Towards Systematic Usability Evaluations for the OR: An Introduction to OR-Use Framework

Ali Bigdelou, Aslı Okur, Max-Emanuel Hoffmann,
Bamshad Azizi, and Nassir Navab

Chair for Computer Aided Medical Procedures (CAMP), TU Munich, Germany

Abstract. It has been shown that usability of intra-operative computer-based systems has a direct impact on patient outcome and patient safety. There are several tools to facilitate the usability testing in other domains, however, there is no such tool for the operating room. In this work, after investigating the features of the existing tools in other domains and observing the practical requirements specific to the OR, we summarize key functionalities that should be provided in a usability testing support tool for intra-operative devices. Furthermore, we introduce the OR-Use framework, a tool developed for supporting usability evaluation for the OR and designed to fulfill the introduced requirements. Finally, we report about several performed tests to evaluate the performance of the proposed framework. We also report about the usability tests which have been conducted up until now using this tool.

1 Introduction

Computer-based systems in the Operating Rooms (OR) have become widely used as a means of improving the treatment process and the patient outcome. On the other hand, this can increase the risks for human error. According to a study published in 2004 [12], between 44,000 to 98,000 patients die each year from preventable medical errors in American hospitals where 69% of these events are rooted in wrong usage of technical equipment. This problem can be targeted by studying the usability of intra-operative devices. Usability can be defined by the ease with which a user can operate, prepare inputs for, and interpret outputs of a system or component. Nielsen states that usability is associated with five main attributes: Learnability, Efficiency, Memorability, Errors, and Satisfaction [10]. This definition has been extended within different industrial standards for certain contexts, e.g. ISO IEC 62366 for the medical domain. Among available usability evaluation approaches, Liljegren [7] suggested that usability testing is more appropriate for medical devices. It consists of four main stages as planning, conducting, analysis and report. Performing a complete usability test is not always an easy endeavor due to many challenges associated with the acquisition and the management of usability data as well as the analysis of huge amount of collected information. In order to overcome these challenges, several usability

P. Abolmaesumi et al. (Eds.): IPCAI 2012, LNAI 7330, pp. 146–156, 2012.

evaluation support tools have been proposed in different domains as reported by Ivory and Hearst [6]. Such tools reduce the required amount of labour and costs assigned to these tests and therefore they can be performed on a frequent, iterative basis, within the development life-cycle. Despite its vital effect, to our knowledge, there is no such support tool for the OR domain and usability study is often neglected due to its complexity and relatively high demand of time and resources. Furthermore, none of the previous frameworks are considering the complexities of the OR, hence making them impractical in this context.

The complex nature of the OR domain makes the production of usable intra-operative devices very challenging. The OR is a collaborative environment where multiple potential users perform together, each with different individual roles e.g. surgeon, nurse. Usually, intra-operative devices are targeting more than one of these roles and therefore their usability should be studied for each individual target role. Additionally, the usages of intra-operative devices are usually fused in activities of a higher level process model known as surgical workflow [9]. In this situation, defining atomic test tasks as it is done for websites or handheld devices is not practical. Moreover, the intra-operative devices are technically much more advanced compared to web sites. They are often compose of additional external hardware such as probes and tracking systems, which are integrated as part of their user interface (UI). Taking into account that interaction with all these external parts of the UI should be also considered within the usability test, the challenges facing the usability specialist in this domain become clear. In [4], we proposed a conceptual model for managing the usability data. This model decomposes the complexities of the OR domain into three views as surgical workflow, human roles and intra-operative device, where cross-view correlations are stored in mapping tables. In this work, we propose the main functionalities required for a usability testing support tool and explain a possible architecture for exploiting models similar to the one proposed in [4] for supporting intra-operative usability testing. Finally, we conclude with a comparison with similar tools and report about our early results from conducted experiments.

1.1 Related Works

There are many tools and frameworks to support usability testing in different domains. Morae [8], Mangold INTERACT and Nodlus Observer are commercially available tools which consist of a recorder and analysis tools. Several tools are further introduced for usability studies of web-based applications such as Web Usability Probe [1]. Technically, they analyze the html source of the web pages and by adding a spy script for each UI control, generate a usage report. Similarly, authors in [3] propose HUIA framework to record and visualize the interaction logs for applications on smart phones. Furthermore, SAVE [5] provides comparable functionalities for augmented reality applications where users interact in a virtual environment. During the analysis stage the recorded data can be played back in the virtual scene, providing interaction information in the same fashion as the recorded videos. In the OR context, there are several methodological studies available on the topic [11]. However, no specialized support tool has been proposed until now. The closest work to our approach is presented by authors in [9] to record surgical workflows. This is a valuable tool to monitor activities within a surgical intervention; however it is not intended to be used for usability studies and does not record any user feedback or interaction logs.

2 Usability Testing for the OR

In this work, we take example of usability study of an intra-operative imaging device, incorporating a navigated handheld gamma probe. This study has been conducted in collaboration with the manufacturer. We present three case studies to highlight some of the issues about the usability of intra-operative devices.

Subjective Satisfaction. This refers to how pleasant users find it to use a system. This is an important measure since it helps to improve the user experience in order to increase customer acceptance. It is mainly evaluated based on heuristic feedback collected from test subjects. As opposed to typical scenarios such as websites, where there is one user working with the system, in the OR this should be evaluated separately for each potential human role. Presenting this information based on different aspects of the OR domain, e.g. workflow stages, it would be possible to prioritize the required improvement based on them.

Learnability per Human Role. Learnability can be defined as a measure of the degree to which interaction with a device can be learned quickly and effectively. It can be measured either with time or comparing number of performed interactions of new users to an optimum set performed by an expert user, accomplishing the very same task. In collaborative environments such as OR, this should be measured for each individual role. Smooth integration and reduced training cost are among the main benefits of a learnable system.

Cross Configurations Efficiency Comparison. In context of iterative evaluations, it is very important to compare the efficiency of the two successive versions. A system is called efficient when a user can use it productively with a minimum amount of resources such as time. One of the most common techniques for evaluating efficiency is measuring the task completion time. This may be achieved using instances running on different locations within multi-center studies.

3 Functionalities

We summarize the requirements of a usability testing support tool based on observations made with medical industry and features available in existing tools.

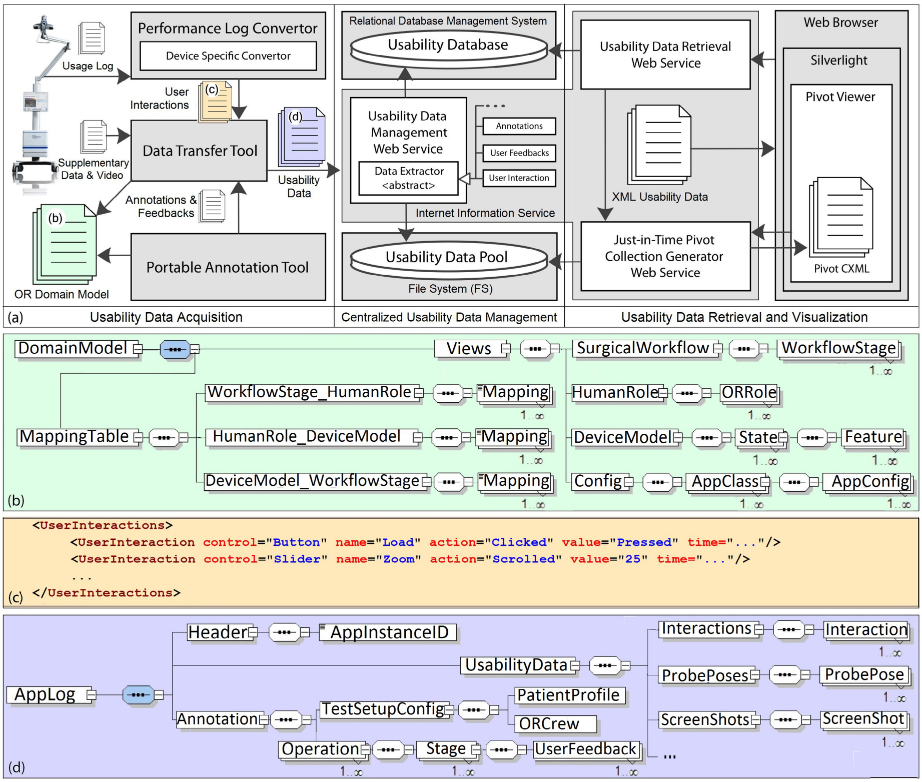
Planning. This is the first step of usability testing and usability support tools in this context should provide the specialist with a proper modeling technique for defining the domain model. For the OR Scenario this includes: (1) *Workflow model* which explains the sequence of activities during a surgical intervention. Having this model, the user interactions can be associated to corresponding stage within the operation. (2) *Human roles* which define different actors within a given surgical intervention. Modeling the OR roles helps to understand the usability measures from the perspective of different users. (3) *Device model* which contains all the features in a system's interface that user can interact with, such as UI controls or handheld probes. This model helps to track the user interactions.

Conducting. During this stage, a wide range of information is synchronously recorded such as: (1) *Video recording*, which is the most common technique in usability testing as it can be used to find out about the effort users make accomplishing a given task. (2) *Performance logging*, which is storing all the users' interactions with different features of the interface. Excessive amounts of clicks, scrolling and probe movements, are possible indicators for poor efficiency. (3) *Recording annotations*, which provides a great insight into the real feelings and comments of the users about the system. A dedicated tool can facilitate this and furthermore be utilized for labeling the start and end of workflow stages. Portability and easy deployment are among the main characteristics of such a tool [9], due to the fact that the monitored surgeries usually take place in different ORs, often with a very short notice. Using a portable device satisfies these needs, however, this tool itself should be highly usable due to time constraints during surgery. Small display size and limited interaction possibilities are among the main challenges for designing an annotation tool for small portable devices like smart phones. (4) *Configuration management*, which includes software and hardware version, test location and date as well as the information about the surgical team such as level of experience. This information provides the required background for analysis of multi-center and cross-configuration tests.

Analysis. Large sets of collected usability data are rarely explanatory on their own and should be analyzed for making conclusions. Several functionalities can be provided to assist the analysis process: (1) *Data retrieval*, which is a fundamental part of many usability support tools that provides a way for searching, filtering and retrieving a subset of the collected usability data. Different aspects of the domain model can be used to filter the data. (2) *Video indexing*, which provides a mean to retrieve the corresponding video segments for each piece of usability data such as user comments or performance logs. This facilitates the analysis stage by reducing the required time for browsing the videos. (3) *Visualization*, including graphs and other diagrams, which should be used where possible to draw attention to the critical issues. These can significantly reduce the time required for exploration and decision making processes.

4 OR-Use Framework

In this section we describe the architecture and main components of the OR-Use framework, shown in Figure 1 (a), developed to achieve the above mentioned requirements. According to the classification of Ivory and Hearst [6], the OR-Use framework solution for usability testing involves: capturing logs generated at client-side and storing them on the server-side, supporting analysis and a number of visualizations to ease the identification of the usability issues, and can



```
<UserInteractions>
    <UserInteraction control="Button" name="Load" action="Clicked" value="Pressed" time="..."/>
    <UserInteraction control="Slider" name="Zoom" action="Scrolled" value="25" time="..."/>
    ...
</UserInteractions>
```

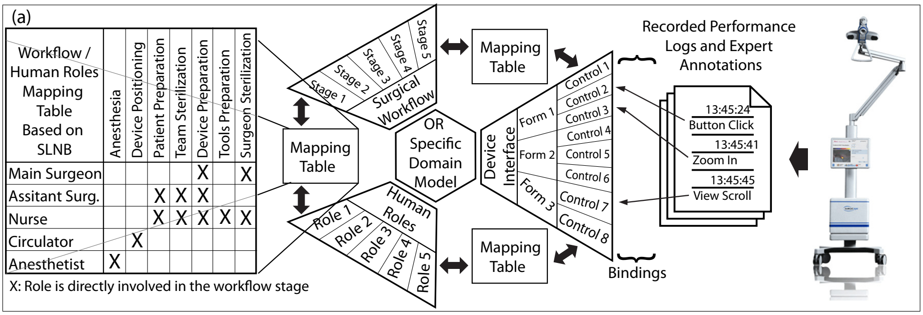
Fig. 1. Architecture of the OR-Use framework and proposed data schema

be used both for test tasks and real usage of the device. In this work we have exploited the modeling technique presented in [4]. Three views of the surgical workflow, human roles, devices and their mapping are defined in an XML file, in the planning phase. Figure 1 (b) demonstrates the schema of this XML file. Different parts of the OR-Use framework use this file as settings for initialization.

4.1 Usability Data Acquisition

In order to collect information during the usability tests, two different tools are provided. Additional to the collected data with these tools, captured video and other device specific files can be attached to the usability data. The timing is synchronized among different modules using an online time web service.

Performance Log Conversion Tool. Wide range of technologies is being used by producers of intra-operative devices, which often varies widely based on their specific requirements. In order to stay independent and extend the functional domain of the OR-Use framework, a conversion kernel is developed to convert the logs generated by a specific device to a uniform XML representation. For



Workflow / Human Roles Mapping Table Based on SLNB	Anesthesia	Device Positioning	Patient Preparation	Team Sterilization	Device Preparation	Tools Preparation	Surgeon Sterilization
Main Surgeon					X		X
Assitant Surg.			X	X	X		
Nurse			X	X	X	X	X
Circulator		X					
Anesthetist	X						

X: Role is directly involved in the workflow stage

View Name	Level 1	Level 2	Level 3
Surgical Workflow	Surgeries	Workflow Stages	
Device Model	Device Configurations	Device States	Device Features
Human Roles	Human Roles		
(b) Configuration	Device Classes	Device Configurations	Device Instances

Fig. 2. (a) Structuring the usability data, (b) Relational database tables

each new device, a component can be provided and added to the conversion kernel. As shown in Figure 1 (c) the performance log XML file consists of a list of interactions. After each test session this tool is used to export the device performance logs.

Portable Annotation Tool. A portable annotation tool is developed for Android smart phone devices. This tool facilitates onsite documentations of specialist's observations. Before starting the test, for each member of the surgical team a test subject profile can be either defined or selected among the previously stored profiles. This profile contains information such as name, role and level of experience. Furthermore, during the surgery this tool is used to follow the surgical workflow stages as defined in the domain model. For each workflow stage, a button is automatically generated on the workflow page and can be used to annotate its start during the intervention. Within each workflow stage, comments can be added in two different ways: typing or voice recording. Additionally the type of OR members' comments, such as complaints or positive feedback can be specified. After the operation, all the recorded data is exported in an XML file.

4.2 Centralized Usability Data Management

For supporting cross configuration and multi-center usability studies, the OR-Use framework has been designed based on client-server architecture, as shown in Figure 1 (a). All the captured data are transferred to a server, hosting this information. Here we explain relevant components of this design.

Data Transfer. A client-side tool is developed to facilitate the uploading process. This application merges and encrypts different generated XML files, performance logs and test annotations, into a single XML message. The structure of this XML message is shown in Figure 1 (d). Multiple instances of this tool can be used simultaneously. Since the message contains all the required information about the test setup configuration, it can facilitate evaluation of different devices with different configurations conducted in various locations and hospitals.

Server Side Extraction. On the server side, the received data are decrypted, extracted and stored on the server. Each part of the XML message is processed with a corresponding data extractor component, loaded at runtime. Data extractors parse the XML data and store them in the corresponding tables of a database. Supplementary materials are separately stored on the local file system of the server, forming a repository named data pool. Each extractor component is developed for processing a special type of data, e.g. user interactions or probe readings. Such an open architecture allows the proposed framework to be extended for new devices and makes it applicable for different usability standards.

Data Management. Retrieved data on the server side is stored in a relational database. As shown in Figure 2 (a), a hierarchical representation is used in each view [4]. The depth of this hierarchy depends on level of granularity required and is defined in the domain model file. As shown in Figure 2 (b), a table is created in the database for each level of this hierarchical representation. A foreign key is used to specify the relation to the parent of each node. Usability data are separately stored, each in a dedicated table such as interactions, comments, etc.

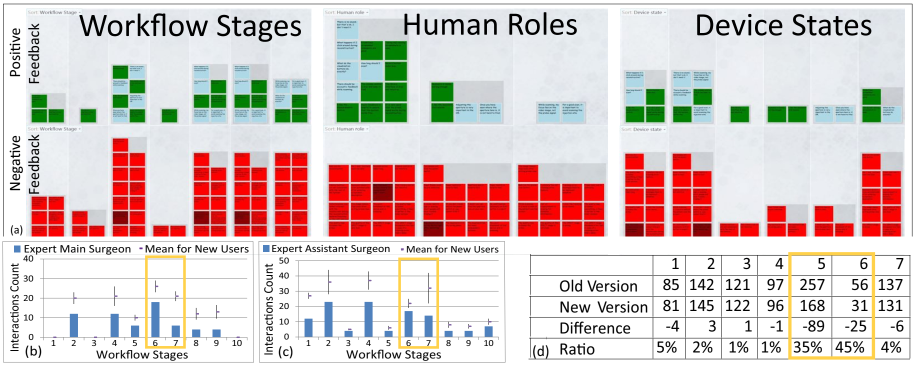
4.3 Usability Data Retrieval and Visualization

Two web services have been developed, to access the stored usability data on the OR-Use server either directly (low-level approach) or using visualization (high-level approach). The web-based nature of these services allows multiple and simultaneous access to the data, which is important for comparison.

Usability Data Retrieval Interface. The stored usability data on the OR-Use server can be retrieved in form of XML, from standard web browsers, sending a query to a data retrieval web service, shown in Figure 1 (a). This query contains a set of key and value pairs, which are used to filter the retrieved results. The first key is *collection*, which specifies the usability data type. For all the database tables shown in Figure 2 (b), a key with similar name is defined. On the data retrieval service, this query is parsed and a SQL command is generated and executed against the database. The retrieved results are then returned as XML, which can be used for the development of additional analysis support tools such as data visualization or data mining. For example, the following query allows the usability engineer to retrieve the interactions that the surgeon has done with the brightness slider, during the scanning stage, on version 2.3 of the device:

```
HostAddress?collection="UserInteraction"&Configuration="2_3"&Workf
lowStage="Scanning"&HumanRole="Surgeon"&DeviceFeature="Brightness"
```

Usability Data Visualization. A high level visualization of the usability data can facilitate the analysis process by reducing the analyzing time. Such interface has been developed, using Microsoft Pivot and CXML file, which is an XML file



	1	2	3	4	5	6	7
Old Version	85	142	121	97	257	56	137
New Version	81	145	122	96	168	31	131
Difference	-4	3	1	-1	-89	-25	-6
Ratio	5%	2%	1%	1%	35%	45%	4%

Fig. 3. Results for case studies: a)Satisfaction, b-c)Learnability, d)Efficiency

expressing a collection of items and their properties. A web service is developed to generate the CXML files using "Just-in-Time" method that makes it possible to dynamically produce collections, based on the user request. Same format as data retrieval interface is used for queries. The Pivot web service generates the CXML file based on the retrieved results from the data retrieval interface. Facets are assigned to each item based on the available information in the corresponding table and additionally other properties traceable via mappings. An image is generated per item, which simply represents its content e.g. user comments are shown as texts with a background color assigned based on their type. Also, a proper segment of the recorded video is extracted and attached, using the time of the usability data item. This is equivalent to typical video indexing.

5 Case Studies

Until the middle of November 2011, the OR-Use is used to conduct usability tests in 12 user studies and 7 surgeries took place in our partner university hospital. 5 surgeons and 10 biomedical students were involved as test subjects. A surgical workflow was modeled with 18 stages and 5 main intra-operative human roles were defined. The device modeled with 122 features in 12 stages. Within 85 performed workflow stages, about 2000 user interactions and 163 user feedbacks, annotated with one of 8 comment types have been collected, using thinking allowed. Here we report on the results for the given case studies.

Subjective satisfaction is evaluated using direct feedbacks received from the users through thinking aloud process. Figure 3 (a) shows these feedbacks, presented based on surgical workflow, human roles and device states.

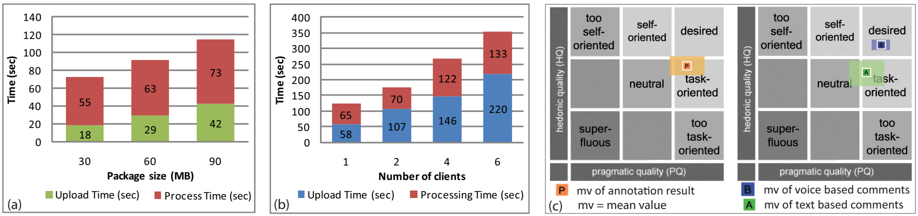
Learnability per human role has been evaluated by comparing the number of interactions, required to accomplish a given task, between an expert user and new users. Figure 3 (b-c) shows these results per human role, distributed over different workflow stages. Several points can be highlighted, e.g. the close results of an expert and new users in stage 6 shows that this stage is much more learnable for surgeons compared to stage 7 where this difference is larger.

Cross configurations efficiency comparison is performed, computing the task completion time using two different versions of the application. Figure 3 (d) shows these times (in seconds), in the newer version a visualization parameter has been computed automatically, removing the need for manual tuning. The difference is higher in stages where this feature is used (5 and 6).

6 Framework Evaluation

The effectiveness of some existing tools which are used in other domains is examined and compared to the OR-Use framework. Figure 4 (d) demonstrates how well each tool meets the functional requirements, discussed in Section 3. In order to evaluate the performance of the proposed client-server architecture, we have measured the uploading and processing time with data of 30, 60 and 90 MB size, as shown in Figure 4 (a), running on a 2.5 GHz Intel Core 2 Duo machine with 4 GB of memory connected via a shared wireless network. This data was collected in 3 different test sessions, lasting about 15 minutes. Moreover, we measured the uploading and processing times when several clients simultaneously accessed the server, uploading data of 100 MB size, as shown in Figure 4 (b).

These diagrams demonstrate that by increasing the size of data and the number of clients, the processing time does not change as much as the uploading time. This means that the main bottleneck of the whole process is the uploading phase and it highlights the importance of the bandwidth in a larger setup. Also, to evaluate the usability of the proposed portable annotation tool, we conducted an ad-



(d)	Functionalities	M	INT	NDL	HUA	WUP	SAV	ORU
Planning	Workflow/Task modeling	X	X	X	X	X	X	O
	Multiple roles	X	X	X	X	X	X	I
	Device model	X	X	X	I	O	X	O
Conducting	Performance logging	I	I	I	I	I	I	I
	Observers annotations	I	O	O	X	X	I	I
	Video & supplementary materials	X	I	I	X	X	I	I
	Configuration management	O	X	X	O	O	O	I
Analysis	Data retrieval	I	I	I	I	I	I	I
	Video indexing	X	I	I	X	X	I	I
	Visualization	I	I	I	I	O	O	I

Legend:
I = Implemented
o = Achieved to some extent
x = Not supported

M = Morae
INT = INTERACT
NDL = Nodlus
HUA = HUIA
WUP = Web Usability Probe
SAV = SAVE
ORU = OR-Use

Fig. 4. OR-Use a-b)Performance evaluation, c)Usability study, d)Comparison

ditional user study with 12 participants (biomedical students aged 22 to 32). After performing a set of tasks based on real activities that a usability specialist should perform during an operation, users were asked to fill three AttrakDiff [2] questionnaires. One about profile management and workflow follow up functionalities and two about the two alternate methods for documenting the user feedbacks. The results, shown in Figure 4 (c), which place this tool in terms of pragmatic quality (PQ) and hedonic quality (HQ). PQ addresses different aspects of human needs and usability factors related to control, learnability and ease of use. HQ deals with human desires for excitement, including novelty and satisfaction. The profile management and workflow follow-up is categorized as "Task-Oriented". High PQ value means that these features have high usability factors. The large confidence area in both dimensions highlights the fact that users had diverse ideas about these features. Furthermore, keyboard-based and voice-based feedback documentation methods have been compared, where the latter is rated as more usable and interesting for intra-operative usability studies.

7 Conclusion

Usability is very important for intra-operative devices, because those with poor usability can increase the chance of human error. Having a tool for supporting different aspects of usability testing can increase testing efficiency and improve the usability of intra-operative devices. Ideally, usability testing support tools should capture a wide range of inputs, manage and organize the collected data on the complete model of the OR and support analysis of the collected data via a proper set of data retrieval and visualization interfaces. Here, we introduced the OR-Use framework and demonstrated how the proposed architecture can address most of these requirements, as a usability support tool for the OR. Conducting more tests with different devices and providing a data mining interface to support the decision making process have to be the central aspects of future work.

References

1. Ardito, C., Costabile, F., De Marsico, M., Lanzilotti, R., Levialdi, S., Roselli, T., Rossano, V.: An approach to usability evaluation of e-learning applications. Univers. Access Inf. Soc. 4, 270–283 (2006)
2. AttrakDiff, http://www.attrakdiff.de/en/Home/
3. Au, F.T.W., Baker, S., Warren, I., Dobbie, G.: Automated usability testing framework. In: Ninth Australasian User Interface Conference, vol. 76, pp. 55–64. ACS, Wollongong (2008)
4. Bigdelou, A., Sterner, T., Wiesner, S., Wendler, T., Matthes, F., Navab, N.: OR Specific Domain Model for Usability Evaluations of Intra-operative Systems. In: Taylor, R.H., Yang, G.-Z. (eds.) IPCAI 2011. LNCS, vol. 6689, pp. 25–35. Springer, Heidelberg (2011)
5. Holm, R., Priglinger, M., Stauder, E., Volkert, J., Wagner, R.: Automatic data acquisition and visualization for usability evaluation of virtual reality systems. In: EUROGRAPHICS 2002. The Eurographics Association (2002)

6. Ivory, M.Y., Hearst, M.A.: The state of the art in automating usability evaluation of user interfaces. ACM Comput. Surv. 33, 470–516 (2001)
7. Liljegren, E.: Usability in a medical technology context assessment of methods for usability evaluation of medical equipment. Int. J. of Ind. Erg. 36(4), 345–352 (2006)
8. Morae, `http://www.techsmith.com/morae.html`
9. Neumuth, T., Durstewitz, N., Fischer, M., Strauss, G., Dietz, A., Meixensberger, J., Jannin, P., Cleary, K., Lemke, H.U., Burgert, O.: Structured recording of intraoperative surgical workflows. In: Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 6145, pp. 54–65 (2006)
10. Nielsen, J.: Usability Engineering. Academic Press Inc., London (1993)
11. Strauss, G., Koulechov, K., Rttger, S., Bahner, J., Trantakis, C., Hofer, M., Korb, W., Burgert, O., Meixensberger, J., Manzey, D., Dietz, A., Lth, T.: Evaluation of a navigation system for ENT with surgical efficiency criteria. The Laryngoscope 116(4), 564–572 (2006)
12. Wachter, R.M.: The End of the Beginning: Patient Safety Five Years After 'To Err Is Human'. Health Affairs (2004)