Hornero: Thunderstorms Characterization using Visual Analytics

Alexandra Diehl[1], Rodrigo Pelorosso[2], Juan Ruiz[2,3,4], Renato Pajarola[1] M. Eduard Gröller[5,6], Stefan Bruckner[7]

[1]University of Zürich, Switzerland, [2]University of Buenos Aires, Argentina, [3]Centro de Investigaciones del Mar y la Atmosfera (CONICET-UBA), [4]Instituto Franco-Argentino sobre Estudios de Clima y sus Impactos (CNRS-UBA-CONICET), [5]Technical University of Vienna, Austria, [6]VRVis, Austria, [7]University of Bergen, Norway

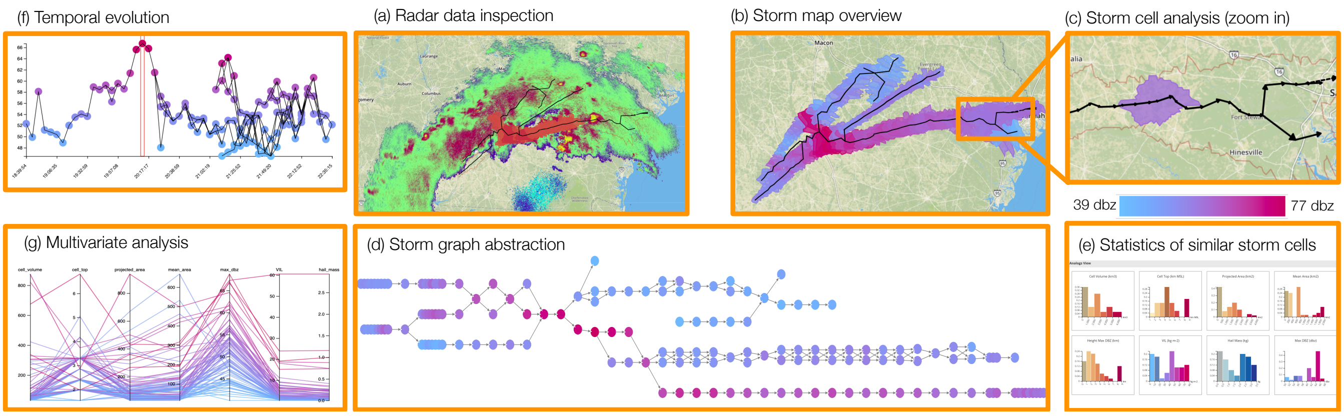

Figure 1: *Hornero: Tornado outbreak March 3rd, 2019, Southeastern US. Hornero shows one of the thunderstorm clusters captured by the radar NEXRAD:KVAX near the city of Valdosta, United States. On the center, (a) shows the intensity of a given storm, (b) the storm trajectories, and (d) the storm graph structure as detected by Titan [DW93]. On the left, (f) shos the temporal evolution of the storm intensity, and (g) a multivariate analysis of all available properties. On the right, (c) shows a detailed view of one storm cell and (e) the probability distributions associated to characteristics of a storm cell to develop (nowcasting).*

Abstract

Analyzing the evolution of thunderstorms is critical in determining the potential for the development of severe weather events. Existing visualization systems for short-term weather forecasting (nowcasting) allow for basic analysis and prediction of storm developments. However, they lack advanced visual features for efficient decision-making. We developed a visual analytics tool for the detection of hazardous thunderstorms and their characterization, using a visual design centered on a reformulated expert task workflow that includes visual features to overview storms and quickly identify high-impact weather events, a novel storm graph visualization to inspect and analyze the storm structure, as well as a set of interactive views for efficient identification of similar storm cells (known as analogs) in historical data and their use for nowcasting. Our tool was designed with and evaluated by meteorologists and expert forecasters working in short-term operational weather forecasting of severe weather events. Results show that our solution suits the forecasters' workflow. Our visual design is expressive, easy to use, and effective for prompt analysis and quick decision-making in the context of short-range operational weather forecasting.

CCS Concepts

• ***Human-centered computing*** → ***Visual analytics; Visualization application domains;***

1. Introduction

According to the UN Office for Disaster Risk Reduction (UNDRR), the indirect economic losses caused by climate-related disasters increased by over 150% during 1998–2017 in comparison to 1978–1997 [WH18]. Among the most prominent high-impact weather events are thunderstorms capable of rapidly developing flooding, large hail and strong surface winds. Forecasting such effects of thunderstorms is still a challenge. During a thunderstorm outbreak, expert forecasters need to analyze in detail a large amount of data. Observation systems (like satellites and radars) provide new information at time frequencies in the order of seconds to minutes and with relevant events taking place simultaneously at different locations over the region of interest. Moreover, some relevant thunderstorm characteristics are difficult to detect and require careful inspection of multidimensional radar features and their temporal evolution over a particular storm cell or cluster. Currently, there are two groups of operational techniques that can be used and combined to provide human forecasters with an adequate guidance for issuing short-time warnings [SXW*13]: numerical weather prediction and data-driven prediction (also known as *nowcasting*). The latter is based on the extrapolation of weather radar data in time using different approaches like optical flow [PNPH*19], tracking of individual convective cells or convective cell clusters [DW93], analog forecasting [AZ15] or deep learning [FSN*19, ASH20]. Analog forecasting and deep learning not only allow nowcasting the storm's movement, but also the anticipation of changes in its intensity or shape.

However, despite the high degree of automation involved in these forecasting techniques, the final decision of issuing a severe-weather warning for a particular region is always taken by human forecasters who detect potentially dangerous thunderstorms and decide which area will be imminently affected by them. In this work, we present a visual analytics framework for the visualization of thunderstorms based on weather radar data, storm tracking by a storm cell identification and tracking software (i.e., Titan [DW93]), and probabilistic forecast generated by an analog-based nowcasting system (e.g., Atencia Zawadzki [AZ15]).

Several efforts have been undertaken in both the meteorology community and the visualization community [RBS*18, AHG*19] to visualize meteorological data. Most of them concern analytical tasks, but not operational weather forecasting. In an operational setup, the human forecaster needs to analyze huge amounts of data in an efficient way, make quick decisions, and communicate fast alerts in the range of minutes to hours. In particular, only a few works [UCAb, UCAa, Nin] address the visualization of very-short-term weather forecasting that takes place between a few minutes and six hours. Although these tools visualize weather radar data and trajectories, there is still a need of integrating storm tracking results and basic radar data and their properties at different levels of abstraction and with enough simplicity to make quick decisions in the context of nowcasting.

To address forecasters' needs, we designed *Hornero*, a web-based visual analytics system that integrates storm tracking data and weather radar data to facilitate the detection and nowcasting of thunderstorms. The novelty of our approach mainly resides in the balanced combination of visualization techniques. The selected techniques are not novel per se, but they are novel in this context of operational weather forecasting. Our contributions consist of:

1. A new visualization framework for operational nowcasting combining overviews of the recent evolution of thunderstorms, storm cells, and cell clusters, seamlessly integrated with a visualization of 2D fields of radar data, interactive multivariate analysis of different storm parameters (e.g., intensity, vertical height, hail production probability, etc.), and past storm cells analysis.
2. A novel visual representation of storms using an abstract graph structure with a geographic cluster-based layout and collapsible nodes to cover a large temporal storm evolution.
3. A visualization overview of analog-based nowcasting results. The overview is performed by the inspection of past storm cells similar to the current state of the latest available time step, or a selected storm cell of interest.

We evaluated our system with domain experts in an iterative process. In the first round, we gathered feedback on our visual design choices and the usability of our tool. This feedback helped us to refine the users' requirements, to improve our design, and to incorporate important functionality needed for the forecasters. We accordingly adjusted our visualization tool, and reevaluated in a second round of interviews the expressiveness, easiness, and effectiveness of our visual design, task abstraction, and provided functionalities. This approach helped us to adjust our visual design to the specifics of the domain tasks, such as rapid detection, prompt analysis, and quick decision making.

2. Related Work

The related work covers two main areas: visualization techniques for spatio-temporal and atmospheric data sets, and visualization tools for nowcasting and issuing of severe weather warnings.

2.1. Visualization Techniques for Spatio-temporal and Atmospheric Data Sets

A recent survey from Afzal et al. [AHG*19] provides a comprehensive overview of the state of the art covering the most popular visual approaches for ocean and atmospheric data sets. The authors present several taxonomies with respect to application area, visualization technique, interaction, and data type. Covered are, for example, visual abstractions and annotations for uncertainty visualization of weather ensembles [ME18, LPCRH18], clustering algorithms [FKRW16, KRRW18], isosurfaces, histograms, and time-series [HMC*13, HMZ*14], among others.

Akiba et al. [AM07] introduced the use of parallel coordinates and time series in the interface for analyzing time-varying multivariate volume data, and studied their usefulness analyzing a hurricane simulation. Inspired by these previous designs, we included parallel coordinate plots (PCP) and time series as part of our design, to help the experts in studying the behavior of storms and their cells. Ma and Entezari [ME18] presented an interactive visualization framework to address the complexity of interpretation of spaghetti plots, for the analysis of uncertainty. They created a visualization named "mode plot" to visually encode high-density clustering results and provide an effective summary of the distribution of ensemble isocontours. Although the focus of this work is on uncertainty analysis and visualization of ensembles, their mode plot abstraction shares similarities with our storm graph structure. However, we differentiate from their work in the visual mapping of 2D clustered regions, named storm structures, into a graph diagram with a geographically clustered and temporally arranged layout.

Kumpf et al. [KTB*17] presented a visual analytics solution to analyze the sensitivity of clustering results with respect to changes of a selected region. Among the visualization components of their solution, they provide 2D maps and spaghetti plots, as well as two abstract views: a cluster-centric view and a member-centric view. The member-centric view shares some similarities with our approach. The authors also used a circular structure and a cluster-based layout based on the two first principal components from a principal components analysis. Contrary to this work, we employ a two level clustering approach. The first clustering pass is done by Titan, which identifies the storm clusters. We show the storm clusters as a connected graph. The second clustering pass is done geographically. We group sub-graphs that are close to each other in geographic space. Later Kumpf et al. [KRRW18] developed a visualization framework based on Met.3D for ensemble sensitivity analysis (ESA) of particular regions, visually encoding statistically coherent regions, automatically tracking them in time, and visualizing the trajectory paths and geospatial evolution of the sensitives. Our approach shares with the work of Kumpf et al. that we extract and represent geographic structures. While they focus on correlation structures, our primary target are storm structures. We also analyze splitting and merging of trajectory paths. However, we do the automated tracking using Titan, and then we post-process and visualize the results. Furthermore, we add a new 2D level of abstraction using a storm graph structure, and we do not use 3D visualizations because they impose a higher cognitive load and therefore are less suitable for an operational forecasting setup.

Andrienko et al. [AAB*13] presented a comprehensive overview of visualization techniques for movement data. Additionally, Andrienko et al. [AAF*15] also presented a novel algorithm for supporting event stream monitoring of spatio-temporal events, their clusters, and their evolution in real-time. The main goal of the paper is the on-the-fly separation of event clusters from noise and the immediate presentation of significant clusters and their evolution. They support the analysis by a visual analytics system with trajectory visualizations, timeline views, and space-time cube visualizations. Contrary to [AAF*15, FKRW16], in our approach, the cell identification, clustering, and tracking of storms us done using Titan's *Storm Cell Identification and Tracking* (SCIT) [DW93]. Moreover, some of the techniques and the general approach for spatio-temporal events do not directly fit our users' needs. For example, 3D visualizations such as isosurfaces used by Ferstl et al. [FKRW16], or the space-time cube [AAF*15] have a high cognitive load. Therefore, they may not be the most suitable option for quick analysis and decision making that operational forecasters need to perform in the context of nowcasting. While Andrienko et al. [AAF*15] and Ferstl et al. [FKRW16] target a similar application area, the main difference is that our visual design and task workflow are focused on decision-making. Our storm graph structure is an abstract representation of the storm clusters as detected by Titan, which facilitates and complements a forecasters' quick analysis and decision making.

2.2. Visualization Tools for Nowcasting and Issuing Severe Weather Warnings

Rautenhaus et al. [RBS*18] covered vast related work in visualization tools used for meteorological data analysis. Some of the most frequently used tools in meteorology include general purpose plotting tools (e.g., matplotlib [Hun07]), and more dedicated visualization tools like GrADS [KI93], VAPOR [Nata, LJP*19], Met.3D [RKSW15], etc. Although these are very powerful tools, they are too complex for the purpose of nowcasting where the focus is on prompt analysis and quick decision making. More specific tools for nowcasting and warning of severe weather hazards include AWIPS [UCAa] (the Advanced Weather Interactive Processing System), a meteorological decoding, display, and analysis package originally developed by the United States' National Weather Service. WarnGen [UCAb] is a tool based on the AWIPS CAVE platform for creating and issuing weather warnings. WarnGen allows for the visualization of radar data and offers storm tracking functionality. However, it does not provide information about the storm structure and topology. NinJo [Nin] is a client-server system for processing and displaying meteorological data. It allows for the visualization of data layers using information coming from station measurements, radar echoes, and model data. NinJo is closest to our work, providing visualization components for the display of Storm Cell Identification and Tracking (SCIT) data. However, it aims to facilitate a distinct set of tasks and thus uses a quite different visual design. We use abstract visualizations such as a directed graph visualization to show storm graph structures, and parallel coordinate plots to easily and quickly provide an overview of storm properties and their relations. Our complementary abstract views have shown to be expressive, easy-to-use, and effective for domain experts. Moreover, NinJo and AWIPS, which have proven to be good tools for analysis and decision making, are currently in use by several national weather services as desktop applications, while Hornero brings the analysis and decision making process to the web.

3. User Tasks

Hornero is designed as a visualization tool for meteorological data to support real-time decision making concerning severe weather warnings. Currently, it is available for weather radar data, but it is extensible to other data types. We work with meteorologists and operational weather forecasters that on a daily basis need to analyze and make quick decisions about severe weather events. The complexity, i.e., spatial resolution, time resolution, and diversity, of data that need to be considered by an expert forecaster to issue a severe weather warning has increased significantly in recent years.

The main goal of Hornero is to provide expert forecasters with a tool that allows them to analyze storms at different levels of abstraction in a limited amount of time. We followed an iterative and participatory design process. We centered our design on the visual analysis of thunderstorms. We collaborated in all stages of the project with a primary domain expert, who is also a co-author of this paper. Then, we interviewed two domain experts for an initial feedback about the visual design. In a second stage, we evaluated our approach with three other domain experts that did not participate in the first interviews. With the help of the domain experts, we identified key tasks that expert forecasters perform on a daily basis, when facing a potential severe weather event associated with thunderstorms (see Figure 2). We summarized the visualization tasks and created a new task workflow based on this information:

T1-Surveillance: visualize data coming from different weather radars to get an overview of which locations are affected by thunderstorms now and will potentially be affected in the near future.
T2-Ranking: visualize additional properties, identify, and rank high intensity storms.
T3-Analysis: visualize further information for a particular storm cell or storm cluster and explore its properties in more detail looking for possible indications of severe weather potential. This task includes the inspection of the recent history of storm cells and storm clusters.
T4-Forecast: based on available nowcasting tools, produce an estimate of the near future evolution of the storm (usually for the next minutes to a few hours) and quantify its uncertainty. Task T4 also implies the decision making itself, generation of reports, and communication to decision makers.

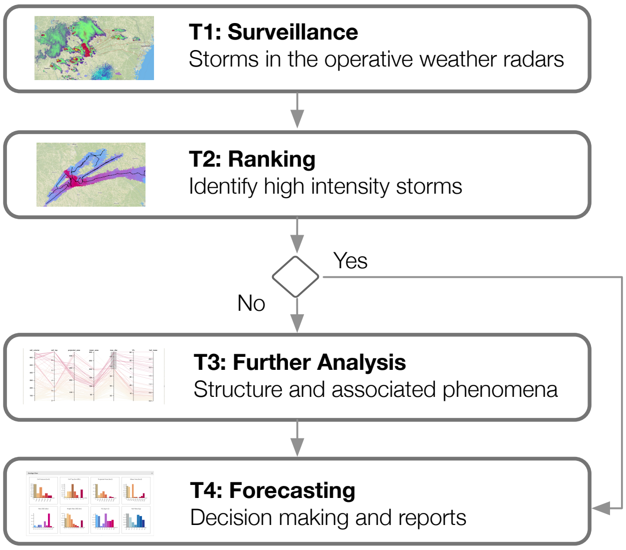

Figure 2: *Hornero's tasks workflow adapted from the domain experts' feedback and task workflow.*

4. Hornero's Working Processes

Hornero is a visualization system that integrates information coming from weather radars, storm identification and tracking systems, as well as nowcasting data through the use of interactive visual analysis.

In the following, we describe the processes and methods used to generate and analyze the data being visualized (i.e., radar data, storm trajectories, and their forecast).

4.1. Storm Identification and Tracking

We use storm identification and tracking data generated by Titan [DW93]. In Titan, a storm structure is described as an object composed of connected pixels in a weather radar reflectivity image (see also Figure 3). Radar reflectivity is a measure of the backscattering produced inside clouds. It depends on the amount of water and its shape contained in the volume illuminated by the radar beam. Titan uses minimum and maximum reflectivity thresholds to identify a storm object also referred to as a *storm cell*. Once the object is defined, several geometric characteristics, like the position of the centroid, its area and orientation are computed. Titan also provides more specific characteristics that are derived from the radar data. These characteristics include storm cell properties such as the maximum cell reflectivity measured in dBZ (Max dBZ), the height of the maximum cell reflectivity (Height Max dBZ), the cell top height (Cell Top), the cell volume (Cell Volume), the cell projected area (Projected Area), the cell mean area (Mean Area), the Vertically Integrated Liquid Water (VIL), and the hail mass (Hail Mass) contained in the storms.

Tracking of storm cells is conducted using an optimization approach in which the solution of the tracking problem is found by minimizing a cost function designed to penalize large storm displacements and abrupt changes of storm properties in time. Particular attention should be paid to the merging and splitting of storm cells since these events are usually associated with relevant physical processes. Titan handles merging events using a two step criterion that involves an extrapolation of terminated trajectories at time t and the superposition of extrapolated trajectories with cells detected at time $t + \delta t$. In the case of splits, all terminated trajectories are extrapolated in time, and if new objects are within the area of the extrapolated objects, then these events are treated as a split. Sometimes storm tracks obtained by Titan result in several short-lived splits followed by merging. This is due to the threshold based cell identification algorithm implemented in Titan. In order to remove these spurious features, when many close storm cells converge into one average centroid, we merge them into one trajectory line.

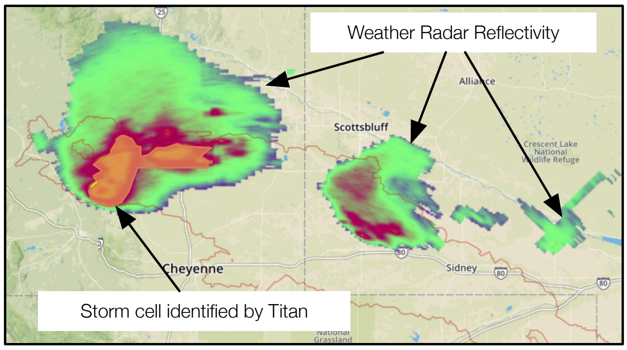

Figure 3: *Radar reflectivity at a specific time step and corresponding to a given storm cluster.*

4.2. Analog-based Nowcasting

Forecasting systems are essential tools on which expert forecasters rely when issuing severe weather warnings. In this work, we generate a probabilistic forecast for the properties of each storm cell or cluster (e.g., maximum reflectivity, hail production potential) based on an analog regression approach inspired by the work of Atencia and Zawadzki [AZ15]. Given the properties of a storm cell detected in the latest available radar data, we search for similar storms in a database of past events. We use a similarity metric based on the storm cell properties provided by Titan and inspired by the SALdEdA (Structure, Amplitude, Location, difference in Eccentricity, and difference in Area) variables and algorithm presented by Shah et al. [SNB15]. In Equation 1, the similarity metric is defined as a weighted sum of the differences between two consecutive storm cells i and j, of volume $dV = |Cell\_Volume_i - Cell\_Volume_j|$, maximum altitude $dAl = |Cell\_Top_i - Cell\_Top_j|$, max reflectivity $dL = |max\_DBZ_i - max\_DBZ_j|$, eccentricity $dE = |Eccentricity_i - Eccentricity_j|$, and mean area $dA = |Area_i - Area_j|$:

$$C_{i,j}(x) = w_1 dV_{i,j} + w_2 dAl_{i,j} + w_3 dL_{i,j} + w_4 dE_{i,j} + w_5 dA_{i,j}. \quad (1)$$

In the current implementation, the variables are equally weighted and normalized. In our approach, the similarity metric relies only on storm properties extracted from radar data, however, other choices are possible including for example a characterization of the storm's environment as in the work of Atencia et al. [AZ15].

Based on the similarity metric, the N most similar storm cells are identified as the analogs. N is a parameter that can be chosen by the user to optimize the method, a value of 20 is used in the examples discussed in this work. The optimal value for this parameter may depend on the database size and on the climatological characteristics of the precipitation systems. Then an ensemble of possible future storm states is obtained from the posterior evolution of the identified analogs. This ensemble provides a forecast for the future storm cell properties and an estimation of its uncertainty, thus allowing for the estimation of the probability associated with future events (e.g., an intensifying storm, an enhancement of hail production potential, etc.). Since the focus of our work is on the integrated visualization of radar data, trajectory data, and nowcasting data, the analog technique is implemented in a rather simplified way. The database of past storm cells and clusters detected by Titan consists of three months of radar data. Given the characteristics of the analog regression, this is not large enough to provide meaningful results in operational applications, but is sufficient to show the advantages of the developed visualization framework.

5. Visualization Design

Our visualization task workflow is based on the task workflow recommended by the Warning Decision Training Division, NOAA, National Weather Service [fEIb], and the specific task workflow identified with our collaborating domain experts (see Figure 2). We adapted their workflow and defined visualization tasks based on Brehmer and Munzner's typology of tasks [BM13].

5.1. Visualizing Radar Reflectivity

In task T1 (*Surveillance*), forecasters first need to see the horizontal distribution of radar reflectivity and other properties derived from the radar data as a basis for their analysis. They progressively add properties and associated parameters such as intensity, presence of hail, topological structure, and others, to understand the severity of the storms. To provide them with a familiar overview of the thunderstorms, we designed a *storm map visualization*. By using the storm map, users can focus on the visualization of radar reflectivity, and overlay extra information as part of the data presentation context. Figure 3 shows the radar reflectivity as a basis for the analysis (focus) and the contour lines corresponding to the complete storm evolution (context). Forecasters can add other properties such as cell volume, hail, etc., as new layers to the map. The user can click on any other geolocated area inside the storm contour lines and see the identified storm cells. Additionally, using a *storm animation* slider, the user can follow the complete storm evolution. The domain expert can also switch the analyzed property through elements in the user interface. Currently, the application can visualize most of the properties computed by Titan. We selected complementary color schemes that are sufficiently different to distinguish between weather properties such as intensity, volume, and hail mass, following well-established guidelines for color visualization of environmental variables [Dat, QM15]. We use the HCL Color Advisor [ZFH*20] for generating colorblind-safe color schemes. We chose a colormap visualization for this task because (1) storms are inherently spatial and (2) our experts are highly trained in spatial visual analysis. They can quickly identify changes on the map, make rough estimations of distances between storm cells or time steps (for example identify rapidly growing storms), and quickly review the temporal evolution of storms over a geographic area, such as areas of recently developed storms. To visualize the storm cell areas, we employed color and contour lines, because our domain experts are familiar with these visual variables, which they utilize on a daily basis.

5.2. Visualizing Storm Tracking Paths

The visualization of storm paths can reveal important aspects of a storm, related to its potential for generating severe weather conditions. Persistent and strong storm cells are often responsible for several high-impact weather events. Also, the split of a storm cell may be indicative of its potential to become a "supercell" [Natb], which is a kind of thunderstorm that is frequently associated with high-impact weather events such as large hail, heavy precipitation, and tornados. The occurrence of merging events can reveal the upscale growth of individual convective cells into storm clusters. These are known as mesoscale convective systems, sometimes associated with events of widespread strong winds and heavy precipitation. Therefore, this view is useful for tasks T2 (*Ranking*) and T3 (*Further Analysis*). We designed a *trajectory view* using the information of storm cells and storm cell paths pre-processed by Titan [DW93]. To visualize the storm paths, we use gray polylines and employ opacity to indicate the passage of time, and line thickness to visualize acceleration. We decided not to use color again for the trajectories, or other visual variables such as texture to reduce the cognitive load of each visual component as required by domain experts. We found simple polylines appropriate to depict storm trajectories since most of the times, thunderstorms follow smooth paths with soft turns. Figure 4 shows an example of a conglomerate of storms that bifurcates to the NE and develops into a supercell storm.

5.3. Identifying Interesting Storms

Forecasters need to quickly identify storms that can turn into hazards, for example based on their large volume, presence of hail, or high intensity. For task T2, we describe them as *interesting storms*.

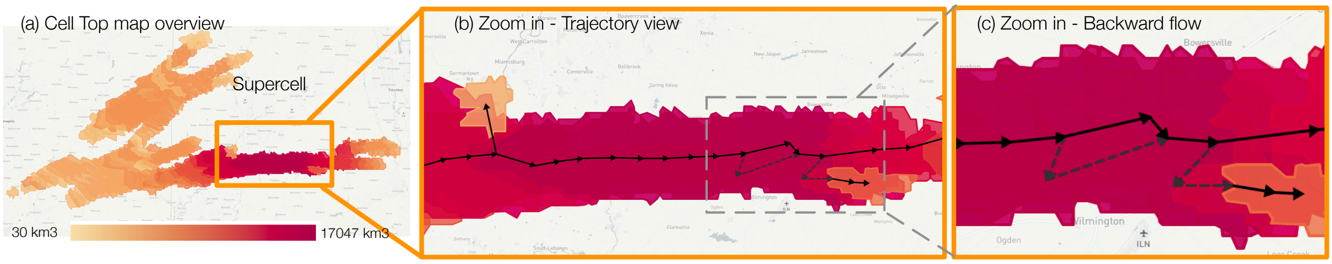

Figure 4: *(a) Storm height overview (cell_top) and two zoom ins (b) and (c) of the trajectory view for one of the storm branches identified by Titan during the tornado outbreak, May, 27, 2019. (c) shows abrupt changes of direction in the storm path, depicted as black dashed arrows.*

We found parallel coordinates plots (PCP) to be a suitable option since they visualize pairwise relationships between multiple storm properties. We designed the *characteristics view* that allows forecasters to easily filter multiple properties of potentially dangerous storms and to inspect potential relations between them. Figure 5(c) shows a filtered geolocated branch.

5.4. Analyzing Storm Splitting and Merging Behavior

Analyzing the splitting and merging of storms together with other properties such as intensity allows forecasters to gain a better understanding of the storm development. Questions concerning which branches grow faster, are new, die, or merge with other branches can be answered. The analysis task T3 helps the experts to identify and focus on the storms that are developing into dangerous phenomena. For example, under certain conditions, the transition of a storm cell into a supercell is preceded by a storm bifurcation. The advantage of the storm graph structure is that the splitting and merging of storm cells can be seen and identified at a glance by the analysts. We designed a graph layout using information of geographic clusters, as it is shown in Figure 6. Storm and storm branches that are closer geographically will be closer in the graph as well, and vice-versa. The designed layout, named cluster-based graph layout, helps the user to easily identify storm branches that are geographically close to each other. This layout is supported by the perceptual Gestalt law of proximity. It especially helps when the storm structure is large, and contains multiple branches coming from different geographic areas. Moreover, the graph provides the option to collapse consecutive time steps into one single glyph. This feature is particularly useful when dealing with large temporal storm structures (see Figure 6). Each graph node is colored by using information of a selected storm property under analysis (e.g., maximum reflectivity, hail production potential, etc.). We developed our visual design with a focus on simplicity and aesthetics provided by the Gestalt laws of proximity (clustered layout), similarity (color encoding), and continuity (collapsible features). By filtering a subset of interesting storm cells, the analyst can inspect their properties, temporal evolution, and topological structures. More details of the algorithms implemented for the storm-graph structure can be found in the appendix.

5.5. Analyzing the Temporal Evolution of Storms

The domain experts expressed the need for simple visualizations as a baseline for their analyses. We provide a line-chart component, named *temporal-evolution view*. It depicts the temporal evolution of storm cells as detected by Titan, for any of the available storm properties. The storm map visualization, the graph-storm structure, and the temporal-evolution view share the same color encoding, associated to the property in analysis. For example, Figure 7(a) depicts the hail mass evolution over time. This view displays the temporal development of a tornado outbreak, presenting hail mass on different branches with a peak at 02:54 UTC, and a small peak at 03:17:54 UTC. The peaks can be visualized on the map for further analysis. By looking at the horizontal axis, it is possible to detect splits and joins of different storm cells that share the same time but have different properties values (see Figure 7(a) and (b)). Together with the temporal animation, these views help to reconstruct the complete evolution of the storms in analysis.

5.6. Forecasting

Expert forecasters need to communicate potentially dangerous storm structures that can transform into severe hazards as outlined in T4 (*Forecasting*). To support this critical task, we developed a visualization component named *analogs view*. The analogs view consists of series of histograms displaying the frequency distribution of future storm properties estimated from its analogs, as defined in Section 4. The main purpose of the analogs view is to enable the expert forecaster to evaluate how likely it is that the current storm cell will develop into a dangerous phenomenon, based on the analysis of its analogs. For the $i-th$ analog, we consider its time evolution after the time frame it was identified as an analog of the current storm under analysis. Then we compute the maximum value of a given storm property during that time period ($P(i)_{max}$). Finally, we construct the histogram of $P(i)_{max}$. We selected a histogram visualization because it is a familiar visualization for the domain experts. The color encoding chosen for the histogram is the same as the color encoding chosen for each available storm property. The bin containing the value of the property for the current storm under analysis is indicated by a shaded/textured bar. This encoding helps to rapidly compare the expected future values of the property, as provided by the analogs, with the current value. Our visualization component supports all ensemble-based techniques that provide samples of the possible storm evolution. The probability

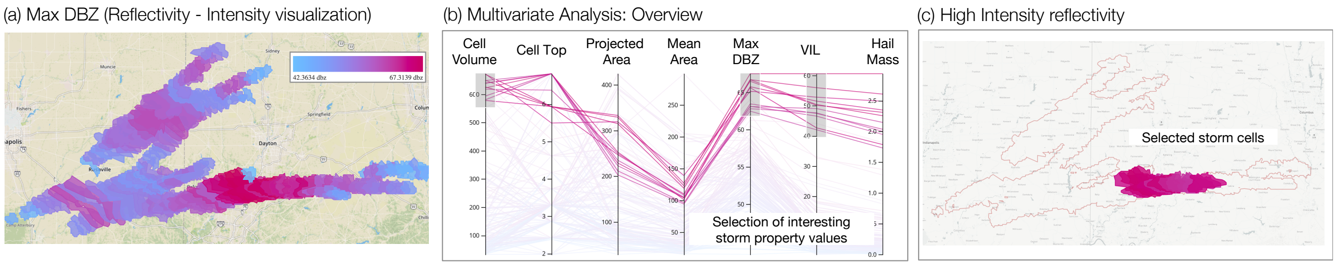

Figure 5: *Using the Characteristics view (b), forecasters can filter storms to analyze the May 27, 2019 outbreak. The left plot (a) shows a storm intensity overview (Max_DBZ). (b) shows the selection of cell volume, intensity (Max_DBZ), and VIL for filtering potentially dangerous storm cells. The right plot (c) shows the resulting query on the map by brushing on the Characteristic view.*

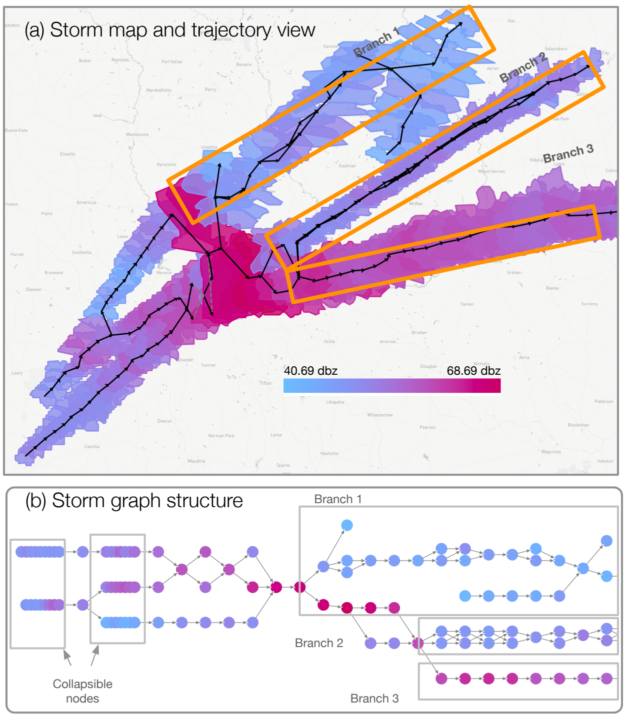

Figure 6: *Storm structure overview. (a) shows the storm map and trajectory view revealing three storm structures with SW-NE direction, (b) shows the corresponding storm-graph structure. Highlighted are the three aforementioned branches. Large chains of nodes belonging to the same storm branch are collapsed*

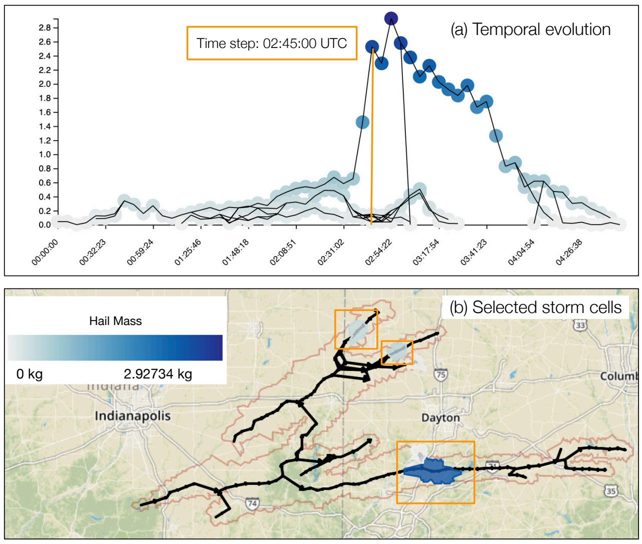

Figure 7: *Temporal-evolution view. Tornado outbreak, May, 27, 2019. The timeline shows at time 02:45:00 UTC three storm cells with a significant estimated hail mass. The most significant storm cell, highlighted in dark blue, can be observed near Dayton, one of the areas most affected by the tornados.*

distribution function can be computed from them and future values for these properties can be approximated.

6. Implementation

Hornero is composed of a backend server and a frontend client, which consumes the information provided by the server (see Figure 8) The server provides a set of API endpoints where requests can be made from the frontend. The basic endpoints provide information about storm structures, storm cells (or nodes), and edges. The frontend is written in Javascript, jQuery, and D3.js. It has several well-defined and extendable view components. Each visualization component can also open a dialog for presenting information. Hornero uses *Leaflet* [Vla] for rendering the underlying map. Elements that are drawn on top of the map are composed of native leaflet objects. The visualization endpoints can, when displaying information in their corresponding dialog window, use the rendering backend that is best suited. We use weather radar data coming from the Next-Generation Radar (NEXRAD) stations provided by the NOAA [fEIa]. NEXRAD provides radar reflectivity data with a spatial resolution of 460 km since June, 1, 1991. For the present study, we use data from selected cases on March 3, 2019, captured by the radar NEXRAD:KVAX near the city of Valdosta, and on May, 27, 2019, captured by the radar NEXRAD:KILN near Cincin-

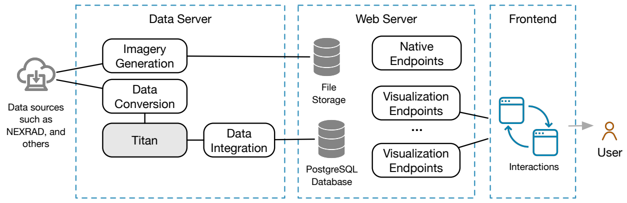

Figure 8: *System architecture. Hornero has a client-server architecture, with visualization components containing frontend and backend layers. It provides endpoints for the retrieval of basic information, such as storm properties and time steps. This information is consumed by a frontend which provides different views and interactions to the user.*

nati, Ohio. The source code and web application are available at http://stormtrack.cg.tuwien.ac.at/.

7. Use Cases

We analyze a tornado outbreak that took place on March 3rd, 2019 over the course of approximately six hours, with a total of 41 tornados spreading over Alabama, Georgia, Florida, and South Carolina. We selected this event due to its devastating power and constitution with several storm clusters happening simultaneously [Sto].

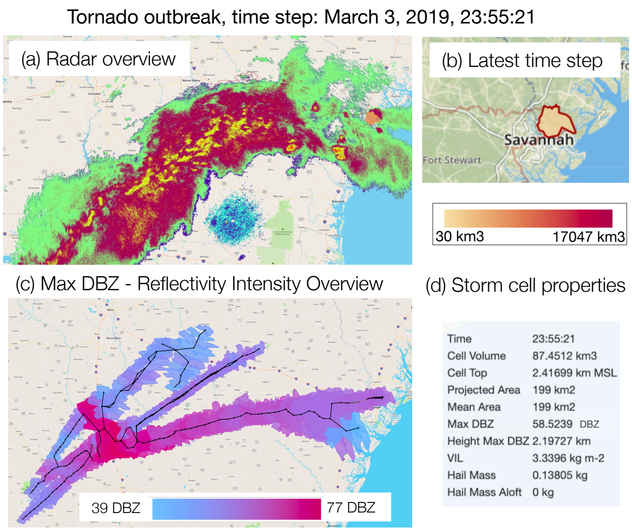

Figure 9: *Thunderstorm that extends towards Florida's coast near Savannah. An overview of the radar data for the latest time step (March 03, 2019 23:55:21) can be seen. (a) the radar reflectivity corresponding to the selected time step, (b) the storm cell detected by Titan at the selected time step, (c) the complete reflectivity intensity evolution (max dbz), and (d) the storm cell properties.*

We first look at the latest radar data, corresponding to T1 (*Surveillance*). The storm map view enables the analysis of radar data coverage, as shown in Figure 9(a), with Figure 9(b) showing the location of a selected storm cell at the same time step as in Figure 9(a). By using the storm animation along the contour lines, we

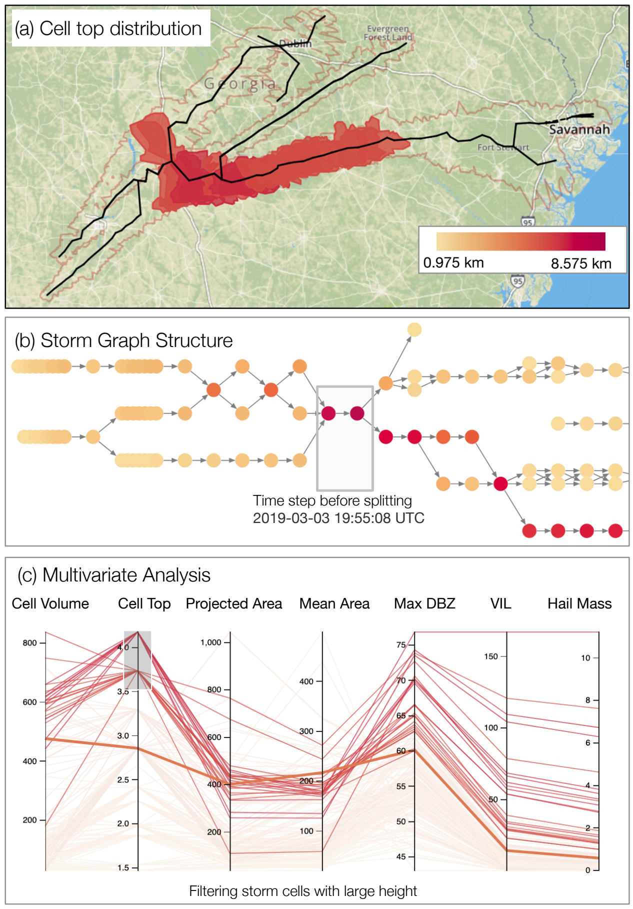

Figure 10: *Analysis of storm cells that can reach the highest altitudes: (a) shows the cell top distribution after applying a filtering to the characteristics view, (b) and (c) show the time step before the bifurcation into the Northeast and the Southeast branch that leads to the development of high altitude storms. All views are linked and color encoded based on the selected variable.*

continue with the analysis of the storm development. The storm-system path starts to the north of Tallahassee. This is possible to see by using the trajectory view shown in Figure 9(c) and Figure 10(a).

By analyzing the storm map and trajectory view, facilitating T2 (*Ranking*), it becomes apparent that most of the storm structures maintain a SW-NE trajectory. However, the southernmost branch slowly turned into a WSW-ENE trajectory possibly indicating a transition into a thunderstorm supercell as shown in Figure 10(a). Its trajectory consisted of two branches that touched each other at 19:50:31 (see Figure 10(b)), and then split again into other three branches: one of them heading towards Georgia, another one towards Evergreen Forest Land, and the Southeast one towards the coast near Savannah. The integrated analysis of reflectivity fields and storm tracking results, by using the storm map together with the trajectory view and storm graph structure, helps the expert to efficiently verify if the behaviour detected in a particular storm cell represents a relevant aspect of the evolution of the storm, or just a

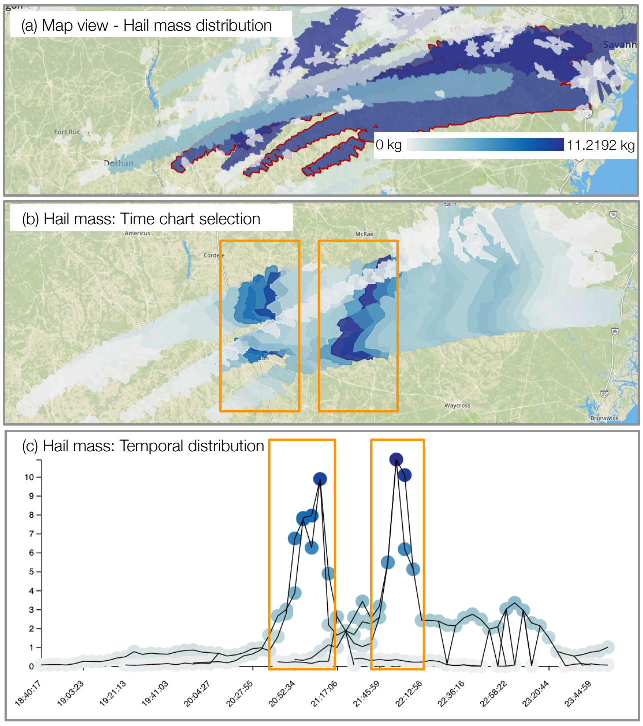

Figure 11: *(a) Hail mass development during the tornado outbreak, March 3rd, 2019. (b) Analysis of hail mass in a thunderstorm. Several independent branches joined and converged into a big structure. (c) Temporal evolution of the selected thunderstorm showing two peaks with high concentrations of hail mass.*

spurious effect associated with the limitations of the tracking system. The interactive visual analysis using the multiple coordinated views offers additional information for the detection of important features in the evolution of storms. For example, a change in the cell movement combined with an intensification in terms of reflectivity provides more certainty on the possible transition into a supercell structure. This analysis corresponds to task T3 (*Further Analysis*).

Figure 10 reveals that after splitting of the storm, the southeast branch gains height. This information adds to the high intensity shown in the storm intensity visualization in Figure 9(c). Higher storms are associated with stronger updraft, which also increases the potential of the storm to produce severe weather events such as large hail or strong surface winds. This phenomenon can also be seen in Figure 10(b) as the chain of storm cells that increases in height (cell top), highlighted in the storm map. This investigation corresponds to T3 (*Further Analysis*) and T4 (*Forecasting*) tasks.

We continued with the analysis of potential large hail in the thunderstorm. Figure 11(a) overviews of the development of hail mass over time and space. Figure 11(b) shows a detailed view for the selected thunderstorm. By the interactive analysis of the temporal evolution view and the storm map view, we notice two peaks associated with a large hail potential (see Figure 11 (b) and (c)). The color encoding and interactive brushing over both views facilitates the quick spatio-temporal analysis of hazardous events, in this case hail. The coordinated and linked views facilitate the analysis of consistency in the evolution of different storm properties. For example, an increasing hail production potential associated with an increasing height or storm intensity provides more certainty to the analysis. If all these attributes were to be analyzed using individual views at different times, the task would be highly time consuming.

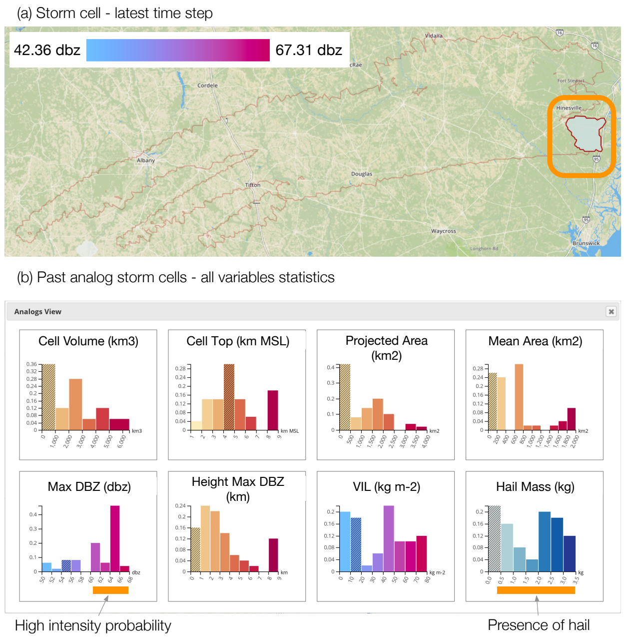

Figure 12: *Probability distributions associated to the latest time step (23:55:21 UTC) affecting the Florida coast near Savannah: (a) shows the selected storm cell used to search for past analogs while (b) shows the frequency distributions among analogs, $P(i)_{max}$, for cell volume, cell top, projected area, mean area, maximum reflectivity, height of maximum reflectivity, vertically integrated liquid and hail mass, corresponding to the storm cell in (a).*

The primary intended use of *Hornero* is the rapid investigation of possible relevant storm cells by expert forecasters for warning issuing applications. Domain experts indicated that the system can also be useful in the context of a post-factum analysis of past storms. One application is the validation of warnings issued in the past by comparing them with the observed storm trajectories and behaviour. Hornero enables the efficient identification of the spatial and time locations associated with intense storms, thus providing valuable information for the comparison with the time frame and region included in a particular warning. Moreover the characteristics and trajectory view can help to characterize storm behaviour prior to the occurrence of a high impact weather. This visual interactive features provide valuable data to elaborate better forecasting rules or better variables that can be useful for forecasting thunderstorms (e.g., by defining better metrics for analog selection).

Finally, we chose the latest radar data available for the selected storm cell (see Figure 12(a)). We visualize the potential hazard of this cell by showing the probability distributions of similar past storm cells (see Figure 12(b)). The probability distributions indicate a high likelihood that the storm will continue to grow in intensity and that hail mass is present. The histograms shown in Figure 12(b) present a summary of how analogs behaved in the past. The forecaster can use this information to forecast the behavior of the selected storm cell (*Forecasting* task).

Analog-based forecasting produces a large amount of data including several possible future storms evolution for each particular storm detected by Titan. The histogram efficiently summarizes this information avoiding a time consuming analysis of each individual analog. This is a crucial aspect of the design since nowcasting of storms usually is issued for lead times in the order of a few minutes.

8. Domain Expert Evaluation

We evaluated our visual design using two rounds of semi-structured interviews with expert meteorologists at the National Weather Services (NWS) in Argentina, all of them with more than three years in operational weather forecasting. One of them is the head of the operational forecasting team at the NWS, and has strong experience in its operational workflow, two of them work as operational forecasters, and another two are researchers working at the NWS as well. We followed the guidelines by Lam et al. [LBI*12] and Sedlmair [Sed16] to sharpen the focus and context of our evaluations. Both interview rounds consisted of a pre-experiment questionnaire, pair analytics session, and post-experiment questionnaire. The interviews lasted approximately one hour. The evaluation form is provided as supplementary material. In the first round of interviews the goals were: (1) to validate the domain experts' requirements previously identified by working with our main domain expert, (2) to evaluate our design choices, and (3) to test the usefulness of our prototype. We presented the initial prototype to two independent domain experts working in operational weather forecasting. During the interviews we followed a "think-aloud" protocol where the interviewees could freely explore the tool and provide us with their feedback. We adapted our visual design and functionalities based on this feedback. In the second round of interviews, three weather forecasters of the NWS participated, including two males and one female researcher, all of them working on operational weather forecasting on a daily basis. The objective was to evaluate the improvements of our visual design, and assess its expressiveness, easiness, and effectiveness. Our solution introduced methodological changes to the weather forecasting workflow, such as the analogs view, which is a first step towards forecasting thunderstorm evolution based on analogs. The interviews were done via video conference with screen and audio recording. We gave the interviewees access to the tool's URL to use it. We recorded their screens and audio with minimal intervention from our side to reduce possible biases. Afterwards, the participants completed an offline questionnaire and provided detailed feedback about the tool. Domain experts found the tool suitable for their daily task workflows. Results from the interviews showed that domain experts found the visualizations very to extremely expressive, very easy, and very useful. They also mentioned in the questionnaire that the tool could be extended to at least the following scenarios:

1. To perform real-time analysis and post-factum analysis. The domain experts foresee the use of our solution in the storm analysis process, prior to issuing alerts or warnings, and in post-factum situations to analyze data and verify alerts issued.
2. To monitor the meteorological situation and to carry out an analysis of the severity potential.
3. To estimate the near-future displacement of thunderstorms within a particular region.

In the future our collaborators would like to evaluate the tool with real-time data, and add additional data sets coming from other sensors (e.g., satellites) and numerical weather prediction models. To do such an evaluation will require the development of high-performance computing algorithms, for example, to query in real-time large historical data sets, and to retrieve multiple storm properties from past situations interactively. Regarding the data volumes, the Nexrad II data for a month requires 5.4GB, while the Titan's post-processed output about 540MB. Currently, the geospatial storm structure is queried through a PostgreSQL database using PostGIS, and the responses are obtained at interactive rates. Future work will include an extensive performance evaluation of the analog-based nowcasting using a larger database. Our tool is extensible and allows for the integration of new data sets and new visualizations components, as described in Section 6.

9. Lessons Learned and Conclusions

Some of the lessons learned during our project reinforce previous research [DPD*15, DPD*17, AHG*19] and discussions during the recent IEEE VIS 2019 "Application Spotlight" session on "Visualization in Meteorology & Climate Sciences" [IEE19]. Visual design for quick and high-impact decision making requires user-centered, easy, expressive, and effective visualizations. Following these guidelines, we focused on the domain experts' requirements, and tried to understand their task workflows (see Section 3). Iterating over two rounds of interviews was very useful to refine our design choices. The participants responded positively and provided constructive feedback, as discussed in Section 8. In summary, all experts stated that the tool is suitable at many levels to complement their daily work. A next challenge will be to integrate *Hornero* into their operational infrastructure that includes working with their radar data sources, additional data sets, and current tools in use.

In conclusion, this paper presents a complete visual analytics framework designed for thunderstorm analysis and characterization, visual storm tracking, and nowcasting based on analogs. Our main contributions are (1) a set of interactive views (the storm map view, characteristics view, temporal evolution view, and storm animation) to overview storms and quickly identify high-impact weather events, (2) a novel storm graph visualization to inspect and analyze the storm structure, and (3) an overview visualization of analog-based nowcasting results.

10. Acknowledgements

This work was partially supported by the UZH Digital Society Initiative, the MetaVis project (#250133) funded by the Research Council of Norway, the VRVis funded in COMET (879730), a program managed by FFG, PICT 2033-2017, from the National Agency for the Promotion of Science and Technology, Argentina, and 20020170100504BA, by the University of Buenos Aires.

References

[AAB*13] ANDRIENKO G., ANDRIENKO N., BAK P., KEIM D., WROBEL S.: *Visual Analytics of Movement*. Springer Science & Business Media, 2013. 3

[AAF*15] ANDRIENKO N., ANDRIENKO G., FUCHS G., RINZIVILLO S., BETZ H.-D.: Detection, tracking, and visualization of spatial event clusters for real time monitoring. In *IEEE International Conference on Data Science and Advanced Analytics* (2015), pp. 1–10. doi:10.1109/DSAA.2015.7344880. 3

[AHG*19] AFZAL S., HITTAWE M. M., GHANI S., JAMIL T., KNIO O., HADWIGER M., HOTEIT I.: The state of the art in visual analysis approaches for ocean and atmospheric datasets. *Computer Graphics Forum 38*, 3 (July 2019), 881–907. doi:10.1111/cgf.13731. 2, 10

[AM07] AKIBA H., MAY K.-L.: A tri-space visualization interface for analyzing time-varying multivariate volume data. In *Proceedings IEEE-VGTC Symposium on Visualization* (January 2007), pp. 115–122. doi:10.2312/VisSym/EuroVis07/115-122. 2

[ASH20] AYZEL G., SCHEFFER T., HEISTERMANN M.: Rainnet v1.0: A convolutional neural network for radar-based precipitation nowcasting. *Geoscientific Model Development 13*, 6 (June 2020), 2631–2644. URL: https://gmd.copernicus.org/articles/13/2631/2020/, doi:10.5194/gmd-13-2631-2020. 2

[AZ15] ATENCIA A., ZAWADZKI I.: A comparison of two techniques for generating nowcasting ensembles. part ii: Analogs selection and comparison of techniques. *Monthly Weather Review 143*, 7 (July 2015), 2890–2908. doi:10.1175/MWR-D-14-00342.1. 2, 4, 5

[BM13] BREHMER M., MUNZNER T.: A multi-level typology of abstract visualization tasks. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (December 2013), 2376–2385. doi:10.1109/TVCG.2013.124. 5

[Dat] DATA SCIENCE AT SCALE. LOS ALAMOS NATIONAL LABORATORY: NSciVisColor: Color Tools and Strategies for Scientific Visualization. https://sciviscolor.org. Accessed: 2020-04-15. 5

[DPD*15] DIEHL A., PELOROSSO L., DELRIEUX C., SAULO C., RUIZ J., GRÖLLER M. E., BRUCKNER S.: Visual analysis of spatio-temporal data: Applications in weather forecasting. *Computer Graphics Forum 34*, 3 (July 2015), 381–390. doi:10.1111/cgf.12650. 10

[DPD*17] DIEHL A., PELOROSSO L., DELRIEUX C., MATKOVIĆ K., RUIZ J., GRÖLLER M. E., BRUCKNER S.: Albero: A visual analytics approach for probabilistic weather forecasting. *Computer Graphics Forum 36*, 7 (October 2017), 135–144. doi:10.1111/cgf.13279. 10

[DW93] DIXON M., WIENER G.: Titan: Thunderstorm identification, tracking, analysis, and nowcasting—a radar-based methodology. *Journal of Atmospheric and Oceanic Technology 10*, 6 (December 1993), 785–797. doi:10.1175/1520-0426(1993)010<0785:TTITAA>2.0.CO;2. 1, 2, 3, 4, 5

[fEIa] FOR ENVIRONMENTAL INFORMATION. N. N. C.: Noaa national weather service (nws) radar operations center (1991): Noaa next generation radar (nexrad) level 2 base data. https://data.nodc.noaa.gov/cgi-bin/iso?id=gov.noaa.ncdc:C00345. Accessed: 2020-04-15. doi:10.7289/V5W9574V. 7

[fEIb] FOR ENVIRONMENTAL INFORMATION N. N. C.: Warning decision training division. https://training.weather.gov/wdtd/. Accessed: 2020-04-30. 5

[FKRW16] FERSTL F., KANZLER M., RAUTENHAUS M., WESTERMANN R.: Time-hierarchical clustering and visualization of weather forecast ensembles. *IEEE Transactions on Visualization and Computer Graphics 23*, 1 (August 2016), 831–840. doi:10.1109/TVCG.2016.2598868. 2, 3

[FSN*19] FORESTI L., SIDERIS I. V., NERINI D., BEUSCH L., GERMANN U.: Using a 10-year radar archive for nowcasting precipitation growth and decay: A probabilistic machine learning approach. *Weather and Forecasting 34*, 5 (October 2019), 1547–1569. doi:10.1175/WAF-D-18-0206.1. 2

[HMC*13] HÖLLT T., MAGDY A., CHEN G., GOPALAKRISHNAN G., HOTEIT I., HANSEN C. D., HADWIGER M.: Visual analysis of uncertainties in ocean forecasts for planning and operation of off-shore structures. pp. 185–192. doi:10.1109/PacificVis.2013.6596144. 2

[HMZ*14] HÖLLT T., MAGDY A., ZHAN P., CHEN G., GOPALAKRISHNAN G., HOTEIT I., HANSEN C. D., HADWIGER M.: Ovis: A framework for visual analysis of ocean forecast ensembles. *IEEE Transactions on Visualization and Computer Graphics 20*, 8 (February 2014), 1114–1126. doi:10.1109/TVCG.2014.2307892. 2

[Hun07] HUNTER J. D.: Matplotlib: A 2d graphics environment. *Computing in Science & Engineering 9*, 3 (May/June 2007), 90–95. doi:10.1109/MCSE.2007.55. 3

[IEE19] IEEE VISUALIZATION CONFERENCE: Visualization in Meteorology and Climate Sciences: Recent Research and Open Challenges. http://ieeevis.org/year/2019/info/application-spotlights, 2019. Accessed: 2020-04-30. 10

[KI93] KINTER III J. L.: The grid analysis and display system (grads): A practical tool for earth science visualization. In *Proceedings Colorado University, Applied Information Systems Research Program (AISRP) Workshop 3* (January 1993). URL: https://ntrs.nasa.gov/citations/19950027810. 3

[KRRW18] KUMPF A., RAUTENHAUS M., RIEMER M., WESTERMANN R.: Visual analysis of the temporal evolution of ensemble forecast sensitivities. *IEEE Transactions on Visualization and Computer graphics 25*, 1 (August 2018), 98–108. doi:10.1109/TVCG.2018.2864901. 2, 3

[KTB*17] KUMPF A., TOST B., BAUMGART M., RIEMER M., WESTERMANN R., RAUTENHAUS M.: Visualizing confidence in cluster-based ensemble weather forecast analyses. *IEEE Transactions on Visualization and Computer Graphics 24*, 1 (August 2017), 109–119. doi:10.1109/TVCG.2017.2745178. 2

[LBI*12] LAM H., BERTINI E., ISENBERG P., PLAISANT C., CARPENDALE S.: Empirical studies in information visualization: Seven scenarios. *IEEE Transactions on Visualization and Computer Graphics 18*, 9 (September 2012), 1520–1536. doi:10.1109/TVCG.2011.279. 10

[LJP*19] LI S., JAROSZYNSKI S., PEARSE S., ORF L., CLYNE J.: Vapor: A visualization package tailored to analyze simulation data in earth system science. *Atmosphere 10*, 9 (August 2019), 488. URL: http://dx.doi.org/10.3390/atmos10090488, doi:10.3390/atmos10090488. 3

[LPCRH18] LIU L., PADILLA L., CREEM-REGEHR S. H., HOUSE D. H.: Visualizing uncertain tropical cyclone predictions using representative samples from ensembles of forecast tracks. *IEEE Transactions on Visualization and Computer Graphics 25*, 1 (August 2018), 882–891. doi:10.1109/TVCG.2018.2865193. 2

[ME18] MA B., ENTEZARI A.: An interactive framework for visualization of weather forecast ensembles. *IEEE Transactions on Visualization and Computer Graphics 25*, 1 (August 2018), 1091–1101. doi:10.1109/TVCG.2018.2864815. 2

[Nata] NATIONAL CENTER FOR ATMOSPHERIC RESEARCH (NCAR): VAPOR: The Visualization and Analysis Platform for Ocean, Atmosphere, and Solar Researchers. https://www.vapor.ucar.edu. Accessed: 2020-04-28. 3

[Natb] NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION. NATIONAL WEATHER SERVICE: What is a Supercell? . https://www.weather.gov/ama/supercell . Accessed: 2020-12-04. 5

[Nin] NINJO WORKSTATION: NinJo Workstation. https://www.ninjo-workstation.com. Accessed: 2020-04-28. 2, 3

[PNPH*19] PULKKINEN S., NERINI D., PÉREZ HORTAL A. A.,

VELASCO-FORERO C., SEED A., GERMANN U., FORESTI L.: Pysteps: an open-source python library for probabilistic precipitation nowcasting. *Geoscientific Model Development 12*, 10 (October 2019), 4185–4219. doi:10.5194/gmd-12-4185-2019. 2

[QM15] QUINAN S. P., MEYER M.: Visually comparing weather features in forecasts. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (August 2015), 389–398. doi:10.1109/TVCG.2015.2467754. 5

[RBS*18] RAUTENHAUS M., BÖTTINGER M., SIEMENM S., HOFFMAN R., KIRBY R. M., MIRZARGAR M., RÖBER N., WESTERMANN R.: Visualization in meteorology—a survey of techniques and tools for data analysis tasks. *IEEE Transactions on Visualization and Computer Graphics 24*, 12 (December 2018), 3268–3296. doi:10.1109/TVCG.2017.2779501. 2, 3

[RKSW15] RAUTENHAUS M., KERN M., SCHÄFLER A., WESTERMANN R.: Three-dimensional Visualization of Ensemble Weather Forecasts - Part 1: The visualization tool Met.3D (version 1.0). *Geoscientific Model Development 8*, 7 (July 2015), 2329–2353. URL: https://www.geosci-model-dev.net/8/2329/2015/, doi:10.5194/gmd-8-2329-2015. 3

[Sed16] SEDLMAIR M.: Design study contributions come in different guises: Seven guiding scenarios. In *Proceedings Workshop on Beyond Time and Errors on Novel Evaluation Methods for Visualization* (December 2016), pp. 152–161. doi:10.1145/2993901.2993913. 10

[SNB15] SHAH S., NOTARPIETRO R., BRANCA M.: Storm identification, tracking and forecasting using high-resolution images of short-range x-band radar. *Atmosphere 6*, 5 (May 2015), 579–606. doi:10.3390/atmos6050579. 5

[Sto] STORM PREDICTION CENTER. NOAA / NATIONAL WEATHER SERVICE NATIONAL CENTERS FOR ENVIRONMENTAL PREDICTION: Day 4-8 Severe Weather Outlook Issued on Feb 28, 2019. https://www.spc.noaa.gov/products/exper/day4-8/archive/2019/day4-8_20190228.html. Accessed: 2020-04-15. 8

[SXW*13] SUN J., XUE M., WILSON J. W., ZAWADZKI I., BALLARD S. P., ONVLEE-HOOIMEYER J., JOE P., BARKER D. M., LI P.-W., GOLDING B., XU M., PINTO J.: Use of nwp for nowcasting convective precipitation: Recent progress and challenges. *Bulletin of the American Meteorological Society 95*, 3 (July 2013), 409–426. doi:10.1175/BAMS-D-11-00263.1. 2

[UCAa] UCAR UNIDATA: AWIPS: Advanced Weather Interactive Processing System. https://www.unidata.ucar.edu/software/awips2/. Accessed: 2020-04-28. 2, 3

[UCAb] UCAR UNIDATA: WarnGen Walkthrough - Unidata AWIPS User Manual. http://unidata.github.io/awips2/cave/warngen/. Accessed: 2020-04-30. 2, 3

[Vla] VLADIMIR AGAFONKIN: Leaflet: An open-source JavaScript library for mobile-friendly interactive maps. https://leafletjs.com. Accessed: 2020-04-24. 7

[WH18] WALLEMACQ P., HOUSE R.: UNISDR and CRED report. economic losses, poverty and disasters (1998–2017). Brussels: Centre for Research on the Epidemiology of Disasters (CRED), 31, October 2018. doi:10.13140/RG.2.2.35610.08643. 1

[ZFH*20] ZEILEIS A., FISHER J. C., HORNIK K., IHAKA R., MCWHITE C. D., MURRELL P., STAUFFER R., WILKE C. O.: Colorspace: A toolbox for manipulating and assessing colors and palettes. *Journal of Statistical Software 96*, 1 (March 2020), 1–49. 5

Appendix: Algorithms

The algorithms implemented to create the visual component *storm graph structure* consist of: (1) a cluster-based layout that groups nodes vertically depending on their spatial closeness, and (2) a collapsible feature that simplifies large consecutive trains of storm cell nodes, in the temporal axis (horizontally). These trains do not present splitting or merging branches.

For the clusterization of the storm branches, the algorithm groups nodes based on the DBScan clustering technique (see Algorithm 1). When computing the layout of the graph, the nodes that correspond to a same geographic cluster are grouped together in the storm graph structure. The layout is created considering larger spaces between clusters of nodes, namely storm branches, and making storm branches geographically distant, also farther distant from their sibling clustered storm branches in the graph.

Algorithm 1: Clustering of nodes in each time step

In: timesteps (list of dates), stepsByDate
Result: Steps clusters

$clustersByStepId \leftarrow \{\}$
$clustersByDate \leftarrow \{\}$
for all $data \in timesteps$ **do**
  $steps \leftarrow stepsByDate[date]$
  $clusters \leftarrow getDBScanClusters(pointsForStep)$
  $clustersList$.push($clusters$)
end for
return $clustersList$

The collapsible feature (see Algorithm 2), starts with no potential interval (indicated by *intervalEnd* and *IntervalStart* set to -1). The first iteration of the loop checks if the last time step could be the end of a new potential interval. For this case, nodes should not be the result of a join (Line 8). The next iteration will try to grow the interval by setting a new start for the interval, or by making it grow if *intervalStart* is not -1. The condition for adding the time step is that its nodes do not result from a join or generate a split (conditions in Line 5 and Line 12). If these conditions are not met, then the interval can be closed if it contains more than one time step. At this point, no potential interval exists and so *intervalEnd* and *intervalStart* are set to -1, so a new end can be searched in the next iteration.

Algorithm 2: Assembling collapsible node chains in the graph

In: dates (list of dates), nodes by date
Result: An array of intervals that can be collapsed

1: $intervalEnd \leftarrow$ -1
2: $intervalStart \leftarrow$ -1
3: $intervals \leftarrow$ []
4: **for** di = dates.length - 1 TO 0 **do**
5:   $eys \leftarrow$ do all nodes in time step di have less that one parent and one child?
6:   **if** $intervalEnd = -1$ **then**
7:     $ps \leftarrow$ do all nodes in time step $di+1$ have less than one parent?
8:     **if** eys and ps **then**
9:       $intervalEnd \leftarrow di$
10:     **end if**
11:   **else**
12:     $is \leftarrow$ do all nodes in time step $d-1$ have less than one child?
13:     **if** eys and is **then**
14:       $intervalStart \leftarrow di$
15:     **else**
16:       **if** $intervalStart < intervalEnd$ and $intervalStart \neq -1$ **then**
17:         $newInterval \leftarrow \{intervalStart, intervalEnd\}$ $intervals$.push($newInterval$)
18:       **end if**
19:       $intervalEnd \leftarrow -1$
20:       $intervalStart \leftarrow -1$
21:     **end if**
22:   **end if**
23: **end for**
24: **if** $intervalEnd \neq -1$ and $intervalStart \neq intervalEnd$ **then**
25:   $newInterval \leftarrow \{0, intervalEnd\}$ $intervals$.push($newInterval$)
26: **end if**
27: return $intervals$