Comparative Eye Tracking Study on Node-Link Visualizations of Trajectories

Rudolf Netzel, Michel Burch, and Daniel Weiskopf, *Member, IEEE Computer Society*

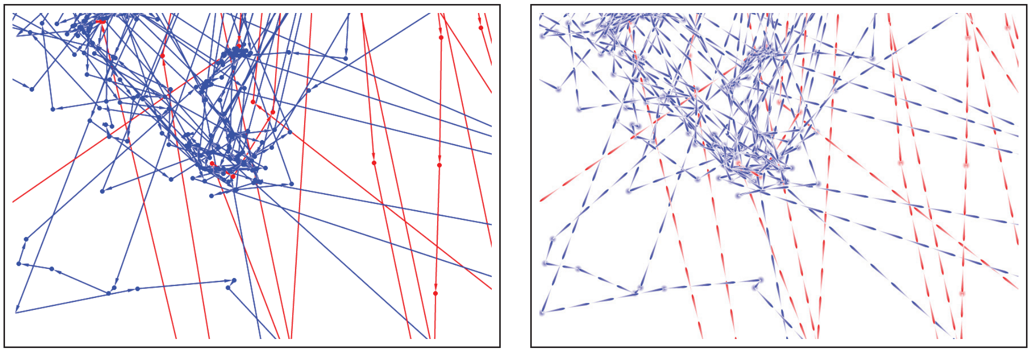


Fig. 1. Examples of different node-link visualizations: they show data from animal movement ecology in the form of GPS tracks of two oystercatcher birds (distinguished by red/blue color). The visualizations show zoomed-in views of a large data set. Due to zooming, there are some links for which neither start nor end node are visible; here, the texture (right image) provides an indication of movement direction that is completely missing with the standard arrow rendering (left image).

Abstract— We present the results of an eye tracking study that compares different visualization methods for long, dense, complex, and piecewise linear spatial trajectories. Typical sources of such data are from temporally discrete measurements of the positions of moving objects, for example, recorded GPS tracks of animals in movement ecology. In the repeated-measures within-subjects user study, four variants of node-link visualization techniques are compared, with the following representations of directed links: standard arrow, tapered, equidistant arrows, and equidistant comets. In addition, we investigate the effect of rendering order for the halo visualization of those links as well as the usefulness of node splatting. All combinations of link visualization techniques are tested for different trajectory density levels. We used three types of tasks: tracing of paths, identification of longest links, and estimation of the density of trajectory clusters. Results are presented in the form of the statistical evaluation of task completion time, task solution accuracy, and two eye tracking metrics. These objective results are complemented by a summary of subjective feedback from the participants. The main result of our study is that tapered links perform very well. However, we discuss that equidistant comets and equidistant arrows are a good option to perceive direction information independent of zoom-level of the display.

Index Terms—User study, eye tracking, evaluation, trajectory visualization, node-link visualization, direction encoding, node splatting, halo rendering

1 Introduction

The fast technological progress allows scientist nowadays to use small and light-weight position tracking devices such as GPS to record time-dependent position information of different kinds of objects or living beings. The common way to visualizing such trajectories just uses a line-based representation, connecting the sample points along the trajectories. Without further information, the typical interpolation model relies on linear connection between sample points, yielding a piecewise linear representation of the trajectory. Such visualizations are, for example, used in animal movement ecology [23, 27] to get an overview of movement behavior, or spot clusters that could indicate nesting sites or hunting grounds.

Performing low-level tasks like following the path of a single trajectory comes with low perceptual and cognitive effort for the user as long as only short and simple trajectories are shown. However, performing path-oriented tasks for complex and dense data can become hard and error-prone. Therefore, there is potential for improved visualization techniques to increase the user's performance for such complex data sets. Figure 1 shows examples of different visualization techniques applied to data from animal movement ecology; here, a zoomed-in view of a large data set with the long-term GPS tracking of positions of oystercatcher birds is shown, provided by collaborators at the Institute for Biodiversity and Ecosystem Dynamics (IBED) at the University of Amsterdam. This figure illustrates that different visualization techniques provide different visual "flavors" and might show different effectiveness.

The goal of our paper is to evaluate and compare different trajectory visualization techniques. All of them follow the same fundamental principle of directed node-link diagrams: nodes represent sample points along the trajectories, directed links connect subsequent sample points. We have chosen to restrict ourselves to node-link diagrams because they follow the congruence principle for effective graphics [29]: the structure and contents of the diagram fit to the structure and content of the internal, mental representation that the visualization tries to achieve. Furthermore, other than more abstract visualizations, they lend themselves for direct overlay on geographic maps that serve as important spatial context for trajectories.

As concrete choices of visualization techniques, we evaluate the following methods for showing directed links: standard arrow, tapered, equidistant arrows, and equidistant comets. The first two are standard approaches known from node-link diagrams for directed graphs; the latter two are variants of existing techniques that we developed to show

- *The authors are with VISUS, University of Stuttgart. E-mail: {rudolf.netzel, michael.burch, weiskopf}@visus.uni-stuttgart.de.*

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.2014.2346420

links of largely varying length in a single diagram. The first technique simply puts a standard arrow at the end of a link, as most commonly used in node-link diagrams. Tapered links performed previously very well in node-link diagrams of abstract directed graphs [17], and their design is inspired by the human ability to perceive direction. Derived from the standard arrow, the equidistant-arrows technique consists of multiple arrows along a link that are placed equidistantly. Finally, the equidistant-comets technique places a "comet texture" at equidistant positions along a link to show direction information along the full length of the link. In addition to the link rendering style, we investigate the effect of rendering order for the halo visualization of the links; halos are used to visually separate crossing links, which often appear in dense and complex data sets. Finally, we test the effectiveness of node splatting for large data sets as an alternative approach to visualize point locations that provides visual feedback about the point density. All combinations of link visualization techniques are tested for two different levels of trajectory density.

Our comparison is based on a repeated-measures within-subjects user study with 25 participants, conducted in a controlled laboratory environment. We used task performance as measure of visualization effectiveness. As tasks, we chose three representative types: tracing of paths, identification of longest links, and estimation of the density of trajectory clusters. Task performance was measured by task completion time and accuracy. In addition, we set up our study as an eye tracking experiment to better understand the reasons for possible differences between visualization techniques. The eye tracking data was analyzed by using average fixation duration and saccade length as metrics.

The main result of our study is that tapered links perform well, even under different data set conditions and tasks. However, depending on the zoom level and variability of link length, the equidistant comets and arrows can show some benefits, too. Also the use of point splatting turned out to be beneficial. The contributions of this paper are:

- A quantitative user study to assess and compare node-link visualization techniques for trajectories.
- An additional eye tracking-based evaluation of the readability of these visualization techniques.
- Equidistant arrows and equidistant comets as two new variants of visualization techniques that we designed in the context of our study.

The user study compares different rendering styles for showing directed links. In addition, it explores the effect of the splatting of nodes and the depth sorting for halo rendering. To complement the objective results from the task performance and the eye tracking-based evaluation, we also include qualitative and subjective feedback from the participants of the study.

2 Related Work

Holten et al. [17, 18] conducted two studies concerning the direction encoding of links of node-link diagrams of directed graphs. In their first study, they tested six different link types: standard arrow, color gradient from green to red and light to dark and vice versa, curved links, and tapered links. Here, tapered links outperformed all other types in solution time as well as accuracy. In their extended study, they tested 15 link types including the previous ones. Additional types were glyph-based, or they used animation. Again, tapered links outperformed all non-animated types in solution time as well as accuracy, whereas animation further increased the accuracy, at the cost of a slightly increased completion time.

Like Holten et al., we also investigate the effectiveness of direction encoding of links in node-link diagrams. However, our study addresses different research questions in the following respects: First, we look into trajectory data instead of abstract directed graphs. Trajectories have node positions that have a geographic interpretation and, thus, cannot be changed by any layout algorithm. In contrast, node-link diagrams of abstract graphs typically follow aesthetic criteria for graph drawing [7, 13]. Therefore, the node-link visualization of trajectories has very different characteristics in the distribution of positions of nodes and lengths of links: both are much less evenly distributed than for abstract graph drawing. Another difference is that trajectories have one incoming and one outgoing link at the most but no arbitrary valence. Second, different tasks are relevant for trajectories. Third, with the restriction to geographic positions, the node-link visualization of trajectories tends to suffer from visual clutter and overdraw in dense or complex regions. Therefore, we investigate additional visualization techniques that are "orthogonal" to the choice of encoding of link direction and that were not part of the studies by Holten et al.: we examine the effect of depth sorting for halo rendering and the usefulness of node splatting. Fourth, we additionally use eye tracking to assess the visual reading characteristics.

Again for node-link diagrams of abstract graphs, there are other papers that address the readability of direction encodings in the presence of link crossings. Burch et al. [6] investigated in their study the readability of directed graphs shown with partially drawn links instead of links of full length. This approach reduces link intersections and can lead to shorter task completion times but at the cost of higher error rates. Bruckdorfer et al. [3] investigated the influence of different parameters in visualizations with partially drawn links. Jianu et al. [20] proposed color encoding of each link according to a closeness metric to address the problem of link crossings. Rusu et al. [25] introduced breaks in links if intersections occur, leading to an effect similar to partially drawn links. Since partially drawn links showed high error rates and were designed for the visualization of abstract graphs, we did not consider them as visualization technique. However, our halo rendering approach adopts the idea of breaking up links at crossings and, thus, incorporates the strategies from the above papers.

We chose node-link visualization because it is very common for representing spatial trajectories in fields like geographic information science [1] and movement ecology [23, 27]. The most common encoding of direction information uses arrows. Even with trajectory clustering and bundling, arrow-style rendering is often employed [14]. Recently, Janetzko et al. [19] have described a method to simplify trajectories and used tapered links for direction indication. We chose arrows and tapered links as the two representatives of traditional rendering techniques for trajectories, to be tested in our user study.

Especially for dense and large data sets of trajectories, there are approaches that create density fields of trajectories. By convolution of trajectories with a low-pass filter, a visualization can be achieved that highlights frequently visited areas [33]. Density maps can also be used to explore multivariate data of trajectories [26]. Similarly, edge splatting can be employed to construct a density field of links in a node-link diagram [5]. We do not use splatting or a density representation of links because they tend to obscure direction information in order to highlight high position density. However, we include splatting of nodes—similar to graph splatting [30]—to show the positions of samples along trajectories.

The encoding of direction information is not only important for node-link diagrams and trajectories, but also in other domains like flow visualization. Flow visualization often encodes direction by glyphs. Pilar et al. [24] investigated this approach in detail. Other techniques use texture-based methods [21] to visualize flow direction. Our comet texture is inspired by oriented line integral convolution (OLIC) [32] and follows the recommendation for stroke design in flow visualization [8]. However, it should be pointed out that flow visualization and our trajectory visualization show many important differences: streamlines, pathlines, etc. in flow visualization are typically smooth and curved, not piecewise linear; streamlines do not intersect, and pathlines do not have the intersection characteristics of trajectories; finally, sample positions along streamlines and pathlines are not visualized.

With this paper, we contribute a user study on visualization techniques for node-link diagrams of trajectories. We also contribute a user-based evaluation of splatting of nodes. In addition to the standard evaluation of task performance, our study comes with the evaluation of eye tracking characteristics. In general, there are only few previous eye tracking studies in information visualization. The following examples use eye tracking to test techniques that do not deal with trajectories or graphs: Goldberg et al. [11] described general aspects of eye tracking and performed a comparison between linear and radial plots. Burch et al. [4] investigated differences of traditional, orthogonal, and radial node-link tree layouts. Garlandini et al. [9] tested different visual variables for the visualization of geographic maps, and performed an empiric evaluation of recorded eye tracking data.

3 Visualization Techniques

In this section, we describe the variants of visualization techniques compared in our user study. First, the four types of rendering directed links are discussed, followed by an explanation of halo rendering with depth sorting. Finally, we describe different ways of visualizing nodes.

Throughout this paper, we will use abbreviations for the different options of link rendering, sorting, and node visualization. These abbreviations are summarized in Table 1.

3.1 Link Types

The simplest and most common type of link visualization uses the standard arrow (A). Here, the end arrow is rendered in order to encode direction information, see Figure 2(a). Another representation is based on tapered links (T): the direction is encoded by a prolonged arrow between the start position (thick end) and the end position (arrow tip), see Figure 2(b). We chose to include standard arrows because of their widespread use, and tapered links due to their very good performance in previous studies on node-link diagrams of abstract graphs [17, 18].

The node-link visualization of spatial trajectories comes with additional challenges compared to the visualization of abstract graphs. In particular, there can be large variations in link length. Furthermore, data sets can become large and complex, requiring zooming for detailed visual analysis. Unfortunately, standard arrows and tapered links are not well suited for very long links. As illustrated in Figures 1 and 3, direction information might be completely missing for standard arrows, if neither their start nor end node is shown (due to zooming). Similarly, the thickness gradient for tapered links might become very small, making it hard to perceive direction information. To address this problem, we developed two new variants of link representation. The technique that we call equidistant arrows (EA) is an extension of the standard arrow: it places multiple arrows along a link at equidistant positions, see Figure 2(c). The distance between arrows is adapted to the zoom level and identical for all links of a zoom level—except for the quantization effect introduced by placing an integer number of intervals along each link.

Finally, we developed the equidistant comets (C) as another technique that is suitable for very long links, see Figure 2(d). The strategy is the same as for the equidistant arrows. By using a repeating direction indicator along a link, it is easy to perceive the direction at any position or scale. The repeated texture is inspired by the metaphor of a comet: the head of the comet is large, and it is followed by a tail that is becoming narrower. A single comet texture is similar to a scaled-down version of the tapered link. The main difference is that direction is essentially reversed: the direction goes from the narrow tail of the comet stroke toward the large end. We chose this opposite encoding scheme because it is inspired by stroke-based rendering in flow visualization: Halley [12] used similar strokes as early as in 1686. The visual design with the thick head is also recommended by Bertin [2] and confirmed by a user study by Fowler and Ware [8]. For more background of the perception of flow visualization, we refer to Ware [31]. We chose the comet texture with the opposite direction encoding of tapered links to extend the design space of our user study compared to Holten et al. and because the equidistant placement along links resembles the visual signature of flow visualization.

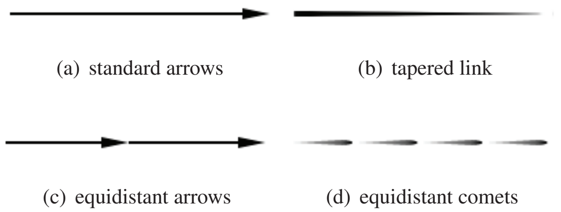
For dense data sets, both arrow rendering methods (A, EA) would suffer from visual clutter and overdraw: important structures like link crossings or other arrows might become occluded. To address this issue, we only render those arrows that do not occlude important structures. This is a trade off because missing arrows also imply the absence of direction information.

Table 1. Abbreviations for the visualization methods of this paper.

N	No direction encoding
A	Direction encoding with standard arrow
EA	Direction encoding with equidistant arrows
C	Direction encoding with equidistant comets
T	Direction encoding with tapered links
RND	Random edge sorting
SF	Shorter links on top of longer links
LF	Longer links on top of shorter links
SP	Splats for point visualization of nodes
NSP	No splats for point visualization of nodes

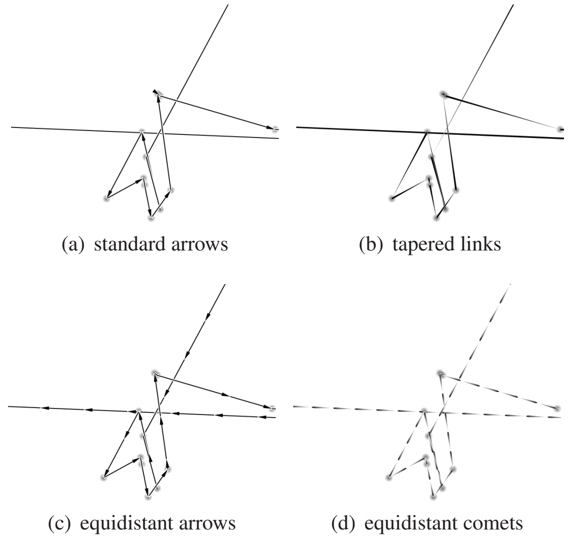
Table 2. Abbreviations for different sizes of data sets used in our study.

D1	Low data density (3 trajectories with 30 points each)
D2	High data density (3 trajectories with 60 points each)


(a) standard arrows (b) tapered link

(c) equidistant arrows (d) equidistant comets

Fig. 2. Types of link visualization.


(a) standard arrows (b) tapered links

(c) equidistant arrows (d) equidistant comets

Fig. 3. Readability of link direction with different types of link visualization, for a zoomed-in view.

3.2 Depth Sorting

We employ halo rendering of links to improve the perception of them at crossings. Therefore, the rendering order plays an important role: links in the background are "cut-through" by the halos of foreground links. To this end, we studied the effectiveness of different depth sorting methods: we compared random ordering (RND) to ordered rendering. The order, respectively the depth, reflects the importance of links. More important links will be rendered over less important ones. Without further information, we are restricted to data from the spatial configuration of the trajectories to control the depth sorting. Therefore, we used the total link length between nodes as importance measure. Here, we evaluate an ordering that puts the shortest links in the foreground (SF) and one that places the longest links in the foreground (LF). Figure 4 illustrates the effect of depth sorting for halo rendering.

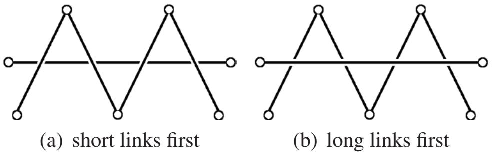

(a) short links first (b) long links first

Fig. 4. Effects of the ordering in combination with a halo rendering.

3.3 Node Visualization

The nodes of a node-link diagram are shown by some kind of point visualization. Typically, these points are rendered as non-filled circles, filled dots, or they are not explicitly shown (only implicitly as ends of links)—the latter being abbreviated as NSP. However, for large and dense data sets, opaque rendering of circles or dots would lead to much overdraw and clutter. Therefore, we include point splatting (SP), similar to the splatting of vertices in graph splatting [30]. Here, each node is rendered as a circular splat with a radially decreasing intensity. Using additive blending to render each splat, results in a density field. Regions with many nodes that are very close to each other will lead to high values in the density field and thus can be identified easily. In contrast, using dots to indicate nodes would lead to much overdraw; in bad cases, a user might perceive a single dot although this dot might occlude dozens of other node representations. Figure 5 illustrates SP and NSP.

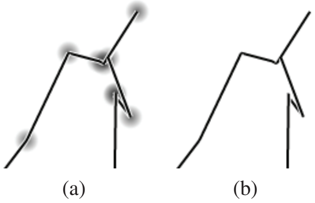

(a) (b)

Fig. 5. (a) Node visualization by splatting, (b) implicit node representation (indirectly through the end points of links).

4 Experiment

This section describes the design, hypotheses, tasks, stimuli, and further details of our user study.

4.1 Design

The user study follows a repeated-measures design. The study was conducted in a controlled laboratory environment; eye tracking information was recorded during the study. The study consisted of three independent parts. The first two parts were designed to compare the different rendering methods for links. Those two parts had the following within-subjects factors to represent the different visualization techniques: link type (A, EA, C, T) and sorting order (RND, SF or LF). In addition, two different data set sizes were tested for each combination of techniques: data density (D1, D2), see Section 4.4. The two first parts mainly differ in the tasks (see Section 4.3). The third part of the study was designed to evaluate the effect of node splatting. It had the following within-subjects factors: splatting or no splatting for point rendering (SP, NSP).

The three parts used the same dependent quantities that served as basis to assess the effectiveness: task solution accuracy, task completion time, average fixation duration, and average Euclidean saccade distance (the latter two were acquired through eye tracking).

4.2 Hypotheses

Our hypotheses were based on previous results from user studies on node-link diagrams of abstract graphs [17, 18] and our own impression and prior experience obtained from working with trajectory visualization for movement ecology:

H1: Tapered links (T) outperform the other link representations for tasks related to following trajectory paths.

H2: Equidistant comets (C) outperform the other link representations for tasks related to recognizing very long links.

H3: Depth sorting (SF) improves correctness of following trajectory paths in dense regions because short links are rendered on top of longer ones.

H4: Depth sorting (LF) improves correctness of tasks related to recognizing very long links because long links are rendered with guaranteed visibility.

H5: Point splatting (SP) improves the perception of clusters of high-density clusters of nodes because it provides a more accurate representation of node distribution than the visualization by the links alone.

Hypotheses H1–H4 refer to the first two parts of the study (related to link rendering), hypothesis H5 refers to the third part of the study (related to node splatting).

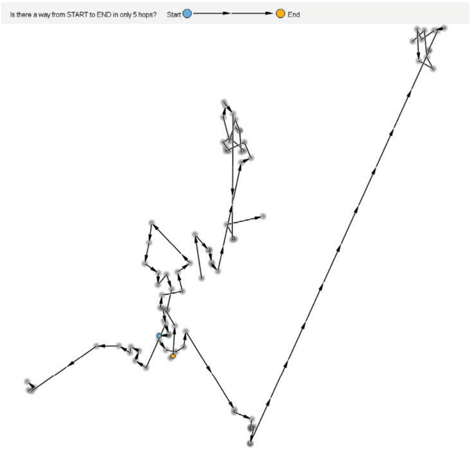
4.3 Tasks

We designed two tasks (Task 1 and Task 2) to compare the techniques for link visualization and one task (Task 3) to evaluate the effect of node splatting. We chose those tasks based on similar tasks used in the visualization of abstract graphs [10, 22] and on tasks specific to the analysis of movement patterns [28]. While the tasks of our user study are presented as abstract tasks to the participants, they have concrete and realistic correspondences to examples in movement ecology: our tasks could be used to identify behavioral patterns like migration and foraging of animals. Nevertheless, the tasks are generic in the sense that they carry over to many other application domains.

Task 1 (Path Following): The first task was to follow the path of a trajectory from a given start point. In particular, we asked the participants to answer the following question for the shown stimulus: “Is there a way from start to end in only 5 hops?” The start and end position were color-coded (start=blue, end=orange) in a way they were easy to recognize. The dots for the two positions were shown alone—without the trajectory visualization—for 2 seconds before the full visualization was presented. In this way, no time for visual search of the two nodes was required, restricting the task to actually following the path of the trajectory. The participants were instructed to answer as correctly and fast as possible; they were told that there was no time limit (and thus no pressure of time). The participants provided their answer by pressing a respective key. The task completion time was measured as the temporal difference between pressing the key and the first appearance of the full visualization. Figure 6 shows the screenshot of an example of Task 1.

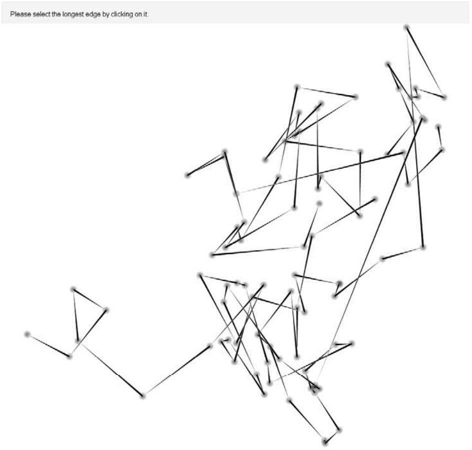
Task 2 (Longest Link): The second task was to identify the longest link. This task was chosen because it tests whether the full visualization can be read effectively, making it complementary to the path tracing of Task 1. In particular, we asked the participants: “Select the longest link by clicking on it.” Again, the participants were instructed to provide correct and fast answers, and that there was no time limit. The task completion time was measured as the temporal difference between mouse click and the first appearance of the visualization. Accuracy was measured by checking whether the mouse click was on the correct link, or not. To avoid unintended errors from slightly wrong mouse positions, we surrounded the longest link with a virtual rectangular bounding box. The width of the box was twice the link width. A click within the bounding box was recognized as correct answer. We instructed the participants that they did not click on the intersection with other links, to avoid ambiguity in the identification of the links. Figure 7 shows the screenshot of an example of Task 2.

Task 3 (Number of Nodes in Clusters): The third task was designed to test the visualization techniques for nodes. We were interested in how well participants could identify the overall structure of the spatial distribution of nodes. To test this, we asked the participants: “Select the cluster with the most elements by clicking.”



Is there a way from START to END in only 5 hops?

Fig. 6. Screenshot of an example of Task 1 (path following). Here, equidistant arrows (EA) with node splatting (SP) and no depth sorting (RND) are shown. The legend at the top contains the question and the description of start and end nodes.



Please select the longest edge by clicking on it.

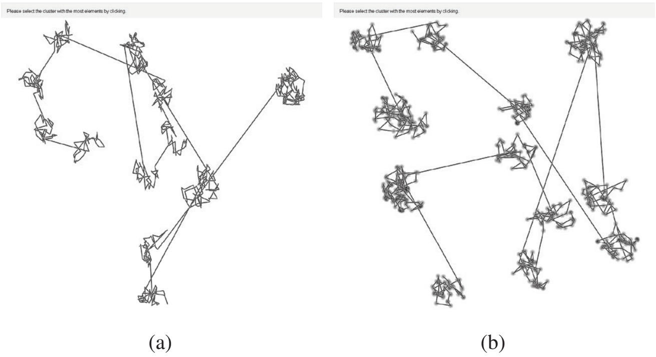
Fig. 7. Screenshot of an example of Task 2 (longest link). Tapered links (T) with node splatting (SP) and depth sorting (LF) is used. The legend at the top contains the question.

We encouraged the participants to make their decision according to the visual appearance of the clusters, not by explicitly counting nodes, because we were interested in how well the participants could obtain a rough impression of the distribution of nodes. Again, the participants were instructed to provide correct and fast answers, and that there was no time limit. The task completion time was measured as the temporal difference between mouse click and the first appearance of the visualization. Accuracy was measured by checking whether the mouse click was on the area of the correct cluster. Figure 8 shows two example screenshots for Task 3.

4.4 Stimuli

The visual stimuli were images rendered with the different variants of visualization techniques. The images were generated with 2x anti-aliasing and had a size of 1100 × 1100 pixels. We enabled halo rendering for all images. The line width was 5 pixels in the virtual high-resolution frame buffer of 2200 × 2200 pixels, i.e., 2.5 pixels in the final image.

The visualization techniques were applied to a generative data model to produce the visual stimuli. To generate realistic data, we used a Markov chain model based on real-world movement data from birds: the input data contained the trajectories of three oystercatcher birds with a total number of 16,434 GPS locations. From that data, we computed transition matrices for both the travel distance between GPS locations and the relative change in direction per location. Each matrix contained a probability distribution for the transition from one state to another, based on the training data. We used 20 states to encode the transitions, i.e., each matrix had a size of 20 × 20. The maximum distance between two locations in the data set was around 4,000 m, so each state covered around 200 m. For the change for movement direction, each state covered $360^\circ/20 = 18^\circ$. With this Markov chain model, we retain important characteristics of real trajectories from movement ecology, serving as a basis for a realistic user study.



Please select the cluster with the most elements by clicking.

Please select the cluster with the most elements by clicking.

(a) (b)

Fig. 8. Screenshots of examples of Task 3 (number of nodes in clusters). Left: no splatting (NSP), right: node splatting (SP).

The stimuli for Task 1 were produced from our generative data model directly. The stimuli were generated so that in 50% of the trials the correct answer for Task 1 was “yes”. We randomly chose the start position and selected an intermediate end position at a random distance of 2 to 5 hops (i.e., a position for which the participants were expected to answer “yes”). Then, with a probability of 0.5, the end position was changed by selecting the nearest position downward the trajectory for which the correct answer was “no”.

For Task 2, it was important that a participant was able to determine the longest link. Therefore, some difference in length was required to distinguish the longest from the second longest link, allowing participants to perform Task 2. We used a threshold of at least 8% length difference. We obtained this 8% threshold from a preliminary study (see Section 4.5).

For both tasks, two different data densities were used: D1 and D2, see Table 2. The total number of points for D1 was 90, distributed across three trajectories each containing 30 points. More than one trajectory was necessary to simulate an appropriate use case. For D2, the number of points on each trajectory was doubled to 60, resulting in a total number of 180 points.

In Task 3, participants should recognize the cluster containing the most GPS locations. In our preliminary study, we determined a threshold of 15% in the difference of number of locations between clusters. Therefore, the data set was generated so that the largest cluster had 15% more nodes than the second largest cluster, leading to a total number of 654 points distributed over 12 clusters. Each cluster had a radius of 200 m and a buffer zone of 50 m, in order to prevent visual merging of clusters. Within each cluster, the trajectories were generated with the Markov chain model. The overall trajectory was constructed by connecting clusters in random order.

4.5 Parameter Choices and Pilot Study

We conducted an informal small-scale study to set up the details of the parameters for the main study, in particular, the choice of thresholds discussed above. In this preliminary study, there were 10 volunteers who were different from the participants of the main study. Eye tracking was not used in the preliminary study.

To set the threshold for Task 2 we asked the participants to select the longest link within 5 different stimuli, shown in random order. The stimuli had varying difference in length between the longest link and the second longest link. We tested difference thresholds of 2%, 4%, 6%, 8%, and 10%. The stimuli were rendered using NSP for point indication, RND for link sorting, and N (standard lines without direction cues) as link type. There were three trajectories with between 30

Table 3. Summary of used methods and multipliers, describing the number of trials.

	Task 1		Task 2		Task 3	
Factor	Method	Multiplier	Method	Multiplier	Method	Multiplier
Depth sorting	(RND, SF)	2	(RND, LF)	2	(SF)	1
Density	(D1, D2)	2	(D1, D2)	2	-	1
Node rendering	(SP)	1	(SP)	1	(SP, NSP)	2
Repetitions		4		5		20
Trials per block		16		20		40
Link rendering	(A, EA, C, T)	4	(A, EA, C, T)	4	(N)	1
Trials per task		64		80		40
Participants	25					
Trials per task		1600		2000		1000

and 60 nodes, randomly chosen; this range of densities was used because it corresponds to the interval of densities defined by D1 and D2. At a threshold of 8%, 9 of 10 participants provided correct answers. Therefore, we assumed that this threshold is sufficient to allow for a fair identification of the longest link.

To determine the threshold for Task 3 we asked the participants to identify the cluster with the largest number of elements. There were 7 different test data sets with varying difference in the number of elements between the largest and the second largest cluster, ranging from 3% to 21% in steps of 3% difference. We used both NSP and SP for node rendering, and N for link rendering. For a difference of 15%, 9 of 10 participants provided correct answers for NSP rendering, and 8 of 10 for SP rendering. Therefore, we assumed that this threshold is sufficient to allow for a fair identification of the largest cluster.

Before the main experiment, we ran a pilot eye tracking study with another four participants. With the pilot study, we were able to refine our study; in particular, we found out that we had to reduce the number of repetitions for each participant to keep the average duration to just above one hour per participant. In detail, we reduced the number of trials per block of Task 2 from 40 to 20, and for Task 1 from 40 to 16. Since Task 1 was more complex than Task 2, the completion time of a trial of Task 1 was higher than that of Task 2. Therefore, we adjusted the trials per block of Task 1 to achieve an overall completion time comparable to Task 2; in this way, participants spent approximately the same amount of time using the same link type in Tasks 1 and 2. We did not have to reduce the number of trials per block of Task 3 because the completion time was not as high as for Tasks 1 and 2. See Table 3 for a detailed overview. Finally, the pilot study also allowed us to set up the number of participants required for the full study.

4.6 Environment Conditions and Technical Setup

Our study was conducted in a laboratory isolated from outside distractions. The room was artificially illuminated with dimmed lights. The laboratory space contained no distracting objects.

The study was conducted with a Windows PC driving a TFT screen with a resolution of 1920 × 1200 pixels. Eye tracking data was recorded by a Tobii T60 XL eye tracking system integrated into the TFT screen. The viewing distance was about 60 cm from the screen to allow for a good calibration of the eye tracking system. Key parameters for the analysis software of the eye tracker were set to a minimum of 10 pixels covering and a minimum of 30 ms fixation duration.

4.7 Participants

There were 25 participants (17 male, 8 female). Their age was between 19 and 66 years, with a median of 24 years. According to a test with a Snellen chart, all participants had normal or corrected-to-normal vision; according to the Ishihara test, all participants had normal color vision. Except for two participants, all were students of our university. 19 participants reported that they were familiar with node-link diagrams. The participants were compensated by €10 each.

4.8 Study Procedure

The within-subjects design comes with a combination of the following factors: link type (A, EA, C, T, N), depth sorting (RND, SF, LF), node visualization technique (SP, NSP), and density of the data set (D1, D2). The concrete combination of factors depends on the task. Table 3 summarizes the relationship between factors and tasks. For Task 1, we tested the depth sorting conditions RND and SF, but not LF, because SF is more helpful for this path following task than LF, especially in dense regions, where the participants have to follow mostly short links. In contrast, we tested RND against LF for Task 2, where good perception of long links is obviously beneficial. For Task 3, we used SF for depth ordering of links because it shows best the detailed structure of the clusters represented by short links. Furthermore, we used N for link rendering because it is the most "neutral" link type; thus, we avoided that a specific link type interferes with NSP or SP as point indication.

At the beginning of the study, there was an introduction to the study and the participants were given general instructions and explanations. Also, the vision tests were performed, and questionnaires about the background of the participant were filled in.

The main part of the study was split into three blocks: one block for each task. In this way, the participants could adapt to the different tasks, and cognitive load from frequent context changes was avoided. At the beginning of each task block, the participants were given detailed information about the task and instructions; they performed exercises on a printed version, followed by a practice phase in front of the screen. The order of the task blocks was counterbalanced between the subjects to compensate for learning effects and fatigue.

For Task 1 and Task 2, there were four internal blocks of link rendering techniques (A, EA, C, T). Again, the order of these internal blocks was counterbalanced between the subjects. Within each internal block, 16 trials (for Task 1) or 20 trials (for Task 2) were performed. Here, the order of depth sorting (RND, SF or LF), data density (D1, D2), and the repetitions of these combinations were randomized.

Task 3 only evaluated the effect of node splatting (SP, NSP). The rendering technique was blocked. The order of the internal blocks was counterbalanced between subjects. The 20 repetitions within each internal block were randomized.

Participants were asked to fill in a questionnaire after each task to obtain subjective feedback on the respective task and techniques. The average duration of the study was about 70 minutes per participant.

5 Results

In this section, we present the results of the statistical analysis of average completion time and correctness for all tasks. Time was measured in milliseconds; a log transformation was applied before statistical inference to conform to the normality assumption. Diagrams with timings are shown in units of seconds. We used the Shapiro Wilk test to check for normal distributions. The time analysis for all tasks was conducted with a repeated-measures ANOVA followed by pairwise t-tests for post-hoc analysis, if significance was indicated.

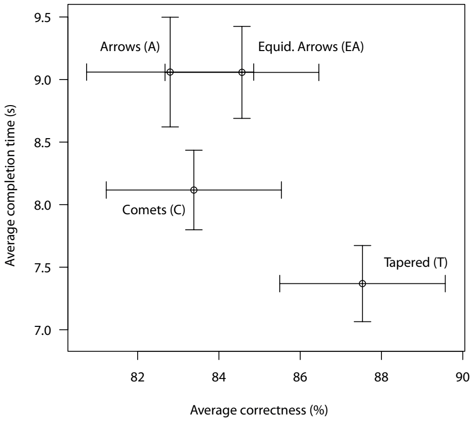
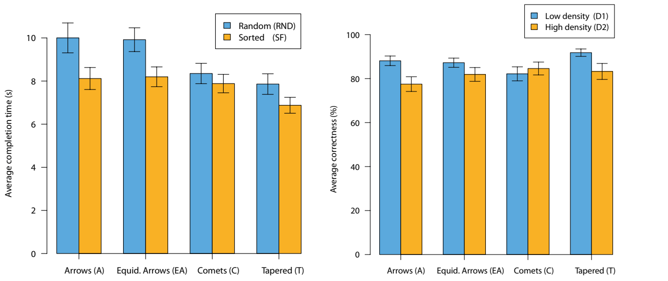


Fig. 9. Average correctness (x-axis) and average completion time (y-axis) for Task 1. Error bars show the standard error of means (SEM).



(a) Average time

(b) Average correctness

Fig. 10. Performance for Task 1 and the different link types: (a) average completion time with different sorting options, (b) average correctness for different data densities.

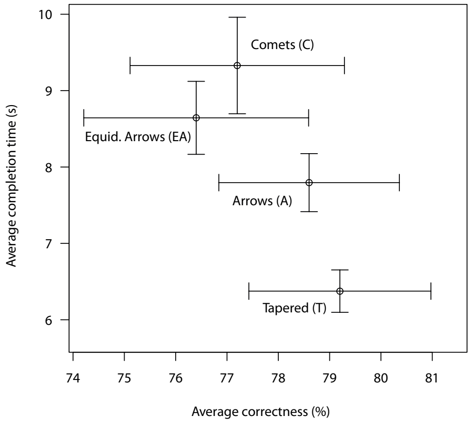
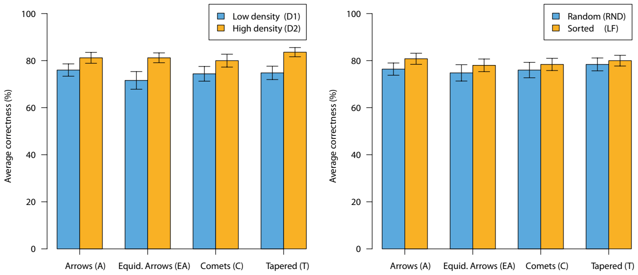


Fig. 11. Average correctness (x-axis) and average completion time (y-axis) for Task 2. Error bars show the standard error of the means (SEM).



(a) Different densities

(b) Different sorting

Fig. 12. Average correctness for Task 2 and the different link types: (a) with different densities, (b) with different sorting methods.

For the correctness, the normality assumption was not valid; therefore, we used for all tasks the non-parametric Friedman test as significance test, and the Wilcoxon signed rank test for post-hoc analysis. In the following, abbreviations will be used for the different rendering methods and data set sizes, as summarized in Tables 1 and 2. The results of the statistical analysis of completion time and correctness are discussed for each task separately.

The results for the ANOVA are in presented in the form $F(a,b) = c$, where a stands for the degrees of freedom (DoF), b for the residuals, and c for the calculated F value. A corresponding partial effect size is indicated through η_p^2. The non-parametric test results are presented in the form $\chi^2(a) = c$. In both cases, p is the corresponding p value. The Holm-Bonferroni method [15] was used to account for multiple comparisons.

Results for Task 1: With a mean correctness of 87.53% and a mean completion time of 7.37 s, T performed best. This is consistent with the results of Holten et al. [17]. C performed better than A and EA with respect to completion time, with a mean time of 8.12 s, and also better than A regarding the correctness with 83.38%. Figure 9 provides a summary of the descriptive statistical graphics of completion time and accuracy for the four link visualization techniques. ANOVA shows a significant effect of the link type ($F(3,72) = 9.45; p < 0.001; \eta_p^2 = 0.28$) and depth sorting ($F(1,24) = 22.27; p < 0.001; \eta_p^2 = 0.48$), concerning completion time. The post-hoc pairwise comparisons show significance for T compared to all other types ($p < 0.001$ for T-A and T-EA; $p = 0.046$ for T-C), and also between C and EA ($p = 0.011$). The influence of the depth sorting, leading to a decrease in time for each link type, is illustrated in Figure 10(a). The correctness analysis exhibited only a significance for the data set density ($\chi^2(1) = 13.5; p < 0.001$). Here, the correctness decreases for all link types, excluding C, see Figure 10(b). This indicates that it is slightly easier to handle C in regions with high density.

Results for Task 2: T performs best with an average completion time of 6.38 s and an average correctness of 79.2%, followed by A with 7.79 s and 78.6%. On average, participants needed the longest time (9.33 s) for C with a correctness of 77.2%, which is still better than the correctness of EA. ANOVA shows significance for the link types concerning the completion time ($F(3,72) = 6.82; p < 0.001; \eta_p^2 = 0.22$). The post-hoc comparison exhibits significant differences between T and all other types ($p < 0.001$), and also between C and A ($p = 0.012$). For correctness, there is a significance for the data density ($\chi^2(1) = 11.64; p < 0.001$) and depth sorting ($\chi^2(1) = 5.2609; p = 0.022$). Figure 12(a) illustrates that the correctness increases with the density summarized over all sorting options. Figure 12(b) shows that sorting also affects the correctness, summarized over all densities.

Results for Task 3: The analysis showed a significant effect for the correctness ($\chi^2(1) = 4.17; p = 0.041$), with an average correctness of 72.8% for SP and 65.4% for NSP; see Figure 13. While SP achieves a better correctness, it comes with a higher average completion time of 8.54 s compared to NSP with 8.04 s (not significant).

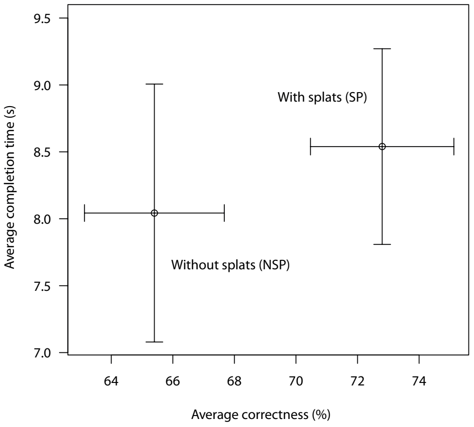


Fig. 13. Average correctness (x-axis) and average completion time (y-axis) for Task 3. Error bars show the standard error of the means (SEM).

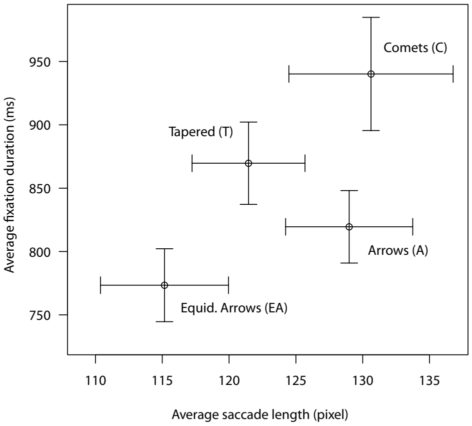


Fig. 14. Average saccade length (x-axis) and average fixation duration (y-axis) for Task 1. Error bars show the standard error of the means (SEM) for the eye tracking data.

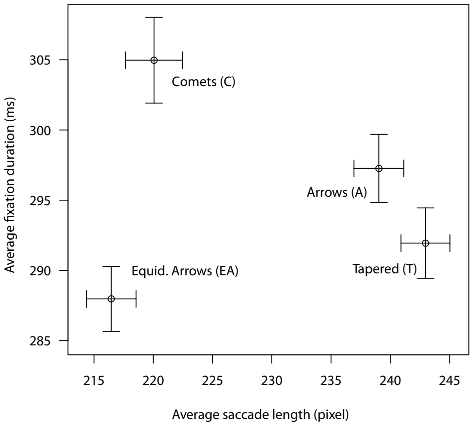
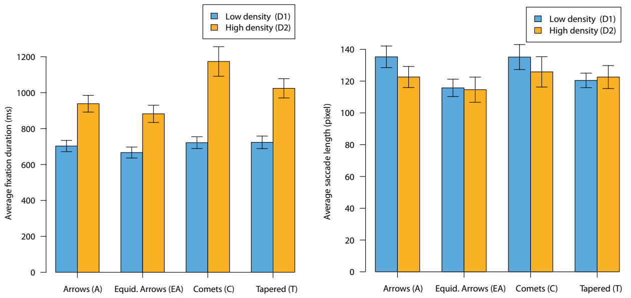
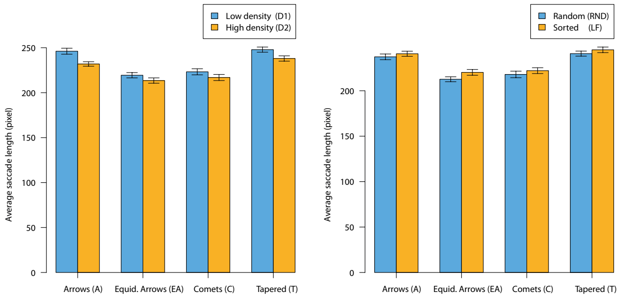


Fig. 16. Average saccade length (x-axis) and average fixation duration (y-axis) for Task 2. Error bars show the standard error of the means (SEM) for the eye tracking data.



(a) Average fixation duration (b) Average saccade length

Fig. 15. Average fixation duration and saccade length for each link type with different densities for Task 1.



(a) Different densities (b) Different sorting

Fig. 17. Average saccade length for Task 2 and the different link types: (a) with different densities, (b) with different sorting methods.

6 Eye Tracking Evaluation

In this section, we present the results of the eye tracking measurements. We performed a statistical analysis of the average fixation duration and average saccade length per displayed image, since these are common metrics in eye tracking. We will explain our choice of metrics and also their interpretation, but we refer to Holmqvist et al. [16] for a more detailed description of these eye tracking metrics and corresponding conducted experiments.

The first metric—average fixation duration—is an indicator for the cognitive processing depth. A high value typically means that a participant spent more time thinking about an area, for example, due to high visual complexity or absence of intuitiveness. Low values in a restricted area can be the result of stress, according to Holmqvist et al. (chapter 11.4.2; page 383). The second metric—the average saccade length or saccade amplitude—is often used to quantify similarity. Long saccade length can be interpreted as explorative eye movement, whereas short saccade lengths may occur when the task difficulty increases as short eye movements are used to collect information from a restricted area to support the current cognitive process, according to Holmqvist et al. (chapter 10.2.1; page 313). In the following, we will identify visual stress by a combined interpretation of the two metrics: short average saccade length and low fixation duration indicate a high level of stress. The perceptual and cognitive interpretation of these eye tracking metrics should be used with caution because of the complex perceptual and cognitive processes involved. However, these metrics add a useful view on our user study.

A common characteristic of eye tracking metrics, including the ones that we employ, is that they do not exhibit normal distribution. Therefore, we used the non-parametric Friedman test as significance test and the Wilcoxon signed rank test for a post-hoc analysis.

Eye Tracking Evaluation for Task 1: Figure 14 shows an overview of the duration and the saccade length of all link types. C has here both the longest average saccade length of 130.62 pixels and the highest average fixation duration of 940.15 ms. For EA, the smallest values for these parameters are obtained: 115.16 pixels and 773.41 ms. The statistical analysis shows a significance in duration for the density ($\chi^2(1) = 25; p < 0.001$). This influence—increasing duration with increasing data set size—is shown in more detail in Figure 15(a). For the saccade length, there is a significance for the density ($\chi^2(1) = 6.67; p = 0.009$). A decline of the saccade length is depicted in Figure 15(b). In both cases, this is consistent with our expectations that a higher data density would lead to a higher average duration and a reduced average saccade length, since more complex images may need more attention (longer duration) and have to be investigated in more detail (shorter saccades).

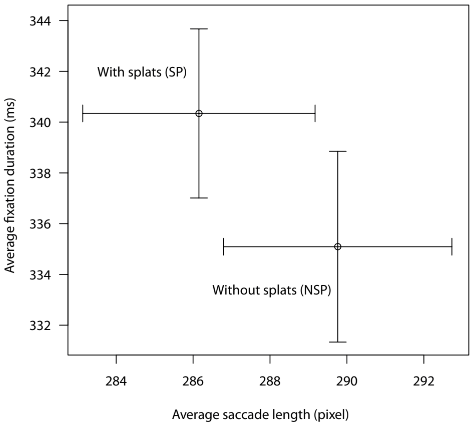


Fig. 18. Average saccade length (x-axis) and average fixation duration (y-axis) of Task 3. Error bars show the standard error of the means (SEM) for the eye tracking data.

Eye Tracking Evaluation for Task 2: In this task, again, the smallest values for the average duration and saccade length were measured for EA (216.46 pixels and 287.97 ms). We obtained the longest duration for C (220.06 pixels with 304.97 ms) and the longest saccade length for T (242.96 pixels with 291.94 ms). Figure 16 shows an overview of the eye tracking metrics. The statistical analysis showed only significance for the average saccade length. Here, all three factors were affected: link types ($\chi^2(3) = 35.35; p < 0.001$), density ($\chi^2(1) = 9; p = 0.003$), and depth sorting ($\chi^2(1) = 11.56; p < 0.001$).

The post-hoc analysis for the link types reveals significance between T-EA ($p < 0.001$), T-C ($p < 0.001$), C-A ($p = 0.003$), and EA-A ($p < 0.001$). The influence of density and sorting for each link type is shown in Figures 17(a) and 17(b). According to these plots, the average saccade length decreases with increasing density, while the sorting has the opposite effect. This may be interpreted as follows: increasing the data density also increases the visual stress resulting in shorter saccades, whereas the opposite holds for the sorting.

Eye Tracking Evaluation for Task 3: On average, NSP achieved a slightly shorter fixation duration of 335.09 ms and also a slightly longer saccade length of 289.76 pixels compared to SP with 340.34 ms and 286.15 pixels. This might vaguely indicate that participants needed more attention investigating clusters while SP was used. However, the statistical analysis showed no significance. Figure 18 provides the respective statistical graphics.

7 Participant Feedback

In this section, we present feedback obtained from questionnaires that the participants filled in after they finished a task. The link types were ranked on a Likert scale from 0 (not helpful) to 5 (very helpful); we did not apply any statistical tests to those data obtained. Furthermore, participants provided free-text comments.

Feedback for Task 1: T was voted as the most helpful link type followed by A, C, and EA. The average ratings were: 3.88 (T), 3.40 (A), 3.28 (C), 2.20 (EA). The comments for this task can be condensed into four statements: Hiding arrows using A and EA, if important structures would be occluded, made sense to all participants. However, for this task, it was confusing because direction information was missing in dense regions. This indicates that the correctness could have suffered because the participants started guessing. In contrast, the completion time might have been increased because it takes longer to come to a decision. Visualization with C and T can be recognized very well in dense regions; this is especially true for T because it is very simple. These two methods are also reported to be pleasant, since they guide the eyes. The direction encoding with C was not intuitive for some participants.

Feedback for Task 2: The vote order of link types is identical to that for Task 1. Average ratings were: 3.84 (T), 3.36 (A), 3.16 (C), 2.48 (EA). The comments indicate here that the missing arrows were regarded as negative, because there was not always an arrow at the end of a line. Also it was possible to count the arrows of EA (when they were there) or the comets of C. This might have led to increased correctness, but also increase the completion time. Again, visualizations with C and T could be recognized very well in dense regions.

Feedback for Task 3: SP was voted as more helpful then NSP with an average of 3.24 over 2.40. The participants commented that the splats used in SP made it hard to decide between two similar clusters because there is nearly no difference in brightness. However, the SP enabled them to identify locations within a cluster where several points are very close to each other. Some had the feeling that they solved the task faster when NSP was used. All participants commented that this task was in general hard to solve.

8 Discussion

One result of our study is that T performed better than the other link types for Task 1, with the highest correctness and the lowest completion time (Figure 9), confirming H1. We could also confirm H3: depth sorting led to a reduction of the solution time (Figure 10(a)). In addition, the evaluation of the eye tracking data revealed that A had the highest average saccade length and the shortest average fixation duration. This may be interpreted as intuitive and efficient usage of A (Figure 14)—with less of short hectic eye movement, exhibiting less visual stress compared to the other methods.

For Task 2, T again performed best, concerning the solution accuracy and time, as depicted in Figure 11. Therefore, we have to reject H2. From Figure 11, we can see that the group (C, EA) with repeating directional cues is outperformed by the techniques A and T, which come with only one cue (an arrow at the end or a prolonged arrow). Figure 16 shows that the techniques A and T have similar eye tracking characteristics in terms of average fixation duration and saccade length. T led to the longest average saccade length and the lowest fixation duration, compared to the other link types; this can be interpreted as an indicator for high efficiency. The high fixation duration and short saccade length for C and EA can be explained by the fact that participants counted the comets or arrows in order to estimate and compare the length of links. Furthermore, we can confirm H4 because sorting leads to higher accuracy, which is true for any link type (Figure 12(b)).

Using SP as point visualization in Task 3 reveals an increase of both accuracy and completion time compared to using NSP (Figure 13). The eye tracking evaluation showed a lower average fixation duration and a longer saccade length for NSP (Figure 18). According to this result, participants might have investigated clusters more carefully when SP was used, which could explain the higher average solution correctness and increased solution time, shown in Figure 13. Since we value accuracy higher than response time, we would argue that our findings support H5.

Overall, T performed very well in all tasks, followed by A. This gives evidence that participants can handle simple link types quite well. Nevertheless, the results for solution accuracy and completion time for C and EA were similar to those of T and A. Also, the participants stated that they liked the equidistant comets. Therefore, the equidistant comets can be recommended for trajectory visualization in large and complex data sets. We base our reasoning on the observations described in Section 3.1 and illustrated in Figure 3: For such data sets, detailed exploration requires frequent zooming. However, zooming affects the readability of direction information in T and A. In fact, that information might be completely missing is such visualizations. In contrast, C and EA are not affected by zooming, which makes it possible to interpret the direction of a link independent of the zoom level. It should be pointed out that we did not include stimuli with zooming because the disadvantages of T and A are obvious from construction, and we did not have to confirm them by the user study.

9 Conclusion

In this paper, we conducted an eye tracking study that compared different visualization methods for long, dense, and complex piecewise linear trajectories. We used the following representations of directed links: standard arrow (A), tapered links (T), equidistant arrows (EA), and equidistant comets (C). Furthermore, we have investigated the effect of rendering order (SF and LF) for the halo visualization of these links, also considering different levels of density (D1 and D2). Finally, we have evaluated the effectiveness of node splatting. Participants in our study were asked to perform three tasks: tracing of paths, identification of longest links, and estimation of the density of trajectory clusters. The tasks were meant to be overview tasks, therefore participants were not provided with interaction techniques like zooming.

According to the results of this study, T performed very well, followed by A, for tracing of paths and identifying longest links. We also have shown that these two link types can lose their ability to encode direction, if interactions, such as zooming, are possible. In this case, C and EA benefit from equidistantly placed direction information on a link, present at different zoom levels. We also were able to confirm that using point splatting increases accuracy.

Acknowledgments

This work was funded by Deutsche Forschungsgemeinschaft (DFG) within project WE 2836/4-1. We thank Emiel van Loon (University of Amsterdam) for the oystercatcher GPS data, and the participants for having attended our study. Many thanks to all reviewers for their constructive comments.

REFERENCES

[1] G. Andrienko, N. Andrienko, P. Bak, D. Keim, and S. Wrobel. *Visual Analytics of Movement*. Springer, 2013.

[2] J. Bertin. *Semiology of Graphics*. University of Wisconsin Press, 1983.

[3] T. Bruckdorfer, S. Cornelsen, C. Gutwenger, M. Kaufmann, F. Montecchiani, M. Nöllenburg, and A. Wolff. Progress on partial edge drawings. In *Proceedings of the International Conference on Graph Drawing*, pages 67–78, 2013.

[4] M. Burch, N. Konevtsova, J. Heinrich, M. Höferlin, and D. Weiskopf. Evaluation of traditional, orthogonal, and radial tree diagrams by an eye tracking study. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2440–2448, 2011.

[5] M. Burch, C. Vehlow, F. Beck, S. Diehl, and D. Weiskopf. Parallel edge splatting for scalable dynamic graph visualization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2344–2353, 2011.

[6] M. Burch, C. Vehlow, N. Konevtsova, and D. Weiskopf. Evaluating partially drawn links for directed graph edges. In *Proceedings of the International Conference on Graph Drawing*, pages 226–237, 2012.

[7] G. di Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, 1999.

[8] D. Fowler and C. Ware. Strokes for representing univariate vector field maps. In *Proceedings of Graphics Interface*, pages 249–253, 1989.

[9] S. Garlandini and S. I. Fabrikant. Evaluating the effectiveness and efficiency of visual variables for geographic information visualization. In *Proceedings of the International Conference on Spatial Information Theory*, pages 195–211, 2009.

[10] M. Ghoniem, J. Fekete, and P. Castagliola. A comparison of the readability of graphs using node-link and matrix-based representations. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 17–24, 2004.

[11] J. Goldberg and J. Helfman. Eye tracking for visualization evaluation: Reading values on linear versus radial graphs. *Information Visualization*, 10(3):182–195, 2011.

[12] E. Halley. An Historical Account of the Trade Winds, and Monsoons, observable in the Seas between and near the Tropicks, with an attempt to Assign the Phisical Cause of the said Winds. *Philosophical Transactions*, 16:153–168, 1686/1692.

[13] I. Herman, G. Melançon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):24–43, 2000.

[14] M. Höferlin, B. Höferlin, G. Heidemann, and D. Weiskopf. Interactive schematic summaries for faceted exploration of surveillance video. *IEEE Transactions on Multimedia*, 15(4):908–920, 2013.

[15] S. Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6:65–70, 1979.

[16] K. Holmqvist, M. Nyström, R. Andersson, R. Dewhurst, J. Halszka, and J. van de Weijer. *Eye Tracking: A Comprehensive Guide to Methods and Measures*. Oxford University Press, 2011.

[17] D. Holten, P. Isenberg, J. van Wijk, and J. Fekete. An extended evaluation of the readability of tapered, animated, and textured directed-edge representations in node-link graphs. In *Proceedings of the IEEE Pacific Visualization Symposium*, pages 195–202, 2011.

[18] D. Holten and J. J. van Wijk. A user study on visualizing directed edges in graphs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 2299–2308, 2009.

[19] H. Janetzko, D. Jäckle, O. Deussen, and D. Keim. Visual abstraction of complex motion patterns. In *Proceedings of S&T/SPIE Electronic Imaging*, volume 9017, pages 90170J–90170J–12, 2013.

[20] R. Jianu, A. Rusu, A. Fabian, and D. Laidlaw. A coloring solution to the edge crossing problem. In *Proceedings of the Conference on Information Visualisation*, pages 691–696, 2009.

[21] R. S. Laramee, H. Hauser, H. Doleisch, B. Vrolijk, F. H. Post, and D. Weiskopf. The state of the art in flow visualization: Dense and texture-based techniques. *Computer Graphics Forum*, 23(2):203–222, 2004.

[22] B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete, and N. Henry. Task taxonomy for graph visualization. In *Proceedings of BELIV*, pages 1–5, 2006.

[23] R. Nathan and L. Giuggioli. A milestone for movement ecology research. *Movement Ecology*, 1:1–3, 2013.

[24] D. Pilar and C. Ware. Representing flow patterns by using streamlines with glyphs. *IEEE Transactions on Visualization and Computer Graphics*, 19(8):1331–1341, 2013.

[25] A. Rusu, A. Fabian, R. Jianu, and A. Rusu. Using the Gestalt principle of closure to alleviate the edge crossing problem in graph drawings. In *Proceedings of the Conference on Information Visualisation*, pages 488–493, 2011.

[26] R. Scheepens, N. Willems, H. van de Wetering, and J. van Wijk. Interactive visualization of multivariate trajectory data with density maps. In *Proceedings of the IEEE Pacific Visualization Symposium*, pages 147–154, 2011.

[27] J. Shamoun-Baranes, E. van Loon, R. Purves, B. Speckmann, D. Weiskopf, and C. Camphuysen. Analysis and visualization of animal movement. *Biology Letters*, 8(1):6–9, 2012.

[28] D. Somayeh, W. Robert, and A.-K. Lautenschütz. Towards a taxonomy of movement patterns. *Information Visualization*, 7(3-4):240–252, 2008.

[29] B. Tversky, J. B. Morrison, and M. Betrancourt. Animation: can it facilitate? *International Journal of Human-Computer Studies*, 57(4):247–262, 2002.

[30] R. Van Liere and W. de Leeuw. GraphSplatting: visualizing graphs as continuous fields. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):206–212, 2003.

[31] C. Ware. Toward a perceptual theory of flow visualization. *IEEE Computer Graphics and Applications*, 28(2):6–11, 2008.

[32] R. Wegenkittl, E. Gröller, and W. Purgathofer. Animating flowfields: Rendering of oriented line integral convolution. In *Proceedings of Computer Animation*, pages 15–21, 1997.

[33] N. Willems, H. van de Wetering, and J. J. van Wijk. Visualization of vessel movements. *Computer Graphics Forum*, 28:959–966, 2009.