INTERACTION BETWEEN A SURGEON AND A COMPUTER ASSISTED SURGERY SYSTEM: AN INTERACTIVE DESIGN SPACE

B. MANSOUX (1,2), L. NIGAY (1), J. TROCCAZ (2)

[1] CLIPS-IMAG lab., HCI team – University of Grenoble – 385, rue de la Bibliothèque – 38041 Grenoble cedex 9 – France
[2] TIMC-IMAG lab., CAS team – Faculté de Médecine – Domaine de la Merci – 38706 La Tronche cedex – France

INTRODUCTION AND GOALS

Computer-Assisted Surgery (CAS) systems aim to help a surgeon in defining and executing an optimal surgical strategy based on a variety of multimodal data inputs. As part of a multidisciplinary project that involves the (Human-Computer Interaction) HCI and the CAS research groups of the University of Grenoble, our research aims at providing elements useful for the design of usable CAS systems by focusing on the interaction between the surgeon and the CAS system.

CAS systems often rely on Augmented Reality (AR) interaction techniques [8], which are based on the fusion of the digital world (e.g. MRI, scan images, computed trajectory) with the real world (e.g. the patient's body, a needle). In [1] common properties of AR techniques have been proposed :
1) combining the digital and real worlds, while maintaining interaction in the real environment,
2) real-time interaction,
3) registration in 3D, that refers to the accurate alignment of digital and real objects.

However, nowadays there is no consensus on a definition of AR techniques highlighting the problem of delimitating the frontier between the digital and real worlds. For instance, let us consider the system of Fig. 1, where a 3D brain model is superimposed onto a real scene video [6]. For [10], such a system is based on an AR technique because the real scene images are modeled as real objects combined with a digital object, the 3D brain model. As opposed to [10], in [2] we consider that the real scene video is a digital object combined with another digital object, the 3D brain model. So as defined in [2], such a system combining digital objects may not rely on an AR technique.

In addition to the absence of consensus on a definition of an AR technique, we must also emphasize the fact that an AR technique does not only rely on the human visual sense. Although most of the existing AR techniques are based on the visual sense by overlaying digital objects on top of physical objects, other human senses can be involved. For instance, guidance information in CAS systems can be provided to the surgeon thanks to haptic devices such as in [13] using electro-stimulation on the tongue via a matrix of 144 electrodes as shown in Fig. 2.

Fig. 1: Video overlay [6].

Fig. 2: Tongue Display Unit (TDU) [13].

The variety of surgical needs and specialties as well as the multitude of interaction devices have led to many specific AR interaction techniques and therefore CAS systems that bring real clinical improvements. Nevertheless the design of such interaction techniques is still ad hoc and generally driven by technologies. We aim at providing a design space for interaction between a surgeon and a CAS system, so as to capitalize on all the previous work on AR interaction techniques and to identify reusable generic design solutions. This work complements our previous study presented in [4] where we proposed an analytical approach for the design of the surgeon's interaction with a CAS system.

In this paper we present a design space that is useful in the context of a top-down (abstract-concrete) design method such as the one we described in [4]. The design space consists of an organized framework of abstract interaction situations. Useful at an early stage of the interaction design, it enables the designer to explore the set of design possibilities without being biased by any particular technology.

METHOD

As part of a top-down interaction design method, in [4] we have presented a set of generic functions for CAS systems that are independent of a surgical specialty. These functions are classified in terms of input functions (exchanges of information from the surgeon to the CAS system), output functions (exchanges of information from the

CAS system to the surgeon) and triggering functions such as the use of a pedal to start an automatic acquisition of images. This set of functions is useful during the early stage of the design as a tool for helping to identify the required interaction functions. As a next step and before selecting the concrete devices and therefore interaction modalities, one design step consists of deciding what the nature (i.e., physical or digital) of the objects involved is, while performing such functions. In other words, the designer decides what the interaction situation is.

In order to describe such interaction situations, we partly reuse the ASUR notation [3] dedicated to describing concrete interaction in CAS systems and more generally AR systems. From the ASUR notation we reuse the entities but we do not reuse the characteristics of entities and relations that are dedicated to concrete interaction.

ASUR entities

Four entities are identified in ASUR: Adapter, System, User and Real objects. Those entities take part in the interaction by exchanging data between each other. Between the user (U) and the computing system (S), the adapters bridge the gap between the physical world and the digital one. They could be input adapters (A_{in}) (e.g. a mouse, an optic tracker) or output ones (A_{out}) (e.g. a screen, audio speakers, a video projector). The real objects belong to the user's physical environment and also take part in the interaction. They could be either the object of the task (R_{object}) (e.g. the patient's body) or a tool (R_{tool}) (e.g. a needle).

ASUR example

Fig. 3: The CASPER system in use.

In order to illustrate the ASUR entities, we present the ASUR diagram that describes the concrete interaction using our CASPER application of Fig. 3. CASPER (Computer ASsisted PERicardial puncture) is a system that we developed for computer assistance in pericardial punctures. In Fig. 4, we present the ASUR diagram of CASPER.

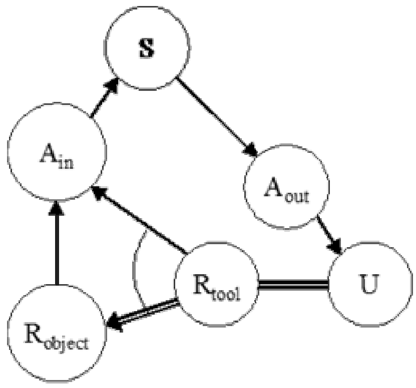

Fig. 4: ASUR diagram of the concrete interaction in CASPER. For a complete ASUR description, the diagram is completed by the characteristics of each entity and relation (see [3]).

During the surgery, CASPER assists the surgeon (U) by providing in real time the position of the puncture needle (R_{tool}) according to the planned trajectory. Two adapters (A_{in}, A_{out}) are necessary: The first one (A_{out}) is the screen for displaying guidance to the surgeon, and the second one (A_{in}) is dedicated to tracking the needle position and orientation as well as the patient's body (R_{object}). The localization of the needle is possible within a predefined volume near the patient's body. Such a constraint is represented in Fig. 4 by an ASUR relation ⇒ (physical activity triggering an action). The concrete interaction description of Fig. 4 is not complete. The ASUR diagram is completed by the characteristics of the identified entities and relations. A complete description of the concrete interaction in ASUR can be found in [3].

In the following section, we present our design space of described interaction situations using the ASUR entities.

RESULTS

Interaction situation design space

Our design space is made of eight interaction situations that are independent of the concrete interaction techniques or interaction modalities. This set of interaction situations enables the designer to explore the set of abstract design possibilities without being biased by any technology. As for our generic functions in [4], our framework is composed of input and output situations. In addition our approach for structuring the framework of interaction situations draws from our distinction of direct or indirect interaction. Indeed we qualify an interaction as being direct when the object of the task (R_{object}) is involved in the situation and indirect when not. We obtain a framework made of eight interaction situations, four input situations amongst which two are indirect and two direct as well as four output situations again two being indirect and two direct.

Input situations:

Indirect

(i-1) $U \rightarrow A_{in} \rightarrow S$

(i-2) $U \rightarrow R_{tool} \rightarrow A_{in} \rightarrow S$

Direct

(i-3) $U \rightarrow [R_{tool}, R_{object}] \rightarrow A_{in} \rightarrow S$

(i-4) $U \rightarrow R_{object} \rightarrow A_{in} \rightarrow S$

Output situations:

Indirect

(o-1) $S \rightarrow A_{out} \rightarrow U$

(o-2) $S \rightarrow A_{out} \rightarrow R_{tool} \rightarrow U$

Direct

(o-3) $S \rightarrow A_{out} \rightarrow [R_{tool}, R_{object}] \rightarrow U$

(o-4) $S \rightarrow A_{out} \rightarrow R_{object} \rightarrow U$

Illustrations of interaction situation

The first situation qualified as input and indirect (i-1) depicts a classical interaction with a computer, for example using a mouse to press a button on screen in order to start the display of the guidance information while interacting with our CASPER system (Fig. 3). The second situation (i-2) describes the case where the user manipulates a physical object (R_{tool}) to interact with the computer via an adapter that captures

the manipulations. An example of such a situation would be the case where the surgeon moves the needle in front of the cameras in order to start the display of guidance information. More generally speaking, examples of such (i-2) input situations, in the HCI domain, are the physical icons that are physical handles to digital objects, "coupling the bits with everyday physical objects and architectural surfaces" [7].
One situation qualified as input and direct is the following one: $U \rightarrow [R_{tool}, R_{object}] \rightarrow A_{in} \rightarrow S$ (i-3). This situation depicts the interaction using our CASPER system. Indeed, during the puncture task, the surgeon is handling the puncture needle (R_{tool}) that touches the patients body ($[R_{tool}, R_{object}]$). Both the needle and the patient are localized by the system via adapters (A_{in}).

An example of output indirect interaction situation is the following one: $S \rightarrow A_{out} \rightarrow U$ (o-1). This situation depicts a classical interaction with a computer. For example using CASPER, during the puncture task, the surgeon (U) perceives guidance information displayed on a screen (A_{out}). The output situation using the PADyC (*Passive Arm with Dynamic Constraints*) system [12] corresponds to the case (o-3): $S \rightarrow A_{out} \rightarrow [R_{tool}, R_{object}] \rightarrow U$. Indeed using PADyC, the surgeon is handling a surgical tool that is linked to a passive arm (A_{out}). The programmable arm enables us to provide haptic guidance information (touch feedback) to the surgeon while performing the surgery. Another example of an output interaction technique is based on the use of a see-through Head-Mounted Display (HMD) to superimpose digital guidance data with the patient's body as shown in Fig. 5. This interaction technique corresponds to the (o-4) interaction situation.

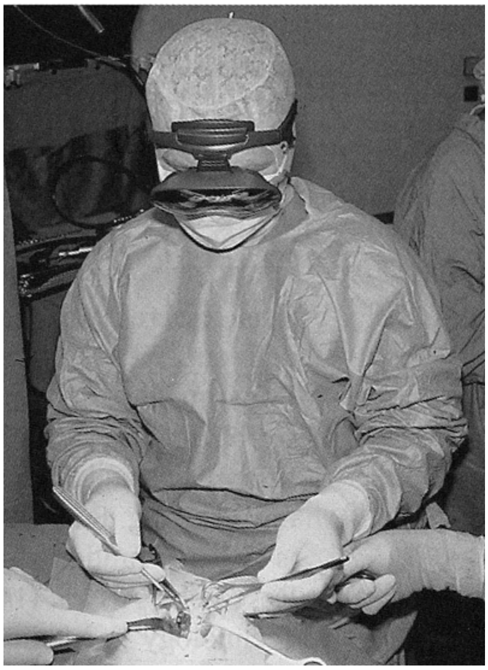

Fig. 5: Overlay using a Head-Mounted Display [11].

Completeness of the interaction situation design space

For each input/output as well as direct/indirect situation, we describe all the combination possibilities of ASUR entities, making the design space complete. Nevertheless for each situation the described chain made of ASUR entities is the minimal one. While making the abstract situation concrete, some ASUR entities may be inserted in the minimal chain. For example in the situations (i-3) and (i-4) we suggest that the user and the object of the task are physically together. In the case of telesurgery for example, the surgeon (user) and the patient (object of the task) are distant. Such situations are described by adding a chain that comprises the computer system (S) between:

- the user (U) and the tool ($[R_{tool}, R_{object}]$) for situation (i-3)
- the user (U) and the object of the task (R_{object}) for situation (i-4).

An example of chain to be added will be: ($A_{in} \rightarrow S \rightarrow A_{out}$). For example the following situation:

$U \rightarrow (A_{in} \rightarrow S \rightarrow A_{out}) \rightarrow [R_{tool}, R_{object}] \rightarrow A_{in} \rightarrow S$

belongs to the (i-3) class of situations.

CONCLUSION AND FURTHER WORK

The interaction situation design space is useful at an early stage of the design of CAS systems: indeed it enables the designer to systematically explore the set of possibilities without being biased by the available technologies. While our situation design space for abstract interaction is complete, further work must be done for concrete interaction.

The transition from interaction situation to concrete interaction is difficult because the set of possibilities in terms of concrete interaction techniques or modalities is huge. Indeed from a given abstract interaction situation, several concrete interaction solutions can be designed. Further work must therefore be done for characterizing input/output interaction modalities. We have started this work for a given device, a mini-screen: in [9] we propose a design space that characterizes the input/output modalities that are based on a mini-screen.

BIBLIOGRAPHY

[1] AZUMA, R. ET AL.
Recent Advances in Augmented Reality
IEEE Computer Graphics and Applications, 2001, 21(6), 34-47

[2] DUBOIS, E. ET AL.
Classification Space for Augmented Surgery, an Augmented Reality Case Study
Proc. of Interact'99, Edinburgh - UK, 1999, 353-359

[3] DUBOIS, E., NIGAY, L., TROCCAZ, J.
Consistency in Augmented Reality Systems
Proc. of EHCI'01, 2001, LNCS 2254, Springer-Verlag, 117-130

[4] DUBOIS, E. ET AL.
Analytical Approach for the Design of the Surgeon's Interaction with a Computer Assisted Surgery System
Proc. of Surgetica 2002, 2002, Sauramps Medical, 211-216

[5] FISHKIN, K.P., MORAN, T.P., HARRISON, B.L.
Embodied User Interfaces: Towards Invisible User Interfaces
Proc. of EHCI'98, 1998, Kluwer Academic, 1-18

[6] GRIMSON, E. ET AL.
An Automatic Registration Method for Frameless Stereotaxy, Image Guided Surgery, and Enhanced Reality Visualization
IEEE Transactions on Medical Imaging, 15(2), (1996), 129-140

[7] ISHII, H., ULLMER, B.
Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms
Proc. of CHI'97, 1997, ACM Press, 234-241

[8] MACKAY, W.
Augmented Reality: linking real and virtual worlds
Proc. of AVI'98, (1998), ACM Press

[9] MANSOUX, B., NIGAY, L., TROCCAZ, J.
Abstract and Concrete Interaction with Mixed Reality Systems
Proc. of International Workshop on Exploring the Design and Engineering of Mixed Reality Systems, MIXER'04, 2004, 12-18

[10] MILGRAM, P.
A taxonomy of mixed reality visual displays
IEICE Transactions on Information Systems, (1994), E77-D(12): 1321-1329

[11] SALB, T. ET AL.
Interoperative Augmented reality for Craniofacial Surgery: The INPRES System
Proc. of Surgetica 2002, 2002, Sauramps Medical, 191-197

[12] TROCCAZ, J., DELNONDEDIEU, Y.
Semi-Active Guiding Systems in Surgery: A Two-DOF Prototype of the Passive Arm with Dynamic Constraints (PADyC)
Mechatronics (1996), 6(4), Elsevier Science, 399-421

[13] VAZQUEZ-BUENOSAIRES, J., PAYAN, J., DEMONGEOT, J.
Evaluation of a lingual interface as a passive surgical guiding system
Proc. of Surgetica 2002, 2002, Sauramps Medical, 232-237