# The Role of Latency and Task Complexity in Predicting Visual Search Behavior

Leilani Battle, R. Jordan Crouser, Audace Nakeshimana, Ananda Montoly, Remco Chang, and Michael Stonebraker

**Abstract**— Latency in a visualization system is widely believed to affect user behavior in measurable ways, such as requiring the user to wait for the visualization system to respond, leading to interruption of the analytic flow. While this effect is frequently observed and widely accepted, precisely how latency affects different analysis scenarios is less well understood. In this paper, we examine the role of latency in the context of visual search, an essential task in data foraging and exploration using visualization. We conduct a series of studies on Amazon Mechanical Turk and find that under certain conditions, latency is a statistically significant predictor of visual search behavior, which is consistent with previous studies. However, our results also suggest that task type, task complexity, and other factors can modulate the effect of latency, in some cases rendering latency statistically insignificant in predicting user behavior. This suggests a more nuanced view of the role of latency than previously reported. Building on these results and the findings of prior studies, we propose design guidelines for measuring and interpreting the effects of latency when evaluating performance on visual search tasks.

**Index Terms**—Visual search, latency, system response time, SRT

## 1 Introduction

Latency in interactive systems is inevitable. Often referred to as *system response time* (SRT), "latency" refers to the elapsed time between a user's input to a system and the time when the system produces a response. In the HCI community, the effect of latency is a well-studied topic. Dating back to work by Miller in 1968, research in SRT has long been driven by the need to identify requirements for responsive software [25]. Although the findings of SRT research are nuanced, design guidelines have nonetheless begun to emerge. In 1984, Shneiderman summarized the existing literature [31], and established 100 milliseconds as the maximum SRT for interactive interfaces.

When compared with the HCI community, latency research in the VIS community is still in its infancy. Although the goal of minimizing latency is commonly shared by visualization developers, there has been limited work in measuring the effect of latency to determine a design guideline similar to that from the HCI community. Notable exceptions include work by Liu & Heer [23] that found that a 500 millisecond latency in a visualization system can negatively affect an analyst's learned insights from exploring data. Zggragen et al. [35] replicated the experimental design by Liu & Heer, and although the authors also found a negative effect when latency is high, they observed no consistent differences between delays of 6 seconds and 12 seconds when using progressive and regular (blocking) visualizations in data exploration. Thus, when evaluating the impact of latency on a user's analytic flow, the objective threshold of when the latency is "too high" in a visualization system is still unclear.

While these disparate results might appear puzzling at first, the findings of SRT research from the HCI community might provide an explanation. As noted by Shneiderman [31] and Dabrowski & Munson [9], the effects of latency can be considered along two dimensions: user expectations and task complexity. When previous results are examined through the lens of task complexity, some of the findings begin to make sense. For example, the 100ms design guideline in HCI is largely based on empirical studies on control activations, such as simple tasks like clicking on a pull-down menu [9]. As task complexity increases, such as in the cases of solving matrix manipulation [14] and multi-parameter optimization using computer aided design (CAD) software [13], users can tolerate a significantly higher SRT to the point where latency no longer has an effect on user performance [14].

In this paper, we examine the role of task complexity when assessing the effect of latency in visualization systems. Our research questions are based on existing HCI and VIS research, but dig deeper into the influence of specific environmental factors within a controlled visualization environment:

1. How does **latency** at various scales affect users' performance?
2. How do changes in **task complexity** alter the influence of latency on user performance?

To answer these questions, we conducted a series of controlled experiments in which participants were asked to complete a visual search task – a common and well-studied task in the visualization community [34]. The relative simplicity of visual search makes it an ideal platform to study and understand the effect of latency. In visual search, a user's *performance* is often influenced by how they choose to conduct their search [7], or their search *strategy*. By studying how latency may influence the way a user performs their search, we can gain insight into how latency can also impact user performance.

In these experiments, participants were asked to find a target image in a pannable interface similar to Google Maps. Task complexity was controlled by providing participants with hints, which were designed to emulate realistic scenarios, such as prior knowledge of common data or image characteristics associated with certain visualizations, in much the same way that people exploit prior knowledge of mountain ranges and ocean coastlines when searching a geographic map [5].

In our first set of experiments, we treat latency as a binned categorical variable, consistent with treatments in prior work [23, 35]. In these experiments, we do find that latency has a statistically significant effect, but only at lower task complexity. These results suggest that as task complexity decreases, the effects of latency increase, which corroborates prior findings in the HCI literature. However, with latency coarsely binned, we are unable to pinpoint the threshold at which latency becomes a significant factor in user search strategies.

In order to identify this inflection point, we repeated the experiments for which latency had a significant effect, but this time treated latency as a *continuous variable*. Surprisingly, we find that latency actually has a more gradual effect than previously reported, which suggests that by treating latency as a coarsely binned categorical variable we may be oversimplifying the relationship of latency and task performance.

Finally, we aim to put these controlled experiment outcomes into a visualization context where visual search is often used. Specifically,

- *Leilani Battle is with the Department of Computer Science at the University of Maryland. E-mail: leilani@cs.umd.edu*
- *R. Jordan Crouser and Ananda Montoly are with the Department of Computer Science at Smith College. Emails: jcrouser@smith.edu and amontoly@smith.edu*
- *Remco Chang is with the Department of Computer Science at Tufts University. Email: remco@cs.tufts.edu*
- *Audace Nakeshimana and Michael Stonebraker are with the Electrical Engineering & Computer Science Department at MIT. Emails:audace@mit.edu and stonebraker@csail.mit.edu*

in visual search interfaces, users are often presented with an overview (small-world view) or with similar items clustered together (e.g. MDS overview). In our final experiment, we cluster the data to be searched and provide a cluster map to participants, to mimic scenarios observed in prior work where users leverage prior dataset knowledge and experience as "hints" for where to search [5]. Our results show that in a more realistic scenario, latency still has a statistically significant relationship with user search strategies, but this effect builds up gradually as latency increases rather than imposing a sudden shift. Furthermore, we see that in some situations, even when the effects of latency are significant, it may not be the most influential variable in predicting search behavior.

These results demonstrate that much like in broader HCI contexts, the effects of latency in visualization contexts can be subtle and interact with other environmental factors in complicated ways. For example, we find that decreasing task complexity can amplify the effects of latency. Furthermore, by considering latency as a continuous variable, we find a more nuanced relationship between latency and user search strategies that is not predicted by existing latency models for visualization.

## 2 Related Work

### 2.1 Latency in HCI Research

Research into the effect of latency / SRT in the HCI community can be categorized into two groups: (1) identifying the **limits** of SRT in the context of tasks and task complexity, and (2) psychological and cognitive factors that influence the **perception** of latency [9, 31].

#### Task and Maximum SRT

The notion that task type / complexity is an important factor when considering latency was first introduced by Miller in 1968 [25]. In this work, the author proposed 17 tasks along with corresponding design guidelines for SRT. Over the next 20 years, numerous studies examined these tasks in more detail. For example, Jota et al. investigated the effects of latency for touch interfaces [18], Tolia et al. for thin-clients [33], and Allison et al. for virtual reality environments [2].

Taken together, these studies resulted in the design guideline wherein 100ms is generally considered the upper limit of SRT in interactive systems. However, with few exceptions, the tasks examined in these studies are typically control activations of basic user interactions such as mouse clicks, keystrokes, or interactions with other graphical user interface (GUI) elements such as buttons, pull-down menus. Unfortunately, it is not clear how findings related to these simple actions would generalize to the more complex tasks inherent in the use of visualization systems such as search and data exploration.

In *Chapter 11: Quality of Service* of Shneiderman & Plaisant's book on *Designing the User Interface*, the authors note that *"response time should be appropriate to the task,"* and suggest that typing, cursor motion, and mouse selections should have SRTs in the range of 50-150ms [32]. For *"simple, frequent tasks, common tasks, and complex tasks"* the SRTs should be 1 second, 2-4 seconds, and 8-12 seconds, respectively. However, the authors do not provide definitions of what precisely constitutes a simple, common, or complex task. To the best of our knowledge, there does not exist a taxonomy of these complex tasks and their appropriate maximum SRTs. We provide our own definition for task complexity in Section 4.

#### Psychological and Cognitive Factors

Card, Robertson & Mackinlay investigated human perception thresholds for creating the illusion that a system runs instantaneously [8]. They determined that a maximum SRT of 100ms has to be maintained, otherwise the user will notice the delay. This 100 ms threshold of perceptual processing was later made popular by Nielsen [28, 29]. Seow further emphasized the importance of user expectations for establishing latency guidelines [30]. Users have certain expectations regarding the responsiveness of the system if a certain task is conducted. For instance, tasks that mimic events in the physical world with instantaneous responses (e.g., pressing a virtual button which mimics pressing a physical button) should yield instantaneous responses (e.g., an audible click). Doherty and Sorenson study how latency disrupts users' perception of immersive experiences (or flow) [10].

Previous work has also investigated how users' perceptions of latency may vary under different conditions. For example, according to Seow [30], tactile feedback after a virtual button press is very similar to the press of a real physical button, therefore the user expects an instantaneous response and might be more sensitive to interaction delays. When considering cognitive Load, previous work has observed that higher load may lead to lowered perception of latency [19, 27]. Other works also consider how users' perception of time can be manipulated [1, 15, 16, 20]. In some cases, the perception of latency is manipulated to ease the unpleasantness of waiting, referred to as benevolent deception by Adar et al. [1].

### 2.2 Latency in Visualization and Data Systems

Related to the research in HCI and operations research, researchers in the field of visualization have recently began to study the effect of latency in interactive data systems. Liu and Heer studied the effect of latency on user behavior and analysis outcomes in an exploratory data analysis task [23]. The primary finding of this study is that participants' behavior when latency exceeds 500ms differed from their behavior performing an identical task using a system with lower latency. However, consistent with HCI research that find task to be a factor in latency, Liu and Heer note that participants' tolerance for latency differed depending on the interaction type. In interactions such as brushing-and-linking, a delay higher than 100ms became noticeable. On the other hand, panning and zooming tasks were more robust to latency, with participants being tolerant to delays up to 1.5 seconds.

In contrast to traditional batch-based blocking systems where the user waits for a continuous period of time and receives the complete result from the system at once, progressive visualization systems provide immediate (but less accurate) information upon receiving a command from a user. In this domain, Fisher et al. [12] and Zgraggen et al. [35] independently observed that analysts preferred and performed with higher efficiency when using systems that provided immediate (and incremental) feedback [12]. Using these progressive systems, analysts can explore data faster and more efficiently when compared to using traditional blocking systems [35].

Many systems have also been developed to support exploratory visual analysis of large datasets, where reducing latency is a key concern in optimizing these systems. For example, imMens employs a specialized data structure called data tiles to support exploration of millions of data points in the browser [24]. ForeCache combines data tiles with predictive data pre-fetching techniques to further reduce latency [5]. Nanocubes are an alternative to data tiles designed to reduce latency when exploring large spatiotemporal datasets [22]. Rather than using specialized data structures or indexes, VizDom uses progressive sampling to produce fast results with low latency, and increasingly accurate results over time [35]. Falcon utilizes predictive aggregation techniques to reduce latency when performing specialized filtering operations over large datasets [26], where cross-filtering is a special case of dynamic query filters over a range of connected histograms.

## 3 Motivation and Research Questions

The diverse research in latency and wait-time mitigation in the HCI and visualization communities serves as the motivation to our work. These existing works suggest a complex relationship between a system's latency and its effects on the user. Unlike the simple "Powers of 10" model, the negative effects of latency do not appear to follow a simple logarithmic function, but are determined in part by the context of the task and the goals of the user. Our work adds to this growing body of literature by examining the effect of latency in a visual search task. We incorporate notions of task complexity in the design of a visualization interface to study how a user's search behavior can be predicted under controlled latency conditions.

A natural research question extending the observations of previous studies of latency in visualization tools [23, 35] is to expect that participants will react negatively to latency in visual search tasks. In particular, we believe that participants will shift their behavior to avoid latency, which leads us to our first research question:

**Research Question 1 (Q1):** *How does latency at various scales affect users' search strategies?*

In particular, we aim to investigate whether users' visual search strategies change in the presence of latency, and to characterize any changes observed, providing insight into the effects of latency on user performance. We will evaluate user strategies both in terms of task outcomes (i.e., whether the visual search was successful), as well as behavioral patterns (e.g., whether users experiencing higher latencies exhibit different search behaviors).

Furthermore, extrapolating from prior HCI research on latency suggests that we could see stronger effects from latency in visual search tasks with lower task complexity. We formulate the following research question regarding how differences in task complexity interact with latency in the visual search process:

**Research Question 2 (Q2):** *How do changes in task complexity alter the influence of latency on user strategy selection?*

Visual search is rarely done in a vacuum; it is generally part of a larger visualization process. As a user continues to search and navigate a given dataset, they become more experienced with this dataset, gaining an intuition for how the data is organized. The complexity of searching this dataset will presumably decrease as more experience is gained, which could alter how latency is perceived over time. We seek to study these effects in a controlled visual search environment.

## 4 Methodology: Pilot Studies

In pursuit of our research questions, we designed our experiments to allow us to explore the effects of latency in a simple visual search task when portions of the data incur substantial latency (i.e. 2500ms or greater), as well as to vary the latency incurred over a wider set of possible values than were observed in previous studies. We also seek to better understand the relationship between latency with task complexity in visualization contexts. Unfortunately, experimental variation of "task complexity" is highly subjective, and any findings would therefore be difficult to generalize. In order to minimize some of the variance introduced by this subjectivity, we elected to vary task complexity by observing the effects of latency not only at the beginning of the interaction, but at various points in the maturation of the visual search:

- **Phase 1 (Baseline)**: at the very beginning (when the user is orienting herself to the interface and data, and the visual search task is consequently most complex)
- **Phase 2 (Search Space Reduction)**: after the initial narrowing of the search space (when the user has gained her bearings, and task complexity has diminished somewhat)
- **Phase 3 (Proposed Locations)** and toward the end where the user is zeroing in on their target of interest (when completing the task is most straightforward)

We define (perceived) task complexity as the degree to which a user can rely on intuition gained through prior knowledge and experience to complete a given task. From this perspective, tasks for which users have no prior knowledge will have higher perceived task complexity. We conducted a series of three related experiments on Amazon Mechanical Turk. Workers with an HIT approval rate of 70+% and at least 50 approved HITs were recruited to participate. Each worker could only participate in one experiment, and exactly once. All experiments use a between-subjects design, and workers were paid up to $2.27 for completing any one of the three studies. The high-level design of these experiments are described in the following section, and additional detail is provided under the heading of each individual experiment.

### Task Design

The search tasks in our experiments were designed primarily to match existing tasks used to study visual search, while also incorporating elements from latency studies in exploratory visual analysis (or EVA) contexts. However, visual search and EVA behaviors are studied very differently, leading to the merging of disparate methodologies in our experiment design, as described below.

Liu and Heer [23] study the effects of latency by analyzing both unit-tasks (i.e., individual interactions) and interaction sequences (e.g., full sequences, transitions). Battle and Heer [6] observe that exploratory tasks vary significantly in open-ended-ness and level of abstraction, and argue that both low-level and high-level tasks should be considered when analyzing visual exploration contexts. Battle and Heer also analyze participant performance using both unit-task analysis and longer interaction sequences. Furthermore, realistic, high-level EVA tasks were an important component of the study designs for Liu and Heer [23] and Battle and Heer [6]; we consider the equivalent for visual search in this work. Given our focus on analyzing user strategies, we focus on sequence analysis rather than unit-tasks in our evaluation.

We sought representative visual search tasks known to be effective for evaluating user strategies and performance. Brown et al. [7] designed an effective abstraction for visual search tasks by having Amazon Mechanical Turk workers play the "Where's Waldo" game. In similar spirit, we seek to study abstract visual search tasks, where the larger image should be easy to partition and re-organize to modulate latency across tiles. Battle et al. [5] study how to reduce latency in real-world exploratory search contexts, combining experiment design elements from prior studies, including those by Brown et al. [7] and Liu and Heer [23]. Battle et al. asked participants to search for geographic regions matching specific visual characteristics (e.g., significant snow cover) within satellite imagery, which we simulate in our experiment design by asking participants to identify particular images within the collage. Furthermore, Battle et al. observe that many participants utilize prior knowledge of satellite imagery data to more quickly navigate and complete the tasks (e.g., searching for specific mountain ranges, and avoiding the oceans). We incorporate these insights into our experiment design through our modulation of task complexity.

### Implementation

Real-world visualization interfaces often contain components that could obscure or otherwise confound the effects of latency, such as formatting and interface configuration widgets. In order to isolate the effects of latency in a visual search task, our initial studies employ a simplified, browser-based visualization interface to reduce distractions. In each of the three experiments, the participant was asked to locate a target image within a 20-by-20 grid of images. Each image within the grid is represented as a distinct *tile*, simulating visualization tiles from existing visualization systems [5, 24].

To ensure that participants' behavior would not be biased by prior visualization experience, the visual stimuli consisted of benign images: the background consisted of images of birds, and the target was an image of a dinosaur that was roughly equivalent to the background with respect to visual salience (Fig. 1). Approximately 1% of the grid is visible through the experiment viewport at a time, and the user is permitted to pan left, right, up, and down. Zooming would have made it trivial in this case to identify the provided visual search targets, hindering our ability to control the complexity of the visual search tasks. As such, zooming was not enabled for these experiments. Targets were positioned such that at least 9 interactions were required to find them. Participants who performed too few interactions (e.g., only 2 interactions) were filtered out.

### Applying Latency

As the user pans around the grid in search of the target, individual tiles are made to incur a pre-specified amount of latency. The latency a participant experiences when a specific tile loads in the interface is defined as the time delay between when the participant finishes performing their panning interaction (i.e., *releases* the mouse drag), and when the image subsequently appears on the screen. For example, if a tile incurs a latency of 1,000ms, then the interface will wait until 1,000ms has elapsed before rendering the image on the screen. In these experiments, we expand the "binning" methodology similar to previous work [23], with maximum latency levels at 0ms, 2500ms, 7000ms, 10000ms, and 14000ms. We tested smaller latency measures in initial pilot studies (e.g., 500 milliseconds), but found no effects of latency for these cases. Thus we omit them from our analysis.

We aim to assess whether latency at various thresholds results in predictable differences in how users search the grid of images. In order to investigate this, we employ a small amount of benevolent deception in our experimental design: although the participant was only asked to locate a single target, **two target images** were actually present within the grid. To modulate latency, the participant is first randomly

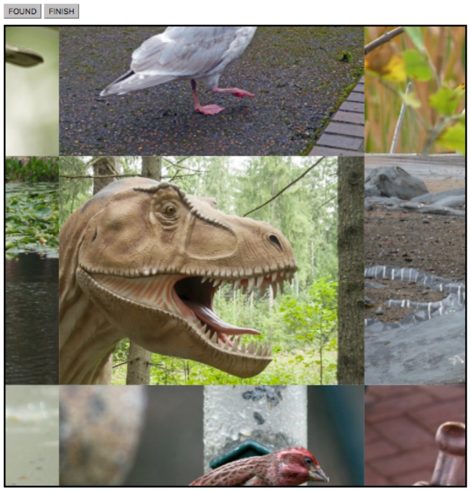
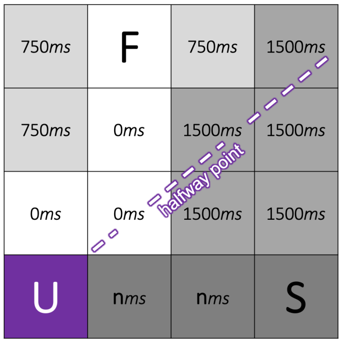


Fig. 1: A snapshot of the visual search interface implemented for our Amazon Mechanical Turk experiments. Participants explored a 20-by-20 grid of images (or *tiles*). Here, the target tile that participants are searching for (a dinosaur) is featured in the center of the viewport.



| | | | |
|---|---|---|---|
| 750*ms* | F | 750*ms* | 1500*ms* |
| 750*ms* | 0*ms* | 1500*ms* | 1500*ms* |
| 0*ms* | 0*ms* | 1500*ms* | 1500*ms* |
| U | n*ms* | n*ms* | S |

Fig. 2: Diagram showing how latency is assigned to tiles, denoted as individual squares. The **purple** square is the user's current location **U**. Tiles along a straight path from the user's current location to the low-latency ("fast") target **F** are assigned 0ms of latency. Tiles along a straight path to the high-latency ("slow") target **S** are assigned the maximum latency value (denoted here as *n*). Of the remaining tiles, those on the same side of the grid as the high-latency target, or that intersect with the halfway point, are assigned a latency of 1500ms; the remaining tiles are assigned a latency of 750ms.

assigned to one of the five latency conditions described above. Next, the two target images are randomly partitioned into a "low-latency" target and a "high-latency" target, which dictates the latency incurred when rendering each target and its surrounding tiles. In this way, we can assess whether people avoid higher latency areas by observing whether they locate the "low-latency" or "high-latency" target first. Having two targets allows us to observe how participants' search strategies influence which target they find. Search strategies are more difficult to reason about with only a single target, because all participants would then be forced to reach the exact same goal states, regardless of latency.

The intuition behind the latency modulation in this experimental design is that tiles along a straight path from the current viewport to the low-latency target appear faster (i.e., incur lower latency) than tiles along a straight path from the current viewport to the slow target. Figure 2 shows an example of how latency is applied to tiles in the interface. After each panning interaction, we draw an invisible line from the participant's current location in the grid to the location of the fast target, and a second line from this location to the slow target. Any tiles that intersect with the line to the fast target incur zero latency, appearing immediately after the user completes the panning interaction. Conversely, any tiles that intersect with the line to the slow target incur the maximum latency for the trial. In order to obscure the sharp contrast between the rendering of tiles along the low-latency trajectory and those along the high-latency trajectory, we also apply smaller amounts of latency to other tiles as follows. If a tile does not intersect with either of these lines, we instead look at whether they lie on the portion of the grid closer to the low-latency or high-latency target. If they are on the same side as the high-latency target, they incur a latency of 1500ms. Otherwise, they are assigned a latency value of 750ms.

However, as observed in relevant visual search systems like ForeCache [5] and Google Maps, individual tiles may appear at different times within the viewport of a visual search interface, due to latency. To mimic this uneven latency behavior, the latency in these experiments is *distributed* and *non-blocking*, in contrast to previous studies [23, 35]. Latency is applied to tiles individually, meaning that different tiles can have different latency values, and these tiles will appear at different times in the viewport. Participants can still perform their desired interactions, even when all tiles are not yet displayed in the viewport. In this case, participants may not see anything in certain parts of the viewport as they continue to explore.

In the following subsections, we will briefly describe the distinguishing characteristics of the three experiments and discuss the findings. In all cases, participants provided consent through a digital consent form, completed a demographics questionnaire, read the instructions for the visual search task, completed said search task using a browser-based visual exploration tool, and filled out a feedback survey.

### 4.1 Experiment 1: Baseline

This experiment aims to simulate the earliest stages of a visual search task. The participant is given no information regarding the target location, and they are left to orient themselves to the dataset. 103 of 111 participants successfully completed the task in this experiment, and they were distributed at random across the five latency conditions.

#### Results of Experiment 1: Baseline

We use Pearson's Chi-Squared test to assess whether users in different latency conditions tend to find the low-latency target first. We observed no statistically-significant relationship between these variables, $\chi^2(4, N = 101) = 2.373, p = 0.6675$. We therefore conclude that the incidence of finding the low-latency target first does not vary with latency in Experiment 1. See Table 1 for observed and expected values.

Table 1: Baseline Experiment

Observed Values

| | 0ms | 2500ms | 7000ms | 10000ms | 14000ms |
|---|---|---|---|---|---|
| Low-First | 11 | 12 | 14 | 13 | 16 |
| High-First | 8 | 9 | 6 | 7 | 5 |

Expected Values

| | | | | | |
|---|---|---|---|---|---|
| Low-First | 12.42 | 13.72 | 13.07 | 13.07 | 13.72 |
| High-First | 6.584 | 7.277 | 6.931 | 6.931 | 7.227 |

### 4.2 Experiment 2: Search Space Reduction

In our second experiment, we repeat the same general design as Experiment 1. However, in this case, we aim to simulate the scenario where the user gleaned a small amount of information on where to search next (i.e., the middle of a visual search process, where the search task becomes somewhat easier). As such, we make one small change to the experiment design for this experiment: we give the participants hints about the general location of the targets. Specifically, we tell the participants that the target is on the *left half of the grid*. In this case, the targets are positioned near the top left and bottom left corners of the grid, and we randomize which target is labeled as the fast target. 109 of 120 participants successfully completed the task in this experiment.

#### Results of Experiment 2

We again observed no statistically-significant relationship between these variables, $\chi^2(4, N = 104) = 3.055, p = 0.5487$. Therefore, we conclude that the incidence of finding the low-latency target first does not vary with latency in Experiment 2 (see Table 2 for details).

Table 2: Search Space Reduction Experiment

| | Observed Values | | | | |
|---|---|---|---|---|---|
| | 0ms | 2500ms | 7000ms | 10000ms | 14000ms |
| Low-First | 10 | 16 | 15 | 14 | 12 |
| High-First | 10 | 6 | 7 | 6 | 8 |
| | Expected Values | | | | |
| Low-First | 12.88 | 14.17 | 14.17 | 12.88 | 12.88 |
| High-First | 7.115 | 7.827 | 7.827 | 7.115 | 7.115 |

### 4.3 Experiment 3: Proposed Locations

In our third experiment, we again re-use the design of Experiment 1. However in this case, we aim to simulate the end of a visual search process, where the user now has sufficient knowledge of the dataset to be able to zero in quickly on the location of a particular visual search target. This effect was observed by Battle et al. [5], where participants (earth science researchers) had prior knowledge of the data being searched (satellite imagery), and exploited this knowledge to hone their searches of the underlying data (e.g., targeting specific mountain ranges). Note that this part of the visual search process is expected to be relatively easy, compared to Experiments 1 and 2. We make a small modification to the experiment design: we give users exact knowledge of two possible locations to find a target, but tell participants they are searching for a single target: the target can either be found along a direct line to their left, or else in a direct line to their right. In this way, the participant only makes a single decision: whether to pan to the left or to the right. Since either choice will end in success, this design enables us to gauge whether latency can influence their decision. 109 of 118 participants successfully completed the task in this experiment.

#### Results of Experiment 3

In contrast to the previous two conditions, we did observe a statistically-significant relationship between latency and the incidence of finding the low-latency target first, $\chi^2(4, N = 106) = 15.63, p = 0.003554$. See Table 3 for observed and expected values. Even though both choices (panning left or right) lead to success, more participants chose to search for the lower-latency target, suggesting that latency does indeed play a role in modulating search strategy when the task itself is less complex.

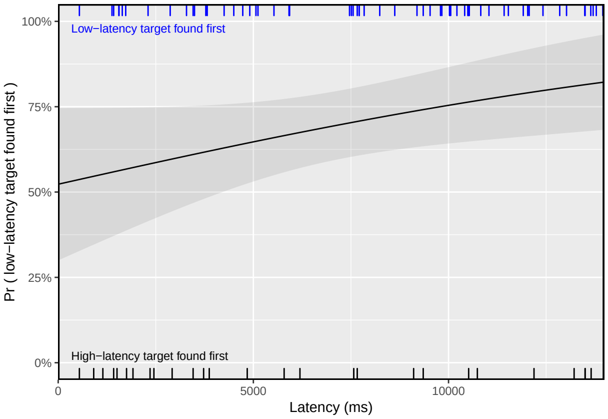
Table 3: Proposed Locations Experiment

| | Observed Values | | | | |
|---|---|---|---|---|---|
| | 0ms | 2500ms | 7000ms | 10000ms | 14000ms |
| Low-First | 10 | 12 | 19 | 18 | 19 |
| High-First | 11 | 8 | 4 | 4 | 1 |
| | Expected Values | | | | |
| Low-First | 15.45 | 14.72 | 16.92 | 16.19 | 14.72 |
| High-First | 5.547 | 5.283 | 6.075 | 5.811 | 5.283 |

### 4.4 Results Summary

In Experiment 1: Baseline, we found no statistically significant effect from latency, leading to our answer to research question **Q1**: *latency alone does not predict search strategy*. Similarly, we found no statistically significant effect from latency in Experiment 2: Search Space Reduction. However, we did find a statistically significant effect ($p < 0.01$) from latency in our third Proposed Locations experiment. When considering these experiments together, the results suggest that latency becomes more prominent in lower complexity visual search tasks, leading us to a tentative answer for **Q2**: *decreasing task complexity seems to amplify the effect of latency to the point of influencing users' search strategies and ultimately their performance.*

These results suggest that latency can influence search behavior in low complexity tasks, but do not tell us where latency shifts from being an insignificant effect to a significant one. In the case of prior work, latency is treated as coarse categorical variable, which does not allow one to determine empirically where latency appears to start having an effect. Therefore, to identify the inflection point for latency, we need to modify how latency is treated from categorical to continuous.



| | *Dependent variable:* |
|---|---|
| | **Low-Latency Target** found first |
| *(Intercept)* | 0.092 (−0.644, 0.827) |
| *latency* | $0.0001^{*}$ (0.00001, 0.0002) |
| Observations | 88 |
| *Note:* | $^{*}$p<0.1; $^{**}$p<0.05; $^{***}$p<0.01 |

Fig. 3: Results for a logistic regression analysis of the results for Experiment 3.2, with latency as the independent variable and the probability of finding the low-latency target first as the dependent variable. This relationship is significant only at the level of $p = 0.1$. When we view these results graphically, we observe the same weak relationship. In this diagram, **blue** tickmarks along the top of the figure represent trials in which the low-latency target was located first, and **black** tickmarks along the bottom of the figure represent trials in which the high-latency target was located first. The grey band around the logistic regression line represents the 95% confidence interval.

## 5 Deeper Analysis of Proposed Locations

In order to further investigate the relationship between latency and search behavior, we re-ran Experiment 3: Proposed Locations, but in version 3.2 treated latency as a *continuous* variable, rather than an ordered factor. Specifically, rather than randomly assigning each participant to one of five latency conditions (0ms, 2500ms, etc.), we instead sample a maximum latency value uniformly from the range [0ms,14,000ms] for each participant, and re-calculate intermediate latencies to be evenly spread between 0ms and the maximum. By drawing uniformly from a continuous range, we are then able to investigate how the probability of finding the low-latency target first varies with latency using logistic regression. 88 of 105 participants successfully completed the visual search task for this experiment.

### 5.1 Results: Logistic Regression Analysis

We find that roughly 30.7% of trials ended in participants finding the high-latency target first, and the remaining 69.3% ended in participants finding the low-latency target first. Thus, without factoring in latency effects, we observe that participants are roughly twice as likely to find the low-latency target first. Next, we built a logistic regression model with latency as the independent variable and the probability of finding the low-latency target first as the dependent variable. The results of this analysis are presented in Figure 3. We see that the relationship between latency and finding the low-latency target first is significant only at the level of $p < 0.1$. Moreover, we do not observe the characteristic S-shaped curve associated with logistic regression models. Instead, we observe at most a very gradual increase in the probability that participants will find the low-latency target first as latency increases.

We see that when latency is set to 0ms (i.e., no latency is applied to any of the tiles), the probability of finding the low-latency target first is roughly 50%. These results are consistent with our intuition: in the absence of additional latency, there is nothing to differentiate the nominal high-latency target from the nominal low-latency target, and so they appear as the first target found with roughly equal probability.

Table 4: Independent variables analyzed in Experiments 3.2 and 4.

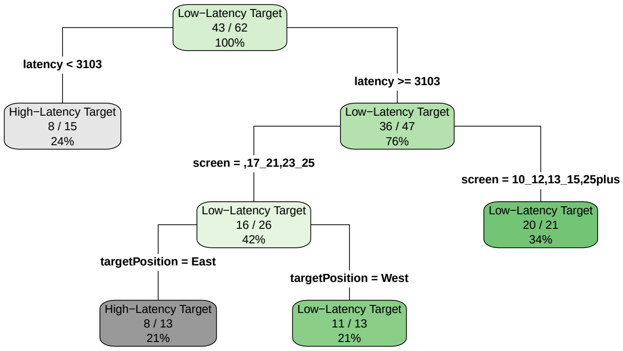
| Variable | Description |
|---:|---|
| *age* | Age: *18-25, 26-35, 36-45, 46-55, or 56+* |
| *cmpexperience* | Level of computer experience:<br>*novice, beginner, intermediate, or expert* |
| *cmphours* | Hours spent on the computer daily<br>*1-3, 3-5, 5-8, or 8+* |
| *cmptype* | Type of computer: *desktop* or *laptop* |
| *pubcmp* | Whether a *public* / *private* computer was used |
| *screen* | Screen dimensions |
| *totalInteractions* | Total interactions performed during the study |
| *latency* | Maximum latency |

As latency increases, we observe a modest increase in the probability of finding the low-latency target first, with a maximum probability of roughly 85%. However, the standard error bands remind us that this effect is weak, indicating that latency may not be the only factor at play. These results suggest that latency may have a gradual effect, rather than the binary effect observed in previous work (i.e., either latency had an effect, or not). We believe that the treatment of latency as a continuous rather than categorical variable may contribute to the differences in our results when compared to prior work: when only a small number of latency categories are evaluated, then latency may appear to be binary.

### 5.2 Results: Recursive Partitioning Analysis

Rather than trying to measure the effect of latency explicitly, we can also consider all factors measured in our experiment, and assess whether latency ranks highly compared to other variables in predicting the visual search behavior (i.e., whether the low-latency target is located first). To compute this ranking, we construct a decision tree using a simple recursive partitioning, where finding the low-latency target first is used as the class label. Table 4 describes the independent variables.

Each node within the decision tree represents a splitting of the remaining data into two sub-populations, based on a dichotomous variable, such as "`latency` $\geq$ `3103ms`". The recursive partitioning analysis selects these splits to maximize correct classification of the dependent variable (i.e., whether the low-latency target was located first). Thus, the most influential variables are generally selected first when constructing the tree. The closer a node is to the root, the more influential the corresponding independent variable is in predicting the outcome of the dependent variable. The model was trained on a random subset containing 70% of the data, and we performed 10-fold cross validation to select the optimal tuning parameter ($cp = 0.05263158$).



| | | Predicted Class | |
|---|---|---|---|
| | | High-First | Low-First |
| True Class | High-First | 6 | 2 |
| | Low-First | 4 | 14 |

Fig. 4: The resulting decision tree for Experiment 3.2. An initial split on $latency > 3103ms$ indicates that latency below this threshold had no discernible effect, classifying just 8 of 15 observations correctly. At higher levels of latency, additional factors such as screen size and target location provide greater predictive capacity. The confusion matrix indicates that this model achieves 76.9% accuracy on the test set.

The result of our recursive partitioning analysis is provided in Figure 4. Here, we see that the root node represents a split on the latency variable, where participants are divided into two groups based on whether the maximum latency is $\geq$ 3103ms. We see that other independent variables also seem to influence participant search behavior, such as screen size or the position of the target tiles within the grid (east or west). However, given that latency is the first node in the decision tree, latency appears to be the most influential variable in terms of predicting whether participants will find the low-latency target first. This model achieves 76.9% accuracy across the reserved test set.

### 5.3 Experiment 3.2: Summary

When revisiting our Proposed Locations experiment with maximum latency treated as a continuous variable, we still find an effect of latency on search behavior for low complexity tasks, however this effect has weak statistical significance ($p < 0.1$) when considered in isolation. In conjunction with additional factors, such as the total number of interactions or the location of the target, this predictive capacity is strengthened. Furthermore, the effects of latency appear to be more gradual, compared to the more binary effect of latency observed in other work, such as observed by Liu and Heer [23].

## 6 A More Realistic Task: Color Clusters

To better understand the effects of latency in more realistic visual search conditions, we updated the design of our Proposed Locations experiment, where we again treat latency as a continuous variable. We want to understand if this effect still exists in a more realistic visualization environment. In other words, we introduced a fourth condition in which each participant is presented with an interface that more closely resembles a real-world visual search environment.

Similar to Experiment 2: Search Space Reduction and Experiment 3: Proposed Locations, Experiment 4: Color Clusters provides hints to participants on where to search, thereby reducing the complexity of the visual search task, compared to Experiment 1: Baseline. The visual search interface remains the same as in previous experiments, but in Experiment 4 we organized the background images into four different color clusters based on the dominant color of the background of each bird image: **green** (e.g., forest, grass), **blue** (e.g., sky, water), **grey** background (e.g., rocks), or **brown** (e.g., soil). We then laid out these images in the grid such that images from the same color cluster were also positioned together in the grid (Fig. 5).

As mentioned in our task methodology, this data organization was designed to mimic well-known visual search scenarios, where certain regions are associated with specific color patterns that are frequently searched (e.g., targeting mountains when searching satellite imagery for snow cover [5]). One target was then embedded within each green region in this map. Participants were provided with this color map, and given a hint that they could find the target in one of the **green** zones. 103 of 121 participants successfully completed the experiment.

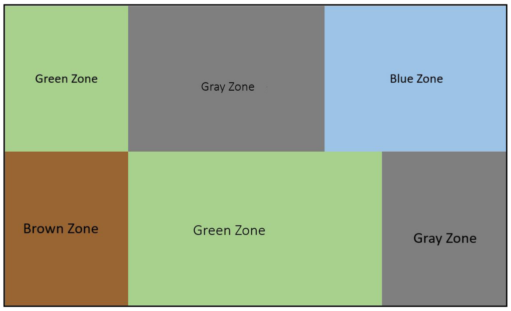


Fig. 5: The organization of background images for Experiment 4.

### 6.1 Results: Logistic Regression Analysis

We repeat our logistic regression analysis from Section 5.1 to evaluate the results of our Color Clusters experiment. We find that latency again has no statistically significant influence ($p = 0.0863$) in predicting whether a participant will find the low-latency target first. We visualize this relationship in Figure 6, where again we see a steady increase in the proportion of participants who find the low-latency target first as latency increases, but that the confidence interval surrounding this trend is quite wide. These results support the findings in Section 5.

### 6.2 Results: Recursive Partitioning Analysis

We also repeat our recursive partitioning analysis for Experiment 4, where we evaluate the same set of independent variables described in Section 5.2 (see Table 4). The model was again trained on a random training set containing 80% of the data, and we performed 10-fold cross validation to select the tuning parameter ($cp = 0.1414141$). In this analysis, we aim to assess the relative importance of latency compared to other variables when predicting whether participants will find the low-latency target first. Our results are provided in Figure 7.

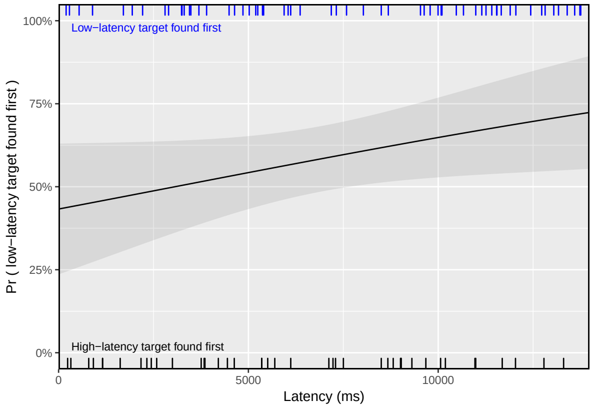
We find that the root node does not involve latency, and instead reflects the total interactions performed: "`totalInteractions` $\geq 50$". However, the child of this node (i.e., the next immediate node), does involve latency: "`latency` $\geq 9420$". In this case, we see that latency does play a role, but the total interactions a participant performs appear to have a stronger predictive capacity. Interestingly, we observed only a weak negative correlation between these two measures (Pearson's $r = -0.145$), and so it is unlikely that the total number of interactions is simply a proxy for latency. Furthermore, we find that latency seems to matter mainly at extremely high values, in this case over nine seconds of latency. These results differ greatly from that of prior work, for example Liu and Heer observe statistically significant effects at 500ms. The confusion matrix for this model appears at the bottom of Figure 7.

### 6.3 Experiment 4: Summary

We find that for a more realistic visual search scenario, latency in isolation again seems to have no statistically significant relationship to the outcome of a visual search task, even at very high levels. However, it does appear useful in predicting outcomes *in conjunction with other factors*, as illustrated by the recursive partitioning model.

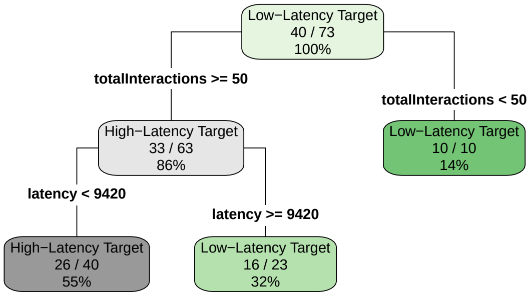
## 7 Search Strategy

The results of the previous experiment suggest that when considered as a continuous variable, latency has limited predictive capacity regarding the ultimate outcome of a visual search task (i.e. the likelihood that users navigate toward low-latency targets first). However, as has been discussed at length within both the VIS and CHI communities, the ultimate (binary) outcome of a task is a relatively blunt instrument for measuring the effects of a stimulus on human behavior within an interactive system. As such, we also wish to investigate the role of latency **during** the visual search task. For example, is latency useful in predicting the kinds of strategies people employ? One intuitive hypothesis could be that latency may have no clear effect on task outcome because users are able to quickly adapt to it, and to adjust their search strategy to circumvent the occurrence of high latency. If that is the case, are we able to observe these strategy switches?



| | *Dependent variable:* |
|---|---|
| | **Low-Latency Target** found first |
| *(Intercept)* | −0.270 (−0.930, 0.389) |
| *latency* | 0.0001$^{*}$ (0.00000, 0.0002) |
| *Observations: 103* | $^{*}$p<0.1; $^{**}$p<0.05; $^{***}$p<0.01 |

Fig. 6: Logistic regression analysis for Experiment 4, with latency as the independent variable and the probability of finding the low-latency target first as the dependent variable.



| | | Predicted Class | |
|---|---|---|---|
| | | High-First | Low-First |
| True Class | High-First | 7 | 3 |
| | Low-First | 9 | 11 |

Fig. 7: The resulting decision tree for Experiment 4, with maximum latencies sampled uniformly from the range [0ms, 14,000ms). An initial split on *totalInteractions* $>= 50$ indicates that in this modified experimental setup, latency was not the most important predictor of which target was located first. However, the confusion matrix indicates that this model achieves just 60% accuracy on the reserved test set.

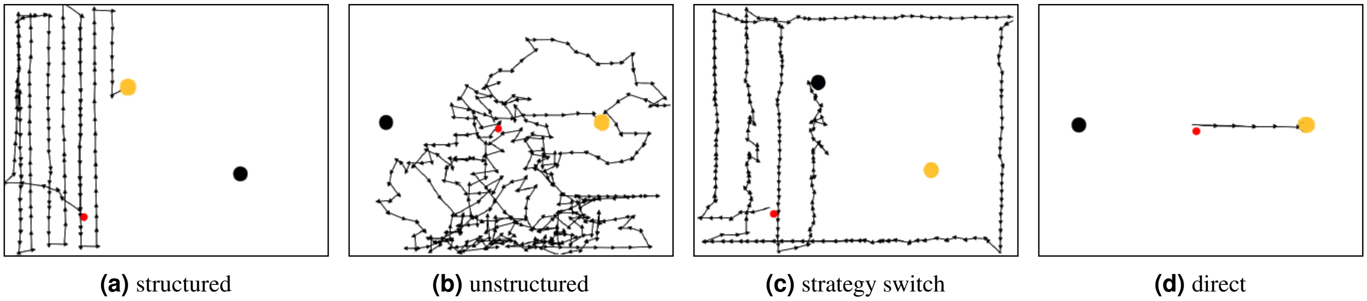
### 7.1 Methodology

In order to answer these questions, we first needed to identify the kinds of search strategies our participants employed when engaged in our visual search tasks. To do this, we first created a simple visualization of the user's trajectory through the collection of images. Our research team performed an initial clustering of these images, and identified four high-level groups of strategies: *structured, unstructured, strategy switch* and *direct* (see Fig. 8). To validate these groupings, we had three different raters independently re-classify the original images under those headings. This achieved an inter-rater agreement of 94.2% ($n = 191$), with discrepancies resolved by simple majority. We then added these strategy labels to the dataset as an additional variable.

### 7.2 Search Strategies in Experiments 1 through 3

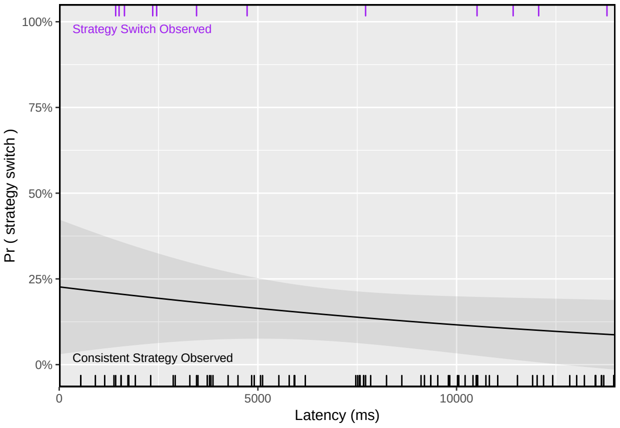
We begin by returning to the data collected in our preliminary experiments, treating latency as an ordinal variable. We combined the data from these three experiments, and determined using Pearson's Chi-Squared test that there is no relationship between strategy and whether or not the participant locates the fast target first, $\chi^2(3, N = 311) = 0.7287, p = 0.8664$, nor does there appear to be any significant relationship between strategy and latency, $\chi^2(12, N = 311) = 19.17, p = 0.08448$. However, further analysis suggests a strong relationship between strategy and the experimental condition to which the participant was assigned, $\chi^2(6, N = 311) = 129.5, p < 0.001$. An exploration of how the distribution of strategies varies among all of our experiments reveals some striking differences (Fig. 11).

For example, we notice that nearly all examples of the *direct* strategy were observed in Experiment 3. This is somewhat intuitive: when given proposed coordinates for the locations of the target images, participants are able to navigate directly to them. In contrast, participants in Experiment 1 had the highest incidence of *strategy switching*. This is again



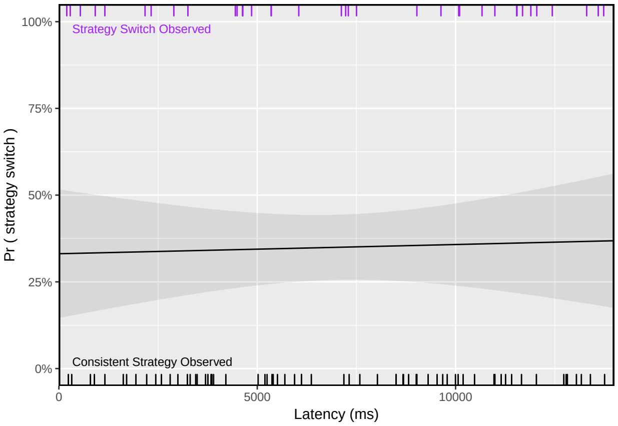
**(a)** structured **(b)** unstructured **(c)** strategy switch **(d)** direct

Fig. 8: Four high-level search strategies emerged during initial clustering of search path diagrams. The **red** dot in each diagram represents the participant's starting location in the collage. The **yellow** dot denotes the position of the fast (low-latency) target; the **black** dot the slow (high-latency) target. Arrows indicate the direction and length of each panning interaction.



| | *Dependent variable:* |
|---|---|
| | **Strategy Switch** observed |
| *(Intercept)* | $-1.227^{**}$ $(-2.149, -0.305)$ |
| *latency* | $-0.0001$ $(-0.0002, 0.00004)$ |
| *Observations: 88* | $^{*}$p<0.1; $^{**}$p<0.05; $^{***}$p<0.01 |

Fig. 9: Logistic regression analysis for Experiment 3.2, with latency as the independent variable and the probability of strategy switching as the dependent variable. Because the incidence is small (12 out of 88 cases), we employ Firth's method [11] rather than maximum likelihood.



| | *Dependent variable:* |
|---|---|
| | **Strategy Switch** observed |
| *(Intercept)* | $-0.703^{*}$ $(-1.391, -0.016)$ |
| *latency* | $0.00001$ $(-0.0001, 0.0001)$ |
| *Observations: 103* | $^{*}$p<0.1; $^{**}$p<0.05; $^{***}$p<0.01 |

Fig. 10: Logistic regression analysis for Experiment 4, with latency as the independent variable and the probability of strategy switching as the dependent variable. This model was fit using maximum likelihood.

intuitive: absent any guidance about the location of the target, many participants began with an inefficient, unstructured search and later switched to a more methodical, structured search.

### 7.3 Strategy Switching and Latency

Given the initial evidence that the effects of latency appear to be moderated at least in part by task difficulty, and the implication above that task difficulty has a strong relationship with strategy, we next explore the relationship between strategy and latency in Experiments 3.2 and 4, wherein latency was treated as a continuous variable. In particular, we focus on determining if there is an amount of latency that is sufficiently large such that strategy switching becomes more likely. If so, this might lend weight to the commonly-held belief that users do in fact alter their approach to solving a problem in order to avoid latency.

However, logistic regression indicates that this is not the case in either experiment (see Figs. 9 and 10). Contrary to our previous intuition, we observe no statistically significant interaction between latency and the incidence of strategy switching. Moreover, there is only minimal correlation between strategy switching and finding the fast target first (Pearson's $r = -0.09464851$), further suggesting that strategy switching is motivated by factors other than avoiding latency.

### 7.4 Summary of Search Strategy Analysis

We classified the strategies exhibited by participants into four different classes, and found no relationship between choice of strategy and participant performance. In cases where participants switched from one strategy to another, we evaluated whether this behavior may be influenced by latency, and found no statistically significant relationship. Finally, we observed notable shifts across experiments in the occurrence of each strategy. When evaluated empirically, we did find a statistically significant relationship between experiment condition and choice of search strategy ($p < 0.01$), suggesting that task complexity may influence a user's choice of visual search strategy. These findings are intuitive in relation to task complexity: the more information a user has about the dataset, the more efficient their search.

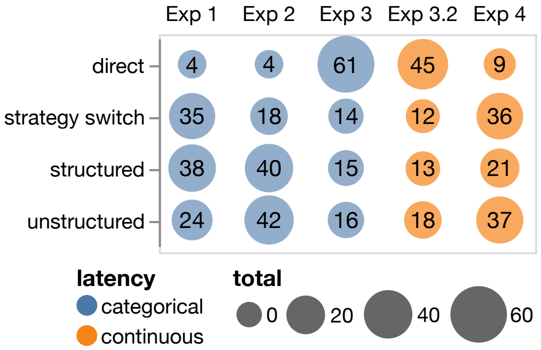


Fig. 11: Distribution of strategies across Experiments 1-4. The observed instances of each strategy appear in the corresponding bubble beneath each experiment. Bubbles are scaled by total observations, and color encodes how latency was treated (categorical or continuous).

## 8 Discussion

In this section, we highlight key takeaways from our results, discuss limitations to our experiments, and identify avenues for future work.

### 8.1 The Complicated Effects of Latency

As evidenced by our results, latency does not appear to be a clear-cut predictor for visual search outcomes; the effects of latency seem more subtle and complex than previously understood, at least for visual search. When we treat latency as categorical, and use existing methods to bin latency into a few factors, we do observe a statistically significant effect. However, latency is not well represented as a categorical variable, and is often one of several factors that can affect user strategy and thus user performance. When we augment our experiments to take these factors into account, latency's measured effect is significantly weaker. Thus, our ability to accurately assess latency's effect on users may be highly dependent on how the latency is modeled (e.g., categorical or continuous), as well as whether other environmental factors are considered (e.g., data layout, task complexity).

To further investigate whether latency may affect a user's visual search strategy, we visualize and analyze user sessions to identify the different ways in which people approach a simple visual search task. We identify three general search strategies, and some cases where users switch between different strategies. We do find that in the case of lower task complexity, participants tend to favor search paths with low latency. However, participants' overall strategy (i.e., panning directly to the target) does not change, regardless of latency. Furthermore, we find this behavior reflected across all of our experiments: latency appears to be less important when choosing a search strategy (i.e., selecting among the categories observed in Section 7). Moreover, choice of strategy does not seem to affect visual search performance.

**Takeaway # 1**: *Latency may not have a strong influence on visual search outcomes, even when the latency is several seconds long.* This challenges the notion that latency is universally detrimental. It may therefore be beneficial to focus on additional factors when optimizing visual search tools [6, 23].

### 8.2 Comparing with Prior Studies

Our findings are consistent with prior HCI research on latency [9, 31], which find that other factors like task complexity [31] can alter a users' perception of latency. However, our results differ significantly from observations made in recent visualization studies, where latency as low as 500ms can significantly affect user performance [23, 35]. This discrepancy is due in part to differences in study methodology. For example, our work integrates methodology from prior studies in two disparate sub-areas, discussed in Section 4: performance in visual search (e.g., [5, 7]), and latency in exploratory visual analysis (e.g., [6, 23]). Thus, some of our results may not translate directly to prior latency studies, which focus on exploratory contexts and not visual search. However, we also believe these differences occur because of how latency was modulated in past experiments. By treating latency as continuous instead of categorical, we are able to model more nuanced relationships between latency and visual search strategy that cannot not be predicted by these existing latency models.

**Takeaway # 2**: *Models of system performance (e.g., latency models) should strive for realism over simplicity.* In similar spirit to the findings of Battle and Heer [6], we find that understanding the context is critical when making assumptions about the performance of visualization systems, otherwise critical details may be missed.

### 8.3 Limitations and Future Work

While the experiment design and analysis methods were informed by prior studies [5, 7, 23, 35], one limitation of this work is the use of a highly simplified interface. For example, our study only considered panning interactions, and did not include zooming interactions. This choice could affect the generalization of our results to other visualization tools used frequently in the real world. However, the interactions between latency and different UI operations (e.g., panning, zooming, brushing and linking) are known to be complex [23], likely warranting a separate study of their own. We see our experiments as a useful starting point, providing a baseline for more complex studies of latency, interaction and visual search in the future.

Another limitation is that participants' mental models of latency were not modulated. As such, participants may have thought the latency was random, and therefore ignored it. Despite this limitation, we still observe participants avoiding latency in cases of low task complexity (Experiment 3, Section 4.3), suggesting that even without prompting, users do notice latency under certain conditions, and may alter their behavior in response. Conducting a study where users are explicitly made aware of latency would be an interesting direction for future work.

A possible limitation is the use of a fully abstracted dataset. Though we chose this design to mitigate variance in participants' dataset expertise, Experiments 1-3 may be difficult to reason about with respect to real-world visualization environments. However, real-world visualization tools have many features and interaction widgets that can confound the results of a controlled study, such as formatting and configuration.

Furthermore, though some aspects may be considered less realistic for visual search, we carefully preserved specific environmental factors in our experiments that are very common in large-scale visual search interfaces. For example, the grid succinctly captures the tile structure and mechanics of existing large-scale visual search systems like Google maps, imMens [24] and ForeCache [5]. We also aimed to address these issues through Experiment 4, where we organize the image data to approximate existing visual search scenarios (e.g., [5]). In the future, we plan to apply our experimental design to relevant tools for visual search, and compare the resulting performance with our current results.

**Takeaway # 3**: *How to best decompose high-level visualization tasks for precise, rigorous study warrants further investigation.* As observed in this work and others [5–7, 23], paring down complex tasks like visual search and exploratory visual analysis is notoriously difficult. More fundamental and rigorous characterizations of these tasks are needed to effectively decompose them for in-depth study.

## 9 Conclusion

It is widely believed that latency in user interfaces can have a significant effect on user performance. In the case of visualization systems, when the result of an interaction is delayed due to high latencies, the user's analytic flow may be interrupted as a result. Though recent studies corroborate such an effect (e.g., [23]), the degree to which latency affects user performance in relation to other factors is less well understood. In this paper, we investigate the relationship between latency, task complexity and user performance. We focus our analysis on the context of visual search, a core sub-task for visual exploration use cases. Through a series of studies on Amazon Mechanical Turk, we find that latency does impact user performance, but only in certain situations: latency appears to only have a statistically significant effect in low-complexity visual search tasks, and does not always seem to affect users' choice of search strategy. Furthermore, other factors may be stronger predictors of users' visual search outcomes, such as the total interactions performed. Our results provide a more nuanced view of the role that latency plays in visual search contexts, which was not predicted by existing visualization latency models. Using the results of both our experiments and prior work, we propose new guidelines for assessing and inferring the role of latency in evaluating visual search systems.

## Acknowledgements

We thank Danyel Fisher, David Karger, Dylan Cashman, and our reviewers for their helpful feedback. This project was supported in part by NSF award IIS-1452977, DARPA awards FA8750-17-2-0107 and FA8750-19-C-0002, and the Intel Science and Technology Center (ISTC) for Big Data.

## Open Access Statement

We are committed to the free and open sharing of experimental data and analyses for the benefit of the broader research community [3, 4, 17, 21]. All datasets, analyses, and generated figures are freely available online:

`https://github.com/leibatt/latency-visual-search`

## References

[1] E. Adar, D. S. Tan, and J. Teevan. Benevolent deception in human computer interaction. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1863–1872. ACM, 2013.

[2] R. S. Allison, L. R. Harris, M. Jenkin, U. Jasiobedzka, and J. E. Zacher. Tolerance of temporal delay in virtual environments. In *Proceedings IEEE Virtual Reality 2001*, pp. 247–254. IEEE, 2001.

[3] L. Battle, M. Angelini, C. Binnig, T. Catarci, P. Eichmann, J.-D. Fekete, G. Santucci, M. Sedlmair, and W. Willett. Evaluating visual data analysis systems: A discussion report. In *Proceedings of the Workshop on Human-In-the-Loop Data Analytics*, p. 4. ACM, 2018.

[4] L. Battle, R. Chang, J. Heer, and M. Stonebraker. Position statement: The case for a visualization performance benchmark. In *2017 IEEE Workshop on Data Systems for Interactive Analysis (DSIA)*, pp. 1–5. IEEE, 2017.

[5] L. Battle, R. Chang, and M. Stonebraker. Dynamic prefetching of data tiles for interactive visualization. In *Proceedings of the 2016 International Conference on Management of Data*, pp. 1363–1375. ACM, 2016.

[6] L. Battle and J. Heer. Characterizing Exploratory Visual Analysis: A Literature Review and Evaluation of Analytic Provenance in Tableau. *Computer Graphics Forum*, 38(3), 2019.

[7] E. T. Brown, A. Ottley, H. Zhao, Q. Lin, R. Souvenir, A. Endert, and R. Chang. Finding waldo: Learning about users from their interactions. *IEEE Transactions on visualization and computer graphics*, 20(12):1663–1672, 2014.

[8] S. K. Card, G. G. Robertson, and J. D. Mackinlay. The information visualizer, an information workspace. In *Proceedings of the SIGCHI Conference on Human factors in computing systems*, pp. 181–186. ACM, 1991.

[9] J. Dabrowski and E. V. Munson. 40 years of searching for the best computer system response time. *Interacting with Computers*, 23(5):555–564, 2011.

[10] R. A. Doherty and P. Sorenson. Keeping users in the flow: mapping system responsiveness with user experience. *Procedia Manufacturing*, 3:4384–4391, 2015.

[11] D. Firth. Bias reduction of maximum likelihood estimates. *Biometrika*, 80(1):27–38, 1993.

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

[13] T. Goodman and R. Spence. The effect of system response time on interactive computer aided problem solving. In *ACM SIGGRAPH Computer Graphics*, vol. 12, pp. 100–104. ACM, 1978.

[14] M. Grossberg, R. A. Wiesen, and D. B. Yntema. An experiment on problem solving with delayed computer responses. *IEEE Transactions on Systems, Man, and Cybernetics*, (3):219–222, 1976.

[15] C. Harrison, B. Amento, S. Kuznetsov, and R. Bell. Rethinking the progress bar. In *Proceedings of the 20th annual ACM symposium on User interface software and technology*, pp. 115–118. ACM, 2007.

[16] C. Harrison, Z. Yeo, and S. E. Hudson. Faster progress bars: manipulating perceived duration with visual augmentations. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1545–1548. ACM, 2010.

[17] L. Harrison, F. Yang, S. Franconeri, and R. Chang. Ranking visualizations of correlation using weber's law. *IEEE transactions on visualization and computer graphics*, 20(12):1943–1952, 2014.

[18] R. Jota, A. Ng, P. Dietz, and D. Wigdor. How fast is fast enough?: a study of the effects of latency in direct-touch pointing tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2291–2300. ACM, 2013.

[19] T. Kaaresoja, S. Brewster, and V. Lantz. Towards the temporally perfect virtual button: touch-feedback simultaneity and perceived quality in mobile touchscreen press interactions. *ACM Transactions on Applied Perception (TAP)*, 11(2):9, 2014.

[20] K. L. Katz, B. M. Larson, and R. C. Larson. Prescription for the waiting-in-line blues: Entertain, enlighten, and engage. *MIT Sloan Management Review*, 32(2):44, 1991.

[21] M. Kay and J. Heer. Beyond weber's law: A second look at ranking visualizations of correlation. *IEEE transactions on visualization and computer graphics*, 22(1):469–478, 2016.

[22] L. Lins, J. T. Klosowski, and C. Scheidegger. Nanocubes for real-time exploration of spatiotemporal datasets. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2456–2465, 2013.

[23] Z. Liu and J. Heer. The effects of interactive latency on exploratory visual analysis. *IEEE transactions on visualization and computer graphics*, 20(12):2122–2131, 2014.

[24] Z. Liu, B. Jiang, and J. Heer. immens: Real-time visual querying of big data. In *Computer Graphics Forum*, vol. 32, pp. 421–430. Wiley Online Library, 2013.

[25] R. B. Miller. Response time in man-computer conversational transactions. In *AFIPS Fall Joint Computing Conference (1)*, pp. 267–277, 1968.

[26] D. Moritz, B. Howe, and J. Heer. Falcon: Balancing interactive latency and resolution sensitivity for scalable linked visualizations. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pp. 694:1–694:11. ACM, New York, NY, USA, 2019. doi: 10.1145/3290605.3300924

[27] A. Ng, M. Annett, P. Dietz, A. Gupta, and W. F. Bischof. In the blink of an eye: investigating latency perception during stylus interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1103–1112. ACM, 2014.

[28] J. Nielsen. Usability inspection methods. In *Conference companion on Human factors in computing systems*, pp. 413–414. ACM, 1994.

[29] J. Nielsen. Powers of 10: Time scales in user experience. *Retrieved January*, 5:2015, 2009.

[30] S. C. Seow. *Designing and Engineering Time: The Psychology of Time Perception in Software (Adobe Reader)*. Addison-Wesley Professional, 2008.

[31] B. Shneiderman. Response time and display rate in human performance with computers. *ACM Computing Surveys (CSUR)*, 16(3):265–285, 1984.

[32] B. Shneiderman. *Designing the user interface: strategies for effective human-computer interaction*. Pearson Education India, 2010.

[33] N. Tolia, D. G. Andersen, and M. Satyanarayanan. Quantifying interactive user experience on thin clients. *Computer*, (3):46–52, 2006.

[34] C. Ware. *Visual thinking for Design*. Morgan Kaufmann, 2008.

[35] E. Zgraggen, A. Galakatos, A. Crotty, J.-D. Fekete, and T. Kraska. How progressive visualizations affect exploratory analysis. *IEEE transactions on visualization and computer graphics*, 23(8):1977–1987, 2017.