Analysis of user generated spatio-temporal data: Learning from collections of geotagged photos

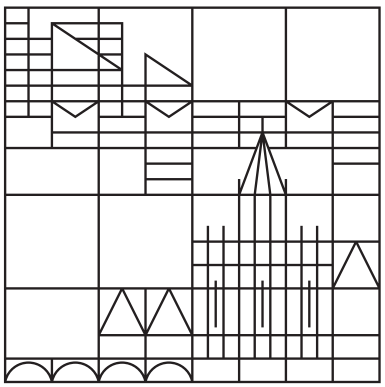
Dissertation zur Erlangung des
akademischen Grades des Doktors der Naturwissenschaften
an der Universität Konstanz
im Fachbereich Informatik und Informationswissenschaft

vorgelegt von

Slava Kisilevich

an der

Universität
Konstanz



Tag der mündlichen Prüfung: 19. März 2012
Referent: Prof. Dr. Daniel A. Keim
Referent: Prof. Dr. Oliver Deussen

Acknowledgements

- I would like to first and foremost thank my advisers Prof. Dr. Daniel Keim and Prof. Dr. Oliver Deussen for their invaluable guidance and support. Their suggestions and encouragement were invaluable during the various stages of my work.

- I would like to thank Dr. Gennady and Dr. Natalia Andrienko for their guidance, support, and patience during my visits in Fraunhofer IAIS, Sankt Augustin. I really learned a lot from them especially about visual analytics, how to conduct a real research and how to write papers.

- I want to thank all my colleagues who helped me in my research and contributed in discussions, generating ideas or writing papers. Special thanks to the following people from my group:

  - Miloš Krstajić
  - Christian Rohrdantz
  - Dr. Florian Mansmann
  - Uwe Nagel (The group of Prof. Dr. Ulrik Brandes)

  Special thanks to external collaborators and co-authors:

  - Dr. Gennady Andrienko and Dr. Natalia Andrienko (Fraunhofer IAIS, Sankt Augustin)
  - Prof. Piotr Jankowski (Department of Geography, San Diego State University)
  - Dr. Lior Rokach (Department of Information Systems Engineering, Ben-Gurion University of the Negev)
  - Dr. Mirco Nanni (Institute of Information Science and Technologies, Italy)
  - Dr. Salvatore Rinzivillo (University of Pisa, Italy)
  - Veronica Maidel (School of Information Studies, Syracuse University).

  Yet special thanks to Pavel Danchenko at Point Carbon Thomson Reuters and Alexander Tchaikin at Vimpelcom for their help in the development of various helper tools at different stages of this thesis and for taking part in the generation of ideas.

- I would like to thank Mrs. Sabine Kuhr for all her help with administration issues.

- My special thanks to Mrs. Anna Dowden-Williams from the Academic Staff Development for her invaluable help in proofreading of the most of my papers. Without her corrections and suggestions my papers would have looked very poor and unprofessional.

- I thank Prof. Mark Last from the Department of Information Systems Engineering, Ben-Gurion University of the Negev for introducing me to Daniel Keim and recommending doing a PhD in his group.

- Finally, I am grateful to my family for giving me their full support and tolerating me during the not so easy period of doctoral studies.

This work was partially funded by the German Research Society (DFG) under grant GK-1042 (Research Training Group "Explorative Analysis and Visualization of Large Information Spaces"), and by the Priority Program (SPP) 1335 ("Visual Spatio-temporal Pattern Analysis of Movement and Event Data").

Parts of this thesis were published in:[1]

[1] Gennady Andrienko, Natalia Andrienko, Peter Bak, Slava Kisilevich, and Daniel A. Keim. Analysis of community-contributed space- and time-referenced data (example of flickr photos). In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST 2009), Poster Paper*, 2009.

[2] Gennady Andrienko, Natalia Andrienko, Peter Bak, Slava Kisilevich, and Daniel A. Keim. Demo-Paper: Analysis of community-contributed space- and time referenced data (by example of Panoramio photos). In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 540–541. ACM SIGSPATIAL GIS 2009, ACM, 2009.

[3] Gennady Andrienko, Natalia Andrienko, Peter Bak, Slava Kisilevich, and Daniel A. Keim. Analysis of community-contributed space- and time referenced data by example of Panoramio photos. In *Proceedings of the Vision, Modeling, and Visualization Workshop (VMV)*, 2009.

[4] Slava Kisilevich, Florian Mansmann, Peter Bak, Daniel A. Keim, and Alexander Tchaikin. Where Would You Go on Your Next Vacation? - A Framework for Visual Exploration of Attractive Places. In *GeoProcessing 2010*, pages 21–26, February 2010.

[5] Slava Kisilevich, Daniel A. Keim, and Lior Rokach. A novel approach to mining travel sequences using collections of geotagged photos. In *Proceedings of the 13th AGILE International Conference on Geographic Information Science*, 2010.

[6] Slava Kisilevich, Daniel A. Keim, and Lior Rokach. GEO-SPADE: A generic Google Earth-based framework for analyzing and exploring spatio-temporal data. In *12th International Conference on Enterprise Information Systems*, volume 5, pages 13–20, 2010.

[7] Slava Kisilevich, Florian Mansmann, and Daniel A. Keim. P-DBSCAN: A density based clustering algorithm for exploration and analysis of attractive areas using collections of geotagged photos. In *1st International Conference on Computing for Geospatial Research & Application*, 2010.

[8] Slava Kisilevich, Milos Krstajic, Daniel A. Keim, Natalia Andrienko, and Gennady Andrienko. Event-based analysis of people's activities and behavior using Flickr and Panoramio geo-tagged photo collections. In *2nd International Symposium Visual Analytics*, 2010.

[9] Slava Kisilevich, Daniel A. Keim, Natalia Andrienko, and Gennady Andrienko. *Towards acquisition of semantics of places and events by multi-perspective analysis of geotagged photo collections.* Lecture Notes in Geoinformation and Cartography (to appear). Springer, 2011.

[10] Slava Kisilevich, Florian Mansmann, Mirco Nanni, and Salvatore Rinzivillo. *Spatio-temporal clustering*, chapter 6, pages 855–874. Springer, 2 edition, 2010.

[1]The paper [12] published in Transactions in GIS, which became part of Chapter 3 was written as part of a joint research. Prof. Piotr Jankowski, Dr. Gennady Andrienko, and Dr. Natalia Andrienko contributed considerably to the success of the paper. All the screenshots presented in Chapter 3 were prepared by Dr. Gennady Andrienko and Dr. Natalia Andrienko by the tool called CommonGIS

[11] Slava Kisilevich, Christian Rohrdantz, and Daniel A. Keim. "Beautiful picture of an ugly place". Exploring photo collections using opinion and sentiment analysis of user comments. In *Computational Linguistics & Applications (CLA 10)*, pages 419–428, October 2010.

[12] Piotr Jankowski, Natalia Andrienko, Gennady Andrienko, and Slava Kisilevich. Discovering Landmark Preferences and Movement Patterns from Photo Postings. *Transactions in GIS*, 14(6):833–852, 2010. 5

[13] Gennady Andrienko, Natalia Andrienko, Peter Bak, Daniel A. Keim, Slava Kisilevich, and Stefan Wrobel. A Conceptual Framework and Taxonomy of Techniques for Analyzing Movement. *Journal of Visual Languages and Computing*, 2011.

[14] Slava Kisilevich, Daniel A. Keim, Amit Lasry, Leon Bam, and Lior Rokach. *Developing Analytical GIS Applications with GEO-SPADE: Three Success Case Studies*, volume 73, pages 495–511. Springer, Heidelberg, 2011.

[15] Slava Kisilevich, Christian Rohrdantz, Maidel Veronica, and Daniel A. Keim. What do you think about this photo? A novel approach to opinion and sentiment analysis of photo comments. *International Journal of Data Mining, Modelling and Management (IJDMMM) - (to appear)*, 2011.

[16] Slava Kisilevich, Piotr Jankowski, Christian Rohrdantz, and Daniel Keim. Exploring geo-tagged photos with density-based clustering and opinion analysis. *Knowledge and Information Systems (submitted)*, 2011.

Additional publications made during the PhD period:

Text analysis:

[1] Marina Litvak, Hagay Lipman, Assaf Ben-Gur, Slava Kisilevich, Daniel A. Keim, and Mark Last. Towards multi-lingual summarization: A comparative analysis of sentence extraction methods on English and Hebrew corpora. In *4th International Workshop On Cross Lingual Information Access*, 2010.

[2] Marina Litvak, Mark Last, Menahem Friedman, and Slava Kisilevich. MUSE - A Multilingual Sentence Extractor. In *Computational Linguistics & Applications (CLA 11) (to appear)*, 2011.

Social Network Analysis:

[1] Slava Kisilevich and Florian Mansmann. Analysis of Privacy in Online Social Networks of Runet. In *3rd International Conference on Security of Information and Networks (SIN 2010)*, 2010.

[2] Slava Kisilevich and Mark Last. Exploring country level gender differences in the context of online dating using classification trees. In *Mining Ubiquitous and Social Environments (MUSE 2010)*, pages 71–87, September 2010.

[3] Slava Kisilevich and Mark Last. *Exploring gender differences in member profiles of an online dating site across 35 countries*, volume 6904, pages 57–78. LNCS/LNAI 6904: Mining and Modeling of Ubiquitous Data in Social Media (to appear), 2011.

[4] Slava Kisilevich, Chee Siang Ang, and Mark Last. Large-scale analysis of self-disclosure patterns among online social networks users: A Russian context. *Knowledge and Information Systems Journal (KAIS) (to appear)*, 2011.

Decision Support Systems:

[1] Slava Kisilevich, Daniel A. Keim, Yossi Palivatkel, and Lior Rokach. Using multiplicative hybrid hedonic pricing model for improving revenue management in hotel business. In *GeoViz Hamburg 2011: Linking Geovisualization with Spatial Analysis and Modeling*, 2011.

[2] Slava Kisilevich, Daniel Keim, Roman Byshko, Michael Tsibelman, and Lior Rokach. Developing a price management decision support system for hotel brokers using free and open source tools. In *13th International Conference on Enterprise Information Systems (ICEIS 2011) (to appear)*, 2011.

[3] Slava Kisilevich, Daniel Keim, and Lior Rokach. A GIS-based decision-support system for hotel room rate estimation and temporal price prediction: The hotel brokers context. *Decision Support Systems (submitted)*, 2011.

Contents

1 Introduction **25**
- 1.1 Aspects of geotagged photo collection analysis . . . 27
  - 1.1.1 Geovisual analytics . . . 27
  - 1.1.2 Discovering attractive places . . . 28
  - 1.1.3 Discovering frequent travel sequential patterns . . . 30
  - 1.1.4 Opinion and sentiment analysis of photo comments . . . 32
  - 1.1.5 GIS-based tools and frameworks . . . 33
- 1.2 Geotagged photos in the spatio-temporal context . . . 35
  - 1.2.1 A classification of spatio-temporal data types . . . 35
  - 1.2.2 Structure of spatio-temporal data . . . 38
  - 1.2.3 Tasks applicable to geotagged photos in the spatio-temporal context . . . 40
- 1.3 Data collection . . . 43
- 1.4 The scope of the thesis and the contributions . . . 43

2 Related Work **47**
- 2.1 Analysis of geotagged photos . . . 47
- 2.2 Clustering methods for spatio-temporal data . . . 48
  - 2.2.1 Descriptive and generative model-based clustering . . . 49
  - 2.2.2 Distance-based clustering methods . . . 49
  - 2.2.3 Density-based methods and the DBSCAN family . . . 50
  - 2.2.4 Visual-aided approaches . . . 52
  - 2.2.5 Important places . . . 53
  - 2.2.6 Patterns and frequent sequences . . . 53
  - 2.2.7 Other clustering methods . . . 55
    - Micro clustering methods . . . 55
    - Flocks and convoys . . . 56
- 2.3 Opinion and Sentiment Analysis . . . 57
- 2.4 Google Earth-based tools and frameworks . . . 58

3 Discovering movement patterns: A geovisual analytics approach **61**
- 3.1 Data . . . 61
- 3.2 Method . . . 61
- 3.3 Analysis . . . 63
  - 3.3.1 Spatiotemporal aggregation . . . 63
  - 3.3.2 Interactive grouping of the places . . . 64
- 3.4 Analysis of photographers' movement in space and time . . . 69
  - 3.4.1 Spatial analysis of movement trajectories . . . 69
    - Aggregation . . . 69
    - Visualization . . . 69
    - Results . . . 69
  - 3.4.2 Spatio-temporal analysis of movement trajectories . . . 71
  - 3.4.3 Summary of findings . . . 75

4 Discovering attractive places **77**
4.1 Data definition and assumptions . . . 77
4.2 Density estimation . . . 79
4.2.1 Density-based clustering . . . 79
4.2.2 DBSCAN algorithm . . . 79
4.2.3 Influence weights . . . 80
4.2.4 Database integration . . . 82
4.2.5 Performance evaluation . . . 83
4.3 Opinion analysis . . . 92
4.4 Visualization and Exploration . . . 93
4.5 P-DBSCAN . . . 97
4.5.1 Problem formulation . . . 97
4.5.2 Definitions . . . 99
4.5.3 Method . . . 100
4.5.4 Evaluation . . . 102

5 Discovering frequent travel sequential patterns **113**
5.1 Method . . . 113
5.1.1 Dataset . . . 113
5.1.2 Photo to POI assignment . . . 114
5.1.3 Sequence Creation . . . 114
5.1.4 Sequence Patterns . . . 114
5.2 Evaluation . . . 115
5.2.1 Case 1. Guimarães, Portugal . . . 116
5.2.2 Case 2. Berlin, Germany . . . 120
5.3 Discussion . . . 124

6 Opinion and sentiment analysis of photo comments **127**
6.1 Development of photo comments corpus . . . 127
6.1.1 Data . . . 127
Region selection . . . 127
Preprocessing . . . 128
6.2 Method . . . 129
6.2.1 Definitions . . . 129
6.2.2 Corpus-based lexicon generation . . . 129
6.2.3 The adjective weighting model . . . 130
6.2.4 Automatic opinion and sentiment analysis . . . 133
Photo features . . . 133
The word orientation list . . . 134
Syntactic opinion reference patterns . . . 134
Identification and separation of photo opinions and general sentiments . . . 134
6.3 Experimental evaluation . . . 136
6.3.1 Design . . . 136
6.3.2 Method . . . 136
6.3.3 Results and discussion . . . 137

Limitations . . . 139
Additional insights from the questionnaire . . . 140

7 A Google Earth-based GIS **141**
7.1 GEO-SPADE application and architecture . . . 141
7.1.1 Overview . . . 141
7.1.2 Main features . . . 142
7.1.3 Architecture . . . 143
7.2 Case studies . . . 144
7.2.1 Analysis of tourist activity . . . 145
7.2.2 Region exploration using geo-tagged photos . . . 149

8 Conclusions **155**
8.1 Discovering movement patterns: A geovisual analytics approach (Chapter 3) . . . 155
8.2 Discovering attractive places (Chapter 4): . . . 157
8.3 Discovering frequent travel sequential patterns (Chapter 5): . . . 158
8.4 Opinion and sentiment analysis of photo comments (Chapter 6): . . . 158
8.5 A Google Earth-based GIS (Chapter 7): . . . 159
8.6 Future work . . . 159

Bibliography **161**

List of Figures

1.1 Spatio-temporal context . . . 37
1.2 Interrelation of chapters and research domains (arrows show the dependence of a technique or method described in a chapter to the one the arrows points to) . . . 46

3.1 Each line in the graph represents the time-series of visits to a given place in the analysis area. There are 2,930 lines in total representing the corresponding number of Voronoi polygons . . . 63
3.2 The distribution of differences between the maximum number of visitors and the 95- and 99-percentiles for 68 places in the Seattle metropolitan area . . . 65
3.3 Grayscale shading indicates the places belonging to four selected classes (4*,6*,7*,8*) categorized earlier in the text as (potentially) interesting . . . 66
3.4 Timeseries of six places comprising cluster 1 on the map of study area (Figure 3.3) in the Freemont district of Seattle. The top part of the figure represents the time series for class 6*: high diff max - 95% AND high diff max - 99% class while the lower part represents the time series for class 7*: higher diff max - 95% AND lower diff max - 99% . . . 67
3.5 Time series graph shows only the line segments where the number of visitors was at least twice as high as in the previous interval . . . 68
3.6 The time series of the line segment selected in the line segmentation mode as shown in Figure 3.5 . . . 68
3.7 Trajectories in the center of Seattle filtered by the city center-only location of start and end, and by the minimum number of moves equal to 50. The irregular mesh of Voronoi polygons, representing places (compartments) is overlaid on the flow map . . . 70
3.8 Trajectories originating in the center and ending outside the city center. The map on the left depict all trajectories. The map on the right depicts only the trajectories with the move count of 10 or more . . . 71
3.9 Trajectories originating and ending outside the city center. The flow map on the left shows the overall pattern comprised of short and long trajectories. The flow map on the right depicts the trajectories that are at the minimum 5 km long and have at least five moves . . . 72
3.10 Long move trajectories (3 km or longer) for the month of June 2006. Similar pattern with moves from Seattle to Bainbridge Island (west) and back was observed for the summer months during the entire 2005-2009 period . . . 72
3.11 Short move trajectories ($0 < length \leq 1500m$). The flow map on the left represents the aggregate moves for 02.2007. The map on the right represents the aggregate moves for 08.2009 . . . 73
3.12 Pattern of movement trajectories in the study area aggregated by 3-month time interval and 5,000m spatial cluster radius. The minimum line thickness (1 pixel) corresponds to five moves and the maximum thickness (36 pixels) corresponds to 176 moves. Flows with less than five moves are not represented . . . 74

4.1 Influence weight calculation . . . 81

4.2 Three regions selected for evaluation . . . 84
4.3 Applying extended DBSCAN with *influence weights* to Washington D.C., Berlin, and London using different neighborhood radius and minimum number of photos generates clusters of different sizes and quantities but does not influence the determination of highly photographed areas. The color hues on the rightmost gradient scale correspond to the places with high influence weights (highly attractive areas). 91
4.4 Visualization using opinion and sentiment scores. The color hues on the rightmost gradient scale correspond to the places with high opinion and sentiment scores. . . 93
4.5 Visualizations using color-coding of photo weights with circles of different sizes inversely proportional to the map scale . . . 95
4.6 Washington D.C. Additional layers of information help to explore interesting areas 96
4.7 Berlin. Opinion scores . . . 96
4.8 Hypothetical examples demonstrating problems in applying general density-based clustering on a photo dataset. . . . 97
4.9 Applying DBSCAN on Washington DC using *MinPts* of 100 photos and neighborhood radius of 30 (red) and 20 (yellow) meters . . . 99
4.10 Washington D.C. DBSCAN and P-DBSCAN comparison using *MinPts* of 100 photos (DBSCAN) and *MinOwners* of 100 owners (P-DBSCAN) and neighborhood radius of 20 meters. DBSCAN - yellow clusters. P-DBSCAN - red clusters . . . 103
4.11 Washington D.C. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters and *grow* adaptive density . . . 104
4.12 Washington D.C. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters and adaptive density with 10% density drop threshold . . . 105
4.13 Washington D.C. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters (green clusters), and adaptive density (red clulters) . . . 106
4.14 Berlin. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters (green clusters), and adaptive density (red clulters) . . . 107
4.15 London. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters (green clusters), and adaptive density (red clulters) . . . 108
4.16 Washington D.C. P-DBSCAN using *MinOwners* of 2 person, neighborhood radius of 100 meters . . . 110
4.17 Berlin. PP-DBSCAN using *MinOwners* of 2 person, neighborhood radius of 100 meters . . . 111
4.18 London. P-DBSCAN using *MinOwners* of 2 person, neighborhood radius of 100 meters . . . 112

5.1 The framework overview for sequence patterns creation . . . 114
5.2 Guimãraes, Portugal. Cluster boundaries of photos assigned to existing POIs (yellow) using a *photo-to-POI* distance threshold of 200 meters. Cluster boundaries forming new areas of POIs were obtained using a distance threshold of 100 meters and a density threshold of three people in a cluster (green) . . . 117
5.3 Guimãraes, Portugal. Cluster boundaries of photos assigned to existing POIs (yellow) using a *photo-to-POI* distance threshold of 400 meters . . . 118

5.4 Berlin, Germany. Cluster boundaries of photos assigned to existing POIs (yellow) using a *photo-to-POI* distance threshold of 200 meters. Cluster boundaries forming new areas of POIs were obtained using a distance threshold of 100 meters and a density threshold of three people in a cluster (green) . . . 121
5.5 Berlin, Germany. Cluster boundaries of photos assigned to existing POIs (yellow) using a *photo-to-POI* distance threshold of 400 meters. Cluster boundaries forming new areas of POIs were obtained using a distance threshold of 100 meters and a density threshold of three people in a cluster (green) . . . 122

6.1 Interdependence of the different core text analysis processes. The numbers correspond to the paragraphs in Section 6.2.4, where details are provided. . . . 133
6.2 Syntactic Opinion Reference Patterns. Word order patterns go from left (before photo features) to right (after photo features), the distance to the photo feature indicates the exact position. . . . 135
6.3 Three POS-annotated example sentences extracted from the photo comments. Each of these sentences contains both a photo opinion and a general sentiment. . . . 135

7.1 Basic GEO-SPADE components . . . 143
7.2 GEO-SPADE architecture . . . 144
7.3 Selection of the area of interest . . . 148
7.4 Monthly photographic activity . . . 148
7.5 Multiple views of regions assigned to some existing POI (left) and regions where no POI was found (right) . . . 149
7.6 Selection of unassigned regions and creation of artificial POIs . . . 150
7.7 Sequence patterns . . . 151
7.8 Combined view of obtained sequence patterns and the map. The regions are highlighted by clicking on the sequences . . . 152
7.9 The control panel for retrieving images according to one of the sorting criteria. The top-N retrieved images are also presented in the control panel. Centering the map view around the top-N photos is implemented by double-clicking the photo . 153
7.10 Two photo representation styles: image thumbnails (left) and color coding according to the image scores of the selected criteria (right) . . . 154

List of Tables

1.1 The type of tasks applicable to geotagged photos in the spatio-temporal context (Part 1) . . . 41
1.2 The type of tasks applicable to geotagged photos in the spatio-temporal context (Part 2) . . . 42

4.1 Evaluation. *Seq* - sequential clustering, *Seq* 2 - sequential clustering with the area divided into two parts, *Seq* 4 - sequential clustering with the area divided into four parts, *Multi* 2 - parallel clustering of the area divided into two, *Multi* 4 - parallel clustering of the area divided into four parts. *Eps* - radius of the neighborhood. *MinPts* - minimum number of photos in the neighborhood. $Owner = 0$ - influence function is not bounded to a finite number of owners. *Avg* - average number of photographers in the neighborhood. The execution time is reported in seconds . . 85
4.2 P-DBSCAN evaluation. . . . 109

5.1 Illustrative results of the Teiresias algorithm . . . 116
5.2 Guimãraes, Portugal. General statistics . . . 118
5.3 Guimãraes, Portugal. Sequence patterns using L=2, W=3 . . . 119
5.4 Berlin, Germany. General statistics . . . 120
5.5 Berlin, Germany. Sequence patterns using L=2, W=3 . . . 121
5.6 Berlin, Germany. Sequence patterns using L=3, W=4 . . . 123

6.1 Statistical information related to five regions selected for analysis . . . 128
6.2 20 most frequent adjectives and their frequency in five selected areas. Words that are commonly used in five regions are colored in yellow, in four regions - gray, in three - pink, in two - green, in one - white . . . 131
6.3 Algorithm-User and User-User inter-rater agreement (IRA) and ICC . . . 138
6.4 Factors that influence the human evaluator . . . 139

7.1 Server-side technologies and tools . . . 145

Abstract

Large collections of geotagged photos publicly accessible through photo-sharing web sites such as Flickr or Panoramio, present an opportunity for the entire Internet community to access the wealth of visual and textual data stored on photos. At the same time, it presents the challenge of how to efficiently and effectively turn this data into information about locations of interest and people's movement preferences.

Geotagged photos can be viewed in a dual way: as independent spatio-temporal events and as trajectories of people in the geographical space. These two views imply, however, two different approaches to an analysis that will yield different kinds of valuable knowledge about places as well as people. In this thesis, geotagged photos are therefore primarily regarded in the context of (event-based) movement rather than multimedia content and present several exploratory and analytical techniques corresponding to event-based and trajectory-based movement analysis. Moreover, geotagged photos are contextually rich data that combine geographical information about the places were photos where taken with textual information. This information includes for the most part titles and tags attached to photos by owners of these photos as well as comments that can be written by other users. This unique combination is what makes the geotagged photos special for analysis of people's movement and behavior in contrast to the general purpose spatio-temporal data like GPS-based records. Analysis of spatio-temporal data is challenging since it always requires combining different analytical methods like geocomputation and geographical analytics, algorithms like aggregation and clustering, technologies in which the data is appropriately visualized on the map and scalable with the amount of the data.

Taking into consideration the above mentioned aspects, we propose in this thesis a systematic approach to an analysis of people's movement and events using geotagged photos. We make contributions in four main research areas: (1) geovisual analytics - interactive exploration of patterns of people's trajectories and interesting routes that they take during photographing; (2) data mining - development of techniques and algorithms for discovery of attractive areas and finding frequent sequential patterns of people's movement; (3) text mining and computational linguistics - development of approaches for extraction and analysis of comments that people write for photos, and (4) systems engineering - development of a GIS-based tool to facilitate handling of geotagged photos. Moreover, while some of the approaches were targeted towards the imaginary domain expert, some approaches proposed in this thesis were deliberately user-centered in order to demonstrate how the proposed approaches can be used in user-centered scenarios if implemented and delivered by a service provider.

Finally, the research demonstrated in this thesis will be useful for practical reasons, such as future research targeting people's preferences for urban locations, landmarks and corresponding travel itineraries. It may also benefit (research) areas like city promotion and advertising, public safety, tourism, and civic minded activities.

Zusammenfassung

Grosse Sammlungen von Fotos mit Geotags, die öffentlich zugänglich sind durch Foto-Sharing-Seiten wie Flickr oder Panoramio, bieten der gesamten Internet-Community die Gelegenheit auf eine Fülle visueller und textlicher Daten von Fotos zuzugreifen. Jedoch stellt sich die Frage, wie effizient und effektiv diese Daten in Informationen über Standorte die von Interesse sind und Präferenzen der menschlichen Bewegungsräume umgewandelt werden können.

Fotos mit Geotags können auf zwei verschiedene Arten betrachtet werden: als eigenständige raum-zeitliche Ereignisse und als Trajektorien der Menschen in einem geographischen Raum. Diese beiden Sichtweisen implizieren jedoch zwei verschiedene Ansätze für eine Analyse, die auf verschiede Weise wertvolles Wissens über Orte und menschliches Verhalten erzeugt. In dieser Arbeit werden Fotos mit Geotags daher mehr im Kontext der (ereignisbasierten) Bewegung als im Kontext von Multimedia-Inhalten betrachtet und präsentieren einige Explorations-und Analyseverfahren in Übereinstimmung mit ereignisbasierten und trajektorien-basierten Analysen. Darüber hinaus sind Fotos mit Geotags kontextuell reichhaltigen Daten, welche geografische Informationen über die Orte an denen die Fotos aufgenommen wurden mit Textinformationen kombinieren. Diese Textinformationen sind zum grössten Teil mit Bildtiteln und Tags versehen, die von den Eigentümern der Fotos hinzugefügt wurden. Sie können ebenfalls Kommentare beinhalten, die von anderen Benutzern geschrieben wurden. Diese einzigartige Kombination macht die Fotos mit Geotags im Gegensatz zu den allgemeinen spatiotemporalen Daten wie GPS-basierten Datensätzen besonders geeignet für die Analyse der Bewegung und das menschliche Verhaltens. Die Analyse von raum-zeitlichen Daten ist eine Herausforderung, da sie immer die Kombination verschiedener analytischer Methoden erfordert wie bespielsweise Geo-Algorithmen, geografische Analysen, Algorithmen wie Aggregation und Clusterbildung, Technologien, in denen die Daten auf der Karte entsprechend visualisiert und gleichzeitig mit der Menge der Daten skalierbar sind.

Unter Berücksichtigung der oben genannten Aspekte, untersuchen wir in dieser Arbeit eine systematische Herangehensweise für die Analyse der Bewegung von Menschen und Ereignissen durch die Nutzung von Fotos mit Geotags. Wir diskutiren vier Forschungsschwerpunkte: (1) Geovisuelle Analysen - interaktive Exploration von Mustern menschlicher Trajektorien und interessante Routen, die sie für das Fotografieren wählen; (2) Data Mining - Entwicklung von Techniken und Algorithmen für die Entdeckung der attraktiven Gebiete und der häufigen sequenziellen Bewegungsmuster; (3) Text Mining und Computerlinguistik -Entwicklung von Ansätzen zur Extraktion und Analyse von Kommentaren, welche Internetnutzer für die Fotos schreiben, und (4) Systemtechnik - Entwicklung eines GIS-basierten Tools zur Bearbeitung von Fotos mit Geotags zu erleichtern. Darüber hinaus, während einige der Ansätze auf imaginären Fachexperten ausgerichtet waren, sind einige der vorgeschlagenen Ansätze in dieser Arbeit gezielt nutzerorientiert, um zu zeigen, wie die von uns vorgesehenen Ansätze in nutzerorientierten Szenarien eingesetzt werden können, wenn sie implementiert und von einem Dienstleister geliefert werden.

Ergänzend zu den von uns vorgestellten Methoden und Ansätzen, können die Ergebnisse dieser Arbeit aus praktischen Gründen sinnvoll sein für zukünftige Forschungsvorhaben, welche sich mit der Präferenzen der Menschen für städtische Standorte, Sehenswürdigkeiten und entsprechende Reiserouten befassen. Weiterhin können (Forschungs) Bereiche wie Stadt und Werbung, öffentliche Sicherheit, Tourismus und zivilgesellschaftliche Aktivitäten ebenfalls von den Ergebnissen dieser Arbeit profitieren.

1 Introduction

Contents

1.1 Aspects of geotagged photo collection analysis 27
1.1.1 Geovisual analytics 27
1.1.2 Discovering attractive places 28
1.1.3 Discovering frequent travel sequential patterns 30
1.1.4 Opinion and sentiment analysis of photo comments 32
1.1.5 GIS-based tools and frameworks 33
1.2 Geotagged photos in the spatio-temporal context 35
1.2.1 A classification of spatio-temporal data types 35
1.2.2 Structure of spatio-temporal data 38
1.2.3 Tasks applicable to geotagged photos in the spatio-temporal context . . 40
1.3 Data collection 43
1.4 The scope of the thesis and the contributions 43

Ubiquity of location-aware devices, cheap storage and fast computing power has enabled collection of large amounts of spatio-temporal data. Different application domains like zoology, activity-based analysis or tourism in which data collection was a tedious and manual process (observations, surveys), benefit from the positioning technology and demand new analysis and techniques to cope with large quantities of these data.

Collections of geotagged photos have recently become available [Goodchild, 2007] as an alternative to the GPS-based data source of geographic data due to the availability of photo-sharing sites such as Flickr[1] and Panoramio[2], in which millions of users from all over the world upload their geo-referenced photos. The basic information provided by a person during photo upload is the location where the photo was taken, the time of the action, and the textual identifiers including title and tags. The advantages of these community-contributed geotagged data can be highlighted by citing Girardin et al. [2008b]:

> Unlike the automatic capturing of traces, the manual disclosure of location in the act of geotagging of photo provides additional qualities: positioning a photo on a map is

[1] http://www.flickr.com
[2] http://www.panoramio.com

not simply adding information about its location; it is also an act of communication which contains what people consider as relevant for themselves and others.

Geotagged photos having spatial and temporal references can be viewed, on the one hand, as independent spatio-temporal events. On the other hand, the entries made by the same person can be considered as a trajectory of this person in the geographical space, which tells something about the movement and behavior of this person. The whole dataset can be viewed as a set of trajectories of multiple people. These two views suppose different approaches to the analysis, which can yield different kinds of knowledge.

The combination of the spatial, temporal and contextual components (title, tags, user comments) in one source of data creates unique possibility to explore such tasks as people's preferences, travel behavior, or places that people visit, and allows to formulate many research questions that could not be answered without the availability of a such contextually rich data.

The fundamental challenge in answering the above questions is the nature of available data, which is voluminous (there were 5 billion (5*109) photos uploaded on Flickr on 2010[3] and 20 million geotagged photos uploaded on Panoramio on 2010[4]) of which the number of geotagged photos on Flickr is more difficult to establish - we managed to collect about 96 million photos, but this number does not include privately held photos.

In addition to a research motivation there may be a number of practical reasons for taking an interest in people's preferences for urban locations, landmarks and corresponding travel itineraries, such as for example, city promotion and advertising, public safety, tourism, and civic minded activities.

The goals of the present research can be summarized as follows:

1. To propose a systematic approach to the analysis of people's movement and events.
2. To explore the potential of publicly available geotagged photos for providing information about people's activities, preferences and behavior in space and time.
3. To explore the potential of publicly available geotagged photos for providing information about temporal events and places visited by people.
4. To extract the knowledge by applying techniques in different domains such as geovisual analytics, data mining, text mining and computational linguistics.
5. To develop new approaches and algorithms for knowledge extraction from geotagged photo collections.
6. To develop tools that facilitate knowledge extraction, exploration and analysis of geotagged photo collections.

To achieve the above mentioned goals we have defined five main aspects for the analysis of geotagged photo collections that we consider in this thesis:

1. Geovisual analytics - a visual interactive exploration of the movement data.

[3]http://blog.flickr.net/en/2010/09/19/5000000000
[4]http://blog.panoramio.com/2010/01/new-time.html

2. Discovery of attractive places - data mining techniques for the event-based analysis of movement data.

3. Discovery of frequent travel sequential patterns - a data mining technique for the analysis of trajectory-based data.

4. Opinion and sentiment analysis of photo comments - a text analysis that is part of *discovery of attractive places*.

5. GIS-based frameworks - an engineering of a dedicated framework to handle geotagged photos.

1.1 Aspects of geotagged photo collection analysis

1.1.1 Geovisual analytics

The increasing volume of geographic data served through intuitive to operate and powerful interfaces has opened access to geographic information for millions of users around the globe. However, the abundance of information gives rise to a new challenge, recognized earlier by Simon [1971], who pointedly stated that

> Information consumes the attention of its recipients. Hence a wealth of information creates a poverty of attention, and a need to allocate that attention efficiently among the overabundance of information sources that might consume it.

Consequently, there is a need to develop methods and tools for finding relevant geographic information in large datasets, which are organized according to some general rules but not necessarily to support specific user information needs. Geovisual analytics [Keim, 2005, Andrienko et al., 2007a] is a methodological approach aimed at facilitating an efficient search for data and information, corresponding to user information needs, in very large data depositories through a synergistic combination of numerical data analysis and visual data exploration.

An exploratory approach to visualization and analysis of spatio-temporal data follows a straightforward workflow comprised of three steps: *overview first, zoom and filter, then details-on-demand* [Shneiderman, 1996]. According to Keim [2005], these steps might not be applicable to very large and complex datasets, such as photos posted on the Flickr website. Therefore, it may be necessary to begin the analysis of data while applying computational methods for reducing the size and complexity, including data aggregation, dimensionality reduction, or feature extraction. Furthermore, it may not be possible to explore thoroughly the whole dataset but instead focus only on what is important. Therefore, Keim [2005] suggests another workflow, called Visual Analytics Mantra, comprised of the following three steps:

- Analyze first - show the important.
- Zoom, filter and analyze further.
- Details on demand.

We suggest the following procedure, corresponding with the Visual Analytics Mantra, for the analysis of a large set of photographs. In the first step, the spatio-temporal dataset is aggregated and summarized. This step of the workflow responds to the challenge of dealing with large datasets, a common trait of spatio-temporal data where there are simply too many records to depict each record directly, and gives the analyst a chance to spot interesting patterns. In the second step, the aggregates of the original data are analyzed using exploratory visual and analytic tools, which by virtue of being linked provide various data views in geographic, attribute, and temporal spaces. In the third step, interesting patterns revealed in the course of exploratory analysis are investigated in more detail by focusing on subsets of original data underlying the observed patterns. This step may involve additional data transformations and exploratory analysis augmented by human interpretation of the photograph content to help interpret the observed data patterns.

By applying geovisual analytics on the collections of geotagged photos, we pursue two goals:

- To explore the potential of publicly volunteered photos for providing information about people's activities in space and time.
- To experiment with geovisual analytic techniques for extracting this information.

The geotagged photos uploaded to Flickr may be regarded as an expression of preference for a given location/landmark revealed by the photo authors. In addition to finding out about geographical preferences for landmarks we also analyze the itineraries of landmark photographers and their spatio-temporal pattern. We explore three questions related to landmark preferences and the movement behavior of photographers:

- Which locations outside the category of the most frequently visited tourist attractions gained the attention of Flickr photographers?
- What were the likely reasons for attracting the attention of photographers?
- What was the spatio-temporal pattern of photographer movement to visit the locations?

We present a spatio-temporal analysis of Flickr photos, in which the information about photo location and time was combined with available photo titles and human interpretation of the photographed objects to discover new potential landmarks.

1.1.2 Discovering attractive places

Google and Yahoo! offer services for interactive exploration of places and photos such as Google Earth photo layer or Flickr World Map. While these services readily reveal where photos were taken by depicting the photo locations with dots or circles on the map, the user has to click on every photo to view the content, which is a very tedious and cognitively difficult process of integrating information contained in photos and their comments. Moreover, it is difficult to focus on places of interest, since there is no visual clue that would guide the user. At the same time, a visual summary of all photos by means of a map (e.g. a density map) is impractical because it requires reading the metadata of thousands or millions of photos. Motivated by the desire to enrich the user experience during the exploration of places of interest by providing

visual clues helping to differentiate between places deemed as highly interesting and the rest of places depicted on photo collections contributed by millions of users worldwide, we propose two complementary approaches to assist in user's search for interesting places.

The first approach is based on the density-based clustering of geotagged photos, while the second is based on the individual scoring of photos using textual analysis of comments (opinion analysis) accompanying the posted photos. The opinion analysis is applied to every photo independently by disregarding the nearby photos, and taking into account only the comments that were written for that particular photo. We proposed our own approach to opinion and sentiment analysis of photo comments making a contribution into the field of computational linguistic. Therefore, we refer the reader to a separate chapter - Chapter 6 for a detailed explanation of the approach used in opinion extraction. In both, the density-based and opinion-based cases, an individual photo is assigned with a weight that corresponds to the degree of its importance or interestingness with respect to photos nearby, which allows visualization of photo locations by mapping photo weights to color scale.

We extend the density-based clustering process by combining a clustering task with the kernel density estimation of the clustered data. The extended density-based clustering yields not only clusters that represent interesting places but it also allows to determine the importance of every photo in a cluster by calculating its importance weight. The incorporation of the importance weight calculation into the clustering process solves also the problem of efficient parameter selection. While clusters can span large areas due to the high density of photos, the different attractive areas within a cluster can be easily found by visualizing photo importance weights that are determined by the photos in the local neighborhood. Therefore, the importance of proper parameter selection appears less critical than in the general density-based clustering tasks and can be mostly important in the trade-off between performance (the more photos are treated as noise, the faster the algorithm converges) and area coverage (the less photos are treated as noise, the larger area is covered by clusters).

The two approaches complement each other and can be useful in different scenarios. For example, if one is interested in finding places of high touristic activity, then the density-based clustering is suitable. However, if the user wants to explore places without known points of interest or characterized by low touristic activity, the density-based clustering may fail to find clusters of sufficient density. In this case, the individual weighting of photos using opinion scores can be useful since the scores do not depend on the photos taken nearby, but rather on the opinions expressed by viewers of that photo. Since the density-based clustering is a time and resource consuming process, we discuss several options for optimizing the clustering runtime complexity, which together with the opinion analysis can support explorations of large but unevenly distributed collections of geotagged photos.

The approach to mapping geotagged photos, developed in this research, differs from interpolation-based techniques (see Section 2.1 for an overview) in that it uses a quantitative measure of the interestingness of an individual photo to generate a visualization of geotagged photo dataset. This approach makes it possible to obtain a visual overview of attractive places, and to view hotspot locations of interesting places. Moreover, unlike the grid-based clustering (Section 2.1) that requires a new interpolation each time the scale changes due to changing user visualization needs, our approach does not require weight recalculation and hence supports faster generation of visualizations at various scales.

In the density-based approach described above the clustering algorithm is combined with a

density estimation to calculate the relative importance of each photo at some place. The primary goal, therefore, was to acquire importance weights and to visualize those weights on the map for user exploration. As a result, the clusters produced by the general purpose density-based clustering algorithms (for example DBSCAN [Ester et al., 1996]) that could potentially be used in the process of importance weight calculation are less applicable to the task of analysis of points of interest using the cluster boundaries (as is done in the classical data mining tasks) since the clusters could contain many different locations of high photographic activity. Drawing upon this observation, we aimed at improving the clustering process in such a way that the clusters produced by the algorithm would describe few points of interest as possible. This led to an extension and refinement of the density-based clustering approach by providing our specialized implementation called P-DBSCAN (photo DBSCAN) of the popular DBSCAN [Ester et al., 1996] algorithm that is able to find clusters around attractive places in a robust way by taking into consideration photo ownership. In addition, in many cases, the obtained clusters may have different densities in different parts of the cluster. Therefore, we introduce a notion of adaptive density to handle such cases. The basic idea is to split the cluster if different local areas of the cluster have large differences in density. The splitting should create small "packed" clusters in which density does not vary much. The combination of the ownership of photos with the adaptive density allows applying the algorithm even without defining the minimum number of owners $MinPts$ that is an important parameter in general purpose density-based clustering (see Section 2.2.3 for an overview) in cases where the initial number is not known in advance or hard to estimate.

1.1.3 Discovering frequent travel sequential patterns

Existing works on analyzing people's mobility mainly concentrate on the trajectories obtained by GPS-enabled devices. Such trajectories usually consist of many space-and-time referenced points measured at a constant interval where the foremost non-trivial task is to extract (semantically) important parts or stay points. Several approaches exist to find the important elements: (1) applying density functions to find regions where intersections of trajectories are high or (2) finding parts of a trajectory where the object stayed for a significant period of time. After the stay points are found, data mining algorithms can be applied to mine frequent sequences. These approaches involve several issues. (1) Important intersection sites for various individuals may seem to be the same but in fact correspond to different sites that were visited. For example, one person visited a bank and another entered a shop. The bank and the shop are situated close to each other and these regions were defined as one stay point in the trajectory of these two persons. (2) Since the stay points are defined mainly using characteristics of the trajectory, without any background knowledge, there is a need to interpret the obtained sequences. The first issue can be tackled by assuming that the regions visited by people are important, making no distinction between the sites visited in these regions. The second issue can be resolved by using external databases of points of interest (POIs) to explain the important places. However, a POI database may be unavailable, inapplicable to the data (shopping, work) or incomplete. Large-scale, GPS-based datasets of people's trajectories are still not available partly because of data acquisition problems. For example, Zheng et al. [2009] reported that a large GPS dataset was created from data collected by 107 users carrying GPS-enabled devices with them for one year. The regions that these users covered included 36 cities in China and various areas in the

USA, South Korea, and Japan. Without regard to the difficulty of data acquisition, the question of whether 107 users are enough to mine travel sequences in different parts of the world remains open.

The geotagged photos differ technically and semantically from raw GPS-based type trajectories. Unlike trajectories recorded by GPS devices and measured at a constant time, photo data can be regarded as a private case of raw trajectories in which an individual is capturing an important event. Using time and location of photos taken by a person, it is possible to construct event-based trajectories, which can then be used to analyze travel activity. The act of sharing the photo with others through photo-sharing sites reveals important information, including time, location, title, tags and the photo itself. Therefore, this data can be directly used in retrieving interesting places, providing us with the opportunity to discover travel sequences and understand in what order people visit such places.

The goal is to suggest an automatic approach for mining semantically annotated travel sequences using geotagged photos by searching for sequence patterns of any length. The sequences obtained may contain patterns that are not necessarily the immediate antecedents. Moreover, the approach that we propose can examine sequences in which the same pattern is repeated more than once in the same sequence.

We address the problem of automatically finding semantically annotated sequences. For instance, consider the following sequences:

- $A \rightarrow B \rightarrow C$
- $A \rightarrow * \rightarrow D$

The first sequence can be interpreted as a route followed by people from place A to place B and from place B to place C. It is important to note that those who reached C from B are the same persons as those who reached B from A. In the second sequence, those who started from A and reach D, did not necessarily visit a particular place, rather they may have visited any possible place before visiting D. Our approach to mining travel sequences consists of four main parts. In the first part, we automatically assign every geotagged photo to a nearby POI using an external POI database. Since we do not perform any image analysis, we cannot really know what was photographed. However, the fact that the photo was taken near some known POI assumes the presence of the photographer in that place. After step one, there are photos that were not assigned to any POI. There are two reasons for this. (1) A photo was taken in an area where there are no POIs (for example a forest or parking lot near someone's house). (2) A photo was taken in an attractive place but a POI is missing in the database. Therefore, there is still a need to analyze these locations and artificially create points of interest using several constraints. For this purpose, we apply a density-based clustering algorithm in order to find dense areas [Maimon and Rokach, 2005]. This allows us to filter out outliers - sparse areas, where the number of people who took photos is less than a predefined threshold. The dense regions that are obtained are new, unknown points of interest which are added to the areas acquired in the first step. The automated process annotates these areas with symbolic names and stores the boundaries of these regions for future access. In the third step, the travel sequence of each person is constructed using the notion of a session: a time frame in which a person takes photos in a particular area. In the fourth step, travel sequence patterns are mined using semantics obtained from the first two steps.

1.1.4 Opinion and sentiment analysis of photo comments

With the fast development of user-centered Internet technologies, we witness a rapid growth of Web resources, which not only allow users to obtain, but also to generate their own textual information. This leads to dramatic improvements of products and services. For example, nowadays it is difficult to imagine that we would book a hotel room without checking the hotel's overall ranking or without reading comments previously written by other users. We are also less inclined to buy a product without reading comments or ratings about its quality. In fact, written opinions have become essential components in decision-making processes and are common in almost all parts of our life. They are essential parts of blogs, news, financial market reports, product reviews, etc. However, textual information generated on the Web grow almost at an uncontrollable pace, and manual skimming through user opinions has become a time consuming process.

A typical task in opinion mining is to determine whether a document (review, comment) is bearing a positive or negative connotation [Hatzivassiloglou and McKeown, 1997, Turney, 2002, Dave et al., 2003, Salvetti et al., 2004, Kennedy and Inkpen, 2006, Das and Chen, 2007, Fahrni and Klenner, 2008, Argamon et al., 2009]. If either connotation is present, the task can be formulated as a classification problem with two class labels (positive and negative) [Liu, 2009]. Three different kinds of approaches have been used: Unsupervised [Turney, 2002], semi-supervised [Argamon et al., 2009] and supervised ones [Gamon, 2004, Salvetti et al., 2004, Pang and Lee, 2005, Kennedy and Inkpen, 2006, Chesley et al., 2006, Das and Chen, 2007, Drake et al., 2008, O'Hare et al., 2009]. Supervised machine learning approaches perform well if sufficient labeled training data exist (for example, in the movie reviews domain users assign ranks to movies along with their written opinions). However, in domains where labels are not easily acquired or where opinion orientation is measured on a real-valued scale [Subrahmanian and Reforgiato, 2008], unsupervised approaches are more favorable.

In this thesis, we consider the problem of opinion and sentiment analysis of users' comments written for photos that are uploaded to photo sharing web sites. Detailed inspection of user comments revealed that comments are noisy, relatively short, and contain only few negations. They may be written in any language, contain arbitrary syntactic structures and typos. Moreover, they may contain a mixture of opinions on the quality of the photo (usually positive) "Great shot", "Nice picture" and sentiments or moods expressed towards objects depicted on the photo ("Sad place"). Further observations revealed that written opinions are mostly accompanied by adjectives, which is in accordance with past findings [Wiebe et al., 1999, Wiebe, 2000]. As mentioned above, a widely used approach is to classify documents using a binary classification. This approach seems inappropriate in our case for two reasons: (1) Photo comments have two subjects of opinions (opinions on the photo and sentiment towards objects). Consequently, we will lose valuable information if the overall score will be a mixture of two opinion scores. (2) Since most of the opinions are positive, we will end up with most of the photo comments classified as positive. In order to provide a workable essential-feature analysis, we propose two improvements over existing approaches. We extract two types of opinions: (1) opinions that relate to the photo quality, and (2) general sentiments targeted towards objects depicted on the photo.

Supervised machine learning approaches are not feasible in our case since it is very hard to find agreements between human annotators on a real-valued scale, e.g. the difference in opinion strength between "Great shot" and "Amazing photo" cannot be clearly defined. For

that reason, we propose an unsupervised approach for opinion scoring using concepts of word importance based on statistical properties derived from the field of information retrieval [Salton and Buckley, 1988a] and using concepts of Zipf's speech regularity [Zipf, 1949] and semantic differentiation [Osgood, 1957].

Based on the observations described above, we generated our own lexicon of adjectives extracted from the corpus of user comments, and analyzed its usage with respect to photo quality opinions and general sentiments, as well as their usage by commenters. We found that in the majority of cases, adjectives are used directly with the subject of the opinion ("Great shot") and that the most frequently used adjectives are the same, even if different regions of the world are considered with photos of different subject matters. The latter suggested that a finite lexicon of adjectives could be used for opinion and sentiment analysis of photo comments in many regions.

1.1.5 GIS-based tools and frameworks

Geographic Information Systems (GIS) handle a wide range of problems that involve spatial distribution data such as housing, healthcare, transportation, cartography, criminology, and many others. Selecting the right GIS is driven by the task the user wishes to perform [Peters, 2008]. Underlying considerations may include the availability of analytical functions, performance, ease of use, interoperability, extensibility, etc. Nowadays, with the proliferation of positioning technology and the rapid growth of spatio-temporal data together with a composite structure of non-spatial data, GIS tools have become more and more complex as researchers seek to address and implement new spatial analysis tasks and techniques. Although many contemporary GIS provide APIs for extending their software for specific needs, this involves adapting the GIS to learn a complex set of proprietary API functions, that may often contain more than hundred or thousand of functions from different logical layers, which is not feasible, particularly in scenarios that demand rapid prototyping like in academic research or in resource-constrained projects or when the tool is supposed to be used by non-professionals. Usually, the analyst (or researcher) requires fast implementation of tasks and the decision to extend the GIS application in cases when some functionality is not available, is rarely favorable. Consequently, to achieve the desired results, the geo-processing task is split between multiple GIS applications where each of them partly provides the required functionality.

The issue of proprietary APIs and interoperability is addressed by Open Geospatial Consortium (OGC)[5], which encourages developers to use standard interfaces and specifications when implementing geo-processing. A need for distributed geo-processing systems was addressed in recent publications [Friis-Christensen et al., 2007, Diaz et al., 2008, Schaeffer and Foerster, 2008, Foerster et al., 2009]. Such systems split the workload between client and server, where clients are often Web applications, accessible by multiple users while the server performs all the required computations and delivers the results to the client using Web service technologies. Such architectures have several advantages over old, closed solutions: (1) there is no need to install a Web application; (2) many users can access the application using a Web browser; (3) the computation is performed on the server side, which relieves the user's computer from performing time-consuming operations; and (4) there is no need to locally store the data (the data can be accessed directly by the service and hosted by the service provider).

[5] http://www.opengeospatial.org

Scalable, distributed architecture, that conforms to open standards, may solve the problem of interoperability and service reusability. However, the client side still has to be developed according to specific tasks defined in advance. Since any changes in the design or in extending the capabilities should be performed by the body responsible for maintaining the Web application, there is a need for a platform, that supports easy extensibility with minimum knowledge of the underlying API. Such a platform should support many of the general visual data exploration approaches like *brushing, focusing, multiple views, linking.* It should also be able to support (geo)visualization approaches such as direct depiction, visualization of abstract data summaries, and extraction and visualization of computationally extracted patterns. It should also be capable of integrating different geo-related tasks to allow the analyst to quickly generate and test her hypotheses [Andrienko et al., 2008]. Of the platforms that combine ease of development and visualization, Google Earth has become popular among researchers dealing with spatial data.

Google Earth has attracted interest ever since it first appeared on the market [Grossner, 2006]. Although debates whether Google Earth is a true GIS and what advantages it brings to scientists continue [Patterson, 2007, Goodchild, 2008, Sheppard and Cizek, 2009, Farman, 2010], the number of publications in which Google Earth is used for visualization and exploration of geo-related results shows the great interest among researchers in this tool. There are several factors that have made Google Earth so successful among geo-browsers: it is freely available; its fast response and real-time exploration of spatial data; its use of satellite imagery; its ease of use and of Keyhole Markup Language (KML)[6], a human readable, XML-based format for geographic visualization. Other attractive factors include Google Earth's ability to implement basic visualization techniques like zooming, panning and tilting, and the availability of many layers of information provided by Google and by user communities. In addition, Google Earth includes a built-in HTML browser to display textual information and allows dynamic content to be streamed from a server in response to changes in a visible frame.

Recent publications demonstrated that Google Earth can be effectively used for data exploration in various areas such as: weather monitoring [Smith and Lakshmanan, 2006], spatio-temporal data exploration [Wood et al., 2007], data mining [Compieta et al., 2007], insurance [Slingsby et al., 2007], crisis management [Pezanowski et al., 2007], spatial OLAP [Martino et al., 2009], and geospatial health [Stensgaard et al., 2009]. Despite its capabilities, Google Earth is still used as a "secondary" tool for data visualization because of its lack of geo-analytical functionality. Developers who want to integrate Google Earth in their applications usually overcome this geo-analytical deficiency by embedding Google Earth into a Web application using its JavaScript API. "Missing" features can then be implemented using a *Network Link* element, which is part of KML, to point to the URL of the underlying server to request a specific operation. Although the *Network Link* feature is used for creating dynamic content, it allows data to flow in only one direction, from the server to Google Earth in response to the changes in the visual boundaries. Clearly, this feature does not allow more complex operations like passing parameters interactively, running the data mining algorithm on the server, and returning the results for a particular setup of objects being visualized by Google Earth. Other mechanisms for performing such operations are required.

Recently, it became possible to embed Google Earth in desktop applications[7] using the Google

[6] http://code.google.com/apis/kml/documentation/kmlreference.html
[7] http://code.google.com/p/winforms-geplugin-control-library/

Earth Web browser plugin[8] COM interface. This makes it possible to write custom applications around Google Earth in high level languages like c# and to completely control the logic of the application thus bypassing the limitations of the stand-alone version of Google Earth. This possibility, on the one hand, and the constraints of current GIS systems, on the other, encouraged us to develop a prototype desktop system called GEO-SPADE (**GEO SPA**tiotemporal **D**ata **E**xploration). The architecture of GEO-SPADE, is based on a thin client paradigm and introduces pluggable components and Service-oriented Architecture (SOA). The framework acts as a bridge between the Google Earth engine and the user defined functionality implemented in a user provided code as a front-end loaded by the framework and the back-end that is completely separated from the client-side (GEO-SPADE) but communicated by means of Web services with the front-end. Unlike the typical APIs, that may often contain more than hundred or thousand of functions from different logical layers, the GEO-SPADE framework provides only the minimal set of functions for plug-in development, networking, and visualization using Google Earth engine. The minimalistic set of API functions and the text-based protocol (KML) for geographical visualization allows for rapid prototyping of different tasks, while the ease of use pertinent to Google Earth allows non-professionals to utilize GEO-SPADE.

We pursued three main goals:

- To present a SOA-based architecture that allows for quick hypothesis generation and testing by implementing extensible components.
- To demonstrate that Google Earth is capable of assisting in geo-processing, visualization and data exploration methods by embedding it into a customizable desktop application.
- To enhance the capabilities of Google Earth by making it a primary tool for geo-related analysis and geotagged photos in particular.

1.2 Geotagged photos in the spatio-temporal context

The spatio-temporal context is a large container, which includes several kinds of data types that exhibit extremely different properties and offer sensibly different opportunities of extracting useful knowledge (usually by means of data clustering). In this section we provide a taxonomy of the data types that are available in the spatio-temporal domain, describe each class of data, present the general structure of the spatio-temporal data and position the geotagged photo data in the spatio-temporal context by outlining possible tasks that can be applied in the context of spatio-temporal domain.

1.2.1 A classification of spatio-temporal data types

Several different forms of spatio-temporal data types are available in real applications. While they all share the availability of some kind of spatial and temporal aspects, the extent of such information and the way they are related can combine to several different kinds of data objects. Figure 1.1 visually depicts a possible classification of such data types, based on two dimensions:

[8] http://earth.google.com/plugin/

- the *temporal dimension* describes to which extent the evolution of the object is captured by the data. The very basic case consists of objects that do not evolve at all, in which case only a static snapshot view of each object is available. In slightly more complex contexts, each object can change its status, yet only its most recent value (i.e., an updated snapshot) is known, therefore without any knowledge about its past history. Finally, we can have the extreme case where the full history of the object is kept, thus forming a time series of the status it traversed;

- the *spatial dimension* describes whether the objects considered are associated to a fixed location (e.g., the information collected by sensors fixed to the ground) or they can move, i.e., their location is dynamic and can change in time.

In addition to these two dimensions, a third, auxiliary one is mentioned in our classification, which is related to the spatial extension of the objects involved. The simplest case, which is also the most popular in real world case studies, considers point-wise objects, while more complex cases can take into consideration objects with an extension, such as lines and areas. In particular, Figure 1.1 focuses on point-wise objects, while their counterparts with spatial extension are omitted because they are not considered in this thesis.

In the following we briefly describe the main classes of data types we obtain for point-wise objects.

ST events. A very basic example of spatio-temporal information are spatio-temporal events, such as earth tremors captured by sensors or geo-referenced records of an epidemic. Each event is usually associated with the location where it was recorded and the corresponding timestamp. Both the spatial and the temporal information associated with the events are static, since no movement or any other kind of evolution is possible. Finding clusters among events means to discover groups that lie close both in time and in space, and possibly share other non-spatial properties. A classical example of that is Kulldorff [1997]'s spatial scan statistics, that searches spatio-temporal cylinders (i.e., circular regions considered within a time interval) where the density of events of the same type is higher than outside, essentially representing areas where the events occurred consistently for a significant amount of time. In some applications, such as epidemiology, such area is expected to change in size and location, therefore extensions of the basic scan statistics have been proposed that consider shapes different from simple cylinders. For instance, Iyengar [2004] introduces (reversed) pyramid shapes, representing a small region (the pinpoint of the pyramid, e.g. the origin of an epidemic) that grows in time (the enlarging section of the pyramid, e.g. the progressive outbreak) till reaching its maximal extension (the base of the pyramid). From another viewpoint, Wang et al. [2006] proposed two spatio-temporal clustering algorithms (ST-GRID and ST-DBSCAN) for analysis of sequences of seismic events. ST-GRID is based on partitioning of the spatial and temporal dimensions into cells. ST-DBSCAN is an extension of the DBSCAN algorithm to handle spatio-temporal clustering. The k-dist graph proposed in [Ester et al., 1996] as a heuristic for determination of the input parameters was used in both approaches. Hence, in the first step, the k-dist graph was created using spatial and temporal dimensions. By means of the graph, the analyst could infer the suitable thresholds for the spatial and temporal cell lengths. In the second step, the inferred cell lengths are provided to ST-GRID algorithm as an input and the dense clusters are extracted.

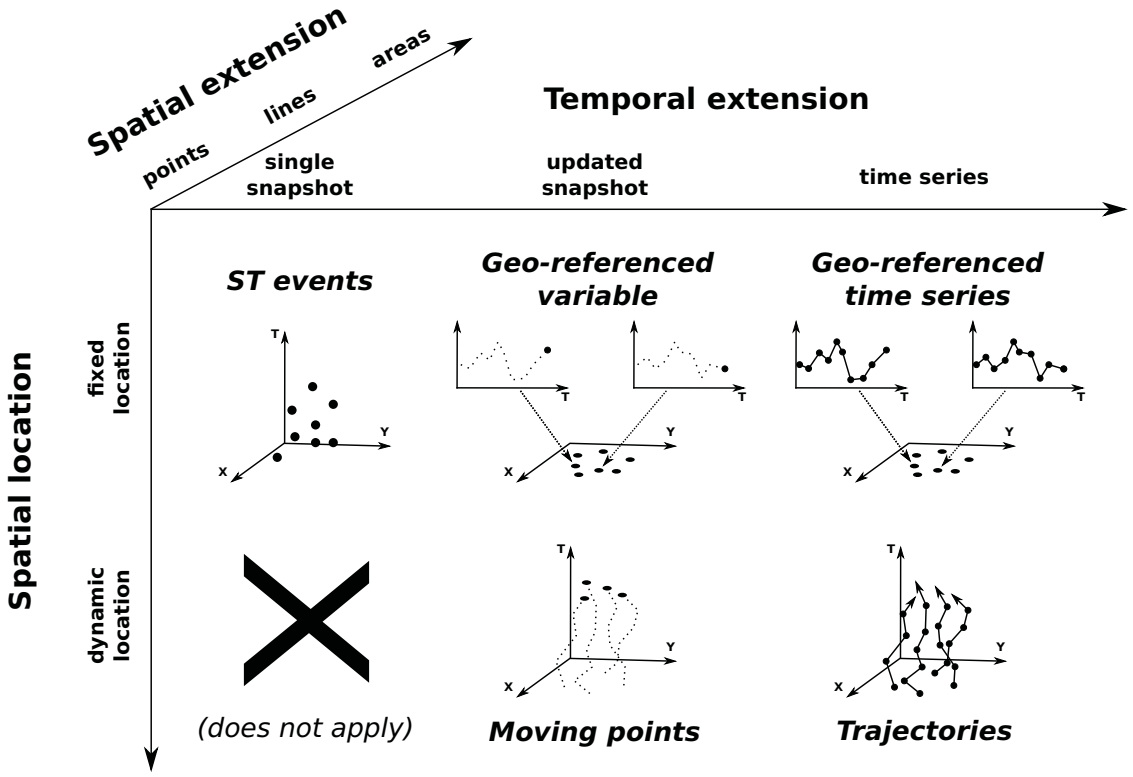


Figure 1.1: Spatio-temporal context

ST-DBSCAN introduced the second parameter of the neighborhood radius in addition to the spatial neighborhood radius ϵ, namely temporal neighborhood radius ϵ_t. These two parameters were determined using k-dist graph and provided to ST-DBSCAN as an input. Thus, point p is considered as *core* when the number of points in the neighborhood is greater or equal to the threshold $MinPts$ within spatial and temporal thresholds.

Geo-referenced variables. When it is possible to observe the evolution in time of some phenomena in a fixed location, we have what is usually called a geo-referenced variable, i.e., the time-changing value of some observed property. In particular, the basic settings might allow only to remember the most recent value of such variable. In this case, the clustering task can be seen as very similar to the case of events discussed above, with the exception that the objects compared refer to the same time instant (the actual time) and their non-spatial features (variables) are not constant. A typical problem in this context consists in efficiently computing a clustering that (i) takes into account both the spatial and non-spatial features, and (ii) exploits the clusters found at the previous time stamp, therefore trying to detect the relevant changes in the data and incrementally update the clusters, rather than computing them from scratch.

Geo-referenced time series. In a more sophisticated situation, it might be possible to store the whole history of the evolving object, therefore providing a (geo-referenced) time-series for the measured variables. When several variables are available, they are usually seen as a single, multidimensional time series. In this case, clustering a set of objects requires to compare the way their time series evolve and to relate that to their spatial position. A classical problem consists in detecting the correlations (and therefore forming clusters) among different time series trying to filter out the effects of spatial auto-correlation, i.e., the mutual interference between objects due to their spatial proximity, e.g., Zhang et al. [2003]. Moreover, spatio-temporal data in the form of sequences of images (e.g., fields describing pressure and ground temperature, remotely sensed from satellites) can be seen as a particular case where location points are regularly distributed in space along a grid.

Moving objects. When (also) the spatial location of the data object is time-changing, we are dealing with moving objects. In the simplest case, the available information about such objects consists in their most recent position, as in the context of real-time monitoring of vehicles for security applications, and no trace of the past locations is kept. As in the case of geo-referenced variables, a typical clustering problem in this context consists in keeping an up-to-date set of clusters through incremental update from previous results, trying to detect the recent changes in the data (in particular, their recent movements) that were significant or that are likely to be followed by large changes in the close future, e.g., due to a change of heading of the object. An example is provided by the work in Li et al. [2004], where a *micro-clustering* technique based on direction and speed of objects is applied to achieve a large scalability.

Trajectories. When the whole history of a moving object is stored and available for analysis, the sequence of spatial locations visited by the object, together with the time-stamps of such visits, form what is called a *trajectory*. Trajectories describe the movement behavior of objects, and therefore clustering can be used to detect groups of objects that behaved in a similar way, for instance by following similar paths (maybe in different time periods), by moving consistently together (i.e., keeping close to each other for long time intervals) or by sharing other properties of movement.

1.2.2 Structure of spatio-temporal data

Data about movers, or, shortly, movement data, represent the movement function $\mu : O \times T \to S$. The most typical format of movement data is a set of position records having the structure <mover identifier, time unit, spatial position>. This structure can also be represented by the formula $O \times T \to S$, which emphasizes that the objects and time units may be, in principle, chosen arbitrarily whereas the spatial position is a measured value depending on the chosen pair of object and time unit. The records may additionally include values of thematic attributes, i.e. the structure may be $O \times T \to S \times A$, where A stands for thematic attributes. Movement data may also be available in the form <mover identifier, trajectory>, where the trajectory specifies the mapping $T \to S$, for instance, by a sequence of pairs ¡time unit, spatial position¿ (in principle, other representations are possible such as sequence of geometric primitives). This form may be encoded as $O \to (T \to S \times A)$. It is equivalent to $O \times T \to S \times A$.

The known methods of position recording include [Andrienko and Andrienko, 2008]:

- Time-based: records are made at regularly spaced time moments, e.g. every 5 minutes
- Change-based: a record is made when mover's position differs from the previous one
- Location-based: a record is made when a mover enters or comes close to a specific location, e.g. where a sensor is installed
- Event-based: positions and times are recorded when certain events occur, in particular, when movers perform certain activities such as mobile phone calling or taking photos
- Various combinations of these basic approaches.

Some methods of data collection may result in movement data with rather fine temporal resolution. This gives a possibility of spatio-temporal interpolation, i.e. using the known positions of a mover for estimating the positions in intermediate time units. In this way, the continuous path of the mover can be approximately reconstructed. Therefore, movement data allowing interpolation between known positions may be called quasi-continuous.

Data about spatial events that do not change their spatial positions have the general structure <event identifier, temporal position, spatial position, values of thematic attributes>, represented by the formula $O \rightarrow T \times S \times A$. For non-spatial events, the data do not have the component representing the spatial position, i.e. the formula is $O \rightarrow T \times A$.

Locations may have static characteristics, which are described by data in the format $S \rightarrow A$, and dynamic characteristics, which may be described by data in the format $S \rightarrow (T \rightarrow A)$. The latter formula means that for each location there is a time series of attribute values. To emphasize the links of places to objects, the formula can be rewritten as $S \rightarrow (T \rightarrow P(O) \times A)$, where $P(O)$ is the power set of the set O. The format $S \rightarrow (T \rightarrow A)$ is equivalent to $S \times T \rightarrow A$, which means that attribute values are specified for various pairs <place, time>.

Characteristics of time units that are not related to locations can be described by data in the format $T \rightarrow A$. To represent spatial configurations of objects, the data structure may be $T \rightarrow (S \rightarrow P(O))$, which is equivalent to $T \times S \rightarrow P(O)$ or $S \times T \rightarrow P(O)$ (the order of T and S in $T \times S$ is irrelevant). Spatial distributions of thematic attribute values can be represented by the structure $T \rightarrow (S \rightarrow A)$, which is equivalent to $T \times S \rightarrow A$ or $S \times T \rightarrow A$. Hence, the same data structures can be used to represent time-dependent characteristics of places and space-dependent characteristics of time units.

Context data describe the environment where the movement takes place: properties of the locations, properties of the time units, spatial objects existing in the space, and/or events that occur during the movement. Context data are not always available. Even when some context data are available, they do not fully describe the context. Therefore, analysis of movement data requires the involvement of analyst's background knowledge about the context. The knowledge may be involved implicitly, when the analyst interprets the data and analytical artefacts obtained, or explicitly, when the analyst constructs new data to be used in the further analysis. Visualization and interactive techniques are required in both cases.

1.2.3 Tasks applicable to geotagged photos in the spatio-temporal context

In terms of the described spatio-temporal framework, the users who take geotagged photos are the *movers*[9]. The sequence of the photo taking events of one mover makes the *trajectory* of this mover. The locations are the spatial positions of the events, which are originally specified as points (by geographical coordinates). By applying spatial generalization, we may generate a discrete set of areas, which will be considered as locations instead of the original points. Depending on the desired temporal scale of analysis, we may choose time intervals of different lengths as time units.

Besides the positions of the movers, the data also describe the photo taking *events*. Each event has its positions in space and time and some thematic attributes: image URL, count of the times the image was viewed, title, and list of tags. However, there are also other kinds of events that can be extracted from the data, in particular, events of several users being in the same location at the same time. These events may be related to other events that occurred in the real world and attracted attention of people: festivals, shows, parties, unusual natural phenomena, etc. They constitute a part of the context in which the movement and activities of the photo owners take place. The context also includes the static geographic environment with the cities, natural areas, landmarks, etc., and events that have no particular positions in space, for example, public holidays. While the geotagged photos do not directly describe the context, some information about the context is contained in the titles of the photos. Tables 1.1 and 1.2 describe the types of tasks for which the geotagged photo data can be used and which partially or fully are covered in subsequent chapters of this thesis.

[9]In the following sections we call the users as *owners* to emphasize that they are the owners of the taken photos

Table 1.1: The type of tasks applicable to geotagged photos in the spatio-temporal context (Part 1)

Focus	**Target characteristics or relations**	**Elementary tasks**	**Synoptic tasks**
Photo taking events	Spatio-temporal positions; relations to locations, times, and context	Positions and relations of particular photos	Spatio-temporal distribution of a set of photos; relations of the distribution to the context
Photo taking events	Spatio-temporal relations among events	Relations (distance, direction) among particular photos	Occurrences of particular types of relations (e.g. concentrations in space and/or time) and their distribution in space and time
Movers (users)	Trajectories; relations to locations, times, and context	Trajectories and relations of particular photographers	Variety of spatial, temporal, and thematic characteristics of trajectories, their spatial, temporal, and frequency distributions and relations to the context
Movers (users)	Spatio-temporal relations among movers	Relations among particular photographers, e.g. proximity, joint travels, similar routes	Occurrences of particular types of relations (e.g. proximity, joint travels, similar routes), their characteristics and distribution in space and time

Table 1.2: The type of tasks applicable to geotagged photos in the spatio-temporal context (Part 2)

Focus	Target characteristics or relations	Elementary tasks	Synoptic tasks
Locations	Presence dynamics; relations to context	Presence dynamics in particular locations and their relations to context	Variety of patterns of presence dynamics (e.g. random fluctuations, seasonal variation, regular peaks, irregular peaks), their spatial distribution and relations to the context
Locations	Relations among locations	Relations among particular locations, e.g. common visitors, sequence of visiting	Occurrences of particular types of relations (e.g. common visitors, sequence of visiting), their characteristics and distribution in space and time
Time units	Spatial configurations; relations to context	Spatial configurations of photos or people in particular time units and their relations to context	Variety of patterns of spatial configurations (e.g. sets of actively visited places), their temporal distribution and relations to context
Time units	Relations among time units	Relations among particular time units, e.g. how the presence differs across the locations, where people moved	Occurrences of particular types of relations (e.g. sudden increase of presence in some/many locations), their characteristics and distribution in space and time

1.3 Data collection

The dataset used in this thesis consists of geotagged photos collected from Flickr, the largest web community for photo sharing, using its publicly available API. The data collection procedure was performed as follows. We seeded initial users by extracting their user ids from several photo groups. First, we extracted information about their geotagged photos and lists of their contacts. Then, iteratively, we retrieved information about geotagged photos of users retrieved from the contact lists. Since June, 2009 until the end of March 2011 we collected metadata for 96,122,832 million geotagged photos covering the entire World.

Geo-referenced (geotagged) photos may introduce positional uncertainty. Flickr offers tools through its web site application that allows a photo to be associated with its corresponding location at various scales ranging from a city or region all the way down to street level. Depending on the choice of scale and the memory of photographer, the accuracy of photo georeferencing will vary. The accuracy may be much improved for more recent postings of photos taken with GPS equipped digital cameras or GPS integrated wireless phones. In some cases, coordinates could refer to the position of the photographer, while in others they could refer to the location of the object being photographed. This justifies a human analyst involvement, in addition to automated analysis, in exploring the data. A review of the collected photos revealed that some temporal information was incorrect. There were 10,925 photos with dates in an incorrect format. In addition, 56,035 photos had dates later than March 2011. There were also 352,538 photographers who uploaded just a single photo; those entries are not suitable for the analysis of photographers' movement (Chapter 3) and frequent sequential patterns (Chapter 5).

Data components that are considered in this thesis include photo ids, photo owners, photo coordinates, comments, comment authors, and timestamps. The set of photos is defined as P, and every photo $p \in P$ is described as a tuple of the following elements: $p = (id, l, u, o, t)$, where id is a unique id of the photo, l the photo's coordinate pair expressed in degrees (latitude and longitude), u the photo's coordinate pair expressed in UTM coordinate system, o is the owner of the photo, t - timestamp associated with the photo (time when the photo was taken). Every photo can contain a set of comments, written by different people including the owner of the photo. Every comment has a timestamp when it was written and all the comments can be sorted according to the timestamps from the oldest to the newest. The set of owners is defined as O, where every owner $o \in O$ can have multiple photos.

Apart for the geotagged photos, we used the Wikipedia database as a source for POI data. This database is an on-going community project aimed at applying geographic annotation to articles describing interesting sites around the world. The database we obtained contains 450,637 entries of various geotagged sites such as cities, landmarks, monuments, buildings, towers, etc. The Wikipedia data was used in Chapter 4.5 for evaluating P-DBSCAN, an extended version of the DBSCAN algorithm, and Chapter 5 for finding frequent sequential patterns of people's movement.

1.4 The scope of the thesis and the contributions

While working with geotagged photos it became clear that different research questions require knowledge in different and sometimes unrelated fields such as:

- (Geo)Visual analytics - visualization, exploration and analysis of spatio-temporal phenomena in combination with digital maps and geocomputation.
- Data mining - clustering and trajectory analysis.
- Text mining and computational linguistics - analysis of textual information (titles, tags, comments).
- Systems engineering - development of analytical tools.

Therefore, the overall goal of this thesis is to contribute to the mentioned domains with regard to the analysis of geotagged photo collections[10].

Chapter 3 presents a geovisual analytics approach to discovering people's preferences for landmarks and movement patterns from photos posted on the Flickr website. The approach combines an exploratory spatio-temporal analysis of geographic coordinates and dates representing locations and time of taking photos with basic thematic information available through the Google Maps Web mapping service, and interpretation of the analyzed area.

In Chapter 4 we discuss two main approaches to discovering attractive places analyzing geotagged photos. The first approach is based on the density of photos taken in the area and is achieved by calculating the relative importance of a photo against other photos taken nearby. The second approach utilizes the opinion and sentiment strength extracted from comments written to a photo in order to estimate its absolute importance. Both approaches allow to build heat maps by converting the importance weights into colors thus facilitating the interactive exploration of attractive areas by the user. In addition, this chapter presents our density-based algorithm P-DBSCAN that improves clustering of attractive areas using the photo ownership information. The contributions of this chapter can be summarized as follows:

- We propose density-based and text analysis methods as ways for enriching existing services and user experience for exploring places and photos.
- We discuss several approaches that improve the running time of the density-based clustering algorithm.
- We discuss the applicability of the discrete point visualization of interesting places using geotagged photos as opposed to interpolation techniques. The discrete point visualization allows displaying layered maps at different scales without recalculating photo weights and at zero runtime overhead.
- We propose a refinement and specialization of general density-based clustering algorithms for analysis of spatial events using geo-tagged photos.
- We present ownership: a new specialized requirement for an object clustering which reduces cluster bias incurred by subjective nature of photos taken by different people and increases robustness to noise.

[10]We, however, completely excluded automated multimedia analysis because it is beyond the scope of this thesis. We refer the reader interested in works that involve image processing of geotagged photos to studies on the estimation of location of photographs and scene summarization: [Simon et al., 2007, Kennedy and Naaman, 2008, Snavely et al., 2010]

In Chapter 5 we provide a practical approach for modeling and mining frequent sequence patterns using the collection of geotagged photos. Our approach is supported by any GIS-based databases and do not require any extensions or special query languages. The contribution of this chapter is the development of a new data mining process that employs concepts that have been developed in various other fields such as bioinformatics and artificial intelligence.

Chapter 6 presents our approach to extracting opinion and sentiments from photo comments. The contribution of the chapter to the area of computational linguistic can be summarized as follows:

- Our model is based on the corpus extracted from users' photo comments.
- We construct and employ a finite lexicon of opinion words in contrast to the majority of approaches in which seed lists are used to infer scores of unknown opinion words.
- We develop a model that consists of two types of scores: *opinion* regarding the photo and *sentiment* towards the subject of the photo. For this purpose, we suggest a semi-automatic extraction of photo features and a set of syntactic opinion reference patterns.
- We model the orientation strength based on word distributions without using any external dictionaries, while the semantic orientation (positive or negative) of a word is determined by the predefined lexicon of positive and negative opinion-bearing words.
- We provide a continuous scale for opinion and sentiment orientation.
- With our approach, we allow for dynamic updates of scores when new comments are added to the system, which makes the whole method readily applicable in real-world tasks.
- As the basis for our approach, we conducted a carefully designed extensive user study. Apart from demonstrating the performance of our approach, the user study provided further interesting insights on how users perceive opinions and sentiments in photo comments.

In Chapter 7 we present an extensible framework that can solve generic spatio-temporal analysis tasks and was developed primarily to handle geotagged photos and support our work on this thesis. The proposed framework, termed GEO-SPADE, uses Google Earth as a primary visualization platform and data interchange system. Pluggable components can easily be integrated into the framework. While most of the Google Earth-based frameworks proposed in the literature are web-based and are designed to perform very narrow tasks, our framework has a plugin architecture, which allows to add different components aimed at solving different spatio-temporal tasks.

The conclusions and future work are summarized in Chapter 8. The interrelation of Chapters and their research domains are schematically shown in Figure 1.2

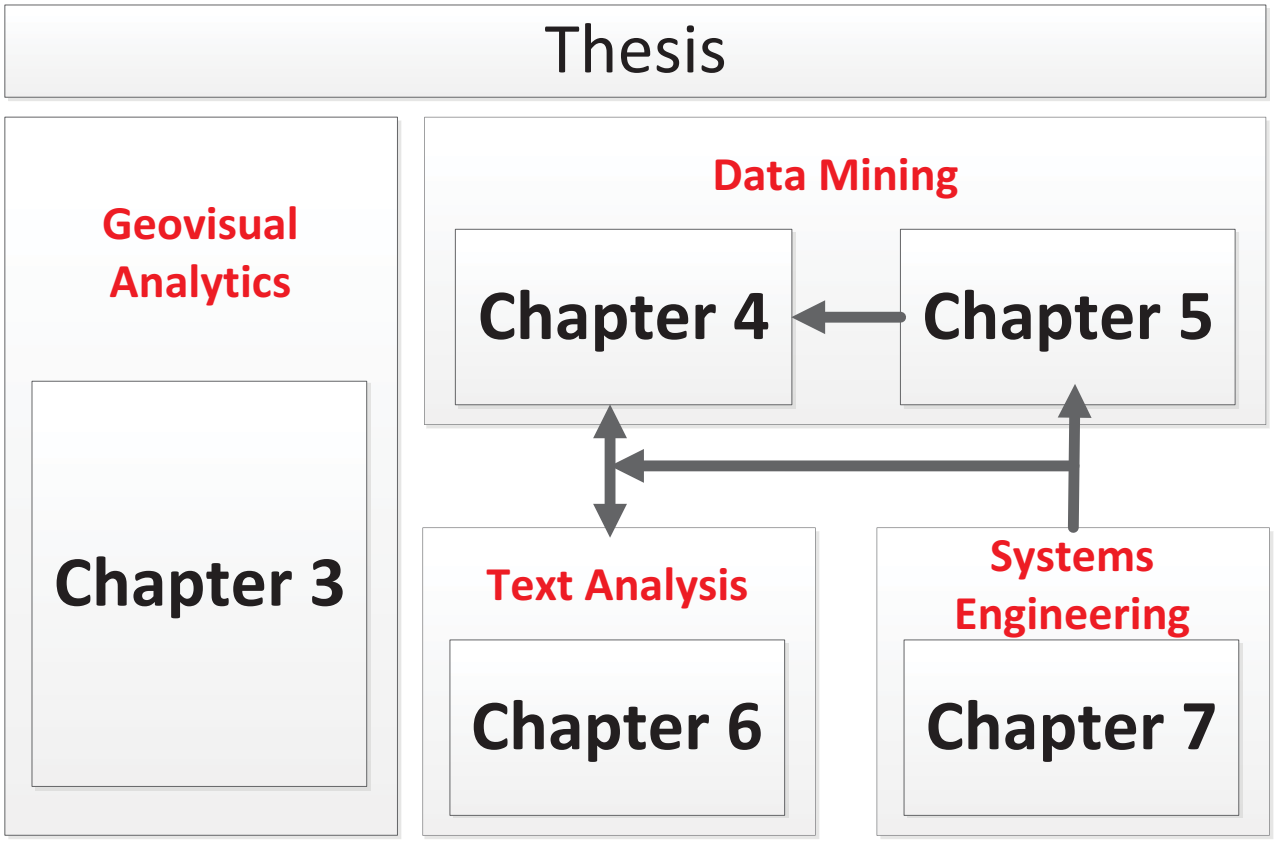


Figure 1.2: Interrelation of chapters and research domains (arrows show the dependence of a technique or method described in a chapter to the one the arrows points to)

2
Related Work

Contents

2.1 Analysis of geotagged photos 47
2.2 Clustering methods for spatio-temporal data 48
2.2.1 Descriptive and generative model-based clustering 49
2.2.2 Distance-based clustering methods 49
2.2.3 Density-based methods and the DBSCAN family 50
2.2.4 Visual-aided approaches 52
2.2.5 Important places 53
2.2.6 Patterns and frequent sequences 53
2.2.7 Other clustering methods 55
2.3 Opinion and Sentiment Analysis 57
2.4 Google Earth-based tools and frameworks 58

We begin this chapter by describing the works that involve geotagged photos. Then we focus on different clustering methods since clustering is the most important method in the analysis of spatio-temporal data including geotagged photos. Clustering of photos using density based approach is covered extensively in Chapter 4. We bring some related work in the area of opinion and sentiment analysis and finish this chapter by presenting state of the art in GIS framework development using Google Earth.

2.1 Analysis of geotagged photos

Text analysis and multimedia communities realized early on a potential of geotagged photos as the valuable source of visual and textual information about people's preferences for landmarks and events. Landmark identification and retrieval of representative images using MeanShift, a non-parametric clustering algorithm, was performed by Crandall et al. [2009]. Jaffe et al. [2006] applied a hierarchical clustering algorithm on the collection of geotagged photos to find tags that characterized a given cluster. In the subsequent work Ahern et al. [2007] used k-means algorithm instead of hierarchical clustering. The scale-structure identification approach was proposed by Rattenbury et al. [2007] to identify the distribution of events utilizing the coordinates of geotagged photos, timestamps, tags and titles, while Kennedy et al. [2007] proposed a method

of retrieving representative images using visual features. Becker et al. [2009] applied an ensemble clustering technique to identify photos posted on Flickr web site that belonged to the same event. Their clustering technique combined multiple textual, temporal and geographical features such as titles, tags, timestamps, and geographic coordinates. Interestingly enough, the best results were obtained when only tags were considered in clustering.

In the mashup tools for visualization and analysis of geotagged photos such as TagMap [Jaffe et al., 2006] or World Explorer [Ahern et al., 2007] clustering of an area was performed using the distribution of photo tags (and/or other textual information). Only one photo representative of a given cluster was displayed on a map, making it difficult to know what else was interesting in the area covered by the cluster. Since the processing time was not reported it would be difficult to plan the deployment of these tools in real time analysis of spatial distribution of geotagged photos. The visual analytics approach proposed in this research (see Chapter 4) differs from the previously developed mashup tools in offering a scalable and flexible photo clustering and exploration of geotagged photo distribution patterns.

Girardin et al. [2007] used heatmaps to visualize concentrations of tourists inferred from geotagged photos of places frequented by tourists. In their mapping approach, the studied area was divided into rectangular cells counting the number of photos and photo owners in each cell as a measure of concentration. The heatmaps were produced using spatial interpolation over every cell. The same approach was used in the subsequent work reported elsewhere [Girardin et al., 2008b,a] albeit without providing a rationale for it and giving no description of interpolation parameters. Fisher [2007] proposed Hotmap, a mashup visualization of the usage of Microsoft's Live Search Maps by displaying heatmaps over the number of downloaded tile images. He also discussed how Hotmap could be used to depict the places of potential touristic interest by analyzing locations the user was looking at while working with Live Search Maps. Fisher (ibid) proposed logarithmic color scaling to increase the variance in color and to differentiate between non-popular and popular places that most people are looking at. The heatmap techniques used in Girardin et al. [2007, 2008b,a], Fisher [2007] are based on interpolation of data between points of known values and are commonly used when data represent smooth, continuous phenomena [Slocum et al., 2008, Kimerling et al., 2009]. Common interpolation methods used for generating heatmaps include triangulation, inverse distance interpolation, kriging [Isaaks and Srivastava, 1989, Stein, 1999] or kernel density estimation [Silverman, 1986, Chainey and Tompson, 2008]. Gaussian filters [Wells, 1986], originally used for image or mesh smoothing, and Alpha Blending [Unwin et al., 2006] are additional interpolation techniques that have been adopted from image processing. Each of these techniques has advantages and disadvantages. The selection of a particular technique is usually a result of many considerations such as the need to report interpolation error variance, sensitivity to the model specification, complexity of interpolation technique, speed of execution, and the ease of using a particular technique. In this research we use a true point visualization of locations where photos were taken to save computational time during interactive exploration of interesting places (see Chapter 4).

2.2 Clustering methods for spatio-temporal data

Here we will focus on the context of moving objects that can be traced along the time, resulting in trajectories that describe their movements. On one hand, trajectories represent the most

complex and promising (from a knowledge extraction viewpoint) form of data among those based on point-wise information. On the other hand, point-wise information is becoming nowadays largely available and usable in real contexts, while spatio-temporal data with more complex forms of spatial components are still rarely seen in real world problems – exception made for a few, very specific contexts, such as climate monitoring.

Clustering is one of the general approaches to a descriptive modeling of a large amount of data, allowing the analyst to focus on a higher level representation of the data. Clustering methods analyze and explore a dataset to associate objects in groups, such that the objects in each groups have common characteristics. These characteristics may be expressed in different ways: for example, one may describe the objects in a cluster as the population generated by a joint distribution, or as the set of objects that minimize the distances from the centroid of the group.

2.2.1 Descriptive and generative model-based clustering

The objective of this kind of methods is to derive a global model capable of describing the whole dataset. Some of these methods rely on a definition of multivariate density distribution and look for a set of fitting parameters for the model. In Gaffney and Smyth [1999] it is proposed a clustering method based on a mixture model for continuous trajectories. The trajectories are represented as functional data, i.e. each individual is modeled as a sequence of measurement given by a function of time depending on a set of parameters that models the interaction of the different distributions. The objects that are likely to be generated from a core trajectory plus gaussian noise are grouped together by means of the EM algorithm. In a successive work Chudova et al. [2003], spatial and temporal shift of trajectories within each cluster is also considered. Another approach based on a model-based technique is presented in Alon et al. [2003], where the representative of a cluster is expressed by means of a Markov model that estimates the transition between successive positions. The parameter estimation task for the model is performed by means of EM algorithm.

2.2.2 Distance-based clustering methods

Another approach to cluster complex form of data, like trajectories, is to transform the complex objects into features vectors, i.e. a set of multidimensional vectors where each dimension represents a single characteristic of the original object, and then to cluster them using generic clustering algorithms, like, for example, k-means. However, the complex structure of the trajectories not alway allows an approach of this kind, since most of these methods require that all the vectors are of equal length. In contrast to this, one of the largely adopted approach to the clustering of trajectories consists in defining distance functions that encapsulate the concept of similarity among the data items.

Using this approach, the problem of clustering a set of trajectories can be reduced to the problem of choosing a generic clustering algorithm, that determines how the trajectories are joined together in a cluster, and a distance function, that determines which trajectories are candidate to be in the same group. The chosen method determines also the "shape" of the resulting clusters: center-based clustering methods, like *k-means*, produce compact, spherical clusters around a set of centroids and are very sensitive to noisy outliers; hierarchical clusters organize the data items

in a multi-level structure; density-based clustering methods form maximal, dense clusters, not limiting the groups number, the groups size and shape.

The concepts of similarities of spatio-temporal trajectories may vary depending on the considered application scenario. For example, two objects may be considered similar if they have followed the same spatio-temporal trajectory within a given interval, i.e. they have been in the same places at the same times. However, the granularity of the observed movements (i.e. the number of sampled spatio-temporal points for each trajectory), the uncertainty on the measured points, and, in general, other variations of the availability of the locations of the two compared objects have required the definition of several similarity measures for spatio-temporal trajectories. The definition of these measures is not only tailored to the cluster analysis task, but it is strongly used in the field of Moving Object Databases for the similarity search problem [Theodoridis, 2003], and it is influenced also by the work on time-series analysis [Agrawal et al., 1993, Berndt and Clifford, 1996, Chan and chee Fu, 1999] and Longest Common Sub Sequence (LCSS) model [Vlachos et al., 2002, 2003, Chen et al., 2005]. The distance functions defined in [Nanni and Pedreschi, 2006, Pelekis et al., 2007] are explicitly defined on the trajectory domain and take into account several spatio-temporal characteristics of the trajectories, like direction, velocity and co-location in space and time.

2.2.3 Density-based methods and the DBSCAN family

The density-based clustering methods use a density threshold around each object to distinguish the relevant data items from noise. DBSCAN [Ester et al., 1996], one of the first examples of density-based clustering, visits the whole dataset and tags each object either as *core object* (i.e. an object that is definitively within a cluster), *border object* (i.e. objects at the border of a cluster), or *noise* (i.e. objects definitively outside any cluster). After this first step, the core objects that are close to each other are joined in a cluster. In this method, the density threshold is espressed by means of two parameters: a maximum radius ϵ around each object, and a minimum number of objects, say *MinPts*, within this interval. An object p is defined a *core object* if its neighborhood of radius ϵ (denoted as $N_\epsilon(p)$) contains at least *MinPts* objects. Using the core object condition, the input dataset is scanned and the status of each object is determined. A cluster is determined both by its core objects and the objects that are reachable from a core object, i.e. the objects that do not satisfy the core object condition but that are contained in the Eps-neighborhood of a core object. The concept of "reachable" is express in terms of the *reachability distance*. It is possible to define two measures of distances for a core object c and an object in its ϵ-neighborhood: the *core distance*, which is the distance of the *MinPts*-th object in the neighborhood of c in order of distance ascending from c, and the *reachability distance*, i.e. the distance of an object p from c except for the case when p's distance is less than the *core distance*; in this case the distance is normalized to the *core distance*. Given a set of *core* and *border* object for a dataset, the clusters are formed by visiting all the objects, starting from a core point: the cluster formed by the single point is extended by including other objects that are within a reachability distance; the process is repeated by including all the objects reachable by the new included items, and so on. The growth of the cluster stops when all the border points of the cluster have been visited and there are no more reachable items. The visit may continue from another core object, if available.

The notion of density-connectivity presented in [Ester et al., 1996], served as a starting

point for a number of density-based clustering algorithms OPTICS [Ankerst et al., 1999], LDBSCAN [Duan et al., 2007], to name a few. Improvements suggested in later research aimed at generalization of clustering approaches [Hinneburg and Keim, 1998], efficient selection of input parameters [Ankerst et al., 1999], performance optimization [Brecheisen et al., 2006], solving the problem of local densities [Duan et al., 2007] and introducing a specialization for particular tasks, such as moving clusters [Kalnis et al., 2005], trajectory clustering [Palma et al., 2008], wild bird migration [Tang et al., 2011] and even document clustering [Zhao et al., 2011].

Due to simplicity in the implementation, DBSCAN became a basis for multitude of specializations and extensions. The number of only recent journal publications including DBSCAN-based implementations reached 10:[Viswanath and Suresh Babu, 2009, Nasibov and Ulutagay, 2009, Folino et al., 2009, Yue et al., 2010, de Oliveira et al., 2010, Ruiz et al., 2010, Jiang et al., 2011, Chen et al., 2011, Zhao et al., 2011, Tang et al., 2011]. The number of recent conference publications is even higher: [Huang et al., 2009, Vieira et al., 2009, Jian et al., 2009, Yang et al., 2009, Ibrahim et al., 2009, Deng et al., 2009, Ali et al., 2010, Tepwankul and Maneewongvatana, 2010, Kramer and Danielsiek, 2010, Rosswog and Ghose, 2010, Chen et al., 2010, Parker et al., 2010, Tepwankul and Maneewongwattana, 2010, Santhisree et al., 2010]. The popularity of DBSCAN and its simplicity in logic and implementation were one of the core reasons for adopting it as a basis for extension in this thesis presented in Chapter 4.

The OPTICS method [Ankerst et al., 1999] proceeds by exploring the dataset and enumerating all the objects. For each object p it checks if the core object conditions are satisfied and, in the positive case, starts to enlarge the potential cluster by checking the condition for all neighbors of p. If the object p is not a core object, the scanning process continues with the next unvisited object of D. The results are summarized in a reachability plot: the objects are represented along the horizontal axis in the order of visiting them and the vertical dimension represents their reachability distances. Intuitively, the reachability distance of an object p_i corresponds to the minimum distance from the set of its predecessors p_j, $0 < j < i$. As a consequence, a high value of the reachability distance roughly means a high distance from the other objects, i.e. indicates that the object is in a sparse area. The actual clusters may be determined by defining a reachability distance threshold and grouping together the consecutive items that are below the chosen threshold in the plot. The result of the OPTICS algorithm is insensitive to the original order of the objects in the dataset. The objects are visited in this order only until a core object is found. After that, the neighborhood of the core object is expanded by adding all density-connected objects. The order of visiting these objects depends on the distances between them and not on their order in the dataset. It is also not important which of density-connected objects will be chosen as the first core object since the algorithm guarantees that all the objects will be put close together in the resulting ordering. A formal proof of this property of the algorithm is given in [Ester et al., 1996].

It is clear that the density methods strongly rely on an efficient implementation of the neighborhood query. In order to improve the performances of such algorithms it is necessary to have the availability of valid index data structure. The density based algorithms are largely used in different context and they take advantages of many indices like R-tree, kd-tree, etc. When dealing with spatio-temporal data, it is necessary to adapt the existing approaches also for the spatio-temporal domain [Frentzos et al., 2007] or use a general distance based index (e.g. M-tree, [Ciaccia et al., 1997])

The approach of choosing a clustering method and a distance function is just a starting

point for a more evolute approach to mining. For example, in [Nanni and Pedreschi, 2006] the basic notion of the distance function is exploited to stress the importance of the temporal characteristics of trajectories. The authors propose a new approach called *temporal focusing* to better exploit the temporal aspect and improve the quality of trajectory clustering. For example, two trajectories may be very different if the whole time interval is considered. However, if only a small sub-interval is considered, these trajectories may be found very similar. Hence, it is very crucial for the algorithm to efficiently work on different spatial and temporal granularities. As mentioned by the authors, usually some parts of trajectories are more important than others. For example, in rush hours it can be expected that many people moving from home to work and vice versa form movement patterns that can be grouped together. On weekends, people's activity can be less ordered where the local distribution of people is more influential than collective movement behavior. Hence, there is a need for discovering the most interesting time intervals in which movement behavior can be organized into meaningful clusters. The general idea of the time focusing approach is to cluster trajectories using all possible time intervals (time windows), evaluate the results and find the best clustering. Since the time focusing method is based on OPTICS, the problem of finding the best clusters converges to finding the best input parameters. The authors proposed several quality functions based on density notion of clusters that measures the quality of the produced clustering and are expressed in terms of average reachability [Ankerst et al., 1999] with respect to a time interval I and reachability threshold ϵ'. In addition, ways of finding optimal values of ϵ' for every time interval I were provided.

2.2.4 Visual-aided approaches

Analysis of movement behavior is a complex process that requires understanding of the nature of the movement and phenomena it incurs. Automatic methods may discover interesting behavioral patterns with respect to the optimization function but it may happen that these patterns are trivial or wrong from the point of view of the phenomena that is under investigation. The (geo)visual analytics field [MacEachren and Kraak, 2001, Dykes et al., 2005, Andrienko and Andrienko, 2006, Keim et al., 2008] tries to overcome the issues of automatic algorithms introducing frameworks implementing various visualization approaches of spatio-temporal data and proposing different methods of analysis including trajectory aggregation, generalization and clustering [Andrienko and Andrienko, 2006, Andrienko et al., 2007b, Andrienko and Andrienko, 2008, Andrienko et al., 2009, Andrienko and Andrienko, 2009]. These tools often target different application domains (movement of people, animals, vehicles) and support many types of movement data [Andrienko et al., 2007b]. The advantages of visual analytics in analysis of movement data is clear. The analyst can control the computational process by setting different input parameters, interpret the results and direct the algorithm towards the solution that better describes the underlying phenomena.

In [Rinzivillo et al., 2008] the authors propose progressive clustering approach to analyze the movement behavior of objects. The main idea of the approach is the following. The analyst or domain expert progressively applies different distance functions that work with spatial, temporal, numerical or categorical variables on the spatio-temporal data to gain understanding of the underlying data in a stepwise manner. This approach is orthogonal to commonly used approaches in machine learning and data mining where the distance functions are combined together to optimize the outcome of the algorithm.

2.2.5 Important places

In the work of Kang et al. [2004], the authors proposed an incremental clustering for identification of important places in a single trajectory. Several factors for the algorithm were defined: arbitrary number of clusters, exclusion of as much unimportant places as possible and being not computationally expensive to allow running on mobile devices. The algorithm is based on finding important places where many location measurements are clustered together. Two parameters controlled the cluster creation - distance between positions and time spent in a cluster. The basic idea is the following. Every new location measurement provided by a location-based device (Place Lab, in this case) is compared to the previous location. If the distance between previous location is less than a threshold, the new location is added to the previously created cluster. Otherwise, the new candidate cluster is created with the new location. The candidate cluster becomes a cluster of important places when the time difference between first point in a cluster and the last point is greater than the threshold. Similar ideas of finding interesting places in trajectories were used in later works [Alvares et al., 2007b, Zheng et al., 2009].

A similar task was performed in [Palma et al., 2008], this time by using speed characteristics. For this, the original definition of DBSCAN was altered to accommodate the temporal aspect. Specifically, the point p of a trajectory called *core* point if the time difference between first and last neighbor points of p was greater or equal to some predefined threshold $MinTime$ (minimum time). This definition corresponds to the maximum average speed condition $\epsilon/MinTime$ in the neighborhood of point p. Since original DBSCAN requires two parameters to be provided for clustering: ϵ - radius of the neighborhood and $MinPts$ - minimum number of points in the neighborhood of p, similarly, the adopted version required providing two parameters: ϵ and $MinTime$. However, without knowing the characteristic of the trajectory it is difficult for the user to provide meaningful parameters. The authors proposed to regard the trajectory as a list of distances between two consecutive points and obtain means and standard deviations of these distances. Then, Gaussian curve can be plotted using these parameters that should give some information about the properties of the trajectory and inverse cumulative distribution function can be constructed expressed in terms of mean and standard deviation. In order to obtain ϵ, the user should provide a value between 0 and 1 that reflects the proportion of points that can be expected in a cluster.

2.2.6 Patterns and frequent sequences

Patterns that are mined from trajectories are called *trajectory patterns* and characterize interesting behaviors of single object or group of moving objects [Fosca and Dino, 2008]. Different approaches exist in mining trajectory patterns. We present two examples. The first one is based on grid-based clustering and finding dense regions [Giannotti et al., 2007], the second is based on partitioning of trajectories and clustering of trajectories' segments [Kang and Yong, 2009].

Giannotti et al. [2007] presented an algorithm to find frequent movement patterns that represent cumulative behavior of moving objects where a pattern, called *T-pattern*, was defined as a sequence of points with temporal transitions between consecutive points. A *T-pattern* is discovered if its spatial and temporal components approximately correspond to the input sequences (trajectories). The meaning of these patterns is that different objects visit the same places with similar time intervals. Once the patterns are discovered, the classical sequence mining algorithms

can be applied to find frequent patterns. Crucial to the determination of *T-patterns* is the definition of the visiting regions. For this, the *Region-of-Interest (RoI)* notion was proposed. A *RoI* is defined as a place visited by many objects. Additionally, the duration of stay can be taken into account. The idea behind *Roi* is to divide the working region into cells and count the number of trajectories that intersect the cell. The algorithm for finding popular regions was proposed, which accepted the grid with cell densities and a density threshold δ as input. The algorithm scans the cells and tries to expand the region in four directions (left, right, up, down). The direction that maximizes the average cell density is selected and the cells are merged. After the regions of interest are obtained, the sequences can be created by following every trajectory and matching the regions of interest they intersect. The timestamps are assigned to the regions in two ways: (1) Using the time when the trajectory entered the region or (2) Using the starting time if the trajectory started in that region. Consequently, the sequences are used in mining frequent *T-patterns*. The proposed approach was evaluated on the trajectories of 273 trucks in Athens, Greece having 112,203 points in total.

Kang and Yong [2009] argues that methods based on partition of the working space into grids may lose some patterns if the cell lengths are too large. In addition, some methods require trajectory discretization according to its recorded timestamps which can lead to creation of redundant and repeating sequences in which temporal aspects are contained in the sequentially ordered region ids. As a workaround to these issues, the authors proposed two refinements: (1) Partitioning trajectories into disjoint segments, which represent meaningful spatio-temporal changes of the movement of the object. The segment is defined as an area having start and end points as well as the time duration within the area. (2) Applying clustering algorithm to group similar segments. A *ST-pattern* (Spatio-temporal pattern) was defined as a sequences of segments (areas) with time duration described as a height of 3-dimensional cube. Thus, the sequences of *ST-patterns* are formed by clustering similar cubes. A four-step approach was proposed to mine frequent *ST-patterns*. In the first step, the trajectories are simplified using the DP (Douglas-Peucker) algorithm dividing the trajectories into segments. The segments are then normalized using linear transformation to allow comparison between segments having different offsets. In the next step, the spatio-temporal segments are clustered using the BIRCH [Zhang et al., 1996] algorithm. In the final step, a DFS-based (depth-first search) method is applied on the clustered regions to find frequent patterns.

The mining of frequent sequential patterns in databases of customer transactions was first presented by Agrawal and Srikant [1994]. The method adopts an a-priori-like approach [Agrawal et al., 1994] where the idea is to find subsets that are common to at least a minimum number of sequences, termed itemsets. The method uses the following observation: if the sequence of length k is not frequent, then neither can the sequence of length $k+1$ ever be frequent. The algorithm can be applied to generic items provided they can be sorted using transaction time. Time, however, is not considered in pattern mining. The limitation of the approach is that it cannot find sequences with repeating patterns and sequences in which patterns are not necessarily immediate antecedents.

There are application domains where time duration between adjacent events is also important. This issue was addressed in MiSTA, a generic algorithm for mining temporally annotated sequences, where frequent patterns are mined using sequence and temporal similarity [Giannotti et al., 2006]. As an extension to MiSTA, Giannotti et al. [2007] presented three different approaches to mining trajectory sequences that are reflecting site visits at approximately the same

time. These two approaches share the idea that the transformation of a trajectory into a sequence of significant parts and the application of semantic meaning are done as a preprocessing step prior to mining the sequence patterns. Since the trajectory is transformed into a sequence of generic events, the MiSTA algorithm can be directly applied to them.

The MiSTA authors suggested two general methods for performing preprocessing. In the first case, background knowledge should be applied to trajectories. To perform this task may require an additional database of POIs or a domain expert. In the second case, significant parts are found without using background knowledge, only the properties of the trajectories themselves. Specifically, the authors proposed to divide the area of investigation into grids and to count the density of trajectories in every grid. Thus, the significant places are defined in terms of frequency of visits by different persons. In contrast to temporal annotated sequences, we define sequences as a frequent move from one place to another without regard to time similarity.

Alvares et al. [2007b] proposed a generic model for semantically annotating trajectories and representing a moving pattern in the geographic database. This approach has two main parts. In the first part, the significant places in a trajectory are found by identifying moves and stops [Spaccapietra et al., 2008]. Stops are significant places that are also called stay points, sites where a person stayed for a certain period of time. The extraction of stops depends on time and distance thresholds. Moves are transitions between consecutive stops. In the second part, stops are integrated into the database along with geographic data like POIs. This makes it possible to perform spatial queries on stop regions by annotating them with semantically meaningful information. They demonstrated this approach for mining frequent trajectory patterns between two stops of conference attendees [Alvares et al., 2007a].

Zheng et al. [2009] mine travel sequences by inferring interesting places from trajectories and the person's experience. The method is based on calculating probabilities that a person will take a specific path using information about how many people move from one place to another. The most interesting sequences of length n can be found by summing the probabilities of every two-length sequence comprising the larger sequence and selecting sequences with high score. However, the notion of such sequences differs from classical sequences based on the frequency of patterns. The authors report that finding sequences of length n is possible but a time consuming process and hypothesize that people would not likely visit many places in a trip. Thus, two-length sequences were only considered in their paper.

2.2.7 Other clustering methods

Micro clustering methods In Hwang et al. [2005] a different approach is proposed, where trajectories are represented as piece-wise segments, possibly with missing intervals. The proposed method tries to determine a *close time interval*, i.e. a maximal time interval where all the trajectories are pair-wise close to each other. The similarity of trajectories is based on the amount of time in which trajectories are close and the mining problem is to find all the trajectory groups that are close within a given threshold.

A similar approach based on an extension of *micro-clustering* is proposed in Li et al. [2004]. In this case, the segments of different trajectories within a given rectangle are grouped together if they occur in similar time intervals. The objective of the method is to determine the maximal group size and temporal dimension within the threshold rectangle.

In Lee et al. [2007], the trajectories are represented as sequences of points without explicit

temporal information and they are partitioned into a set of quasi-linear segments. All the segments are grouped by means of a density based clustering method and a representative trajectory for each cluster is determined.

Flocks and convoys In some application domains there is a need in discovering group of objects that move together during a given period of time. For example, migrating animals, flocks of birds or convoys of vehicles. Kalnis et al. [2005] proposed the notion of *moving clusters* to describe the problem of discovery of sequence of clusters in which objects may leave or enter the cluster during some time interval but having the portion of common objects higher than a predefined threshold. Other patterns of moving clusters were proposed in the literature: Gudmundsson and van Kreveld [2006], Vieira et al. [2009] define a flock pattern, in which the same set of objects stay together in a circular region of a predefined radius, while Jeung et al. [2008] defines a convoy pattern, in which the same set of objects stay together in a region of arbitrary shape and extent.

Kalnis et al. [2005] proposed three algorithms for discovery of moving clusters. The basic idea of these algorithms is the following. Assuming that the locations of each object were sampled at every timestamp during the lifetime of the object, a snapshot $S_{t=i}$ of objects' positions is taken at every timestamp $t = i$. Then, DBSCAN [Ester et al., 1996], a density-based clustering algorithm, is applied on the snapshot forming clusters $c_{t=i}$ using density constraints of *MinPts* (minimum points in the neighborhood) and ϵ (radius of the neighborhood). Having two snapshots clusters $c_{t=i}$ and $c_{t=i+1}$, the moving cluster $c_{t=i}c_{t=i+1}$ is formed if $\frac{|c_{t=i} \cap c_{t=i+1}|}{|c_{t=i} \cup c_{t=i+1}|} > \theta$, where θ is an integrity threshold between 0 and 1.

Jeung et al. [2008] adopts DBSCAN algorithm to find candidate convoy patterns. The authors proposed three algorithms that incorporate trajectory simplification techniques in the first step. The distance measures are performed on the segments of trajectories as opposed to commonly used point based distance measures. They show that the clustering of trajectories at every timestamp as it is performed in moving clusters is not applicable to the problem of convoy patterns because the global integrity threshold θ may be not known in advance and time constraint (lifetime) is not taken into account, which is important in convoy patterns. Another problem is related to the trajectory representation: Some trajectories may have missing timestamps or be measured at different time intervals. Therefore, the density measures cannot be applied between trajectories with different timestamps. To handle the problem of missing timestamps, the authors proposed to interpolate the trajectories creating virtual time points and apply density measures on segments of the trajectories. Additionally, the convoy was defined as candidate when it had at least k clusters during k consequent timestamps.

Five on-line algorithms for discovery flock patterns in spatio-temporal databases were presented in [Vieira et al., 2009]. The flock pattern Φ is defined as the maximal number of trajectories and greater or equal to density threshold μ that move together during minimum time period δ. Additionally, the disc with radius $\epsilon/2$ with the center $c_k^{t_i}$ of the flock k at time t_i should cover all the points of flock trajectories at time t_i. All the algorithms employ the grid-based structure. The input space is divided into cells with edge size ϵ. Every trajectory location sampled at time t_i is placed in one of the cells. After processing all the trajectories at time t_i, a range query with radius ϵ is performed on every point p to find neighbor points whose distance from p is at most ϵ and the number of neighbor points is not less than μ. Then, for every pairs of points

found, density of neighbor points with minimum radius $\epsilon/2$ is determined. If the density of a disk is less than μ, the disk is discarded otherwise the common points of two valid disks are found. If the number of common points is above the threshold then the disk is added to a list of candidate disks. In the basic algorithm that generate flock patterns, the candidate disk at time t_i is compared to the candidate disk at time t_{i-1} and augmented together if they have the common number of points above the threshold. The flock is generated if the augmented clusters satisfy the time constraint δ. In other four proposed algorithms, different heuristics were applied to speed up the performance by improving generation of candidate disks. In one of the approaches called *Cluster Filtering Evaluation*, DBSCAN with parameters μ as a density threshold and ϵ for neighborhood radius is used to generate candidate disks. Once candidate disks are obtained, the basic algorithm for finding flocks is applied. This approach works particularly well when trajectory dataset is relatively small and many trajectories have similar moving patterns.

2.3 Opinion and Sentiment Analysis

Existing approaches in the context of opinion analysis can be broadly divided into several categories. The following categories are closely related to our work: *opinion classification*, *lexicon generation*, and *feature-based opinion analysis*. A more detailed overview can be found in Liu [2009].

A Naïve Bayes Classifier was used in Salvetti et al. [2004] for classifying movie reviews, while Das and Chen [2007] used Naïve Bayes as one of five classifiers with majority voting. A Support Vector Machine (SVM) classifier was used by Gamon [2004] for classifying customer feedback data. O'Hare et al. [2009] applied SVM on financial blogs. An unsupervised approach for review classification was applied in Turney [2002] based on the calculation of pointwise mutual information (PMI) among potential opinion phrases in a large-scale web corpus. Subrahmanian and Reforgiato [2008] proposed a real-valued scale opinion orientation based on a classification of adverbs, different verb categories and complex relationships of adverbs, adjectives and verbs in the text. The mentioned classifications are used to separate comments according to their opinion orientation or in order to asses opinion strength. In our approach, we additionally separate opinions about the photo quality from sentiments about the content.

Additional approaches to learn the semantic orientation of words utilize external resources like WordNet [Fellbaum, 1998] by measuring relative distance of an arbitrary word to words "good" and "bad" [Kamps et al., 2004] or by utilizing a random walk model on the graph of word relations [Hassan and Radev, 2010]. Esuli and Sebastiani [2006] generated a dictionary called SentiWordNet using WordNet with three sentiment scores (positive, negative and objective) for each WordNet synset. Other approaches rely on seed lists containing words with a known semantic orientation and search corpora for specified adjective-adjective relations [Hatzivassiloglou and McKeown, 1997], adjective-product feature relations [Qiu et al., 2009, Jijkoun et al., 2010] or bag-of-word vector space similarities [Sahlgren et al., 2007]. Chesley et al. [2006] used a Wikipedia dictionary to determine the polarity of adjectives. We use a predefined opinion lexicon, the Internet General Inquirer lexicon[1], and complement it with a statistically motivated adjective weighting model.

[1] http://www.webuse.umd.edu:9090/

In addition to the approaches that try to detect the sentiment of sentences or even documents as a whole, the task of feature-based analysis is to investigate to which feature (e.g. entity, topic, attribute) sentiments or opinions refer. Some of the feature-based analysis methods use distance-based heuristics [Ding et al., 2008, Oelke et al., 2009]: The closer an opinion word is to a feature word, the higher its influence on the feature is. Other approaches exploit advanced natural language processing methods, like dependency parsers, to resolve linguistic references from opinion words to features. Popescu and Etzioni [2005] extract pairs (opinion word, feature) based on 10 extraction rules that work on dependency relations involving subjects, predicates and objects. Riloff and Wiebe [2003] use lexico-syntactic patterns in a bootstrapping approach to resolve relations between opinion holders and verbs for subjectivity classification.

2.4 Google Earth-based tools and frameworks

Google Earth has been used to show weather related information in Smith and Lakshmanan [2006] by producing snapshot updates of the current weather every 120 seconds and utilizing the KML file format to read these updates into the Google Earth. GeoSphereSearch, a context-aware geographic Web search, integrated a Web-based Google Earth to visualize results of a query Graupmann and Schenkel [2006]. Wood et al. [2007] demonstrated how visually exploring mobile directory service log files with spatial, temporal and attribute components can be performed using Google Earth in combination with other open source technologies like MySQL, PHP and LandSerf. Wood et al. [2007] discusses several aspects, like visual encodings, color and overplotting and shows how Google Earth handles these features. Slingsby et al. [2007] presented the feasibility of exploring catastrophic events and potential loss of information. The tasks that must be performed are outlined. The methods, with which Google Earth can handle such tasks, include: mapping, filtering by attribute, space, time, spatial aggregation, and creation of new views. Bringing geo-processing capabilities and Google Earth together is addressed by Pezanowski et al. [2007] to support crisis management scenarios. For this task, "Google Earth Dashboard" was proposed. It combines several technologies: Adobe Flex, Web Map Service (WMS), Web Feature Service (WFS), Web Processing Service (WPS) services and Google Earth as the cornerstone. The potential in combining geo-browsers like Google Earth in analyzing geospatial health sciences is discussed in Stensgaard et al. [2009]. The applicability of Google Earth in Online Analytical Processing (OLAP) is demonstrated in Martino et al. [2009], Ferraz and Santos [2010].

A desktop-based exploratory spatial association mining tool integrated with Google Earth for visualization was proposed in Compieta et al. [2007]. The tool consisted of two visualization views: the first was based on Google Earth for viewing and exploring related phenomena; the second view was developed in-house using Java 3D technology to interact with the association rules produced by the mining algorithm.

Integrating WPS-based services into Google Earth is addressed by Foerster et al. [2009]. Their approach consists of two parts. In the first part, the WPS client, built on top of the uDig (desktop, Java-based GIS system) exports configured processes into a KML file. In the second part, the KML file is loaded into Google Earth, which triggers WPS processes using the *Network Link* element. The advantage of the WPS approach is that the exported processes can be consumed by the vast community of people using applications that handle KML formats such

as Google Earth or Google Maps. This advantage was addressed in recent publications Smith and Lakshmanan [2006], Slingsby et al. [2007]. However, the proposed approach has several disadvantages. Although the approach has only two parts, they are not autonomous and require manual implementation (configuration of the processes using WPS client, creation and export of KML and dissemination of the results). Another aspect is lack of interaction and control over the geo-process exported into KML. As soon as KML is loaded into the application with *Network Link*, the process will be triggered and the results returned.

In general, the client side, which usually involves data visualization and manipulation, still remains more heterogeneous than the server side, since the way the data is presented cannot be enforced by standard specifications and interfaces. Here the choice of an appropriate software is driven by the needs of an expert. As was already mentioned, dozens of GIS applications are available, capable of solving specific spatial processing tasks and of supporting geo-visualization. When these applications are incapable of dealing with the problem, custom solutions and architectures are proposed Wachowicz et al. [2002], Compieta et al. [2007], Lundblad et al. [2009]. However, the current trend in data visualization and exploration is to use mass-market applications such as Google Earth for visualization, combined with open source technologies for data processing and storage Wood et al. [2007], Pezanowski et al. [2007], Martino et al. [2009], Ferraz and Santos [2010].

3

Discovering movement patterns: A geovisual analytics approach

Contents

3.1 Data 61
3.2 Method 61
3.3 Analysis 63
3.3.1 Spatiotemporal aggregation 63
3.3.2 Interactive grouping of the places 64
3.4 Analysis of photographers' movement in space and time 69
3.4.1 Spatial analysis of movement trajectories 69
3.4.2 Spatio-temporal analysis of movement trajectories 71
3.4.3 Summary of findings 75

This chapter presents a study of the geovisual analytics approach to discovering preferences for urban landmarks and pertinent travel itineraries revealed through photographs posted on the photo sharing and social networking website Flickr.

3.1 Data

The data selected for analysis comprises of 577,053 geo-referenced photos of locations in the Puget Sound metropolitan area, Washington State, USA, including the cities of Seattle, Bellevue, Kirkland, Redmond and the neighboring communities, made by 9,324 photographers between January 1, 2005 and August 31, 2009.

3.2 Method

Spatial event and movement data can be transformed into spatially referenced time series (see Chapter 1.2 for types of spatio-temporal data) by means of spatial and temporal aggregation. There are also aggregation techniques that transform movement data into spatial flows. Thus, in our analysis, we first deal with spatial event data (each photo shot is an event). In the process

of analysis, we apply spatial and temporal aggregation to the events and thereby obtain time series of attribute values characterizing places. Next, from the original event data, we derive movement data. For this purpose, the photos made by the same photographer are linked into a chronologically ordered sequence, which is interpreted as the trajectory of the photographer. Then, we transform the movement data into spatial flows data.

To accomplish spatio-temporal aggregation, the geographical space is divided into compartments and the time span of the data is divided into intervals. The events fitting into each compartment and time interval are collected and their statistics, in particular, the number of events are computed. Consequently, each compartment is characterized by a time-series of event counts. Given the analysis task, the tessellation of the territory should be sufficiently fine and result in a potentially large number of compartments. However, the individual inspection of each compartment may be too expensive, given their number. A reasonable approach is grouping the compartments by the similarity of their characteristics and further processing of groups instead of individual compartments. Before any further processing of grouped compartments can happen, though, one has to answer the question of what are the characteristics differentiating potentially interesting locations from uninteresting ones. In the analysis of Flickr posted photos we used the following criteria to help discriminate between potentially interesting and uninteresting locations.

1. Locations never visited or visited only by a few photographers are uninteresting.

2. Locations which were consistently visited by a relatively high number of photographers are uninteresting (these are major tourist places and as such of no interest to us).

3. Locations that were consistently visited by a relatively moderate number of photographers are potentially interesting (not crowded with tourists but attracting stable attention).

4. Locations that periodically or occasionally attracted unusually many photographers are interesting. Such locations may be attractive for organizing special activities for tourists and/or locals.

Suitable characteristics for the classification of locations may be obtained by describing the time series of photo counts using simple descriptive statistics such as maximum and quartiles or percentiles and using the following selection conditions:

- A low maximum value means that the place was never sufficiently interesting.

- A high value of the 1st quartile means that the place was visited by relatively many photographers.

- A low value of the third quartile together with a high maximum value means that the place attracted high attention in less than 25% of the time intervals. This may be a periodically or occasionally visited place (but not necessarily). The use of percentiles such as 90, 95, or 99 can help to find places that attracted high attention in less than 10%, 5%, or 1% of time intervals, respectively. To figure out whether the increase of interest in these places was periodic or occasional, the analyst may need to look at the place's time series of photo postings ("details on demand").

The aggregation of movement data proceeds as follows. The trajectories are divided into segments corresponding to the division of time into intervals. Then, for each pair of spatial compartments s_1 and s_2 and time interval t all trajectory segments that start in s_1 and end in s_2 are collected and counted. The count gives the magnitude of the flow from s_1 to s_2 during the interval t.

3.3 Analysis

3.3.1 Spatiotemporal aggregation

We started the analysis by aggregating the locations of photos into spatial clusters with the radius of $500m$. The length of the radius is a rough approximation of the length/ width of a parcel accommodating a city landmark, such as a building, park, stadium, etc. [Adrienko and Adrienko, 2011]. We used the cluster centroids as the seeds for generating 2,930 Voronoi polygons covering the analysis area. Unlike a regular grid, the Voronoi polygon tessellation better matches the spatial distribution of photos, since the cluster centroids tend to be in the centers of spatial concentrations of photos that do not necessarily align with a regular shape of a grid cell. For each polygon, the total number of photos and the number of photographers were computed. The polygons are henceforth referred to in the article as "places".

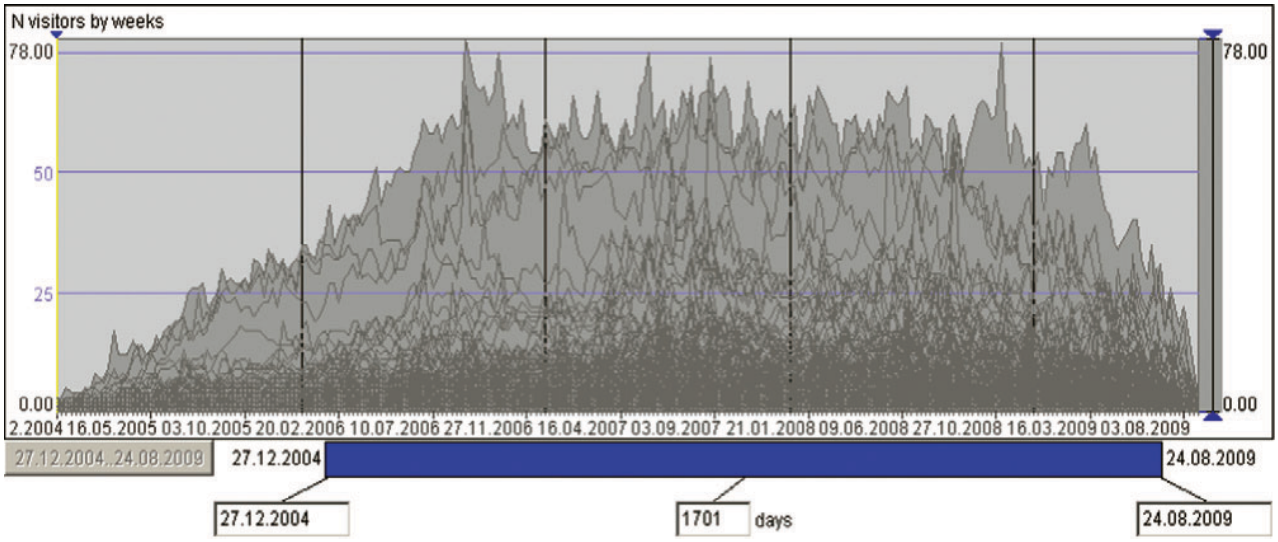


Figure 3.1: Each line in the graph represents the time-series of visits to a given place in the analysis area. There are 2,930 lines in total representing the corresponding number of Voronoi polygons

Next, we divided the 1/1/2005-31/8/2009 time period into 244 weekly intervals. For each place, the number of visits (number of photographers) in each week was computed. Hence, the

data were transformed into the spatially referenced time series of 244 observations. The time series can be visualized on a time graph in Figure 3.1.

The time graph cannot be conveniently used for data exploration because of the large number of intersecting and overlapping lines. However, it is suitable for examining the time series of particular places by selecting only the corresponding time-series lines.

3.3.2 Interactive grouping of the places

For each place, the following statistics were computed from the 244 weekly observations: 1st quartile (25-percentile), median (50-percentile), 95-percentile, 99-percentile, and maximum. We used standard attribute query and classification tools to identify interesting places, characterized by high count values in the top percentiles. We also used the following classification corresponding to the criteria presented in Section 3.2 to differentiate between potentially interesting (denoted by the asterisk symbol) and uninteresting places:

- Class 1: places with less than 10 visitors for the entire time-series (2,103 places) were classified as not interesting.
- Class 2: places with the maximum number of visitors in a week of less than five (548 places) were classified as not interesting.
- Class 3: places with the 1st quartile of weekly visitors greater than 10 (nine places) were regarded as uninteresting as these are the usual locations attracting many visitors otherwise known as the main tourist attractions.

Places with the maximum of weekly visitors ranging between 5 and 10 (213 places) were subdivided further into:

- Class 4*: places with relatively moderate values of the 1st quartile (25-percentile) and/or the median (50-percentile) indicating moderate but stable interest of photographers-these places (17) may be regarded as potentially interesting.
- Class 5: Places with relatively low values of the 1st quartile and/or the median signifying the lack of stable interest (196). Those places were deemed uninteresting.

For the remaining 68 places, we were interested in finding locations characterized by periodic or occasional peaks in the number of visitors. A high difference between the maximum number of visitors and the 95-percentile means that the place attracted high attention in less than 5% of the 244 time intervals (weeks), i.e. no more than in 12 weeks. A high difference between the maximum number of visitors and the 99-percentile means that the place attracted high attention in less than 1% of the time intervals (weeks), i.e. no more than in 2 weeks. To find places with such characteristics, we computed the differences between the maximum and the 95- and 99-percentiles. This resulted in two new attributes and their distribution values depicted in Figure 3.2. The scatterplot reveals two interesting subsets of places: (1) points corresponding to high difference values between the maximum and both 95- and 99-percentiles (shown as dark hollow circles); (2) points corresponding to higher difference values between the maximum and 95-percentile, and lower difference values between the maximum and 99-percentile (shown as dark filled circles). Thereby one can create three additional classes:

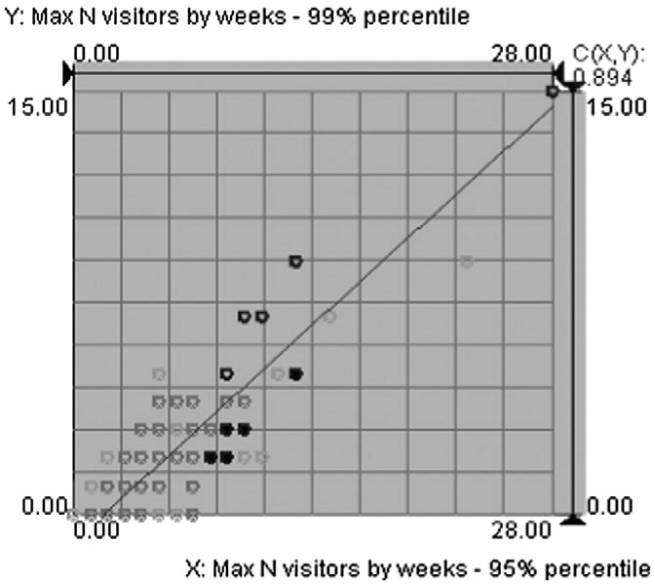


Figure 3.2: The distribution of differences between the maximum number of visitors and the 95- and 99-percentiles for 68 places in the Seattle metropolitan area

- Class 6*: places with one or two occasional peaks of interest (five places represented by hollow points).
- Class 7*: places with more than two peaks of interest (six places represented by filled circles; there are two overlapping points in Figure 3.2), which, possibly occur periodically.
- Class 8*: the residual of 57 places characterized by the lack of occasional peaks representative of isolated one-week long intervals and by longer (2-3 consecutive weeks) periods of relatively high interest. These places are characterized by relatively high maximum values and low differences between the maximum and 95-percentile indicating locations that received high interest in more than 5% of time intervals, i.e. in more than 12 weeks. Additionally, these places are characterized by high differences between the maximum and 90-percentile thus indicating the number of "high interest" time intervals to be less than 10% or 24 weeks.

Next, we explored each place belonging to one of two classes in detail by visualizing their locations (Figure 3.3). The map reveals a spatial cluster formed by six places belonging to classes 6* and 7* in the Seattle's Fremont district (Figure 3.2, location 1). Two of the places belong to class 6* representing high difference values between the maximum and both 95- and 99-percentiles (a few occasional peaks) while the other four belong to class 7* representing higher difference values between the maximum and 95-percentile, and lower difference values between the maximum and 99-percentile (still occasional but more frequent peaks than in class 6*). The other places, belonging to both classes and numbered from 2 to 6, are clustered around the Seattle downtown area.

The time series of places belonging to classes 6* and 7* and located in Seattle's Fremont district are depicted in a time graph (Figure 3.4). The two highest numbers of visitor photog-

raphers in the time series happened in the summers of 2007 and 2008, more precisely, in the weeks starting from 11.06.2007 and 16.06.2008 and coincided with the Summer Solstice Parade and Pageant in Freemont - the artsy district of Seattle.

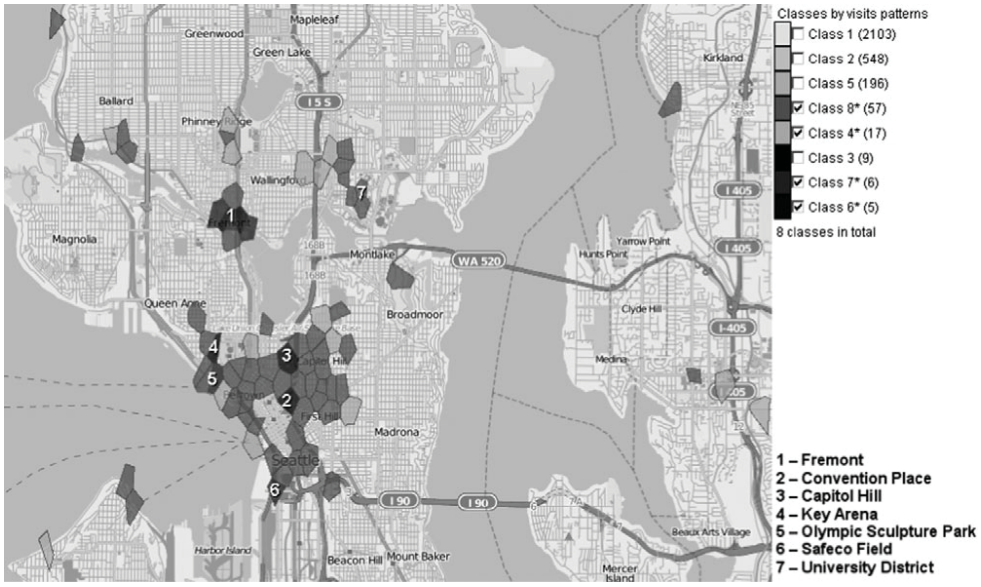


Figure 3.3: Grayscale shading indicates the places belonging to four selected classes (4*,6*,7*,8*) categorized earlier in the text as (potentially) interesting

Citing the http://en.wikipedia.org/wiki/Fremont,_Seattle,_Washington: "The Fremont Arts Council sponsors several highly attended annual events in Fremont. One of those events is the Summer Solstice Parade & Pageant, which has made Fremont famous for its nude Solstice Cyclists" (http://en.wikipedia.org/wiki/Solstice_Cyclists).

The interpretation of the cluster was corroborated by the inspection of corresponding photos. We retrieved the titles of the photos made in the places belonging to the cluster during the weeks with the highest number of visitors. Many titles indeed included "Fremont solstice parade". In 2007, most of the photos were made on 16.06.2007, which is the Saturday before the summer solstice. In 2008, the parade took place on Saturday 21.06.2008. Searching the Internet sources we found out that in 2009 the parade was held on Saturday 20.06.2009. The corresponding peak in the time-series graph (Figure 3.4), as well as the peak of 12.06.2006 and a smaller one of 13.06.2005 point out to a periodically high interest associated with this regular event.

Further examination of the time-series for the cluster of places in the Freemont area revealed also a high number of visits re-occurring periodically in September, more precisely, in the weeks starting on 18.09.2006, 17.09.2007, and 15.09.2008, respectively. The inspection of the titles of photos provided an explanation for the higher than usual number of visits: the Fremont beer festival, which regularly takes place in the middle of September.

The remaining five places belonging to classes 6* and 7* and labeled 2-6 in Figure 3.3 do not form spatial clusters but they do concentrate around Seattle's city center. The examination of their respective time series corroborated by the inspection of the photos and related Internet sources revealed the following locations and reasons that attracted the attention of photographers:

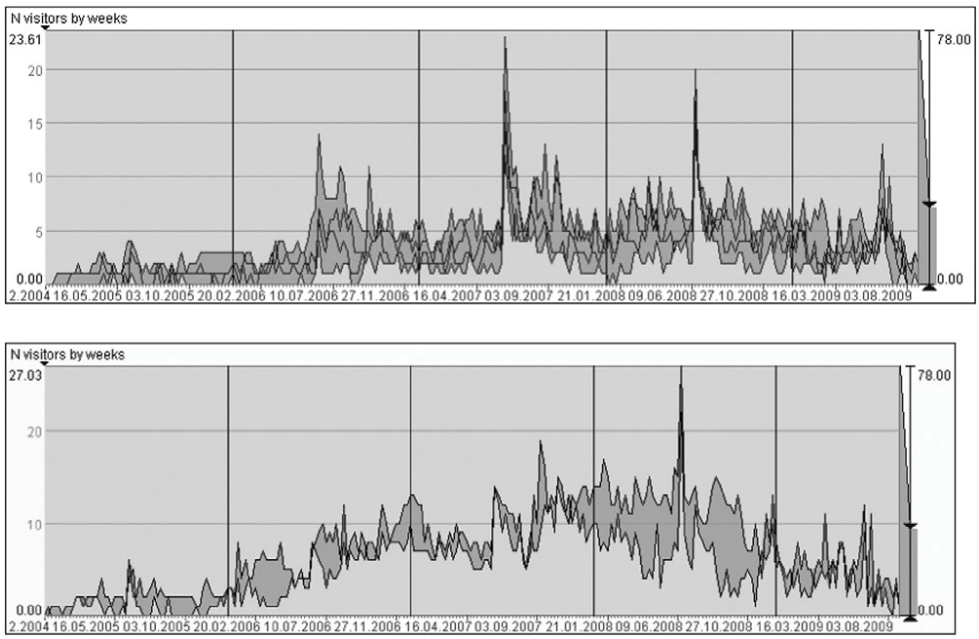


Figure 3.4: Timeseries of six places comprising cluster 1 on the map of study area (Figure 3.3) in the Freemont district of Seattle. The top part of the figure represents the time series for class 6*: high diff max - 95% AND high diff max - 99% class while the lower part represents the time series for class 7*: higher diff max - 95% AND lower diff max - 99%

- Place #2: Convention Place. Highest number of visitors: the weeks starting on 20.08.2007 and 25.08.2008. Explanation: game festivals PAX 2007 and PAX 2008 taking place at the Convention Place during 24.08-26.08.2007 and 29.08-31.08.2008.
- Place #3: Capitol Hill. Highest number of visitors: 15.12.2008. Explanation: snowfall and traffic accident.
- Place #4: Key Arena. Highest number of visitors: 04.02.2008. Explanation: Barack Obama's visit on 08.02.2008.
- Place #5: Olympic Sculpture Park. Highest number of visitors: the week of 15.01.2007-21.01.2007. Explanation: the Olympic Sculpture Park opened on January 20, 2007.
- Place #6: Safeco Field. Highest number of visitors: the time period from the week of 11-17.06.2006 until the week of 30.07-05.08.2007. Explanation: popular game season coinciding with better than average performance of the Seattle Mariners baseball team.

Further search for places of interest to photographers led us to examine the temporal patterns of photo taking activities in class 8* (Figure 3.3). The time series graph (Figure 3.1) can be transformed to show only line segments corresponding to substantial increases (or decreases) of

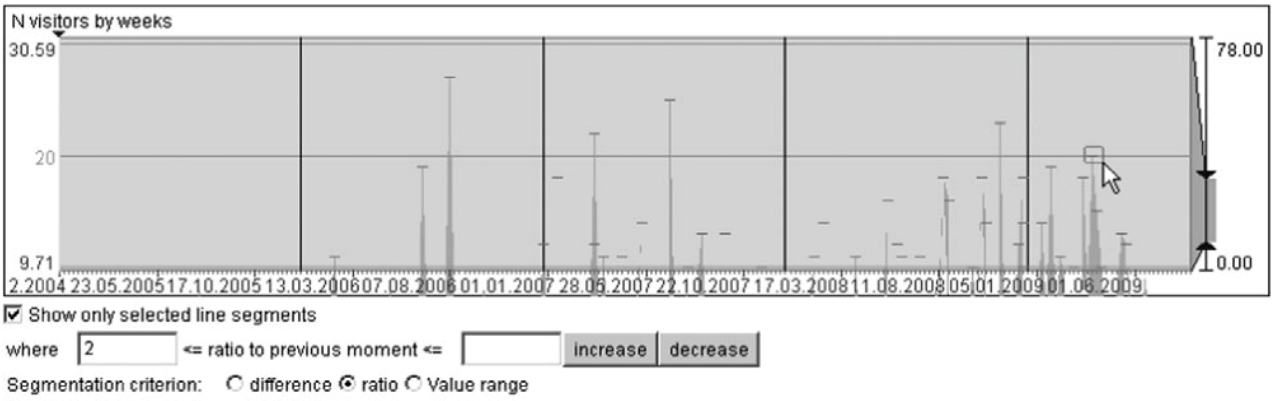


Figure 3.5: Time series graph shows only the line segments where the number of visitors was at least twice as high as in the previous interval

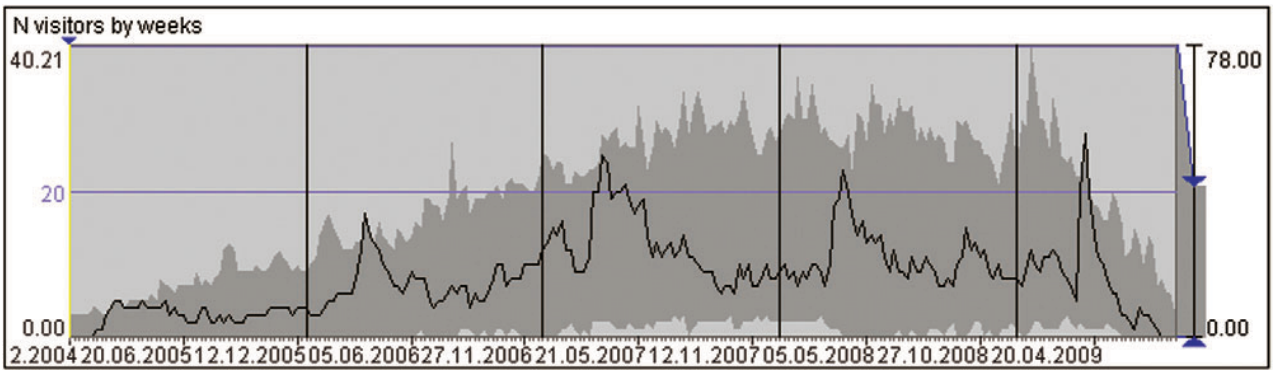


Figure 3.6: The time series of the line segment selected in the line segmentation mode as shown in Figure 3.5

the weekly number of visitors with respect to the previous week. Thus only the line segments representing the number of visitors at least twice as high as for the previous time interval (week) are visible in the time graph depicted in Figure 3.5. The switching between the standard mode and the line segmentation mode is accomplished by selecting and unselecting the checkbox "Show only selected line segments" at the bottom of the time graph.

In the segmentation mode (Figure 3.5), the analyst may select a particular line segment with the mouse (Figure 3.6) and then return the graph to the standard mode in order to see the whole time series line, to which the given segment belongs. The line is highlighted in black, as shown in Figure 3.6.

The examination of the time series characterized by repeatedly large increases in the weekly number of visitors led us to discover several unusual temporal patterns. A particularly interesting pattern, depicted in Figure 3.6, corresponds to location 7 on the map in Figure 3.3. The location is on the campus of University of Washington. The location attracted periodically a relatively high number of visitors during the weeks of 18-31.03.2006; 12-25.03.2007; 17-31.03.2008; 26.03-06.04.2009 (see the peaks in the time series in Figure 3.6). Since the increases are not occasional but consist of several high values during consecutive weeks (compare "wide" peaks in Figure 3.6

with "narrow" peaks in Figure 3.4), this pattern is representative of class 8*. The main campus of the University of Washington located in Seattle's University District is famous for its early spring cherry tree blossoms attracting the local as well as national and international visitors.

3.4 Analysis of photographers' movement in space and time

The objective of movement analysis was to find out whether there was a spatio-temporal pattern of visiting the locations of interest in the greater metropolitan area of Seattle. To facilitate the analysis the sequences of photos were extracted for each photographer who took more than one photo. A photo was regarded to be a part of a sequence if the time interval separating it from the previous photo was less than or equal to eight hours. Otherwise the photo was considered to be the beginning of a new photo session and hence a new sequence. This pre-processing of the data resulted in extracting 78,871 sequences created by 9,324 photographers. Thanks to their geographic coordinates photos belonging to a sequence can be plotted as point locations and aligned into a movement trajectory of photographers.

3.4.1 Spatial analysis of movement trajectories

Aggregation

Movement trajectories can either be enclosed within one place (polygon container) or originate in one place and end in another. For every pair of places (A, B) the total number of times a photographer moved from A to B was counted. The resulting records (place A, place B, count) are called "aggregate moves". There are also records of the type (place A, place A, count), which aggregate the trajectories fully contained in place A.

Visualization

Flow mapping [Slocum, 1999] is a standard cartographic technique to visualize aggregate moves. Since movement trajectories between any two places can proceed in both directions, our movement visualization tool employs "half-arrow" symbols to represent movements between the places in two opposite directions. This symbol was proposed by Tobler [1987] for discrete flow maps. The symbols are integrated with lines connecting the centroids of the polygon compartments. The widths of the symbols are proportional to the numbers of aggregate moves they represent. The aggregate moves with coinciding starts and ends are represented by circular symbols with the radius proportional to the number of moves.

Results

More than one third of the trajectories (29,072 or 37%) were contained within one compartment. Such trajectories were transformed into the aggregate moves of the A, A, count type, i.e. where the start coincided with the end. Most of such moves were located in the city center. The circular symbols dominated the map and made the linear symbols representing the aggregate moves of the A, B, count type hardly visible. Hence, in the subsequent analysis the aggregate moves with the coinciding starts and ends were filtered out.

The remaining 49,799 trajectories represent the aggregate moves of the A, B, count type, i.e. where the end differs from the start. The large number of trajectories makes a visual pattern assessment nearly impossible and necessitates data filtering. We interactively filtered the trajectories by the place of start and/or end, the move count, and the length of the move (the distance between the centroids of place A and place B).

Figure 3.7 depicts the trajectories filtered by the places of start and end, where the place location is restricted to the city center and the move count is at least 50. There are two distinct areas: Pacific Science Center (northwestern section of the maps) and Pike Place Market (south-eastern section) where the places are connected with strong bi-directional flows. Moves in the south-central section are more frequent than in other sections of the city center.

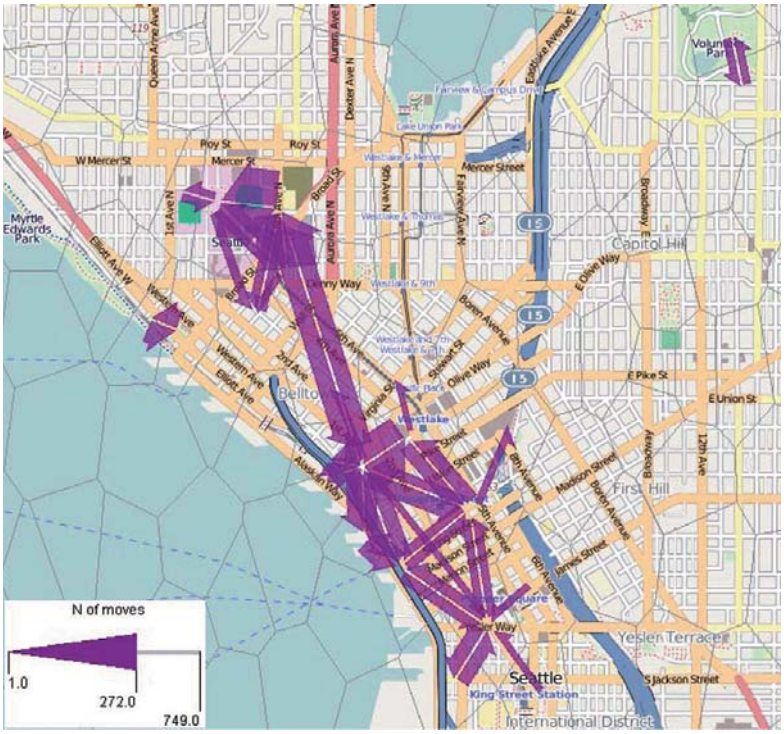
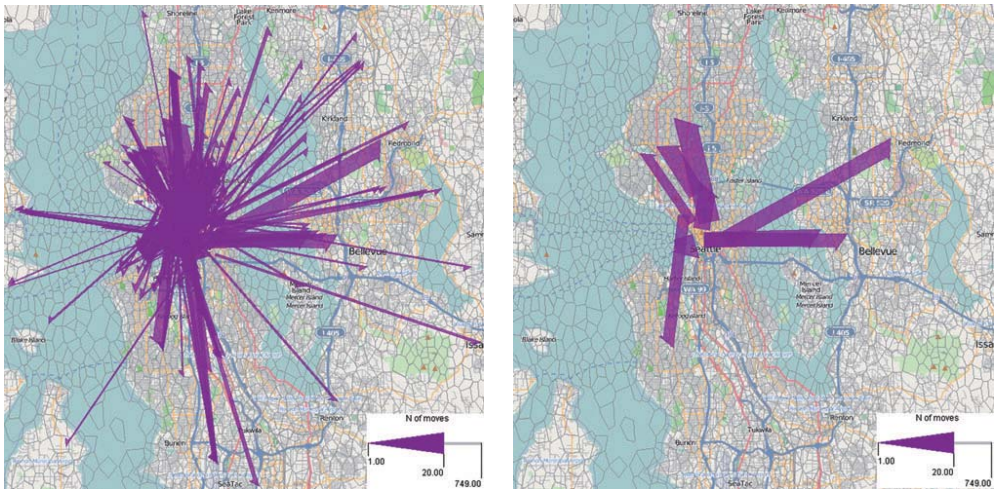


Figure 3.7: Trajectories in the center of Seattle filtered by the city center-only location of start and end, and by the minimum number of moves equal to 50. The irregular mesh of Voronoi polygons, representing places (compartments) is overlaid on the flow map

Figure 3.8 depicts the trajectories filtered by the places of start and end, where the start locations are in the city center and the end locations are outside the city center. Further filtering using the minimum number of moves along a trajectory equal to 10 reveals the movement pattern, in which the dominating trajectories connect the city center with the Green Lake and Magnolia areas (north and north-west of the city center), southwestern Seattle, and the areas east of Seattle including the cities of Bellevue and Redmond. Filtering the trajectories using the reversed locations of start and end (starting outside the city center and ending in the center) returns a similar pattern.

Figure 3.9 depicts the trajectories filtered by the locations of start and end being outside the city center and by the length of moves. The flow map on the left shows that the majority of



(a) (b)

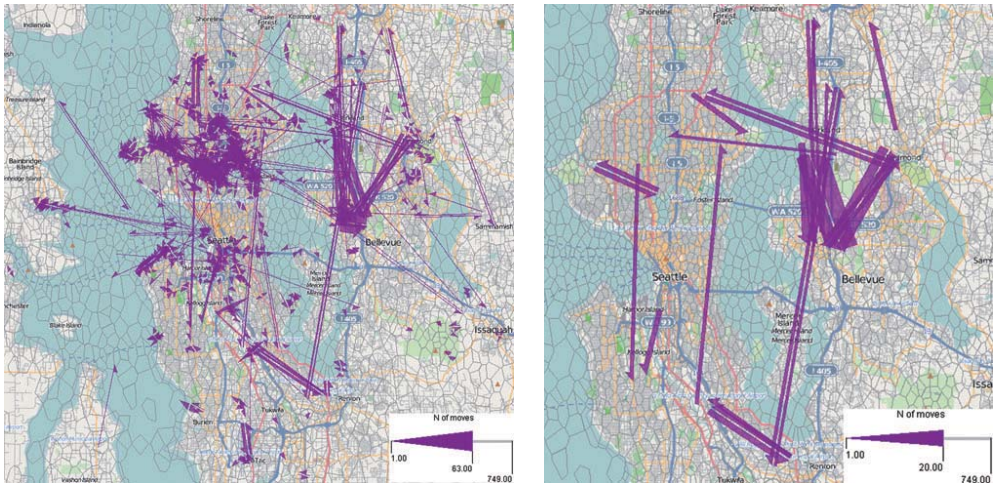
Figure 3.8: Trajectories originating in the center and ending outside the city center. The map on the left depict all trajectories. The map on the right depicts only the trajectories with the move count of 10 or more

trajectories, located outside the city center, are dominated by short moves (e.g. Fremont area and around). Long moves are much less frequent than the short moves; the lines depicting long moves are mostly thin representing a small number of photographers who moved over longer distances within a course of one photo session. The largest move count for the short trajectories is 63 while for the long trajectories it is 20 (Figure 3.9). The flow map on the right shows that long trajectories most notably connected Bellevue with Redmond and Bellevue with Kirkland - cities on the east side of Lake Washington - and that there were only a few, low-count trajectories connecting different parts of the greater Seattle metropolitan area. The pattern of short moves versus long moves is consistent with the density of development in Seattle where the areas outside the city center are characterized by a fairly dense development in contrast to other neighboring cities (Bellevue, Kirkland, Redmond) where a low density, suburban development pattern is dominant outside the urban core. Only a small number of photographers moved over longer distances within a short time span of 8 hours. Most tended to visit fairly localized areas, as demonstrated by the predominance of short itineraries in Figure 3.9.

3.4.2 Spatio-temporal analysis of movement trajectories

The analysis was facilitated by data aggregation, in which for every pair of places (A, B) and time interval $[t_i, t_{i+1}]$, where i initially equalled one month, the number of moves from A to B was counted. For the monthly time interval there is no obvious trajectory pattern. We considered separately long moves ($length \geq 3km$) and short moves ($0 < length \leq 1,500m$). For long moves there was an absence of prominent re-occurring patterns that could be detected except for the trajectories across the Puget Sound, most notably between Seattle and Bainbridge Island, occurring mostly during summer months when weather is the best (Figure 3.10).

For the short moves ($0 < length \leq 1,500m$), interesting patterns appear in some months.



(a) (b)

Figure 3.9: Trajectories originating and ending outside the city center. The flow map on the left shows the overall pattern comprised of short and long trajectories. The flow map on the right depicts the trajectories that are at the minimum 5 km long and have at least five moves

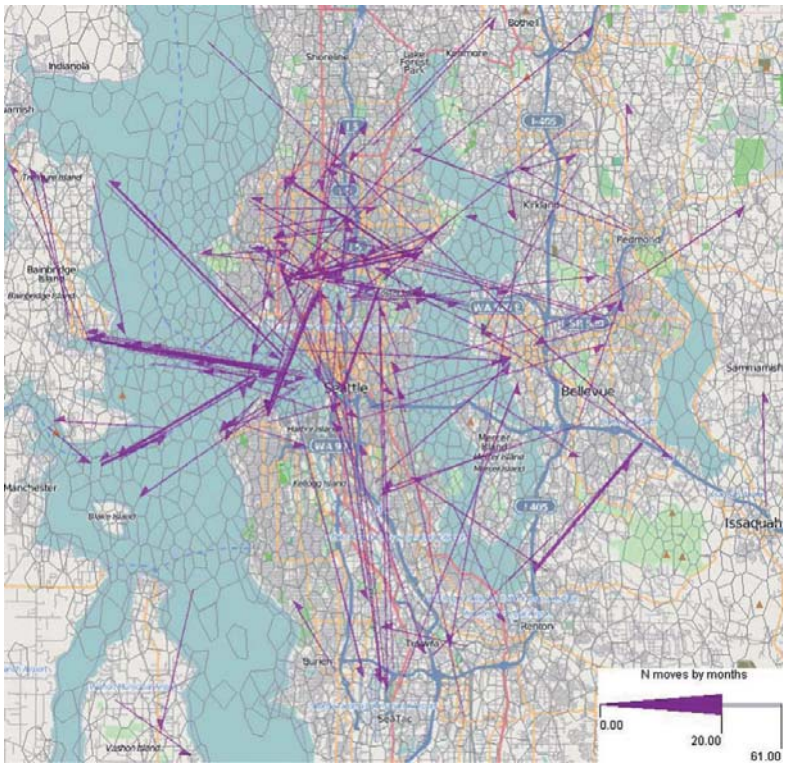
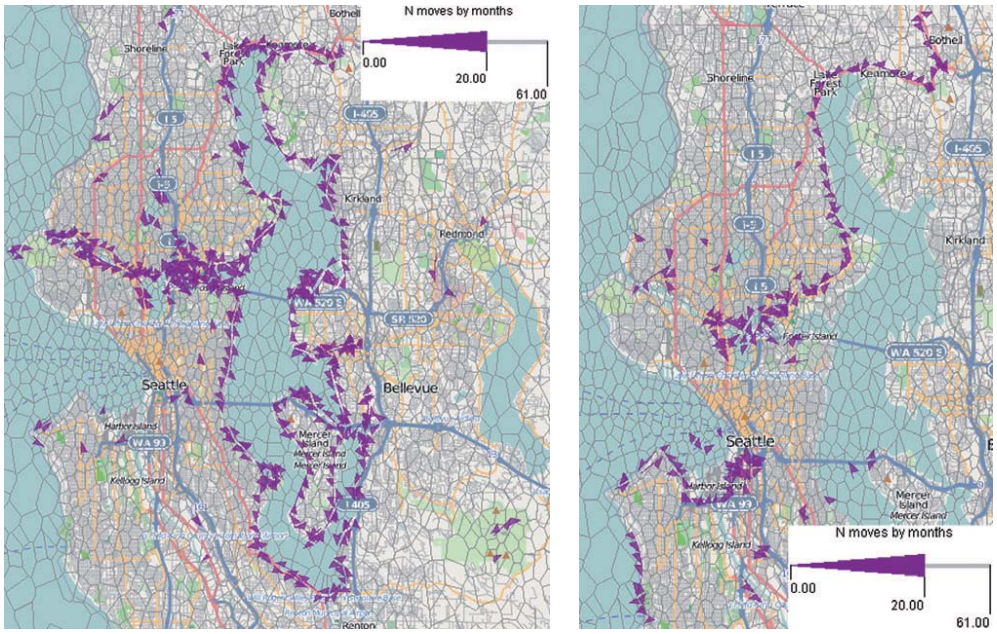


Figure 3.10: Long move trajectories (3 km or longer) for the month of June 2006. Similar pattern with moves from Seattle to Bainbridge Island (west) and back was observed for the summer months during the entire 2005-2009 period

Some of them, such as those depicted in Figure 3.11, represent long sequences of photographing urban and suburban landscape, during which photographers moved and frequently took photos along the way in almost every compartment of the sequence. Such sequences might be analyzed further for their potential of becoming designated as possible scenic walking routes.

Next, the data aggregation was repeated for a longer time interval i equal to three months and spatial clustering with a radius of 5,000m instead of the initial radius of 500m. The increased time aggregation interval reflects the seasonality of travel patterns in the study area, which roughly corresponds to quarters of the year. The increased radius of spatial clustering corresponds to our interest in capturing major flows in the study area without focusing necessarily on moves between specific landmarks. The results of analyzing movement patterns in space and time using the increased temporal and spatial clustering parameters are presented in Figure 3.12. Two patterns become apparent when summer trajectories are compared with winter trajectories across four years (2005-2008). First, the frequency of moves, represented by the width of the line, steadily increases across both seasons (summer and winter) from 2005 through 2008. This increase may simply reflect the fact of the growing popularity of Flickr resulting in the growing number of photographers between 2005 and 2008 posting their photos. Secondly, the seasonality (summer versus winter) is reflected only marginally in the movement trajectories connecting Seattle's downtown area with Bainbridge Island to the west of downtown and across Puget Sound. These trajectories are of low frequency representing a relatively low number of photographers moving across Puget Sound. The vast majority of trajectories concentrate, regardless of the season, around the city center and north of it. This pattern underscores the fact that Seattle has been a year-around destination for photographers.



(a) (b)

Figure 3.11: Short move trajectories ($0 < length \leq 1500m$). The flow map on the left represents the aggregate moves for 02.2007. The map on the right represents the aggregate moves for 08.2009

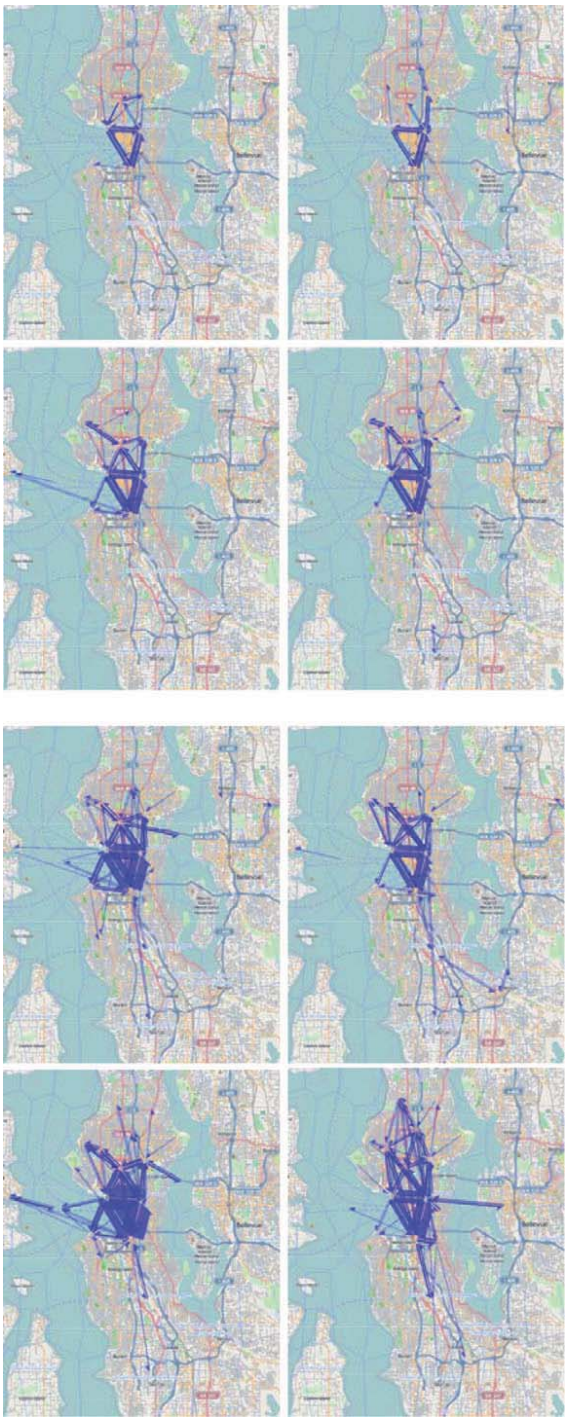


Figure 3.12: Pattern of movement trajectories in the study area aggregated by 3-month time interval and 5,000m spatial cluster radius. The minimum line thickness (1 pixel) corresponds to five moves and the maximum thickness (36 pixels) corresponds to 176 moves. Flows with less than five moves are not represented

3.4.3 Summary of findings

The majority of photos analyzed in this study were taken within small areas (approximated by circles with radii below 500m). This indicates that the photographers focused on specific locations or events rather than on photographing an extended metropolitan area. There were two larger areas in the city center accounting for most of the intra- and inter-area moves associated with photo taking activity. One area covers a large part of Seattle's downtown, from Pioneer Square to Pike Place Market. The other area extends around the Pacific Science Center north of Seattle's downtown.

Photographers' moves away from the city center to other locations were not as frequent as within the city center. Some of those locations to the north, south, and east (e.g. Redmond and Bellevue) of the city center are better connected than others by bi-directional flows of photographers. This means that during the period of 2005-2009 there were people interested in visiting in the same day the city center and some of outlying locations in the Puget Sound Metropolitan Area.

Both in the city center and outside of it, short moves between neighboring locations prevailed over distant moves. This may reflect a relative mobility of photographers who tour an area rather than staying in one location and often find interesting landmarks/ events for taking photos as they move from one place to another. Long moves correspond to cases, in which photographers travel from one place to another without taking photos on the way. Such moves were relatively infrequent in our study. Some photographers were interested in walking or driving longer distances along Puget Sound, Lake Washington, and Lake Union waterfronts and taking photos on the way. There was a lack of clear seasonal dependency in the frequency and spatial direction of photographers' moves, meaning that similar move patterns occurred in different seasons (summer and winter).

4

Discovering attractive places

Contents

4.1 Data definition and assumptions 77
4.2 Density estimation 79
4.2.1 Density-based clustering 79
4.2.2 DBSCAN algorithm 79
4.2.3 Influence weights 80
4.2.4 Database integration 82
4.2.5 Performance evaluation 83
4.3 Opinion analysis 92
4.4 Visualization and Exploration 93
4.5 P-DBSCAN 97
4.5.1 Problem formulation 97
4.5.2 Definitions 99
4.5.3 Method 100
4.5.4 Evaluation 102

In this chapter we present two approaches to discovering attractive areas by analyzing geo-tagged photos and photo comments using density-based clustering and text analysis techniques. We present a new DBSCAN-based algorithm termed P-DBSCAN that improves clustering of attractive areas using the notion of photo ownership and adaptive density.

4.1 Data definition and assumptions

Data components that are considered in this section include photo ids, photo owners, photo coordinates, comments, comment authors, and timestamps. The set of photos is defined as P, and every photo $p \in P$ is described as a tuple of the following elements: $p = (id, l, u, o, t)$, where id is a unique id of the photo, l the photo's coordinate pair expressed in degrees (latitude and longitude), u the photo's coordinate pair expressed in UTM coordinate system, o is the owner of the photo, t - timestamp associated with the photo (time when the photo was taken). Every photo can contain a set of comments, written by different people including the owner of the

photo. Every comment has a timestamp when it was written and all the comments can be sorted according to the timestamps from the oldest to the newest. The set of owners is defined as O, where every owner $o \in O$ can have multiple photos.

Every photo $p \in P$ can be assigned a numerical weight $w_p \in R$ representing its importance or interestingness. In the case of density estimation, we call these weights *influence weights* expressing the influence of nearby photos on the given photo, while in the analysis of opinions and sentiments, the weights are called *opinion* and *sentiment* scores respectively. We define an *interesting place* as a region containing photos with high (influence) weights. These regions do not have precise shapes and do not necessarily form clusters, but they are visually distinguishable from less interesting places.

The photo importance can be defined in various ways and depends on elements that are considered in the analysis. Ahern et al. [2007], for example, defined the photo importance as tag representativeness based on the rational that a photo tag represents the importance of a given photo and the place where it was taken. Here, we suggest several heuristics for establishing the photo importance in each of the two presented approaches.

Photo importance and place interestingness using the density estimation approach can be established in the following ways:

1. The interestingness of a place is determined by the number of photos taken at or near the place and by the number of people (owners) who have taken the photos.

2. The importance of a photo is determined by a relative distance between the photo and other photos around it; the shorter the distance between photos the more important the given photo is (it exerts more influence over the other photos).

3. The importance of a photo should not be biased by other (multiple) photos of the place taken by the same person.

Photo importance/interestingness using opinion and sentiment scores can be established in the following ways:

1. The photo importance is determined by the number of people commenting on the photo.

2. The photo importance is determined by the number of sentences containing opinions and/or sentiments.

3. The photo importance should not be biased by comments of the person who is the author of the photo.

4. The photo importance should not be biased by subsequent comments of the same person (replies to earlier comments).

5. When the opinion or sentiment score increases, the photo interestingness increases.

6. In case of sentiments, the interestingness of a photo can be represented by negative scores (negative sentiments).

4.2 Density estimation

The influence weights are calculated during the cluster expansion process based on our modification of the density-based clustering algorithm DBSCAN [Ester et al., 1996]. We describe basic features of density-based clustering followed by the definition of DBSCAN and its modification introduced in this research.

4.2.1 Density-based clustering

The density-based clustering has the following basic characteristics:

1. It usually requires only two input parameters: the minimum radius of the neighborhood ϵ and the minimum number of points inside a neighborhood $MinPts$. Regions with high density form density clusters. There is an intuitive notion that for any point in a cluster, the local point density around that point has to exceed some threshold.

2. There is no need to preset the number of clusters. The number of clusters is determined by using only two parameters: ϵ and $MinPts$.

3. The clusters can be of arbitrary shapes. Unlike grid-based clustering, where one needs to define rectangular grids, density clusters can be of any shape depending on the density of the regions.

4. Density-based clustering can handle noise. Points located in non-dense regions are not included in clusters and are considered as outliers.

4.2.2 DBSCAN algorithm

Here we define DBSCAN algorithm adapted to the problem of geotagged photo collection clustering. For more detailed explanation of the algorithm we refer the reader to the original paper by Ester et al. [1996].
Definition 1 (*Eps-neighborhood*): the Eps-neighborhood of a photo p is defined as a set of photos q whose distance from p is not more than Eps: $N_{Eps}(p) = \{q \in P | dist(p,q) \leq Eps\}$, where P is a set of all photos.
Definition 2 (*Core photo*): A photo p is a core when the *Eps*-neighborhood of a photo p contains at least a minimum number *MinPts* of other photos.
Definition 3 (*Directly density-reachable*): A photo p is directly density-reachable from the photo q if p is within the *Eps*-neighborhood of q, and q is a core photo.
Definition 4 (*Directly reachable*): A photo p is directly-reachable from the photo q with respect to *Eps* and *MinPts* if there is a chain of photos $p_1, \ldots p_n, p_1 = q$ and $p_n = p$ such that p_{i+1} is directly density-reachable from p_i.
Definition 5 (*Density-connected*): A photo p is density-connected to photo q with respect to *Eps* and *MinPts* if there is a photo $o \in P$ such that photos p and q are density-reachable from o with respect to *Eps* and *MinPts*.
Definition 6 (*Density-based cluster*): A cluster C is a non-empty subset of P satisfying the following requirements:

1. $\forall p, q$: if $q \in C$ and p is density-reachable from q with respect to *Eps* and *MinPts*, then $p \in C$.

2. $\forall p, q \in C$: p is density-connected to q with respect to *Eps* and *MinPts*.

Definition 7 (*Border photo*): A photo p is a border photo if it is not a core photo but density-reachable from another core photo.
Definition 8 (*Noise*): Let $C_1, \ldots C_n$ be the clusters of the photo dataset P. Then the noise is the set of photos in the dataset P not belonging to any cluster C_i, where $i = 1, \ldots n$. noise $= \{p \in P | \forall i : p \notin C_i\}$

The definition of the DBSCAN algorithm with respect to photo clustering can be explained in the following way. The DBSCAN requires two input parameters: the minimum radius of the neighborhood ϵ and the minimum number of photos inside a neighborhood $MinPts$. Starting with an arbitrary photo p, the algorithm checks how many photos are around the photo p within a radius ϵ. If there are fewer photos than $MinPts$, the photo p is marked as *noise*, otherwise all the photos in the neighborhood are added to the cluster the photo p belongs to. In such a case, the photo p is called a *core photo*. The photos that were found in the neighborhood are called *directly density-reachable*. Photo p marked as noise can still be assigned to some cluster if it is *directly density-reachable* from another photo q. In that case, photo p is called a *border photo*.

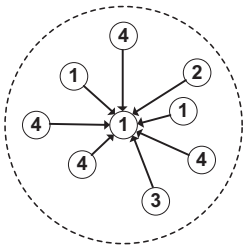
4.2.3 Influence weights

The definitions of photo importance described in Section 4.1 include the number of photos, the number of people who took the photos and the distance between the photos. The photo importance is modeled using a notion that we call *influence weight*, adopted from [Hinneburg and Keim, 1998], which is expressed as the sum of influence functions between a photo and all photos in the neighborhood. Thus, the influence function expresses the impact between any two photos p and p_n within a neighborhood and can be described as follows:

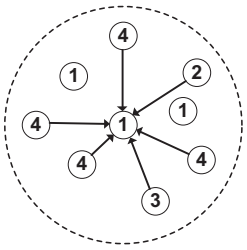
$$f(p) = K(p, p_n) = f(p_n)$$

where K is a symmetric kernel function based on Euclidean distance with bandwidth h as a smoothing parameter. Therefore, the requirements of DBSCAN-clustering and influence weight calculation using the sum of kernel functions present an opportunity for combining the two processes together such that during the cluster expansion process not only the neighboring photos are retrieved for a particular photo but also the influence weights are calculated for that photo. Moreover, the smoothing bandwidth is the same as the neighborhood radius ϵ used by DBSCAN.

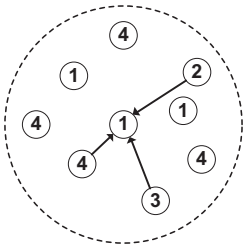
Figure 4.1(a) shows an example of the neighborhood of a photo labeled as 1 (the label indicates the owner of the photo) in the middle of the circle (neighborhood of radius ϵ). Let us assume that the number of photos in the neighborhood is larger than $MinPts$ threshold. Thus, all neighbor photos in Figure 4.1(a) are assigned to the cluster of the central photo labeled as 1. Consequently, the influence weight the central photo 1 receives equals the sum of individual influence functions between the central photo 1 and all the photos in the neighborhood. However, two types of biases exist when all the photos are considered in the influence weight calculation: the bias of multiple photos taken by the author of the weighted photo (photos labeled as 1 in



(a) Influence weight calculation using all photos in the neighborhood



(b) Influence weight calculation excluding photos of the same owner



(c) Influence weight calculation using only the nearest photo of each owner

Figure 4.1: Influence weight calculation

Figure 4.1(a)) and the bias of photos taken by other photographers. In order to eliminate the first bias, we exclude the influence of photos taken by the same owner (Figure 4.1(b)). In the case of the second bias, we use the following observation. Unlike the photos of owners 2 and 3, three photos of owner 4 contribute to the resulting influence weight. Owner 4 represents the case of a photographer uploading a number of photos of the same place, and consequently biasing the weight attached to the given place. In order to overcome this problem, we propose to select only those photos of every owner, whose distance to the central photo is the shortest. In such a case, only the nearest photo of owner 4 will be taken into account in the calculation of an influence weight (Figure 4.1(c)). Below is the formal definition of this approach.

$$S_p = \{p_i \in P : d(p, p_i) < r\}$$

where S_p is the set of photos in the neighborhood of p and r is the radius of the neighborhood and $d(p, p_i)$ is the distance between photos p and p_i

$$O(S_p) = \{o(p_i) : p_i \in S_p\} \setminus \{o(p)\}$$

where $O(S_p)$ is the set of owners having photos in space S and $o(p_i)$ is the owner of the photo p_i,

$$\sum_{o \in O(S_p)} K(p, f_{min}(p, o)) \tag{4.1}$$

where $f_{min}(p,o) = \arg\min_{p' \in p(o)} d(p',p)$ and $p(o)$ is set of photos of owner o.

As we have shown, the calculation of weights is based on the nearest neighbors of the photo of each of the photographer. However, the number of nearest neighbors is not bounded and theoretically can be very large if the neighborhood contains hundreds or thousands of photos taken by different photographers. The question then is whether it is practical to consider all the photographers for weight calculation and whether after selecting a limited subset of photographers, the user will notice any difference between the whole set and its subset, when the weights are mapped to colors. In any case, it is practical to introduce an upper limit of the number of photographers that can be considered in the weight calculation. Instead of taking all the photographers that satisfy the above formulation (Equation 4.1), we limit the number photographers to n. The formal definition is presented below.

$$L = \bigcup_{o \in O \setminus \{o(p)\}} f_{min}(p,o)$$

assign an order

$$O_p : L \to \mathbb{N}$$

with

$$O_p(p_i) \leq O_p(p_j) \Leftrightarrow d(p_i,p) \leq d(p_j,p)$$
$$N_p(n) = \{p_i : O_p(p_i) \leq n\}$$

4.2.4 Database integration

Usually, spatial indexing techniques like R-Trees [Guttman, 1984] or R^*-Tree [Beckmann et al., 1990] are used to index spatial information. In any such case, the points in the neighborhood can be retrieved in $O(log\ n)$ and the total complexity is $O(n{*}log\ n)$. However, the points are returned in no specific order since the order is not required for the density-based clustering algorithm. In our definition of *influence weights*, the requirement is that photos are sorted according to their distances. One of the straightforward solutions is to use one of the many implementations of indexers available on the Net for querying the photo neighborhood, and to sort the photos according to the distances in the custom code. However, another solution, which simplifies the two-step process, is to use a database with spatial extensions to perform such queries. Most of the free and commercial databases like PostgreSQL, Microsoft SQL Server 2008 and Oracle support spatial queries using Open Geospatial Consortium (OGC) [Ryden, 2005] specifications and implement fast indexing techniques. In case of the database integration, the core of the computation is performed by the database engine using two spatial queries (Listings 4.1 and 4.2).

Listing 4.1: The query counts the number of photos in the neighborhood

```
SELECT COUNT(photo_id) FROM TABLE
WHERE ST_DWITHIN(p, photos, radius)
```

where ST_DWITHIN returns *true* if photos *photos* are within neighborhood radius *radius* from a photo p.

The second query returns all the photos found in the neighborhood with the distances between photos and photo p calculated by the database and sorted in ascending order. In order to calculate

influence weights, there is a need to iterate over the returned photos and skip those whose owners were already taken into consideration.

Listing 4.2: The query returns all the photos found in the neighborhood with the distances between photos and photo p calculated by the database and sorted in ascending order

```
SELECT photo, ST_DISTANCE(p, photo)
FROM TABLE
WHERE ST_DWITHIN(p, photos, radius)
AND
p.photo_id <> photo.photo_id
ORDER BY distance ASC
```

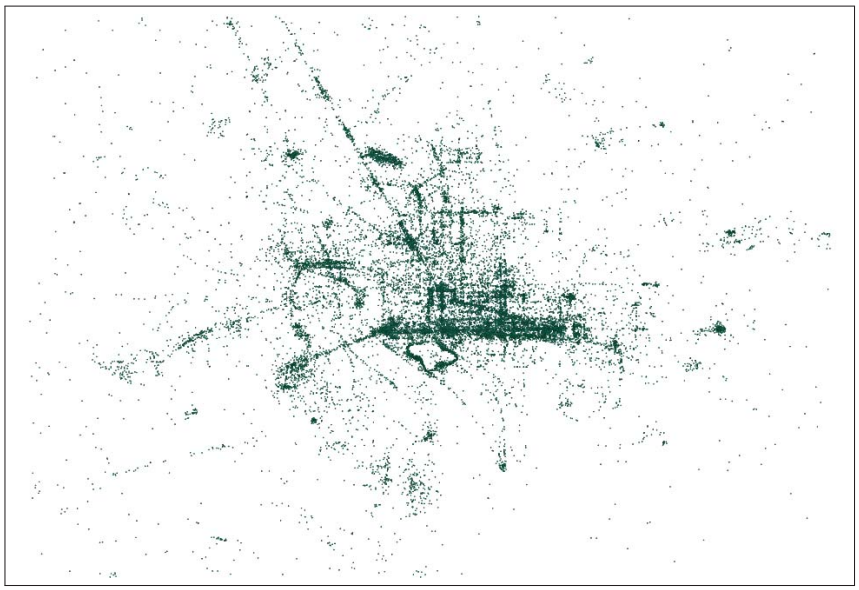
The clusters produced by the algorithm guarantee that each of them contains at least *MinPts* photos. However, there is a possibility to select a subset of clusters that comply to a specific number of photos or owners using a series of SQL queries (Listing 4.3).

Listing 4.3: The query returns all clusters that contain more than 100 photographers

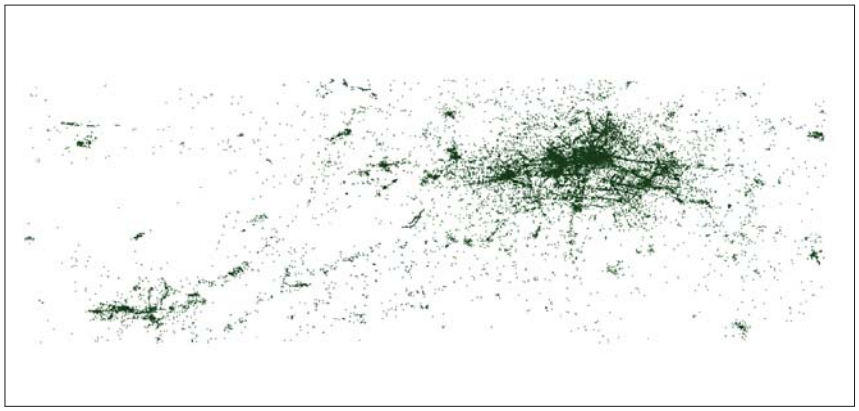
```
SELECT cluster_id FROM TABLE GROUP BY
cluster_id HAVING COUNT(distinct(owner_id)) > 100
```

4.2.5 Performance evaluation

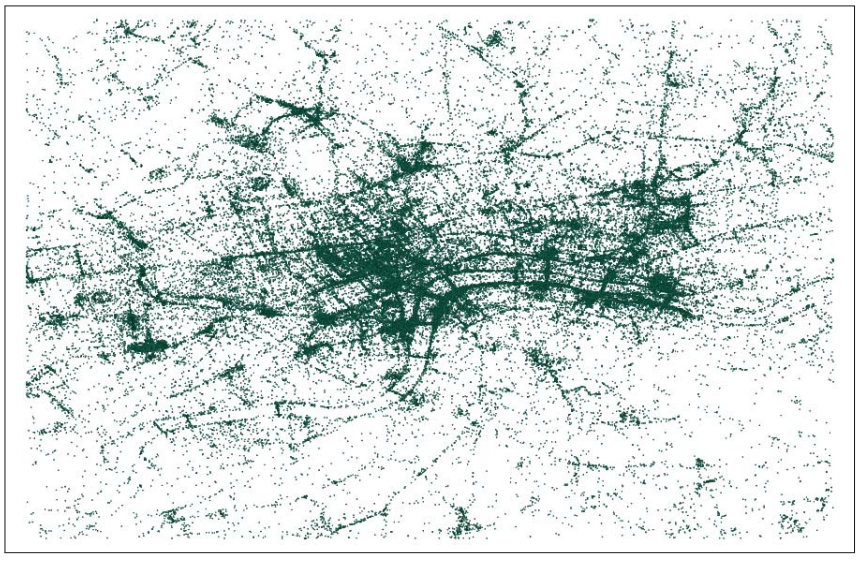
We evaluated the runtime complexity of DBSCAN algorithm with influence weights for three highly photographed cities: Washington D.C., Berlin and London. In cases where overlapping photos had identical coordinates, we removed all but one photo in order to reduce the runtime complexity. The areal extent, number of photos and photographers selected for the evaluation were as follows: Washington D.C. - area: $306km^2$, photos: 28,698, photographers: 4,160. Berlin - area: $1{,}834km^2$, photos: 43,718, photographers: 4,089. London - area: $126km^2$, photos: 107,982, photographers: 11,356. Figure 4.2 illustrates the distribution of photos in three cities with each photo represented by a single pixel.



(a) Washington D.C. area. 28698 photos.



(b) Berlin area. 43718 photos.



(c) London area. 107982 photos.

Figure 4.2: Three regions selected for evaluation

Table 4.1: Evaluation. *Seq* - sequential clustering, *Seq* 2 - sequential clustering with the area divided into two parts, *Seq* 4 - sequential clustering with the area divided into four parts, *Multi* 2 - parallel clustering of the area divided into two, *Multi* 4 - parallel clustering of the area divided into four parts. *Eps* - radius of the neighborhood. *MinPts* - minimum number of photos in the neighborhood. *Owner* = 0 - influence function is not bounded to a finite number of owners. *Avg* - average number of photographers in the neighborhood. The execution time is reported in seconds

Washington D.C					
Eps=30	**Seq**	**Seq 2**	**Seq 4**	**Multi 2**	**Multi 4**
owner=0 (Avg=26)	114	139	130	115	66
owner=10	105	132	133	105	68
Eps=50					
owner=0 (Avg=33)	178	146	120	97	74
owner=10	161	143	117	97	66
Eps=150					
owner=0 (Avg=60)	207	185	187	130	116
owner=10	182	184	175	114	106

Berlin					
Eps=30	**Seq**	**Seq 2**	**Seq 4**	**Multi 2**	**Multi 4**
owner=0 (Avg=22)	180	177	178	177	161
owner=10	180	177	170	171	161
Eps=50					
owner=0 (Avg=33)	187	189	171	174	173
owner=10	179	189	167	165	154
Eps=150					
owner=0 (Avg=57)	306	310	289	300	282
owner=10	294	297	280	280	266

London					
Eps=30	**Seq**	**Seq 2**	**Seq 4**	**Multi 2**	**Multi 4**
owner=0 (Avg=33)	427	389	384	266	242
owner=10	425	390	390	246	236
Eps=50					
owner=0 (Avg=54)	548	531	503	335	320
owner=10	532	511	483	327	297
Eps=150					
owner=0 (Avg=118)	1816	1554	1478	1060	900
owner=10	1771	1497	1416	939	877

The evaluation was performed on a 32-bit Windows platform with Intel Core 2 CPU (T7300, 2.00 GHz, 3GB memory) using PostgreSQL 8.3 database with PostGIS extension. The algorithm was implemented in Java 6. We compared the runtime performance (expressed in seconds) using three different settings for the radius of the neighborhood ϵ: 30, 50 and 150 meters, fixing the minimum number of photos $MinPts$ to 50 (Table 4.1). Additionally, we tested the algorithm using five execution modes: sequential clustering (labeled as Seq), sequential clustering when the area is vertically divided into two parts (labeled as Seq 2), sequential clustering when the area is divided into four parts (labeled as Seq 4), parallel clustering of two vertically divided areas (labeled as $Multi$ 2), and parallel clustering of the area divided into four parts (labeled as $Multi$ 4). The reason for dividing the area into parts is as follows: the clustering algorithm is used as a means of generating the influence weights. The weights are calculated for photos in the neighborhood controlled by the parameter ϵ, which is usually far smaller than the area under investigation. The photos that are situated far away from the current neighborhood do not contribute to the overall influence weight. This provides a rationale for splitting the analyzed area into several parts applying the algorithm sequentially or in parallel for every part. We evaluated the algorithm twice for every configuration setting and mode by first setting the number of photographers who contributed to the overall influence function to be unbounded (denoted as $owner = 0$, the average number of photographers in the neighborhood is reported next to it in parentheses), and then limiting the number of contributing photographers to ten. The computation time is reported in seconds. We used Gaussian kernel to calculate the influence weights.

The performance of the algorithm depends on many factors, such as clustering parameters, size of the area, density of photos, and number of photographers. For example, the difference between the computation time for London at $\epsilon = 30$ and $\epsilon = 150$ reaches 23 minutes. This can be explained by the fact that the area of London contains the majority of photos distributed across the whole region with a sufficiently high number of owners. The neighborhood radius of 30 meters is a very strong constraint to finding at least 50 photos inside the neighborhood (only some highly visited points of interest can meet this condition) and thus, many photos will be treated as noise, which decreases considerably the computation time. However, the neighborhood radius of 150 meters is enough to cover most of the areas of London, which means that spatial queries will find a large number of owners and photos (more than 50) warranting computation of the influence weights.

While limiting the number of owners and hence, photos, improves almost always the run time of computing the influence weights, the differences are not significant. This can be attributed to the fact that the time required by the database to find all the photos in the neighborhood and sorting them according to the distance is longer than computing weights between a limited number of photos.

Division of the region into parts and running the algorithm sequentially also improves the running time. The difference is perceivable in case of the division into four parts. The reason why there is almost no difference in the case of Berlin is that the majority of photos are concentrated in one part of the city. When the area is divided into parts, only one additional cluster is created when moving from the two part division (Seq 2) to the four part division (Seq 4) at the neighborhood radius set to 30m. In cases, in which the photos are distributed evenly (London area), sequential clustering of divided regions improves the running time. The biggest difference in running time can be observed in cases of parallel clustering of divided regions. In many cases

the running time is about twice as fast when using parallel clustering on four regions then when running sequential processing clustering for the whole area.

We evaluated the runtime performance of the clustering algorithm using several extreme cases with thousands of photos and owners. As we have demonstrated, the runtime performance depends on many factors, some of which are hard to predict in advance. It is evident that the most crucial factors are the combination of clustering parameters including the spatial distribution of photos, number of photos and owners, and the size of the area. The size of the area can be controlled by the user in a user-oriented environment (e.g. web-based application). However, the number of photos and owners, while depending on the area, cannot be predicted. In this case, in order to reduce the number of photos that will be used for calculating weights, further preprocessing can be employed as follows: (1) Removing not only those photos that were taken at the same place (having the same coordinates), but also the photos that were taken at some user-specified distance away, (2) Using sampling techniques prior to clustering where the area is divided into cells and photos that belong to a certain percentage of owners are retrieved from every grid cell.

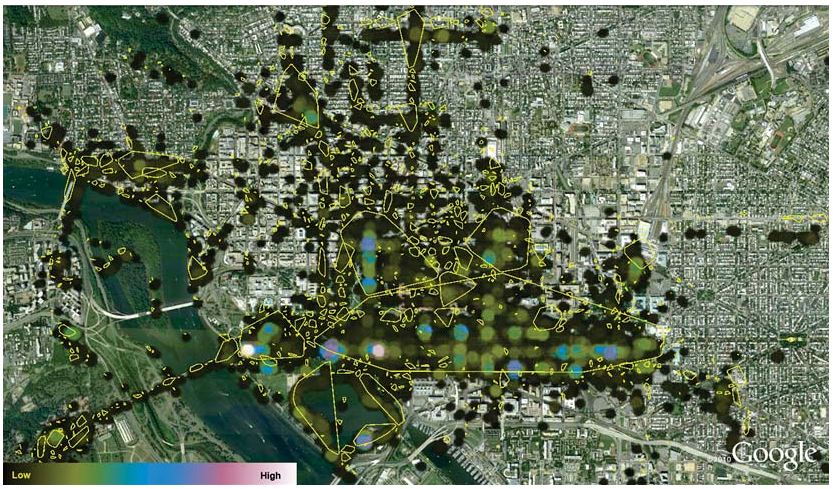
Another important aspect of the algorithm performance is the selection of clustering parameters. Selection of the parameters is a general issue in the density-based clustering algorithms and depends on the task at hand. Several recommendations can be found in the literature [Ester et al., 1996, Hinneburg and Keim, 1998, Gan and Li, 2003] but these suggestions are hardly applicable to the problem of clustering photo points given their ownership. As a consequence, the generated clusters may cover large areas. However, in the case of the visual exploration of tourist attractions and landmarks attracting many visitors the parameter selection becomes less critical due to the incorporation of *influence weights* in the cluster generation process. As it was presented in Section 4.2.3, the *influence weight* of a photo is a cumulative sum of influences between the given photo and its neighboring photos. Therefore, the size of a cluster has no influence on the determination of highly attractive places, which is carried out by assigning photos that belong to such places into one cluster. Hence, the role of parameter choice in the presented approach is reduced to determining a trade-off between runtime performance and area coverage, while the number of attractive areas determined by visual exploration remains almost unchanged. The runtime performance is influenced by the selection of the neighborhood radius (Table 4.1). With the number of photos in the neighborhood fixed, the runtime performance increases when the neighborhood radius is small because it becomes more difficult to find neighborhoods with the sufficient number of photos inside a smaller neighborhood. Thus more photos are classified as noise and are not further processed. Likewise, the area coverage increases (less photos are treated as noise) when the neighborhood is increased provided one keeps the number of photos in the neighborhood fixed or sets to a lower value.

We show how the cluster size and area coverage changes in the three highly visited cities including Washington D.C. (Figures 4.3(a)-4.3(c)), Berlin (Figures 4.3(d)-4.3(f)), and London (Figures 4.3(g)-4.3(i)) by setting different clustering parameters (Figure 4.3). Specifically, Figures 4.3(a), 4.3(d), and 4.3(g) show attractive areas and cluster boundaries generated by setting the neighborhood radius to 20 meters without providing a minimum number of photos in the neigborhood ($MinPts$=1). Cluster boundaries generated by Convex Hull algorithm are depicted in yellow. The color hues on the rightmost gradient scale correspond to the places with high influence weights (highly attractive areas), while the hues on the leftmost gradient scale correspond to the places with low influence weights. Since no minimum number of photos was provided, all

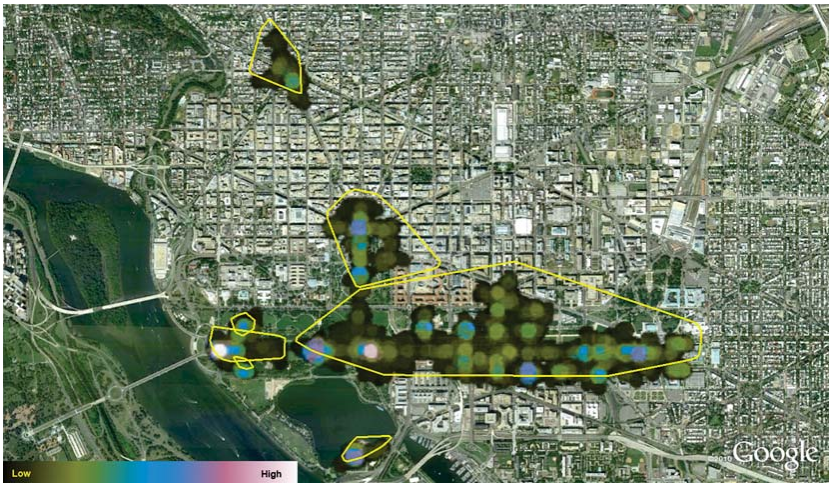
the photos were assigned to a cluster that contained at least one photo. It can be clearly seen that a number of clusters were generated with only a few photos inside, which had no importance in terms of our definition of *place interestingness* discussed in Section 4.1. However, some interesting places are discernible: *The Lincoln Memorial, The World War II Memorial, The Thomas Jefferson Memorial, The White House, Capitol Hill, etc.* (Washington), *The Reichstag Building, The Brandenburg Gate, Memorial to the Murdered Jews of Europe, Potsdamer Platz, etc.* (Berlin), *Piccadilly Circus, Trafalgar Square, London Eye, The British Museum, etc.* (London). Figures 4.3(b), 4.3(e), and 4.3(h) show the attractive areas of Washington D.C., Berlin, and London with the neighborhood radius of 20 meters and minimum number of photos set to 100 photos. The increase in the number of photos in the neighborhood from 1 to 100 results in a smaller number of clusters as was expected but the number of attractive areas does not change. The same interesting places that were found when the minimum number of photos was 1 can be still found. Figures 4.3(c), 4.3(f), and 4.3(i) show the difference in the area coverage due to increase of the neighborhood radius keeping the minimum number of photos fixed. Due to increase in the neighborhood radius, the clusters became larger (less photos were classified as noise). However, as in the previous example with the neighorhood radius of 20 meters, the same attractive places can be clearly seen.

The shown examples confirm that the selection of parameters in the case of visual exploration of geotagged photos has almost no impact on the results. The number of parameters can be even reduced to one (radius size) by omitting the minimum number of photos in the neighborhood which is not desirable in the case of DBSCAN in general. The selection of clusters with potentially attractive areas can be performed in the post-processing step by selecting those clusters that contain a minimum number of photos (for example clusters that contain one photo are of no interest) and can be performed as discussed in Section 4.2.4. There is still an advantage in using DBSCAN-like approach to clustering since it handles non-important areas (photos classified as noise in sparse areas) during the clustering process provided the minimum number of photos is higher than 1.

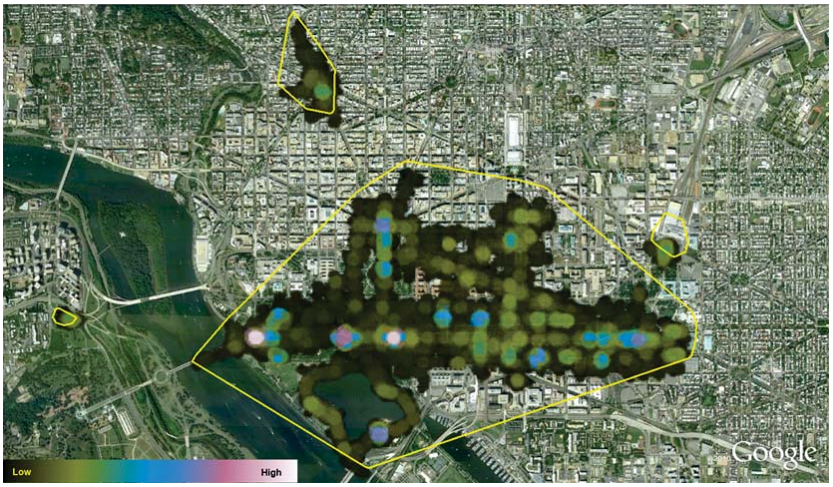
The division of areas into parts and the subsequent clustering can create a border (boundary) problem in a general-purpose clustering since a cluster can be potentially split. However, as we showed, the clusters and their spatial extents are less important than the weights generated during the clustering process. Moreover, the clusters are much larger than the neighborhood size used for the calculation of *influence weights*. Therefore, there is no impact of split areas on weight calculation and even if there is one, it is unlikely that it can be distinguished by the human eye when weights are transformed into color hues.



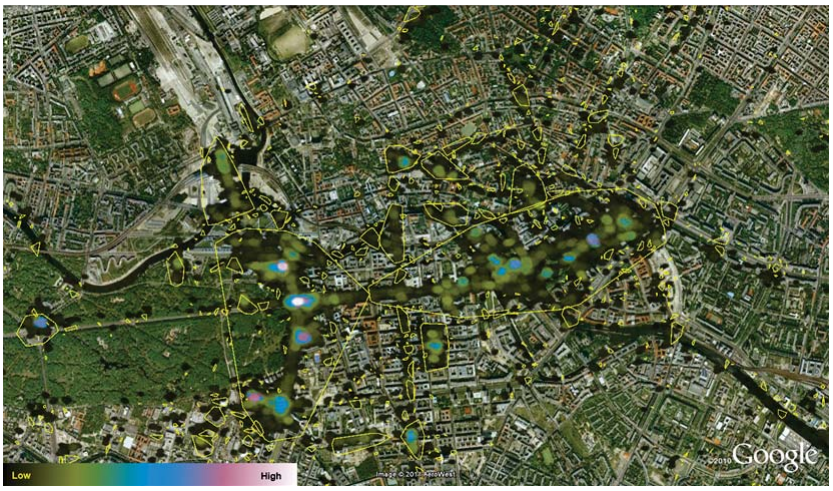
(a) Washington D.C. Neighborhood radius $\epsilon = 20$ meters, minimum number of photos $MinPts = 1$



(b) Washington D.C. Neighborhood radius $\epsilon = 20$ meters, minimum number of photos $MinPts = 100$

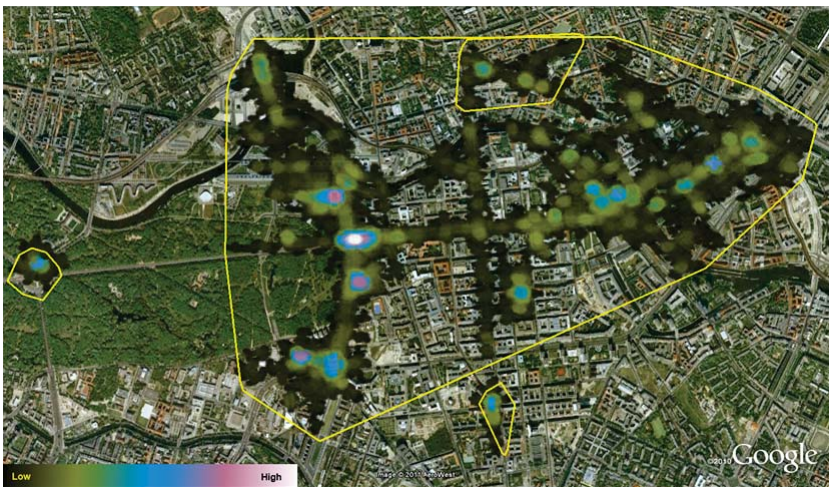


(c) Washington D.C. Neighborhood radius $\epsilon = 30$ meters, minimum number of photos $MinPts = 100$

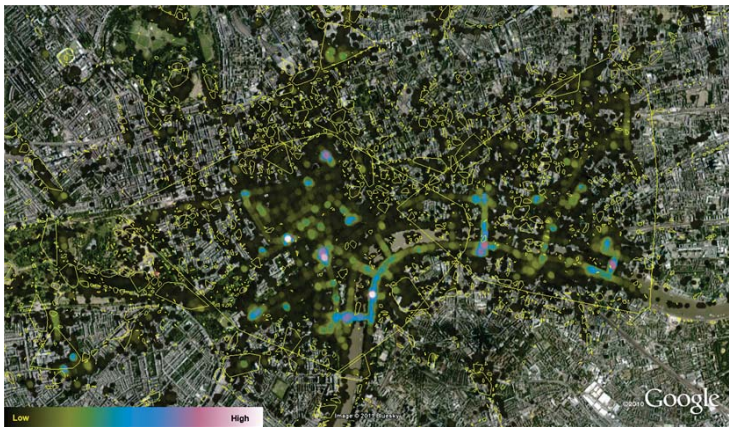


(d) Berlin. Neighborhood radius $\epsilon = 20$ meters, minimum number of photos $MinPts = 1$

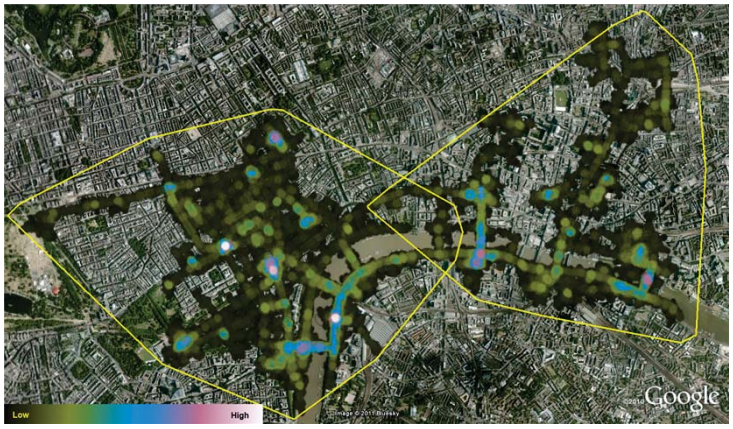
(e) Berlin. Neighborhood radius $\epsilon = 20$ meters, minimum number of photos $MinPts = 100$



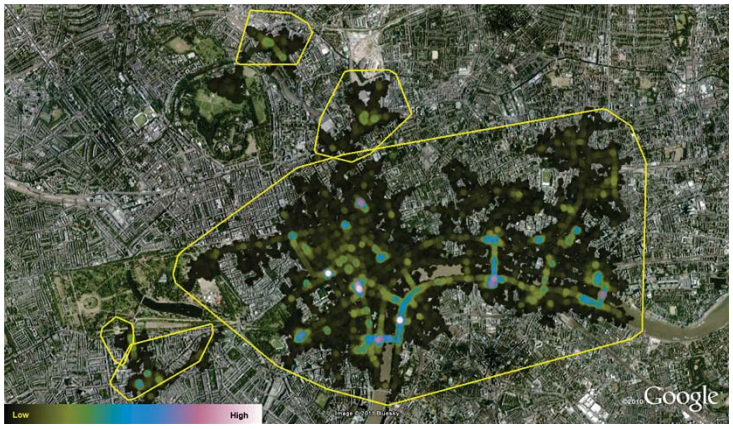
(f) Berlin. Neighborhood radius $\epsilon = 30$ meters, minimum number of photos $MinPts = 100$



(g) London. Neighborhood radius $\epsilon = 20$ meters, minimum number of photos $MinPts = 1$



(h) London. Neighborhood radius $\epsilon = 20$ meters, minimum number of photos $MinPts = 100$



(i) London. Neighborhood radius $\epsilon = 30$ meters, minimum number of photos $MinPts = 100$

Figure 4.3: Applying extended DBSCAN with *influence weights* to Washington D.C., Berlin, and London using different neighborhood radius and minimum number of photos generates clusters of different sizes and quantities but does not influence the determination of highly photographed areas. The color hues on the rightmost gradient scale correspond to the places with high influence weights (highly attractive areas).

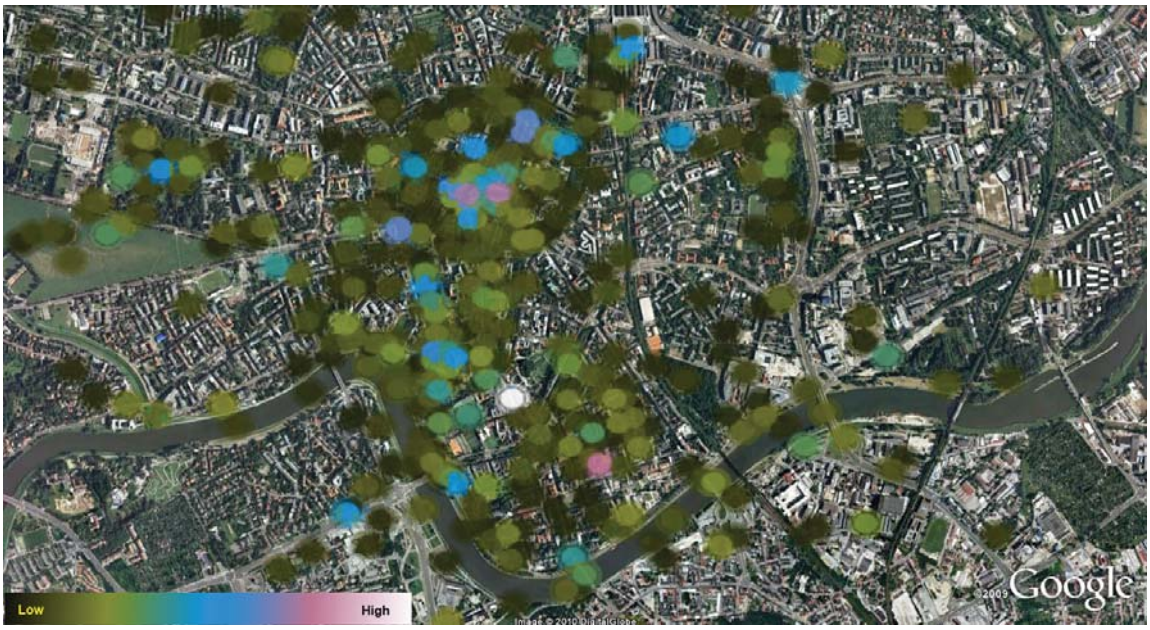
4.3 Opinion analysis

In Section 4.2 we presented an approach for finding locations of interest (e.g. tourist attractions) using density estimation. In this section we present an alternative approach, which rates an individual photo using any of the two features extracted from user comments: opinions and/or sentiments. We define opinions as negative or positive user statements related to the quality aspects of the photos (e.g., "vivid colors", "ambient light"), while sentiments are negative or positive user statements expressing feelings about a photo and about the objects depicted on the photo (e.g., "beautiful tree", "sad place"). There are two issues involved in this task. First, we need to differentiate between two types of user statements, which usually can be included in a single sentence. Having separated the two types of user statements, we need to determine the strength of user statements rather than merely report whether they are negative or positive. The solution to the first issue is to prepare a list of nouns that are commonly used by people to describe various photo features (e.g., "color", "composition", "photo quality"). The solution to the second problem lies in analyzing the choice of words used by people to describe features. In the English language these words are adjectives (e.g., "good", "bad"). By analyzing the usage of adjectives in the photo commentaries, we can classify words that are used more frequently than others. This allows us to rank adjectives by frequency similar to the ranking of important words in text mining communities [Salton and Buckley, 1988b].

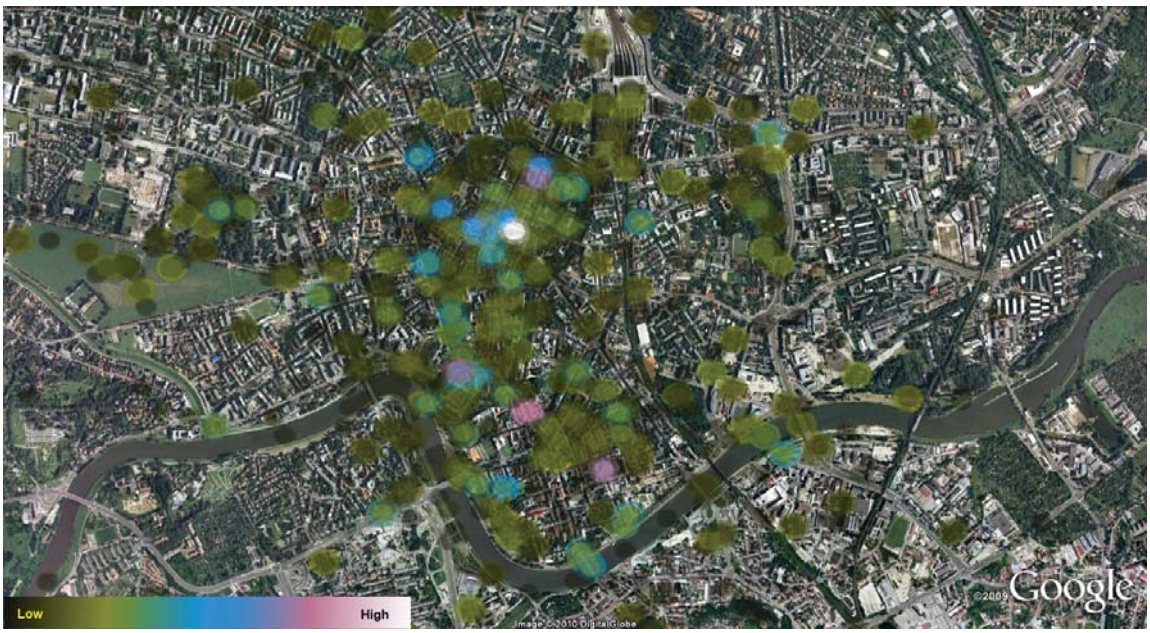
The approach used here consists of several preprocessing and linguistic analysis steps listed below, and is described in more details in Chapter 6:

1. Selection of photos that have at least one comment.
2. Removing comments that are written by the owner of the photo as a response to an earlier comment.
3. Removing comments of the same person if he/she wrote a number of comments about the same photo.
4. Cleaning comments by removing irrelevant text passages (URLs, tags).
5. Removing comments that are written in languages other than English (this is a current limitation of the approach).
6. Applying Part-of-speech tagger allowing to retrieve parts of speech (adjectives, nouns).
7. Applying automatic text analysis and extraction of opinion and sentiments.
8. Calculation of opinion and sentiment scores based on the scores of individual commenters.

The resulting opinion and sentiments scores are symbolized on a map. This enables the user to visualize interesting places and photos according to the opinions about the quality of photos and/or according to sentiments expressed about the objects depicted on the photos. While the opinion scores enable the user to focus on high quality photos, the sentiment scores enable the user to find places with different emotional connotations (e.g., "happy" or "sad"). Figure 4.4 illustrates locations and strength of user statements related to sentiments (Fig 4.4(a)) and opinions (Fig 4.4(b)) for the city of Krakow, Poland. In our experiments, the average processing time was 38 photos with comments per second.



(a) Krakow. Location and strength of user statements related to sentiments



(b) Krakow. Location and strength of user statements related to opinions

Figure 4.4: Visualization using opinion and sentiment scores. The color hues on the rightmost gradient scale correspond to the places with high opinion and sentiment scores.

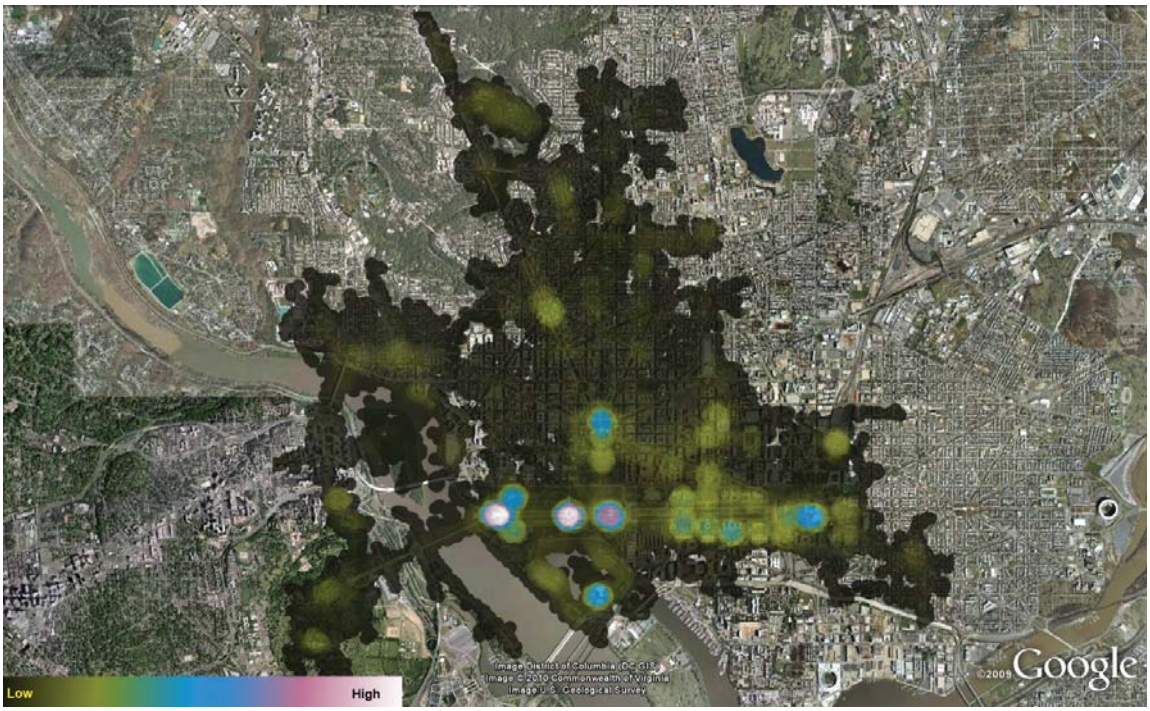
4.4 Visualization and Exploration

The visualization procedure generates heatmaps of photos by drawing colored filled circles for every photo, using the photo's geographic coordinates as the center point of the circle. The

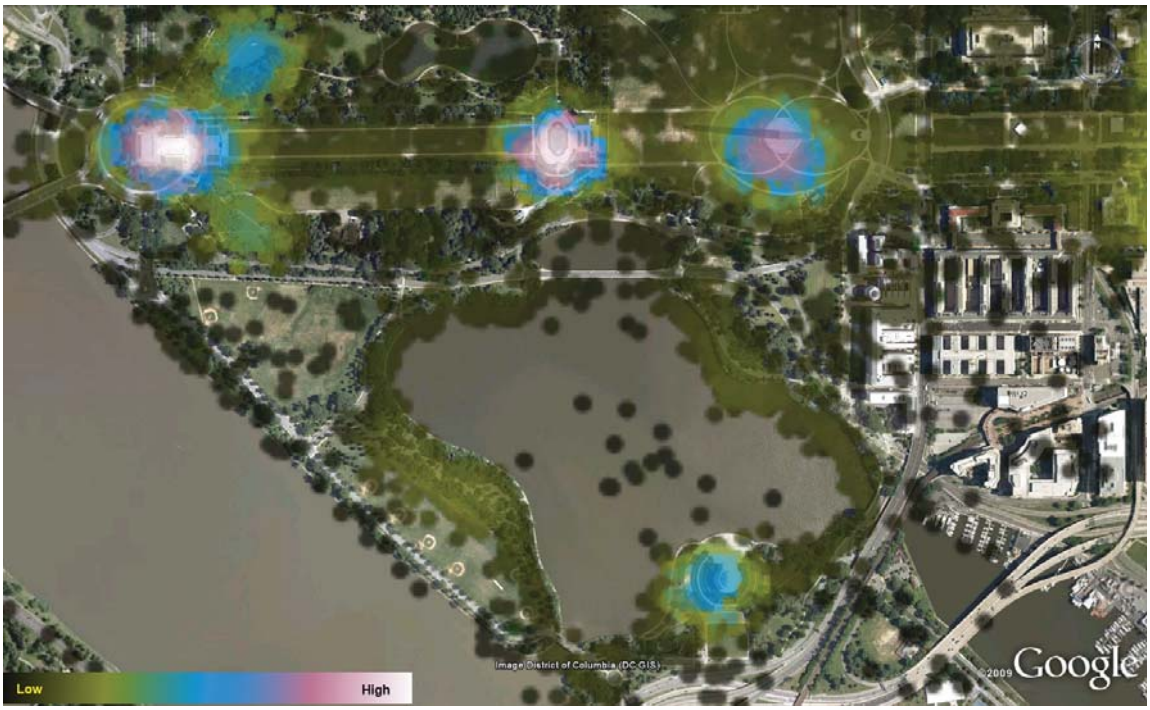
photo's weight, computed as part of density-based clustering procedure or as part of opinion analysis, is used for color-coding. Circle size is inversely proportional to the map scale. For example, when displaying information at the city level, the system displays significantly larger circles in order to depict the respective interesting places, whereas a detailed street level view needs finer grained points. Figure 4.5(a) shows the heatmap of interesting places for Washington D.C. using the radius of 20 pixels for every photo, while the heatmap shown in Figure 4.5(b) displays interesting places in Washington D.C. zoomed in to the area of Lincoln and Jefferson Memorials, National World War 2 Memorial, and Washington Monument, using the five pixel radius.

The exploration of places of interest is performed by overlaying a heatmap on an interactive map such as Google Earth using KML file format. The advantages of widely-used Web mapping technologies such as Google Earth include easy access to different layers of information and the wide-spread familiarity with the mapping interface. When combined with heatmaps of interesting places, these technologies facilitate the exploration by offering areal imagery as a background layer and navigational tools of a virtual globe. Figure 4.6 shows an area of Washington D.C. with several layers of information: a heatmap of interesting places, other prominent places provided as a reference, Panoramio photos, and Wikipedia geotagged articles.

The two approaches for the exploration of interesting places described in Sections 4.2 and 4.3 also suggest additional exploration steps the user can perform. In the case of density estimation, when a place of potential interest is indicated by the density of photos taken in the neighborhood, an interesting place can be found on a number of photos. The user can focus on the area, zoom in and check the name of the place using, for example, the Wikipedia layer, and next inspect some of the photos. In the case of opinion and sentiment analysis, the interestingness of a place is defined by a single photo. Therefore, it is essential for the user to see that particular photo, and to retrieve all the relevant information including comments that people wrote about it. This can be achieved by creating an additional layer embedded in the KML along with the heatmap that includes URLs to photos and comments associated with those photos (see Figure 4.7)).



(a) Washington D.C. City zoom level using 20 pixels as the radius of a photo



(b) Washington D.C. District zoom level using 5 pixels as the radius of a photo

Figure 4.5: Visualizations using color-coding of photo weights with circles of different sizes inversely proportional to the map scale

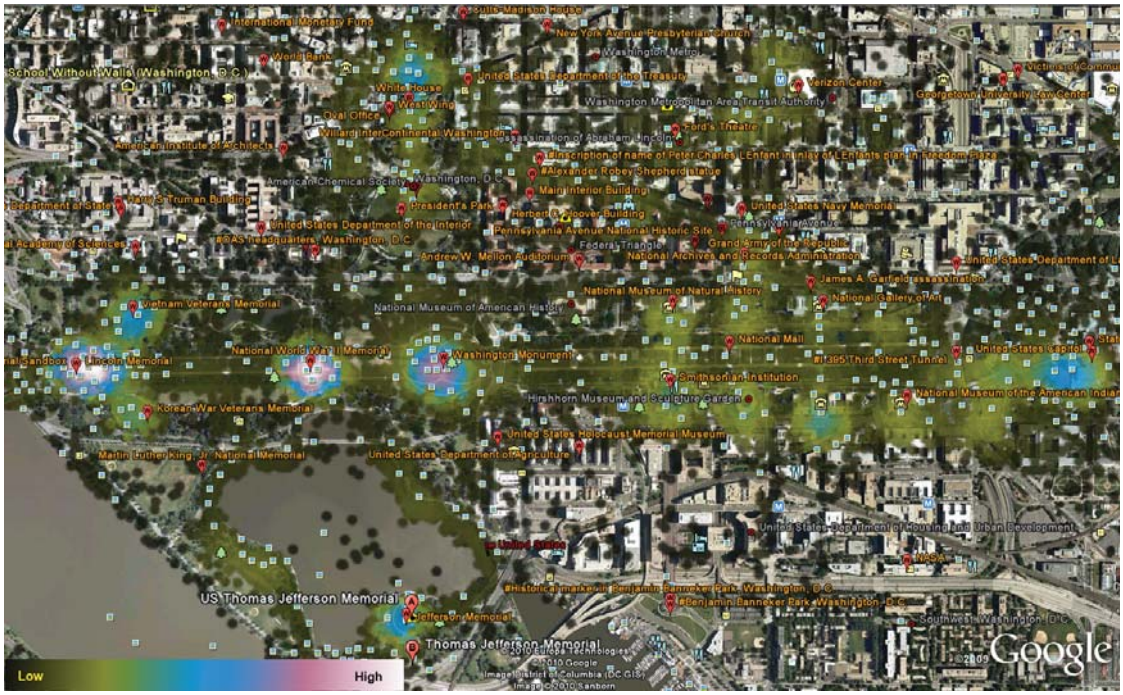


Figure 4.6: Washington D.C. Additional layers of information help to explore interesting areas

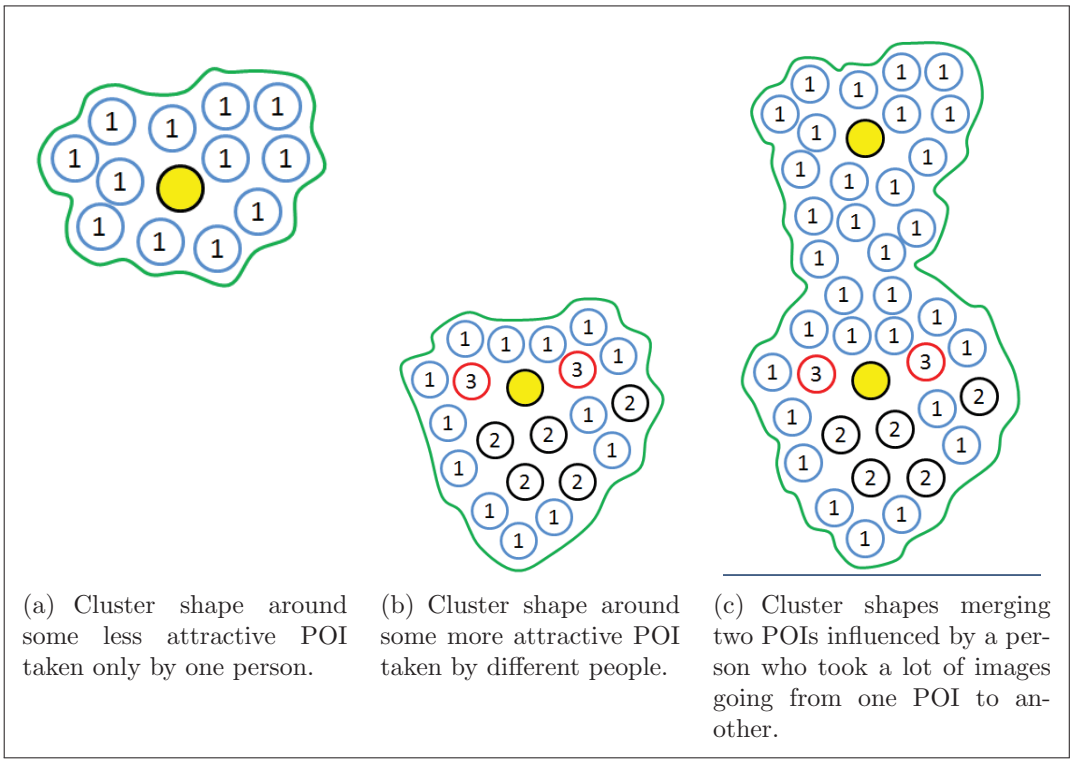
Figure 4.7: Berlin. Opinion scores

4.5 P-DBSCAN

4.5.1 Problem formulation

Density-based clustering algorithms can find clusters of different shapes and filter out noise. Thus, the analyst is not required to set the number of clusters in advance. Instead, he/she is required to set a distance threshold ϵ between two points in a cluster and a minimum number of points $MinPts$ (cardinality) around every point. However, having two such parameters, makes it difficult to agree on values and it requires a lot of trial-and-error before getting meaningful results [Ankerst et al., 1999]. Moreover, the usage of a single $MinPts$ parameter ignores the possibility that the density can vary not even in different regions, but also inside a certain cluster [Duan et al., 2007].

Several suggestions [Ankerst et al., 1999, Duan et al., 2007] were proposed to cope with problem of effectiveness of produced density clusters. However, the existing algorithms still provide a general-purpose solution which cannot utilize the unique properties of photo-based user generated data. Imagine, that people take pictures around some place. We can classify such places as attractive or not and our method is precisely proposed to be able to differentiate such places. We also assume that except for existing objective properties of a place which makes it attractive, people contribute to its value by their subjective behavior by means of taking photos there. Thus, the behavior of people who take photos influences a lot on the analysis of places.



(a) Cluster shape around some less attractive POI taken only by one person.

(b) Cluster shape around some more attractive POI taken by different people.

(c) Cluster shapes merging two POIs influenced by a person who took a lot of images going from one POI to another.

Figure 4.8: Hypothetical examples demonstrating problems in applying general density-based clustering on a photo dataset.

We consider two types of people for better illustration. The first type likes to take a lot

of pictures around while the second takes only few, high quality images. The first example on Fig. 4.8(a) illustrates the case when only one person takes photos around some place which he/she considers as interesting. The possible cluster is also presented, had a generic density-based clustering algorithm was applied on those points provided it complies to the basic requirement of a density-based clustering: every point in a cluster has a predefined minimum number of points around it in the neighborhood of a given radius. The reasons why only one person took pictures in a specific area can be infinite. It could be photos of his/her own car or a property, therefore the cluster is clearly not interesting to others, or it can be high quality photos of a lone photographer in a remote areas where nobody visits on ordinary occasions. Since we lack the semantic information, we assume that places can be graded according to the number of people who make photos there, thus, the place where only one person took photos has the minimum attractiveness. Fig 4.8(b) represents a more attractive place where 3 people took photos. Points labeled 1 and 2 represent people who take a lot of photos, while points labeled 3 represent a person who takes few photos. The possible cluster shape is also outlined provided the generic density-based clustering is applied. It can be seen that the shape of the cluster is mainly influenced by person 1 and 2 since generic density clustering algorithms do not distinguish types of points. This drawback leads us to the following observation: the shape of the cluster and number of points assigned to the cluster will change if we add or remove some points. Fig 4.8(c) illustrates such a case when person 1 continues to take a lot of photos by moving from one place to another. The new cluster encompasses two places and the number of points in a cluster includes points from two previous clusters presented in Fig 4.8(a) and Fig 4.8(b). Since generic density-based clustering algorithms will assign points to a cluster using only distance threshold and cardinality of every point, the cluster in Fig 4.8(c) looks very reasonable, however if we look on the ownership of every point, then we understand that the two clusters merged in Fig 4.8(c) are due to only one person which contributed a lot of images taking them going from one place to another. It is seen that addition of photos has an immediate influence on the shape and number of points in a cluster. It clearly shows the lack of robustness of clustering algorithm to the changes in the data, which can lead to poor results during analysis. Fig 4.9 illustrates the real example of clustering photos using DBSCAN in a highly attractive place such as Washington D.C. having many points of interest such as monuments, museums, and buildings. We fixed the number of photos to 100 and applied clustering using a neighborhood radius ϵ of 20 and 30 meters. Even though the neighborhood radius of 30 meters is relatively small, only three clusters are produced: the smallest one encircles Washington Union Station, the second one encircles Dupont Circle, while the largest one contains all the central points of interest of Washington D.C. Changing the neighborhood radius to 20 meters results in more clusters but the size of the clusters are still very large and contain many attractive areas inside.

From an observation over three theoretical examples and illustration in Figure 4.9 we clearly see that additional meta-information except for coordinates and distance between points is required. This meta-information is an ownership of a point. In addition, in many cases, the obtained clusters may have different densities in different parts of the cluster. Therefore, we introduce a notion of *adaptive density* to handle such cases. The basic idea is to split the cluster if different local areas of the cluster have large differences in density. The splitting should create

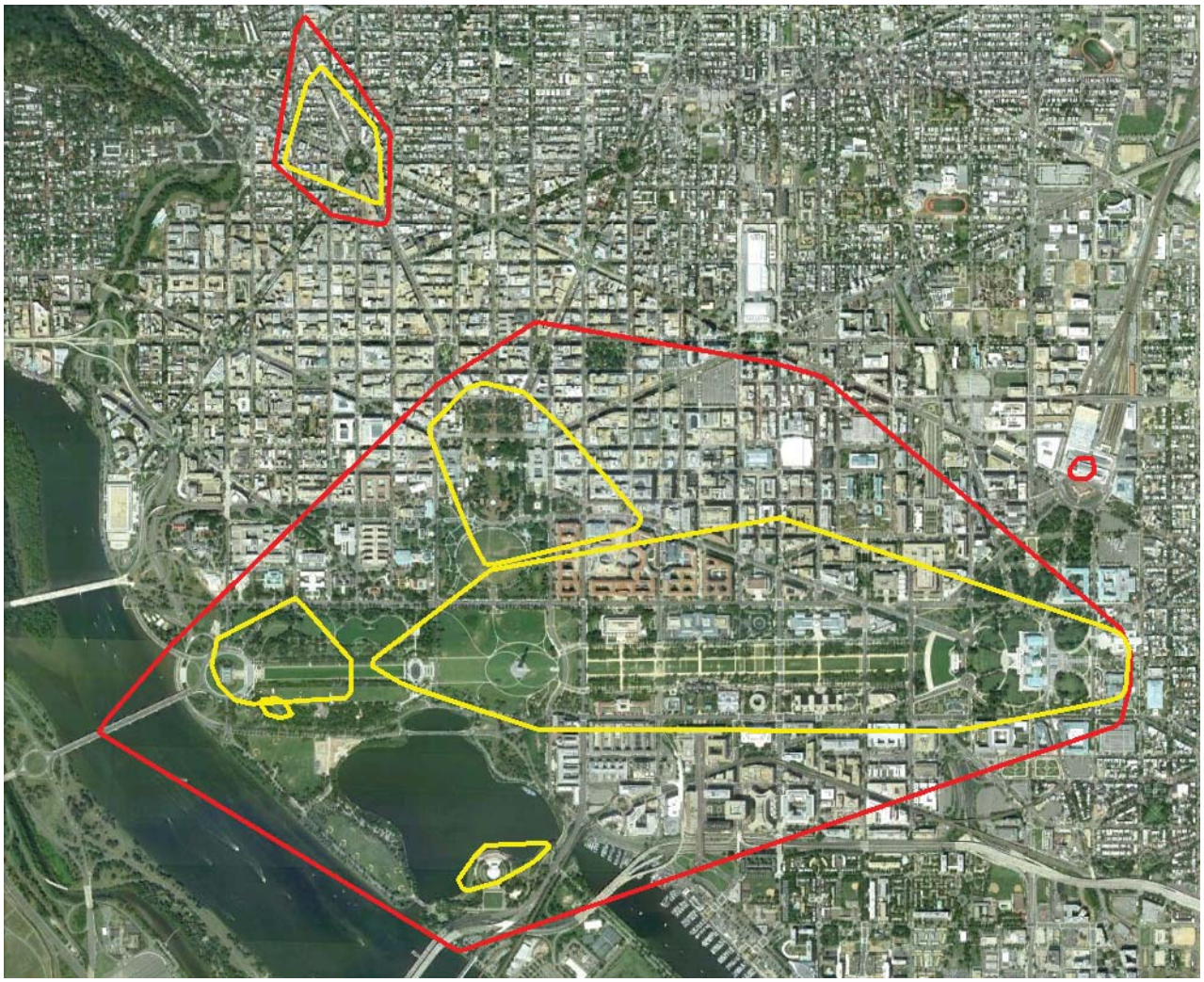


Figure 4.9: Applying DBSCAN on Washington DC using *MinPts* of 100 photos and neighborhood radius of 30 (red) and 20 (yellow) meters

small "packed" clusters in which density does not vary much. The combination of the ownership of photos with the adaptive density allows applying the algorithm even without defining the minimum number of owners in cases where the initial number is not known in advance or hard to estimate. The algorithm will create clusters of different densities while the selection of clusters of the required density (e.g. clusters that contain more than 100 photos or 100 owners) can be done in the post-processing step, by querying the database as described in Section 4.2.4.

4.5.2 Definitions

Following the terminology of the original work on DBSCAN [Ester et al., 1996], we provide our basic definitions of P-DBSCAN with respect to the new definition of density based on the number of people (owners of photos).

Definition 1. The **neighborhood** of a photo point p, denoted as $N_\epsilon(p)$, is defined by

$$N_\epsilon(p) = (q \in D, Owner(q) \neq Owner(p) | Dist(p,q) \leq \epsilon)$$

where $(o_{i_p} \in O) = Owner(p)$ is an ownership function and $Dist(p,q)$ is distance between points p and q. We require that for a photo point p, which belongs to the owner o_i, we find at least one point q whose owner is not o_i in a neighborhood of radius ϵ.
Definition 2. A **core photo** is a photo point where at least a minimum number of owners *MinOwners* not including the owner of the photo p took photos in the neighborhood of the photo p.
Definition 3. A photo point q is **directly ownership-reachable** from a point p when $q \in N_\epsilon(p)$.
Definition 4. A photo point p is **ownership-reachable** if there is a chain of photo points p_1, $p_2 \ldots, p_n = p$ such that p_{i+1} is directly ownership-reachable from p_i.
Definition 5. A photo is a **border photo** when it is not a core, but ownership-reachable from a core photo point.
Definition 6. Adaptive density is defined as the ratio of the current density of the neighborhood of a photo point p according to the **Definition 2** and the previous density. The neighbors of the photo are assigned to the current cluster until the density ratio is greater or equal to 1 (density increase).
We introduce two variations of *adaptive density*: *grow*, and *adaptive*.
Grow. The cluster expansion process continues while density grows or remains equal to the density on the previous iteration. On every iteration, the density threshold is updated with the current density.
Adaptive. The cluster expansion process continues as in the case of the *grow* adaptive density type. In addition, adaptive density drop threshold (0-100%) is introduced to soften the requirement for cluster expansion. In this case, the density drop which is less than the predefined adaptive density drop threshold indicates that the cluster expansion should continue even if the current density became lower then the density on the previous iteration.

4.5.3 Method

In this section we describe P-DBSCAN algorithm. Fig. 1 and Fig. 2 shows the pseudocode of P-DBSCAN.

The algorithm starts with arbitrary photo that is not yet assigned to any cluster and not defined as noise. If the photo is not core according to **Definition 2**, it is marked as noise (line 1.5). If the photo is a core, it is assigned to the current cluster and all the neighbors of the photo are queued for further processing (line 1.9), skipping the photos that were already processed or that are already in Q. The processing and assignment of photos to the current cluster continues until the queue is empty (line 1.10). The next photo is retrieved from the queue and assigned to the current cluster. If the static version of P-DBSCAN is running (line 1.12) (without adaptive density), the neighborhood of a photo p is checked and the neighboring photos are added to the queue if the number of owners exceeds *MinOwners* threshold, otherwise, the function AdaptiveDensity is invoked (line 1.18). In AdaptiveDensity (Alg. 2), several additional conditions are checked. The number of owners in the neighborhood of the point p is checked

Input: D - dataset of points with coordinates and ownership attributes, ϵ - neighborhood radius, Ad - adaptive density flag, Addt - adaptive density drop threshold (percents)
Output: Set of clusters

```
1  cluster-id = 0
2  while ((p = getUnprocessedPhoto(D)) ∉ ∅) do
3  |  CurrentDensity = MinOwners
4  |  if (|Neighborhood(p)| < CurrentDensity) then
5  |  |  MarkPhotoAsNoise(p)
6  |  else
7  |  |  cluster-id = cluster-id + 1
8  |  |  AssignPhotoToCluster(p,cluster-id)
9  |  |  UniqueQueue(Q,GetNeighborhoodPhotos(p))
10 |  |  while (Q is not empty) do
11 |  |  |  p = DeQueue(Q)
12 |  |  |  if (Ad == false) then
13 |  |  |  |  if |(Neighborhood(p)| > MinOwners) then
14 |  |  |  |  |  AssignPhotoToCluster(p,cluster-id)
15 |  |  |  |  |  UniqueQueue(Q,GetNeighborhoodPhotos(p))
16 |  |  |  |  end
17 |  |  |  else
18 |  |  |  |  AdaptiveDensity(...)
19 |  |  |  end
20 |  |  end
21 |  end
22 end
```

Algorithm 1: P-DBSCAN

against the current density. If the number of owners in the neighborhood is equal or greater than the current density, the neighbor photos of the photo p are queued and the current density is updated (line 2.9). If the number of owners in the neighborhood is less than the current density, the adaptive density drop threshold is checked. If the number of owners in the neighborhood drops below the threshold, the neighbors of the point p are not processed, otherwise the neighbor photos of the photo p are queued and current density is updated (line 2.5).

```
   Input: p
1  DensityDrop = 1 − |Neighborhood(p)|/CurrentDensity
2  if |(Neighborhood(p)| < CurrentDensity) then
3  |  if (DensityDrop < Addt) then
4  |  |  CurrentDensity = |(Neighborhood(p)|
5  |  |  UniqueQueue(Q,GetNeighborhoodPhotos(p))
6  |  |  AssignPhotoToCluster(p,cluster-id)
7  |  end
8  else
9  |  CurrentDensity = |(Neighborhood(p)|
10 |  UniqueQueue(Q,GetNeighborhoodPhotos(p))
11 |  AssignPhotoToCluster(p,cluster-id)
12 end
```

Algorithm 2: P-DBSCAN with adaptive density

4.5.4 Evaluation

For the experimental evaluation, we concentrated on the area of Washington D.C. spanning 308 km^2, Berlin (70 km^2), and London (126 km^2). We retrieved photos that were taken in year 2009. From this set, we removed photos having the same coordinate (regardless of the owner), leaving only one photo. Finally, we were left with 50,687 photos from 5,626 owners (Washington DC), 35,985 photos from 4,571 owners (Berlin), and 124,206 from 14,311 owners (London).

In the first evaluation, we applied $MinPts = 100$ (DBSCAN), $MinOwners = 100$ (P-DBSCAN) and $\epsilon = 20$. The goal of the evaluation is to compare DBSCAN and P-DBSCAN with and without the ownership information.

Fig. 4.10 shows the results of the clustering. The boundaries of the clusters were obtained using the PostgreSQL's Convex Hull spatial query. It can be seen that ownership information has a great influence on the clustering. The clusters produced by P-DBSCAN are the highly visited places in the region of Washington D.C. including Lincoln Memorial, Jefferson Memorial, National World War II Memorial, United States Capitol, National Museum of American History, National Natural History Museum, and two spots where people take photos of the White House.

In the second evaluation, we applied $MinOwners = 100$ (P-DBSCAN), $\epsilon = 100$ with and without adaptive density parameter. In the case of adaptive density we used *grow* and *adaptive* types of adaptive density with 10% for the density drop threshold. The goal of this evaluation is to compare P-DBSCAN with and without adaptive density.

Figure 4.11 and 4.12 show the results of the clustering using *grow* and *adaptive* adaptive types of adaptive density. The visual inspection of the clusters suggest that the density drop

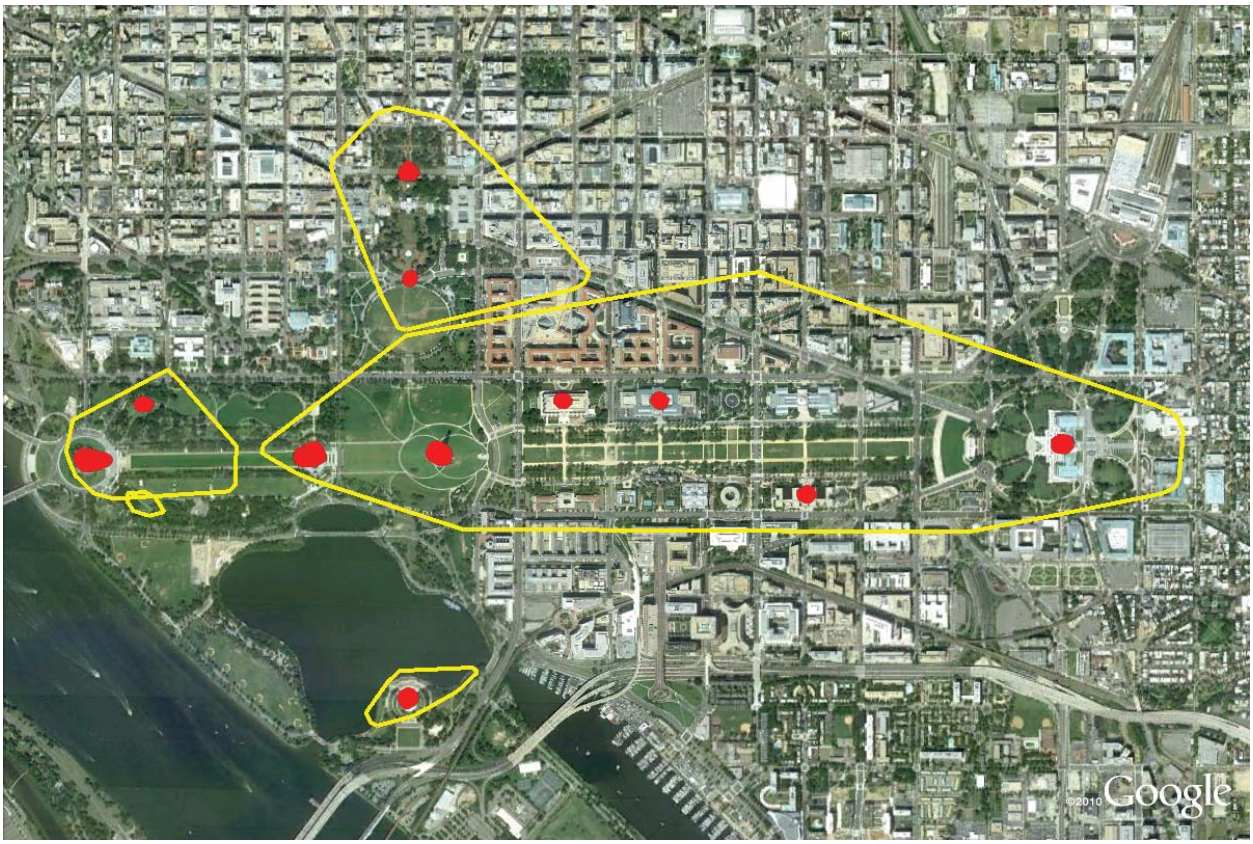


Figure 4.10: Washington D.C. DBSCAN and P-DBSCAN comparison using *MinPts* of 100 photos (DBSCAN) and *MinOwners* of 100 owners (P-DBSCAN) and neighborhood radius of 20 meters. DBSCAN - yellow clusters. P-DBSCAN - red clusters

threshold has small effect on the resulting clusters using the provided parameters. However, adaptive density threshold can be effective in cases when the algorithm starts extending clusters in the area with the highest density. In the case of *grow* type, the expansion will stop as soon as the current density is lower than the previous one but in the case of density drop threshold the expansion will continue.

Figure 4.13 (Washington D.C), 4.14 (Berlin), 4.15 (London) show the difference between the static density and the *grow* adaptive density.

The version of P-DBSCAN that uses static density of 100 people generates 9 clusters in Washington D.C. The problem with these clusters is similar to the one described in Section 4.5.1: the clusters are very large in the areas with a lot of people taking photos. For example, the area between World War II Memorial and the United States Capitol is described by one cluster that contains 2,869 people and 13,429 photos. However, P-DBSCAN with the adaptive density splits this cluster into 23 small clusters that are generated around some points of interest. In all the cases P-DBSCAN with adaptive density generates several smaller clusters instead of one large cluster. Similar behavior repeats on Berlin and London areas. In Berlin, 11 clusters are generated using static density. The two largest clusters cover the large area of Berlin with many attractive areas and contain 4,987 photos from 1,669 people (the cluster that covers Bundestag,

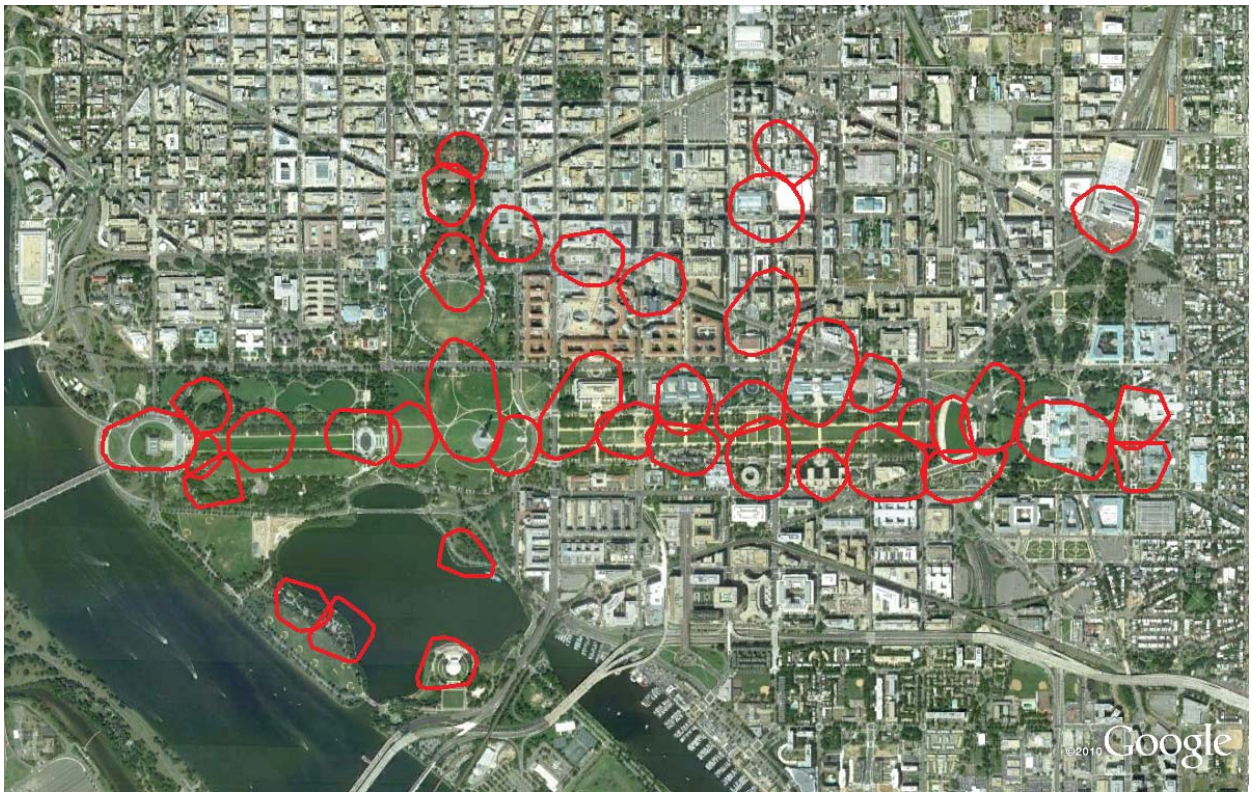


Figure 4.11: Washington D.C. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters and *grow* adaptive density

the Brandenburg Gate, the Memorial to the Murdered Jews of Europe), and 5,550 photos from 1,559 people (the cluster that covers the Near East Museum, the Old Museum, the German Historical Museum, the Red City Hall, the Neptune Fountain and other points of interest). In London, the largest cluster created using static density spans about $14km^2$. It contains 58,140 photos from 9,709 people - almost half of the total number of photos and people in London area. The P-DBSCAN with adaptive density splits this cluster into many clusters better reflecting the attractive places visited by people.

In general, it is difficult to evaluate density-based clustering algorithms because they always produce correct results in terms of the provided parameters. However, our goal was to show that P-DBSCAN is capable of generating clusters that reflect attractive areas as good as possible. Therefore, the goal of the evaluation is to show that P-DBSCAN improves the acquisition of various attractive areas using only coordinates of the geotagged photos and ownership information, without any contextual information (types of points of interest for example). For this case, we used the Wikipedia database as a source for POI data (Section 1.3). We evaluated three versions of P-DBSCAN (static density, *grow* and *adaptive*) on three large cities Washington D.C, Berlin, and London fixing the minimum number of owners *MinOwners* and neighborhood radius to 100 people and 100 meters. The results of the evaluation are presented in Table 4.2. According to the results, static density results in fewer but larger clusters that contain many POIs. The difference between *grow* and *adaptive* adaptive densities are not significant. However, the two

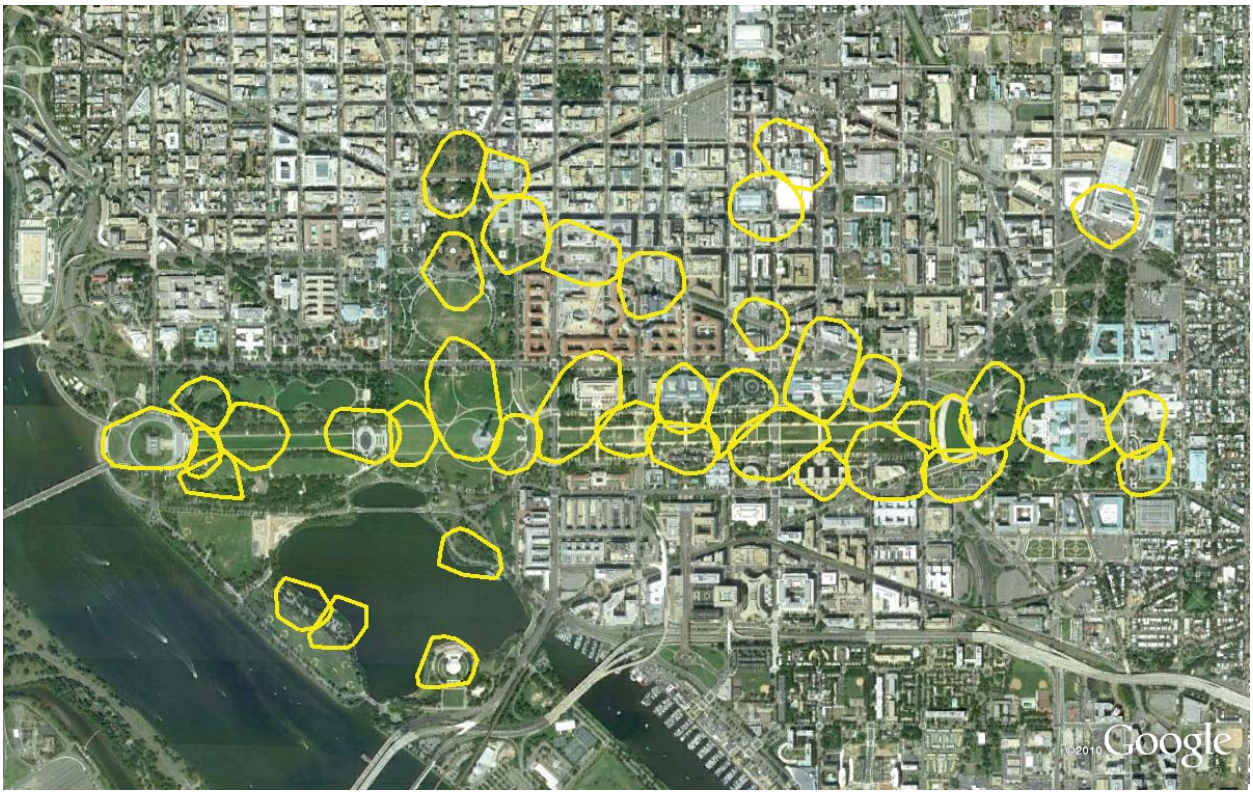


Figure 4.12: Washington D.C. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters and adaptive density with 10% density drop threshold

types of adaptive density considerably improves the cluster acquisition. The number of clusters generated by P-DBSCAN with adaptive density is two times higher at the lowest in Berlin and four times higher at the greatest in London. While the minimum distance to a POI in a cluster does not change considerably, the average maximal distance to a POI in a cluster drops almost by three times. Likewise, the average number of POIs in clusters falls by 70% for Berlin, by 80% in Washington D.C., and by almost 99.8% in London.

As was discussed in Section 4.5.1, P-DBSCAN with adaptive density solves the problem of initial parameter provision. The incorporation of the ownership information into the cluster expansion process compensates for the uncertainty with the selection of the density parameter. Figure 4.16 (Washington D.C), 4.17 (Berlin), and 4.18 (London) show the result of the clustering when *MinOwners* parameter was set to a minimum of 2 people using neighborhood radius of 100 meters. In the post-processing step we selected only those clusters that contain at least 100 people as described in Section 4.2.4.

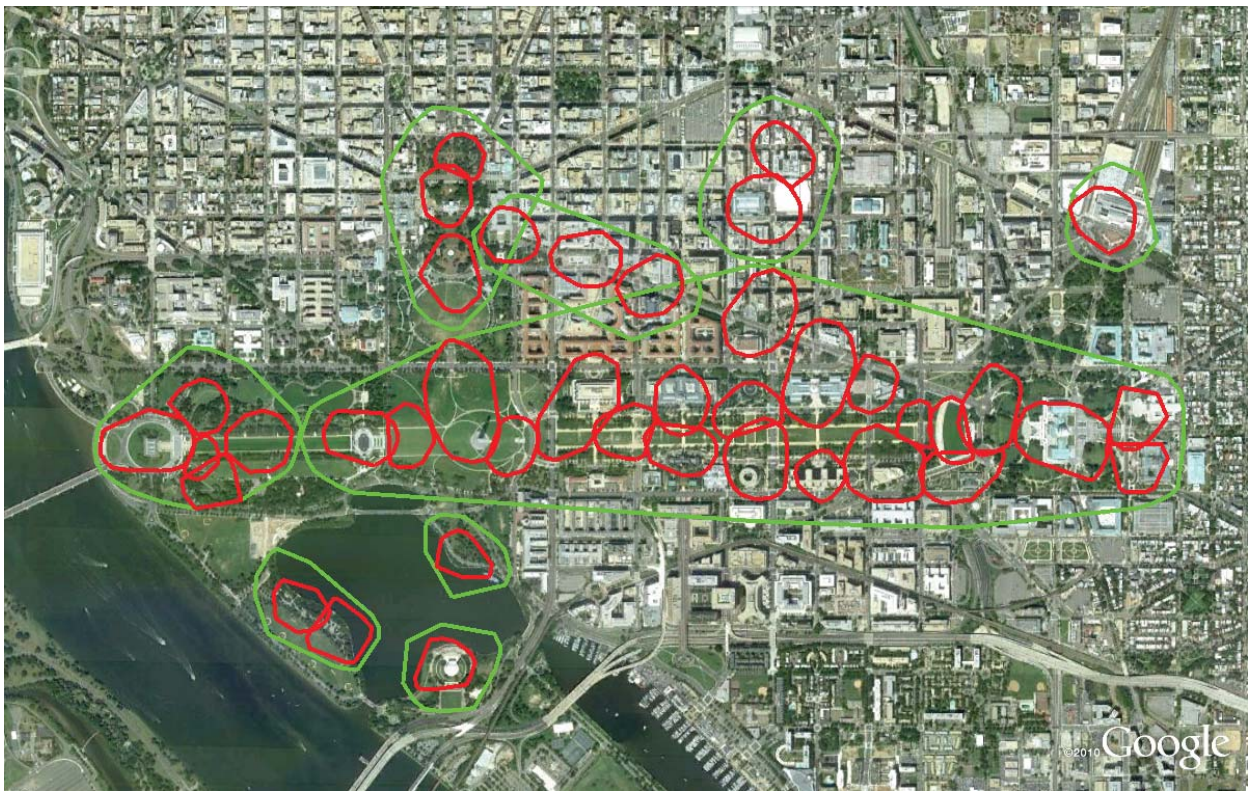


Figure 4.13: Washington D.C. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters (green clusters), and adaptive density (red clulters)

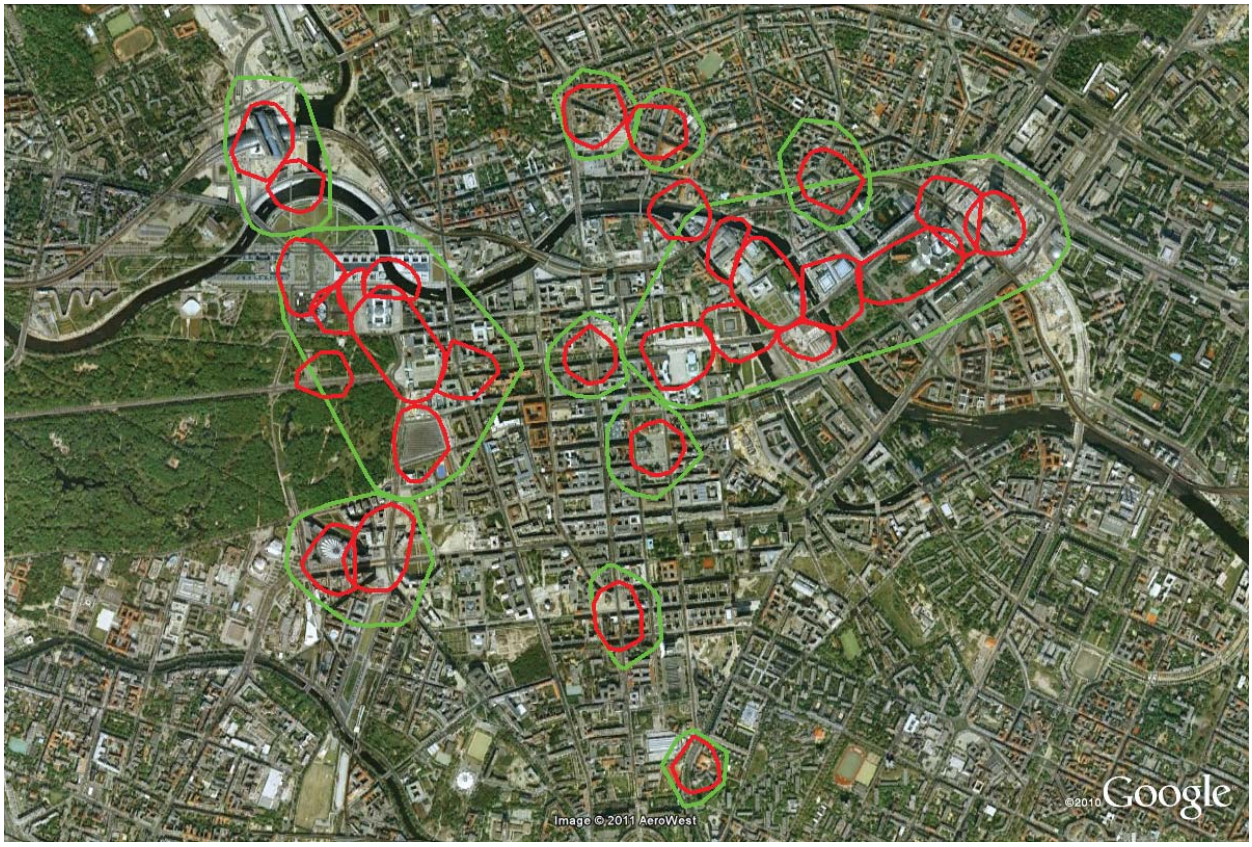


Figure 4.14: Berlin. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters (green clusters), and adaptive density (red clulters)

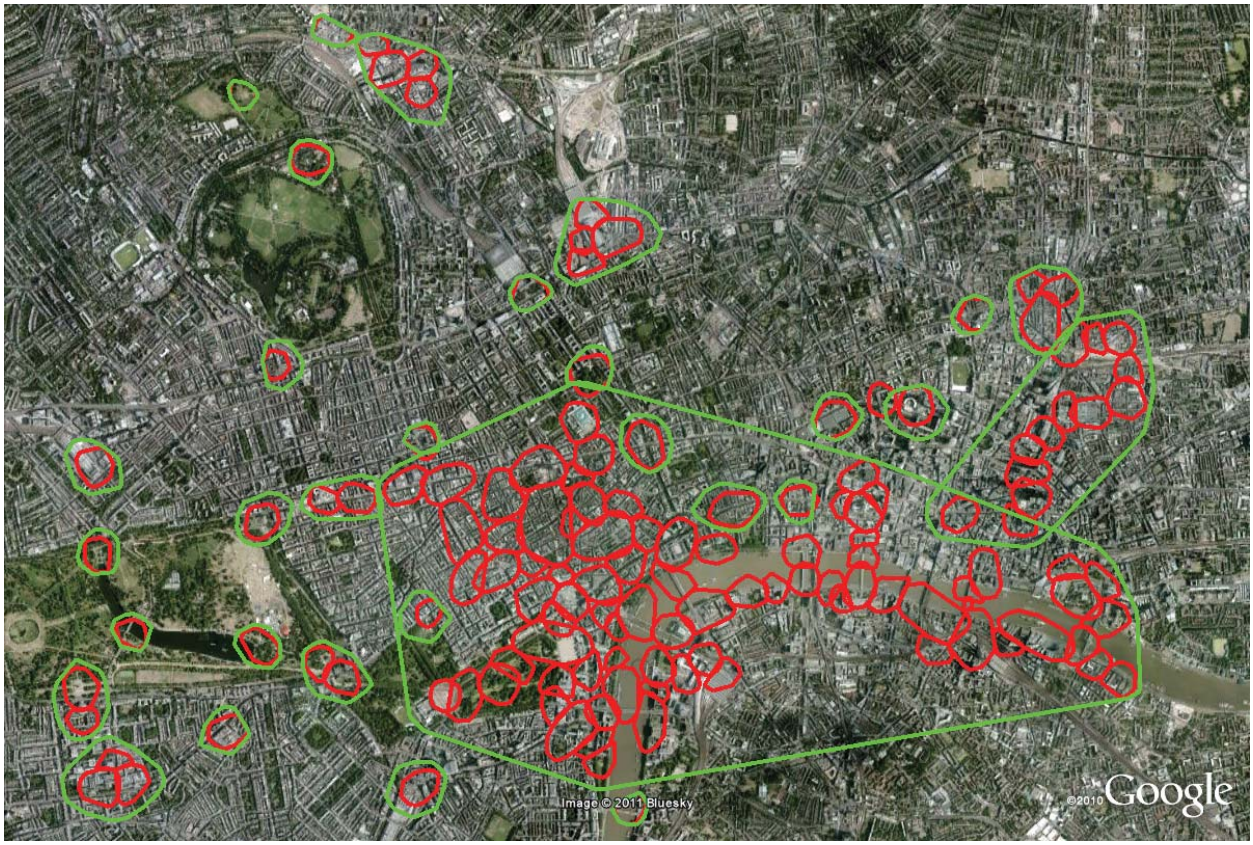


Figure 4.15: London. P-DBSCAN using *MinOwners* of 100 owners, neighborhood radius of 100 meters (green clusters), and adaptive density (red clulters)

Table 4.2: P-DBSCAN evaluation.

Washington D.C					
Method	**# of Clusters**	**Clusters with POIs**	**Average Minimal Distance to POI (m)**	**Average Maximal Distance to POI (m)**	**Average POIs in Cluster**
Static Density	15	13	43.95	278.78	12.00
Grow	46	37	49.07	82.69	2.49
Adaptive	47	36	47.74	85.17	2.50

Berlin					
Method	**# of Clusters**	**Clusters with POIs**	**Average Minimal Distance to POI (m)**	**Average Maximal Distance to POI (m)**	**Average POIs in Cluster**
Static Density	15	13	29.22	272.00	19.92
Grow	33	29	39.67	106.94	6.00
Adaptive	35	31	38.78	101.13	5.90

London					
Method	**# of Clusters**	**Clusters with POIs**	**Average Minimal Distance to POI (m)**	**Average Maximal Distance to POI (m)**	**Average POIs in Cluster**
Static Density	32	29	61.35	281.00	30.52
Grow	133	114	49.74	108.15	4.39
Adaptive	131	118	50.68	105.36	4.14

Figure 4.16: Washington D.C. P-DBSCAN using *MinOwners* of 2 person, neighborhood radius of 100 meters

Figure 4.17: Berlin. PP-DBSCAN using *MinOwners* of 2 person, neighborhood radius of 100 meters

Figure 4.18: London. P-DBSCAN using *MinOwners* of 2 person, neighborhood radius of 100 meters

5

Discovering frequent travel sequential patterns

Contents

5.1 Method 113

5.1.1 Dataset 113

5.1.2 Photo to POI assignment 114

5.1.3 Sequence Creation 114

5.1.4 Sequence Patterns 114

5.2 Evaluation 115

5.2.1 Case 1. Guimarães, Portugal 116

5.2.2 Case 2. Berlin, Germany 120

5.3 Discussion 124

This chapter presents a novel approach for analyzing the trajectories of people by finding semantically annotated frequent sequence patterns of people's movement.

5.1 Method

Fig 5.1 presents the proposed framework. First, we try to match photo coordinates with known POIs. Then the remaining unassigned photos are clustered and new POIs are identified. This is followed by converting the individual's trajectory into sequences of POIs. These sequences are analyzed and new sequence patterns are discovered. The following subsections describe each step in more detail.

5.1.1 Dataset

We used the Wikipedia database as a source for POI data (Section 1.3). For our purposes, the most important information that the entries contained were *id, title*, and *coordinates*.

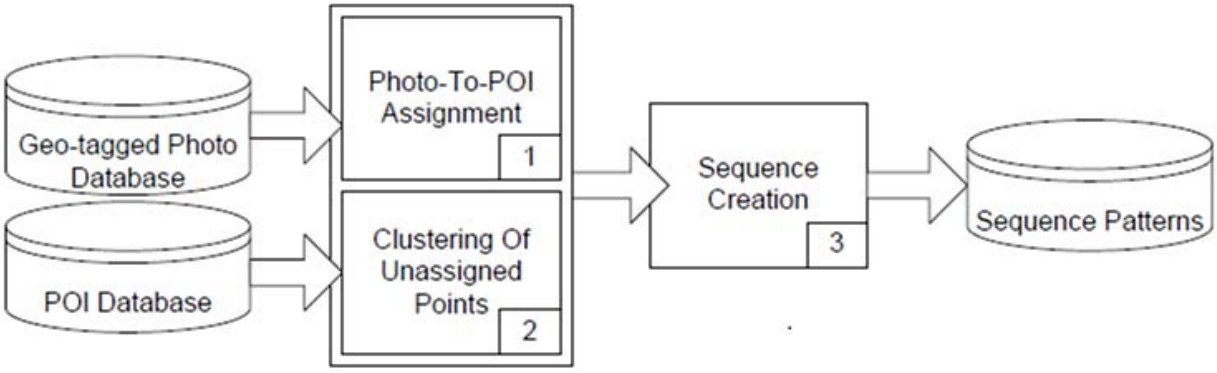


Figure 5.1: The framework overview for sequence patterns creation

5.1.2 Photo to POI assignment

In this step, every geotagged photo from the database is matched to a nearby POI using a distance threshold called *photo-to-POI*. If the distance between the photo and a POI is not longer than the photo-to-POI distance threshold, the photo is assigned to that POI. If there are several POIs within the distance threshold, the photo is assigned to the closest POI. In cases when there is a region in which no known POI is found in our database, we apply a density-based clustering algorithm to create regions of unknown POIs using the unassigned photos.

5.1.3 Sequence Creation

In this step, we assemble the POIs visited by a person into a sequence of places using the time stamp of the photo. If two consecutive photos are assigned to the same POI, only one photo is taken into consideration. We discard sequences that have only one POI since they do not contribute to discovering new sequence patterns. In general, sequences of any length can be built in this step. However, sequence creation can be constrained using such criteria as a time interval between every two consecutive photos or a total time interval between first and last photo. For example, Girardin et al. [2009] applied a 30-day interval threshold to differentiate between tourists, whose photo sessions lasted less than 30 days and locals whose sessions were longer. This heuristic approach can be used for differentiating between travel patterns of various groups of visitors. In our experiments, we implemented the same idea.

5.1.4 Sequence Patterns

The term "sequence pattern" usually refers to a set of short sequences that is precisely specified by some formalism. As is the practice in bioinformatics research, we are also adopting a regular expression in order to represent sequence patterns. A pattern is defined as any string consisting of a letter of the alphabet and the wild-card character '*'. The wild-card (also known as the "don't care" character) is used to denote a position that can be occupied by any letter of the alphabet. Here, we consider the Teiresias algorithm [Rigoutsos and Floratos, 1998] which was originally

developed as a combinatorial pattern discovery algorithm in bioinformatics for analyzing DNA sequences. The algorithm identifies recurrent maximal patterns within sequences. Although the method is combinatorial in nature and able to produce all patterns that appear in at least a (user-defined) minimum number of sequences, it achieves a high degree of efficiency by avoiding the enumeration of the entire pattern space. The algorithm, which has also been successfully used for information retrieval and intelligent manufacturing [Rokach et al., 2008b,a], performs a well-organized exhaustive search. In the worst case, the algorithm is exponential, but works very well for usual inputs. Furthermore, the reported patterns are maximal; any reported pattern cannot be made more specific and still keep on appearing at the exact same positions within the input sequences. Teiresias searches for patterns that satisfy certain density constraints, limiting the number of wild-cards occurring in any stretch of pattern. More specifically, Teiresias looks for maximal <L,W>patterns with support of at least K (i.e. in the corpus there are at least K distinct sequences that match this pattern). A pattern P is called <L,W>pattern if every sub-pattern of P with length of at least W operations (combination of specific operations and "*" wild-card operations) contains at least L specific operations. For example, given the following corpus of 6 trajectory sequences:

1. Reichstag → Der Bevölkerung → Brandenburg Gate → Memorial to the Roma and Sinti Holocaust Victims → Pariser Platz
2. Reichstag → Marienviertel → Memorial to the Murdered Jews of Europe → Brandenburg Gate
3. Reichstag → Berliner Dom → Liebknecht Bridge → Checkpoint Charlie → Brandenburg Gate → Treptower Park → Pariser Platz
4. Reichstag → 18th March Square → Brandenburg Gate
5. Potsdamer Platz → Zoological Garden → Marienviertel → Reichstag → 18th March Square → Brandenburg Gate
6. Sony Center → Pleasure Garden → Reichstag → Der Bevölkerung → Unter den Linden → Memorial to the Murdered Jews of Europe → Brandenburg Gate

The Teiresias program (L=K=2 and W=3) discovers 5 recurring patterns shown in Table 5.1. The first column represents the support of the pattern.

5.2 Evaluation

In this section, we present an experimental evaluation using two case studies of areas in Guimãraes, Portugal and Berlin, Germany. In particular, this experimental study has the following goals:

- To examine whether the proposed method can be applied to regions with different scales, number of persons and their photos, and several points of interest.
- To examine the effect on travel patterns of such parameters as the *photo-to-POI* threshold, the distance threshold for density-based clustering and the minimum number of people in a cluster (Section 5.1.2), and session length (Section 5.1.3).

Table 5.1: Illustrative results of the Teiresias algorithm

#	Sequence patterns
2	Reichstag → 18th March Square → Brandenburg Gate
2	Reichstag → Der Bevölkerung
2	Memorial to the Murdered Jews of Europe → Brandenburg Gate
3	Reichstag → * → Brandenburg Gate
2	Brandenburg Gate → * → Pariser Platz

Throughout the entire experimental process, we observed a constant session time of 10 days, a cluster threshold of three people and a minimum support K=5 of sequence patterns . We used session time as a heuristic for classifying people into locals and tourists. We classified a person as a tourist if she took photos during a period of no more than 10 days. Otherwise, he/she was considered as a local resident and his/her sequences were discarded. The following subsections describe the experimental study in detail.

5.2.1 Case 1. Guimarães, Portugal

Guimaraes is a relatively small city with historical roots going back to the 9th century. The city was the first capital of Portugal and is often called "the birthplace of the Portuguese nationality". UNESCO declared its historical section as a World Heritage site. In spite of its historical importance, only a very small number of people shared their photos on Flickr compared to the sharing of photos that is generally derived from other cities.

We defined an area of approximately 8.5 square kilometers around the center of Guimaraes with the following boundaries: longitude = 8.318° West and 8.276° East; latitude = 41.435° South and 41.457° North. From 2005 until October 2009, we were able to obtain only 391 photos from 152 people. The Wiki database contains only 11 POIs in the defined area: *Nossa Senhora da Oliveira, Guimaraes Castle, Palace of the Dukes of Braganza, Church of Sao Miguel do Castelo, Guimaraes Historical Center, Sao Paio, Dom-Afonso-Henriques-Stadion, Azurem University, Sao Sebastiao, Pousada de Santa Marinha, Oliveira do Castelo.* We used 200 and 400 meters as a distance for a *photo-to-POI* assignment (Section 5.1.2) in order to obtain the sequence patterns. We applied DBSCAN [Ester et al., 1996] on unassigned photos using a distance threshold of 100 meters and identified unknown POIs (Section 5.1.2). These new POIs were added to the existing POIs. A total of 342 photos from 127 individuals were assigned to existing and unknown POIs. Figures 5.2 and 5.3 show regions of existing and unknown POIs using a *photo-to-POI* threshold of 200 and 400 meters respectively.

The Teiresias algorithm [Rigoutsos and Floratos, 1998] discovered frequent sequence patterns of length two only. The general statistics pertaining to sequences and patterns are presented

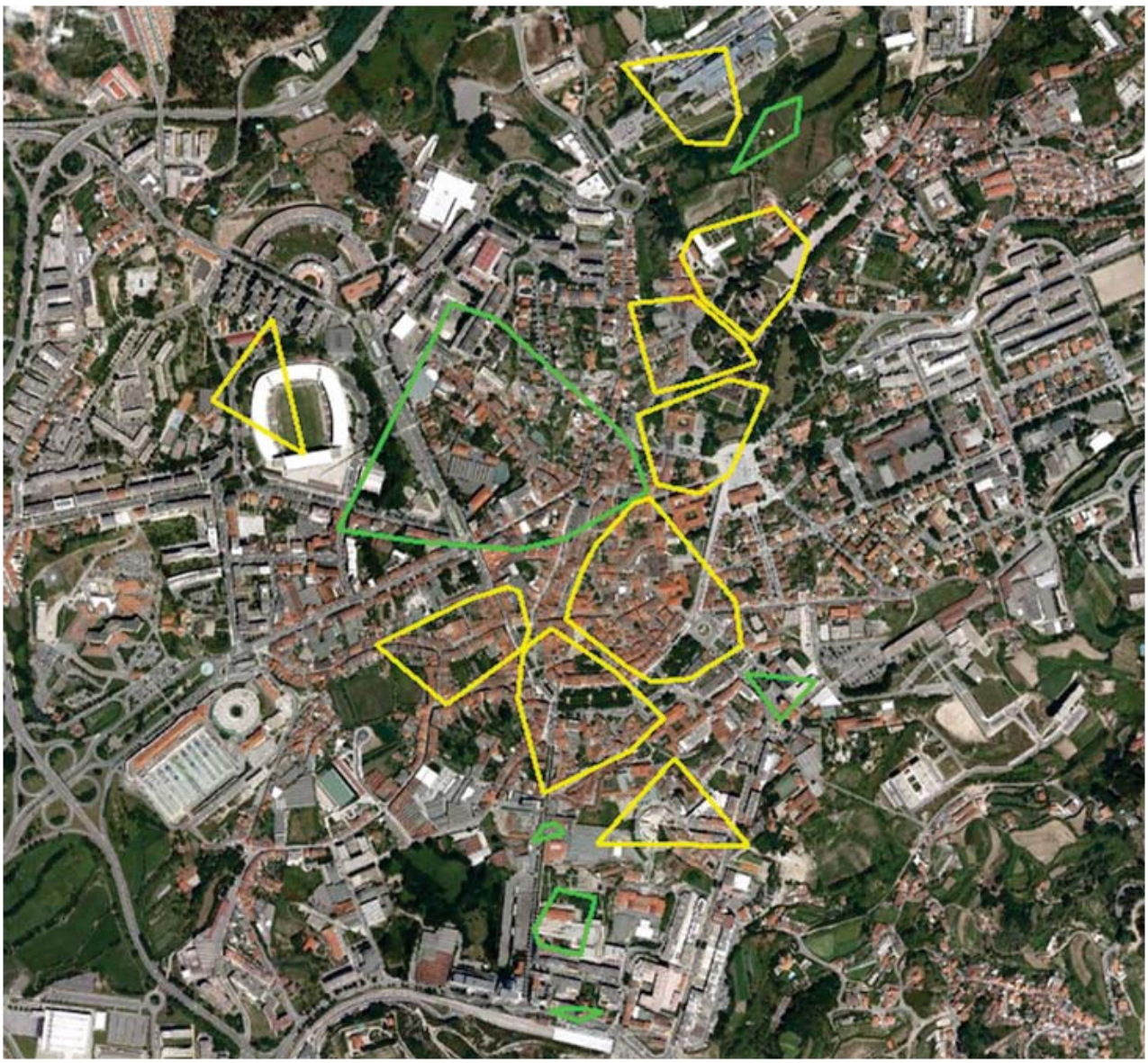


Figure 5.2: Guimãraes, Portugal. Cluster boundaries of photos assigned to existing POIs (yellow) using a *photo-to-POI* distance threshold of 200 meters. Cluster boundaries forming new areas of POIs were obtained using a distance threshold of 100 meters and a density threshold of three people in a cluster (green)

in Table 5.2. It can be seen that only 18 out of 127 sequences for a *photo-to-POI* threshold of 200 meters and 24 out of 138 sequences for a *photo-to-POI* threshold of 400 meters were created. There are two reasons for this. Firstly, the majority of people took photos in only one place. Secondly, some of the sequences were discarded because their length exceeded the 10-day threshold. Teiresias discovered 8 patterns using a *photo-to-POI* threshold of 200 and 7 patterns using 400 meters respectively. Table 5.3 shows five most frequent sequence patterns for every *photo-to-POI* threshold, where three generated sequences do not differ in two cases. The

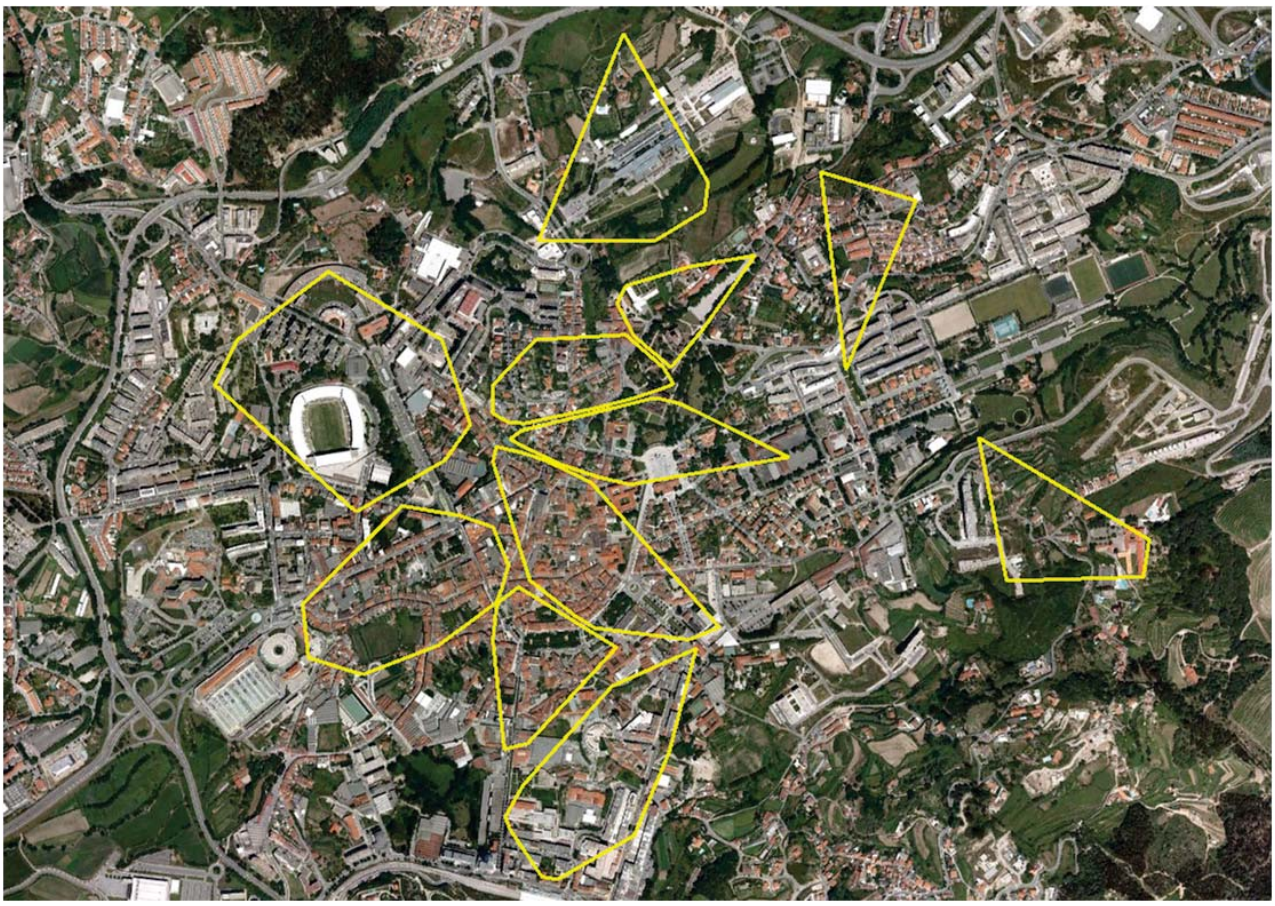


Figure 5.3: Guimãraes, Portugal. Cluster boundaries of photos assigned to existing POIs (yellow) using a *photo-to-POI* distance threshold of 400 meters

sequences that are different for 200 and 400-meter threshold are marked in bold.

Table 5.2: Guimãraes, Portugal. General statistics

Photo-to-POI threshold	<L,W>	# of people in sequences	# of valid sequences	# of sequence patterns
200	<2,3>	127	18	8
400	<2,3>	138	24	7

Table 5.3: Guimãraes, Portugal. Sequence patterns using L=2, W=3

Photo-to-POI threshold	# of input sequences	Sequence patterns
200	5	Guimaraes Historical Center → Nossa Senhora da Oliveira
	3	Guimaraes Castle → Church of Sao Miguel do Castelo
	3	Nossa Senhora da Oliveira → Church of Sao Miguel do Castelo
	3	**Church of Sao Miguel do Castelo → Nossa Senhora da Oliveira**
	2	Guimaraes Castle → Nossa Senhora da Oliveira
400	4	Guimaraes Historical Center → Nossa Senhora da Oliveira
	4	Guimaraes Castle → Nossa Senhora da Oliveira
	3	**Nossa Senhora da Oliveira → * → Nossa Senhora da Oliveira**
	2	Church of Sao Miguel do Castelo → Nossa Senhora da Oliveira
	3	Guimaraes Castle → Church of Sao Miguel do Castelo

5.2.2 Case 2. Berlin, Germany

Berlin is the capital of Germany and its largest city. It is one of the most popular tourist destinations in the EU. In 2008, a total of 17,758,591 persons visited Berlin according to European Cities Tourism Site[1]. Of this total, 7,033,593 people were classified as foreign visitors.

We defined an area of approximately 59 square kilometers around the center of Berlin with the following boundaries: longitude = 13.321° West, 13.474° East; latitude = 52.494° South and 52.543° North. We retrieved 76,824 photos from 9,401 people between 2005 and October 2009. The Wiki database contains 857 POIs in the defined area. We used 200 and 400 meters as a threshold for a *photo-to-POI* assignment. A total of 71,532 photos from 8,928 users (200 meters *photo-to-POI*) and 76,609 photos from 9,379 users (400 meters *photo-to-POI*) were assigned to existing. Of the rest unassigned photos, 5,292 photos from 2,239 users where clustered into 143 clusters using DBSCAN algorithm [Ester et al., 1996] during the second step of assignment of unassigned photos as described in Section 5.1.2 in case of *photo-to-POI* threshold of 200 meters. In case of *photo-to-POI* threshold of 400 meters, only 215 previously unassigned photos from 140 users were assigned to 22 clusters. Figures 5.4 and 5.5 show regions of existing (yellow cluster boundaries) and unknown POIs (green cluster boundaries) using a *photo-to-POI* distance threshold of 200 meters and 400 meters respectively.

The general statistics pertaining to sequences and patterns are presented in Table 5.4. Tables 5.5 and 5.6 present the five most frequent patterns of length two and three discovered by the Teiresias algorithm [Rigoutsos and Floratos, 1998].

Table 5.4: Berlin, Germany. General statistics

Photo-to-POI threshold	<L,W>	# of people in sequences	# of valid sequences	# of sequence patterns
200	<2,3>	8952	2844	2047
	<3,4>			186
400	<2,3>	8968	2845	2086
	<3,4>			195

From Table 5.4 we can see that using 2,844 sequences from a total of 8,952 sequences and a *photo-to-POI* distance threshold of 200 meters, the algorithm discovered 2,047 patterns of length 2 and 186 patterns of length 3. Using 2,845 sequences from a total of 8,968 sequences with a *photo-to-POI* distance threshold of 400 meters, the algorithm discovered 2,086 patterns of length 2 and 195 patterns of length 3. The first four sequence patterns of length 2 and 3 are identical for two *photo-to-POI* distance thresholds (Tables 5.5-5.6). The first three sequence patterns of

[1]http://www.europeancitiestourism.com/

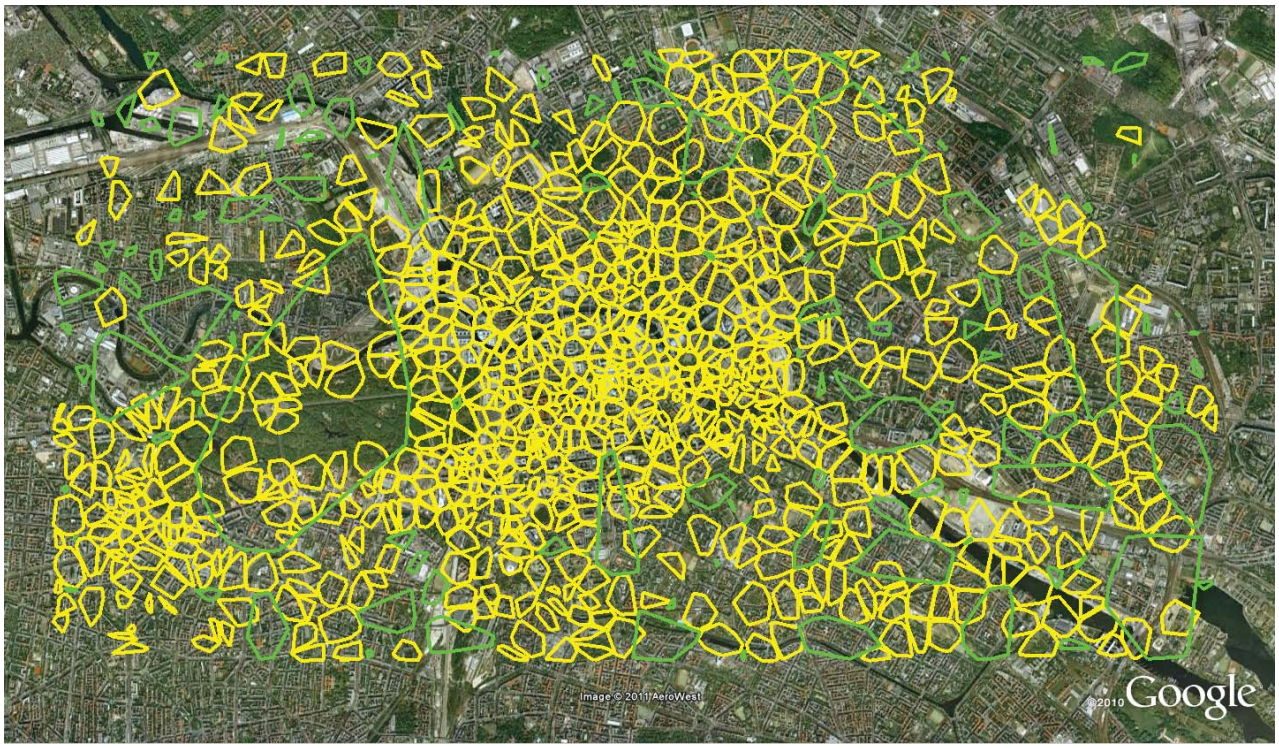


Figure 5.4: Berlin, Germany. Cluster boundaries of photos assigned to existing POIs (yellow) using a *photo-to-POI* distance threshold of 200 meters. Cluster boundaries forming new areas of POIs were obtained using a distance threshold of 100 meters and a density threshold of three people in a cluster (green)

Table 5.5: Berlin, Germany. Sequence patterns using L=2, W=3

Photo-to-POI threshold	# of input sequences	Sequence patterns
200	74	Brandenburg Gate → Reichstag
	53	Brandenburg Gate → Memorial to the Murdered Jews of Europe
	46	Brandenburg Gate → * → Reichstag
	41	Reichstag → Brandenburg Gate
	36	Pariser Platz → Brandenburg Gate
400	71	Brandenburg Gate → Reichstag
	51	Brandenburg Gate → Memorial to the Murdered Jews of Europe
	47	Brandenburg Gate → * → Reichstag
	43	Reichstag → Brandenburg Gate
	34	Reichstag → * → Reichstag

length 2 (Table 5.5) suggest that people began photographing at Brandenburg Gate and then continued to other places. The third sequence pattern in Table 5.5 contains a wild character

Figure 5.5: Berlin, Germany. Cluster boundaries of photos assigned to existing POIs (yellow) using a *photo-to-POI* distance threshold of 400 meters. Cluster boundaries forming new areas of POIs were obtained using a distance threshold of 100 meters and a density threshold of three people in a cluster (green)

indicating that that people started from Brandenburg Gate, then visited any POI and finished at the Reichstag. We should also note that unknown POIs created by applying density-based clustering are not part of the most frequent sequence patterns.

Table 5.6: Berlin, Germany. Sequence patterns using L=3, W=4

Photo-to-POI threshold	# of input sequences	Sequence patterns
200	13	Reichstag → Der Bevölkerung → Reichstag
	10	Brandenburg Gate → Memorial to the Roma and Sinti Holocaust Victims → Reichstag
	8	Pariser Platz → Brandenburg Gate → 18th March Square
	8	Reichstag → Brandenburg Gate → Memorial to the Murdered Jews of Europe
	7	Der Bevölkerung → Reichstag → Der Bevölkerung
400	14	Reichstag → Der Bevölkerung → Reichstag
	10	Brandenburg Gate → Memorial to the Roma and Sinti Holocaust Victims → Reichstag
	9	Pariser Platz → Brandenburg Gate → 18th March Square
	8	Reichstag → Brandenburg Gate → Memorial to the Murdered Jews of Europe
	7	Zeughaus → Alte Kommandantur → Lustgarten

5.3 Discussion

We demonstrated how an automatic data mining process could be used in finding travel patterns from a collection of geotagged photos. However, geographical data mining is far more complex process than its "classical" counterpart. There are several reasons for this:

1. Data quality, spatial precision and uncertainty play a crucial role in a spatio-temporal analysis.

2. Many spatial problems are ill-defined. This makes it impossible to apply fully automatic data-mining process to solving particular problems [Andrienko et al., 2007a].

3. The geographical analysis is very sensitive to the length or area over which an attribute is distributed [Miller and Han, 2009].

Data quality (spatial and temporal) and precision depends on the way the data is generated and should be taken into consideration during analysis and validation of results. Movement data is usually collected using GPS-enabled devices attached to an object or by geotagging images shared on the Web. For example, when a person enters a building a GPS signal can be lost or the positioning may be inaccurate due to a weak connection to satellites. These concerns are valid for geotagged photo data as well. Specifically, there are two ways to geotag a photo and upload it on the Web. One way involves attaching a GPS to a camera. In this case, the geotagging is performed automatically and the person can face the same problems as with conventional GPS devices described above. Alternative solution would be to manually annotate a photo during upload. In this case, several possibilities exist: the individual photographer may geo-annotate the object being photographed instead of the exact place where it was taken or the exact place could be geotagged with a different level of precision. In addition, the timestamp of a taken photo may not correspond to the correct time at which the photo was taken because of: (1) time zones differences between the user's country of origin and the visiting country, (2) careless setting of the camera's clock to some unrealistic time or (3) a software failure reading the timestamp of a photo.

In regard to the second issue raised in this section, there are two basic approaches for discovery of interesting sequence patterns: user-driven and data-driven. The user-driven approach is based on an expert's knowledge. However, it is not always efficient when an expert is required to find interesting sequences from thousands of sequence patterns such as was the case of Berlin. In our examples, we used frequency of patterns as a selection measure. However, frequent sequences do not necessarily constitute the most interesting patterns. In fact, frequent sequences usually represent the obvious patterns. Therefore, different interestingness measures for ranking patterns [Piatetsky-Shapiro, 1991] can be combined with the expert's knowledge to find some new unexpected patterns.

The difficulties associated with spatio-temporal data mining indicate that an analyst should select the parameter values very carefully and it is often done according to the expert's experience and knowledge. Unfortunately, we could not cover all the possible combinations of parameter values in our experiments. However, we demonstrated that changing only the distance threshold of the *photo-to-POI* while keeping all other parameters constant, may produce slightly different pattern sequences. Changing parameters at every step of our approach could lead to completely

new sequence patterns. While background knowledge of an analyst or domain expert could help overcome the weakness of the automatic process, some degree of human involvement is necessary for inspecting the data, tuning the parameters, controlling the analysis process and revising the obtained results. For example, an unknown POI can be discovered using the procedure presented in Section 5.1.2. The newly discovered POI may be adjacent to the region of an existing POI. An automatic process treats these two regions as distinct. However, visual inspection might reveal that the unknown POI belongs to the existing POI and that the two regions should be merged into one. Therefore, the solution to this issue is the incorporation of data mining techniques into geovisual analytics systems. Chapter 7 presents a GIS-based framework that we developed to perform geovisual analytics tasks and presents an example of the system usage applied to the task of finding frequent sequence patterns (Section 7.2.1).

6

Opinion and sentiment analysis of photo comments

Contents

6.1 Development of photo comments corpus 127
6.1.1 Data 127
6.2 Method 129
6.2.1 Definitions 129
6.2.2 Corpus-based lexicon generation 129
6.2.3 The adjective weighting model 130
6.2.4 Automatic opinion and sentiment analysis 133
6.3 Experimental evaluation 136
6.3.1 Design 136
6.3.2 Method 136
6.3.3 Results and discussion 137

This chapter presents a practical unsupervised approach to opinion and sentiment analysis of photo comments with a real-valued strength orientation. Our approach combines linguistic features for part of speech tagging, traditional statistical methods for modeling word importance in the photo comment corpus (in a real-valued scale), and a predefined lexicon for detecting negative and positive opinion orientation.

6.1 Development of photo comments corpus

6.1.1 Data

Region selection

Five regions (Dachau, Auschwitz, Wisła, Krakow and Warsaw) were defined for analysis. The rationale behind selecting these regions pertains to the following three goals:
(1) To find differences in comment types between regions.
(2) To find differences in the usage of parts of speech (adjectives and nouns).

(3) To build a model that represents the nature of photo comments.

We assumed that Dachau and Auschwitz concentration camps should contain special kinds of comments (negative emotions) that would differ from comments in general tourist locations. Wisła, we assumed, is a neutral region without many attractions while Krakow and Warsaw were selected as large Polish cities that include many tourist attractions. Table 6.1 summarizes the statistics related to the selected regions.

Table 6.1: Statistical information related to five regions selected for analysis

Region	Area	# commented photos	# owners	# commenters	# commented photos after preprocessing
Krakow	$120km^2$	8127	1257	23045	4214
Warsaw	$60km^2$	8690	1140	22695	4098
Wisła	$43km^2$	117	39	603	56
Auschwitz	$12km^2$	505	138	1687	311
Dachau	$14km^2$	329	121	1062	179

Preprocessing

Having manually examined hundreds of user comments, we found a similarity to blogs [Chesley et al., 2006], where opinions are stated in the beginning of the paragraph. Similar to blogs, the same user can write several comments about the same photo, but usually the first comment contains the opinions and sentiments, while subsequent comments mostly include neutral information like responses to comments of others or the photo owner. The following example shows two comments from the same user. In the first comment, there is an expression of sentiment ("Powerful place and story"). The second comment was made after the owner of the photo wrote his response.

(1) *This is great. I visited Dachau, but don't remember this part. but I hear they have added some things in the last 5 years. Powerful place and story, thanks for sharing*

(2) *I was there about 8 years ago and I don t recall this hall way. Was this one of the houses, or near the main complex where the museum and films were?*

As already mentioned, the owner of the photo can also participate in the discussion about his own photo. The following is a short example of two comments written by the owner of the photo to people as a response to their comments.

(1) *Thanks for the comments. I also found the colors both beautiful and chilling...a very creepy place for sure*

(2) *Thanks! I was fortunate to actually capture the impression it made on me standing there in person*

In this case, his opinions can introduce a certain bias, which suggests that comments of the photo owner should be excluded from the analysis.

For every region, we selected photos that contain at least one comment. We removed HTML tags and irrelevant sections (URL links, invitations to join a group). Next, we applied a language

guesser to remove comments written in languages other than English and applied Stanford POS Tagger [Toutanova and Manning, 2000] on the remained comments. Table 6.1 shows the number of remaining commented photos after the preprocessing.

6.2 Method

6.2.1 Definitions

Different terminology definitions are provided in the sentiment and opinion analysis literature. The terminology used in this research mostly follows the definitions given in Liu [2009], but makes a clear distinction between opinions and sentiments. The important terms and their definitions:

Photo Feature: Nouns that describe the photo features – attributes, components or characteristics of the photo, e.g. "shot", "photo", "colour", "composition", "light". Photo features in our case are usually related directly to the quality of the photo. It is common to distinguish between explicit and implicit features, i.e. features that are mentioned in a sentence and features that are not explicitly mentioned but implicitly referenced.

Orientation: The semantic orientation of a word or a comment as a binary categorical variable with the parameter values "negative" and "positive". Sentences or words that cannot be assigned to one of these two categories are implicitly rated as "neutral" and ignored in the further analysis.

Orientation Strength: The numerical strength of the orientation value ranging from 0 to ∞ in absolute numbers, whereas negative orientations are indicated by the algebraic sign "-".

Photo Opinion (PO): Negative or positive user statements, that clearly refer to photo features of a certain photo, are summarized as the respective photo opinion. They express the users' opinions on the technical and artistic photo quality. For simplicity, we will only speak of *opinions* when we refer to *photo opinions*.

General Sentiment (GS): Negatively or positively connoted user statements that cannot be attributed to a photo feature. As implied by the denotation, the general sentiment shall capture orientation statements that have a broader nature than opinions, i.e. sentiments and emotions that are evoked by the photo content. For simplicity's sake, we will only speak of *sentiments* when we refer to *general sentiments*.

6.2.2 Corpus-based lexicon generation

Opinion mining is heavily dependent on an opinion lexicon. To generate a lexicon there are the two common approaches, the dictionary-based and the corpus-based approach. The former is based on bootstrapping a seed of opinion words from dictionaries like WordNet [Fellbaum, 1998], SentiWordNet [Esuli and Sebastiani, 2006] or Wikipedia[1], the latter is based on the corpus and, thus, inherently domain dependent. We extend an existing general lexicon, the Internet General Inquirer lexicon, and adapt it to our domain computing an adjective-weighting model.

[1] http://www.wikipedia.org/

We applied a corpus-based lexicon generation due to different reasons:
(1) We want to generate a new lexicon in the domain of photo comments since currently, at least to our knowledge, no such lexicon is publicly available.
(2) Dictionaries like SentiWordNet may supply only a binary opinion orientation, while our task is to model opinion orientations on a real-valued scale.
(3) We want to investigate statistical properties of words used for commenting.

In order to acquire word distributions, we extracted adjectives and nouns from the corpus, counted their occurrences in the five selected regions separately, and sorted them according to their frequency from the highest to the lowest. Nouns were extracted in order to learn what words are commonly used as photo features. We used the Yago-Naga stemmer[2] to convert all nouns into a singular form.

To minimize the bias of some very active commenters, we counted word occurrence only once for each person for each region. The reason why we selected five separate regions is because word occurrences may differ due to different subject matters. Moreover, the number of commented photos is different from region to region and the word distribution would inevitably be biased towards words used in regions with many comments.

An inspection of the adjective distribution is quite surprising: The words *Great, Nice* and *Beautiful* are the most frequent and equally ranked adjectives in all five regions. For the complete list of 20 most frequent adjectives in the five regions please refer to Table 6.2. Among 100 frequent adjectives, 36 adjectives are unique, 58% of the adjectives are found in more than one region and 42% of frequent adjectives are found only in one region. This suggests that the vocabulary that people use to express opinions or sentiments is relatively small and contains many common words even if the context of photos is very different (e.g. Dachau concentration camp and Nature).

Next, we obtained the slope coefficients of word frequencies to check for existence of Zipfian distribution. The slope coefficients are the following: Krakow (-1.138), Warsaw (-1.136) , Auschwitz (-0.988) and Dachau: (-0.95) (Wisła was excluded because it does not have enough words for a reliable slope estimation). The results show that Zipf's law holds true not only for the English language as a whole but also for a particular parts of speech usage in photo comments.

6.2.3 The adjective weighting model

Having shown the statistical properties of the distributions of adjectives in the photo comments corpus, we are now ready to discuss the linguistic interpretation of adjective usage and propose an adjective weighting model for opinion orientation.

It was shown in past research that there is a strong correlation between the presence of adjectives and opinions [Wiebe et al., 1999, Wiebe, 2000]. Indeed, a careful analysis of photo comments showed that people often use short sentences like "Great photo", "Nice picture", "Sad place" to express their opinions or sentiments. The analysis also showed that the number of positive adjectives used in photo comments is higher than the number of negative adjectives and that overall, the number of positive comments is much higher than the number of negative comments. Any lexicon of positive and negative words will show that the words "Great" and "Nice" are positive. However, it is difficult to estimate which of these two words is "more positive than the other" using lexical features alone. As Osgood pointed out in [Osgood, 1957]

[2] http://www.mpi-inf.mpg.de/yago-naga/

Table 6.2: 20 most frequent adjectives and their frequency in five selected areas. Words that are commonly used in five regions are colored in yellow, in four regions - gray, in three - pink, in two - green, in one - white

Krakow	Warsaw	Wisla	Auschwitz	Dachau
great,1469	great,1403	great,26	great,129	great,65
nice,864	nice,856	nice,14	nice,61	nice,29
beautiful,829	beautiful,756	beautiful,13	beautiful,57	beautiful,29
good,311	good,306	lovely,8	good,42	fantastic,17
wonderful,271	wonderful,257	awesome,7	powerful,31	powerful,14
lovely,238	amazing,215	amazing,6	amazing,30	excellent,14
amazing,202	cool,191	cute,6	impressive,27	awesome,11
interesting,200	lovely,184	good,5	sad,24	amazing,11
cool,196	fantastic,181	such,3	wonderful,24	sad,10
fantastic,168	excellent,174	excellent,3	excellent,22	impressive,10
excellent,153	interesting,173	wonderful,3	fantastic,18	very,8
awesome,137	awesome,166	right,2	awesome,17	interesting,8
very,129	very,133	pretty,2	interesting,16	such,7
perfect,116	perfect,104	cool,2	very,15	wonderful,7
gorgeous,74	gorgeous,79	new,2	strong,13	dark,7
such,71	cute,78	very,2	many,12	lovely,6
cute,68	little,61	fantastic,2	same,11	cool,6
much,62	such,55	terrific,1	white,11	scary,6
little,58	stunning,47	fierce,1	such,11	dramatic,6
black,55	impressive,45	perfect,1	cool,11	good,6

a difference in "feeling-tone" exists even between synonyms such as "Good" and "Nice", but people are unable to verbalize the difference.

One of the simple approaches is to treat all positive words as equally positive, assigning a score of 1 for every occurrence of a positive word and counting the total number of positive words in a sentence or a document. Likewise, the negative words could be assigned a score of -1. Consequently, the final orientation of a sentence or a document would be the overall score (positive or negative) calculated by addition of all positive and negative scores [Turney, 2002]. This approach was used in previous research in the context of classifying the documents into positive or negative classes. Our task is different since we are interested in not only classifying the documents but also in ranking them according to the opinion or sentiment strength. As we mentioned, the majority of comments are relatively short and according to the statistics acquired from the five regions that we investigated, the vocabulary that people use to express their opinions or sentiments is relatively small. Thus, we hypothesize that a mere counting of positively and negatively oriented words will result in lack of sensitivity between the ranked comments. We claim that opinion or sentiment words should be scaled on a continuous scale denoting the difference in opinion or sentiment strength between those words. Therefore, we base our claim reflecting upon the seminal work of Osgood "The measurement of meaning" and using the Least Effort Principle and word distribution regularity presented by Zipf in his "Human

behavior and the principle of least effort". As we showed in Section 6.2.2, the orderliness of word distribution is preserved not only for particular parts of speech but also in every region. This indicates that even in special cases where photo comments are written by non-native speakers of English as well as by native English speakers, the fundamental principle that governs the word usage in a language is preserved even if a person is not aware of its existence as suggested by Zipf [Zipf, 1949]. Moreover, if any regularity or law did not govern the word usage it could mean that people do not attach any meaning to what they are saying or that they do not differentiate between words that describe the same concept. In the former case, we could observe a completely random word occurrence, in the second case we could observe that the frequency of word usage is the same no matter what word is used.

Similar to Osgood's measurement of meaning to compare "the output of two different subjects" measuring similarity or difference in meanings of a term, our goal is to quantitatively measure the opinion or sentiment strength. Unlike Osgood that builds differential scales for every concept (good-bad, slow-fast), we utilize Zipf's fundamental law of word usage by comparing how words that denote the same concept (positive or negative in our case) are used by people. This can be compared to a TF-IDF measure often used in information retrieval [Spärck Jones, 1972]. Let us assume that "Good" and "Nice" are the two words with equal frequency (TF is equal). According to TF-IDF, the least important word is the one which is found in most of the documents. Similarly, the word with the highest importance is the word that is found in the least number of documents. In this case, one of the words, let us say "Good", which is found in most of the documents will receive a lower score than the word "Nice". Similarly to TF-IDF, in our case, the most frequent word is the one which is found in most of the comments. Thus, its score will be lower than the score of the next most frequent word.

We define the word opinion strength w_{oo} using the principles of word importance as defined in the TF-IDF measure and word distribution properties of Zipf's law as follows:

$$w_{oo} = orientation(w) * log(\frac{f_{w,r=1}}{f_w} + 1) \tag{6.1}$$

where $orientation(w)$ is a function which assigns 1 if the word w is positive and -1 if it is negative, $f_{w,r=1}$ is the frequency of the word having the rank 1 (a most frequent word) and f_w is the frequency of the word w in the whole corpus.

The difference between TF-IDF and our approach is that the importance of the word in TF-IDF is measured for every word independently, while the opinion orientation score is calculated relatively to the most frequent word in the corpus. Thus, if the most frequent word is "Great" with a frequency of 1469 and the word ranked second is "Nice" with a frequency of 864, "Great" will receive a score of 0.30 (log (1469/1469 + 1)), while the score of "Nice" will be 0.43 (log (1469/864) + 1). One is added to *log* to avoid a zero score of the most frequent word.

We should note, that the word frequency in Equation 6.1 is absolute and can be applied to five regions separately. In order to create a global model that takes into account different word distributions, we need to find the relative order of all words from five regions. We proceeded as follows:

- We calculated a ratio $\frac{f_{w,r=1}}{f_w}$ for every word
- An average of ratios for every word was calculated taking these ratios for the same word $w_{i,n}$ from every region n

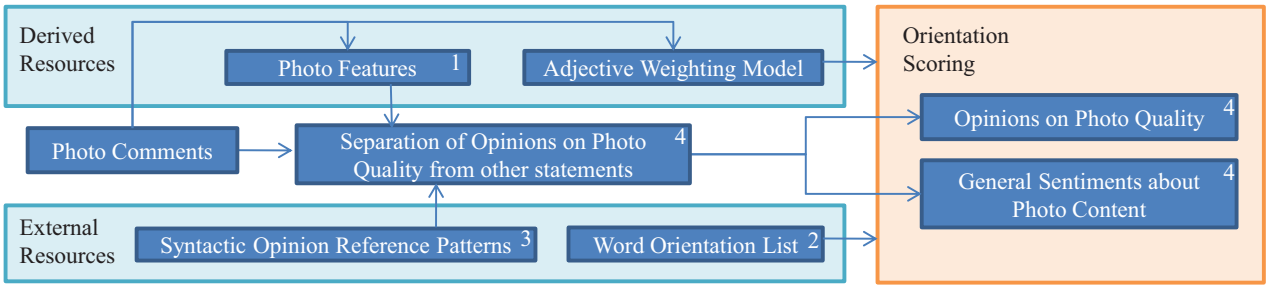


Figure 6.1: Interdependence of the different core text analysis processes. The numbers correspond to the paragraphs in Section 6.2.4, where details are provided.

- If the word $w_{i,n}$ was not found among the lexicon of the region n, its ratio was assumed to have the ratio of the last word in the lexicon of the region n

After building a weighted ratio for every word, we applied Equation 6.1 to obtain the global adjective weighting model.

6.2.4 Automatic opinion and sentiment analysis

The automatic opinion and sentiment analysis consists of several interdependent steps as outlined in Figure 6.1. The analysis relies on both resources derived from the photo comment corpus itself and external resources. The details are provided in the following subsections.

Photo features

In order to determine which opinions relate to the photo, first a list of photo features had to be compiled. For this purpose a term extraction method was created that exploits certain characteristics of photo features: (1) features usually correspond to nouns, (2) features should not depend significantly on the photo location, and (3) features should be frequent in photo comments. Consequently, (1) all nouns were extracted, that (2) appeared in photo comments of at least 4 out of 5 locations and finally (3) the 100 most frequent among these terms were extracted as candidate photo features. The list was then manually revised and finally, 50 out of these nouns were considered in the analysis as photo features. The top ten frequent nouns present in at least four locations were, in decreasing frequency order, “shot”, “photo”, “colour/color”, “composition”, “light”, “picture”, “capture”, “love”, “image”, “work”. Here, “love” is one example that was manually deleted. In this case we could observe that the high frequency of the noun “love” was due to a repeated error of the part-of-speech tagger, when occurrences of the verb “love” in very short sentences (e.g. “Love it!”) were misclassified as nouns.
Implicit features: A number of very short sentences implicitly refer to the photo quality without explicitly mentioning a photo feature (e.g. “I love it.”, “Well done.”, “Very nice.”). The common characteristic of such sentences is that they are very short and do not contain any nouns, i.e. do not contain any explicit target word for sentiments or opinions. Therefore, for very short sentences (less than 6 words) that did not contain any nouns, it was assumed that they implicitly related to the photo quality.

The word orientation list

A manually enhanced version of the widely used Internet General Inquirer lexicon was used as a word orientation list. It was applied to determine the orientation of the word and incorporate it into Equation 6.1, i.e. +1 for positive, -1 for negative and 0 for neutral words (not contained in the orientation list). All the words which were not contained in the adjective weighting model (Section 6.2.3), were allocated the weight of 1, because they either had not appeared in the photo comments or because they belonged to a different part-of-speech category.

Syntactic opinion reference patterns

In order to detect references of opinion words to photo features, a set of syntactic opinion reference patterns was defined, based on linear word order part-of-speech sequences[3]. A very simple example is the pattern "JJ NN", which stands for an adjective (JJ) directly followed by a noun (NN). In this case, we could be sure that the adjective referred to the noun. Hence, if the noun is a photo feature then the adjective and its orientation can be assigned to this feature. While in theory recursive patterns of arbitrary length (e.g. JJ* NN) are possible in natural language, in practice such patterns do not appear to a noteworthy extent in the domain under investigation. When we defined the pattern set, we started with including some very obvious cases like "JJ NN" and "NN VB JJ" and then skimmed through the data in search for further patterns. We could observe that the limited pattern set we defined covered the vast majority of cases. To verify the observation, we randomly drew sentences from the corpus until having encountered 100 opinion reference examples. While 90 were correctly covered by our patterns, no false positives occurred. The whole pattern set is provided in Figure 6.2. One main advantage is that the patterns encode the available linguistic knowledge about opinion references without requiring the time-consuming parsing of a full syntax structure tree or a typed dependencies graph.

Our syntactic reference patterns cover most of the cases that other approaches detect with dependency parses. This is because in English, adjectives are usually very close to the nouns they refer to or modify. Only very exceptional and infrequent cases like a relational phrase, e.g. the hypothetical sentence "the photo, that shows a tree, is really nice" cannot be resolved by our means. In case of verbs, our approach is not able to distinguish explicitly whether the feature is the subject or the object of the verb. In our tests, however, we could observe that this is not a problem. In addition, our method is less error-prone than dependency parsing, especially when applied to less formalized and sometimes sloppy and incorrect writing, as in user-generated content.

Identification and separation of photo opinions and general sentiments

A crucial part of the automatic text analysis is the detection and separation of (1) opinions about the photo quality (PO) and (2) general sentiments expressed about the photo content (GS).
The first part (1) is based on the extraction of photo features and the mapping of opinion statements to photo features. The described set of syntactic opinion reference patterns was

[3]The used part-of-speech tags follow the Penn Treebank Tag-set definition: http://www.comp.leeds.ac.uk/ccalas/tagsets/upenn.html

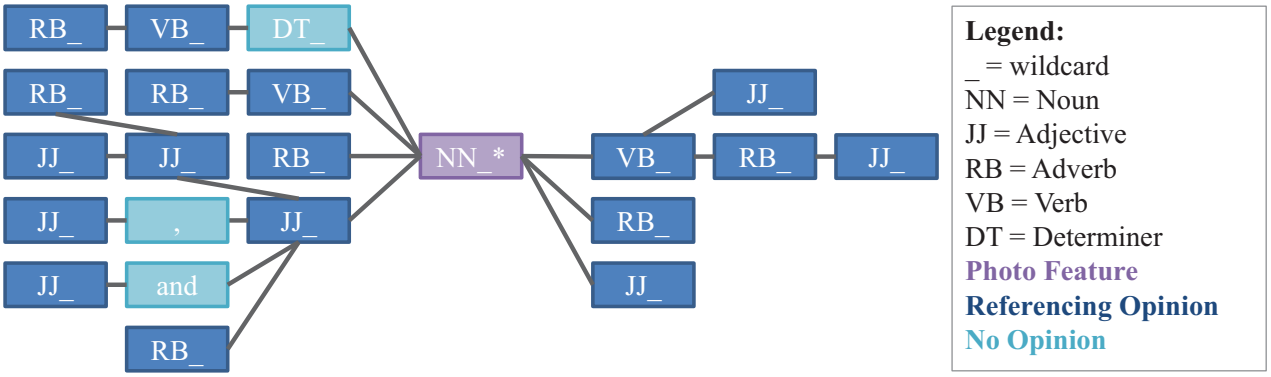


Figure 6.2: Syntactic Opinion Reference Patterns. Word order patterns go from left (before photo features) to right (after photo features), the distance to the photo feature indicates the exact position.

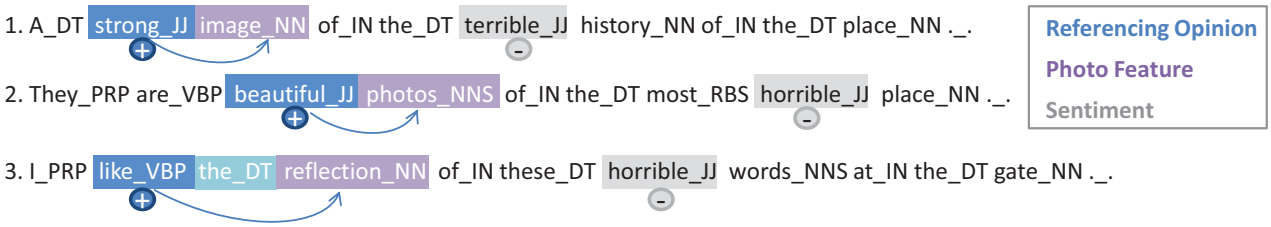


Figure 6.3: Three POS-annotated example sentences extracted from the photo comments. Each of these sentences contains both a photo opinion and a general sentiment.

applied for this mapping. For each photo feature in a sentence, all words were extracted that describe the feature according to one of the syntactic opinion reference patterns. The orientation scores of these words were then summed up to yield a photo opinion value. In this process, a simple heuristic is used to invert the orientation of negated words.

Accordingly, part (2) is based on all sentiment expressions that could not be attributed to photo features during step (1). This means that all the words that do not refer to photo features were considered and their orientation scores were summed up to yield a general sentiment value. Figure 6.3 provides some example sentences extracted from the photo comments, which contain both photo opinions and general sentiments. In all three cases, we have positive opinions ("strong", "beautiful", "to like") on the artistic quality of the photo, represented by the photo features ("image", "photos", "reflection"). The corresponding part-of-speech sequences are included in the syntactic reference patterns. The remaining opinion words ("terrible", "horrible") that could not be attributed to a photo feature are consequently considered as referring to general sentiments on the photo content.

It should be noted that clauses related to general sentiments are falsely classified as photo opinions only in very rare cases. The opposite situation, where photo opinions are falsely classified as general sentiments, could be observed in a couple of cases, due to different reasons (missing photo feature, implicitness).

6.3 Experimental evaluation

The goal of the experimental evaluation is to compare the performance of the proposed approach to the performance of human evaluators and to determine the factors that influence non-expert evaluators during opinion and sentiment strength assessment.

6.3.1 Design

A total of 78 participants were recruited to participate in the user study through Amazon's Mechanical Turk[4] of which 49 participants (31 females, 18 males) completed the assignment. The age of participants ranged from 18 to 67 (Mean 31.3, Std. 11.14, Median 28). The user study lasted for one week and was restricted to users from the US. Each person that accepted the assignment, received a questionnaire and a set of five text files containing user comments, which were gathered from photos randomly selected by the automatic procedure. The evaluator had to judge comments according to the criteria (opinion or sentiment), manually assigned to him/her by the user study manager. The evaluation procedure consisted of three steps.

In the first step, the participants provided some demographic information about themselves, such as age and gender. We also asked the non-native English participants to assess their level of English (basic, intermediary, high). 37 people were native English speakers. Six people stated that their level of English was high, five stated that their English was on an intermediary level, and one person stated to have basic knowledge of English.

In the second step, the participants had to read the comments in the files and rank them according to the opinion or sentiment strength from the most positive opinion or sentiment to the most negative. In total, 60 sets of comments were prepared for five regions: Auschwitz, Dachau, Krakow, Warsaw, and Berlin. Berlin was chosen as an additional region on which we applied our global weighting model. Sentiments assessment criteria was applied on comments from Auschwitz, Dachau, Berlin while opinions assessment criteria was applied on comments from Krakow, Warsaw, and Berlin. Every set contained comments from five photos. Every set was generated by randomly selecting photos from a particular region. 49 participants evaluated 137 sets reading 685 photo comments, which yields 2.79 sets per participant on average and 2.28 evaluations per set (some sets were evaluated by 3 participants).

The third step included eight closed-ended questions (see Table 6.4) to assess the additional factors that might have influenced the evaluator and one open-ended question to be filled by the evaluator in case if there was a factor that was not mentioned. A five point Likert scale was used for the closed-ended questions ranging from *strong disagree* to *strong agree*.

6.3.2 Method

Kendall's tau rank correlation was used to assess the degree of inter-rater agreement (IRA) between the ranks produced by the algorithm and the ranking of users. We applied the Intra-class correlation coefficient (ICC) [Shrout and Fleiss, 1979] to assess the differences in the opinion or sentiment scores assigned by the algorithm and the users. Since the users were asked to provide only ranks and not scores, an item ranked at the ith place by the user, got the score assigned to

[4] http://www.mturk.com/

it by the algorithm. This allowed us to avoid unnecessary complications with differences in user scoring. In all cases, the average IRA and ICC was calculated for every set for all users and then, the averaged IRA and ICC were averaged across sets in every region, across sets belonging to the same evaluation criterion (All Sentiments, All Opinions) and across all sets (All) without regard to a criterion (see Table 6.3). Finally, the Mann-Whitney U two-tailed test with significance level α of 0.05 was used to answer the question whether the rankings and the score differences between the algorithm and human evaluators are not statistically significant, i.e. whether the performance of the algorithm and the performance of the human evaluators are the same (the null hypothesis). The answers to the closed-ended questions were numerically encoded from -2 (*strong disagree*) to 2 (*strong agree*), and the mean, standard deviation, and median were calculated (see Table 6.4).

6.3.3 Results and discussion

Table 6.3 shows the average results of the algorithm-user and user-user rank and score agreements combined with standard deviation. In the case of Auschwitz and Warsaw, the rank and score agreements between the algorithm and the users are considerably higher than the agreement between users. In all other cases except for Dachau, the level of agreement is similar between the algorithm and users. We can observe a notably big difference between the algorithm and the users for the Dachau region where the user-user rank and score agreements are higher. However, the significance test (denoted as p-value) shows no evidence for statistical difference in all cases. Table 6.4 shows the answers of participants to eight questions.

The difficulties users experienced, e.g. completing the task and working with different interpretations, are reflected only on a moderate level of user-user agreement on the same comment sets. This tendency shows that for both opinions and sentiments criteria, the users' level of opinion is more similar to the algorithm than the level of agreement among the users. The fact that the user-algorithm agreement is about the same as the user-user agreement is a strong support for the algorithmic approach. It could not be expected that a user-algorithm agreement would exceed the user-user agreement in such a difficult task. The conclusion that can be drawn is that the algorithm in essence is equal to or as good as an average human user, which is promising.

As may be expected, the user-algorithm agreement is generally higher on opinions than on sentiments. As mentioned in Section 6.2.4, the algorithmic separation has a slight tendency to misclassify opinions as sentiments. While some opinions might be missed, the opinion score remains unaffected by falsely regarded sentiments and thus remains accurate.

The opinion analysis for the different regions Krakow, Warsaw and Berlin worked quite well. The sentiment analysis, in contrast, is more heterogeneous. While the algorithm worked well for Auschwitz, the results were less convincing for Berlin and especially for Dachau. A deeper investigation revealed the cause. Apparently, the comparatively low user-algorithm agreement in the Dachau Region was strongly influenced by one document set, in which three users heavily agreed in disagreeing with the algorithmic result. The user-algorithm agreement was -0.667 (IRA) and -0.65 (ICC), while the user-user agreement was 0.733 (IRA) and 0.985 (ICC). With the purpose of learning about the reasons for this strong deviation in agreement a further analysis was conducted. Interestingly, all users rated the top-ranked comment file as last and the second as penultimate, which was the main cause for the extreme user-algorithm disagreement. The respective algorithmically top-ranked comment file included many more comments than the

Table 6.3: Algorithm-User and User-User inter-rater agreement (IRA) and ICC

Data Set	Test	IRA (Avg/SD)	ICC (Avg/SD)
Auschwitz (Sentiments)	Alg-User	0.325 ± 0.419	0.457 ± 0.436
	User-User	0.164 ± 0.436	0.230 ± 0.481
Dachau (Sentiments)	Alg-User	0.038 ± 0.492	0.064 ± 0.607
	User-User	0.352 ± 0.398	0.275 ± 0.546
Berlin (Sentiments)	Alg-User	0.157 ± 0.415	0.073 ± 0.459
	User-User	0.187 ± 0.218	0.235 ± 0.613
Krakow (Opinion)	Alg-User	0.285 ± 0.427	0.319 ± 0.507
	User-User	0.411 ± 0.414	0.436 ± 0.553
Warsaw (Opinion)	Alg-User	0.440 ± 0.378	0.429 ± 0.399
	User-User	0.160 ± 0.488	0.155 ± 0.573
Berlin (Opinion)	Alg-User	0.380 ± 0.358	0.314 ± 0.521
	User-User	0.333 ± 0.563	0.248 ± 0.630
All Sentiments	Alg-User	0.213 ± 0.436	0.257 ± 0.506
	User-User	0.226 ± 0.382	0.245 ± 0.505
All Opinions	Alg-User	0.347 ± 0.400	0.345 ± 0.477
	User-User	0.324 ± 0.469	0.316 ± 0.573
All	Alg-User	0.287 ± 0.419	0.306 ± 0.489
	User-User	0.285 ± 0.436	0.288 ± 0.544

algorithmically low-ranked ones. While the latter ones each contained only one sentence expressing negative sentiment and no opinion at all, the two top-ranked comments contained both many positive photo opinions and many very negative sentiments. While our algorithm is tuned to ignore opinions when evaluating sentiments, the users in this case apparently behaved differently, as revealed by some of their answers.

One of the three users answered in the questionnaire that she agreed with the ranking order of the algorithm, despite ranking quite differently herself. The same user also agreed that her rating had been influenced by the overall number of comments in the comment file. It seems she down-ranked the files with more comments. The second user disagreed with the algorithmic ranking, but did not reveal any further details. The third one strongly disagreed with the algorithmic ranking order and stated not to have been influenced by the number of comments. However, her textual explanation was interesting: "*I looked at the sentiments expressed to determine if they were positive or negative. I did not take into account grammar or punctuation, I looked at what the comments had to say. Even though the one comment file had lots of comments, I felt many of them were more positive or actually opinions of the photo so I said this was the photo with the most positive sentiments contrary of what the algorithm concluded.*" Apparently her decision had been influenced by the large number of positive opinions, which somehow attenuated the impression of the negative sentiments.

Thus, it can be concluded that this case reflects weaknesses of the user study rather than

Table 6.4: Factors that influence the human evaluator

Question	Mean	Std	Median
I give lower ratings to comments with many typos	-0.531	1.174	-1
I give lower ratings to comments written in bad English	-0.347	1.251	-1
I give higher ratings to well-thought comments (the comments where people discuss what is so unique in the picture instead of just saying that the photo is good)	1.102	1.141	1
I give higher ratings to comments with many exclamation marks	-0.918	1.037	-1
I give higher ratings to comments if I encounter some type of words (among all possible) that relate to sentiment\opinion expressions	0.796	0.865	1
I weigh equally all adjectives with positive meaning Example: There is no perceptual difference between the two sentences (1) Beautiful place and (2) Moving environment	0	1.099	0
I weigh equally all adjectives with negative meaning Example: There is no perceptual difference between the two sentences (1) Ugly place and (2) Sad place	-0.224	1.104	0
My rating decision was influenced by the overall number of comments for a particular image	-0.449	1.081	-1

weaknesses of the algorithm.

Limitations

Photo comments may be quite long and each photo may have many comments. Memorizing several comment texts with respect to certain criteria and evaluating them in relation to each other is demanding. Therefore, we found 5 comment text files to be a good trade-off between providing the user with enough data to make his/her reply meaningful and at the same time not to overburden the evaluator. It is also not practical to ask users for real-valued scores without providing them with a sound basis for their decisions, since we did not want them to simulate any kind of algorithmic behaviour. Even the mere ranking of only a limited number of comments does not seem to be a trivial task. According to the users' remarks, differentiating between sentiments and opinion was especially difficult.

We tried to minimize the potential bias introduced by the outlined problems by designing the user task as simple and clear as possible. The drawback of giving only a small set of comments to each user was reduced by averaging over many different sets. Still, in the Dachau region one of the randomly drawn sets considerably influenced the overall result for the whole region. In addition, the users apparently had varying notions of opinions and sentiments. We tried to prevent this by explaining the differences carefully and providing a large list of examples at the beginning of the study. However, the results reflected that some people did not perceive negative

sentiments that strongly if they were coupled with positive opinions.

Additional insights from the questionnaire

In addition to the ranking, we requested the users to fill out a questionnaire in order to learn more about human behaviour when rating opinions or sentiments. The results show that users do not have the tendency to give lower ratings to comments if they contain many typos or are written in bad English. This is consistent with the behaviour of our algorithm, which does not provide special treatment for those cases, unless an opinion/sentiment word or photo feature could not be detected because of a typo. Additionally, users do not give higher ratings to comments with many exclamation marks, which are also ignored in our algorithm. Similar to our algorithm and to our expectation, users take the occurrence of sentiment and opinion words into account and do not tend to weigh them equal. In addition most users declared that they had not been influenced by the overall number of comments for a particular image, which is the only point where user behaviour deviates from the scoring strategy used by the algorithm. To a certain extent, our algorithm tends to give higher ratings to photos with larger number of comments since the opinion and sentiment scores are calculated based on a sentence, and then added up. In the case of online photo collections, this makes sense as more comments show a higher interest in the photo. Yet, the sentence by sentence score combination can be easily changed to a different strategy. For example by picking only one sentence that has the highest opinion or sentiment strength.

7 A Google Earth-based GIS

Contents

7.1 GEO-SPADE application and architecture 141

7.1.1 Overview 141

7.1.2 Main features 142

7.1.3 Architecture 143

7.2 Case studies 144

7.2.1 Analysis of tourist activity 145

7.2.2 Region exploration using geo-tagged photos 149

This chapter presents an extensible Google Earth-based framework, called GEO-SPADE, for handling geotagged photos. The two use cases presented in this chapter showcase the applicability of the Google Earth-based framework for different analytical tasks.

7.1 GEO-SPADE application and architecture

In this section we present the main features of the system and discuss the proposed architecture.

7.1.1 Overview

The client is an MDI (Multiple Document Interface) desktop application that allows multiple windows to reside under a single parent window. The client is written in c# programming language. We tried to keep the application as generic as possible by implementing the core features shared by all tasks. However, further extensibility is possible. The main user interface is presented in Fig. 7.1.

Views (labeled as 1 in Fig. 7.1) are the main views that encapsulate instances of Google Earth. The user can open several windows and multiple instance of Google Earth will be instantiated. The basic functionality of these views is the same as in the stand-alone Google Earth application: navigation to a specific location, panning, zooming, and KML layer loading. We also added support for custom shape creation by drawing on top of the map. This feature is described in further detail in the Section 7.1.2.

The session tree window (labeled as 2 in Fig. 7.1) maintains a list of open Google Earth views (sessions) and KML objects that belong to every session. This window supports removing, renaming and hiding of any KML object. Moreover, the textual representation of a KML object (labeled as 3 in Fig. 7.1) can be obtained by selecting it using a session tree node.

The toolbox window (labeled as 4 in Fig. 7.1) is utilized for basic geo-coding and defining the width and color of a line used for manual region selection on top of the map of the current session.

The window labeled as 5 in Fig. 7.1 records event logs and exception messages.

7.1.2 Main features

In this section we outline the main features of the proposed framework:

Multiple instances of the Google Earth - every view encapsulates a new Google Earth instance (labeled as 1 in Fig. 7.1). This feature allows the analyst to work independently on different regions or to separate the visualization of different objects applied to the same area.

Synchronization between views - when the exact positioning of several views is required for comparison, the view's boundaries can be synchronized to match the boundary of another view.

Session-based views - we introduced a notion of session when working with views. As was already mentioned, the analyst may work simultaneously with multiple views. If several time-consuming algorithms are activated, the application should know in which view the results of computation are directed. Since Web services are usually stateless, that is to say each individual request sent from the client should contain all the information because the server does not store the session state data [MicrosoftTeam, 2009], the session id of the view is sent along with other service specific information. When the service produces the KML result, the original session id is attached in the response. Therefore, when the client receives the server response, the framework tries to match the session id extracted from the KML to all existing sessions. If one is found, the result is visualized in that view, otherwise, the result is discarded. The discarding of a resul can occur, for example, when the analyst closes the view that was active when the service request was performed. Sessions and objects that belong to the session are displayed in a session tree (labeled as 2 in Fig. 7.1).

Shape drawing on top of Google Earth - we anticipated that some tasks may require manual region selection on top of the Google Earth map. This selection can be used in different scenarios such as finding objects that are located within the boundaries of the selection. The region creation is described by coordinates, line color and width, and stored as a regular KML object that can later be saved and reused.

Pluggable components - to keep the core system as simple as possible, it includes only the components necessary for storing and visualizing KML files, maintaining views, logging and exception management. The complexity of geo-related tasks is delegated to the custom implementation using pluggable components. The components follow specific requirements and conceptually do not differ from the plugins used in other frameworks like MapWindow GIS or OpenJUMP.

View separability - this is not strictly the feature of the framework, but the outcome of the MDI architecture. If two or more displays are available, the views and pluggable components can be positioned on different displays in order to isolate different tasks and to free more visual space. For example, the analyst may want to position two maps on separate displays or separate between session views and the main UI.

Chaining of analysis steps - results obtained from the server are visualized using KML. However, the elements of the KML are objects that hold some state and relevant information (for example, cluster boundaries or area). The analyst can further refine the results by focusing on the subset of the obtained objects. Consequently, a new invocation of an algorithm can be accomplished using the selected objects to pass information.
Service oriented architecture - allows service reuse and interoperability across programming languages and platforms. Changes in the service implementation remain transparent to the client as long as the number and naming of parameters in a service contract do not change.

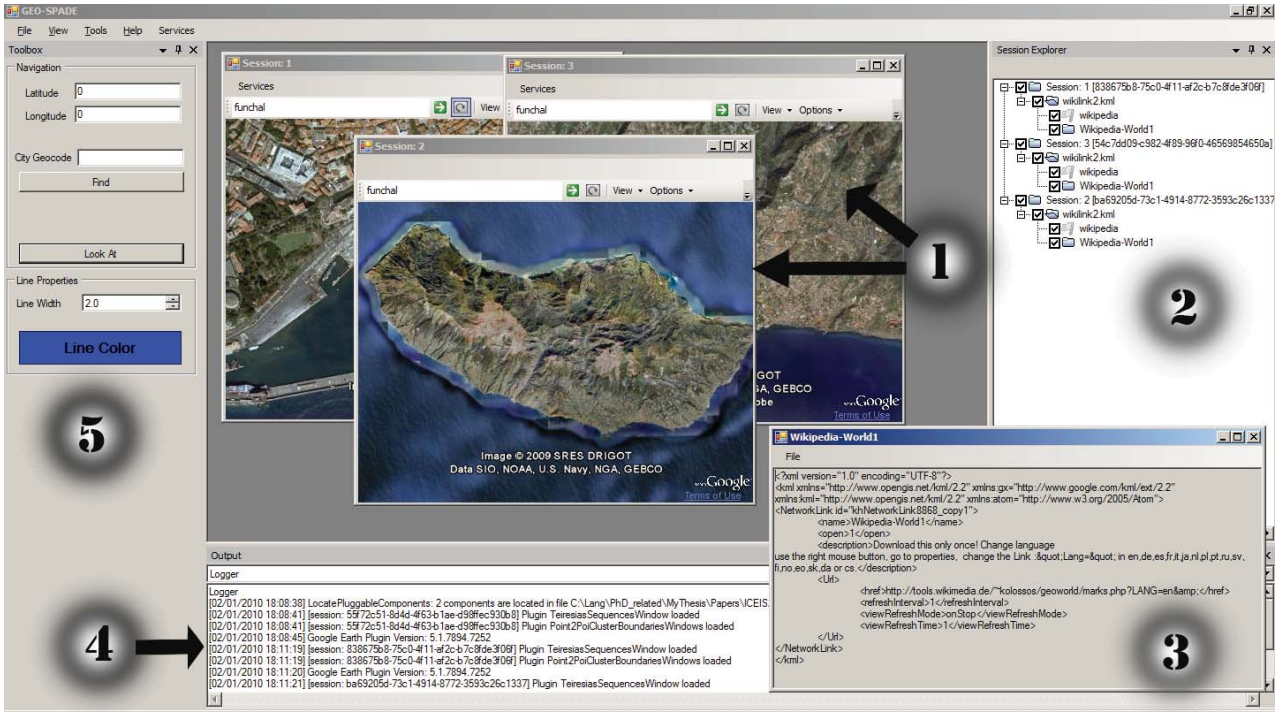


Figure 7.1: Basic GEO-SPADE components

7.1.3 Architecture

The proposed architecture is based on a thin-client paradigm whose main purpose is to pass off geoprocessing to a remote server using Web services. The core system visualizes the results, drawing upon any of the Google Earth wrapped views as described in Section 7.1.2. However, the provider of the plugin component is responsible for the basic UI view and complex interaction if needed (see Section 7.2 for illustration).

The selection of a communication protocol depends on the service provider and different protocols can be integrated as extension points. However, the application proposes asynchronous REST (representational state transfer) services [Fielding, 2000]. REST is an architectural style of networked systems and relies on HTTP protocol and data exchange based on XML. Since REST

does not deal with implementation details, the Web services based on REST can be created and consumed by different programming languages. The advantages of using REST-like architecture in GIS are described in WPS standard specifications. The extension mechanism is performed by creating pluggable UI components whose basic purpose is to obtain from the user the parameters for invoking the Web service. The overall architecture of the application can be seen in Fig. 7.2. The geoprocessing task can be provided as one pluggable component and/or can be split into a chain of steps. The server side is presented as a collection of geo-processing services.

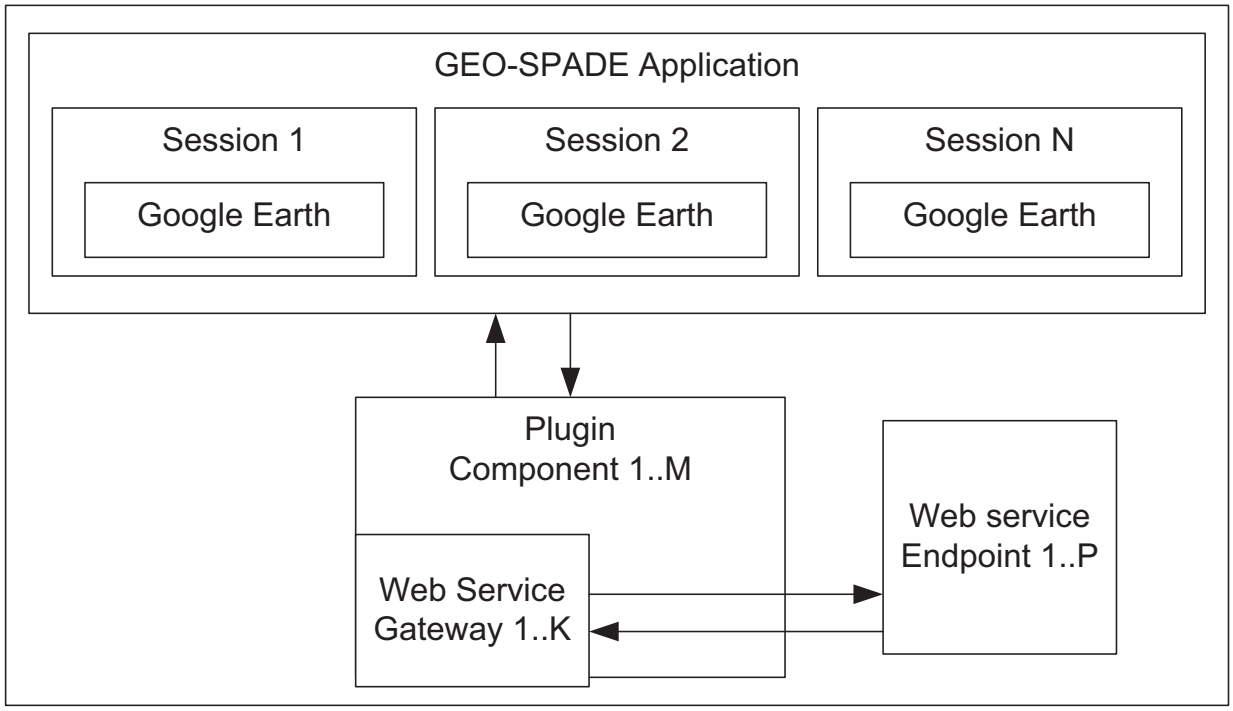


Figure 7.2: GEO-SPADE architecture

7.2 Case studies

In this section, we present two case studies in which GEO-SPADE is used as an analytical, exploratory and decision support tool. In the first case study presented in Section 7.2.1, GEO-SPADE is used as analytical tool for analyzing tourist activity using geo-tagged photos. The method itself was described in detail in Section XXX. We extend the case study by discussing the details of data interchange and technologies used in the analysis. The second case study (Section 7.2.2) discusses the integration of GEO-SPADE in a region exploration scenario using geo-tagged photos sorted according to various criteria.

Before moving to a specific case study, we would like to draw the reader's attention to Table 7.1. It shows the technologies and tools that were used in each case study. It can be seen

Table 7.1: Server-side technologies and tools

Technologies	Tourist activity Section 7.2.1	Region exploration Section 7.2.2
Communication	Java API for RESTful Web Services Windows Communication Foundation	Java API for RESTful Web Services
Database	PostGis	MySql
Data Interchange	KML Class Serialization	KML User-defined
Additional tools	JfreeChart Teiresias [Rigoutsos and Floratos, 1998] DBSCAN [Ester et al., 1996]	

that KML is shared between all case studies because it is a format for geographical visualization used by Google Earth. Java API for RESTful Web Services[1] is also used in the three case studies because Java is one of the most popular, general-purpose languages and the language of choice of many Web programmers. However, Windows Communication Foundation (WCF)[2] was used in the first case study along with Java to simplify complex data interchanges between the client-side (GEO-SPADE) written in c# and the server-side. Specifically, class serialization was used to transmit complex data structure, and rebuild the native object. Likewise, different databases were used. PostGis[3] was used in the first case study because of its support for spatial queries and free availability. MySql[4] was used in the second case study because of its ease of maintenance. Thus, the loosely coupled architecture gives a developer freedom in choosing whatever technology to use to accomplish a specific task.

7.2.1 Analysis of tourist activity

We outlined six major steps for completing the task of finding travel sequences in an arbitrary region of the world. These steps actively involve the analyst in the process of finding sequence patterns by reviewing the results of every step in the process, changing the parameters and activating the new steps using results from the previous step.

In the first step, presented in Figure 7.3, the analyst selects the desired area. The visible frame constitutes the boundaries of a region near Funchal, Madeira. The analyst can check the photographic activity in the selected region. Figure 7.4 shows time-series graph of the number of people who took photos in the selected area from January 2008 to September 2009 aggregated by month. The server receives the boundary of the region and queries the database counting the number of people per month. We used JFreeChart[5] to generate the image of the graph. The

[1] http://jcp.org/aboutJava/communityprocess/mrel/jsr311/index.html
[2] http://msdn.microsoft.com/en-us/netframework/aa663324.aspx
[3] http://postgis.refractions.net/
[4] http://www.mysql.com/
[5] http://www.jfree.org/jfreechart/

image is stored in the temporal directory on the server, while the KML, which is sent to the client includes the URL to the graph. The example of the KML is presented in Listing 7.1.

Listing 7.1: Server generated KML that shows photographic activity in the region. The activity graph is stored on the server

```
<Placemark>
<name>Funchal (regional events)</name>
 <description><![CDATA[<table border="1">
  <tr><td>south bound: 32.6445708362354</td></tr>
  <tr><td>north bound: 32.6594099577288</td></tr>
  <tr><td>west bound: -16.922443384542</td></tr>
  <tr><td>east bound: -16.891952135182</td></tr>
  <tr><td>lower time bound: 2008-01-01 00:00:00</td></tr>
  <tr><td>upper time bound: 2009-08-31 00:00:00</td></tr>
  <tr><td>
  <img src="http://192.168.2.100/t1/monthlyEvents.png">
  </td></tr>
  </table>]]>
 </description>
</Placemark>
```

The process of finding travel sequences is divided into two parts as described in Section 5.1. In the first part, every photo location is matched against a database of points of interest (we used the Wikipedia database[6]) and the closest POI is assigned to the photo. This creates clusters in which every photo is assigned to existing points of interest (POIs). In the second part, the remaining unassigned photos are clustered using DBSCAN [Ester et al., 1996], a density-based clustering algorithm. The results of this part are presented in Figure 7.5. The left part of the figure shows clusters in which photos were assigned to known POIs (assigned clusters in our terms), while the right part of the figure presents clusters of photos that were not assigned to a POI (unassigned clusters). Listing 7.2 shows part of the KML generated by the server that describes information about unassigned clusters including its symbolic name that begins with "[uc]", identifier number (id), information regarding the number of photos, the number of people in the cluster, and cluster boundaries.

Listing 7.2: Server generated KML that describes information about unassigned clusters

```
<Placemark>
 <name>[uc] unassigned cluster id: 23</name>
 <description>
  <![CDATA[<br>owners: 29<br>photos: 32]]>
 </description>
 <Polygon>
  <outerBoundaryIs>
   <LinearRing>
    <coordinates>
```

[6] http://toolserver.org/~kolossos/wp-world/pg-dumps/

```
        -16.926498,32.639085,0  -16.928043,32.640748,0
        -16.925811,32.642844,0  -16.922807,32.643494,0
        -16.922003,32.642939,0  -16.920404,32.640675,0
        -16.921896,32.639835,0  -16.926498,32.639085,0
      </coordinates>
    </LinearRing>
  </outerBoundaryIs>
 </Polygon>
</Placemark>
```

Next, the analyst visually inspects the created clusters and performs the clustering again if needed (for example, when the size of clusters should be changed). She may remove some clusters that are irrelevant or unimportant using either the statistical information of the cluster (number of photos and people in a cluster) or her background knowledge of the area. Figure 7.6 summarizes this step by presenting a view in which the analyst can select unassigned clusters and artificially create a new POI identification for every selected cluster, either by giving a symbolic identifier or some meaningful name based on the knowledge of the area. Since Google Earth may contain different objects in a view, the system iterates over the objects and selects only those whose names begin with the "[uc]" identifier (see Listing 7.2). The identifiers of unassigned clusters that the analyst decides to add to the list of clusters that are important for the analysis, are sent to the server, which, in turn, queries the database, matches the received identifiers to the ids stored in the database, adds these clusters to the list of assigned clusters and assigns symbolic names.

In the final step, the analyst generates sequence patterns from the clusters that were assigned to existing and artificial POIs. Figure 7.7 shows a form in which the analyst can select such parameters as: database properties and the length of generated patterns. We used the Teiresias algorithm [Rigoutsos and Floratos, 1998] for generating sequence patterns. Since the information about the generated sequences is not used directly for visualization, the server does not send the data in the KML format. Instead, we used a class serialization mechanism available in WCF to transfer the complex data structure to the client. The data includes the frequency of the sequence, the identifiers of the clusters in the sequence, the names of points of interest that constitute the travel sequence, the coordinates and centroids of every cluster in a sequence. The inspection of the travel sequences can be performed without referring to the exact location of clusters that are part of the sequence if the names of clusters belong to some existing POIs. However, when the sequence contains a cluster with a generated symbolic name, the analyst has to see the position of the cluster on the map. This is demonstrated by the following case.

The most frequent pattern generated is *Santa Lucia* $\Rightarrow$ *newpoi-3*. While Santa Lucia is a parish in the district of Funchal and may be known to the domain expert, *newpoi-3* is a name assigned by the system to an area in which the Wikipedia database does not contain a known POI. Clearly, this area should be located to give the analyst a hint about the sequence pattern. Sequences can be highlighted by clicking on the sequence pattern. Placemarks are added to the centers of the areas that are part of the selected sequence and the number assigned to them highlights the relative order of the area in the sequence. In the case of *Santa Lucia* $\Rightarrow$ *newpoi-3*, Santa Lucia will by highlighted with the number one, while *newpoi-3* receives the number two (Figure 7.8).

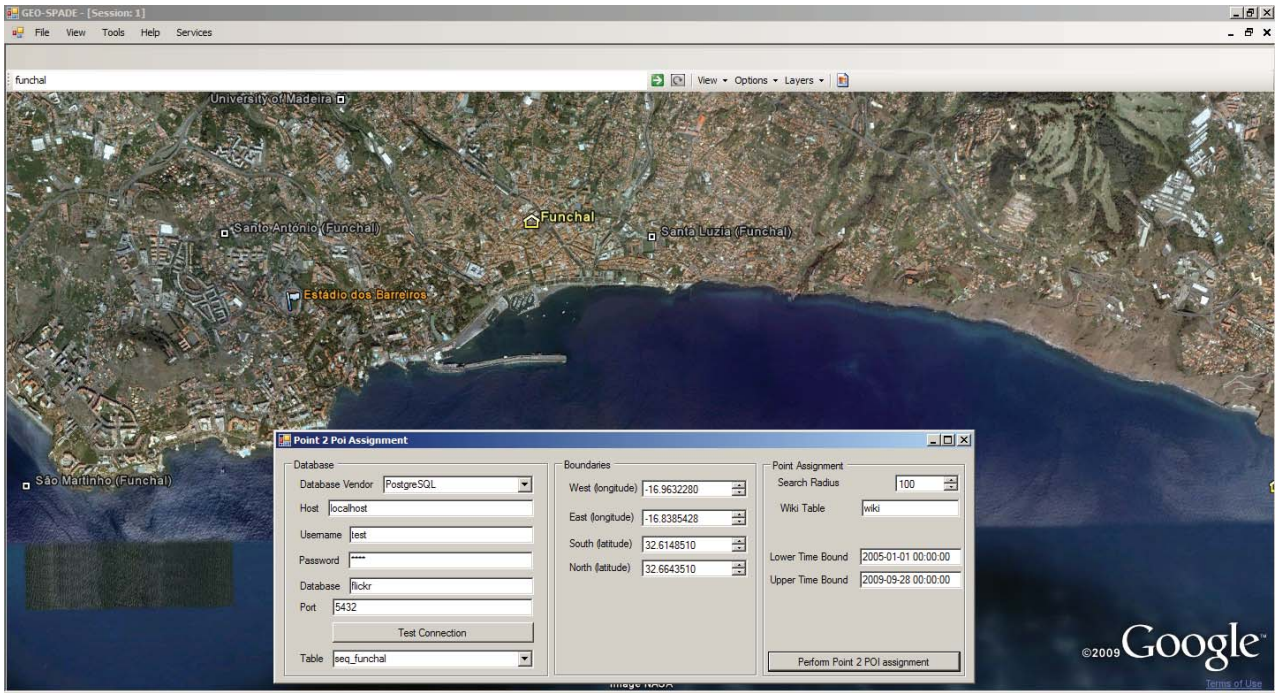


Figure 7.3: Selection of the area of interest

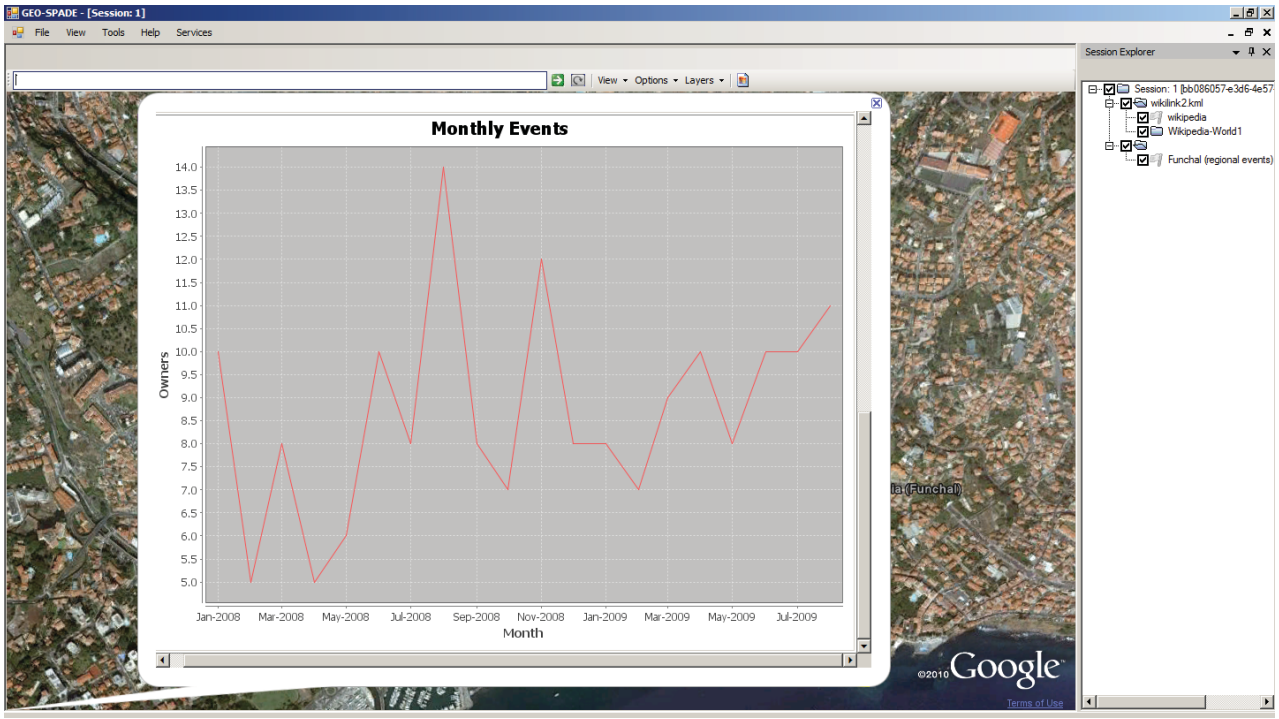


Figure 7.4: Monthly photographic activity

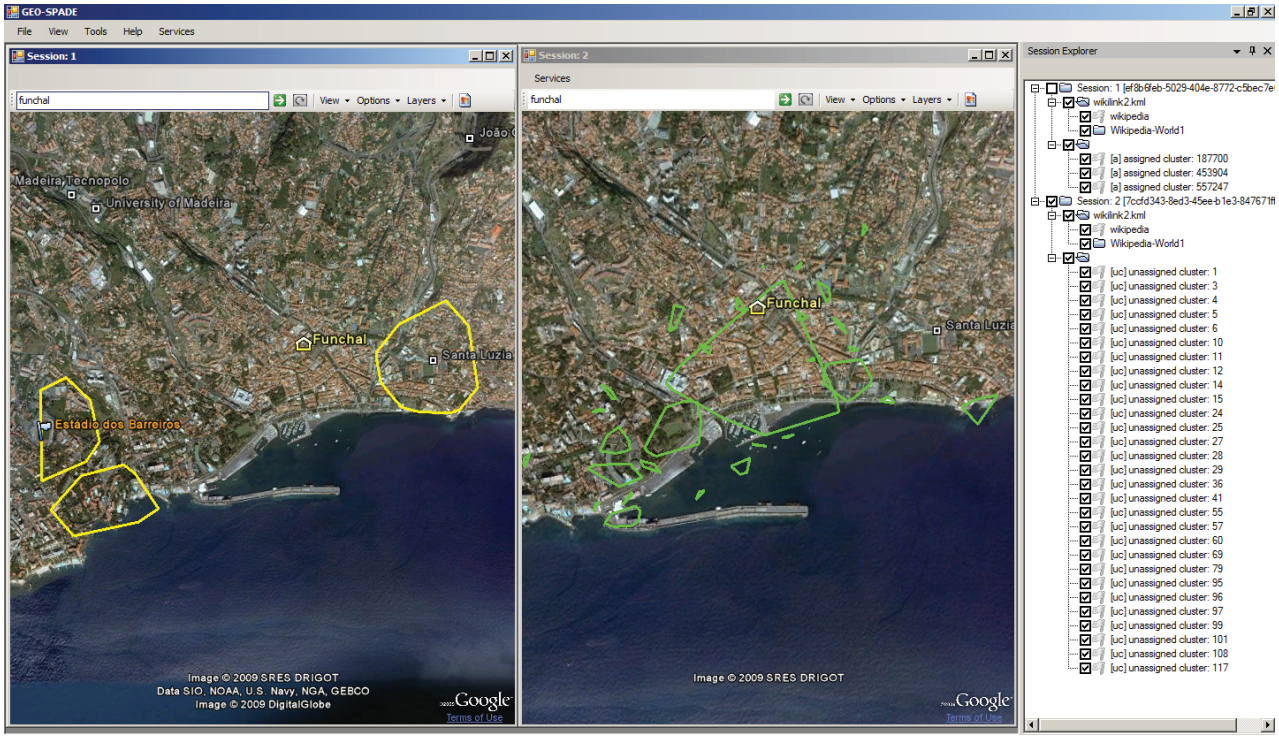


Figure 7.5: Multiple views of regions assigned to some existing POI (left) and regions where no POI was found (right)

7.2.2 Region exploration using geo-tagged photos

In Section 6, we presented a method for analyzing photo comments in terms of user opinions and sentiments that are present in comments. When the text parts that describe opinions (attitude towards the quality of a photo) and/or sentiments (attitudes towards the objects depicted on a photo or mood expressions) are located in the text, the strength of the opinion and/or sentiments can be calculated and mapped to a continuous numerical scale. This allows searching for photos according to the opinion or sentiment scores. We demonstrate how GEO-SPADE can be used by the user to perform the task of the area exploration using opinion and sentiment scores and show some technical details of the implementation.

The exploration begins by navigating to an area of interest. The control panel shown in Figure 7.9 is the main control panel for filtering photos according to different criteria (including

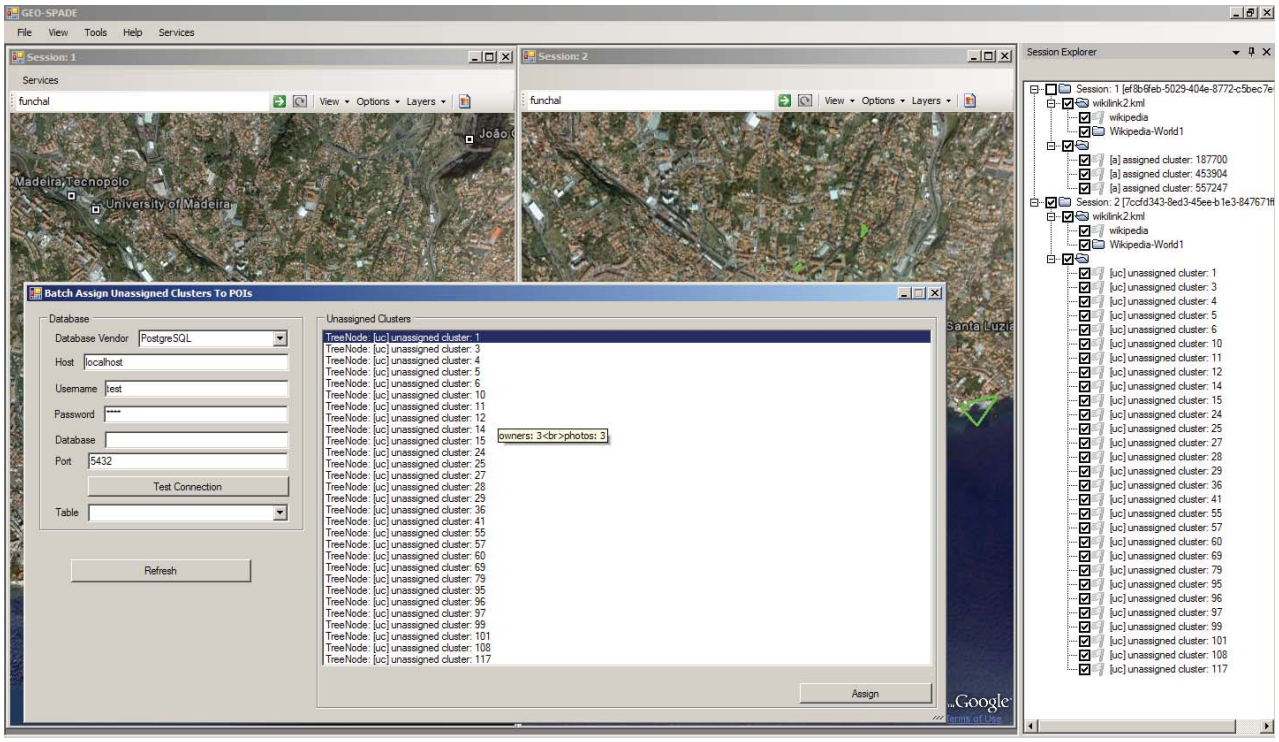


Figure 7.6: Selection of unassigned regions and creation of artificial POIs

opinion and sentiment scores). When the focus is on the control panel, the current visual boundaries of the map view are sent to the server. The server connects to the database and fetches the information about all photos found in the area. The response structure consists of key-value pairs separated by ";". An example of the response string is the following:
service=statistics;photos=501;minOpinion=-0.70;maxOpinion=21.30;....
The first key-value parameter denotes the type of the response in such a way that the response handler can delegate the response to the appropriate service handler. The service handler extracts key-value pairs using regular expressions. This allows extraction of the known parameters without breaking the code even if the new parameters were added to the response string without updating the client side. The general statistics are displayed on the control panel, which allows the user to see how many photos are found in the region, what are the minimum and maximum sentiments and opinion scores of these photos and other relevant information.

The next step is to retrieve the photos by applying one of the available filters (opinion, sentiment, etc.) and by restricting the number of photos on the map and on the control panel as well as by selecting the photos that were taken in a specific time period. The server's response is similar to the one described above. However, it contains two parts. The first part is intended for parsing by the control panel and includes the information for N selected photos to be displayed in the control panel and sorted according to the selected filtering criteria (opinion, sentiment). The second part is a KML string that will be delegated to the Google Earth engine. Figure 7.10 shows an example of the KML response that consists of two representation styles. The left side of Figure 7.10 shows the map view of a region of Warsaw, Poland with thumbnails of images taken in that area filtered by sentiment scores. The right side of Figure 7.10, shows the

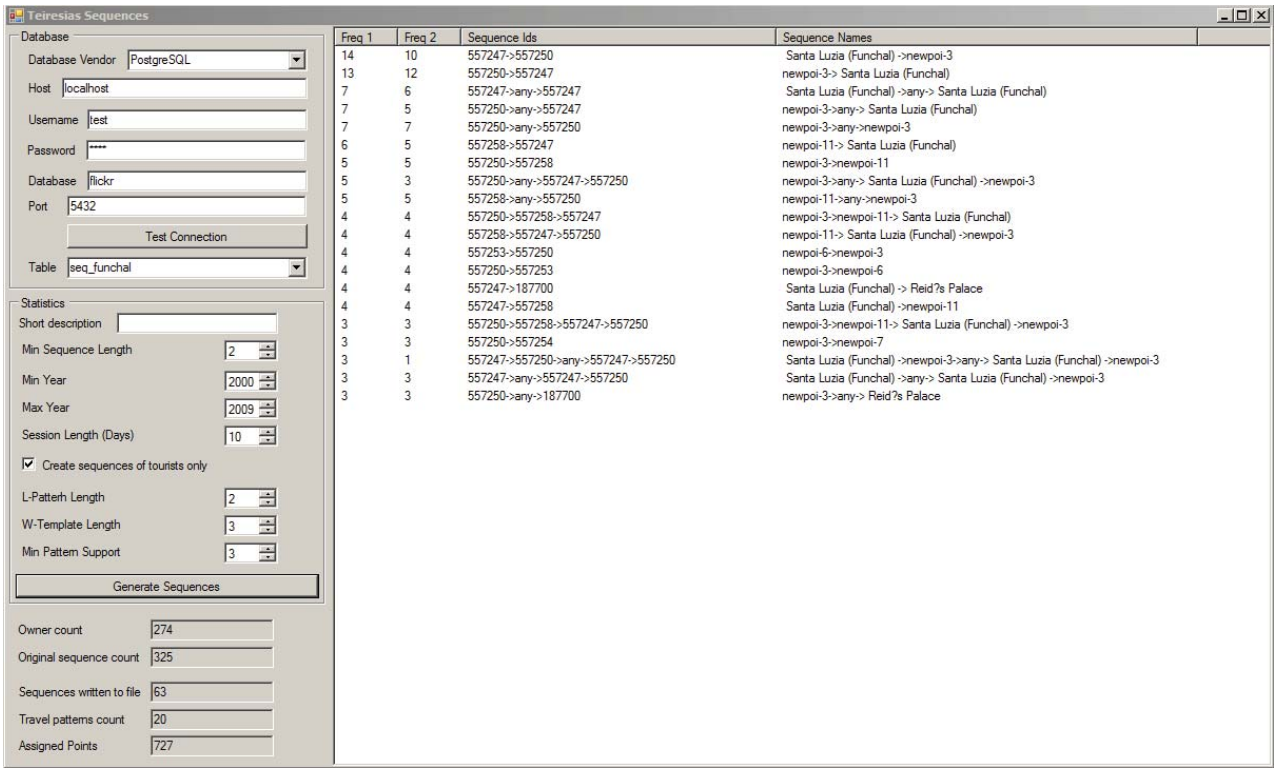


Figure 7.7: Sequence patterns

same map view but instead of image thumbnails, the sentiment scores are mapped to colors. Both representations allow the user to quickly explore the area either by observing the image thumbnails or by navigating to a place where higher scores are given to images, which can be seen by looking at the color of the circles. Both ways make it possible to see the image itself and to retrieve more information about it (opinion or sentiment scores, and comments) by clicking on the thumbnail.

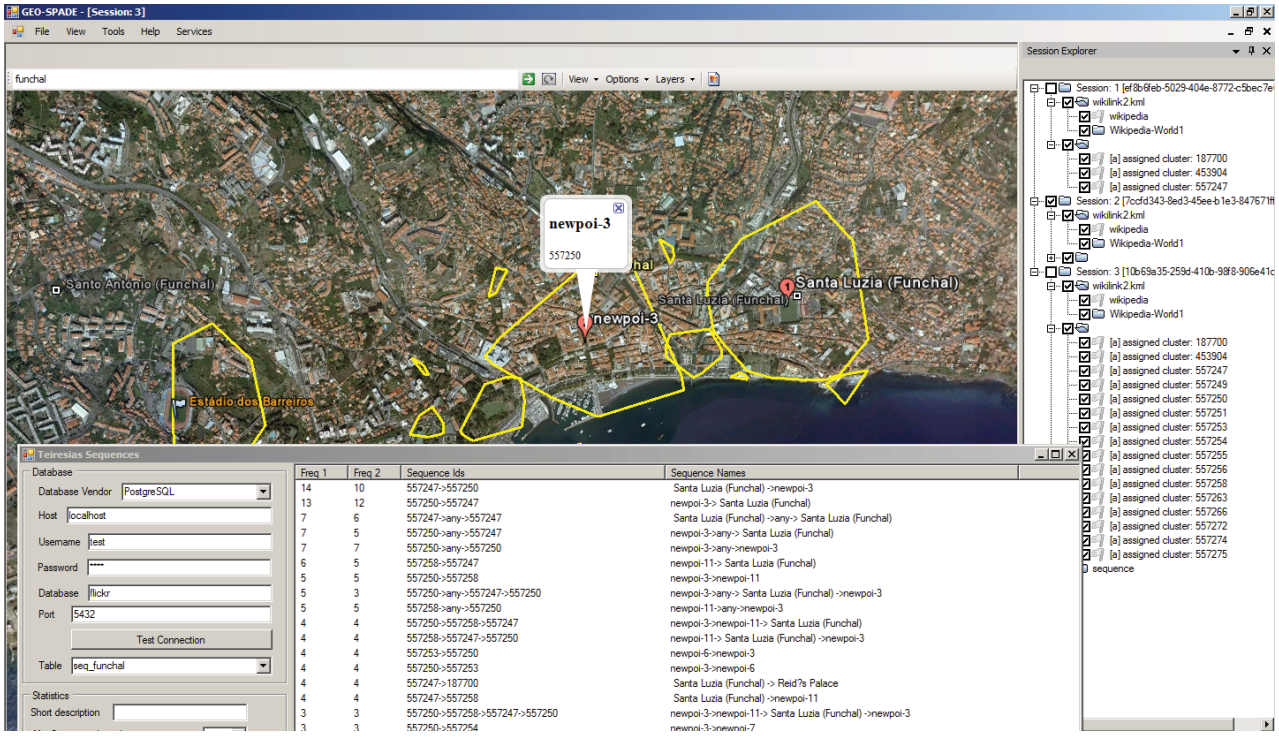


Figure 7.8: Combined view of obtained sequence patterns and the map. The regions are highlighted by clicking on the sequences

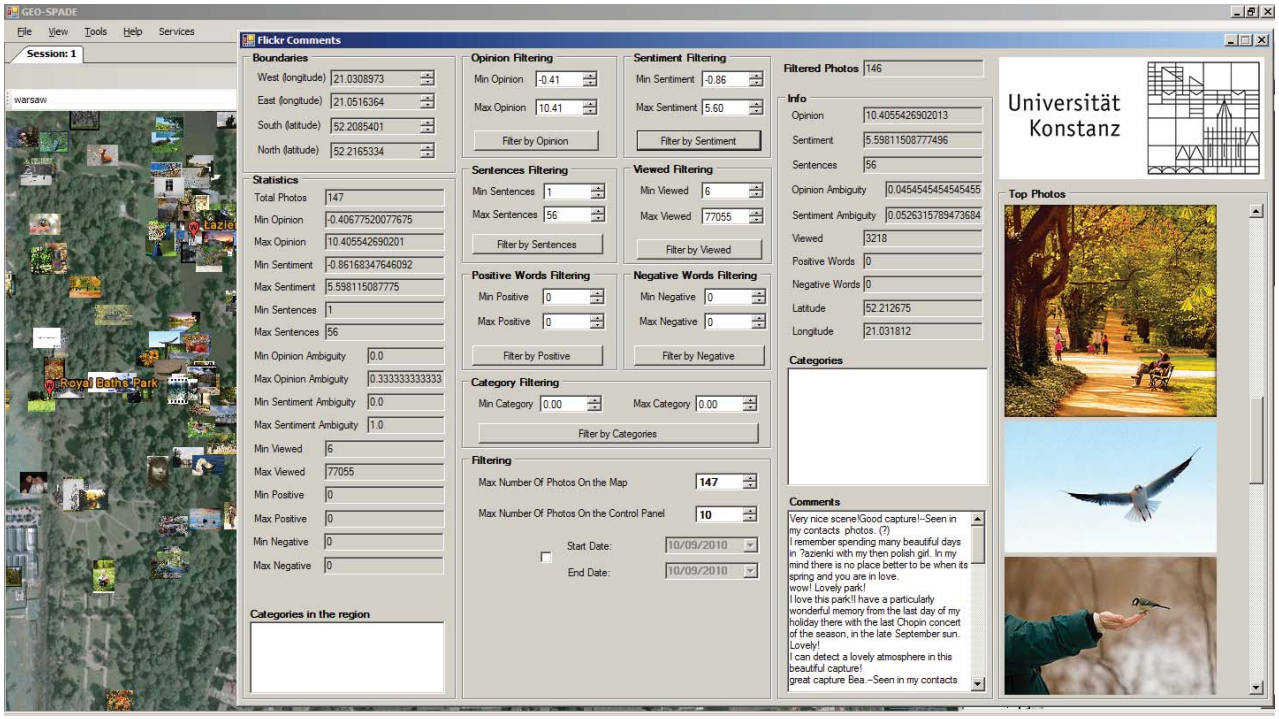


Figure 7.9: The control panel for retrieving images according to one of the sorting criteria. The top-N retrieved images are also presented in the control panel. Centering the map view around the top-N photos is implemented by double-clicking the photo

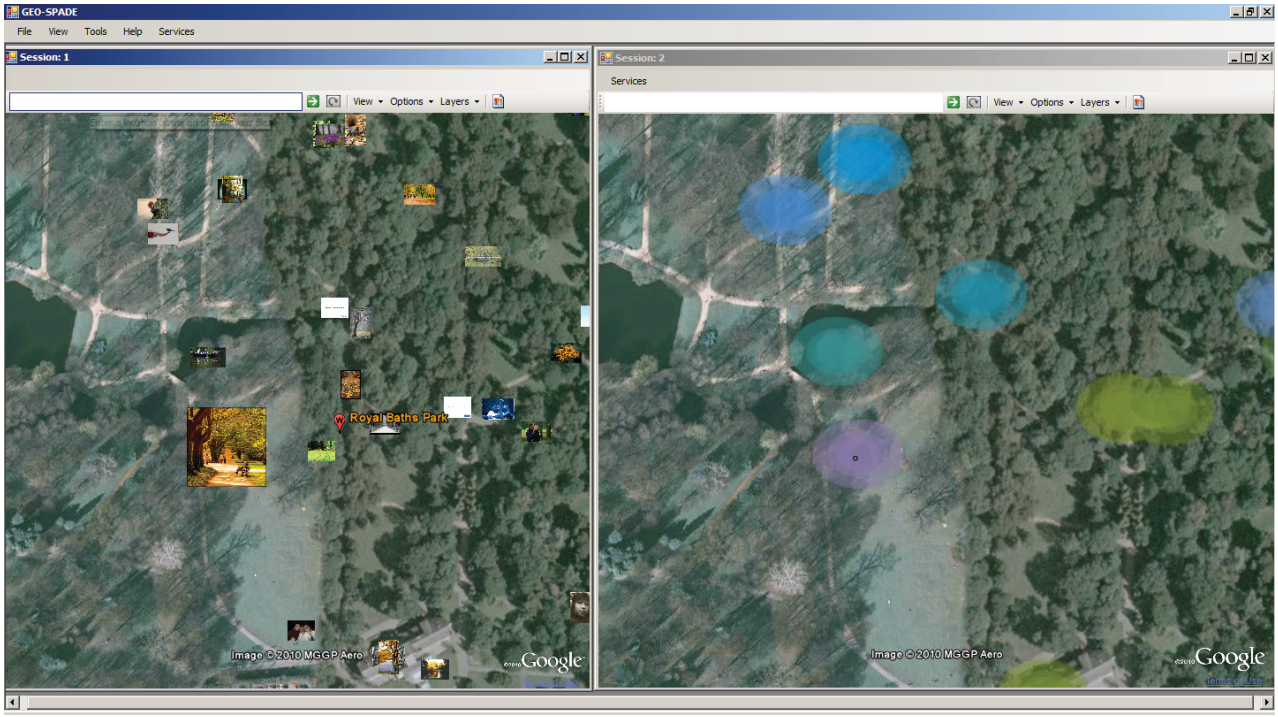


Figure 7.10: Two photo representation styles: image thumbnails (left) and color coding according to the image scores of the selected criteria (right)

8 Conclusions

The main goals of this thesis are to propose a systematic approach to the analysis of people's movement and events using geotagged photos and to extract knowledge from geotagged photos using approaches from different fields such as geovisual analytics and data mining (spatio-temporal and text mining). We presented several aspects of geotagged photo analysis in Chapter 1 such as geovisual analysis of people's movement, discovery of attractive places (event-based analysis), discovery of frequent travel sequential patterns (trajectory movement). In some tasks where the location of a taken photo was not enough for analysis we used additional information available for a photo such as comments written by users in order to estimate the importance of a particular photo. This led to a contribution on its own in the field of computational linguistics (Chapter 6). Finally, the analysis of spatio-temporal data is not possible without appropriate tools. Therefore, the development of appropriate tools to handle geotagged photos was also part of this research and became a contribution into the field of systems engineering (Chapter 7).

We emphasized that the analysis of geotagged photos can be performed in both ways: as an analysis of an event-based or as an analysis of trajectory movement, which requires different analytical approaches under an umbrella of movement analysis. For this, we put the analysis of geotagged photos in the broad context of spatio-temporal analysis of movement and presented a classification of spatio-temporal data types and tasks applicable to geotagged photos in the spatio-temporal context.

In Chapter 2 we outline several directions related to the research such as works that involve analysis of geotagged photos, opinion and sentiment analysis, and GIS frameworks. However, the most of Chapter 2 is devoted to outlining spatio-temporal clustering, which is the basic approach in dealing with spatio-temporal and movement data.

Chapters 3-7 is the core of this thesis. The conclusions for each chapter are given below.

8.1 Discovering movement patterns: A geovisual analytics approach (Chapter 3)

The goals of this study were: (1) to explore the potential of volunteered geographic information, using the example of Flickr posted photos, for providing information about people's activities in space and time, and (2) to experiment with geovisual analytics techniques for extracting this information. In respect to the first goal, social networking websites such as Flickr and Panoramio, hosting databases of georeferenced photos, offer information on the:

- Spatiality of people's interests; locations of landmarks and events that are of interest to photographers.
- Temporality of people's interests; dates of photographing places and events and the seasonality of people's interests.
- Spatial extent of people's interests; boundaries of areas and events represented on photographs.
- Connectivity between photographed places represented by a network of moves connecting places of interest.
- Travel patterns of photographers and their temporal characteristics.

Similar information can be found in the databases of other social networking services storing spatial and temporal references of information created by their user, such as for example the georeferenced Twitter messages. Users of these services act as voluntary or sometimes involuntary sensors, collecting potentially useful geographic information [Goodchild, 2007]. At the same time, it is important to be aware of the limitations of volunteered geographic information including spatial and temporal coverage as well as demographic and social representativeness. The question of how closely do landmark preferences and travel itineraries of social media users represent the preferences and itineraries of other groups, who do not use social media to communicate their preferences, remains unanswered.

In respect to the second goal, the techniques of geovisual analytics used in this study proved to be effective in data aggregation and search for spatio-temporal patterns. There were two time-consuming tasks in the analysis of places involving: (1) search for interesting temporal patterns of place visits, and (2) acquisition of additional information for interpreting the detected patterns (interpreting specific locations within their larger geographic context, retrieving and reading the titles of the photos, and searching the Web for information related to locations of interest). These tasks require an improvement in analytical support. More specifically, statistical techniques can help in detecting particular types of patterns such as sudden peak and periodic variation. The interpretation of patterns could be supported in future studies by:

- Automated comparison of geotagged photo locations with locations of known objects (e.g. landmarks) stored in a geodatabase.
- Automated text analysis of photo titles and supplementary comments to extract place names and information identifying photographed events. Keywords extracted from photo titles and accompanying time references can be passed to a Web search machine for retrieving relevant information from the Web.

In the analysis of movement (Section 3.4), the main problem of visual display clutter was overcome by interactive dynamic filtering, which enabled the visualization of filtered data. Further automation could be achieved by applying data mining techniques for extracting frequent item combinations and frequent item sequences (see Chapter 5). In this case, items are the places occurring in the trajectories.

Further on, the investigation of the temporal variation of movement trajectories (Section 3.4.2) could be supported by two-way clustering similar to the approach suggested by Andrienko et al. [2010] for data aggregated by spatial compartments. In this approach, compartments are grouped according to the temporal variation of the respective attribute values (e.g. frequency of photographs). Complementary to this, time intervals are grouped according to the respective spatial distributions of the attribute values. The same idea can be applied to aggregated movement data by taking aggregate moves between places instead of spatial compartments. The time-variant counts of people are the attribute of the moves used for the clustering.

8.2 Discovering attractive places (Chapter 4):

In this chapter, we presented two approaches for analysis of places of interest, such as for example tourist attractions, using geotagged photos as the primary data source. In the first, density-based clustering, approach, places of potential interest are characterized by a high concentration of photos and photographers. We introduced the notion of photo importance, which measures the degree of influence on the given photo by the neighboring photos taken by different photographers. Therefore, the interestingness of a place is characterized by the degree of importance of photos taken in that place expressed by photo's weight.

We presented a DBSCAN-based clustering algorithm that combines the clustering and weight calculation into one process. We also presented the overall runtime complexity by evaluating the algorithm for three urban areas in the World. We showed that the runtime complexity depends on a number of reasons. The most influential factors, however, are the clustering parameters, the size of the area, and the number of photos and owners. The results of the evaluation showed that parallel execution can significantly reduce the runtime complexity. Other methods for improving the runtime complexity such as sampling were mentioned.

The second approach is based on the individual interestingness of a photo, which is determined by the analysis of comments. We showed that two types of interestingness can be derived from user comments: one, which is based on opinions concerning the quality of photos and the other, which is based on sentiments expressed towards objects depicted on photos.

In both cases, an individual photo is assigned a weight expressing its interestingness. These weights are mapped to a color scale and a heatmap is generated reflecting the places of potential interest with distinguishable colors. The photo weights are depicted on a graduated circle, in which the circle radius in inversely proportional to the map scale. This approach makes it possible to use the same weights in interactive explorations by zooming in to a street level or zooming out to a city or country scale without the need to recalculate the weights. After the user has zoomed to the street level, the colored circles representing the place where a photo was taken are redisplayed with a proportionally smaller radius. Then, when the user zooms out, the circle sizes are increased proportionally to the zoom level.

Both types of analysis can be used in different scenarios. The density-based approach is appropriate when the user is interested in exploring known, highly visited touristic places, such as city capitals or widely known landmarks. When the user is interested in the exploration of places with rare touristic activity, the second approach, which is based on the individual ranking of photos, is more suitable.

We extended the density-based clustering by introducing our algorithm termed P-DBSCAN,

which is based on DBSCAN. We introduced two improvements to the original definition of DBSCAN. (1) We defined neighborhood density as the number of people who take photos in the area. (2) We proposed a notion of adaptive density for optimizing search for dense areas and faster convergence of the algorithm towards clusters with high density. This resulted in a creation of larger number of small-size clusters that contain as few as possible different points of interest in a cluster. The experiments on several highly visited places in the World showed improvement by about 80% compared to the static version of P-DBSCAN without adaptive density (a comparison to the original DBSCAN would have yielded even better results and therefore were omitted from the experiment). We also showed that it is possible to eliminate the *MinPts* parameter and still get meaningful clusters using only one parameter of neighborhood radius.

8.3 Discovering frequent travel sequential patterns (Chapter 5):

The goal of the study was to suggest an automatic approach for mining semantically annotated travel sequences using geotagged photos by searching for sequence patterns of any length. The sequences obtained may contain patterns that are not necessarily the immediate antecedents. Moreover, the approach that we proposed can examine sequences in which the same pattern is repeated more than once in the same sequence. We showed that the method is capable of mining semantically annotated sequences of any length with patterns that are not necessarily immediate antecedents. We demonstrated the feasibility of our approach on two different cities using real data. We showed that the approach could be applied to different spatial scales - to places that have a great number of visitors and points of interest, and to locations that have relatively few visitors and points of interest.

8.4 Opinion and sentiment analysis of photo comments (Chapter 6):

This chapter introduced a practical unsupervised approach to the analysis of opinions in photo comments. Our approach is capable of identifying two types of opinions from the comments: opinions that are related to the quality of the photo and general sentiments or moods expressed towards the objects shown on the photo. Unlike most of the existing approaches in which binary (negative or positive) opinion orientation is used, we model opinion orientation using a real-valued scale.

Using linguistic features, we built a finite lexicon of adjectives and calculate their opinion strength using a word importance paradigm borrowed from the information retrieval field combined with the concepts of Zipf's Least Effort and regularity in word usage, and semantic differentiation of Osgood. The opinion orientation (negative or positive sign) is calculated using a predefined lexicon of positive and negative opinion-bearing words. The identification and separation of photo opinions is based on a semi-automatic method for photo feature extraction and a set of predefined syntactic opinion reference patterns. We applied the cumulative sum to calculate the overall opinion and sentiment scores of comments. This allows a dynamic update of scores if new comments are added to the photo. However, other strategies for overall opinion and sentiment scores can be easily applied.

We conducted a user study in which we analyzed factors that influence the human evaluator during the ranking of photo comments. Our study included 49 participants, who evaluated photo comments from five different regions across Central Europe on a predefined criterion (opinions or sentiments). The results of the user study showed that there is a high variability in agreements between participants themselves, and between the algorithm and the users. However, there was no statistical significance to the difference between the algorithm and the participants, which allows us to conclude that the performance of our algorithm is comparable to the performance of the average user.

The approach is potentially useful in other domains where different kinds of opinions have to be separated. One popular example are movie reviews where one aspect in comments is the plot of the movie and another aspect is the opinion about the movie. For example, a character might be identified as being evil, while the actor does a good job of embodying it. Hence, opinion words relating to the plot should be separated from user opinions. The presented approach to opinion and sentiment analysis was used in Chapter 4 to discover attractive places where people take photos.

8.5 A Google Earth-based GIS (Chapter 7):

This chapter focused on real-life scenarios of spatial analysis and exploration. Specifically, we presented two case studies in which our framework, called GEO-SPADE, is successfully evaluated on problems such as analysis of tourist activity and discovery of interesting photo locations using geotagged photo data. This is achieved by the architecture that is based on a thin client paradigm, pluggable components and SOA-based architecture in which the core of the functionality is developed separately in any programming language and run on the server, while the results of the computation are transferred using Web services. The results that are targeted for direct visualization are communicated in the KML format and delivered to the Google Earth engine. The results that are required for further processing are communicated to the client-side plug-in component in any format suitable for the developer. GEO-SPADE was extensively used on different stages of this thesis contributing to writing of several papers such as [Kisilevich et al., 2010a,c,b,d]

8.6 Future work

While we tried to cover many possible directions of the analysis of geotagged photos in the context of event-based and trajectory movement and to make our work systematic in this respect we could not cover all possible directions which leaves plenty of room to continue research on user generated spatio-temporal data. We can list several prominent directions that were not covered in the thesis in the context of movement analysis: travel recommender systems or travel guides and semantic analysis. The former direction has many commonalities with our approach of finding frequent sequential patterns of people's movement described in Chapter 5 but it also requires a recommender engine to learn not only where people move but also what they like, which usually requires the analysis of multimedia content to understand the preferences of the photographer in order to create his/her activity profile. This leads to a latter direction that we mentioned, namely

semantic analysis. In our paper titled "Towards acquisition of semantics of places and events by multi-perspective analysis of geotagged photo collections" [Kisilevich et al., 2011] presented at GeoCart 2010 conference in Auckland, New-Zealand we developed a conceptual framework and proposed a methodology to acquire semantical information from geotagged photos for analyzing events and places using spatio-temporal clustering combined with time-series analysis of events and text clustering. As a result, we succeeded in finding and explaining the events occurred in a selected region without any prior background knowledge. However, further research in this direction is required.

Bibliography

Adrienko, N. and Adrienko, G. (2011). Spatial generalization and aggregation of massive movement data. *Visualization and Computer Graphics, IEEE Transactions on*, 17(2):205–219. 63

Agrawal, R., Faloutsos, C., and Swami, A. N. (1993). Efficient Similarity Search In Sequence Databases. In Lomet, D., editor, *Proceedings of the 4th International Conference of Foundations of Data Organization and Algorithms (FODO)*, pages 69–84, Chicago, Illinois. Springer Verlag. 50

Agrawal, R. and Srikant, R. (1994). Mining sequential patterns. In *Proceedings of the 11th International Conference on Data Engineering (ICDE)*, pages 3–14. IEEE. 54

Agrawal, R., Srikant, R., et al. (1994). Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB)*, volume 1215, pages 487–499. 54

Ahern, S., Naaman, M., Nair, R., and Yang, J. (2007). World explorer: visualizing aggregate data from unstructured text in geo-referenced collections. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, page 10. 47, 48, 78

Ali, T., Asghar, S., and Sajid, N. (2010). Critical analysis of dbscan variations. In *Information and Emerging Technologies (ICIET), 2010 International Conference on*, pages 1–6. IEEE. 51

Alon, J., Sclaroff, S., Kollios, G., and Pavlovic, V. (2003). Discovering clusters in motion time-series data. In *CVPR (1)*, pages 375–381. 49

Alvares, L., Bogorny, V., de Macedo, J., Moelans, B., and Spaccapietra, S. (2007a). Dynamic modeling of trajectory patterns using data mining and reverse engineering. In *Proceedings of the 26th international conference on conceptual modeling*, volume 83, pages 149–154. 55

Alvares, L. O., Bogorny, V., Kuijpers, B., de Macedo, J. A. F., Moelans, B., and Vaisman, A. (2007b). A model for enriching trajectories with semantic geographical information. In *GIS '07: Proceedings of the 15th annual ACM international symposium on Advances in geographic information systems*, pages 1–8. 53, 55

Andrienko, G. and Andrienko, N. (2008). Spatio-temporal aggregation for visual analysis of movements. In *Proceedings of IEEE Symposium on Visual Analytics Science and Technology (VAST 2008), IEEE Computer Society Press*, pages 51–58. 38, 52

Andrienko, G. and Andrienko, N. (2009). Interactive cluster analysis of diverse types of spatiotemporal data. *ACM SIGKDD Explorations*. 52

Andrienko, G., Andrienko, N., Bremm, S., Schreck, T., Von Landesberger, T., Bak, P., and Keim, D. (2010). Space-in-time and time-in-space self-organizing maps for exploring spatiotemporal patterns. In *Computer Graphics Forum*, volume 29, pages 913–922. Wiley Online Library. 157

Andrienko, G., Andrienko, N., Dykes, J., Fabrikant, S., and Wachowicz, M. (2008). Geovisualization of dynamics, movement and change: key issues and developing approaches in visualization research. *Information Visualization*, 7(3):173–180. 34

Andrienko, G., Andrienko, N., Jankowski, P., Keim, D., Kraak, M., MacEachren, A., and Wrobel, S. (2007a). Geovisual analytics for spatial decision support: Setting the research agenda. *International Journal of Geographical Information Science*, 21(8):839–857. 27, 124

Andrienko, G., Andrienko, N., Rinzivillo, S., Nanni, M., Pedreschi, D., and Giannotti, F. (2009). Interactive Visual Clustering of Large Collections of Trajectories. *VAST 2009*. 52

Andrienko, G., Andrienko, N., and Wrobel, S. (2007b). Visual analytics tools for analysis of movement data. *SIGKDD Explorations Newsletter*, 9(2):38–46. 52

Andrienko, N. and Andrienko, G. (2006). *Exploratory analysis of spatial and temporal data: a systematic approach*. Springer Verlag. 52

Ankerst, M., Breunig, M. M., Kriegel, H.-P., and Sander, J. (1999). Optics: ordering points to identify the clustering structure. *SIGMOD Rec.*, 28(2):49–60. 51, 52, 97

Argamon, S., Bloom, K., Esuli, A., and Sebastiani, F. (2009). Automatically determining attitude type and force for sentiment analysis. *Human Language Technology. Challenges of the Information Society*, pages 218–231. 32

Becker, H., Naaman, M., and Gravano, L. (2009). Event identification in social media. In *Proceedings of the ACM SIGMOD Workshop on the Web and Databases (WebDB 09)*. 48

Beckmann, N., Kriegel, H., Schneider, R., and Seeger, B. (1990). The R*-tree: an efficient and robust access method for points and rectangles. *ACM SIGMOD Record*, 19(2):322–331. 82

Berndt, D. J. and Clifford, J. (1996). Finding patterns in time series: a dynamic programming approach. *Advances in knowledge discovery and data mining*, pages 229–248. 50

Brecheisen, S., Kriegel, H., and Pfeifle, M. (2006). Multi-step density-based clustering. *Knowledge and Information Systems*, 9(3):284–308. 51

Chainey, S. and Tompson, L. (2008). *Crime Mapping Case Studies. Practice and Research*. Wiley. 48

Chan, K.-P. and chee Fu, A. W. (1999). Efficient time series matching by wavelets. In *In ICDE*, pages 126–133. 50

Chen, L., Özsu, M. T., and Oria, V. (2005). Robust and fast similarity search for moving object trajectories. In *SIGMOD '05: Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 491–502, New York, NY, USA. ACM. 50

Chen, M., Gao, X., and Li, H. (2010). Parallel dbscan with priority r-tree. In *Information Management and Engineering (ICIME), 2010 The 2nd IEEE International Conference on*, pages 508–511. IEEE. 51

Chen, X., Liu, W., Qiu, H., and Lai, J. (2011). Apscan: A parameter free algorithm for clustering. *Pattern Recognition Letters*. 51

Chesley, P., Vincent, B., Xu, L., and Srihari, R. (2006). Using verbs and adjectives to automatically classify blog sentiment. *Training*, 580(263):233. 32, 57, 128

Chudova, D., Gaffney, S., Mjolsness, E., and Smyth, P. (2003). Translation-invariant mixture models for curve clustering. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 79–88, New York, NY, USA. ACM. 49

Ciaccia, P., Patella, M., and Zezula, P. (1997). M-tree: An efficient access method for similarity search in metric spaces. In Jarke, M., Carey, M., Dittrich, K. R., Lochovsky, F., Loucopoulos, P., and Jeusfeld, M. A., editors, *Proceedings of the 23rd International Conference on Very Large Data Bases (VLDB'97)*, pages 426–435, Athens, Greece. Morgan Kaufmann Publishers, Inc. 51

Compieta, P., Di Martino, S., Bertolotto, M., Ferrucci, F., and Kechadi, T. (2007). Exploratory spatio-temporal data mining and visualization. *Journal of Visual Languages and Computing*, 18(3):255–279. 34, 58, 59

Crandall, D., Backstrom, L., Huttenlocher, D., and Kleinberg, J. (2009). Mapping the world's photos. In *Proceedings of the 18th international conference on World wide web*, pages 761–770. 47

Das, S. and Chen, M. (2007). Yahoo! for Amazon: Sentiment extraction from small talk on the web. *Management Science*, 53(9):1375–1388. 32, 57

Dave, K., Lawrence, S., and Pennock, D. (2003). Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, page 528. 32

de Oliveira, D., Garrett Jr, J., and Soibelman, L. (2010). A density-based spatial clustering approach for defining local indicators of drinking water distribution pipe breakage. *Advanced Engineering Informatics*. 51

Deng, D., Chuang, T., and Lemmens, R. (2009). Conceptualization of place via spatial clustering and co-occurrence analysis. In *Proceedings of the 2009 international Workshop on Location Based Social Networks*, pages 49–56. ACM. 51

Diaz, L., Granel, C., Gould, M., and Olaya, V. (2008). An open service network for geospatial data processing. In *An Open Service Network for Geospatial Data Processing: Free and Open Source Software for Geospatial (FOSS4G) Conference*. 33

Ding, X., Liu, B., and Yu, P. (2008). A holistic lexicon-based approach to opinion mining. In *Proceedings of the international conference on Web search and web data mining*, pages 231–240. 58

Drake, A., Ringger, E., and Ventura, D. (2008). Sentiment Regression: Using Real-Valued Scores to Summarize Overall Document Sentiment. In *2008 IEEE International Conference on Semantic Computing*, pages 152–157. 32

Duan, L., Xu, L., Guo, F., Lee, J., and Yan, B. (2007). A local-density based spatial clustering algorithm with noise. *Inf. Syst.*, 32(7):978–986. 51, 97

Dykes, J., MacEachren, A., and Kraak, M. (2005). *Exploring geovisualization*, volume 1. Pergamon. 52

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *Data Mining and Knowledge Discovery*, pages 226–231. 30, 36, 50, 51, 56, 79, 87, 99, 116, 120, 145, 146

Esuli, A. and Sebastiani, F. (2006). SentiWordNet: A publicly available lexical resource for opinion mining. In *Proceedings of LREC*, volume 6, pages 417–422. 57, 129

Fahrni, A. and Klenner, M. (2008). Old wine or warm beer: Target-specific sentiment analysis of adjectives. In *AISB 2008 Convention Communication, Interaction and Social Intelligence*, volume 1, page 60. 32

Farman, J. (2010). Mapping the digital empire: Google Earth and the process of postmodern cartography. *New Media & Society*, 12(6):869–888. 34

Fellbaum, C. (1998). *WordNet: An electronic lexical database*. MIT press Cambridge, MA. 57, 129

Ferraz, V. R. T. and Santos, M. T. P. (2010). Globeolap: Improving the geospatial realism in multidimensional analysis environment. In *12th International Conference on Enterprise Information Systems*, pages 99–107. 58, 59

Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures*. PhD thesis. Chair-Taylor, Richard N. 143

Fisher, D. (2007). Hotmap: Looking at geographic attention. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1184–1191. 48

Foerster, T., Schaeffer, B., Brauner, J., Jirka, S., and Muenster, G. (2009). Integrating ogc web processing services into geospatial mass-market applications. *International Conference on Advanced Geographic Information Systems & Web Services*, pages 99–103. 33, 58

Folino, G., Forestiero, A., and Spezzano, G. (2009). An adaptive flocking algorithm for performing approximate clustering. *Information Sciences*, 179(18):3059–3078. 51

Fosca, G. and Dino, P. (2008). *Mobility, Data Mining and Privacy: Geographic Knowledge Discovery*. Springer. 53

Frentzos, E., Gratsias, K., and Theodoridis, Y. (2007). Index-based most similar trajectory search. In *ICDE*, pages 816–825. 51

Friis-Christensen, A., Ostlander, N., Lutz, M., and Bernard, L. (2007). Designing service architectures for distributed geoprocessing: Challenges and future directions. *Transactions in GIS*, 11(6):799. 33

Gaffney, S. and Smyth, P. (1999). Trajectory clustering with mixtures of regression models. In *KDD '99: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 63–72, New York, NY, USA. ACM. 49

Gamon, M. (2004). Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In *Proceedings of the International Conference on Computational Linguistics (COLING)*. 32, 57

Gan, W. and Li, D. (2003). Optimal choice of parameters for a density-based clustering algorithm. In *Proceedings of the 9th international conference on Rough sets, fuzzy sets, data mining, and granular computing*, pages 603–606. 87

Giannotti, F., Nanni, M., and Pedreschi, D. (2006). Efficient mining of temporally annotated sequences. In *Proceedings of the 6th SIAM International Conference on Data Mining (SDM)*, pages 346–357. 54

Giannotti, F., Nanni, M., Pinelli, F., and Pedreschi, D. (2007). Trajectory pattern mining. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, page 339. ACM. 53, 54

Girardin, F., Calabrese, F., Fiore, F. D., Ratti, C., and Blat, J. (2008a). Digital footprinting: Uncovering tourists with user-generated content. *Pervasive Computing, IEEE*, 7(4):36–43. 48

Girardin, F., Fiore, F. D., Blat, J., and Ratti, C. (2007). Understanding of tourist dynamics from explicitly disclosed location information. *In 4th International Symposium on LBS and Telecartography, Hong-Kong, China*. 48

Girardin, F., Fiore, F. D., Ratti, C., and Blat, J. (2008b). Leveraging explicitly disclosed location information to understand tourist dynamics: a case study. *Location Based Services*, 2(1):41–56. 25, 48

Girardin, F., Vaccari, A., Gerber, A., and Ratti, C. (2009). Quantifying urban attractiveness from the distribution and density of digital footprints. *Journal of Spatial Data Infrastructure Research*, 4:175–200. 114

Goodchild, M. (2007). Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69(4):211–221. 25, 156

Goodchild, M. (2008). The use cases of digital earth. *International Journal of Digital Earth*, 1(1):31–42. 34

Graupmann, J. and Schenkel, R. (2006). GeoSphere-Search: Context-Aware Geographic Web Search. In *3rd Workshop on Geographic Information Retrieval*. 58

Grossner, K. E. (2006). Is google earth, "digital earth?" - defining a vision. *University Consortium of Geographic Information Science, Summer Assembly, Vancouver, WA*. 34

Gudmundsson, J. and van Kreveld, M. (2006). Computing longest duration flocks in trajectory data. In *GIS '06: Proceedings of the 14th annual ACM international symposium on Advances in geographic information systems*, pages 35–42, New York, NY, USA. ACM. 56

Guttman, A. (1984). R-trees: a dynamic index structure for spatial searching. In *Proceedings of the 1984 ACM SIGMOD international conference on Management of data*, pages 47–57. 82

Hassan, A. and Radev, D. (2010). Identifying text polarity using random walks. In *ACL '10: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 395–403, Morristown, NJ, USA. Association for Computational Linguistics. 57

Hatzivassiloglou, V. and McKeown, K. (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, page 181. 32, 57

Hinneburg, A. and Keim, D. A. (1998). An efficient approach to clustering in large multimedia databases with noise. *Data Mining and Knowledge Discovery*, 5865:58–65. 51, 80, 87

Huang, T., Yu, Y., Li, K., and Zeng, W. (2009). Reckon the parameter of dbscan for multi-density data sets with constraints. In *2009 International Conference on Artificial Intelligence and Computational Intelligence*, pages 375–379. IEEE. 51

Hwang, S.-Y., Liu, Y.-H., Chiu, J.-K., and Lim, E.-P. (2005). Mining mobile group patterns: A trajectory-based approach. In *PAKDD*, pages 713–718. 55

Ibrahim, L., Minshawi, W., Ekkab, I., Al-Jurf, N., Babrahim, A., and Al-Halees, S. (2009). Enhancing the dbscan and agglomerative clustering algorithms to solve network planning problem. In *2009 IEEE International Conference on Data Mining Workshops*, pages 662–667. IEEE. 51

Isaaks, E. and Srivastava, R. (1989). *An introduction to geostatistics*. Oxford University Press. 48

Iyengar, V. S. (2004). On detecting space-time clusters. In *Proceedings of the 10th International Conference on Knowledge Discovery and Data Mining (KDD'04)*, pages 587–592. ACM. 36

Jaffe, A., Naaman, M., Tassa, T., and Davis, M. (2006). Generating summaries and visualization for large collections of geo-referenced photographs. In *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*, pages 89–98. 47, 48

Jeung, H., Yiu, M. L., Zhou, X., Jensen, C. S., and Shen, H. T. (2008). Discovery of convoys in trajectory databases. *Proc. VLDB Endow.*, 1(1):1068–1080. 56

Jian, L., Wei, Y., and Bao-Ping, Y. (2009). Memory effect in dbscan algorithm. In *Computer Science & Education, 2009. ICCSE'09. 4th International Conference on*, pages 31–36. IEEE. 51

Jiang, H., Li, J., Yi, S., Wang, X., and Hu, X. (2011). A new hybrid method based on partitioning-based dbscan and ant clustering. *Expert Systems with Applications*. 51

Jijkoun, V., de Rijke, M., and Weerkamp, W. (2010). Generating focused topic-specific sentiment lexicons. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 585–594. Association for Computational Linguistics. 57

Kalnis, P., Mamoulis, N., and Bakiras, S. (2005). On discovering moving clusters in spatio-temporal data. *Advances in Spatial and Temporal Databases*, pages 364–381. 51, 56

Kamps, J., Marx, M., Mokken, R., and De Rijke, M. (2004). Using WordNet to measure semantic orientation of adjectives. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004*, volume 4, pages 1115–1118. 57

Kang, J. and Yong, H.-S. (2009). Mining Trajectory Patterns by Incorporating Temporal Properties. *Proceedings of the 1st International Conference on Emerging Databases*. 53, 54

Kang, J. H., Welbourne, W., Stewart, B., and Borriello, G. (2004). Extracting places from traces of locations. In *WMASH '04: Proceedings of the 2nd ACM international workshop on Wireless mobile applications and services on WLAN hotspots*, pages 110–118, New York, NY, USA. ACM. 53

Keim, D. (2005). Scaling visual analytics to very large data sets. In *Workshop on Visual Analytics*. 27

Keim, D., Andrienko, G., Fekete, J., Goerg, C., Kohlhammer, J., and Melancon, G. (2008). Visual analytics: Definition, process, and challenges. *Information Visualization*, pages 154–175. 52

Kennedy, A. and Inkpen, D. (2006). Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence*, 22(2):110–125. 32

Kennedy, L. and Naaman, M. (2008). Generating diverse and representative image search results for landmarks. In *Proceeding of the 17th international conference on World Wide Web*, pages 297–306. ACM. 44

Kennedy, L., Naaman, M., Ahern, S., Nair, R., and Rattenbury, T. (2007). How flickr helps us make sense of the world: context and content in community-contributed media collections. In *Proceedings of the 15th international Conference on Multimedia*, page 640. 47

Kimerling, A., Buckley, A., Muehrcke, P., and Muehrcke, J. (2009). *Map Use: Reading and Analysis*. Esri Press. 48

Kisilevich, S., Keim, D. A., Andrienko, N., and Andrienko, G. (2011). *Towards acquisition of semantics of places and events by multi-perspective analysis of geotagged photo collections*. Lecture Notes in Geoinformation and Cartography (to appear). Springer. 160

Kisilevich, S., Keim, D. A., and Rokach, L. (2010a). A novel approach to mining travel sequences using collections of geotagged photos. In *Proceedings of the 13th AGILE International Conference on Geographic Information Science*. 159

Kisilevich, S., Krstajic, M., Keim, D. A., Andrienko, N., and Andrienko, G. (2010b). Event-based analysis of people's activities and behavior using Flickr and Panoramio geo-tagged photo collections. In *2nd International Symposium Visual Analytics*. 159

Kisilevich, S., Mansmann, F., and Keim, D. A. (2010c). P-DBSCAN: A density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos. In *1st International Conference on Computing for Geospatial Research & Application*. 159

Kisilevich, S., Rohrdantz, C., and Keim, D. (2010d). "beautiful picture of an ugly place". exploring photo collections using opinion and sentiment analysis of user comments. In *Computational Linguistics & Applications (CLA 10)*, pages 419–428. 159

Kramer, O. and Danielsiek, H. (2010). Dbscan-based multi-objective niching to approximate equivalent pareto-subsets. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, pages 503–510. ACM. 51

Kulldorff, M. (1997). A spatial scan statistic. *Communications in Statistics: Theory and Methods*, 26(6):1481–1496. 36

Lee, J.-G., Han, J., and Whang, K.-Y. (2007). Trajectory clustering: a partition-and-group framework. In *SIGMOD Conference*, pages 593–604. 55

Li, Y., Han, J., and Yang, J. (2004). Clustering moving objects. In *Proceedings of the 10th International Conference on Knowledge Discovery and Data Mining (KDD'04)*, pages 617–622. ACM. 38, 55

Liu, B. (2009). Sentiment Analysis and Subjectivity. *Handbook of Natural Language Processing, Second Edition,(editors: N. Indurkhya and FJ Damerau)*. 32, 57, 129

Lundblad, P., Eurenius, O., and Heldring, T. (2009). Interactive visualization of weather and ship data. In *Proceedings of the 13th International Conference Information Visualisation*, pages 379–386, Washington, DC, USA. IEEE Computer Society. 59

MacEachren, A. and Kraak, M. (2001). Research challenges in geovisualization. *Cartography and Geographic Information Science*, 28(1):3–12. 52

Maimon, O. and Rokach, L. (2005). *Data mining and knowledge discovery handbook*. Springer-Verlag New York Inc. 31

Martino, S. D., Bimonte, S., Bertolotto, M., and Ferrucci, F. (2009). Integrating google earth within olap tools for multidimensional exploration and analysis of spatial data. In *Proceedings of the 11th International Conference on Enterprise Information Systems*, pages 940–951. 34, 58, 59

MicrosoftTeam (2009). *Microsoft Application Architecture Guide*. Microsoft Press. 142

Miller, H. J. and Han, J. (2009). *Geographic data mining and knowledge discovery*. Chapman & Hall/CRC. 124

Nanni, M. and Pedreschi, D. (2006). Time-focused clustering of trajectories of moving objects. *Journal of Intelligent Information Systems*, 27(3):267–289. 50, 52

Nasibov, E. and Ulutagay, G. (2009). Robustness of density-based clustering methods with various neighborhood relations. *Fuzzy Sets and Systems*, 160(24):3601–3615. 51

Oelke, D., Hao, M., Rohrdantz, C., Keim, D. A., Dayal, U., Haug, L.-E., and Janetzko, H. (2009). Visual opinion analysis of customer feedback data. In *VAST '09: Proceedings of the 2009 IEEE Symposium on Visual Analytics Science and Technology*, pages 187–194. 58

O'Hare, N., Davy, M., Bermingham, A., Ferguson, P., Sheridan, P., Gurrin, C., and Smeaton, A. (2009). Topic-dependent sentiment analysis of financial blogs. In *Proceeding of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, pages 9–16. 32, 57

Osgood, C. (1957). *The measurement of meaning*. University of Illinois Press. 33, 130

Palma, A. T., Bogorny, V., Kuijpers, B., and Alvares, L. O. (2008). A clustering-based approach for discovering interesting places in trajectories. In *SAC '08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 863–868. 51, 53

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, volume 43, pages 115–124. 32

Parker, J., Hall, L., and Kandel, A. (2010). Scalable fuzzy neighborhood dbscan. In *Fuzzy Systems (FUZZ), 2010 IEEE International Conference on*, pages 1–8. IEEE. 51

Patterson, T. (2007). Google Earth as a (not just) geography education tool. *Journal of Geography*, 106(4):145–152. 34

Pelekis, N., Kopanakis, I., Marketos, G., Ntoutsi, I., Andrienko, G., and Theodoridis, Y. (2007). Similarity search in trajectory databases. In *TIME '07: Proceedings of the 14th International Symposium on Temporal Representation and Reasoning*, pages 129–140, Washington, DC, USA. IEEE Computer Society. 50

Peters, D. (2008). *Building a GIS: System Architecture Design Strategies for Managers*. ESRI Press. 33

Pezanowski, S., Tomaszewski, B., and MacEachren, A. (2007). An open geospatial standards-enabled google earth application to support crisis management. *Geomatics Solutions for Disaster Management*, pages 225–238. 34, 58, 59

Piatetsky-Shapiro, G. (1991). Discovery, analysis, and presentation of strong rules. *Knowledge discovery in databases*, pages 229–238. 124

Popescu, A.-M. and Etzioni, O. (2005). Extracting product features and opinions from reviews. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 339–346, Morristown, NJ, USA. 58

Qiu, G., Liu, B., Bu, J., and Chen, C. (2009). Expanding domain sentiment lexicon through double propagation. In *Proceedings of the 21st international jont conference on Artifical intelligence*, pages 1199–1204. 57

Rattenbury, T., Good, N., and Naaman, M. (2007). Towards automatic extraction of event and place semantics from flickr tags. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, page 110. 47

Rigoutsos, I. and Floratos, A. (1998). Combinatorial pattern discovery in biological sequences: The teiresias algorithm. *Bioinformatics*, 14(1):55. 114, 116, 120, 145, 147

Riloff, E. and Wiebe, J. (2003). Learning extraction patterns for subjective expressions. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 105–112. 58

Rinzivillo, S., Pedreschi, D., Nanni, M., Giannotti, F., Andrienko, N., and Andrienko, G. (2008). Visually driven analysis of movement data by progressive clustering. *Information Visualization*, 7(3):225–239. 52

Rokach, L., Romano, R., and Maimon, O. (2008a). Mining manufacturing databases to discover the effect of operation sequence on the product quality. *Journal of Intelligent Manufacturing*, 19(3):313–325. 115

Rokach, L., Romano, R., and Maimon, O. (2008b). Negation recognition in medical narrative reports. *Information Retrieval*, 11(6):499–538. 115

Rosswog, J. and Ghose, K. (2010). Efficiently detecting clusters of mobile objects in the presence of dense noise. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, pages 1095–1102. ACM. 51

Ruiz, C., Spiliopoulou, M., and Menasalvas, E. (2010). Density-based semi-supervised clustering. *Data Mining and Knowledge Discovery*, 21(3):345–370. 51

Ryden, K. (2005). OpenGIS® Implementation Specification for Geographic information - Simple feature access - Part 2: SQL option. *Open Geospatial Consortium*. http://www.opengeospatial.org/standards/sfs. 82

Sahlgren, M., Karlgren, J., and Eriksson, G. (2007). SICS: Valence annotation based on seeds in word space. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 296–299. 57

Salton, G. and Buckley, C. (1988a). Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523. 33

Salton, G. and Buckley, C. (1988b). Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523. 92

Salvetti, F., Lewis, S., and Reichenbach, C. (2004). Automatic opinion polarity classification of movie reviews. *Colorado research in linguistics*, 17(1). 32, 57

Santhisree, K., Damodaram, A., Appaji, S., and Nagarjunadevi, D. (2010). Web usage data clustering using dbscan algorithm and set similarities. In *2010 International Conference on Data Storage and Data Engineering*, pages 220–224. IEEE. 51

Schaeffer, B. and Foerster, T. (2008). A client for distributed geo-processing and workflow design. *Location Based Services Journal*, 2(3):194–210. 33

Sheppard, S. and Cizek, P. (2009). The ethics of Google Earth: Crossing thresholds from spatial data to landscape visualisation. *Journal of environmental management*, 90(6):2102–2117. 34

Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. pages 336–43. 27

Shrout, P. and Fleiss, J. (1979). Intraclass correlations: uses in assessing rater reliability. *Psychol Bull*, 86(2):420–428. 136

Silverman, B. (1986). *Density estimation for statistics and data analysis*. Chapman & Hall/CRC. 48

Simon, H. A. (1971). Designing Organizations for an Information-Rich World. *Computers, communications, and the public interest*, pages 37–72. 27

Simon, I., Snavely, N., and Seitz, S. (2007). Scene summarization for online image collections. 44

Slingsby, A., Dykes, J., Wood, J., and Clarke, K. (2007). Interactive tag maps and tag clouds for the multiscale exploration of large spatio-temporal datasets. In *IV '07: Proceedings of the 11th International Conference Information Visualization*, pages 497–504, Washington, DC, USA. IEEE Computer Society. 34, 58, 59

Slocum, T. (1999). *Thematic cartography and visualization*. Prentice Hall Upper Saddle River, NJ. 69

Slocum, T. A., McMaster, R. B., Kessler, F. C., and Howard, H. H. (2008). *Thematic Cartography and Geovisualization*. Prentice Hall. 48

Smith, T. and Lakshmanan, V. (2006). Utilizing google earth as a gis platform for weather applications. In *22nd International Conference on Interactive Information Processing Systems for Meteorology, Oceanography, and Hydrology, Atlanta, GA, 29 January-2 February 2006*. 34, 58, 59

Snavely, N., Simon, I., Goesele, M., Szeliski, R., and Seitz, S. (2010). Scene reconstruction and visualization from community photo collections. *Proceedings of the IEEE*, 98(8):1370–1390. 44

Spaccapietra, S., Parent, C., Damiani, M., De Macedo, J., Porto, F., and Vangenot, C. (2008). A conceptual view on trajectories. *Data & knowledge engineering*, 65(1):126–146. 55

Spärck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21. 132

Stein, M. L. (1999). *Interpolation of Spatial Data: Some Theory for Kriging*. Springer Series in Statistics. 48

Stensgaard, A., Saarnak, C., Utzinger, J., Vounatsou, P., Simoonga, C., Mushinge, G., Rahbek, C., Møhlenberg, F., and Kristensen, T. (2009). Virtual globes and geospatial health: the potential of new tools in the management and control of vector-borne diseases. *Geospatial Health*, 3(2):127–141. 34, 58

Subrahmanian, V. and Reforgiato, D. (2008). AVA: Adjective-Verb-Adverb Combinations for Sentiment Analysis. *IEEE Intelligent Systems*, 23(4):43–50. 32, 57

Tang, M., Zhou, Y., Li, J., Wang, W., Cui, P., Hou, Y., Luo, Z., Li, J., Lei, F., and Yan, B. (2011). Exploring the wild birds' migration data for the disease spread study of h5n1: a clustering and association approach. *Knowledge and Information Systems*, pages 1–25. 51

Tepwankul, A. and Maneewongvatana, S. (2010). Customized dbscan for clustering uncertain objects. In *2010 Third International Conference on Knowledge Discovery and Data Mining*, pages 90–93. IEEE. 51

Tepwankul, A. and Maneewongwattana, S. (2010). U-dbscan: A density-based clustering algorithm for uncertain objects. In *Data Engineering Workshops (ICDEW), 2010 IEEE 26th International Conference on*, pages 136–143. IEEE. 51

Theodoridis, Y. (2003). Ten benchmark database queries for location-based services. *The Computer Journal*, 46(6):713–725. 50

Tobler, W. (1987). Experiments in migration mapping by computer. *Cartography and Geographic Information Science*, 14(2):155–163. 69

Toutanova, K. and Manning, C. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora*, pages 63–70. Association for Computational Linguistics. 129

Turney, P. (2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 417–424. 32, 57, 131

Unwin, A., Theus, M., and Hofmann, H. (2006). *Graphics of Large Datasets: Visualizing a Million Series, In: Statistics and Computing*. Springer. 48

Vieira, M. R., Bakalov, P., and Tsotras, V. J. (2009). On-line discovery of flock patterns in spatio-temporal data. In *GIS '09: Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 286–295, New York, NY, USA. ACM. 51, 56

Viswanath, P. and Suresh Babu, V. (2009). Rough-dbscan: A fast hybrid density based clustering method for large data sets. *Pattern Recognition Letters*, 30(16):1477–1488. 51

Vlachos, M., Hadjieleftheriou, M., Gunopulos, D., and Keogh, E. (2003). Indexing multidimensional time-series with support for multiple distance measures. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 216–225, New York, NY, USA. ACM. 50

Vlachos, M., Kollios, G., and Gunopulos, D. (2002). Discovering similar multidimensional trajectories. In *Proceedings of the International Conference on Data Engineering*, pages 673–684. 50

Wachowicz, M., Ying, X., Ligtenberg, A., and Ur, W. (2002). Land use change explorer: A tool for geographic knowledge discovery. In *In Anseling, L., Rey S.J. (eds), New Tools for Spatial Data Analysis, Proceedings of the CSISS specialist meeting.* 59

Wang, M., Wang, A., and Li, A. (2006). Mining Spatial-temporal Clusters from Geo-databases. *Lecture Notes in Computer Science*, 4093:263. 36

Wells, W. M. (1986). Efficient synthesis of gaussian filters by cascaded uniform filters. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(2):234–239. 48

Wiebe, J. (2000). Learning subjective adjectives from corpora. In *Proceedings of the National Conference on Artificial Intelligence*, pages 735–741. 32, 130

Wiebe, J., Brace, R., and O'Hara, T. (1999). Development and use of a gold-standard data set for subjectivity classifications. In *Annual meeting-association for computational linguistics*, volume 37, pages 246–253. 32, 130

Wood, J., Dykes, J., Slingsby, A., and Clarke, K. (2007). Interactive Visual Exploration of a Large Spatio-temporal Dataset: Reflections on a Geovisualization Mashup. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1176–1183. 34, 58, 59

Yang, C., Wang, F., and Huang, B. (2009). Internet traffic classification using dbscan. In *2009 WASE International Conference on Information Engineering*, pages 163–166. IEEE. 51

Yue, S., Wang, J., Wu, T., and Wang, H. (2010). A new separation measure for improving the effectiveness of validity indices. *Information Sciences*, 180(5):748–764. 51

Zhang, P., Huang, Y., Shekhar, S., and Kumar, V. (2003). Correlation analysis of spatial time series datasets: A filter-and-refine approach. In *In the Proc. of the 7th PAKDD.* 38

Zhang, T., Ramakrishnan, R., and Livny, M. (1996). BIRCH: an efficient data clustering method for very large databases. *ACM SIGMOD Record*, 25(2):103–114. 54

Zhao, W., He, Q., Ma, H., and Shi, Z. (2011). Effective semi-supervised document clustering via active learning with instance-level constraints. *Knowledge and Information Systems*, pages 1–19. 51

Zheng, Y., Zhang, L., Xie, X., and Ma, W.-Y. (2009). Mining interesting locations and travel sequences from gps trajectories. In *WWW '09: Proceedings of the 18th international conference on World wide web*, pages 791–800. 30, 53, 55

Zipf, G. (1949). *Human behavior and the principle of least effort: An introduction to human ecology.* Addison-Wesley Press. 33, 132