Abstract and Concrete Interaction with Mixed Reality Systems

The case of the mini screen, a new interaction device in Computer-Assisted Surgery

Benoit Mansoux, Laurence Nigay

Laboratoire CLIPS-IMAG

BP 53

38041 Grenoble cedex 9

{mansoux, nigay}@imag.fr

Jocelyne Troccaz

Laboratoire TIMC-IMAG

I. I. I. S. -Faculté de Médecine

38706 La Tronche cedex

jocelyne.troccaz@imag.fr

ABSTRACT

In this paper we focus on the design of mixed reality (MR) systems. We propose two design spaces that can be useful in a top-down (abstract to concrete) design method for MR systems. The first design space consists of an organized framework of abstract interaction situations for describing mixed systems. Each situation is depicted by an ASUR diagram and describes the exchange of information between the entities involved in a mixed system. The situations are abstract because they are independent of the interaction modalities (both interaction languages and devices). The abstract interaction situations are illustrated by several Computer-Assisted Surgery (CAS) systems. Such a framework is useful for the designer in order to systematically explore the set of possibilities at an early stage of the interaction design, without being biased by a particular technology. With the interaction situation described, the designer can then focus on the modalities to be used: both passive and active modalities can be elected. This design stage consists of concretizing the interaction situation by selecting the modalities. For this stage of the design, we propose a design space that characterizes the possible usages of one particular innovative interaction device for CAS systems: a mini screen. We illustrate the complementarity of our two design spaces by presenting two CAS systems that embed a mini screen for different purposes in the interaction: one system is based on a localized mini screen fixed on the surgical tool while the other involves the surgeon handling the mini screen on top of the patient's body.

Keywords

Mixed Reality, Computer Assisted Surgery, Design Space, Interaction Device, Mini screen

INTRODUCTION

In this paper, we focus on the design of Mixed Reality (MR) systems. We focus on MR systems that assist a user in performing a task on a physical object (a class of MR systems called "Augmented Reality" in [2]). One of our main application domains for such MR systems is Computer Assisted Surgery (CAS), in the context of a multidisciplinary project that involves the HCI and the CAS research groups of the University of Grenoble. The main objective of CAS systems is to help a surgeon in defining and executing an optimal surgical strategy based on a variety of multi-modal data inputs. MR systems play a central role in the CAS domain because the key point of a CAS system is to "augment" the physical world of the surgeon: the operating theater, the patient, the surgical tools etc., by providing pre-operative information during the surgery. MR systems are now entering many surgical specialties and such systems can take on the most varied forms. Although many CAS systems have been developed and provide real clinical improvements, their design is ad-hoc and principally driven by technologies.

In this context, our research aims at providing elements useful for the design of usable MR systems by focusing on the interaction between the user and the MR system. We present two design spaces that can be useful in a top-down design method for MR systems. The first design space, presented in the second section of the paper, consists of an organized framework of abstract interaction situations for describing MR systems. This first result is useful at an early stage of the design of MR systems: indeed it enables the designer to systematically explore the set of possibilities without being biased by the available technologies. While this first design space focuses on abstract interaction (i.e., independent of the interaction technologies), our second design space, presented in the third section of the paper, characterizes the possible usages of one particular interaction device, a mini screen. Our two design spaces are therefore complementary and address different stages of a top-down design method of MR systems: abstract versus concrete interaction. Before presenting our two design spaces, we first clarify the two interaction design steps, i.e. the design of the abstract and concrete interaction.

ABSTRACT AND CONCRETE INTERACTION

We call interaction situation, an abstract description of the interaction involved in an MR system. Such a description is independent of the interaction modalities. We define in [7] a modality as the coupling of a physical device with an interaction language. After describing the interaction situation, the following step in the design consists of concretizing the abstract situation by choosing the modalities: the description of the interaction is then concrete. For describing the abstract and concrete interaction, we use the ASUR notation [2][3]. In the following paragraph, we summarize the main characteristics of the notation. We then describe how to use the ASUR notation for describing the abstract and concrete interaction.

ASUR notation

ASUR [2][3] stands for "Adapter", "System", "User", "Real objects". In user-centered MR systems are described in terms of entities (A, S, U, R) taking part in the interaction and the relations between those entities. Between the user (U) and the computer 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, a localization mechanism) or output ones (A_{out}) (e.g., a video projector, audio speakers). Physicality is one key feature of MR systems: real objects are involved in the task. Within the ASUR notation we distinguish physical objects that are tools (Rt_{ool}) for performing the task, from the ones that are the objects of the task (R_{object}).

Three kinds of relationship between two ASUR entities are identified:

Exchange of data is represented by an arrowed line between two ASUR entities ($A{\rightarrow}B$).

Physical activity triggering an action: a double-line arrow ($A{\Rightarrow}B$) denotes the fact that when the entity A meets a given spatial constraint with respect to entity B, data will be exchanged along another specified relationship ($C{\rightarrow}D$).

Physical collocation is represented by a non-directed double line ($A=B$). This refers to a persistent physical proximity of two entities.

Finally, the ASUR entities and relationships are described by a set of characteristics. Table 1 presents some of them. For example the first characteristic induced by the use of a real object (R) or an adapter (A) is the human sense involved in perceiving data from such an entity or in performing actions using such entity. The most common used ones are the haptic, visual and auditory senses. A second characteristic is the location where the user has to focus with the required sense, in order to perceive/manipulate the real entity as well as to manipulate the adapter or perceive the data provided by it. In addition one characteristic of a relation between two ASUR entities is the interaction language used to express data carried by the relation. If we refer to our definition of a modality [7] as the coupling of a physical device with an interaction language, the device is described by an ASUR entity while the interaction language is a characteristic of the relation from this entity (device) to another ASUR entity.

Entities (R and A) characteristics	Relationships characteristics
- Perceptual/Action sense and location	- Interaction language

Table 1: Characteristics of ASUR entities and relationships.

Two levels of abstraction in describing interaction using ASUR

In [3], we explained how we use the ASUR notation during the requirements definition phase for describing usage scenarios and during the external specification phase for describing the concrete designed interaction. Going one step further, we define here two levels of abstraction in describing the interaction in an MR system, as part of a top-down (abstract to concrete) method for designing the interaction. Interaction situations are described using ASUR at the most abstract level. Nevertheless, for analytical reasons, we describe the two levels of interaction description in reverse order, from the concrete one to the abstract one.

(1) The most concrete description is the final stage of the external specification phase. Interaction is fully depicted by a set of ASUR entities and relations that are described by the ASUR characteristics. The interaction modalities (devices and languages) are therefore chosen: we distinguish two types of modalities in an MR system, the active and passive modalities. Active and passive modalities are defined for the MR systems we are concerned with in this paper: the object/target of the main task is physical, for example the patient for CAS systems.

For inputs, active modalities are used by the user to issue a command to the computer such as a pedal to move a laparoscope in a CAS system. Passive modalities are used to capture relevant information for enhancing the realization of the task, information that is not explicitly expressed by the user to the computer ("perceptual user interfaces" [9]). For example, in our CASPER (Computer ASsisted PERicardial puncture) system, presented in Figure 1, a system for computer assistance in pericardial punctures, a passive modality is used for tracking the position of the puncture needle.

For outputs, active modalities, conveying information from the computer to the user, imply that the user explicitly switches attention from her/his current task focus to a new focus in order to perceive the provided information. For example in our CASPER system, visual guidance information during the puncture task is displayed on a screen. While using CASPER (Figure 1), the surgeon must always shift between looking at the screen and looking at the patient and the needle (i.e., the task environment). As opposed to active modalities, passive output modalities convey information to the user that is integrated in her/his task environment, for example displaying anatomical information onto the patient's body during a surgery. For the case of passive output modalities, the user does not have to switch

attention from her/his current task focus in order to perceive the provided information.

Fig. 1: CASPER in use during the intervention.

In Figure 2, we illustrate this level of interaction description, by presenting the ASUR diagram of the CASPER system. 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 Figure 2 by an ASUR relation $\Rightarrow$ (physical activity triggering an action).

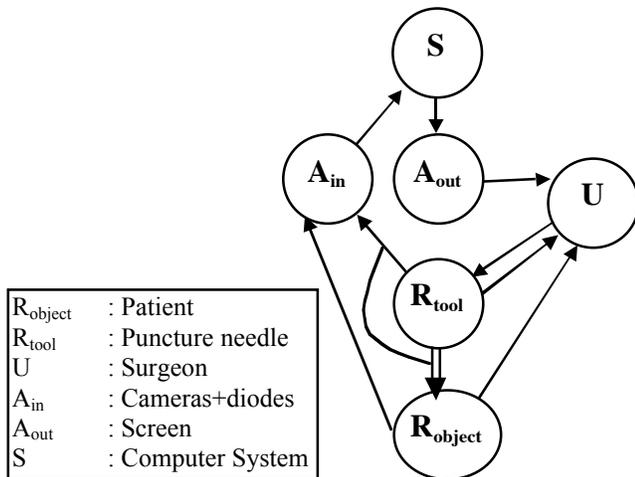| R_{object} | : Patient |
R_{tool}	: Puncture needle
U	: Surgeon
A_{in}	: Cameras+diodes
A_{out}	: Screen
S	: Computer System

Fig. 2: 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 [2]).

The concrete interaction description of Figure 2 is not complete. The ASUR diagram is completed by the characteristics of the identified entities and relations. For example the interaction language (one of the characteristics) used to convey the guidance information on screen (A_{out}) must be described: Using CASPER, in the same window on screen, the current position and orientation of the needle

is represented by two mobile crosses, while one stationary cross represents the planned trajectory. A complete description of the concrete interaction in ASUR can be found in [2].

(2) A more abstract level of description of interaction consists of focusing on the exchange of information between the involved entities during interaction. By doing so, we describe what we call the interaction situation. Interaction modalities are not yet chosen but the elementary tasks are identified. The role of the adapters are therefore defined (for example, a localization mechanism, a data presenter) but the concrete adapters (physical devices) as well as the forms of the data conveyed along the relation are not yet defined. In addition the physical setting is not yet defined, physical relationships between entities are not decided. In conclusion, such level of description consists of an ASUR diagram:

without characterization of the entities and relations.

with one kind of relation: exchange of data (A→B).

Figure 3 illustrates this level of description using our CASPER system. Figure 3 is therefore a more abstract description of the interaction described in Figure 2.

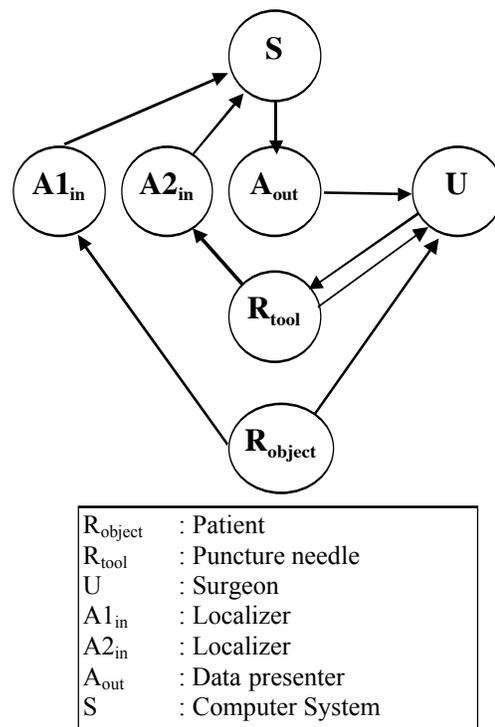| R_{object} | : Patient |
R_{tool}	: Puncture needle
U	: Surgeon
$A1_{in}$	: Localizer
$A2_{in}$	: Localizer
A_{out}	: Data presenter
S	: Computer System

Fig. 3: ASUR description of the abstract interaction in CASPER. As opposed to Figure 2, the interaction modalities as well as the physical relationships are not yet defined at this stage of the design.

In a top-down (abstract to concrete) design method, the designer first focuses on the interaction situation (i.e., abstract description of the interaction) and will then select the modalities for concretizing the interaction. Our first design space identifying a set of interaction situations is therefore useful at an early stage of the interaction design

for reasoning on the interaction without being biased by the interaction technologies. Our second design space characterizes the possible usages of one particular innovative interaction device (output adapter) for CAS systems: a mini screen. This second design space is therefore useful for designing concrete interactions involving a mini screen.

INTERACTION SITUATION DESIGN SPACE

Our design space is made of interaction situations that are independent of the interaction modalities. A situation is dedicated to a particular task. For example, in Figure 3, the diagram depicts the interaction situation for the task of pericardial puncture while using CASPER. A situation describes both the abstract input and output interaction. Our framework is composed of input and output situations. Our approach for establishing the framework of interaction situations draws from our distinction of active and passive modalities.

Input interaction situations

For inputs (user to computer), we identify four situations, two of them involve active modalities while the other two involve passive modalities.

(1) The two situations, Class I-input and Class II-input, involve active modalities. In these situations, the user explicitly issues a command to the computer system. The user must switch attention from the task's focus (R_{object}) to a new focus in order to interact with the computer. As a consequence, in the ASUR diagram that depicts these two situations, there is no R_{object} involved. Without R_{object}, the two remaining possibilities are:

Class I-input: $U \rightarrow A_{in} \rightarrow S$

Class II-input: $U \rightarrow R_{tool} \rightarrow A_{in} \rightarrow S$

The first situation (Class I-input) depicts a classical interaction with a computer, for example using a mouse. The second situation (Class II-input) describes the case where the user manipulates a physical object (R_{tool}) to interact with the computer via an adapter that captures the manipulations. Examples of such input situations are the physical icons that are physical handles to digital objects, "*coupling the bits with everyday physical objects and architectural surfaces*" [6].

(2) We identify two situations that involve passive modalities. The user is performing a task in the physical world on an R_{object} while the computer captures relevant information for enhancing the realization of the task, thanks to passive modalities. Two situations are possible whether the user manipulates R_{object} using a tool ([R_{tool}, R_{object}]) or directly manipulates R_{object}.

Class III-input: $U \rightarrow [R_{tool}, R_{object}] \rightarrow A_{in} \rightarrow S$

Class IV-input: $U \rightarrow R_{object} \rightarrow A_{in} \rightarrow S$

A Class III-input example is the CASPER input situation described in Figure 3: During the puncture task, the surgeon is handling the puncture needle (R_{tool}) that touches the patient body ([R_{tool}, R_{object}]). Both the needle and the patient are localized by the system via adapters.

For these two situations that involve passive modalities, 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 using ASUR by adding an ASUR chain that comprises the computer system (S) between:

the user (U) and the tool ([R_{tool}, R_{object}]) for Class III-input,

the user (U) and the object of the task (R_{object}) for Class IV-input.

The ASUR chain to be added is either:

(a) ($A_{in} \rightarrow S \rightarrow A_{out}$)

(b) ($R_{tool} \rightarrow A_{in} \rightarrow S \rightarrow A_{out}$)

The two ASUR chains differ by the way the user interacts with the computer system (S). The two chains (a) and (b) respectively correspond to Class I-input and Class II-input.

We therefore obtain four classes:

Class III-input-a (U and R_{object} distant):

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

Class III-input-b (U and R_{object} distant):

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

Class IV-input-a (U and R_{object} distant):

$U \rightarrow (A_{in} \rightarrow S \rightarrow A_{out}) \rightarrow R_{object} \rightarrow A_{in} \rightarrow S$

Class IV-input-b (U and R_{object} distant):

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

For example, the input interaction situation of the telesurgery system described in [5] belongs to Class III-input-b: The surgeon (U) remotely controls a slave robot (A_{out}), that holds the surgical tools ($A_{out} \rightarrow [R_{tool}, R_{object}]$), by manipulating force-feedback arm-mounted tools ($U \rightarrow R_{tool} \rightarrow A_{in}$).

Output interaction situations

For outputs (computer to user), we identify four situations, two involving active modalities and two involving passive ones. This is the symmetric case of input situations.

Class I-output and Class II-output correspond to situations involving active modalities. The user must switch attention (explicit action of the user) from her/his current task focus (R_{object}) to a new focus in order to perceive the provided information carried by the active modalities. The ASUR diagrams of these two situations therefore do not comprise an entity R_{object}.

Class I-output: $S \rightarrow A_{out} \rightarrow U$

Class II-output: $S \rightarrow A_{out} \rightarrow R_{tool} \rightarrow U$

A Class I-output example is the CASPER output situation described in Figure 3: During the puncture task, the surgeon perceives guidance information displayed on a screen. An example of Class II-output situation would correspond to a CAS system that displays information on the wall of the operating theater: Although a surface of the physical environment is used for displaying information

(R_{tool}), it implies that the surgeon consciously switch attention from the environment of the task (the operating field) to the wall in order to perceive the information.

(2) As for inputs, two output situations involve passive modalities. These situations describe the cases where the user is perceiving the information provided by the system within her/his task environment (R_{object}). The ASUR diagrams that describe these two situations therefore involve an R_{object}.

Class III-output: $S{\rightarrow}A_{out}{\rightarrow}[R_{tool}, R_{object}]{\rightarrow}U$

Class IV-output: $S{\rightarrow}A_{out}{\rightarrow}R_{object}{\rightarrow}U$

The output situation using the PADyC (*Passive Arm with Dynamic Constraints*) system [8] belongs to Class-III-output. 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 output situation of this class that involves a mini screen will be described in the last section of the paper.

A Class IV-output example is the situation using the second version of CASPER [2] that involves a see-through head-mounted display (HMD), instead of a screen as in the first version of CASPER (Figure 1). Thanks to the HMD, the surgeon directly perceives the guidance information displayed on top of the patient. Another example is the Image Overlay system [1] presented in Figure 5. The guidance information is displayed onto a see-through surface located in between the surgeon and the patient's body. Such an interaction situation belongs to Class IV-output.

The same reasoning as the one for inputs can be applied for studying the case where the user and the object of the task are distant. The two chains to be added to Class III-output and Class IV-output, in between R_{object} and U are:

$$(a) (A_{in}{\rightarrow}S{\rightarrow}A_{out})$$
$$(b) (A_{in}{\rightarrow}S{\rightarrow}A_{out}{\rightarrow}R_{tool})$$

For example:

Class IV-output-a: $S{\rightarrow}A_{out}{\rightarrow}R_{object}{\rightarrow}(A_{in}{\rightarrow}S{\rightarrow}A_{out}){\rightarrow}U$

One example of such a situation will be the following one: a telesurgery system displays anatomical information on top of the patient's body ($S{\rightarrow}A_{out}{\rightarrow}R_{object}$), while a camera ($A_{in}$) facing the patient's body enables the distant surgeon (U) to see on her/his screen (A_{out}) the image of the patient enhanced by the anatomical information.

Completeness of the situation design space
For each input as well as output situation, we described all the combination possibilities of ASUR entities, making the design space complete. Nevertheless for each situation the described ASUR chain is the minimal one. While concretizing the abstract situation, some ASUR entities may be inserted in the minimal chain.

The completeness of the framework makes it a useful tool for the designer to systematically explore the set of possibilities at an early stage of the interaction design, without being biased by a particular technology. With the interaction situation described, the designer can then focus on the modalities (device and language) that are passive or active according to the situation, as well as on the physical setting (physical relations described in ASUR). From an abstract interaction situation, several concrete interaction solutions can be designed. In the following paragraph, we focus on concrete interaction involving a particular device: a mini screen.

CONCRETE INTERACTION INVOLVING A MINI SCREEN
The transition from interaction situation to concrete interaction is difficult because the set of possibilities in terms of modalities (device and language) is huge. As a first step for accompanying this transition, we propose a design space that describes the possible modalities that involve a mini screen.

Small devices are increasingly being used in MR systems as in [10], and offer new interaction techniques, like the Embodied User Interfaces defined in [4]. For CAS systems, a small screen is an innovative device.

Beyond standard technical features of an LCD screen like size, weight, resolution, frame rate, number of colors, luminance, viewing angle, and thickness, we propose a design space based on more interaction-centered characteristics, that are inspired from our situation design space. As shown in Figure 4, our framework is comprised of four dimensions, namely Input, Output, Manipulation and DOF.

Input
The Input dimension is used to characterize how the screen is used by the user to convey information to the computer system. Five values are identified along this dimension: none, tactile, pressure, acceleration, localization. The value none means that the screen is not used as part of an input modality. Tactile is the common input modality with a PDA (tactile screen). Moreover sensors can be embedded within the device. Thus pressure or acceleration can be detected as in [4]. Finally the localization of the screen can be known by the computer system thanks to a tracking mechanism.

Output
The Output dimension is used to describe how the device conveys information to the user. We focus here on visual data but other non visual interaction languages can be used, including haptic feedback. Along this dimension, two values are identified showing whether the displayed data are dependent on the screen's position or not. For instance, if the screen is tied to a tool handled by the surgeon, and it conveys guidance information, then the output data may be dependent on the screen's position: the displayed data change according to the screen's positions over the patient's body. Other kind of data (e.g., blood pressure, body temperature) may be independent on the screen's position in that same case.
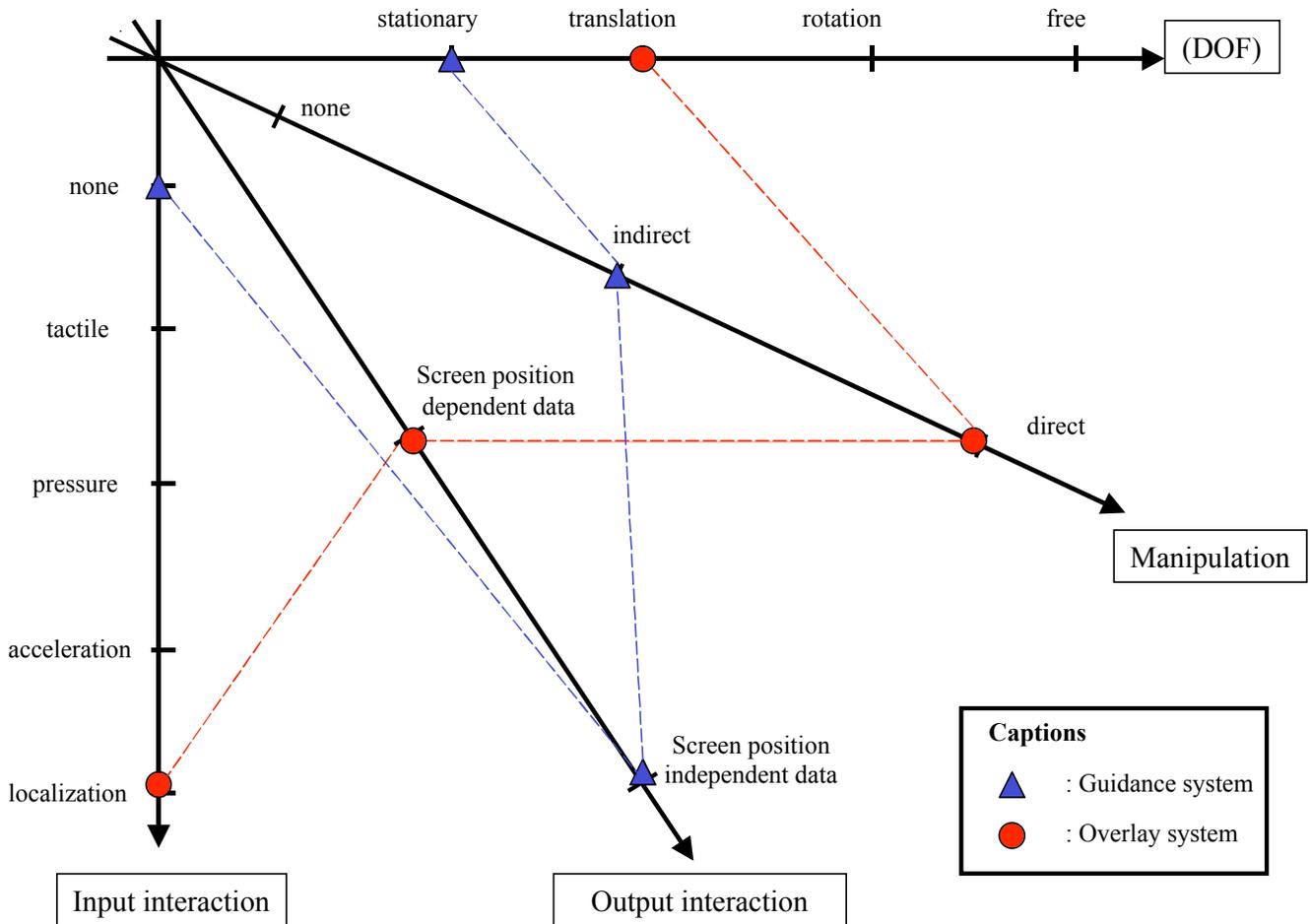
Fig. 4: Mini screen design space.

Manipulation

The Manipulation dimension expresses the context of use of the screen. Two values, direct and indirect, are identified along this dimension. The manipulation is direct if the user holds the device. The manipulation is indirect when the device is bound to another entity (e.g., an automatic arm), which itself is manipulated by the user.

Degree of Freedom (DOF)

The DOF dimension is used to describe the number of different ways in which the screen can move. The screen can be stationary, move only in translation or in rotation, or accept free motions. Those values are always based on a referential. For instance, if a screen is tied to a surgical tool (e.g., a drill), the screen is stationary in the tool's referential, but freely mobile in a more global referential: Its position and orientation are therefore tool-dependent. That referential is often defined thanks to the context of use (Manipulation).

Two CAS systems involving a mini screen

We present two usages of a mini screen that we designed and are currently developing. They correspond to different interaction situations as well as characterizations within our mini screen design space.

Guidance system

An immediate usage of the mini screen consists of using it as an output adapter to display guidance information. We therefore obtain the same situation as in CASPER presented in Figure 3 (Class III-input and Class I-output).

While concretizing the interaction, we decided that the mini screen will be tied directly to the tool (e.g., a drill) or a tool guide if the tool itself is too fragile (e.g., a needle). The ASUR description of the concrete interaction therefore includes: ($A_{out}=R_{tool}$). Within the mini screen design space, this design decision is described by (none, screen position independent data, indirect, stationary) as shown in Figure 4. Taking this design decision was driven by the need to reduce the perceptual discontinuity as defined in [2] and experimentally observed in CASPER. Linking the screen and the tool may indeed reduce the perceptual discontinuity.

As a CAS system to integrate our prototype, we have chosen puncture applications, either pericardial or renal. Guidance information in these systems is limited and easy to represent (tool direction, tool orientation, and tool depth).

Overlaying data system

Another possible usage of a mini screen consists of not reducing it to an output adapter only, as in the previous system, but allowing it to be manipulated as a tool by the surgeon. The mini screen can then be used as a magnifying glass or "magical lens" on top of the patient's body. Our design is inspired from the Image Overlay system [1] presented in Figure 5. As opposed to the interaction situation of the Image Overlay system where the surface is an output adapter (A_{out}), the mini screen in our system is both an A_{out} and a R_{tool}. Indeed the surgeon is no longer manipulating surgical tools but the mini screen. The interaction situation therefore belongs to Class III-output as opposed to Class IV-output for the Image Overlay system. For inputs, the interaction situation corresponds to the same one as in CASPER. The mini screen as a tool (R_{tool}) is localized by an input adapter.

While concretizing the interaction, the same localizer (A_{in}) can be used for both the patient and the mini screen (as in CASPER). We fixed the value "translation on top of the patient's body" along the dimension DOF, as shown in Figure 4. The ASUR description of the concrete interaction therefore includes: ($R_{tool} \Rightarrow R_{object}$).

Fig. 5 : Image Overlay system.

CONCLUSION

In this paper we presented two design spaces for MR systems.

The interaction situation design space is useful for the designer in order to systematically explore the set of possibilities at an early stage of the interaction design, without being biased by a particular technology.

The mini screen design space helps the transition between the abstract and concrete interaction by characterizing possible usages of one particular innovative interaction device for CAS systems: a mini screen.

As on-going work, we are studying the interaction situations of other types of MR systems and not only the ones that assist a user in performing a task on a physical object as in CAS systems.

During the workshop we would like to discuss the completeness of the mini screen design space and apply our interaction situation design space for describing the interaction situations of the presented systems.

REFERENCES
1. Blackwell, M., Nikou, C., DiGiola, A.M., and Kanade, T. An Image Overlay System for Medical Data Visualization. Proceedings of MICCAI'98, (1998), LNCS 1496, Springer-Verlag, 232-240.

2. Dubois, E., Nigay, L., and Troccaz, J. Consistency in Augmented Reality Systems. Proceedings of EHCI'01, (2001), IFIP WG2.7 (13.2) Conference, LNCS 2254, Springer-Verlag, 117-130.

3. Dubois, E., Gray, P., Nigay, L. ASUR++: a Design Notation for Mobile Mixed Systems. Interacting With Computers (2003), IWC, 15(4), Elsevier Science, 497-520.

4. Fishkin, K.P., Moran, T.P. and Harrison, B.L. Embodied User Interfaces : Towards Invisible User Interfaces. Proceedings of EHCI'98, (1998), IFIP WG2.7 (13.2) Conference , Kluwer academic, 1-18.

5. Green, P. S., Jensen, J. F., Hill, J. W., Shah, A. Mobile Telepresence Surgery. Proceedings of MRCAS'95 (1995), International Symposium on Medical Robotics and Computer Assisted Surgery, Wiley, New-York, 98-103.

6. Ishii, H., Ullmer, B. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms. Proceedings of CHI'97 (1997), ACM Press, 234-241.

7. Nigay, L., Coutaz, J. A Generic Platform for Addressing the Multimodal Challenge. Proceedings of CHI'95 (1995), ACM Press, 98-105.

8. 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.

9. Turk, M., Robertson, G., Eds. Perceptual user Interfaces. Communications of the ACM (2000), 43(3), ACM Press, 32-70.

10. Wagner, D., and Schmalstieg, D. First steps Towards Handheld Augmented Reality. Proceedings of ISWC03 (2003), IEEE International Symposium on Wearable Computers, IEEE Computer Society.