DOI: 10.1111/cgf.13882

COMPUTER GRAPHICS *forum*
Volume 00 (2019), number 00 pp. 1–16

Visualizing Dynamics of Urban Regions Through a Geo-Semantic Graph-Based Method

Yunzhe Wang[1], George Baciu[1] and Chenhui Li[2]

[1]Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Hong Kong
yunzhew1991@gmail.com, csgeorge@polyu.edu.hk
[2]School of Computer Science and Technology, East China Normal University, Shanghai, China
dawn.chli@gmail.com

Abstract

In urban analysis, it is desirable to find regions where a primary socio-economic activity dominates as a key endeavour. This can be accomplished by aggregating neighbouring locations where similar activities take place. However, people move and their activities change over time. Furthermore, the boundaries of regions are not stationary. Thus, it is challenging to update region divisions and track their evolution. Geo-textual data embody geographical information and activity descriptions. We obtain changes in regional boundaries by iteratively applying a community detection *process to a sequence of latent graphs that are constructed from geo-textual data. Region characteristics are interpreted by topics learned by the latent Dirichlet allocation model. We also propose a matching algorithm to expose region transformations between different timestamps. Interesting patterns of evolution emerge after clustering the migration trajectories of region centroids. In our visual system, users can explore the evolution of regions through animations and linked snapshots. To facilitate visual comparisons, we represent regions by hexagonal tiling that better construct arbitrary regional shapes. The effectiveness of our method is evaluated on two case studies using real-world datasets, and a user study shows that our visual analytics system is highly effective in performing studies on such regional maps.*

Keywords: human–computer interfaces, interaction, information visualization, visualization, visual analytics

ACM CCS: Human-centred computing → Visualization

1. Introduction

Understanding the dynamics of urban activity in large cities and highly developed areas can be a daunting task due to a large number of parameters and the complexity of interactions between structure, accessibility and the movement of people and vehicles. This complexity increases substantially when additional time constraints are imposed as competing agendas for resource allocation arise and fade over shorter time periods. To reduce this complexity in urban analysis, we propose to segment the larger areas into smaller components and focus on their dominant functional units, one at a time.

There are various segmentation criteria available. For example, based on physical characteristics, the surface of our planet can be largely divided into continents and oceans. As people predominantly shape the large features of our planet, we can think of administrative divisions, transportation, tourism, restricted areas and so on. For consistency, we use the term *region* to denote non-overlapping sub-areas, and *venue* to indicate the location of designated social activities. The feature that a region carries is called a *function*. Functions reflect the context of human activities.

Region segmentation allows for the overall understanding of the influence or action that a specific function has over a certain geographical area. From this segmentation, urban planners can learn about land uses [PQW*13], retailers can decide where to set up advertisements and warehouses [LWL*16] and law enforcement officers can rank locations by their security risks [MJR*11].

Existing researches on region segmentation rely on various approaches and data sources. Yuan *et al.* [YZX*15] treat the space separated by main roads as regions, and they require extra information like the statistics of points of interest (POI) to interpret region functions. However, since road networks update slowly, their

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

segmentation can hardly demonstrate changes of region shapes in real time. Wu *et al.* build a system called Mobiseg [WZC*17] that splits regions by interactively clustering local districts that have similar activity patterns. The patterns are extracted from mobility data, and they vary across time. Hence, the segmentation can be updated accordingly. But, Mobiseg lacks the exposure of the relationship between regions at different time steps, and does not differentiate between types of regional transformations. This makes it difficult to analyse the evolution of various regions and their functional characteristics over time.

In this paper, we describe the implementation of a visualization system for presenting and exploring region evolution. The system consists of two collaborating views, the *overall view* and *comparison view*.

The *overall view* supports the interactions on the entire map. It shows the region centroids at different time steps. From time t to $t+1$, our matching algorithm aims to find pairs of centroids and their successors that belong to identical regions. Then, the migration trajectories of centroids are displayed by animated sequences. In this view, we also demonstrate evolutionary patterns obtained by summarizing all trajectories. From the patterns, we may find that some regions only exist at one time step, which promotes our conjecture that some temporary events reshape these regions but do not have a large impact over the evolution of the regions. Some regions involve only small changes and their centroids shift within a limited area. It is likely that these regions contain stable infrastructures, like educational institutes and commercial buildings.

The *comparison view* consists of multiple snapshots from which users can observe region territories. We calculate territories as the minimum geographical area that covers closely located venues, where similar human activities occur. Links between snapshots connect matched pairs of region centroids, and they pass through a stack chart that shows the statistics of the region transformations. We categorize transformations into seven types: *appear*, *disappear*, *expand*, *shrink*, *remain*, *merge* and *split*, all indicating changes in territory. Topic changes can be identified by comparing keywords.

The *overall view* and *comparison view* are complementary to each other. The former outperforms the latter in visual scalability. Users can access long-term evolution of regions and roughly judge their stability. The latter integrates multiple visual techniques, and it is more suitable for comparison tasks and inspections of details of an area of interest (AOI).

Our main contributions are as follows:

- **Region delimitation**: we put forward a graph-based method to delimit functional regions. Based on geo-textual data, we construct graphs that encode the information about the density of population and the consistency of activities. We take communities detected from the social graph as the reference for calculating region territories.
- **Evolution interpretation**: we propose a matching algorithm to expose region transformations between consecutive time steps. Transformations are categorized into seven types. The evolution of a region can then be interpreted by a series of transformations in chronological order.
- **Evolutionary patterns**: we obtain migration trajectories by connecting region centroids, and abstract them as graphs. By clustering graphs of similar topologies, we reduce the diversity of trajectories and reveal representative patterns of evolution.
- **Visualization system**: we represent the evolution of a region by using a rich set of visual designs. Animations show coherent transformations, and snapshots depict different region segmentation across time. We evaluate the system by two application scenarios and a study on users' performance of conducting analytic tasks.

2. Related Work

2.1. Urban analysis

Existing work has taken various data sources and methods to learn characteristics of urban areas. For example, we can obtain an overview of land usages from satellite images [VDVJC11], and use POI data to categorize areas into different types [KKK*10, LFYX13, WGDQ18]. Recently, geo-textual data have been widely used to develop applications of situation awareness [MJR*11], user recommendations [ZLY*16] and data queries [CCC13, CFZ16]. Unlike conventional urban analysis that focuses on regional semantics, SemanticTraj [ADWK*16] concentrates on street-level information and supports queries of various syntaxes. The system leverages taxi trajectory data and transforms them into documents with rich semantics.

Lansley and Longley [LL16] classify geo-tagged tweet streams into 20 groups with distinctive topics. The objective is to figure out the influence of time, place and people on tweet contents. They study the temporal intensity of topics and present the spatial distribution of topics by heatmaps. Urban Space Explorer [KCW*17] constructs movement flows from geo-located tweets by computing the shortest path on street networks between consecutive tweet locations. The visual system contains multiple views that show flows go inside or out of a selected region, as well as flow patterns that are distinguished by length and duration. Kling and Pozdnoukhov [KP12] investigate the temporal usage of venues, and then inspect the functional structure of neighbourhoods by clustering city areas. Given a specific region, alluvial diagrams show topic transitions during a period. All these works are based on geo-textual data and extract topics by using the latent Dirichlet allocation (LDA) model. However, they care less about dynamic region segmentation, and lack utilities that allow users to conduct map-based comparisons of regions.

Pan *et al.* [PQW*13] adapt the density-based spatial clustering of applications with noise (DBSCAN) algorithm to data with heterogeneous densities, and represent a region by a cluster of venues. Spatial temporal-DBSCAN (ST-DBSCAN) [BK07] combines spatial and non-spatial similarity metrics to cluster venues. Our previous work decides venue groupings by a graph-based method [WBL17, WBL18]. Cranshaw *et al.* [CSHS12] find regions where businesses have similar check-in patterns. But they are unable to interpret the function of regions unless extra information is given. Based on regions separated by main roads, Yuan *et al.* [YZX*15] use POI statistics to infer functions of regions. The issue is that their method cannot further divide a large region that consists of

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

multiple functions, neither update the segmentation in time. MobiSeg [WZC*17] solves the issue by dynamically clustering districts that have similar activity patterns. Districts are cells of a tessellation grid on the focal area. Data from multiple sources are fused to mitigate the sparsity problem. Methods and systems mentioned above devote to static or dynamic region segmentation, but they still need improvement on demonstrating the evolutionary procedure of regions.

2.2. Graph-based geographical analysis

Modelling geographical data by graph structures can reduce the complexity of analysis and exert the strength of graph-based techniques. Compared to an ordinary graph, node positions of a geographical graph are fixed, and the relationship between nodes depends on different scenarios.

To find regions where people communicate with each other frequently, Ratti *et al.* [RSC*10] detect communities from graphs that model the interactions between users over telephones. Huang *et al.* [HZM*16] transform mobility trajectories to graphs so that road segments are abstracted as graph nodes and their accessibility is indicated by graph links. Their goal is to measure the importance of roads and expose areas that involve with intense human activities. Hence, they adopt the centrality metrics and apply partitioning methods to the graph. To build a location-based recommendation system, Xie *et al.* [XYW*16] construct three types of graphs to simulate the relationship between POIs, time and words. As a result, they can calibrate the recommendation by capturing the geographical, temporal and semantic influences. Finding similar mobility trajectories at different time steps is difficult. But converting trajectories into graphs and then clustering them makes it easy to reveal mobility patterns during the evolution [VLBR*16].

2.3. Temporal-geographical data visualization

Given temporal data, there are basically two streams of visualization techniques. One is the *static timeline-based representation* containing a sequence of small multiples, each representing a snapshot of the data. The other is the *animation-based representation* that continuously displays data in chronological order.

Each method has advantages and disadvantages. The former provides an overview and facilitates comparisons of different time steps; however, it is constrained by the limited [LBH18, LBWZ18] and [BBDW17]. The latter decreases the complexity of expression by multiplexing states to time rather than space [BPF14], but it hardly preserves the mental map [MELS95]. In some circumstances, special techniques, like space-time cube [BDA*14] and functional boxplots [SG12], can also be used.

As for visualizing geographical data, there are many other issues to deal with. First, within a limited area on the map, embedding massive temporal information might raise severe visual clutters [AAG03]. To alleviate the problem, solutions include level-of-details [FPV*13], linked views [LWL*17, CYW*16] or improving the compactness of visual designs [SLQW17]. Nevertheless, temporal-geographical visualization should support mining tasks which include: clustering, predictive learning, change detection, frequent pattern mining, anomaly detection and relationship mining [AKK18].

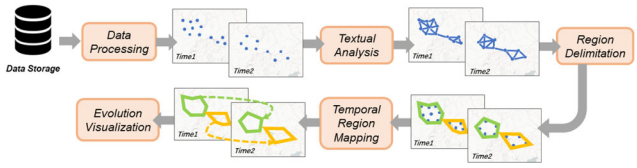


Figure 1: *The overview of the visual system implementation.*

Another issue is how to transform visual elements into comparable scales, as comparisons are frequently made in both geographical and temporal dimensions. Grid-based methods have at least three benefits. Firstly, it becomes straightforward to compare the size of regions after tiling them with grid cells [MH17]. Because cells have a uniform size. Secondly, better aesthetics can be achieved [CCW*16]. Thirdly, a constant grid provides a consistent reference for temporal explorations. Classic shapes of grid cells include triangles, rectangles and hexagons [CBC*15].

3. System Overview

As shown in Figure 1, we build latent graphs on discrete geographical venues. Close venues are linked. Through textual analysis, the link weight is measured by the similarity score of semantics of end venues. In the procedure of region delimitation, dense venues that also have similar semantics are grouped to form disjoint regions. Applying above steps repeatedly to each snapshot of geo-textual data, we can get independent region distributions. Furthermore, the trend of evolution is exposed by mapping regions at different snapshots. Details can be found in Sections 4 and 5.

4. Region Delimitation

Our objective is to separate regions that reflect the natural gathering of people and distinguish from each other by the theme of inside activities. Density-based algorithms, like DBSCAN, can be used to cluster venues by their spatial closeness, but the corresponding activities in one cluster not necessarily have similar semantics. Many previous work employed mobility data, such as taxi routes and commuting flows [RSC*10, CSHS12, WZC*17]. However, extra information, like POI statistics, was needed for region interpretation. We propose a graph-based method. It encodes the spatial and semantic proximity of venues simultaneously. With geo-textual data, we can directly learn topics to characterize regions.

4.1. Data description

We use a four-tuple $< time, venue, text, meta\_data >$ to depict a geo-textual item, where $time$ and $venue$ denote when and where an activity happens and $text$ describes the context of the activity. A $venue$ is identified by a coordinate of longitude and latitude, and $text$ consists of a list of words. $meta\_data$ include supplementary information, such as a photo taken at the venue. During the preprocessing stage, we aggregate data items by uniform time intervals, e. g. day, month or year, for the ease of analysis.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

4.2. Semantics extraction

The words that people use to describe activities are highly diverse and are likely to be meaningless, such as stop words. We reduce the difficulty of analysis by extracting insightful semantics and then calculating their similarities. Specifically, the LDA model is adopted, because it returns interpretable topics distributed at all venues.

LDA represents a document as a probabilistic mixture of topics learnt from the corpus. A topic corresponds to a cluster of words. At one time step, the *corpus* is composed of words collected from the text component of all geo-textual items, excluding meaningless words. The text component of each data item is regarded as a single *document*. Assuming that there are n topics $\{t_1, t_2, \ldots, t_n\}$ in the corpus, we compute:

- **Venue Topics**: the topic distribution at a venue v, denoted by $T_v = [p_{t_1}, p_{t_2}, \ldots, p_{t_n}]$. $\sum_{i=1}^{n} p_{t_i} = 1$, where p_{t_i} is the probability of topic t_i.
- **Region Topics**: the average topic distribution of all venues within a region r, represented by $T_r = \frac{1}{|V_r|} \sum_{i=1}^{|V_r|} T_i$, and $|V_r|$ is the number of venues inside r.

For venues that associate with multiple items, T_v is an averaged topic distribution of these items. To update topics, we conduct LDA repeatedly for each time step.

We can measure the extent of consistency of activities in r by the information entropy of T_r, which is defined as:

$$E_r = -\sum_{i=1}^{n} \overline{p_i} \log \overline{p_i}, \tag{1}$$

where $\overline{p_i}$ is the averaged probability of topic t_i of all $|V_r|$ venues. A low entropy value implies that the region tends to have an explicit function that can be interpreted by a dominate topic in T_r. Otherwise, we can find various types of activities involved in the region.

4.3. Geo-semantic graph

We abstract venues as graph vertexes and each venue only connects with its k nearest neighbours. The weight of connections equals to the similarity score between topic distributions of two end venues v_i and v_j, which is defined by the reciprocal of symmetric Kullback–Leibler divergence:

$$KL(T_{v_i}, T_{v_j}) = \sum_{t=1}^{n} T_{v_i,t} \cdot \frac{T_{v_i,t}}{T_{v_j,t}}, \tag{2}$$

$$KL_{sym}(T_{v_i}, T_{v_j}) = \frac{1}{2} \cdot (KL(T_{v_i}, T_{v_j}) + KL(T_{v_j}, T_{v_i})). \tag{3}$$

Then, we cluster venues by applying the Louvain method [BGLL08], which aims at maximizing the graph modularity:

$$Q = \frac{1}{2m} \sum_{uv} \left[A_{uv} - \frac{k_u k_v}{2m} \right] \delta(c_u, c_v), \tag{4}$$

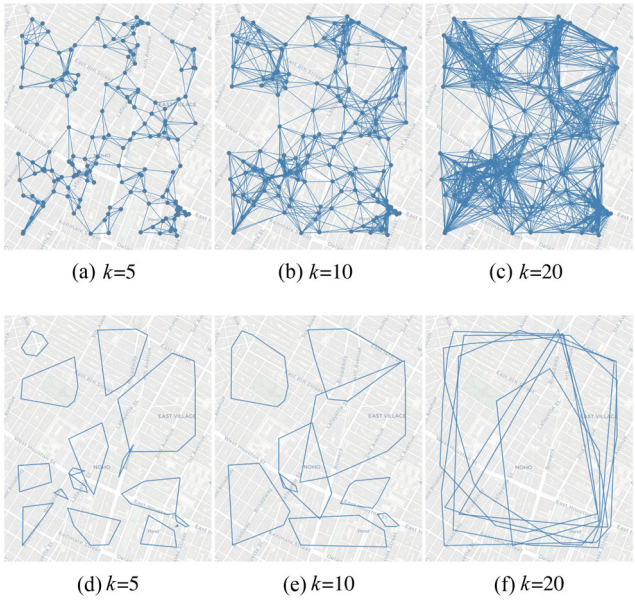


Figure 2: *Given an identical set of venues, the latent geo-semantic graphs (first row) and the obtained regions (second row).*

where A is the adjacency matrix, $A_{uv} = KL_{sym}(T_u, T_v)$; k_u is the sum of weights of links attached to vertex u; $2m$ is the sum of all link weights; c_u indicates the membership of u to community. $\delta(\cdot)$ equals to 1 if $c_u = c_v$, otherwise, its value is 0. Modularity measures the ratio of connections between venues in same clusters to connections between venues in different clusters. Initially, to represent region territories, we encompass venues belonging to same clusters with convex hulls. Later, in Section 7.2.1, we refine visual aesthetics by tiling convex polygons with hexagonal grids.

An example is shown in Figure 2. Given the same set of venues, we can generate graphs with different k values. Normally, a lower value results in a sparser graph and more regions. Besides, regions are relatively smaller and less overlapped. With the increase of k, each venue gets connected to more neighbours that are further away. An extreme case is that the graph is fully connected, and we get only one region covering all venues.

Based on the way of graph construction, closely located venues are more densely connected, as we can see in the first row of Figure 2. Hence, they are more likely to be in the same cluster. Besides, according to Equation (4), if a link has a heavier weight, then its incident vertexes are also more likely to be assigned to the same cluster. In a word, we consider both the density of population and the consistency of activities to separate regions.

The graph is latent because it is invisible to users. We repeat graph construction for individual time steps to update the region segmentation. Venues are organized into a *k-d* tree to speed up the search for nearest neighbours. Due to the sparsity of data at specific areas, a venue might be far away from its neighbours. In case of getting large regions with few venues inside, we can remove those connections whose end-venues have a geographical distance that exceeds the limitation.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

Besides the Louvain method for clustering graph vertexes, a lot of other approaches are also available [FOR10], such as the traditional partitioning methods [HW04, RMJ07], spectral clustering methods [CSCC05], etc. The Girvan and Newman algorithm is the first modern technique of community detection. It is implemented by iteratively removing links based on their betweenness. However, it has a high computational complexity ($O(n^3)$ on a sparse graph, and n is the number of nodes). We chose the Louvain algorithm because of its high efficiency in processing large graphs. It runs approximately in time $O(n \log n)$. Also, it can find communities that contribute to a high modularity without knowing their count in advance. According to the comparison made by Lancichinetti and Fortunato [LF09], the Infomap [RB08] and the multi-resolutional method [RN09] can be alternatives that have the similar performance.

5. Region Matching

To understand how a region evolves from time t to $t+1$, we need to find its target regions after transformation, and then decide the type of transformation. Hence, we propose a matching algorithm as shown in Algorithm 1. The output are two sets of matching results S and S' for regions at t and $t+1$, respectively. M and M' denote the set of matched regions. By applying matching operations to a sequence of snapshots, we can track the temporal changes of regions.

Algorithm 1. Matching Regions

Require:

R_t, R_{t+1}: the set of regions at time t and $t+1$, respectively;
$R_t = \{r_1, r_2, \cdots, r_m\}$, $R_{t+1} = \{r'_1, r'_2, \cdots, r'_n\}$.

Ensure

$S = \{(r, M_r) | \forall r \in R_t, M_r \subseteq R_{t+1}\}$;
$S' = \{(r', M'_{r'}) | \forall r' \in R_{t+1}, M'_{r'} \subseteq R_t\}$

1: **for** $r_i \in R_t$ **do**
2: $NR_i \Leftarrow k$ nearest regions to r_i in R_{t+1};
3: **for** $nr_i \in NR_i$ **do**
4: **if** $Overlapped(r_i, nr_i) = True$ **then**
5: $o \Leftarrow$ overlapping area of r_i and nr_i
6: $v_i \Leftarrow$ venues within r_i and also locate in o;
7: $nv_i \Leftarrow$ venues within nr_i and also locate in o;
8: **if** $(\frac{|v_i|}{|r_i|} > threshold)$ and $\frac{|nv_i|}{|nr_i|} > threshold)$ **then**
9: add nr_i to M_{r_i};
10: add r_i to M'_{nr_i};
11: **end if**
12: **else**
13: break;
14: **end if**
15: **end for**
16: **end for**

For two regions to be matched, they need to fulfil the following conditions: first, their areas overlap; second, they should have a certain proportion of designated venues inside the overlapping area. The second condition refines the matching result by inspecting the importance of the overlapping area. The importance depends on the density of venues rather than the area size, because a large area containing only a few venues is perceived to be less important.

In *line 2*, we firstly build a *k-d* tree in $O(n \log n)$ time to organize centroids of all regions in R_t and R_{t+1}, and n is the total number of regions. Then, searching the nearest region is an $O(\log n)$ operation on average. In *line 4–5*, we calculate o by adopting the Weiler–Atherton clipping algorithm [WA77]. In *line 6–7*, we use the ray casting method [ROT82] to determine if venues are inside o. Increasing the threshold in *line 8* will decrease the number of matched regions.

To accelerate the matching procedure, we reduce the candidate targets twice. First, in *line 2*, r_i only compares to its k nearest rather than all regions in R_{t+1}. Because the probability that two regions are overlapped decreases when their distance increases. In our tests, we find that a region rarely overlaps with more than six other regions, i.e. $k = 6$. The distance between regions is measured by the distance between their centroids that are obtained by averaging all venue coordinates. Second, we sort regions in NR_i in ascending order based on their distances to r_i. In *line 12*, the matching process stops once a candidate does not overlap r_i. It is noteworthy that acceleration might compromise the matching accuracy in that certain regions probably overlap with more than k neighbours. Also, we should have used the single-linkage criteria [SR05] to measure the distance between region borders, but it takes too many calculations.

The algorithm is applicable to any region sets, no matter they are adjacent in time or not. Matching results can be categorized into five different types, and their relationship to region transformations is as follows. For the ease of explanation, we use s_1 and s_2 to denote the former and latter set of regions, respectively. Region r_1 belongs to s_1 and r_2 is in s_2.

- ***One-to-Zero***: no regions in s_2 match with r_1. r_1 *disappears*;
- ***Zero-to-One***: no regions in s_1 match with r_2. r_2 *appears*;
- ***One-to-One***: only r_1 matches r_2 in s_1 and vice versa. The region territory may *expand*, *shrink* or *remain*;
- ***One-to-Multi***: r_1 matches with more than one region in s_2. r_1 is *split* into multiple regions;
- ***Multi-to-One***: r_2 matches with more than one region in s_1. Multiple regions *merge* into r_2.

6. Evolution Patterns

We create migration trajectories based on the temporal matching results, which can be obtained by repeatedly applying matching operations to regions at every two consecutive time steps. A trajectory represents the connections between centroids of a region, but embodies no information about the differences in size and function.

Many regions often undergo similar transitions of evolution. This can be exposed via representative evolution patterns by abstracting trajectories as graphs. Region centroids are then mapped to graph vertexes, which are stamped by their time steps. Graph edges

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

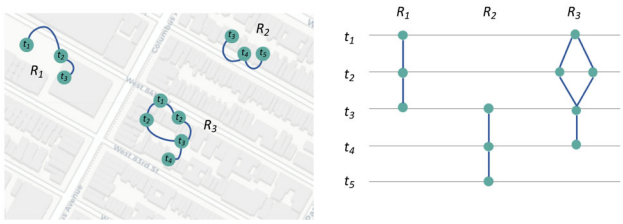


Figure 3: *Conversion from migration trajectories to graphs. Nodes represent region centroids and they are stamped with time steps.*

indicate the original connections between centroids. Figure 3 shows an example. R_1 and R_2 have only *one-to-one* transformations, hence their corresponding graphs have the same topology. A *split* and a *merge* transformation occur to R_3, so there is a cycle in the graph.

Graph abstraction simplifies trajectories by ignoring details, such as the spatial location of centroids and the distance between them. Then, we can easily find regions that undergo similar evolution procedures by clustering abstract graphs based on topology. Netsimile [BKEF12] is an approach for quantifying the similarity between size-independent graphs. It firstly converts graphs into signature vectors by aggregating topological attributes of vertexes, including degree, clustering coefficient, etc. Then, we can apply a standard clustering method (e.g. *k*-means) on these vectors with the cosine distance. Evolution patterns are revealed by randomly choosing a representative graph from each cluster.

7. Visualization

We build a visual system to facilitate the exploration of region evolution. The system consists of an *overall view,* where users can access details of venues, like their descriptions and distributions, and a *comparison view* that displays results of dynamic region segmentation.

7.1. Requirement analysis

Our region delimitation method is based on discrete venues, along with their contextual information. Therefore, we focus on representing both low-level (i.e. venues) and high-level (i.e. regions) geographical elements. Their visualization requirements are as follows:

- ***Venues***: mark their locations on the map, and allow users to interactively select venues and investigate the meta data, such as time, photos, comments, etc.
- ***Regions***: improve aesthetics of the representations, and make regions visually comparable.

Based on the visual task model proposed by Andrienko *et al.* [AAG03], our system provides *static* and *dynamic* information. At a specific time step, static information includes:

- ***Venue distribution*** (**S1**): Which part has a higher density of venues within the focal area? Where is the centroid of venues?

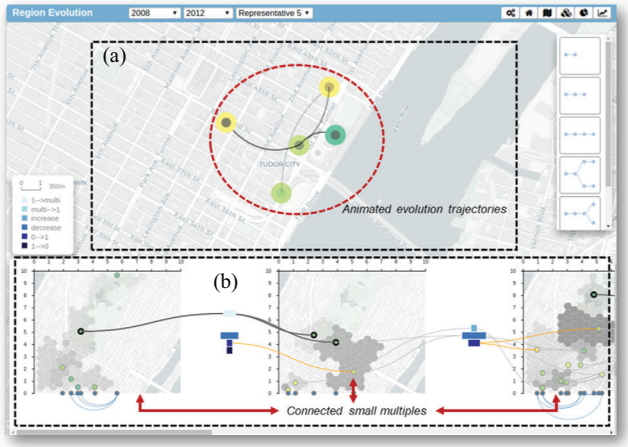


Figure 4: *Interface of the visual system. (a) Overall view plays the animation of region evolution; (b) comparison view contains snapshots of region distributions in an AOI.*

- ***Region distribution*** (**S2**): How many regions does the focal area possess? How large is the territory of a region? What is the relationship between the region segmentation and the venue distribution?
- ***Region semantics*** (**S3**): What is the entropy level of region topics? Does a region have an explicit function? What are the regions that have similar functions?

During a period, dynamic information includes:

- ***Territory transformation*** (**S4**): How does a region territory change over time? Does it remain stable, become larger/smaller or even split/merge into other regions?
- ***Semantic transition*** (**S5**): What is the difference of topic distribution between consecutive time steps, regarding individual regions?
- ***Area dynamics*** (**S6**): How many regions contribute to each type of transformations, within an area of interest?

7.2. Visual design

Figure 4 shows the interface of the visual system. We have two collaborating views, each characterized by animations and small-multiple representations. The *overall view* provides both static and dynamic information at a global level. It plays animations of the movement of region centroids. Animations are effective in solving the scalability issues of visualizing temporal data. However, users may suffer from visual clutter due to trajectory crossings [LBH18]. Also, it is difficult to show region territories at different time steps on the same map because of visual overlaps. Therefore, we allow users to select an AOI and perform further explorations in the *comparison view*. A sequence of snapshots is juxtaposed along the timeline, each exposing region details at a specific time step. Adjacent snapshots are connected by links that demonstrate the matching relationship between regions.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

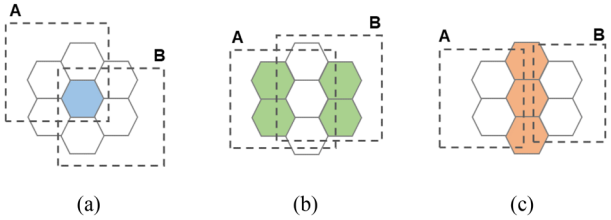


Figure 5: *Controversial situations of assigning cells at the borders of regions. (a) The hexagon (blue) is inside both regions A and B; (b) hexagons (green) are inside one region and intersect with another; (c) hexagons (orange) intersect with both regions A and B.*

7.2.1. *Overall view*

In this view, we provide utilities for users to access low-level static information. A scatter plot is used to display the geographical distribution of discrete venues and a heatmap shows the density distribution. Our concerns of visualizing high-level information are two-fold: first, what is the aesthetic way to illustrate regions that are derived by applying the delimitation method introduced in Section 4; second, how to make region evolution straightforward to users.

Using convex hulls to represent regions is not a good option, because their irregular shapes are hard to compare and they might overlap slightly at borders. The overlapping is caused by peripheral venues, and we later measure their importance to decide if they should be treated as outliers. In this work, we adopt the tiling techniques [MH17, CBC*15] to convert convex polygons to hexagonal grids. Besides, we achieve non-overlapping representations by exclusively assigning grid cells to a region.

Region tiling facilitates visual comparisons by making the territory size countable, but it is difficult to preserve the boundary of convex polygons and minimize the time cost of tiling. In general, smaller hexagons can be aggregated into arbitrary polygon shapes while maintaining uniform convexity internally and the topic entropy in one cell is lower. But the time cost may increase as more hexagons need to be calculated for the same region.

The primary operation of tiling is using the ScanLine method [WREE67] to determine if a cell completely resides in a region polygon. The procedure can be accelerated by indexing grid cells with an r-tree. Meanwhile, we have to deal with some ambiguous situations at boundaries. As shown in Figure 5, cells might intersect with or belong to multiple polygons simultaneously. Let $\mathcal{R}$ be the set of regions involved with a cell c. For $r \in \mathcal{R}$, v_{rc} is the set of venues inside the overlapping area of r and c. $|r|$ is the total number of venues in r, and c_r is the centroid of r; $dist(\cdot)$ returns the distance between two venues. The contentious cell c will be allocated to the region which satisfies:

$$r = \arg\max_{r \in \mathcal{R}} \frac{1}{|r|} \sum_{v \in v_{rc}} \frac{1}{1 + dist(v, c_r)}. \quad (5)$$

Equation (5) implies the impact that venues inside the overlapping area have on the whole region. If the impact is negligible, we skip c as a part of the tiling. Also, for reducing holes in the tiling, we prefer to assign c to regions that totally cover rather then intersect with c.

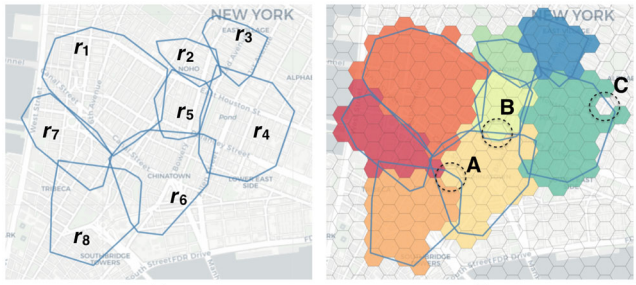


Figure 6: *Region tiling with hexagonal cells. (a) Polygonal region representations; (b) hexagonal tiling. Different regions are distinguished by using a ColorBrewer platte [HB03].*

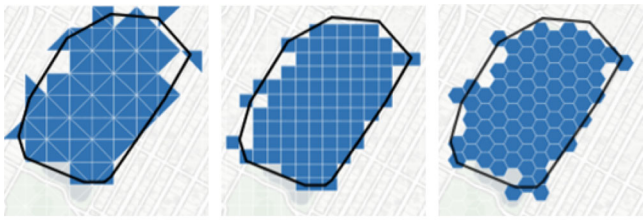


Figure 7: *Tiling a region by (a) triangles, (b) squares and (c) hexagons.*

Figure 6(a) shows an example of the original polygonal representations of regions, and in Figure 6(b), we can see the tiling result based on the strategy defined in Equation (5). It is worthy noting that, in Figure 6(b), cell **A** is inside r_6 and intersects with r_8, and it is finally allocated to r_6. Cell **B** intersects with both r_5 and r_6, and it finally goes to r_5. Cell **C** intersects with r_4, but its inner venues have weak impact on r_4, hence it is discarded.

Alternative geometries include squares and triangles. In Figure 7, we compare the three types of tiling given an identical region. Single cells of each grid have the same size, and the ratio of the side length of cells is approximately 24:15:10. According to Chen *et al.* [CCW*16], the triangle grid introduces visual ambiguities, because it consists of triangles oriented at two directions. The square grid brings illusions of stretching along the vertical and horizontal levels. Hence, we choose the hexagonal grid for obtaining better aesthetics and visual compactness.

An animation that shows the movement of region centroids helps users to understand the region evolution. We take centroids as the delegates of regions, because directly drawing region territories causes visual overlaps on a single map, like the example in Figure 8. Users are still allowed to inspect the underlying territory by clicking a centroid. We map the chronological order to the opacity of inner circles of centroids and the curved links. The higher the opacity, the closer the timestamp of a centroid to the current time. Alternatively, arrows can represent the order. But when centroids are close to each

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

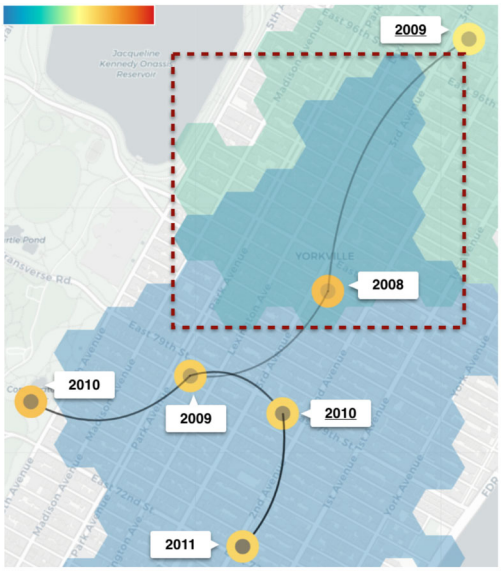


Figure 8: *The migration trajectory of a region. Each centroid is denoted by two concentric circles. The opacity of inner circles indicates the chronological order and the colour of outer circles reflects the entropy level of region topics. Territories in the years 2009 and 2010 are represented by blue and green hexagonal grids, respectively, and their overlapping area is marked by a dashed rectangle.*

other, the links are short and inserting arrows to such kinds of links aggravates visual clutters. The default colour of inner circles and curved links is *black*. Outer circles of centroids are filled according to the linear gradient of colour shown in the top-left corner of Figure 8. The warmer the colour, the higher the topic entropy of the region in all regions.

7.2.2. *Comparison view*

In the *overall view*, users can select a rectangular AOI by dragging. Then, they can access and compare the details of multiple snapshots simultaneously in the *comparison view*, without being interrupted by visual clutters caused by territory overlaps. Figure 9 shows an example of the *comparison view* that contains two snapshots, each representing the region delimitation result at a specific time.

We use various visual designs to encode the statistics of venues and regions. Thick lines highlight the boundaries of regions. The opacity of region fillings indicates the number of inner venues, and the default colour is *black*. The darker the filling, the more venues the region contains. Heatmaps located at the horizontal and vertical axes reflect the distribution of venues along the longitude (*green*) and latitude (*magenta*) directions, respectively. Besides, the graduated axes are helpful in calculating the migration distances. Users can switch between different regions by clicking their centroids, which are drawn as small circles. The circle fillings reveal the entropy level of region topics and they follow the same colour scheme adopted in Figure 8. An arc diagram at the bottom of each snapshot depicts the similarity of topic distributions between each pair of regions. Nodes in the diagram are projections of centroids to the bottom axis. The

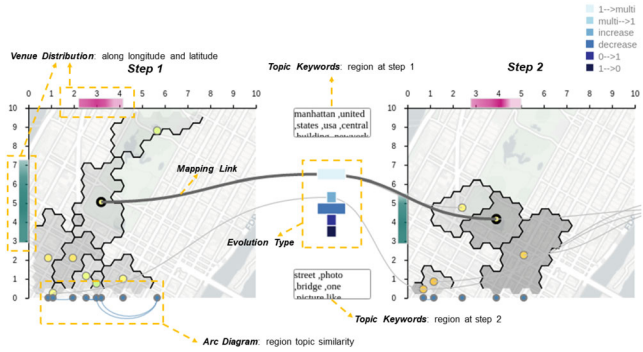


Figure 9: *Two snapshots from the comparison view, each displaying the distribution of regions at a specific timestamp. Snapshots are connected by links showing the matching results of regions.*

range of similarity values is [0,1], and users can decide the number of visible arcs by controlling the threshold.

Adjacent snapshots are connected by links whose end points are centroids of matched regions. Then, it becomes convenient to track the temporal transformation of regions. Links are categorized according to different types of matching results defined in Section 5. Among them, the *One-to-One* type is divided into *increase* and *decrease* categories, indicating the size change of regions. The six categories are denoted by stacked rectangles whose width is proportional to the number of regions belonging to that category. We note that a link is invisible if any of its end centroids goes out of the AOI. When clicking on a link, it is highlighted with a thicker stroke. Meanwhile, the keywords of topics of two end regions are listed separately in the boxes that are located at the top and bottom space between snapshots.

8. Evaluation

We demonstrate the effectiveness of our methods by presenting two application cases and a user study. All experiments were conducted on a desktop machine with 2.20GHz, Intel Core i5 CPU, 16.0 GB RAM and 1680×1050 pixel resolution. The visual system was built by using JavaScript and D3, and interactions on the map were supported by Leaflet.

8.1. Case 1: Flickr

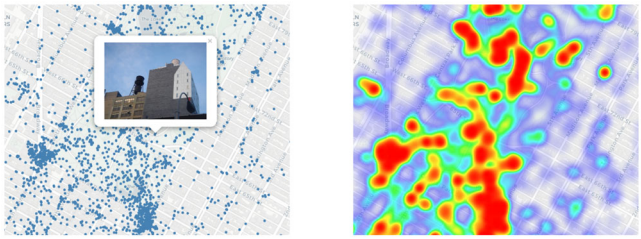
YFCC100M dataset [TSF*16] contains records of photos or videos uploaded by Flickr users during the years 2004–2014. In this case, we focused on the period from the year 2008 to 2012 and filtered out the records whose textual descriptions and geo-locations were both available. The focal area was bounded to New York City.

Data items were manipulated to fit the format that we defined in Section 4.1. We then aggregated items based on the time interval of 1 year. By doing experiments with different k values, we decided to connect each venue to its 20 nearest neighbours. Consequently, region overlaps were minimized. Statistic samples of the venues and regions can be found in Table 1.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

Table 1: *For the Flickr dataset, given the number of venues ($|V|$) for each year from the year 2008 to 2011, we show the ratio of the number of edges of the latent graph to the number of regions ($|E|/|R|$) for each of the three values of* k*=5, 10 and 20.*

	2008	2009	2010	2011
$\|V\|$	26 784	30 771	35 459	43 321
k=5	85 061/384	97 517/465	112 520/502	137 007/586
k=10	166 696/168	190 491/187	220 187/214	268 422/228
k=20	329 901/99	377 000/112	435 585/124	531 500/137



(a) (b)

Figure 10: *Within an area of NYC in the year 2008, (a) a scatter plot in which blue circles denote venues, and (b) the corresponding heatmap that reflects the popularity of different places.*

We allowed users to view the information of individual venues. Figure 10(a) is a scatter plot of venues. When clicking on a venue, a popup window displays the corresponding media object. Figure 10(b) shows a heatmap of the venue distribution. The warmer the colour, the denser the venues. We found that most of the blocks in New York City were crowded with Flickr users, especially in midtown and downtown (**S1**).

One significant discovery was that our region delimitation procedure successfully located landmarks in New York City (**S3**). The LDA model extracted 12 topics at each time step. With the help of topic keywords listed in Table 2, we identified regions covering landmarks, such as Times Square and The Empire State Building. The result is shown in Figure 11. These regions were detected mostly because people tended to add the name of landmarks in textual descriptions. Most of the corresponding evolution trajectories only involved with *One-to-One* transformations. However, the trajectory of the Central Park region experienced a few times of split. An explanation could be that, within the vast area, people were more likely to move around and gather at different locations. Based on the

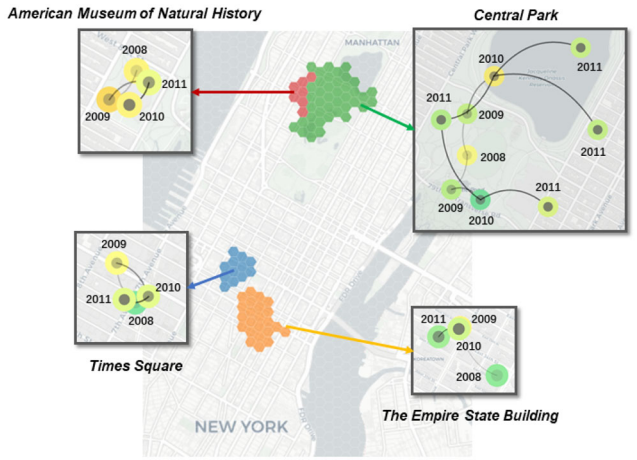


Figure 11: *Four landmark regions in the year 2008. Their evolution trajectories are shown in the enlarged rectangles.*

topic keywords, we also inferred that the assembly locations could be the baseball fields. We then confirmed the inference by *Google Maps* API.

We set the threshold in Algorithm 1 to 0.6 to get a balanced distribution of different types of transformations. Some regions remained stable across time. Their evolution trajectories did not embody *split* or *merge* transformations. Besides, region centroids shifted within a limited area (**S4**), and the entropy of the topic distributions remained at a low level. We conclude that such regions provide constant utilities and services, like the Museum in Figure 11. Conversely, regions without an explicit function were potentially more dynamic and their evolution trajectories contained more types of transformations. By clustering the trajectories, we revealed the evolution patterns and the representatives, which are shown in Figure 12. Patterns 1–3 vary in duration. In fact, many regions appeared at only one timestamp, and the trajectory graph contained only a single node. Figure 12 also displays the spatial distribution of regions that match the patterns 3–5 (**S2**). In this case, a majority of regions contributed to pattern 3. Regions that were assigned pattern 4 were more likely to be detected around transportation hubs. A minority of regions went through a combination of transformations, and their evolution trajectories did not match with any of the five patterns, such as the Central Park region shown in Figure 11.

We selected an AOI with diagonal corners at $[-74.0012, 40.7822]$ and $[-73.9599, 40.7562]$. The radius of hexagonal cells was set to 100 m, and the unit of longitude and latitude axes was set to 350 m.

Table 2: *Keywords of the dominating topics in four landmark regions.*

Topic index	Keywords	Landmark
t_{10}(41%)	brooklyn, times, bridge, east, art, street, square, manhattan	Times Square
t_4(23%)	st, college, years, day, natural, fair, benoit, amnh, history, american, parade, museum	American Museum of Natural History
t_1(52%)	central, building, park, baseball, stadium, manhattan, city, states, yankee	Central Park
t_{12}(44%)	street, manhattan, state, building, center, avenue, empire, governors, nuev, apple, big	The Empire State Building

The first column lists the index and the probability of topics.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

Figure 12: *Representative patterns of region evolution. Three map views show the spatial distribution of regions that contribute to patterns 3–5, respectively.*

The *comparison view* shown in Figure 13 contains four snapshots from the year 2008 to 2012. Between the first two snapshots, thick links represent a *split* transformation from one region to two regions. The stacked rectangles in the middle space indicate that no regions involve with *merge* and *increase* transformations (**S6**). In fact, the *merge* transformation is missing during the whole period. The *orange* link suggests that a region appears, and it is invisible by default because there is no matched centroid in the first snapshot. Users can inspect the hidden links by clicking on rectangles. Among the last three snapshots, thick links demonstrate the evolution of regions in Central Park. From the opacity of fillings, we can see that the popularity of regions does not vary dramatically. The *split* transformation happens twice. One descendant region is beyond the range of AOI in the fourth snapshot. In the last snapshot, the two descendant regions have similar topics and possess a communal segment of boundary. The reason of their separation is probably because region centroids are far from each other, and few venues locate at the periphery. We can confirm the inference of venue distributions by observing heatmaps along the graduated axes (**S1**).

8.2. Case 2: Yelp

We collected the reviews of businesses in Las Vegas from *Yelp Dataset Challenge* [YEL17]. Data items went through a preprocessing stage. The time interval was 1 month. Table 3 lists statistic samples of the dataset.

Table 3: *The number of venues ($|V|$) in the Yelp dataset.*

	January	February	March	April
$\|V\|$	6802	6792	7249	7199
k=5	21 709/156	21 581/148	23 211/162	22 948/155
k=15	63 230/57	63 283/57	67 541/54	66 999/57
k=25 ($dist_{nbr} < \infty$)	105 062/38	104 928/41	112 044/40	111 416/40
k=25 ($dist_{nbr} < 1.5km$)	100 949/57	100 962/55	107 728/56	107 368/53

When k=5, 15, 25, the ratio of the number of edges of latent graphs to the number of regions ($|E|/|R|$). $dist_{nbr}$ denotes the distance between connected venues.

There are two notable distinctions between the Yelp and Flickr datasets. First, the temporal distribution of venues in the Yelp dataset is more stable. Because venues in the Flickr dataset are arbitrary places where people may take photos on a whim. However, in this case, business venues are fixed. The difference between venues at different timestamps mainly reflects the establishment of new businesses or the termination of old businesses. Second, the Yelp dataset has less topic varieties as the comments are all about the businesses. Therefore, regions detected in this case are supposed to have stable territories and consistent topic themes over time.

We tested different k values when constructing latent graphs and the final value was 25. Besides, we compared the region delimitation results when the distance between connected venues was or was not constrained. The heatmap in Figure 14 shows the uneven distribution of businesses. We can see that businesses are only crowded in two areas (red colour). If there was no distance limitation, venues in sparse areas would find neighbours in far places. Then, we would obtain large regions in low-density areas, such as R_1 and R_2 in Figure 15(a). When we set the threshold to 1500 m, the number of edges in latent graphs decreased as shown in the last row of Table 3. In addition, we eliminated the impact of isolated venues that were marked by rectangles in Figure 15(b). Tiny regions were ignored. We also found that the distance limitation did not drastically affect the region delimitation in high-density areas marked by ellipses in Figure 15.

We compared the region evolution of two AOIs. One was the high-density area B in Figure 14. It was located at the centre of

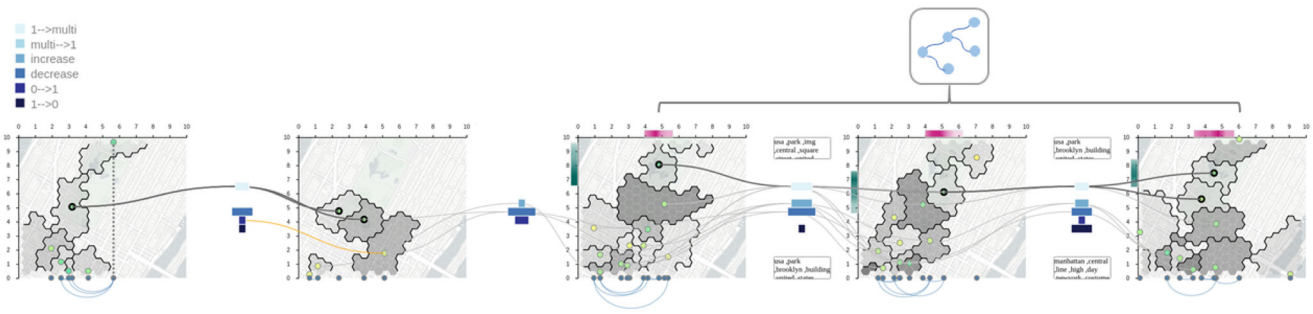


Figure 13: *Five snapshots of an AOI in NYC. The pattern graph depicts the evolution of regions in Central Park.*

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

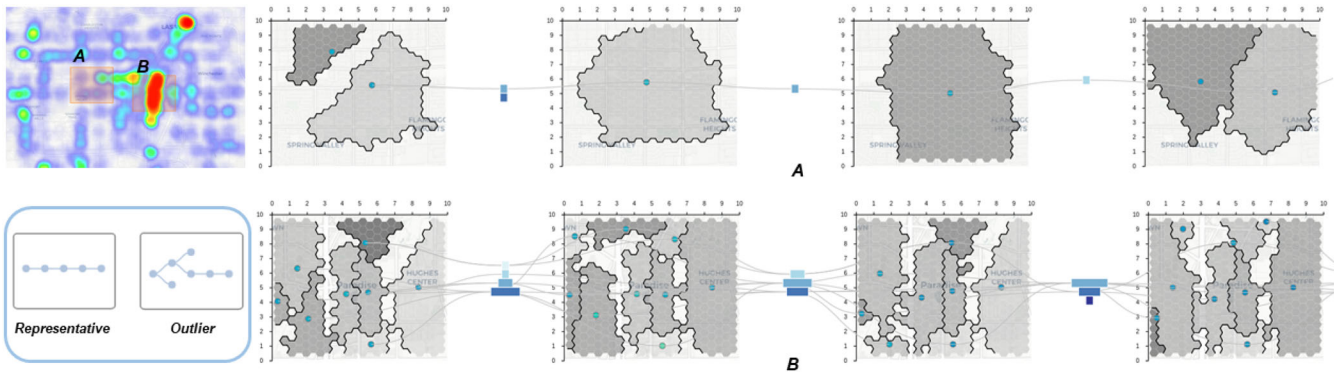
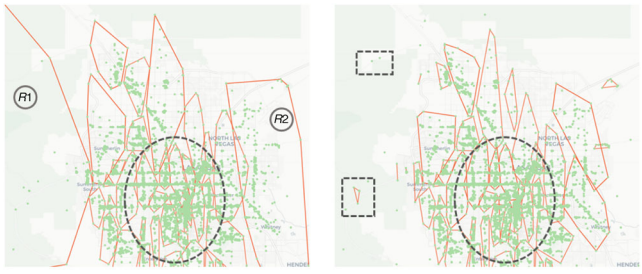


Figure 14: A *and* B *are AOIs of the same size. Venues are sparse in* A *and dense in* B. *Snapshots of* A *and* B *from January to April are displayed in the up and bottom rows, respectively.*

Las Vegas. The other one was the low-density area *A*. *A* and *B* had the same size. Their snapshots at four timestamps are displayed. From the opacity of region fillings, we see that most of the regions in *B* encompass more venues than regions in *A*, even though they have smaller territories. Besides, there are more regions in *B* and the stacked rectangles indicate that these regions involve with higher dynamics that are caused by various types of transformations. However, regions in *A* are relatively stable except the size changes.

A majority of regions in Las Vegas had the same evolution pattern as the representative shown in Figure 14. It reflected the stable distribution of venues and regions. However, outlier patterns, such as the one in Figure 14, were found especially at high-density areas like *B*. Frequent *split* and *merge* transformations occurred to the corresponding regions. Venue descriptions had similar semantics because they were about the characteristics of businesses, such as the quality of service. Hence, edges of individual latent graphs had similar weights. Region delimitation depended more on the closeness of venues than the proximity of semantics (**S2**). The boxplot in Figure 16 represents the entropy of region topics. We extracted six topics at each time step. When comparing region topics in the Yelp and Flickr cases, the former has lower entropy levels and smaller variation ranges of entropy values. It implies that Yelp regions have more explicit and consistent topics, while Flickr regions possess multiple topics and none of them plays a dominant role.



(a) (b)

Figure 15: *Convex polygons (red) represent regions that are detected when the distance between connected venues ($dist_{nbr}$) is: (a) not limited; (b) shorter than 1500 m.*

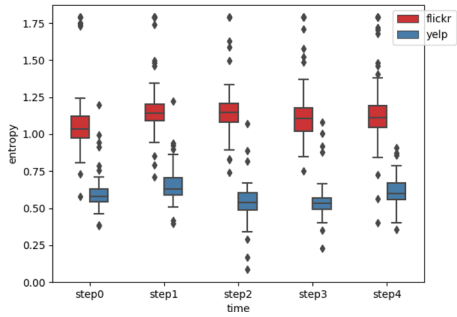


Figure 16: *Statistics of topic entropy among all regions. Yelp regions have lower level entropy, implying that they embody more explicit topics.*

Table 4 lists the time cost of the main procedures in our method. Areas considered for the Flickr and Yelp cases are NYC and Las Vegas, respectively. The graph construction procedure includes calculations of *k* nearest neighbours and semantic similarities between venues. The second column shows the running time of our delimitation algorithm. The most time-consuming part of tiling is generating a hexagonal grid for the whole city area, and the tool we use is Turf.

8.3. User study

Our objective is to compare users' performance on conducting the same visual tasks in two views (*view 1: the overall view*; *view 2: the comparison view*). We invited 12 users to attend the study. One of them was a male expert of geographical information systems. Two female graduate students had knowledge about visual analytics, and the remaining users were undergraduate students (six males) with diverse majors.

Users were given a standard laptop (1.1GHz Intel Core M3 CPU and 8GB memory) and the system interface was presented in the Google's Chrome browser. Animations in *view 1* were played region by region. The playing time was not recorded. Users could start a replay if needed, and they were allowed to take simple notes during the test.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

Table 4: *Among all time steps, the average time cost of main procedures (s).*

	Graph construction	Region delimitation	Matching	Tiling
Yelp	0.03	0.44	1.01	0.21
Flickr	0.21	2.37	2.49	0.83

The average number of vertexes and links of latent graphs are 7,000 and 108,360 in the Yelp case, and 34,002 and 418,370 in the Flickr case.

Table 5: *Visual tasks that need to be conducted in both the overall view and comparison view.*

T_1	What is the duration of a region?
T_2	What kind of transformations does a region involve?
T_3	Which region experiences only *one-to-one* transformations?
T_4	Which region has its centroids move across the largest area?
T_5	How many regions does the AOI consist of at a time step?
T_6	Which region has the largest territory at a time step?
T_7	Which region has a higher venue density at a time step?
T_8	How many new regions appear at a time step?
T_9	How many regions shrink between two time steps?
T_{10}	How many regions split between two time steps?

At the beginning, we gave a brief tutorial on the visual designs and the usage of the system. Then, users conducted the visual tasks listed in Table 5. The study was based on experiments of both Flickr and Yelp datasets, and we focused on region B in the Yelp case.

Each user needed to complete 40 tasks (10 for each view and 20 for each dataset). We designed the tasks according to the requirements introduced in Section 7.1. For a pair of inverse transformations, we chose one of them as a representative. Therefore, alternative tasks for T_8-T_{10} could be: *How many regions disappear/expand/merge between two time steps?* Considering that the impression on one view will affect the answers of another view, we asked users to finish all tasks for both datasets in *view 1*, and then repeat the procedure in *view 2*. We anticipated that users' performance would be different in two views but similar for two datasets.

Results. We recorded the accuracy and time cost of each task, and the results are presented in Table 6. We find that T_1-T_4 have lower accuracy and longer response time in *view 2*, and the same applies to T_6–T_9 in *view 1*. The reason might be that T_1–T_4 require users to have an overall understanding of the evolution. *view 1* outperforms *view 2* in these tasks because animations will not be affected by the growth of time. However, in timeline representations, we have to view long-term information by scrolling which causes interruptions in memory and extra time expenses. T_6–T_9 associate with comparisons of region territories. In *view 1*, some users provided wrong answers as they did comparisons by observation. Others compared the number of hexagonal cells and obtained accurate answers. In *view 2*, tasks can be easily completed by inspecting the width of the stacked rectangles.

Feedback. All users thought the system was helpful. It was easy to use and the two views complemented each other. Some undergraduate students said that, even if they had little experience in urban analysis, they could quickly understand what the system presented and start exploring interesting patterns. The graduate students liked the aesthetic representations and the interactive features. They also agreed that links between snapshots were essential as they led the movement of eyes when tracking the evolution of regions. The expert commented our system like this: *I can clearly see the difference between region distributions at different time steps, and the categorization of region transformations is especially helpful.* He also pointed out that it took some time to figure out the usage of widgets. We then attached hint texts to these widgets.

9. Discussion and Limitations

The studies have shown that our methods are effective in discovering dynamic regions, and the visual system is helpful in assisting users with exploration tasks. However, there are limitations. First, users need to choose a sufficient k for constructing latent graphs.

Table 6: *The accuracy and the average time cost of visual tasks.*

	Accuracy				Time (s)			
	Flickr		*Yelp*		*Flickr*		*Yelp*	
	view 1	*view 2*	*view 1*	*view 2*	*view 1*	*view 2*	*view 1*	*view 2*
T_1	**100%**	75%	**91.7%**	75%	**5.5**	12.3	**4.8**	10.6
T_2	**83.3%**	75%	**91.7%**	83.3%	**6.1**	13.5	**5.9**	11.2
T_3	**100%**	83.3%	**100%**	66.7%	**4.8**	19.9	**4.7**	20.3
T_4	**91.7%**	41.7%	**83.3%**	33.3%	**5.6**	36.5	**6.4**	39.8
T_5	100%	100%	91.7%	**100%**	8.3	**5.4**	8.8	**5.2**
T_6	50%	**100%**	33.3%	**83.3%**	38.6	**5.5**	43.2	**8.1**
T_7	58.3%	**91.7%**	58.3%	**83.3%**	7.5	**4.6**	8.8	**5.4**
T_8	91.7%	**100%**	100%	100%	6.2	**4.2**	6.3	**4.5**
T_9	41.7%	**91.7%**	41.7%	**100%**	55.7	**4.2**	64.5	**4.1**
T_{10}	100%	100%	91.7%	**100%**	5.8	**4.3**	7.9	**5.7**

The better performance achieved from the two views is highlighted.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

As we mentioned in Section 4.3, a larger k corresponds to less regions with more overlaps. Conversely, we obtain a greater portion of tiny regions (e.g. the area is less than 100 m^2) with a smaller k. To decide on a value, we can progressively increase k until the number of tiny regions (R_t) and the size of the overlapping areas (A_o) are both minimized, but the calculations are time-consuming. Therefore, we provide several k values that have achieved good results in our experiments, and their corresponding R_t and A_o on the current datasets have been calculated. Users can compare them and select the most satisfying one. Second, we need to determine the number of topics for the LDA model. The hierarchical Dirichlet process [TJBB06] is helpful, but it might be infeasible on large corpuses. Similar to the way of k selection, we train models on varying number of topics, and choose the one with the best performance. A model works better if it causes a lower perplexity on a test set of documents. Third, there is no guarantee that venues belonging to the same region are mutually accessible. For the ease of data processing, we use the Eclidean distance to measure the geographical proximity of venues. However, a topological network-based distance, such as the length of the shortest path on a street network, would be a better option.

The reason that we propose a graph-based method rather than using the traditional density-based methods like DBSCAN to cluster venues is two-fold. First, we need to consider two similarity metrics at the same time, one is in geographical space and the other is in semantic space. In the graph-based method, we can conveniently map them to the connectivity of venues and the weight of connections, respectively. Whereas in DBSCAN, we have to design a distance measure that combines the two metrics, like the implementation of ST-DBSCAN [BK07]. Second, DBSCAN requires two parameters, ϵ (the maximum distance between points in a cluster) and $MinPts$ (the minimum number of points required to form a cluster) [EKSX96]. It is non-trivial to estimate them. In our method, we only need to determine a k value for constructing graphs, and the frequently adopted values, such as 10 and 20, suit most of the scenarios in our tests.

Besides, we can extend our work on revealing evolution patterns by abstracting migration trajectories as attributed graphs. For example, node attributes may include the number of venues, the entropy of topics and edge attributes can be the distance between region centroids. By applying clustering methods for attributed graphs [XKW*12], we are able to find regions that have similar evolution procedures more precisely.

Our method is applicable to the analysis of any spatio-temporal data, e. g. telecommunication records and commuting flows. Such kind of data might be inherently related, and their connection weights should be measured differently, rather than by calculating the semantic similarity of topics. For example, the vertexes of graphs that are constructed from commuting flows can be the departure and destination venues of commuters, and edges can be the routes connecting them. The edge weight denotes the number of trips between two end-venues [VLBR*16]. Regions that are detected from the graphs are not necessarily areas that provide specific functions, but areas that cover a group of closely related venues and also reflect a high frequency of human activities. Our matching algorithm can expose the popularity of regions at different time and the migration of activity centres.

10. Conclusion

We propose a new visual system for analysing the evolution of geographical regions. It consists of two interactive visualization views, one for displaying the animated region migrations and the other for showing the differences and connections between a sequence of snapshots. Compared to conventional methods of urban analysis, we leverage dynamic geo-textual data and update region divisions by employing a graph-based method. Furthermore, the matching algorithm aims to identify different types of region transformations. Users can further explore whether regions have similar evolution patterns. We have conducted two case studies and a user study on real datasets. The results show that the system can effectively assist users in analysing dynamic features of regions. In the future, we plan to enhance the visual system by automating the selection of parameter values and also apply our methods to exploring more geographical data. Besides, we will enrich interactive features to promote analysis.

Acknowledgements

The authors would like to acknowledge the partial support of IGRF PolyU 152142/15E and Project 4-ZZFF from the Department of Computing, The Hong Kong Polytechnic University. This work was partially supported by the National Natural Science Foundation of China under Grants (No. 61802128).

References

[AAG03] Andrienko N., Andrienko G., Gatalsky P.: Exploratory spatio-temporal visualization: An analytical review. *Journal of Visual Languages & Computing 14*, 6 (2003), 503–541.

[ADWK*16] Al-Dohuki S., Wu Y., Kamw F., Yang J., Li X., Zhao Y., Ye X., Chen W., Ma C., Wang F.: Semantictraj: A new approach to interacting with massive taxi trajectories. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (2016), 11–20.

[AKK18] Atluri G., Karpatne A., Kumar V.: Spatio-temporal data mining: A survey of problems and methods. *ACM Computing Surveys (CSUR) 51*, 4 (2018), 83.

[BBDW17] Beck F., Burch M., Diehl S., Weiskopf D.: A taxonomy and survey of dynamic graph visualization. *Computer Graphics Forum 36*, 1 (2017), 133–159.

[BDA*14] Bach B., Dragicevic P., Archambault D., Hurter C., Carpendale S.: A review of temporal data visualizations based on space-time cube operations. In *Eurographics Conference on Visualization* (2014).

[BGLL08] Blondel V. D., Guillaume J.-L., Lambiotte R., Lefebvre E.: Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment 2008*, 10 (2008), P10008.

[BK07] Birant D., Kut A.: St-dbscan: An algorithm for clustering spatial–temporal data. *Data & Knowledge Engineering 60*, 1 (2007), 208–221.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

[BKEF12] Berlingerio M., Koutra D., Eliassi T., Faloutsos C.: Netsimile: A scalable approach to size-independent network similarity. *CoRR abs/1209.2684* (2012).

[BPF14] Bach B., Pietriga E., Fekete J.-D.: Graphdiaries: Animated transitions and temporal navigation for dynamic networks. *IEEE Transactions on Visualization and Computer Graphics 20*, 5 (2014), 740–754.

[CBC*15] Cano R. G., Buchin K., Castermans T., Pieterse A., Sonke W., Speckmann B.: Mosaic drawings and cartograms. *Computer Graphics Forum 34*, 3 (2015), 361–370.

[CCC13] Chen L., Cong G., Cao X.: An efficient query indexing mechanism for filtering geo-textual data. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data* (2013), ACM, pp. 749–760.

[CCW*16] Chen S., Chen S., Wang Z., Liang J., Yuan X., Cao N., Wu Y.: D-map: Visual analysis of ego-centric information diffusion patterns in social media. In *2016 IEEE Conference on Visual Analytics Science and Technology (VAST)* (2016), IEEE, pp. 41–50.

[CFZ16] Cong G., Feng K., Zhao K.: Querying and mining geo-textual data for exploration: Challenges and opportunities. In *2016 IEEE 32nd International Conference on Data Engineering Workshops (ICDEW)* (2016), IEEE, pp. 165–168.

[CSCC05] Capocci A., Servedio V. D., Caldarelli G., Colaiori F.: Detecting communities in large networks. *Physica A: Statistical Mechanics and its Applications 352*, *2–4* (2005), 669–676.

[CSHS12] Cranshaw J., Schwartz R., Hong J. I., Sadeh N. M.: The livehoods project: Utilizing social media to understand the dynamics of a city. In *ICWSM* (2012).

[CYW*16] Chen S., Yuan X., Wang Z., Guo C., Liang J., Wang Z., Zhang X. L., Zhang J.: Interactive visual discovering of movement patterns from sparsely sampled geo-tagged social media data. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 270–279.

[EKSX96] Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, KDD'96 (1996), AAAI Press, pp. 226–231.

[FOR10] FORTUnato S.: Community detection in graphs. *Physics Reports 486*, *3–5* (2010), 75–174.

[FPV*13] Ferreira N., Poco J., Vo H. T., Freire J., Silva C. T.: Visual exploration of big spatio-temporal urban data: A study of N York City taxi trips. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2149–2158.

[HB03] Harrower M., Brewer C. A.: Colorbrewer. org: An online tool for selecting colour schemes for maps. *Cartographic Journal 40*, 1 (2003), 27–37.

[HW04] Hlaoui A., Wang S.: A direct approach to graph clustering. *Neural Networks and Computational Intelligence 4*, 8 (2004), 158–163.

[HZM*16] Huang X., Zhao Y., Ma C., Yang J., Ye X., Zhang C.: Trajgraph: A graph-based visual analytics approach to studying urban network centralities using taxi trajectory data. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 160–169.

[KCW*17] Karduni A., Cho I., Wessel G., Ribarsky W., Sauda E., Dou W.: Urban Space Explorer: A visual analytics system for urban planning. *IEEE Computer Graphics and Applications 37*, 5 (2017), 50–60.

[KKK*10] Kisilevich S., Krstajic M., Keim D., Andrienko N., Andrienko G.: Event-based analysis of people's activities and behavior using Flickr and Panoramio geotagged photo collections. In *2010 14th International Conference on Information Visualisation (IV)* (2010), IEEE, pp. 289–296.

[KP12] Kling F., Pozdnoukhov A.: When a city tells a story: Urban topic analysis. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems* (2012), ACM, pp. 482–485.

[LBH18] Li C., Baciu G., Han Y.: Streammap: Smooth dynamic visualization of high-density streaming points. *IEEE Transactions on Visualization and Computer Graphics 24*, 3 (March 2018), 1381–1393.

[LBWZ18] Li C., Baciu G., Wang Y., Zhang X.: Fast content-aware resizing of multi-layer information visualization via adaptive triangulation. *Journal of Visual Languages & Computing 45* (2018), 61–73.

[LF09] Lancichinetti A., Fortunato S.: Community detection algorithms: A comparative analysis. *Physical Review E 80*, 5 (2009), 056117.

[LFYX13] Liu B., Fu Y., Yao Z., Xiong H.: Learning geographical preferences for point-of-interest recommendation. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2013), ACM, pp. 1043–1051.

[LL16] Lansley G., Longley P. A.: The geography of twitter topics in london. *Computers, Environment and Urban Systems 58* (2016), 85–96.

[LWL*16] Liu D., Weng D., Li Y., Bao J., Zheng Y., Qu H., Wu Y.: Smartadp: Visual analytics of large-scale taxi trajectories for selecting billboard locations. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (2016), 1–10.

[LWL*17] Liu D., Weng D., Li Y., Bao J., Zheng Y., Qu H., Wu Y.: Smartadp: Visual analytics of large-scale taxi trajectories for selecting billboard locations. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (2017), 1–10.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd

[MELS95] Misue K., Eades P., Lai W., Sugiyama K.: Layout adjustment and the mental map. *Journal of Visual Languages & Computing 6*, 2 (1995), 183–210.

[MH17] McNeill G., Hale S. A.: Generating tile maps. *Computer Graphics Forum 36*, 3 (2017), 435–445.

[MJR*11] MacEachren A. M., Jaiswal A., Robinson A. C., Pezanowski S., Savelyev A., Mitra P., Zhang X., Blanford J.: Senseplace2: Geotwitter analytics support for situational awareness. In *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)* (2011), IEEE, pp. 181–190.

[PQW*13] Pan G., Qi G., Wu Z., Zhang D., Li S.: Land-use classification using taxi GPS traces. *IEEE Transactions on Intelligent Transportation Systems 14*, 1 (2013), 113–123.

[RB08] Rosvall M., Bergstrom C. T.: Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences of the United States of America 105*, 4 (2008), 1118–1123.

[RMJ07] Rattigan M. J., Maier M., Jensen D.: Graph clustering with network structure indices. In *Proceedings of the 24th International Conference on Machine Learning* (2007), ACM, pp. 783–790.

[RN09] Ronhovde P., Nussinov Z.: Multiresolution community detection for megascale networks by information-based replica correlations. *Physical Review E 80*, 1 (2009), 016109.

[ROT82] Roth S. D.: Ray casting for modeling solids. *Computer Graphics and Image Processing 18*, 2 (1982), 109–144.

[RSC*10] Ratti C., Sobolevsky S., Calabrese F., Andris C., Reades J., Martino M., Claxton R., Strogatz S. H.: Redrawing the map of Great Britain from a network of human interactions. *PLoS One 5*, 12 (2010), e14248.

[SG12] Sun Y., Genton M. G.: Adjusted functional boxplots for spatio-temporal data visualization and outlier detection. *Environmetrics 23*, 1 (2012), 54–64.

[SLQW17] Sun G., Liang R., Qu H., Wu Y.: Embedding spatio-temporal information into maps by route-zooming. *IEEE Transactions on Visualization & Computer Graphics*, *23*, *5* (2017), 1506–1519.

[SR05] Szekely G. J., Rizzo M. L.: Hierarchical clustering via joint between-within distances: Extending Ward's minimum variance method. *Journal of Classification 22*, 2 (2005), 151–183.

[TJBB06] Teh Y. W., Jordan M. I., Beal M. J., Blei D. M.: Hierarchical Dirichlet processes. *Journal of the American Statistical Association 101*, 476 (2006), 1566–1581.

[TSF*16] Thomee B., Shamma D. A., Friedland G., Elizalde B., Ni K., Poland D., Borth D., Li L.-J.: Yfcc100m: The new data in multimedia research. *Communications of the ACM 59*, 2 (2016), 64–73.

[VDVJC11] Van De Voorde T., Jacquet W., Canters F.: Mapping form and function in urban areas: An approach based on urban metrics and continuous impervious surface data. *Landscape and Urban Planning 102*, 3 (2011), 143–155.

[VLBR*16] Von Landesberger T., Brodkorb F., Roskosch P., Andrienko N., Andrienko G., Kerren A.: Mobilitygraphs: Visual analysis of mass mobility dynamics via spatio-temporal graphs and clustering. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 11–20.

[WA77] Weiler K., Atherton P. R.: Hidden surface removal using polygon area sorting. In *SIGGRAPH* (1977).

[WBL17] Wang Y., Baciu G., Li C.: Cognitive exploration of regions through analyzing geo-tagged social media data. In *2017 IEEE 16th International Conference on Cognitive Informatics Cognitive Computing (ICCI*CC)* (July 2017), pp. 59–64.

[WBL18] Wang Y., Baciu G., Li C.: Visualizing functional regions by analysis of geo-textual data. In *Proceedings of the Eurographics/IEEE VGTC Conference on Visualization: Short Papers* (Goslar Germany, Germany, 2018), EuroVis '18, Eurographics Association, pp. 25–29.

[WGDQ18] Wang Y., Gu Y., Dou M., Qiao M.: Using spatial semantics and interactions to identify urban functional regions. *ISPRS International Journal of Geo-Information 7*, 4 (2018), 130.

[WREE67] Wylie C., Romney G., Evans D., Erdahl A.: Halftone perspective drawings by computer. In *Proceedings of the November 14–16, 1967, Fall Joint Computer Conference* (1967), ACM, pp. 49–58.

[WZC*17] Wu W., Zheng Y., Cao N., Zeng H., Ni B., Qu H., Ni L. M.: Mobiseg: Interactive region segmentation using heterogeneous mobility data. In *2017 IEEE Pacific Visualization Symposium (PacificVis)* (2017), IEEE, pp. 91–100.

[XKW*12] Xu Z., Ke Y., Wang Y., Cheng H., Cheng J.: A model-based approach to attributed graph clustering. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data* (2012), ACM, pp. 505–516.

[XYW*16] Xie M., Yin H., Wang H., Xu F., Chen W., Wang S.: Learning graph-based POI embedding for location-based recommendation. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management* (2016), ACM, pp. 15–24.

[YEL17] Yelp dataset challenge. https://www.yelp.com/dataset/challenge (2017). Accessed 20 December 2017.

[YZX*15] Yuan N. J., Zheng Y., Xie X., Wang Y., Zheng K., Xiong H.: Discovering urban functional zones using latent activity trajectories. *IEEE Transactions on Knowledge and Data Engineering 27*, 3 (2015), 712–725.

[ZLY*16] Zhao K., Liu Y., Yuan Q., Chen L., Chen Z., Cong G.: Towards personalized maps: Mining user preferences from geo-textual data. *Proceedings of the VLDB Endowment 9*, 13 (2016), 1545–1548.

© 2019 The Authors Computer Graphics Forum © 2019 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd