

Visually Comparing Weather Features in Forecasts

P. Samuel Quinan and Miriah Meyer

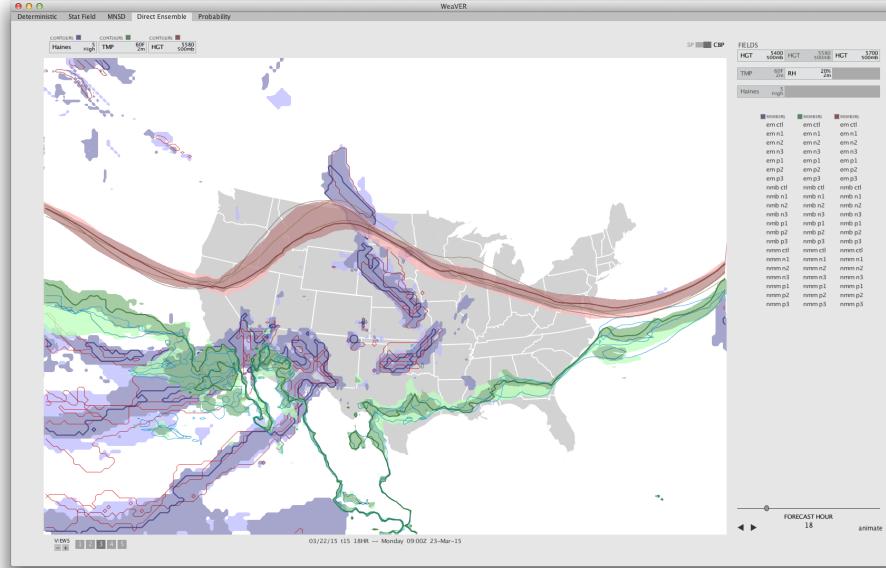


Fig. 1: An overview of the interface for WeaVER, an open-source tool developed for supporting meteorological analysis, shown here visually relating multiple isocontour features across an ensemble using contour boxplots.

Abstract— Meteorologists process and analyze weather forecasts using visualization in order to examine the behaviors of and relationships among weather features. In this design study conducted with meteorologists in decision support roles, we identified and attempted to address two significant common challenges in weather visualization: the employment of inconsistent and often ineffective visual encoding practices across a wide range of visualizations, and a lack of support for directly visualizing how different weather features relate across an ensemble of possible forecast outcomes. In this work, we present a characterization of the problems and data associated with meteorological forecasting, we propose a set of informed default encoding choices that integrate existing meteorological conventions with effective visualization practice, and we extend a set of techniques as an initial step toward directly visualizing the interactions of multiple features over an ensemble forecast. We discuss the integration of these contributions into a functional prototype tool, and also reflect on the many practical challenges that arise when working with weather data.

Index Terms—Design study, weather, geographic/geospatial visualization, ensemble data.

1 INTRODUCTION

A wide variety of domains depend on weather predictions for making critical decisions, such as wildfire response, avalanche prediction, and hurricane evacuation. Meteorologists working in these domains make predictions based on numerically simulated forecasts, the outputs of which include many different variables and time points. Furthermore, as with most numerical simulations, various sources of error lead to inherent uncertainty in the resulting forecasts. To account for some of the uncertainty, forecasts can use multiple simulations to sample the space of possible outcomes, creating an ensemble of results for each variable and time point.

Because the simulations are large and computationally expensive, only a few large governmental and intergovernmental agencies run and distribute the majority of the forecast simulations. The resulting data is very large, on the order of hundreds of gigabytes for a single day's forecasts, which has fostered a prevalence of tools and third-party or-

ganizations that create static visualizations of the forecasts for use and distribution. The challenge for the meteorologists who use these visualizations is that these tools and third-party organizations produce visualizations with vastly different visual conventions, many of which go against well-known visualization principles, and seldom offer support for exploring the uncertainty in the simulations.

To better understand these challenges, we conducted a two-year design study that involved meteorologists in decision-making contexts across a variety of application areas ranging from wildfire prediction to air quality assessment. We identified two common visualization challenges. The first challenge stems from meteorologists attempting to mentally integrate information from sets of visualizations with inconsistent and even conflicting visual encodings. The second challenge is the current, limited support for working with ensemble forecasts, which includes no effective methods for the direct comparison of multiple features across an ensemble.

In addressing these two challenges we provide several contributions: a characterization of both the problems and data associated with meteorological forecasting, a concise treatment of which we were unable to locate in the existing body of visualization research; a data driven formulation of **informed, default** encoding choices that integrate existing meteorological conventions with good visualization principles; and, as a secondary contribution, the extension of state-of-the-art-techniques for visualizing ensembles to enable the direct com-

• P. Samuel Quinan is at the University of Utah. E-mail: psq@cs.utah.edu.
• Miriah Meyer is at the University of Utah. E-mail: miriah@cs.utah.edu.

Manuscript received 31 Mar. 2015; accepted 1 Aug. 2015; date of publication xx Aug. 2015; date of current version 25 Oct. 2015.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

parison of multiple meteorological features. We evaluated these contributions with our collaborators by integrating them into WeaVER, a proof-of-concept system for weather visualization encoding recommendation packaged into an open-source tool for visualizing weather forecasts, shown in Figure 1. Finally, we reflect on several challenges we encountered working in the domain of meteorology and discuss several pitfalls.

2 DESIGN PROCESS

Over the course of the project we worked closely with meteorologists from a variety of domains. Our collaborators included: two meteorologists supporting resource allocation decisions for wildfire emergency management in the southwest region of the United States; a meteorologist working with air quality simulations for Utah’s state-level regulatory Department of Air Quality; a meteorologist in the Science and Technology Infusion Division of the National Weather Service; and a professor in the Department of Atmospheric Sciences at the University of Utah who specializes in cold-weather mountain climatology and runs a popular public facing blog about winter weather and ski conditions along the Wasatch Front.

Our design process was heavily influenced both by Sedlmair et al.’s nine-stage framework [31] and by the processes and recommendations for human centered design in geo-visualization outlined by Lloyd and Dykes [19]. We began with a series of contextual interviews where we observed the daily workflows of several of our collaborators. These formative interviews shaped our initial domain problem characterization. Due to our lack of background knowledge in meteorology, we sought assistance from the Atmospheric Sciences Department at the University of Utah, which led us to several other collaborators. A series of additional interviews with these individuals led to insights that added significantly to our understanding of the tasks and workflows of meteorologists.

We then moved on to a prototyping phase that began with the parallel development of multiple wireframe prototypes. After internal review, the designs from our parallel prototyping session were integrated into a full scale paper prototype that we presented to our collaborators for feedback. We then created additional digital prototypes that focused separately on the ideas of informed, default encodings and directly visualizing multiple features across ensembles. An overview of these wireframe, paper, and digital prototypes is included in Supplemental Materials. We subsequently presented the digital prototypes to our collaborators in three separate feedback sessions. The feedback on these prototypes was integrated into the development of WeaVER, the open-source tool that we developed as a full-scale proof of concept for evaluating our proposed ideas, described in Section 7. WeaVER was, in turn, presented to our collaborators for additional tweaks and final evaluation as we discuss in Section 8.

3 FOUNDATIONS

In this section we provide an overview of weather forecasts, as well as the tasks and workflows of our collaborators in meteorology. We also discuss the current role of visualization in these workflows, along with the areas where improved visualization methods are needed.

3.1 Problem Characterization

In general, a *weather forecast* refers to one or more outputs of numerical weather prediction simulations. The simulations are run using various models, parameterizations, input conditions and boundary conditions in order to simulate different sets of variables over varying geographical extents, predictive time frames, and grid resolutions. To ensure that meteorologists have the most up to date information, many of these simulations are run multiple times throughout the day.

There are two main types of forecasts: deterministic forecasts and ensemble forecasts. *Deterministic forecasts* are the output of a single simulation. *Ensemble forecasts*, on the other hand, are a collection of two or more simulation outputs that cover overlapping geographical extents and predictive timeframes, but use different models, different parameterizations, or different initial or boundary conditions. The simulations comprising an ensemble are generally run at lower

resolutions than their deterministic counterparts due to limited computational resources. Because each simulation, or *member*, within an ensemble represents a possible forecast outcome, the ensemble as a whole can be thought of as sampling the space of possible forecast outcomes, providing a mechanism for approximating uncertainty in the simulation process. This uncertainty may be due to the propagation of initial error through the simulation, the use of approximations in modeling the underlying physical processes, or numerical error within the computation itself. These various sources of error grow over the predictive time frame, propagating to larger and larger scale features, thereby increasing the divergence among different simulations [32].

Due to the associated computational complexity, many organizations that rely on forecasts have neither the time nor the resources to run their own weather simulations. As such, the majority of the weather forecast simulations used today are run by a small number of large governmental and intergovernmental organizations such as the United States’ National Oceanic and Atmospheric Administration (NOAA) and the European Centre for Medium-Range Weather Forecasts (ECMWF) — we collectively refer to these organizations as gatekeepers [31]. For the many meteorologists who rely primarily or solely on the forecasts run and disseminated by these gatekeepers, predefined choices of models, variables, and grids become limiting factors in all weather predictions and decisions.

The workflow of most meteorologists centers around the tasks of locating and relating specific *features*. Features generally correspond to events, trends, or boundaries: a cold front, a low pressure system, the freezing point temperature boundary, etc. Decisions are rarely based on a single feature in the data; rather, they are based on the intersection of multiple features. The threshold for critical wildfire conditions, for example, equates to surface temperatures above 60°F, surface relative humidity under 20%, and sustained surface winds over 20 mph [33]. Meteorologists also rely heavily on experiential knowledge in making judgements based on features. When a simulation’s resolution does not accurately resolve mountainous terrain, for example, the resulting forecasts will fail to show major precipitation events; yet one of our collaborators can often still accurately predict those precipitation events. By looking at the same forecasts day after day, meteorologists begin to recognize patterns in how the simulations’ biases relate to their specific problems and begin to account for those biases in their decision-making process.

We observed that meteorologists generally use visualizations in order to understand the big-picture status of a forecast. Many meteorological visualizations, however, fail to present the forecast data in a way that enables effective visual comprehension of feature relationships. As we will go on to discuss, this is largely due to the problematic usage of visual encodings, along with unsatisfactory methods for relating features across ensembles. As such, meteorologists are often forced look at a large number of visualizations to locate all the features relevant to their current problem, mentally aggregate those features, and then use that aggregated mental model to make their judgements [35]. As one collaborator summarized, “*Forecasters are dealing with a firehose of data, and they need to be able to process it fast.*” Pattern recognition plays a critical role in this regard, but only if an individual can look at a consistent set of visualizations over time.

While a number of different tools and systems exist for visualizing forecast data, there are also many third-party organizations who take the same data, derive additional data products, and release static visualizations over the web. These visualizations are often created using the same or similar tools to those meteorologists already have access to. In some cases, however, these third-party visualizations provide access to new or experimental research not yet integrated into available forecasting tools, such as derived probabilistic predictions for dry lightning [12]. Other times the data has been post-processed or bias-corrected to target specific meteorologists’ problems, as is the case for a large number of visualizations generated by the Storm Prediction Center in Norman, OK [6]. Sometimes the third-party visualizations are simply easier to use or more readily available outside the office. Whatever the reason, bookmarks and browser tabs for accessing a combination of these static visualizations factor heavily into the

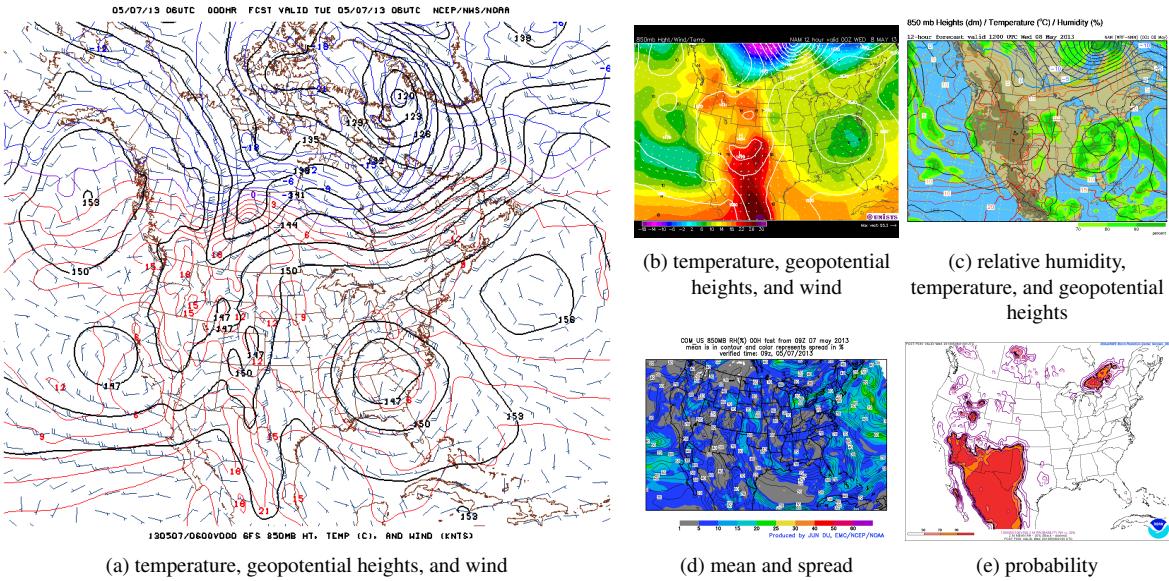


Fig. 2: Examples of existing static forecast visualizations that show various meteorological conventions for encoding combinations of variables.

The images in 2a, 2d, and 2e are reproduced courtesy of various subsidiaries of NOAA and the US National Weather Service: the National Center for Environmental Protection (NCEP), NCEP’s Environmental Modeling Center, and the Storm Prediction Center, respectively. The image in 2b is reproduced courtesy of Unisys Weather, while the image in 2c is reproduced courtesy of the US National Center for Atmospheric Research (NCAR).

workflows of the majority of our collaborators.

As Figure 2 illustrates, different variables within a forecast are visualized using a variety of encodings. The visualizations often layer multiple encodings to allow for the comparison of multiple variables and their constituent features of interest. Unfortunately, when showing multiple variables simultaneously, these visualizations often combine encodings in problematic ways, as evidenced by the usage of color in Figure 2c. Additionally, the static nature of the visualizations often leads to significant visual clutter, like in Figures 2a and 2d. An expanded discussion of these issues is presented in Section 5.1.

The notion of visually relating features and feature relationships is more complicated when dealing with ensembles. An ensemble of possible simulated outcomes provides an ensemble of possible instantiations for each feature of interest. Visualization methods for *directly* examining the space of possible features across an ensemble are currently limited to looking at isocontour-based features. Moreover, these visualization methods are ineffective for tasks that involve relating multiple features. As such, meteorologists only examine the interactions among multiple features across the ensemble *indirectly*, looking at how the features relate under some derived-data transformation. The transformations are usually statistical in nature: averaging the possible outcomes, taking the maximum or minimum possible values, calculating the probability of an event based on how many ensemble members predict it, etc. Unfortunately, these transformations often mask outliers, gradient based features, and edge cases that are otherwise significant [33]. Additionally, the resulting derived forecasts often depict physically impossible features and feature relationships.

3.2 Data Abstraction

Forecast simulations are multi-dimensional, and multivariate. Each forecast simulation is a time-varying set of height layers, where each height layer is a 2D grid with variable values at every grid point. Each height layer may have tens to hundreds of variables. Ensemble forecasts can analogously be thought of as multivalued, with the set of simulation members defining possible values for a particular variable at a specific point in simulated space-time [29].

Notably, while a set of 2D height layers can be characterized as a single 3D dataset, the meteorologists we spoke with generally do not think about them as such. While the simulations do cover a continuous 3D extent, the height layers are far enough apart that any dependencies between them are considered negligible.

A *field* refers to a particular variable at a particular height: a 2D

grid of generally scalar, but possibly vector, values. Our contextual interviews and subsequent conversations indicated that, in practice, a given meteorologist only cares about a subset of around 50 fields out of the hundreds often available within a given forecast. Which 50 fields, however, ranges from meteorologist to meteorologist according to their particular problems and prior experience. Meteorologists additionally incorporate many *derived fields* into their workflows. Derived fields represent a mapping from a set of fields to a single field. This mapping may combine multiple instances of the same field from different simulations, as would be required to calculate the average field over an ensemble. Alternatively, the mapping may combine multiple fields from a single simulation, as in the case of calculating the Haines Index. The Haines Index takes values between two (low risk) and six (high risk) indicating the potential for accelerated wildfire growth based on temperature and dew point differences across various atmospheric height layers [27]. For all of the derived fields we discuss, values at different grid locations are treated as completely independent. This notion of grid-location independence is neither a necessary condition for a derived field, nor a reflection of the physical processes being simulated. It is an artifact of how the meteorological community computes derived quantities in practice.

A *feature* refers to a significant artifact, usually an event, trend, or boundary, that can be located within a given field. For scalar fields, features are generally visually represented by isocontours or gradients. For vector fields features generally correspond to topological structures such as a source, sink, saddle-point, or closed orbit.

4 RELATED WORK

The related work for this paper loosely falls into two categories: work that addresses the visualization of meteorological data and work that addresses the visualization of ensembles and uncertainty.

4.1 Visualizing Weather

A number of software systems have been designed for the visual exploration of meteorological data in both operational and research environments [1, 2, 4, 9, 15, 17, 21, 23, 24, 29]. These systems provide varying levels of user control for selecting which fields to visualize and how to encode them. Some rely entirely on scripting-based interfaces [1, 2, 15], as opposed to offering a graphical UI; some deal solely with gridded information, such as forecasts, while others integrate point-based observational data [1, 15], as well as real-time satellite and radar data [4, 9, 21, 23]. As Table 1 illustrates, however, these

Tool	2D	3D	Deterministic	Single Feature Direct Ensemble	Multiple Feature Indirect Ensemble	Multiple Feature Direct Ensemble	Movement toward Informed Defaults
AWIPS [9, 21]	●	●	●	●	●		
Metview/ Magics++ [4]	●		●	●	●		
GrADS [15]	●		●	●	●		
IDV [23]	●	●	●	●	●		■
Vis5D [17]	●		●	▲		▲	
Ensemble Vis [29]	●	●		●	●		●
SimEnvVis [24]	●	●	●		◊		●
FERRET [1]	●	●	●	●	◆		
GMT [2]	●	●	●		◊		

Table 1: Overview of various visualization mechanisms in common meteorological systems.

- : full support
- : colormaps created for specific variables
- ▲: small multiple 3D views only

- ◊: can theoretically visualize any pre-derived data using available deterministic mechanisms
- ◆: like ◊, but also supports a data aggregation operation

systems have done very little with either informed, default encodings or the direct visualization of multiple features across an ensemble.

The majority of the aforementioned systems leave encoding choices to users who lack training in visualization principles. So far, initial steps towards addressing this issue have taken orthogonal approaches. Unidata’s Integrated Data Viewer (IDV) [23] provides a set of color map recommendations for different types of fields, though most appear to be spectral and ignore effective visualization practices. Potter et al.’s Ensemble Vis framework, on the other hand, reuses a single set of two sequential and two categorical color maps based on visualization principles for all fields of interest [29]. Taking an entirely different tack, Nocke et al.’s SimEnvVis allows for the creation of rules that assign encodings based on field metadata and user preferences; however, a user’s lack of training regarding visualization principles remains an issue. Alternatively, Ware and Plumlee propose a set of perceptually motivated design alternatives for encoding weather data [36].

There are also a number of websites that provide users with a range of static forecast visualizations. The Short-Range Ensemble Forecast (SREF) website run by the Storm Prediction Center [6] is one example, but a list of others can be found in our Supplemental Materials. The static visualizations provided on these websites are largely created using tools or systems like those described at the beginning of this section. The users of these web-based systems, however, exercise even less control over which fields they can look at and no control over how the information is encoded.

4.2 Uncertainty

A significant amount of work addresses the visualization of uncertainty — Potter et al.’s taxonomy [28] contains a thorough overview of the state-of-the-art approaches. In terms of research relevant to visualizing uncertainty in meteorological data, the early work focuses primarily on uncertainty in Geospatial Information Systems. Work by both Pang [25] and MacEachran [20] outline various issues and recommendations with regards to the visualization of geospatial uncertainty, while Thompson et al. present a typology for uncertainty in geospatially referenced data for intelligence analysis [34].

Attempting to better address the needs of meteorologists, a number of papers focus on understanding uncertainty from simulation ensembles. Vis5D [17] employs small multiples as a way of looking at multiple ensemble members, while SimEnvVis [24] explores various comparative techniques for investigating differences among ensemble members. EnsembleVis [29], on the other hand, enables the generation of visual summaries for ensemble forecasts that specifically emphasize the probabilistic characteristics of the ensemble. Additionally, Sanyal et al.’s Noodles tool [30], Póthkow et al.’s probabilistic marching cubes technique [26], and Whittaker et al.’s contour boxplot technique [37] present methods for directly summarizing ensembles of isocontours in visualizations. None of this work, however, has addressed the task of relating multiple features across an ensemble.

5 INFORMED DEFAULTS

Many meteorologists turn to visualization as a way to quickly locate relevant weather features within the forecast data. These meteorologists often mentally integrate information from multiple visualizations, which make use of a wide range of encoding conventions. The design flaws in many of these visualizations can lead to misinterpretation, inaccuracy, and inefficiency, among other consequences [14]. In an attempt to address this issue, we present a set of informed, default encoding recommendations that integrate existing meteorological conventions with visualization principles.

5.1 Understanding Meteorological Encoding Conventions

We collected representative samples of visualization products from a variety of sources used extensively by our collaborators. In total, we compiled 41 images representative of the encoding conventions used by 9 different meteorological sources — we include these images in the Supplemental Materials. This sampling was intended to be representative rather than exhaustive, encompassing the most popularly referenced visualization products along with those specifically referenced by our collaborators. We coded these images based on the forecast variables shown and the encoding channels used. Using these codes, we then grouped the images based on their similarity.

In general, we found that visualizations generated by a given source have consistencies, but those consistencies rarely extend across sources. The encodings employed in these visualizations include contours, color maps, texture maps, streamlines, glyphs, and wind barbs [36]. Contours, color maps and wind barbs are favored heavily, with contours and color maps showing up in 85% and 78% of the samples, respectively, and wind barbs comprising over 80% of all glyph usage. For two-thirds of the samples, either 2 or 3 fields are encoded in a single visualization, though we did find isolated examples of attempts to encode up to 4 fields simultaneously. For both contours and color maps, a range of common intervals exist for certain variables: relative humidity and probabilities usually step by 10 percent, temperature usually steps by 3, 4, 5, or 10 degrees, and geopotential heights usually step by 30, 40, 60, or 120 meters. Due to the static nature of these images, a label is required for each connected component of each contour in a given image. As a result, increasing the number of contours in a given visualization, either by using a smaller interval step-size or by layering additional fields, quickly leads to increasingly cluttered visualizations like those in Figures 2a and 2d.

In terms of color usage, there are a number of problematic trends. It is common for colored contours to be overlaid on top of color maps, or for colored contours and color maps with transparent portions to be overlaid on top of multi-colored backgrounds, both of which add difficulty to resolving features — this latter case is exemplified in Figure 2c. Additionally, rainbow color maps, which are widely considered a poor choice by the visualization community [11], comprise nearly

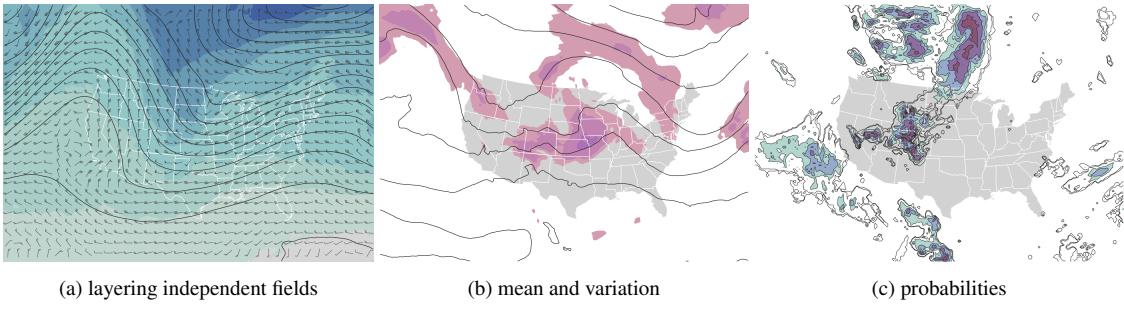


Fig. 3: Examples of visualizations generated by each of our three cases for informed defaults.

70% of the color maps in our sample set. We also noted multiple cases of cyclical color map design, including the cyclical rainbow color map used in Figure 2b.

There are other color trends, however, that do not violate visualization principles. For example, meteorologists almost always use discrete color maps, rather than continuous; in fact, our sample set did not include a single continuous color map. As another example, when a mean field is displayed in concert with a variational measure such as standard deviation or spread for an ensemble, the variational metric is virtually always color mapped and the mean overlaid as a set of contours, as shown in Figure 2d — we encountered only one example where this configuration was reversed. We also noted a handful of trends in color use tied to specific variables. Geopotential heights are almost never color mapped and usually shown as black contours, like in Figures 2a and 2c. When color mapped, relative humidity is nearly always represented using 3 shades of green, usually denoting the 70–80, 80–90, and 90–100 percentage intervals, with the lower intervals being entirely ignored; Figure 2c shows an example of this. When relative humidity is contoured, on the other hand, isocontours are generated for every interval of 10 percent. The color map for temperature is always a rainbow color map with blues on the low end and reds on the high end, though we also noticed a secondary trend where temperature is displayed using a colored contour scheme with the color blue specifying values below freezing and red specifying values above freezing, as seen in Figures 2a and 2c. Additionally, while probability fields are usually encoded using either contours or a color map, there were also notable examples, such as the one reproduced in Figure 2e, where the probabilities are dual encoded using both. The contours provide a constant overview of the entire field, while the color map is specifically used to highlight value intervals of significance.

5.2 Proposed Defaults

Based on the trends we observed, we propose the following set of informed defaults. We outline three cases. The first involves the simultaneous display of independent fields. These may be original or derived fields from either a deterministic or ensemble forecast. The notion of *dependence* used here is based on whether the interpretation of a given field necessarily depends on knowledge of some other field. The only dependent fields that arise within the context of this work are standard deviation fields derived from an ensemble, as a measure of standard deviation is generally meaningless without the corresponding mean. Thus, our second case involves the simultaneous display of an ensemble-derived dependent variation field with its corresponding mean field. Our third case involves the display of the uncalibrated probability of a given event or condition derived from the ensemble of predicted outcomes. The *uncalibrated* probability refers specifically to the percentage of the ensemble members that predict the event or condition of interest. It is important to distinguish this from the actual expected frequency of the condition or event, as most ensembles underestimate the true range of possible forecast outcomes [32]. For each of these three cases we have abstracted a recommended set of encoding choices, examples of which can be seen in Figure 3.

For simultaneously displaying independent fields, we recommend staying within combinations of three encoding choices: a base color map, a set of contours, and a set of glyphs or texture map (though we do not currently support texture maps in WeaVER). This set of encod-

ing choices allows for the simultaneous display of two 2D scalar fields along with a third 2D field of either scalar or vector values. Ware and Plumlee propose an alternative configuration that uses color, texture with optional contour boundaries, and animation to encode a similar set of fields [36], however this configuration is a departure from the conventions at the core of current meteorological visualizations. Given that meteorologists, especially those in decision-making contexts, have significant training and experience using current meteorological conventions, we tried to incorporate those conventions as much as possible while remaining within the bounds of accepted visualization practices.

For the other two cases, our recommendations mirror existing meteorological convention. For the simultaneous display of an ensemble-derived mean field with its dependent variation field, we recommend a single combined choice that encodes the variation metric as a color map and overlays the mean field using a set of contours. For encoding the uncalibrated probabilities of a condition, we recommend a single dual-encoded choice where value intervals of significance are highlighted by a color map, while all other value intervals are represented only by contours. The notion of a *condition* used here may refer to either a single individual condition (e.g. relative humidity less than 10 percent) or the joint condition of a set of individual conditions. While multiple user choices may go into specifying a set of individual conditions, that set still represents a single joint condition choice.

We additionally provide a set of recommendations of low-level encoding behaviors for specific variables or fields. For both the color map and the contour intervals, the handful of trends discussed in Section 5.1 represent reasonable choices. For variables without common trends in interval spacing, we use spacings specified by our collaborators. Wind bars are evenly spaced to prevent intersection of the glyphs, though other placement strategies may also be acceptable [10].

We propose a set of color maps for a number of variables, shown in Figure 4. Aside from the categorical map for representing the Haines index in Figure 4g, these color maps are designed to approximate perceptually uniform steps in luminance. We do not claim these color maps guarantee perceptually equivalent luminance steps, as that would require a calibrated monitor, control over the ambient light in the viewing environment, etc. [18]. The color maps are instead defined in device dependent HSV space, but use a nonlinear function in order to vary brightness in a way that creates convincingly uniform perceptual steps. Initial versions of the color maps, which can be found in Supplemental Materials, were generated algorithmically using a cube-root approximation to the Munsell value scale in combination with linearly varying hue and saturation. Additional hand-tweaking, however, was still required to create the final proposals.

Integrating an interactive routine to approximate a monitor’s gamma value, such as that outlined by Kindlmann et al. [18], would allow for more accurate control of luminance variation across devices. We wanted to ensure, however, that the color maps were specified such that they could also be used in other meteorological tools.

While the proposed color maps cover similarly large portions of the luminance spectrum, we specifically did not use the entire spectrum in order to allow for grey values for both contours and wind bars that would remain distinguishable from the color mapped background when overlaid. Based on color usage trends that we observed, we created a green color map covering the 70%–100% value range for the

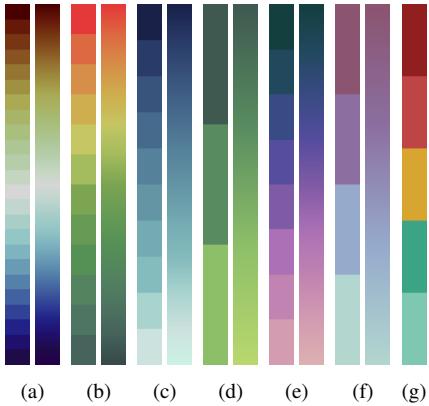


Fig. 4: Discrete and continuous colormap proposals for: (a) temperature, (b) wind speed, (c) precipitation, (d) relative humidity, (e) standard deviation, (f) probability, and (g) Haines index.

display of relative humidity and a diverging spectral color map for temperature, as shown in Figures 4d and 4a. Also included in Figure 4 are separate, distinguishable color maps for other common variables: accumulated precipitation, high wind speeds, ensemble standard deviation, and ensemble probabilities. While these additional color maps are based solely on principles of effective color usage [22], rather than any meteorological convention, the unique color maps for each variable within a system are meant to improve users' efficiency in dealing with multiple visualizations. With this goal in mind, the majority of the color maps are defined on an absolute scale covering the range of values taken by that variable across all height layers. The color map for standard deviation is the notable exception, as meaningful thresholds of uncertainty change for different variables.

While we recognize that our choice of a spectral color map for temperature is generally considered a poor choice for univariate color map design [11], there is a history of justified use of spectral color schemes for visualizing weather [29, 36]. While hue does not have an inherent perceptual ordering, its familiarity and widespread use have led to an expected ordering within meteorology. This is strengthened by matching the intuitive mappings that exist for certain colors in spectral schemes (e.g. red is hot, blue is cold) [36]. By controlling for luminance variation, the proposed spectral color map does not suffer from the perceptual irregularities that traditionally plague spectral schemes [11]. We employ a diverging luminance scheme in order to emphasize the nature of the freezing point boundary as a critical center point for temperature. Moreover, well-designed diverging spectral schemes behave just as well as widely-accepted two-hue diverging color schemes for modeling data distributions with a critical mid-range value [13]. Given all this, we believe that spectral-nature alone is an insufficient reason to go against one of the strongest meteorological conventions that we encountered.

The low-level encoding behaviors for the variables presented in this section do not comprise a comprehensive solution for meteorological visualization, but we believe they represent a reasonable solution for some portion of the meteorological community. Our early interviews indicated that most meteorologists refer to only 50 or so fields and, in the case of our collaborators, those fields only covered a handful of different variables. This made the specification of unique color maps on a per-variable basis possible for our prototype. Given the large number of variables present in forecasts, however, having separate, distinguishable, absolute color scales for every possible variable will never be feasible. Similarly, while our diverging spectral color scheme for temperature results in clear visual differences at intervals of ten or even five degrees, if our collaborators had needed to be able to resolve steps of three degrees, the luminance differences become too small for smoothly varying hue and saturation to result in sufficient visual differences. As such, it is important to remember that enabling effective defaults will always require understanding the needs of the target users. The specifications we have outlined here are simply one possible configuration that results in effective visualizations for the majority

of the cases handled by our collaborators.

6 ENSEMBLES OF FEATURES

Understanding the variability and associations among features across ensembles is often critical to decision-making. This is especially true in areas such as wildfire emergency management, which rely on forecasts multiple days into the future when the forecasts remain incredibly uncertain. Many meteorologists rely heavily on visualization to develop this understanding, yet no existing visualization methods allow users to effectively explore the variability of feature relationships across an ensemble. Given that most decisions in meteorology are based on the intersection of multiple features, this represents a major gap in visualization's current support for meteorological tasks.

In meteorology, the only conventional technique for directly examining the distribution of behaviors for a given feature across an ensemble is the use of *spaghetti plots*. Spaghetti plots display the set of isocontours associated with a specific value, one for each ensemble member, within the same, generally static plot. Color is routinely used to distinguish which ensemble member a particular contour is derived from. Spaghetti plots for multiple features, especially intersecting or overlapping features, are rarely overlaid within visualizations, because the results do not effectively present the range of feature relationships in the ensemble. If color is used on a per-member basis, visually separating overlapping or intersecting contour sets into their respective features becomes incredibly difficult. Alternatively, when color is used to explicitly differentiate feature contour sets, it becomes nearly impossible to pick out the contours associated with a particular ensemble-member. These problems are compounded as the number of ensemble members or the number different features increases. In both cases, picking distinguishable colors becomes more challenging and the occlusion of contours by one another, regardless of which feature they belong to, becomes increasingly problematic.

While the use of small multiples for each ensemble member in Vis5D [17] represents one possible step towards addressing these deficiencies, the idea has never been generalized back to 2D plots. We opted for a different approach, developing a modified formulation of the spaghetti plot technique that uses interactive highlighting as the primary mechanism for distinguishing between members. We refer to these modified spaghetti plots as **interactive spaghetti plots**. As shown in Figure 5a, this formulation frees up the color encoding channel, allowing the use of color to differentiate and enable direct comparisons among the distributions of multiple isocontour features within a single plot. It also mitigates contour distinguishability and occlusion issues by bringing the highlighted contour(s) to the front of the view and decreasing the visual saliency of other features' contours. Users are able to look at multiple features simultaneously and interactively highlight contour sets at both the member and feature level. This supports a variety of both exploratory and investigatory tasks.

Even with interactive highlighting, spaghetti plots still do not scale well, quickly becoming visually cluttered. Given conversations with our collaborators heralding the advent of *super ensembles*, ensembles with hundreds of members, we also wanted to provide users with a technique that could scale. To this end, we integrated Whitaker et al.'s *contour boxplots* [37] as a state-of-the-art encoding technique that directly summarizes an ensemble of isocontour features using a boxplot-like summarization of a set of 2D contours.

There are a number of benefits to using contour boxplots for simultaneously visualizing multiple isocontour based feature sets. Being analogous to box-and-whisker plots, they provide a commonly understood statistical framework for summarizing a distribution of isocontour-based features across the ensemble. Because contour boxplots were motivated in part by the idea of aggregation preserving shape, the characteristic details of physically plausible features are still present in the summarizations, which is often not the case for the corresponding features derived using the ensemble-mean. Moreover, contour boxplots are scalable as their visual representation remains consistent for any arbitrary number of ensemble members.

Like with the interactive spaghetti plots, we use a modified formulation of **interactive contour boxplots**. Color is, again, used to

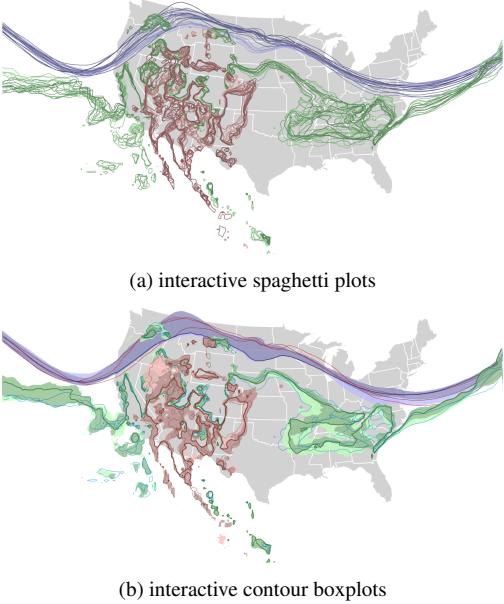


Fig. 5: The simultaneous display of three isocontour features using modified ensemble visualization techniques.

differentiate between the summarizations of different features over the ensemble, as shown in Figure 5b. We duplicate the functionality for feature-level highlighting, bringing that feature’s contour boxplot to the front of the view and decreasing the visual saliency of any other contour boxplots in the display, mitigating distinguishability and occlusion issues. Additionally, we allow users to query and highlight contours from the ensemble on a per-member basis, thereby retaining the capability for interactive exploration of the entire distribution of feature relationships across the ensemble.

While we chose to limit users to looking at up to three features simultaneously, the interactive mechanisms we have outlined could allow for the simultaneous inclusion of a larger number of features in a single view. The added benefit to simultaneously visualizing a fourth or even fifth feature, however, is unclear, when users are provided with a mechanism for easily swapping out features.

7 WEAVER

In this section, we provide an overview of WeaVER, an open-source tool developed to test our informed defaults and extended ensemble visualization techniques. WeaVER is not intended to be a fully viable alternative to current operational forecasting tools; rather, it is a proof-of-concept, designed to allow our collaborators to evaluate the proposed ideas.

7.1 Data Processing

WeaVER is designed to visualize NCEP’s SREF (Short-Range Ensemble Forecast), which currently contains 21 member simulations: 7 different sets of initial conditions run over 3 models. The simulation is run 4 times per day (at 03, 09, 15, and 21 UTC) and includes predictions at 3-hour intervals out to 87 hours into the future. We use the version of the SREF run on NCEP’s 212 Grid [8], which is a Lambert Conic Conformal grid over the continental United States with approximately 40 km grid spacing. The forecast data is retrieved from the NOAA Operational Model Archive and Distribution System (NO-MADS) server (nomads.ncep.noaa.gov) where it is released in the binary GRIB2 [8] format. The wgrib2 command line utility [7], which is made publicly available by NCEP’s Climate Prediction Center, is used to parse the data into a csv format for preprocessing.

A fairly significant amount of data preprocessing is required for WeaVER to achieve interactive rates. We generate a number of derived fields, such as the Haines index, for each ensemble member, along with various statistical derived fields (max, min, mean, and standard derivation) across the ensemble. We also precompute the statistical

quantities required to generate interactive contour boxplots for various isolines. Additionally, we derive a number of *condition fields* for calculating the uncalibrated probabilities of arbitrary joint conditions on the fly. Because an individual condition applied to a single field results in a boolean value at each grid point, we can concisely represent a condition applied to an ensemble of fields as a bit-set at each grid point. This reduces the computation of both arbitrary joint conditions and uncalibrated probabilities to a small number of per grid-point operations. The bit-set representing any joint condition can be computed using a series of bitwise AND (&) operations, while calculating the uncalibrated probability simply requires dividing the Hamming weight of a bit-set by the total number of ensemble members.

7.2 System Overview

WeaVER consists of five interchangeable views: a *deterministic* view, an ensemble statistic (*stat*) view, an ensemble mean and standard deviation (*mnsd*) view, an ensemble *probability* view, and a *direct ensemble* view. The *deterministic* and *stat* views both handle layering of multiple independent fields, the first of the three cases discussed in Section 5.2. The ensemble *mnsd* and *probability* views handle the second and third cases, respectively. The *direct ensemble* view, on the other hand, allows users to switch back and forth between interactive spaghetti plots and interactive contour boxplot summarizations using the mechanisms described in Section 6. The specific fields, isolines, and conditions supported across these views were explicitly requested by our collaborators.

Across all five views, fields may be dragged from a library on the right-hand side and dropped onto various encoding targets at the top of the view, as illustrated in Figure 1. The currently configured visualization is automatically updated according to the user’s choices. Mouseover of any of the drop targets reduces the saliency of the other encodings in the visualization. When color maps and contours are controlled by the same layer (as in the ensemble *mnsd* and *probability* views), the visual saliency of the contours is reduced in favor of showing the color map more clearly. In all views, a tab-like mechanism allows users to create and quickly switch back and forth between multiple configured visualizations by either using the GUI interface or the numeric keys on the keyboard. Every view also offers independent time manipulation through animation, a slider-based interface, and forward or backward time stepping using either the GUI interface or the arrow keys on the keyboard. We include a video overview of the interactive features in Supplemental Materials.

Contrary to meteorological convention, we take a detail-on-demand approach to contours labels. There are no labels on the contours by default; rather, labels appear on mouse-over. This action simultaneously highlights the full isocontour, which is especially helpful for contours with multiple connected components. Clicking when a contour is highlighted creates a persistent *sticky label* that moves with the contour across time steps. At any time a user can reposition a sticky label along the contour by dragging, or delete it by double clicking. Limiting the labels to those requested by the user not only reduces clutter but also reduces the number of targets for visual search.

7.3 Implementation

All of the data fetch and preprocessing code was implemented using a combination of bash scripts and C++ programs that have been tested and run on both Mac and Linux. This code has several dependencies: we use wgrib2 [7] to transform the data from its original binary GRIB2 format, and the contour boxplot implementation relies on ITK [3].

WeaVER itself is implemented entirely in Processing [5]. The resulting Java application has similarly been tested and runs on both Mac and Linux. We are making the source code for WeaVER publicly available at <http://www.sci.utah.edu/~samquinan/software/WeaVER/>. Specifications of the proposed default color maps are included within this source code, along with both example preprocessed data and the data processing code.

We note that the current handling of geographic projection within WeaVER is an approximation of a Lambert Conic Conformal projection. As NCEP’s SREF is run on a Lambert Conic Conformal grid,

we found that treating the forecast data as an equidistant grid overlaid on top of a pre-projected Lambert Conic Conformal map provided a sufficient visual approximation to the proper projection.

Additionally, our design decisions to use the 40 km resolution version of the SREF as well as to visualize the forecast over the entire continental US were based specifically on needs of our collaborators. In general, meteorological phenomena exist at a range of different geographic scales, ranging anywhere from under a single kilometer to thousands of kilometers. As such, the appropriate geographic scale for visualizing weather data depends both on the scale of the features of interest and on the resolution of the underlying forecast.

8 VALIDATION

We validated WeaVER through a series of semistructured interviews with our collaborators. A set of initial interviews were conducted in which we elicited feedback from various collaborators after walking them through a full demo of WeaVER. We additionally provided custom installations of the software for an extended evaluation period to our two collaborators working in wildfire prediction, allowing them to evaluate the tool in the context of their work environment. We provide a description of how one of our collaborators used WeaVER with historical forecast data, as well as informal feedback from our other collaborators.

8.1 The Diego Fire

For one of our wildfire prediction collaborators, we supplied archived forecasts from three mornings leading up to the Diego fire, a lightning-strike fire that began in northern New Mexico during June, 2014 — our collaborator requested these specific forecasts in order to retrace his predictions of the Diego fire using WeaVER. He began by investigating the forecasts through the *deterministic* and *stat* views in order to gain a sense of the big picture atmospheric conditions over the forecasts. He was particularly excited about the ability to easily create visualizations with combinations of fields tailored to his own problems and preferences. He also noted that the resulting visualizations were more visually appealing and easier to read than what he usually looks at. Using these views he was able to determine that the forecasts showed the expected signals for a threat of a lightning-started fire: moisture, indicating lighting potential, on the front end of the forecasts, followed by windy, dry, unstable conditions for a day or so after.

After gaining a sense of the big picture three days out, our collaborator began investigating the forecasts using the *direct ensemble* view. He stated that he needed a sense of the spread or variation across the ensemble, but more importantly, he also needed to be able to understand how that variation differs from a particular model or member. He explained that organizations such as the National Weather Service still key their recommendations off a deterministic forecast, so understanding how the rest of the ensemble compares to that particular member is incredibly important. As such, he appreciated being able to interactively highlight a particular member from the ensemble. He was also particularly impressed by the contour boxplot summarizations. He stated that, while it would take training for forecasters to understand exactly what they are looking at, the contour boxplots provide the same visual cues of the forecast as spaghetti plots, but much more quickly and concisely. When pressed to look at multiple features simultaneously, our collaborator noted that he could see expected behaviors and interactions. He also noted that interactive spaghetti plots had significantly decreased utility compared to the interactive contour boxplots for contour-sets generated from non-well-behaved fields such as the Haines Index.

Finally, our collaborator looked at the *probability* view in order to determine which areas had a high likelihood of a critical combination of dry, windy, and unstable conditions in the latter portions of the forecast. As we have reproduced in Figure 6, the combined condition of surface temperatures greater than 60° F, surface wind speeds greater than 20 mph, and a Haines index of 5 or greater highlighted the area over northern New Mexico as favorable for fire spread after lighting ignition. Our collaborator noted that this highlighted area, which he would have been worried about, is where the Diego fire originated.

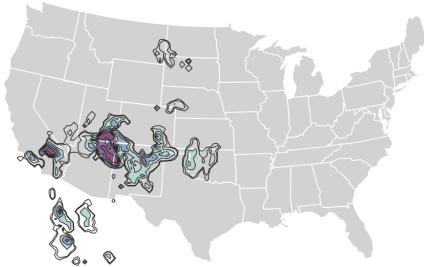


Fig. 6: The ensemble-derived probability of conditions favorable for wildfire growth forecasted during the time frame of the Diego fire outbreak, highlighting northern New Mexico where the fire originated.

8.2 Informal Feedback

We also received a significant amount of supporting feedback from the interviews with our other collaborators. In general, the addition of interactivity was well-received. Our collaborators commented that it was straight-forward to create a wide variety of views, and that the sticky labels and interactive highlighting were a significant improvement over the traditional static visualizations. Several of our collaborators noted that in existing tools they must either choose to label every contour or none, which leaves them in a dilemma as they either have to obscure information in order to provide context or forego context entirely. The sticky labels and interactive highlighting, alternatively, made contour labels visible only where needed.

We received multiple independent confirmations of the meteorological encoding conventions we derived in Section 5.1. Several of our collaborators commented that WeaVER’s informed defaults highlight the fact that effective color usage is something that forecasters generally struggle with. They also noted, however, that some of the meteorological conventions encoded in the informed defaults were not always sufficient for their specific problems. For example, in wildfire prediction low relative humidity is the primary concern, making a color map highlighting the 70%–100% value range, largely useless. Similarly, while our default temperature increments of 5°C are fine for an overview, when forecasting the rain-snow transition line for winter weather predictions, meteorologists need to see a rapid change localized around 0°C with half degree increments.

According to our collaborators, interactive highlighting was the primary key to understanding feature relationships in both the interactive spaghetti plot and interactive contour boxplot techniques. They were particularly intrigued by contour boxplots as this visualization provides a fundamentally different way to get at the statistics or uncertainty of a feature’s behavior across the ensemble. One collaborator observed how quantile bands can immediately indicate that half of an ensemble’s members are located within a specific geographic region, an observation that would have required him to count individual members in a spaghetti plot. The collaborator also noted that the latter procedure simply does not scale, stating that scalability is becoming increasingly important as the number of ensemble members is expected to increase by an order of magnitude over the next couple decades.

9 DISCUSSION

The feedback from our collaborators indicates that our informed defaults were largely a success. Provided with simple, transparent visualization encoding choices, our collaborators were able to interactively generate a wide range of effective visualizations tailored to a variety of their own needs and problems. Limiting the user choices to effective combinations of encodings and specifying low-level behaviors regarding color usage and contour spacing at the system level resulted in the users’ visualizations maintaining several meteorological conventions, while widely being considered easier to read. Labels do not clutter the view, masking important features; rather, meteorologists can interactively place labels exactly where needed based on the current context, regardless of how that context shifts.

The feedback regarding both interactive spaghetti plots and interactive contour boxplots was more mixed. Interactivity was the key to

enabling the exploration of feature relationships across the ensemble; however, not in the way that we had anticipated. We figured that interactivity would alleviate differentiation and occlusion issues, but that even statically, when those issues still exist, both techniques would convey some understanding of the distribution of feature relationships. Instead, interactively highlighting the relationships for each ensemble member did a better job showing the range of feature interactions than the static display of either technique. Additionally, interactive highlighting allows users to efficiently compare the behavior of a given member to the rest of the ensemble, which turns out to be a critical task for meteorologists who need to be able to resolve their own predictions against the recommendations of others. Both techniques offer a clear improvement over standard spaghetti plots with regards to relating multiple uncertain features.

Our feedback also suggests that contour boxplots generally represent an improvement over spaghetti plots. Contour boxplots can show physically meaningful statistical variation within the spread of features in a simpler and more concise manner, improving forecasting speed. They can also create meaningful summaries for more complex fields, such as the Haines index, where a lack of grouping among the contours makes it extremely difficult to visually extract a meaningful summary from spaghetti plots. Additionally, contour boxplots are significantly more scalable, which will likely become important in the future.

That said, there were several aspects of our evaluation which were not particularly successful. For reasons we discuss in Section 10, we were unable provide several of our collaborators with their preferred forecast. Additionally, WeaVER does not support investigating ensembles of non-isocontour features, such as cold fronts. These limitations forced our collaborators to make judgements out of context regarding the efficacy of WeaVER.

More generally, our evaluation of informed defaults suffers from a mismatched scope between our designs and validation. We designed the informed defaults using general meteorological conventions in the hope that they would be applicable to a wide range of meteorologists. We then attempted to validate them with a relatively small set of meteorologists with differing needs and problems. Unsurprisingly, our low-level encoding choices were not always deemed appropriate by all our collaborators. Several alternative options exist. One option, common in design studies, would be to gear the default encoding behaviors to a specific subset of meteorologists, such as those forecasting wildfires in the southwestern US. Another option would be to provide users with the ability to interactively modify low-level encoding behaviors in order to support a wider range tasks. The first option reduces the generalizability of the informed defaults, while the second option provides the opportunity to create increasingly ineffective visualizations. Neither alternative represents a best-case scenario. This raises an interesting question: what *is* the proper way to design for a set of experts who have similar goals but individualized processes and domains? From an evaluation standpoint, such a design may require a new model for validation.

10 CHALLENGES WORKING WITH WEATHER DATA

A number of challenges and hurdles made this design study more difficult than we initially anticipated. Here we reflect on a number of these issues to provide guidance for others working with meteorological data in the future.

Because our collaborators were purely consumers of meteorological visualizations, it was difficult for us to get direct answers about how existing visualizations were created. For example, confirming that the data processing for derived fields assumes grid-location independence required going through multiple levels of contacts. Similarly, none of our collaborators could answer our questions regarding geographic projection or the smoothing and down-sampling of forecast data. As a result, there were certain critical complexities that it took us a very long time to understand.

Of these complexities, we spent significant time understanding the issues with geographic projections. Because NCEP’s grids use a non-standard geodetic datum [8], we needed to understand what issues underly simultaneously projecting simulations and maps based on dif-

ferent geodetic data. While there *are* errors that arise from simply modifying the projection equations to use different geodetic data, we were eventually able to confirm that these differences are primarily significant when dealing with high resolution local-scale simulations. Additionally, despite the prevalence of interpolated contours and color maps, it remains unclear both how the associated latitude and longitude values needed for projection should be interpolated and how existing tools are interpolating those values in practice. For visualizations designed for deployment in decision-making environments, these sorts of considerations need to be accounted for.

As another example, only one of our collaborators was aware that many calculations of the uncalibrated probabilities of joint conditions over an ensemble assume the individual conditions are independent. As the individual conditions are derived from the same ensemble, however, this assumption is not only unfounded but generally misleading. While the resulting differences in predicted probability at a given grid point can be fairly large, we generally found fairly subtle differences when visualizing the entire field. Still, such a disconnect between the data presented in visualizations that meteorologists look at and the meteorologists’ interpretation of that data is problematic.

We also ran into a number of practical and engineering challenges that forced us to scale back our designs. The scope of the forecast data was significantly greater than we anticipated and initially designed for, and it is only going to expand in the future. Luckily, related work on climatological systems [38] could offer insight into dealing with this data increase. Additionally, internet connectivity issues and a lack of install permissions in several of our collaborators’ work offices became non-trivial design hurdles.

We also experienced major issues getting the data desired by our collaborators. Several of our collaborators primarily look at long-range ensembles, such as the Global Ensemble Forecast System (GEFS) ensemble. While the GEFS is both run and used internally by the National Weather Service at a 40 km resolution [16], it is only made publicly available at less than half that resolution. This forced us to test WeaVER with the SREF, which does not provide significant enough lead times for many of our collaborators’ decisions. Moreover, a lack of public research archives for the SREF made it difficult to track down meaningful data for our collaborators to look at when eliciting feedback during their off-seasons.

11 CONCLUSIONS AND FUTURE WORK

In this work we present a characterization of both the problems and data associated with meteorological forecasting. We outline a system for informed defaults that allow meteorologists without visualization expertise to generate a wide variety of effective visualizations based on current meteorological conventions and visualization principles. We also extend state-of-the-art visualization techniques in order to allow users to effectively relate multiple isocontour features. As a proof-of-concept for these ideas we present WeaVER, an open-source tool for interactively visualizing weather forecasts.

As future work we are interested in conducting a formal evaluation of both interactive contour boxplots and interactive spaghetti plots as mechanisms for looking at multiple isocontour based features simultaneously. We are also interested in investigating how to design a user study that can account for highly individualized analysis processes like those found in the meteorological community. Additionally, there remains no current visualization methods that enable the direct visualization or comparison of non-isocontour based features across an ensemble, providing another interesting avenue for future work.

ACKNOWLEDGMENTS

The authors wish to thank the following individuals for their feedback and assistance: Mike Kirby, Ross Whitaker, Mahsa Mirzargar, Shelby Law, Chuck Maxwell, Lance Avey, Jim Steenburgh, Trevor Alcott, Jack Kain, Kent Knopfmeier, Don House, Mike Goodchild, Steve Kruger, Jason Dykes, and Alan MacEachran. This work was funded by NSF grant IIS-1212806.

REFERENCES

- [1] Ferret. <http://ferret.pmel.noaa.gov/Ferret/>. Accessed: February 2015.
- [2] GMT. <http://gmt.soest.hawaii.edu/>. Accessed: February 2015.
- [3] Insight segmentation and registration toolkit (ITK). <http://www.itk.org/>. Accessed: March 2015.
- [4] Metview. <https://software.ecmwf.int/wiki/display/METV/Metview>. Accessed: February 2015.
- [5] Processing. <https://processing.org/>. Accessed: March 2015.
- [6] SREF products from the storm prediction center in norman, OK. <http://www.spc.noaa.gov/exper/sref/>. Accessed: March 2015.
- [7] wgrib2. <http://www.cpc.noaa.gov/products/wesley/wgrib2/>. Accessed: March 2015.
- [8] *Office Note 388 — GRIB, revised*. NOAA National Centers for Environmental Prediction, February 2005. <http://www.nco.ncep.noaa.gov/pmb/docs/on388/>.
- [9] *The AWIPS CAVE-D2D User's Manual, OB 13.4.1*. NOAA National Weather Service and Raytheon, July 2013. https://www.unidata.ucar.edu/software/awips2/doc/CAVE_D2D_Manual.pdf.
- [10] R. Borgo, J. Kehrer, D. H. S. Chung, E. Maguire, R. S. Laramee, H. Hauser, M. Ward, and M. Chen. Glyph-based Visualization: Foundations, Design Guidelines, Techniques and Applications. In M. Shert and L. Szirmay-Kalos, editors, *Eurographics 2013 - State of the Art Reports*. The Eurographics Association, 2012.
- [11] D. Borland and R. M. Taylor. Rainbow color map (still) considered harmful. *Computer Graphics and Applications, IEEE*, 27(2):14–17, March 2007.
- [12] P. D. Bothwell and L. M. Richardson. Forecasting lightning using a perfect prog technique applied to multiple operational models. In *15th International Conference on Atmospheric Electricity*, Norman, OK, June 2014. Preprint.
- [13] C. A. Brewer. Spectral schemes: Controversial color use on maps. *Cartography and Geographic Information Systems*, 24(4):203–220, 1997.
- [14] A. Dasgupta, J. Poco, Y. Wei, R. Cook, E. Bertini, and C. T. Silva. Bridging theory with practice: An exploratory study of visualization use and design for climate model comparison. *IEEE Transactions on Visualization and Computer Graphics*, 2015. Preprint.
- [15] B. Doty and J. Kinter. Geophysical data analysis and visualization using the grid analysis and display system. In *American Geophysical Union 1993 Fall Meeting*, pages 6–10, 1993.
- [16] T. M. Hamill, G. T. Bates, J. S. Whitaker, D. R. Murray, M. Fiorino, T. J. Galarneau, Y. Zhu, and W. Lapenta. NOAA’s second-generation global medium-range ensemble reforecast dataset. *Bulletin of the American Meteorological Society*, 94(10):1553–1565, 2014/12/03 2013.
- [17] B. Hibbard and D. Santek. The VIS-5D system for easy interactive visualization. In *Visualization, 1990. Visualization '90., Proceedings of the First IEEE Conference on*, pages 28–35, 462, 1990.
- [18] G. Kindlmann, E. Reinhard, and S. Creem. Face-based luminance matching for perceptual colormap generation. In *Proceedings of the Conference on Visualization '02, VIS '02*, pages 299–306, Washington, DC, USA, 2002. IEEE Computer Society.
- [19] D. Lloyd and J. Dykes. Human-centered approaches in geovisualization design: Investigating multiple methods through a long-term case study. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2498–2507, Dec 2011.
- [20] A. M. MacEachren. Visualizing uncertain information. *Cartographic Perspective*, (13):10–19, 1992.
- [21] P. McCaslin. *Display Three-Dimensional (D3D) User's Guide, Version 4.1*. NOAA Environmental Research Laboratories Forecast System Laboratory, Boulder, CO, September 2001. http://laps.noaa.gov/d3d/home/D3DUG_Ver4.1.pdf.
- [22] K. Moreland. Diverging color maps for scientific visualization. In *Proceedings of the 5th International Symposium on Advances in Visual Computing: Part II, ISVC '09*, pages 92–103, Berlin, Heidelberg, 2009. Springer-Verlag.
- [23] D. Murray, J. McWhirter, S. Wier, and S. Emmerson. The Integrated Data Viewer: a Web-Enabled Application for Scientific Analysis and Visualization. In *19th Conference on International Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology*. American Meteorological Society, 2003.
- [24] T. Nocke, M. Flechsig, and U. Bohm. Visual exploration and evaluation of climate-related simulation data. In *Simulation Conference, 2007 Winter*, pages 703–711, 2007.
- [25] A. Pang. Visualizing uncertainty in geo-spatial data. In *Proceedings of the Workshop on the Intersections between Geospatial Information and Information Technology*, 2001.
- [26] K. Pöthkow, B. Weber, and H.-C. Hege. Probabilistic marching cubes. In *Proceedings of the 13th Eurographics / IEEE - VGTC Conference on Visualization*, EuroVis'11, pages 931–940, Chichester, UK, 2011. The Eurographics Association & John Wiley & Sons, Ltd.
- [27] B. E. Potter, J. A. Winkler, D. F. Wilhelm, R. P. Shadbolt, and X. Bian. Computing the low-elevation variant of the haines index for fire weather forecasts. *Weather and Forecasting*, 23(1):159–167, 2015/01/05 2008.
- [28] K. Potter, P. Rosen, and C. Johnson. From quantification to visualization: A taxonomy of uncertainty visualization approaches. In A. M. Dienstfrey and R. F. Boisvert, editors, *Uncertainty Quantification in Scientific Computing*, volume 377 of *IFIP Advances in Information and Communication Technology*, pages 226–249. Springer Berlin Heidelberg, 2012.
- [29] K. Potter, A. Wilson, P.-T. Bremer, D. Williams, C. Doutriaux, V. Pasucci, and C. R. Johnson. Ensemble-Vis: A framework for the statistical visualization of ensemble data. In *IEEE International Conference on Data Mining Workshops*, pages 233–240, 2009.
- [30] J. Sanyal, S. Zhang, J. Dyer, A. Mercer, P. Amburn, and R. J. Moorhead. Noodles: A tool for visualization of numerical weather model ensemble uncertainty. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1421 – 1430, 2010.
- [31] M. Sedlmair, M. Meyer, and T. Munzner. Design study methodology: Reflections from the trenches and the stacks. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2431–2440, Dec 2012.
- [32] J. K. Sivillo, J. E. Ahlquist, and Z. Toth. An ensemble forecasting primer. *Weather and Forecasting*, 12:809 – 818, 12/1997 1997.
- [33] S. J. Taylor, R. Bright, G. Carbin, P. Bothwell, and R. Naden. Using short range ensemble model data in national fire weather outlooks. In *Fifth Symposium on Fire and Forest Meteorology*, Orlando, FL, November 2003. Preprint.
- [34] J. Thomson, E. Hetzler, A. MacEachren, M. Gahegan, and M. Pavel. A typology for visualizing uncertainty. In *Proceedings of SPIE. Vol. SPIE-5669*, pages 146–157, 2005.
- [35] J. G. Trafton and R. R. Hoffman. Computer-aided visualization in meteorology. In R. R. Hoffman, editor, *Expertise Out of Context: Proceedings of the Sixth International Conference on Naturalistic Decision Making*. Expertise: Research and Applications Series. Psychology Press, New York, NY, 2007.
- [36] C. Ware and M. D. Plumlee. Designing a better weather display. *Information Visualization*, 12(3-4):221–239, 2013.
- [37] R. T. Whitaker, M. Mirzargar, and R. M. Kirby. Contour boxplots: A method for characterizing uncertainty in feature sets from simulation ensembles. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2713–2722, Dec. 2013.
- [38] D. Williams, T. Bremer, C. Doutriaux, J. Patchett, S. Williams, G. Shipman, R. Miller, D. Pugmire, B. Smith, C. Steed, E. Bethel, H. Childs, H. Krishnan, P. Prabhat, M. Wehner, C. Silva, E. Santos, D. Koop, T. Elmqvist, J. Poco, B. Geveci, A. Chaudhary, A. Bauer, A. Pletzer, D. Kindig, G. Potter, and T. Maxwell. Ultrascale visualization of climate data. *Computer*, 46(9):68–76, September 2013.