A Polygon-based Methodology for Mining Related Spatial Datasets

Sujing Wang, Chun-Sheng Chen, Vadeerat Rinsurongkawong, Fatih Akdag, Christoph F. Eick
Department of Computer Science, University of Houston
Houston, TX 77204-3010, USA
{sujingwa, lyon19, vadeerat, fatihak, ceick}@cs.uh.edu

ABSTRACT

Polygons can serve an important role in the analysis of geo-referenced data as they provide a natural representation for particular types of spatial objects and in that they can be used as models for spatial clusters. This paper claims that polygon analysis is particularly useful for mining related, spatial datasets. A novel methodology for clustering polygons that have been extracted from different spatial datasets is proposed which consists of a meta clustering module that clusters polygons and a summary generation module that creates a final clustering from a polygonal meta clustering based on user preferences. Moreover, a density-based polygon clustering algorithm is introduced. Our methodology is evaluated in a real-world case study involving ozone pollution in Texas; it was able to reveal interesting relationships between different ozone hotspots and interesting associations between ozone hotspots and other meteorological variables.

Keywords

spatial data mining, polygon clustering algorithms, mining related datasets, polygon analysis, polygon distance functions.

1. INTRODUCTION

Tools that visualize and analyze geo-referenced datasets have gained importance in the last decade, as can be witnessed by the increased popularity of products, such as Google Earth, Microsoft Virtual Earth and ArcGIS. Polygons play an important role in the analysis of geo-referenced data as they provide a natural representation of geographical objects, such as countries, and in that they can be used for the modeling of spatial events, such as air pollution. Moreover, polygons can serve as models for spatial clusters and can model nested and overlapping clusters. Finally, polygons have been studied thoroughly in geometry and they are therefore mathematically well understood; moreover, powerful software libraries are available to manipulate and to analyze and quantify relationships between polygons. Spatial extensions of popular database systems, such as ORACLE and Microsoft SQL Server 2008, support polygon search and polygon manipulation in extended versions of SQL. Surprisingly, past and current data mining research has mostly ignored the capabilities polygon analysis has to offer.

In general, as we will argue in the remainder of the paper, polygon analysis is particularly useful to mine relationships between multiple, related datasets, as it provides a useful tool to analyze discrepancies, progression, change, and emergent events. This work centers on clustering polygons that have been extracted from multiple, related datasets. In particular, a new methodology to mine related, spatial datasets is introduced that consists of a meta clustering module that clusters polygons and a user driven summary generation module that creates a final clustering and other summaries from a polygonal meta clustering. The paper's main contributions include:

- A novel polygon-based methodology for analyzing related, spatial datasets is introduced.
- In contrast to past research, our approach puts a lot of emphasis on the analysis of overlapping polygons that originate from different datasets. Novel distance functions to assess the similarity of overlapping polygons are introduced for this purpose.
- A density-based polygonal meta-clustering algorithm is introduced.
- Summary generation algorithms that create the final clustering from meta clusters are proposed. The algorithms rely on a plug-in fitness function to capture user preferences, which is maximized when generating the final cluster.
- The proposed framework is evaluated in a challenging real-world case study involving ozone pollution in the Houston Metropolitan area.

The paper is organized as follows. Section 2 discusses related work. Section 3 introduces distance functions and clustering algorithms for polygons. Section 4 introduces algorithms that generate a final clustering from polygonal meta clusters. Finally, Section 5 evaluates the proposed methodology using ozone pollution case studies and Section 6 summarizes our findings.

2. RELATED WORK

In [1], Joshi et al. propose a DBSCAN-style clustering algorithm for polygons; the algorithm works by replacing point objects in the original DBSCAN algorithm with the polygon objects. In [2], Joshi et al. introduce a dissimilarity function for clustering non-overlapping polygons that considers both spatial and non-spatial attributes. Buchin et al. [12] propose a polygonal time algorithm to compute the Fréchet distance between two polygons. Several papers [21], [6] propose algorithms to compute the Hausdorff distance between polygons. Sander et al. [7] propose GDBSCAN, an algorithm generalizing DBSCAN in two directions: First, generic object neighborhoods are supported instead of distance-based neighborhoods. Second, it proposes other, more complicated measures to define the density of the neighborhood of an object instead of simply counting the number of objects within a given radius of a query point.

Zeng et al. [3] propose a meta clustering approach to obtain better clustering results by comparing and selectively combining results of different clustering techniques. In [4] Gionis et al. present clustering aggregation algorithms which produce a single clustering that minimizes the total number of disagreements among input clusterings. The proposed algorithms apply the concept of correlation clustering [5]. Caruana et al. [6.] propose a mean to automatically create many diversity clusterings and then measures

the distance between the generated clusterings. Next, the hierarchical meta clusters are created. Finally an interactive interface is provided to allow users to choose the most appropriate clustering from meta clusters based on their preferences. In general, [3], [4], and [6] perform meta clustering on a single dataset, whereas our proposed methodology uses meta clustering to analyze relationship between polygons from multiple related datasets.

Our work also relates to correspondence clustering, coupled clustering, and co-clustering which all mine related datasets. Coupled clustering [3] is introduced to discover relationships between two textual datasets by partitioning the datasets into corresponding clusters where each cluster in one dataset is matched with its counterpart in the other dataset. Co-clustering has been successfully used for applications in text mining [4], market-basket data analysis, and bioinformatics [5]. In general, co-clustering clusters two datasets with different schemas by rearranging the datasets. The objects in two datasets are represented as rows and columns of a dataset. Then co-clustering partitions rows and columns of the data matrix and creates clusters which are subsets of the original matrix. Correspondence clustering [9] is introduced by Rinsurongkawong et al. to cluster two or more spatial datasets by maximizing cluster interestingness and correspondence between clusters. Cluster interestingness and correspondence interestingness are captured in a plug-in fitness functions and prototype-based clustering algorithms are proposed that cluster multiple datasets in parallel. In conclusion, coupled clustering [3] and co-clustering [4], [5] are not designed for spatial data and they cluster point objects using traditional clustering algorithms. The techniques introduced in correspondence clustering [9] are applicable to point objects in the spatial space whereas this paper focuses on clustering spatial clusters that originate from different, related datasets that are approximated using polygons.

3. DISTANCE FUNCTIONS AND CLUSTERING ALGORITHM FOR POLYGONS

This paper introduces a methodology that uses polygon analysis to mine related datasets, which consists of 3 steps:

1. Collect/Generate polygonal clusters for multiple related datasets
2. Meta cluster polygonal clusters
3. Extract interesting patterns / create summaries from polygonal meta clusters

As far as polygon generation is concerned, our work uses a contouring algorithm called DCONTOUR [14] to generate polygons from continuous density functions or interpolation functions as described in [13], [14]. Moreover, if spatial cluster extensions are given instead, Characteristic shapes [15] and Alpha shapes [16] can be used to wrap polygons around objects that belong to a particular spatial cluster. Both Characteristic shapes and Alpha shapes algorithms create the Delaunay triangulation of the point set and reduce it to a non-convex hull. Polygon generation (Step 1) will not be discussed any further in this paper; this section focuses on Step 2.

3.1 Distance Functions for Polygons

One unique characteristic of our work is that we have to cope with overlapping polygons; past work on polygonal clustering usually assumes that polygons do not overlap and most uses the Hausdorff distance [11] to assess polygon similarity. However, we believe that considering polygon overlap is of critical importance for polygonal clustering of related datasets. Therefore, in addition to the Hausdorff distance, our work proposes two novel distance functions called overlay and hybrid distance functions.

We define a polygon A as a sequence of points A= p_1,…, p_n, with point p_1 being connected to the point p_n to close the polygon. Moreover, we assume that boundary of the polygon does not cross itself and polygons can have holes inside. Throughout the paper we use the term polygon to refer to such polygons.

3.1.1 Distance Functions for Polygons

3.1.1.1 Hausdorff Distance

The Hausdorff distance measures the distance between two point sets. It is the maximum distance of a point in any set to the nearest point in the other set. Using the same notation as [11], let A and B be two point sets, the Hausdorff distance $D_{Hausdorff}(A,B)$ for the two sets is defined as:

$$D_{Hausdorff}(A,B) = \max\{max_{a\in A} min_{b\in B} d(A,B), max_{b\in B} min_{a\in A} d(A,B)\}$$

where *d(A,B)* is the Euclidean distance between point *A* and point *B*.

In order to use the Hausdorff distance for polygons, we firstly have to determine how to associate a point set with a polygon. One straight forward choice is to define this point set as the points that lie on the boundary of a polygon. However, computing the distance between point sets that consist of unlimited number of points is considerably expensive. An algorithm that solves this problem for trajectories has been proposed by [20] and the same technique can be applied to polygons.

3.1.1.2 Overlay Distance

The overlay distance measures the distance between two polygons based on their degree of overlap. The overlay distance $D_{Overlay}(A,B)$ between polygons A and B is defined as:

$$D_{Overlay}(A,B) = 1 - \frac{area(Intersection(A,B))}{area(Union(A,B))}$$

where the function area(X) returns the area a polygon X covers. Basically, the overlay distance is the quotient of the size of the intersection of the two polygons over the size of the union of the two polygons. The overlay distance is 1 for pairs of polygons that do not overlap at all.

3.1.1.3 Hybrid Distance

The hybrid distance function uses a linear combination of the Hausdorff distance and the overlay distance. Because the overlay distance between two disjoint polygons is always 1, regardless of the actual location in space, additionally using the Hausdorff distance provides more precise approximations of the distance between polygons. The hybrid distance function is defined as:

$$D_{Hybrid}(A,B) = \left(w \times D_{Overlay}(A,B)\right) + \left((1-w) \times D_{Hausdorff}(A,B)\right)$$

where w is the weight associated with each distance function ($1 \geq w \geq 0$). Due to the fact that our goal is spatial clustering and we are interested in obtaining meta clusters whose polygons overlap a lot, typically much more weight will be associated with the overlay distance function.

3.2 The POLY_SNN Algorithm

The SNN (Shared Nearest Neighbors) algorithm [8] is a density-based clustering algorithm which assesses the similarity between two points using the number of nearest neighbors that they share. SNN clusters data as DBSCAN does, except that the number of shared neighbors is used to access the similarity instead of Euclidean distance.

Similar to DBSCAN, SNN is able to find clusters of different sizes and shapes, and can cope with noise in the dataset. However, SNN copes better with high dimensional data and responds better to datasets with varying densities.

In SNN, similarity between two points p_1 and p_2 is the number of points they share among their K nearest neighbors as follows:

$$similarity(p1, p2) = size\ of\ (NN(p1) \cap NN(p2))$$

where $NN(p_i)$ is the K nearest neighbors of point pi.

SNN density of a point p is defined as the sum of the similarities between point p and its K nearest neighbors as follows:

$$density(p) = \sum_{i=1}^{k} similarity\ (p, pi)$$

where p_i is point p's i^{th} nearest neighbor.

After assessing the SNN density of each point, SNN algorithm finds the core points (points with high SNN density) and forms the clusters around the core points like DBSCAN. If two core points are similar to each other, then they are placed in the same cluster. All non-core points which are not similar to any core point are identified as noise points. All non-noise and non-core points are assigned to the cluster of the nearest core point.

When using SNN to cluster polygons, we first calculate the distances between all pairs of polygons using the distance functions discussed in section 2. Next, we identify the K nearest neighbors for each polygon. SNN calculates the SNN density of each polygon using the K nearest neighbors list and clusters the polygons around core polygons using the DBSCAN like algorithm described above.

4. CREATING FINAL CLUSTERINGS FROM POLYGONAL META CLUSTERS

Several forms of summaries can be generated from polygonal meta clusters:

1. Signatures for meta clusters that summarize what characteristics objects in the same meta clusters share.
2. Discrepancy mining can be used to create knowledge of how the clusters in a particular meta cluster differ from the clusters in another meta cluster.
3. Final clusterings can be created from a meta clustering.

Section 5 gives some examples of summaries with respect to characteristics and discrepancies of ozone hotspot polygons. The remainder of this section will discuss how to create a "good" final clustering from a set of meta clusters.

Although clustering has been studied for more than 40 years, its objectives and how to evaluate different clustering results is still subject to a lot of controversy; moreover, current research, particularly most ensemble clustering research is still relying on the misconception that a universal, optimal clustering of a dataset exists. However, in general, domain experts seek for clusters based on their domain-driven notion of "interestingness" which usually differs from generic characteristics used by clustering algorithms; moreover, for a given dataset there usually are many plausible clusterings whose value really has to be determined by the domain expert. Finally, even for the same domain expert multiple clusterings, e.g. clusterings at different levels of granularity, are of value. A key idea of this work is to collect a large number of frequently overlapping clusters which are organized in form of meta-clusters; a final clustering is then created from those meta clusters based on a user's notion of interestingness.

To reflect what was discussed in the previous paragraph, we assume that our final cluster generation algorithms provide plug-in fitness functions that capture a domain expert's notion of interestingness which are maximized when generating the final clustering. Meta clustering provides an alternative approach to the traditional ensemble clustering approach by creating a more structured input for generating a final clustering, also reducing algorithm complexity by restricting choices. In this section, we propose algorithms that create a final clustering by selecting at most one cluster from each meta cluster. Moreover, due to the fact that polygons originating from different datasets typically overlap a lot, we provide an option for the user to restrict cluster overlap in the final clustering. More formally, we develop algorithms that create a final clustering from a meta clustering by solving the following optimization problem:

Inputs:

1. A meta clustering $M=\{X_1, \ldots, X_k\}$ —at most one object will be selected from each meta cluster X_i (i=1,...k).
2. The user provides her own individual cluster reward function $Reward_U$ whose values are in $[0,\infty)$.
3. A reward threshold θ_U—low reward clusters are not included in the final clustering.
4. A cluster distance threshold θ_d which expresses how much cluster overlap/coverage she likes to tolerate.
5. A cluster distance function *dist*.

Find $Z \subseteq X_1 \cup \ldots \cup X_k$ that maximizes:

$$q(Z) = \sum\nolimits_{c \epsilon Z} reward_U(c)$$

subject to:

1. $\forall\ x \in Z\ \forall x' \in Z\ \ (x \neq x' \Rightarrow Dist(x,x') > \theta_d)$
2. $\forall\ x \in Z\ (Reward_U(x) > \theta_U)$
3. $\forall\ x \in Z\ \forall x' \in Z\ \ ((x \in X_i \wedge x' \in X_k \wedge x \neq x') \Rightarrow i \neq k)$

The goal is to maximize the sum of the rewards of clusters that have been selected from meta clusters. Constraint 1 prevents that two clusters that are too close to each other are both included in the final clustering. Constraint 3 makes sure that at most one cluster from each meta cluster is selected.

Assuming that we have n meta clusters each containing an average of m clusters, there are roughly $(m+1)^n$ final clusterings; for each meta cluster we can either select one of its clusters for inclusion or we might decide not to take any cluster of the meta cluster due to violations of constraints 1 and 2. Constraint 2 is easy to handle by removing clusters below threshold from the meta clusters prior to running the final cluster generation algorithm.

Many different algorithms can be developed to solve this optimization problem, three of which we are currently investigating:

- A greedy algorithm: A greedy algorithm that always selects the cluster with the highest reward from the unprocessed meta clusters whose inclusion in the final clustering does not violate constraints 1 and 2. If there are no such clusters left, no more clusters will be added from the remaining meta clusters to the final clustering.
- An anytime backtracking algorithm: An anytime backtracking algorithm that explores the choices in descending order of cluster rewards; every time a new final clustering is obtained, the best solution found so far is potentially updated. If runtime expires, the algorithm reports the best solution found so far.
- An evolutionary computing algorithm that relies integer chromosomal representations; e.g. (1,2,3,0) represents a solution where cluster 1 is selected from meta clustering 1, cluster 2 from meta cluster 2,…, and no cluster is selected from meta cluster 4. Traditional mutation and crossover operators are used to create new solutions, and a simple repair approach is used to deal with violations of constraint 1.

The greedy algorithm is very fast ($O(m \times n)$) but far from optimal, the backtracking algorithms explore the complete search space ($O(m^n)$) and—if not stopped earlier—finds the optimal solution if n and m are not very large; however, the anytime approach can be used for large values of m and n. Finally, the evolutionary computing algorithm covers a middle ground, providing acceptable solutions that are found in medium runtime.

5. EXPERIMENTAL EVALUATION

5. 1 The Ozone Dataset

Recently, it has been reported by the American Lung Association [24] that Houston Metropolitan area is the 7th worst ozone zone in the US. The Texas Commission on Environmental Quality (TCEQ) is a state agency responsible for environmental issues including the monitoring of environmental pollution in the Texas. TCEQ collects hourly ozone concentration data for metropolitan areas across the state and publishes the data on its website [22]. TCEQ uses a network of 44 ozone-monitoring stations in the Houston-Galveston area. The area covers the geographical region within [-95.8070, -94.7870] longitude and [29.0108, 30.7440] latitude. We downloaded the hourly ozone concentration data from TCEQ's website between the timeframe of April 1, 2009 at 0:00 to November 30, 2009 at 23:00. In addition to the ozone concentrations, we also downloaded the meteorology data including average wind speed, average solar radiation, and average outdoor temperature for the same time slots as the ozone measurements.

Basically, we create polygons that capture ozone hotspots for particular time slot; for each time slot we obtain a set of polygons. In particular the polygons were generated as follows: First, we download the ozone concentration monitored by 44 monitoring sites from TCEQ's website. Next, a standard Kriging interpolation method [25] is used to compute the ozone concentrations on 20×27 grids that cover the Houston metropolitan area. Finally, we feed the interpolation function into the DCONTOUR algorithm with a defined threshold to create sets of polygons, describing polygon hotspots—areas in the spatial dataset whose ozone concentration is above the input threshold. Two polygon datasets are created by using two different density thresholds as inputs for DCONTOUR algorithm. The use of the density threshold 180 creates 255 polygons. These polygons represent areas where the average one hour ozone concentration is above 80 ppb (parts per billion). The density threshold 200 generates 162 polygons that have one hour ozone concentration more than 90 ppb. The current EAP ozone standard is based on an eight-hour average measurement. In order to meet the standard, the eight-hour average ozone concentration has to be less than 0.08ppm (80 ppb). Therefore, we can consider the polygons that we created are areas where the ozone level exceeds the EPA standard in that hour. Our experiments were conducted using the polygon dataset generated by DCONTOUR with threshold equal to 200.

We evaluate our methodology in two case studies. The goal of the first case study is to verify that our new polygon distance functions and clustering algorithms for geospatial polygons can effectively cluster overlapped spatial polygons originated from different related datasets. By analyzing additional meteorological attributes such as outdoor temperature, solar radiation, wind speed and time of day associated with polygons, we can characterize each cluster and identify interesting patterns associated with these hotspots. To accomplish this goal we cluster all polygons at all time slots for certain threshold as a single pool of clusters.

In the second case study, we are interested to generate final clusters that capture a domain expert's notation of interestingness by plugging in different reward functions, e.g., possible maximum range of ozone pollution represented by area of polygons. In order to summarize final clusters generated by our model, we also compute the statistical results of ozone pollution control variables.

5.2 Case Study 1: Anaylizing Meta Clusters of Ozone Hotspots

An ozone polygon is a hotspot area that has ozone concentration above a certain threshold. In the first case study, we apply the POLY_SNN clustering algorithm to cluster all the polygons in the ozone dataset in order to find clusters of hotspots.

Figure 1 displays the meta-clustering result of 30 clusters found by POLY_SNN using the hybrid distance function and the number of nearest neighbors k set to 5. The X and Y coordinates are the latitude and longitude of each polygon. The dataset consists of 162 polygons created by DCONTOUR using density threshold equal to 200 (90 ppb). Out of 162 polygons, 30% of polygons in the dataset are considered outliers by POLY_SNN. Polygons marked by the same color belong to the same cluster.

Ozone formation is a complicated chemical reaction. There are several control factors involved:

1. Sunlight measured by solar radiation is needed to produce ozone.
2. High outdoor temperatures cause the ozone formation reaction to speed up.
3. Wind transports ozone pollution from the source point.

4. Time of Day: ozone levels can continue to rise all day long on a clear day, and then decrease after sunset.

In general, by analyzing the meteorological characteristics of polygons domain experts may find some interesting phenomena that could lead to further scientific investigation. Therefore, we also compute some statistics of 4 meteorological variables involved in ozone formation. Table 1 lists the statistical results of four control factors discussed above associated with the meta clustering in Figure 1. As expected, meta clustering shown in Figure 1 representing one hour ozone concentration higher than 90 ppd is characterized by high outdoor temperature (average of 90.6 and standard deviation of 5.3) and strong solar radiation(average of 0.80 and standard deviation of 0.36), which usually happens between 1 pm to 4 pm each day. The wind speed affects the range of ozone pollution represented by the size of polygons. Since the standard deviation of the wind speed (1.90) compared with the average wind speed (6.05) is nontrivial, the variation of the size of the polygons is significant in Figure 1.

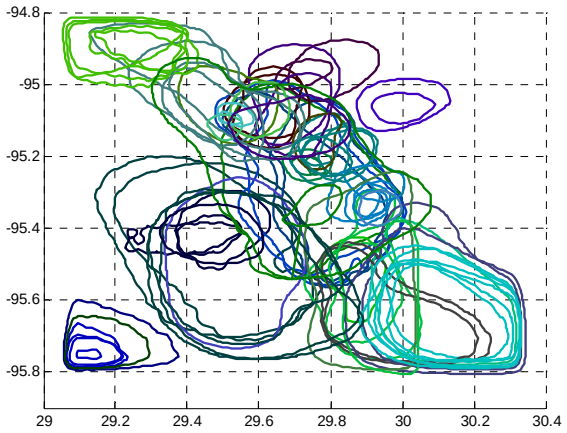

Figure 1. Meta clustering generated by POLY_SNN using the Hybrid distance function.

Table 1. The statistical results of meteorological variables for meta clustering shown in Figure 1

	Mean	Std	Max	Min
Temperature	90.6	5.3	102.8	78.6
Solar Radiation	0.8	0.36	1.4	0.03
Wind Speed	6.1	1.9	15.7	0.3
Time of Day	2:30 pm	1.8	10 am	8 pm

It is hard to visualize clustering results as polygons overlap a lot as can be seen in Figure 1. Figure 2 and Figure 3 give a picture of eight polygonal meta clusters from Figure 1. As expected, the hybrid distance function that employs both overlay distance function and Hausdorff distance function creates clusters of polygons that are similar in shape, size and location. Particularly, since we give more weights to the overlay distance function, the clusters in Figure 2 and Figure 3 are highly overlapped. The clustering results prove that our POLY_SNN clustering algorithm in conjunction with the hybrid distance function can effectively find clusters of overlapping polygons similar in size, shape and location.

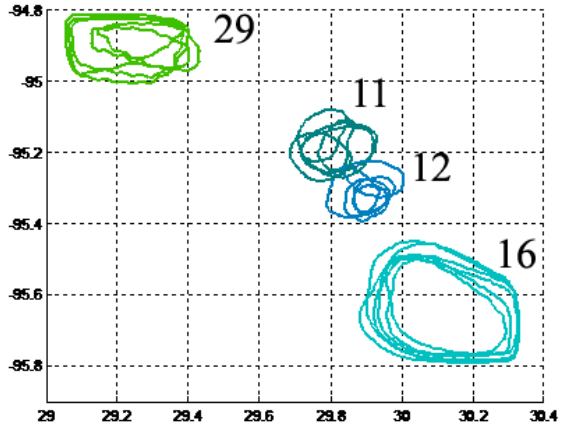

Figure 2. Visualization of 4 meta clusters (ID: 11, 12, 16, and 29) discovered by POLY-SNN in Figure 1.

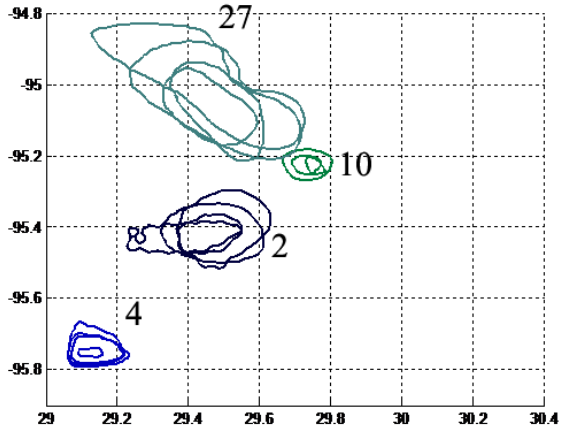

Figure 3. Visualization of 4 meta clusters (ID: 2, 4, 10, and 27) discovered by POLY_SNN in Figure 1.

Table 2 and Table 3 list the mean and standard deviation of outdoor temperature, solar radiation, wind speed and time of day associated with eight meta clusters in Figure 2 and Figure 3. The solar radiation information related to cluster 2 and 4 are not available from TCEQ's website. Certainly, ozone formation is far more complicated than only considering those four control factors. Our polygon-based methodology has the capability of handling more non-spatial attributes.

Based on Table 2, we can see that ozone polygons in clusters 11 and 12 are characterized by very high outdoor temperature (98.83 and 99.10) compared with entire meta clustering (90.6) and strong solar radiation (0.90 and 0.86) compared with entire meta clustering (0.8). The wind speed of cluster 11 and cluster 12 (5.16 and 4.86) are slow compared with entire meta clustering (6.1) so that the average size of the polygons in cluster 11 and cluster 12 are relatively small compared with all other polygons shown in Figure 1. Also, Clusters 11 and 12 are captured around 2 pm. The statistical results associated with Cluster 16 are very close to the entire meta clustering in Table 1.

Table 2. The statistical results of meteorological variables for 4 meta clusters shown in Figure 2

Meta Cluster Id		11	12	16	29
Temperature	mean	98.83	99.10	90.94	85.48
	std	1.05	2.89	4.26	1.04
Solar Radiation	mean	0.90	0.86	0.70	0.69
	std	0.34	0.0.28	0.28	0.46
Wind Speed	mean	5.16	4.86	5.84	8.34
	std	0.46	0.97	0.93	2.58
Time of Day	mean	2 pm	2 pm	3 pm	12 pm
	std	0.88	1.62	1.63	1.92

Table 3. The Statistical results of meteorological variables for 4 meta clusters shown in Figure 3

Meta Cluster Id		2	4	10	27
Outdoor Temperature	Mean	83.41	88.51	85.95	92.3
	Std	3.81	1.61	2.06	2.86
Solar Radiation	Mean	N/a	n/a	0.65	0.6155
	Std	N/a	n/a	0	0.27
Wind Speed	Mean	6.84	6.15	4.8	6.51
	Std	1.04	0.52	0.79	0.51
Time of Day	Mean	2 pm	1 pm	4 pm	3 pm
	Std	1.70	0.86	0.81	0.83

Based on Table 3, cluster 10 has lower outdoor temperature (85.95) compared with entire meta clustering (90.6), lower solar radiation (0.65) compared with entire meta clustering (0.80) and lower wind speed (4.8) compared with entire meta clustering (6.05). The average time of day for cluster 4 is about 4 pm. All those 4 lower meteorological values contribute to smaller polygon sizes inside cluster 4 shown in Figure 3.

5.3 Case Study 2: Final Cluster Generation

The greedy algorithm introduced in section 4 is used to generate the final cluster from polygonal meta clusters shown in Figure 1. We use several reward functions to capture different notations of interestingness of domain experts. The final cluster generated by our model can be used to summarize what characteristics ozone polygons in the same meta clusters share.

The domain experts are usually interested in recognizing the possible maximal range of ozone pollution. The range of ozone pollution represented by polygon area in our model is selected as the first cluster reward function $Reward_U$. By selecting different reward threshold and distance threshold, different final clusters could be generated. Figure 4 shows one final cluster using reward threshold 0.04 and Hybrid distance threshold 0.5. There are 5 polygons in the final cluster. A small polygon inside the big dark green polygon is a hole inside the polygon. Our framework allows for polygons with holes inside. Those 5 polygons in Figure 4 clearly capture the dominant ozone hotspots in Houston-Galveston area found in Figure 1. Table 4 shows statistical results of meteorological variables of final cluster showed in Figure 4. Since the standard deviations of these four variables are relatively small for each polygon, we did not discuss the standard deviation in this section. Based on Table 4, Polygon 21, 80 and 150 covers larger area with higher outdoor temperature, high wind speed and strong solar radiation compared with polygon 12 and 125. Polygon 150 is interesting because it has a hole inside. Further analysis could be done to help understand the formation of hole inside polygons.

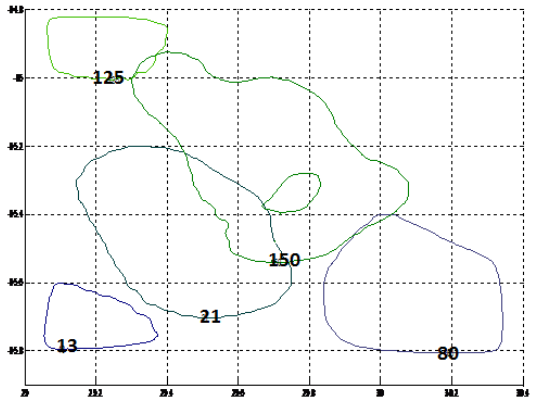

Figure 4. Final cluster for area of polygon reward threshold 0.04 and Hybrid distance threshold 0.5.

Table 4. The mean of meteorological variables for final cluster shown in Figure 4

Polygon ID	13	21	80	125	150
Outdoor Temperature	79.0	86.35	89.10	84.10	88.87
Solar Radiation	N/A	1.33	1.17	0.13	1.10
Wind Speed	4.50	6.10	6.20	4.90	5.39
Time of Day	6 pm	1 pm	2 pm	2 pm	12 pm

The reciprocal of the area of each polygon is used as the second reward function for smaller granularity which may be useful to identify the ozone pollution point source and enable the domain experts to analyze patterns at different levels of granularity. By decreasing either the reward threshold or the distance threshold, we are able to get different final clusters. Figure 5 shows the final clusters with reward threshold set to 10 and distance threshold set to 0.45. Table 5 lists statistical results of four meteorological variables of all polygons in the final cluster shown in Figure 5. Some of the values are not available in the original ozone pollution datasets downloaded from TCEQ website [22]. All of those polygons with relative smaller size shown in Figure 5 occur either before 1 pm or after 4 pm. According to Table 1, the average time of entire meta clustering shown in Figure 1 is 2:30 pm with a standard deviation of 1.8. The time slot from 1 pm to 4 pm everyday is definitely a hotspot for ozone formation which could change the range and the concentration density of ozone pollution

significantly. More analysis should be done specially for this time slot.

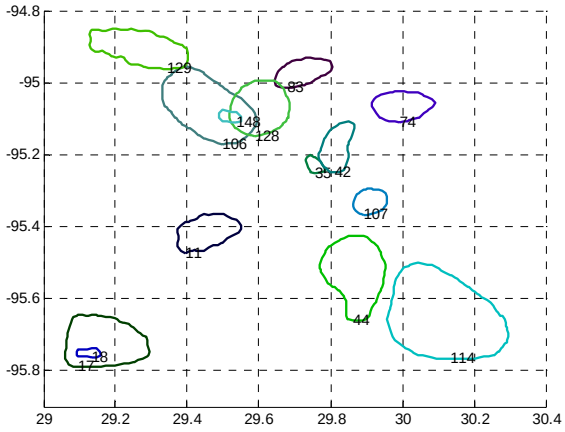

Figure 5. Final cluster for the reciprocal of area reward threshold 10 and Hybrid distance threshold 0.45.

Table 5. The mean of meteorological variables for final cluster shown in Figure 4

Polygon ID	Outdoor temperature	Solar radiation	Wind speed	Time of day
11	81.4	N/A	6.3	4 pm
17	88.2	N/A	6.0	3 pm
18	N/A	N/A	N/A	4 pm
35	86.3	N/A	6.2	5 pm
42	N/A	N/A	N/A	1 pm
44	N/A	N/A	N/A	3 pm
74	N/A	N/A	N/A	4 pm
83	N/A	N/A	5.9	10 am
106	93.5	0.18	5.9	4 pm
107	94.4	1.21	4.6	11 am
114	94.6	0.63	5.8	4 pm
128	86.4	0.13	5.4	5 pm
129	86.2	1.09	8.8	10 am
148	N/a	N/A	N/A	N/A

The outdoor temperatures, wind speed and solar radiation also play a very important role in ozone pollution. We use average outdoor temperature associated with each polygon as the third reward function in our model. Figure 6 shows one final cluster with average outdoor temperature threshold set to 90 and distance threshold set to 0.55. The statistics results of meteorological variables are summarized in Table 6. Obviously, all the polygons with high temperatures occur during 2 pm to 4 pm. The lower the wind speed, the smaller the area of the polygon. For example, polygon 67 has the lowest wind speed of 4.1 compared with all the other four polygons in Figure 6.

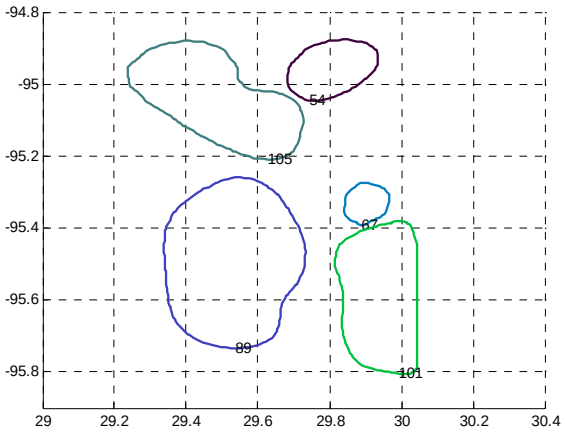

Figure 6. Final cluster for polygon average temperature reward threshold 90 and Hybrid distance threshold 0.55.

Table 6. The mean of meteorological variables of final cluster shown in Figure 5

Polygon ID	54	67	89	101	105
Outdoor Temperature	100.3	102.8	92.4	99.4	94.5
Solar Radiation	N/A	0.96	0.91	0.70	0.72
Wind Speed	6.0	4.1	8.533	8.2	6.04
Time of day	2 pm	3 pm	3 pm	4 pm	3 pm

6. CONCLUSION

This paper claims that polygon analysis is particularly useful for mining multiple, related spatial datasets. In particular, a novel methodology for clustering polygons that have been extracted from multiple, spatial datasets is proposed which consists of a meta-clustering module that clusters the obtained polygons and a summary generation module that extracts patterns and creates summaries from a polygonal meta clustering. In general, this work has the capability to cluster overlapping polygons and use novel distance functions to assess the similarity between polygons which have been proposed for this purpose. Moreover, a density-based polygonal clustering algorithm called POLY_SNN is proposed by extending SSN. Finally, three algorithms for generating a final clustering from a given meta clustering based on user preferences were discussed. To the best of our knowledge, this is the first paper that proposes a comprehensive methodology that relies on polygon analysis to mine related spatial datasets.

Our methodology is evaluated in a real-world case study involving ozone pollution in the Houston Metropolitan area. It is able to reveal interesting relationships between different ozone hotspots and interesting associations between ozone hotspots and other variables.

7. REFERENCES

[1] Joshi, D., Samal, A. K., Soh, L.K., "Density-based clustering of polygons," in Proc. of IEEE Symposium on Computational Intelligence and Data Mining, 2009.

[2] Joshi, D., Samal, A. K., Soh, L.K., "A dissimilarity function for clustering geospatial polygons," in Proc. of the ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2009.

[3] Zeng, Y., Tang, J., Garcia-Frias, J., Gao, R.G., "An adaptive meta-clustering approach: combining the information from different clustering results," in Proc. of IEEE Computer Society Conference on Bioinformatics, 2002.

[4] Gionis, A., Mannila, H., Tsaparas, P., "Clustering aggregation," in Proc. of the International Conference on Data Engineering, 2005.

[5] Bansal, N., Blum, A., Chawla, S., "Correlation clustering," in Proc. of Symposium on Foundations of Computer Science, 2002.

[6] Caruana, R., Elhawary, M., Nguyen, N., Smith, C., "Meta clustering," in Proc of IEEE International Conference on Data Mining, 2006.

[7] Sander J., Ester M., Kriegel H.-P., Xu X., "Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and its Applications," Data Mining and Knowledge Discovery, Vol. 2, No. 2, pp. 169-194, 1998.

[8] Ertoz, L., Steinback, M., Kumar, V., "Finding clusters of different sizes, shapes, and density in noisy, high dimensional data," in: Proc. of SIAM International Conference on Data Mining, 2003.

[9] Rinsurongkawong, V., Eick, C.F., "Correspondence clustering: an approach to cluster multiple related datasets," in: Proc. of Asia-Pacific Conference on Knowledge Discovery and Data Mining, 2010.

[10] Eick, C.F., Parmar, R., Ding, W., Stepinki, T., Nicot, J.P., "Finding regional co-location patterns for sets of continuous variables in spatial datasets," in Proc. of ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2008.

[11] Zhang, Z., Huang, K., Tan, T., "Comparison of similarity measures for trajectory clustering in outdoor surveillance scenes," in Proc. of International Conference on Pattern Recognition, 2006.

[12] Buchin, K., Buchin, M., Wenk, C., "Computing the Fréchet distance between simple polygons in polynomial time," In Proc. of Symposium on Computational Geometry, 2006.

[13] Rinsurongkawong, V. Chen, C.S., Eick, C. F., Twa, M., "Analyzing change in spatial data by utilizing polygon models," in Proc. of International Conference on Computing for Geospatial Research & Application, 2010.

[14] Chen, C.S., Rinsurongkawong, V., Eick, C.F., Twa, M., "Change analysis in spatial data by combining contouring algorithms with supervised density functions, in Proc. Of Asia-Pacific Conference on Knowledge Discovery and Data Mining, 2009.

[15] Duckham, M., Kulik, L., Worboys, M., Galton, A., "Efficient generation of simple polygons for characterizing the shape of a set of points in the plane," Pattern Recognition. 41, pp. 3224-3236, 2008.

[16] Edelsbrunner, H., Kirkpatrick, D. G., Seidel, R., "On the shape of a set of points in the plane," IEEE Transactions on Information Theory, vol. IT–29, no. 4, pp. 551-558, 1983.

[17] Marx, Z., Dagan, I., Buhmann, J.M., Shamir, E., "Coupled clustering: a method for detecting structural correspondence," Journal of Machine Learning Research, pp. 747-780, 2002.

[18] Dhillon, I.S., "Co-clustering documents and words using bipartite spectral graph partitioning," in: Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2001.

[19] Cheng, Y., Church, C.M., "Biclustering of Expression Data," in: Proc. of International Conference on Intelligent Systems for Molecular Biology, 2000.

[20] Hangouet, J., "Computing of the Hausdorff distance between plane vector polylines," in Proc. of Symposium on Computer-Assisted Cartography, 1995.

[21] Atallah M.J., Ribeiro, C.C., Lifschitz, S., "Computing some distance functions between polygons," Pattern Recognition, Vol. 24, Issue 8, pp. 775-781, 1991.

[22] Texas Commission on Environmental Quality, http://www.tceq.state.tx.us

[23] MacQueen, J.B., "Some methods for classification and analysis of multivariate observations," in Proc. Of Berkeley Symposium on Mathematical Statistics and Probability, 1967.

American Lung Association. http://www.lungusa.org/

[24] Cressie, N., Statistics for spatial data. New York: Wiley, 1993.