

Evaluating Pan and Zoom Timelines and Sliders

Michail Schwab
Northeastern University
Boston, Massachusetts
michaschwab@ccs.neu.edu

Sicheng Hao
Northeastern University
Boston, Massachusetts
hao.sic@husky.neu.edu

Olga Vitek
Northeastern University
Boston, Massachusetts
o.vitek@northeastern.edu

James Tompkin
Brown University
Providence, Rhode Island
james_tompkin@brown.edu

Jeff Huang
Brown University
Providence, Rhode Island
jeff_huang@brown.edu

Michelle A. Borkin
Northeastern University
Boston, Massachusetts
m.borkin@northeastern.edu

ABSTRACT

Pan and zoom timelines and sliders help us navigate large time series data. However, designing efficient interactions can be difficult. We study pan and zoom methods via crowd-sourced experiments on mobile and computer devices, asking which designs and interactions provide faster target acquisition. We find that visual context should be limited for low-distance navigation, but added for far-distance navigation; that timelines should be oriented along the longer axis, especially on mobile; and that, as compared to default techniques, *double click*, *hold*, and *rub* zoom appear to scale worse with task difficulty, whereas *brush* and especially *ortho* zoom seem to scale better. Software and data used in this research are available as open source.

CCS CONCEPTS

• **Human-centered computing** → **Empirical studies in visualization**; **Empirical studies in interaction design**.

KEYWORDS

Pan, zoom, visualization, timeline, interaction, evaluation.

ACM Reference Format:

Michail Schwab, Sicheng Hao, Olga Vitek, James Tompkin, Jeff Huang, and Michelle A. Borkin. 2019. Evaluating Pan and Zoom Timelines and Sliders. In *CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019), May 4–9, 2019, Glasgow, Scotland Uk*. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3290605.3300786>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org. *CHI 2019, May 4–9, 2019, Glasgow, Scotland Uk*

© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-5970-2/19/05...\$15.00

<https://doi.org/10.1145/3290605.3300786>

1 INTRODUCTION

Pan and zoom interactions allow us to explore multi-scale timeline data with 10^{6+} spans on fixed-size displays [8]. As large-span datasets increase in prevalence, visualization creators will increasingly ask how best to maximize usability in their applications. Navigation efficiency is integral to usability: it can be dramatically affected by timeline design and interaction technique; being able to turn an interface from one that frustrates users to one that supports their goals. Brehmer et al. recently reviewed timeline appearance, and identified many different layouts, shapes, and scale types [5]. However, little is known about the effectiveness of these timeline designs for multi-scale navigation across the many options for pan and zoom interaction techniques. To provide timeline creators with useful guidance, studies must be run which are applicable to the Web—a highly uncontrolled environment accessed through a multitude of devices.

We present a systematic study of multi-scale timeline and interaction designs in an experiment with 318 participants across **mobile and desktop** devices. With this study, we answer the following three questions:

- Q1: **Do participants navigate a date-based timeline faster than a pure numberline?** We find that visual context from dates can be helpful for far-distance navigation, but decreases short-distance navigation speed.
- Q2: **Do participants navigate timelines faster with a horizontal or a vertical layout?** We find that orienting timelines along the longer axis of devices improves navigation efficiency, and an orientation which maps intuitively to hardware also improves performance, e.g., *scroll wheel* pan is faster vertically than horizontally.
- Q3: **From a set of eight different pan and zoom interaction pairings from the literature, which is fastest for timelines?** We find the default *pinch* zoom on mobile and *scroll wheel* zoom on computer are fastest overall, as might be expected. We find that the pan technique is important for overall navigation speed, and that compared to default techniques, *2x click*, *hold*, and *rub* zoom seem to scale worse with task difficulty. In contrast, *brush*

and especially *ortho* zoom seem to scale better than default techniques. Given users' unfamiliarity with these techniques, this is a strong indicator that brush and especially *ortho* zoom may outperform the default technique with more practice, especially for high task difficulties.

We interpret our study outcomes and discuss their impact on multi-scale timeline design. In our supplemental material, we review 10 different pan and zoom methods considered for the study, including a matrix of method compatibility and all parameters used. To maximize reproducibility, we publish all technique implementations as open source software.

2 RELATED WORK

Much work addresses multi-scale navigation, e.g., space-scale diagrams [12], map visualization choice [18], smooth animation [39], visual aids for multi-scale navigation [9, 34, 40], and topology-based interaction techniques [21]. We focus on timelines and general-purpose interaction techniques.

Timeline Visualization

Timelines are an essential and historic visual encoding of events in chronological order and a basic building block of many visualizations [33, 36]. Brehmer et al. review the design space of timeline shapes (e.g., linear, radial, grid), scales (e.g., chronological, relative, logarithmic) and layouts (e.g., unified, faceted, segmented) [5]. This helps us comprehend visual choices, but little is known about the consequences of these choices: which is easiest to understand or to interact with. While Brehmer et al. published an evaluation of understandability of two timeline designs [6], performance in most of the design space is still unknown. In particular, no study reports the influence of interaction on design, which is a critical component of any multi-scale timeline.

We contribute an evaluation of two timeline design questions about visual context and timeline orientation, and so provide design recommendations from our navigation study.

Multi-scale Interaction

Existing Interactions: A large body of work exists on interaction techniques for pan and zoom in multi-scale spaces, and new techniques are frequently proposed. Early multi-scale or zoomable interfaces, such as Pad, Pad++, and MuSE [4, 13, 32] defined zoom based on holding a mouse button down or dragging a slider. The interaction design space has been explored through brush or marquee zoom, anchor or *ortho* zoom [1, 2], pinch to zoom [24], elastic interaction for precise manipulation [28], pan-speed-dependent zooming [19], flick pan or zoom [35], and cyclic gestures based on rubbing [31] or circles [27]. Moving beyond 2D pointing devices, the literature explores mid-air gestures [30], Webcam-based zooms for desktop [16] and mobile [22], plus mobile-specific tilt sensors [17]

and camera-based device spatial positioning [38]. This dense network of techniques is difficult to grasp.

New Interactions: When new interaction techniques are proposed, they are typically compared against only a few related techniques. This makes it difficult to gain an overview of technique performance across the field. Further, these are often implemented in isolation and closed source, which makes reproduction and comparison difficult, with researchers even going so far as to reverse engineer interaction techniques to understand their mechanics [35]. This is also true of parameter selection, which can be critical to performance.

We contribute 32 open-source implementations to aid reproducibility, and provide a systematic evaluation of seven of these combined techniques (Figure 2).

Motor Control Theory: Design and technique comparisons are possible thanks to human motor control theory [3]. Guiard and Beaudouin-Lafon empirically showed that Fitts' Law [11, 26] can apply to multi-scale pointing [15]. Across indices of difficulty (*ID*) from 5–30 bits, target pointing measurement time *T* varied linearly with target *ID*, or logarithmically with target distance, with performance variation represented by the slope of the line. This models different multi-scale interaction technique performance, where intercept and slope parameters *a* and *b* vary per technique:

$$T = a + b \cdot ID = a + b \cdot \log_2 \left(\frac{\text{Distance}}{\text{Target Size}} + 1 \right) \quad (1)$$

We observe that Fitts' law holds for our data obtained in the wild, which helps us compare visualization and interaction techniques for this uncontrolled environment.

Guiard and Beaudouin-Lafon continue to describe how the user must decide how to use the pan, zoom in, and zoom out functions during multi-scale interaction. Below ~8 bit *ID*s, the user may pan only, but high-bit *ID* targets require 10+ seconds of interaction with a combination of zoom out, zoom in, and pan [15]. Interaction time is affected by timeline appearance, as the user must orient the data view to their mental map of the data, e.g., higher cognitive load can increase 'idle' time. Both the interaction difficulty and the human data processing can cause 'desert fog' [23], where a user becomes lost in multi-scale space, e.g., by zooming in to an incorrect target.

Summary

Little guidance exists for multi-scale timeline visualization and interaction design, in part because technique details are not available and so systematic and reproducible pan and zoom evaluation is difficult. To address this, we implement and evaluate multiple timeline designs and interactions to provide actionable design recommendations to timeline creators, and open source our software and data.



Figure 1: The study setup as seen by participants: Pinch zooming (left) and brush zooming (right) on a horizontal timeline. Top: Timeline as seen when zoomed out. Bottom: Timeline as seen when zoomed in all the way to a time span of four minutes.

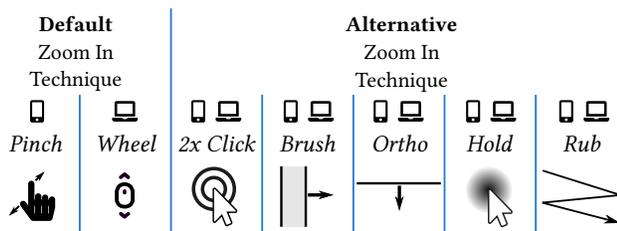


Figure 2: Schematic overview of zoom in interactions used in this study. Ortho zoom is dragging perpendicularly to the axis of the timeline, rub zoom is dragging back and forth.

3 OPEN-SOURCING INTERACTION TECHNIQUES

To enable systematic and fair evaluation of techniques, we implemented 32 pan and zoom techniques, including variations, and contributed them to EasyPZ [37], an open-source pan and zoom JavaScript library which works across mobile and desktop computers. EasyPZ enables easy use and tuning of these techniques on any data visualization. Contribution of these techniques to this open-source platform allows future research and evaluations to build on this work.

Implemented Interactions

The implemented interactions are largely from existing literature, although some have received improvements from their original design, such as a continuous zoom on rub zoom as opposed to the original step-wise implementation.

We chose these techniques based on their popularity in existing software, availability in the literature, and compatibility with both computers and mobile phones, and added some simple variations such as quadratic scaling to pinch zoom.

Pan: We implement default desktop dragging with a linear transfer function (*drag pan*), default mobile dragging with flick momentum (*flick pan* [35]), scroll wheel on a desktop mouse or a scroll gesture on a touchpad (*scroll pan*), and using the touchscreen with a two-finger motion (*two-finger pan*).

Zoom: The zoom techniques are shown in Figure 2. We implement double click or tap (*2x click zoom*), holding down a button or pressing a touchscreen without releasing the finger

(*hold zoom* [4, 32]), using the scroll wheel on a desktop mouse or a scroll gesture on a touchpad (*scroll zoom*), two-finger pinch to zoom (*pinch zoom* [24]), drag-selecting a rectangular zoom region, often called marquee zoom or brushing (*brush zoom*), dragging along the orthogonal axis for continuous zoom granularity control (*ortho zoom* or *dynamic zoom* [1, 2]), and the cyclic gesture of bi-directional rubbing to zoom in and out (*rub zoom* [31]), e.g., where rubbing horizontally zooms in and rubbing vertically zooms out.

Compatibility: Given a pointing device, only some interaction combinations are compatible. This design space includes parameters like zoom speed, minimum interaction time, or pointer speed bounds during the beginning of an interaction. We tabulate these compatibilities in our supplemental material as a reference to visualization creators.

4 USER STUDY DESIGN AND PROCEDURE

Next, we describe the experimental procedure, the gathered data, the data analysis and results of the user study designed for this task. We use 1D timelines as in Figure 1.

Motivation and Design Rationale

In-the-wild Study: We wish to find practical insights into which timeline visualizations and interactions perform best given the wide range of screens and input devices in use, in contrast to well-defined laboratory settings which may or may not be applicable in the real world, as online studies include more user errors than lab data [10]. For this reason, our study does not control user nor input devices, and is run instead as an *in situ* study on Amazon Mechanical Turk. This approach conserves the unknown real-world variability which faces Web application designers today.

Mobile Phones and Orientation: The growing use of smart phones, and increased use of visualizations on mobile phones, motivated us to find interaction design insights for both laptops/PCs (called “computers” here), as well as mobile smart phones (called “mobile” here, excluding tablets). Given varying aspect ratios across different devices, we decided that evaluation of impact of timeline orientation was essential.

Table 1: Our four participant sets, their chosen fixed study settings and compared factor levels, and to which questions their data were applied. Each Q1-3  and Q1-3  symbolizes that the data was used for question 1-3 for mobile or desktop, and represents one of six analysis models. Each column is one set of ≈40 people who all underwent the same conditions. For Q1 and Q2, the study design is balanced, whereas for Q3 the study design is unbalanced.

Participant Set	Participant Set 1		Participant Set 2		Participant Set 3		Participant Set 4	
Question	Q1: Representation		Q2 & Q3: Orientation & Method		Q3: Method		Q3: Method	
Representation Orientation	(Varied) Horizontal		Dates (Varied)		Dates Vertical		Dates Vertical	
Platform # Participants	Mobile 40	Computer 43	Mobile 38	Computer 42	Mobile 35	Computer 40	Mobile 41	Computer 39
Orientation Setting: Horizontal								
Dates								
Numbers								
Representation Setting: Dates, Orientation Setting: Horizontal								
Default								
Ortho								
Brush								
Representation Setting: Dates, Orientation Setting: Vertical								
Default			 	 				
Ortho			 					
Hold								
2x Click								
Brush				 				
Rub								

Simulated “Locate” Task: In a real pan and zoom timeline navigation application, the target location is not precisely known in a “locate” task (target known, location unknown [7]). If the target were known (“lookup”), then a text or other direct input option would be more efficient for navigation than navigating a timeline. To simulate a “locate” task, participants were instructed to navigate to specific targets on the timeline without visually showing the target location on the timeline. This is a difference to traditional Fitts’ law experiments, where targets are typically shown. We believe a design that visually displays the target on the timeline would simplify the navigation task too much to be applicable to a *locate* task, where maintaining context by visually parsing the scene is important. This design accounts better for target acquisition issues such as *desert fog* (see Section 2).

Number Challenges: In a pilot study with a numbers-based pan and zoom timeline, we observed navigation difficulties and errors among our participants: As the length of the target numbers was very long—8 digits—pilot participants commonly forgot the exact target number and struggled to find it. As years, months, and dates provide a natural hierarchical grouping (e.g., Miller’s “magical number 7” memory

capacity maxim [29]), we hypothesized an increased recall and improved target acquisition when using dates over numbers. This can be a visualization choice: application examples include ‘friendly dates’ where recent events are described in relative terms (“last week”) but distant events are absolute (“Jan 2012”), and video editor time encoding as hours, minutes, and seconds, or as absolute number of frames (e.g., Adobe Premiere Pro). Thus, we study the effect of this additional context on navigation speed. We chose a minimal timeline layout to reduce the influence of any particular design, but balanced this goal with the observation that far-distance navigation can be difficult without visual context.

Tasks for Range of Difficulties: From pilot lab studies, we concluded that method combinations may be more or less suited for a task depending on the task difficulty. For example, scroll wheel panning on a vertical timeline may allow to pan quickly to nearby targets, but may not scale well for navigating to far targets, depending on the paired technique for zooming. Different applications have different typical navigation distances, such as subway maps typically not requiring pan and zoom between targets very far apart, whereas navigating a global map can involve pan and zoom between points

much farther apart. We decided in our study to cover a set of tasks which show these different strengths so that applications with a particular need could choose the most appropriate techniques. Since our target size was fixed, the task difficulty is a function of only the target distance.

Procedure: Data Collection and Cleaning

An overview of the study design is presented in Table 1 and includes participant sets, compared factor levels, fixed settings, and questions answered. Below, we describe the experiment procedure, data collection, and data cleaning.

Participant Requirements: Participants were required to be located in the United States, have a HIT Approval Rate greater than or equal to 98%, and have more than 500 approved HITs. Participants were monetarily compensated for their participation following approved IRB protocol.

Pay and Restrictions: Pay was fixed per task, which provided a soft incentive to perform the task quickly to increase wage per hour. The average pay per hour was \$8.24. Participants were allowed to participate once per study to minimize learning effects of the methods. This included restricting participants from taking the mobile version of a study after completing the computer version, and vice versa. As each study had its own settings and methods, we did allow participants to participate in multiple studies, e.g. they could participate up to once per “participant set” in Table 1.

Participant Statistics: A total of 318 participants (225 unique) successfully completed the studies. Of those, 135 identified as female, 178 as male, 4 as other gender, and one participant did not wish to disclose their gender. Participants in our study ranged from age 18 to over 55, with the majority of participants between the ages of 18–24 (147 participants). More details about participants, such as average computer use, are available in our supplemental material.

Study Process: Each study was split into a mobile and a desktop version, and participants chose their input devices freely. This reflects the study goal of real-world applicability, as visualization creators can not practically force specific input devices to be used. All participants were asked to perform the study in the Chrome web browser for practical considerations such as ensuring consistent appearance. Each participant was informed of the duration, pay, procedure, and goal of the study, informed of their rights, and asked for their informed consent through our institution-approved IRB protocol. The order of the compared factors was randomized, such as representation, orientation and technique. For each compared factor, there was a short description of the current timeline design and interaction method, plus free time to test the interaction before the start of the experiment. Then, three sets of task ‘runs’ proceeded: first, a test run with four multi-scale pointing tasks in a random order (this data was not used for analysis); then, two study runs one after the other, each

with the full set of eight multi-scale pointing task distances in a random order. On average, participants spent about 8 minutes per experimental factor on desktop, and 13 minutes on mobile. In total, participants took between 20 and 60 minutes to complete the study, depending on whether they were in participant set 1, 2, 3 or 4.

Task: In each trial, participants were asked to find a known target time/date on a pan and zoom timeline or number line. To investigate performance over multi-scale data, we used a linear number line spanning from zero to 20 million, which corresponds to all times to the minute over a period of 38 years for the timeline. With this number line, we asked participants to navigate by 10, 20, 40, 100, 1,000, 100,000, and 5,000,000 numbers, or 10, 20, and 40 minutes, 1.66 and 16 hours, 69 days, and ~10 years on the timeline. These represent index of difficulties of 3.3, 4.3, 5.3, 6.6, 10, 16.6, and 22.3 bits.

Study Design: The study design, with all data collected separated by participant set, is presented in Table 1. It shows which data was used for which of the three questions and their respective models, which settings were fixed for each participant set, and which factor was varied and compared. As the goal of this study is not to compare visualizations and interaction techniques on mobile versus desktop, but rather examine them independently for each platform, we did not compare methods across mobile/desktop in our analysis and modeled them separately. Hence, our study design is set up for one model per question and platform, resulting in six models in total. Each of the six models is represented in the table as Q1-3  and Q1-3 .

Data Collection: For each task, completion times were collected from the moment a participant started a task until the target was found and selected by clicking. In addition, data was collected about the user behavior during the task. In particular, we detected whether users were zooming in, zooming out, panning, or idling, and created a timeline for each participants’ tasks. Some of this data is shown in the Supplemental Material, along with graphs presenting which type of interaction is prevalent for each distance.

Outlier Removal: Task times were only removed when users were not paying attention: If a user was not navigating for more than 10 seconds during a task, then we removed that participant’s data for that task (task duration typically ranged from 5–50 seconds). In total, this filtered out 7% of the data, in line with other crowdsourced work (cf. 11% [14]).

Data Validation via Fitts’ Law

This predicts that task time for a pointing task will increase logarithmically with target distance, or linearly with task difficulty (Section 2). Figure 3 presents task times across all studies and study factors in aggregate (representation, orientation and method). Although there is some variation, we see

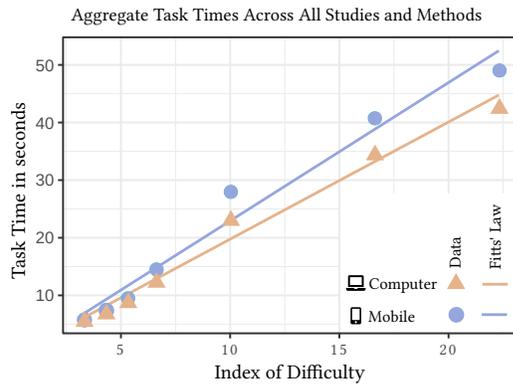


Figure 3: All study data aggregated over all representations, orientations, and methods, for computers (orange triangles) and mobile (blue circles). For both platforms, the linear Fitts' law relationship between task time and task difficulty is visible. Error bars are not shown since this data does not follow a normal distribution and is not used for our analysis.

a logarithmic trend of task time to target distance, or a linear trend to task difficulty, on both platforms. The R^2 scores for computer and mobile are 0.983 and 0.972, respectively. While not perfect, our data conform quite well to Fitts' Law in this highly uncontrolled environment, which helps validate this type of real-world study.

Analysis Procedure

To test our hypotheses H1–3, we develop a model based on Fitts' law. This varies slightly from classical Fitts' to consider the challenges of an in-the-wild study, such as right-skewed user performance distributions. To assess the strength of associations, we fit a linear mixed effects model which includes task difficulty, compared factor level, and participant ID. The model considers:

Fitt's Law: This [11] predicts task times to increase linearly with the index of difficulty (denoted α).

Comparisons: A change in timeline design or interaction can cause a constant change in task time (denoted β), such as extra needed time to orient independent of difficulty of task, as well as a change in task time that scales with task difficulty (denoted γ), such as slower navigation. This corresponds to a factor scaling differently with task difficulty compared to the baseline: γ is the difference in slope in Fitts' law.

Individual differences: We find that some participants typically performed tasks faster than other participants, but that there was no significant effect caused by some participants being particularly efficient with some specific methods or tasks. We assume the participant as a random effect to be Normally distributed and uncorrelated with the independent variables such as task difficulty and method (denoted id). This model is able to explain the variation in the data significantly better than a linear model without random effect.

Log transformation: To test our hypotheses, the model assumes task times to be Normally distributed. However, the measured task times were heavily right-skewed. We apply a log transformation to conform the task times to a Normal distribution, which is the standard approach to right-skewed data in statistics (skewness of residuals is reduced from 2.2 to 0.4; cf. Q–Q plots in supplemental analysis).

Model: If T denotes task completion time, β_i and γ_i the intercept and slope contribution from the representation or orientation or method, j the index of the participant, and r the index of the repeated measurement, then the model is:

$$\begin{aligned} \log(T)_{ijr} = & \text{Intercept} + \alpha \cdot \log(\text{difficulty}) & (2) \\ & + \beta_i + \gamma_i \cdot \log(\text{difficulty}) \\ & + id_j + \epsilon_{ijr}, \text{ where} \\ & \gamma_1 = 0, & \beta_1 = 0 \text{ as baseline levels,} \\ & id_j \stackrel{iid}{\sim} \mathcal{N}(0, \tau^2), & \epsilon_{ijr} \stackrel{iid}{\sim} \mathcal{N}(0, \sigma^2) \end{aligned}$$

All id_j and ϵ_{ijr} are assumed to be independent. Hypothesis testing on the terms $\beta_i + \gamma_i \cdot \log(\text{difficulty})$ evaluates the contribution of each representation, orientation, and method against the baseline factor level. We compute confidence bands for the expected values, and use these for hypothesis testing: If they do not include 0, this corresponds to a p -value of $\alpha = 5\%$ as family-wise error rate. To arrive at this family-wise error rate, we use an individual test error rate of $\alpha/16 \approx 0.3\%$ to account for multiple testing using a Bonferroni correction of 16. We use confidence bands instead of p -values because of their better visual representation of both direction of change and uncertainty, as well as for their taking into account multiple testing within each comparison, as they consist of many confidence intervals. We use R's LMER package to fit the linear mixed effects model, and use INTERPLOT to compute and plot confidence bands.

5 USER STUDIES RESULTS

Q1: Impact of Visual Context and Hierarchy

Introduction: Our first study examines basic linear sequential data representations: Q1: "Do participants navigate a date-based timeline faster than a pure numberline?" This represents a visualization choice, and validates the importance of context when navigating multi-scale data. As hypothesis H1, we expect visual context to aid navigation, and hence expect dates to perform faster than numbers.

Design: This study had two conditions: a timeline with a visual hierarchy denoting decades, years, months and dates, and numbers marked on a horizontal line without a visual hierarchy. To minimize impact from other factors, we kept other settings close to a 'default' interaction per platform: *pinch zoom* and *flick pan* for mobile, and *scroll zoom* and *drag pan* on desktop. Participants performed tasks for all factors

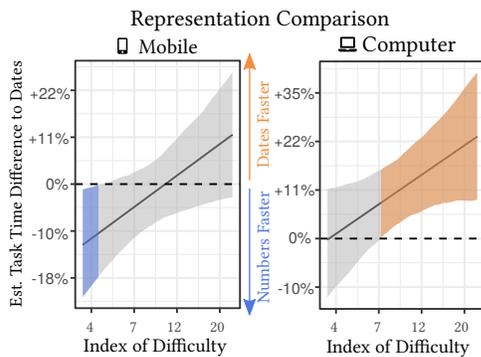


Figure 4: Confidence bands for the expected value of the numbers vs. dates representation for mobile (left) and computer (right). If the slanted black line is below zero, then numbers are faster, else dates are faster. The gray shading indicates the 95% confidence band, and significant results are colored in. On mobile, numbers are faster for low difficulties, whereas there is no significant difference on computer for low difficulties, and numbers are slower for high difficulties. Axes are non-linear due to analyzing the data in log space.

in a balanced study design. This can be seen for participant set 1 in Table 1, with the models $Q1_{\text{M}}$ and $Q1_{\text{C}}$.

We chose the dates representation as the baseline, i.e., $\beta_1 = \gamma_1 = 0$, and β_2 and γ_2 represent the variation of the numbers-based representation from the dates-based representation.

Results—Figure 4: For mobile, numbers are significantly faster (~11% or 1s) than dates for low difficulty tasks which primarily involve panning. For more difficult tasks which involve zooming, the trend reverses, but is insignificant. For low difficulty tasks on computer, there is no significant difference between dates and numbers, but for difficult tasks, dates are significantly faster than numbers (8–22% or 1.1–9.7s). Compared to numbers-based timelines, timelines with visual context and hierarchy from dates were navigated slower for low-distance targets, but faster for far-distance targets.

Q2: Impact of Timeline Orientation

Introduction: Our second study assesses a common interface design decision: Q2: “Do participants navigate pan and zoom timelines faster with horizontal or vertical layout?” Smart phones were required to be used in portrait mode, so we know that vertical timelines were longer than horizontal ones. This is practical for real-world visualizations on smart phones: at initial Webpage load, smart phones are in portrait mode about 90% of the time [20]. For hypothesis H2, we expect horizontal timelines to perform faster than vertical ones because of user familiarity with the more common format for data visualizations (e.g., in a Google image search for “d3 timeline”, the first 100 linear timelines are all horizontal).

Design: We compare timelines with horizontal and vertical orientations given common display size constraints: phones

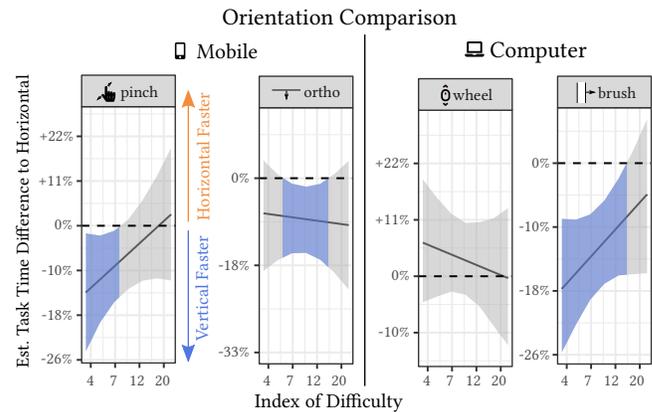


Figure 5: Confidence bands for the expected value of the vertical vs. horizontal orientation across difficulties for pinch and ortho zoom on mobile (left) and wheel and brush zoom on computer (right). A line below zero means that vertical is faster; above zero means that horizontal is faster.

are used in portrait mode, and for computers, we limit the width to 1000 pixels (a common maximum width on Websites), and use the maximum height available – 810 pixels on average. Hence for mobile phones, the vertical orientation results in a significantly longer timeline, whereas on computer, the timeline lengths are more comparable between vertical and horizontal.

We use the devices’ ‘default’ interactions. However, we also pick an additional interaction combination for each platform where we assume performance to be affected by the orientation change. For mobile, we chose *ortho* zoom and *flick* pan. For desktop, we chose *brush* zoom and *scroll* pan. These choices are not intended to be equivalent across devices; merely illustrative of a change in interaction performance. Interaction and orientation were varied within-subjects in a balanced study design. The data used for the two models in the analysis is indicated by $Q2_{\text{M}}$ and $Q2_{\text{C}}$ in Table 1.

The analysis was executed with orientation and method as a combined four-level factor, with the baselines $\beta_1 = \gamma_1 = 0$ chosen to be the horizontal orientation of the techniques.

Results—Figure 5: On mobile, participants performed significantly faster (8–14% or 1–1.2s) using the longer vertical orientation with pinch zoom compared to the shorter horizontal orientation for low difficulties, whereas there is no significant difference for high difficulties. For *ortho* zoom on mobile, the vertical orientation is significantly faster for medium difficulties (~10% or 2.7s). For computer, *wheel* zoom shows no significant impact of orientation. *Brush* zoom with *scroll wheel* pan performs significantly better vertically than horizontally (9–18% or 1.1–3s) for low to medium difficulties that primarily involve panning. In summary, we find that timelines oriented along the longer axis of devices are navigated faster, and that *scroll wheel* pan performs better vertically.

Table 2: Interaction combinations used in Study 3. Default zoom in refers to the device defaults of pinch zoom on mobile, and wheel zoom on desktop.

Zoom In	Zoom Out		Pan	
	☑ Mobile	☒ Computer	☑	☒
<i>Default</i>	Pinch	Wheel	Flick	Drag
<i>2x Click</i>	Hold	Hold	Flick	Drag
<i>Brush</i>	2x Click	2x Click	2 Finger	Wheel
<i>Ortho</i>	Ortho	Ortho	Flick	Drag
<i>Hold</i>	Click + Hold	Click + Hold	Flick	Drag
<i>Rub</i>	Rub Y	Rub Y	Flick	Drag

Q3: Impact of interaction techniques

Introduction: From Q2 and Q3, we picked the representation which performed best across participants—vertical timelines—for use in our third study on interaction design, Q3: “From a set of eight different pan and zoom interaction pairings from the literature, which is fastest for timelines?” As hypothesis **H3**, we expect default techniques to perform better than less familiar techniques (brush, hold zoom) or steeper-learning-curve techniques (ortho and rub zoom).

Design: For the study, we must pick a subset of technique combinations to test as there are too many combinations to compare in a reasonable amount of participant time. Given many possible technique comparisons and limits on user fatigue, we chose to compare popular techniques (*pinch*, *wheel*, *2x click*, *brush*) along with a sample of less common techniques from recent literature (*hold*, *rub*, *ortho*). We exclude less common smartphone-specific techniques like accelerometer or camera-based interfaces, and focus on techniques for both computer and smartphones. With this constraint, we paired each zoom in technique with the most complementary zoom out and pan methods as assessed by our judgment and in pilot studies. Table 2 lists our final combinations.

While we wished to maintain a within-subjects design, six combinations per platform would lead to a total study time per participant of 80 minutes or more on mobile. At this length, fatigue effects are likely to affect any differences between interaction designs. As such, we increased the participant pool and split the study for this question into the three participant sets. Each set compared between one and three interaction techniques to the ‘default’ platform method. How each of the participant sets 2–4 contributed to Q3☑ and Q3☒ can be seen in Table 1. Because of this split, Question 3 is investigated in an unbalanced study design.

The analysis was completed with a six-level factor as the method. The baseline was chosen as the default set of interaction techniques from each platform. Hence, $\beta_1 = \gamma_1 = 0$ represent the default technique, and $\beta_{2...6}$ and $\gamma_{2...6}$ represent the compared techniques.

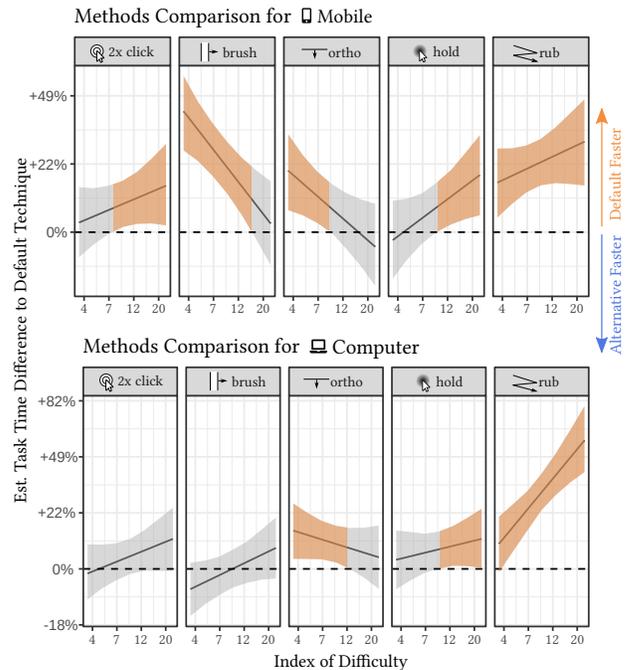


Figure 6: Confidence bands for the expected value of the different methods vs. platform default methods (☑ : pinch zoom; ☒ : scroll wheel zoom), across difficulties, for mobile and computer. Lines below zero show that the alternative method is faster; above zero show that the default method is faster.

Results—Figure 6: As expected, all techniques are significantly slower than the default technique on at least one platform. For high difficulty tasks, *2x click* and *hold* are 11% (5.8s) and 19% (10s) slower than *pinch* zoom on mobile, respectively, and *hold* is 9% (4s) slower on computer. *Rub* zoom is significantly slower than the default on both platforms across all difficulties by 9–58% (0.6–25s).

Results generally agree between the platforms, except for *brush* zoom: due to it requiring click/tap and drag, it must pair with an alternative pan technique: *two-finger pan* for mobile (2.4s slower for short-distance navigation), and *scroll wheel pan* for computer. The large difference between the two platforms’ results for brush zoom highlights that the pan technique is important for overall performance.

Finally, compared to the default technique, *2x click*, *hold* and *rub* zoom have worse scaling with task difficulty, whereas *brush* and especially *ortho* zoom scale better. *Ortho* zoom is the only technique that is estimated to perform comparably better as difficulty increases across platforms. This indicates it is particularly effective at large scales. However, the slope itself is not hypothesis-tested.

To analyze user behavior, we compare both time spent actively navigating vs. idling. For high difficulty tasks, idle time was about ~30% of task time on mobile and ~40% on computers, with similar trends across platforms. The technique with the

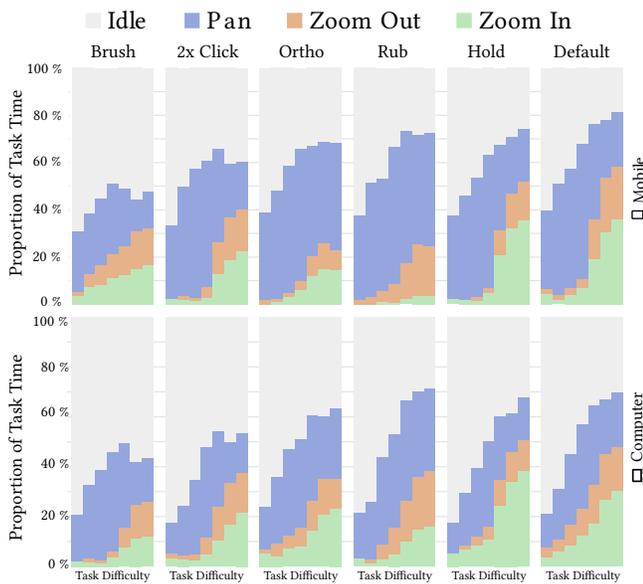


Figure 7: Proportion of task time per technique and task difficulty, for \square mobile and \square computer. Participants idle most for brush zoom (left), and least for default techniques (right).

highest idle proportion is brush zoom, with ~50%. This is indicative of fast context switches leading to ‘desert fog’. There is less difference among the other techniques, with default techniques having the lowest idle time at ~25% on mobile (Fig. 7).

Further, to quantify desert fog, we define an error as zooming in for at least 4 seconds before realizing there was a mistake and zooming out for at least 4 seconds. We observe that brush zoom led to fewer errors on computer than on mobile (0.6 vs. 2 tasks with errors per participant), presumably due to the lack of a simple alternative pan technique on mobile. Rub zoom has a low error rate on mobile at 0.2, with a high error rate on computers, presumably because the gesture is intended for touchscreens. Ortho zoom fares best overall, possibly due to the fine control of scale. Pinch fares the worst of all techniques on mobile, with participants making errors on 2.9 of 14 tasks, and we speculate that familiarity with a technique may have the side effect of premature navigation, whereas unfamiliar techniques may lead to more deliberate navigation with fewer errors.

6 DESIGN IMPLICATIONS SUMMARY

Target Visualization

From the number vs. timelines study results (Figure 4), we have two recommendations: First, long-distance timeline navigation benefits from visual aids. Whenever possible, designers should display information in groups that can be individually processed, as opposed to long numbers. This agrees with our hypothesis, H1. However, we also found a result counter to our hypothesis: For short distance tasks, we recommend not

displaying hierarchical target information as the additional complexity can slow down users. This affects timeline design, e.g., Knightlab’s timeline.js [25] always shows visual context, but this can slow down users.

Timeline Orientation

Drawing on our results in Figure 5, we realize that vertical timelines outperforming horizontal ones is counter to our hypothesis H2. We make two recommendations: First, we recommend maximizing timeline length by aligning it with the device’s long axis, especially on space-limited devices. For instance, we recommend vertical timelines for phones in portrait mode. Second, we recommend using an orientation that allows for an intuitive mapping between hardware and interaction. Specifically, we suggest not using *scroll wheel* pan for horizontal timelines, as this has a negative effect on the *brush* zoom navigation. Otherwise, orientation does not need to be a concern for timeline creators for computer.

Techniques

Based on the results of study 3 in Figure 6, we realize that standard techniques do perform best overall, agreeing with our hypothesis H3, but that *brush* and *ortho* zoom scale better than default techniques with increased task difficulty, and that *ortho* performed better than pinch zoom for high difficulty. We recommend use of *brush* and *ortho* zoom for far-distance navigation on both platforms.

Brush Zoom: *Brush* zoom’s incompatibility with *drag* pan is challenging, and needs to be paired with an additional technique to zoom out. In our study, we used *2x click* to zoom out. When the default pan method is replaced with *two-finger* panning, such as in our mobile study, it has a negative effect on performance. We recommend exploring other options to pair with *brush* zoom on mobile. For computer, replacing default pan with *scroll wheel* pan could lead to improved navigation, especially for close by targets. We recommend taking care when selecting the pan method.

Ortho Zoom: *Ortho* zoom shows potential and is the only method that has a negative trend on both platforms, meaning that it tends to fare better compared to the default method as the target distance increases. More research is needed to confirm whether this technique can significantly surpass the default methods for far targets, but this scaling, as well as the estimate for high difficulties being about on par with the default technique, are strong indicators that users will likely outperform the default technique with *ortho* zoom with more practice, especially for high difficulties. Hence, we recommend using *ortho* zoom for far distance navigation. *Ortho* zoom requires familiarity, so applications with many short-time users may prefer a default scroll zoom. However, applications with users that stay for at least 15 minutes, and

ideally longer, should consider using *ortho* zoom to enable users with experience to navigate faster.

Context Switching: The two best methods for high difficulty navigation on both platforms—*ortho* and *brush* zoom—both enable the user to make vast scale changes with a single interaction. This enables fast navigation, but can also have drawbacks. For example, fast context switches can create the need for users to spend time to re-orient themselves. We recommend taking care when designing context switches, and using a longer animation time. To further unlock the potential of techniques that require little interaction time, designers must aid users in visual navigation as much as possible, e.g., by providing visual cues to anchor the zoom operation, or by animating at a speed which reduces the need for data reorientation.

Novice Audiences: Some visualizations have novice users as audiences, and thus tend to use simple or familiar techniques such as *pinch* zoom. However, care must be taken to use this interaction method in long timelines with extensive manipulations: because it scales linearly with the physical motion of users' fingers, limited learning can take place for long-time users. To make this interaction more scalable, this core problem needs to be addressed, and options such as a non-linear mapping between finger motion and timeline scale change need to be considered.

7 DISCUSSION AND LIMITATIONS

Until now, there has been no evaluation-based guidance on how to design pan and zoom timelines for effective navigation on either computer or mobile. Given the large design space of visualization and interaction options, it is difficult to make an informed decision as a data visualization creator. Due to the diversity of hardware, contexts, and users to target with interactive timelines on the Web, creators have even less information about navigation effectiveness and efficiency.

Our work contributes the first design recommendations for timeline visualizations and interactions, on both mobile and desktop platforms. This was achieved by running a systematic in-the-wild study. Not controlling for input devices or contexts, as opposed to a laboratory setting, improves the applicability of our results as it is closer to how a timeline visualization would be accessed on the Web today. Future shifts in hardware use will limit the applicability of these results, as different input devices have individual characteristics. We believe that our results are applicable to different display sizes, such as tablets. For very large displays, other physiological factors may come into play and again limit applicability.

The results of our study most appropriately apply to novice users as our study participants only interacted with each timeline and method for under 15 minutes. This may have negatively impacted our results for uncommon and difficult interaction techniques such as *ortho* zoom. A preliminary analysis of user behaviors confirms that *ortho* zoom was primarily

used in small steps of zooming which do not fully use the potential of the technique. We anticipate that more experienced users will take effective advantage of this technique.

While many questions remain due to the large design space of pan and zoom timelines, we have found practical and useful answers to questions such as the effect of visual context for navigation, the role of orientation of timelines for mobile and computer environments, and have identified promising interaction techniques. We anticipate the results to generalize well to other timeline designs where the target is not clearly visible, and look forward to studies investigating pan and zoom timelines further to validate our results.

8 CONCLUSION

We have investigated navigation on pan and zoom timelines on desktop and mobile, and have identified design recommendations for efficient visualization and interaction between selecting dates and numbers, between horizontal and vertical timelines, and between interaction techniques.

We found that (1) the visual context from dates can be helpful for far-distance navigation, but harmful to short-distance navigation speed; (2) orienting timelines along the longer axis of devices improves navigation efficiency; (3) an orientation that allows for an intuitive mapping between hardware and interaction improves performance; (4) default techniques have the best navigation speed; (5) the choice of pan technique is important for overall performance; and (6) compared to default techniques, *2x click*, *hold*, and *rub* zoom seem to scale worse with task difficulty, whereas *brush* and especially *ortho* zoom seem to scale better. Given users' unfamiliarity with these techniques, this is a strong indicator that brush and especially *ortho* zoom may outperform the default technique with more practice, especially for high difficulties.

A large linear and non-linear design space of timeline visualizations and sliders still awaits evaluation. With this work, we aim to provide timeline visualization creators with more guidance for effective navigation. All implementations, data and data analysis routines used in this work are open source (<http://multiscale-timelines.ccs.neu.edu/>). Future research can build on this work for repeatable and reproducible user studies to improve navigating multi-scale data visualizations.

ACKNOWLEDGMENTS

We thank Alex Ahmed, Cody Dunne, Lulu Liu, Aditeya Pandey, Laura South, and Yixuan Zhang for their feedback on this paper. This work was supported in part by the College of Computer and Information Science and a Tier 1 seed grant at Northeastern University.

REFERENCES

- [1] Christopher Ahlberg and Ben Shneiderman. 1994. The Alphaslider: A Compact and Rapid Selector. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '94)*. ACM, New York, NY, USA, 365–371. <https://doi.org/10.1145/191666.191790>
- [2] Caroline Appert and Jean-Daniel Fekete. 2006. OrthoZoom Scroller: 1D Multi-scale Navigation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, New York, NY, USA, 21–30. <https://doi.org/10.1145/1124772.1124776>
- [3] Ravin Balakrishnan. 2004. "Beating" Fitts' Law: Virtual Enhancements for Pointing Facilitation. *Int. J. Hum.-Comput. Stud.* 61, 6 (Dec. 2004), 857–874. <https://doi.org/10.1016/j.ijhcs.2004.09.002>
- [4] Benjamin B. Bederson and James D. Hollan. 1994. Pad++: A Zooming Graphical Interface for Exploring Alternate Interface Physics. In *Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology (UIST '94)*. ACM, New York, NY, USA, 17–26. <https://doi.org/10.1145/192426.192435>
- [5] Matthew Brehmer, Bongshin Lee, Benjamin Bach, Nathalie Henry Riche, and Tamara Munzner. 2017. Timelines Revisited: A Design Space and Considerations for Expressive Storytelling. *IEEE Transactions on Visualization and Computer Graphics* 23, 9 (Sept 2017), 2151–2164. <https://doi.org/10.1109/TVCG.2016.2614803>
- [6] Matthew Brehmer, Bongshin Lee, Petra Isenberg, and Eun K. Choe. 2019. Visualizing Ranges over Time on Mobile Phones: A Task-Based Crowdsourced Evaluation. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (January 2019), 619–629. <https://doi.org/10.1109/TVCG.2018.2865234>
- [7] Matthew Brehmer and Tamara Munzner. 2013. A Multi-Level Typology of Abstract Visualization Tasks. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (Dec 2013), 2376–2385. <https://doi.org/10.1109/TVCG.2013.124>
- [8] Andy Cockburn, Amy Karlson, and Benjamin B. Bederson. 2009. A Review of Overview+Detail, Zooming, and Focus+Context Interfaces. *ACM Comput. Surv.* 41, 1, Article 2 (Jan. 2009), 31 pages. <https://doi.org/10.1145/1456650.1456652>
- [9] Raimund Dachselt and Markus Weiland. 2006. TimeZoom: A Flexible Detail and Context Timeline. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems (CHI EA '06)*. ACM, New York, NY, USA, 682–687. <https://doi.org/10.1145/1125451.1125590>
- [10] Leah Findlater, Joan Zhang, Jon E. Froehlich, and Karyn Moffatt. 2017. Differences in Crowdsourced vs. Lab-based Mobile and Desktop Input Performance Data. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 6813–6824. <https://doi.org/10.1145/3025453.3025820>
- [11] Paul M. Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology* 47, 6 (June 1954), 381–391. <https://doi.org/10.1037/0096-3445.121.3.262>
- [12] George W. Furnas and Benjamin B. Bederson. 1995. Space-scale Diagrams: Understanding Multiscale Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '95)*. ACM, New York, NY, USA, 234–241. <https://doi.org/10.1145/223904.223934>
- [13] George W. Furnas and Xiaolong Zhang. 1998. MuSE: A Multiscale Editor. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology (UIST '98)*. ACM, New York, NY, USA, 107–116. <https://doi.org/10.1145/288392.288579>
- [14] Krzysztof Z. Gajos, Katharina Reinecke, and Charles Herrmann. 2012. Accurate Measurements of Pointing Performance from In Situ Observations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 3157–3166. <https://doi.org/10.1145/2207676.2208733>
- [15] Yves Guiard and Michel Beaudouin-Lafon. 2004. Target acquisition in multiscale electronic worlds. *International Journal of Human-Computer Studies* 61, 6 (2004), 875–905. <https://doi.org/10.1016/j.ijhcs.2004.09.005>
- [16] Chris Harrison and Anind K. Dey. 2008. Lean and Zoom: Proximity-aware User Interface and Content Magnification. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. ACM, New York, NY, USA, 507–510. <https://doi.org/10.1145/1357054.1357135>
- [17] Ken Hinckley and Hyunyoung Song. 2011. Sensor Synaesthesia: Touch in Motion, and Motion in Touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 801–810. <https://doi.org/10.1145/1978942.1979059>
- [18] Kasper Hornbæk, Benjamin B. Bederson, and Catherine Plaisant. 2002. Navigation Patterns and Usability of Zoomable User Interfaces with and Without an Overview. *Transactions on Computer-Human Interaction* 9, 4 (Dec. 2002), 362–389. <https://doi.org/10.1145/586081.586086>
- [19] Takeo Igarashi and Ken Hinckley. 2000. Speed-dependent Automatic Zooming for Browsing Large Documents. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology (UIST '00)*. ACM, New York, NY, USA, 139–148. <https://doi.org/10.1145/354401.354435>
- [20] ScientiaMobile Inc. 2017. Smartphone vs Tablet Orientation: Who's Using What? <https://www.scientiamobile.com/page/smartphone-vs-tablet-orientation-whos-using-what>. Accessed: March 30, 2018.
- [21] Waqas Javed, Sohaib Ghani, and Niklas Elmqvist. 2012. GravNav: Using a Gravity Model for Multi-scale Navigation. In *Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI '12)*. ACM, New York, NY, USA, 217–224. <https://doi.org/10.1145/2254556.2254597>
- [22] Neel Joshi, Abhishek Kar, and Michael Cohen. 2012. Looking at You: Fused Gyro and Face Tracking for Viewing Large Imagery on Mobile Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2211–2220. <https://doi.org/10.1145/2207676.2208375>
- [23] Susanne Jul and George W. Furnas. 1998. Critical Zones in Desert Fog: Aids to Multiscale Navigation. In *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology (UIST '98)*. ACM, New York, NY, USA, 97–106. <https://doi.org/10.1145/288392.288578>
- [24] Myron W. Krueger, Thomas Gionfriddo, and Katrin Hinrichsen. 1985. VIDEOPPLACE - an Artificial Reality. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '85)*. ACM, New York, NY, USA, 35–40. <https://doi.org/10.1145/317456.317463>
- [25] Knight Lab. 2018. TimelineJS. <http://timeline.knightlab.com/>
- [26] I. Scott Mackenzie. 1989. A Note on the Information-Theoretic Basis for Fitts' Law. *Journal of Motor Behavior* 21, 3 (1989), 323–330. <https://doi.org/10.1080/00222895.1989.10735486>
- [27] Sylvain Malacria, Eric Lecolinet, and Yves Guiard. 2010. Clutch-free Panning and Integrated Pan-zoom Control on Touch-sensitive Surfaces: The Cyclostar Approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 2615–2624. <https://doi.org/10.1145/1753326.1753724>
- [28] Toshiyuki Masui, Kouichi Kashiwagi, and George R. Borden, IV. 1995. Elastic Graphical Interfaces to Precise Data Manipulation. In *Conference Companion on Human Factors in Computing Systems (CHI '95)*. ACM, New York, NY, USA, 143–144. <https://doi.org/10.1145/223355.223471>
- [29] George A. Miller. 1956. The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological review* 63, 2 (1956), 81–97. <https://doi.org/10.1037/h0043158>
- [30] Mathieu Nancel, Julie Wagner, Emmanuel Pietriga, Olivier Chapuis, and Wendy Mackay. 2011. Mid-air Pan-and-zoom on Wall-sized Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 177–186.

- <https://doi.org/10.1145/1978942.1978969>
- [31] Alex Olwal, Steven Feiner, and Susanna Heyman. 2008. Rubbing and Tapping for Precise and Rapid Selection on Touch-screen Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. ACM, New York, NY, USA, 295–304. <https://doi.org/10.1145/1357054.1357105>
- [32] Ken Perlin and David Fox. 1993. Pad: An Alternative Approach to the Computer Interface. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '93)*. ACM, New York, NY, USA, 57–64. <https://doi.org/10.1145/166117.166125>
- [33] William Playfair. 1786. *The Commercial and Political Atlas and Statistical Breviary*.
- [34] Stuart Pook, Eric Lecolinet, Guy Vaysseix, and Emmanuel Barillot. 2000. Context and Interaction in Zoomable User Interfaces. In *Proceedings of the Working Conference on Advanced Visual Interfaces (AVI '00)*. ACM, New York, NY, USA, 227–231. <https://doi.org/10.1145/345513.345323>
- [35] Philip Quinn, Sylvain Malacria, and Andy Cockburn. 2013. Touch Scrolling Transfer Functions. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13)*. ACM, New York, NY, USA, 61–70. <https://doi.org/10.1145/2501988.2501995>
- [36] Daniel Rosenberg and Anthony Grafton. 2010. *Cartographies of Time: A History of the Timeline*. Princeton Architectural Press.
- [37] Michail Schwab, James Tompkin, Jeff Huang, and Michelle A. Borkin. 2018. EasyPZ.js: Simple & Extendable Pan & Zoom JS Library for Mobile & Desktop. <https://doi.org/10.5281/zenodo.1137780> Instructions for use at <https://easypz.io>.
- [38] Martin Spindler, Martin Schuessler, Marcel Martsch, and Raimund Dachsel. 2014. Pinch-drag-flick vs. Spatial Input: Rethinking Zoom & Pan on Mobile Displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 1113–1122. <https://doi.org/10.1145/2556288.2557028>
- [39] Jarke J. Van Wijk and Wim A. A. Nuij. 2003. Smooth and Efficient Zooming and Panning. In *Proceedings of the Ninth Annual IEEE Conference on Information Visualization (INFOVIS'03)*. IEEE Computer Society, Washington, DC, USA, 15–22. <https://doi.org/10.1109/INFVIS.2003.1249004>
- [40] Colin Ware and Marlon Lewis. 1995. The DragMag Image Magnifier. In *Conference Companion on Human Factors in Computing Systems (CHI '95)*. ACM, New York, NY, USA, 407–408. <https://doi.org/10.1145/223355.223749>