From Appearing to Disappearing Ephemeral Adaptation for Small Screens

Sara Bouzit[12] **Denis Chêne**[1] **Gaelle Calvary**[2]

[1]Orange Labs
28 chemin du Vieux chêne
38240 Meylan, France
prenom.nom@orange.com

[2]Univ. Grenoble Alpes, LIG, F-38000 Grenoble, France
CNRS, LIG, F-38000 Grenoble, France
prenom.nom@imag.fr

ABSTRACT

This paper presents two forms of adaptive menus for small devices (smart phones). Contrary to the Ephemeral *appearing* adaptation proposed by Findlater *et al.* [7], we claim for *disappearing* adaptation. The first form is named *In Context Disappearing* (ICD); the second one *Out of Context Disappearing* (OCD). The principle of ICD is to display predictive information in a prompting window placed above the main list. The prompting window disappears gradually while maintaining the context always visible and directly accessible. In case of low level prediction, ICD enables user to reach its target without waiting for disappearing effect. OCD principle is almost the same except that the disappearing prompting window covers the full page and thus is out of context like Findlater's approach. Our study shows that for small devices "fading out" a contextual window is better than "fading in". We demonstrate the benefit of these new forms of adaptation through an experiment with 24 subjects. We conclude that (1) ICD and OCD adaptive lists support faster selection than Control condition when the level of prediction is high, slower in case of bad prediction, and that (2) ICD is faster than OCD in case of bad prediction.

Author Keywords

Adaptive interfaces, interaction techniques, gradual onset, gradual disappearance, Ephemeral adaptation, disappearing adaptation.

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION

For a long time, adaptation of user interfaces to user profile and user context has become an urgent necessity. According to Findlater *et al.* [8], different users tend to use different functions. This suggests that interfaces must be customized for each individual user. Adaptation of Graphical User Interfaces (GUIs) [2, 5, 20] might be a suitable approach. It can tune spatial and graphical features of UIs.

The purpose of spatial adaptation approaches [13] is to reduce space navigation time and to facilitate visual search. Those techniques involve changing order and position of UI elements as it is done in split menus [19].

OzCHI '14 , December 02 - 05 2014, Sydney, NSW, Australia
Copyright 2014 ACM 978-1-4503-0653-9/14/12 $15.00
http://dx.doi.org/10.1145/2686612.2686619

Graphical adaptation techniques [10, 14] are intended to reduce visual search time. They adjust visual rendering, e.g. by highlighting some menu items or by changing font (type, size, case) and/or foreground and background colors.

In some situations, the use of adaptive interfaces has become indispensable. In 2008, Findlater *et al.* [6] [9] showed that adaptive interfaces are relevant for constrained devices such as smartphones where screen size is a major constraint. This study [9] confirmed that interaction benefits more from adaptive interfaces on small screens than on large screens. So the presence of adaptive interfaces on some supports becomes more and more a necessity, since the number of features used on small devices is increasing.

Grounded in universal design [4], we are interested in applying adaptive approach as it enables to generalize to other kinds of users (e.g., vision impaired) and constraint contexts of use such as small sceens, low vision, low accuracy due to mobility). The aim of our study is to translate the Ephemeral adaptation proposed by Findlater *et al.* [7] to smart phones where screen size is significantly reduced and where context of use might have an impact on usability. Initially we planned to verify performance of Ephemeral adaptation on smartphones. But quickly some main constraints appeared making it quite difficult to generalize Ephemeral adaptation to small screens. Inspired by these limitations, two adaptive approaches (ICD and OCD) are applied to smart phones.

Next section reports state of the art. Then, the focus is set on Ephemeral adaptation [7]. Two new approaches are presented and evaluated. Finally, a discussion of disappearing Ephemeral adaptation is proposed.

RELATED WORKS

Many studies have contributed to adaptive interfaces. In 1989, Mitchell and Shneiderman [17] proposed an adaptive menu which items are ordered according to their frequency of use: frequent options are in the top. However, Findlater *et al* [8] reports a user disturbance due to frequent changes of items position.

In 2007, Cockburn *et al.* [3] defined a new approach to adaptive interfaces. This time, items order does not change, but font size does in order to facilitate identification of the most used items. The motivation is gounded in the Fitt's law to speed up items selection.

However in case of bad prediction, targets selection becomes difficult.

Among most promising approaches, the split menu of Sears *et al.* in 1994 [19] attracted much attention. The split menu can be seen as a combination of two sub-menus. The first contains frequently used items or most frequent actions in frequency order. The second includes others in the original order. This split menu was already quite used, but it has the same problems as the Mitchell's menu.

In 2006, Gajos *et al.* [12] proposed an adaptive split interface that is regarded as a mixture between static and adaptive menu. This approach consists of a menu separated into two parts. The first part includes frequently used items and is therefore adaptive. The second part is static: predicted items are replicated. This is a strong limitation for smart phones which screen size is significantly reduced.

A modification was introduced in Microsoft Office 2000 © where only frequently used functionalities are displayed first. This variant has been quite criticized due to limited visibility of other functions.

In 2000, Bederson applied fisheye concept to menus giving rise to Fisheye Menus [1]. All items are displayed on a single screen or window that is completely visible. Items near the cursor are written with a larger font size. Thus the entire list can fit on a single screen. Of course, in case of many items, small options need to be focused to become legible.

Another kind of spatially stable menu (order of menu items stay unchanged) was proposed in order to reduce visual search time: frequently used items are highlighted by changing their background color or font color. Tsandilas and Schraefel [21] compared traditional highlighted items to highlighting in a fisheye menu [1]. On smart phones, Highlighting menus require a lot of concentration from the user, especially when predicted item is at the bottom of the screen. Indeed user must scroll window in order to see the predicted item.

In 2004, Lee and Yoon [15] proposed a new style of adaptive menu: temporal menus. A temporal menu presents items in two steps. At menu opening, the user finds only high priority items (relative frequency, importance or relevance within current context). Those items appear in the same position as in the full menu. After 170ms (100ms used for perception and 70ms for cognition), remaining items are displayed directly without any transition.

In 2009, Findlater and Gajos [8] reported an analysis of different adaptation approaches. In 2009, Findlater *et al.* proposed an interesting alternative [7] so called Ephemeral adaptation. Ephemeral adaptive menus combine Lee and Yoon's Temporal menus with transition. They use gradual onset of items. As for Temporal menus, this approach is a new way to improve performance by reducing visual search time while maintaining spatial consistency. In other words, items order in a menu remains unchanged. At opening, predicted items (prediction algorithm is based on the frequency and recency of use) are displayed. Other items appear gradually along a process that takes 500ms. Findlater *et al* showed that the performance is strongly related to prediction algorithm accuracy.

In 2013, Matejka *et al.* [16] proposed a system called Patina. Patina allows collecting and visualizing data using software applications. Patina provides visual benchmarks with dynamic graphic overlay. A colored heatmap is used to indicate the functions that are commonly versus rarely used in the interface, and adapts to the provision of the current interface. Patina uses an automatic transient display similar to the Ephemeral adaptation [7]. Findlater's and Matejka's solutions are only applied to large screens.

As part of small devices, Wavelet menus were developed by Francone *et al.* [11] for iPhones ©. They consist in concentric inverted hierarchical Marking menus based on simple gestures. The Wavelet menu allows user to interact with large hierarchies by using circular and linear forms.

Advanced menus are discussed in literature as Leaf menus proposed by Roudaut *et al.* [18]. However we focus on menus with the simplest possible form (lists) with regard to accessibility (disabled people and constraint of use).

This study is centered on Ephemeral adaptation. The key points of Findlater *et al.* study [7] are reported below as well as a critical analysis.

EPHEMERAL ADAPTATION

Principles and properties

Findlater *et al.* investigated Ephemeral adaptation on personal computers. The goal was to reduce navigation and visual search time.

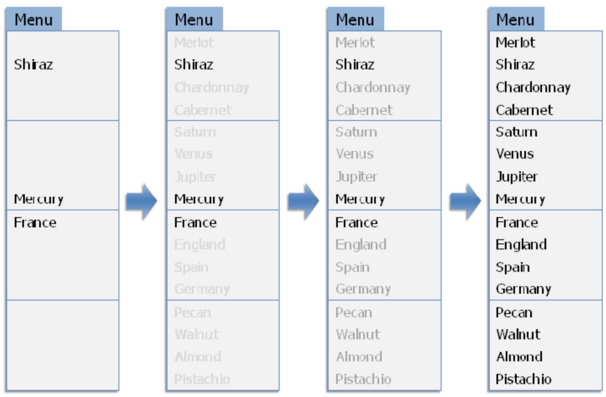

Figure 1. Ephemeral adaptation applied to menus: predicted items appear immediately, while remaining items gradually fade in [7].

The principle of Ephemeral adaptation is to display three predicted items in a first stage of interaction. The prediction is based on the frequency and recency of use. The purpose is to attract user attention on this set of items supposed to be immediately useful. After a delay and in a second step of interaction, the rest of items begins to appear gradually, until the total occurrence of all items. The best delay has been identified as 500 ms by Findlater *et al.* [7].

For the transition from the first stage of interaction to the second, Findlater *et al.* proposed to use progressive gray characters. They also propose to keep a spatial stability within the menu. That is to say that order and position of items in menu do not change. This spatial stability is very important because, after a certain numbers of uses, the user creates a memory model of this menu. The fact that position items in the menu stay unchanged allows user to rely on visual memory. This results in reducing cognitive effort, which actually reduces visual search time. Figure 1 shows the principle of Ephemeral adaptation applied to menus on personal computer.

Findlater *et al.* evaluated Ephemeral adaptation compared to the Control condition (without adaptation) and to the Highlighting approach through two experimental studies with 24 subjects each. The first experimental study aims to determine the best transition time between first to second stages of interaction. Findlater *et al.* tested 250, 500 and 1000 ms. 250 ms appears to be very short and users could not see predicted items while other items are already starting to appear. For 1000 ms users have found that this period is too long, especially when the prediction is incorrect. In this case users will be required to wait for a long time. 500 ms is the good trade-off between the three possible delays.

In the second study, Findlater *et al.* tested the performance of Ephemeral adaptation compared to Control condition and to Highlighting approach, when the level of prediction is high. Similarly it shows that when the level of prediction is low, the Ephemeral menu is not significantly longer.

Two prediction levels were used: 79% as high level prediction, and 50% as low level. The process is a set of selections with instructions indicating which item user must select. Results found show that on the one hand, when the level of prediction is high, Ephemeral menu is faster than Highlighting menu and Control menu. On the other hand, in case of low prediction level, Ephemeral approach is acceptable but is not quite different from Highlighting condition. In this case, spatial stability does not seem to have a particular effect.

Critical analysis

Within a framework of universal design, our goal is to translate Ephemeral adaptation to constrained usage, as low vision users or smart phones tiny screen sizes. Indeed, in such contexts, the screen size is significantly reduced, and as such may have impact on usability for regular and impaired users (as light variations, imprecise handling). Moreover the concept itself of pull-down menu is affected. Indeed, on a smart phone or for a user constrained to sequential interaction (blind user or motor impaired user), a pull-down menu, once opened, is totally similar to a simple plain page list of items.

One key element of Ephemeral adaptation is spatial stability. One can quickly see that spatial stability cannot be maintained when applying Ephemeral adaptation to smart phones. Indeed, all elements cannot be displayed on a single screen. This makes the implementation of Ephemeral adaptation on smart phones quite difficult. If predicted items are rendered within two different screens, the user must scroll. This requires from the user some effort and it needs adjusted concentration in order to explore all predicted items.

This constraint related to screen size display is similar to visually impaired user experience on wide screen: the user has to zoom in, and to handle a split list on multiple parts of the screen, browsable by scrolling. Spatial stability can be maintained by using multiple virtual screens at the expense of strong movement and performance constraints.

In addition, in the Ephemeral approach, the use of progressive gray to display non-predicted items generates some lack of performance. In case of incorrect prediction, it leads user to wait for target availability. Generally in human computer interaction, gray elements and semi-transparent ones convey items unavailability. This may have an impact on user who, in this case, would wait for items availability. In addition, on smart phones, gray font is not readable. In some places, such as outside under sunlight, visibility is becoming low making it difficult for the user to achieve his/her task.

Finally, efficiency of prediction is crucial, not to slow down interaction in case of bad prediction. Most adaptation approaches aim to give the most optimal and most efficient way to display the (good) prediction in order to speed up user interaction. However, on the other side, they do not really deal with the question of preventing slow down user interaction when the (bad) prediction doesn’t answer his/her needs. Of course it is not the desired case, but it needs to be taken into account at least at the same level as efficiency in case of good prediction.

In the line of Ephemeral adaptation, two adaptation approaches, namely In / Out of Context Disappearing adaptation (ICD and OCD) adapted to smart phones are tested. As Ephemeral adaptation, both provide a way to display prediction in order to accelerate user interaction, while reducing navigation and visual search time. Moreover, those methods do not slow down user interaction in case of wrong prediction.

In the following section, we present ICD and OCD approaches.

DISAPPEARING EPHEMERAL ADAPTATION

Inspired by Ephemeral adaptation, this paper aims at improving efficiency in case of both correct and incorrect prediction. The principle lies in gradual disappearance of useless information rather than gradual appearance of needed information. Speeding up interaction in case of wrong prediction should lead to better performances.

In Context Disappearing (ICD) adaptation

The principles of ICD are to keep the context (main list of items) visible and accessible at any stage of interaction, and to display the prediction in a prompting window. This latter appears above the main list of items. The prompting window contains the three predicted items with regard to frequency. The prompting window disappears gradually within 500 ms. We use the same time as in [7]. Figure 2 shows the principles of ICD.

When the menu opens, user sees a superposition of the small predicted list prompting window) and the main list.

The user searches his/her target in the prompting window. When prediction is correct, target is in the predicted list. The user selects it directly in the prompting window. Otherwise, in case of incorrect prediction, the user can immediately navigate inside the main list without being required to wait until complete disappearance of the prompting window. In this case, the user can select any item that is not hidden by the prompting window or can even start to scroll before complete disappearance of the prompting window. Only some hidden items are not clickable until total disappearance of the prompting window.

In summary, the prompting window is not a blocking modal window. It is an informative window dedicated to presentation of the predicted items. Moreover, ICD approach pushes a predicted list, contextualized within a complete list of items, and this last one can already be partially manipulated.

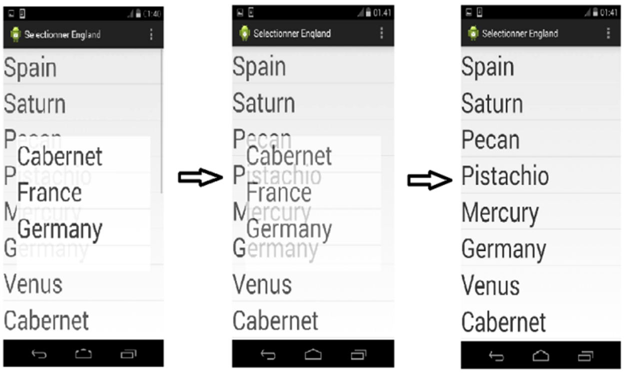

Figure 2. ICD adaptation applied to menus: a prompting window containing predicted items appears above the main list of items and then disappears gradually.

Out of Context Disappearing (OCD) adaptation

OCD adaptation looks like Ephemeral adaptation [7] but for tiny screens. It is a combination of progressive disappearance and gradual onset. In OCD, items are presented in two steps. When menu opens, the user sees three predicted items. These items are displayed at top of the menu. The second step triggers within 500 ms. In this step, the predicted items disappear gradually and the full list of items appears gradually. Figure 3 shows OCD principle.

The user starts by searching the target inside the predicted list. When the prediction is correct, he/she selects his/her target directly.

When the prediction is incorrect, the user waits for the total appearance of the main list for selecting his/her target.

In summary, OCD combines two different effects: prediction list fading out and main list fading in (gradual onset). As in Ephemeral adaptation, OCD follows a two-step process, the first step being dedicated to prediction only, without any access to the contextual main list. Contrary to Ephemeral adaptation, there is no spatial stability (the predicted items are not displayed at the same place at the first and second step).

Topological versus frequency-based factor

Findlater *et al.* claim for spatial stability. This is certainly useful and important, but as we have seen, it is difficult to apply or retain this spatial stability in smart phones where the main list is split onto several screens.

Nevertheless, it may be useful to keep some spatial properties as much as possible. First, a factor of order in the predicted list is handled. Three predicted items are in the same order than in the complete main list. Secondly, predicted items are organized according to a probabilistic . Items that have the highest probability of being used by the user will be at the top of the predicted list. These ordering and probabilistic conditions are worst regarding spatial stability as they are used in Findlater's study. They are repeated in both ICD and OCD conditions.

EXPERIMENT

The purpose of our experiment is to compare (1) ICD and OCD performance to Control condition (static non adaptive), (2) ICD performance to OCD, and (3) topological to frequency factor.

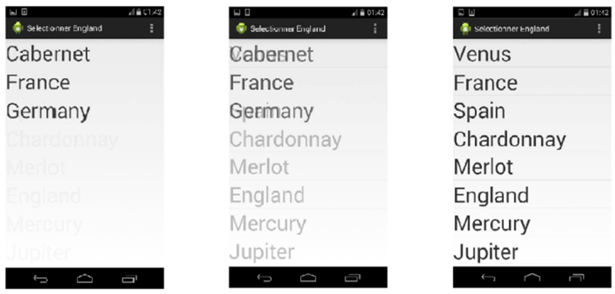

Figure 3. OCD adaptation applied to menus: the predicted list disappears gradually while the main list appears gradually.

Three tests on smart phones were implemented, with a list of 16 items. The items are those from Findlater *et al.* [7]. In each test, the user has a target, and has to reach it as fast as possible. The target may be on the prediction window or not (ICD and OCD first step condition), the target is always duplicated in the main list (ICD and OCD second step, and control first step), and inside this main list, the target may be on the first screen, or on the second screen as a result requiring some scrolling to be viewed.

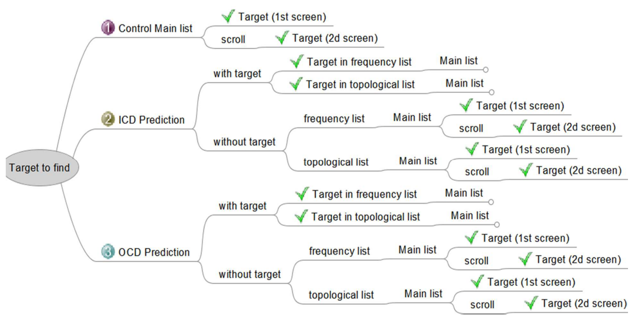

Figure 4. Tests overview.

First test is the Control condition (static approach), without any prediction at all.

Second test is the ICD approach, where prediction is contextual. The predicted list is displayed as a prompting window placed above the main list. In the prompting window, items are sometimes in the same order as in the main list (topological order condition), and sometimes organized according to the probabilistic criterion.

Third test is the OCD approach that consists in a sequential appearance of predicted list followed by full main list. Position of items in the predicted list is sometimes in topological order and sometimes in probabilistic organization.

Hypothesis

There are two hypothesis:

H1. Speed

- For a high level prediction

ICD and OCD are faster than Control. Since, in this case, target will be in first stage of interaction (in the predicted list), the user does not have to search as in the case of Control condition, where target can be on the first as on the second screen. Thereby, ICD and OCD conditions should be faster than Control condition.

ICD probabilistic display of predicted list is faster than ICD ordinal display of predicted list.

OCD probabilistic display of predicted list is faster than OCD ordinal display of predicted list.

These two hypotheses are based on the fact that lists are commonly read top-down and that for tiny screens, topological references are difficult to do for users.

- For a low level prediction

ICD is faster than OCD. When the target is not in the predicted list, user can already manipulate the main list without waiting for complete disappearance of the prompting window. In OCD condition, user must wait for complete main list full display.

ICD and Control are faster than OCD. Based on the previous assumption, since user must wait until total appearance of main list items in OCD condition, interaction should be slower than in ICD where the waiting time should be reduced. Compared to Control condition where user has direct access to the complete main list, OCD should be slower.

Whatever the list (Control, ICD and OCD) is, when the target is on the first screen, interaction is faster than when it's on the second screen. Indeed access to the first screen doesn't require scrolling and can be achieved immediately.

H2. User preference

- For a high level prediction

At least ICD or OCD is preferred to Control. In adaptive conditions (ICD and OCD), user attention is drawn to the predictive list. This makes target selection faster and simpler than in Control condition. In this latter, the user must find the target that can be on the first or on the second screen.

- For a low level prediction

Control is not preferred to ICD or OCD. As the desired objective of ICD and OCD is not to slow down the user interaction even when the prediction is incorrect, control should not be preferred to ICD or OCD.

Methodology

There are five independent factors. The first is the presence or absence of prediction (control vs. ICD/OCD). Control list is static without prediction (non adaptive) and ICD and OCD lists are with prediction.

The second factor is the kind of display of the prediction window (ICD vs OCD). ICD list is the simultaneous appearance of the predicted list (order or probability) and main list. The prompting window is placed above the main list and disappears within 500ms. OCD list is the plain screen window display of predicted list (order or probability) followed by gradual appearance of the main list within 500ms.

The third factor is the kind of display of the prediction list (topological vs frequency). For both ICD and OCD, the predicted list contains three items. In case of positive prediction, the way of presenting items may differ from order display to a probabilistic display.

The fourth factor is target location. In the three main lists, we used 16 items. Each list is divided into two screens. Each screen contains 8 items in order to have the same number of items in each screen. Targets distribution among the two screens is controlled.

The fifth factor is prediction accuracy. In case of accurate prediction, the expected target is included in the predicted list. In case of low accuracy prediction, the target isn't in the predicted list. In both cases, the target is inside the main complete list, on the first or second screen.

Task

The experimental task is a sequence of target selections. In each test, a message at the top of the screen indicates the item to be selected. The user can select it in the predicted list and/or in the full main list. When selection is right, the next target to be selected is displayed. In the case of incorrect selection, an error message is displayed and the subject has to try again.

Generally the users hold the smart phone in the left hand and select the target with the right hand (index finger).

We controlled by random draw the order of items in each list. The selection sequence (instructions) was also controlled by random draw. For conditions order, we have six distributions, and users were randomly assigned to a distribution. We also control the position of the target on the first or second screen, and the level of prediction.

Quantitative and qualitative measures

There are three dependent variables.

The first dependent variable is the selection speed. The speed is measured by time taken from opening menu until selecting the correct target.

The second dependent variable is the task achievement. Error rate was recorded.

The third dependent variable is the scrolling delay.

Finally, we collected subjective data about perceived difficulty, satisfaction and aesthetic using a Likert scale of 5 points and preferential ranking.

Technical settings

Android smart phones were used with prototypes coded in Java for Android. Probes recorded the selection time, scrolling time and error rate.

Participants

Twenty-four persons (5 women, 19 men) participated in this experiment. All participants were recruited internally in Orange Labs. All participants were regular tactile smart phones users and were around 23 and 57 years old.

Procedure

Before starting the test, a pre-test was performed allowing users to train with lists containing various items. Once the user successfully selected 10 targets during the pre-test, he/she was allowed to start the test. The latter is composed of 100 targets.

Results

Data were analyzed with ANOVA Randomized Blocks after verification of a variance homogeneity test (Brown & Forsythe's $W(9{,}230) = 2.64$, $p = .006$). Task achievement statistical analysis was performed by χ^2 and paired student T tests were helpful for detailed analysis.

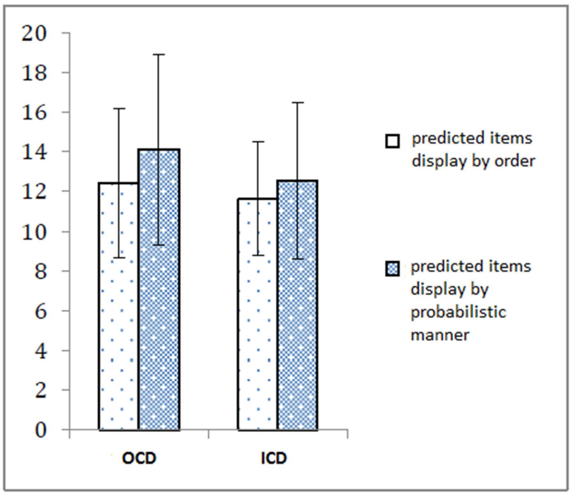

Figure 5. Selection speed for ICD and OCD when prediction is correct.

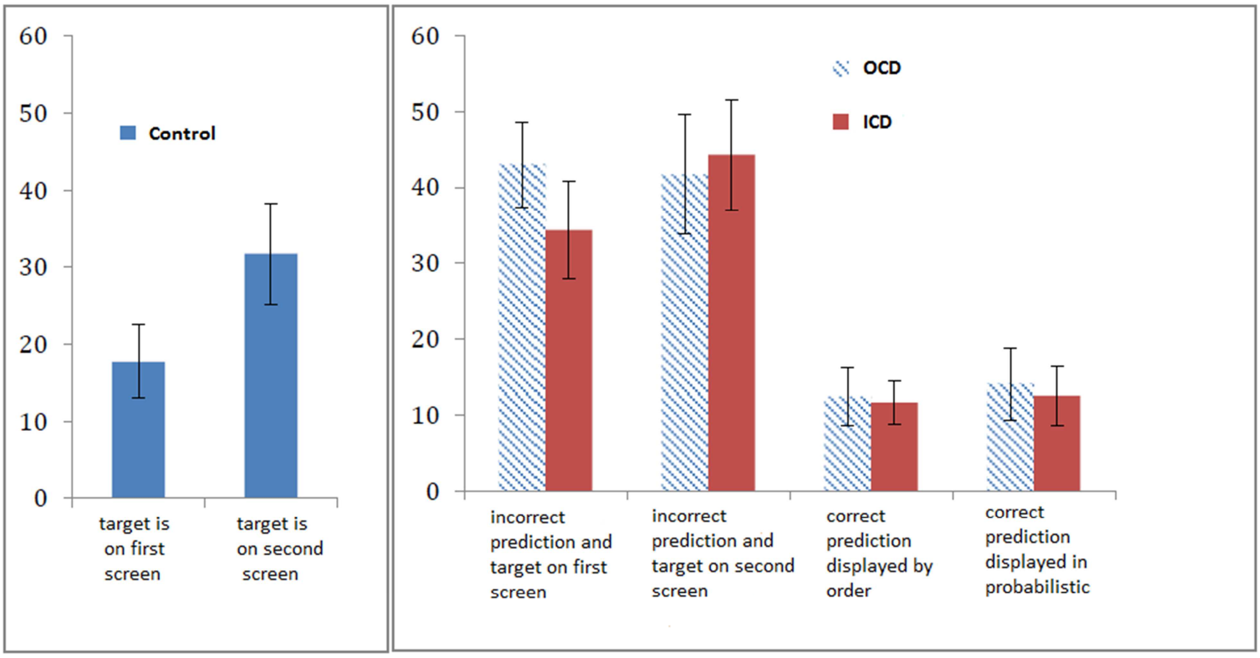

Figure 6. Selection time for all conditions.

Control condition consistency

Results show that in the Control condition (where there is no prediction), subjects make fewer errors than in ICD and OCD conditions $\chi^2(1, N = 2400) = 17{,}4$, $p < .01$. In the Control condition, when the target is located on the first screen, subjects were significantly faster ($M = 17.75$, $SD = 4.74$) than when target is located on the second screen ($M = 31.75$, $SD = 6.54$), $t(23) = 10.3$, $p < .01$ one-tailed. Moreover, this is for an equivalent error rate. This confirms the reliability of the Control condition.

Diseappearing effect

When users are on a predictive condition (ICD or OCD), they are significantly faster (respectively M =

12.08, SD = 3.45 and M = 13.27, SD = 4.36) than in Control condition (M = 24.75, SD = 9.05), for ICD vs. control $t(47) = 9.9$, $p < .001$ and for OCD vs. control $t(47)= 9.85$, $p < .001$. Implicitly, this justifies the use of a prediction. Furthermore, in ICD, subjects are significantly faster than in OCD $t(47) = 2.12$, $p < .02$, and that effect is for an equivalent error rate.

Target position on the first or on the second screen

Errors in OCD are more frequent when targets are on the first screen (31) than on the second screen (18) $\chi^2(1, N = 480) = 3.841$, $p < .05$. However we cannot show such significant difference with regard to time. This error rate is probably due to the succession of similar windows and to the overlap between the first and the second stage of interaction.

In contrast, in ICD, interaction steps are simultaneous and windows are clearly different. Indeed, in this condition, when the target is not in the first step on the predicted window, the user can click directly on a visible part of the list. Users are significantly faster when the target is located on the first screen (M = 34.38, SD = 6.43), than when it's located on the second screen (M = 44.25, SD = 7.24), $t(23) = 6.41$, $p < .001$. Scrolling takes time for both ICD and OCD (respectively M = 39.31, SD = 8.41 and M = 42.33, SD = 6.78), but it is still significantly shorter with ICD $t(47) = 2.08$, $p < .02$, and for an equivalent error rate.

Topological ordering effect

Similarly, when targets are presented on predicted window, predicted items presented by order (topology prediction) are significantly faster (M = 12.02, SD = 3.36) than when they are presented in a probabilistic way (M = 13.3, SD = 4.41), $t(47) = 2.79$, $p < .003$, with an equivalent error rate between the two. It seems that spatial stability is useful for quickly retrieving the target.

For user preference, we didn't find any significant difference.

Discussion

H1. Speed

- For a high level prediction

ICD and OCD are faster than Control. Supported.

When the prediction is correct and corresponds to user need, the latter reaches his/her target more rapidly. This further justifies the importance of prediction and shows that it is crucial in adaptation. Although the user after some number of uses can learn the position of items in the UI, but this does not prevent that prediction remains crucial, especially in certain contexts that were previously mentioned. Smart phone is a good example, the number of applications being growing, and the screen size reducing (even if screens are wider and wider). In this case, even if the user comes to learn the position of all elements, access to them is not always obvious.

ICD probabilistic display of predicted list is faster than ICD ordinal display of predicted list.. Not supported

OCD probabilistic display of predicted list is faster than OCD ordinal display of predicted list. Not supported

Results show that probabilistic display of prediction in ICD and OCD is not faster than ordinal display. This might be explained by the fact that spatial stability plays an important role in UIs. It helps the user to create a memory model of the UI. This allows him to rely on his/her visual memory when searching for targets.

- For a low level prediction

ICD is faster than OCD. Supported

Results show that when the prediction does not correspond to what the user is looking for, ICD is faster than OCD. This is justified by the fact that access to the complete list of items is easier in ICD than in OCD. In ICD, the main list is available at all stages of interaction. On the contrary, in OCD user must wait for the total appearance of the main list. Moreover, in ICD the user can select an item (which is not hidden by the prompting window) without waiting for the complete disappearance of the prompting window, and the possibility to scroll immediately, speed up user interaction. However they may be risky for novice users.

In summary, when prediction is incorrect, the user must move as quickly as possible to the main list. Therefore, the use of gradual disappearance in the main list (ICD) of this bad prediction, with the ability to directly manipulate the next step items (items of first screen, or scrollbar for second screen) is better than the use of gradual onset of the main list (OCD) forcing user to wait for the second interaction step.

In conclusion, ICD using progressive disappearance does not slow down user interaction in case of incorrect prediction. Therefore, gradual disappearance is better than gradual onset.

ICD and Control are faster than OCD. Supported

ICD is faster than OCD. Similarly, when the prediction is incorrect, navigation is easier in Control condition as user directly accesses to the main list of items. Unlike in OCD, the user must wait for the total appearance of the list to reach its target.

Whatever the list (Control, ICD and OCD) is, when the target is on the first screen, interaction is faster than when it is on the second screen. Supported

Results confirm our hypothesis. It is clear that access to the target on the first screen is faster than on the second screen. To access the second screen, the user must scroll, contrary to the first screen. More important, access to the first screen in ICD is faster than in OCD. Indeed, keeping the context available (main list items) at all stages of interaction is important. In OCD there is a waiting time until the total appearance of all items, which can be annoying. This confirms again that the approach of gradual disappearance is better than the gradual onset.

CONCLUSION

This paper introduces *In Context Disappearing* (ICD) and *Out of Context Disappearing* (OCD) adaptations, two forms of Evanescent Ephemeral adaptation. The point is to overcome contextual limitations, in particular those related to more and more reduced screen sizes.

We have experimentally shown that, when the level of prediction is high, ICD and OCD are faster than Control. Also, ICD is faster than OCD.

Results also showed that ICD is not slower than Control in case of wrong prediction. This makes of ICD a good solution whatever the quality of the prediction is (correct or incorrect): interaction is speed up when the prediction is good; interaction is not slowed down when prediction is incorrect.

Thus, while keeping the benefits of ephemeral adaptation [7],, ICD overcomes its limitations by being more efficient and faster overall.

To conclude, the use of progressive disappearance (ICD) is better than the use of progressive appearance (OCD).

In future work, we propose to continue this study by generalizing this concept of ICD as part of a mosaic of icons (home screen of smart phones for example). Adding a spatial dimension may have an effect on topological factor. Visual transition between the prediction and the main window could also power up interaction. In this vein, continuity between ordered predicted items and their real topological position in the main list could be animated. We also propose to add elements in the interface allowing user to act directly on ephemeral delay in order to adapt it to his/her level of expertise. Taking into account novice users and possible increasing errors in ICD could also be an interesting perspective. Finally, displaying effects of prediction is only part of the overall interaction problem, it will be quite interesting to add different kinds of interaction modalities (gestural, vocal…) and to observe display-commands interaction.

REFERENCES

1. Bederson, B.B. Fisheye menus. ACM UIST, (2000). San Diego, USA.p. 217-225.
2. Bridle, R., and McCreath, E. Inducing shortcuts on a mobile phone interface. In. Proc. IUI, (2006), 327-329.
3. Cockburn, A., Gutwin, C. and Greenbrerg, S. A predictive modal of menu performance. In. Proc CHI'07, (2007), 627-636.
4. Constantine, S. Adaptive Techniques for Universal Access. User Modeling and User-Adapted Interaction 11: 159-179, 2001.
5. Findlater, L. and McGrenere, J. A comparison of static, adaptive, and adaptable menus. In proceedings of the SIGCHI conference on human factors in computing systems, pages 89-96. ACM, 2004.
6. Findlater, L. and McGrenere, J. Impact of Screen Size on Performance, Awareness and User Satisfaction With Adaptive Graphical User Interfaces. In Proceedings of the twenty- sixth annual SIGCHI.
7. Findlater, L., Moffatt, K., McGrenere, J. and Dawson, J. Ephemeral Adaptation: The Use of Gradual Onset to Improve Menu Selection Performance. ACM CHI, (2009), 1655-1664.
8. Findlater, L. and Gajos., K.Z. Design space and evaluation challenges of adaptive graphical user interfaces. AI Magazine, 30(4): 68, 2009.
9. Findlater, L. and McGrenere, J. Evaluating reduced functionality interfaces according to feature findability and awareness. In. Proc. IFIP Interact 2007, (2007), 592-605.
10. Florins, M. and Vanderdonckt, J. Graceful degradation of user interfaces as a design method for multiplatform systems. ACM IUI 2004: 140-147.
11. Francone, J., Bailly, G., Lecolinet, E., Mandran, N. and Nigay, L. Walvet Menus on Handheld Devices: Stacking Metaphor for Novice Mode and Eyes-Free Selection for Expert Mode. ACM AVI, 2010.
12. Gajos, K.Z., Czerwinski, M., Tan, D.S. and Weld, D.S. Exploring the design space for adaptive graphical user interfaces. In Proc. AVI'06, (2006), 201-208.
13. Gajos, K.Z., Everitt, K., Tan, D.S, Czerwinski, M. and Weld, D.S. Predictability and accuracy in adaptive interfaces. In Proc. CHI'08, (2008), 1271-1274.
14. John Christie, Raymond, M. and Carolyn, R. Watters: A comparison of simple hierarchy and grid metaphors for option layouts on small-size screens. Int. J. Hum.-Comput. Stud. 60(5-6): 564-584 (2004).
15. Lee, D.S. and Yoon, W.C. Quantitative results assessing design issues of selection-supportive menus. International Journal of Industrial Ergonomics 33 (1), 2004, pp. 41-52.
16. Matejka, J., Grossman, T. and Fitzmaurice, G. Patina: Dynamic Heatmaps for Visualizing Application Usage. ACM CHI, 2013.
17. Mitchell, J. and Shneiderman, B. Dynamic versus static menus: An exploratory comparison. SIGCHI Bulletin 20, 4(1989), 33-37.
18. Roudaut, A., Bailly, G., Lecolinet, E. and Nigay, L. Leaf Menus: Linear Menus with Stroke Shortcuts for Small Handheld Devices. In Conference Proceedings of INTERACT'09, 2009.
19. Sears, A., and Shneiderman, B. Split menus: Effectively using selection frequency to organize menus. ACM TOCHI 1, 1(1994), 27-51.
20. Shneiderman, B. (2003). Prompting universal usability with multi-layer interface design. In Proc of CUU 2003, 1-8. ACM Press.
21. Tsandilas, T. and Schraefel, M.C. An empirical assessment of adaptation techniques: In CHI'05 Extended Abstracts, (2005), 2009-2012.