Analyzing the Composition of Cities Using Spatial Clustering

Zechun Cao
Dept. of Computer Science
University of Houston
Houston, TX 77004

Sujing Wang
Dept. of Computer Science
University of Houston
Houston, TX 77004

Germain Forestier
University of Haute Alsace
MIPS-EA 2332
Mulhouse, France

Anne Puissant
University of Strasbourg
LIVE -ERL 7230, Strasbourg, France

Christoph F. Eick
Dept. of Computer Science
University of Houston, Houston, TX 77004

ABSTRACT

Cities all around the world are in constant evolution due to numerous factors, such as fast urbanization and new ways of communication and transportation. Since understanding the composition of cities is the key to intelligent urbanization, there is a growing need to develop urban computing and analysis tools to guide the orderly development of cities, as well as to enhance their smooth and beneficiary evolution. This paper presents a spatial clustering approach to discover interesting regions and regions which serve different functions in cities. Spatial clustering groups the objects in a spatial dataset and identifies contiguous regions in the space of the spatial attributes. We formally define the task of finding uniform regions in spatial data as a maximization problem of a plug-in measure of uniformity and introduce a prototype-based clustering algorithm named CLEVER to find such regions. Moreover, polygon models which capture the scope of a spatial cluster and histogram-style distribution signatures are used to annotate the content of a spatial cluster in the proposed methodology; they play a key role in summarizing the composition of a spatial dataset. Furthermore, algorithms for identifying popular distribution signatures and approaches for identifying regions which express a particular distribution signature will be presented. The proposed methodology is demonstrated and evaluated in a challenging real-world case study centering on analyzing the composition of the city of Strasbourg in France.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: data mining, spatial databases and GIS

General Terms

Algorithm, Design, Experimentation, Performance

Keywords

Urban computing, spatial data mining, spatial clustering, finding uniform regions in spatial datasets, algorithms to discover the spatial structure of a city, region discovery

UrbComp'13, August 11–14, 2013, Chicago, Illinois, USA.

1. INTRODUCTION

Urbanization is the physical growth of urban areas as a result of global change where increasing proportion of the total population becomes concentrated in towns. The United Nations reported that since 2008 more than half of the world's population is living in urban areas [20]. There is a growing need to develop urban computing and analysis tools to guide the orderly development of cities. Recently, data describing cities is widely available, offering a great opportunity to develop urban computing techniques for urban planners to make smarter decisions because they can provide deep insights into city development dynamics. Moreover, it offers an opportunity to improve people's knowledge about the impacts from urbanization on the territory.

The step of urbanization leads to different functional regions in a city, called urban patches throughout the remainder of this paper, such as residential areas, business districts, industrial and recreational areas. Different types of urban patches support different needs of people's lives and "*serve as a valuable organization technique for framing detailed knowledge of a metropolitan area*" [18].

Improvement in scanning devices, gps, and image processing leads to an abundance of geo-referenced data. For example, tracking devices are now available to capture movement of human and animals in form of trajectories [19]. Furthermore, more and more Point of Interest (POI) databases are created which annotate spatial objects with categories, e.g. buildings are identified as restaurants, and systems, such as Google Earth, already fully support the visualization of POI objects on maps. As more and more data become available for a spatial area, it is desirable to identify different functions and roles which different parts of this spatial area play; in particular, it is desirable to identify homogeneous regions in spatial data and to describe their characteristics, creating high-level summaries for spatial datasets which are valuable for planners, scientists, and policy makers. For example, ecologists might be interested in partitioning a wetland area into uniform regions based on what animals and plants occupy this area and on other environmental characteristics [15]. Similarly, city planners might be interested in identifying uniform regions of a city with respect to the functions they serve for the people who live in or visit this part of a city [18].

More specifically in this work, we are interested in developing spatial clustering frameworks which are capable of creating summaries for an area of interest by identifying the spatial structure in spatial data and capturing its spatial heterogeneity. It should be stressed that traditional clustering algorithms are not suitable for this task—as they minimize distance-based objective functions or employ distance-based density estimation

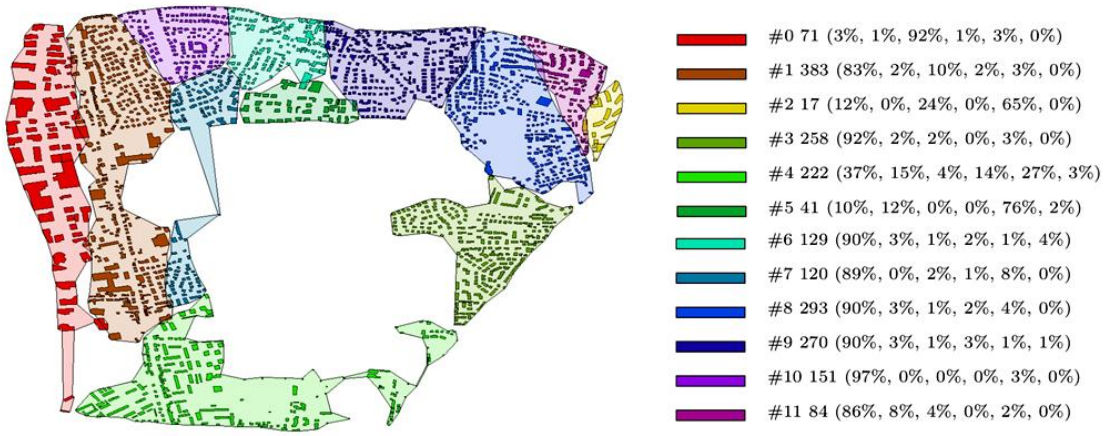

Figure 1: Example of a Spatial Clustering of Buildings Belonging to Different Building Types.

techniques—whereas assessing uniformity relies on non-distance based uniformity measures which operate on non-spatial attributes, such as purity, entropy or variance with respect to continuous non-spatial attributes. The focus of this paper is the introduction of a methodology which identifies uniform regions in spatial data and provides analysis functions to create summaries for the identified uniform regions. Its main technical contributions include:

1. It formally defines the problem of finding uniform regions in spatial data as a maximization problem.
2. A novel spatial clustering approach is proposed for identifying regions based on uniformity measures, which have to be expressed as reward-based fitness functions which are then maximized by the spatial clustering algorithm. The approach models the scope of spatial clusters as polygons and describes their characteristics using histogram-style distribution signature. Moreover, two novel interestingness measures which capture different notions of uniformity are introduced.
3. Popular signatures are proposed which are frequently occurring distribution signatures in the subspaces of a spatial area of interest. A novel approach which summarizes the composition of a spatial dataset by annotating regions with popular signatures is presented and algorithms to identify regions which match particular signatures are introduced.
4. The proposed framework is evaluated in a case study involving the building structure of the city of Strasbourg, France; in particular, the city is partitioned into uniform regions which are annotated with signatures and the benefit for domain experts of having such summaries is discussed.

Section 2 formally defines the problem of finding uniform regions in spatial dataset and introduces the spatial clustering approach for this task. Section 3 gives an experimental evaluation of the methodology. Section 4 discusses related work and Section 5 concludes the paper.

2. USING SPATIAL CLUSTERING TO DISCOVER UNIFORM REGIONS

2.1 Finding Uniform Regions in Spatial Data

Figure 1 gives an example spatial clustering result in which buildings of different types (e.g., schools and industrial buildings) of a city are clustered. The proposed methodology characterizes spatial clusters using their scope and signature.

The scope of a spatial cluster captures the model of a cluster. In our approach, we use concave polygons as models for spatial clusters as depicted in Figure 1; that is, if a spatial object is inside the polygon which describes the scope of a spatial cluster, it belongs to that spatial cluster. Secondly, the proposed methodology uses signatures to annotate spatial clusters. Signatures summarize the distribution of the objects that belong to a cluster. As the clusters in the example contain buildings belonging to different types, building type histograms are used as signatures to annotate spatial clusters. In our case study, there are six building types: single house, garages, industrial buildings, light buildings, collective buildings and schools. In this case, the signature of a cluster c is a vector $s(c)=(s_1,...,s_p)$ with $s_1+...+s_p = 1$ giving the proportions of categories of objects belonging cluster c. For example, the leftmost cluster is identified as cluster 0 and contains 71 buildings, and its building type signature is (3%, 1%, 92%, 1%, 3%, 0%), indicating that 3% of the buildings in cluster 0 are single houses, 1% are garages, 92% are industrial buildings, 1% are light buildings, 3% are collective houses and there are no schools in this spatial cluster.

So far we did not clearly discuss what distinguishes a uniform region from a non-uniform region in a spatial dataset. More formally, we are interested in obtaining spatial clusters which are uniform with respect to their signatures, using the following maximization procedure:

Input: a dataset O containing spatial objects belonging to p classes

Task: Find a spatial[1] clustering $X = \{c_1,...,c_k\}$ of O such that

(1) $c_i \subseteq O$ for i =1, ..., k

(2) $c_p \cap c_q = \varnothing$ *for* $p \neq q$

which maximizes the following objective function $\phi(X)$

$$\phi(X) = \sum_{c \in X, c' \in X} \frac{d(s(c), s(c'))}{b} \qquad (1)$$

where b is the number of pairs of neighboring clusters in X, s(c) denotes the signature of cluster c and d is a distance function which assesses the similarity of two signatures.

[1] A spatial Cluster is assumed to be contiguous in the space of the spatial attributes.

In summary, we are interested in obtaining a spatial clustering in which the average Euclidian distance between the signatures of neighboring clusters is as large as possible. It should be emphasized that only distances between neighboring clusters are considered in the definition of ϕ. In order to find uniform partitions, we can devise a search procedure which maximizes the disagreement of neighboring clusters with respect to their signatures.

However, developing a spatial clustering algorithm which directly maximizes $\phi(X)$ is quite challenging, as this would require to identify and to keep track of which spatial clusters are neighboring in order to compute $\phi(X)$, which leads to quite significant clustering overhead, and to theoretical problems[2]. Consequently, we are using different heuristics to find uniform spatial clusters without having to deal with the question which clusters are neighboring, and rely on approaches which use simplified versions of $\phi(X)$ instead; in particular:

1. We use prototype-based spatial clustering algorithms that are guaranteed to obtain contiguous spatial clusters without the necessity of knowing which clusters are neighboring. These algorithms maximize reward functions which encourage the merging of similar neighboring clusters and the splitting of non-homogeneous clusters if it leads to a significant increase in the total reward.
2. We reformulate the above optimization task in two ways:
   i. We make the problem supervised, by using interestingness functions which assess the quality of spatial clusters based on uniformity measures which capture a domain expert's notion of uniformity. Moreover, as we will see later, those uniformity measures assume that certain signatures are more desirable than other signatures. One such interestingness function is introduced in Section 3.1.
   ii. Instead of comparing the signatures of all neighboring clusters—as ϕ does—we employ an approach which identifies a set of popular[3] signatures and then uses those signatures to annotate clusters. In particular, this approach seeks for a spatial clustering which maximizes the match of a cluster's signature with the closest signature in the popular signature set, as will be explained in Section 3.2.

2.2 CLEVER—a Spatial Clustering Algorithm Supporting Plug-in Interestingness Measures

In order to employ these two approaches outlined in section 2.1, we need a spatial clustering algorithm capable of finding contiguous spatial clusters by maximizing a plug-in reward function which captures a particular notion of uniformity. A spatial clustering algorithm named CLEVER [3, 8] will be adapted for this task. In general, CLEVER is a prototype-based, k-medoid-style [10] spatial clustering algorithm which employs randomized hill climbing to maximize a plug-in reward function. Reward functions are assumed to have the following form when assessing the quality of a clustering $X = \{c_1,...,c_k\}$:

$$q(X) = \sum_{c \in X} reward(c) = \sum_{c \in X} i(c) \times |c|^{\beta} \qquad (2)$$

where $|c|$ denotes the number of objects in a cluster c, $i(c)$ is an interestingness function which assesses how interesting the cluster c is, and $\beta \geq 1$ is a parameter which determines how much reward is put on cluster size; β indirectly controls the numbers of clusters in X. As cluster size is rewarded using a non-linear function, usually fewer clusters are obtained when larger values for β are used. Moreover, the rewarding scheme encourages the merging of neighboring clusters with the same or similar signatures. The reward function assesses the quality of a clustering as the sum of the rewards of the individual clusters; two such interestingness functions will be introduced in Section 3. The pseudo-code of CLEVER is given in Algorithm 1.

Algorithm 1: CLEVER.

Input: Dataset O, distance-function d or distance matrix M, k', $i(c)$, β, sampling rate p

Output: Clustering X, quality $q(X)$, rewards for clusters in X

1: Randomly create a set of k' representatives

2: Sample p solutions in the neighborhood of the current representative set

3: If the best solution of the p solutions improves the clustering quality of the current solution; its representative set becomes the current set of representatives and search continues with Step 2; otherwise, terminate returning the current clustering.

CLEVER maintains a current set of representatives which are objects in the dataset and forms clusters by assigning the remaining objects to the closest representative in the representative set. It samples p representative sets in the neighborhood of the current representative set by adding, deleting, and replacing representatives. This process continues as long as a better clustering with respect to $q(X)$ is found. The algorithm begins its search from a randomly created set of k' representatives, where k' is an input parameter of the algorithm.

To give an example, let us assume we cluster a dataset $O = \{o_1,...,o_{200}\}$ with k' set to 3; in this case, the algorithm starts with a random representative set, let us say $\{o_3,o_9,o_{88}\}$, and forms clusters by assigning the remaining 197 objects to the closest representative which takes $O(k*(n-k))$ where n is the number of objects in the dataset and k is current number of representatives. Next, the algorithm samples p new clusterings in the neighborhood of the current solution by inserting, deleting or replacing representatives; for example, assuming p is 3, the algorithm might create clusterings for the representative sets $\{o_3,o_9,o_{88},o_{92}\}$,$\{o_3,o_{88}\}$, and $\{o_3,o_{17},o_{88}\}$ all of which have been obtained by a single insertion/deletion/replacement applied to the current representative set $\{o_3,o_9,o_{88}\}$. Next, the algorithm computes $q(X)$ for these three clusterings, and if the best of the three clusterings improves the clustering quality, its representative set becomes the new current solution; otherwise, the algorithm terminates. In general, assuming that CLEVER runs for t iterations its complexity is of the order of $O(t*p*k*n)$ with t and k usually being much smaller than n.

[2] If prototype-based clustering algorithms, such as K-medoids or K-means are used, a Voronoi tessellation can be used to derive cluster models from the set of cluster prototype which are convex polygons; unfortunately, it is not computationally feasible to compute Voronoi cells in higher dimensional spaces, as the complexity of the algorithm is exponential with respect to the dimensionality of the dataset. Consequently, it is only feasible to compute the Voronoi tessellation in 1D, 2D, and for small datasets in 3D. For density-based clustering algorithm the situation is even worse; for example, we are not aware of any methods which are capable of producing cluster models from a DBSCAN clustering.

[3] Popular signatures are distribution characteristics which occur frequently in contiguous subspaces of a spatial dataset.

2.3 Spatial Homogeneity Between Neighboring Clusters and Within Clusters

In Section 2.1, we stated that ideally signatures of two neighboring clusters should be significantly different from each other. In order to discuss this issue further, let us assume we have two neighboring clusters c_1 and c_2 containing a_1 and a_2 objects, respectively, which have exactly the same signature s, whose interestingness is $i(s)$. As we explained earlier, our reward framework employs a parameter $\beta>1$ that puts a reward on cluster size. We claim that in the discussed scenario our reward structure assigns a higher reward to a clustering which merges clusters c_1 and c_2 into a single cluster c as this clustering receives a higher reward, because of the following:

$$i(s) \times (a_1 + a_2)^{\beta} > i(s) \times \left(a_1^{\beta} + a_2^{\beta}\right) \qquad (3)$$

For example, if we have two neighboring clusters with purity *90%* that are dominated by instances belonging to the same class, merging the two clusters leads to better clustering with respect to $q(X)$, introduced earlier. Moreover, merging clusters frequently leads to a drop in interestingness/purity in the merged cluster; however, if the cluster size reward measured by $(clustersize)^{\beta}$ makes up for this loss of interestingness with respect to the cluster signature $s(c)$ the two clusters should still be merged. Therefore, the distribution of the objects belonging to a spatial cluster should be spatially homogeneous with respect to their associated signature.

2.4 Determining the Scope of a Spatial Cluster

In general, determining the scope of a spatial cluster is a challenging task. The goal is to create a spatial representation of a set of spatial objects in order to easily visualize it on the plane. One of easiest approaches is to compute the convex hull of the spatial objects in the cluster. However, the obtained convex hull polygon is usually not very tight and frequently enclosing empty spaces. This is especially the case when the spatial objects are spread and exhibit a low spatial density. Alpha shapes [7] and the concave hull [11] algorithm generalize the convex hull algorithm, allowing for the generation of much tighter polygons which might contain holes. In our proposed methodology, we use the PostGIS Concave Hull algorithm [21] for computing the scope of spatial clusters; we believe this approach is more effective than the convex hull algorithm, as it wraps a much tighter line around a set of spatial objects, resulting in less overlap with respect to the scope of neighboring clusters and less empty spaces in clusters, as as can be seen in Figure 1.

3. IDENTIFYING UNIFORM REGIONS IN A CITY

Since understanding the evolution of cities is the key to intelligent urbanization, there is a growing need to develop urban planning and analysis tools to guide the orderly development of cities, as well as to enhance their smooth and beneficiary evolution. However, it is a big challenge for urban planners to come up with methodologies to analyze how cities are changing. Partitioning a city into uniform regions facilitates this task, as change can be analyzed based on higher level of granularity instead on the raw data. In this section, we present a set of experiments which use the methodology, which was introduced in Section 2, to extract urban patches from a building dataset. In this context, metrics for evaluating the homogeneity of a group of buildings are very important as they impact how a city is partitioned into urban patches characterized by signatures. In particular, two such metrics, one based on purity and one based on popular signatures will be introduced in this section. In particular, we report the results of a series of experiments in which the CLEVER spatial clustering algorithm is used in conjunction with those two uniformity metrics to obtain interesting, uniform regions for the city of Strasbourg, France. As part of the GeOpenSim project, a temporal topographic database of the city of Strasbourg has been acquired [12]. As buildings are represented as polygons, we use Hausdorff distance [5] to compute the distance between buildings in the experiments.

3.1 Building Type Purity Experiments

This section introduces a purity interestingness function which measures uniformity by the degree of dominance of instances belonging to a single category and discusses spatial clustering results obtained with this interestingness function.

The purity interestingness function is used for analyzing interestingness with respect to a categorical non-spatial attribute. Purity interestingness $i_{PUR}(c)$ of a cluster c is computed using the following formula:

Let $R = max_t p_t\,(c), t \in cl(O)$

$$i_{PUR}(c) = \begin{cases} 0\,, & R < th \\ (R - th)^{\eta}\,, & otherwise \end{cases} \qquad (4)$$

where $cl(O)$ is the set of classes in the dataset O and p_t is a function that computes the proportions of the objects of class t belonging cluster c; $\eta > 0$ is the scaling factor and $th > 0$ is the threshold. For example, assuming that $th = 0.4$, $\eta = 1$, and $s(c)=(0.6, 0, 0, 0, 0.4, 0)$ indicating that 60% of the objects belong to the first category, and 40% of the objects belong to the fourth category, we obtain: $i_{PUR}(c) = 0.6 - 0.4 = 0.2$ for cluster c. In general when using the purity interestingness function, we are interested in obtaining clusters which are dominated by instances of a single category.

There are six different building types in the dataset: single house, garage, commercial building, light building, collective house, and school. In year 2008 78% of the buildings are single houses; commercial buildings take 7%; collective houses take 8%; 4% of the buildings are garages and 3% of the buildings are light buildings; finally, 1% of the buildings are schools. Building type signatures describe the characteristics of each urban patch which can help domain experts to better understand the composition of a city.

Figure 1 visualizes and lists the building type signatures of 12 clusters for the year 2008; they were generated by CLEVER using the purity interestingness function with $th = 0.5$, $\eta = 2$ and $\beta = 1.2$. Cluster *0* contains 92% commercial buildings; therefore, cluster 0 is labeled as a business urban patch. Cluster 10 is a residential area because 97% of the buildings in cluster 10 are single houses. There are 76% of collective houses in cluster 5, which indicates a living area with a lot of apartment complexes. Both garages and schools have very small percentage in the whole dataset, but garages and schools are more frequent in the collective housing areas in clusters 4 and 5, but surprisingly are not present in cluster 2. Figure 1 verifies that our approach is able to identify contiguous urban patches dominated by buildings of a single type.

3.2 Using Popular Signatures to Find Uniform Regions in a City

Many uniform regions are characterized by particular proportions of class densities without having a dominating class; for example, collective houses usually have a lot of garages next to them. This is the motivation for the following alternative approach which

seeks to find popular signatures which occur frequently in contiguous subspaces of the area of interest and then uses these signatures to annotate urban patches, as depicted in Figure 2.

Figure 2: Example of a Spatial Clustering of Buildings Annotated by Popular Signatures.

As we can see in Figure 2, the popular signature S4 is used to annotate regions in the northwest and southwest corner of the display. The challenge of generating such maps is that if we annotate a region by a popular signature, this makes only sense if the region's signature is close to the popular signature associated with it. To accomplish that, we need a spatial clustering algorithm to partition the spatial dataset into regions whose signatures are a good match with respect to a given set of popular signatures.

In the remainder of this section we will propose a framework for annotating regions with matching popular signatures. It first collects signatures using a sampling approach; second, it identifies a set of popular signatures from the collected signatures using a clustering approach; third, it uses a spatial clustering algorithm to identify regions with a good match with the set of popular signatures. As step 1 and 2 are kind of straightforward, we will not discuss those further.

As far as the third step is concerned, we run CLEVER using the following popular signature interestingness function $i_{POP}(c)$:
Let $cld = d(s(c), closest(s(c),P))$

$$i_{POP}(c) = \begin{cases} 0, & cld > D \\ (D - cld)^{\eta}, & otherwise \end{cases} \quad (5)$$

where $s(c)$ is the signature of cluster c, $closest(s,P)$ computes the closest signature in P to s, d denotes Euclidian distance, D is a match threshold, and η is a form parameter having value in $(0, \infty)$.

In summary, the interestingness $i_{POP}(c)$ of a cluster c is inversely proportional to the Euclidean distance of the cluster signature $s(c)$ to the closest popular signature in P. The interestingness function $i_{POP}(c)$ uses a match threshold D that serves the following purpose: if the distance of cluster c's signature and the closest popular signature is above D, we say c's signature does not match any signature in P, and c will not receive any reward and we will not annotate c with any signature.

Popular building type signatures describe compositions of urban patches which frequently occur in different parts of a city. To obtain a set of popular signatures, we first randomly created 1000 small spatial clusters and extracted their building type signature. Next, we apply a distance-based outlier detection technique to remove 10% of the building type signatures as outliers—signatures were sorted by their 3-nearest neighbor distance to the other signatures in the set. Signatures with the largest 3-nearest neighbor distance were removed from the signature set. Next, we clustered the remaining signature set using K-means with different k values ranging between 6 and 10 several times, and identified the clustering with the lowest squared average distance of the objects in the dataset to the cluster centroid they belong to. Finally, we extracted the centroids from the best clustering as popular signatures. Table 1 lists nine popular building type signatures that were obtained as the result of this process.

Table 1: Popular Building Type Signatures in 2008

Signature ID	Single House	Garage	Commercial Building	Light Building	Collective House	School
S1	77%	3%	2%	2%	17%	0%
S2	87%	4%	1%	3%	4%	1%
S3	2%	6%	0%	0%	92%	0%
S4	99%	0%	0%	0%	0%	0%
S5	48%	1%	46%	3%	2%	0%
S6	4%	0%	96%	0%	0%	0%
S7	37%	22%	4%	1%	32%	4%
S8	62%	6%	13%	12%	4%	1%
S9	85%	1%	14%	0%	0%	0%
Dataset	78%	4%	7%	3%	8%	1%

Table 2 summarizes a popular signature clustering result which was created using CLEVER and the popular signature interestingness function with parameters k'= 20, β =1.005, D =0.1 and θ =2. We use 0.1 as the threshold for the Euclidian distance of the cluster signature to its closest popular signature to indicate a good match. 14 out of the 16 urban patches have good matches with their popular signatures. Cluster 3 is quite unusual as it is dominated by light buildings and is not close to any popular signature in Table 1 at all, which is indicated by its very high Euclidian distance of 0.49 to its closest popular signature.

Our approach uses a spatial clustering algorithm—and not predetermined regions as suggested by [17, 18]—to identify the scope of a popular signature. We claim that the urban patches identified by our approach, exhibit a much better match with the popular signature set.

3.3 Querying a Spatial Dataset with Signatures

Although the presented popular signature mining algorithm has been originally developed to determine the scope of a set of popular signatures, it can be used in conjunction with any signature set P. This enables us to use the same algorithm for querying spatial datasets for the presence of particular "*query signatures*". For example, in the experiment summarized in Table 2, we came across cluster 3, which was dominated by light buildings and it might be interesting to see if its signature $Q1$=(29%,2%,9%,45%,15%,0%) occurs in other areas of the city; along the same line we might want to see, if there are regions with a high density of schools in a residential area captured by signature $Q2$=(70%,0%,0%,0%,0%,30%). Finally, we like to see if the popular signature $Q3$=(2%, 6%, 0%, 0%, 92%, 0%) (named S3 in Table 1) occurs anywhere in the dataset, as it did not match any cluster signature.

Figure 3 and Table 3 gives the result of running CLEVER with the popular signature interestingness function for signature set $P = \{Q1, Q2, Q3\}$ with parameters D = 0.1, η = 3, and β = 1.2. The spatial clusters in Figure 3 are annotated with corresponding signature if the distance of the cluster signature to its closest query signature in P is 0.1 or less. Table 3 lists the signatures for three clusters that are close to query signatures as well as the closest query signature and the distance to the closest query signature.

Table 2: Popular Building Type Signature Clustering Results for 2008

Cluster ID	Single House	Garage	Commercial Building	Light Building	Collective House	School	No. of Building	Closest Signature	Distance
0	89%	4%	2%	0%	5%	0%	56	S2	0.04
1	75%	7%	4%	0%	13%	0%	69	S1	0.07
2	73%	8%	6%	2%	12%	0%	52	S1	0.09
3	29%	2%	9%	45%	15%	0%	55	S8	0.49
4	72%	6%	11%	1%	10%	0%	157	S1	0.13
5	88%	4%	2%	3%	5%	0%	199	S2	0.02
6	100%	0%	0%	0%	0%	0%	112	S4	0.01
7	44%	1%	46%	5%	4%	0%	100	S5	0.05
8	87%	4%	1%	3%	3%	1%	335	S2	0.01
9	85%	1%	13%	1%	1%	0%	320	S9	0.01
10	77%	5%	8%	0%	10%	0%	39	S1	0.09
11	77%	3%	1%	1%	17%	2%	198	S1	0.03
12	36%	20%	3%	4%	34%	4%	142	S7	0.05
13	99%	1%	0%	0%	0%	0%	121	S4	0.01
14	98%	2%	0%	0%	0%	0%	57	S4	0.02
15	89%	0%	0%	0%	11%	0%	27	S2	0.09

Table 3: Clusters Matching Query Signatures

Cluster ID	Matched Signature	Single House	Garage	Commercial Building	Light Building	Collective House	School	Distance
5	Q1	29.63%	1.85%	9.26%	44.44%	14.81%	0%	0.009
11	Q3	2.78%	5.56%	0%	0%	91.67%	0%	0.010
13	Q2	66.67%	0%	0%	0%	0%	33.33%	0.047

Figure 3: Visualization of Clusters Matching Query Signatures

As can be seen, the algorithm rediscovered the same region with a majority of light buildings identified by the popular signature clustering algorithm but no other regions which match this signature. Moreover, a single region which almost perfectly matches the popular signature *Q*3 was found. Finally, we were able to find a single region with a mixture of schools and single houses, but the match of the regions' signature with *Q2* is of medium quality, as the Euclidian distance between the two signatures is about 0.047.

3.4 Sensitivity Analysis

CLEVER has been designed to find a "good" solution for an in general NP-hard problem relying on randomized hill climbing. As all optimization procedures which start with randomly created initial solutions, CLEVER—as K-means—is sensitive to initialization, as different initializations may lead to different alternative solutions. In this section, we discuss the result of an experiment which analyzes CLEVER's sensitivity to initialization.

To analyze CLEVER's sensitivity to initialization, we ran the building type purity clustering procedure 20 times with parameters $k' = 20$, $\beta = 1.05$, $\eta = 3$ and $th = 0.5$ and collected the following run characteristics: $q(X)$, number of the clusters in the final clustering, number of iterations, and the number of clusterings generated during the run. The sampling procedure used in this experiment first samples 15 clusterings in the neighborhood of the current clustering, then—if there is no improvement — 30 solutions, and finally 180 solutions; if none of the 225 sampled clusterings improves the current clustering, the search ends. According to the results reported in Table 4, CLEVER terminated after at an average 32 iterations and searched at an average 1400 clusterings. Although CLEVER starts from different initial clusterings, the quality of the clustering results are relatively stable around 729 with a standard deviation of 24. However, the number of final clusters obtained differs quite significantly between the twenty runs, ranging between 3 and 23. This fact indicates that the obtained 20 final clusterings—although having a similar quality with respect to $q(X)$—differ from each other significantly.

Table 4: Building Type Purity Sensitivity Results

Run ID	$q(X)$	No. of Clusters	No. of Iterations	Generated Clusterings
1	776.81	7	38	1635
2	764.68	8	43	1920
3	756.20	10	25	645
4	747.56	11	39	1830
5	746.39	12	29	1245
6	744.51	9	30	1470
7	741.23	11	24	1170
8	738.21	3	31	1470
9	737.03	13	29	1245
10	736.27	16	45	1950
11	727.90	11	39	2010
12	726.31	8	48	2175
13	719.12	10	23	960
14	716.62	23	36	1395
15	715.18	14	20	525
16	710.86	16	26	1380
17	707.44	9	31	1140
18	693.47	18	37	1605
19	688.78	16	31	1665
20	685.85	16	24	1005
Mean	729.02	12.05	32.40	1422
STD	24.63	4.55	7.88	444.40
Max	776.81	23.00	48.00	2175.00
Min	685.85	3.00	20.00	525.00

3.5 Performance Analysis for CLEVER

Table 5 gives some performance characteristics for the clustering results that were reported in Sections 3.1 to Section 3.4 in terms of iterations needed, number of clusterings generated, and wall clock time. CLEVER was run on a dataset containing 2039 objects on a computer with the processor running at 3 GHz and 8 GB main memory.

Table 5: Performance Characteristics of the Reported Clustering Results

	No.of Iterations	No.of Clusterings Generated	Time Elapsed
Section 3.1	30	1485	32.92*s*
Section 3.2	35	1590	33.65*s*
Section 3.3	44	2670	38.26*s*
Section 3.4	34	1422	31.15*s*

4. RELATED WORK

Work in [6, 9] proposed a region discovery framework based on a fitness function to maximize. The framework adapts four representative clustering algorithms, exemplifying grid-based, prototype-based, density-based, and agglomerative clustering algorithms to optimize the fitness function. The fitness function is defined according to the application, and the goal is to model the interestingness of a region. Other work seeks to find uniform regions for spatial regression [2, 14]; using quite different methods, both approaches partition the space into regions, associating different regression functions with different regions; uniformity in this work is associated with point sets sharing the same or a similar relationship between a dependent variable and a set of independent variables. Sheng et al. [13] introduces a search algorithm which finds the top-k regions with a similar distribution of POIs on a spatial map.

One key idea of this paper is to use signatures to annotate spatial clusters and to propose a framework to mine cluster signatures in spatial datasets. We are not aware of any work that uses signatures in conjunction with clustering; however, signatures have been used for other purposes. Applegate et al. [1] state that "*signatures are compact representations…that capture important characteristics of massive datasets*" and then investigate a special family of signatures for multidimensional distributions that represent the distribution of probability mass over a manifold and introduce a novel distance function for such signatures. Cortes et al. [4] discuss the use of signatures for mining massive telecommunications data to find communities of interest, and for fraud detection. Wong et al. [16] demonstrate the benefits of using data signatures to guide the visualization of complex scientific datasets.

Joshi et al. [22] proposes a dissimilarity function for clustering geo-spatial polygons. The proposed dissimilarity function takes into account different characteristics of the polygon separated in different groups: non-spatial attributes, intrinsic spatial attributes and extrinsic spatial attributes. The dissimilarity function computes the dissimilarity between polygons as a weighted function that compute the distance between two polygons in the different attribute spaces. This approach is different from our approach which supports plug-in interestingness functions that allow assessing cluster quality using non-distance based interestingness measures; moreover, our approach generates clusters which are contiguous in the subspace of the spatial attributes.

The use of topic discovery approaches [17, 18] to annotate spatial regions has gained some popularity recently. There are two major differences between our approach and the topic discovery approach: First, our approach is supervised based on a domain expert's notion of uniformity, which has to be expressed by a plug-in interestingness function, whereas in the other approach popular signatures are identified by an unsupervised topic discovery approach. Second, the topic discovery approach requires an a priori given partitioning of the city as an input, whereas our approach uses spatial clustering algorithms to determine such a partitioning which is optimal with respect to a given notion of uniformity.

5. SIGNIFICANCE AND IMPACT

This paper introduces a spatial clustering methodology which identifies contiguous regions in the space of the spatial attributes which are uniform with respect to their signatures, which represent statistical summaries for the objects belonging to a particular cluster. The second idea advocated in the paper is to mine spatial data for the presence of particular signatures. We claim that these two types of signature-based spatial clustering have broad applications in urban computing.

The proposed methodology defines the task of finding uniform regions formally as a maximization problem. Various objective functions and corresponding algorithms are introduced. In particular, we introduce a prototype-based clustering algorithm named CLEVER, which identifies uniform regions in a spatial dataset by maximizing a plug-in measure of uniformity, relying on a randomized hill climbing approach. Moreover, polygon models which capture the scope of a spatial cluster and histogram-style distribution signatures are used to annotate the content of a spatial cluster; both play a key role in summarizing the composition of a spatial dataset. We claim that the presented approach is novel and unique as existing clustering algorithms are

not suitable for this task as they minimize distance-based objective functions, whereas assessing uniformity relies on non-distance based uniformity measures. The efficacy of the proposed methodology is demonstrated by a challenging real-world case study centering on analyzing the composition of the city of Strasbourg in France based on building characteristics.

Applying the methodology, presented in this paper, faces several challenges, such as sensitivity to initialization, finding more suitable algorithms to compute the scope of a set of spatial clusters, providing a better theoretical foundation for signature mining, the capability to identify spatial clusters of arbitrary shape, and the need to run spatial clustering algorithms multiple times. Finally, as the computational complexity of signature mining is usually very high, there is a need for parallel signature mining algorithms. Our current and future work centers on dealing with these challenges.

6. REFERENCES

[1] D. Applegate, T. Dasu, S. Krishnan, and S. Urbanek. Unsupervised clustering of multidimensional distributions using earth mover distance. In Proceeding of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2011.

[2] O. U. Celepcikay and C. F. Eick. A regional regression framework for geo-referenced datasets. In Proc. 17th ACM SIGSPATIAL International Conference on Advances in GIS (GIS), pages 326–335, November 2009.

[3] C.-S. Chen, N. Shaikh, P. Charoenrattanaruk, C. F. Eick, N. Rizk, and E. Gabriel. Design and evaluation of a parallel execution framework for the clever clustering algorithm. In Proc. ParCo Conference, 2011.

[4] C. Cortes and D. Pregibon. Signature-based methods for data streams. Data Mining and Knowledge Discovery, 5:167–182, 2001.

[5] N. Cressie. Statistics for spatial data. Addison-Wesley Publishing Company, 1993.

[6] W. Ding, R. Jiamthapthaksin, R. Parmar, D. Jiang, T. Stepinski, and C. F. Eick. Towards region discovery in spatial datasets. In Proc. Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining (PAKDD), pages 88–99, September 2008.

[7] H. Edelsbrunner, D. Kirkpartick, and R. Seidel. On the shape of points in the plane. IEEE Transactions on Information Theory, 29(4):551–559, 1983.

[8] C. F. Eick, R. Parmar, W. Ding, T. Stepinki, and J.-P. Nicot. Finding regional co-location patterns for sets of continuous variables in spatial datasets. In Proc. 16th ACM SIGSPATIAL International Conference on Advances in GIS (GIS), November 2008.

[9] C. F. Eick, B. Vaezian, D. Jiang, and J. Wang. Discovery of interesting regions in spatial datasets using supervised clustering. In Proc. 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD), September 2006.

[10] L. Kaufman and P. Rousseeuw. Finding Groups in Data: An Introduction to Cluster Analysis. Addison-Wesley Publishing Company, New York, 1990.

[11] A. Moreira and M. Santos. Concave hull: a k-nearest neighbors approach for the computation of the region occupied by a set of points. In Proc. International Conference on Computer Graphics Theory and Applications (GRAPP), pages 61–68, 2007.

[12] A. Ruas, J. Perret, F. Curie, A. Mas, A. Puissant, G. Skupinski, D. Badariotti, C. Weber, P. Gancarski, N. Lachiche, A. Braud, and J. Lesbegueries. Conception of a gis platform to study and simulate urban densification based on the analysis of topographic data. Cartography and GIScience, 1:413–430, 2011.

[13] C. Sheng, Y. Zheng, W. Hsu, M. L. Lee, and X. Xie. Answering top-k similar region queries. Database Systems for Advanced Applications, Lecture Notes in Computer Science, 5981:186–201, 2010.

[14] S. Vucetic and Z. Obradovic. Discovering homogeneous regions in spatial data through competition. In Proc. ICML Conference, pages 1095–1102, 2000.

[15] L. Windham, M. Laska, and J. Wollenberg. Evaluating urban wetland restorations: Case studies for assessing connectivity and function. URBAN HABITATS, 2(1):130–146, 2004.

[16] P. Wong, H. Foote, R. Leung, D. Adams, and J. Thomas. Data signatures and visualization of scientific data sets. IEEE Computer Graphics and Applications, 2000.

[17] Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang. Geographical topic discovery and comparison. In Proc. WWW Conference, 2011.

[18] J. Yuan, Y. Zheng, and X. Xie. Discovering regions of different functions in a city using human mobility and pois. In Proc. KDD Conference, August 2012.

[19] Y. Zheng and X. Zhou. Computing with spatial trajectories. Springer-Verlag New York Inc., 2011.

[20] The Associated Press 2008, *UN says half the world's population will live in urban areas by end of 2008*, February 2008.

[21] Boston Geographic Information Systems, http://www.bostongis.com/postgis_concavehull.snippet

[22] D. Joshi, A. Samal, and L. Soh. A dissimilarity function for clustering geospatial polygons. In Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS), pages 384-387, 2009.