Human Factors in Streaming Data Analysis: Challenges and Opportunities for Information Visualization

Aritra Dasgupta, Dustin L. Arendt, Lyndsey R. Franklin, Pak Chung Wong, Kristin A. Cook

Pacific Northwest National Laboratory

Abstract

Real-world systems change continuously. In domains such as traffic monitoring or cyber security, such changes occur within short time scales. This results in a streaming data problem and leads to unique challenges for the human in the loop, as analysts have to ingest and make sense of dynamic patterns in real time. While visualizations are being increasingly used by analysts to derive insights from streaming data, we lack a thorough characterization of the human-centered design problems and a critical analysis of the state-of-the-art solutions that exist for addressing these problems. In this paper, our goal is to fill this gap by studying how the state of the art in streaming data visualization handles the challenges and reflect on the gaps and opportunities. To this end, we have three contributions in this paper: i) problem characterization for identifying domain-specific goals and challenges for handling streaming data, ii) a survey and analysis of the state of the art in streaming data visualization research with a focus on how visualization design meets challenges specific to change perception, and iii) reflections on the design trade-offs, and an outline of potential research directions for addressing the gaps in the state of the art.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Real-time data streams pose unique challenges for human-in-the-loop data analysis processes. Monitoring heterogeneous streams for emergency response or building situational awareness about potential cyber attacks are complex and demanding analytical tasks. Information visualization techniques are being widely adopted in such scenarios for helping analysts detect and synthesize fast-changing patterns and keep their mental model about the data in sync with the evolving stream. A key challenge in streaming visualizations is in presenting salient changes to the data in such a way that analysts can understand the context and relevance of the changes, and reason about their causes and implications in real time (Figure 1).

The field of streaming data visualization is maturing quickly, with a number of techniques being developed for event detection, handling text streams, analyzing social network data, etc. However, there is a need to develop a deeper understanding of how human perception and cognition can cope with complex changes in continually evolving data streams. Despite our high perceptual bandwidth, human attention span is limited. This implies that visualizations not only need to adapt to the fast rates of data streams but also need to pre-attentively present and emphasize salient changes by updating the underlying data through optimal encoding strategies. Two open questions in this context are: i) How does the state-of-the-art streaming visualization design address these challenges in change perception? ii) Can we systematically identify goals, tasks, and related design challenges for improving upon the state of the art?

To address these questions, in this paper we look at the streaming problem through the lens of perceptually motivated design problems for streaming data visualization. Existing surveys on state-of-the-art streaming data analysis have focused on techniques for mining patterns [ILG07] or methods for addressing the problem of scale [BHKP10, Joy09]. Researchers have also looked at the challenges for developing visual analytics methods [MFK12], as visualization techniques alone might not be able to solve many challenges associated with interactive streaming analysis at

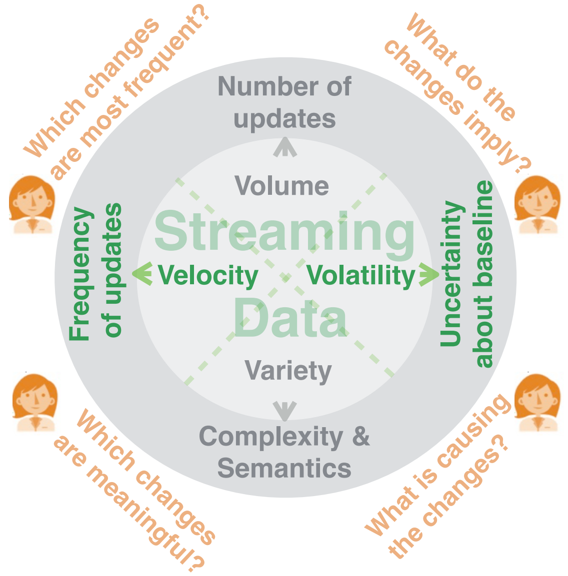

Figure 1: **Mapping properties of streaming data to challenges in change perception.** In addition to volume and variety that characterize much of the modern real-world data, velocity and volatility are key attributes of streaming data. While high velocity data leads to frequent updates that are hard for a human to track, volatility of the data implies unknown baseline behavior that can make it difficult for analysts to understand the causes and implications of the changes.

scale. In a complementary approach to these studies, we aim to understand the human-centered streaming-specific goals cutting across different domains, how they can be translated into visualization tasks, and how state-of-the-art visual representations are adapted to influence change perception in high-velocity streaming environments.

To this effect, we have three specific contributions in this state-of-the-art report. First, we describe the streaming-specific analysis requirements across different domains that can be synthesized into high-level goals and visualization tasks. Second, we study the mapping between these goals and the design space of information visualization techniques developed for handling data streams. In the process we highlight how machine-centered data transformations and human-centered design approaches have been used and provide a comparative analysis of these approaches. Third, we analyze the design challenges and trade-offs in a streaming context, the gaps in current research, and identify research directions that can address these gaps.

2. Methodology

Streaming data poses challenges for both automated methods like data mining and machine learning for extracting key patterns, and for visualization techniques that communicate the changing patterns to the analyst. In this section we define the scope of our work with respect to previous research and describe the analysis workflow that helped us critique the state of the art in streaming data visualization.

2.1. Definition and Scope

We adopt the definition of streaming data as proposed by Babcock et al. [BBD*02], where a stream is defined to be a continuous flow of data, where the system has no control over the volume of the arriving data or the number of updates, and only a small fraction of the whole stream is archived at any point of time, while the rest is discarded. These properties pose challenges for analysts observing the streams, who have to detect and understand fast-changing patterns and their implications in a dynamic environment (Figure 1). In this paper, we restrict our scope to understanding how streams influence change perception through visualizations. The challenges for streaming data for automated methods such as data mining or machine learning are outside the scope of this work, and can be found in other literature surveys [GZK05, Gam12].

A streaming visualization is one which adapts to the continuous flow of new data and follows certain strategies to display salient changes in the context of the past data. These strategies entail adjusting the encoding properties of static visualizations, such as the choice of visual variables or layouts, adding new elements to or removing old elements from the visualization as necessary, all while appropriately conveying changes in the underlying data to the end user. We do not consider streaming visualization techniques to have a new kind of design that is different from the design of static visualizations, instead, they build on the existing tenets of information visualization design principles. For this reason, the field of streaming data visualization also overlaps with many related research areas in visualization, such as time-series visualization, dynamic network visualization, and event detection techniques. What differentiates streaming visualizations from visualizations of dynamic or time-oriented data lies in how the visualization is to be used for real-time change perception. Streaming visualizations are characterized by needing a more immediate decision or action from the user, which often rules out batch-oriented analyses and traditional exploratory tasks.

Researchers have surveyed visualization techniques like dynamic graphs [BBDW14] and those applicable to unstructured data [WSJ*14]. Our goal in this paper is to go beyond specific techniques or data types and focus on the challenge of letting human analysts efficiently and effectively perceive changing patterns and derive streaming insights [EPC14]. We investigate how strengths and limitations of the human perception system influence real-time data visualization tasks and techniques, and reflect on the outstanding research challenges in this regard.

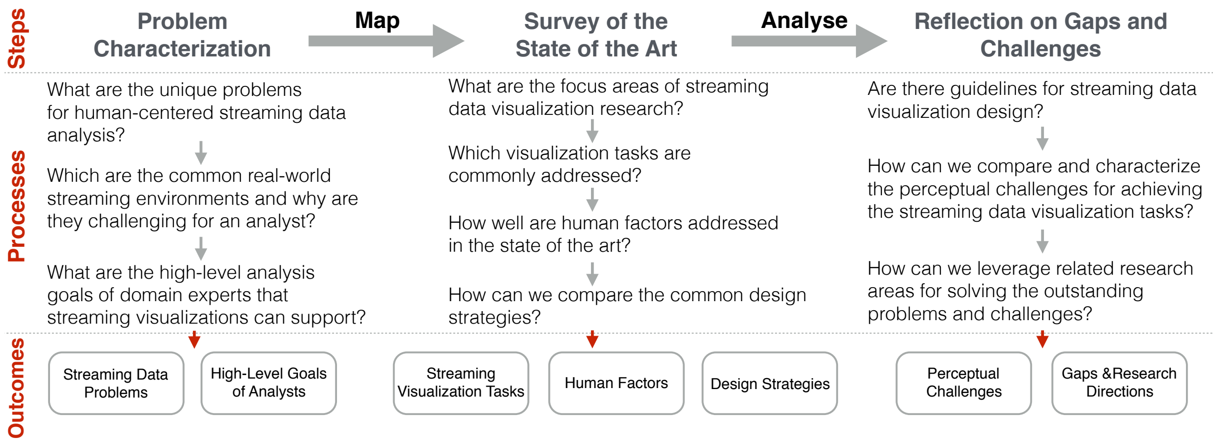

Figure 2: **Our workflow** was aimed at characterizing the human-centered challenges for streaming data analysis, mapping those to the landscape of existing techniques and models through a critique of the state of the art, and presenting an analysis of our findings reflecting on design trade-offs and future research opportunities.

2.2. Analysis workflow

When we started collecting papers related to streaming data visualization, we realized that unlike the other more mature areas of visualization research, such as techniques for high-dimensional or temporal data analysis, the area of streaming data visualization lacks a thorough characterization of the problems and an analysis of the existing solution space. This led us to develop a three-step workflow (Figure 2) as part of our methodology, including a problem characterization of domain-specific goals, their translation and mapping into visualization tasks for understanding change (state of the art), and an analysis of the visualization design space for change perception and their gaps and challenges. We describe them below:

Problem Characterization: This phase in our study (Section 3) was inspired by the domain and data characterization phases of Munzner's nested model for visualization design [Mun09]. We identified key domains in which streaming data analysis is an integral part by looking into past literature on streaming data analysis in the research areas of visualization, data mining, and machine learning. We leveraged the experience of two co-authors on this paper, both of whom have more than 15 years of experience in visualization and visual analytics, for grouping these domains based on similarity of the intended domain-specific goals. Next, we studied the literature related to the domain-specific goals and tasks in these domains. For example, for research related to cyber intrusion detection, we looked at survey papers that provide a descriptive summary of the cyber data analysis goals [JLSW10]. This helped us distill a set of high-level goals and tasks based on which streaming data domains can be grouped, characterized, and supported by visualization techniques. The outcome of this process (Figure 2) was a characterization of the complexity of streaming data and the associated goals and tasks across different domains.

Survey of the State of the Art: In this phase (Section 4), we first collected research papers related to streaming and dynamic data visualization, by focusing our search on leading visualization publications from the past twenty years. These included proceedings of the Information Visualization Symposium/Conference, and journals such as *IEEE Transactions of Visualization and Computer Graphics (TVCG)*, *Computer Graphics Forum*, *ACM CHI Conference*, and *IEEE PacificVis Symposium*. We initially collected a list of papers by searching with keywords "streaming", "dynamic", and "real-time". Starting from this initial seed, we also looked into their citations and included relevant papers from other venues. We noted the type of visualization technique or model, the evaluation strategies, and the key contributions of each paper for streaming data analysis. In parallel, we investigated the literature on human perception and cognition for identifying key visualization design challenges relevant to real-time assimilation of streaming patterns. Examples of such papers included those related to change perception and blindness [NHT01, ROC97], attention deficiency of human analysts under load [Lav05, MR98, MSCS05], memorability criteria for visualizations [BVB*13], and general principles of human perception as applicable to visualization [HBE95]. Using this analysis we described each paper based on how they addressed human-centered and machine-level problems and solutions for streaming data. The outcome of this phase (Figure 2) was a set of commonly used techniques and a systematic understanding of how specific human-centered challenges are addressed through the streaming visualization design space.

Reflection on Gaps and Challenges: In this phase we syn-

Velocity	Volatility	Domains	Representative Scenario	Main Challenge for Analysts
		Emergency Response Cyber Threat Mitigation	Which areas affected by flood need immediate action? What are the signatures of threats and how can they be mitigated?	Reason about changes based on unknown or evolving baseline behavior.
		Air-Traffic Control Stock Market Analysis	Are all flight paths adhering to the plan? Is this a significant event compared to the past?	Detect key changes that deviate from baseline behavior.
		Social/News Media Cyber Intrusion Detection	In these particular geographical areas what are people talking about? How many nodes are currently affected?	Apply heuristics to determine meaningful changes.
		Fraud Detection Simulation Modeling	Compared to the transaction history, is this a fraudulent transaction? How to reset the model parameters to get better simulation performance?	Use historical context and current data to decide on actions.

Figure 3: **Illustrating the relationships between complexity of streaming data and analysis scenarios** across different domains. High velocity and volatility of streaming data have posed unique challenges for an analyst. For different combinations of velocity (high, low) and volatility (high, low), we highlight the domains, representative scenarios, and analysis goals.

thesized our findings by analyzing the gaps in the state of the art by analyzing the perceptually motivated challenges and design trade-offs for handling change sensitivity of different visual representations (Section 5). This was followed by weighting these perceptually motivated challenges based on the high-level goals and tasks. This was because only a subset of design challenges are relevant for a certain goal and it was important to identify them for guiding visualization design. We focused our discussion based on the existing streaming visualization techniques and also based on techniques that can be potentially applied to address design challenges in a streaming scenario. We describe the different design trade-offs based on the visualization tasks and high-level goals. We draw connections with related research areas that can be utilized for further exploration and evaluation of these trade-offs. Finally we conclude by pointing out the outstanding gaps (Section 6) and the associated research directions that can be pursued for addressing those challenges.

3. Problem Characterization

In this section we characterize the problem of human-centered streaming data analysis through two key steps: i) analyzing the complexity of streaming data and its implications on change perception (Figure 1), and ii) identifying key domains that involve streaming data reasoning [DVCVHF09] and cross-cutting high-level goals of domain experts (Figure 3).

3.1. Streaming Data-Driven Change Perception

Streaming data is characterized by its continuous flow [BBD*02] and is often distinguished by its high velocity and volatility as compared to static data sources. As shown in Figure 1, the four dimensions of streaming data: volume, variety, velocity, and volatility [ZE*11, KŽB*14] have unique implications for the perception of *change* for an analyst in a dynamic environment.

Volume of changes in streaming data is quantified by the number of updates to the existing data and **velocity** is determined by the frequency of the updates. Large numbers of updates at a fast rate make it difficult for analysts to keep track of changes in a system. The frequency of updates can also be unpredictable, where there can be an influx of bursty data within a short span of time. In all these cases, an analyst tries to understand: *what are the frequent changes* in the system. In domains that require real-time monitoring for immediate decision making, like in air traffic control or emergency response, analysts have to be constantly careful not to miss any actionable changes and they typically depend on alerts from the system for making them aware of changes that need attention.

Variety of streaming data is characterized by the underlying heterogeneous data sources, leading to complex changes resulting from a combination of structured and unstructured data (e.g., social media). In such cases analysts have to make sense of change semantics, i.e., changes that are meaningful and significant, by understanding the context provided by the data.

Volatility of streaming data is mainly caused by unknown baseline behavior of the attributes of data that are being tracked. As described by Krempl et al., [KŽB*14], ever-changing patterns in a stream can cause a change of target variable or feature availability to explain the changes, and this can lead to change uncertainty. In such cases, analysts need help from the system in understanding the causes of the changes and what they imply for the immediate future. Examples of such changes are drifts and concept evolution in social networks. In domains such as social network analysis or cyber threat detection such volatile data implies that analysts are dependent on the system for summarizing and semantically integrating multiple pieces of information at different instances during the stream, otherwise such information might become unusable due to the evolution of the stream.

Examples: In Figure 3 we provide a high-level overview of the different domains that can be grouped based on the velocity and volatility of the data and provide representative scenarios for each. We categorize velocity into three classes based on the general frequency of the updates.

In the case of high velocity data, the update frequency is in seconds or minutes, as in emergency response situations when an event has happened and an analyst is constantly supervising a scenario. When combined with high volatility, the main challenge for the analyst is to often grapple with unknown baseline behavior at a fast rate. For example, the exact signatures of threats in a cyber system might be unknown, yet streaming patterns of activities might raise suspicion that defenders need to monitor the system and against which they need to take preventive action. Even in emergency response scenarios, situations are often dynamically evolving and analysts need to learn and decide in real-time about possible mitigation mechanisms.

In the case of medium velocity data, the update frequency is generally in minutes or hours, as in the case of air traffic control [LJFJ14] or stock market analysis. Combined with medium volatility, where the baseline behavior is mostly known, the main challenge for an analyst is to compare evolving changes to known baseline behavior and accordingly decide on a course of action. Medium or low velocity and highly volatile streaming data can be found in domains such as social media or cyber intrusion detection. In those cases, although baseline behavior might be unknown, certain heuristics (e.g., topic modeling in social media) can be used to provide insights into the data.

In the case of low velocity data, update frequency is of the order of hours to days, or periodic. An example scenario is fraud detection while credit card monitoring, which depends on a person's transaction frequency. In such cases, the main goal of the analyst is to compare current behavior of individuals or other entities with respect to historically known baseline behavior and accordingly decide whether action needs to be taken to flag anomalies or influence future patterns of behavior (e.g., model performance during simulation runs).

3.2. Domain-specific Goals

We consider a set of common domains where analysts are faced with streaming data such as social media, cyber security, emergency response, and financial domains. We selected these domains based on their usage in streaming data analysis literature and also by analyzing the difference in goals across these domains. We aimed for a more fine-grained analysis of goals and tasks than the one presented by Rohrdantz et al. [ROKF11] and came up with the following high-level streaming data analysis goals based on which different domains can be classified: building Situational Awareness (SA), Monitoring (Mon), and Event-Tracking (ET).

These domains span the three states of a data stream: past, current, and future. The different goals and scenarios are schematically represented in Figure 4, and described below. The transition among the past, current, and future states are dependent on the analysis scenario.

In the case of high velocity data, where updates are of the order of seconds or minutes, the transition between past and present is extremely rapid and often indistinguishable to the analyst. In the case of low-velocity data, where updates are of the order of hours or days, there is a distinguishable past. The future state indicates a state of the system that is a function of the current changes perceived by the analysts and the actions taken by them to influence the outcome of the changes. The future state is especially relevant where the analyst wants to take an action for influencing the stream: an emergency responder or cyber security analyst seeking to mitigate situations that demand urgent attention and that can can be resolved by their action.

The categorization of tasks ultimately helps us in identifying the visualization-specific design challenges in a streaming scenario.

Situational Awareness (SA): In building situational awareness analysts are mostly concerned with getting actionable insight from the system to influence the *future.* We adopt Endsley's [End95] definition of situational awareness, which is "a three-part process of perception, comprehension, and projection (into the future to make predictions) that leads to decision making and then to actions." The perception and comprehension task mainly involves reasoning about the current state, followed by a projection or prediction about the future. As noted by MacEachren et al. [MJR*11], these stages are closely related with the visual analytic loop where sense making and information foraging are key tasks for the analyst. The tasks can be exploratory in nature, as the analyst is often searching for causal relationships at different levels of detail. Exploring mitigation strategies or discovering unknown signatures for defending against cyber threats [JLSW10] or use of social media for enhancing awareness about an emergency situation [YLC*12] are common examples of such sense-making tasks, that generally involve highly volatile data. Reasoning and projection are key tasks towards the overall goal, and these tasks typically need the data to be represented at different levels of abstraction and in coordinated multiple views for achieving those tasks.

The SA-related analytical tasks can be synthesized into the following main types:

SA1: Summarize information from heterogeneous streaming

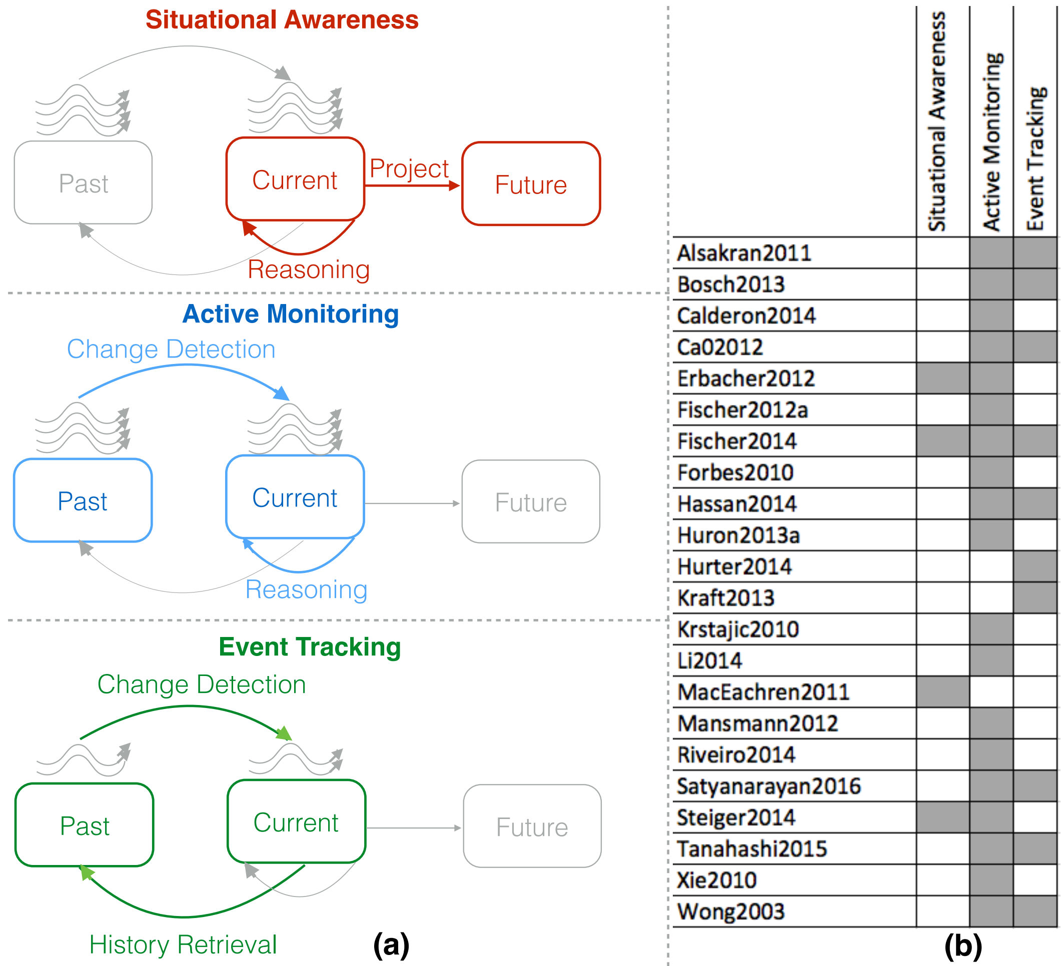

Figure 4: **High-level streaming-specific analysis goals as found in the literature.** The final set of goals we derived were: *Situational Awareness* (SA), *Monitoring* (Mon), and *Event Tracking* (ET). Compared to SA and Mon, ET is a goal in cases where the stream rate is low, that is, the temporal granularity of data update is of the order of days or weeks. In (a) the goals are broken down into relevant analytical tasks, and in (b) we show the distribution of these goals in state-of-the-art techniques.

data sources for identifying causal relationships behind the changing patterns.

SA2: Take domain knowledge into account and let analysts explore dynamic what-if scenarios.

Based on our survey, we found that very few papers (with the exception of [Erb12, FK14, MJR*11, SBM*14]) address situational awareness scenarios. In Figure 5 we see one such example where interactive feature selection is used for summarizing and reasoning purposes.

Active Monitoring (Mon): Active monitoring is the most common streaming data analysis goal where an analyst supervises a system in real-time in the face of high-velocity data. In most monitoring cases, baseline behavior is known by the analyst and they are aware of which changes need their attention. The problem that the human analysts face in monitoring tasks is that due to the velocity of data, in the absence of effective analysis tools, they might miss detecting a change. Monitoring can also be combined for more complex situational awareness tasks in domains with higher volatility, as when social network data is used for immediate emergency response or potential cyber threats need to be flagged by cyber defenders. Detection of patterns and reasoning about them in real time are the main tasks of the analysts. Active monitoring was the most common application scenario that we found, and the visualization tasks involved:

Mon1: Highlight changes in the stream and communicate trends and anomalies in the stream.

Mon2: Present salient changes that require human attention and differentiate among small and significant changes.

Common examples of stand-alone active monitoring

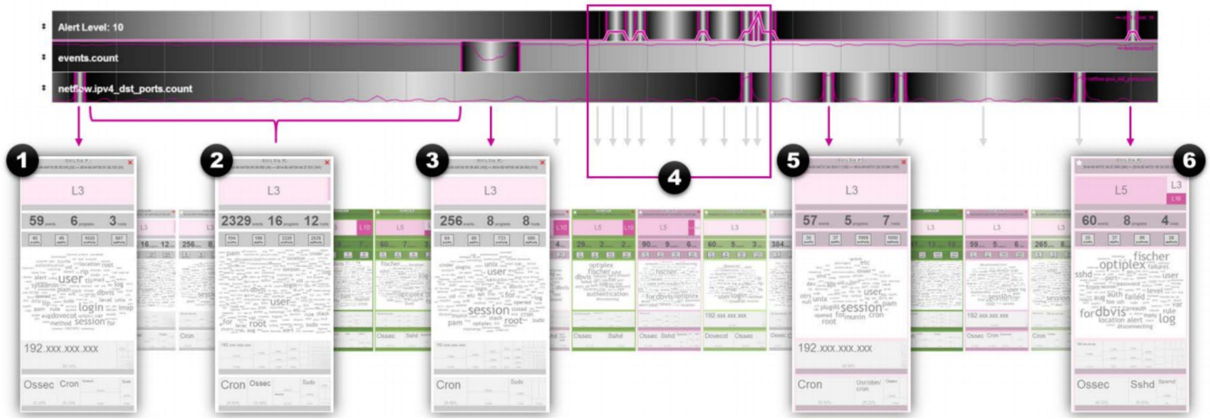

Figure 5: **Designing for enhancing situational awareness [FK14]** where exploratory feature selection is used for summarizing multiple time slices. Such exploration and summarization are necessary for stream reasoning and projection of future patterns.

tasks are air-traffic control or network intrusion detection [Axe00], where the analysts try to detect very specific trends or anomalies and know what they are looking for.

Event Tracking (ET): Event tracking is a goal when the current state of the system needs to be understood in the context of the historical information, and the analyst either wants to identify, compare, or retrieve events [WSJ*14]. Historical information is typically retrieved on demand, and a computational model is integrated with the data, with rules that flag whether an event has occurred.

Common examples are social media data analysis or fraud detection in credit card transactions. For comparing the significance of the incoming data stream, analysts are often interested in retrieving the history. Such event detection may also help predict future events, as in the case of simulation modeling [SEH*15], where relationships between model parameters and outputs can be understood based on the definition of events. ET involves the following tasks:

ET1: Understand the importance of current patterns based on the past context.

ET2: Compare current events to the past ones on demand.

4. Survey of the State of the Art

Our survey (Figure 6) focused on collecting examples of visualization techniques designed for addressing streaming-specific problems and solutions. In this regard, we differentiate between techniques developed for handling dynamic data, as opposed to streaming data. In the case of streaming data, an analyst only looks at a specific snapshot of the data at a particular instance of time and data is not stored in a machine's memory, as it is in the case of emergency response systems. In the case of dynamic data, while the data may be changing over time, old data can be stored in memory and used for event detection, as in credit card monitoring. In this section, we present an exhaustive list of real-time streaming visualization techniques, and a representative sample of dynamic visualization techniques which can be adapted to handle real-time streams. We use the following heuristic to decide a representative sample: we group dynamic visualization techniques into groups based on the visual representations and the tasks, and randomly select one technique that is representative of that group. For example, dynamic graphs form one group and papers such as [APP11, BPF14] are representatives of that group. For a more detailed survey on dynamic graph visualizations, we refer readers to the STAR paper by Beck et al. [BBDW14]. For each of the listed techniques, we study the data-specific problems and solutions, human-centered problems and solutions, and the associated evaluation strategies. Since the goal of this paper is to study problems and solutions specific to the human-in-the-loop situation, we present a detailed analysis of how streaming-specific challenges are handled by encoding strategies and compare their advantages and disadvantages.

4.1. Machine-level Problems and Solutions

The nature of streaming data is to naturally accumulate with little to no bounds, requiring visualizations to accommodate this growth. Along with data accumulation, missing or incomplete data, and heterogeneity of data are other problems that have been addressed in the visualization literature [FHL10,HSP14,Riv14,WSJ*14]. As shown in Figure 6, the following are the different data transformation strategies employed while designing visualizations for handling these problems.

Binning: Roughly a third of the papers we surveyed relied on binning. Binning is effective in streaming environments where the volume of data is so large that representing individual records is not practical. This technique introduces uncertainty into the visualization by foregoing a direct repre-

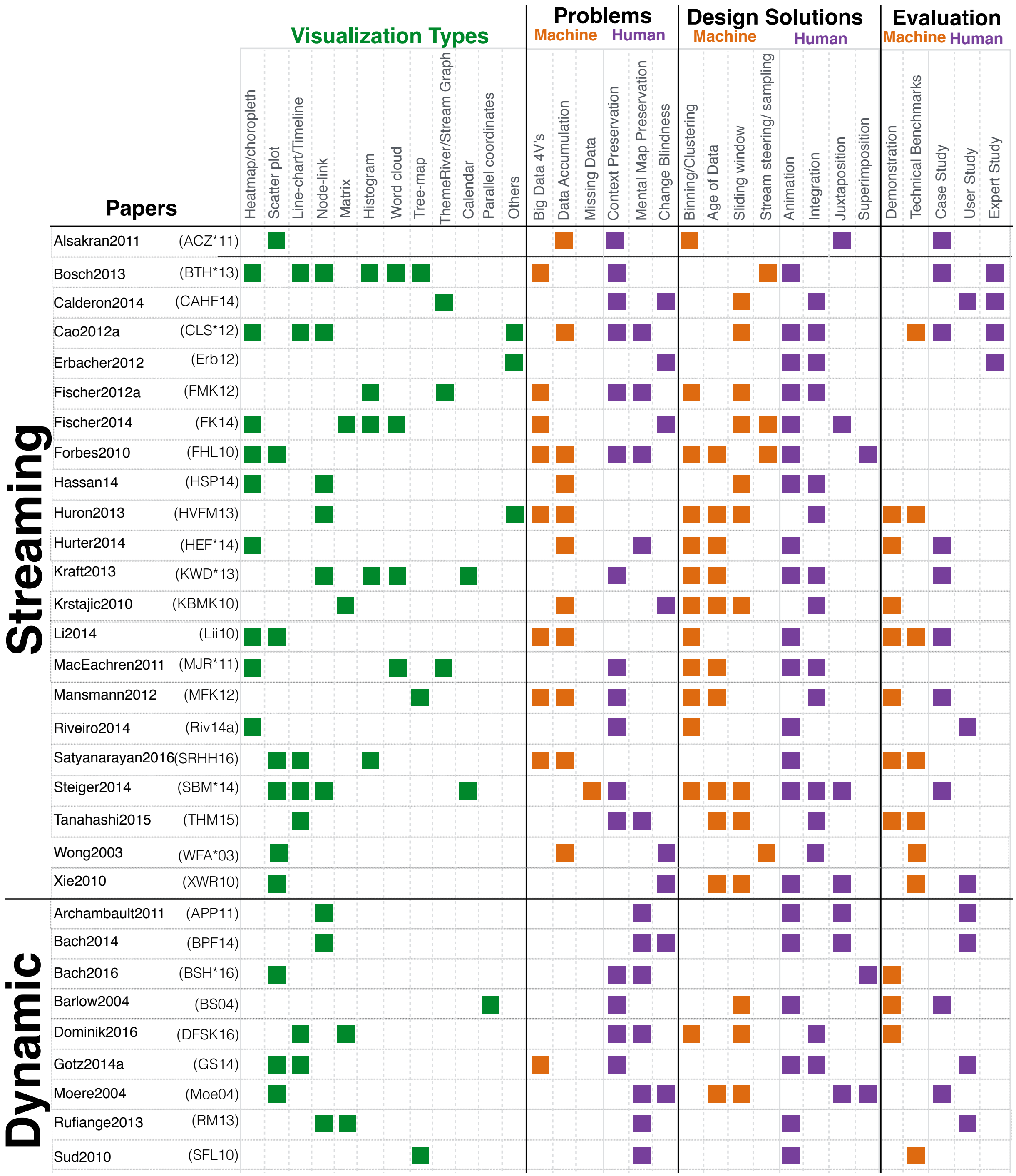

Figure 6: **Surveying the state of the art** based on the visualization techniques, machine-level and human-centered problems and solutions, and evaluation strategies adopted in the papers. We focused our analysis on how human factors are addressed by the design solutions and based on those factors we identified relevant design challenges and gaps.

sentation of each data record and instead representing counts of records across a set of intervals or faceted by a categorical attribute. This allows an arbitrarily large number of data records to be accounted for in the visualization by communicating the shape of the distribution of records to the user. The counts can be represented using a bar chart (e.g., spatial encoding of counts across intervals) to facilitate accurate visual comparison of counts. However, we have also observed binning used in conjunction with a map (e.g., heatmap, calendar, choropleth), where counts within geographical regions are shown, as is the case in [FHL10]. Time discretizes in natural ways (e.g., months, weeks, days) which can be used as bins to count records. This technique was used in [KWD*13] with a calendar visualization to represent the temporal intensity of tweets on a particular subject using a calendar heatmap. Clustering and binning can be used in conjunction when uniform discretization is not appropriate, as it was in [SBM*14].

Age of Data: The age of the data itself can be visually encoded for providing temporally relevant information to the user. Obvious ways to represent the age of data include encoding time spatially (e.g., on a timeline) or encoding age with size or color. Making younger data larger than older data, or giving younger data more contrast from the background compared to older data can draw the user's attention towards these elements. This is helpful under the assumption that younger data is more relevant or interesting than older data. Below we discuss several unique solutions to data aging from our literature survey.

Erbacher visualized streaming data in concentric circles, where newly arrived data was added to the outside of the circle, giving it more screen space [Erb12]. Conversely, less screen space was available to older data, which was assumed to be of lesser importance. Huron et al. use a sedimentation metaphor to facilitate graceful aging of data [HVFM13]. New data records accumulate on top of a pile and form layers like sediment. As the data ages, it compacts (data records become smaller) and is eventually replaced by an aggregate representation where individual data points are indistinguishable. Mansmann et al. also rely on compaction and aggregation to handle graceful aging of data in their tool, StreamSqueeze [MKF12]. Newer data is shown with the most detail. As data ages, it gets less screen resolution and is moved towards the right, exhibiting less level of detail than the new data.

Sliding Window: Many papers in our survey employ a sliding window as a method for handling the data aggregation problem of the stream. This is commonly implemented as an age-off policy for the data where any data older than an arbitrary window size of t is either archived or discarded. The visualization will often show high fidelity and interactive views of all data younger than t to help preserve context, however any context older than t is not available. In practice, the sliding window approach can be combined with binning or sampling techniques, but it is common to see integrated views (e.g., spatial encoding of time) used with a sliding window, by simply sliding the view to the left (when time is on the horizontal axis) as time elapses. The solution presented by Mansmann et al. [MKF12] is notable because it does not employ a fixed-width sliding window; instead the tool maintains a representation of all data over time. It addresses the data accumulation problem by reducing the fidelity of the information shown as time elapses. Newer data points are given much more screen space; older items get less space and are eventually only are represented within a histogram.

Sampling and Stream Steering: Sampling is a technique that can be used when the underlying data stream contains too much information for the back end systems to process in real time. In this case, sampling techniques, which likely involve machine learning models, can be used to pick and choose what data to process and eventually show to the user. Stream steering is an emerging area of research [FDCD12] that investigates how to allow users to influence the sampling process.This is accomplished in [BTH*13] by allowing users to create sophisticated filters that can identify interesting documents (e.g., tweets, news articles, blog posts) for the user in real time according to their topical content, and not just based on keyword matching.

4.2. Human-centered Problems and Solutions

In this section, we first discuss the human-centered perceptual design challenges by linking them to the properties of streaming velocity and volatility. Next, we discuss the different time encoding strategies we found in the literature that are leveraged to handle these challenges.

4.2.1. Perceptually Motivated Design Challenges

In our survey, we found three perceptually motivated design challenges that are explicitly or implicitly handled in the papers.

Context Preservation: In streaming scenarios, the perception of change is often influenced by the context. For example, a feature subset might be predictive of a particular target variable, but when target variables change, so do those subsets. In this case, the target variable provides the context for the importance of the feature subsets. Preserving the context is most important for event tracking scenarios [BS04, BPF14], where there is a need to capture the provenance and larger impact of data at different instants of time. With respect to streaming data, this often means preserving how the data have changed over time and even supporting the recall of older information which has suddenly regained relevance.

Mental Map Preservation: Changing patterns in a stream can be complex, and that can affect the stability of a visual display. If the magnitude of changes between time

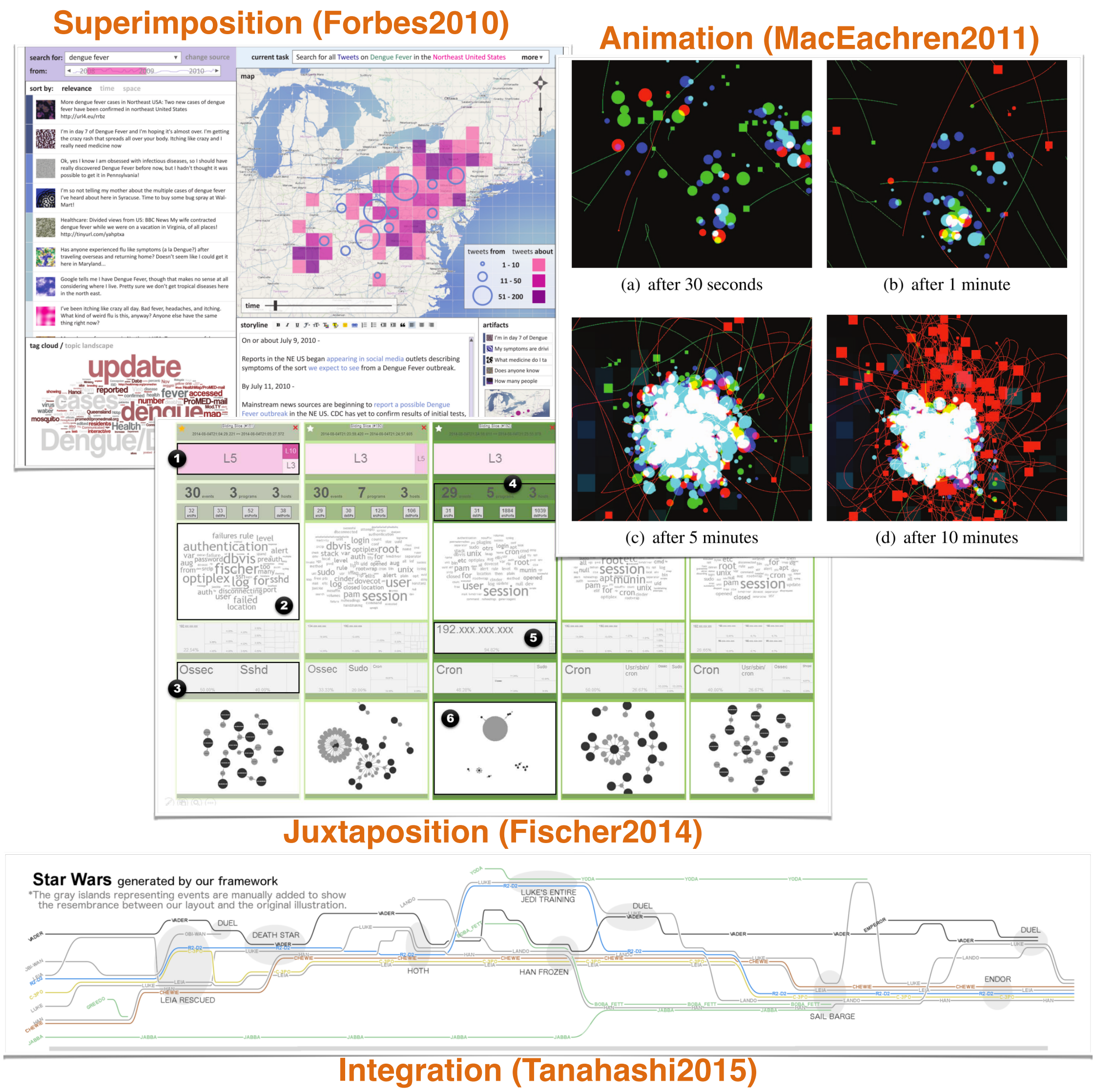

Figure 7: **Examples of time encoding strategies for streaming data visualization**, which are: superimposition, juxtaposition, integration, and animation. These strategies are discussed in Section 4.2.2.

steps are too large or too complex, analysts may find it difficult to preserve their mental map about emerging patterns [FHL10, HEF*14]. This is especially important for building situational awareness and exploring what-if scenarios in the face of volatile data. In such cases, the analyst is not always sure which patterns to look for, and what is causing them. To preserve the mental map, it is important for visual representations to optimize visual quality of a display, while at the same time provide multiple perspectives into the data for the analyst's mental model to be in sync with the evolving stream.

Change-blindness Prevention: The change-blindness problem is caused by high frequency and large number of changes in the stream, where in absence of pre-attentive visual cues [ROC97], the human vision system is unable to perceive changes even when able to see them. Ideally, a streaming visualization system should leverage the pre-attentive nature of visual variables for encoding changes in the data. These changes must either be analyzed for importance and selectively displayed or the design of a visualization must account for constant visual adjustment. Efforts to address the prevention of change blindness often focus on reducing visual clutter so as to make changes visually salient and visualizations with explicit time encoding as in ThemeRiver [HHN00]. In some cases, even explicit representations of time are augmented with eye-catching animations or other decorations to mark changes [CAHF14]. Prevention of change blindness should be a key design goal, especially in the case of active monitoring scenarios. Even outside the streaming scenario, researchers have aimed to characterize visualization techniques based on their tendency to cause change blindness [NHT01].

4.2.2. Time Encoding

All streaming visualizations result from design decisions related to how to handle time, or more specifically, that the data of interest will change over time. Our survey revealed four design patterns (Figure 7) for handling time for streaming data visualization. Drawing from the nomenclature introduced by Javed et al. [JE12] for static composite visualizations and later adapted by Beck et al. [BBDW14] for dynamic graph visualization, the four design patterns we identified for handling time in streaming visualization are: integration, juxtaposition, superimposition, and animation.

To help illustrate these concepts, we will use a "toy" streaming data example of visualizing the trajectory of a falling ball in real-time (Figure 8). Throughout these examples we use the term "model state" to describe the set of salient features and corresponding values that need to be communicated to the user to support the user's streaming analytics task. In the falling ball example, the model state is the height of the ball. Integration and juxtaposition map time into space; integration does so explicitly, by definition, and juxtaposition does so implicitly by ordering views chronologically. Animation shows only one version of the model state, which is updated in place, whereas the other three methods show copies of the model states at different times. Usually animation, juxtaposition, and superimposition will maintain the same scaling for colors and scales in order to facilitate accurate comparison across different times, and to preserve the mental map.

Animation: If we visually encode only the current height of the falling ball, and update this visualization as the position changes, we are using animation. The ball is always represented as a single entity in the visualization. Animation is a commonly used design pattern in streaming visualization, apparently because it is often straightforward (from an implementation standpoint) to use animation to adapt a static visualization to a streaming context. If a static visualization already exists that is suitable to communicate the model state within a snapshot or time window, then that visualization can be updated "in place" to reflect the current model state. If the frame rate of the visualization is high enough, and the model state changes smoothly, then this directly results in an animation effect. When this is not the case and updates to the model state occur at longer intervals, interpolation can be used so that visual elements transition smoothly, which might help preserve the user's mental map. Animation alone makes comparisons between the past and the present awkward for the user (e.g., requires playback, and seeking back and forth), which can cause problems with preserving the user's context. For monitoring, animations might be most effective as minor changes can be relevant [Blo05].

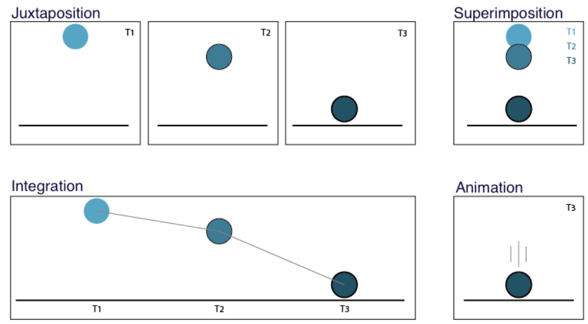

Figure 8: **Illustrating the different time encoding techniques we found in the literature** using a *falling ball* example, where T1, T2, T3 denotes the different states of the ball.

Integration: If we show the position of the ball as a function of time in a single visualization, such that time is spatially encoded, we are using integration. A new visual object representing the ball is added to the visualization for each new time frame. The visual "copies" of the ball can be connected (i.e., integrated) using lines to communicate they are the same object. When the position of the ball is drawn as a continuous function, this is often referred to as a "timeline" or "space-time plot." We consider these special cases of the use of integration for streaming data. Integrated streaming visualizations generally should explicitly handle data age-off. As time elapses, more and more copies of the model state will accumulate in the visualization because new copies are being added during each frame, and old objects that are no longer relevant should be removed to reduce clutter. A common solution is to remove all visual encodings of the model state older than some arbitrary time duration. In cases where time is encoded on the horizontal axis, this manifests as a sliding window effect, which is a commonly used solution. Integration can be helpful in preserving context because the present model state can be more easily contrasted against past states versus animation.

A unique example of integration is the use of storylines to convey changing relationships over time [THM15], which is shown in Figure 7. In this case the visualization is built of alternating representations of the relationships within a time window, and integrating lines connecting adjacent time windows. The integrating lines are styled to appear identical to the lines within the time window to give the appearance of a continuous timeline for each "character" in the storyline.

Juxtaposition: If we were to repeatedly take a snapshot of the ball as it falls, and then arrange those snapshots according to time (similar to a comic strip), we would be using juxtaposition. Similar to integration, an additional visual representation of the model is added to each frame. In this case, juxtaposition can be similar to integration, however, we consider juxtaposition to employ an implicit spatial encoding of time, whereas it is explicit in the case of integration. Juxtaposition often employs small multiples and is used in place of integration in a streaming context when views of the model state are too complex or challenging to be combined into a single view.

For example, juxtaposition was used in [FK14] to communicate how topics (distribution over words, represented as a word cloud) evolve over time; an example of this is shown in Figure 7. We see juxtaposition used frequently for dynamic graph visualization due to the abundance of effective open source graph layout algorithms for static graphs. A new view of the graph can be generated each frame using static methods, and then juxtaposed with the previous frames to show the change. Graphs during different time windows are drawn in separate, adjacent views to help the user understand how the topology of the graph is changing over time [BPF14, RM13]. Dynamic graph researchers hypothesize that minimizing changes across juxtaposed views over time is important to preserve the user's mental map, which should improve their ability to understand change over time in these data sets [APP11].

Superimposition: If we took a multiple exposure photograph of the falling ball (from the previous example), we would be employing superimposition. Superimposition compresses the model state at different times onto the same view using the same visual encoding. While we no longer would have an implicit or explicit spatial encoding of time, time can be communicated using other retinal properties such as color or size. Superimposition can help with preserving context and preserving the mental map, because the visual encodings are consistent over time, and because past and present are comparable in the same view. Furthermore, superimposition can be more space-efficient than integration and juxtaposition due to the reuse of space. Because views of the model state are closer together, making comparisons between past and present can be more efficient with superimposition as compared to the other methods discussed. A trade-off is that superimposed views will likely suffer from visual clutter and over plotting more readily than the alternatives.

Our survey revealed that superimposition was the least commonly used technique for handling time in streaming data. This was surprising given that this technique can be fairly easily implemented from a pre-existing static visualization, and has the advantage of showing data in context, as well as helping to preserve the mental map. Animation combined with superimposition (e.g., by showing the recent paths taken) is used in [FHL10, Moe04]. In [BSH*16] superimposition is used by projecting high-dimensional data at different time windows into a common two-dimensional space. Different instances of the model state are represented as single points to allow the user to see the relatedness of the model state over time, and detect patterns such as oscillation, stagnation, divergence, or recurrence.

4.3. Evaluation

Evaluations of the research efforts to address these streaming challenges have been diverse. In this work, we categorize evaluations into one of several forms. Case studies involve research that is presented as proof of concept and applied to a particular domain. During a case study, the application may have been provided to users not associated with the research team, but was done so without controlled tasks or conditions. User studies evaluate prototype research under controlled conditions with participants to produce quantitative results. Expert studies constitute evaluations which make use of presented research as a prototype and include review by domain experts not associated with the presenting researchers. Finally, technical benchmarks involve demonstrating the efficiency or data handling abilities of a prototype or algorithm without consideration of users or tasks. As shown in Figure 6, the problems of data accumulation, heterogeneous, and missing data are lacking in user-oriented evaluations. With the exception of [CAHF14] and [SBM*14], most research relies on technical benchmarks and demonstrations without determining if the visualization strategies are in fact effective and supporting user tasks. Conversely, most work addressing change blindness, context preservation, and mental map preservation relies on at least expert feedback or case studies to determine efficacy. In our survey we also found a general lack of quantitative user studies that either simulate real-world streaming environments or let domain experts perform certain tasks in a controlled setting. For some of the quantitative studies we also found update rates of the stream to be of the order of several minutes, which might not be a realistic scenario, especially where active monitoring of high-velocity streams is necessary.

In Figure 9 we describe how well human-centered design challenges are addressed by the techniques in terms of scalability, expressiveness [Mac86] for depicting change, change-blindness prevention, context-preservation, and mental-map preservation. Visualization types highlighted in bold have been explicitly applied in a streaming context. Connected scatter plots [HKF16], slope graphs [Sch14] and parallel coordinates [DKG15] can be used for encoding change, but have not been applied on streaming data.

5. Analyzing Design Challenges and Trade-offs

In this section we analyze the change sensitivity of the common streaming data visualization techniques. We classify the visualizations based on two encoding properties: which visual variables induces a perception of change in response to a stream, and what type of time encoding the visualization employs (Figure 9). Based on this classification we identify the

How Change is Perceived	Visualization Type	Visual Variable: position	orientation	size	color	area	Time Encoding: superimposition	integrated	juxtaposition	Design Trade-Off: Main Advantage	Main Drawback
Change in Position	**line chart/ timeline**	■					■			expressiveness	scalability
	connected scatter plot	■						■		context preservation	expressiveness
	slope graph	■						■			expressiveness
	parallel coordinates	■						■			mental map preservation
Change in Layout	**MDS plots**	■	■						■	scalability	mental map preservation
	temporal MDS plots	■	■				■				
	node-link diagram	■	■						■		
Change in Retinal Properties & Layout	**circle packing**			■			■			scalability	change blindness
	tree map					■	■				
Change in Position & Retinal Properties	**stream graph**	■				■	■		■	expressiveness	scalability
	visual sedimentation	■		■				■		context preservation	change blindness
	heat map	■			■		■			scalability	context preservation

Figure 9: **Investigating the design-trade-offs for the common streaming data visualization techniques that we surveyed**. The visualizations that are bold highlighted have been applied in the context of streaming data, while the other visualizations, such as the connected scatter plot [Kos16], slope graph [Sch14], and the parallel coordinates, have mostly been applied in the context of static data analysis.

merits and drawbacks of the techniques. For such identification we consider two more change sensitivity criteria in addition to the criteria of **context preservation**, **mental-map preservation** and **change-blindness prevention**. These are: **scalability** of a technique, which determines how a large number of changes can be accommodated, and **expressiveness** [Mac86], which determines if the salient changing patterns are clearly communicated to the analyst. To simplify our analysis, we identify the main advantage of a technique and its main disadvantage. We recognize that a more nuanced analysis might be necessary and a weighted score might be ideal in classifying the techniques, but we leave that for future work, as an extension to the analysis presented here. In addition to the techniques found in the survey, we analyzed techniques like slope graphs [Sch14], connected scatter plots [HKF16], and temporal MDS plots [JFSK16] that are relevant to the time encoding strategies and can be applied in the context of streaming data visualization.

5.1. Change in Position

Encoding change through *position* is the most common strategy used across many techniques. In a basic line graph or in timeline plots [SRHH16, DFSK16, GS14, SBM*14] time is encoded on one of the axes and superposing lines can encode the temporal trajectory of a stream. Line graphs are highly expressive and an optimal choice for showing temporal trends as they can communicate changing patterns quickly and effectively. For univariate data, especially in scenarios where analysts are looking for specific trends (e.g, variation in trajectories for air-traffic monitoring) as in the case of active monitoring, there are few better encoding choices than superimposing lines. However in presence of bursty data, a rapid change in scale of a variable can affect how the magnitude of change is perceived. Also for this approach, encoding change for multiple variables and a high volume of data can be challenging due to a lack of scalability.

Connected scatter plots [HKF16] and slope graphs [Sch14] are techniques used by the news media for integrating temporal information within a two-dimensional plot. While they are able to preserve the past context directly, and can be effective in historical retrieval tasks, they can be difficult to read and suffer from lack of expressiveness about key trends and anomalies. These plots are not suitable in monitoring situations, but might be used for reasoning and projection tasks for building situational awareness.

Parallel Coordinate Plots (PCP) represent multivariate data by encoding samples as contiguous line segments connecting pairs of variables assigned to vertical axes [ID87], and time can be an additional axis [HW13]. In a PCP the ordering of the axes determines which $n-1$ out of a possible $\frac{n\cdot(n-1)}{2}$ axis pairs are visible. A good ordering of the axes might reduce clutter and reveal patterns (e.g., clusters, trends) that are not otherwise visible with a different axis order. Many quality metrics for PCPs have been proposed with the assumption that an axis ordering that optimizes a given metric will improve the user's performance in certain analytical tasks [DK10]. PCPs have been used to show how a collection of multivariate objects changes over time by mapping time to the user's time (i.e., animation) [BS04, BBP08, The06]. In a streaming context, PCPs could be used to show recently collected samples (e.g., samples no older than t, or the k most recent samples). To our knowledge there has been no demonstration or systematic evaluation of a technique to automatically reorder PCP axes in a dynamic or streaming context and preserve the mental map of a user at the same time.

5.2. Change in Layout

Change in position of points coupled with the change in orientation of their groupings or their connections can lead to a perception of change in the overall layout of a particular view. This is most common in the case of two-dimensional projections of multidimensional data or node-link diagrams. Dimension embedding techniques project high dimensional data (e.g., multivariate or connectivity data) into a lower dimensional space (usually $D=2$) by placing similar samples (e.g., nearby in the feature space) close together in the lower dimensional embedding. Juxtaposed views of such MDS plots [WFA*03, XWR10] can be used to represent different slices of time. Force directed graph drawing algorithms can be considered a special case of this broader problem, where similarity is a binary relation equivalent to connectivity. Preserving high dimensional similarity (or distance) in low dimensional space is usually addressed directly by an optimization algorithm (e.g., MDS) or is an emergent outcome of the algorithm. This can help reveal features in the global structure of the data including clusters, holes, or relationships (when attributes not used by the embedding are encoded as shape, size, or color). The main advantage of this approach is that they are scalable with respect to large dimensionality of the data.

When data changes, the current spatial layout may become very sub-optimal if it is not adjusted to reflect the new relationships in the data. This problem has been explored in a streaming context by the dynamic graph drawing community, where a long standing hypothesis is that "preserving the user's mental map" is extremely important [APP11, BBDW14, PHG06]. There are also examples of MDS and related dimensional embedding techniques being used in a streaming context [ACZ*11, WFA*03], but these approaches have explicitly focused on preserving the user's mental mapping as the data changes, except in the case of [GHN12] where the problem of computing the positions of the dynamic multivariate data (i.e., streaming text) was transformed into a dynamic graph layout problem. This illustrates the close relationship between graph drawing and dimension embedding techniques–problems that appear to be unsolved, yet dynamic dimension embedding techniques may have good solutions, or at least starting points, in the dynamic graph drawing community.

5.3. Change in Retinal Properties and Layout

Retinal properties such as area, size, etc. have been used in several visualization techniques for encoding change. Techniques such as treemaps [JS91] and circle packing [WWDW06] are effective at representing large hierarchical datasets where each element also has a primary attribute that is encoded visually as area or size, and other secondary attributes that can be encoded with color, texture, etc. Updated data is generally superposed on the old data. These general techniques use visual containment to represent the hierarchy, and solutions generally use heuristics to quickly determine how to effectively place child elements within their parent container. In the case of treemaps, a popular heuristic produces squarified treemaps [BHvW00], where the aspect ratio of the elements is usually low. The circle packing algorithm tries to produce layouts where the area of the parent circle is not much larger than the total area of the child circles–in other words, it reduces unnecessary whitespace.

In a streaming context, change encoding using retinal variables and time encoding using the superposition strategy can quickly show a large number relevant changes, leading to high scalability, but in presence of a high frequency and large number of changes, analysts' attention might not be focused on relevant changes, and thus change blindness can occur. Additionally, small changes to the hierarchy or primary attributes might have a large effect on the layout of the visualization, and therefore the user's perception of the change. For example, with treemaps, the order of child nodes within a parent is sometimes determined by the primary attribute in order to improve the quality of the visualization. However, a relatively small change to the primary attribute of a single node can have a disproportionately large change to its order, which could cause confusion and lead to change blindness. Mental-map preservation is also a problem in this case. This problem has been investigated and addressed using dynamic voronoi treemaps [SFL10], which are designed to be stable against changes to the primary attribute, as well as to zooming. Dealing with changing hierarchies in a streaming context seems even more problematic, but because trees are special cases of graphs, solutions might be drawn from techniques for dynamic graph drawing.

5.4. Change in Position and Retinal Properties

A combination of position and retinal properties have been used in several techniques, and they vary based on the time encoding strategy used. Streamgraphs show the aggregate of many univariate time series by superposing individual time series, which are referred to as streams [BW08, DBH16, FMK12]. Color is used to allow the user to differentiate between the different streams, and to understand how the whole is composed of the individual parts (streams) over time. Time is usually encoded on the horizontal axis, leaving the primary attribute of the time series to be encoded as the thickness of the stream at that timepoint.

Therefore, the vertical position of a stream is dependent on the sum of the thicknesses of all the streams below it, plus a baseline. The height of the baseline is adjusted to improve the aesthetic quality of the visualization by decreasing, on average, the magnitude of "wiggles" throughout the visualization. The order of the streams also affects the legibility, and different heuristics can be applied, or the order can be determined intrinsically from the data. Streamgraphs have the advantage that the baseline at the "current" time can be calculated directly from the data and baseline at the previous time step, so accumulating new data will not necessarily require a recalculation of the entire baseline as data arrives. However, the optimal ordering of the streams is likely to change over time, so an open problem is determining when reordering is necessary, and managing this change in a way that preserves the users' context. The visual sedimentation [HVFM13] technique is able to better preserve the user's context by retaining the old data and by using an integrated encoding approach, but a large number of changes happening simultaneously can lead to change blindness due to the rapid transition between tokens and sediments.

Matrices and heatmaps [CLS*12,Riv14,LB14,KBMK10] are used to reveal relationships between pairs of variables in multi-dimensional data by mapping each variable to a spatial dimension to produce an image, and encoding the data at that point in the image with an appropriate color. Though it is common for the encoded variables to be continuous, many cases also exist where variables are categorical, as occurs in combinatorial data analysis. In such cases, the order of the rows and columns is arbitrary, but can have a significant impact on the usability of the visualization. The technique of reordering matrices to reveal patterns (which typically emerge as block diagonal structures) is known as matrix seriation [Lii10],

The optimal ordering of a matrix is likely to change as the underlying data changes. To the best of our knowledge, the visualization community has not explored matrix reordering in a streaming or dynamic environment. Similar to dynamic graph visualization and PCPs, questions arise such as: when should the matrix be reordered, and how does one compromise between minimizing the amount of change in the visualization and revealing interesting structures?

6. Research Directions for Addressing Gaps

Based on our survey and analysis presented in Sections 4 and 5, we reflect on the gaps in the state of the art in streaming data visualization and potential research directions that can help address these gaps.

Role of Visualization in Situational Awareness (SA): We found very few papers addressing the goal of achieving situational awareness for analysts using visualization. Cutting across various domains like cyber threat mitigation, or use of social media for emergency response, there is an urgent need for analytical tools that can be leveraged by analysts for achieving situational awareness. The opportunities for visualization and visual analytics in this regard were also highlighted in the visual analytics arena [TC06].

As mentioned earlier, SA involves dynamic reasoning over emerging streaming patterns while projecting the implications of these patterns on the future for decision-making. We posit that building exploratory, faceted visualizations on top of the integration of heterogeneous streaming data will be an important research direction towards this end. While such visualizations have been used for event detection [DGWC10], faceted displays can also help the analyst achieve situational awareness.

Task Models for Streaming Data: Our second finding was that there is a lack of systematic approaches towards translating high-level streaming data analysis goals into concrete visualization tasks. In this paper we have differentiated among three high-level goals: *SA*, *Mon*, and *ET*. We found that most papers explicitly address trend and anomaly detection tasks in a monitoring context. As mentioned by MacEachren [MJR*11], SA involves complex information foraging and sense-making tasks. However, there is little introspection on the instantiations of these tasks that need to be accomplished for reasoning, and exploring the implications of change in a streaming context. We also need to understand the varying levels of task complexities for a human analyst, as these tasks can often be demanding and lead to significant cognitive load [Lav05].

We posit that task models for streaming data need to be developed for a nuanced analysis of the low-level tasks an analyst has to perform in different scenarios. These will also be helpful in understanding how to leverage relevant research areas in visualization to solve streaming-related problems. For example, it has been shown that high-resolution displays are more effective when analysts have to perform complex sense-making tasks for synthesizing multiple pieces of information [AEN10]. Formal task models can help users bridge these gaps.

Handling Inattentional Blindness: We also found that the design space of visually encoding *change* needs to evolve for addressing the various streaming-specific challenges. Streaming changes are often unpredictable and bursty: patterns may appear and disappear at a later time, they can rapidly change across subsequent time steps, or their updates can vary based on different domains. In the face of such data, an important research problem that needs to be addressed is the problem of inattentional blindness on the human side. While change blindness can happen due to velocity of the data that is too high for a human observer to detect a change, high velocity coupled with high volume of changes can lead to important patterns escaping human attention. Since a data stream is constantly evolving, human analysts may struggle to pay attention to important states or transitions and this can lead to inattentional blindness [MSCS05]. This is especially relevant for the active monitoring task [MR98], where visualizations need to adapt for engaging analysts about the most salient changes in the system. Healey et al. [HE12] has pointed out the need to leverage the pre-attentive properties of retinal variables while designing visualizations to encode change. Effective use of color, motion etc., can effectively capture human attention, which is especially needed in the case of monitoring tasks. Related research areas where human reaction time in response to dynamic data [MDH95] has been studied, can be utilized for addressing the problem of inattentional blindness.

Change Presentation Versus Exploration: In many of the papers we surveyed, the visual encoding is an output of an underlying computational model that detects and quantifies the significant changes in the evolving patterns. In scenarios like situational awareness, it is important to present these patterns in a transparent manner so that the key insights can be efficiently discovered by the analysts. The goals of such presentation-based encoding can be very different from pure exploratory visualization encoding, as has been highlighted in the evolving research area of presentation-oriented visualizations [Kos16]. As pointed out by MacEachren [MJR*11], the data needs to be presented at appropriate levels of abstraction for the user to shift between different perspectives about the data. This is especially needed for complex sense-making tasks when an analyst is aiming for building situational awareness about the stream by understanding the causes and implications of the changes. However, the understanding behind the building blocks of such abstraction-based design is still in its infancy. Even an abstract representation has to generate enough confidence in an analyst to inspire confident decision making by highlighting the underlying causalities of the visual patterns. An important direction will be to investigate what design criteria should be used for generating presentation-oriented visualizations that effectively summarize and communicate important changes and the relevant context to the user.

Handling Uncertainty due to Incremental Updates: We found that the issue of incompleteness of information or uncertainty in a stream has not been substantially addressed in the visualization design phase. Often incremental approaches are used that incorporate information as it becomes available [Gam10] and this implies that the mental model of the analyst has to constantly adapt to the temporally evolving information. While machine-level pre-processing like binning, clustering, or use of sliding windows have been used, related research areas like incremental visualization [GFWS96, AS13] can be leveraged for designing and evaluating visualizations that handle irregular updates. Such incremental visualization will be needed to adapt to the changing frequency of data updates, handle bursty data, and still preserve the mental map of users and the context of past information. An open issue for incremental visualizations is that analysts might not be confident while making decisions due to the uncertainty caused by partially available information, and that can lead to a lack of trust [FPD12]. Novel incremental visualizations of the data [KBK11] that inform the analyst of both the uncertainty in the visual representation, as well as the context of the changes, can help bridge this gap between analytical uncertainty and trust.

Evaluation Studies, Metrics, and Benchmarks: A significant gap in streaming data visualization research is the lack of formal evaluation methods for qualitatively or quantitatively comparing the various design trade-offs [DPW*15]. We posit that not only do we require formal approaches like design studies with domain experts and controlled user studies by simulating or in actual streaming environments, we also need to establish new metrics based on the design trade-offs, some of which were described in Section 5. For example, in monitoring scenarios with high-velocity data, user engagement can be an important metric. The trade-off there is that human attention might be fixated on one set of patterns, while missing another set of patterns due to inattentional blindness. The issue of change blindness should also be investigated further: what metrics do we use to measure the effect of the time encoding strategies on change blindness?

Related areas of research that can be leveraged towards building such visualizations are metrics like memorability [BVB*13]. In the face of rapidly changing data and context, memorable visualizations can help analysts quickly recall past patterns and enable them to act on the currently observed patterns. Tasks where the search targets are not clearly defined, can benefit from explicit design criteria for increasing the memorability of key temporal trends and anomalies.

7. Conclusion and Future Work

We have presented a survey and analysis of the state of the art in streaming data visualization, by focusing exclusively on the relationship between the design space and complexities of change perception. In the process we have looked at how the different change dimensions such as frequency, amount, uncertainty, and complexity are accentuated by the velocity and volatility of data across common streaming domains, and affect human perception. We mapped these data-oriented problems to domain-specific goals for understanding challenges that domain experts face while analyzing streaming patterns. To meet these challenges visualization techniques have evolved over the years and our survey analyzed how well different encoding strategies are able to address the human factors. In turn, our analysis revealed several design challenges and trade-offs that can eventually be synthesized into a set of criteria for effective streaming data visualization design.

We believe that the findings and gap analysis in our study can be leveraged for developing a sustained research agenda around investigating how visualizations can better facilitate change perception in a streaming environment, and how different views can be integrated to provide a holistic perspective about the stream. The research agenda will be complementary to our recent focus in areas of stream data mining and big data visualization, and potentially lead to effective integration of automated methods and perceptually motivated visualization techniques for human-in-the-loop streaming data exploration.

8. Acknowledgment

The research described in this paper is part of the Analysis in Motion Initiative at Pacific NorthWest National Laboratory (PNNL). It was conducted under the Laboratory Directed Research and Development Program at PNNL, a multi-program national laboratory operated by Battelle. Battelle operates PNNL for the U.S. Department of Energy (DOE) under contract DE-AC05-76RLO01830.

References

[ACZ*11] ALSAKRAN J., CHEN Y., ZHAO Y., YANG J., LUO D.: STREAMIT: Dynamic visualization and interactive exploration of text streams. In *IEEE PacificVis Symposium* (2011), pp. 131–138. 14

[AEN10] ANDREWS C., ENDERT A., NORTH C.: Space to think: large high-resolution displays for sensemaking. In *SIGCHI Conference on Human Factors in Computing Systems* (2010), ACM, pp. 55–64. 16

[APP11] ARCHAMBAULT D., PURCHASE H., PINAUD B.: Animation, small multiples, and the effect of mental map preservation in dynamic graphs. *IEEE Transactions on Visualization and Computer Graphics 17*, 4 (2011), 539–552. 7, 12, 14

[AS13] ANGELINI M., SANTUCCI G.: Modeling incremental visualizations. In *EuroVis Workshop on Visual Analytics* (2013), pp. 13–17. 16

[Axe00] AXELSSON S.: *Intrusion detection systems: A survey and taxonomy*. Tech. Rep. 99-15, Department of Computer Engineering, Chalmers University of Technology, Sweden, March 2000. 7

[BBD*02] BABCOCK B., BABU S., DATAR M., MOTWANI R., WIDOM J.: Models and issues in data stream systems. In *Symposium on Principles of database systems* (2002), ACM, pp. 1–16. 2, 4

[BBDW14] BECK F., BURCH M., DIEHL S., WEISKOPF D.: The State of the Art in Visualizing Dynamic Graphs. In *EuroVis STAR* (2014). 2, 7, 11, 14

[BBP08] BLAAS J., BOTHA C. P., POST F. H.: Extensions of parallel coordinates for interactive exploration of large multi-timepoint data sets. *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (2008), 1436–43. 14

[BHKP10] BIFET A., HOLMES G., KIRKBY R., PFAHRINGER B.: Moa: Massive online analysis. *The Journal of Machine Learning Research 11* (2010), 1601–1604. 1

[BHvW00] BRULS M., HUIZING K., VAN WIJK J. J.: Squarified treemaps. *Joint Eurographics and IEEE TVCG Symposium on Visualization* (2000), 33–42. 14

[Blo05] BLOK C.: Dynamic visualization variables in animation to support monitoring. In *22nd International Cartographic Conference.* (2005). 11

[BPF14] BACH B., PIETRIGA E., FEKETE J.-D.: GraphDiaries: Animated Transitions and Temporal Navigation for Dynamic Networks. *IEEE Transactions on Visualization and Computer Graphics 20*, 5 (2014), 740–754. 7, 9, 12

[BS04] BARLOW N., STUART L. J.: Animator: A Tool for the Animation of Parallel Coordinates. *International Conference on Information Visualisation* (2004), 725–730. 9, 14

[BSH*16] BACH B., SHI C., HEULOT N., MADHYASTHA T., GRABOWSKI T., DRAGICEVIC P.: Time Curves : Folding Time to Visualize Patterns of Temporal Evolution in Data. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 559–568. 12

[BTH*13] BOSCH H., THOM D., HEIMERL F., PÜTTMANN E., KOCH S., KRÜGER R., WÖRNER M., ERTL T.: ScatterBlogs2: real-time monitoring of microblog messages through user-guided filtering. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (dec 2013), 2022–31. 9

[BVB*13] BORKIN M. A., VO A. A., BYLINSKII Z., ISOLA P., SUNKAVALLI S., OLIVA A., PFISTER H.: What makes a visualization memorable? *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2306–2315. 3, 17

[BW08] BYRON L., WATTENBERG M.: Stacked graphs - Geometry & aesthetics. *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (2008), 1245–1252. 15

[CAHF14] Calderon N. A., Arias-Hernandez R., Fisher B.: Studying animation for real-time visual analytics: A design study of social media analytics in emergency management. In *System Sciences (HICSS), 2014 47th Hawaii International Conference on* (2014), IEEE, pp. 1364–1373. 11, 12

[CLS*12] Cao N., Lin Y. R., Sun X., Lazer D., Liu S., Qu H.: Whisper: Tracing the spatiotemporal process of information diffusion in real time. *IEEE Transactions on Visualization and Computer Graphics 18*, 12 (2012), 2649–2658. 15

[DBH16] Di Bartolomeo M., Hu Y.: There is more to streamgraphs than movies: Better aesthetics via ordering and lassoing. In *Computer Graphics Forum* (2016), vol. 35, Wiley Online Library, pp. 341–350. 15

[DFSK16] Dominik J., Fischer F., Schreck T., Keim D. A.: Temporal MDS plots for analysis of multivariate data. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 141–150. 13

[DGWC10] Dörk M., Gruen D., Williamson C., Carpendale S.: A visual backchannel for large-scale events. *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (2010), 1129–1138. 15

[DK10] Dasgupta A., Kosara R.: Pargnostics: Screen-space metrics for parallel coordinates. *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (2010), 1017–1026. 14

[DKG15] Dasgupta A., Kosara R., Gosink L.: Vimtex: A visualization interface for multivariate, time-varying, geological data exploration. In *Computer Graphics Forum* (2015), vol. 34, Wiley Online Library, pp. 341–350. 12

[DPW*15] Dasgupta A., Poco J., Wei Y., Cook R., Bertini E., Silva C. T.: Bridging theory with practice: An exploratory study of visualization use and design for climate model comparison. *IEEE Transactions on Visualization and Computer Graphics 21*, 9 (2015), 996–1014. 16

[DVCVHF09] Della Valle E., Ceri S., Van Harmelen F., Fensel D.: It's a streaming world! reasoning upon rapidly changing information. *IEEE Intelligent Systems*, 6 (2009), 83–89. 4

[End95] Endsley M. R.: Toward a theory of situation awareness in dynamic systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society 37*, 1 (1995), 32–64. 5

[EPC14] Endert A., Pike W. A., Cook K.: From streaming data to streaming insights: The impact of data velocities on mental models. In *Proceedings of the 2014 Workshop on Human Centered Big Data Research* (2014), ACM, p. 24. 2

[Erb12] Erbacher R. F.: Visualization design for immediate high-level situational assessment. In *International Symposium on Visualization for Cyber Security* (oct 2012), ACM Press, pp. 17–24. 6, 9

[FDCD12] Fisher D., DeLine R., Czerwinski M., Drucker S.: Interactions with big data analytics. *interactions 19*, 3 (2012), 50–59. 9

[FHL10] Forbes A. G., Höllerer T., Legrady G.: behaviorism: a framework for dynamic data visualization. *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (jan 2010), 1164–1171. 7, 9, 10, 12

[FK14] Fischer F., Keim D. A.: NStreamAware: real-time visual analytics for data streams to enhance situational awareness. In *Proceedings of the Eleventh Workshop on Visualization for Cyber Security* (New York, New York, USA, nov 2014), ACM Press, pp. 65–72. 6, 7, 12

[FMK12] Fischer F., Mansmann F., Keim D. A.: Real-time visual analytics for event data streams. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing* (2012), ACM, pp. 801–806. 15

[FPD12] Fisher D., Popov I., Drucker S.: Trust me, i'm partially right: incremental visualization lets analysts explore large datasets faster. In *SIGCHI Conference on Human Factors in Computing Systems* (2012), ACM, pp. 1673–1682. 16

[Gam10] Gama J.: *Knowledge discovery from data streams*. CRC Press, Florida, USA, 2010. 16

[Gam12] Gama J.: A survey on learning from data streams: current and future trends. *Progress in Artificial Intelligence 1*, 1 (2012), 45–55. 2

[GFWS96] Garrett W. F., Fuchs H., Whitton M. C., State A.: Real-time incremental visualization of dynamic ultrasound volumes using parallel bsp trees. In *IEEE Visualization* (1996), pp. 235–240. 16

[GHN12] Gansner E. R., Hu Y., North S.: Visualizing streaming text data with dynamic graphs and maps. In *Graph Drawing* (2012), pp. 439–450. 14

[GS14] Gotz D., Stavropoulos H.: Decisionflow: Visual analytics for high-dimensional temporal event sequence data. *IEEE transactions on Visualization and Computer Graphics 20*, 12 (2014), 1783–1792. 13

[GZK05] Gaber M. M., Zaslavsky A., Krishnaswamy S.: Mining data streams: a review. *ACM Sigmod Record 34*, 2 (2005), 18–26. 2

[HBE95] Healey C. G., Booth K. S., Enns J. T.: Visualizing real-time multivariate data using preattentive processing. *ACM Transactions on Modeling and Computer Simulation 5*, 3 (1995), 190–221. 3

[HE12] Healey C., Enns J.: Attention and visual memory in visualization and computer graphics. *IEEE Transactions on Visualization and Computer Graphics 18*, 7 (2012), 1170–1188. 16

[HEF*14] Hurter C., Ersoy O., Fabrikant S. I., Klein T. R., Telea A. C.: Bundled visualization of dynamic graph and trail data. *IEEE Transactions on Visualization and Computer Graphics 20*, 8 (2014), 1141–1157. 10

[HHN00] Havre S., Hetzler B., Nowell L.: Themeriver: Visualizing theme changes over time. In *IEEE Symposium on Information Visualization* (2000), pp. 115–123. 10

[HKF16] Haroz S., Kosara R., Franconeri S. L.: The connected scatterplot for presenting paired time series. *IEEE transactions on visualization and computer graphics 22*, 9 (2016), 2174–2186. 12, 13, 14

[HSP14] Hassan S., Sänger J., Pernul G.: Soda: dynamic visual analytics of big social data. In *International Conference on Big Data and Smart Computing (BIGCOMP)* (2014), IEEE, pp. 183–188. 7

[HVFM13] Huron S., Vuillemot R., Fekete J.-d., Member S.: Visual Sedimentation. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2446–2455. 9, 15

[HW13] Heinrich J., Weiskopf D.: State of the Art of Parallel Coordinates. *STAR Proceedings of Eurographics* (2013), 95–116. 14

[ID87] Inselberg A., Dimsdale B.: Parallel coordinates for visualizing multi-dimensional geometry. In *Computer Graphics*. Springer, Tokyo, 1987, pp. 25–44. 14

[ILG07] Ikonomovska E., Loskovska S., Gjorgjevik D.: A survey of stream data mining. In *National Conference with International Participation (ETAI)* (2007), pp. 19–25. 1

[JE12] JAVED W., ELMQVIST N.: Exploring the design space of composite visualization. *IEEE Pacific Visualization Symposium (PacificVis)* (2012), 1–8. 11

[JFSK16] JAÌĹCKLE D., FISCHER F., SCHRECK T., KEIM D. A.: Temporal mds plots for analysis of multivariate data. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 141–150. 13

[JLSW10] JAJODIA S., LIU P., SWARUP V., WANG C.: *Cyber situational awareness*, vol. 14. Springer, New York, NY, 2010. 3, 5

[Joy09] JOY K.: *Massive Data Visualization: A Survey*. Mathematics and Visualization. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. 1

[JS91] JOHNSON B., SHNEIDERMAN B.: Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *IEEE Conference on Visualization* (1991), pp. 284–291. 14

[KBK11] KRSTAJIC M., BERTINI E., KEIM D.: Cloudlines: Compact display of event episodes in multiple time-series. *IEEE transactions on visualization and computer graphics 17*, 12 (2011), 2432–2439. 16

[KBMK10] KRSTAJIĆ M., BERTINI E., MANSMANN F., KEIM D. A.: Visual analysis of news streams with article threads. *International Workshop on Novel Data Stream Pattern Mining Techniques* (2010), 39–46. 15

[Kos16] KOSARA R.: Presentation-oriented visualization techniques. *Computer Graphics and Applications 36*, 1 (2016), 80–85. 13, 16

[KWD*13] KRAFT T., WANG D. X., DELAWDER J., DOU W., RIBARSKY W.: Less After-the-Fact: Investigative visual analysis of events from streaming twitter. In *IEEE Symposium on Large-Scale Data Analysis and Visualization (LDAV)* (oct 2013), pp. 95–103. 9

[KŽB*14] KREMPL G., ŽLIOBAITE I., BRZEZIŃSKI D., HÜLLERMEIER E., LAST M., LEMAIRE V., NOACK T., SHAKER A., SIEVI S., SPILIOPOULOU M., STEFANOWSKI J.: Open challenges for data stream mining research. *ACM SIGKDD explorations newsletter 16*, 1 (2014), 1–10. 4

[Lav05] LAVIE N.: Distracted and confused?: Selective attention under load. *Trends in cognitive sciences 9*, 2 (2005), 75–82. 3, 16

[LB14] LI C., BACIU G.: VALID: A web framework for visual analytics of large streaming data. In *International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)* (2014), pp. 686–692. 15

[Lii10] LIIV I.: Seriation and matrix reordering methods: An historical overview. *Statistical analysis and data mining 3*, 2 (2010), 70–91. 15

[LJFJ14] LUNDBERG J., JOHANSSON J., FORSELL C., JOSEFSSON B.: The use of conflict detection tools in air traffic management: an unobtrusive eye tracking field experiment during controller competence assurance. In *Proceedings of the International Conference on Human-Computer Interaction in Aerospace* (2014), ACM, p. 12. 5

[Mac86] MACKINLAY J.: Automating the design of graphical presentations of relational information. *ACM Transactions On Graphics (Tog) 5*, 2 (1986), 110–141. 12, 13

[MDH95] MATEEFF S., DIMITROV G., HOHNSBEIN J.: Temporal thresholds and reaction time to changes in velocity of visual motion. *Vision research 35*, 3 (1995), 355–363. 16

[MFK12] MANSMANN F., FISCHER F., KEIM D. A.: *Dynamic visual analytics facing the real-time challenge*. Springer, London, 2012. 1

[MJR*11] MACEACHREN A. M., JAISWAL A., ROBINSON A. C., PEZANOWSKI S., SAVELYEV A., MITRA P., ZHANG X., BLANFORD J.: SensePlace2: GeoTwitter analytics support for situational awareness. In *IEEE Conference on Visual Analytics Science and Technology (VAST)* (2011), pp. 181–190. 5, 6, 15, 16

[MKF12] MANSMANN F., KRSTAJIC M., FISCHER F.: StreamSqueeze: a dynamic stream visualization for monitoring of event data. In *Visualization and Data Analysis (VDA)* (2012), pp. 1–12. 9

[Moe04] MOERE A. V.: Time-Varying Data Visualization Using Information Flocking Boids. *IEEE Symposium on Information Visualization* (2004), 97–104. 12

[MR98] MACK A., ROCK I.: *Inattentional blindness*. MIT press Cambridge, MA, 1998. 3, 16

[MSCS05] MOST S. B., SCHOLL B. J., CLIFFORD E. R., SIMONS D. J.: What you see is what you set: sustained inattentional blindness and the capture of awareness. *Psychological review 112*, 1 (2005), 217. 3, 16

[Mun09] MUNZNER T.: A nested model for visualization design and validation. *IEEE Transactions on Visualization and Computer Graphics 15*, 6 (2009), 921–928. 3

[NHT01] NOWELL L., HETZLER E., TANASSE T.: Change blindness in information visualization: A case study. In *IEEE Information Visualization* (2001), pp. 15–22. 3, 11

[PHG06] PURCHASE H. C., HOGGAN E., GÖRG C.: How important is the"Mental Map"? - An empirical investigation of a dynamic graph layout algorithm. In *Graph Drawing* (2006), Springer Berlin Heidelberg, pp. 184–195. 14

[Riv14] RIVEIRO M.: Evaluation of normal model visualization for anomaly detection in maritime traffic. *ACM Transactions on Interactive Intelligent Systems 4*, 1 (2014), 1–24. 7, 15

[RM13] RUFIANGE S., MCGUFFIN M. J.: DiffAni: Visualizing dynamic graphs with a hybrid of difference maps and animation. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2556–2565. 12

[ROC97] RENSINK R. A., O'REGAN J. K., CLARK J. J.: To see or not to see: The need for attention to perceive changes in scenes. *Psychological science 8*, 5 (1997), 368–373. 3, 10

[ROKF11] ROHRDANTZ C., OELKE D., KRSTAJIC M., FISCHER F.: Real-Time Visualization of Streaming Text Data : Tasks and Challenges. *IEEE Workshop on Interactive Visual Text Analytics for Decision Making* (2011). 5

[SBM*14] STEIGER M., BERNARD J., MITTELSTÄDT S., LÜCKE-TIEKE H., KEIM D., MAY T., KOHLHAMMER J.: Visual analysis of time-series similarities for anomaly detection in sensor networks. *Computer Graphics Forum 33*, 3 (2014), 401–410. 6, 9, 12, 13

[Sch14] SCHWABISH J. A.: An economist's guide to visualizing data. *The Journal of Economic Perspectives 28*, 1 (2014), 209–233. 12, 13, 14

[SEH*15] STEED C. A., EVANS K. J., HARNEY J. F., JEWELL B. C., SHIPMAN G., SMITH B. E., THORNTON P. E., WILLIAMS D. N.: Web-based visual analytics for extreme scale climate science. *IEEE International Conference on Big Data* (2015), 383–392. 7

[SFL10] SUD A., FISHER D., LEE H. P.: Fast dynamic Voronoi treemaps. In *International Symposium on Voronoi Diagrams in Science and Engineering (ISVD)* (2010), pp. 85–94. 15

[SRHH16] SATYANARAYAN A., RUSSELL R., HOFFSWELL J., HEER J.: Reactive Vega : A Streaming Dataflow Architecture for Declarative Interactive Visualization. *IEEE Transactions on Visualization and Computer Graphics 22*, 1 (2016), 659–668. 13

[TC06] THOMAS J. J., COOK K.: A visual analytics agenda. *Computer Graphics and Applications 26*, 1 (2006), 10–13. 15

[The06] THERON R.: Visual analytics of paleoceanographic conditions. In *IEEE Symposium on Visual Analytics Science and Technology* (2006), pp. 19–26. 14

[THM15] TANAHASHI Y., HSUEH C.-H., MA K.-L.: An Efficient Framework for Generating Storyline Visualizations from Streaming Data. *IEEE Transactions on Visualization and Computer Graphics 21*, 6 (2015), 730–742. 11

[WFA*03] WONG P. C., FOOTE H., ADAMS D., COWLEY W., THOMAS J.: Dynamic visualization of transient data streams. In *IEEE Symposium on Information Visualization* (2003), IEEE, pp. 97–104. 14

[WSJ*14] WANNER F., STOFFEL A., JÄCKLE D., KWON B. C., WEILER A., KEIM D. A.: State-of-the-art report of visual analysis for event detection in text data streams. *Computer Graphics Forum 33*, 3 (2014), 1–15. 2, 7

[WWDW06] WANG W., WANG H., DAI G., WANG H.: Visualization of large hierarchical data by circle packing. In *SIGCHI conference on Human Factors in computing systems* (2006), pp. 517–520. 14

[XWR10] XIE Z., WARD M. O., RUNDENSTEINER E. A.: Visual exploration of stream pattern changes using a data-driven framework. In *International Symposium on Visual Computing* (2010), Springer, pp. 522–532. 14

[YLC*12] YIN J., LAMPERT A., CAMERON M., ROBINSON B., POWER R.: Using social media to enhance emergency situation awareness. *IEEE Intelligent Systems 27*, 6 (2012), 52–59. 5

[ZE*11] ZIKOPOULOS P., EATON C., ET AL.: *Understanding big data: Analytics for enterprise class hadoop and streaming data*. McGraw-Hill Osborne Media, IBM, 2011. 4