Detecting Plumes in Mobile Air Quality Monitoring Time Series with Density-based Spatial Clustering of Applications with Noise (DBSCAN)

Blake Actkinson1, Robert J. Griffin1,2\*

1Department of Civil and Environmental Engineering, Rice University, Houston, TX 77005, USA

2School of Engineering, Computing, and Construction Management, Roger Williams University, Bristol, RI 02809, USA

\*Corresponding Author: Robert Griffin, *E-mail:* [*rgriffin@rwu.edu*](mailto:rgriffin@rwu.edu)

**ABSTRACT**

Mobile monitoring is becoming an increasingly popular technique to analyze air pollution measurements on fine spatial scales, but methods to determine specific source contributions are sorely needed. One approach is to isolate plumes from mobile monitoring time series and analyze them separately, but methods which are suitable for large mobile monitoring time series are lacking. Here we discuss a novel method used to detect and isolate plumes from an extensive mobile monitoring data set. The new method relies on Density-based Spatial Clustering of Applications with Noise (DBSCAN), an unsupervised machine learning technique. The new method systematically runs DBSCAN on mobile monitoring time series by day and identifies a subset of points as anomalies for further analysis. When applied to a mobile monitoring data set collected in Houston, Texas, analyzed anomalies reveal patterns previously associated with different types of vehicle emission profiles. We observe spatial differences in these patterns and reveal striking disparities by census tract. These results can be used to inform stakeholders of spatial variations in emission profiles not obvious through the use of data from stationary monitors alone.

**TOC ART**

**Map

Description automatically generated**

1. **INTRODUCTION**

A central question of air pollution studies is to identify the varied sources that contribute to measured pollutant concentrations. This question becomes more complicated in a mobile monitoring context because measurements and concentrations vary as a function of both space and time, making conventional source apportionment techniques such as positive matrix factorization and principal component analysis (PCA) harder to apply effectively.1

Recently published work took several approaches to source apportionment on measured pollutants in a mobile monitoring context. One approach involves using PCA on background subtracted measurements, such as in Larson et al,1 whose approach has limitations when applied to extensive mobile monitoring campaigns, in that it defines defining a rolling minimum across a static time window which may not be feasible for extensive mobile monitoring campaigns with 20-30x the temporal coverage. Other approaches have focused on using Land Use Regression (LUR) models to identify relationships between pollutants and land use variables, such as in Messier et al.2 However, use of LUR models often involves aggregating measurements in time, dropping temporal resolution. Additionally, derived relationships do not necessarily imply causality, and use of these models is limited through the availability of sophisticated land use variable databases. There is a need for the development of methods that can identify source influences in large mobile monitoring data sets at high time resolution without being subject to the availability of pre-existing, sophisticated databases.

Another factor that aggravates source identification in mobile monitoring contexts is the nature of mobile monitoring data itself. If a mobile monitoring campaign were conducted focusing largely on residential areas with brief excursions onto busier thoroughfares, performing a PCA or other dimension reduction techniques to describe patterns in the data would likely return results that are weighted towards data in which source influences are largely negligible. This type of analysis could frequently generate solutions in which there is a demarcation between the majority of points with little source influences and a smaller subset of source-influenced points elevated in all pollutants, which may not be compelling if one’s objective is to determine the specific sources affecting the measurements.

This raises the question of how to identify source influences within mobile monitoring time series that cover locations ranging from ‘background’ to ‘highly influenced by sources.’ If one could identify source spikes, or plumes within mobile monitoring time series, one could restrict their analysis to these plumes to categorize the different types of sources that affected their mobile monitoring measurements. Plume identification within mobile monitoring time series has been addressed previously. Hagler et al. use a rolling coefficient of variation across a 5-s time interval, then flag points with a coefficient greater than 2.3 Drewnick et al. use a different moving window algorithm that calculates the standard deviation of points below a defined background threshold () and flags points which are greater than 3 of the previous point. The algorithm then flags subsequent points, increasing the threshold necessary (by a factor of , in which n is the total number of flagged points) for flagging for every subsequent point beyond the first flagged.4 Others have addressed the plume identification question indirectly through background estimation and removal methods.5

These methods all have drawbacks. In the data used in the present work, the method of Hagler et al. flags few to no points at all, suggesting that the method is sensitive to the time series utilized. The algorithm of Drewnick et al. suffers in situations where many plumes appear consecutively to one another, frequently leading to poor performance in those circumstances.4 Other methods depend on time window, which presents problems for complex, multi-day mobile monitoring time series.

Here we discuss an algorithm to identify plumes in a different manner. The algorithm relies on Density-based Spatial Clustering of Applications with Noise (DBSCAN), a nearest neighbor clustering algorithm. DBSCAN clusters points based on whether they fall into predetermined neighborhoods with other points. The technique can cluster points with more complicated shapes (e.g. an “S” embedded in noise in 2D space) and is not sensitive to starting values compared to other clustering techniques such as k-means.6 Additionally, the algorithm does not require every single point to be clustered, allowing for those points that do not neatly fall into a given cluster to be defined as noise.

We utilize DBSCAN by envisioning daily mobile monitoring time series consisting of black carbon (BC), carbon dioxide (CO2), oxides of nitrogen (NO+NO2 = NOx) and ultrafine particle number concentrations (UFP) as large numbers of points clustered around a 4-dimensional origin with plumes scattered outwards from this origin. In the DBSCAN context, plumes would be labeled as noise. We first describe DBSCAN, then detail how we adapt it for application to mobile monitoring time series. To evaluate performance, we construct a validation set by manually flagging plumes via visual inspection from a randomly chosen subset of days from a Houston mobile monitoring campaign.7,8 We use the validation set to tune DBSCAN and other time series-based models and compare performance of all models. We apply the algorithm to the Houston mobile monitoring dataset to identify anomalies, which are then clustered into anomaly types linked to specific emission sources. We tabulate the number of these different anomaly types by census tract and discuss the implications of the method, the results, and future directions for this research.

1. **METHODS**
   1. **Data**

Data were collected during the Houston mobile monitoring campaign and are described in detail elsewhere.7,8 In brief, two Google Street View cars drove different census tracts in the Greater Houston area systematically to evaluate spatial differences in the concentrations of 7 pollutants. In this analysis, we restrict the set of analyzed pollutants to be BC, CO2, UFP, and NOx. A table providing the instruments used to measure each respective pollutant is given in Table S1.

We create a multivariate dataset consisting of these four air pollution variables, along with corresponding latitude/longitude coordinates and timestamps, that spans 277 separate days of sampling, for a total of 5,301,507 observations. The BC data were smoothed with a 10-s time window to limit the effects of noise on subsequent analysis. Missing data were imputed up to a 6-s time gap with monotone Hermitian splines. Any multivariate realization with at least one missing observation in a variable not imputed was excluded otherwise. Days had to possess a minimum of 600 realizations to be included in the analysis. To be consistent with Miller and Actkinson et al.,7,8 we restrict our analysis to points with logged latitude/longitude coordinates on primary, secondary, local, and private roads, as well as ramps and service drives as defined by the TigerLINE census.9 Additionally, we remove logged GPS coordinates > 30m away from a given TigerLINE shapefile point, as these points were subject to GPS error, as well as points < 30 m from a dead end in a road due to potential self-sampling contamination.

The following R packages were used in the analysis and visualization of results: tidyverse,10 ggpubr,11 caret,12 dbscan,13 leaflet,14 leafem,15 sf,16 mapview,17 scattermore,18 and data.table.19

* 1. **DBSCAN – A Brief Description**

DBSCAN is a clustering routine originally conceived by Ester et al.20 Using two predefined parameters, epsilon () and MinPts, DBSCAN seeks to label points that have MinPts points within a neighborhood defined with radius as core points, points that do not meet the MinPts criteria but have a core point within their - neighborhood as border points, and points that do not fit either above criteria as noise.

More formally, the -neighborhood is defined using the notation of Hahsler13 as

where N is the neighborhood, D is the set of points, and d is a distance measure such as the Euclidean distance. A point is defined as a core point if

where MinPts is the minimum points parameter and || denotes cardinality. The algorithm systematically labels points as core points, border points, or noise points depending on these criteria.

* 1. **Validation Set Construction**

To tune parameters and evaluate algorithm performance, a validation set was constructed from the mobile monitoring data by manually flagging visible plumes within 30 randomly selected daily mobile monitoring time series (out of a possible total of 277). The total number of points in the validation set was 564,107, which amounts to 10% of the entire set (5,712,090 total points). A graphical user interface in IgorPro was used to flag plumes by visually inspecting the time series for spikes in pollutant concentrations for each pollutant (BC, CO2, NOx, UFP). Any time series realization that had a spike in at least one pollutant was flagged.

* 1. **Algorithm Description**

An algorithm incorporating DBSCAN was created to systematically label anomalies within the Houston mobile monitoring campaign. A flowchart depicting this algorithm’s steps is given in Figure S1. The algorithm estimates eps () and MinPts parameters for daily time series in the campaign via statistical heuristics and performs DBSCAN using these estimated parameters. The MinPts parameter was defined to be the product of the total number of points in the daily time series, , and a fractional value parameter, . More formally,

where

An appropriate value of of 0.03 was determined using the external validation set and is described in Section 2.6. Values of greater than 0.5 were not considered due to rapidly increasing computational cost and apparent poor performance at higher values.

The value of was determined by a k-nearest-neighbor distance ordering procedure in which the value of k was set to be equal to MinPts. We constructed an ordered k-nearest-neighbor distance set and determine the mean and standard deviation of the first 30 ordered distances. Then, was set to the first subsequent distance in the remaining ordered set that met the criteria:

where is the ordered k-nearest-neighbor distance, is the mean of the ordered distances between indices 1 and , and is the standard deviation of ordered distances

between 1 and . A graphical example illustrating the estimated epsilon value for a knn ordered distance graph is given in Figure 1. Given both and MinPts, DBSCAN was run on the daily time series observations in which core points were labeled as normal and both border and noise points labeled as anomalies. An example of labeled DBSCAN output for a scatterplot of daily BC/CO2 time series is given in Figure 2.

Diagram

Description automatically generated with medium confidence

**Figure 1.** Graphical depiction of the selection criteria in equation (5).

Chart, scatter chart

Description automatically generated **Figure 2.** Daily scatterplot example of DBSCAN labeled anomalies for CO2 against BC. Points labeled as normal are 2/3 of the time series realizations in this example.

**2.5 Description of Other Algorithms**

To put the performance of the DBSCAN anomaly detection algorithm in context, we compare its labeled anomalies with output from the previously described plume detection technique of Drewnick et al.4 (referred to as “Drewnick” moving forward) or base-case 90th-quantile algorithms. These two base-case algorithms, the Quantile-OR (QOR) and the Quantile-AND (QAND) algorithms, flag points as anomalous based on criteria centered around the 90th quantile of pollutant distributions. In the QOR case, points are flagged as anomalous if any one pollutant measurement (BC, , , or UFP) is above the 90th quantile for the given daily time series (if > 90th BC OR > 90th OR > 90th OR > 90th UFP). In the QAND case, points are flagged as anomalous if *all* pollutant measurements are greater than their respective 90th quantiles (if > 90th BC AND > 90th AND > 90th AND > 90th UFP). These algorithms, along with the Drewnick algorithm,4 were run on all daily time series to assess performance.

**2.6 Using the External Validation Set to Tune Parameters and Evaluate Performance**

To determine an appropriate value of for use in the DBSCAN algorithm, we perform grid search on values in [0.01,0.10] in increments of 0.01 and [0.15,0.50] in increments of 0.05. Values above 0.5 were not considered due to computational cost and apparent poor performance at higher values of . Performance was validated using percentage agreement, defined as

Where is the indicator function which evaluates to 1 if the condition is true and 0 otherwise, is the prediction label at point , is the validation set label at point , and is the total number of points in the validation set. Tuning results indicated that a value of 0.03 was most appropriate for , which was used in subsequent analyses. In addition to the parameter, the quantile parameter was tuned with the external validation set. Quantiles near the 90th were found to return only modest improvements, and thus the 90th quantile was analyzed for simplicity.

To evaluate whether we overfit to this validation set, we perform k-fold cross validation with the number of folds k = 5. We train our models on 4 out of 5 folds, tuning the parameter such that the model performance agreement is maximized on the testing set. We find that the value of which results in superior performance to be 0.03, suggesting that our work above generalizes appropriately. K-fold cross validation results are given in Table S2.

We also use this same validation set to compare performance across all 4 algorithms examined in this study. We evaluate performance of each by calculating the percentage agreement between each algorithm’s labels and the validation set labels.

**2.7 Interpretation: K-Means Clustering and Principal Component Analysis**

We perform k-means clustering on the extracted anomalies in R using the kmeans function available in base.21 The number of centers (clusters) was set to 3 and 200 iterations with different random starts were chosen to ensure the derived result was robust to utilized starting values. Cluster labels were assigned based on the cluster means to ensure consistency in label assignment. Principal component loadings and scores for visualization were calculated using prcomp available in R base on scaled measurements.21 Visualization itself was done using R packages scattermore18 and tidyverse.10 Varimax rotation was performed using R package psych22 to compare with results in a previously published study.1

Boxplots of assigned roadway trucking variables were created to probe potential meanings of clustered anomalies. Roadway trucking variables were extracted from TxDOT’s roadway inventory with processing performed using R package sf.23 Records along the same road segment were averaged with weights equivalent to the distance between fields in the shapefile FROM\_DFO and TO\_DFO, which are distance measures representing starting and ending points for those records in the shapefile. Extracted roadway variables from the shapefile included Truck Annual Average Daily Traffic Percentage (TRUCK\_AADT\_PCT) and the number of all trucks in AADT (AADT\_TRUCKS).

**2.8 Census Tract Assignment**

To determine differences in anomaly frequency between census tracts, points were assigned16 to census tracts using tract boundaries stored in a shapefile used in previously published analyses of the same campaign data.7,8 Anomalies of a given cluster assignment were counted and divided by the total recorded measurements in each polygon. Because each census tract was imperfectly sampled, we implement a rescaling procedure described in detail in the SI. As part of that procedure, we restrict the comparisons to 19 of the 35 census tracts, to measurements taken between 8 AM and 4 PM, and to measurements taken on weekdays. To account for different polygons containing differing numbers of measurements, we divide the total amount of rescaled anomaly types by the total number of measurements made in the census tract, deriving a probability of encountering the specified anomaly type during the campaign in the restricted time interval described above.

1. **RESULTS**

**3.1 External Validation**

All four algorithms - Drewnick, QOR, QAND, and DBSCAN - were run on the Houston mobile monitoring campaign data. To compare performance, each algorithm’s labeled anomalies were compared with the anomalies of the validation set on the same subset of days, which were taken to be the ground truth.

Of the 4 algorithms, DBSCAN had the best performance, with its labels exhibiting 86.9% agreement with the validation set’s labels. The QOR, QAND, and Drewnick algorithms exhibit 85.5%, 77.0%, and 81.8% agreement, respectively. For context, an algorithm that simply labeled all points as normal would generate 74.7% agreement with the validation set. Because this baseline agreement is so high, confusion matrices were created to probe sources of agreement and disagreement between each algorithm’s predicted anomalies and the validation set labeled anomalies. These confusion matrices tabulate the number of points that a given algorithm labels as normal or anomaly which agree or disagree with the corresponding labels in the validation set. Confusion matrices for the four algorithms are presented in Figure 3.

Figure 3 illustrates that even though the DBSCAN algorithm exhibits greater overall agreement with the validation set, it predicts anomalies less successfully compared to the QOR algorithm. However, the DBSCAN algorithm outperforms the QOR algorithm in its ability to not predict normal points as anomalous. This suggests that the QOR algorithm captures the most anomalies but is a coarse approach to doing so; the DBSCAN algorithm captures fewer anomalies but is less likely to predict something as anomalous when it is not. Table S4 contains counts of instances in which one algorithm made a mistake of a given type when the other did not. Table S4 serves to provide further evidence that the DBSCAN algorithm is inferior in its ability to label anomalous points compared to the QOR algorithm, while the QOR algorithm is inferior in its ability to not label normal points as anomalous. For the purposes of further analysis, we focus our attention on DBSCAN derived anomalies, bringing in QOR derived anomalies periodically for comparison. We choose to focus on results from DBSCAN as the approach is more conservative; it does not result in as many false positives as the QOR algorithm and provides confidence that what is being analyzed is an anomaly. The QAND and Drewnick algorithms do not offer superior performance over the DBSCAN and QOR algorithmsChart, bar chart

Description automatically generated. They will not be considered for further analysis.

**Figure 3.** Confusion matrices corresponding to the performance of (a) DBSCAN, (b) Drewnick, (c) QOR, and (d) QAND. Overall agreement between each algorithm and the validation set was (a) 86.9%, (b) 85.5%, (c) 81.8%, and (d) 77.0%

**3.2 K-Means Clustering and PCA**

Detected anomalies were clustered using R function kmeans and consistently yielded one cluster rich in concentrations (“CO2 Cluster”), another cluster rich in (“BC/UFP Cluster”) concentrations, and a third cluster that contains lower concentrations of all 4 for both QOR and DBSCAN derived anomalies (“Transition Cluster”). Table 1 contains cluster means of the three derived clusters from the QOR and DBSCAN algorithms. Figure S2 shows cluster assignments of DBSCAN labeled anomalies projected onto the first two principal component axes in a PCA biplot. Figure 4 displays boxplots of the clustered DBSCAN pollutant anomalies. In Figure S2, it is evident that the CO2-rich cluster points in the direction of CO2 loadings, while the BC/UFP-rich cluster points in the direction of the BC/UFP loadings. These loadings when varimax rotated are consistent in split with those reported in Larson et al.; loadings are sequestered into BC/UFP-rich and CO2-rich factors.1 However, they differ in value. These loadings are given in Table S3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | (ppm) | BC (ng/m3) | (ppb) | UFP (p/cc) |
| DBSCAN | | | | |
| 1st cluster | 556 | 1893 | 73 | 16298 |
| 2nd cluster | 444 | 1540 | 43 | 15411 |
| 3rd cluster | 493 | 6326 | 179 | 50244 |
| QOR | | | | |
| 1st cluster | 547 | 2142 | 83 | 17463 |
| 2nd cluster | 444 | 1597 | 42 | 16616 |
| 3rd cluster | 495 | 6639 | 184 | 51112 |

**Table 1.** DBSCAN and QOR k-means cluster means.

**Chart

Description automatically generated with medium confidenceFigure 4.** Boxplots of clustered DBSCAN anomalies by cluster label. Red rectangles correspond to insets of CO2 and BC that are displayed on the right side of the plot.

To aid in interpretation of these clusters, we extract traffic variables from the TxDOT roadway inventory and assign these values to our clustered anomalies based on nearest neighbor assignment between the logged GPS coordinates of each clustered point and the latitude/longitude coordinates of the inventory’s features.23 We plot these assignments in Figure 5. Panel (a) in Figure 5 contains the overall AADT counts. Panel (b) in Figure 5 shows percentages of trucks in the estimated annual AADT counts. The boxplots illustrate that the 3rd cluster potentially corresponds to trucking activity, while the 1st cluster potentially corresponds to gasoline-powered vehicle activity. The high percentage of trucks in AADT in the 3rd cluster clearly points to trucking activity, while the lower trucking percentage in combination with elevated AADT compared to the 2nd cluster suggest that the 1st cluster could be capturing some light duty gasoline vehicle activity. These assignments, in conjunction with their principal component loading assignments, are consistent with previously published studies of vehicle source apportionment.1,24,25

**Chart, box and whisker chart

Description automatically generatedFigure 5.** Boxplot of traffic attributes corresponding to anomalies in labelled clusters (1- “CO2 Cluster”, 2 – “Transition Cluster”, 3 – “BC/UFP Cluster”). (a) Annual average daily traffic (AADT) by cluster label. (b) Percentages of trucks in the annual average daily traffic counts (AADT % Truck).

**3.3 Detected Anomaly Type by Census Tract**

We evaluate spatial differences in these clustered anomaly types across the city of Houston by tabulating anomaly types for visited census tracts. We report rescaled total number of detected anomalies of a given cluster type (“CO2 Cluster” for “CO2 -rich”, “Transition Cluster”, “BC/UFP Cluster”) divided by the total number of measurements made in that census tract. Normalizing by the total number of measurements in this manner yields the probability of encountering that anomaly in the census tract during the study period, which is from 8 AM to 4 PM during weekdays. Figure S3 displays bar plots showing detected DBSCAN anomaly type by each census tract, while Figures 6 and 7 map the census tracts colored by their CO2 and BC/UFP anomaly detection probabilities.

**Map

Description automatically generatedFigure 6.** Map depicting analyzed census tracts colored by their calculated CO2 anomaly detection probabilities (%). Basemap courtesy of Wikimedia.

**Map

Description automatically generatedFigure 7.** Map depicting analyzed census tracts normalized by their calculated BC/UFP anomaly detection probabilities (%). Basemap courtesy of Wikimedia.

The bar plot and maps illustrate stark spatial divides in distributions of anomaly type. With respect to cluster anomalies, neighborhoods in the western parts of Houston (North Rice, South Rice, South River Oaks) consistently rank higher than neighborhoods in the eastern part of Houston, with neighborhoods surrounding Rice University ranking the highest. The neighborhoods near Rice campus consists of busy thoroughfares that are often congested with traffic from light duty gasoline powered vehicles, especially around rush hour (8 AM). With regards to the BC/UFP clusters, heavily industrialized neighborhoods in the eastern part of Houston (Milby Park, West Galena Park, Manchester, Clinton) are ranked the highest, with the Milby Park census tract exhibiting the highest probability of encountering one of these anomaly types (10.6%) during the study period. Many of these anomaly detection types were found to occur on highway; Figure S7 illustrates the differences in BC/UFP anomaly detection probabilities when highways are included and excluded from the analysis. Even with highways removed from the analysis, neighborhoods in the Eastern part of Houston still rank consistently higher than those neighborhoods in the western part of Houston. The mapped census tracts highlight spatial discrepancies in that some areas are CO2 dominated and others are BC/UFP dominated with respect to probability of anomaly type detection. Table 2 contains probabilities of detecting each anomaly type by census tract which underscores these spatial disparities. For example, the bold, italicized entries in Table 2 indicate a 10x greater chance of encountering a BC/UFP anomaly type in the Manchester census tract compared to the North Rice census tract. These disparities, and the presented evidence suggesting that the BC/UFP anomalies are closely related to heavy duty vehicles, are consistent with previous modeling studies which show how large contributions of air pollution in Houston’s Ship Channel neighborhoods can be attributed to heavy-duty vehicles.26

**Table 2.** Tabulated anomaly detection probability type (“CO2 – rich” = “CO2 %”, “Transition” = “Transition %”, “BC/UFP – rich” = “BC/UFP %”) by census tract.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Census Tract | CO2 % | Transition % | BC/UFP % | Total |
| Bayland Park | 1.7 | 8.6 | 0.8 | 138367 |
| Washington Corridor | 2.8 | 13.3 | 1.9 | 206611 |
| Manchester | 0.8 | 19.6 | ***5.6*** | 97374 |
| East Galena Park | 0.7 | 8.6 | 0.7 | 77046 |
| Milby Park | 1.2 | 16.8 | 10.6 | 110019 |
| Sharpstown | 4.6 | 17.8 | 2.8 | 80560 |
| Sharpstown South | 2.2 | 9.5 | 1.3 | 114595 |
| West Galena Park | 1.5 | 16.5 | 6.0 | 134501 |
| North Spring Branch | 2.1 | 12.0 | 1.0 | 100391 |
| North Rice | 5.8 | 14.4 | ***0.6*** | 263585 |
| Clinton | 1.2 | 20.1 | 4.4 | 185196 |
| West Eastex | 1.1 | 12.8 | 2.5 | 144963 |
| North Heights | 1.4 | 10.4 | 1.4 | 246103 |
| South Rice | 5.0 | 13.4 | 0.6 | 139313 |
| Harrisburg | 1.0 | 16.9 | 4.2 | 127736 |
| Sharpstown North | 3.6 | 18.7 | 1.2 | 98743 |
| Westchase | 3.4 | 12.7 | 1.3 | 68620 |
| South Spring Branch | 2.3 | 13.3 | 2.4 | 78195 |
| South Beltway Central | 0.9 | 16.3 | 2.2 | 311589 |

1. **DISCUSSION AND FUTURE DIRECTIONS**

We discuss the successful development of a new approach to detect plumes in mobile monitoring time series using an anomaly detection algorithm based on DBSCAN. While previous work has implemented DBSCAN in conjunction with deep learning models to analyze satellite PM2.5 measurements27 or used it to define microenvironments in air pollution exposure contexts (e.g. home, work, or restaurant),28 this is the first study (to our knowledge) to incorporate DBSCAN in plume detection efforts. The algorithm offers comparable, if not superior, performance to previously published plume detection techniques for mobile monitoring time series and is justified in analyses warranting a conservative approach. In this work, we use the algorithm to illustrate different emission profiles in census tracts around the city of Houston and showcase disparities. Specifically, we show how BC/UFP anomaly types were 10x more likely to be detected in census tracts in the Eastern part of Houston compared to neighborhoods in the western part of Houston. While it is not definitive that this cluster type represents impacts from heavy duty vehicles, for there is no observational evidence to connect those observations to those vehicle types directly, the results are consistent with previously published studies analyzing emissions from light and heavy-duty vehicles (e.g. Larson et al.1 and references therein), as well as with modeling studies elucidating the large impacts of trucking on PM formation in the Houston Ship channel area.26 It is evident that more investigation is needed into the trucking activity in ship channel neighborhoods.

There are opportunities to improve this algorithm in future work. This algorithm should be evaluated using different external validation sets. Alternative nearest neighbor clustering techniques could be explored; local outlier factors could be used to address situations where DBSCAN does not exhibit great performance.6 Finally, an ensemble approach utilizing both DBSCAN and other clustering techniques could be investigated for improved performance.4,8

ASSOCIATED CONTENT

**Supporting Information**.

Section S1 describes rescaling procedure for census tract comparisons.

Figures S1-S8

Tables S1-S5

AUTHOR INFORMATION

Corresponding Author

\*Rob Griffin (rgriffin@rwu.edu)

Author Contributions

BA1 conceived, wrote, and analyzed the plume detection algorithm with helpful insight from RG2. BA1 wrote the manuscript. RG2 provided helpful edits and suggestions.

Funding Sources

Authors Blake Actkinson and Robert Griffin both received funding from NIEHS grant #R01ES028819-01.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of NIEHS (grant #R01ES028819-01). Additionally, we appreciate the support of Environmental Defense Fund for the collection and provision of the mobile data used to develop this algorithm. Finally, we’d like to thank Dr. Katherine Ensor for her useful suggestions and input.

REFERENCES

(1) Larson, T.; Gould, T.; Riley, E. A.; Austin, E.; Fintzi, J.; Sheppard, L.; Yost, M.; Simpson, C. Ambient Air Quality Measurements from a Continuously Moving Mobile Platform: Estimation of Area-Wide, Fuel-Based, Mobile Source Emission Factors Using Absolute Principal Component Scores. *Atmos. Environ.* **2017**, *152*, 201–211. https://doi.org/10.1016/j.atmosenv.2016.12.037.

(2) Messier, K. P.; Chambliss, S. E.; Gani, S.; Alvarez, R.; Brauer, M.; Choi, J. J.; Hamburg, S. P.; Kerckhoffs, J.; LaFranchi, B.; Lunden, M. M.; Marshall, J. D.; Portier, C. J.; Roy, A.; Szpiro, A. A.; Vermeulen, R. C. H.; Apte, J. S. Mapping Air Pollution with Google Street View Cars: Efficient Approaches with Mobile Monitoring and Land Use Regression. *Environ. Sci. Technol.* **2018**, *52* (21), 12563–12572. https://doi.org/10.1021/acs.est.8b03395.

(3) Hagler, G. S. W.; Lin, M.-Y.; Khlystov, A.; Baldauf, R. W.; Isakov, V.; Faircloth, J.; Jackson, L. E. Field Investigation of Roadside Vegetative and Structural Barrier Impact on Near-Road Ultrafine Particle Concentrations under a Variety of Wind Conditions. *Sci. Total Environ.* **2012**, *419*, 7–15. https://doi.org/10.1016/j.scitotenv.2011.12.002.

(4) Drewnick, F.; Böttger, T.; von der Weiden-Reinmüller, S.-L.; Zorn, S. R.; Klimach, T.; Schneider, J.; Borrmann, S. Design of a Mobile Aerosol Research Laboratory and Data Processing Tools for Effective Stationary and Mobile Field Measurements. *Atmospheric Meas. Tech.* **2012**, *5* (6), 1443–1457. https://doi.org/10.5194/amt-5-1443-2012.

(5) Choi, W.; He, M.; Barbesant, V.; Kozawa, K. H.; Mara, S.; Winer, A. M.; Paulson, S. E. Prevalence of Wide Area Impacts Downwind of Freeways under Pre-Sunrise Stable Atmospheric Conditions. *Atmos. Environ.* **2012**, *62*, 318–327. https://doi.org/10.1016/j.atmosenv.2012.07.084.

(6) Tan, P.-N.; Steinbach, M.; Karpatne, A.; Kumar, V. *Introduction to Data Mining*; Pearson, 2019.

(7) Miller, D. J.; Actkinson, B.; Padilla, L.; Griffin, R. J.; Moore, K.; Lewis, P. G. T.; Gardner-Frolick, R.; Craft, E.; Portier, C. J.; Hamburg, S. P.; Alvarez, R. A. Characterizing Elevated Urban Air Pollutant Spatial Patterns with Mobile Monitoring in Houston, Texas. *Environ. Sci. Technol.* **2020**, *54* (4), 2133–2142. https://doi.org/10.1021/acs.est.9b05523.

(8) Actkinson, B.; Ensor, K.; Griffin, R. J. SIBaR: A New Method for Background Quantification and Removal from Mobile Air Pollution Measurements. *Atmospheric Meas. Tech.* **2021**, *14* (8), 5809–5821. https://doi.org/10.5194/amt-14-5809-2021.

(9) TIGER/Line Shapefile, 2018, county, Harris County, TX, All Roads County-based Shapefile - Data.gov https://catalog.data.gov/dataset/tiger-line-shapefile-2018-county-harris-county-tx-all-roads-county-based-shapefile (accessed 2020 -12 -14).

(10) Tidyverse https://www.tidyverse.org/ (accessed 2022 -03 -31).

(11) Kassambara, A. *Ggpubr: “ggplot2” Based Publication Ready Plots*; 2020.

(12) Kuhn, M. *The Caret Package*.

(13) Hahsler, M.; Piekenbrock, M.; Doran, D. Dbscan: Fast Density-Based Clustering with R. *J. Stat. Softw.* **2019**, *91*, 1–30. https://doi.org/10.18637/jss.v091.i01.

(14) Leaflet for R - Introduction https://rstudio.github.io/leaflet/ (accessed 2022 -03 -31).

(15) Appelhans, T.; Reudenbach, C.; Russell, K.; Darley, J.; plugin), D. M. (Leaflet E.; Busetto, L.; Ranghetti, L.; McBain, M.; Gatscha, S.; plugin), B. H. (FlatGeobuf; Dufour (georaster-layer-for-leaflet), D.; Neuwirth, Y.; Cazelles, K. *Leafem: “leaflet” Extensions for “Mapview”;* 2021.

(16) Pebesma, E.; Bivand, R.; Racine, E.; Sumner, M.; Cook, I.; Keitt, T.; Lovelace, R.; Wickham, H.; Ooms, J.; Müller, K.; Pedersen, T. L.; Baston, D.; Dunnington, D. *Sf: Simple Features for R*; 2022.

(17) Interactive Viewing of Spatial Data in R https://r-spatial.github.io/mapview/ (accessed 2022 -03 -31).

(18) Kratochvil, M. *Scattermore: Scatterplots with More Points*; 2022.

(19) Dowle, M.; Srinivasan, A.; Gorecki, J.; Chirico, M.; Stetsenko, P.; Short, T.; Lianoglou, S.; Antonyan, E.; Bonsch, M.; Parsonage, H.; Ritchie, S.; Ren, K.; Tan, X.; Saporta, R.; Seiskari, O.; Dong, X.; Lang, M.; Iwasaki, W.; Wenchel, S.; Broman, K.; Schmidt, T.; Arenburg, D.; Smith, E.; Cocquemas, F.; Gomez, M.; Chataignon, P.; Blaser, N.; Selivanov, D.; Riabushenko, A.; Lee, C.; Groves, D.; Possenriede, D.; Parages, F.; Toth, D.; Yaramaz-David, M.; Perumal, A.; Sams, J.; Morgan, M.; Quinn, M.; @javrucebo; @marc-outins; Storey, R.; Saraswat, M.; Jacob, M.; Schubmehl, M.; Vaughan, D.; Hocking, T.; Silvestri, L.; Barrett, T.; Hester, J.; Damico, A.; Freundt, S.; Simons, D.; Andrade, E. S. de; Miller, C.; Meldgaard, J. P.; Tlapak, V.; Ushey, K.; Eddelbuettel, D.; Schwen, B. *Data.Table: Extension of “Data.Frame”;* 2021.

(20) Ester, M.; Kriegel, H.-P.; Xu, X. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. 6.

(21) R: The R Project for Statistical Computing https://www.r-project.org/ (accessed 2021 -06 -22).

(22) Revelle, W. *Psych: Procedures for Psychological, Psychometric, and Personality Research*; 2022.

(23) Roadway Inventory https://www.txdot.gov/inside-txdot/division/transportation-planning/roadway-inventory.html (accessed 2022 -03 -04).

(24) Pachon, J. E.; Balachandran, S.; Hu, Y.; Mulholland, J. A.; Darrow, L. A.; Sarnat, J. A.; Tolbert, P. E.; Russell, A. G. Development of Outcome-Based, Multipollutant Mobile Source Indicators. *J. Air Waste Manag. Assoc.* **2012**, *62* (4), 431–442. https://doi.org/10.1080/10473289.2012.656218.

(25) Dallmann, T. R.; Kirchstetter, T. W.; DeMartini, S. J.; Harley, R. A. Quantifying On-Road Emissions from Gasoline-Powered Motor Vehicles: Accounting for the Presence of Medium- and Heavy-Duty Diesel Trucks. *Environ. Sci. Technol.* **2013**, *47* (23), 13873–13881. https://doi.org/10.1021/es402875u.

(26) Zhang, X.; Craft, E.; Zhang, K. Characterizing Spatial Variability of Air Pollution from Vehicle Traffic around the Houston Ship Channel Area. *Atmos. Environ.* **2017**, *161*, 167–175. https://doi.org/10.1016/j.atmosenv.2017.04.032.

(27) Lu, X.; Wang, J.; Yan, Y.; Zhou, L.; Ma, W. Estimating Hourly PM2.5 Concentrations Using Himawari-8 AOD and a DBSCAN-Modified Deep Learning Model over the YRDUA, China. *Atmospheric Pollut. Res.* **2021**, *12* (2), 183–192. https://doi.org/10.1016/j.apr.2020.10.020.

(28) Do, K.; Yu, H.; Velasquez, J.; Grell-Brisk, M.; Smith, H.; Ivey, C. E. A Data-Driven Approach for Characterizing Community Scale Air Pollution Exposure Disparities in Inland Southern California. *J. Aerosol Sci.* **2021**, *152*, 105704. https://doi.org/10.1016/j.jaerosci.2020.105704.