The Not-so-Staggering Effect of Staggered Animated Transitions on Visual Tracking

Fanny Chevalier, Pierre Dragicevic and Steven Franconeri

Abstract—Interactive visual applications often rely on animation to transition from one display state to another. There are multiple animation techniques to choose from, and it is not always clear which should produce the best visual correspondences between display elements. One major factor is whether the animation relies on staggering—an incremental delay in start times across the moving elements. It has been suggested that staggering may reduce occlusion, while also reducing display complexity and producing less overwhelming animations, though no empirical evidence has demonstrated these advantages. Work in perceptual psychology does show that reducing occlusion, and reducing inter-object proximity (*crowding*) more generally, improves performance in multiple object tracking. We ran simulations confirming that staggering can in some cases reduce crowding in animated transitions involving dot clouds (as found in, e.g., animated 2D scatterplots). We empirically evaluated the effect of two staggering techniques on tracking tasks, focusing on cases that should most favour staggering. We found that introducing staggering has a negligible, or even negative, impact on multiple object tracking performance. The potential benefits of staggering may be outweighed by strong costs: a loss of common-motion grouping information about which objects travel in similar paths, and less predictability about when any specific object would begin to move. Staggering may be beneficial in some conditions, but they have yet to be demonstrated. The present results are a significant step toward a better understanding of animation pacing, and provide direction for further research.

Index Terms—Animated transitions, staggered animation, visual tracking

1 Introduction

The purpose of an interactive visualization is to allow analysts to view a dataset from multiple perspectives. When these perspectives change, it is critical that the observer understands how the data in a new display corresponds to data from a display that has now vanished. For example, when switching between two different 2D scatterplots of a multi-dimensional dataset, the analyst needs to understand how data points map from the old projection to the new one [18]. When displays undergo large changes in an abrupt manner, understanding changes can be extremely difficult [39]. A common solution is to smoothly animate the transition between displays, allowing elements to gradually move from one position or visual state to the other, and thus allowing the observer to perceive the correspondence between displays.

Animated transitions have been shown to yield a number of advantages, such as helping users keep oriented during pan and zoom operations [6, 37] and helping them understand changes between different visual representations of statistical data [24]. But little is known on how to best design animated transitions. There are multiple animation techniques to choose from, and it is not always clear which should produce the best visual correspondence between display elements. For example, elements can be set to follow different trajectories, or move at different times and/or accelerate or slow down (i.e., animation pacing). Staggering is one animation pacing technique: rather than having all the elements move at the same time, it introduces an incremental delay in start and stop times. Several visual applications have adopted this design, including Pivot [1], DynaVis [24] and Histomages [11]. Staggering has been suggested to help reduce inter-elements occlusion and to provide less overwhelming visual transitions [24]. However, there is no empirical evidence for these purported advantages.

One way to put this question to the test is to determine whether or not staggering makes it easier for users to visually track elements from one display state to the other. Researchers in perceptual psychology have extensively studied the processing limitations of the human visual system [20] and more specifically, the mechanisms involved in the tracking of multiple objects over time. Past work indicates that inter-object proximity (i.e., *crowding*) has an influence on visual tracking performance [19]. These results align with the intuition that an animated transition technique that reduces crowding may have a facilitating effect. Yet, many questions remain to be addressed. One question is whether staggering does reduce crowding in the first place. Furthermore, the effects of crowding have been generally observed on animations where all objects move at the same time, and the potential benefits of staggering may be outweighed by strong costs associated with non-simultaneous object motions.

This article attempts to move closer to a better understanding of animated transition design by examining the effect of two types of staggered animation on multiple object tracking performance. We ran a series of simulations and controlled experiments on randomly moving dots as found in, e.g., 2D scatterplot transitions. These experiments address the following questions: *i)* which animation characteristics (crowding or otherwise) affect the difficulty of tracking multiple objects? *ii)* can these characteristics be manipulated using staggering? and *iii)* provided this is the case, will staggering yield observable benefits on object tracking performance, or will these benefits be outweighed by the costs associated with non-simultaneous object motions? We first present an overview of related work, then the general design rationale for our study. We then report on our experiments, and finally conclude with general implications for the design of animated transitions in interactive information visualization systems.

2 Background

Animations consist of creating the illusion of continuous visual changes through the rapid display of a sequence of static images. Partly building on the tradition of cartoon animations [9, 28], animations are widely used in graphical user interfaces for helping users understand dynamic processes and time-varying data, or simply for their compelling value. Many benefits of animations have been pointed out [5], as well as a certain number of pitfalls [40]. Despite their popularity, animations are still poorly understood. Here we discuss animated transitions, i.e., a particular class of computer animations whose purpose is to turn abrupt visual changes into smooth ones. We discuss prior work on animated transitions both in general human-computer interaction (HCI) and in information visualization (infovis). We also discuss how previous work in perceptual psychology has attempted to shed light on how we apprehend changes in dynamic displays.

- *Fanny Chevalier is with Inria. E-mail: fanny.chevalier@inria.fr*
- *Pierre Dragicevic is with Inria. E-mail: pierre.dragicevic@inria.fr*
- *Steven Franconeri is with Northwestern University. E-mail: franconeri@northwestern.edu*

Manuscript received 31 Mar. 2014; accepted 1 Aug. 2014. Date of publication 11 Aug. 2014; date of current version 9 Nov. 2014.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.
Digital Object Identifier 10.1109/TVCG.2014.2346424

2.1 Animated Transitions

Animated transitions have been used in a variety of interactive applications for a variety of reasons, including for facilitating the use of zoomable user interfaces [37], to aid the tracking of changes in dynamic data [4, 10, 38], and to facilitate the understanding of transitions between different representations of the same data [7, 11, 17, 18, 24].

Although prior work has demonstrated the benefits of animated transitions for particular tasks, this work has mainly focused on particular instances of animated transitions. There are many possible designs for animations, and with a few exceptions [15, 24], there exist very little empirical research comparing the effectiveness of different types of designs. A better understanding of good animated transition design is particularly important for interactive information visualization applications, where the analyst needs to constantly navigate between different views of the same data. Understanding how views relate to each other is an integral part of the visual exploration activity.

A few general design recommendations have been proposed for animations [40], such as *apprehension* – the information conveyed by the animation should be accurately perceived – and *congruence* – the animation should be predictable and understandable. Other general recommendations have been derived from cartoon animation principles [9, 28]. Many of these recommendations are too general to effectively guide the design of animated transitions. While some animations can be authored manually, an animated transition technique must produce an effective and meaningful sequence of frames from *any* pair of initial and final frames. For developers of information visualization applications, animated transitions are functions with a multitude of free parameters (e.g, the duration, the temporal pacing, the interpolation function between positions, etc.), and they need operational recommendations as to which parameter to choose and when. Only a few studies have attempted to empirically investigate these [15, 24, 35].

Animated transitions are generally characterized by spatial and temporal parameters, i.e., parameters that involve object trajectories and parameters that involve object speed or pacing. One important aspect of temporal pacing is whether all the display elements move at the same time, or if there is delays in motion start times. For example, a transition can be broken down into different *stages* based on the structure of the data (e.g., delete first, then move, then add [31]. Staged animations can be valuable when the inherent structure of the data allows for the definition of meanigful stages, but their benefit is not systematic [24]. The sequencing the transition into several stages can also dramatically increase the animation duration.

Another pacing effect is *staggering*, where the start time of elements is delayed incrementally [24]. Staggering is commonly found in visual applications [1, 11] and generally preferred over other animations [24]. It has been suggested that staggering may reduce occlusion, while also reducing display complexity and producing less overwhelming animations. However, no study has explicitely demonstrated these advantages. This work attempts to address this lack.

2.2 Studies on Visual Tracking

In many animated transitions, individual objects (points, bars, pixels, icons, ...) shift their position over time, requiring the observer to understand which points moved to which new locations. A substantial perceptual psychology literature examines how well the visual system can deal with such animations, testing how many objects can be successfully tracked under different conditions. These tests rely on Multiple Object Tracking (MOT) tasks, which require observers to mentally track a set of target objects moving among distractor objects [33]. The challenge is similar to tracking a single target cup as a street magician shuffles it rapidly among identical distractor cups but here there are multiple target cups. In a typical task, a set of circles are presented on a display, and a subset is briefly cued by a color change. All of the objects then move randomly across the screen for several seconds. During this phase, all objects are visually identical in color and shape, in order to ensure that the task tests an observer's ability to track via object location alone. When the objects stop, the observer must click on the original highlighted circles, and experimenters examine their accuracy depending on display conditions.

The results from such studies reveal a clear set of conditions affecting performance. Perhaps the most important factor is crowding, the spacing among objects [19]. When objects become crowded, targets and distractors can become confused with each other [30]. These results also reveal that people can track a maximum of 7-8 objects under carefully controlled conditions, though this limit appears to drop to 3-4 objects in more general cases [19]. There are also factors that surprisingly do not seem to limit tracking performance. Object speed only appears to impair performance at extremes - the types of speeds that strain the refresh rate of typical monitors [19]. Trajectory changes and curved paths appear to have only minimal impacts on performance [23]. Extreme magnification changes of the display do not have a substantial impact [22]. Object occlusion is surprisingly undisruptive when the occluding surface is clearly distinguishable from the tracked objects [36], but when objects occlude each other performance can be impaired because of the extreme crowding that this entails.

In infovis, animated transitions can be used for switching between different views of the same data, or for switching between different temporal snapshots. In these contexts, it is important to be able to track data points when the display changes. In some cases, e.g., when analyzing clusters or during faceted navigation, it is enough to be able to track collections of objects as single entities. In these cases, individual identity may not be critical. Visual objects can also have labels or color encodings that mark their individual identities. But in other cases, several visual objects need to be tracked as distinct individuals, such that interchanging them would lead to incorrect or inefficient use of the displayed information. For example, a user who swaps data points during a scatterplot axis transition [18] may get an erroneous perception of how dimensions correlate.

Work in the perceptual psychology literature also examines these distinct conditions, cases where tracking identity is not required (*NoID*), and cases where it is critical to the task (*ID*). This work clearly shows that while capacity for *NoID* tracking can reach several objects, *ID* tracking is strikingly hard, with capacities as low as 1-2 objects [25, 34, 32]. It appears that while separate systems can track object positions and object identities, coordinating these systems is a strongly resource-limited operation [21].

Despite a wealth of empirical data available from the psychology literature, the data is still too incomplete and controversial to have clear and direct implications to animated transition design. Furthermore, the stimuli used in these studies are primarily designed to capture everyday human experiences. Typical computer animated transitions differ from these stimuli in several respects:

- Most animated transitions employ object paths reflecting purposeful transitions, instead of animations containing complex behavior involving wandering, collision avoidance, bouncing, and changes in direction [2], or rotation [19].
- Most animated transitions are short (the generally recommended duration is 1s [15, 24]), instead of animations lasting e.g., 8s [23]).
- Many animated transitions use a slow in/out pacing [15], while past psychology studies often use abrupt transitions [33].
- When interacting with a computer eye movements are not restricted, while previous studies use a fixation point ("+") [19]

In summary, though despite there exist extensive research in HCI and infovis on animations, prior work has typically focused on particular instances of these animations making it difficult to derive general guidelines for the design of animated transtions. In particular, staggering is believed to has a potential benefit on crowding, but such advantages remain speculative. This work builds on methods employed in perceptual psychology to study to adress this question.

3 Study Design Rationale

The goal of our study is to *i)* determine the factors that affect the difficulty of tracking multiple objects during animated transitions, and *ii)* determine whether these factors can be manipulated using staggering in order to facilitate tracking. We discuss the rationales behind our task design and describe the difficulty metrics and error measures that we introduced for this study.

3.1 Basic Animation Terminology

In HCI and infovis, an *animated transition* between two distinct visual states smooths out an otherwise abrupt visual change. In this article, we refer to the terms *transition* and *animation* as follows:

- A *transition* is a pair of visual states – an initial one and a final one. A visual state can be thought of as a structured image.
- An *animation* is a (usually perceptually continuous) sequence of intermediary images that give the illusion of a smooth progression from a transition's initial visual state to its final visual state.

Transitions are *given*, i.e., they are outside the control of the animation designer. For example, when a data change causes a bar chart to be updated, or when a user chooses to remap a scatterplot's axes, both the initial and the final images are given. In contrast, animations are *designed*, i.e., the designer has total freedom on how to choose all intermediary visual states (i.e. *frames*). In infovis, animation frames do not necessarily map to real data or well-formed visual encodings [24].

Since it is unrealistic to manually design animations for every possible transition, the frame generation process is usually automated:

- An *animated transition technique* is an algorithm for generating animations from transitions.

3.2 Tasks

The effectiveness of animated transition techniques can be estimated empirically by giving users tasks, i.e., by presenting them with various animations and testing their ability to follow these animations. Similar methods are used in visual tracking studies in psychology [33], but these involve long and complex animations which are not representative of animated transitions in infovis. An infovis study requires manipulation of animations – which are a characteristic of the technique being evaluated – independently of the transitions – which are a characteristic of the task. Therefore a task is only defined by a *transition* and a *test* for measuring how well this transition can be understood.

We are interested in evaluating the effectiveness of animated transition techniques at a low perceptual level. We chose to give visual tracking tests that require following one or more moving objects (*targets*) while ignoring other moving objects (*distractors*). Although this is an elementary low-level task, higher level infovis tasks will likely be equally or more difficult as they heavily rely on such perceptual capabilities (see a discussion in [15]).

3.2.1 Choice of Transitions

Because we study visual tracking performance, we focus on transitions that involve changes in object location (transitions can also involve changes in, e.g., color or shape). In order to control for other perceptual processes such as preattentive color processing and visual search, we focus on collections of *visually identical* objects that move from a location to another. The rationale is that if an animated transition technique makes it easier to follow visually identical objects, it should also make it easier to follow visually dissimilar objects (also see [15] for a discussion).

We choose to focus on simple objects that are small enough (*dots*) so that the likelihood of overlap or occlusion is reasonably small when few of them are present. Dots capture the visual marks that can be found in many high-density visualizations such as scatterplots, glyph-based visualizations or node-link diagrams.

3.2.2 Choice of Tests

While Dragicevic *et al.*used tracking of a single object [15], many analytical tasks likely require to follow more than one object at a time. We therefore focus on tests involving multiple object tracking (MOT) instead. We consider two types of MOT tests that have been previously studied in psychology: *i)* group tracking, i.e., tracking collections of moving objects as a whole, and *ii)* identity tracking, i.e., tracking a set of targets with distinct identities. While the first task only requires to identify target objects' final location, the second task also requires to specify which target is which. To summarize, the tasks we consider are defined by:

- A *transition*: a set of dots with initial and final positions.
- A set of *targets*: a subset of dots to be tracked.
- A *test type*: either tracking targets as a group (*NoID*) or tracking target identities (*ID*).

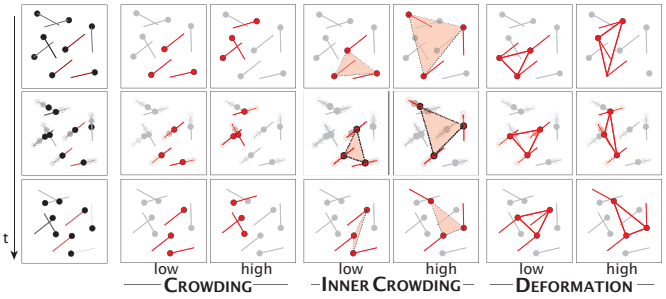

Fig. 1. Illustration of the complexity measures.

3.3 Complexity metrics

Depending on the nature of a task, i.e., a transition, a set of targets and a test type, its difficulty will vary greatly. For example, difficulty likely increases with additional targets or distractors – the number of distractors being correlated with crowding. Assuming the number of targets is known, based on our literature review and informal observations we identified three characteristics of animations likely to influence tracking difficulty: *i) target crowding*, *ii) inner crowding* and *iii) deformation*. We refer to them as *complexity metrics* and provide operational definitions in the following.

3.3.1 General Notations

In the following, we use $\mathcal{P}$ to refer to the set of dots in a transition or animation, and $\mathcal{T} \subset \mathcal{P}$ to refer to the set of targets to track. The symbol p refers to a particular dot in $\mathcal{P}$, and p^t refers to its location at the instant t of the animation. Similarly, $\mathcal{P}^t$ refers to the particular configuration of dots at the instant t. All dots have the same size, s. Dot sizes and coordinates are all normalized between 0 and 1, i.e., divided by the dimensions of the animation window.

3.3.2 Target Crowding

As seen in the background section, previous studies on object tracking suggest that the difficulty of a tracking task highly depends on how often distractors cross the targets' path or come in their vicinity [22]. We define a *target crowding* metric that captures interactions between distractors and targets, and between the targets themselves.

We first define the *instantaneous crowding* of a single dot $p \in \mathcal{P}$ at instant t as follows:

$$crowd(p,t) = \begin{cases} 1 & \text{if } d(p^t, p^t_{close}) \leq s \\ \frac{1/d(p^t,p^t_{close}) - 1}{1/s - 1} & \text{if } s < d(p^t, p^t_{close}) < 1 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where d is the distance between two points and p^t_{close} is the closest neighbor of p at t.

This metric ensures that *i)* crowding is between 0 and 1; *ii)* crowding is 1 when p and p_{close} touch each other or overlap, *iii)* crowding is less than 1 when they do not overlap, *iv)* crowding is 0 when their distance is greater than 1, and *v)* crowding decreases rapidly with distractor distance.

For a given animation A, we define the *crowding* of the dot p, noted $\overline{crowd}(p,A)$, as the mean value of its instantaneous crowding across the entire animation. Finally, we define the *crowding* of a set of dots P as the mean value of all their crowdings:

$$\overline{\overline{crowd}}(P,A) = \frac{1}{|P|} \sum_{p_i \in P} \overline{crowd}(p_i,A) \quad (2)$$

where P is a sets of dots, $|P|$ is the number of dots in this set, and A is an animation.

For any given task (with a set of targets to track), we define *target crowding* as the crowding of $\mathcal{T}$. Note that this metric depends on both the *task* and the *animation* chosen. In order to get a task complexity metric, we define the TARGETCROWDING metric of a tracking task as the crowding of $\mathcal{T}$ computed on the simplest possible animated transition, i.e., a direct linear interpolation of all dot positions. Figure 1 illustrates cases of a low and a high target complexity.

3.3.3 Inner Crowding

As discussed in a background section, previous work has suggested that tracking multiple objects involves attending the entire convex spatial region formed by the targets [41]. This hypothesis together with our informal observations suggest that distractors located between targets may interfere with tracking more than distractors located outside of the convex region. We therefore introduce another metric, *inner crowding*, that captures the average number of distractors intersecting the convex hull formed by the targets (see Figure 1).

We define the *instantaneous inner crowding* of a set of dots P at instant t as:

$$incrowd(P,t) = \sum_{p_i \in \mathcal{P} \setminus P} inside(p_i^t, conv(P^t)) \quad (3)$$

where *conv* refers to the convex hull and *inside* is 1 if the point belongs to the polygon and 0 otherwise. In other terms, *incrowd* captures the number of distractors within the convex hull defined by P.

As before, we derive a measure of average inner crowding $\overline{incrowd}(P,A)$ to capture the amount of inner crowding occurring across an entire animation. This allows us to define *inner crowding* for any particular combination of task and animation by setting $P = \mathcal{T}$. As before, we also define a general task complexity metric InnerCrowding by setting A to a direct linear interpolation.

3.3.4 Deformation

Previous work also suggests that when tracking three targets, the task is harder when the triangle formed by these targets undergoes important distortions over time [41]. This was confirmed by our informal observations. We introduce a *deformation* metric that captures these variations.

We first define the *instantaneous deformation* of a set of dots P at the instant t as:

$$deform(P,t) = \sum_{\{p_i,p_j\} \in P,\, i<j} |d(p_i^t, p_j^t) - d(p_i^{t-\delta}, p_j^{t-\delta})| \quad (4)$$

where d is the distance between two points and δ is the duration of an animation frame.

In other terms, this metrics captures the instantaneous change in length of all possible segments connecting the points P.

We use all segments instead of, e.g., the convex hull or a triangulated mesh, because this structure remains stable over time.

We derive a measure of cumulative *deformation* over an animation, noted $\overline{deform}(P,A)$, by summing up all instantaneous deformations across the entire animation. Note that while instantaneous deformation depends on animation pacing – including its total length and framerate – cumulative deformation does not. Animation framerate only affects the precision with which cumulative deformation is computed. As before, we derive a deformation metric for a specific task and animation by setting $P = \mathcal{T}$, and we define a task complexity metric Deformation by setting A to a linear interpolation.

3.4 Difficulty Metrics

The complexity metrics we introduced only capture characteristics of tasks and animations that we think are correlated with task difficulty, but not task difficulty per se. Task difficulty can only be empirically determined by looking at actual user performance, i.e., by measuring the average precision with which users can track targets. Tracking precision can be estimated through different metrics.

We refer to $\mathcal{S}$ as the set of dots selected by the participant in the test phase of the task. An error metric captures the difference between the selection $\mathcal{S}$ and the set of initial targets $\mathcal{T}$. It can be either binary (correct vs. incorrect), discrete (e.g., number of correct targets) or continuous – capturing how "far" the participant was from the actual answer. In psychology, binary and discrete measurements are commonly used and are aggregated across trials to yield proportions of correct answers. This aggregated measure is commonly referred to as *accuracy* [26]. Another approach is to report the average distance to the correct answer [15, 29]. Both are useful, so we provide operational definitions for both.

3.4.1 Accuracy

For any given trial, we define the *accuracy* of a participant's answer as the number of correctly marked dots divided by the total number of targets. Correctly marked dots refer to all $s_i \in \mathcal{S}$ that belong to $\mathcal{T}$ and, in the case of *ID* tests, are assigned the correct identity.

To guarantee that this proportion is meaningful, we enforce $|\mathcal{T}| = |\mathcal{S}|$ by requiring participants to mark as many dots as targets to track before they can proceed. In the case of *ID* tests, we also require that all identities (colors) are selected once and only once.

3.4.2 Error

Although accuracy metrics have the advantage of simplicity, they tell little about how far the answer is from the actual answer. In our case, a selection that is far from any target should be penalized more than a selection that is right next to a correct target. In infovis, for example, the latter error can be negligible if the position of the dot is used to read values. We therefore introduce a continuous *error* metric that captures the distance to the correct answer.

Dragicevic *et al* introduced an error metric for single object tracking [15]. We generalize their definition to multiple targets and define the *error* between a selection $\mathcal{S}$ and a set of targets $\mathcal{T}$ as:

$$error(\mathcal{S},\mathcal{T}) = \frac{\sum_i err(s_i, t_i^1)}{E(err(\mathcal{P},\mathcal{T}))}, \qquad err(a,b) = ||a-b|| \quad (5)$$

The numerator captures the total distance between targets and selected dots. t_i^1 is the final position of the target of index i, while s_i is the matching selection (a dot). In the *ID* case, targets and selections are matched according to their identity (color), whereas in the *NoID* case, they are matched in a way that yields the lowest possible error.

The denominator is a normalizer as in [15]: $E(err(\mathcal{P},\mathcal{T}))$ corresponds to the expected error that would have been measured had the participant selected random targets, estimated using Monte Carlo methods. Thus, an average error of 1 would mean no knowledge whatsoever on target locations.

4 Experiment 1: Validation of Task Complexity Metrics

In this first experiment, we set out to determine whether task complexity – with regard to the three metrics previously defined – correlates with task difficulty – as measured by our tracking accuracy and error metrics. If lower task complexity yields higher tracking accuracy and lower errors, then our complexity metrics can be used as a proxy for task difficulty, and help design staggered animated transition techniques that improve visual tracking performance without the need to collect empirical data.

Since the purpose of this experiment is only to validate our task complexity metrics, it does not involve staggered animations: all dots move at the same time from their initial to their final position.

4.1 Task Generation

The following describes how we generated the tasks in order to manipulate task complexity. A task consists of a transition, a set of targets, and a test type. Transitions and sets of targets are randomly generated, then selected based on their complexity.

4.1.1 Generation of Random Tasks

We generated random dot transitions of 30 dots in total including targets, with a dot size of 0.03 (recall all measures of distance are normalized between 0 and 1). In addition, each transition had to meet the following requirements: i) dots are at a minimum distance of 0.08 from each other at the initial and final states, and ii) dots travel a fixed distance of 0.5. The goal of requirement i) was to facilitate target cueing and selection. The goal of requirement ii) was to reduce the space of possible transitions and to control for dot speed, since dot speed depends on both animation duration and the distance covered. Transitions were therefore generated by taking segments of fixed length (corresponding to a dot's path), and randomly drawing their midpoint and orientation. All values were chosen based on pilot studies in order to obtain tasks that are neither too difficult nor too easy, and avoid ceiling and floor effects across the different complexity conditions.

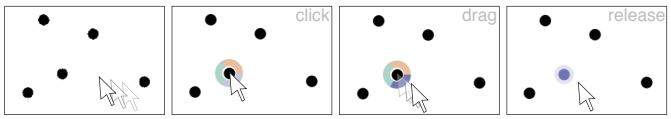

Fig. 2. Marking a blue target with the pie menu widget in the *ID* task.

We generated tasks for both *NoID* and *ID* test types by randomly selecting a set of $n = 3$ targets among the 30 dots. For the *ID* test these 3 target were randomly labeled *red*, *green* and *blue*.

4.1.2 Orthogonality of Task Complexity Metrics

Before proceeding, we tested the orthogonality of our task complexity metrics. The reason is that a task that is high or low on a certain metric (say, target crowding) could also tend to be high or low on another metric (say, deformation). This would mean that our three metrics are not independent and could be reduced to two or one metrics.

We used a Monte Carlo simulation on 10,000 randomly generated tasks to measure the correlation between our three complexity metrics, and obtained the following results:

- inner crowding / crowding: $r = -0.045$, 95% CI [-0.065, -0.026],
- inner crowding / deformation: $r = -0.078$, 95% CI [-0.098, -0.059],
- deformation / target crowding: $r = 0.080$, 95% CI [0.060, 0.100].

Correlations are remarkably low, suggesting that our three metrics are close to orthogonal, and successfully capture different, independent characteristics of the tasks.

4.1.3 Manipulation of Task Complexity

We used the same Monte Carlo simulation to derive a value distribution for each of our metrics. We then took the first and last decile of these distributions in order to classify task complexity into three levels: *Low*, *Medium* and *High*.

In other terms, a task is said to have *Low* target crowding if the target crowding measure belongs to the left tail of the overall target crowding distribution, and to have a *High* target crowding if it belongs to the right tail. The same is true for inner crowding and deformation.

We ignored *Medium* complexity tasks in order to get more sensitive measures. This left us with eight (2^3) different task complexity profiles: *LLL* (for *Low* target crowding, *Low* inner crowding, *Low* deformation), *LLH*, *LHL*, *LHH*, *HLL*, *HLH*, *HHL*, and *HHH*. Tasks following any of these profiles were obtained by selecting from a pool of randomly-generated tasks as detailed above. With two test type conditions (*NoID* and *ID*), the experiment involved a total of $2 \times 8 = 16$ different types of task.

4.2 Procedure and Apparatus

Here we cover the experimental setup and procedure. The experimental software can be downloaded and tested at http://fannychevalier.net/animations

4.2.1 Procedure

Participants first filled out a background questionnaire, then were shown a slideshow explaining the experimental conditions and interactions. They were then asked to perform a series of visual tracking tasks under different complexity conditions and test types. The experiment was broken down into two parts, one for each test type (*NoID* and *ID*). At the beginning of each part, participants were prompted with an instruction screen followed by a block of practice trials. Each trial (practice or measured) consisted in the following sequence:

The participant was first presented with a set of black dots, then asked to press and hold the space bar to reveal the three targets – displayed in red for *NoID*, and red, green and blue for *ID*. Releasing the space bar removed the target highlights (we ensured a minimum highlight duration of 0.5 seconds). Then the participant was asked to press the space bar again to trigger the animation of the dots to their final position. After the animation was completed, the participant was then asked to select the targets in the final point cloud using the mouse. Once satisfied with her answer, the participant had to press and hold the space bar in order to validate her answer and reveal the solution.

For the *NoID* condition, a target was selected by clicking on a dot, and could be deselected by clicking again. For the *ID* condition, we designed a pie menu widget for quick and effortless ID selection (see Figure 2). To mark a dot blue, for example, the participant just had to perform a mouse press on the dot and drag to the blue area of the pie menu. Clicking without dragging would deselect an already-selected target. Participants could not proceed until they provided a complete answer, i.e., three red dots in the *NoID* test, and one dot of each color in the *ID* test. The correct answer was shown in order to motivate participants and reduce the likelihood that they misunderstood the tasks.

Participants were instructed to answer as accurately as possible, and make their best guess whenever they did not know the answer. Pilot testing suggested that *ID* tracking required effortful memorization unless the initial target colors were subvocalized. Since we wanted to test visual tracking and not short-term memory, we allowed subvocalization and even explicitly encouraged its use in order to level out strategies. Participants were instructed not to point at the screen. Participants were regularly prompted with an invite to rest.

After completing all trials, participants filled out a qualitative questionnaire. The whole experiment took approximately one hour.

4.2.2 Apparatus and Setup

The experiment was conducted on a desktop computer equipped with a mouse, keyboard, and an LCD display of resolution 1280×1024 pixels, 2.95×2.96 mm pixel size, and 60 Hz refresh rate. Dots were shown in a 600×600 pixels area (17.72×17.75 cm), and were displayed as black circles of 18 pixels (∼0.53 cm). Participants were sitting at a distance of approximately 65 cm to the display.

4.3 Participants

We recruited 20 unpaid participants (12 female) aged 18-20 (mean 18.5). All were students in psychology and were given course credit in compensation for participating in this experiment.

4.4 Experimental Design

Our independent variables were test type (*NoID* or *ID*) and task complexity profile. The task complexity profile has eight levels that can be broken down into two levels (*Low* or *High*) for each of the three complexity metrics. Each type of task was repeated eight times. To summarize, our factors were:

20	participants
× 2	TEST (*NoID*, *ID*)
× 2	TARGETCROWDING (*Low, High*)
× 2	INNERCROWDING (*Low, High*)
× 2	DEFORMATION (*Low, High*)
× 8	repetitions
2560	trials

Animated transition technique was not a factor in this experiment. All tasks were presented with a non-staggered, direct animation, where all dots moved at the same time, at the same speed and on a straight path. All animations lasted one second and used a slow-in slow-out pacing as recommended in [15].

The trials were blocked by TEST, yielding two blocks of $2 \times 2 \times 2 \times 8 = 64$ trials. Each block was preceded by 8 practice trials (1 per complexity profile). Trials were fully randomized within each block and the block presentation order was counterbalanced across participants.

Our two dependent variables were tracking ACCURACY and tracking ERROR, as defined in the previous section.

4.5 Hypotheses

Based on our previous analysis, our hypothesis was that our three task complexity metrics all correlate with task difficulty, and that all three metrics are equally important. In other terms, we predicted that:

- For all three metrics (TARGETCROWDING, INNERCROWDING and DEFORMATION), tasks of high complexity will be clearly more difficult on average than tasks of low complexity,
- These effects will be observed for both difficulty metrics (ACCURACY and ERROR),
- The effects will be similar for all three task complexity metrics.

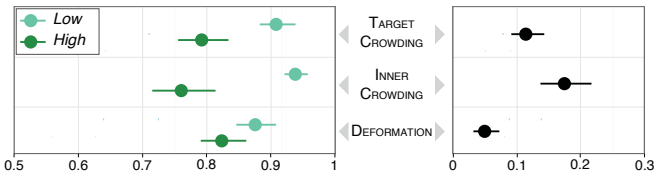

Fig. 3. Mean Accuracy and mean Accuracy difference as a function of task complexity for the *NoID* tracking task. Error bars are 95% CIs.

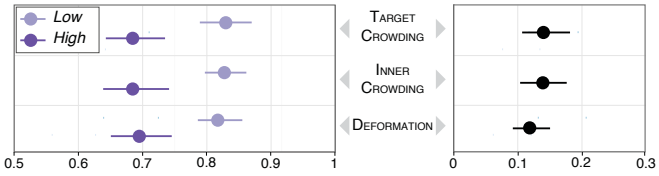

Fig. 4. Mean Accuracy and mean Accuracy difference as a function of task complexity for the *ID* tracking task. Error bars are 95% CIs.

4.6 Results

Due to growing concerns in various research fields over the limits of null hypothesis significance testing for reporting and interpreting experimental results [13, 16], we base all our analyses and discussions on estimation, i.e., effect sizes with confidence intervals [14]. This approach also aligns with the latest recommendations from the APA [3].

We break down our analysis into a confirmatory and an exploratory section [13]. All analyses in the confirmatory section were prespecified before conducting the experiment, i.e., the R scripts used for computing effect sizes, confidence intervals and for generating initial drafts of the figures were written in advance and tested on pilot data. Analyses in the exploratory section were conducted to address additional questions raised while looking at the experimental data.

4.6.1 Confirmatory analysis

We observed similar trends for the two difficulty metrics, so we focus our analysis on Accuracy. The effect of task complexity on Accuracy is reported in Figures 3 and 4, for the *NoID* and *ID* tasks respectively.

On Figure 3, the left plot shows the effect of each complexity metric (one per row). The effect of each metric was assessed by performing a contrast, i.e., aggregating all trials where the metric was *Low* and all trials where the metric was *High*. For each metric, two mean accuracies were therefore derived for each participant (one for *Low* and one for *High*), yielding a total of 20 data points for *Low* and 20 data points for *High*. Point estimates and 95% confidence intervals were computed using bootstrapping [27]. On the Figure, higher values (to the right) mean higher accuracy. Points indicate best estimates while intervals indicate all plausible values, the point estimates being about 7 times more likely than interval endpoints [12]. Thus the effect of complexity on accuracy is clear, except for Deformation. The right plot shows point and interval estimates of the difference of mean accuracy between *Low* and *High*, computed for each participant (yielding 20 data points in total). This method gives more precise effect size estimates (as often in within-subjects designs) that leave no doubt as to the effect of Deformation.

Figure 4 shows the results of the same analysis considering *ID* tasks instead of *NoID* tasks. Clear effects can also be seen for all three metrics, and this time the effects are quite similar.

Overall, our data are consistent with the trends initially predicted by our hypothesis, except for the relative importance of our three metrics. We initially conjectured that the three metrics would have roughly the same importance, which was confirmed for the *ID* tracking task but not the *NoID* tracking task: when the task only requires to track targets as a group, Deformation has a measurable detrimental effect but it is the least important factor. Both TargetCrowding and InnerCrowding have a higher influence on performance, with InnerCrowding being particularly detrimental to tracking accuracy.

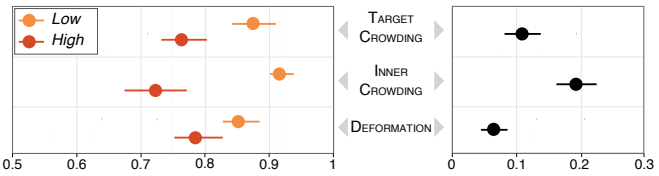

Fig. 5. Mean Accuracy and mean Accuracy difference of target *selection* only (no identification) as a function of task complexity for the *ID* tracking task. Error bars are 95% CIs.

4.6.2 Exploratory analysis

Here we address the question as to why the three metrics have a comparable effect for *ID* tasks, but quite different effects for *NoID* tasks. Recall the *ID* tasks involved two components and thus two possible sources of error: *i)* identifying where the three targets ended up and *ii)* identifying which target is which. To get a better understanding of the results for the *ID* task, we examined the two types of error separately.

We first reanalyzed the answers to the *ID* task by considering participants' accuracy in selecting targets irrespective of their identities, as we did for the *NoID* task. Figure 5 shows the mean accuracy and mean accuracy difference for the *ID* task estimated using the same metric as the *NoID* task. The results were remarkably close to the *NoID* task (Figure 3), suggesting that participants were able to follow the three targets as if no identity tracking was required, but mixed up their identities more or less frequently depending on the task's complexity profile. This in turn suggests that the difficulty of tracking dot identities varies depending on the task's complexity profile, since the trends are very different (i.e., accuracies even out) when both sources of error are accounted for (see Figure 4).

Correctly selected targets:		1	2	3
TargetCrowding	*Low*	37	140	457
	High	86	219	317
InnerCrowding	*Low*	10	128	501
	High	113	231	273
Deformation	*Low*	50	148	433
	High	73	211	341

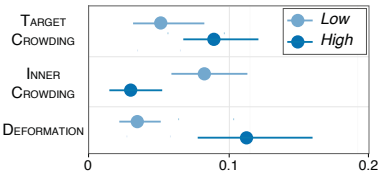

Fig. 6. Left: Total number of trials per successfully selected targets and complexity. Right: Percentage of misidentified targets per correctly selected targets as a function of task complexity for 3 correctly selected targets. Error bars are 95% CI.

This conjecture is confirmed when analyzing the proportion of misidentification errors (i.e., wrongly selected colors) among the dots that have been correctly identified as targets. The plot in Figure 6 shows the proportion of misidentification errors for trials where participants successfully selected all three targets (∼30% of all trials). Trends were similar in the other cases, though the interval estimates were too wide for a meaningful analysis.

On Figure 6, higher estimates (i.e., more to the right) mean more identification errors. It can be seen that deformation seriously impairs target identity tracking, as participants are much more likely to mix up colors than when the triangle formed by the three targets only translates or rotates. Target crowding also seems to play a role, although to a much lesser extent. Most remarkably, a task with *higher* inner crowding makes participants *less* likely to mix up targets. One possible explanation is that when inner crowding is low, the three targets are mostly separated by empty space and tend to be close to each other, and thus are more likely to be merged into a single object by visual attention processes [23]. This attentional merging can lead participants to "forget" target identities. In contrast, in high inner crowding conditions, the larger spacing between targets and the intervening distractors may facilitate the tracking of the three targets as separate objects. This however only concerns trials where dots have been successfully tracked in the first place, which are much less numerous when inner crowding is high (see Figure 5). Nevertheless, these findings can have interesting applications for situations where switching target identities is much more costly than loosing track of them.

4.6.3 Questionnaire

Since participants were told that some tasks were more difficult than others without further details, we only asked which strategy they developed, and when such strategy was challenged. Their feedback provides valuable insights on the *perceived* difficulty of the tasks.

Among the developed strategies, participants i) pictured the dots as a virtual triangle (4 participants) – which they reported to be challenging when the shape collapsed, i.e., when a vertex crossed the opposite edge, especially in the *ID* test; ii) "blurred" their focus and relied on peripheral vision during the animation (5 participants) – a strategy that was perceived more difficult when the targets were crowded; or iii) eventually gave up on tracking all three targets and kept their focus on only two of them, hoping to guess the third right without conviction.

These results suggest that though the difference in complexity between the *Low* and *High* conditions are rather low values (10% to 20% of the total range of complexities), these variations for TARGETCROWDING and DEFORMATION are still *perceived* as being highly different.

In summary, this experiment confirms that task complexity as we defined it can be used as a proxy for task difficulty, and that each of the three metrics captures a different aspect of difficulty. The following examines whether staggering can be used to manipulate these factors.

5 ANALYSIS: CAN STAGGERING REDUCE TASK COMPLEXITY?

We know from the previous experiment that our complexity measures are correlated to task difficulty. As discussed in Section 3, task complexity is a function of the animated transition chosen. An animated transition technique that reduces task complexity on average may be therefore more promising than a technique that leaves it unchanged or increases it. In this analysis, we set out to determine whether staggering can reduce task complexity when very few assumptions are made about the nature of the visual transition.

5.1 Staggered Animation Designs

By staggered animation we mean an animation where all elements move at different times and *i)* each element starts moving with a constant temporal delay δ_t after the previous one, and *ii)* all elements move for the same duration. Such animations can be characterized by two parameters:

- the *order* in which the individual elements move, and,
- a *dwell* factor, capturing the degree of motion sequentiality:

$$dwell(|\mathcal{P}|, t(\mathcal{A}), \delta_t) = \delta_t \cdot \frac{|\mathcal{P}|}{t(\mathcal{A})}$$

where $|\mathcal{P}|$ is the number of moving elements, $t(\mathcal{A})$ is the total animation duration, and δ_t is the delay previously mentioned. In other terms, a dwell of 0 corresponds to an animation where all dots move simultaneously (no sequentiality), and a dwell of 1 corresponds to an animation where they all move in sequence (maximum sequentiality).

When designing staggered animation techniques, there are many possible orders to chose from and many possible values for dwell, and these choices might influence the effectiveness of the technique for a number of reasons. While our primary concern is reducing task complexity, staggering may cause additional perceptual difficulties due to *i)* a lower predictability as to which elements will move and when, and *ii)* the increased speed with which elements move. While *i)* is impacted by the choice of ordering, *ii)* is impacted by the choice of dwell. We explain which designs we chose for the purposes of this analysis.

5.1.1 Choice of Order

The predictability of staggered transitions may be facilitated by moving the dots in a systematic order, such as based on their initial location (e.g., from top to bottom). However, fixing the ordering also limits the possible animations to choose from, making it less easy to optimize complexity. We therefore chose to test two ordering schemes: one that optimizes predictability, and another one that optimizes complexity.

- *Spatial ordering* consists in having dots move according to their initial y-coordinate, starting from the topmost dot and finishing with the bottom-most dot.
- *Smart ordering* consists in having dots move in a way that tries to minimize crowding.

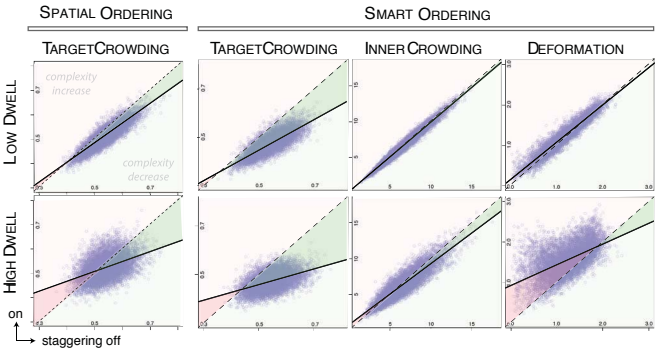

Fig. 7. Complexity of transitions without staggering vs. with staggering animation for SPATIAL, and *Low* (top) and *High* (bottom) dwell.

For *smart ordering*, we focus on minimizing crowding primarily because informal experimentation suggested it is the complexity metric that is the easiest to minimize. This will be later confirmed by our analyses. Our current implementation uses a simple Monte Carlo approach that generates 2,000 random orders and picks the one with the lowest crowding. Crowding is computed by setting all dots as targets (i.e., we compute TARGETCROWDING, for $\mathcal{T} = \mathcal{P}$). Although one could possibly optimize complexity only for the targets, such a technique would be of little use for real computer applications, where the system is rarely aware of which elements of the display the user chose to focus on. Nevertheless, minimizing global crowding makes it more likely that target crowding is also reduced. Finally, smart ordering is computed based on the dwell value, discussed in the next section.

5.1.2 Choice of Dwell

The same way different staggering orders come with benefits and drawbacks, different dwell times also have trade-offs. In particular, a higher dwell increases motion sequentiality, which could help focus on individual dots. However, for a fixed animation duration, a higher dwell also increases motion speed, since dots have less time to reach their final destination. Though speed is not an issue per se, devoting fewer animation frames to each dot motion may be detrimental [19].

After informal testing with different dwell values on transitions involving 30 dots, we converged to three values: a LOW DWELL (0.2), a MEDIUM DWELL (0.4) and a HIGH DWELL (0.6). These values provide a good coverage of all reasonable dwell values, as values higher than 0.6 yielded animations close to impossible to follow for these transitions.

5.2 Analysis Results

We tested the six different staggering techniques obtained by fully crossing the ORDER conditions (SPATIAL, SMART) and the DWELL conditions (LOW, MEDIUM, HIGH). We generated 10,000 random tasks using the same criteria as in the previous experiment, and measured the differences in task complexity profiles compared to no staggering.

In Figure 7, scatterplots show the complexity for all 10,000 tasks, for all complexity metrics (on columns) and dwell values (on rows)—due to space limitation, only the most illustrative conditions are showed. The x-axis of each scatterplot is the complexity measurement without staggering. The y-axis reflects the complexity measurement of the same task with staggering. The dashed line is the identity line. The thick line is the linear regression. Any data point located under the dashed line (the green area) indicates that staggering has successfully reduced task complexity with regard to the complexity metric.

The plots for target crowding (two left columns) suggest that SPATIAL staggering slightly reduces crowding overall, both with LOW and HIGH dwell, while SMART yields more improvement. However, there is a large variability in both cases (sometimes staggering makes things worse) and the reduction is overall modest for SPATIAL in particular. Reassuringly, the slope of the regression lines suggest that staggering reduces crowding more when it is already high. These results confirm previous intuitions that staggering can reduce crowding by having visual objects move at different times and "avoid" each other, but the average reduction on unstructured transitions is surprisingly low.

The plots for inner crowding (third column) suggest that SMART staggering has little impact on inner crowding, as the linear regression and identity line are almost confounded. With HIGH dwell, the effect is observable but still small. We observed similar trends for SPATIAL. That staggering does not affect inner crowding is hardly surprising: inner crowding is generally high when targets are spaced out, and staggering does little in bringing targets closer to each other.

The plots for deformation (to the right) suggest that SMART with LOW dwell has little impact on deformation, while its effect with HIGH dwell is mostly detrimental, especially when deformation is initially low. Here too, SPATIAL yielded similar trends, though to a lesser extent. Again, this result is not surprising, since if objects move together in a coherent fashion, staggering will tend to break this coherence.

In summary, our analyses confirm that staggering can help reduce task complexity is some cases, but these cases are rare and the difference on average is modest (for target crowding) or negligible (for inner crowding). Moreover, staggering tends to increase deformation. Given the possibly detrimental effects introduced by staggering that are independent from the task complexity profile, the benefits of staggering —at least when systematically applied to unstructured transitions— are doubtful. The question however remains as to whether staggering can be beneficial for those rare cases where it does significantly reduce task complexity. This is the subject of our second experiment.

6 EXPERIMENT 2: MEASURING THE EFFECT OF STAGGERING

So far we have learned that staggering fails to consistently reduce task complexity metrics. Furthermore, staggering can decrease predictability and increase motion speed, which may make tracking tasks even harder. A pending question is whether staggering still helps when it successfully reduces task complexity. We conducted a second experiment where we tested SPATIAL and SMART staggering on the most favorable tasks. While such tasks are unrepresentative of unstructured transitions, we seek to understand whether staggering can sometimes be beneficial, and whether one should try to minimize complexity as much as possible at the detriment of predictability, or vice versa.

6.1 Task Generation

In contrast with our first experiment where the goal was to test tasks with different complexity profiles, the goal in this second experiment is to test the tasks that are likely to benefit the most from staggering. Since all three complexity metrics correlate with difficulty, we must choose which importance to give to each metric. We found *ID* tasks to be equally impacted by all metrics (Figure 4), while *NoID* tasks were the most impacted by DEFORMATION (Figure 3). Our simulations also revealed that staggering has very little impact on INNERCROWDING and tends to increase DEFORMATION more than it reduces it. Therefore, we selected tasks where staggering yields the best reduction in TARGETCROWDING without increasing INNERCROWDING and DEFORMATION.

One difficulty is that a task that favors a given staggering technique may not favor another technique. To guarantee a fair comparison, we select the most favorable tasks *per technique*. Thus we compare techniques by the *gain in performance* they yield in the best cases scenarios, using direct animation as a baseline.

For each technique defined by an ORDER and a DWELL, we picked the n most favorable tasks as follows: we generate $n \times 10,000$ random tasks for the *ID* task, and prune those where INNERCROWDING or DEFORMATION is increased by staggering. From the remaining pool, we keep the n tasks that reduce TARGETCROWDING the most. In other words, we select the 0.01% most favorable cases for each staggering technique.

6.2 Procedure and Apparatus

We used the same setup and followed the same procedure as Experiment 1 (Sec. 4.2) with a few changes. First, we tested the *ID* tasks only, based on the results of Experiment 1 suggesting that performance for *NoID* tasks can be estimated fairly well by using *ID* tasks and the accuracy metric from the *NoID* task (compare Figures 3 and 5). The experiment was also broken down into 16 blocks instead of 2. Each block consisted of a series of trials with the same animation technique preceded by instructions (more below on the experiment design).

6.3 Participants

We recruited 20 new participants (14 female) aged 18-35 (mean 22.5). All were students or University staff and were compensated $10.

6.4 Experimental design

Since we were interested in the improvement provided by staggering over direct animations, we presented each task twice, with and without staggering, to allow for pairwise comparison. Due to the block design, a task was never presented twice in the same block.

Our independent variables were presence of staggering (*on*, *off*), ORDER (SPATIAL, SMART) and DWELL (LOW, HIGH). Participants were presented with two blocks of five repetitions each for each combination of staggering, ORDER and DWELL. To summarize, our factors were:

20	participants
× 2	staggering (*on*, *off*)
× 2	ORDER (SPATIAL, SMART)
× 2	DWELL (LOW, HIGH)
× 2	blocks
× 5	repetitions
1600	Trials

The presentation order of all $2\times2\times2\times2 = 16$ blocks was fully randomized, as well as all 5 repetitions within each block. The experiment was preceded by 32 practice trials, 4 for each technique.

We computed *accuracy gain* and *error gain* by taking the difference in ACCURACY and ERROR between staggering and no staggering for the same task. Both gains were measured by taking into account identification errors (*ID* metric), and without taking into account identification errors (*NoID* metric). Therefore we had four dependent variables.

6.5 Hypotheses

Based on our simulations and on further informal observations, our hypothesis was that for the favorable tasks, staggering will slightly improve accuracy if the dwell is low and if the order is predictable. We indeed noticed that both fast and unpredictable dot movements seemed to make tracking more challenging. In other terms, we predicted that:

- Tasks will be easier on average for staggered animations using SPATIAL ordering and LOW dwell, compared to direct animations.
- The SMART technique will exhibit no observable improvement over direct animations or will be detrimental, for both dwell conditions.
- HIGH dwell will exhibit no observable improvement over direct animations or will be detrimental, for both order conditions.

6.6 Results

As before, we use an estimation approach to data analysis and break down our analyses into confirmatory and exploratory.

6.6.1 Confirmatory Analysis

The effects of staggering ORDER and DWELL on accuracy gain are reported in Figure 8. The left plot shows the accuracy gain for each of the four staggering techniques, using the *ID* performance metric. For each staggering condition, a mean accuracy gain was derived for each participant, yielding a total of 20 data points. Point estimates and 95% confidence intervals were computed using bootstraping methods. The right plot shows the effects measured when the *NoID* performance metric is used. Figure 9 shows the same analyses for error gain. On all Figures, values to the right (high for accuracy difference and low for error difference) indicate that staggering is beneficial.

Contrary to what we predicted, SMART staggering with a HIGH dwell yields an observable gain in accuracy, particularly for the *ID* metric (Figure 8, left plot). We can also be fairly confident that SPATIAL staggering with a HIGH gain yields a small increase in error, particularly for the *NoID* metric (Figure 9, right plot). For all other conditions, the sampling error is too large for us to be able to conclude, but the general trend seems to be that SMART ordering outperforms SPATIAL ordering.

Overall these results suggest that contrary to what we initially expected, reducing TARGETCROWDING – which SMART staggering with HIGH dwell does best (Figure 7) – seems to be more important than enforcing predictability. This seems to be true both for *ID* and *NoID* tasks.

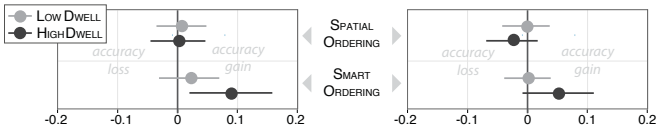

Fig. 8. Mean accuracy gain as a function of staggering technique, for the *ID* metric (left) and the *NoID* metric (right). Error bars are 95% CIs.

Fig. 9. Mean error gain as a function of staggering technique, for the *ID* metric (left) and for the *NoID* metric. Error bars are 95% CIs.

Despite these trends, if we consider effect sizes, the benefits of staggering are lower than we could have expected. For the best technique, the increase in accuracy most likely does not go beyond 0.15 and the reduction in error does not go beyond 0.1. These very liberal estimates are comparable to the gains brought by, e.g., switching from high to low deformation when performing *NoID* tracking tasks on non-staggered animations (see Figure 3). These gains are substantial but surprisingly low given that we selected the 0.01% (1 out of 10,000) most favorable cases from our pool of randomly-generated tasks. Given this, we can fairly conclude that the facilitating effects of staggered animated transitions are far from staggering when applied to complex dot transitions as found in, e.g., dynamic scatterplots.

6.6.2 Exploratory Analysis

We focus the rest of our exploratory analysis on the questionnaire.

In contrast with our first experiment, participants were fully aware of the set of experimental conditions (i.e. animation techniques). We asked them to rank task difficulty for each condition on a five-point Likert scale (very easy to very difficult). Figure 10 shows the results.

Fig. 10. Participants' assessment of difficulty per condition.

An interesting result is that the tasks where participants performed the best (Smart, High) were also perceived as the most difficult overall. Answers align with our first conjecture that both fast and unpredictable dot movements appear to be highly challenging. Similarly, participants ranked the Spatial ordering with Low dwell as easier than no staggering overall, though the gain was not measurable in practice.

This suggests that even though a staggering technique improves performance, the observer may still feel overwheled by the perceived difficulty. Conversley, it seems that staggering that is not beneficial for visual tracking, can give the illusion of facilitating. From an animation design perspective, this means that there may still be a value of using staggered animations in applications that do not require high accuracy.

In summary, this last experiment confirms that for best performance, animated transitions should try to reduce TargetCrowding first, though participants found unpredictability harder. We found some minor evidence of gains of staggering in some particular conditions. The effect is discouragingly poor given that we tested the best case scenarios for staggering. On all possible transitions, we suspect that the gains would have most likely been non-measurable, or even possibly negative given the non-predictability and the increase in deformation.

7 Discussion

Our study confirms that task difficulty can be assessed by the means of our three complexity metrics. In particular, we were able to validate prior work in perceptual psychology that visual tracking is impaired by crowding. We also found that higher inner crowding harms tracking of the target end position, whereas it surprisingly facilitates identity tracking. In contrast, deformation is harmful to tracking, but only when individual target identities must be maintained.

Follow-up investigations showed that staggering can reduce complexity as measured by crowding (keeping the two other metrics constant), under two different animation techniques. However, such cases are rare, and the gain is modest overall. We tested tracking performance on the 0.01% best case scenarios to assess whether reducing crowding with staggering was beneficial keeping in mind that any improvement could be at the cost of additional difficulties introduced by staggering. Interestingly, experimental data provide evidence that it is preferable to reduce crowding than maintain predictability and low motion speed, with a measurable gain of performance.

What makes these 0.01% of trials so special? These rare trials where crowding is reduced by staggering must have a particular type of spatial structure, or exhibit a particular type of motion pattern. Previous work that explores staggering focus on visualizations with specific configurations, such as 1-dimensional bar charts, pie charts [24] and histograms [11]. In contrast, our study focused on the more general case of 2D random point clouds. We attemped to find generalized structures or motion patterns among the top 0.001% of least-crowded staggered trials, to no avail. Our selected tasks reduce complexity for the arbitrarily-chosen targets only, and it is possible that no staggered transition technique will reduce complexity in a target-agnostic way.

All in all, while more work is needed to identify the potential benefits for staggering, this work contributes several methodological tools, and overall design considerations:

- All three complexity metrics *target crowding*, *inner crowding* and *deformation* are a good proxy for visual tracking task difficulty and can be used as reference measures in further experiments.
- Among the three complexity metrics, *target crowding* is the one that yields the most benefit from staggering. We thus recommend to primarily consider this factor when designing staggered animations.
- From a viewer's perspective, the positive aspect of staggering – a potential to bring tiny improvements in crowding and accuracy – is likely to be outwheighted by its negatives – a loss of predictability in motion start times, and faster motion of individual elements.

The above implications for design, though informative, must be considered with caution since we tested the effect of staggering on very specific, low level tasks which may not be strongly representative of typical infovis tasks. We have chosen to carefully control the details that may affect the study results in order to isolate the effect of our independant variables on our dependant variables. We thus favoured internal validity over external validity [8], and future work should confirm that our results apply more widely in real-world contexts. This said, as previously discussed, if users are unable to perform such elementary tasks, then more complex tasks involving visual tracking will likely be as much or more difficult to carry out (see also [15]). This work helps us better understand animations, and we hope that it will inspire further development in the area.

8 Conclusion and Future Work

This work examines the potential for staggering to improve visual tracking tasks in animated transitions. We contribute the definition of complexity metrics to assess task difficulty, and measure if and to which extent reducing complexity through staggering yields better performance. Our results suggest that staggering may be beneficial in some conditions, but they have yet to be demonstrated.

This work is a significant step toward a better understanding of pacing in animated transitions. While our results do not demonstrate strong benefits for staggering, this work provide useful directions for further research. More work is needed to identify the potential of staggered animations beyond their aesthetic value. In particular, we plan to study effects on data that exhibit meaningful spatio-temporal structure and extend our investigation to sequential animations.

Acknowledgments

We thank Bruno de Araujo and Mathieu Nancel for their valuable input on this paper, Kevin Hartstein and Alan Pan for their help running the experiments, and our reviewers for their useful comments.

REFERENCES

[1] Microsoft pivot. http://tinyurl.com/pivotlabs, 2009. *Retrieved March 2014*.

[2] G. A. Alvarez and S. L. Franconeri. How many objects can you track?: Evidence for a resource-limited attentive tracking mechanism. *Journal of Vision*, 7(13):14, 2007.

[3] American Psychological Association. *The Publication manual of the American psychological association (6th ed.)*. Washington, DC, 2010.

[4] B. Bach, E. Pietriga, and J. Fekete. Graphdiaries: Animated transitions and temporal navigation for dynamic networks. 2013.

[5] R. Baecker and I. Small. Animation at the interface, 1990.

[6] B. B. Bederson and A. Boltman. Does animation help users build mental maps of spatial information? In *Information Visualization, 1999.(Info Vis' 99) Proceedings. 1999 IEEE Symposium on*, pages 28–35. IEEE, 1999.

[7] A. Bezerianos, F. Chevalier, P. Dragicevic, N. Elmqvist, and J.-D. Fekete. Graphdice: A system for exploring multivariate social networks. In *Computer Graphics Forum*, volume 29, pages 863–872. Wiley Online Library, 2010.

[8] B. J. Calder, L. W. Phillips, and A. M. Tybout. The concept of external validity. *Journal of Consumer Research*, pages 240–244, 1982.

[9] B.-W. Chang and D. Ungar. Animation: from cartoons to the user interface. 1995.

[10] F. Chevalier, P. Dragicevic, A. Bezerianos, and J.-D. Fekete. Using text animated transitions to support navigation in document histories. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 683–692. ACM, 2010.

[11] F. Chevalier, P. Dragicevic, and C. Hurter. Histomages: fully synchronized views for image editing. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, pages 281–286. ACM, 2012.

[12] G. Cumming. *Understanding the new statistics: Effect sizes, confidence intervals, and meta-analysis*. Routledge, 2013.

[13] G. Cumming. The new statistics: Why and how. *Psychological science*, 25(1):7–29, 2014.

[14] G. Cumming and S. Finch. Inference by eye: Confidence intervals and how to read pictures of data. *American Psychologist*, 60(2):170, 2005.

[15] P. Dragicevic, A. Bezerianos, W. Javed, N. Elmqvist, and J.-D. Fekete. Temporal distortion for animated transitions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2009–2018. ACM, 2011.

[16] P. Dragicevic, F. Chevalier, and S. Huot. Running an hci experiment in multiple parallel universes. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, pages 607–618. ACM, 2014.

[17] P. Dragicevic, S. Huot, and F. Chevalier. Gliimpse: Animating from markup code to rendered documents and vice versa. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 257–262. ACM, 2011.

[18] N. Elmqvist, P. Dragicevic, and J.-D. Fekete. Rolling the dice: Multidimensional visual exploration using scatterplot matrix navigation. *Visualization and Computer Graphics, IEEE Transactions on*, 14(6):1539–1148, 2008.

[19] S. Franconeri, S. Jonathan, and J. Scimeca. Tracking multiple objects is limited only by object spacing, not by speed, time, or capacity. *Psychological Science*, 21(7):920–925, 2010.

[20] S. L. Franconeri. The nature and status of visual resources. *Oxford handbook of cognitive psychology*, 8481, 2013.

[21] S. L. Franconeri, G. A. Alvarez, and P. Cavanagh. Flexible cognitive resources: competitive content maps for attention and memory. *Trends in cognitive sciences*, 17(3):134–141, 2013.

[22] S. L. Franconeri, J. Y. Lin, J. T. Enns, Z. W. Pylyshyn, and B. Fisher. Evidence against a speed limit in multiple-object tracking. *Psychonomic Bulletin & Review*, 15(4):802–808, 2008.

[23] S. L. Franconeri, Z. W. Pylyshyn, and B. J. Scholl. A simple proximity heuristic allows tracking of multiple objects through occlusion. *Attention, Perception, & Psychophysics*, 74(4):691–702, 2012.

[24] J. Heer and G. G. Robertson. Animated transitions in statistical data graphics. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1240–1247, 2007.

[25] T. S. Horowitz, S. B. Klieger, D. E. Fencsik, K. K. Yang, G. A. Alvarez, and J. M. Wolfe. Tracking unique objects. *Perception & psychophysics*, 69(2):172–184, 2007.

[26] J. Hulleman. The mathematics of multiple object tracking: From proportions correct to number of objects tracked. *Vision research*, 45(17):2298–2309, 2005.

[27] K. N. Kirby and D. Gerlanc. Bootes: An r package for bootstrap confidence intervals on effect sizes. *Behavior research methods*, 45(4):905–927, 2013.

[28] J. Lasseter. Principles of traditional animation applied to 3d computer animation. In *ACM Siggraph Computer Graphics*, volume 21, pages 35–44. ACM, 1987.

[29] L. Micallef, P. Dragicevic, and J. Fekete. Assessing the effect of visualizations on bayesian reasoning through crowdsourcing. *Visualization and Computer Graphics, IEEE Transactions on*, 18(12):2536–2545, 2012.

[30] D. G. Pelli, M. Palomares, and N. J. Majaj. Crowding is unlike ordinary masking: Distinguishing feature integration from detection. *Journal of vision*, 4(12):12, 2004.

[31] C. Plaisant, J. Grosjean, and B. B. Bederson. Spacetree: Supporting exploration in large node link tree, design evolution and empirical evaluation. In *Information Visualization, 2002. INFOVIS 2002. IEEE Symposium on*, pages 57–64. IEEE, 2002.

[32] Z. Pylyshyn. Some puzzling findings in multiple object tracking: I. tracking without keeping track of object identities. *Visual cognition*, 11(7):801–822, 2004.

[33] Z. W. Pylyshyn and R. W. Storm. Tracking multiple independent targets: Evidence for a parallel tracking mechanism. *Spatial vision*, 3(3):179–197, 1988.

[34] J. Saiki. Multiple-object permanence tracking: Limitation in maintenance and transformation of perceptual objects. *Progress in brain research*, 140:133–148, 2002.

[35] C. Schlienger, P. Dragicevic, C. Ollagnon, and S. Chatty. Les transitions visuelles différenciées: principes et applications. In *Proceedings of the 18th International Conferenceof the Association Francophone d'Interaction Homme-Machine*, pages 59–66. ACM, 2006.

[36] B. J. Scholl and Z. W. Pylyshyn. Tracking multiple items through occlusion: Clues to visual objecthood. *Cognitive psychology*, 38(2):259–290, 1999.

[37] M. Shanmugasundaram and P. Irani. The effect of animated transitions in zooming interfaces. In *Proceedings of the working conference on Advanced visual interfaces*, pages 396–399. ACM, 2008.

[38] M. Shanmugasundaram, P. Irani, and C. Gutwin. Can smooth view transitions facilitate perceptual constancy in node-link diagrams? In *Proceedings of Graphics Interface 2007*, pages 71–78. ACM, 2007.

[39] D. J. Simons and D. T. Levin. Change blindness. *Trends in cognitive sciences*, 1(7):261–267, 1997.

[40] B. Tversky, J. B. Morrison, and M. Betrancourt. Animation: can it facilitate? *International journal of human-computer studies*, 57(4):247–262, 2002.

[41] S. Yantis. Multielement visual tracking: Attention and perceptual organization. *Cognitive psychology*, 24(3):295–340, 1992.