Middle East Technical University

Informatics Institute

VISUALIZATION OF SPATIAL DATA WITH USING OPEN SOURCE TECHNOLOGIES

Advisor Name: Assoc. Prof. Dr. Altan KOÇYİĞİT

(METU)

Student Name: Önder ALTINTAŞ

(SM)

January 2019

TECHNICAL REPORT

METU/II-TR-2019-

Orta Doğu Teknik Üniversitesi
Enformatik Enstitüsü

AÇIK KAYNAK TEKNOLOJİLERİ KULLANILARAK MEKANSAL VERİNİN GÖRÜNTÜLENMESİ

Danışman Adı: Doç. Dr. Altan KOÇYİĞİT
(ODTÜ)

Öğrenci Adı: Önder ALTINTAŞ
(SM)

Ocak 2019

TEKNİK RAPOR
ODTÜ/II-TR-2019-

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY (Internal Use)	2. REPORT DATE 15-01-2019

3. TITLE AND SUBTITLE

VISUALIZATION OF SPATIAL DATA WITH USING OPEN SOURCE TECHNOLOGIES

4. AUTHOR (S) Önder ALTINTAŞ	5. REPORT NUMBER (Internal Use) METU/II-TR-2019-

6. SPONSORING/ MONITORING AGENCY NAME(S) AND SIGNATURE(S)

Software Management Master's Programme, Department of Information Systems, Informatics Institute, METU

Advisor: Assoc. Prof. Dr. Altan KOÇYİĞİT Signature:

7. SUPPLEMENTARY NOTES

8. ABSTRACT (MAXIMUM 200 WORDS)

Every day, more data are being created in the world and stored in data stores. A considerable amount of these data contains spatial information. It is hard to make intense spatial data understandable for users also it is hard to visualize dense spatial data because of the performance problems. This report presents common spatial data visualization techniques with their advantages and disadvantages and explains a system to achieve spatial data visualization. In this project, common spatial visualization techniques were explained on a GIS software application created for the purpose. It can use any kind of point feature data that has values to be selected. As it was explained, all visualization methods have their own advantages and disadvantages.

9. SUBJECT TERMS GIS, Spatial, Data Visualization, Software	10. NUMBER OF PAGES 25

Table of Contents

Table of Tables

No table of figures entries found.

Table of Figures

1 Introduction

Every day, with increasing acceleration, more data are being created in the world and stored in data stores. A considerable amount of these data contains spatial information. By comparing traditional data visualization techniques, understandable with high performance visualization of dense spatial data creates new problem domain for GIS applications.

The purpose of the project is to describing most common spatial data visualization techniques which are point representation, clustering, heat map, interpolation, choropleth and, reviewing their advantages and disadvantages with creating a GIS software application.

In the scope of this project, software that obtains and consolidates twitter data from various points in most crowded cities in Turkey and a software that is able to achieve visualization of obtained spatial data with most common visualization techniques will be developed with using open source software development tools. These open source tools are OpenLayers as a map framework, MongoDB as a database system and node.js as an application container to be able to run JavaScript codes. In the project, only open source tools were used just to show their ease of use and usefulness. Thus, explanation of spatial data visualization techniques and their advantages and disadvantages will be presented by hands-on experiences on a live application.

2 Background

2.1 Spatial Data

Spatial data also known as Geospatial data is data about objects, events, or phenomena that have a location on the surface of the earth [1] .Spatial data consist of points, lines, polygons and other geographic data which can be mapped by location. Currently, the spatial data can be stored and used in two main forms which are vector and raster.

2.1.1 Vector Data

Vector Data is a type of data to be able to representation spatial information with using points, lines and polygons. Although different GIS programs might have different kinds of geometries (rectangles, square, circle, ellipse, arc etc.); basically all the geometries can be defined with using points, lines and polygons.

2.1.2 Raster Data

Raster data is a type of data which contains pixels carrying value as information. For example on temperature raster data each pixel has one band gray value is temperature as celcius degree or in elevation raster each pixel has color value as elevation value as meter. In addition, also digital aerial photographs, satellite images, scanned maps are also considered as raster data to represent real world.

2.2 GIS Software

GIS Software is a type of system to be able to store, retrieve, manage, display, and analyze all types of geographic and spatial data. Today, most of the GIS software products are focused on decision support task for different problem domains like defense industry, mining/oil operations, emergency and site selection. Therefore, to be able to support decisions; visualization of spatial data is crucial for GIS software products.

3 Related Works

Although there is not much information exists for academically documented resources, mainly the GIS software applications provide the information about their own visualization standards.

Currently, Open Geospatial Consortium (OGC) is working to be able to create standards for GIS applications, data models and naming.

Right now, there are various kinds of applications which differ on data representation.

3.1 Esri ArcGIS

The licensed software application is one of the first GIS applications. By the definition in their own website, ArcGIS is a platform for organizations to create, manage, share, and analyze spatial data. It consists of server components, mobile and desktop applications, and developer tools. This platform can be deployed on-premises or in the cloud (Amazon, Azure) with ArcGIS Enterprise, or used via ArcGIS Online which is hosted and managed by Esri [2].

The company is one of the first GIS application development companies. So, with higher experience level, their current toolsets and capabilities are higher than most of the other GIS applications.

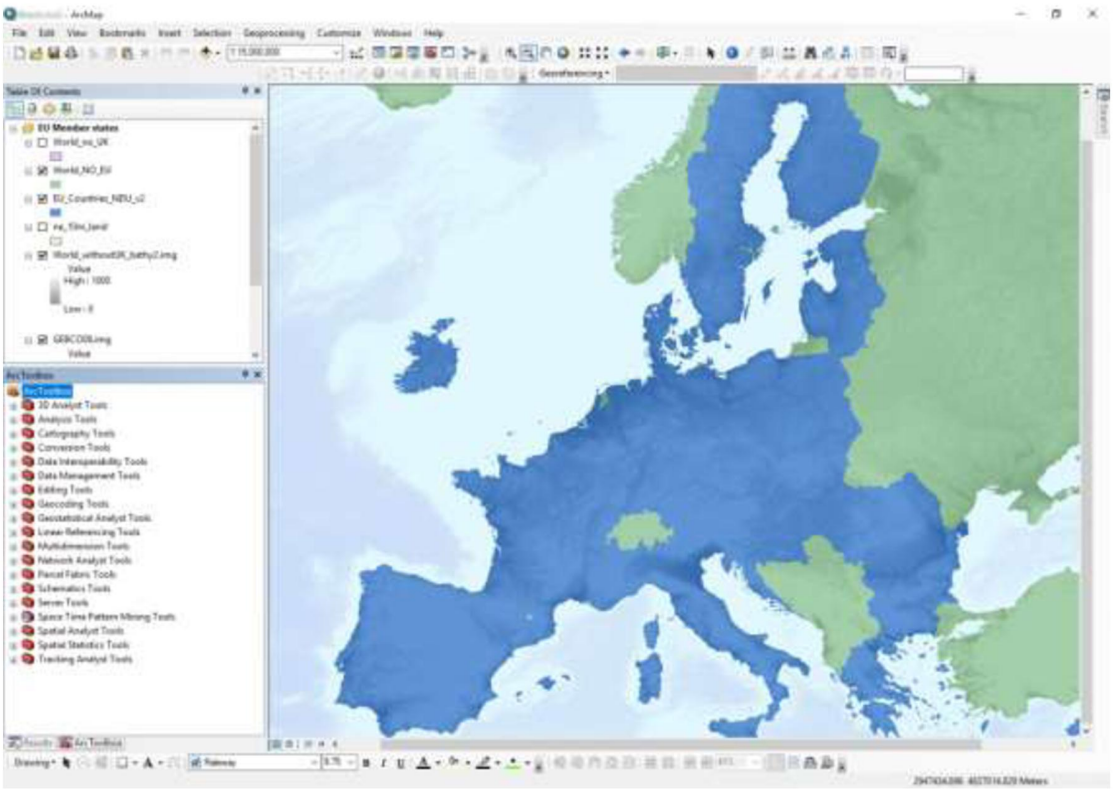

Figure 1 – ArcGIS User Interface

3.2 QGIS

QGIS –formerly known as Quantum GIS- is a professional GIS application that is built as Free and Open Source Software (FOSS) licensed under the GNU General Public License.

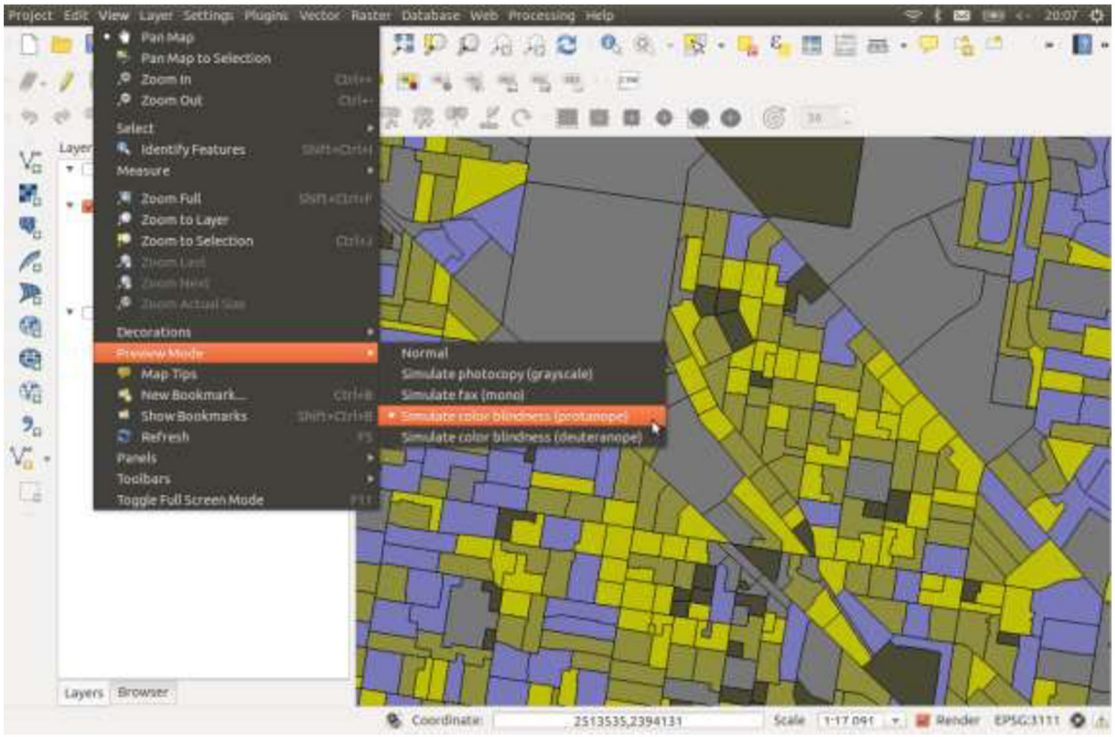

Figure 2 – QGIS User Interface

4 The System

The system that was created for the project mainly consists of 2 different software products. Twitter Crawler and Spatial Data Visualizer. Basically, Twitter Crawler gathers data from Twitter API and consolidates by adding some attributes and saves the data to database and Spatial Data Visualizer is a web based application that gathers data with using user's queries and represents on a map with different visualization techniques.

The open-source technologies used in the system are node.js, express.js, OpenLayers 3, MongoDB. In addition to these technologies, for both back-end and front-end javascript was used as a programming language. Both crawler and visualizer uses node.js and MongoDB. OpenLayers 3 was used for presentation (front end) and express.js was used to create web services and web server (back-end).

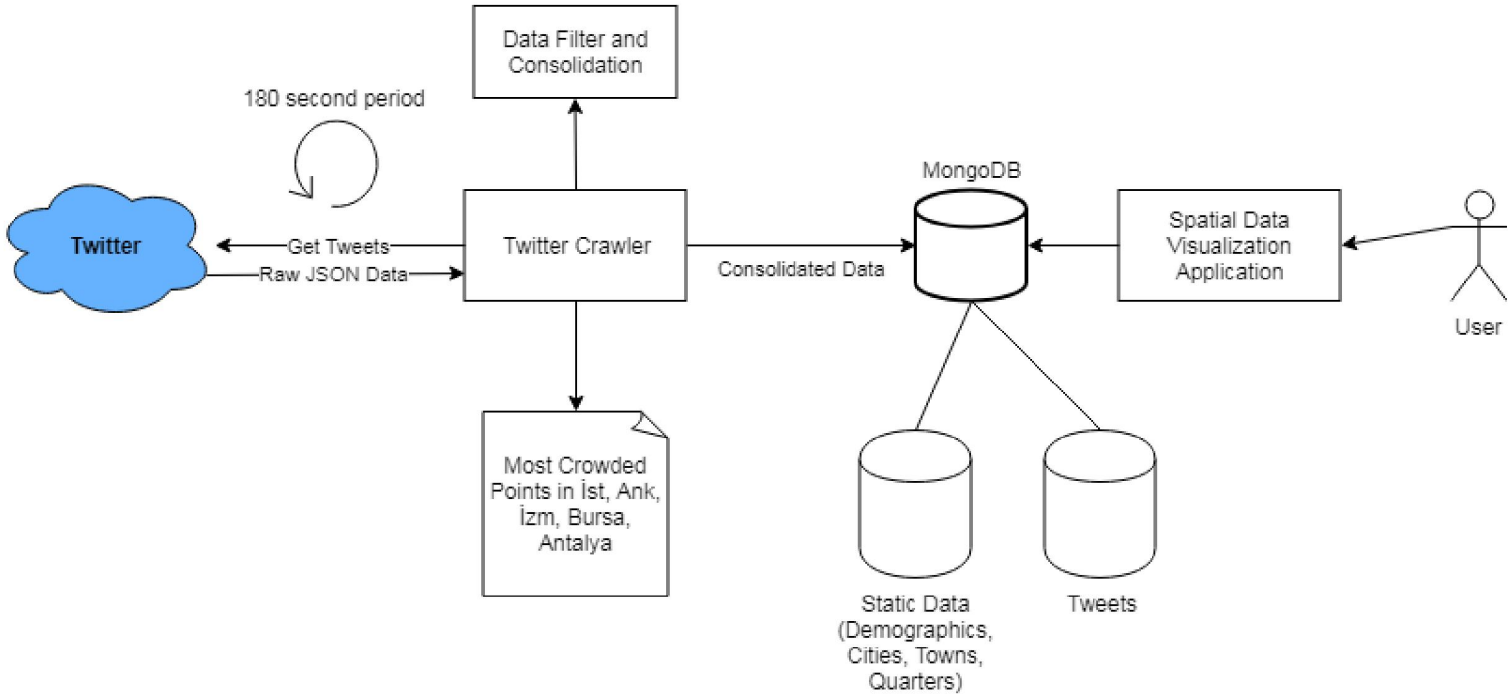

Figure 3 - Context Diagram of System

4.1 Twitter Crawler

While visualizing spatial data, firstly, the data should be exists to be reached. In addition to that also the data should be arranged for visualization requirements. Therefore, it was required to create a software product that gathers data, arranges the attributes and save the data to database as a part of the project. In the scope of the project, there were only public tweets with locations gathered.

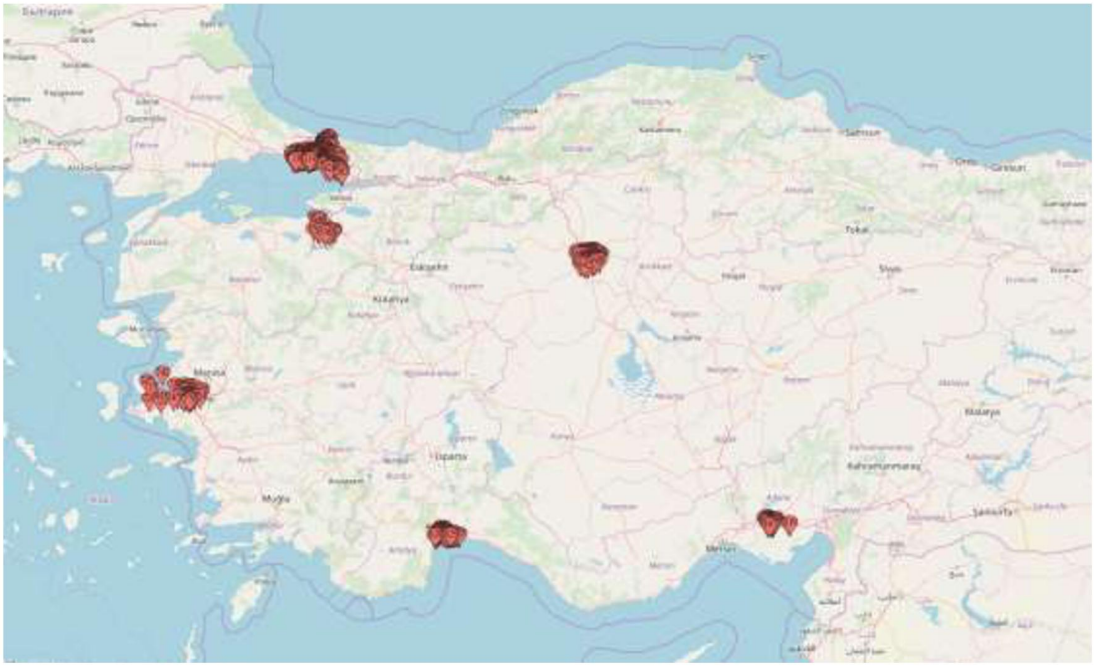

Figure 4 - Selected cities for gathering tweets

While the crawler does not have any complex task, it just uses Twitter API with API's limitations [3]. There are 6 cities –Ankara, İstanbul, İzmir, Antalya, Bursa and Adana- were selected to gather twitter data and 831 points defined in points.dat file for crawler's usage. The points have least 1km distance between each other since the queries for api are searching the tweets in 1 km radius. Since sending queries are limited by Twitter, mostly crawded points are selected and eliminated.

Figure 5 - Selected points for crawling

Mainly, crawler starts to work by reading data from points.dat file. It selects first 83 points and sends to the twitter api search endpoint with geocode parameter. After obtaining tweets from Twitter, firstly the same tweets gets filtered and only unique tweets stay. The program creates Lat (as latitude), Lon (as longitude) and Location (as GeoJSON point location) attributes for each object and saves to the MongoDB. After saving obtained tweets to database, the program starts to sleep for 180 seconds. 180 seconds later, program wakes up and selects next 83 points from defined points. This procedure continues forever and cycle starts from first 83 points when all the points were queried.

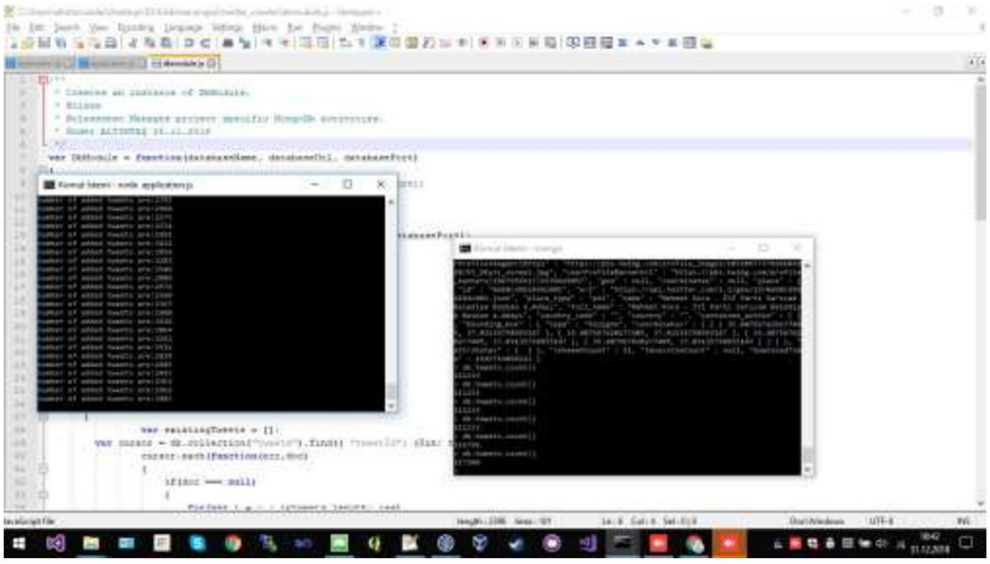

Figure 6 - Twitter Crawler is gathering tweets

As chosen a standard the attributes added by the program starts with capital letter. These attributes were required because while some of the tweets have location as a point, some of them have no location or have location as different format. The tweets that have "place" attribute were bound to point which are the center points of their places' bounding boxes. So, after arrangement, all the tweets had Latitude, Longitude attributes for point in box queries and Location attributes for geospatial queries.

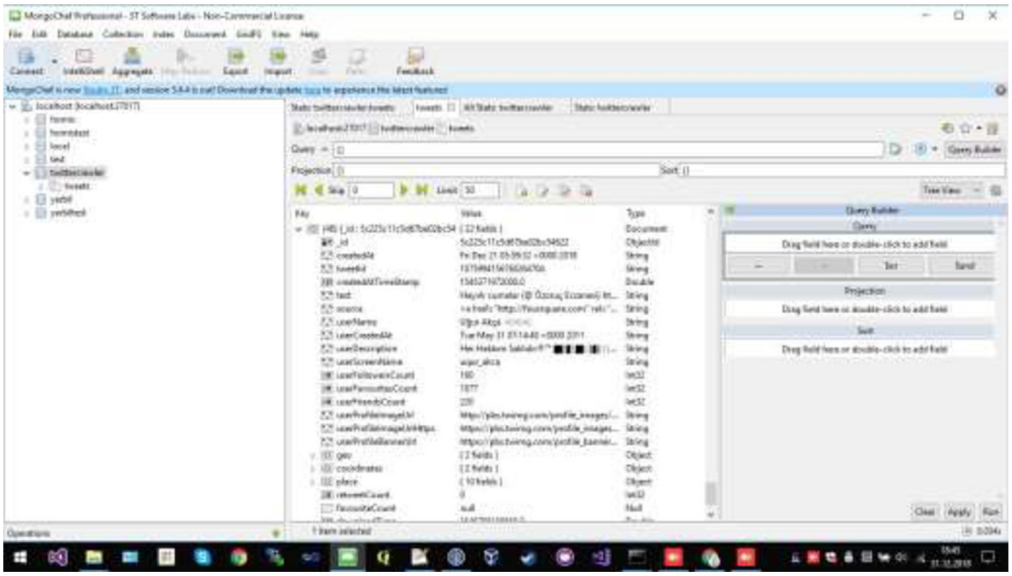

Figure 7 - Tweet data structure

4.2 Spatial Data Visualizer

Spatial Data Visualizer is a web based software product that represents spatial data – geo-located tweets for this project- on the map with various spatial data visualization techniques. Back-End (also labeled as Service Layer) side was built on node.js and express.js with using javascript for creating web server, serving the content and services. The Front-End (also labeled as Presentation Layer) side is also using JavaScript for communicating with back-end rest services and showing the data. In addition, OpenLayers is being used for visualizing data and map. The rest of the application was written with using html5, javascript and css3 technologies.

4.2.1 Back-End (Service Layer)

The service layer was created to serve the spatial data. Rest services were created in GeoDataWebServiceManager class. Database connection classes are using DbManager class which is a generically created class for all projects that are using node.js and MongoDB. GeoDataDb classes are using the DbManager for obtaining required data (static data which are city, town, quarter, demographic data and dynamic data which is twitter data). Server is a non-class application that creates web-server and sets configurations for application.

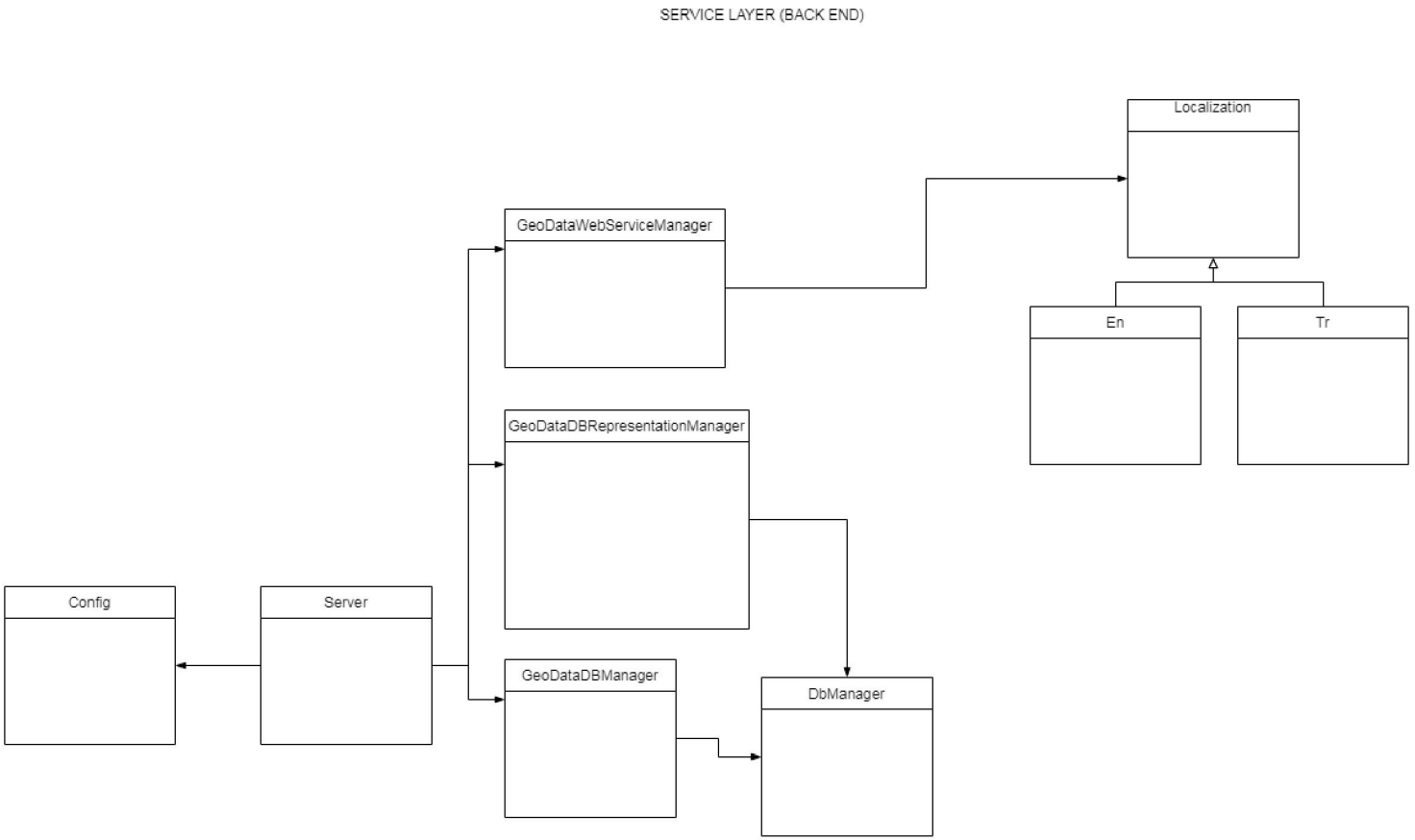

Figure 8 - Back End (Service Layer)

4.2.2 Front-End (Presentation Layer)

The presentation layer is for visualization of the data. It only does service calls to obtain data and creates visualization on client side. Application class is for just wrapping the applications inside. The map application calls window manager for creating windows to place different visualization windows with different dom query selectors. Window manager's purpose is just creating moveable windows that has close and minimize buttons. Map manager is responsible for managing map activities including creating visualization layers and converting data to format which OpenLayers can accept.

Visualization is for doing different kinds of spatial data visualizations. Except choropleth quarters, all visualizations call "pointrepresentation" service to show the data but differs on presentation layer. The choropleth quarters, on the other hand, requires polygonal data to be able to colorize polygons up to total twitter counts.

The common query parameters are date and a bounding geometry for all visualizations except choropleth quarters. First and second date selectors define time interval (e.g. 12 to 18 december 2018) and the geometry limits the tweets that will come from server to a specific area defined by the user. This area either can be a quarter or can be a drawn area. For size of the drawn area, defined maximum area size is 25 km^2. But to show performance differences, for all visualizations except "inverse distance weighting" the limits were lifted.

For inverse distance weighting, there are some other parameters like search radius, inverse distance power, pixel width and colors defined. User can change these parameters by his own but the pixel width will adapt performance criteria. Since idw algorithm complexity n^2+n for reducing overload pixel size increases when it detects slow performance for this application.

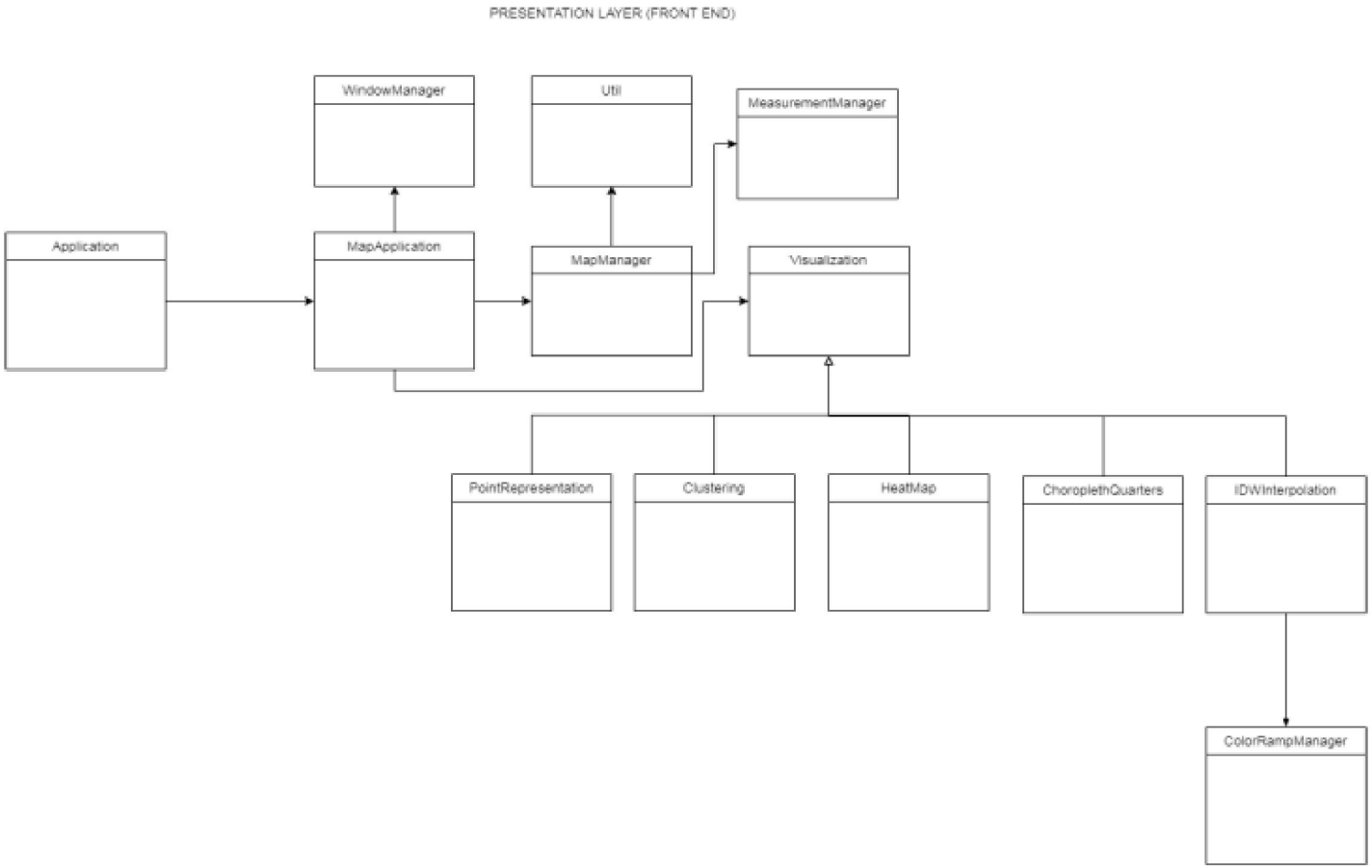

Figure 9 - Front End (Presentation Layer)

5 Spatial Data Visualization Techniques

There were 5 spatial visualization techniques implemented for the project. These techniques are: point representation, point clustering, choropleth map, interpolation and heat map. Each technique has its own advantages and disadvantages. In addition to these visualization categories, there were also tooltip and tabular form representations and border lines to limit spatial data queries were implemented as supporting visualizations to create more clear and understandable GIS environment.

5.1 Supporting Visualizations

5.1.1 Tooltip

Tooltip is a small visual element to be able to show summary information about hovered map feature. It should hold the data as minimal as it can be not to close the map and data visualization.

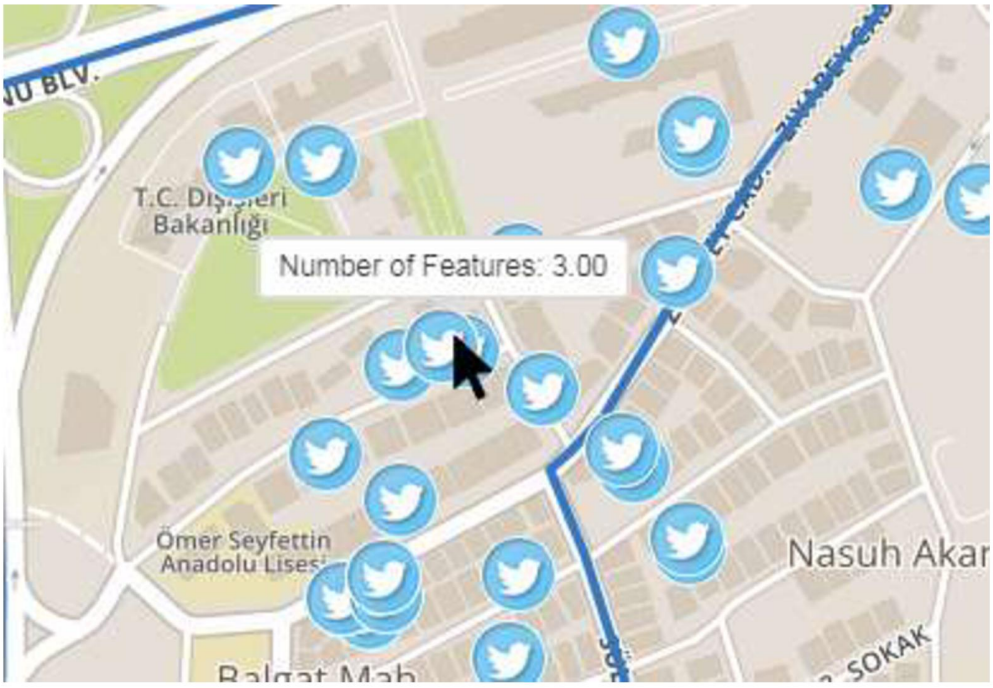

Figure 10 - Tooltip is showing the number of features hovered

5.1.2 Tabular Form

Tabular form can be used to show wider informations for the selected feature or features. It can be both used for data visualization and textual data creation. In the example shown in Figure 9, the demographic information for selected quarter (Balgat Mahallesi) can be seen in a window which has tables.

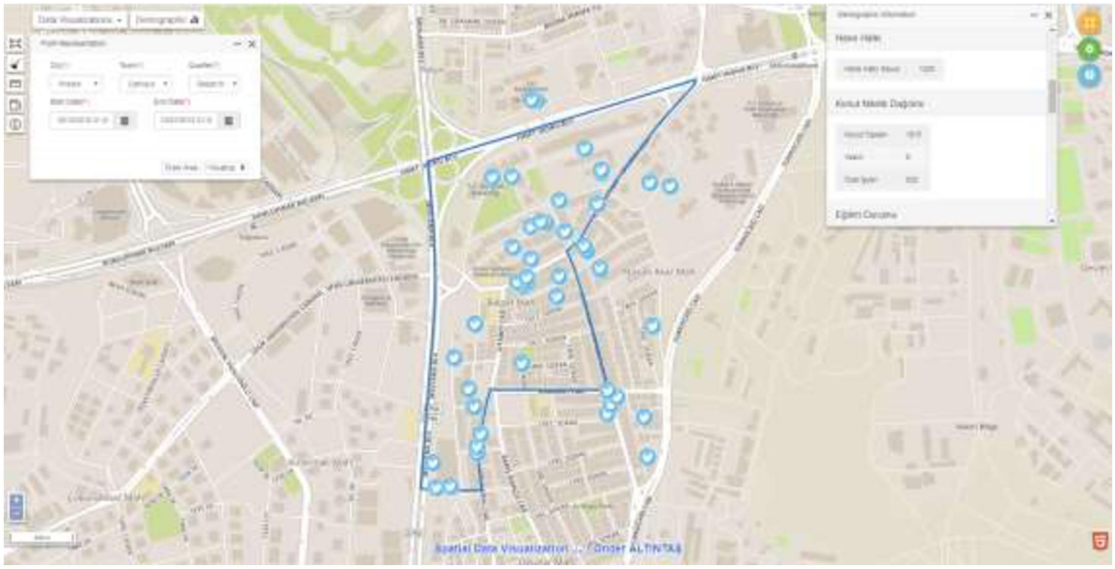

Figure 11 - Tabular information for selected quarter and Point Representation

5.2 Main Visualization Techniques

5.2.1 Point Representation

As it can be seen on Figure 11, point representation is a visualization that shows each feature as points. Although it is useful and understandable for limited areas like quarters, when the area gets wider, understandability of points will be reduced since they will be stacked on top of each other. In addition, the increasing the size of selected area will also be harmful to application performance because of the problem of showing too many points at same application screen.

5.2.2 Point Clustering

Point clustering is gathering nearest points at defined area and representing them together as one point. When the points are getting stacked on top of each other, point clustering technique can be used with both performance and understandability reasons. Oh the project, all the clusters are showing the number of point features they include and the sizes of the clusters differ from 15 pixels to 30 pixels up to their number of points. Each cluster group search 50 pixel range for gathering points in the cluster and the cluster's locations are being determined by the mean value of all locations of features in the cluster. This visualization is good for wider areas for making data more understandable for the users and it is also reduces the performance impact since less points will be visualized. On the other hand, when the user requires more detailed information, the clusters hide some of the feature and location information at lower scales. Therefore, it can be more useful to use both point representation and point clustering together or give option to user to choose which one to be shown.

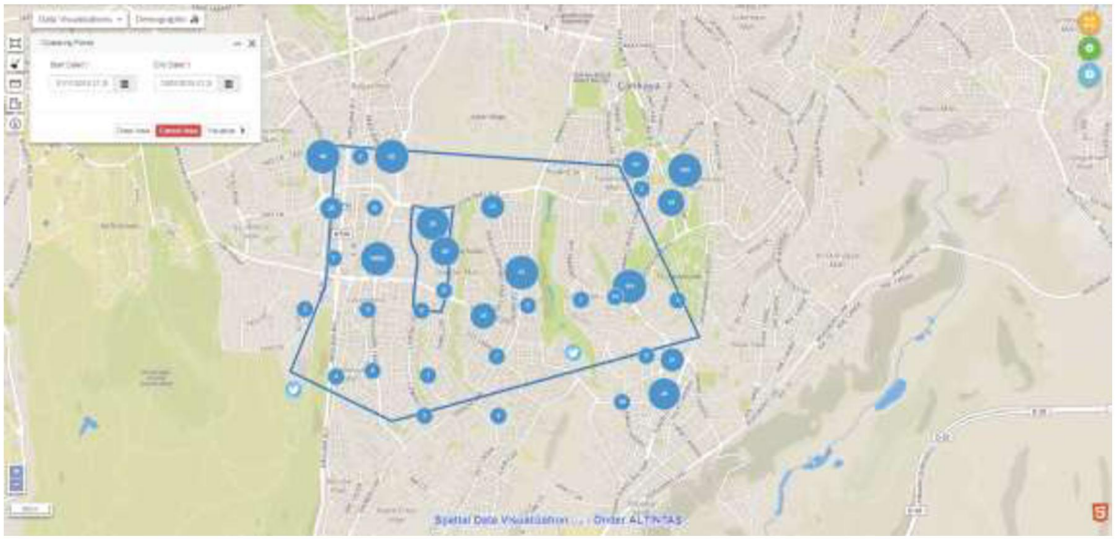

Figure 12 - Point Clustering

5.2.3 Choropleth Map

A choropleth map is a thematic map where geographic regions are colored, shaded, or patterned in relation to a value [4]. Colors show the different values between maximum and minimum defined values. In the project, the value is decided as number of tweets. Each quarter shows how many number of tweets that has with a color. Although, this visualization summarizes the whole data as little pieces; it only gives information about massive amount of data together instead of individuals. For example, in the project, the colors give information about number of tweets but they don't give information about tweet creators, texts or tweet create times.

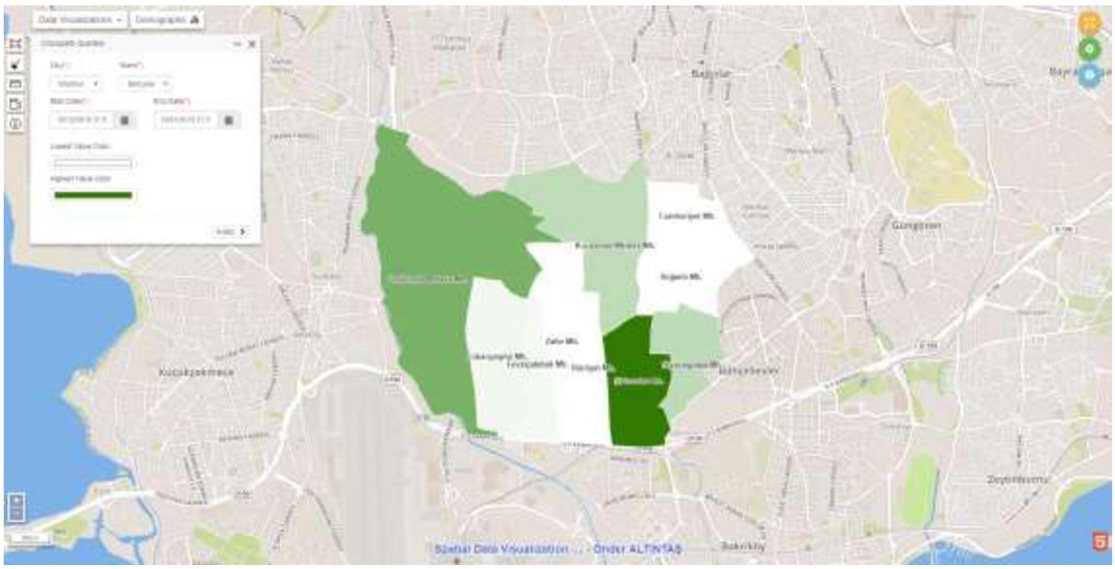

Figure 13 - Choropleth Map

5.2.4 Interpolation

Interpolation predicts values for cells in a raster from a limited number of sample data points [5]. In this project, the prediction calculated by inverse distance weighting algorithm. This interpolation technique is mostly used for mining operations to predict probability of new mine areas from given known mine areas with predicted volumes.

Interpolation is useful for predictions and to be able to lower performance impacts. Once the pixel values were calculated, the program doesn't need to re-calculate the raster once more. On the other hand, initial calculation time can be longer since the related algorithms require all the pixels to be iterated. Therefore, it can be better decision to make pre-calculations and create raster before dynamic usage.

Figure 14 - Inverse Distance Weighting Interpolation

5.2.5 Heat Map

Heat map is a visualization technique that shows dense points from low density areas. Heat maps are useful for identifying patterns, especially "hot spots" or regions of high concentration of the variable. This is useful when hotspot areas required to be identified. On the other hand, when the density increases or for evenly distributed points/values, heat map can lose its usefulness. In addition to that, when the number of data increases, the calculations for heat maps take longer time.

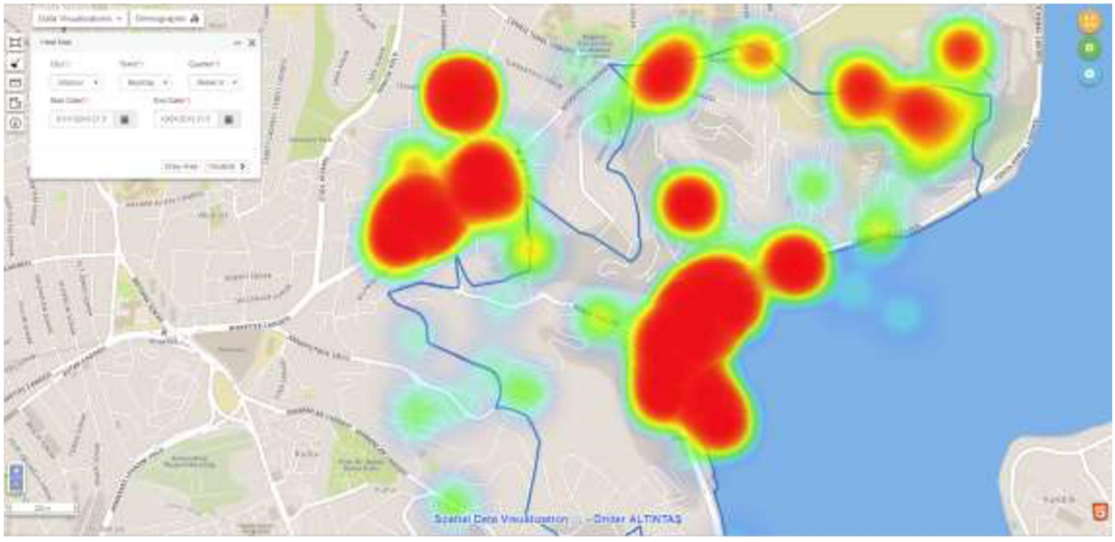

Figure 15 - Heat Map

6 Conclusion and Future Works

In this project, common spatial visualization techniques were explained on a GIS software application created for this purpose. It can use any kind of point feature data which have values to be selected. As it was explained, all visualization methods have their own advantages and disadvantages.

As a result, point representation and heat map visualizations were found to be better visualization techniques on limited and high scale areas. Furthermore, clustering was better on wider areas for creating clearer visualization and interpolation was usable to be able to create visualizations with using lower computation resources. Lastly choropleth visualization found to be better for summarization of high volume of point spatial data.

In the future, with usage of more data, the scalability issues can be tested on big data problems. Categorization, arrangement and standardization of spatial visualization techniques can be created since there is no documented standardization on data visualization techniques. In addition, hashtag and text search can be added to application to be able to visualize tweets' contents.

References

[1] Kristin Stock., Hans Guesgen, "Geospatial Reasoning with Open Data", 2016. https://www.sciencedirect.com/science/article/pii/B9780128029169000105#!

[2] "Data Visualization with ArcGIS" Retrieved from : https://developers.arcgis.com/features/visualization/

[3] "Rate Limiting" Retrieved from: https://developer.twitter.com/en/docs/basics/rate-limiting.html

[4] "5 Popular Thematic Map Types and Techniques for Spatial Data" Retrieved from: https://carto.com/blog/popular-thematic-map-types-techniques-spatial-data/

[5] "What is a spatial interpolation?" Retrieved from: http://planet.botany.uwc.ac.za/nisl/GIS/spatial/chap_1_11.htm