systems. Evaluating traffic congestion patterns across metropolitan road networks is critical for effective traffic management. This assessment allows researchers to accurately determine the operational status of network traffic, such as congested routes. Perimeter control is a promising technique for reducing traffic congestion across networks rather than considering individual routes (Bichiou et al., 2020). Identifying geographically connected homogeneously congested regions within transportation networks is critical for implementing this perimeter control technique (Lukas Ambühl, 2023). The identification of such regions is the objective of this research effort.

Technological developments in database management and data mining have simplified the handling of spatial data and allowed for the identification of intricate connections, patterns, and attributes (Henriques et al., 2012). Clustering is one way to solve problems presented by large databases. It consists of dividing the data into groups of related objects, using machine learning algorithms (Kim et al., 2023). The K-means clustering approach is widely used in studies that employ regional cluster analysis to classify important variables found in urban and rural areas (Ran et al., 2021; Kim et al., 2014). The K-means clustering method provides the benefit of reducing and displaying high-dimensional data, however, it does not take topological factors into account (Kim et al., 2023).

The concept of self-organizing maps (SOM), often

[a] https://orcid.org/0000-0002-4719-4987
[b] https://orcid.org/0000-0002-5845-2929
[c] https://orcid.org/0000-0001-5701-0523
[d] https://orcid.org/0000-0003-4396-5913

Elouni, M., Rakha, H., Menendez, M. and Abdelghaffar, H.
Geographical Self-Organizing Map Clustering in Large-Scale Urban Networks for Perimeter Control.
DOI: 10.5220/0012729300003702

In *Proceedings of the 10th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2024)*, pages 465-472
ISBN: 978-989-758-703-0; ISSN: 2184-495X

referred to as self-organizing feature maps (SOFM), was introduced by Kohonen (Kohonen, 2013). SOM involves the mapping of an input space onto a lower-dimensional output space composed of units known as neurons, forming the basis of this idea. The input space's topology is maintained by this mapping. This implies that neurons nearby are mapped to inputs nearby as well, and vice versa. Particularly in high dimensional spaces, it is an extremely helpful tool for understanding the data structure (Pires et al., 2007). SOM is essentially a data-driven technique for data compression and dimensionality reduction. It is frequently utilized for data clustering and graph mining. Numerous applications use SOM including object recognition, learning a motion map, image segmentation, density modeling, gene expression analysis, object classification, skin detection, robot behavior learning, text mining, and information management (Le Thi and Nguyen, 2014).

Although SOM is extensively utilized in clustering tasks, it typically does not consider geographic location. In clustering geo-referenced data, Baccao et al. (Bação et al., 2005) emphasized how crucial it is to take physical locations into account. "Everything is related to everything else, but near things are more related than distant things," according to the first law of geography, which they referenced (Waters, 2017). This implies that even if two parts are identical in every other way, they shouldn't be grouped if they are far apart geographically. To incorporate geographical features into the SOM clustering, two approaches were investigated by Baccao et al. With one approach, they added geographic coordinates to the input vectors and assigned them a high weight to indicate their significance. With the second approach they created a new architecture called Geographical SOM (GeoSOM) (Bação et al., 2004).

SOM and GeoSOM are used in various clustering applications including forest management (Kim et al., 2023), illness spreading patterns (Melin et al., 2020), tourism patterns (Provenzano and Giambrone, 2023), and water quality studies (Feng et al., 2023). However, as far as we are aware, they have never been applied to transportation network clustering, where the deployment of traffic controllers to alleviate congestion requires spatially connected and homogeneously congested clusters.

Therefore, in this research, the objective is to identify a homogenously congested area using the GeoSOM algorithm within a large-scale network, specifically the Los Angeles (LA) downtown. Additionally, the effectiveness of the proposed GeoSOM algorithm will be evaluated through comparison with two well-known clustering algorithms, namely, DBSCAN and K-means (Wulandari et al., 2024).

The paper is structured as follows: Section 2 describes the GeoSOM algorithm. Section 3 presents the LA case study and the simulation configuration. Section 4 performs a sensitivity analysis of the GeoSOM algorithm to determine the best set of parameters. Section 5 evaluates the effectiveness of GeoSOM by comparing it to DBSCAN and K-means clustering algorithms. Finally, Section 6 discusses the conclusions drawn from the study and future work.

2 GeoSOM ALGORITHM

The SOM is a competitive learning-based artificial neural network (Kohonen, 2013). It operates by computing the degree of similarity between the weights of each neuron and the input vector. The neuron whose weights most closely resemble the input vector is designated as the winning neuron, often referred to as the best matching unit (BMU). Subsequently, the neighborhood of the winning neuron is adjusted closer to the input vector through the updating of neuron weights.

The primary distinction between GeoSOM and SOM lies in their two-stage BMU search process. In GeoSOM, a geographical BMU (geoBMU) is initially identified, where the search relies solely on the geographic locations of neurons (Bação et al., 2004). The geoBMU is the neuron closest to the input vector in terms of geography. In the subsequent stage, non-geographical features are considered to select the final BMU, which is located within a defined radius (R) of the geoBMU. This radius, denoted as the geographical tolerance, limits the neighboring units within physical proximity to the input vector as the only contenders for becoming a BMU.

The set of N input vectors is represented as X. Each input vector X is structured as $X = [x_{geo}, x_{att}]$, where x_{att} represents the non-geographical attributes of the input, and x_{geo} denotes the geographical coordinates. In our specific context, the non-geographical attribute refers to the densities (k) of roads, indicating the number of vehicles per unit distance and serving as a metric for road congestion.

Let G denote a grid consisting of N_{neu} neurons with weights W, where $W = [w_{geo}, w_{att}]$ represents the weight of each neuron. Here, w_{att} represents the non-geographical weights, while w_{geo} represents the geographical weights. Initially, W is randomly selected. The GeoSOM algorithm, outlined in Algorithm 1, comprises three stages:

- Competition. Neurons compete to determine the best matching unit (BMU), striving to be the closest to the input.

- Collaboration. By stimulating its surrounding neurons through the neighborhood function $h_{BMU_{final},j}(t)$ often modeled as a Gaussian function, the BMU shares its success with them. Initially, the degree of collaboration is strong, but it diminishes over time. The neighborhood function is defined as follows:

$$h_{BMU_{final},j}(t) = \exp\left(-\frac{||r_{BMU_{final}} - r_j||^2}{2\sigma(t)^2}\right)$$

  where $r_{BMU_{final}}$ and r_j denote the positions of the BMU_{final} and the neuron j on the neurons' grid, respectively, and $\sigma(t)$ is the radius of the neighborhood, and it decreases with time.

- Weight Update. Neurons adjust their weights towards the input vector through a weight update process guided by a decreasing learning rate $\alpha(t)$ with time.

initialize $\alpha(1), \sigma(1), w(1), TrainingSteps, R$;

for $t = 1 : TrainingSteps$ **do**
  for $i = 1 : N$ **do**
    $BMU_{geo} = \arg\min_j (x_{i,geo} - w_{j,geo})$;
    $S_R := \{neuron : ||w_{neuron_{geo}} - w_{BMU_{geo}}|| < R\}$;
    $BMU_{final} = \arg\min_{neuron \in S_R} \left(||x_{i,att} - w_{neuron_{att}}||\right)$;
    for $j = 1 : N_{neu}$ **do**
      $h_{BMU_{final},j}(t) = e^{\frac{-||r_{BMU_{final}} - r_j||^2}{2\sigma(t)^2}}$
      $w_j(t+1) = w_j(t) + \alpha(t) h_{BMU_{final},j}(t) \left(x_i - w_j(t)\right)$;
    end
  end
  update $\alpha(t) = \alpha(1) * e^{-t/TrainingSteps}$;
  update $\sigma(t) = \sigma(1) * e^{-t/TrainingSteps}$
end

Algorithm 1: GeoSOM Algorithm.

3 CASE STUDY: NETWORK DESCRIPTION AND SIMULATION SETUP

The real-life Los Angeles (LA) downtown network shown in Figure 1 is a large-scale network composed of 3,556 links (Abdelghaffar and Rakha, 2019). It is characterized by congested traffic, long travel times,

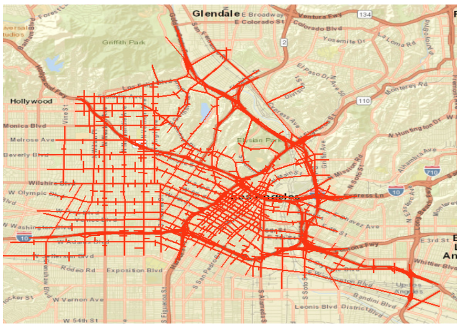


Figure 1: Downtown Los Angeles network.

and frequent delays. To alleviate the congestion, our ultimate goal is to apply the network perimeter control strategy in the congested region (Elouni et al., 2021). For the control system to function properly, the area where it will be implemented must exhibit homogeneous congestion. In other words, a densely homogeneous area that is geographically connected must be located. Therefore, the GeoSOM algorithm is deployed across the network, considering both the link densities and their geographic locations, to achieve this objective.

Every network link is characterized by two features: its density (k), considered a non-geographical feature, and its midpoint location, represented by x and y coordinates, considered a geographical feature. To prevent any feature from being weighted more than the others, x, y, and k are normalized to values between 0 and 1.

Throughout the simulation, the neuron weights are updated until they converge to a state where either the change becomes minimal or ceases altogether. Then, the distances between each neuron and its neighboring neurons are computed using the unified distance matrix, commonly referred to as the U-matrix (Hamel and Brown, 2011). Neurons are grouped into the same clusters when the distances between them are relatively small, while cluster boundaries are delineated by larger distances between neurons. To define a color code for each network link, the U-matrix results are projected and interpolated into the input space. Based on these colors, the clusters are then visually identified.

4 SENSITIVITY ANALYSIS

The objective of this research is to identify a heavily congested homogeneous area to manage congestion within it. In evaluating GeoSOM's effectiveness, our emphasis was on identifying a specific area characterized by a high volume of vehicles, i.e., high traffic density. The peak density observed in the LA real traffic volume during rush hour is represented by the yellow zone in Figure 2.

GeoSOM operates on artificial neural networks and encompasses multiple parameters that necessitate fine-tuning. These parameters include the number of neurons, the learning rate, and geographical tolerance. The best configuration is determined through a sensitivity analysis, as presented below.

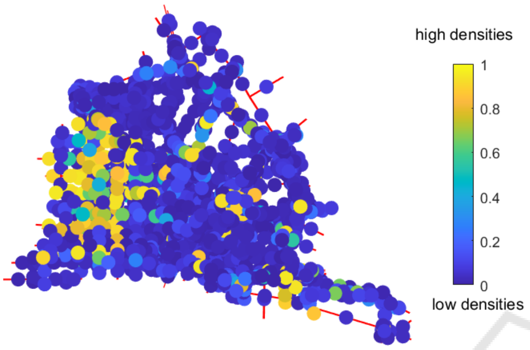


Figure 2: LA real link densities.

4.1 Number of Neurons

GeoSOM was executed with varying numbers of neurons, ranging from a 5×5 to a 25×25 grid, on the LA network. Figure 3 depicts three cases, corresponding to grid sizes of 5×5, 15×15, and 25×25, respectively, from left to right.

Figure 3a illustrates the neurons' grid at the end of the simulation, following Algorithm 1. Meanwhile, Figure 3b displays a color-coded interpolation of the neurons' weights onto the network links. Blue indicates similar densities among the links, implying they could potentially be clustered together, whereas yellow indicates a notable disparity in densities, implying they should not be clustered together. The leftmost plot in Figure 3a shows that the neurons did not adequately cover the map, implying that a 5×5 neuron configuration is inadequate. Furthermore, the outcome for the 25×25 neuron configuration, as depicted in Figure 3b, reveals that all the links are grouped into only a single large cluster. Therefore, 25×25 neurons are ineffective. Figure 3 shows that the best number of neurons is 15×15. The neuron grid efficiently covers the network, as evidenced by the middle plot in Figure 3a, and it also effectively achieves homogeneous clusters, as indicated by the dark blue color with clear light color boundaries in the middle plot of Figure 3b.

4.2 Learning Rate

In this section, various learning rates (α) were experimented within the GeoSOM algorithm. As α falls between 0 and 1, the tested values ranged between 0.05 to 0.8. Figure 4 shows GeoSOM results for three different α values. It is evident that for small α values (e.g., $\alpha = 0.05$), the homogeneous regions, characterized by a dark blue color, are quite small, which may not be optimal for the control objective. Conversely, with a high α value (e.g., $\alpha = 0.8$), the network is consolidated into a single cluster. The best α value appears to be 0.4, where a well-defined homogeneous dark blue area is discernible with clear distinct borders, exhibited by yellow and green colors, making it more suitable for control purposes.

4.3 Geographical Tolerance

This section investigates the impact of geographical tolerance (R) on GeoSOM results, with low R values indicating that geographical attributes are prioritized and higher R values indicating a preference for non-geographical attributes over geography. The R values tested with the GeoSOM algorithm ranged between 0 and 8.

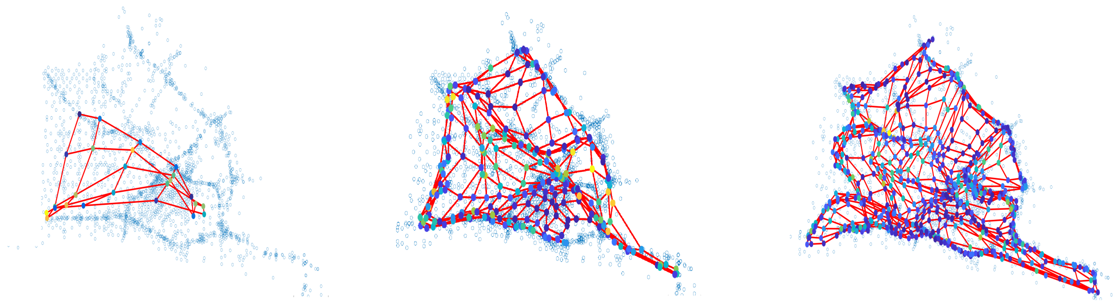
Figure 5 illustrates that for $R = 0$, the clustering results did not yield a homogeneous dark blue area. As R increases, the dark blue area expands. A well-defined cluster emerged when $R = 5$. However, as the geographical tolerance increases to $R = 8$, cluster borders start to blur, and all links are grouped into a single cluster.

4.4 Sensitivity Analysis Conclusion

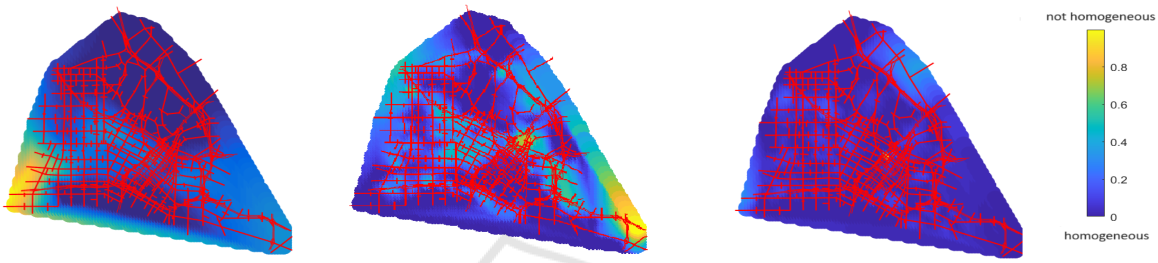
The best configuration for the GeoSOM clustering algorithm is identified as 15×15 neurons, $\alpha = 0.4$, and $R = 5$. Within the various homogeneous areas depicted by the dark blue color in Figure 6, the area of primary interest is the one with the highest mean density, representing the most congested area, as illustrated in Figure 2, and depicted by the black contour in Figure 6.

5 COMPARING GeoSOM TO OTHER CLUSTERING ALGORITHMS

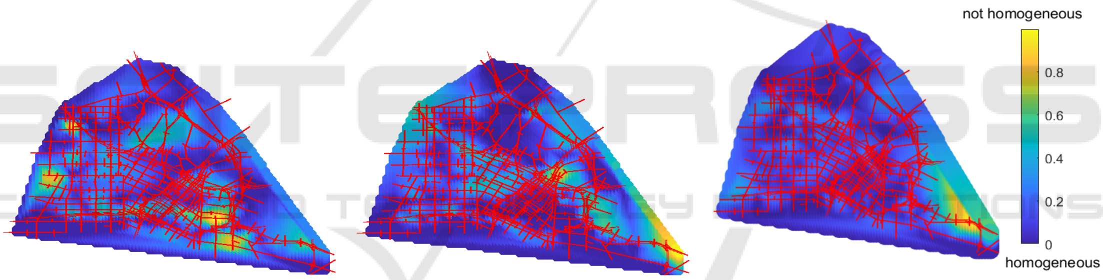
In this Section, the GeoSOM algorithm is compared with two other widely used clustering techniques: DBSCAN and K-means, to evaluate its performance.



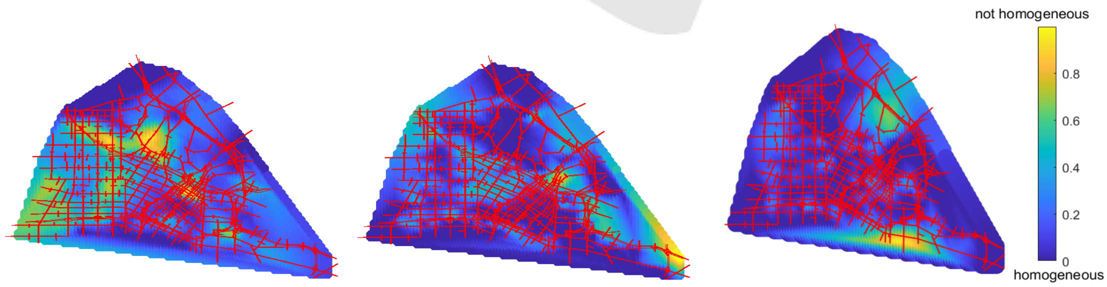
(a) Grid of neurons.



(b) Color-coded distances between the network links.
Figure 3: GeoSOM with different number of neurons; from left to right, 5×5, 15×15 and 25×25.



(a) $\alpha = 0.05$. (b) $\alpha = 0.4$. (c) $\alpha = 0.8$.
Figure 4: GeoSOM maps for various α.



(a) R=0. (b) R=5. (c) R=8.
Figure 5: GeoSOM maps for various R.

While these two methods have been employed in the literature for clustering transport networks, they lack the explicit incorporation of geographical data, a feature inherent in GeoSOM (Lopez et al., 2017; Lin and Xu, 2020).

5.1 DBSCAN

The density-based spatial clustering algorithm (DBSCAN) (Sahu et al., 2023) clusters points based on their proximity to neighboring points. It relies on two key parameters: ε, representing the radius of a neigh-

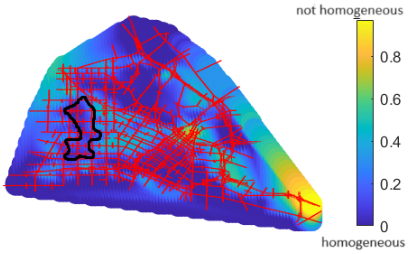


Figure 6: GeoSOM cluster identification using best configuration.

borhood relative to a point, and *minPts*, which specifies the minimum number of points needed to constitute a dense region, i.e., the minimum cluster size.

DBSCAN inputs are (x, y, k), where x and y represent geographical coordinates and k represents the link density. DBSCAN's sensitivity analysis utilizes a k-distance graph to determine ε for each selected *minPts*. The optimal values of ε are identified where this graph exhibits an "elbow" as shown in Figure 7. Ultimately, the best parameters were found to be $minPts = 9$ and $\varepsilon = 0.8$.

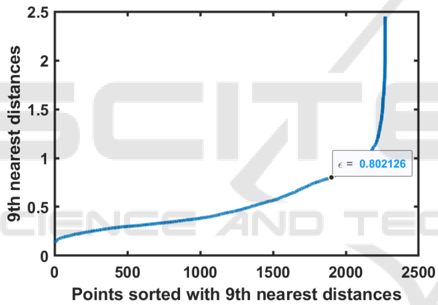


Figure 7: K-distance graph for DBSCAN.

Figure 8 shows the outcomes of the clustering process. It is observed that clusters 1, 4, and 5 exhibit high congestion levels and are geographically connected based on the data in Figure 2. To facilitate comparison with the GeoSOM cluster, these three clusters were merged. The combined cluster is illustrated in blue on the lower plot of Figure 8.

5.2 K-Means

K-means is a clustering algorithm that partitions data points into k_m clusters and assigns each point to the cluster with the nearest mean. Often, it serves as a benchmark to assess the performance of other clustering algorithms (Lopez et al., 2017; Saeedmanesh and Geroliminis, 2016). Unlike DBSCAN, K-means only requires the determination of one parameter, which is the number of clusters.

The Calinski-Harabasz method (Aik et al., 2023)

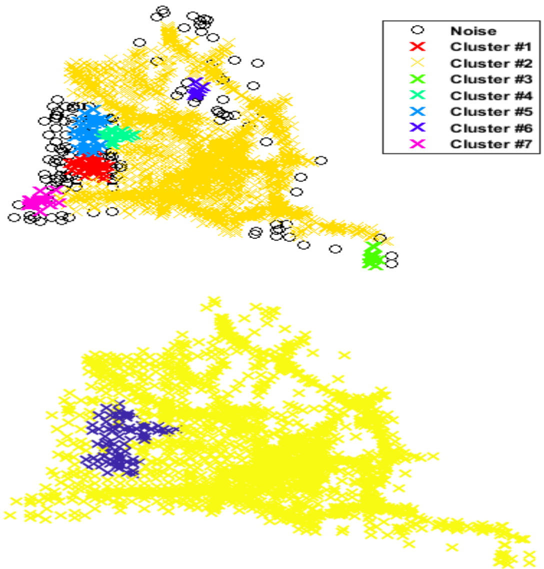


Figure 8: DBSCAN.

is employed to identify the best number of clusters. The results obtained using this method for k_m values ranging from 2 to 30 indicate that $k_m = 12$ is the best number of clusters, i.e., the maximum value in Figure 9.

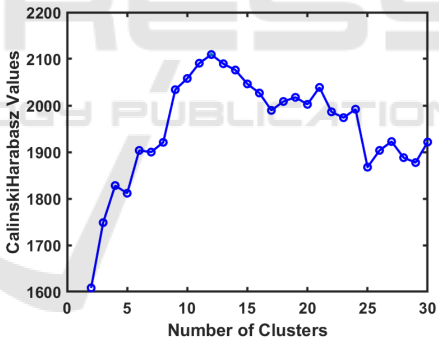


Figure 9: Calinski Harabasz for K-means.

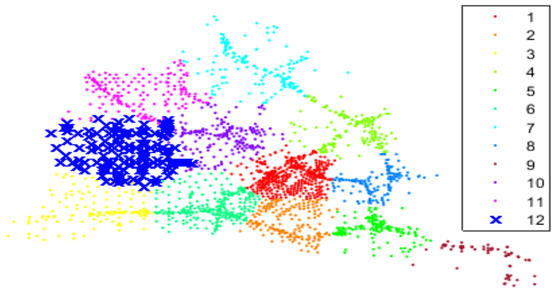
The clustering outcomes are displayed in Figure 10, where cluster 12 corresponds to the most congested area, as inferred from the data presented in Figure 2.

5.3 Performance Metrics

To evaluate the performance of the GeoSOM algorithm, we compare it to the DBSCAN and K-means algorithms using the following metrics:

- Quantization Error (QE):

$$QE = \frac{||k(cluster) - \bar{k}(cluster)||}{length(cluster)}$$



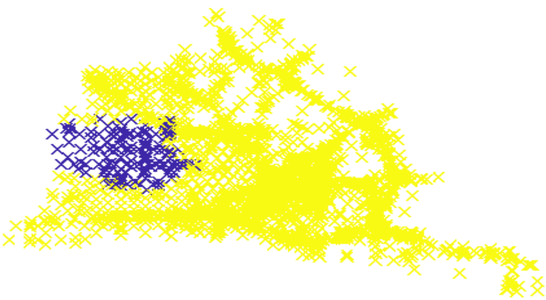


Figure 10: K-means.

where $\bar{k}(cluster)$ represents the cluster's mean density, and $length(cluster)$ represents the cluster's number of elements.

- Spatial Quantization Error (SQE):

$$SQE = \frac{||xy(cluster) - \bar{xy}(cluster)||}{length(cluster)}$$

where $\bar{xy}(cluster)$ represents the x and y coordinates of the cluster's centre.

- Density Variance (DV): It refers to the density variance within the cluster, reflecting how much the densities of individual points within the cluster deviate from the mean density.

Table 1 illustrates the three performance metrics for the three clustering algorithms. The performance is improved when QE, SQE, and DV are minimized. Table 1 demonstrates that GeoSOM outperforms the other two algorithms.

Table 1: Performance metrics.

	QE	SQE	DV
DBSCAN	0.0396	0.0095	0.1241
K-means	0.0333	0.0055	0.1494
GeoSOM	0.0336	0.0037	0.0837

Table 2 presents the percentage improvements of GeoSOM over DBSCAN and K-means across various performance metrics. The most significant enhancements are observed in SQE and DV, aligning with GeoSOM's objectives of fostering spatially connected links within clusters (yielding the best SQE) and achieving clusters with minimal density variance.

Table 2: Improvement percentage (%).

	GeoSOM over DBSCAN	GeoSOM over K-means
QE	15.15	-0.9
SQE	61.05	32.72
DV	32.55	43.98

The results reveal that the GeoSOM approach successfully achieved the research goal of identifying highly congested and geographically compact clusters with low-density variance, surpassing DBSCAN and K-means with improvement percentages reaching up to 43% in DV, up to 61% in SQE, and up to 15% in QE. These findings could ultimately be used for the development of traffic control systems aimed at alleviating congestion within the network.

6 CONCLUSION

In this research, a novel clustering algorithm is proposed (GeoSOM) to identify densely congested and spatially compact areas within transportation networks. This study provides valuable insights for developing effective traffic control strategies to alleviate traffic congestion. The proposed clustering algorithm is evaluated on a large-scale urban network in downtown Los Angeles, and its performance is compared to two other clustering algorithms, namely: DBSCAN and K-means. The results demonstrated enhancements across all three performance metrics when using GeoSOM compared to DBSCAN and K-means. Specifically, there was a 15% reduction in the quantization error, a reduction of up to 43% in the traffic stream density variance within the cluster, and a reduction of up to 61% in the spatial quantization error. The results also demonstrate GeoSOM's effectiveness in accurately delineating spatially homogeneous, congested areas within a large-scale network. This research establishes a framework for harnessing advanced clustering algorithms to tackle intricate traffic management challenges, thereby paving the way for more efficient urban mobility solutions. Future research will involve leveraging this study's findings to develop network perimeter control strategies targeting congested areas to reduce congestion and improve the overall efficiency of urban transport systems.

ACKNOWLEDGEMENTS

This work was partially funded by the Department of Energy through the Office of Energy Efficiency and Renewable Energy (EERE), Vehicle Technologies Office, Energy Efficient Mobility Systems Program under award number DE-EE0008209. M. Elouni acknowledges the receipt of the Chenery Grant from Randolph-Macon College. M. Menendez acknowledges the support of the NYUAD Center for Interacting Urban Networks (CITIES), funded by Tamkeen under the NYUAD Research Institute Award CG001.

REFERENCES

Abdelghaffar, H. M. and Rakha, H. A. (2019). A novel decentralized game-theoretic adaptive traffic signal controller: Large-scale testing. *Sensors*, 19(10).

Aik, L. E., Choon, T. W., and Abu, M. S. (2023). K-means algorithm based on flower pollination algorithm and calinski-harabasz index. In *Journal of Physics: Conference Series*, volume 2643, page 012019. IOP Publishing.

Bação, F., Lobo, V., and Painho, M. (2004). Geo-self-organizing map (geo-som) for building and exploring homogeneous regions. In *International Conference on Geographic Information Science*, pages 22–37. Springer.

Bação, F., Lobo, V., and Painho, M. (2005). Geo-som and its integration with geographic information systems. In *Proc. Workshop on Self-Organizing Maps, Paris, France*.

Bichiou, Y., Elouni, M., Abdelghaffar, H. M., and Rakha, H. A. (2020). Sliding mode network perimeter control. *IEEE Transactions on Intelligent Transportation Systems*, 22(5):2933–2942.

Elouni, M., Abdelghaffar, H. M., and Rakha, H. A. (2021). Adaptive traffic signal control: Game-theoretic decentralized vs. centralized perimeter control. *Sensors*, 21(1):274.

Feng, Z., Xu, C., Zuo, Y., Luo, X., Wang, L., Chen, H., Xie, X., Yan, D., and Liang, T. (2023). Analysis of water quality indexes and their relationships with vegetation using self-organizing map and geographically and temporally weighted regression. *Environmental Research*, 216:114587.

Hamel, L. and Brown, C. W. (2011). Improved interpretability of the unified distance matrix with connected components. In *Proceedings of the International Conference on Data Science (ICDATA)*, page 1. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).

Henriques, R., Bacao, F., and Lobo, V. (2012). Exploratory geospatial data analysis using the geosom suite. *Computers, Environment and Urban Systems*, 36(3):218–232.

Kim, J.-Y., Gim, U. S., and Oh, M. T. (2014). Characteristic analysis and classification of rural areas: Based on the eup and myon areas of chungcheongnam-do. *The Korean Regional Development Association*, 26(1):27–44.

Kim, T.-S., Dhakal, T., Kim, S.-H., Lee, J.-H., Kim, S.-J., and Jang, G.-S. (2023). Examining village characteristics for forest management using self-and geographic self-organizing maps: A case from the baekdudaegan mountain range network in korea. *Ecological Indicators*, 148:110070.

Kohonen, T. (2013). Essentials of the self-organizing map. *Neural networks*, 37:52–65.

Le Thi, H. A. and Nguyen, M. C. (2014). Self-organizing maps by difference of convex functions optimization. *Data Mining and Knowledge Discovery*, 28(5-6):1336–1365.

Lin, X. and Xu, J. (2020). Road network partitioning method based on canopy-kmeans clustering algorithm. *Archives of Transport*, 54(2):95–106.

Lopez, C., Krishnakumari, P., Leclercq, L., Chiabaut, N., and van Lint, H. (2017). Spatiotemporal partitioning of transportation network using travel time data. *Transportation Research Record*, 2623(1):98–107.

Lukas Ambühl, Monica Menendez, M. C. G. (2023). Understanding congestion propagation by combining percolation theory with the macroscopic fundamental diagram. *Communications Physics*, 6(26).

Melin, P., Monica, J. C., Sanchez, D., and Castillo, O. (2020). Analysis of spatial spread relationships of coronavirus (covid-19) pandemic in the world using self organizing maps. *Chaos, Solitons & Fractals*, 138:109917.

Pires, F. J., Lobo, V., and Bação, F. (2007). Insights on the interpretation of som and u-matrices with an example clustering based in oceanographic data. In *10th AGILE international conference on geographic information science, Aalborg University, Denmark*.

Provenzano, D. and Giambrone, R. (2023). Clustering of tourism patterns with self-organizing maps: The case of sicily. *Tourism Analysis*.

Ran, X., Zhou, X., Lei, M., Tepsan, W., and Deng, W. (2021). A novel k-means clustering algorithm with a noise algorithm for capturing urban hotspots. *Applied Sciences*, 11(23):11202.

Saeedmanesh, M. and Geroliminis, N. (2016). Clustering of heterogeneous networks with directional flows based on "snake" similarities. *Transportation Research Part B: Methodological*, 91:250–269.

Sahu, R. T., Verma, M. K., and Ahmad, I. (2023). Density-based spatial clustering of application with noise approach for regionalisation and its effect on hierarchical clustering. *International Journal of Hydrology Science and Technology*, 16(3):240–269.

Waters, N. (2017). Tobler's first law of geography. *The international encyclopedia of geography*, pages 1–13.

Wulandari, V., Syarif, Y., Alfian, Z., Althof, M. A., and Mufidah, M. (2024). Comparison of density-based spatial clustering of applications with noise (dbscan), k-means and x-means algorithms on shopping trends data. *IJATIS: Indonesian Journal of Applied Technology and Innovation Science*, 1(1):1–8.