

Using Animation to Visualize Spatio-Temporal Varying COVID-19 Data

Hanan Samet
University of Maryland
hjs@umd.edu

John Kastner
University of Maryland
kastner@umd.edu

Yunheng Han
University of Maryland
yhhan@umd.edu

Hong Wei
University of Maryland
hyw@umd.edu

ABSTRACT

CoronaViz (<http://coronaviz.umiacs.io>) is a research prototype developed by us to enable the dynamic map visualization of COVID-19 related variables including the number of confirmed cases, active cases, recoveries, and deaths all on a daily basis from the Johns Hopkins University web site at ter.ps/coronajhu, by allowing the underlying spatial region and the spanned time interval to vary. Any combination of the variables can be viewed, subject to a possibility of clutter which is avoided by the use of concentric circles (termed geo-circles) whose radius values correspond to the variable values. The variable values are provided both on cumulative and day-by-day bases. The visualization enables spatial and temporal variation.

CCS CONCEPTS

- Information systems → Geographic information systems;
- Human-centered computing → Geographic visualization.

KEYWORDS

COVID-19, Coronavirus, Geographic Information Systems

ACM Reference Format:

Hanan Samet, Yunheng Han, John Kastner, and Hong Wei. 2020. Using Animation to Visualize Spatio-Temporal Varying COVID-19 Data. In *1st ACM SIGSPATIAL International Workshop on Modeling and Understanding the Spread of COVID-19 (COVID-19)*, November 3, 2020, Seattle, WA, USA. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3423459.3430761>

1 INTRODUCTION

CoronaViz (<http://coronaviz.umiacs.io>) is a research prototype developed at the University of Maryland to enable the dynamic map visualization of COVID-19 related variables including the number of confirmed cases, active cases, recoveries, and deaths all on a daily basis from the Johns Hopkins University web site at ter.ps/coronajhu, by allowing the underlying spatial region and the spanned time interval to vary. Any combination of the variables can be viewed, subject to a possibility of clutter which is avoided by the use of concentric circles (termed geo-circles) whose radius values correspond

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

COVID-19, November 3, 2020, Seattle, WA, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-8168-0/20/11...\$15.00
<https://doi.org/10.1145/3423459.3430761>

to the variable values. The variable values are provided both on cumulative and day-by-day bases. Some like the number of confirmed cases and deaths are also reported as a result of normalization with respect to a measure such as per 100,000 inhabitants (termed an incidence rate). Others like the number of deaths are normalized with respect to the number of confirmed cases (termed a mortality rate). The visualization enables spatial and temporal variation.

CoronaViz was motivated by the continuing spread of COVID-19 which led to the desire to track its progress over time to be better prepared to anticipate its emergence in new regions. There exist numerous systems to monitor and map officially released numbers of cases [3], which are the current established means of keeping track of the progress of the virus. However, as we mentioned earlier, these systems do not necessarily paint a complete picture. For example, they are primarily mashups in that they do not support zooming in on the map in the sense that they just increase the resolution of the map but do not show the data for the additional units (e.g., states/provinces, counties, etc.) that have become visible as a result of the zoom. The visualization enables the comparison of disease-related variables pairwise or region-wise. Particular attention is paid to proper scaling of the disease-related variables so that we can visualize them even if they are all small values or large values in terms of magnitude. To run the system, preferably using the Google Chrome or Microsoft Edge browsers on a laptop or desktop, go to <http://coronaviz.umiacs.io>

The rest of this paper is organized as follows. Section 2 discusses the queries our system is able to support. Section 3 reviews related work by discussing existing disease monitoring systems. Section 4 describes the CoronaViz user interface, while Section 5 provides examples of the use of CoronaViz that highlight its utility. Section 6 contains concluding remarks and discusses future work.

2 QUERIES

The values of all of the variables in CoronaViz are presented in a time-varying manner as time moves on with the aid of a time slider thereby leading them to be characterized as *dynamic* variables. This is in contrast to visualization tools where such variables are presented in a graph where time is the horizontal axis and the variable value is the vertical axis thereby leading them to be characterized as *static*. Thus we see that the presentation manner is the key to the characterization. It is not easy to present several static variables as they tend to clutter the display regardless of whether they are represented as one graph for the set of all variables or one graph per variable. The situation becomes more complex when values of the variables vary in a spatially-varying manner. In this case the

only way to deal with the static variables is to repeat the graph at each location. This is OK when the data is spatially sparse but this is not something we can count on.

In contrast, CoronaViz deals with variables that are both time-varying and spatially varying by re-examining the dimensionality of the data in the sense that a time slider is a natural representation of one-dimensional data (i.e., time) while a two-dimensional map is a natural representation of two-dimensional data. The problem is how we represent the values of the variables. One possible solution is via a histogram but this leads to clutter on the display and is cumbersome when the data is not spatially sparse. Moreover, there may be a layout problem here in the sense that we cannot allow the histograms to overlap. An alternative common solution with the same overlap issues is to use solid concentric circles where the radii of the circles correspond to the value and the color corresponds to the identity of the variable. This type of visualization is known in cartography as a *proportional symbol map* [7]. The problem here is when we have multiple dynamic variables as is the case for our application, then only the one with the largest magnitude can be viewed. One solution is to vary the colors of the circles but if this method is used, then we must pay close attention to the order in which we display the circles so that the one with the largest radius is displayed first and the remaining circles be displayed in decreasing order of radius values (this is analogous to the “back-to-front” z-buffer display algorithm used in computer graphics). We can avoid the need to worry about the order in which we display the circles by using hollow concentric circles where again the color indicates the identity of the variable while the radius corresponds to a scaled variable magnitude. We use the term *geo-circle* to describe this approach.

The visual strain posed by having a large number of circles can be relieved by drawing the circles using broken lines of the same width. At times, the width of the broken lines can be increased with the goal of drawing attention to a particular set of concentric circles (i.e., a location whose variable values at a particular instance of time) which is of interest. We do this in the case of a hover operation while panning on the map to show the spatially closest location with nonzero variable values. This operation is common in computer graphics where it is known as a “pick” operation (e.g., see [5]). However, care must be exercised when implementing it in the sense that we don’t always want the closest geo-circle. For example if we are hovering in Brazil, then we want the geo-circle of Brazil even though the geo-circles of Paraguay or Bolivia may be closer to the hover location in Brazil.

CoronaViz makes use of 7 dynamic variables comparing the number of confirmed cases, active cases, recoveries (although not reported by all jurisdictions), and deaths, as well as normalized variants which include the incidence rate (number of confirmed cases per 100,000 inhabitants), mortality rate (number of deaths divided by the number of confirmed cases), and the recovery rate (number of recoveries divided by the sum of the numbers of deaths and recoveries). No active rate is tabulated as the number of active cases is simply the number of confirmed cases minus the number of deaths and recoveries and thus the only possible rate measure is a normalized active cases value per 100,000 inhabitants which is similar to the incidence rate and thus we do not provide it. Concentric circles (i.e., geo-circles) drawn with broken lines are used for

th 4 disease related variable values while concentric circles drawn with solid lines are used for the 3 disease related rates. They are drawn with different colors with the same color being used for the corresponding variable and rate.

The concentric circles make it easy to spot trends and similar values on the map by looking at the magnitude of the radii. Other observations of interest involve trends such as noting lower confirmed case and death counts over time as the circles get smaller. Another encouraging trend is when confirmed case counts become smaller than death counts. Of particular interest is the situation when concentric circles intersect and change their relative order. Of course this must be treated with caution as the magnitudes of the variables change). In particular, of a comparison is only meaningful when comparing variable values and not rates.

There are a number of ways of presenting the variable values. The default in our case is of a cumulative nature. However, it is possible to normalize the values over population, or even area. Normalizing over the area is of possible interest as it could be used to see if densely populated areas are more likely to lead to a higher number of confirmed cases of COVID-19 and deaths.

Our goal is to endow CoronaViz with a full compliment of queries that are consistent with its role as a spatiotemporal database. First of all, we have two types of queries:

- (1) location-based: given a location or time, what are the values of certain variables and rates.
- (2) feature-based: given a variable or rate value, where or when is its value present. This is also known as spatial data mining [2]. In CoronaViz we might be looking for locations or time instances where there are no deaths.

The location-based queries are supported by the ability to pan the map with a hover operation and always returning the variable values with the nearest location for which we have data. CoronaViz supports this query by a PR quadtree where we have one PR quadtree for each of the disease-related dynamic queries or variables or rates. Feature-based queries require the use of a pyramid-like data structure on each of the disease-related dynamic queries or variables or rates.

The animation window is a very important feature as it enables the execution of a range query where the range is temporal. Users can vary the start and end times of the query as well as the animation step size. In addition, users can specify what statistic is being computed for the temporal window. It can be cumulative, or a time period whose length can be in terms of days, weeks, months, or even years. Average values for the window can also be computed. This is particularly useful for the “reopen” discussion which is often based on a rolling weekly daily average computation involving the number of confirmed cases.

Spatial range (also known as window) queries are of great interest. In this case, users use pan and zoom operations to get a map that is focused on a particular desired spatial region (e.g., the minimum bounding areal box that contains Italy). Note that here we find overlap with San Marino and the Holy See (i.e., the Vatican in Rome). In particular, we have one geo-circle that displays the sum of the values of the dynamic variables for all three of these spatial entities. In order to restrict the visualization to Italy, users must zoom in further so that San Marino and the Holy See are not

in the window (i.e., the displayed geo-circle). Alternatively if users only want the Italy, then they could simply pose the textual query with “Italy” as the search string as well as the name of a region such as “Liguria” or city such as “Genoa” for which appropriate indexes exist. Note that as Coronaviz zooms into a region, it has access to more data (as low as county or city level data).

CoronaViz enables the execution of the full compliment of spatiotemporal queries as it supports keeping location fixed while varying time via the time slider, keeping time fixed and letting location vary via the hover, panning, and zooming operations. We can also pick any range of time or space. Users can also take advantage of spatial synonyms when they don’t know the exact name of the location of interest. For example, when seeking a “Rock Concert in Manhattan,” concerts in Harlem, New York City, and Brooklyn are all good answers because of being contained in Manhattan, containing Manhattan, and being a spatially adjacent borough, respectively. This is an example of a proximity query which we saw previously via the use of a hover operation in the case of spatial proximity, and the time slider in the case of temporal proximity. Note that for temporal proximity, we provide the capability to halt an animation at arbitrary time instances as well as resuming or terminating it. In addition, users are also able to set the speed of the animation, as well as to step through an animation by a specific time interval both forward and backward in time.

3 RELATED WORK

Here we first briefly consider prior work dealing with the visualization of spatiotemporal data and then review existing systems designed specifically for monitoring the spread of COVID-19.

3.1 Spatiotemporal Data Visualization

Visualization and analysis of temporally varying geospatial data is a difficult task; as such, it has been the subject of substantial prior work. The difficulty comes from the inherently multidimensional nature of the data: there are at a minimum two spatial dimensions and one temporal dimension, in addition to the dimensionality added by the actual variables being visualized. All of these dimensions must be projected onto a two dimensions screen. We can broadly break spatiotemporal visualization techniques into two groups: those that use animation to capture the time dimensions, and those that attempt to encode temporally varying information into a single static visualization.

An example of this second variant is presented by Du et al. [4] who modify the traditional choropleth map to encode temporal information inside each area unit. Rather than picking a single color for each areal unit, units are divided either into bands of either equal width or equal area. Each band is then assigned a color in the same way areal units are assigned colors in traditional choropleth maps (e.g. Howard et al. [7]).

Li et al. [11] do not use a fully animated approach, but neither do they commit to showing the full temporal data range in a single image. Instead, they use an interface termed the “Event View” to display images generated for discrete time intervals side-by-side. To link these images together into a single cohesive visualization, the authors overlay a “trend line” that connects the time intervals. This trend line is used to link events extracted by a separate component of their system.

Very often a temporal variant of a well known cartographic visualization technique can be obtained by applying the existing technique to data within a time window for a series of time windows. An animation is obtained by collecting the individual visualization and displaying them in order by time. This is approach the basis of Ouyang and Revesz [15] who develop an algorithm to generate spatiotemporal cartogram animations.

3.2 Existing COVID-19 Monitoring Systems

In this subsection we review a number of existing systems designed specifically for monitoring the spread of COVID-19. These systems are described below with an emphasis on pointing out their drawbacks thereby motivating our work in developing CoronaViz.

- (1) <https://coronavirus.jhu.edu/> Coronavirus COVID-19 global cases (Johns Hopkins)
- (2) <https://www.healthmap.org/ncov2019/> Novel Coronavirus (COVID-19) outbreak timeline map (HealthMap)
- (3) <https://news.google.com/covid19/map> (Google News)
- (4) <https://hgis.uw.edu/virus/> Novel coronavirus infection map (University of Washington)
- (5) <http://nssac.bii.virginia.edu/covid-19/dashboard/> COVID-19 surveillance dashboard (University of Virginia)
- (6) <https://covid19.who.int/> Novel coronavirus (COVID-19) situation dashboard (WHO)
- (7) <https://www.cdc.gov/coronavirus/2019-ncov/cases-in-us.html> Coronavirus disease 2019 in the US (CDC)
- (8) <https://www.ecdc.europa.eu/en/geographical-distribution-2019-ncov-cases> Geographical distribution of COVID-19 cases worldwide (ECDC)
- (9) <https://www.kff.org/global-health-policy/fact-sheet/coronavirus-tracker/> COVID-19 coronavirus tracker (Kaiser Family Foundation)
- (10) <https://www.worldometers.info/coronavirus/> COVID-19 coronavirus outbreak (Worldometer)
- (11) <https://multimedia.scmp.com/infographics/news/china/article/3047038/wuhan-virus/index.html> Coronavirus: the new disease Covid-19 explained (South China Morning Post)
- (12) <https://storymaps.arcgis.com/stories/4fdc0d03d3a34aa485de1fb0d2650ee0> Mapping the Wuhan coronavirus outbreak (ESRI StoryMaps)
- (13) <https://public.flourish.studio/visualisation/1539110> (Flourish)
- (14) <https://coronavirus.1point3acres.com/en> (1point3acres)
- (15) <https://geods.geography.wisc.edu/covid19/physical-distancing/> (University of Wisconsin)

The Johns Hopkins system tabulates cumulative numbers of confirmed, active, deaths, and recoveries. The cumulative numbers of confirmed and active cases in some of the countries are displayed on the map for some of the larger countries (in terms of area). A drawback of the maps is that zooming in on the map simply increases the resolution of the map but does not show the data for additional countries. This is a common drawback of many of the systems that have been created for visualizing the coronavirus. This is not the case for CoronaViz.

The HealthMap system shows the spread of the disease by tabulating the number of new confirmed cases of the disease on a daily

basis and displaying it with a circle of a particular size and color anchored at the location where it was reported (e.g., a city, state, country, etc.). HealthMap still has the drawback that zooming in only increases the resolution of the map but does not show a finer allocation of the tabulated property to the location.

The Google News system makes use of a map query interface and allows zooming in and reports the variable values for the smaller subunits. It uses a hover operation to yield the variable values for the spatial unit being hovered over, as well as disease-related news at times. It does not have the ability to provide variable values for a combination of units that make up the viewing window when these units are small (e.g., counties) or bigger (countries) as is done in CoronaViz. It is static as it has no temporal component other than precomputed graphs of variable values over a predetermined range of days unlike CoronaViz where the range is set by the user.

The University of Washington system shows the total number of confirmed cases, deaths, and recovered for the countries of the world as one pans the world map. For the US, zooming in has a greater granularity and results in showing how the number of confirmed cases are spatially distributed in each state. Descriptive data is also provided for the confirmed individuals when the region is sufficiently small.

The Flourish system enables the visualization of just one dynamic variable such as the number of confirmed cases in a number of countries at the same instance of time. Although the data is spatially-referenced by name (i.e., the names of the countries) no use is made of a map nor are there any input or output controls. The one advantage of the system is that it is fast which conveys the urgency of the need to stop the spread of the disease.

Both the 1point3acres and Worldometer systems provide comprehensive data and graphs for the dynamic variables but no animation or maps. The dynamic aspect of the variables is captured by the various plots of the variable values and combinations thereof. They make a distinction between cumulative variable values as well as new values. The 1point3acres system prides itself in its data collection ability and is more focused on the virus while the Worldometer system also provides statistics related to the impact of the disease such as unemployment.

The University of Virginia system displays the number of cumulative confirmed cases, deaths, and recovered over time using a time slider. The countries are colored according to the range of the number of individuals for the variable being displayed. Zooming in results in more locations being placed on the map as well as the inconsistent decomposition into smaller units such as states for the US and provinces for China but not for Canada or Australia.

The remaining systems are quite similar in that they only map the number of confirmed cases in each country in the case of the WHO and ECDC systems and in each state for the CDC system. The Kaiser Family Foundation system also maps the deaths. None of the WHO, ECDC, CDC, and the Kaiser Family Foundation systems permit zooming in to get additional data. Non-interactive maps are used to tell the story of the coronavirus outbreak in the South China Post using ESRI StoryMaps. Instead of the disease-related variables some systems like that from the University of Wisconsin look at a variable that monitors the mobility of the population with a map query interface that makes use of cell phone data.

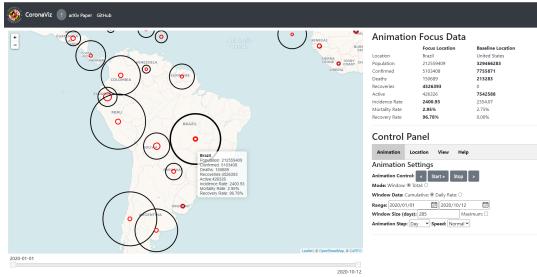
4 USER INTERFACE

CoronaViz's user interface is anchored by the "Control Panel" which is partitioned into four components corresponding to the three tasks of the system which are data animation, location specification, and data viewing, and help. They are accessed by appropriately named buttons.

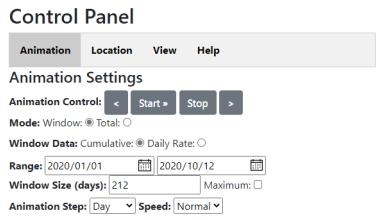
Figure 1a shows the incidence and mortality rates in South America for a 212 day wide window in an animation range spanning from the first of this year 2020 through October 12 of the same year. We see that the query window and the animation range are the same, and thus no animation can be performed as we just have one time instance and an animation requires at least two time instances. Therefore, the result of the query is just a screenshot of the geo-circles each one of which consists of a linearly scaled incidence rate and a linearly scaled mortality rate corresponding to the countries in South America. From the figure we see that the incidence rates are relatively similar for these countries. Mortality rates are much smaller and thus we may wish to scale them so that we can better differentiate between them. Figures 1b-1e show the different control panels and their settings for the query whose result is shown in Figure 1a. The different control panels are described in greater detail in the rest of this section.

The "Animation" button controls the animation process. CoronaViz can be run in two animation modes: "Total" and "Window". In Window mode we provide a temporal region w (termed the "Animation Window") which is specified in terms of days, and a location (i.e., spatial region) which is a geographic entity. In order to simplify the explanation, we use a variant of the example query of animating the progression of COVID-19 in Brazil (See Figure 2 for its result) and its neighboring South American countries in terms of the values of the confirmed cases and deaths disease-related variables. This is done for the "Animation Range" which is set by default to the period between the first of this year 2020 though October 12 of the same year. The animation can provide either the cumulative values of these variables or the daily average value for the days making up the window w . Note that when the window duration is one day, then the cumulative value and the average daily rate are the same. This information is provided on a daily basis on the last day of the animation window for each day of the animation range. In contrast, recall that the maximum possible size of the animation window is the duration of the animation range in which case there is no effective animation as the result is the cumulative value of the variables and the daily average value of the variables over the entire "Animation Range" and only reported on the last day of the animation window. Users don't have to know the value of this maximum as it is specified by checking the "Maximum" checkbox which appears to the right of the "Window Size" in the "Animation Control Panel".

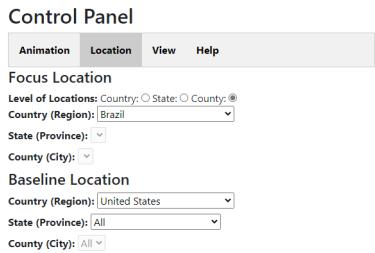
Users who wish to see the cumulative as well as the average daily rates for all of the disease-related variables in an animated manner, should use the "Total" mode. In this case, we do have an animation on a daily basis with the final frame of the animation yielding the cumulative values of the disease-related variables for the temporal "Animation Range" for all spatial ranges that can be viewed. These features are all accessed by clicking on the "Animation" button in



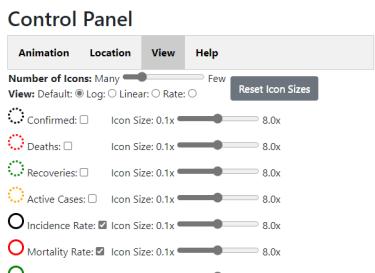
(a) overview



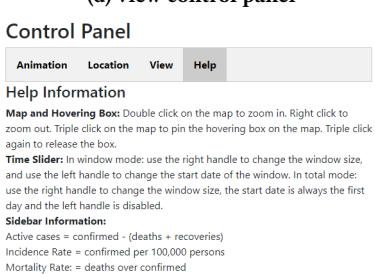
(b) animation control panel



(c) location control panel



(d) view control panel



(e) help information

Figure 1: user interface

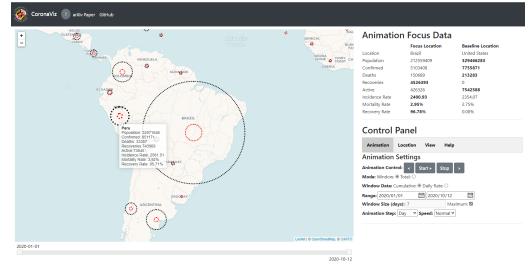


Figure 2: example CoronaViz screenshot for Brazil in South America.

the “Control Panel”. Figure 2 is the final screenshot for the animation of the cumulative values of the number of confirmed cases, deaths and recoveries for the Total mode query for the countries in the vicinity of Brazil for the time period between the first of this year 2020 though October 12 of the same year. In essence, the screenshot differs from the one in Figure 1a for Figures 1b-1e by displaying the values of the disease-related variables instead of the rates. Note the larger geo-circles on account of no normalization which is the case when we used rates for the disease-related variables.

The “Location” button activates the “Location Specification” process which identifies the spatial entity for which we wish to animate and view the disease-related variables. This location is known as the “Animation Focus”. It can be the name of a country/region, state/province, or county/city all of which are obtained from an appropriately named pull-down menu. Alternatively, the geographical entities can also be specified graphically using direct manipulation actions like pan, zoom, and hover. In this case we usually start with a map from which a new map is constructed using pan and zoom operations as well as possibly dilation. Once the desired location has been identified on the map, then a single left click on the mouse is sufficient to initialize or reset the “Animation Focus”.

The advantage of the direct manipulation approach is that it provides the query poser the opportunity to specify the exact shape and boundary, as well as the resolution, of the query region. The “Location” button can also be used in the same manner to set what we call a “Baseline Location” for comparing disease-related data as the animation proceeds. This location can only be set using the pull down menus. It cannot be set using direct manipulation. As the animation proceeds, the values of all of the disease-related variables and rates are displayed side-by-side in the “Animation Focus Data Panel” for the two locations.

At this point, the animation can be started by clicking on the “Start Animation” button in the “Animation Control Panel”. In our example, we see the animation of the progression of COVID-19 in all of the South American countries with a focus on Brazil in terms of the counts of confirmed cases, deaths, and recoveries (see Figure 2).

The “View” button in the “Control Panel” controls the viewing process by providing a number of options of viewing the 7 disease-related rates and variables. The user selects them by clicking the checkbox to their right. Some combinations of variables are predefined such as the “Default” view corresponding to just displaying the incidence and mortality rates, while the “Rate” view corresponds

to just displaying the incidence, mortality, and recovery rates. As the animation proceeds, the values of the selected disease-related variables and rates are displayed using geo-circles anchored at the corresponding geographic location and whose radii provides an indication of their relative magnitudes. We have several options for the radii: linear and logarithmic, neither of which are always satisfactory. Using a "linear" relationship breaks down when radius values differ by more than one order of magnitude. A "logarithmic" relationship is fine for differentiating between small radius values while making it harder to differentiate between large radius values as well as needing to make an appropriate choice of the base (e.g. 2 or natural logarithm instead of 10 which is not good for large radius values). The radii can be scaled by factors ranging from .1X to 8.0X which is useful when the radius values are very large or very small, respectively.

In our example, Brazil is said to be the "Focus Location" which means that as the animation proceeds, users can see additional data for Brazil corresponding to the daily variation of all of the disease-related variables and rates. It is constantly updated by looking at the panel with the heading "Animation Focus Data" to the right of the map which shows Brazil in this case (see Figure 2).

In addition, during the animation, the mouse can be moved over the visible part of the map (termed "hover") and the data associated with closest geo-circle (e.g., Peru in Figure 2) is displayed in what we call the "Hover Box" which is initially anchored on the mouse and moves with the mouse until three quick left clicks are performed at which time the Hover Box" is detached and remains in that position until the performance of the next three quick left clicks. However, even though the hover box has been detached from the mouse, it continues to display the data associated with the nearest geo-circle to the mouse. This nearest geo-circle is highlighted with a thicker outline. Figure 2 shows the result of the animation for a time period since the first of this year which is set in the "Animation Control" panel. Note that the "Animation Control" panel enables users to pause, resume, halt, and restart the animation process by clicking on the appropriate button. Users can also run the animation in a day-by-day manner one day at a time in the forward and backward temporal directions via the buttons labeled "<" and ">", respectively. It is especially interesting to go backwards at the end of the animation by repeatedly clicking on the "<" button found to the left of the "Start/Pause/Resume/Stop" button. The above "playback" can be achieved in a continuous manner by using the mouse to define the width of a window by varying the positions of the left and right tabs of the time slider. This process proceeds by fixing the right tab and varying the left tab as needed. The "playback" is achieved by dragging the left tab in either of the two temporal directions. The right tab is left alone and it follows the motion of the the left tab.

5 EXAMPLES OF THE UTILITY OF CORONAVIZ

in this section we provide use cases of CoronaViz that demonstrate its utility. Notice that we provide both figures and animations. The figures usually correspond to the last screenshot (frame) of an animation. In most cases we also provide a link to video for the entire animation. The animation can be viewed by clicking on the

link in the paper or by cutting the link from the paper and pasting it in the browser (preferably the Chrome or Microsoft Edge browsers).

We first compare the dynamic visualization provided by CoronaViz with conventional methods as used, for example, by newspapers such as the Washington Post for the incidence rates for some of the states in the US (Figure 3) that are two-dimensional graphs where the *x* axis corresponds to the date while the *y* axis corresponds to the value of disease-related variables and rates, Figures 3a, 3b and 3c, show the incidence and mortality rates for the United States during April, July, and September of 2020, respectively. They are screenshots from the entire animation¹. From these three figures we see that there are more confirmed cases in New York than other states in April while New York reported far fewer cases in July. Meanwhile, the number of confirmed cases grows rapidly in the southern and western states. In September, a growing trend of confirmed cases moves to the western and north-central states. This data and the corresponding trends (Figure 3c) are also available on the Washington Post website but it is difficult to draw conclusions about the spatial significance of the relationship between the incidence rates of some spatially adjacent states. Moreover, we can visualize the data of all states on the map in CoronaViz while it is impossible to fit 50 graphs in one page. In addition, CoronaViz can show multiple variables/rates on the map while the graphs could be confusing when many variables/rates are plotted as the graph can support at most two different *y*-axes interpretations (i.e., one on the left and one on the right ends of the *x*-axis).

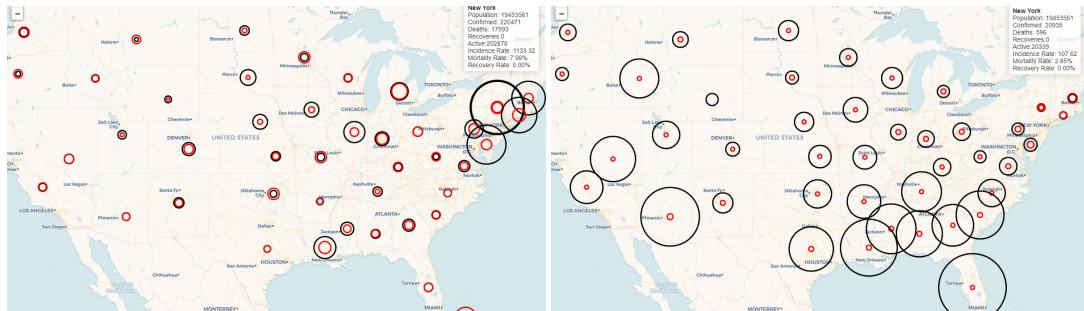
It is often the case that the data values have very small magnitudes as is the case for mortality rates especially when compared with the number of confirmed cases or incidence rates thereby making it difficult to compare their values for different locations. Users can perform more meaningful comparisons of locations with very small data values by changing the scaling factor². For example, Figures 4a and 4c show the mortality rate in the US for September 2020 (window mode) and for Africa in January through October 2020 (Total mode), respectively. It is very small and consequently, the geo-circles representing the mortality rates are also small, which makes it difficult to compare the rates of different locations. By changing the scaling factor of the mortality rate geo-circles, their sizes become larger simultaneously (see Figure 4b and 4d), respectively. After that, the differences in the mortality rate between two locations are clearer. Interestingly, the variation in Africa is much bigger than in the US.

Besides using raw data directly, we often also normalize the data based on population³. Figures 5a and 5c show the number of confirmed cases and deaths in South America in September 2020 (window mode) and in North America in January through October 2020 (Total mode), respectively. Here we see that some countries like the U.S. and Brazil have a large population and thus they have many confirmed cases, which results in geo-circles with large radii when the raw data is plotted directly. After normalization, the values of the confirmed cases are represented by the incidence rate, which is defined as the number of confirmed cases per 100,000 population. The incidence rate is rarely greater than 3,000 and hence the values of the radii of the geo-circles become reasonable

¹https://www.youtube.com/watch?v=UcDjFLa3I_Y

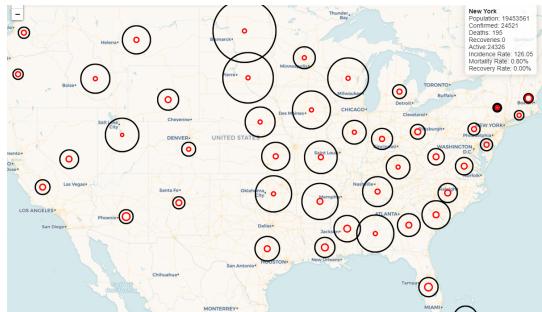
²<https://www.youtube.com/watch?v=VLiWoWtYHQo>

³<https://www.youtube.com/watch?v=cCGWQ4jaChw>

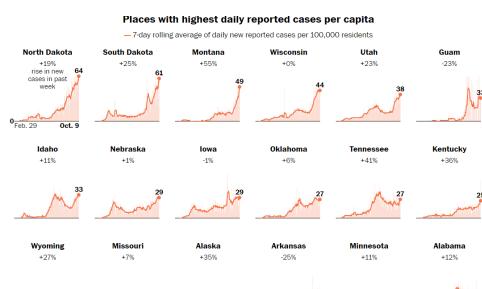


(a) U.S. cases in the Apr. window

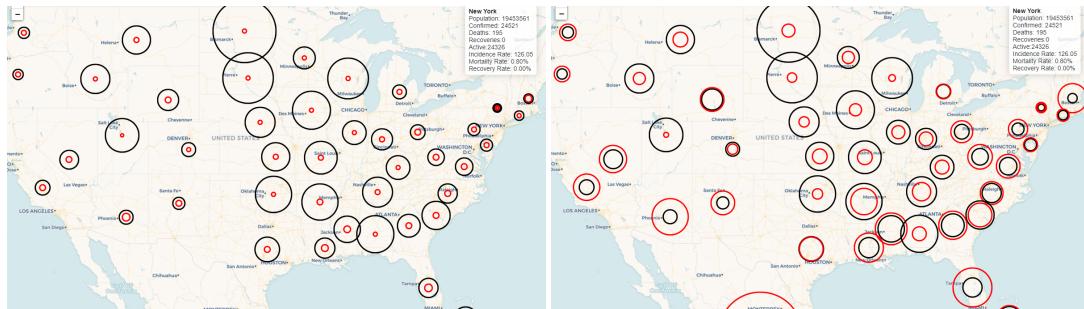
(b) U.S. cases in the Jul. window



(c) U.S. cases in the Sep. window

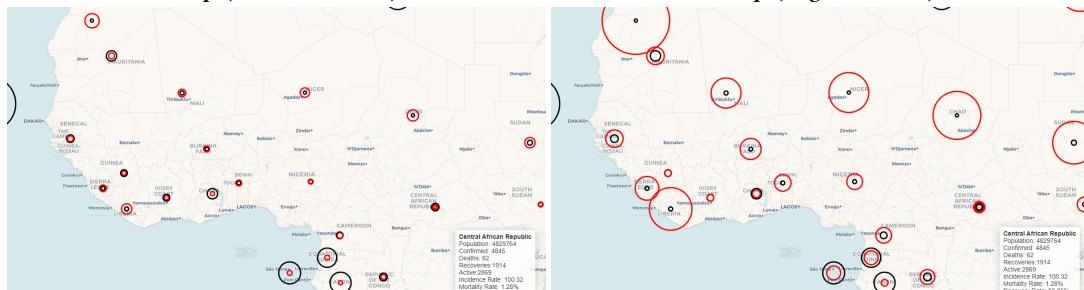


(d) charts from Washington Post

Figure 3: CoronaViz map vs Washington Post charts ([video](#))

(a) U.S. mortality rate in Sep. (normal icon size)

(b) U.S. mortality rate in Sep. (larger icon size)



(c) Africa mortality rate in Jan.-Oct. (normal icon size)

(d) Africa mortality rate in Jan.-Oct. (larger icon size)

Figure 4: choose a proper scaling factor ([video](#))

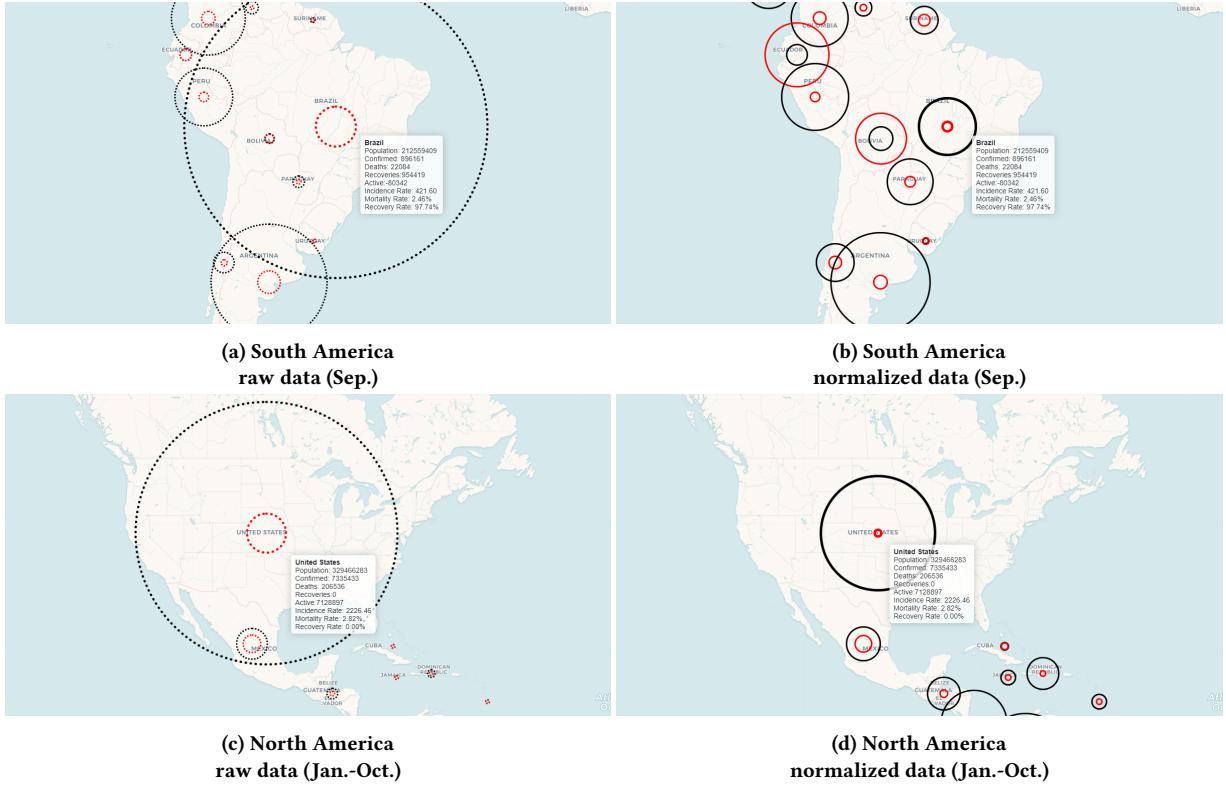
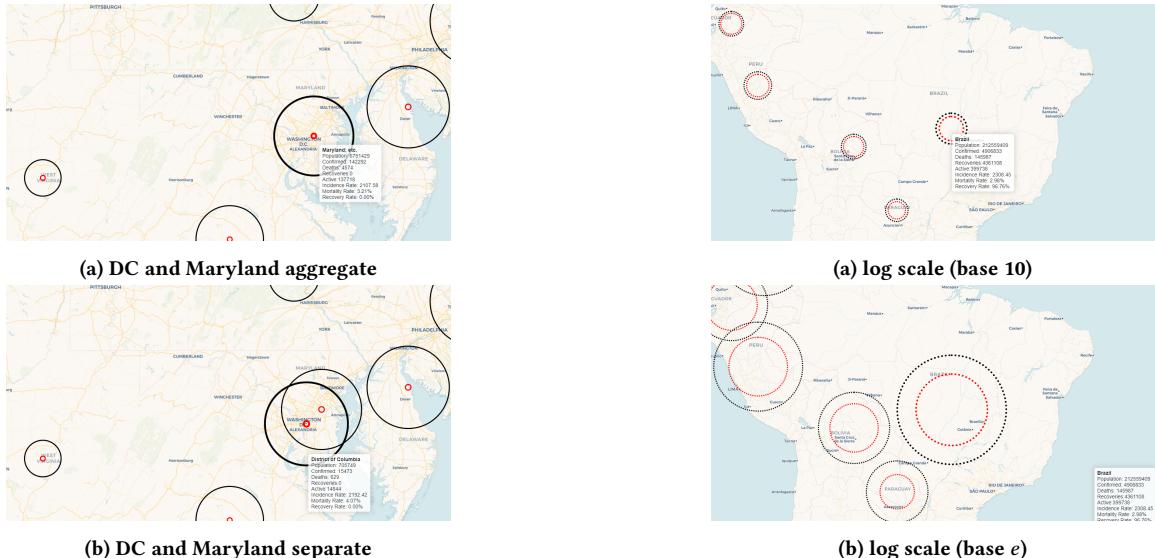
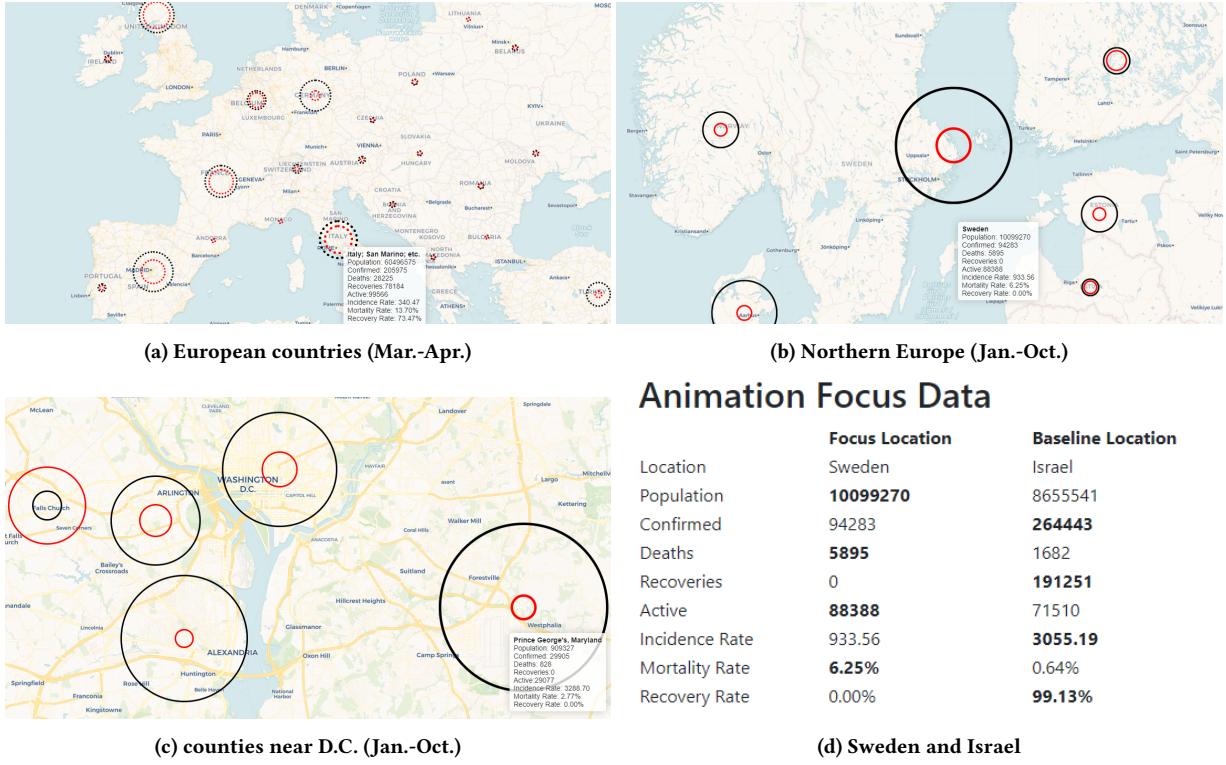
Figure 5: normalization ([video](#))Figure 6: limit number of icons ([video](#))

Figure 7: log scale

after normalization (see Figures 5b and 5d of (window mode) in January through October 2020 (Total mode), which correspond to Figures 5a and 5c, respectively.

Users can control the number of geo-circles on the map⁴. In CoronaViz, geo-circles aggregate automatically if they are close

⁴<https://www.youtube.com/watch?v=DYHk5XmGXKA>

Figure 8: case study ([video](#))

to each other. As shown in Figure 6a, the geo-circles of Washington, D.C. and Maryland usually aggregate unless we zoom in substantially because they are geographically close. By increasing the number of geo-circles plotted on the map, close geo-circles (e.g., D.C. and Maryland in Figure 6b), and more details are shown on the map. This makes comparisons among geographically-proximate locations feasible.

Besides using normalization to reduce the radii of the geo-circles of the disease-related variables, we can also reduce their values by subjecting them to a logarithmic scale. As shown in Figure 7, the geo-circles become smaller when we use the logarithm of the raw values as the radius. However, this also makes the difference between two geo-circles less noticeable. As a result, a smaller base for the logarithm is preferable. For example, a natural logarithm (Figure 7b) is better than the base 10 common logarithm (Figure 7a). Note also that using a small base is equivalent to using a larger scaling factor.

We also study some typical cases to show the utility of CoronaViz. In Europe, the pandemic first peaked in late March to early April. As shown in Figure 8a (confirmed cases and deaths), there were several hot spots. Setting the temporal window to be March and April finds them to be the U.K., France, Germany, Italy, and Spain. Another example is Sweden which let the Coronavirus spread in the hope that the population would develop “herd immunity”. Figure 8b shows the incidence and mortality rates for Sweden and its neighboring countries for January through October 2020 (Total mode). From Figure 8b, we see that Sweden has higher incidence

Animation Focus Data

Location	Focus Location	Baseline Location
Population	10099270	8655541
Confirmed	94283	264443
Deaths	5895	1682
Recoveries	0	191251
Active	88388	71510
Incidence Rate	933.56	3055.19
Mortality Rate	6.25%	0.64%
Recovery Rate	0.00%	99.13%

(d) Sweden and Israel

and mortality rates than its neighboring countries. We can also compare the data through the text information provided in the sidebar⁵. In Figure 8d, we use Israel as a baseline, whose population is close to Sweden. We observe that Israel has a higher incidence rate but a lower mortality rate compared with Sweden. Note that we do not have recovery data than Sweden so it is not shown in Figure 8d. Observe that CoronaViz not only visualizes data of countries but also other administrative divisions like states and counties. For example, some counties near Washington D.C. are plotted in Figure 8c for January through October 2020 (Total mode)

6 CONCLUDING REMARKS AND DIRECTIONS FOR FUTURE RESEARCH

We have seen the utility of animation to keep track of the spread of disease by examining disease-related variables and rates. Our visualization relies heavily on the availability of quantitative data about the presence of the disease provided by the Johns Hopkins University. Additional useful knowledge about the potential progression of the disease can be gained by keeping track of spatially-referenced mentions in news articles as in NewsStand [10, 12, 17], tweets as in TwitterStand [6, 8, 18], documents such as PubMed [13] and ProMED-mail [9, 13], and spreadsheets [1]. This involves geotagging which is the process of recognizing textual references to location as in [14, 16]. Presently we do not make use of such data

⁵<https://www.youtube.com/watch?v=QSkI8htZQQo>

although we do feel that such an approach is a direction for future research.

Note that we have not provided “positivity” data which indicates the percentage of tests that are positive (i.e., the ratio of the number of confirmed cases and the total number of tests). The problem here is that the number of tests is unevenly reported thereby making it impossible to report this rate accurately. We will incorporate this measure in CoronaViz once testing centers adopt more complete reporting procedures that include this data. Finally, another topic for future investigation is normalization by a country’s area.

ACKNOWLEDGMENTS

This work was sponsored in part by the NSF under Grants IIS-18-16889 and IIS-20-41415.

REFERENCES

- [1] M. D. Adelfio and H. Samet. Schema extraction for tabular data on the web. *PVLDB*, 6(6):421–432, April 2013. Also *Proceedings of the 39th International Conference on Very Large Data Bases (VLDB)*.
- [2] W. G. Aref and H. Samet. Efficient processing of window queries in the pyramid data structure. In *Proceedings of the 9th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 265–272, Nashville, TN, April 1990. Also in *Proceedings of the Fifth Brazilian Symposium on Databases*, pages 15–26, Rio de Janeiro, Brazil, April 1990.
- [3] E. Dong, H. Du, and L. Gardner. An interactive web-based dashboard to track covid-19 in real time. *The Lancet Infectious Diseases*, 2020.
- [4] Y. Du, L. Ren, Y. Zhou, J. Li, F. Tian, and G. Dai. Banded choropleth map. *Personal and Ubiquitous Computing*, 22(3):503–510, 2018.
- [5] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley, Reading, MA, second edition, 1990.
- [6] N. Gramsky and H. Samet. Seeder finder - identifying additional needles in the Twitter haystack. In A. Pozdnukhov, editor, *Proceedings of the 6th ACM SIGSPATIAL International Workshop on Location-Based Social Networks (LBSN’13)*, pages 44–53, Orlando, FL, November 2013.
- [7] H. Howard, R. McMaster, T. Slocum, and F. Kessler. Thematic cartography and geovisualization. 2008.
- [8] A. Jackoway, H. Samet, and J. Sankaranarayanan. Identification of live news events using Twitter. In Y. Zheng and M. F. Mokbel, editors, *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location-Based Social Networks (LBSN’11)*, pages 25–32, Chicago, November 2011.
- [9] R. Lan, M. D. Lieberman, and H. Samet. The picture of health: map-based, collaborative spatio-temporal disease tracking. In *Proceedings of the 1st ACM SIGSPATIAL International Workshop on the Use of GIS in Public Health (HealthGIS 2012)*, pages 27–35, Redondo Beach, CA, November 2012.
- [10] R. Lan, M. D. Adelfio, and H. Samet. Spatio-temporal disease tracking using news articles. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on the Use of GIS in Public Health (HealthGIS 2014)*, pages 31–38, Dallas, TX, November 2014.
- [11] Jie Li, Siming Chen, Kang Zhang, Gennady Andrienko, and Natalia Andrienko. Cope: Interactive exploration of co-occurrence patterns in spatial time series. *IEEE transactions on visualization and computer graphics*, 25(8):2554–2567, 2018.
- [12] M. D. Lieberman and H. Samet. Supporting rapid processing and interactive map-based exploration of streaming news. In I. Cruz, C. A. Knoblock, P. Kröger, E. Tanin, and P. Widmayer, editors, *Proceedings of the 20th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 179–188, Redondo Beach, CA, November 2012.
- [13] M. D. Lieberman, H. Samet, J. Sankaranarayanan, and J. Sperling. STEWARD: architecture of a spatio-textual search engine. In H. Samet, M. Schneider, and C. Shahabi, editors, *Proceedings of the 15th ACM International Symposium on Advances in Geographic Information Systems*, pages 186–193, Seattle, WA, November 2007.
- [14] M. D. Lieberman, H. Samet, and J. Sankaranarayanan. Geotagging: Using proximity, sibling, and prominence clues to understand comma groups. In R. Purves, C. Jones, and P. Clough, editors, *Proceedings of 6th Workshop on Geographic Information Retrieval*, Zurich, Switzerland, February 2010. Article 6.
- [15] M. Ouyang and P. Revesz. Algorithms for cartogram animation. In *Proceedings 2000 International Database Engineering and Applications Symposium (Cat. No. PR00789)*, pages 231–235. IEEE, 2000.
- [16] G. Quercini, H. Samet, J. Sankaranarayanan, and M. D. Lieberman. Determining the spatial reader scopes of news sources using local lexicons. In A. El Abbadi, D. Agrawal, M. Mokbel, and P. Zhang, editors, *Proceedings of the 18th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 43–52, San Jose, CA, November 2010.
- [17] H. Samet, J. Sankaranarayanan, M. D. Lieberman, M. D. Adelfio, B. C. Fruin, J. M. Lotkowski, D. Panizzo, J. Sperling, and B. E. Teitler. Reading news with maps by exploiting spatial synonyms. *Communications of the ACM*, 57(10):64–77, October 2014.
- [18] J. Sankaranarayanan, H. Samet, B. Teitler, M. D. Lieberman, and J. Sperling. TwitterStand: News in tweets. In D. Agrawal, W. G. Aref, C.-T. Lu, M. F. Mokbel, P. Scheuermann, C. Shahabi, and O. Wolfson, editors, *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 42–51, Seattle, WA, November 2009.