

EVALUATING THE USABILITY OF 1D FISHEYE LISTS

Christoph Wimmer, Martin Tomitsch, Thomas Grechenig
*Research Group for Industrial Software (INSO), Vienna University of Technology
Wiedner Hauptstr. 76 / 2 / 2, A- 1040 Vienna, Austria*

ABSTRACT

Fisheye views represent a promising solution to the problem of visualizing and interacting with large amounts of data. Such lists often require tedious scrolling, which can be cumbersome for users. In a controlled experiment we compared performance and user preferences of a fisheye list and a traditional list with a scrollbar for selection and drag & drop tasks. Users performed significantly better using the fisheye list for drag & drop tasks, whereas no significant differences were found with regard to selection tasks. In terms of satisfaction, users generally preferred the fisheye list to the standard list. A folder view with alphabetically grouped items served as a baseline condition and performed best overall.

KEYWORDS

Fisheye views, fisheye lists, drag and drop, interaction techniques

1. INTRODUCTION

Fisheye views provide a balance between the global overview of large structures and local detail within these structures. Local areas of interest are shown in great detail to make interaction with them feasible, while other areas are shown in less detail to always give an overview of the global context.

Since the introduction of fisheye views in user interfaces by Furnas (1982) for providing a balance between focus and context, their applicability to various user interface design problems has been intensely examined yielding mixed results. Bederson's (2000) fisheye menus introduced the application of a one-dimensional visual fisheye distortion for drop-down menus. A preliminary study showed that users preferred fisheye menus for browsing tasks (Bederson, 2000). Hornbæk and Hertzum (2007) further explored the concept of fisheye menus and presented a user study on variations of fisheye menus. Their empirical results suggested that the hierarchical menu was superior to the fisheye menus in respect to accuracy and selection time for finding known items. However, for browsing tasks all menu types performed the same.

This paper presents an evaluation of the applicability of visual fisheye distortion techniques for embodiments of lists other than drop-down menus. List widgets are often used for displaying long lists of data items, containers or folders. Thus, these widgets typically involve two types of tasks – item selection and drag & drop, whereas drop-down menus generally only facilitate selection tasks. The focus of our study was on one-dimensional lists, which are often used in applications or contexts that do not allow the employment of hierarchical structures. The goal of our user study was to evaluate the usability of a one-dimensional visual fisheye distortion for typical desktop user interfaces such as a list widget.

2. METHOD

To evaluate the usability of our presented graphical user interface (GUI) widget, we compared the performance and user preference of our fisheye list implementation with a traditional list with scrollbar. The fisheye list enables task completion in one continuous movement. In contrast, in traditional lists users have to perform a series of mouse movements and clicks in order to fulfill the selection or drag & drop task. For drag & drop tasks they can use auto-scroll functionality, however, this also increases task completion time. Therefore we expected the fisheye list to perform better or at least the same as the list.

A hierarchical folder view served as a baseline condition in our experiment. Items were alphabetically grouped in folders for this view. We anticipated that the folder view would be superior to both the list and fisheye list, because the hierarchical organization of data allows faster selection and better orientation.

2.1 Participants

Ten volunteer participants (3 female, 8 right-handed) were recruited for our study. The age of the participants was between 20 and 25 years ($M=23.6$, $SD=2.0$). All participants had several years of experience in computer usage and were therefore proficient in mouse usage and familiar with typical mouse interactions in a desktop computing environment. All participants had extensive experience in using lists and folder views as part of GUIs, but had no prior experience in using fisheye views.

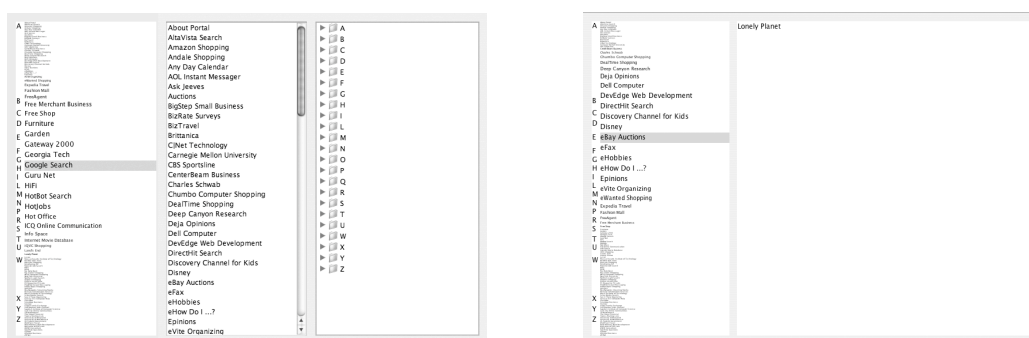
2.2. Apparatus

We developed a test application using Java including all three different widgets for our experiment, namely: our own implementation of a fisheye list, a traditional list with scrollbars, and a folder view (Figure 1). In the test application the list widgets are presented on the left hand side of the window. The basic layout of our test interfaces mimics typical desktop applications such as mail clients or media players in widespread use. This design was chosen to evoke familiarity in the study participants and ensure the applicability of our test results to typical desktop applications already in practical use.

The fisheye list was implemented as a list widget utilizing a one-dimensional visual fisheye distortion for visualization. Our design and implementation of a fisheye list was based on fisheye menus (Bederson, 2000), using a similar type of visualization and features such as fixed alphabetic indices. However, we omitted the focus lock mode because its usage turned out to be not intuitive for users in Bederson's previous study. The fisheye list shows list items in the focus region at full size while shrinking list items outside the focus region. This distortion effect is applied dynamically depending on the mouse pointer position. The fisheye list displays all of the list items in a single, completely visible window of fixed size with no need to scroll. The traditional list with scrollbars offers auto-scroll functionality and the folder view provides auto-expand functionality when the focus is moved over a node. Items are grouped alphabetically in the folder view.

As content for the widgets we used the titles of 100 popular websites as our data pool. The data pool is based on one of the pools used by Bederson (2000) for his fisheye menu implementation. All tests were done on a laptop computer with no peripherals attached except for a standard optical mouse as an input device. The resolution of the laptop display (13.3" widescreen) was 1280 x 800 pixels. The widgets were all equal in size (240 x 545 pixels). No additional windows were visible on the laptop screen during testing.

Figure 1. Three variations of widgets used for our experiment – fisheye list, traditional list with scrollbar and folder view (left). Layout of the test interface (right)



2.3. Task and Procedure

We used two typical mouse interaction tasks to evaluate our widgets: selection and drag & drop. In the beginning of both tasks a single item was displayed on the top right hand side of the widget. This item was randomly selected from our data pool. The selection tasks consisted of selecting the corresponding item from

the widget using a mouse click. The drag & drop tasks consisted of dragging the actionable item from the right into the widget on the left hand side and dropping it onto the corresponding item.

We conducted the tests in a quiet environment to minimize the effect of disturbances on our study. The experiment started with an explanation of its purpose, followed by a survey of each participant to learn more about their general level of computer proficiency. After an oral introduction of the test application, the different widget types and tasks at hand, the tests started with a training session for every testing condition. Here all participants performed each task 10 times using every widget to familiarize themselves with the interface and testing procedure. After these training sessions, the participants were instructed to complete the tasks as quickly as possible without making errors. After the tests, participants were asked to fill out a post-test survey for gathering feedback about subjective user satisfaction and preferences regarding the three evaluated interface types. Subjects were asked which interface type they liked best and least. They further had to rank each of the three interface types on a 9-point semantic-differential scale according to 7 characteristics (see Table 2). Finally, a post-test interview allowed room for additional comments.

2.4. Design

The experiment had two factors: widget (fisheye list, list, folder view) and task (selection, drag & drop). For each widget, participants conducted the given task 30 times. Using a within-subject design, each participant had to complete the test for all 6 testing conditions. The sequence of the testing conditions per user was counterbalanced to avoid learning effects. To measure performance and accuracy, we used task completion time and error rate respectively. Both measures represented the dependent variables in our experiment and were automatically logged by the test application. We defined errors as selection of the wrong item in the selection task as well as dropping the actionable item on the wrong item in the widget in the drag & drop task. We calculated pair-wise comparisons using a paired samples t test. User preference rankings were analyzed using Wilcoxon's signed ranks test.

3. RESULTS

We performed the experiment to look for effects of widgets on task performance. Since differences between selection and drag & drop were expected, the analysis was performed individually for each task. The mean task completion time per widget was calculated for each participant. Task completion times for erroneous tasks were discarded from this calculation.

3.1. Completion Time

Based on mean task completion times per participant, we calculated the mean task completion time per widget as shown in Table 1. Pair-wise comparisons of widget showed a significant difference between task completion time of folder view and both fisheye list ($t(9)=2.84$, $p<.05$) and list ($t(9)=4.04$, $p<.01$). Thus, folder view was significantly faster than fisheye list and list with regard to the selection tasks. No significant difference between fisheye list and list was found.

For the drag & drop task, pair-wise comparisons of interface types showed a significant difference between all three widgets. Folder view was significantly faster ($t(9)=2.46$, $p<.05$) than fisheye list and the traditional list ($t(9)=3.32$, $p<.01$). Hence, the folder view was also superior in the drag & drop task condition. However, fisheye list was significantly faster ($t(9)=2.58$, $p<.05$) than the traditional list. Therefore we concluded that the fisheye list outperforms the list with scrollbar for the drag & drop tasks.

Table 1. Means (and standard deviations) of task completion time in seconds. Bold denotes a significant difference between fisheye list and list

Task Type	Fisheye list	List	Folder View
Selection	4.05 (0.73)	4.15 (0.66)	3.57 (0.43)
Drag & Drop	5.02 (0.75)	5.62 (1.27)	4.66 (0.64)

3.2. Accuracy

Our analysis showed no significant effects on accuracy for any of the evaluated test conditions. With regard to selection tasks, participants achieved 98% accuracy for the list, 97% for the fisheye list, and 99% for the folder view. Regarding drag & drop tasks, participants achieved 99% accuracy using the fisheye list and the list, and 100% accuracy when using the folder view.

3.3. Preferences and Satisfaction

Table 2 shows participants' ratings of the three widgets. Comparing the fisheye list and the traditional list we found significant differences for several questions. Participants rated the fisheye list more wonderful ($z=2.32$, $p<.05$), more satisfying ($z=2.36$, $p<.05$), faster ($z=2.84$, $p<.001$), more fun ($z=2.38$, $p<.05$) and more pleasant ($z=2.03$, $p<.05$) than traditional lists.

Comparing the fisheye list with the folder view we found two significant differences. Participants reported on the one hand that the fisheye list was more fun ($z=2.14$, $p<.05$) to use than the folder view, but on the other hand that the folder view was easier ($z=1.98$, $p<.05$) to use than the fisheye list.

Comparing the traditional list with the folder view we found significant differences in our results for all questions except "hard - easy to learn". Thus, participants found the folder view more wonderful ($z=2.81$, $p<.001$), more satisfying ($z=2.83$, $p<.001$), easier ($z=2.03$, $p<.05$), faster ($z=2.72$, $p<.001$), more fun ($z=2.21$, $p<.05$) and more pleasant ($z=2.83$, $p<.001$) than traditional lists.

Additionally, participants were asked to name the widget they liked most and least. Eight participants preferred the folder view and 2 participants preferred the fisheye list. Eight participants said they liked the list interface least, and 2 participants stated they liked the fisheye list interface least. The variation in preferences of fisheye views was also shown in other publications (Bederson, 2000).

Table 2. Preference ranking of widgets. The scale is from 1 associated with the negative extreme to 9 associated with the positive extreme. Bold denotes a significant difference between fisheye list and list

Scale 1 – 9	Fisheye list	List	Folder View
Terrible – Wonderful	6.2 (1.4)	3.3 (1.6)	7.6 (1.3)
Frustrating – Satisfying	6.7 (1.6)	3.9 (1.4)	7.3 (1.6)
Difficult – Easy	7.0 (1.8)	7.4 (1.3)	8.5 (0.8)
Slow – Fast	6.7 (1.3)	3.6 (1.2)	7.4 (1.5)
Hard – Easy to learn	7.2 (2.1)	8.0 (1.2)	8.5 (0.7)
Boring – Fun	7.0 (1.7)	3.6 (1.3)	5.1 (1.2)
Annoying – Pleasant	6.0 (2.0)	3.4 (1.6)	7.3 (1.6)

4. DISCUSSION

In conclusion, we found a significant difference in task completion times between fisheye list and traditional list with respect to drag & drop tasks. This can be explained by the fact that the fisheye list allowed drag & drop operations to be performed in one continuous motion. In contrast, the traditional list often required scrolling to perform the drag & drop operation or using the auto-scroll feature when the target item was not shown in the initial viewport of the list. We observed that participants hardly ever used the auto-scroll feature for drag & drop tasks. In post-test interviews, they stated that they found it too cumbersome. Using the fisheye list widget, participants could perform the drag & drop operations without decomposing the movement into two components or resorting to the auto-scroll feature, leading to faster task completion times.

Preference rankings suggest that study participants generally preferred the fisheye list compared to the list with scrollbars. All participants found the fisheye list intuitive and easy to understand. Looking at individual preference rankings reveals that participant preferences and satisfaction regarding the fisheye list varied widely: two participants liked the fisheye view list best and two liked it least.

One apparent problem of our fisheye list observed by most study participants was the difficulty to precisely acquire targets at the end of the targeting process. This is a consequence of the decoupling of visual size and motor size, necessitated by the fact that all list elements must be selectable by pointer movement in the fixed space of the list (Bederson, 2000). A possible solution to this problem represents the focus lock mode, introduced by Bederson (2000) and further explored by Hornbæk and Hertzum (2007). Semantic pointing (Blanch et al., 2004) might be another promising solution to this problem. Implementing similar mechanisms in our test application could possibly improve the performance of the fisheye list.

One main problem of scrolling in traditional lists is its negative impact on spatial memory. In contrast, the position of list items never changes inside a fisheye list, which takes advantage of the user's spatial memory. A study on list revisitation has shown that users find common items more quickly when revisiting them repeatedly (Gutwin and Cockburn, 2006).

As anticipated, our results indicate that the folder view widget, which served as the baseline for our experiment, is superior to the fisheye list interface and the list interface for both selection and drag & drop tasks concerning task completion time. This is due to the fact that hierarchical organization of data items better supports navigation and orientation for users. However, our findings suggest that fisheye lists are a viable alternative to folder views when list items cannot be easily grouped into folders because of interface constraints or data-specific characteristics.

In accordance with our studies, Bederson (2000) and Hornbæk and Hertzum (2007) found out that users generally preferred hierarchical organization and alphabetical grouping of items. Bederson's preliminary study further confirmed our observation that the fisheye view was preferred over the one-dimensional list views. The empirical data collected by Hornbæk and Hertzum (2007) showed that the hierarchical folder view performed better than all variations of fisheye menus, especially for finding known items. This confirms our results regarding the evaluation of selection tasks for fisheye lists and folder views.

5. CONCLUSION

For drag & drop tasks our implementation of a fisheye list performed significantly better than the traditional list in respect to task performance, while accuracy levels stayed the same. With regard to selection tasks, there were no significant differences between the performance of participants using the fisheye list and their performance when using the list, while accuracy levels slightly decreased when applying the fisheye list. The folder view, which served as a baseline, performed better than the other widgets in all conditions. Moreover, study participants generally preferred the folder view. The fisheye list received significantly better ratings in all but one category than the traditional list. The results of our study therefore indicate that the application of a visual fisheye distortion on one-dimensional lists can be beneficial, especially for drag & drop tasks.

REFERENCES

- Bederson, B. B., 2000. Fisheye Menus. *Proc. UIST 2000*. ACM Press (2000), 217-225.
- Blanch, R., Guiard, Y., and Beaudouin-Lafon, M., 2004. Semantic pointing: improving target acquisition with control-display ratio adaptation. *Proc. CHI 2004*. ACM Press (2004), 519-526.
- Furnas, G., 1982. The FISHEYE View: A new look at structured files. *Bell Laboratories Technical Memorandum, #82-11221-22, Oct. 18, 1982*. Reprinted in Card, S., Mackinlay, J. and Shneiderman, B., 1999. *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufman.
- Gutwin, C., and Cockburn, A., 2006. Improving List Revisitation with ListMaps. *Proc. AVI 2006*. ACM Press (2006), 396-403.
- Hornbæk, K., and Hertzum, M., 2007. Untangling the usability of fisheye menus. *Trans. Comput.-Hum. Interact. 14(2)*. ACM Press (2007).