DensityBars: A Space-Efficient Visualization for Event Temporal Distribution

Mingwei Lin*
School of Software Engineering
South China University of Technology
Guangzhou, Guangdong, China
ptlmw666@gmail.com

Qin Huang*
School of Software Engineering
South China University of Technology
Guangzhou, Guangdong, China
h2513561399@gmail.com

Zikun Deng†
School of Software Engineering
South China University of Technology
Guangzhou, Guangdong, China
Key Laboratory of Big Data and Intelligent Robot (SCUT), Ministry of Education
Guangzhou, Guangdong, China
zkdeng@scut.edu.cn

Tobias Schreck
Institute of Visual Computing
Graz University of Technology
Graz, Austria
tobias.schreck@tugraz.at

Yi Cai
School of Software Engineering
South China University of Technology
Guangzhou, Guangdong, China
ycai@scut.edu.cn

Abstract

Event temporal distribution analysis aims to capture both global (e.g., rises and peaks) and local patterns (e.g., frequent occurrences and sudden absences). Traditional charts typically rely on adjusting binning granularities to reveal such patterns. However, this strategy forces a trade-off between global clarity and local detail and may require considerably more screen space as the number of bins increases, which limits its applicability in space-constrained visual interface design. In this paper, we propose DensityBars, a space-efficient visualization that embeds fine-grained density heatmaps of event occurrences into the coarse-grained bar chart to convey both global and local patterns simultaneously. Two real-world use cases and two formal user studies demonstrate its effectiveness and usability. Insights from studies provide valuable implications for the visual design of temporal distribution visualizations.

CCS Concepts

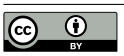
• **Human-centered computing** → **Visualization**.

Keywords

Event occurrence, temporal visualization, temporal distribution

ACM Reference Format:
Mingwei Lin, Qin Huang, Zikun Deng, Tobias Schreck, and Yi Cai. 2026. DensityBars: A Space-Efficient Visualization for Event Temporal Distribution. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26), April 13–17, 2026, Barcelona, Spain.* ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3772318.3791169

*Both authors contributed equally to this research.
†Zikun Deng is the corresponding author.



CHI '26, Barcelona, Spain

ACM ISBN 979-8-4007-2278-3/26/04
https://doi.org/10.1145/3772318.3791169

1 Introduction

Events, such as server downtime, urban traffic congestion, and heavy air pollution, represent critical data across various domains. A core objective in event analysis is temporal distribution analysis, which explores how events occur over time. In this context, an "event" refers specifically to "event occurrence" and can be recorded as only a timestamp t when it occurs. Temporal distribution analysis can be performed from global and local perspectives, each offering unique insights. For example, in studying air pollution events in Beijing, the global perspective assesses patterns across an entire year, addressing questions like, "*Which season has the most air pollution events?* [20]" This perspective reveals long-term trends. Conversely, the local perspective focuses on shorter periods, such as weeks compared to the season above, to answer more specific questions like, "*Which week had the highest frequency of air pollution events?* [23]" This detailed analysis uncovers short-term patterns that may be missed in the global perspective. Together, these perspectives offer a comprehensive understanding of temporal dynamics [5, 68].

Many visualizations, such as point plots (Fig. 1a), line charts (Fig. 1b), and bar charts (Fig. 1c), are popularly used for visualizing temporal distributions. When utilizing these charts for event temporal distribution analysis, event occurrences must be binned, typically by aggregating them over specific time intervals or periods, such as weeks or months. We denote the length of the aggregating periods as **bin size**. Take Fig. 1c as an example. The bin size is one month, and the height of the first bar represents the number of event occurrences falling into January. The choice of bin size critically influences a chart's effectiveness in representing temporal distributions [13, 52]. A large bin size (e.g., one month) is often selected to reveal global patterns but inevitably obscures finer details. Conversely, a small bin size (e.g., one day) can expose such details, yet it may clutter global patterns and consume much screen space. This trade-off becomes challenging in multi-view visual analytics systems, where designers must balance clear temporal pattern representation with the spatial constraints of coordinated layouts.

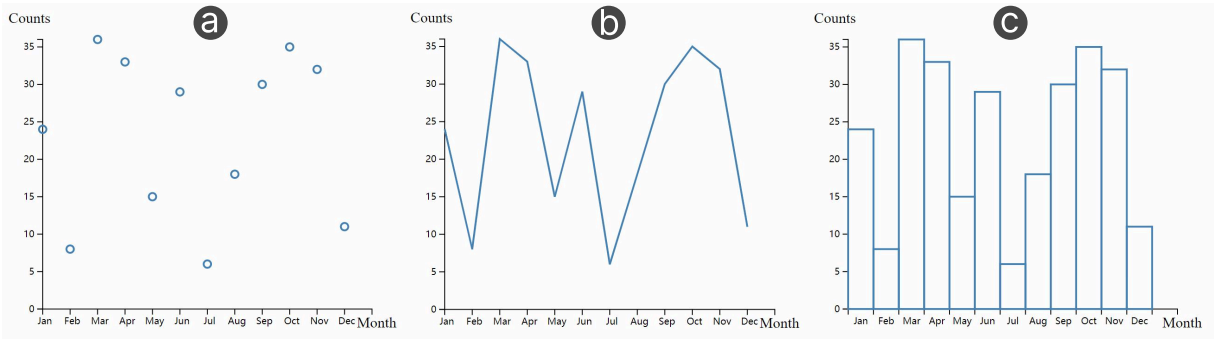


Figure 1: The same dataset is visualized with (a) the point plot, (b) the line chart, and (c) the bar chart, respectively.

A **concrete scenario** is presented in Fig. 2, where Tom is designing a visual analytics system to visualize the spatiotemporal characteristics of earthquake events. Fig. 2a1, a2, and a3 are designed to show spatial distribution, temporal distribution, and magnitude distribution of events, respectively. From Fig. 2a2, Tom finds it hard to identify which week has the most earthquakes because the bin size is one month. With a smaller bin size (Fig. 2b), Tom can find it is the last week of February, but in contrast, he cannot quickly identify October, the month with the most earthquakes. Moreover, such a bin size consumes much horizontal space, making the chart incompatible with other charts in layout. Inclusion of additional view for supplement (Fig. 2d) suffers from the same issue.

To address these limitations, prior work has proposed automatic bin size selection [18, 24] or variable bin sizes [65] to preserve both global and local patterns. However, these methods can yield unintuitive bins (e.g., 4.3 days or 3.8 hours). Other approaches rely on interaction, allowing users to adjust bin sizes on the fly [4] or explore local patterns within the global context using focus+context techniques [26], but incurring additional user effort. Observing that the vertical space within bars remains underutilized, we introduce visual encoding within them, enhancing bar charts without using additional screen space and preserving the possibility of further improvements through interaction.

In this study, we propose a fine-grained, space-efficient bar chart that embeds local temporal distributions within the vertical space (Fig. 2e). We name it DensityBars. As usual, the height of each bar represents the total number of events within the given period, while an embedded heatmap within each bar indicates the frequency of events during the period, with darker colors signifying higher event densities. This design overcomes the limitations of traditional charts by eliminating concerns over bin size, retaining the global information of the original chart while simultaneously offering enhanced detail within a constrained space. With DensityBars, Tom can adopt Fig. 2e without changing the interface layout and identify that the most earthquakes occurred in October and three short periods (enclosed by orange dashed rectangles) when earthquakes frequently occurred.

To assess the effectiveness of DensityBars, we first present real-world use cases, demonstrating that it is versatile and applicable across various domains and capable of supporting both global and local temporal analyses. We also conduct two task-based user studies, comparing DensityBars against alternative designs. The results show that our design performs as well as or better than these alternatives in both task accuracy and completion time, while using less screen space, making it ideal for space-constrained interfaces.

In sum, the contributions of this study are as follows:

- We formulate a visualization problem of designing visualizations that effectively display event temporal distributions under space constraints posed by multi-view layouts.
- We propose a novel visual design, DensityBars, by efficiently embedding fine-grained density heatmaps of event occurrences within a coarse-grained bar chart.
- We evaluate DensityBars with real-world cases and user studies, providing reflections on event temporal distribution visualization.

2 Related Work

This section discusses the prior studies related to temporal data visualization and static visualization of event occurrences. Finally, we drill down into the prior studies related to the bar chart.

2.1 Temporal Data Visualization

Event occurrences are essentially a kind of time-varying data. One of the fundamental tasks is to identify the temporal patterns of event occurrences, such as periodicity [43], rising/falling trends [56], and frequent/infrequently occurring time periods [19]. In the field of data visualization, approaches for visualizing time-varying data can be broadly categorized into two types: animation-based methods and static charts [21].

Animation-based methods require users to track changes over time via animated visual elements [27]. Specifically, events can be modeled as units, and then the temporal patterns of them can be reflected by dynamically displaying these units, for example, appearing one after another [8, 64]. However, animation-based methods often impose cognitive loads on users as they have to keep different frames of visualization in mind [46].

Static charts, in contrast, present dynamic data in a static format, potentially reducing this additional burden. Commonly seen static visualizations for the event temporal distribution include point plots, line charts, bar charts, and heatmaps [3, 42, 45]. These visualizations usually demand that the events be binned to

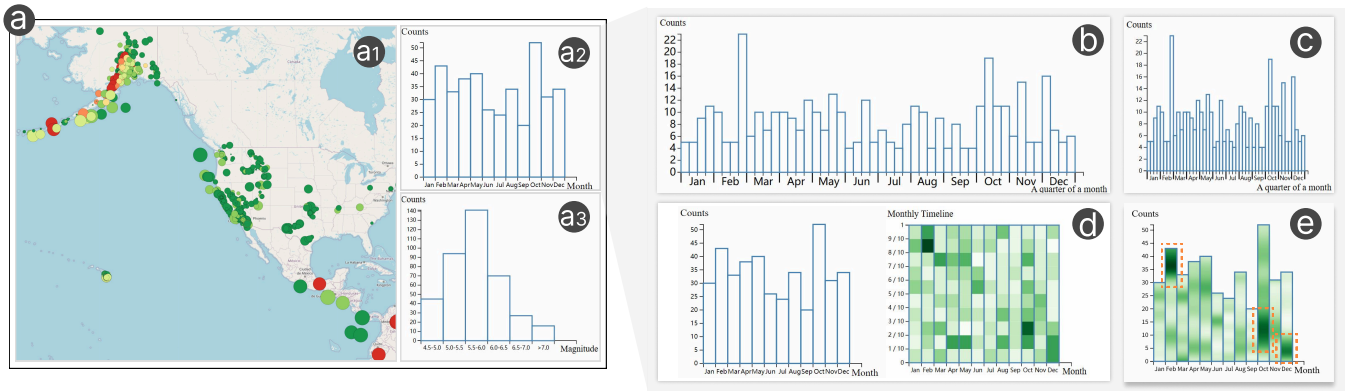


Figure 2: The scenario motivating this study. In this scenario, Tom is designing (a) a visual analytics system to visualize the spatiotemporal characteristics of earthquake events. The system interface comprises three views to visualize (a1) the spatial distribution, (a2) the temporal distribution, (a3) the magnitude distribution of events. The bar chart with the bin size of one month for the temporal distribution cannot reveal local or finer-grained temporal patterns. Tom explores various solutions. (b, c) The bar chart with a smaller bin size cannot effectively show global patterns. (b) A wider chart or (d) multiple views requires significantly changing the layout of the original system interface. (e) Our DensityBars is a good choice for Tom.

produce interpretable summaries and maintain readability. **Point plots** [2, 72] encode occurrences as discrete symbols positioned by time and count, but their sparse visual continuity limits effective trend perception [59]. **Line charts** [11, 44, 75] enhance continuity by connecting adjacent bins and are effective for revealing global temporal trends [51, 70]. **Bar charts** [49, 69, 73] represent binned counts using aligned rectangular bars, for example, Google Scholar citation statistics. This design makes them effective for explicit time–value mapping and accurate cross-bin comparison [2, 22], at the cost of reduced effectiveness in conveying continuous trends. **Heatmaps** [25, 29] use color to encode binned counts across two temporal dimensions and support multi-resolution analysis, but color-based encodings are generally less effective than position for estimating precise values and comparing patterns across bins [71].

Furthermore, prior work has explored kernel density estimation-based temporal visualizations to eliminate the need for binning. CloudLines [41] applies a kernel density estimation of the series of events to obtain a continuous density function, which is then encoded in the size and transparency of circular marks to provide a continuous overview of the event distribution. Similarly, Seebacher et al. [54] modeled event distributions as a continuous density function using kernel density estimation and visualized the resulting densities as area charts along a timeline. These approaches are well-suited for conveying continuous distributions when ample horizontal space is available; however, their continuous encodings typically rely on a sufficiently long time axis, which can make it difficult to display the complete distribution in constrained layouts.

We focus on bar charts in this study due to their strength in supporting direct and reliable comparison of event counts across time. Building on bar charts, we first examine the widely used binning strategy in static temporal visualizations and distill design insights that can inform other binned temporal charts. Motivated by these insights, we propose DensityBars, a new visual design for revealing event temporal distributions. Compared to kernel density estimation-based temporal visualizations, DensityBars integrates density cues into existing bins without increasing the horizontal space. We also discuss the opportunities and challenges of adapting this design concept to other chart types, such as line charts (see Section 7.3 for details).

2.2 Bar Chart

As a classic and fundamental visualization method, the bar chart has been extensively studied in terms of user perception and improvement to its inherent limitations.

General perception. Cleveland and McGill [12] investigated the impact of various bar chart designs on the accuracy of viewers' perceptual tasks, demonstrating that aligned bar charts performed better than stacked bar charts. Building on Cleveland and McGill's findings, Talbot et al. [62] further explored the sources of errors in participants' responses. They found that short bars are more challenging to compare, and the labeling of bars also influenced accuracy. The visual arrangement of bars in bar charts has also been studied [34, 35, 74]. For example, Indratmo et al. [34] compared three types of stacked bar charts (classic, inverting, and diverging) in attribute-comparison tasks. They found that the inverting design enabled faster overall-attribute comparisons, largely because its visual alignment is more consistent. Jardine et al. [35] examined how different spatial arrangements (e.g., stacked, superposed, mirrored) affect users' ability to compare statistical summaries such as mean and range across bar charts. Results showed that mean comparison mainly depends on global features, whereas range comparison depends more on local features. This suggests that different statistical judgments naturally invoke different perceptual levels, offering a perceptual basis for multi-level visual encodings.

Our study also examines users' perception of bar charts, but with an emphasis on their ability to discern event temporal distributions.

Bin size selection. When analyzing event temporal distributions, it is important to capture both global, long-term trends and local, short-term patterns [5]. The choice of bin size critically shapes how bar charts reveal these patterns [55]. For example, Sahann et

al. [52] showed that appropriately reducing the bin size can improve the recognition of underlying distributions when the data are well-defined and sufficiently sampled. Correll et al. [13] found that a smaller bin size makes local patterns, such as spikes and gaps, more salient.

Existing guidelines, such as the square root [47], Sturges [61], Doane [18], Scott [53], and Freedman-Diaconis [24] rules, offer methods for determining appropriate bin sizes to reveal the inherent data distribution. However, these guidelines ignore the inherent temporal dynamics of event occurrences and overlook the constraints imposed by the available pixel space. Some strategies involve adopting irregular or adaptive binning [31, 37] but may obscure meaningful semantic [55] and thereby hour-of-day, day-of-week, or day-of-year trends. For instance, these methods could yield inconsistent bin sizes, such as one spanning 13 days or 53 hours, thereby undermining the clarity and interpretability of the temporal distribution. Furthermore, a recent study [65] suggested that irregular bar widths may increase perceptual bias.

Our study does not aim to propose methods for determining bin sizes; rather, it seeks to improve bar chart design to enhance its ability to present data given a specified bin size effectively.

Improvement with visual encodings. Beyond bin size selection, prior work has explored a variety of visual encoding–based extensions to bar charts. For example, Karduni et al. [38] folded and wrapped excessively high bars to simultaneously represent both extremely high and low values within the same chart. Díaz et al. [17] suggested that internal visual cues within bars themselves can improve absolute value estimation. Besides, bar charts often struggle to convey additional attributes without extra coordinated visualizations. A common approach to address this is by integrating additional visualization into the bars to create a composite visualization [14, 36]. For example, Huang et al. [33] embedded treemaps within bars to visualize the associated hierarchical data. Heatmaps and bar charts can be combined to increase the amount of data that can be shown. In Pixel-Barcharts [40], each individual data value can be shown by a color-coded pixel. The technique has high scalability and user-defined binning. It is not optimized for temporal data. In particular, the placement of pixels within each bar adheres to a distinct x-y coordinate system, which may result in temporal patterns being misrepresented. To the best of our knowledge, no existing bar chart variants are capable of effectively representing both global and local patterns within event temporal distributions.

Improvement with user interactions. Complementary to visual encodings, interaction techniques have been explored to augment the analytical capability of bar charts. Highlighting and linking guide user attention to subsets of interest and reduce visual clutter during exploration [58]. Interactive adjustment of bin size [4] allows users to dynamically control temporal aggregation, supporting the exploration of patterns across different levels of granularity. Focus+context techniques enable multi-scale analysis by distributing detailed views and context views across multiple coordinated views. In contrast, fisheye-based techniques [26, 50], including lens-based interaction [67] achieve focus+context within a single view by magnifying regions of interest while compressing surrounding context. Some techniques, such as LineUp [28], allow users to switch the arrangement of bar charts (e.g., aligned by value or by attribute) to facilitate ranking, comparison, and pattern discovery. Multiple interaction techniques can also be integrated to support complex visual analysis scenarios. For example, Keim et al. [39] introduced interactive control mechanisms into Pixel-Barcharts [40] to enable hierarchical exploration and level-of-detail adjustment for dense distributions, such as, drill-down and drill-up, thresholding, and bounding-box selection. More recently, interaction paradigms have been extended toward conversational settings, where visual question answering (VQA) [32] enables users to query bar charts through natural language, further lowering the barrier for interactive analysis.

We conduct a formative study to examine whether adjusting the bin size alone could improve the bar chart's ability to visualize event temporal distribution in practice. Drawing on the studies' findings and inspired by composite visualizations, we refine the bar chart to effectively present both global patterns and fine-grained details within a limited space. Our design, named DensityBars, integrates heatmaps of event occurrences into the bar chart, offering a high degree of regularity. We also contribute to a deeper understanding of the value and limitations of DensityBars through two use cases and two user studies. In this work, we focus on improvements via visual encodings to enable richer information extraction without imposing additional interaction costs; interaction techniques remain a complementary direction for future work.

3 Formative Study

Both global and local patterns are essential for analyzing event temporal distributions. A common approach to reveal these patterns is to adjust the bin size of temporal charts [13, 52]. This strategy provides partial solutions but may lack comprehensive support for simultaneous exploration of both global and local patterns, particularly under the screen space constraints of visual interfaces. To systematically investigate these issues, we conducted a preliminary user study to evaluate bin size adjustment strategy and performed expert interviews to understand design practices in multi-view visual interfaces.

3.1 Preliminary User Study

Sahann et al.'s study [52] investigated the impact of bin size on the perception of well-defined distributions, such as normal or bimodal, given a sufficient sample size. In contrast, we focus on the common and practical scenario in which users have to use records sampled at an unknown sampling rate to perceive the distribution features, such as peaks, valleys, spikes, and gaps. Correll et al.'s study [13], a study the most similar to ours, suggested that using a smaller bin size enhances the visibility of local patterns like spikes and gaps. Building on these insights, we want to further test two hypotheses: **(H1) using a smaller bin size (i.e., more bins) reduces users' accuracy in perceiving global patterns and increases their response time** and **(H2) horizontally compressing the bar chart to fit limited space reduces users' accuracy in perceiving global and local patterns, and increases their response time.** We conduct a preliminary user study to test the hypotheses, which was approved by the relevant Ethics Review Board.

3.1.1 Synthetic Distribution. We used synthetic data for experiments. In this way, we can set up global and local temporal patterns that are common in real-world event time series as needed.

Rather than attempting to include every possible type of temporal pattern, we focus on a concise set of representative patterns that commonly appear in real-world datasets and have been widely discussed in prior work. In addition, we can remove confounding external factors by precisely controlling the strength and occurrences of patterns. Specifically, for global patterns, we consider **peaks** and **valleys** [52], and for local patterns, we consider **dense** and **sparse** occurrences [13, 15]. Besides their own meaningfulness, these patterns can constitute higher-level patterns. Peaks and valleys characterize long-term highs and lows in the temporal distribution; once such extrema are identified, it becomes easier to reason about rising and falling trends and periodicity. Dense occurrences correspond to short periods with unusually high event frequency, which instantiate temporal clusters, whereas sparse occurrences reflect the converse condition, revealing local anomalies in the event distribution.

To ensure that these patterns can be unambiguously identified and measured during the experiment, we introduced operational definitions that translate the intuitive notions of peaks, valleys, dense, and sparse into reproducible criteria. Specifically, for global patterns, a peak is defined as a long time period where the number of event occurrences is greater than that in both of the two preceding and two succeeding periods, with the event counts in neighboring periods progressively decreasing away from the peak. Conversely, a valley indicates that the number of event occurrences in this long time period is lower than that in the two preceding and two succeeding periods, with event counts in neighboring periods progressively increasing away from the valley. Two peaks usually result in a single valley between the peaks. For local patterns, a dense occurrence signifies a short period with a significantly higher event frequency compared to the neighboring periods, whereas a sparse occurrence refers to a short period with few or no events, surrounded by periods with relatively higher event occurrences.

In this study, we used synthetic yearly event distributions with hourly granularity as the experimental dataset. The hour was chosen as the basic unit of time because it carries a clear semantic meaning and is commonly used in real-world datasets, such as the air pollution and electricity consumption datasets discussed in Section 5. Each event distribution is represented as a list of timestamps $\{\ldots, t, \ldots\}$, where $0 \leq t < 8760$ (i.e., 365×24 hours), with each timestamp t in the list indicating an event occurs at t. While this is a reasonable and intuitive choice for our study, other granularities can be set depending on the scale of the data of concern.

For each synthetic distribution, we first sampled the total number of events at random from the range [1200, 1500]. This range keeps the distributions neither too sparse nor too dense for the experimental tasks, while allowing some natural variability in overall activity levels across conditions. We then instantiated the desired global and local patterns by assigning these events to hourly timestamps under the following constraints. For global patterns, the specific time span for peaks and valleys was set to one month, with their positions randomly assigned within specific months. We considered three scenarios: (1) no peaks (**0P**), (2) one peak (**1P**), and (3) two peaks with one valley (**2P/1V**). Particularly, the time interval between the two peaks must be greater than 4 months. These three configurations correspond to flat, single-peak, and two-peak yearly profiles that we commonly observe in real-world datasets, while keeping the number of global patterns small enough for a controlled study. For local patterns, dense and sparse occurrences were defined to span a quarter of a month. These occurrences were also randomly assigned within the dataset. We chose three scenarios: (1) no dense occurrences with two sparse occurrences (**0D+2S**), (2) one dense occurrence with one sparse occurrence (**1D+1S**), and (3) two dense occurrences with no sparse occurrences (**2D+0S**). Particularly, the time interval between two dense or sparse occurrences must be greater than 1 month. These three configurations capture local deviations that appear as gaps, bursts, or a mixture of both, while keeping the number of local patterns small enough for a controlled study. In sum, there are 3 (global patterns) × 3 (local patterns) = 9 distributions. These cover cases with or without global features and with or without local features, ensuring diversity while keeping the experimental scale manageable.

3.1.2 Charts. We selected three distinct bar chart configurations as follows:

C1: Coarse-grained bar chart (Fig. 2a2). In this chart, the event occurrences are aggregated by the bin size of one month, resulting in 12 bins. The chart occupies a screen space of 515px × 375px.

C2: Fine-grained bar chart (Fig. 2c). The second one has a smaller bin size than **C1**. Specifically, the event occurrences are aggregated by the bin size of a quarter of a month, resulting in 48 bins. It retains the same chart width as **C1**, which is 515px × 375px.

C3: Wider fine-grained bar chart (Fig. 2b). The last one has a bin size of a quarter of a month, smaller than **C1** and the same as **C2**, resulting in 48 bins. It has a larger chart width than **C1** and **C2**, occupying a screen space of 1050px × 375px.

Comparison between **C1** and **C3** is to figure out **H1** while comparison between **C2** and **C3** is to figure out **H2**.

3.1.3 Tasks and Questions. We design four tasks, **T1** to **T4**, in the form of questions in two aspects:

Global Patterns Identification. Subjects were tasked with identifying peaks and valleys in the event distribution from the chart. For instance, they compare the total number of events across months to determine which month is a peak or valley. Specifically, the two questions subjects should answer are **T1**: *"In which month does the peak occur?"* and **T2**: *"In which month does the valley occur?"*

Local Patterns Identification. Subjects were asked to identify the dense and sparse occurrences in the event distribution from the chart. For instance, they compare different weeks throughout the year to determine which weeks have obviously higher and lower event densities. The two specific questions are **T3**: *"During which periods are the observed counts dense?"* and **T4**: *"During which periods are the observed counts sparse?"*

These tasks are representative and serve as fundamental building blocks for more advanced event time series analysis. As discussed in Section 3.1.1, once peaks and valleys are identified, the underlying rising and falling trends can become clearer. Similarly, locating dense and sparse periods highlights short-term bursts and gaps in the event distribution, which underlie higher-level tasks such as detecting temporal clusters or local anomalies.

Each question was accompanied by six answer options. Four of these options corresponded to specific time periods, while the remaining two were "no" and "not sure." The "no" option indicated the absence of the specified time periods in the chart, while the

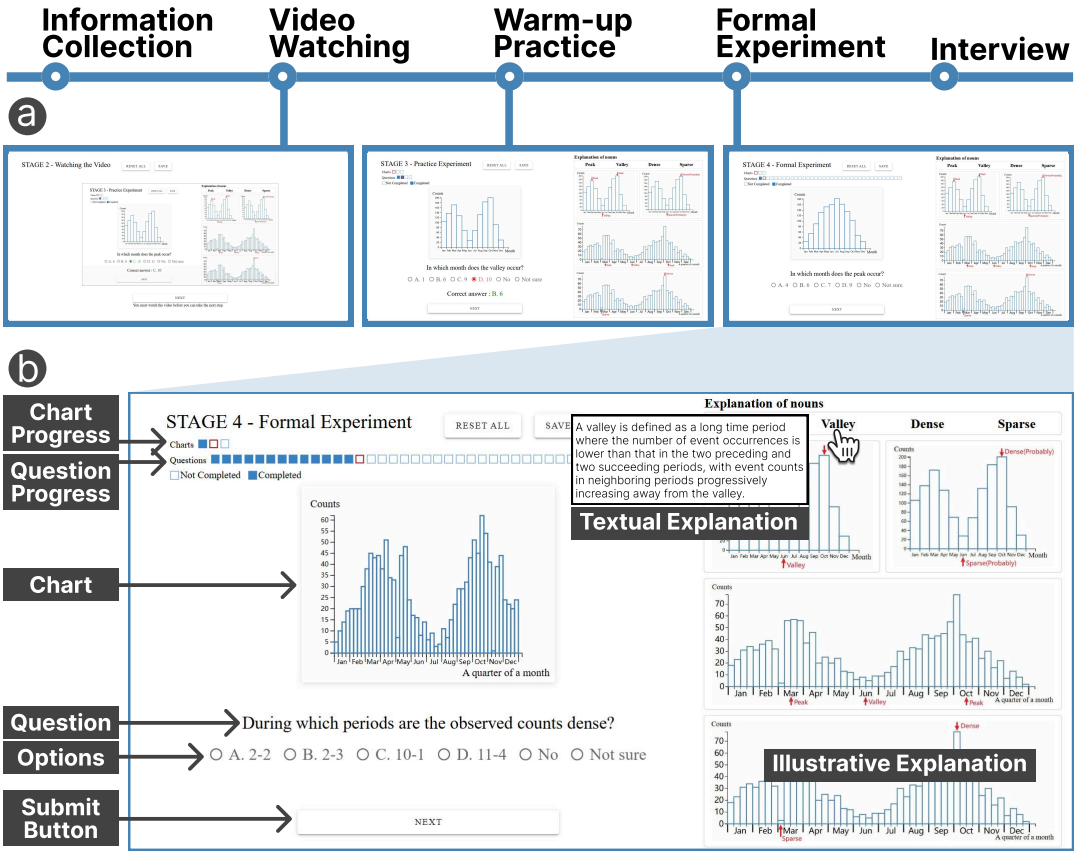


Figure 3: The procedure of preliminary user study. (a) The procedure consists of five stages, namely, information collection, video watching, warm-up practice, formal experiment, and interview. (b) The user interface snapshot of the formal experiment.

"not sure" option allowed subjects to express uncertainty, thereby preventing excessive time spent on a single question.

3.1.4 Participants. We published a participant recruitment notice within our college. A total of 16 participants (13 males and 3 females), aged between 18 and 24 years (mean age: 21.1 years), were recruited for this preliminary user study. The majority (93.75%) of participants majored in computer science or software engineering, and one (6.25%) specialized in logistics engineering. Familiarity with data visualization varied among the participants: 1 (6.25%) had no prior exposure, 7 (43.75%) were novices, 6 (37.5%) had basic knowledge, 1 (6.25%) was an experienced user, and 1 (6.25%) was highly proficient. Familiarity diversity ensures the participants' representativeness. Each participant was compensated 20 RMB for their participation.

3.1.5 Procedure. We develop an experiment system where the sizes of every chart are fixed. The system was deployed on the cloud, and participants accessed the experimental system via desktop or laptop computers using the provided link from anywhere as long as it was convenient for them. Participant variability in terms of device and environmental conditions introduced differences in screen brightness and resolution. Despite these inconsistencies, prior research has established the viability of such experiment environments [30]. The experiment was divided into five stages: basic information collection, instructional video viewing, warm-up practice, formal experiment, and interview (Fig. 3). Participants were required to complete each stage sequentially. On average, the entire experiment took 40 minutes to complete. These stages are explained as follows:

Stage 1. Firstly, participants provided informed consent and were fully aware of the nature, purpose, and potential risks of the research. They filled in background information, including name, gender, age, major, and familiarity with data visualization. **Stage 2.** Secondly, participants watched an instructional video that introduced the charts, explained key terminology, and described the experimental tasks. In the video, we asked users to adjust the browser zoom ratio to 100% to ensure that the effect of the bin size and chart width can be accurately assessed. Watching the video was mandatory before proceeding to the next step. **Stage 3.** Thirdly, participants were presented with 12 questions (3 charts × 4 questions) designed to familiarize them with the experimental content and procedures. Each question required selecting one of six possible options based on the information presented in the corresponding chart. Feedback was provided after each response, displaying the correct answer to reinforce participants' understanding. **Stage 4.** Participants completed three sessions, each focused on one chart and consisted of 36

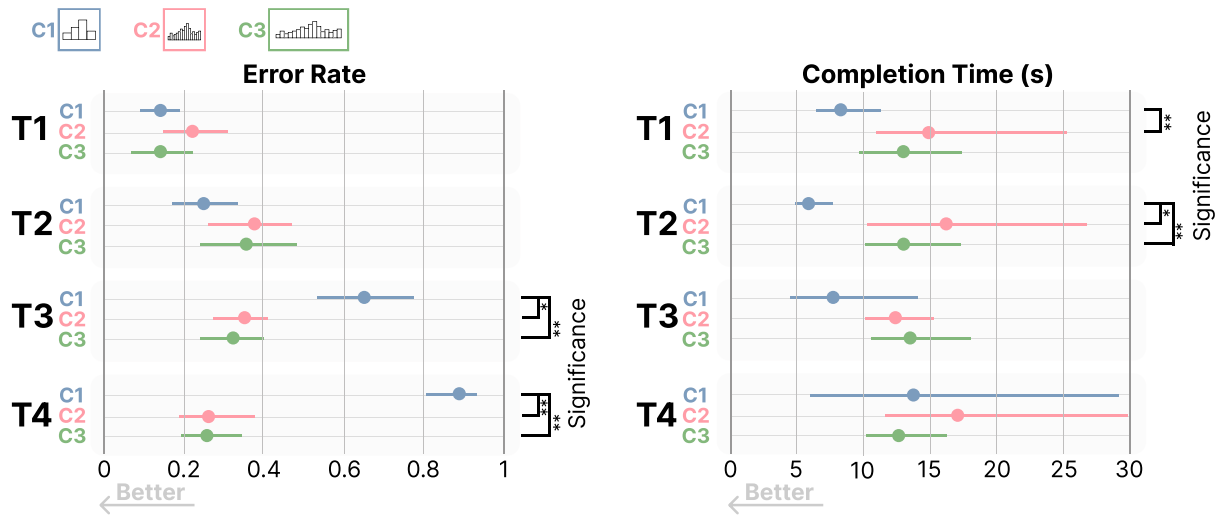


Figure 4: The means of error rate and completion time with their 95% confidence intervals (CIs) for three charts in the preliminary user study. Note that links on the right side indicate pair-wise significance. The significance ($p < 0.05$) is annotated by an asterisk*, and the strong significance ($p < 0.01$) is annotated by two asterisks.**

questions. Participants were instructed to complete each session as accurately and quickly as possible. Unlike in the practice stage, no feedback on correct answers was given during the formal trial. The order of the chart presentation (session) was randomized. Within each session, the order of the questions was also randomized. After completing each session, participants were given a 5-minute break to mitigate their fatigue. All answers were submitted at the end of the final session, concluding this stage. In addition, the response time for each question and the total duration of the experiment were recorded for further analysis. In sum, each participant was required to complete 9 distributions × 4 tasks × 3 charts = 108 trials.

Stage 5. Finally, we conducted a short online interview with each participant and collected feedback. To collect timely feedback, we ask participants to contact us via instant messaging immediately once they finish the stages above.

3.1.6 Results. **Metrics.** Following previous studies [1, 6, 7, 48, 76], we use two quantitative metrics for each task. The first one is the *error rate*, i.e., the proportion of incorrect answers made by participants relative to the total number of questions in the task. The second one is the *completion time*, i.e., the duration from when participants first viewed the chart to when they submitted their answer. To assess participants' performance in identifying correct responses, our analysis focused exclusively on completion time for correctly answered questions.

Statistics. We computed the means of error rate and completion time for each task across the three charts used in this preliminary user study, employing BCa bootstrapping with 10,000 iterations to estimate their associated 95% confidence intervals (CIs). These results are presented in Fig. 4. We also conducted a significance analysis to evaluate differences among the charts. Given the non-normal distribution of the data, the Friedman test was employed [63, 76]. We used Kendall's W as the effect size for Friedman tests and interpreted it using conventional thresholds: small ($0.1 \leq W < 0.3$), medium ($0.3 \leq W < 0.5$), and large ($W \geq 0.5$). The three charts have significant differences regarding the error rate in **T3** and **T4** with ($\chi^2 = 15.75$, $p < 0.01$, Kendall's $W = 0.49$, medium effect) and ($\chi^2 = 26.42$, $p < 0.01$, Kendall's $W = 0.83$, large effect), respectively. The three charts also have significant differences regarding the completion time in **T1** and **T2** with ($\chi^2 = 10.77$, $p < 0.01$, Kendall's $W = 0.34$, medium effect) and ($\chi^2 = 12.11$, $p < 0.01$, Kendall's $W = 0.38$, medium effect), respectively. For further pair-wise comparison, we employed the Nemenyi post-hoc test. The pair-wise significant differences were indicated by the links in Fig. 4.

Findings. We can obtain the following findings from Fig. 4:

First, coarse-grained binning with fewer bins (**C1**) is risky in revealing local patterns, while using more bins (**C2** and **C3**) can alleviate this issue, as also suggested by prior studies [13]. Second, with respect to our hypothesis **H1**, the error rate in **T1** Peak-Identification and **T2** Valley-Identification is slightly higher for **C2** and **C3** compared to **C1**, but the difference is not statistically significant, which partially contradicts our hypothesis **H1**. On the other hand, the completion time is significantly longer for **C2** and **C3** than for **C1**, which supports **H1** in terms of response time. Many participants provided an explanation: "We need to mentally aggregate four bins into one month first to perform peak and valley recognition." This highlights the additional cognitive load caused by finer binning. Third, regarding **H2**, chart width shows little impact on error rate or completion time, as **C2** and **C3** generally performed comparably. This result contradicts **H2**. Nevertheless, many participants preferred **C3**, citing its lower visual density, which they felt improved readability and facilitated interpretation of temporal distributions. Overall, to reveal both global and local patterns, fine-grained binning with sufficient chart width (**C3**) appears to be the most effective choice.

3.2 Expert Interviews

The preliminary user study highlights the limitations of bin size adjustment strategy. In particular, the most effective configuration (**C3**) required more screen space. In real-world interface design, space and layout constraints are often unavoidable, which makes this issue even more critical. To better understand how designers handle these constraints in practice, we conduct semi-structured interviews with three visual interface and dashboard designers. Each designer has more than three years of experience in visual interface design and has independently designed ten real-world multi-view dashboards or visual analytics interfaces each. The interviews are conducted individually, each lasting about half an hour. We mainly asked each designer three questions: 1) Do you usually follow a specific layout when designing a multi-view visual interface? 2) When a view becomes visually dense, do you allocate additional screen space to improve readability? 3) When one view requires additional space to present its information, how do you adjust the layout of the other views? Based on the responses, we would ask for details or ask about their personal experiences. The findings of the interviews are summarized as follows:

Layout Conventions. For the first question, the interviews reveal two major conventions in designing visual interface layouts. The first concerns the spatial arrangement of views. As one designer noted, "*Control panels are often placed in the upper menu bar or the leftmost sidebar, while auxiliary views that provide detailed information are often placed in the rightmost sidebar.*" This convention is called configuration patterns in multiple-view visualizations in the visualization community [10]. The second convention concerns alignment between views. Each view in a multi-view visual interface should be aligned by any of its four borders with other views to make the interface visually neat and aesthetic. For example, in Fig. 2a, the top and bottom edges of Fig. 2a1 align with those of Fig. 2a2 and Fig. 2a3, while the left and right edges of Fig. 2a2 and Fig. 2a3 are also aligned. One designer emphasized, "*I have never seen an interface design anywhere where the views are not aligned.*"

Visual Clarity. For the second question, all designers agreed that they usually allocate additional screen space to improve readability when a view becomes visually dense. One designer explained, "*If a view contains too many data points or dense visual elements, I usually increase its width or height so that users can read the information comfortably.*" When asked to compare Fig. 2b and Fig. 2c, all designers judged Fig. 2b to be visually clearer. However, they also acknowledged that under strict spatial constraints, they might be forced to adopt a more compact layout similar to Fig. 2c. These responses illustrate that while designers value clarity, spatial limitations often dictate compromises in multi-view layouts.

Space Reallocation. For the third question, designers reported that when one view requires additional space, the surrounding views must be adjusted to maintain overall alignment and visual neatness. We asked them to provide a scenario using Fig. 2a. Two designers said that, "*In this layout, if the width of a2 increases to display more information, the width of a3 also needs to increase*". One of the designers further explained, "*If a2 and a3 become too wide, compressing the horizontal space of a1, I may need to consider placing a2 and a3 side by side and left to right at the bottom.*" These comments highlight the complex dependencies and constraints that designers face when reallocating space in multi-view layouts.

Taken together, the interviews reveal that the limitations of bin size adjustment strategy become more pronounced in real-world interface design. Space constraints and alignment conventions often force designers to compromise, suggesting that alternative or complementary approaches may be necessary.

4 Design

To overcome the limitations found in the preliminary studies, we introduce a novel visualization technique, DensityBars. This design integrates heatmaps within individual bars to encode fine-grained temporal details while the bar height still encodes aggregated counts. In this way, DensityBars achieves dual-level encoding: users can quickly grasp global patterns from bar heights and local patterns from the inner heatmap texture, all within the same spatial footprint. In the following, we first describe an alternative design that we explored, and then present how we arrived at our final composite design, DensityBars.

Alternative. We first considered a heatmap-based design, as shown in Fig 5a, which is widely used for visualizing temporal distribution. In such a matrix-like design, each column denotes a bin size as the bar chart in Fig. 5b, and the column is divided into multiple cells, each of which denotes a smaller time period than the bin size. The darkness of the cell encodes the number of event occurrences during the time period. This representation works well for examining fine-grained temporal fluctuations when the analytical window is at or near the cell granularity. Local patterns become immediately visible through intensity variations, for instance, the second week of March in Fig. 5a stands out as denser than adjacent weeks. However, the design can be less suitable for tasks involving the analysis of much longer time spans. In such tasks, users often need to mentally integrate numerous color-encoded values, a process that is perceptually non-linear and cognitively demanding. For example, in Fig. 5a, it is difficult to determine whether March or April (or October or November) represents the true peak month, depending on how users mentally aggregate the cell intensities.

Final Design. Our goal is to provide a visualization that supports both global and local analysis within a compact visual space. This requirement motivated the design of DensityBars (Fig. 5c), which embeds a continuous density-based texture into bar charts so that bar height conveys aggregated event counts while the inner texture reveals within-bin temporal variations. The composition procedure is as follows.

Step 1: Tick Generation. Given event occurrences $\{\ldots, t, \ldots\}$, we map them onto a timeline, generating event ticks (Fig. 5d). Event ticks offer a clear depiction of the temporal distribution, akin to the ticker-tape timer in physics experiments, where the tick density conveys the event frequency. *Step 2: Heatmap Generation.* Instead of the discrete heatmap in Fig. 5a, we obtain the continuous tick density by applying kernel density estimation (KDE) to the horizontal ticks in the whole timeline, using the optimal bandwidth computed by the method of Shimazaki et al. [57]. Continuous is more suitable than discrete for the subsequent composition process. We normalize the density values to the range [0, 1]. These normalized densities are subsequently mapped to a color gradient,

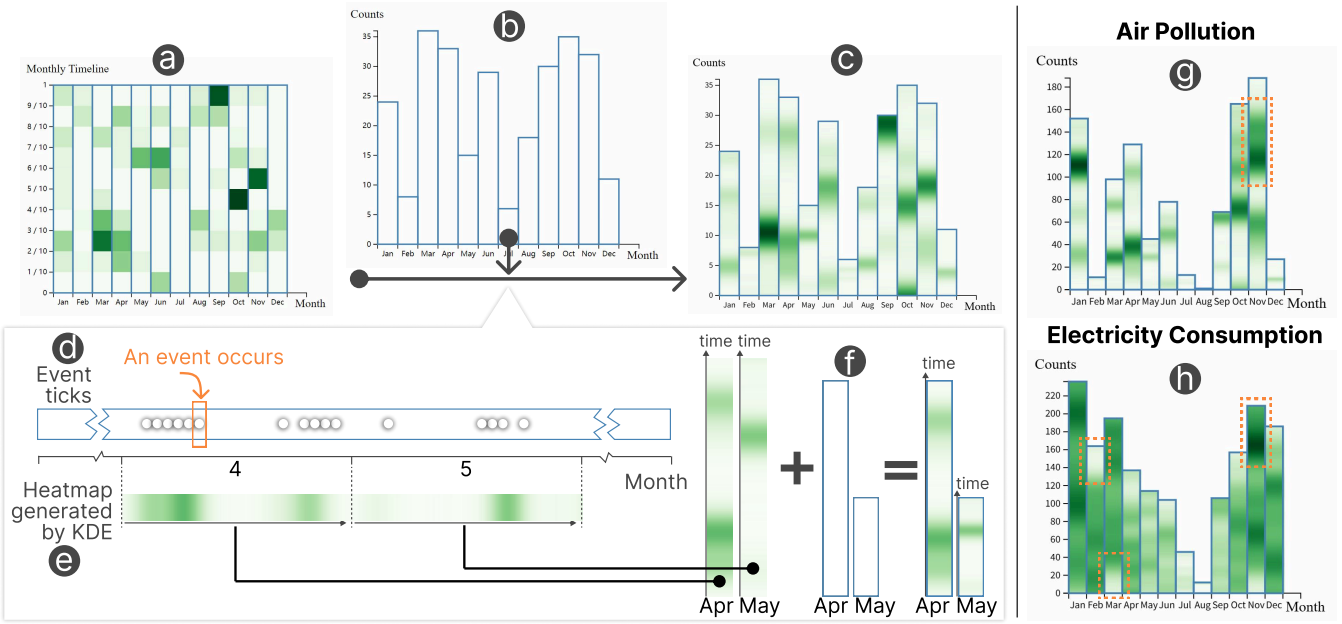


Figure 5: Design process (a-f) and two use cases (g-h) of DensityBars. (a) A heatmap can provide a more detailed temporal distribution of event occurrences than (b) the traditional bar chart. (c) Our design, named DensityBars, is proposed by integrating (a) the heatmap and (b) the traditional bar chart. (d) The event ticks are first generated. (e) A heatmap is generated by KDE based on the event ticks and is sliced into multiple sub-heatmaps according to the bin size. (f) DensityBars is created by scaling the sub-heatmaps and embedding them into bars. (g) Air pollution analysis and (h) electricity consumption analysis supported by DensityBars.

Algorithm 1 Construction of DensityBars

Input: Event timestamps $T = \{t_1, \ldots, t_N\}$; bar intervals $\{I_1, \ldots, I_M\}$
Output: Bars whose heights encode event counts and whose internal textures encode local densities
1. $\rho(t) \leftarrow \mathrm{KDE}(T)$ ▷ *Continuous global tick density*
2. $H(t) \leftarrow \textsc{NormalizeToColor}(\rho(t))$ ▷ *Normalize density and map to 1D heatmap*
3. for $m = 1$ **to** M **do**
4. $c_m \leftarrow |\{t_i \in T \mid t_i \in I_m\}|$ ▷ *Event count for bar height*
5. $H_m \leftarrow H(t)|_{t \in I_m}$ ▷ *Slice local density pattern*
6. $H_m \leftarrow \textsc{Scale}(H_m, c_m)$ ▷ *Adapt to bar height*
7. $\textsc{Embed}(m, H_m)$ ▷ *Embed density texture H_m into bar m*
8. end for

with darker shades indicating higher event frequencies and lighter shades indicating lower frequencies, generating the heatmap as Fig. 5e. *Step 3: Composition.* The heatmap is then sliced according to the bin size into multiple sub-heatmaps. Each part corresponds to a bar. Afterward, these sub-heatmaps are scaled according to the number of event occurrences. Fig. 5f illustrates how the heatmaps of April and May are scaled and embedded into the bars accordingly. To make this design easier to reproduce, we provide its detailed implementation in pseudocode form in Algorithm 1.

5 Use Cases

We present two use cases with DensityBars where real-world event datasets are analyzed from both the global and local perspectives, revealing the temporal patterns of corresponding events. These cases demonstrate the effectiveness and usability of DensityBars.

5.1 Air Pollution Analysis

Dataset. Air pollution can cause chronic damage to people's health, and the patterns and causes behind it have received widespread attention. We have an air quality index (AQI) dataset, which records the hourly air quality of Beijing in 2022. An air pollution event occurs when the AQI reading is higher than 100, according to China's policy. DensityBars of this dataset is shown in Fig. 5g. Specifically, the bin size is one month, i.e., the event occurrences are aggregated by months.

Global Patterns. Both the traditional bar chart and DensityBars can effectively capture the monthly trends in air pollution events over the course of the year. The visualizations show that November exhibited the highest frequency of air pollution events, indicating peak severity, while August had the lowest, marking the least pollution. This pattern may be attributed to the increased rainfall during the summer months, which helps to remove airborne pollutants, thereby improving air quality. We can also see that there were fewer pollution events in February compared to January and March. The first half of February corresponds to the Chinese New Year holiday, spanning from the first to the fifteenth day of the Lunar New Year.

During this period, reduced urban traffic and the suspension of industrial activities likely led to a significant decrease in emissions and overall pollution. In addition, an upward trend in pollution events is observed during the second half of 2022, culminating in a decline in December.

Local Patterns. Unlike traditional bar charts, our design presents detailed information at a finer granularity without increasing screen space. An obvious visual pattern is that every heatmap in each bar is not uniform. Thus, air pollution events do not occur consistently within a given month. We also found an interesting observation: the third week of November (enclosed by the orange dashed rectangle) had the highest frequency of air pollution events, as shown by the darkest color, surpassing all other periods. The reason might be as follows. In 2022, heating in Beijing commenced on November 15. The widespread use of coal for heating across northern China likely contributed to the increased frequency of air pollution events observed during this period.

5.2 Electricity Consumption Analysis

Dataset. Electricity is a fundamental cornerstone of human civilization, essential to nearly every aspect of modern society. We have a dataset that records the hourly occurrence of electricity consumption exceeding 1.8 kWh of a household in Paris in 2008. We denote these occurrences as high consumption events. We also set the bin size as one month. Fig. 5h is DensityBars for these occurrences.

Global Pattern. Overall, the monthly frequency of high consumption events generally shows a decreasing trend followed by an increase throughout the year, with a lowest point in August. That is to say, electricity usage is higher in winter and lower in summer. This pattern can be attributed to the mild Parisian summer, where most households relied on electric fans for cooling. In contrast, the cold winter led to widespread use of electric heating, resulting in significantly higher electricity consumption.

Local Pattern. The density heatmap further displays that high consumption events occurred most frequently in the last week of November (enclosed by the orange dashed rectangle). The daily low in Paris for the winter of 2008 fell below 0°C for the first time this week, which might explain the frequency of events. We also noticed an empty area in late February and early March (enclosed by the orange dashed rectangles), indicating fewer high-consumption events during this period. During this period, the temperature in Paris rose. The daily minimum temperature no longer fell below 0°C as it did in the rest of February. Moreover, on February 24, the maximum temperature exceeded 15°C (for the first time since 2008), reaching 17°C.

6 User Studies

Based on the design process and exploration of use cases, we derive the following three hypotheses: **(H3) compared with two representative baselines (i.e., bar charts and discrete heatmaps), our design enhances users' accuracy and response time in perceiving both global and local patterns**, **(H4) our design performs comparably to the combined chart (i.e., a bar chart and a concrete heatmap placed side by side) in terms of users' accuracy and response time for perceiving both global and local patterns**, and a supplementary hypothesis **(H5) its visual composition that scales the heatmap timeline differently within each bar does not affect users' accuracy and response time in perceiving local patterns.** To validate these hypotheses, we design and conduct two user studies, further exploring DensityBars's performance.

The user studies are designed similarly to the preliminary study. In particular, we adopt the same *synthetic distributions* to generate the experiment dataset and the same *tasks and questions*. The charts to be compared and the participants are different from the preliminary study. Both user studies were also approved by the relevant Ethics Review Board.

6.1 User Study #1: Comparing DensityBars with Alternative Single Charts

The first study compares DensityBars with the other two representative single charts, with the goal of understanding whether it can present more information about global and local patterns in a single visualization (related to **H3**).

6.1.1 Charts. We select three charts for comparison:

C3: Wider fine-grained bar chart (Fig. 2b). The first one is the most effective chart in the preliminary study. Its bin size is a quarter of a month, resulting in 48 bins. The chart occupies a screen space of 1050px × 375px, the same as in the preliminary study.

C4: Discrete heatmap (Fig. 5a). The second one is the alternative design. Compared to **C3**, it has a larger bin size, a month, also resulting in 12 bins in total. It has a smaller chart width than **C3**, occupying a screen space of 515px × 375px. For each column in the heatmap, the timespan of a month is divided into 10 cells, each of which denotes approximately three days. In this setting, two to three cells (i.e., 6–9 days) correspond to the temporal scale of local dense or sparse episodes considered in our tasks. Using one- or two-day cells would result in 15–30 tiny cells per month, leading to visual clutter, whereas using five- or six-day cells would make the temporal granularity too coarse to localize whether an episode occurs in the early, middle, or late part of a month.

C5: DensityBars (Fig. 5c). The last one is our proposed design, with a larger bin size than **C3** and the same bin size as **C4**, also resulting in 12 bins in total. Its chart width is the same as that in **C4** and smaller than that in**C3**, occupying 515px × 375px.

This user study consists of 9 (synthetic distributions) × 3 (charts) × 4 (tasks or questions) = 108 trials.

6.1.2 Participants. Participants for this study were recruited using the same method as in the preliminary study. We recruited 36 participants (30 males and 6 females). They did not participate in the preliminary study. They are aged between 19 and 26 years (mean age: 20.7 years). 19 participants majored in computer science or software engineering, while the remaining participants had diverse majors spread across fields such as medicine, accounting, design, chemistry, and agriculture. Familiarity with data visualization varied among the participants: 14 (38.89%) had no prior exposure, 11 (30.56%) were novices, 7 (19.44%) had basic knowledge, 3 (8.33%) was an experienced user, and 1 (2.78%) was highly proficient. Such diversity ensures the participants' representativeness. Each participant was compensated 20 RMB for their participation.

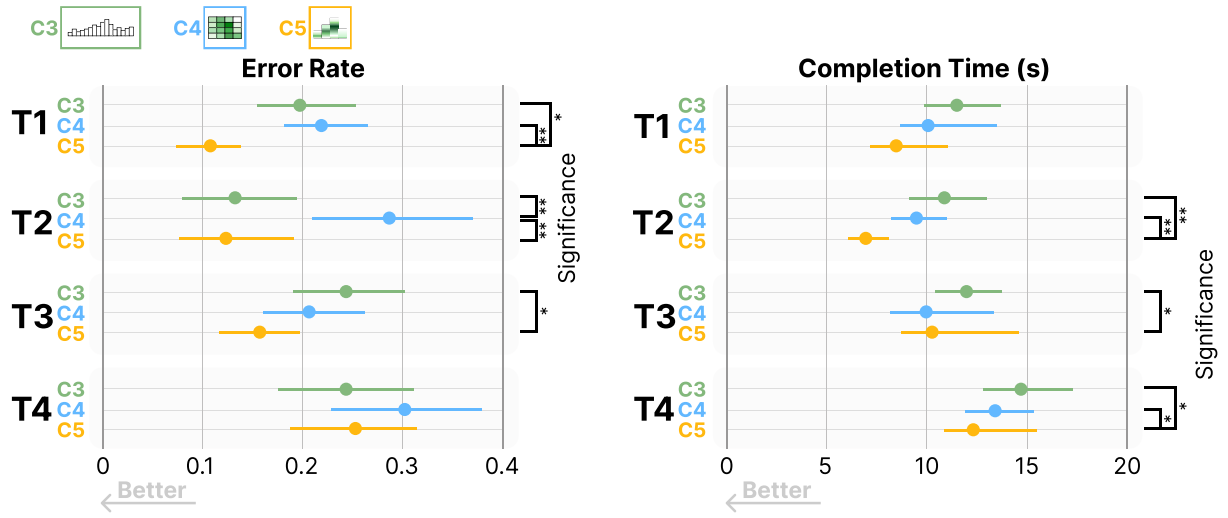


Figure 6: The means of error rate and completion time with their 95% confidence intervals (CIs) for three charts in User Study #1. Note that links on the right side indicate pair-wise significance. The significance ($p < 0.05$) is annotated by an asterisk*, and the strong significance ($p < 0.01$) is annotated by two asterisks.**

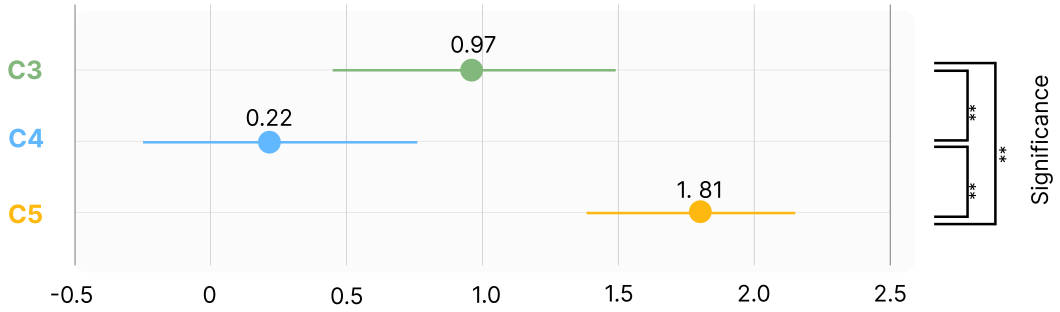


Figure 7: The average ranking scores with standard deviations. The links on the right side indicate the pair-wise significance. Two asterisks annotate the strong significance ($p < 0.01$).**

6.1.3 Procedures. The procedures were largely consistent with those in the preliminary study, except for modifications introduced in **Stage 5** (interview). In this stage, participants were asked to rank the three charts based on their experimental experience and to articulate their rationale. These modifications aimed to obtain more detailed insights into participant perspectives, facilitating a more comprehensive evaluation of DensityBars's effectiveness. On average, the entire experiment took 35 minutes to complete.

6.1.4 Results. **Metric.** We use the error rate and completion time as metrics, the same as in the preliminary study. We also quantify the participants' ranking of the three charts as an additional metric.

Statistics. We calculated the means of error rate and completion time for each task across the three charts, and estimated the corresponding 95% confidence intervals (CIs), following the same methodology as in the preliminary study. The results are shown in Fig. 6. We also performed a significance analysis to evaluate differences among the charts using the Friedman test followed by a Nemenyi post-hoc test. The results revealed significant differences among the three charts regarding the error rate, in **T1**, **T2** and **T3**, with ($\chi^2 = 17.26$, $p < 0.01$, Kendall's $W = 0.24$, small effect), ($\chi^2 = 27.78$, $p < 0.01$, Kendall's $W = 0.39$, medium effect), and ($\chi^2 = 10.67$, $p < 0.01$, Kendall's $W = 0.15$, small effect), respectively. Significant differences regarding the completion time were observed in **T2**, **T3** and **T4**, with ($\chi^2 = 16.72$, $p < 0.01$, Kendall's $W = 0.23$, small effect), ($\chi^2 = 8.39$, $p < 0.05$, Kendall's $W = 0.12$, small effect) and ($\chi^2 = 9.39$, $p < 0.01$, Kendall's $W = 0.13$, small effect). The pair-wise significant differences were represented by the links in Fig. 6.

Finally, we assigned each chart a series of scores ranging from 0 to 2 based on the participants' rankings of the charts, with higher scores indicating better rankings. Specifically, the top-ranked chart

received 2 points, the middle-ranked chart 1 point, and the lowest-ranked chart 0 points, inspired by Tian et al. [66]. The scores for each chart are illustrated in Fig. 7. The Friedman test indicated a significant difference in scores across the charts (χ^2 = 45.17, p < 0.01, Kendall's W = 0.63, large effect). Post-hoc analysis using the Nemenyi test revealed the pair-wise significance, as shown by the links on the right side of Fig. 7.

Findings. The performance of **C5** is better than that of **C3** and **C4** in terms of the error rate for **T1** Peak-Identification and **T3** Dense-Identification, as well as completion time for **T2** Valley-Identification and **T4** Sparse-Identification. Some of these advantages are statistically significant. These results partially support our hypothesis **H3** and also indicate that **C5** performs comparably or better than **C4** in **T3** and **T4**, thereby supporting **H5**. The rankings of participants also underscore their distinct preference for **C5**, with a significant difference from the other two. **C5**'s benefits can be explained in detail as follows.

First, compared to **C3**, **C5** uses a larger bin size, which makes differences in monthly event counts more salient. Several participants noted, "*With **C3**, we have to merge multiple small bars in our heads to see which month is higher, whereas with **C5** we can just look at which bars are taller.*" This aligns with the feedback from the preliminary user study and explains why **C5** supports global patterns identification at least as well as **C3**. Moreover, although **C5** uses a larger bin size, local patterns remain clearly visible thanks to the embedded density heatmap. **C5** even significantly reduces the completion time compared to **C3** for identifying local patterns. Many participants pointed out the reason, "*We have to browse many bars in **C3**, which often causes visual fatigue.*" In contrast, the local patterns are encoded by the sparse and dense areas in the heatmap, which can be identified by participants at a glance.

Second, compared to **C4**'s color-based encoding of fine-grained event counts, **C5** maintains the use of bar height for monthly totals, providing a clearer representation of global patterns. As many participants put it, "*With **C4**, we often need to mentally integrate the color intensities across an entire column of small squares to estimate which month is higher, and when two columns look similar, we hesitate and spend more time.*" This observation is consistent with our analysis of the heatmap-based alternative and helps explain why **C5** outperforms **C4** in peak and valley identification. For local patterns, **C5** performs similarly to, and in some cases slightly better than, **C4**. However, three participants noted difficulty in identifying specific timestamps or time intervals within the heatmaps embedded in bars of varying heights. For example, one participant reported, "*I am often unsure which quarter of a month an identified dense area is located.*" In contrast, **C4** performed better in this regard, as each cell clearly represents a uniform time interval. This feedback indicates a tension: although the quantitative results support **H5**, suggesting that DensityBars does not compromise task accuracy or efficiency, the subjective experience of some users highlights a potential drawback in precise temporal localization.

In short, the quantitative results show that DensityBars performs better than the other two baselines for the global- and local-pattern tasks we studied. Participants' subjective feedback is consistent with this trend. Many described **C5** as "easier" or "more straightforward." One participant explained: "*The coarse bars make it easy to see which months stand out, and the heatmap inside each bar makes it easy to see where events are dense or sparse.*" This comment illustrates that, when using DensityBars, users naturally rely on bar height for global patterns and on the inner color for local details, treating them as two simple and separable channels rather than a single overloaded encoding. Another participant remarked, "***C5** naturally combines a bar chart and a heatmap without taking extra screen space; I find this kind of chart very interesting.*" Together, these reports suggest that the dual-level encoding in DensityBars feels natural and space-efficient to participants, rather than adding noticeable mental overhead compared to the baseline charts.

6.2 User Study #2: Comparing DensityBars with Combination of Heatmap and Bar Chart

Given the space constraints imposed by an interface layout, DensityBars could be the best option for visualizing the temporal distribution of events. Although the first user study suggests DensityBars's advantage in showing global and local patterns, one remaining question is whether such a design that nests the two charts loses any effectiveness compared to simply placing them side by side (related to **H4**). Thus, we conduct the second user study.

6.2.1 Study Design. In this user study, we consider the side-by-side placement of **C1** and **C4** (Fig. 2d) as a whole, denoted by **C1 + C4**, and compare it with DensityBars, **C5**. 12 participants (8 males and 4 females) were recruited. None of them had participated in the preliminary study and User Study #1. They are between 20 and 32 years old (mean age: 26.8 years). The majority (83.33%) of them majored in software engineering, and two specialized in management. Three (25.00%) of them had basic knowledge, five (41.67%) were novices, and four (33.33%) had no prior exposure. The procedure was almost the same as the preliminary user study, particularly with informed consent first. In Stage 4, each participant only needs to complete 9 distributions×4 tasks×2 charts = 72 trials. The whole procedure for each participant took about 28 minutes. Each participant was compensated 20 RMB for their participation.

6.2.2 Results. **Metric.** We again use the error rate and completion time as metrics.

Statistics. As before, we calculated the means of error rate and completion time for each task across the two charts and estimated the corresponding 95% confidence intervals (CIs). The results are shown in Fig. 8. As there are only two groups, we used the Wilcoxon test to detect significant differences. No significant differences were detected. In addition, we computed Rosenthal's r as effect sizes for each Wilcoxon test. For the error rate, many participants made zero or very few errors in both conditions, leading to highly discrete data with numerous ties, so these effect sizes are not very informative. For completion time, the effect sizes all fall in the small-to-medium range, with $|r|$ between 0.21 and 0.48.

Findings. Overall, **C5** and **C1 + C4** achieve comparable error rate and completion time across all tasks. Error rate was already very low for both designs, and no significant differences were detected, suggesting that neither visualization incurs a noticeable accuracy penalty. For completion time, the mean values in Fig. 8 consistently favor **C5**, and the corresponding effect sizes are small to medium. However, the confidence intervals overlap substantially, and the Wilcoxon tests do not show significant differences. Overall,

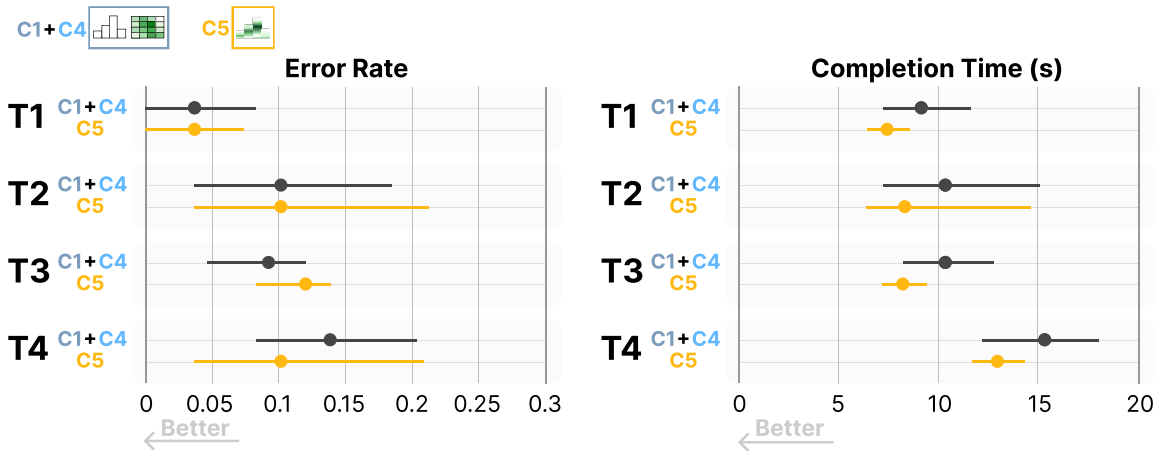


Figure 8: The means of error rate and completion time with their 95% confidence intervals (CIs) for two charts in User Study #2.

both visualizations appear to yield comparable completion times, with insufficient evidence for a meaningful difference to reject **H4**. In sum, DensityBars (**C5**), the space-efficient integration of a bar chart and a density heatmap, is as effective as the straightforward side-by-side placement of the two charts (**C1 + C4**). Focusing on **T3** Dense-Identification and **T4** Sparse-Identification, the error rate and completion time of **C5** and **C1 + C4** are likewise similar. This suggests that the visual transformation with inconsistent height of each bar in DensityBars has a negligible impact, which further supports **H5**.

Participants' feedback helps explain these comparable performances. First, as two participants remarked, "*Although these are two charts in **C1 + C4**, the amount of information provided is the same as in **C5**.*" In other words, both conditions visualize the same underlying data at comparable temporal granularity. Second, most participants reported very similar strategies across the two conditions. In both **C1 + C4** and **C5**, they relied on coarse-grained bars to identify peaks or valleys and on fine-grained heatmaps to locate dense and sparse periods. These comments suggest that DensityBars preserves the familiar division of labor between bars and heatmaps for global and local tasks, and therefore does not introduce noticeable additional cognitive burden compared to **C1 + C4**. Finally, for **C5**, participants generally considered the impact of scaling the heatmap timeline differently within each bar to be minor. Several explained, "*We typically first scan for relatively darker or lighter stretches within a bar to spot dense or sparse periods, and only then check the corresponding time position if needed.*"

7 Discussion

This section discusses the implications of our study, the applicability, and versatility of our design, and future work.

7.1 Implications for Visual Design

Our study offers several insights into effective visual design.

Task-oriented bin size selection. Selecting an appropriate bin size is a critical factor in distribution visualization across various chart types. Correll et al.'s study [13] suggested that using a smaller bin size will make local patterns, like spikes and gaps, clearer. As a supplement, our study, Preliminary User Study, demonstrates that excessively small bin sizes can obscure global patterns. This suggests that the selection of bin size should be tailored to the specific analytical task at hand. Larger bin sizes are more effective when the primary objective is to analyze global patterns, such as long-term trends. Conversely, smaller bin sizes are preferable when the focus is on detecting local features, such as anomalies. If both levels of analysis are crucial, DensityBars offers a balanced solution, accommodating both global patterns and fine-grained details.

Support for joint global and local analysis. Analyzing global and local temporal patterns are two common analytic tasks. To support both within a single view, DensityBars uses bar height to encode global patterns and embedded heatmaps to encode local patterns. Results from User Studies #1 and #2 show that DensityBars achieves comparable or better performance than the baselines on these individual tasks, suggesting that the combined encoding does not degrade single-task performance. In practice, when tasks involve the joint analysis of both global and local patterns, users can still interpret them in a stepwise manner, and DensityBars provides effective support for each step without introducing extra burden.

Color channel V.S. position channel. In User Study #1, charts using the position channel for encoding numerical values (e.g., bar height) yielded lower error rates than the discrete heatmap that relies solely on color intensity. This observation is consistent with prior perceptual findings that position generally affords higher precision than color for quantitative judgments [71]. One possible extension of position-based encoding is to embed an additional bar chart within each primary bar. However, the available horizontal space is extremely limited, and prior studies have shown that short bars [62] or bars with extreme aspect ratios [9] can impair accurate perception of length. Therefore, embedding bars within such constrained regions would likely introduce perceptual errors rather than improve performance. Overall, these findings indicate

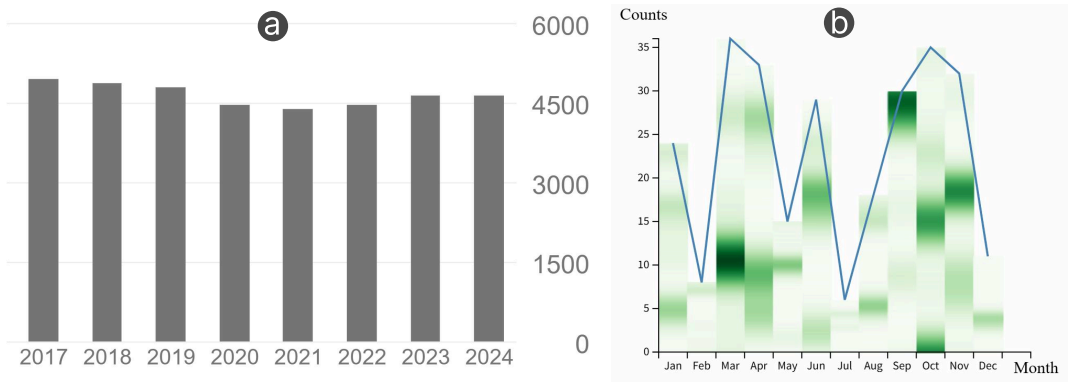


Figure 9: Two illustrative examples used for discussion. (a) Annual citation counts for Ben Shneiderman, obtained from his Google Scholar profile, without finer-grained temporal variation. (b) A synthetic time series with finer-grained event counts, visualized using a line chart augmented with scaled density heatmaps, with remaining visual design issues.

that the observed differences reflect an interaction between encoding choices and the demands of the tasks, rather than being solely caused by the intrinsic limitations of color.

Visual interface design. In the design of multi-view visualization interfaces (e.g., dashboards [60] and visual analytics systems [16]), the width and height of each view are often adjusted to align with the overall layout, ensuring a visually cohesive interface. For instance, in a left-right layout, the horizontal space available for side views and chart components is typically constrained. We found that the width of the bar chart had little effect on its validity in visualizing the temporal distribution of events. Therefore, if necessary, designers can reduce the width of the chart appropriately. In addition, DensityBars eases the layout constraints by providing a solution that effectively presents detailed information without requiring large horizontal space, thereby enhancing the flexibility of multi-view visualization interfaces.

7.2 Applicability and Versatility

DensityBars offers a highly versatile approach to event visualization, applicable across a wide range of domains. We have demonstrated its utility in two distinct fields, namely environmental science and energy, where it uncovers patterns that traditional bar charts fail to reveal. The potential applications of this technique extend well beyond these examples. For instance, it could be employed on Google Scholar to expose more granular citation trends. An interesting question to explore is whether citation counts for scholars in human-computer interaction (e.g., Ben Shneiderman's citations in Fig. 9a) exhibit noticeable increases around the annual ACM CHI Conference on Human Factors in Computing Systems each April. Furthermore, as an extension of basic charts, DensityBars benefits from being linked and coordinated with other visualizations. We believe it has broad applicability in various visualization interfaces, including dashboards and visual analytics systems. Its space-efficient design offers greater flexibility for integration, making it an attractive option for interface designers.

7.3 Future Work

Our study can be further improved in the following areas.

Plug-and-play package. While we have successfully designed DensityBars, the current implementation requires users to preprocess their datasets and manually configure parameters such as bin size. This process can be cumbersome, particularly for users with limited experience in data processing, creating a barrier to effective utilization. To address this, future work should focus on publishing a plug-and-play package that allows users to seamlessly import their data and generate charts with minimal configuration. Such a package would facilitate quicker adoption and broaden the applicability of our method across various research contexts.

Enhancing temporal precision. DensityBars effectively integrates global and local patterns while optimizing screen space, yet its design introduces a unique challenge: varying bar heights result in inconsistent heatmap scaling. Despite this, users can still readily perceive the overall temporal distribution. For example, in Fig. 5g, although September and October differ substantially in bar height, it is easy to see that September events are concentrated at the end of the month (the last few days), while October events are concentrated in the middle. At the same time, mapping the vertical position (e.g., middle) of a bar to the exact temporal interval (e.g., mid-month) requires extra cognitive effort. Future work could explore lightweight interaction to enhance temporal precision, such as hover-based tooltips that reveal exact dates and values, or smooth animated transitions between DensityBars and a standard heatmap that let users switch seamlessly between overview and precise inspection without increasing screen space usage.

Cognitive load measurement. Our discussion of cognitive load is based on indirect evidence from task performance and participants' feedback, rather than on dedicated cognitive-load measures. We did not collect subjective workload ratings (e.g., NASA-TLX) or run studies that directly measure users' mental effort. Consequently, our findings can only be interpreted as indirect evidence that the dual-level encoding in DensityBars is workable in our study settings; they do not quantify how much cognitive load it imposes on users. A promising direction for future work is to complement performance-based evaluation with subjective workload questionnaires and other objective indicators of cognitive effort.

Broader analytic tasks. Our studies focus on four fundamental perceptual tasks: identifying peaks, valleys, dense periods, and sparse periods. This design allows us to attribute differences in accuracy and efficiency primarily to the visual encodings. However, real-world analysis often involves higher-level tasks such as assessing long-term trends and seasonality, characterizing short-term irregularities, comparing multiple groups, and relating patterns to external factors. Evaluating DensityBars in such realistic visual analytics settings over longer-term use remains an important direction for future work, complementing our current perceptual findings.

Extending to more complex data. DensityBars currently visualizes the temporal distribution of a single event type. Temporal data can become more complex when events are grouped by category, such as type or region. One potential extension is a stacked variant, where each bar is divided into segments for different categories, and each segment embeds its own local density heatmap. This could preserve per-bin totals and reveal within-category patterns, though it may also increase visual and cognitive load. Another simple extension is to add interactive highlighting of category structure. For example, hovering over a category (e.g., in a legend) could switch the embedded heatmaps within DensityBars to show that category's density. Exploring such multi-category extensions of DensityBars and evaluating them on richer temporal datasets is a promising direction for future work.

Adaptation of line charts. Line charts effectively convey temporal trends by connecting data points to illustrate continuity over time. We explored embedding heatmaps into line charts but found the visual results to be suboptimal (e.g., Fig. 9b). Unlike bar charts, where distinct bins are clearly delineated by bars, line charts lack explicit boundaries, making it challenging to distinguish between individual heatmaps. Moreover, the bar chart's upper edge is flat and segmented, while the line chart follows a zigzagging path, and thus, heatmaps are difficult to embed properly. These characteristics, including the absence of segmentation and the undulating path, reduce the clarity and effectiveness of the line chart when combined with density heatmaps. In the future, we will explore novel heatmap generation algorithms or layout methods to better leverage the vertical space in line charts for visualizing local temporal patterns.

8 Conclusion

This study explores the fundamental issues encountered by traditional charts in visualizing event temporal distributions and introduces DensityBars, a novel visualization technique designed to overcome these limitations. We first conduct a preliminary user study to reveal the limitations of bin-size adjustment strategies in bar charts, particularly their demand for additional screen space. Subsequently, expert interviews further highlighted how such limitations become more pronounced in real-world visualization interface design. In response to these limitations, DensityBars leverages vertical space efficiently, embedding fine-grained density heatmaps of event occurrences within the coarse-grained bar chart. To validate the effectiveness of DensityBars, we present two real-world use cases and conduct two user studies, showing its superior capability in revealing both global and local patterns. This work concludes with insights into visual design that hold relevance for practitioners and researchers in the visualization community.

Acknowledgments

We sincerely thank all reviewers for their constructive comments. This study was supported by the Guangdong Basic and Applied Basic Research Foundation (2025A1515010162), the National Natural Science Foundation of China (62402184), the Guangdong Provincial Fund for Basic and Applied Basic Research—Regional Joint Fund Project (Key Project) (2023B1515120078), the Guangdong Provincial Natural Science Foundation for Outstanding Youth Team Project (2024B1515040010), the Science and Technology Planning Project of Guangdong Province (2025B0101120003). The work of Tobias Schreck was supported by the Austrian FFG-COMET-K1 Center Pro^{2}Future (Products and Production Systems of the Future) (No.881844).

References

[1] Moataz Abdelaal, Nathan D Schiele, Katrin Angerbauer, Kuno Kurzhals, Michael Sedlmair, and Daniel Weiskopf. 2022. Comparative Evaluation of Bipartite, Node-Link, and Mtrix-Based Network Representations. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (2022), 896–906. doi:10.1109/TVCG.2022.3209427

[2] Wolfgang Aigner, Silvia Miksch, Heidrun Schumann, and Christian Tominski. 2023. *Visualization of Time-Oriented Data, Second Edition.* Springer. doi:10.1007/978-1-4471-7527-8

[3] Leilani Battle, Peitong Duan, Zachery Miranda, Dana Mukusheva, Remco Chang, and Michael Stonebraker. 2018. Beagle: Automated Extraction and Interpretation of Visualizations from the Web. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*. ACM, 1–8. doi:10.1145/3173574.3174168

[4] Lior Berry and Tamara Munzner. 2004. BinX: Dynamic exploration of time series datasets across aggregation levels. In *IEEE Symposium on Information Visualization*. IEEE Computer Society, p2–p2. doi:10.1109/INFVIS.2004.11

[5] Michael Blumenschein, Luka J Debbeler, Nadine C Lages, Britta Renner, Daniel A Keim, and Mennatallah El-Assady. 2020. v-plots: Designing Hybrid Charts for the Comparative Analysis of Data Distributions. *Computer Graphics Forum* 39, 3 (2020), 565–577. doi:10.1111/cgf.14002

[6] Matthew Brehmer, Bongshin Lee, Petra Isenberg, and Eun Kyoung Choe. 2018. Visualizing Ranges over Time on Mobile Phones: A Task-Based Crowdsourced Evaluation. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2018), 619–629. doi:10.1109/TVCG.2018.2865234

[7] Xiwen Cai, Konstantinos Efstathiou, Xiao Xie, Yingcai Wu, Y Shi, and Lingyun Yu. 2018. A Study of the Effect of Doughnut Chart Parameters on Proportion Estimation Accuracy. *Computer Graphics Forum* 37, 6 (2018), 300–312. doi:10.1111/cgf.13325

[8] Yining Cao, Jane L E, Zhutian Chen, and Haijun Xia. 2023. DataParticles: Block-Based and Language-Oriented Authoring of Animated Unit Visualizations. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*. ACM, 1–15. doi:10.1145/3544548.3581472

[9] Cristina R Ceja, Caitlyn M McColeman, Cindy Xiong, and Steven L Franconeri. 2020. Truth or square: Aspect ratio biases recall of position encodings. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2020), 1054–1062. doi:10.1109/TVCG.2020.3030422

[10] Xi Chen, Wei Zeng, Yanna Lin, Hayder Mahdi Al-Maneea, Jonathan Roberts, and Remco Chang. 2021. Composition and Configuration Patterns in Multiple-View Visualizations. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2021), 1514–1524. doi:10.1109/TVCG.2020.3030338

[11] William S Cleveland. 1993. *Visualizing Data.* Hobart press.

[12] William S Cleveland and Robert McGill. 1984. Graphical Perception: Theory, Experimentation, and Application to the Development of Graphical Methods. *J. Amer. Statist. Assoc.* 79, 387 (1984), 531–554. doi:10.1080/01621459.1984.10478080

[13] Michael Correll, Mingwei Li, Gordon Kindlmann, and Carlos Scheidegger. 2019. Looks Good To Me: Visualizations As Sanity Checks. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 830–839. doi:10.1109/TVCG.2018.2864907

[14] Dazhen Deng, Weiwei Cui, Xiyu Meng, Mengye Xu, Yu Liao, Haidong Zhang, and Yingcai Wu. 2023. Revisiting the Design Patterns of Composite Visualizations. *IEEE Transactions on Visualization and Computer Graphics* 29, 12 (2023), 5406–5421. doi:10.1109/TVCG.2022.3213565

[15] Zikun Deng, Shifu Chen, Tobias Schreck, Dazhen Deng, Tan Tang, Mingliang Xu, Di Weng, and Yingcai Wu. 2024. Visualizing Large-Scale Spatial Time Series

with GeoChron. *IEEE Transactions on Visualization and Computer Graphics* 30, 1 (2024), 1194–1204. doi:10.1109/TVCG.2023.3327162

[16] Zikun Deng, Di Weng, Shuhan Liu, Yuan Tian, Mingliang Xu, and Yingcai Wu. 2023. A Survey of Urban Visual Analytics: Advances and Future Directions. *Computational Visual Media* 9, 1 (2023), 3–39. doi:10.1007/S41095-022-0275-7

[17] Jose Díaz, Oscar Meruvia-Pastor, and Pere-Pau Vázquez. 2018. Improving perception accuracy in bar charts with internal contrast and framing enhancements. In *Proceedings of IEEE International Conference Information Visualisation*. IEEE Computer Society, 159–168. doi:10.1109/IV.2018.00037

[18] David P Doane. 1976. Aesthetic Frequency Classifications. *The American Statistician* 30, 4 (1976), 181–183. doi:10.1080/00031305.1976.10479172

[19] Ensheng Dong, Hongru Du, and Lauren Gardner. 2020. An Interactive Web-Based Dashboard to Track COVID-19 in Real Time. *The Lancet Infectious Diseases* 20, 5 (2020), 533–534. doi:10.1016/S1473-3099(20)30120-1

[20] Hao Fan, Chuanfeng Zhao, and Yikun Yang. 2020. A Comprehensive Analysis of the Spatio-Temporal Variation of Urban Air Pollution in China during 2014–2018. *Atmospheric Environment* 220 (2020), 117066. doi:10.1016/j.atmosenv.2019.117066

[21] Yujie Fang, Hui Xu, and Jie Jiang. 2020. A Survey of Time Series Data Visualization Research. *IOP Conference Series: Materials Science and Engineering* 782, 2 (2020), 022013. doi:10.1088/1757-899X/782/2/022013

[22] Stephen Few. 2004. Show Me the Numbers. *Analytics Pres* 2 (2004).

[23] Mikalai Filonchyk, Michael Peterson, and Volha Hurynovich. 2021. Air Pollution in the Gobi Desert Region: Analysis of Dust-storm Events. *Quarterly Journal of the Royal Meteorological Society* 147, 735 (2021), 1097–1111. doi:10.1002/qj.3961

[24] David Freedman and Persi Diaconis. 1981. On the Histogram as a Density Estimator: L_2 Theory. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete* 57, 4 (1981), 453–476. doi:10.1007/BF01025868

[25] Siwei Fu, Jian Zhao, Hao Fei Cheng, Haiyi Zhu, and Jennifer Marlow. 2018. T-Cal: Understanding Team Conversational Data with Calendar-Based Visualization. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*. ACM, 1–13. doi:10.1145/3173574.3174074

[26] Peter S Games and Alark Joshi. 2014. Visualization of off-screen data on tablets using context-providing bar graphs and scatter plots. In *Visualization and Data Analysis*, Vol. 9017. SPIE, 132–146. doi:10.1117/12.2038456

[27] Tong Ge, Bongshin Lee, and Yunhai Wang. 2021. CAST: Authoring Data-Driven Chart Animations. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*. ACM, 1–15. doi:10.1145/3411764.3445452

[28] Samuel Gratzl, Alexander Lex, Nils Gehlenborg, Hanspeter Pfister, and Marc Streit. 2013. LineUp: Visual Analysis of Multi-Attribute Rankings. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2277–2286. doi:10.1109/TVCG.2013.173

[29] Ming C. Hao, Umeshwar Dayal, Daniel A. Keim, and Tobias Schreck. 2007. Multi-Resolution Techniques for Visual Exploration of Large Time-Series Data. In *Proceedings of Joint Eurographics - IEEE VGTC Symposium on Visualization, EuroVis*. Eurographics Association, 27–34. doi:10.2312/VisSym/EuroVis07/027-034

[30] Jeffrey Heer and Michael Bostock. 2010. Crowdsourcing Graphical Perception: Using Mechanical Turk to Assess Visualization Design. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*. ACM, 203–212. doi:10.1145/1753326.1753357

[31] Anja Heim, Alexander Gall, Manuela Waldner, Eduard Gröller, and Christoph Heinzl. 2024. AccuStripes: Visual Exploration and Comparison of Univariate Data Distributions Using Color and Binning. *Computers & Graphics* 119 (2024), 103906. doi:10.1016/J.CAG.2024.103906

[32] Enamul Hoque, Parsa Kavehzadeh, and Ahmed Masry. 2022. Chart Question Answering: State of the Art and Future Directions. *Computer Graphics Forum* 41, 3 (2022), 555–572. doi:10.1111/CGF.14573

[33] Mao Lin Huang, Tze-Haw Huang, and Jiawan Zhang. 2009. TreemapBar: Visualizing Additional Dimensions of Data in Bar Chart. In *Proceedings of International Conference on Information Visualisation*. IEEE Computer Society, 98–103. doi:10.1109/IV.2009.22

[34] Indratmo, Lee Howorko, Joyce Maria Boedianto, and Ben Daniel. 2018. The Efficacy of Stacked Bar Charts in Supporting Single-Attribute and Overall-Attribute Comparisons. *Visual Informatics* 2, 3 (2018), 155–165. doi:10.1016/J.VISINF.2018.09.002

[35] Nicole Jardine, Brian D. Ondov, Niklas Elmqvist, and Steven Franconeri. 2020. The Perceptual Proxies of Visual Comparison. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (2020), 1012–1021. doi:10.1109/TVCG.2019.2934786

[36] Waqas Javed and Niklas Elmqvist. 2012. Exploring the Design Space of Composite Visualization. In *Proceedings of IEEE Pacific Visualization Symposium*. IEEE Computer Society, 1–8. doi:10.1109/PACIFICVIS.2012.6183556

[37] George F Jenks. 1977. Optimal Data Classification for Choropleth Maps. *Department of Geographiy, University of Kansas Occasional Paper* (1977).

[38] Alireza Karduni, Ryan Wesslen, Isaac Cho, and Wenwen Dou. 2020. Du Bois Wrapped Bar Chart: Visualizing Categorical Data with Disproportionate Values. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*. ACM, 1–12. doi:10.1145/3313831.3376365

[39] Daniel A Keim, Ming C Hao, and Umeshwar Dayal. 2002. Hierarchical pixel bar charts. *IEEE Transactions on Visualization and Computer Graphics* 8, 3 (2002), 255–269. doi:10.1109/TVCG.2002.1021578

[40] Daniel A. Keim, Ming C. Hao, Umeshwar Dayal, and Meichun Hsu. 2002. Pixel Bar Charts: A Visualization Technique for Very Large Multi-Attribute Data Sets. *Information Visualization* 1, 1 (2002), 20–34. doi:10.1057/palgrave.ivs.9500003

[41] Milos Krstajic, Enrico Bertini, and Daniel A. Keim. 2011. CloudLines: Compact Display of Event Episodes in Multiple Time-Series. *IEEE Transactions on Visualization and Computer Graphics* 17, 12 (2011), 2432–2439. doi:10.1109/TVCG.2011.179

[42] Satsuki Kumatani, Takayuki Itoh, Yousuke Motohashi, Keisuke Umezu, and Masahiro Takatsuka. 2016. Time-Varying Data Visualization Using Clustered Heatmap and Dual Scatterplots. In *Proceedings of International Conference on Information Visualisation*. IEEE Computer Society, 63–68. doi:10.1109/IV.2016.50

[43] Zhenhui Li, Jingjing Wang, and Jiawei Han. 2014. ePeriodicity: Mining Event Periodicity from Incomplete Observations. *IEEE Transactions on Knowledge and Data Engineering* 27, 5 (2014), 1219–1232. doi:10.1109/TKDE.2014.2365801

[44] Shuhan Liu, Yuan Tian, Zikun Deng, Weiwei Cui, Haidong Zhang, Di Weng, and Yingcai Wu. 2025. Relation-Driven Query of Multiple Time Series. *IEEE Transactions on Visualization and Computer Graphics* 31, 8 (2025), 4210–4225. doi:10.1109/TVCG.2024.3397554

[45] Muller and Schumann. 2003. Visualization Methods for Time-Dependent Data-An Overview. In *Proceedings of the Winter Simulation Conference*. IEEE Computer Society, 737–745. doi:10.1109/WSC.2003.1261490

[46] Kumiyo Nakakoji, Akio Takashima, and Yasuhiro Yamamoto. 2001. Cognitive Effects of Animated Visualization in Exploratory Visual Data Analysis. In *Proceedings of International Conference on Information Visualisation*. IEEE Computer Society, 77–84. doi:10.1109/IV.2001.942042

[47] Karl Pearson. 1892. The Grammar of Science. *Nature* 46, 1185 (1892), 247–247. doi:10.1038/046247b0

[48] Vanessa Peña-Araya, Emmanuel Pietriga, and Anastasia Bezerianos. 2020. A Comparison of Visualizations for Identifying Correlation over Space and Time. *IEEE Transactions on Visualization and Computer Graphics* 26, 1 (2020), 375–385. doi:10.1109/TVCG.2019.2934807

[49] Lei Peng, Ziyue Lin, Natalia V. Andrienko, Gennady L. Andrienko, and Siming Chen. 2025. Contextualized visual analytics for multivariate events. *Visual Informatics* 9, 2 (2025), 100234. doi:10.1016/j.visinf.2025.100234

[50] Ramana Rao and Stuart K Card. 1994. The Table Lens: Merging graphical and symbolic representations in an interactive focus+context visualization for tabular information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 318–322. doi:10.1145/191666.191776

[51] Paul Rosen and Ghulam Jilani Quadri. 2020. LineSmooth: An Analytical Framework for Evaluating the Effectiveness of Smoothing Techniques on Line Charts. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2020), 1536–1546. doi:10.1109/TVCG.2020.3030421

[52] Raphael Sahann, Torsten Müller, and Johanna Schmidt. 2021. Histogram Binning Revisited with a Focus on Human Perception. In *Proceedings of IEEE Visualization Conference*. IEEE, 66–70. doi:10.1109/VIS49827.2021.9623301

[53] David W Scott. 1979. On Optimal and Data-Based Histograms. *Biometrika* 66, 3 (1979), 605–610. doi:10.1093/biomet/66.3.605

[54] Daniel Seebacher, Maximilian T. Fischer, Rita Sevastjanova, Daniel A. Keim, and Mennatallah El-Assady. 2019. Visual Analytics of Conversational Dynamics. In *Proceedings of International EuroVis Workshop on Visual Analytics*. Eurographics Association, 83–87. doi:10.2312/EUROVA.20191130

[55] Vidya Setlur, Michael Correll, and Sarah Battersby. 2022. Oscar: A Semantic-Based Data Binning Approach. In *Proceedings of IEEE Visualization and Visual Analytics*. IEEE, 100–104. doi:10.1109/VIS54862.2022.00029

[56] Conglei Shi, Siwei Fu, Qing Chen, and Huamin Qu. 2015. VisMOOC: Visualizing video clickstream data from massive open online courses. In *Proceedings of IEEE Pacific Visualization Symposium*. IEEE Computer Society, 159–166. doi:10.1109/PACIFICVIS.2015.7156373

[57] Hideaki Shimazaki and Shigeru Shinomoto. 2010. Kernel Bandwidth Optimization in Spike Rate Estimation. *Journal of Computational Neuroscience* 29, 1-2 (2010), 171–182. doi:10.1007/S10827-009-0180-4

[58] Ben Shneiderman. 2003. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *The Craft of Information Visualization*. Elsevier, 364–371. doi:10.1016/B978-155860915-0/50046-9

[59] Rober H Shumway. 2006. Time Series Analysis and Its Applications: With R Examples.

[60] Praveen Soni, Cyril de Runz, Fatma Bouali, and Gilles Venturini. 2024. A Survey on Automatic Dashboard Recommendation Systems. *Visual Informatics* 8, 1 (2024), 67–79. doi:10.1016/J.VISINF.2024.01.002

[61] Herbert A Sturges. 1926. The Choice of a Class Interval. *J. Amer. Statist. Assoc.* 21, 153 (1926), 65–66.

[62] Justin Talbot, Vidya Setlur, and Anushka Anand. 2014. Four Experiments on the Perception of Bar Charts. *IEEE Transactions on Visualization and Computer Graphics* 20, 12 (2014), 2152–2160. doi:10.1109/TVCG.2014.2346320

[63] Junxiu Tang, Fumeng Yang, Jiang Wu, Yifang Wang, Jiayi Zhou, Xiwen Cai, Lingyun Yu, and Yingcai Wu. 2024. A Comparative Study on Fixed-Order Event Sequence Visualizations: Gantt, Extended Gantt, and Stringline Charts. *IEEE Transactions on Visualization and Computer Graphics* 30, 12 (2024), 7687–7701.

doi:10.1109/TVCG.2024.3358919

[64] Junxiu Tang, Yuhua Zhou, Tan Tang, Di Weng, Boyang Xie, Lingyun Yu, Huaqiang Zhang, and Yingcai Wu. 2022. A Visualization Approach for Monitoring Order Processing in E-Commerce Warehouse. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2022), 857–867. doi:10.1109/TVCG.2021.3114878

[65] Chatpong Tangmanee and Phattapang Suriyakul Na Ayutthaya. 2024. How Bar Chart Display Features Can Skew Perception. *Journal of Ecohumanism* 3, 7 (2024), 971–981.

[66] Yuan Tian, Weiwei Cui, Dazhen Deng, Xinjing Yi, Yurun Yang, Haidong Zhang, and Yingcai Wu. 2025. ChartGpt: Leveraging LLMs to Generate Charts from Abstract Natural Language. *IEEE Transactions on Visualization and Computer Graphics* 31, 3 (2025), 1731–1745. doi:10.1109/TVCG.2024.3368621

[67] Christian Tominski, Stefan Gladisch, Ulrike Kister, Raimund Dachselt, and Heidrun Schumann. 2017. Interactive Lenses for Visualization: An Extended Survey. *Computer Graphics Forum* 36, 6 (2017), 173–200. doi:10.1111/CGF.12871

[68] Jarke J Van Wijk and Edward R Van Selow. 1999. Cluster and Calendar based Visualization of Time Series Data. In *Proceedings of IEEE Symposium on Information Visualization*. IEEE Computer Society, 4–9. doi:10.1109/INFVIS.1999.801851

[69] Guijuan Wang, Yadong Wu, Jiaxin Yang, Han Su, Ziyi Tang, Changwei Luo, Lu Tong, Xiaorong Zhang, Wenjuan Bai, and Lijuan Peng. 2025. COVIDashboardsWild: An Interactive Review on COVID-19 Visualization Dashboards in the Wild. *Visual Informatics* (2025), 100287. doi:10.1016/j.visinf.2025.100287

[70] Yunhai Wang, Fubo Han, Lifeng Zhu, Oliver Deussen, and Baoquan Chen. 2017. Line Graph or Scatter Plot? Automatic Selection of Methods for Visualizing Trends in Time Series. *IEEE Transactions on Visualization and Computer Graphics* 24, 2 (2017), 1141–1154. doi:10.1109/TVCG.2017.2653106

[71] Colin Ware. 2019. *Information Visualization: Perception for Design*. Morgan Kaufmann.

[72] Leland Wilkinson. 1999. Dot Plots. *The American Statistician* 53, 3 (1999), 276–281. doi:10.1080/00031305.1999.10474474

[73] Yingcai Wu, Di Weng, Zikun Deng, Jie Bao, Mingliang Xu, Zhangye Wang, Yu Zheng, Zhiyu Ding, and Wei Chen. 2021. Towards Better Detection and Analysis of Massive Spatiotemporal Co-Occurrence Patterns. *IEEE Transactions on Intelligent Transportation Systems* 22, 6 (2021), 3387–3402. doi:10.1109/TITS.2020.2983226

[74] Cindy Xiong, Vidya Setlur, Benjamin Bach, Eunyee Koh, Kylie Lin, and Steven Franconeri. 2022. Visual Arrangements of Bar Charts Influence Comparisons in Viewer Takeaways. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2022), 955–965. doi:10.1109/TVCG.2021.3114823

[75] Fan Yan, Yong Wang, Xuanwu Yue, Kamkwai Wong, Ketian Mao, Rong Zhang, Huamin Qu, Haiyang Zhu, Minfeng Zhu, and Wei Chen. 2025. FundSelector: A visual analysis system for mutual fund selection. *Visual Informatics* 9, 4 (2025), 100258. doi:10.1016/J.VISINF.2025.100258

[76] Yalong Yang, Wenyu Xia, Fritz Lekschas, Carolina Nobre, Robert Krüger, and Hanspeter Pfister. 2022. The Pattern is in the Details: An Evaluation of Interaction Techniques for Locating, Searching, and Contextualizing Details in Multivariate Matrix Visualizations. In *Proceedings of ACM CHI Conference on Human Factors in Computing Systems*. ACM, 1–15. doi:10.1145/3491102.3517673