Improving the User-Interface to Increase Patient Throughput

A. Gupta, B.Sc., Principal Scientist, Philips Research Laboratories, Redhill, UK

J. Masthoff, Ph.D., Research Scientist, Philips Research Laboratories, Redhill, UK

P. Zwart, M.A., Philips Medical Systems Ned. B.V., Best, The Netherlands

Abstract

One of the main goals of a radiology department is to optimise patient throughput. We have observed a number of factors which reduce patient throughput, one of them being sub-optimal system usage. In this paper, we distinguish and discuss two ways to reduce sub-optimal operation: improved design of the user-interface, and active support for learning during system usage i.e. during examinations. We outline the rationale for this by looking at the current situation and trends in the radiology departments. We have based our work firmly on the principles of user-centred design: observations, task modelling, user involvement and prototyping have been undertaken.

Keywords: Medical, Error, Guidance, Learning, Task Analysis, User Interface, Help

1. Human error in the radiology department

In a radiology department, patients are exposed to X-rays to produce images for diagnosis or to assist in an operation. Important goals in the radiology department are to:

- maximise patient *safety and comfort*, for instance, by minimising the X-ray dose used (which is harmful to the patient), and by accurate diagnosis,
- maximise *patient throughput*, by reducing the time needed per examination,
- minimise the *resources* used per patient, such as film, contrast fluid, and staff time.

Leplat states that "a human error is produced when a human behaviour or its effect on a system exceeds a limit of acceptability" (as quoted in reference 1). Another definition of human error is "all the occasions in which a planned sequence of mental or physical activities fail to achieve its intended outcome, and when these failures cannot be attributed to the intervention of some chance agency" (ref. 2). Both of these definitions associate human error with not reaching a set goal ("exceed a limit of acceptability" and "fail to achieve its intended outcome").

Based on this, we see human error in the radiology department as *human behaviour resulting in sub-optimal achievement of the goals listed above*. Accordingly, the following are examples of errors:

- Applying a higher dose than strictly necessary (even when not violating safety limits!)
- Taking more time for an examination than necessary
- Producing sub-optimal images and thereby endangering good diagnosis.

There are many reasons why an error may occur:

- Sub-optimal operation of the system, i.e., not using all functions of a system or not operating the system in the most efficient way.

  Example: Suppose there is a function to delete a set of images. A radiographer needs to delete a large set of images, but instead of using the function, she deletes the images one by one, wasting time repeating the same action.
- Sub-optimal system design, i.e., the system does not support users' tasks sufficiently, the system is difficult or slow to operate, or the system is not designed for the particular examinations for which it is being used.

  Example: Suppose that often a large set of images needs to be printed but the system only provides a function to print one image at a time, or it does provide a function to print a set of images but that function is hidden.
- Sub-optimal workflow management ("providing the right work for the right person at the right time", ref. 3), i.e., different parts of the work do not connect seamlessly.

  Example: Suppose the radiographer taking the images has to wait for a long time for the next patient to be brought in. Better scheduling or better communication between the different work spots could prevent this, thereby increasing patient throughput.
- Non-optimal diagnosis, for instance, due to inadequate awareness of medical research. This issue is addressed in evidence-based medicine (ref. 4).

  Example: Suppose a radiologist does not interpret images in the correct way, because he is not aware of the latest research on the connection between certain artefacts in images and a certain rare disease.

Often a combination of these occurs: the clinical user does not use a certain function because the design of the system does not make the purpose of the function clear. In this paper we describe our work on reducing sub-optimal operation of systems as caused by sub-optimal design. We are interested in optimising a design process, such that human errors are prevented. The rationale for this will be given in the next section. Workflow management and evidence-based medicine are outside the scope of this paper.

2. Current situation and trends

2.1 Observational studies

In order to assess the current situation and trends in radiology departments, visits were organised to five hospitals in the Netherlands, Belgium, and the UK. An application specialist of Philips Medical Systems would accompany us. Application specialists are attached to hospitals and maintain frequent contact with clinical staff at all levels; they also deliver on-site training and support. The whole process was guided by a checklist of questions. The checklist was based on core user-centred design practices (ref. 5) to make profiles of users, the environment in which they work (technical, organisational, and social), and the tasks they perform. In addition, it contained questions on trends.

Summary of process followed:

1. Interview of the application specialist. We asked about the type of hospital, the kinds of examinations conducted, the number and types of people working there, the types of systems used, the process used for introducing new systems into the hospital, trends in the hospital, and problems experienced and anticipated.
2. Observations of examinations (on average 8 per hospital). We would draw plans of the rooms, and record the actions of and communications between the clinical staff and patient, noting information present on walls, and ambient conditions, for example, lighting and noise. We would not only look at what happened during an examination, but also between examinations.
3. Interviews of staff present. We talked to radiographers, radiologists, cardiologists, and nurses, both to very experienced staff and trainees. We would ask about the nature of their work, their roles, the problems they experienced, the changes they would like, the trends in their work, their training, and what they liked in their job.
4. Writing hospital visit reports (on average 25 pages, typical sections: users, systems, environment, tasks, information, training, UI issues, and trends). Observations and interviews were done in pairs, so that one

observer's notes could be checked and complemented with the notes of the second observer. The application specialist checked the resulting report for correctness.

2.2 Current situation

The following observations relate to the topic of this paper and were consistently made during hospital visits. It is beyond the scope of this paper to provide a complete overview of the profiles of users, environment, and tasks obtained.

- [Obs1] In all hospitals, we observed that radiologists and radiographers often only know part of the available system functionality, and not always how to accomplish their task efficiently.
- [Obs2] In all radiology departments visited, systems are operated by people with different backgrounds. For instance, surgery systems are often operated by cardiac radiologists as well as by nurses, for different types of examinations. According to hospital staff and application specialists, the growing and complex functionality, for instance, for cardiac applications, further raises the threshold for the nurses to operate the systems efficiently.
- [Obs3] In all control rooms, we observed hand-written reminders, tips, and other information, for instance on allergic reactions to contrast fluid, being stuck to the walls or the lead glass screen.
- [Obs4] In all radiology departments visited, training on a new system takes place during on average three days on installation. On average, three clinical users are trained: initially by giving them a system overview, and then by assisting them during real examinations. These clinical users then go on to train their colleagues. So, a substantial part of training is conducted on the job. User manuals were observed in only one control room.
- [Obs5] In all radiology departments visited, clinical users must repeatedly move from one system to another, and get confused by controls varying from system to system due to the range of systems from different manufacturers, and the lack of industry standards.

2.3 Trends

The interviews during the observational studies highlighted the following trends in the radiology departments. These trends were confirmed by market intelligence in Philips Medical Systems.

- [Trend1] An increasing amount of functionality is being offered in systems. For instance, the manual for Philips' Cardiology X-ray system has almost doubled in size, increasing from 120 pages for the BH3000 to 245 pages for the following system, the BH5000.
- [Trend2] Boundaries between domains are becoming fuzzy. For instance, the boundaries between diagnostic and interventional radiology, and between cardiology and surgery are diminishing. Increasingly, for example, cardiologists prefer when performing a minimal invasive by-pass operation to also place an abdominal stent (i.e., a small intervention in a leg normally performed by a radiologist) when needed.
- [Trend3] There is a shift from stand-alone to networked systems where the medical equipment is effectively a peripheral. Increasingly, there is a requirement for equipment to be based on open standards, and for inter-operability with systems from other manufacturers.
- [Trend4] Information is increasingly stored digitally, giving rise to film-less departments and to the need for image access and archiving functionality in the systems.

2.4 Conclusions from the above

The main conclusions from the observations are that X-ray systems should be usable by a broad user group [Obs2] without much training [Obs4,5] and that this objective is currently sub-optimally met [Obs1,3,5]. Therefore improved design of the user-interface and active support of the learning process seems appropriate (Note: It is not suggested that these are the only approaches to address the problems).

The trends indicate that the user group of X-ray system is broadening even further [Trend2], and that sub-optimal use will continue to increase in line with system functionality and complexity [Trend1,2,3,4]. This indicates that good user interface design and active support of learning become even more important in future. These are now described in turn.

3. Improved design of the user-interface

Designing a user-interface requires answering the following questions: (1) which user tasks to support, (2) which functions to provide in order to support these tasks and how to enable the user to operate these functions, and (3) what navigational structure and screen layout to use to make these functions available.

3.1 Task Modelling

A Task Model represents the tasks undertaken to achieve a goal. It is a hierarchical drawing as tasks may be decomposed into sub-tasks. It is read from left to right and captures the process used by the subject in going about his or her job. It is also independent of any technology, user-interface or other, as the level of abstraction is much higher than mouse and keyboard interactions. Task Models are of two types: those representing currently used systems, and those representing the target system to be designed. An example-fragment of the former kind is shown in Figure 1. We have based our task modelling work in part on the MUSE method described by Lim & Long (ref. 6). Note: The relation between a Task Model and a system is a subtle one: though a task model is rooted in one or more systems it is only a description of tasks, and is entirely independent of system-level details.

Notation:

- *Task Ordering*. Two types of ordering are distinguished in the task model diagrams: Sequential Ordering (indicated by a single horizontal line, tasks ordered from left to right) and Free Ordering (indicated by a double horizontal line).
- *Repeatable tasks*. These are indicated by multiple vertical bars connecting a task to its parent task.
- *Optional Tasks*. These are represented by boxes drawn with dashed lines.
- *Location of tasks*. As we are only interested in two locations (examination and control room) we have used simple colour coding in the top of the box to represent this property. Brown (which prints as a dark shade in grey-scale) represents the examination room; blue (a light scale) represents the control room. An absence of colour implies no restriction on location.
- *Person*. This indicates any demarcation on tasks amongst radiologists and radiographers. Red (which prints as a dark shade in grey-scale) represents tasks done by the former; green (a light shade) the latter.

The observations of examinations during the hospital visits resulted in the construction of a generalised task model of X-ray imaging. This task model together with the user and environment profiles and trends, was used to generate concepts for a future system. Five concepts were worked out in more detail, into scenarios, and made more realistic. The task model was used as a basis for this work: we identified which sub-tasks were closely related to a concept and visualised how the concept could impact these tasks. The scenarios were evaluated on the basis of technical feasibility and usefulness and one concept was chosen for further development.

Based on the chosen concept and the generalised task model of currently used systems, a new task model was constructed for the target system. A fragment of a task model for a target application supporting Viewing and Post-processing of images is presented in Figure 2. The notation is similar to the notation above, except that location and person indications have been omitted, as all the tasks are intended to be performed in the same location and by any user type.

Benefits of task modelling:

1. *Capturing important characteristics*: In the observations we conducted it soon became clear that task characteristics such as location, communication, co-operation and duration would be useful to record. These could constrain design solutions (e.g. co-operation between users), or point to where improvements would be most effective (e.g. duration). These can easily be recorded as annotations on the Task Model.
2. *Communication*: Having a common frame of reference is useful both within the project team, and when communicating with partners. The task model proved to be very useful when collaborating with others during idea generation and interaction design.
3. *As a basis for design*: Comparing task models is a good starting point to find commonalities and differences which impact user-interface design. Once agreement on the desired task structure has been obtained, candidate design solutions can be brainstormed. Optimal functional allocation between system and user, can then be followed by interaction and graphic design.
4. *Validation*: Task Models can be shown to application experts to check that descriptions are correct. The notation is simple enough to understand.

3.2 Interaction styles

To determine which functions to provide a user to assist with the leaf-tasks of the task model, and which interaction to provide for these functions, we have analysed the way in which these tasks have been implemented in existing systems (both in the medical and non-medical domains). For instance, the task “Optimise the brightness of an image” has been implemented in X-ray systems, but also in televisions and image processing software on PC. Such an analysis leads to a choice inspired by best practise across many domains, as much as the need to balance consistency with existing practise in the medical domain.

The resulting user interactions are specified in the form of Storyboards. These contain:

- a description of the purpose of the storyboard (for instance, “Setting image brightness by touch”)
- a description of the appearance, in terms of sample screen content and sound output (for instance, two buttons, one with a “+” and one with a “-” and a warning sound if value outside the allowable brightness range is set)
- the ways to access (enter) and exit the storyboard, including voice commands (for instance, access by pressing an image quality button, exit by pressing an okay button)
- the interactions possible at the top level of the storyboard (for instance, pressing the + button resulting in an increase in brightness, and pressing the – button resulting in a decrease)
- the sub-interactions, which are of two kinds; either further branches i.e. pointers to other storyboards, or invocation of functions
- the syntax of the interactions in modified Backus-Naur form, and
- miscellaneous remarks

How can one determine which interaction is best? What are the relevant questions to ask? We applied the Cognitive Walkthrough technique (ref. 7). The primary purpose of this technique is to analyse user interfaces in terms of the problems users may encounter due to lack of understanding of the user interface. The technique involves simulating the way users explore and gain familiarity with interactive systems, with the aid of a simple step-by-step model:

- The user starts with a rough plan of what she wants to achieve - i.e. of the task to be performed
- she explores the interface looking for actions that might contribute to performing the task
- she selects the action whose appearance or description most closely matches what she is trying to do
- she then interprets the system's response to see whether progress has been made toward completing the task.

The analysis involves simulating these steps and asking questions of the form:

1. Will the correct action be made sufficiently evident to the user ?
2. Will the user connect the correct action with what she is trying to do?
3. Will the user interpret the system's response and feedback correctly?

Evaluating the interface in this way is an iterative process, and an important form of validation.

3.3 Navigation structure

The navigation structure is based of the task model, the functions and interactions chosen, and the available space on the interface. The ordering in the task model influences the navigation structure and screen layout of the system as follows. A task that has to be performed before other tasks, might be better represented on an earlier screen. Tasks that can be performed in any order should preferably be present on the same screen. In western systems, tasks on the same screen will normally be ordered from top to bottom and from left to right. For example, in the case of a sequential ordering of tasks A and B, the functions to enable task A should either be on a screen that precedes the screens with the functions of task B, or they should be to the top (and left). It should be noted that even in the case of free ordering, the layout of the task model may lead to conclusions from, say, the screen-layout designers: there is a high probability that in a free ordering of a number of tasks, the order from left to right will be used as a normal (though not required) ordering. Therefore, we have ordered the tasks in the task model within a free ordering from left to right according to a likely sequence. Notes have been added to the task model to indicate more than one likely sequence, or to state conditions under which a certain sequence is likely to occur.

The grouping of subtasks in the task model, reflects the fact that certain tasks are conceptually more related to each other than to other tasks. In the user interface this could result in visually grouping functions related to those tasks (for instance by proximity).

4. Active support for the learning process

In order to promote optimal use of a system, clinical users need to know:

- *Which functions can be useful in a certain situation*. For instance, when performing a Subtraction task (i.e. subtracting a reference mask image from another image), motion artefacts can occur due to patient movement. The user needs to know that the 'Pixelshift' function can correct this (by shifting the relative positions of the mask and source images).
- *When to apply a certain function*. For instance, Brightness can be set whenever viewing an image, but it is advisable to set Polarity, electronic shutters, and optionally to perform subtraction, beforehand.
- *How to operate a certain function in a certain situation.* For instance, in a future system, a Zoom function could be activated by a voice command and then, touching the area that is to be the centre of the zoomed image. Alternatively, during some interventions, hands-free or even eyes-free operation may be needed. If ambient noise is too high, speech recognition is impaired and another way of operating the function may be more appropriate.
- *What the scope of a function is.* For instance, when changing Brightness while viewing an image, the effect can be a change in Brightness of the image being viewed, or of all images in the set, or of all subtracted images.
- *How to undo the effect of applying a certain function.*

It is impossible to give a user all the knowledge mentioned above during a short training period. However, more training time is unlikely: users generally have to learn on the job. Therefore, we are working on active support by the system of the learning process. The support has to be highly customised, depending on the hospital, the experience and preferences of clinical users, the examination being conducted, and user roles.

There has been quite some research into the design of intelligent help (see, e.g., ref 8), focusing on providing the users with good ways to ask questions and on volunteering answers when the user is observed to have a problem. In that research, the user-interface is seen as a constant factor. We have taken a different approach, and are focussing on changing the user-interface to increase learnability. We see both approaches as being complementary.

Learnability is mainly related to efficiency and effectiveness over time. Preferably, the system should be highly intuitive to use the first time, i.e., have an immediate high effectiveness without much effort on the side of the user, and efficiency should increase over time, leading to a highly efficient system for experienced users. Moreover, systems should support the rapid progress of users from initial to experienced level.

In the system we consider (and this holds for most systems), only a limited amount of space is available for the presentation of functions and information. Nevertheless, we know for the X-ray viewing and post-processing domain that the users want many functions to be available at the same time, without having to take extra navigational actions ("within one button click" in manner of speaking). By developing a good user-interface design, and making only those functions available which are appropriate in a certain situation (see above) it is possible to meet this criterion to a certain extent. If we include the use of speech commands to operate functions, it is even possible to have functions available that are not immediately visible on the system. However, to present all allowable functions on the system or hide them by using speech commands, has a rather negative impact on learnability. There is insufficient room to explain what all the functions do, and how to operate them, and this is particularly acute when the user is presented with a wide choice. So, our concern for novice users conflicts with our desire to provide highly efficient systems to experienced users.

One option may be to provide separate interfaces for novice and experienced users. However, this has only limited advantages: a user who is familiar with the use of certain functions (such as setting Brightness and Contrast) may never have heard of other functions (like Pixelshift) which can be very useful for his current situation (say, performing a Subtraction with motion artefacts). This is something we have actually observed in our hospital visits. The option of two separate interfaces also poses the danger that the gap between novice use and experienced use becomes too large.

What we would like, is that, for instance, when the user is performing a Subtraction, and has not used Pixelshift before, the system indicates that this function can be used to remove motion artefacts, and how to use it. This means that we will not use two separate interfaces, but one interface which gradually changes to reflect the experience of the user (i.e., the navigation structure and information is customised to the experience of the user).

The (simplified) example illustrates how this could work (see Figure 3, note that it uses a touch sensitive screen). If the user is not familiar with the concept of subtraction then this concept is introduced (image 1). Next, the user is guided through the subtraction task. At each step, it is explained how the user should do it, and how he or she can correct mistakes (images 2-4).

Note that the initial sequence presented (images 2-4) is directly based on the task model (see fragment of the task model in Figure 2). Initially the user is forced to do the tasks in the task model from left to right, even though the model allows free ordering. When the user gets more experienced, the free ordering is introduced in the user interface (compare images 2,3 and 6). Note also that the information presented reflects the types of information mentioned above. What information is presented to the user depends on his experience. If the user has never used a certain function before (and not seen the information related to that function recently), then information regarding the purpose of the function is given, and the user is instructed how to operate it. More detailed information regarding the steps to take is given only when needed. For instance, if the user already has experience with selecting images, as part of another task, then no information is given anymore on how to (de)select an image (compare images 2 and 5). The user can get this next level of information if he wants it after all, by explicitly asking for it with the help function (available from the button marked "?"). We envisage that users will also be able to explicitly select the level of information they like for certain functions. This enables experienced users to still use more structured interfaces if they prefer that.

The work described in this section is very much work in progress. Evaluations have to be conducted to judge the effectiveness and acceptability of the approach.

5. Conclusions

Our main interest is in exploring how to reduce sub-optimal system usage in the X-ray examination and control rooms. This has a direct impact on patient throughput, which is expected to continue to be an important measure in the hospital radiology department. We have described how we have applied user-centred design methods basing our work on first-hand observations conducted in hospitals, and on Task Analysis and Modelling.

Various problems have been observed during field studies. In this paper, we have suggested two ways these could be addressed:

(1) Improved user-interface design based on user-centred design. We have described how such a user-centred design process can work.

(2) Active support by the system for the learning process (needed as training takes place on the job). We have sketched a way in which the user's learning process is supported by gradually changing the interface to reflect the user's growing experience.

These promote optimal and full system usage. Our work is ongoing; we are presently in the midst of building an integrated research prototype, which will be tested in usability evaluations.

Acknowledgements

We would like to thank the application specialists who accompanied us in the hospital visits: Erik Vossen, Theo Loohuis, Jacques Lukassen, and Peter Sutton. Thanks are due to two designers: Job Rutgers and Jude Rattle. Boris de Ruyter was involved in some discussions on the notation for the task model. The support provided by Eric von Reth and Franklin Schuling is acknowledged.

References

1. Bés, M-O., Analysis of a human error in a dynamic environment: The case of air traffic control*. Proceedings of the workshop on human error and systems development*, Glasgow, 1997.
2. Reason, J. *Human error*. Cambridge: Cambridge University Press, 1990.
3. Schmidt, J., Meetz, K., & Wendler, Th. The application of workflow management systems in radiology. *Proceedings of the 15th international EuroPACS meeting*, Pisa, Italy, 1997.

4. Sackett, D.L., Rosenberg, W., Gray J.A., Haynes, R.B., & Richardson, W.S. *Evidence-based medicine: what it is and what it isn't*. Centre for Evidence-Based Medicine, http://cebm.jr2.ox.ac.uk/ebmisisnt.html, 1996.

5. Newman W., & Lamming M., Interactive System Design, Addison-Wesley, 1995.

6. Lim, K.Y, & Long. J. *The MUSE method for usability engineering*. Cambridge series on human-computer interaction, 8, Cambridge: Cambridge University Press, 1994.

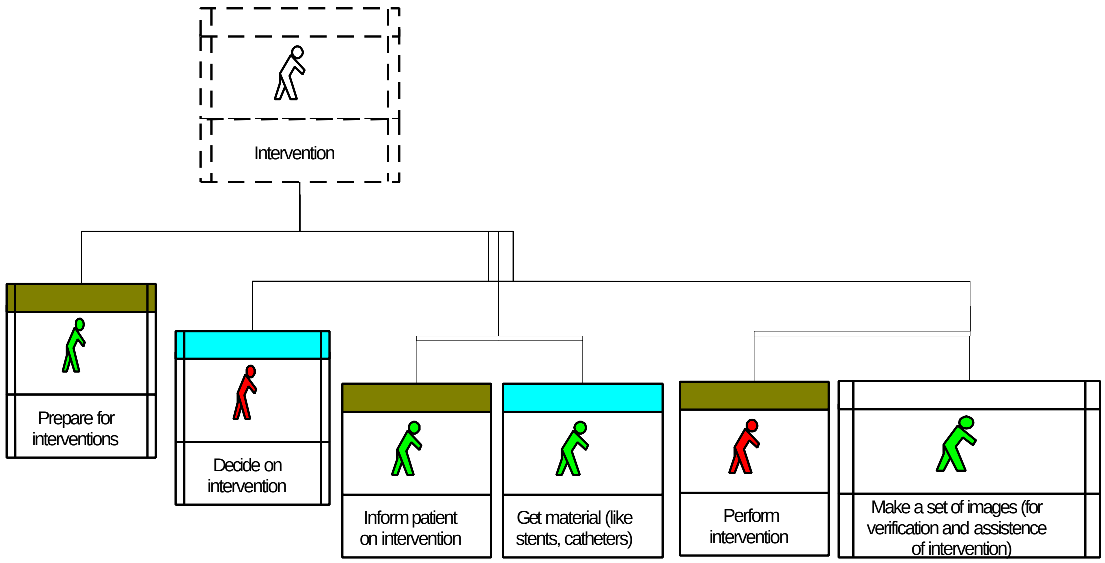
7. Polson P.G., Lewis C., Rieman J. & Wharton C., "Cognitive walkthroughs: A method for theory-based evaluation if user interfaces.", *International Journal of Man-machine Studies*, 36 (1992), pp 741-773.

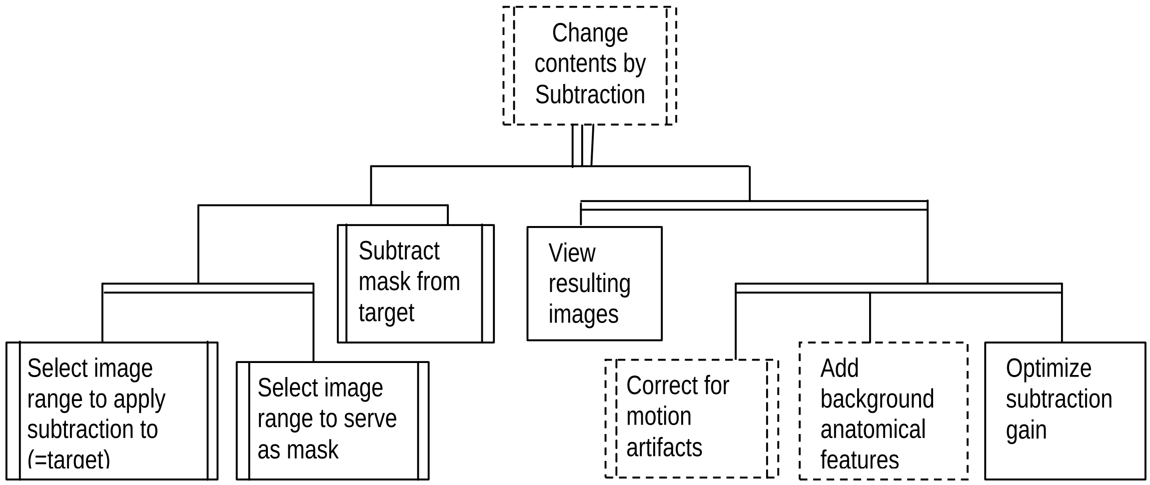
8. Mallen, C. Designing intelligent help for information processing systems. *International Journal of Human-Computer Studies, 45* (1996), 349-377.

Figure 1 - Fragment of a Generalised Task Model of Existing Systems

Figure 2 - Fragment of a Task Model for the Target Application

Figure 3 - Example of Active Support for the Learning Process


Intervention
Prepare for interventions
Decide on intervention
Inform patient on intervention
Get material (like stents, catheters)
Perform intervention
Make a set of images (for verification and assistence of intervention)


Change contents by Subtraction
Subtract mask from target
View resulting images
Select image range to apply subtraction to (=target)
Select image range to serve as mask
Correct for motion artifacts
Add background anatomical features
Optimize subtraction gain

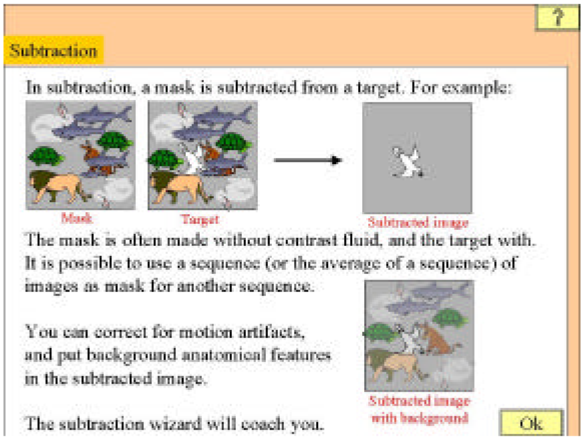


Image 1

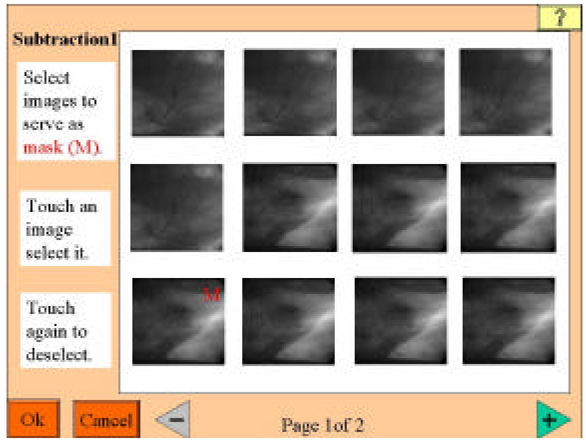


Image 2

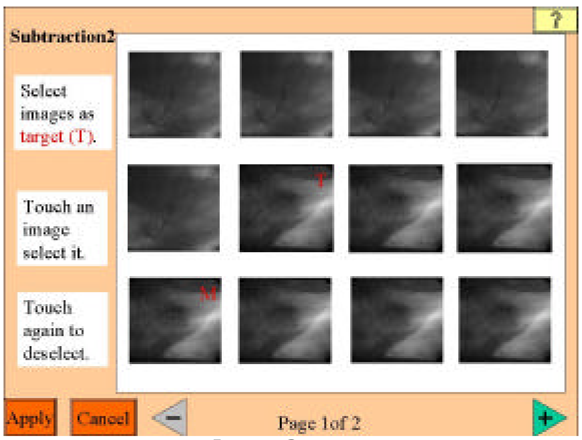


Image 3

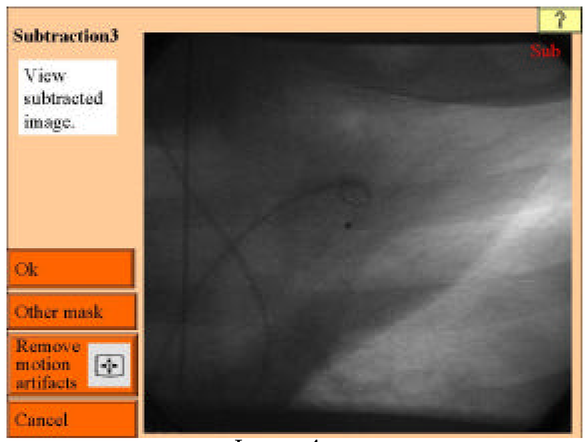


Image 4

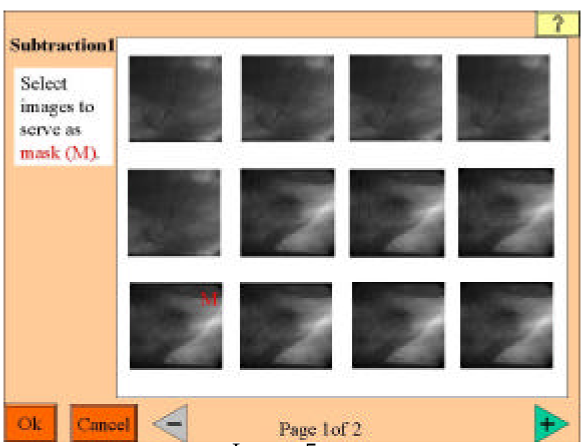


Image 5

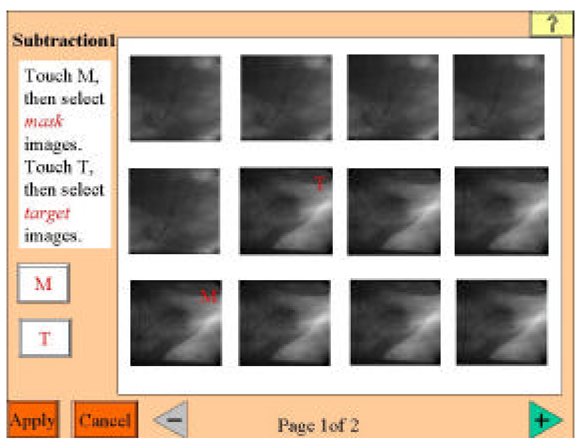


Image 6