

# GeoTemCo: Comparative Visualization of Geospatial-Temporal Data with Clutter Removal Based on Dynamic Delaunay Triangulations

Stefan Jänicke<sup>1</sup>, Christian Heine<sup>2</sup>, and Gerik Scheuermann<sup>1</sup>

<sup>1</sup>Image and Signal Processing Group, Institute for Computer Science,  
University of Leipzig, Germany

{stjaenicke,scheuermann}@informatik.uni-leipzig.de

<sup>2</sup>ETH Zürich, Switzerland

cheine@inf.ethz.ch

**Abstract.** The amount of online data annotated with geospatial and temporal metadata has grown rapidly in the recent years. Providers like Flickr and Twitter are popular, but hard to browse. Many systems exist that, in multiple linked views, show the data under geospatial, temporal, and topical aspects. We unify and extend these systems in a Web application to support comparison of multiple, potentially large result sets of textual queries with extended interaction capabilities. We present a novel fast algorithm using a dynamic Delaunay triangulation for merging glyphs in the map view into so-called circle groups to avoid visual clutter, which is critical for the comparative setting. We evaluate our design by qualitative comparison with existing systems.

**Keywords:** Visual data exploration, geovisualization, comparative visualization, dynamic Delaunay triangulation.

## 1 Introduction

Although the amount and types of data available through public Web resources is seemingly endless, finding information is still largely performed by text queries. Popular search engines rank the typically huge amounts of query results based on relevance and popularity. When too many irrelevant items remain, the user is required to restate the query by adding or replacing search terms. Unfortunately, repeated refinement can lead to frustration. An alternative is to spatialize the data and allow the user to refine queries using mouse-based navigation.

While the data amount is increasing, data also become more structured. Websites such as Flickr and Twitter provide rich data sources annotated with geospatial and temporal metadata. This metadata can be used to provide a contextual overview of the data in many forms: topical, geospatial, and temporal, being some of the most popular. Users already familiar with searching in geographic environments like Google maps, can find results presented directly on a map rather than in a list, emphasizing the geospatial aspect. A tool providing more contextual overview and filter capabilities allows for synergistic effects.

In this paper, we present a Web application that enables the synergetic exploration of multiple topical queries in a geospatial and temporal context. It employs a map view for the geospatial context, a time view for the temporal context, and provides on-demand tag clouds for topical refinements. All views provide linked brushing. This allows, for instance, to juxtapose and compare spatial distribution and temporal trends of multiple queries. Our design is based on a number of published systems that each allow only a subset of our requirements: comparing multiple datasets, refining temporal context, and scaling to a large number of items, both computationally and visually. We achieve the later by zoom-dependent aggregation of result locations into non-overlapping circles, thus avoiding visual clutter in the map view.

We extend our design, which we reported on in a previous version of this paper [1], to reflect the results of a usability study. In particular, we made circles transparent to mitigate potential occlusions of map labels, provide a new tag cloud design for detailed comparison of result sets, and present a novel aggregation algorithm that removes the run-time complexity in the start-up phase of our system from  $O(n^2)$  to  $O(n \log n)$ .

## 2 Related Work

Our work can be placed in the domain of thematic cartography and geovisualization. Overviews of the field are given by, e.g. Dent [2] and Slocum *et al.* [3]. An overview of tools, principles, challenges, and the concept of the analysis of geotemporal data are given by Andrienko and Andrienko [4, 5]. An important role plays the representation of data at multiple levels of detail to allow both gaining an overview and an interactive drilling down onto the details, as well as linking different views on the data, i.e. allowing interaction in one view to affect other views. This typically makes it easier to find causal relationships and unforeseen connections.

Because of limited space we can address only a few representative designs that exploit synergetic effects of linked views for geospatial-temporal data.

The Web application *VisGets* [6] employs four linked views: a location view, showing the result items as small glyphs on a map, a time view, showing histograms of results for year, month, and day resolution, a tag view, showing most-frequent words in a size proportional to their importance, and a results view, showing small textual or image thumbnails of results arranged as a table. *VisGets* offers query refinement in space (selecting a glyph), time (selecting a year, month, or day), and by topic (selecting a tag). Each refinement affects the presentation in all other views. *VisGets* also provides many mechanisms to interact with the results of a single query, but a comparison of different queries is not directly supported. Because no aggregation of closely positioned glyphs takes place, visual clutter often ensues. The time view supports only query refinement for months within the same year, or days within the same month. The provided time resolutions make it also inconvenient to work with datasets spanning centuries or just hours and minutes.

*GeoVISTA CrimeViz* [7] enables comparison and analysis of different crime incident types. The map view shows individual circles only when zoomed in. In overviews, the incidents of all types are aggregated into hexagonal bins, thereby disabling comparison. The display of the bins using transparency interacts with the map: in dense areas the map is occluded and the underlying map can bias the transparency perception. In the CrimeViz time view, the number of incidents per time period of different types are stacked. It can show the total trend well, but is insufficient for comparing the different incident types [8]. The supported linked interactions are asymmetric: a selection in the time view (year, month, week, no time ranges) filters items on the map but not vice versa. CrimeViz’s data source offers 8 different incident types, of which only the 3 smallest (with a maximum total of 637 incidents per year) are used. The other 5 types, ranging from 2,600 to 9,300 incidents per year each, are omitted.

The problem with many results in a map view becomes apparent in the visualization of the Iraq conflict incidents by *The Guardian* [9]. It does not make use of the different casualty types present in the data source or time information, and suffers from “red-dot fever” [10]: glyphs overlap in the map to an extent where the spatial distribution can no longer be determined reliably. To support large datasets, our tool thus automatically aggregates result hits in the map to non-overlapping circles. It becomes indispensable when presenting glyphs of different visual properties, since overlaps could bias the perceived distributions.

The increasing data sizes to be shown in scatter plots, forced the development of binning strategies. Data-independent, top-down approaches like rectangular binning [11] and hexagonal binning [12] split the plane into rectangular or hexagonal bins. The number of items per bin can be reflected in different ways, e.g., a bin coloring with associated colors from a predefined color map. Furthermore, specific shapes (e.g., circles, hexagons) reflecting the bin count with size can be placed in the bin area. A special case for shapes are so called sunflower plots [13], which reflect the number of items in one bin of the scatter plot with a sunflower glyph, that has a specific number of petals for a specific number of points.

These top-down binning strategies are widely used in geoapplications [5]. However, Novotny [14] remarked that the result can be misleading since cluster centers might be split into distinct bins. He proposes  $K$ -means [15] as a bottom-up binning approach, but it requires an appropriate selection of the number of clusters. General clustering algorithms cannot make assumptions on the dimensionality of the data and can therefore be slow. In this paper, we propose to use a fast data-driven clustering employing a dynamic 2D Delaunay triangulation algorithm. Delaunay triangulations are typically used in geographic information systems to model terrain [3], but to the best of our knowledge, we are the first to employ it as a clustering algorithm for clutter removal of glyphs.

### 3 Design

Our system’s design is inspired by Dörk *et al.* [6]. It also consists of a map view showing the position of query results, a time view showing the distribution of

results in a time span, and a detail view showing textual contents or thumbnails of data items arranged in a table. Our system differs in that we allow comparison of multiple result sets from a classical keyword or term-based topical search, support time-span data, show tag clouds for selected data on demand in the map, provide more flexible selections in the time domain, and render glyphs in the map avoiding visual clutter. Internally, we support any simple statistical graph, with the time view being a special case, and a designer can add more views based on the data to show. Fig. 4c gives an example view composition.

We chose colors to mark the different result sets, because of their effective use to discriminate categorical data [16]. An alternative is to use small multiples similar to the system *LISTA-Viz* [17], but we found the use of small multiples makes comparison of scattered data with irregular spatial distribution difficult. Furthermore, colors can serve as the visual link between the different views. As the number of colors that can be easily distinguished by humans is limited to 12 [18], we restricted our method to four datasets, to ensure both a good distinction from the map, as well as allowing colors to mix in the time view. We presume map colors to be mostly dark, cold, or unsaturated, and select very light colors for deselected and very saturated colors for selected circles to ensure that the thematic overlay pops out in comparison to the base map. The four base colors used for datasets 1 through 4 are red, blue, green, and yellow – to accommodate color impaired users while preserving red for single type datasets.

Each view provides native navigation and selection that results in updates of the other views. The map and the time views provide simple zoom and pan. Because of the way we aggregate data in these views an animation between zoom levels or time resolution changes is not performed. We reflect selections by marking table entries and the corresponding fraction of map glyphs and time graphs with saturated versions of the datasets’ base colors. When selections are performed by a mouse drag gesture, the impact of releasing the mouse at this point is immediately reflected in the other views. Selections can be modified by dragging shapes or clicking on table entries in the detail view.

### 3.1 Map View

The dominant view in our visualization in terms of screen space is the *map view*. It is a thematic map [3] comprising a base map and a thematic overlay. The base map can be a contemporary map or one of 23 different historical maps showing political borders from 2000 BC to 1994 AD provided by Thinkquest<sup>1</sup>. Overlaying data over historic maps can benefit applications in the humanities (see e.g. Tshipidis et al. [19] for archaeological data, or the HESTIA project [20], which investigates the differences between imagined geographic distances and real distances in ancient Mediterranean space). Because a dataset may span a time range for which multiple maps are available and there is no concept of “average political border” we show the map closest to the median time stamp occurring in the dataset as default and allow the user to switch maps.

<sup>1</sup> <http://library.thinkquest.org/C006628/>

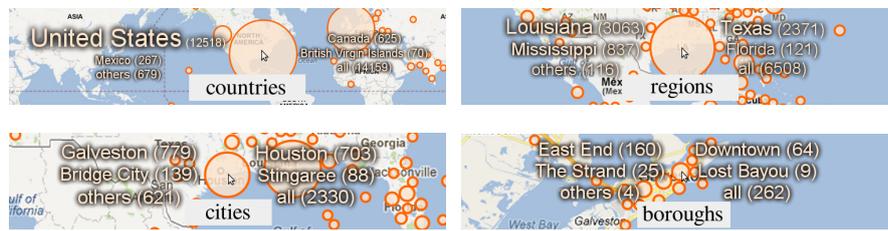


Fig. 1: Place name tag clouds with different levels of detail

For the overlay, we chose a proportional glyph map over isopleth and choropleth maps due to the scattered nature of our data. We can use neither dasy-metric nor dot maps, as these require ancillary information and a cartographer to apply this information correctly. The spatial distribution could also be shown via heat map, but to preserve legibility of the base map and to show multiple datasets, color mixing would ensue, against which Ware [18] argues. Also, humans are more accurate judging areas than they are judging color tones, making areas a better candidate for quantitative values [8]. Using glyphs also allows to group glyphs and to make every data item individually accessible for interaction.

We disallow the glyphs to overlap in order to avoid visual clutter. While drawing rules such as “always let the smaller overlap the larger” can reduce the risk of occluding small glyphs, this is only a solution if glyphs do not differ in their other visual attributes, like shape or color. Instead, we merge circles based on their size, distances, and the current scale in an iterative process.

The overlap removal algorithm, which can be directly attached for single input sets ( $m = 1$ ), is described in the next section. In the case of multiple ( $m = 2, 3, 4$ ) input sets, we compose multiple circles  $c_1, \dots, c_m$  into a more complex glyph – a *circle group* – whose bounding circle  $b$  will be used for the aggregation. We chose them over pie charts, because these improve comparisons of data at the same point at the expense of comparison of the global distribution.

The grouping process is illustrated in Fig. 2d-f. Initially we place the centers of  $m$  prototype circles on the vertices of a regular  $m$ -polygon. The prototype circles’ radius is set equal to the largest circle to group. Then we move the  $m$  circles using their correct radius from their polygon vertex closest to the polygon center without leaving the prototype. Finally we construct the bounding circle by moving it from the polygon center towards the center of the largest circle until its boundary touches two circles. Although this wastes some space, it is quick to compute and allows the map to be seen through. To ensure the map legibility underneath large circles, we draw all circles semi-transparent.

Selections in the map view can be specified by clicking on items or by drawing circles, rectangles, polygons, or clicking on an administrative region (e.g. country), which then selects the items in the surrounding polygon. Each of the map’s circles is associated with a details-on-demand tag cloud showing the most

frequent place names in a font scaled proportional to their frequency. The cloud provides a preview of how a glyph arising from agglomeration would split if zoomed in. If the data offers different levels of detail for a place, we choose the label dependent on the current zoom level. We distinguish between country, region, city, and borough level. We replace missing levels by the next coarser or the next finer level. Fig. 1 gives example place name tag clouds. In the comparison setting we use lines to link tags between datasets, as can be seen in Fig. 5b.

### 3.2 Time View

Using terminology of Harris [21], our time view is a segmented area graph with time for the  $x$ -axis. A segmented area graph is a line chart where the area under the line is filled.  $T_1, \dots, T_n$  partition the interval  $T = [t_{min}, t_{max}]$  of the given dataset into intervals of regular duration: either seconds, minutes, hours, days, weeks, months, quarters, years, or decades. It is chosen to maximize the number of intervals without exceeding 400. Short units typically arise from dynamic data sources and large units arise from data with historic context. The resolution unit changes automatically when the user zooms inside the time view.

Whereas the  $x$ -axis of the time view is directly defined by  $T_1, \dots, T_n$ , the  $y$ -direction shows the number of data items that fall in each interval using binning. For data items with time stamps the counting is straight-forward, for data items with time spans we add a value proportional to the amount of overlap with each bin. Although this can lead to an over-representation of items with long time spans in the time view, we found this to be no problem. For the datasets we considered, either the effect was benign as the time spans had approximately the same duration or items with longer time spans were also more interesting.

In the comparative setting we perform the bin counting per group. In the final visualization the bins' sizes are shown as overlapping segmented area graphs rather than bar charts because the former is better suited to direct comparison of the groups' time distribution. We shade the area under each line using a semi-transparent version of that datasets color, ensuring that all curves are visible and also hinting at the area that would have been present in the bar chart. Through the use of blending, the limitation to four colors as well as the stacked area graphs' shape mitigates ambiguities. The user can switch to logarithmic scale if datasets to be compared have largely different totals.

The time view allows both the clicking on one bin and the selection of a time range using a mouse drag gesture. A toolbar is then shown, that offers to add a "gray zone" which blends between selected and unselected elements. A play button starts an animation mode causing the selected time window to loop.

### 3.3 Detail View

For inspection of single data items that match the current filtering we present small textual or image thumbnails presented as a table in the detail view. This view is the only one which does not include any aggregation, but results are presented on multiple pages if they exceed a certain fixed number.

The system also allows to export elements of a selection as a new dataset. This can be used for the temporal comparison of different geographical regions of one dataset or the geographical comparison of different time periods is possible.

## 4 Overlap Removal Algorithm

In the following let  $k$  denote the number of supported scales (i.e. magnifications). The scale doubles with each level  $l$  ( $1 \leq l \leq k$ ). Furthermore, let  $N$  denote the number of points  $P$ ,  $p_i = (0.5 + lon_i/360^\circ, 0.5 + lat_i/180^\circ)$  in a normalized space. Each circle  $i$  represents  $n_i$  points. We define a minimum radius  $r_{min}$  dependent on the average font size of common Web mapping services' labels (e.g. Google Maps, Bing Maps) so that circles have salience no smaller than labels. The maximum radius was found empirically as  $r_{max} = 4 \log_2(N + 1)$ . A circle's area  $A_i$  is a linear interpolation between the corresponding minimum and maximum circle areas  $A_{min}$  and  $A_{max}$  based on  $n_i$ :

$$A_i = A_{min} + \frac{n_i - 1}{N - 1}(A_{max} - A_{min}).$$

Because neighboring circles are most likely to overlap and we want to merge close circles before far circles, we use a dynamic Delaunay triangulation as supporting data structure, allowing us to quickly find and merge overlapping circles.

The algorithm is detailed in Algorithm 1 and illustrated in Fig. 2a-c. It initially creates a dynamic Delaunay triangulation of the given points in the normalized space merging duplicates. After initialization, all edges of the triangulation are inserted into a priority queue with a priority  $\psi$  depending on the amount of overlap.  $\psi$  relates the distances after transformation from normalized space to screen space and the radii of the circles in pixel. The factor containing  $l$  in the formula for  $\psi$  assumes that the map image for level 1 has dimensions  $256 \times 256$ . For simplicity and speed we use the ratio of the minimum distance desired plus  $\epsilon \geq 0$  and the real distance as priority. Circles overlap or are too close when  $\psi > 1$ . After constructing the priority queue, we repeatedly find the overlap of highest priority and remove it by merging the circles, which affects both the Delaunay triangulation and the priority queue, and finish when there are no more overlaps. Then we proceed with the next scale, which implicitly halves point distances but not radii in the formula for  $\psi$ .

In our implementation we use the algorithm proposed by Kao *et al.* [22]. It is a randomized incremental algorithm that constructs the initial triangulation in  $O(N \log N)$  expected time, and can find a duplicate to a given point in  $O(\log N)$ . As the triangulation contains  $O(N)$  edges, construction of the priority queue using a simple heap requires  $O(N \log N)$  time. Each merging of circles requires 3 elementary operations on the triangulation, each with  $O(\log N)$  expected time, and an expected constant number of updates to the priority queue, each with  $O(\log N)$  time. As building the priority queue and merging circles takes place for each scale we can give the trivial upper bound of our algorithm as  $O(kN \log N)$ . This is a significant improvement over the agglomeration algorithm using clustering presented in the previous version of this paper [1].

---

**Algorithm 1** OverlapRemoval(P)

---

```

 $D \leftarrow$  empty Delaunay triangulation
for  $i = 1$  to  $|P|$  do
  if  $p_i \in D$  then
     $p_j \leftarrow$  duplicate of  $p_i$  in  $D$ 
     $n_j \leftarrow n_j + 1$ 
  else
    Insert( $p_i, D$ )
     $n_i \leftarrow 1$ 
  end if
end for
for  $l = k$  to 1 do
   $Q \leftarrow$  empty priority queue
  for all edges  $\{p_i, p_j\}$  in  $D$  do
     $\psi \leftarrow (\epsilon + r_i + r_j) / (2^{7+l} \|p_i - p_j\|)$ 
    if  $\psi > 1$  then
      Insert( $\{p_i, p_j\}, \psi, Q$ )
    end if
  end for
  while  $Q$  not empty do
     $\{p_i, p_j\} \leftarrow$  highest priority element of  $Q$ 
     $n_{ij} \leftarrow n_i + n_j$ 
     $p_{ij} \leftarrow \frac{n_i}{n_{ij}} p_i + \frac{n_j}{n_{ij}} p_j$ 
    Delete( $p_i, D$ )
    Delete( $p_j, D$ )
    Insert( $p_{ij}, D$ )
    Update( $Q$ )
  end while
end for

```

---

## 5 Results

*GeoTemCo* is the main application which implements the design described in Section 3. *GeoTemCo* works completely within the client's browser, performing filtering, visualization, and interaction, and allows to load data locally or using a server that is an adapter for dynamic data source. Within the development phase for *GeoTemCo* we used among others the following three dynamic sources: Europeana<sup>2</sup>, an online library of several million digitized objects of European's cultural heritage, eAQUA<sup>3</sup>, a project studying topic migrations in the ancient Mediterranean based on words extracted from ancient Greek texts, and Flickr<sup>4</sup>: a database of millions of public user-provided photos. All items are annotated with creation place and date.

<sup>2</sup> <http://www.europeana.eu/>

<sup>3</sup> <http://www.eaqua.net/>

<sup>4</sup> <http://www.flickr.com/>

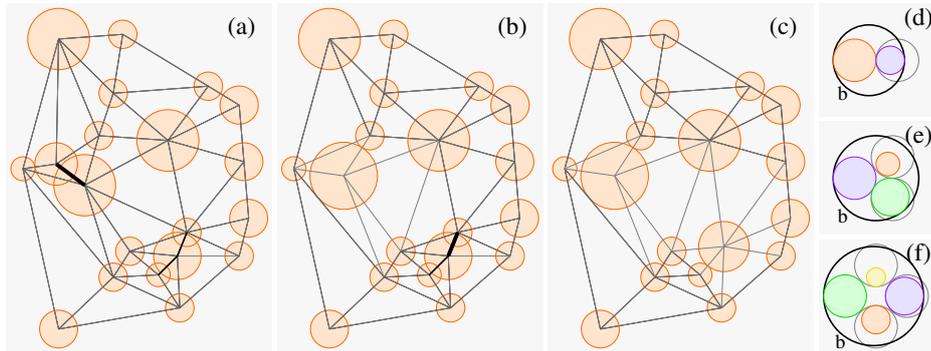


Fig. 2: Aggregation of single and multiple items. The next occlusion to be resolved is marked by line thickness.

*GeoTemCo* supports both KML and JSON exchange formats. Since the overall support for JavaScript as well as its browser performance has increased in the last years, we decided to implement *GeoTemCo* completely in JavaScript. We adapted two OpenSource JavaScript libraries for implementing our views: OpenLayers<sup>5</sup> is used to provide both the thematic layer and the base map from different Web mapping services, e.g., Google Maps, Open Street Map or historic maps, hosted on our own GeoServer<sup>6</sup> instance. Also, we extended the Simile Widgets Timeplot<sup>7</sup> for displaying statistical graphs to support our interaction.

### 5.1 Crime Incidents

We compared the *CrimeViz* application (Fig. 3a) to our visualization (Fig. 3b) for crime incidents in 2009. Both show the distribution of homicide, arson, and sex abuse incidents in the Washington D.C. metropolitan area. Our tool easily shows that there was neither homicide (blue) nor arson (red) incidents in the north western neighborhoods, which is not directly visible in the *CrimeViz* map, because of the hexagonal binning, that furthermore causes a loss of map context in dense regions. When considering further crime types, we find patterns for thefts and robberies near populated places like metro stations or shopping centers. Unlike *CrimeViz*, we aggregate preserving incident types, hence we find more relations, e.g. by comparing the crime types stolen cars (red), burglaries (blue), thefts (green), and robberies (yellow) (Fig. 3c). We detect relatively few stolen car (20%) and burglary incidents (23%) in comparison to thefts (45%) and robberies (35%) in Washington D.C.'s downtown.

Crime analysts can use our visualization to detect connected crime incidents of different types. For instance, a daily exploration reveals, that stolen car in-

<sup>5</sup> <http://openlayers.org/>

<sup>6</sup> <http://geoserver.org/>

<sup>7</sup> <http://www.simile-widgets.org/timeplot/>

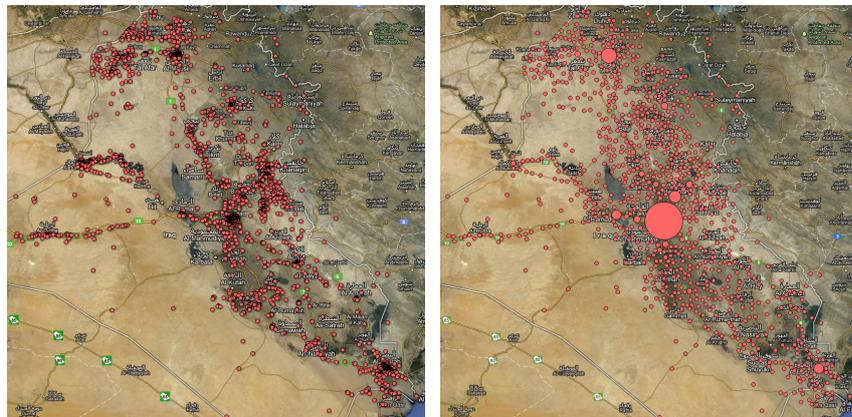


Fig. 3: Crimes in Washington D.C. [Fig. 3a reproduced with permission from [7]]

incidents are often grouped with a theft, burglary or robbery incident, e.g., co-located incidents at 27th of June (Fig. 3d). The number of correlations increases further by choosing a two day time range. The original *CrimeViz* did not allow such fine-grained selection of time ranges. The analyst can furthermore compare different districts using the administrative region selection offered by the map view. Another application could be a decision making support for apartment search, based on regions of low burglary probability, or where it would be safe to rent a garage for a car. Fig. 3e indicates Neighborhood 13 (top) as substantially safer than Neighborhood 39 (bottom) with respect to stolen cars and burglaries.

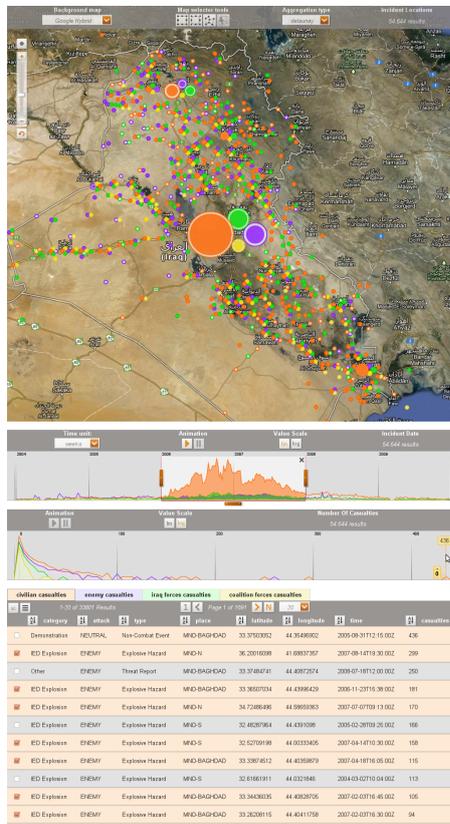
## 5.2 Guardian Data

The *Iraq war logs* dataset as published by *The Guardian*, contains around 60,000 entries; one for each incident with at least one casualty during the Iraq conflict from 2004 to 2009. Each entry states place and time, as well as the number of casualties by type (civilian, enemy, Iraq forces, coalition forces).

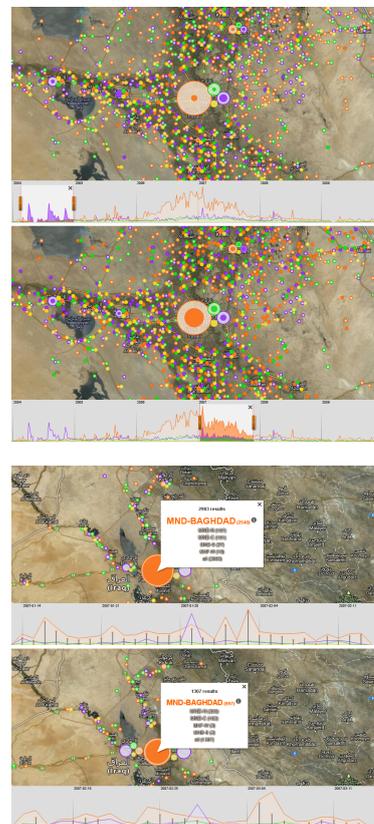


(a) Guardian map

(b) Our map view



(c) Incidents by casualty type



(d) Operation Imposing Law

Fig. 4: Iraq War Logs: Analysis [Fig. 4a reproduced with permission from [9]]

The Guardian visualization [9] for the *Iraq war logs* is a map containing one circle for each incident. It produces a lot of clutter as the result of overlapping glyphs distorting perception of incident densities. In Fig. 4a three conflict centers can be guessed: Baghdad in the center, Al Mausi in the north, and Al Basrah in the south-east, but easily confirmed using the overlap removal algorithm (Fig. 4b) in our tool. Baghdad clearly stands out as the region with most incidents.

For comparison, we split all incidents based on the casualty type into four different datasets. Fig. 4c shows an increased number of incidents with civilian casualties (red) in 2006 and 2007. A second histogram using logarithmic scale shows the number of incidents by casualty total. We discover the incident with most casualties, which is known as the *2005 Baghdad bridge stampede*. In contrast to various newspaper reports of around 1,000 casualties, the data shows only 437 civilian and 7 Iraq forces casualties (green).

Furthermore, we prepared four datasets containing one item for each casualty to point out regions and time periods with lots of casualties. Fig. 4d (top) shows an increased number of casualties in 2007 compared to 2004, except for enemy casualties, in particular in Baghdad. The *Operation Imposing Law* (February 14th - November 24th 2007) reduced the overall number of victims, most notably in Baghdad. On March 14th 2007, the Iraq military stated, that there were only 265 civilian casualties in the first month of *Operation Imposing Law*, which is a low compared to the month before the operation (1440). By filtering using the proper time ranges and clicking the tag cloud for Baghdad, we find 2,540 and 997 civilian casualties, respectively, for these time periods in Fig. 4d (bottom).

### 5.3 Biodiversity Data

The European project *BioVeL*<sup>8</sup> combines several biodiversity databases with the goal to implement and provide flexible user interfaces for biologists for their research on biodiversity issues. In this context GeoTemCo is integrated into a complex workflow process as one of the major visual interfaces for researchers. Biologists select a list of specific species and initialize GeoTemCo with observation entries of different time periods used as different datasets. Exploration and filtering abilities help to detect geospatially migrating species over time. Furthermore, GeoTemCo is used to clean the underlying databases from wrong entries. The results of these user interactions are exploited for the onward workflow.

Fig. 5a gives an example for two observation cycles for marine organisms at the south-eastern coast of Sweden, the first from 1916 to 1938 and the second from 2006 to 2009. For each item, the species' identifier replaces the place name field, so that place name tag clouds show most frequently observed species for geographical areas (Fig. 5b). Hovering, a circle pack's boundary shows tag lists for both circles. Tags which occur in both lists are connected to each other. Hovering an unconnected tag in one list shows an additional link. If the tag also exists in the opposite list, it appears as a new entry (top). Otherwise, the

<sup>8</sup> <http://www.biovel.eu/>

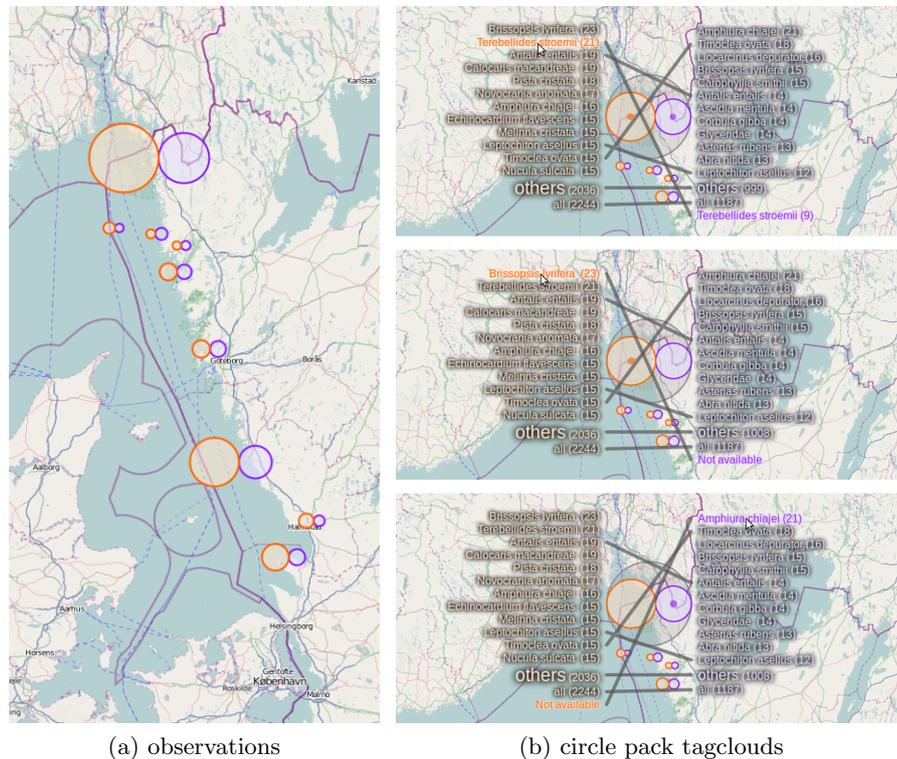


Fig. 5: Observation cycles from 1916-1938 (red) and 2006-2009 (blue)

information “*Not available*” is shown, which indicates migrated (middle) and non-native species (bottom).

#### 5.4 Books from Google Ancient Places

Google Ancient Places (GAP) [23] is concerned with the analysis of books on history. *GapVis*<sup>9</sup> is a visualization from GAP where historic places mentioned in books are plotted onto a map. Page numbers are used as a second dimension, so that an analysis of geospatially migrating topics in books is possible. *GapVis* does not avoid visual clutter on the map and a comparative view for books is also lacking.

We use the same data and utilize our time view for the books’ page dimension. Fig. 6 shows an example for two books about the Roman empire, written by Livy approximately 2000 years ago: *Roman History* and *The History of Rome, Vol. 5*. The political map of 1BC reflects proper historian circumstances. Instantly

<sup>9</sup> <http://nrabinowitz.github.com/gapvis>

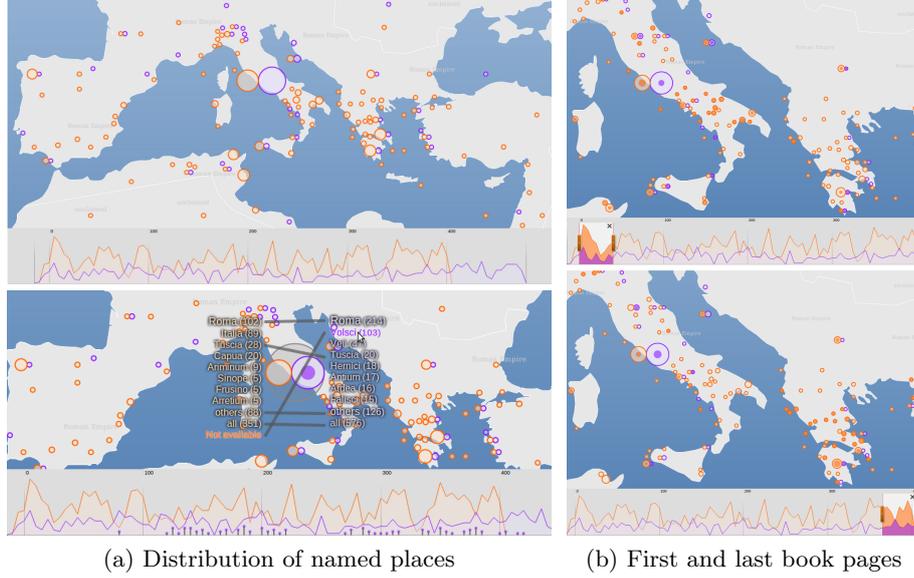


Fig. 6: Livy books: *Roman History* (red) and *The History of Rome, Vol. 5* (blue)

we see quite different geospatial references. Fig. 6a (bottom) clearly indicates that the second book thematizes the conflict between the ancient Rome and the Volscian territory starting at around page 100. In contrast, this period is not discussed in *Roman History*, rather we detect a thematic migration (Fig. 6b) from the ancient Italian (top) to the ancient Greek region (bottom).

### 5.5 Timing Evaluation

We compared the performance of the Delaunay overlap removal algorithm (DOR) with the hierarchical agglomerative clustering method (HAC) as described in [1]. We used the *Iraq war logs* data as benchmark because of its high number of data items non-uniformly spread. We split the incidents randomly into sets of different size and performed several iterations in different browsers for each set. For the test with Ubuntu 12.04, we used a quadcore Intel(R) Core(TM) i5-2410M CPU @ 2.30GHz and 8GB memory. As Internet browsers, we chose Google Chrome 19 and Mozilla Firefox 13. The following table shows the median runtimes of 11 iterations for all datasets with both algorithms in both browsers.

	558	1,185	2,523	4,628	8,917	15,716	33,801	52,048
<b>DOR (Chrome19)</b>	0.08s	0.17s	0.42s	0.86s	1.69s	3.64s	6.85s	10.79s
<b>DOR (FF13)</b>	0.11s	0.26s	0.5s	1.03s	2.1s	3.98s	7.46s	12.45s
<b>HAC (Chrome19)</b>	0.16s	0.37s	1.56s	4.17s	14.41s	42.21s	193.38s	518.24s
<b>HAC (FF13)</b>	0.71s	0.16s	3.07s	7.58s	30.39s	110.35s	483.19s	1038.78s

## 6 Conclusion and Future Work

We presented a novel approach and Web application to show, compare, and explore multiple topical query results in a geographical and temporal context. We were able to utilize, combine, and improve approaches from several prior works. In contrast to CrimeViz [7], which also offers comparative visualization, we display items without aggregating them into the same representatives for coarser zoom levels. We aggregate map glyphs to avoid visual clutter, which is an issue in the Guardian visualization for the Iraq war logs [9]. Compared to the similar system VisGets [6], which only works for one set of results, we also made use of the linked views approach (map, time line, detail view) to extend the users exploration abilities. Furthermore, we enriched the filter capabilities in both geospatial (e.g., selecting all results inside a country) and temporal dimension (e.g., selecting dynamic time ranges).

Our case studies show that visually comparable datasets extend the exploration and analysis abilities of the user in an effective way. It helps to detect equalities and varieties between distinct data contents that unveil their relationships in space and time. Our method is limited to four datasets at a time mainly to ensure that the colors used for discrimination are properly distinguishable, the splitting of circles does not waste too much screen space, and the overlapping segmented area graphs do not occlude each other too much.

In the future we will direct our attention to the extension of our method to very large datasets by searching for client-server communication where most of the data remains on the server but a working set is transmitted to the client for quick interaction. We aim to extend our system to extract and show trajectories of distinct sets containing movement data, potentially leading to new insights. At some point we also want to support live update of dynamic data sources like Twitter feeds to see how topics proliferate. We would also like to extend our system to show the uncertainty in data with historic context, which is often annotated by hand and uses places' names instead of longitude and latitude. These names can be resolved, but often yield polygonal regions.

**Acknowledgments.** We thank Sebastian Bothe, Vera Hernandez-Ernst, Karl-Heinz Sylla, and Robert Kuwaliak from Fraunhofer IAIS for collaborating on the extension of GeoTemCo for several projects, and providing data for Section 5.3. We thank the contributors of Google Ancient Places, especially Leif Isaksen, for providing the underlying data of GapVis.

## References

1. Jänicke, S., Heine, C., Stockmann, R., Scheuermann, G.: Comparative visualization of geospatial-temporal data. In: GRAPP/IVAPP. (2012) 613–625
2. Dent, B.D.: Cartography: Thematic Map Design. 5th edn. McGraw-Hill (1999)
3. Slocum, T.A., McMaster, R.B., Kessler, F.C., Howard, H.H.: Thematic Cartography and Geovisualization. 3rd, international edn. Prentice Hall Series in Geographic Information Science. Prentice Hall (2009)

4. Andrienko, G., Andrienko, N.: Visual Data Exploration: Tools, Principles, and Problems. In: Classics from IJGIS: twenty years of the International journal of geographical information science and systems. CRC Press (2006) 475–479
5. Andrienko, N., Andrienko, G.: Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach. Springer (2005)
6. Dörk, M., Carpendale, S., Collins, C., Williamson, C.: VisGets: Coordinated Visualizations for Web-based Information Exploration and Discovery. *IEEE Transactions on Visualization and Computer Graphics* **14**(6) (2008) 1205–1212
7. Roth, R.E., Ross, K.S., Finch, B.G., Luo, W., MacEachren, A.M.: A User-Centered Approach for Designing and Developing Spatiotemporal Crime Analysis Tools. In Purves, R., Weibel, R., eds.: *Proceedings of GIScience*. (2010)
8. Cleveland, W.S., McGill, R.: Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods. *Journal of the American Statistical Association* **79**(387) (1984) 531–554
9. Rogers, S.: The Guardian - Wikileaks Iraq war logs: every death mapped (2010) <http://www.guardian.co.uk/world/datablog/interactive/2010/oct/23/wikileaks-iraq-deaths-map> (Retrieved 2011-10-07).
10. Schuyler: Web map API roundup: Mapping hacks (2006) <http://mappinghacks.com/2006/04/07/web-map-api-roundup/>.
11. Wilkinson, L.: *The Grammar of Graphics (Statistics and Computing)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA (2005)
12. Carr, D.B., Littlefield, R.J., Nichloson, W.L.: Scatterplot matrix techniques for large n. In: *Proceedings of the Seventeenth Symposium on the interface of computer sciences and statistics on Computer science and statistics*, New York, NY, USA, Elsevier North-Holland, Inc. (1986) 297–306
13. Cleveland, W.S., McGill, R.: The Many Faces of a Scatterplot. *Journal of the American Statistical Association* **79**(388) (1984) 807–822+
14. Novotny, M.: Visually effective information visualization of large data. In: *In 8th Central European Seminar on Computer Graphics (CESCG 2004)*, CRC Press (2004) 41–48
15. Lloyd, S.P.: Least squares quantization in pcm. *IEEE Transactions on Information Theory* **28** (1982) 129–137
16. Bertin, J.: *Semiology of graphics*. University of Wisconsin Press (1983)
17. Hardisty, F., Klippel, A.: Analysing Spatio-Temporal Autocorrelation with LISTA-Viz. *Int. J. Geogr. Inf. Sci.* **24** (October 2010) 1515–1526
18. Ware, C.: *Information Visualization: Perception for Design*. 3rd edn. Morgan Kaufmann (2004)
19. Tsipidis, S., Koussoulakou, A., Kotsakis, K.: Geovisualization and Archaeology: supporting Excavation Site Research. In Ruas, A., ed.: *Advances in Cartography and GIScience*. Volume 2. Volume 6 of *Lecture Notes in Geoinformation and Cartography*., Springer Berlin Heidelberg (2011) 85–107
20. Barker, E., Bouzarovski, S., Pelling, C., Isaksen, L.: Mapping an ancient historian in a digital age: the Herodotus Encoded Space-Text-Image Archive (HESTIA). *Leeds International Classical Studies* **9**(1) (March 2010)
21. Harris, R.L.: *Information Graphics: A Comprehensive Illustrated Reference*. Oxford University Press (1999)
22. Kao, T., Mount, D.M., Saalfeld, A.: Dynamic maintenance of delaunay triangulations. Technical report, University of Maryland at College Park, College Park, MD, USA (1991)
23. Barker, E., Isaksen, L., Byrne, K., Kansa, E.: GAP: a neogeo approach to classical resources. In: *European Conference on Complex Systems 2010*. (2011)