Adaptive support for user interface customization: a study in radiology

Wiard Jorritsma a,⁎, Fokie Cnossen b, Peter M.A. van Ooijen c

a University of Groningen, University Medical Center Groningen, Dept. Radiology, Hanzeplein 1, 9713 GZ, Groningen, The Netherlands
b University of Groningen, Institute of Artificial Intelligence and Cognitive Engineering, Nijenborgh 9, 9747 AG, Groningen, The Netherlands
c University of Groningen, University Medical Center Groningen, CMI-NEN, Dept. Radiology, Hanzeplein 1, 9713 GZ, Groningen, The Netherlands

ABSTRACT

Objectives: This study aimed to evaluate the usefulness of adaptive customization support in a natural work environment: the Picture Archiving and Communication System (PACS) in radiology.

Methods: Adaptive support was given in the form of customization suggestions, generated based on behavioral user data, which participants could choose to accept or ignore. Twelve radiologists worked with the standard adaptable version of the PACS for six weeks, during which their actions on the PACS interface were logged. Based on these logging data, customization support was generated for each specific participant. Half of the participants received support and half did not. After the support was given, logging continued for another six weeks. Participants’ customization behavior and performance, measured as the average time needed to review a radiography study, were compared between the groups. Subjective responses to the customization support were also measured.

Results: Participants who received support used the PACS’s customization facilities more effectively than participants who did not receive support. Participants receiving support accepted most of the customization suggestions and all participants indicated that the support was useful. We did not find an increase in efficiency due to the support, possibly because the performance measure we used was not sensitive enough. Subjective responses did show that participants perceived the support as increasing their efficiency.

Conclusion: Adaptive customization support would be a useful addition to the standard adaptable PACS interface, because it allows radiologists to customize their interface more effectively.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Over the past decades, digital filmless radiology has largely replaced its traditional film-based counterpart. In the modern hospital, the distribution of patient images is controlled by a Picture Archiving and Communication System (PACS). The PACS also provides the user interface through which the radiologists retrieve, view and manipulate images. As in many other modern software packages, the number and complexity of functions in the PACS is very high and continues to increase. This poses the challenge of creating a user interface that presents these functions to radiologists in an appropriate way and allows them to interact with the software efficiently. Because different radiologists use the software in different ways, depending on their goals and interaction preferences, creating an interface that suits each radiologist is a difficult task.

As a solution to this problem, most PACSs have an adaptable interface, which allows radiologists to customize several aspects of the PACS (e.g. the items in the toolbar, the functions of the mouse, and the display protocols1) according to their personal needs and preferences. Research in the field of human-computer interaction has shown the potential benefits of adaptable interfaces compared to static ones (e.g. Findlater and McGrenere, 2004; McGrenere et al., 2002), but there are also problems with this approach. Users do not always customize effectively (Bunt et al., 2004), or they do not customize at all (Mackay, 1991). This means that they will never interact with the system in a maximally efficient way.

An alternative approach to interface personalization is the adaptive approach, where the interface changes automatically based on the user’s behavior. This approach ensures that the interface is customized to the user, without requiring any effort on the user’s part. However, automatically changing the interface can reduce the system’s predictability and transparency, and can

1 A display protocol automatically arranges images on the screen in a way that is appropriate for the types of images that need to be displayed.
undermine the user’s sense of control over the system (Höök, 2000).

The mixed-initiative interface combines elements from both approaches, creating an adaptable interface with adaptive components that help the user customize effectively. The adaptive support is usually presented in the form of user-specific customization suggestions, which the user can choose to accept or dismiss. For example, the system could suggest to add or remove functions from a toolbar based on their frequency of use. Compared to a purely adaptive interface, a mixed initiative interface increases the system’s predictability and transparency (the system does not change itself without the user’s knowledge and consent), and increases the user’s control over the system (users can dismiss the system’s customization suggestions and can initiate their own customizations). Compared to a purely adaptable interface, a mixed-initiative interface increases the likelihood of customization (users may be more likely to customize because the adaptive support reduces the effort to customize, or because it simply reminds them of the possibility to customize). It can also increase the quality of customization, because the set of customization suggestions may contain useful customizations that users would not have considered on their own.

An example of a mixed initiative interface can be found in Debevc et al. (1996). This system provided adaptive customization support for Microsoft Word 6.0 by suggesting functions to be added to or removed from Word’s toolbar based on their frequency of use. Both novice and expert Word users performed faster with this mixed-initiative interface than with the standard adaptable interface, although this effect was only found in a picture and text formatting task and not in a table editing task. The measure of task performance included the time users spent customizing the interface.

The adaptive support also reduced the average time needed to perform a customization. With support, users had to evoke the support mechanism and if they decided to accept a customization suggestion, it was performed automatically. This was faster than manually performing a customization, as done by users using the standard interface. Novices using the adaptable interface did not add any functions to the toolbar, while novices using the mixed-initiative interface did. Interestingly, experts in the mixed-initiative condition added fewer functions to the toolbar than experts in the adaptable condition. However, because a between-subject design was used, the possibility of a bias in customization behavior between the mixed-initiative and the control group cannot be excluded.

Bunt et al. (2007) implemented a similar mixed-initiative interface for Word 2003 and compared this interface with an adaptable one. They found that the mean task performance (both when excluding and including customization time) was faster in the mixed-initiative condition than in the adaptable condition, but this effect was not statistically significant (although the effect size was large), possibly due to the small number of participants used. Participants customized more effectively and efficiently when using the mixed-initiative interface. There was also a large subjective preference for the mixed-initiative interface.

Park and Han (2011) compared two mixed-initiative interfaces (adaptable with system support and adaptive with user control) for a PDA software prototype to an adaptable, adaptive and static alternative. Participants performed a controlled menu selection task (excluding customization time) slower and with more button presses on the static interface than on the others, but there were no differences between the other interfaces. The adaptable interface with system support did reduce the customization time compared to the standard adaptable interface. This system presented participants with their function usage frequencies, which allowed them to make faster decisions about which customization to perform. The perceived efficiency and overall preference was higher for the two mixed-initiative and the adaptive interfaces than for the adaptable and static ones.

These studies show that adding adaptive customization support to an adaptable interface has the potential to increase customization effectiveness and efficiency, interaction efficiency (although this is not strongly supported), and user satisfaction. However, all of these studies were conducted in laboratory conditions over short periods of time (the longest study lasted for 2.5 h). Users’ customization behavior might be very different in real-world situations, where different (perceived) costs and benefits of customization apply and where users are not artificially manipulated to customize. Furthermore, little is known about the long-term effects of customization and how customization behavior changes over time.

The fact that our hospital had just purchased a new PACS presented us with the unique opportunity to study the effects of adaptive customization support on users working with a new interface in a real work environment. This study aimed to evaluate whether adaptive customization support would be a useful addition to an adaptable PACS interface.

1.1. Adaptable PACS interface

The graphical user interface of the PACS used in this study is shown in Fig. 1. At the top of the interface is a toolbar consisting of multiple tabs, each with a different set of functions (similar to the Ribbon in Microsoft Office 2007 and later). One of these tabs, called My Tab, is customizable; allowing users to add functions to it. A similar customizable region called Right Click Menu (RCM) is located at the top of the pop-up menu that appears when the right mouse button is clicked within the image. Users can add functions to these customizable regions by right-clicking on a function button and selecting add to My Tab or add to RCM. The default My Tab and RCM consist of eight pre-defined functions each. For most discussions in this article, the distinction between these two customizable regions is irrelevant. In these cases we will use the term custom region for both My Tab and RCM.

1.2. Adaptive customization support

The adaptive customization support was designed to help users customize the PACS’s custom region effectively. It was based on users’ function usage, which was logged by the PACS’s built-in logging tool, and consisted of a table that gave insight into a user’s function usage and a set of suggestions about which functions the user should add or remove to his or her custom region.

1.3. Research questions and hypotheses

In order to determine the usefulness of adaptive customization support for an adaptable PACS interface, the following questions were addressed:

(Q1) Does customization support increase the effectiveness with which users use the PACS’s customization facilities?
(Q2) Does customization support increase the efficiency with which users interact with the PACS?
(Q3) To what extent do users accept the customization suggestions?
(Q4) Do users perceive customization support as being useful?

Effective customization (i.e. constructing a high-quality custom region) means trading off custom region coverage (the percentage of total function usage that the functions in the custom region cover) against custom region complexity (the number of functions
Because the optimal trade-off is user-dependent (some users tolerate a higher complexity than others), it is impossible to objectively measure custom region quality. However, there are three measures that provide strong indications for effective customization: the frequency of adding a function to the custom region, the frequency of removing a function from the custom region, and the custom region usage, i.e., the proportion of function selections made from the custom region relative to the regular interface. Effectiveness was therefore measured in terms of these three variables.

Efficiency was measured by the average time needed to review a radiography study. To assess the extent to which users accept the customization suggestions, we measured the number of functions users in the support group accepted in their custom region, broken down into three categories: functions from the suggested optimal custom region, additional frequently used functions, and other functions. Perceived usefulness was measured by asking users whether they found the support useful and why, and whether they thought the support increased the efficiency with which they could work with the interface and why.

To answer these questions, we conducted an experiment using a mixed factorial design with customization support as a between-subject factor and time as a within-subject factor. All users worked with the PACS for twelve weeks and half of them were given customization support after six weeks. Customization behavior and performance were compared between the groups and the time periods (week 1–6 and week 7–12). Subjective responses to the customization support were also measured.

We hypothesized that the effectiveness with which users used the custom region (measured in terms of customization frequency and custom region usage) would increase more in week 7–12 compared to week 1–6 for users who received support than for users who did not receive support (H1), and that this would lead to a larger increase in efficiency for the support group (H2). We expected users to accept most of the customization suggestions (H3) and to perceive the support as being useful (H4).

2. Method

2.1. Function usage logging

Users' function usage within the PACS was logged by the PACS's built-in logging tool. This tool works on a workstation level (in contrast with a user level), which means that it logs data of all users working on a specific workstation and not all data of a specific user on all workstations. After collecting the log files from the workstations, they were converted from workstation-specific to user-specific, which was possible because users' usernames were logged when they logged into the PACS. A user-specific log file contained a log of all functions a specific user used on all the workstations that were logged.

The logger was turned on eighteen (out of the total of 70) randomly selected workstations at various locations within the hospital. Unfortunately, it was not possible to log more workstations due to constraints imposed by the hospital's IT department. Logging
started on the first day the new PACS was being used. Five weeks after the start of the logging, the log files were collected from the workstations. The log files were collected again after twelve weeks.

A limitation of the workstation-specific logging was that if a user worked on a workstation that was not part of the eighteen workstations that were logged, these data would not be present in this user’s log file. For most users, this was not a serious issue since they have a private workstations on which they do (most of) their work, but it could be a problem for users who mainly use “public” workstations (this is common for radiology residents).

The logging yielded data of 35 unique users. The number of data points (i.e. function and tab selections) per user ranged from 2 to 6456. Only users with a log file containing more than 400 data points were included as participants in the experiment, because these log files were considered a good representation of a user’s total function usage (i.e. the number of collected interactions was large enough to warrant generalization to the total number of interactions).

Only functions from the PACS’s main viewer were analyzed. This excluded functions from specialized applications within the PACS (e.g. Vessel Analysis, Calcium Scoring and Volume Matching).

2.2. Participants

Twelve users (eight certified radiologists and four radiology residents; six female) met the inclusion criterion described above and were therefore included as participants in the experiment. We tried to log as many workstations as possible, and thereby get as many users as possible, but we were only allowed to log eighteen workstations by our IT department. Therefore, twelve users was the best we could do, even though we aimed for more.

2.3. Apparatus

Participants interacted with the PACS as part of their normal daily routine on a standard workstation. The PACS used in the experiment was Carestream Vue version 11.3 (Carestream Health, 2012).

2.4. Materials, design and procedure

All radiologists and residents within the hospital received training of the new PACS before they started working with it. The PACS’s customization options were demonstrated in this training, so participants were aware of the fact that they could customize the PACS’s interface.

Because integrating a customization support component into the PACS was not feasible (mainly because we did not have access to the PACS’s source code), we used a simulated mixed-initiative setup in which the customization support was given by a human experimenter rather than by the computer.

Participants did not have a close relationship with the experimenter. Of the six participants in the support group, five had never met him and one had participated in one of his previous studies.

The support was based on users’ function usage, which was logged by the PACS’s logging tool. Based on a participant’s function usage, his or her “optimal” custom region, which was defined as the region containing the smallest set of functions making up at least 80% of the total function usage, was determined. Participants were presented with a list of all the functions they had used, ordered from high to low usage frequency, showing the absolute frequency and the percentage of total use for each function. The list also showed the percentage of function usage accessible from the custom region if it included this function and all functions with a higher frequency (incremental percentage). The optimal custom region was highlighted. An example is shown in Table 1.

Table 1

<table>
<thead>
<tr>
<th>Function</th>
<th>Frequency</th>
<th>Percentage of total use</th>
<th>Incremental percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pointer</td>
<td>314</td>
<td>20.4</td>
<td>20.4</td>
</tr>
<tr>
<td>Mark as key image</td>
<td>296</td>
<td>19.3</td>
<td>39.7</td>
</tr>
<tr>
<td>Line measurement</td>
<td>243</td>
<td>15.8</td>
<td>55.5</td>
</tr>
<tr>
<td>Change image rendition</td>
<td>219</td>
<td>14.2</td>
<td>69.7</td>
</tr>
<tr>
<td>Oval measurement</td>
<td>207</td>
<td>13.5</td>
<td>83.2</td>
</tr>
<tr>
<td>Windowing</td>
<td>98</td>
<td>6.4</td>
<td>89.6</td>
</tr>
<tr>
<td>Start aquatic net</td>
<td>39</td>
<td>2.5</td>
<td>92.1</td>
</tr>
<tr>
<td>Zoom from cursor</td>
<td>30</td>
<td>2.0</td>
<td>94.1</td>
</tr>
</tbody>
</table>

Participants were suggested to add all functions from the optimal custom region to their custom region (either My Tab, RCM, or both). They could accept or decline the entire suggested custom region, or accept a subset of its functions.

In addition to the functions in the suggested optimal custom region, participants could add more functions to the custom region, based on their frequency of use, so that it covered an even larger percentage of their total function usage.

When participants had decided which functions they wanted to include in their custom region, the experimenter added them. Participants were then suggested to remove all functions from the custom region (either default functions or functions added by participants themselves) that were not part of the suggested optimal custom region or the additionally accepted functions. Participants could choose to keep functions that were suggested to be removed. Participants were also suggested to set My Tab as the default tab.

We used a mixed factorial design with customization support as a between-subject factor and time as a within-subject factor. Participants were randomly assigned to the support and no-support groups, while the number of radiologists and residents in each group was balanced.

The experiment lasted for twelve weeks. After five weeks, the log files were collected from the workstations and were used to generate the customization support. After six weeks, the experimenter planned a session with each participant during which the support was given. Participants in the support group were interviewed immediately after this session and again at the end of the experiment in order to obtain subjective responses to the customization support.

2.5. Measures and data analysis

To analyze the effectiveness with which users used the PACS’s customization facilities, we measured the number of times participants added functions to and removed functions from the custom region (customization frequency), and the proportion of function selections made from the custom region relative to the regular interface (custom region usage). Three functions (zoom, automatic registration and change image rendition) were excluded from the latter analysis because due to an error in the logging tool they were always logged as being selected from the regular interface, even when they were selected from the custom region.

The efficiency of interaction with the PACS was measured by the average time needed to review a radiography study in one of the two time periods. This time was determined based on the report submission times retrieved from the Radiology Information System. Because radiologists often review multiple radiography studies in quick succession, the time between two radiography
report submissions in such ‘reviewing blocks’ represents the time needed to review one radiography study. Report submission time differences longer than twenty minutes were excluded because they were not considered part of the same block. One participant, a resident from the no-support group, was excluded from this analysis because he only reviewed four radiography studies during the time course of the experiment. Because radiography interpretation is (usually) relatively straightforward and radiologists review many of these studies in six weeks, we can assume equal average image complexity in the two time periods.

Boxplots consisting of four data sets, one for each group (support and no support) in each time period (week 1–6 and week 7–12), were used to assess the differences between the groups and the time periods. Because not all participants in the support group received the support on the same day, the time periods for each of these participants were relative to the day at which they received the support (i.e. six weeks before this day, and six weeks after and including this day). Statistical tests were not used because of our small sample size.

To analyze the level of acceptance of customization support, we measured the number of functions from the suggested optimal custom region participants in the support group accepted, the number of additional functions they added based on their usage frequency (presenting usage frequency information was also part of the support), and the number of infrequently used functions they added or declined to remove.

Immediately after they had received support, participants in the support group were interviewed. They were asked whether they found the support useful (yes or no) and why (open-ended). Then they were told how customization support could be implemented in the PACS interface, by using a fully adaptive or mixed-initiative approach, and asked which of these methods they would prefer (fully adaptive, mixed-initiative, or no preference) and why (open-ended). At the end of the experiment (six weeks after the first interview), participants were asked whether they thought the support had improved the efficiency with which they worked with the interface (yes or no) and why (open-ended) (subjective responses to customization support).

3. Results

3.1. Customization frequency (Q1)

Fig. 2 shows the number of times participants added a function to the custom region in both time periods for participants in the support and no-support group. In week 1–6, some participants added functions to the custom region, but the median number of functions added was approximately zero for both groups. In week 7–12, only one participant in the no-support group added functions to the custom region. In the support group on the other hand, all participants added more functions in week 7–12 (after they had received support) than in week 1–6.

Participants in the support group added most functions during the session in which the support was given (median number of 'add' customizations = 8.5, range = 5–11) and a few in the weeks afterwards (median number of 'add' customizations = 2, range = 0–9).

Fig. 3 shows the number of times participants removed a function from the custom region in the two time periods for participants in both groups. In week 1–6, most participants did not remove any functions from the custom region. In week 7–12, most participants in the no-support group still did not remove any functions, while participants in the support group did.

Participants in the support group removed most functions during the session in which the support was given (median number of 'remove' customizations = 6.5, range = 1–18) and almost no functions in the weeks afterwards (median number of 'remove' customizations = 0, range = 0–1).

3.2. Custom region usage (Q1)

Fig. 4 shows the percentage of function selections made from the custom region, as opposed to other parts of the graphical user interface, in the two time periods for participants in both groups. Custom region usage increased over time for both the support and the no-support group and this increase was larger for the support group. A notable difference between the groups was that the minimum of the support group increased dramatically, while it
remained the same for the no-support group. The between-participant variability also decreased in the support group, but not in the no-support group (as can be seen from the IQRs in Fig. 4).

3.3. Efficiency (Q2)

Fig. 5 shows the average time participants in both groups needed to review one radiography study in each of the two time periods. This measure did not reveal any clear differences between the groups or the time periods.

3.4. Acceptance of customization support (Q3)

Participants who received support accepted most of the functions from the optimal custom region (median = 93%, range = 50–100%). In addition to these functions, they added many other functions. This is illustrated in Fig. 6, which shows the total number of functions in the custom region of all participants who received support immediately after support was given, broken down into three categories: functions from the suggested optimal custom region, additional frequently used functions (defined as functions that made up more than 1.4% of the total function usage), and other functions (either infrequently used functions or functions that were not included in the support because they were not part of the PACS’s main viewer).

All participants added at least one infrequently used function to the custom region, or declined to remove it from the custom region. The reason for this was that they expected to use these functions frequently in the future. Four of the six participants actually used these functions frequently during the remaining six weeks of the experiment. The other two participants still used them infrequently or not at all, but did keep them in their custom region.

3.5. Subjective responses to customization support (Q4)

Immediately after participants received the support, they were asked whether they found it useful. All participants indicated that they did, giving the following reasons: “The support makes frequently used functions easier to access” (five participants); “It is useful to gain insight into your own function usage” (two participants); “You tend to just continue working the way you do”; “I already forgot the training with respect to customization by the time I actually started working with the PACS”; “I was not even aware that the My Tab existed”.

At the end of the experiment, participants were asked whether they thought the support increased the efficiency with which they could interact with the PACS interface. Five participants indicated that it did and one participant indicated that it did so a little bit. The following reasons were given for this increase in perceived efficiency: “The support improved the accessibility of functions” (four participants); “The support improved my user experience”; “I don’t have to switch between [toolbar] tabs anymore and I don’t
have to remember which function is in which tab. The custom region creates order in the chaos of the extremely large amount of functions”; “The support increased the quality of the content of my custom region”.

Participants were told how customization support could be implemented in the PACS interface: by using a fully adaptive or a mixed-initiative approach, and were asked to indicate their preference. All participants indicated that they would prefer a mixed-initiative customization interface for the following reasons: “Otherwise you don’t know what has happened to the interface”; “I hate it when things are adjusted automatically”; “Then you know what’s going on”; “I would like to be able to choose for myself. Also, I don’t want the interface to change all the time”; “Suggestions are always welcome, but I want to have the final say”; “So that I know when something changes”.

4. Discussion

In this study, we aimed to determine whether adaptive customization support would be a useful addition to an adaptable PACS interface. We conducted an experiment in which participants worked with the PACS for twelve weeks and half of them received adaptive customization support after six weeks. Customization behavior and performance were compared between the groups and the time periods (week 1–6 and week 7–12). Subjective responses to the customization support were also measured. We hypothesized that the effectiveness with which users used the custom region would increase more in week 7–12 compared to week 1–6 for users who received support than for users who did not receive support, and that this would lead to a larger increase in efficiency for this group. We also expected users to accept most of the customization suggestions and to perceive the support as being useful.

We found that participants who received support customized more frequently than participants who did not receive support. The fact that participants who received support not only added more functions to the custom region, but also removed more functions from it indicates that the support increased the quality of functions in the custom region and not just the quantity (useful functions were added and useless functions were removed). Bunt et al. (2007) and Debevc et al. (1996) did not find a difference in the number of customizations between participants working with a mixed-initiative and an adaptable interface. A possible explanation for this difference between these studies and ours is that the artificial nature of the experimental tasks used in these studies prompted participants to customize, even when they did not receive customization support, thereby obscuring the effects of the support. In our study, there were no artificial factors that prompted participants who did not receive support to customize.

The number of function selections made from the custom region relative to other parts of the interface (custom region usage) increased over time for participants in both groups. Two factors played a role in this effect: (1) participants added functions to the custom region, thereby increasing the number of functions that could be selected from the custom region, and (2) participants became more aware of the fact that they could select functions more quickly from the custom region than from the rest of the interface. The second factor was especially important for participants in the support group who were only vaguely aware of the custom region’s existence before they received support, because the support dramatically increased their awareness of the custom region’s usefulness.

The increase in custom region usage over time was larger for participants who received support, but since there was considerable overlap between the data of the groups, the evidence for this difference was not very strong. However, these data underrepresented the actual custom region usage because three very frequently used functions were excluded from the analysis due to an error in the logging tool (they were always logged as being selected from the normal interface, even when they were selected from the custom region). Because these functions were so frequently used, they were very likely to be in the optimal suggested custom region and therefore added to the custom region by participants in the support group. Therefore, the actual custom region usage was likely to be much higher for participants in the support group relative to participants in the no-support group. The effect of the support would therefore be more visible if these functions had been correctly logged and included in the analysis.

A clear difference between the groups was that the minimum custom region usage of participants in the no-support group remained the same over time, while it increased dramatically for participants in the support group. The support also decreased between-participant variability. This indicates that users who are unlikely to use the custom region a priori will not become more likely to do so unless they receive support. Obviously, customization support is most useful for users who do not use the customization facilities on their own. The support therefore increases custom region usage more for these users than for users who already customize of their own accord, and thereby reduces the variability between users.

We did not find a difference in performance between the two groups, possibly because the performance measure we used (the average time needed to review a radiography study in one of the two time periods) was not sensitive enough to reveal any effects of customization support. We used this performance measure because it was natural, yielded many data points, and did not require participants to abandon their daily work. In hindsight however, the measure was too coarse and subject to too many interfering factors. It would have been better to measure performance using a controlled set of tasks that focused on interaction with the PACS interface and excluded diagnosis and interaction with other software on the radiology workstation.

Despite the fact that we did not find an effect of support in the objective performance data, the subjective responses did show that all participants who received support perceived it as increasing the efficiency with which they could work with the PACS.

All participants who received support found it useful and they accepted most of the functions from the suggested optimal custom region. The reason that some functions were rejected was that these suggested functions could be accessed through other equally efficient methods (e.g. shortcut keys or action regions within the viewports) and participants already accessed the functions in this way. It was not possible to take this factor into account when generating the support due to limitations of the logging tool.

The large number of additional frequently used functions in the custom region indicates that participants wanted the custom region to cover more than 80% of their total function usage. Participants also accepted a substantial number of infrequently used functions in the custom region. There were two reasons for this: (1) the customization support motivated participants to add functions they had not used frequently so far because they were difficult to access, but they expected to use frequently in the future if they would be in the custom region. For example, one participant commented that he only used the arrow marker to mark abnormalities in an image before he received support because this was the easiest to access marker function. He expected to use the square region marker much more often when he realized that he could add both the arrow marker and the square region marker to the custom region. (2) Participants did not want to remove functions they rarely used (either part of the default custom region or added by themselves) because they expected to use
them frequently in the future. The expected increase in usage frequency came true for some participants, but not for others, yet all of them kept the functions in their custom region. This shows that some users are able to accurately predict their future function usage and customize accordingly, while others are not. There were also functions that became more frequently used over time, but were not added to the custom region. For example, one participant removed the magnifying glass function from the default custom region because she never used it. In the following weeks however, she reviewed a lot of mammograms, for which the magnifying glass is quite useful. Even though her usage of the magnifying glass increased dramatically, she did not add it to the custom region again. These findings suggest that users do not always adapt their interface to changes in their work patterns, which makes it useful to continuously provide customization support.

In line with our hypothesis and the findings of Bunt et al. (2007), our results show that participants who received customization support used the PACS's customization facilities more effectively than participants who did not receive support. Contrary to our expectations, this increase in customization effectiveness did not lead to improvements in performance. However, due to the aforementioned limitations of our performance measure and the fact that the perceived efficiency did increase for all participants who received support, we believe that an effect of customization support on performance would have been found if a more fine-grained measure of performance had been used. The increase in customization effectiveness and the positive subjective responses to the support lead us to conclude that adaptive support would be a useful addition to an adaptable PACS interface.

It is possible that participants were more likely to accept customization support in our experiment because the support was given by a human experimenter instead of a computer. While this is a valid concern, it does not change the fact that customization support in itself was found to be useful. The challenge now remains to design a computer interface that delivers the support as effectively as a human does. We believe that if an appropriate interface is designed to present the support, which provides the rationale behind the suggestions as was done in our experiment, there will be a negligible difference in acceptance between support given by a human and support given by a computer.

One could argue that the results were simply due to the fact that users were reminded of the possibility to customize and that a simple reminder would be just as effective as the adaptive support. While the fact that customization support serves as a reminder to customize is arguably its most important property, its adaptive nature is also valuable for the following reasons: (1) a static reminder only prompts users to customize whereas adaptive support also assists them in deciding what to customize. The fact that several participants explicitly indicated that it was useful to gain insight into their function usage in order to decide which customizations to perform supports this and is indeed a useful property. (2) Adaptivity allows the support to be displayed at an appropriate time: when there is a discrepancy between the user's function usage and his or her custom region. (3) Adaptivity allows specific customizations to be performed automatically by the system (when the user approves them), Users may be more likely to act on the support if it provides an easy and fast way to perform the customizations.

A factor that might have influenced custom region usage is users' sense of involvement and ownership of the custom region. Users might be more inclined to use the custom region not only because it improves their user experience and efficiency, but also simply because they constructed it themselves. While this factor might have played a role in our study, participants' subjective responses showed that they (also) had more valid reasons for using the custom region (e.g. improved accessibility of functions, reduced cognitive load). Furthermore, custom region usage and custom region quality are not independent, even if users use the custom region solely because of involvement and ownership. If the custom region is not customized effectively, most functions can simply not be selected from it.

All participants who received support indicated that they would prefer a mixed-initiative customization interface over a fully adaptive interface. The reasons for this preference coincided with the limitations of adaptive interfaces described by Höök (2000): participants feared a lack of predictability and transparency in a fully adaptive interface and they wanted to have control over the changes made to the interface. This suggests that a mixed-initiative approach to PACS interface customization support would be more appropriate than a fully adaptive approach. However, an empirical study comparing a mixed-initiative against a fully adaptive PACS interface is needed to truly determine which is the most appropriate.

The adaptive support used in this study was based on the overall frequency of function usage over a five-week time period. Other adaptive interfaces often also take the recency of function usage into account (e.g. Debevc et al., 1996; Findlater and McGrenere, 2004). The danger of assigning a large weight to recency information is that it decreases the stability of the interface. In a mixed-initiative interface, this would lead to a large number of irrelevant customization suggestions that are likely to annoy users and decrease their overall trust in the suggestions. We propose using frequency information over a moving time window as the basis for the adaptive support. This ensures stable suggestions, while implicitly taking recency into account. An appropriate size for the time window depends on the type of interface and the nature of the tasks users perform on it and should be determined empirically.

The adaptive support used in this study only concerned customization of the custom region. Even though this rather simple support was already useful, much more advanced support could be given if radiologists' behavior would be logged more elaborately. For example, if study information (imaging modality, body part scanned etc.) and the way radiologists arrange images on the screen were logged, the adaptive support could include suggestions for more effective display protocols (which automatically arrange images on the screen based on their characteristics). For example, consider a display protocol for a knee X-ray that displays the frontal view of the knee on the screen. If a radiologist reviewing knee X-rays with this protocol always manually places the lateral view of the knee next to the frontal view, this could be suggested as a permanent change to the knee X-ray display protocol.

The advent of structured reporting also presents opportunities for adaptive support to aid customization of the content and layout of the radiology report. Possibilities for report customization have been described by Reiner (2010) and a promising interface that adaptively displays reports has been developed in the cardiology domain (Chédira et al., 2002).

The usefulness of such content customization has been demonstrated in various domains (e.g. customization of routes generated by an itinerary recommender system (Schaller and Elsweiler, 2014), and customization of the decision-making and transaction processes of online shops (Thirumalai and Sinha, 2011)).

Another possibility would be to promote feature awareness by suggesting features that radiologists are currently unaware of (in our experience, most radiologists do not know what all buttons in the PACS interface are for), but that are likely to be be useful to them based on their behavior (as in Matejka et al., 2009). More elaborate logging also presents other interesting possibilities. For example, the log files could be used for usability evaluation of the system (as in Atterer and Schmidt, 2007;...
Atterer et al., 2006; Hong and Landay, 2001), and to monitor radiologists’ performance on a more fine-grained level than the high-level productivity measures that are currently used by many hospitals.

Radiologists often review multiple types of studies (e.g. chest X-rays and lung CTs). Some functions in the PACS are only useful for certain types of studies. It could therefore be beneficial to make the custom region context-specific, so that it only contains functions that are relevant for the study the radiologist is currently reviewing. An advantage of this approach is that it reduces the complexity of the custom region by hiding all irrelevant functions. A disadvantage is that it reduces the system’s predictability (the interface changes with each different type of study) and increases the complexity of the customization process if the radiologist has to specify which functions should be present for which types of studies, or reduces the system’s transparency if this is determined by the system automatically. More research is needed to determine if and how a context-specific custom region should be implemented in the PACS.

5. Conclusion

The adaptable PACS interface allows radiologists to customize several aspects of the PACS according to their personal needs and preferences. However, in line with previous studies on customization behavior we found that radiologists do not always customize effectively, and sometimes do not customize at all. This study has shown that adaptive customization support helps radiologists use the PACS’s customization facilities more effectively. Radiologists perceived the support as being useful and accepted most of the customization suggestions. The increase in customization effectiveness and the positive subjective responses to the support lead us to conclude that adaptive customization support would be a useful addition to an adaptable PACS interface.

Acknowledgments

We would like to thank André Broekema and the people at Carestream Health for their assistance in the setup and data retrieval of the logging tool. We also thank Tjibbe Hoogstins for providing the data on which we based our performance measure, and the radiologists and radiology residents who participated in this study.

References


Wiard Jorritsma is a researcher with an interest in human-computer interaction, usability engineering and artificial intelligence; he is a PhD student in the radiology department of the University of Groningen, University Medical Center Groningen.

Fokke Cossen is an experimental psychologist with an interest in human error and cognitive engineering in patient safety; she is an assistant professor of Cognitive Engineering in the Cognitive Modeling group of the Institute of Artificial Intelligence and Cognitive Engineering (ALICE) at the University of Groningen.

Peter M.A. van Ooijen is a scientific researcher with an interest in Medical Imaging Informatics ranging from information systems and data distribution to advanced visualization and usability; he is an associate professor in the department of radiology of the University of Groningen, University Medical Center Groningen.