Virtual and Augmented Reality Tools for Teleoperation: Improving Distant Immersion and Perception

Nicolas Mollet, Ryad Chellali, and Luca Brayda

TEleRobotics and Applications dept.
Italian Institute of Technology
Via Morego, 30 16163 Genoa, Italy
{luca.brayda,nicolas.mollet,ryad.chellali}@iit.it

Abstract. This paper reports on the development of a collaborative system enabling to tele-operate groups of robots. The general aim is to allow a group of tele-operators to share the control of robots. This system enables the joint team of operators and robots to achieve complex tasks such as inspecting an area or exploring unknown parts of an unknown environment.

Thanks to virtual and augmented reality techniques, a Virtual and Augmented Collaborative Environment (VACE) is built. This last supports a $N * M * K$ scheme: $N_{1;N}$ tele-operators control $M_{1;M}$ robots at $K_{1;K}$ abstraction levels. Indeed, our VACE allows to N people to control any robot at different abstraction's levels (from individual actuator's control $K = 1$ to final status specification $K = 3$). On the other hand, the VACE enables to build synthetic representations of the robots and their world. Robots may appear to tele-operators as individuals or reduced to a single virtual entity.

We present in this paper an overview of this system and an application, namely a museum visit. We show how visitors can control robots and improve their immersion using a head-tracking system combined with a VR helmet to control the active vision systems on the remote mobile robots. We also introduce the ability to control the remote robots configuration, at a *group* level. We finally show how Augmented and Virtual Reality add-ons are included to ease the execution of remote tasks.

1 Introduction

Tele-operation is dealing with controlling robots to remotely intervene in unknown and/or hazardous environments. This topic is addressed since the 40's as a peer to peer (P2P) system: a single human or tele-operator controls distantly a single robot. From information exchanges point of view, classical tele-operation systems are one to one-based information streams: the human sends commands to a single robot while this last sends sensory feedbacks to a single user. The forward stream is constructed by capturing human commands and translated into robot's controls. The backward stream is derived from the robot's status

Z. Pan et al. (Eds.): Transactions on Edutainment II, LNCS 5660, pp. 135–159, 2009.

and its sensing data to be displayed to the tele-operator. This scheme, e.g. one to one tele-operation, has evolved this last decade thanks to the advances and achievements in robotics, sensing and virtual-augmented realities technologies: these last ones allow to create interfaces that manipulate information streams to synthesize artificial representations or stimulus to be displayed to users or to derive adapted controls to be sent to the robots. Following these new abilities, more complex systems having more combinations and configurations became possible. Mainly, systems supporting N tele-operators for M robots has been built to intervene after disasters or within hazardous environments. Needless to say that the consequent complexity in both interface design and interactions handling between the two groups and/or intra-groups has dramatically increased. Thus and as a fundamental consequence the "one to one" or "old fashion" tele-operation scheme must be reconsidered from both control and sensory feedback point of views: instead of having a unique bidirectional stream, we have to manage $N * M$ bidirectional streams. One user may be able to control a set of robots, or, a group of users may share the control of a single robot or more generally, N users co-operate and share the control of M co-operating robots. To support the previous configurations, the N to M system must have strong capabilities enabling co-ordination and co-operation within three subsets:

- Humans
- Robots
- Human(s) and Robot(s)

The previous subdivision follows a homogeneity-based criteria: one use or develop the same tools to handle the aimed relationships and to carry out modern tele-operation. For instance, humans use verbal, gesture and written language to co-operate and to develop strategies and planning. This problem was largely addressed through collaborative environments (CE).

Likely, robots use computational and numerical-based exchanges to co-operate and to co-ordinate their activities to achieve physical interactions within the remote world. Known as swarm robotics, robots' groups behavior is a very active field and we will not consider it within this paper.

For human(s)-robot(s) relationships, the problem is different: humans and robots belong to two separate sensory-motor spaces: humans issue commands in their motor space that robots must interpret and execute the corresponding motor actions through actuators. Conversely, robots inform humans about their status, namely they produce sensing data sets to be displayed to users' sensory channels. Human-Machine Interfaces (HMI) could be seen here as spaces converters: from robot space to human space and vice versa. The key issue thus is to guarantee the bijection between the two spaces. This problem is expressed as a direct mapping for the one-to-one $(1 * 1)$ systems. For the $N * M$ systems, the direct mapping is inherently impossible. Indeed, when considering a $1 * M$ system for instance, any aim of the single user must be dispatched to the M robots. Likely, one needs to construct an understandable representation of M robots to be displayed to the single user. We can also think about the $N * 1$

systems: how to combine the aims of the N users to derive actions the single robot must perform?

This paper reports on developments we are conducting in our Lab to study bijective Human-Robot interfaces design. First, we present the platform and its capabilities to integrate and abstract any robots into Virtual and Augmented worlds. We give the general framework of the platform and some associated tools, such as scenario languages, to manage Robots' organization, Augmented Reality (tracked head-mounted display) system to improve teleoperators' immersion through the control of an active vision system on a remote mobile robot. We finally present an example of the actual deployment of the platform: remote artwork perception within a museum.

2 State of the Art

Robots are entities being used more and more to both extend the human senses and to perform particular tasks involving repetition, manipulation, precision. Particularly in the first case, the wide range of sensors available today allows a robot to collect several kind of environmental data (images and sound at almost any spectral band, temperature, pressure...). Depending on the application, such data can be internally processed for achieving complete autonomy [1,2] or, in case a human intervention is required, the observed data can be analyzed off-line (robots for medical imaging, [3]) or in real time (robots for surgical manipulations such as the Da Vinci Surgical System by Intuitive Surgical Inc., or [4]). An interesting characteristic of robots with real-time access is to be remotely managed by operators (Teleoperation), thus leading to the concept of *Telerobotics* [5,6] anytime it is impossible or undesirable for the user to be where the robot is: this is the case when unaccessible or dangerous sites are to be explored, to avoid life threatening situations for humans (subterranean, submarine or space sites, buildings with excessive temperature or concentration of gas).

Research in Robotics, particularly in Teleoperation, is now considering cognitive approaches for the design of an *intelligent* interface between men and machines. This is because interacting with a robot or a (inherently complex) multi-robot system in a potentially unknown environment is a very high skill demanding and high concentration task. Moreover, the increasing ability of robots to be equipped with many small - though useful - sensors, is demanding an effort to avoid any data flood towards a teleoperators, which would dramatically drawn the pertinent information. Clearly, sharing the tasks in a collaborative and cooperative way between *all* the $N * M$ participants (humans, machines) is preferable to a classical $1 * 1$ model.

Any teleoperation task is as much effective as an acceptable degree of immersion is achieved: if not, operators have distorted perception of distant world, potentially compromising the task with artifacts, such as the well know tunneling effect [7]. Research has focused in making Teleoperation evolve into Telepresence [8,9], where the user feels the distant environment as it would be local, up to Telexistence [10], where the user is no more aware of the local environment and he is entirely projected in the distant location. For this projection to be feasible, immersion is the key feature. VR is used in a variety of disciplines and applications: its main advantage consists in providing immersive solutions to a given Human-Machine Interface (HMI): the use of 3D vision can be coupled with multi-dimensional audio and tactile or haptic feedback, thus fully exploiting the available external human senses.

A long history of common developments, where VR offers new tools for teleoperation, can be found in [11][9][12][8]. These works address techniques for better simulations, immersions, controls, simplifications, additional information, force feedbacks, abstractions and metaphors, etc. The use of VR has been strongly facilitated during the last ten years: techniques are mature, costs have been strongly reduced and computers and devices are powerful enough for real-time interactions with realistic environments. Collaborative teleoperation is also possible [13], because through VR more users can interact in Real-Time with the remote robots and between them. The relatively easy access to such interaction tool (generally no specific hardware/software knowledge are required), the possibility of integrating physics laws in the virtual model of objects and the interesting properties of abstracting reality make VR the optimal form of exploring imaginary or distant worlds. A proof is represented by the design of highly interactive computer games, involving more and more a VR-like interface and by VR-based simulation tools used for training in various professional fields (production, medical, military [14]).

Furthermore, in this multi-robots teleoperation context, complex tasks have to be *specified*, involving several actors with time and resource constraints, with synchronizations problems and potential dynamic modifications according to teleoperators' actions. In the literature, first solutions used in Virtual Environments to describe tasks were based on low-level languages (state-machine, parallel, hierarchical, etc.). As those languages could describe complex behaviors, they could be used to describe complex tasks involving several behaviors [15]. Quickly, the need of abstraction became a priority, in order to simplify scenario authoring and their separation with low-level languages for future usages. Many studies have been conducted to create new dedicated *scenario languages*, allowing elegant descriptions of complex situations, involving complex actors. Applications have been made in many fields: tutoring systems [16], fire-fighting simulation and training for collaborative actions [17], interactive museum with virtual actors [18], huge military training environments on maintenance operations [19]. To the best of our knowledge, such an advanced approach about Scenario Languages has not been directly applied in the field of Robotics.

3 Contributions

We firstly describe an overview of our framework. Then we present the abstraction layers proposed to teleoperators. Finally, we give an overview of the interactions process within our system.

3.1 Overview of the System

In our framework we first use a VACE for abstracting and standardizing real robots. The VACE is a way to integrate in a standardized way of interaction heterogenous robots from different manufacturers in the same environment, with the same level of abstraction. We intend in fact to integrate robots being shipped with the related drivers and robots internally assembled together with their special-purpose operating system. By providing a unique way of interaction, any robot can be manipulated through standard interfaces and commands, and any communication can be done easily: heterogenous robots are thus *standardized* by the use of a VACE. An example of such an environment is depicted in Figure 1: a team of teleoperators $N_{1;N}$ is able to simultaneously act on a set of robots $M_{1;M}$ through the VACE. This implies that this environment provides a suitable interface for teleoperators, who are able to access a certain number of robots altogether, or also just one robot sensor in function of the task.

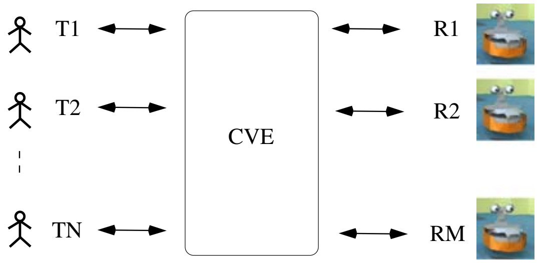

Fig. 1. Basic principle of a Virtual-Augmented Collaborative Environment: N teleoperators can interact with M robots

3.2 Abstraction Layers for Interactions

In the VACE of our framework, several Teleoperators can interact simultaneously with $K = 3$ layers of abstraction, from the lowest to the highest (Figure 2) :

1. the Control Layer
2. the Augmented Virtuality (AV) Layer
3. the Group Manager Interface (GMI) Layer

In this scheme layers concerning ways of teleoperating closer to the human way of acting are closer to the depicted teleoperator. We detail in the following the role of those layers. Note that a more complete description of the VACE and the global system can be found in [20].

The *Control layer* is the lowest level of abstraction, where a teleoperator can take full and direct control of a robot. The purpose is to provide a precise control of sensors and actuators, including wheel motors, vision and audio system, distance estimators etc... Directly accessing such layer is useful when delicate, new or atypic operations have to be performed with a single component of a

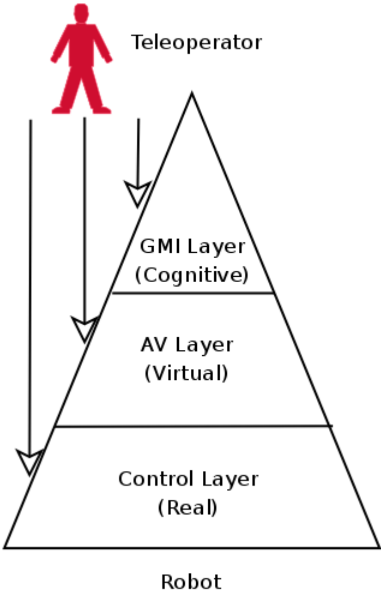

Fig. 2. In our VACE three abstraction layers (GMI, AV, Control) are available for teleoperation

single robot. Operations at this layer are equivalent to using software primitives on top of the robots' hardware, which can be of primary importance when the robot is potentially blocked in certain physical states and no recovering routine is present. Note that directly acting on the Control Layer implies a certain amount of training for the operator: minimizing human adaptation through the layers above is within the targets of a VACE.

The remaining operations, generally classified as simple, repetitive or already learnt by the robots, are executed by the Control Layer without human assistance; whether it is the case to perform them or not it is delegated above to the *Augmented Virtuality Layer*. Such layer offers a medium level of abstraction: teleoperators take advantage of the standardized abstracted level, can manipulate several robots with the same interface, which provides commands close to *what* an operator wants to do instead of *how*. This is achieved by presenting a Human-Machine Interface (HMI) with a purely virtual scene of the environment, where virtual robots move and act. Using that, teleoperators can concentrate on a minimum-information world, where just the essential data (for a given task) is dynamically represented. Peaces of reality (i.e. data from sensors) can be embedded in such world by the operators, thus making the Virtual space Augmented. A typical AV-Layer command is asking to a robot to reach a target, visible in the virtual world through the HMI.

Finally, the highest level of abstraction is offered by the *Groups Manager Interface (GMI)*. Its role is to organize groups of robots according to a set of tasks, given a set of resources. Teleoperators communicate with the GMI, which in turns combines all the requests to adjust priorities and actions on robots. No action is required to teleoperator concerning selection of robots in function of their capabilities or their current availability state: the GMI handles everything. In the example mentioned, the GMI (transparently called via the HMI) would choose the best set of robots able to reach the target, drive the group of robots and report success or failure.

3.3 Overview of Interactions

The figure 3 presents more details about the global interactions system, and the way Teleoperators can interact within groups of robots.

- At the top of the figure, the real world is composed of mobile robots (with wheels, legs, etc.) which have to evolve in their environment, for example a task of surveillance or exploration. By definition, the real world is complex, and can be considered as a *noisy* environment as it can provide many information which are not useful for the task the robots have to realize. For the illustration, we have defined 2 groups of robots: R1, R2, R3, R4 and R6 for the first group, R5 and R7 for the second.
- In the middle the model of the distant environment is depicted. This virtual world includes 3D aspects, and all the necessary informations according to the task the groups of robots have to do. It can be the current state of a 2D or 3D map the robots are building of their environments, or a full 3D map of a known area. It can also be for example particular objects, like doors, or of course the other robots in their current state (position in the space, physical state, etc.). Those virtual robots can contain the definition of the robots behaviors, which can be their autonomy part. Note that this behavior can also be embedded, so the virtual robot is more used as a representation and an interface of control. There's a direct bijection between the real world and the virtual world, as much precise as possible: robots' locations, states, etc.
- At the bottom, 4 human teleoperators are represented. They have to teleoperate the two groups of robots to realize their task. In order to simplify the teleoperation process, we designed the GMI, the role of which is to allow a *cooperation* with the groups of robots. Teleoperators (in the illustrative figure: T1 and T4) can give some basic orders to a *group*, and the GMI will take in charge the virtual robots to realize it. So the real robots. However, the teleoperator can also take a direct control on one precise virtual robot, in the picture the teleoperator T3 directly manipulates the virtual robot VR2. Thus, the precise movements made on the virtual robot are reproduced by the real robot R2. Finally, a teleoperator can have an access on the real robot through its virtual avatar: this is the case of teleoperator T2 which, connected to the virtual robot VR6, has taken a direct control on the robot 6. As presented in section 4.3, the teleoperator can see what the robot see, and act directly on the real robot's head and gaze.

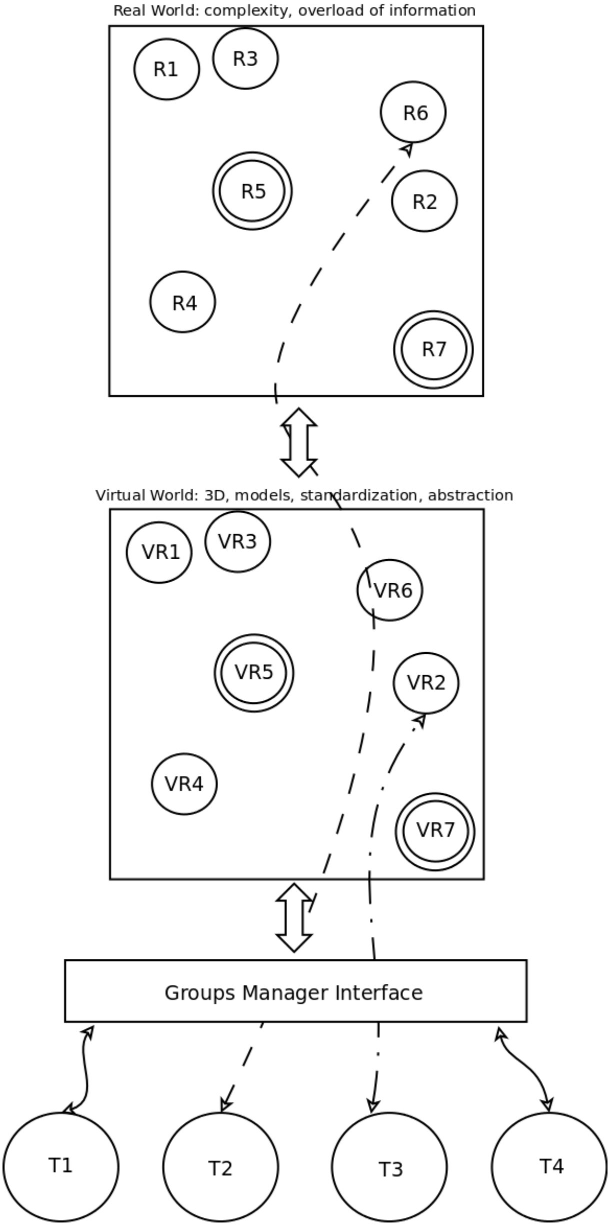
Real World: complexity, overload of information
R1
R3
R6
R5
R2
R4
R7
Virtual World: 3D, models, standardization, abstraction
VR1
VR3
VR6
VR5
VR2
VR4
VR7
Groups Manager Interface
T1
T2
T3
T4

Fig. 3. Overview of the system. Teleoperators *Ti* can interact with the virtual robots *VRi* through the Groups Manager Interface (GMI), or can take directly the control on one virtual entity (*ex: T3 and VR2*). Real and virtual robots' states are linked together. It's also possible to access to the distant real robot via the virtual robot (*ex: T2, VR6 and R6*), in particular for the vision in our case.

4 Developments: VR and AR Tools for Abstraction, Standardization and Immersion

This section deals with some of the developments we made in the context of VACE, abstraction, standardization and immersion. We first present the platform we are developing that supports our researches, then the first characteristics and needs of the GMI system through a Scenario Language, and finally our head-tracking system which improves distant immersion for teleoperation.

4.1 The ViRAT Platform as a Collaborative Virtual Environment

We are developing a multi-purposes platform, namely ViRAT (Virtual Reality for Advanced Teleoperation [21][22]), the role of which is to allow several users to control in real time and in a collaborative and efficient way groups of heterogeneous robots from any manufacturer. We presented in the paper [22] different tools and platform, and the choices we made to build this one. The ViRAT platform offers teleoperation tools in several contexts: VR, AR, Cognition, groups management. Virtual Reality, through its Virtual and Augmented Collaborative Environment, is used to abstract robots in a general way, from individual and simple robots to groups of complex and heterogenous ones. Internal ViRAT's VR robots represents exactly the states and positions of the real robots, but VR offers in fact a total control on the interfaces and the representations depending on users, tasks and robots, thus innovative interfaces and metaphors have been developed. Basic group management is provided at the GMI Layer, through a first implementation of a Scenario Language engine. The interaction with robots tends to be natural, while a form of inter-robot collaboration, and behavioral modeling, is implemented. The platform is continuously evolving to include more teleoperation modes and robots.

To explain how ViRAT is built and how it works, we are now presenting the running basic demonstration.

The basic demonstration in detail. We use for this demonstration two real and distant environments. One operator is acting through a PC, equipped with classical physical interfaces (mouse, keyboard, monitor) and more immersive devices (head mounted display, joystick). In ViRAT we find the three Layers of Figure 2. The operator manages the three robots through a Human Machine Interface (HMI): such interface offers a unified virtual world and a menu-based decision console. Figure 4 depicts the data flow and the mutual inference between GMI, scenarios, tasks and resources: orders passed to the HMI are processed by the GMI, at the moment resident on the PC. The GMI is responsible of scheduling and executing scenarios (in function of the current Language) and tasks in function of the available resources; GMI also generates new scenarios, task, and possibly new resources. The AV Layer is represented by the HMI. Note that, though the GMI is conceptually closer to the cognitive level of the operator, it is hidden behind the AV Layer. This complies with the main characteristic of the GMI: it does what the teleoperator would do, but it does not require

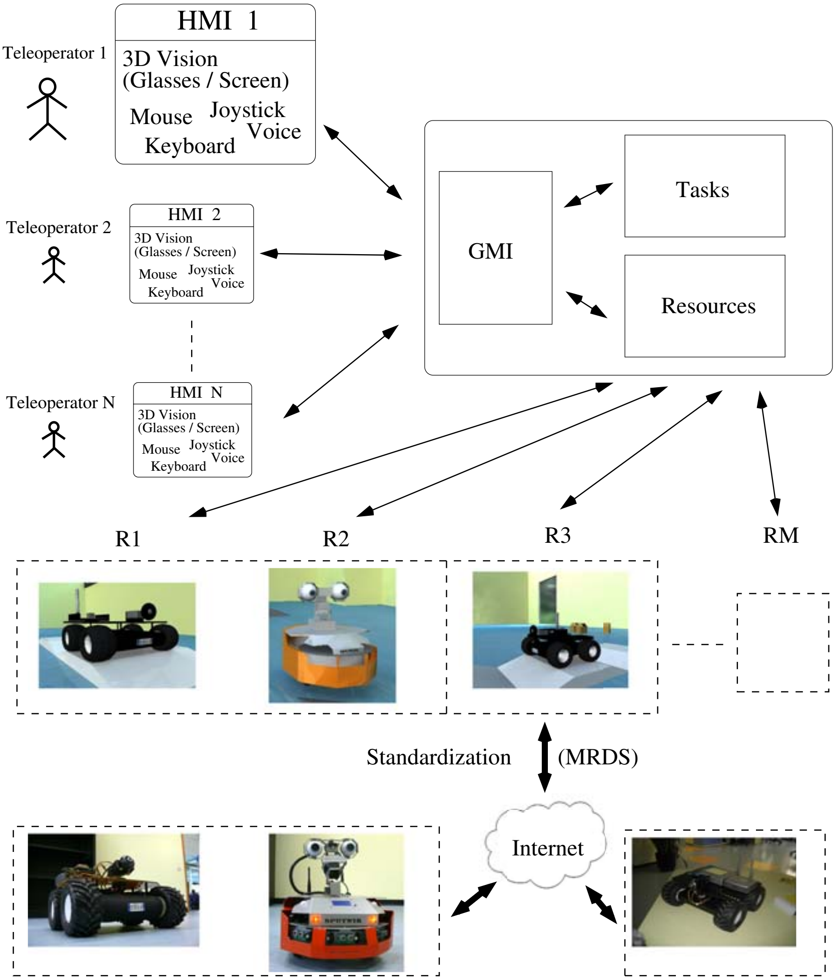

Fig. 4. Interactions scheme of the ViRAT platform

him to think about it. The output from the GMI is then dispatched through the internet to the distant rooms. Robots, each of them hosting an embedded system equipped with Wifi cards, are connected to the internet and receive the orders at the Control Layer. Finally, movements and actions of the robots can be finally followed in the virtual world by the operator. As a result, real, distant robots are connected together with their virtual avatars. More specifically, we recall Figure 1, in ViRAT $N = 1$ and $M = 3$; *Room1* contains two Robots (*R1* and *R2*), while *Room2* contains one robot (*R3*). Figure 5 depicts the virtual world and the decision console in such configuration. In our current prototype, we use Microsoft Robotics Developer Studio (MRDS) as the tool to provide virtual immersion to teleoperators and to encapsulate as services our algorithms which

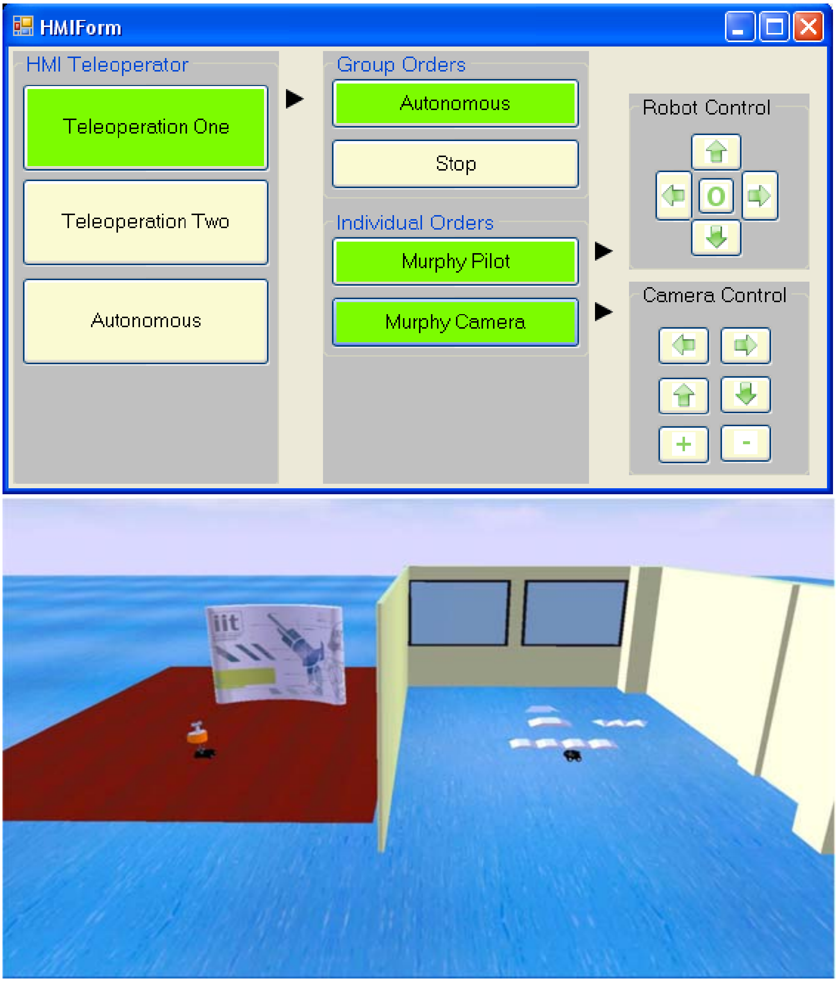

Fig. 5. The HMI console and unified virtual world seen by the teleoperator

handle human-robot and robot-robot communication. In fact MRDS acts as a server, offers a way to implement and standardize the abstractions of the real robots, and allows us to integrate synchronization aspects between the virtual and the real world. The advantage of using such a versatile interface is that teleoperators can navigate freely in the two rooms, both when robots are moving or not. Operators can choose the most suitable world for the task: AV Layer or Control Layer. An example of the two view from these layers is available in Figure 6, (the two images in the bottom). Note that the operator can include or exclude some details (such as the plant), which may increase the cognitive load of the teleoperator, depending on the current task. Note also that a switch from the real on-board view of the camera to the virtual camera is possible *anytime*, because the virtual world is constantly re-calibrated to match the real world. The real-time tracking of the position and velocity of the real robots (mirrored by the locations of avatars) is achieved thanks to a calibrated camera system, able to locate the real position of robots and input them in the AV Layer. Finally, note that virtual avatars appear in the same virtual room, while real robots are separated in their real rooms. Thus, a *distributed, complex* space is represented via a *unique, simple* virtual room. We emphasize that one of the big advantaged of using the AV Layer is that teleoperators can freely navigate in the virtual system, while the real system evolves without their interference. Concerning teleoperation modes, the teleoperator interacts by default with the GMI through the HMI. In ViRAT there are substantially two macro-modes : an autonomous mode, where the GMI manages alone the two distant worlds, without the need of input from the teleoperator; a teleoperation mode, where human input is required. Such initial interaction is achieved with a speech recognition system, recognizing high-level commands. These high priority orders, depicted in Figure 5, offer three possible controls: teleoperation of the Room1, of the Room2, or actions on the global scenario. Such orders are just an illustration of the high-level abstraction access which is provided by the GMI.

Autonomous mode. The default scenario in ViRAT is the autonomous mode, which is a session composed by two parallel tasks: R1 and R2 interact in a typical master-slave paradigm, while R3 accomplishes a video-surveillance task. The tasks are independent, thus with no possible conflict.

The interaction in Room1 includes the following events:

- R1 follows a known path, which pairs with a virtual path in the virtual environment.
- R2 follows R1 with its vision tracking system (an onboard digital camera).
- R1 is able to randomly stop himself, stop R2, order R2 to reach R1. Orders are spoken by R1 through an audio system.

In Room2, R3 evolves alone, avoiding obstacles, thanks to an exact replica of the camera system. The role of R3 is to continuously verify that there's nothing abnormal in the room: a new obstacle, or an alert given on an object in particular. Because the virtually unified environment is in fact physically separated, particular metaphors show that robots can not switch between rooms.

Teleoperation : a request of interaction given to the GMI. Teleoperators have the ability, through the speech recognition system, to start/stop the global autonomous mode, or to start/stop the two sub-scenarios (these are the *Group Orders* in Figure 5). This because one may want to interact with a specific room, while in the other room everything is still autonomous.

In *Room 1*, AV Layer-specific commands are: "Go To Target" (includes an automatic obstacles avoidance), commands transmitted to Robot R2 ("Stop to follow" / "Come here"). Control Layer-specific commands come from mouse, keyboard or joystick for fully controlled robot displacements. On-board camera can also be commanded by varying its pan and tilt. Note that directly controlling

R1 will still keep R2 controlled by the GMI, depending on the last orders. Recalibration between real and virtual is always performed.

In *Room 2* the AV Control Layer commands for R3 are similar. Unless otherwise specified, the GMI keeps on handling the tasks assigned to R1 and R2 while the teleoperator manages R3. The system randomly simulates a problem in the room 2 (in ViRAT a simple light switch is turned off). For example, the teleoperator can be alerted in case a problem occurs to an electrical device in Room 2. When using the embedded real camera of the robot, virtual arrows on the HMI indicate the correct direction of the target.

Note that this demo shows a general group organization through the GMI, and a collaboration of this one with a human. This basic demo also allows several teleoperators to act simultaneously: for example it's possible for a human to control R3 displacements while another teleoperator controls R3's embedded camera through the head-tracking system presented in the next sub-section. While the GMI continue to manage R1 and R2.

Goals of ViRAT. The design and tests of ViRAT allows us to claim that this platform achieves a certain number of goals:

- *Unification and Simplification*: there is a unified, simplified CVE accessing two distant rooms, which are potentially rich of details. Distant robots are part of the same environment.
- *Standardization*: we use a unified Virtual Environment to integrate heterogenous robots coming from from different manufacturers): 3D visualization, integration of physics laws into the 3D model, multiple devices for interaction are robot-independent. This framework (achieved with, but not limited to MRDS) is potentially extensible to many other robots.
- *Reusability*: behaviors and algorithms are robot-independent as well and built as services: their implementation is re-usable on other robots.
- *Pertinence via Abstraction*: a robot can be teleoperated on three layers: it can be controlled directly (Control Layer), it can be abstracted for general commands (AV Layer), and groups of robots can be teleoperated through the GMI Layer.
- *Collaboration*: several, distant robots collaborate to achieve several tasks (exploration, video-surveillance, robot following) with one teleoperator in real time. We plan to extend ViRAT to multiple teleoperators.
- *Interactive Prototyping* can be achieved for the robots (conception, behaviors, etc.) and the simulation.
- *Advanced teleoperation interfaces*: we provided interfaces which start considering cognitive aspects (voice commands) and reach a certain degree of efficiency and time control.
- *Time and space navigation* are for the moment limited in the current version of ViRAT, but the platform is open for the next steps, and teleoperators can already navigate freely in the virtual space at runtime.
- *Scenario Languages applicability.* The first tests we made with our first and limited implementation of the Scenario Language for the GMI allow us to organize this whole demonstration which mixes real and virtual actors.

4.2 Scenario Languages Requirements for Robots' Management

Scenario Languages allow simple management of complex actors to achieve complex tasks, according to their skills, behaviors, availability and current states. Scenario Languages are well known in the field of Virtual Reality, and we are applying those technics for robots' management, in the development of the GMI (Note that a complete description of the Scenario Language and its engine is behind the scope of this paper). In fact, common features addressing both virtual environments and multi-robot systems are:

- Synchronization. Particular tasks demand synchronized actions from robots, in order to obtain a precise collaboration for complex manipulations. Low-level synchronizations, which have to be handled by the platform, are not included.
- Multi-threading. Authors of scenarios have to be able to specify parallel tasks.
- Job scheduling. Time-driven decision are necessary for multiple aspects: duration of tasks, synchronization, launch particular actions at a certain time, etc.
- Dynamism. As teleoperators have to interact in real-time with groups of robots, Scenario Languages cannot be static: they must adapt to the current state of the whole system. Being dynamic entails such languages to be editable at runtime. This avoids just *playing* a known pre-compiled scenario. A dynamic language allows teleoperators to make requests of interactions, set priorities, possibly change tasks/goals.
- Tasks/goals. Scenarios have to be organized around tasks to achieve. There are several kind of Scenario Languages: a linear description of a scenario is not enough. An initial state is represented by the notion of task itself, which provides a basis to obtain an automatic organization according to this dynamic context.
- Resources. The Scenario Language has to provide a clear notion of *resources.* Those define entities with set of skills and states. The combinations of resources and tasks allow scenario engines to dynamically generate other scenarios to follow.
- Hierarchy. In order to simplify the organization of scenarios and tasks, the notion of hierarchy has to be proposed.
- Uncertainty. To be very adaptable and to allow dynamic arrangements according to states, problems or modifications on priorities, the Scenario Language has to integrate the ability to define tasks and associated resources through an *uncertain* approach: the full way to proceed has not to be clearly specified, except for goals.

We developed a first version of a Scenario Language and its engine, which is in the heart of the GMI. This first prototype is very limited according to the possibilities that can be offered by Scenario Languages, but it allowed us to validate them through the ViRAT project.

4.3 Head-Tracking System for VR/AR Remote Observations

In ViRAT we make use of advanced immersive interfaces: a helmet equipped with displays, earphones and microphone. Worn by a human, all of his head-movements are transmitted to the robot's webcam, so the teleoperator can feel fully immersed in this distant environment, and can see in a very natural way. Note that the camera view takes advantages of the Virtual environment: teleoperators can directly obtain Augmented Reality features, the virtual additions already accessible in the pure virtual world (Virtual arrows for instance).

Clearly, seeing what the robot see is an important opportunity, and it's fundamental in many cases: direct manipulation on a sensible object, a too complex environment where the robot can not evolve alone, the detection of a problem which require a human attention, etc. In order to offer an efficient capacity of distant vision for the teleoperators, we designed and prototyped a system which uses a VR helmet and a head-tracking system:

- As the teleoperator's vision is limited to what the VR helmet shows, it allows an interesting *immersion* sensation, which makes the operators feeling *in* the distant environment, as far as the resolutions of the helmet and the distant camera(s) is good enough.
- The tracking system is used here to track the movements of the head. Usually, such a system is used to improve the feeling of immersion in a pure 3D world,

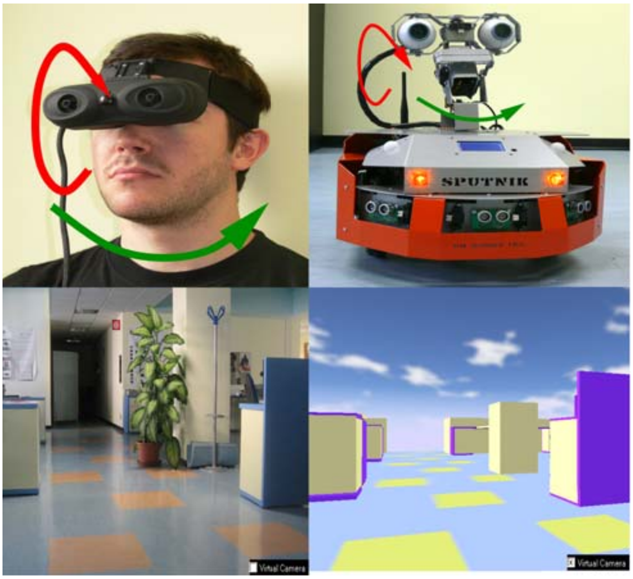

Fig. 6. Relationship between the robot vision system, carried out to the human vision capabilities. Note that the user can switch anytime from the real to the virtual view (both are real-time re-calibrated), driven by his/her quasi real-time head position and orientation.

with any system of visualization. Here, we use the same approach to be fully immersed in the distant world: the movements of the head make the distant camera(s) moving in the same way. The pan-tilt capability is emphasized in figure 6. Though ideally the system would be stereo, currently we use only one of the cameras of this robot. The tracking system has been thus limited to the rotations, so it's not a full head-tracking system. We currently evaluate the possibility of an extension of such a system to allow some small translation movements in the teleoperators area, which would be transformed and adapted to translations on the robots. For the moment, the immersion and its feeling is already enough for facilitating the observation of the distant site; and robots movement can still be control with a joystick for example.

We developed a generic system (figure 7) which allows us to easily deploy vision capacities and remote access on every robots we build or buy. The system is composed of a wired Hi-Res webcam, plugged on a *control module* which offers GPS positions, WiFi and Bluetooth communications capabilities, and inputs/outputs access for controlling small motors and sending the videos. The webcam is articulated with two motors, controlled by the module. We developed a basic remote control program which receives the videos and sends commands to the motors; this program is currently under integration in MSRS as a standard service. The control module has been developed in our team[20].

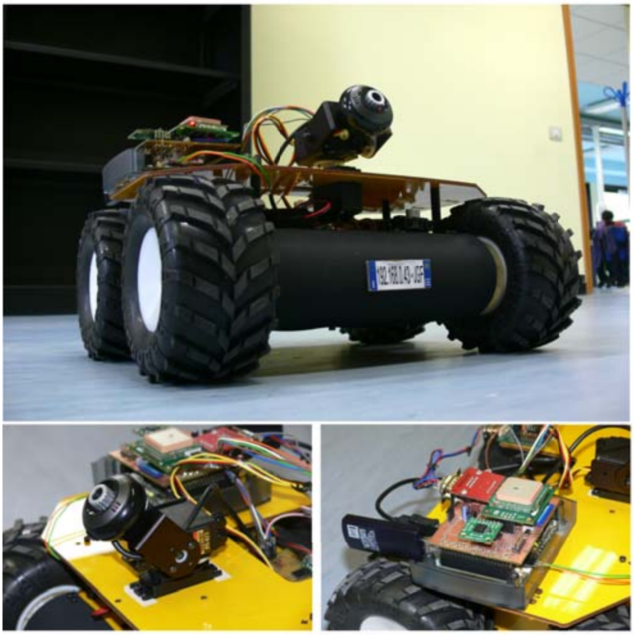

Fig. 7. A wheeled robot, with an integration of our generic vision system. The webcam moves with two motors, managed by the control module shown on the last photo.

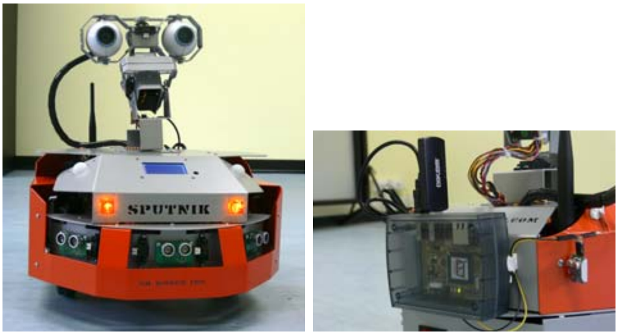

Fig. 8. The integration of the control module on the *Sputnik* robot

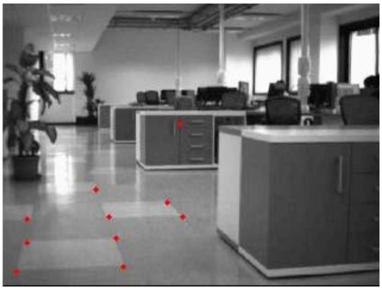

Fig. 9. Some Augmented Reality content inserted on what the teleoperators can see: here, the ground analyze and a target to reach (red points)

The figure 7 shows the system in action on one of our simple wheeled robot. We can easily see the camera with the motors, and the control module. The figure 8 presents the current integration of the module with *Sputnik*[1].

In the current version of our prototype, we use a *ARvision-3D* VR helmet (see figure 6, top left image). It shows the videos received by the teleoperator's computer from the distant robot. That's already an interesting way for a better comprehension of the distant environment: the teleoperator is really immersed. As previously introduced, we use a head-tracking system to improve this feeling and the efficiency of the teleoperations. We can use here two systems: a magnetic tracker (*Polhemus* system) or an accelerometer. The tracking, as previously said, is limited on the analysis of the rotations. Those informations are then converted into commands for the motors which are sent in Real-Time to the distant robot (control-module) and to the virtual robot.

In order to help the teleoperator in his task for this usage of the distant vision, we introduced some Augmented Reality (AR) technics. Based on the internal modeling / representation (of the objects, the entities, their behaviors and states,

[1] Check for Dr Robot manufacturer.

interactions capabilities, etc.) managed in the VACE, it's easy to add for example the current map of the area, or an indicator on known entities (in particular on other robots). Our system can overlay live vision images by some additional signs to enable to teleoperators to communicate and to achieve a collaborative inspection of an area. The process uses a top view representation of the targeted area as a pointing surface for teleoperators (T1 in the example) in the virtual world. The process calculates then the localization (position and orientation) of the stereo system regarding an absolute framework (well known on the top view map). Using a projective geometry (namely a homographic transformation), we then project on the floor any needed sign (crosses for instance to show the aimed position - see figure 9). This projection become an AR indicator in the vision of the real-world seen by the teleoperator who use the VR helmet.

5 Example of Real Case Application: Improving Immersion in Artwork Perception by Mixing Telerobotics and VR

The ViRAT platform proposes now several demonstrations that are focused on interfaces or human analysis. We are also deploying some real-case projects. One of those is a collaboration with a museum, where the major goal is to offer the ability for distant people to visit a real museum. We'll see in this section that we are interesting in improving the sensation of immersion, of *real visits* for virtual visitors, and that such a system may have different usages such as surveillance when the museum is closed.

The existing VR system for virtual visits of museum, like the excellent *Musée du Louvre*[23], are still limited, with for example the lack any natural light conditions in the Virtual Environment. Another interesting point is that the user is always alone in exploring such virtual worlds. The technologic effort to make an exploration more immersive should also take into account such human factors: should navigation compromise with details when dealing with immersion? We believe this is the case. Does the precise observation of an artwork need the same precise observation during motion? Up to a certain degree, no. We propose a platform able to convey realistic sensation of visiting a room rich of artistic content, while demanding the task of a more precise exploration to a virtual reality-based tool.

5.1 Deployment of the ViRAT Platform

We deployed our platform according to the particularities of this application and the museum needs. Those particularities deal mainly with high-definition textures to acquire for VR, and new interfaces that are integrated to the platform. In this first deployment, consisting in a prototype which is used to test and adapt interfaces, we only had to install two wheeled robots with embedded cameras that we have developed internally (a more complete description of those robots can be found in [20]), and a set of cameras accessible from outside trough

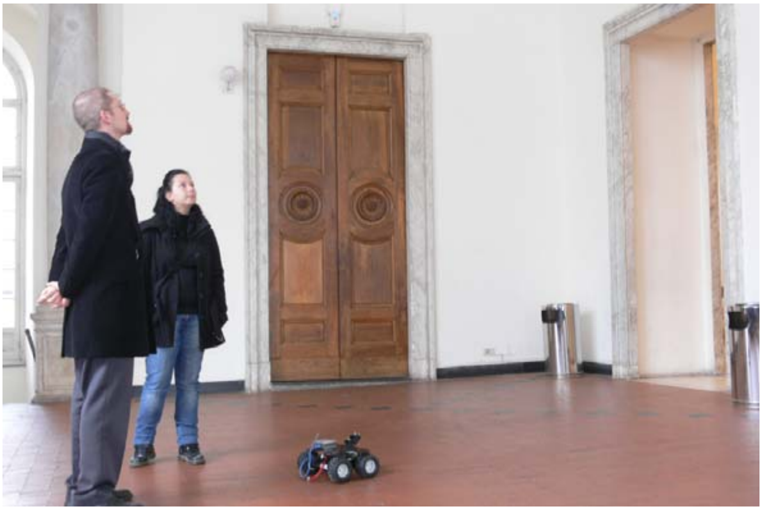

Fig. 10. A robot, controlled by distant users, is *visiting* the museum like other *traditional* visitors

internet (those cameras are used to track the robot, in order to match Virtual Robots' locations and Real Robots' locations). We modeled the 3D scene of the part of the museum where the robots are planned evolve. A computer, where the ViRAT platform is installed, is used to control the local robots and cameras. It runs the platform, so the VR environment. From our lab, on a local computer, we launch the platform which use internet to connect to the distant computer, robots and cameras. Once the system is ready, we can interact with the robots, and visit the museum, virtually or really.

5.2 Usage of Telerobotics and VR for Artwork Perception

As presented in [23], existing works with VR offer the ability to virtually visit a distant museum for example, but suffer from lacks of sensations: first, users are generally alone in the VR environment, and second, the degree and sensation of immersion is highly variable. The success of 3D games like *second life* comes from the ability to really *feel* the virtual world as a real world, where we can have numerous interactions, in particular in meeting other *real* people. Moreover, when we really visit a place, we have a certain atmosphere and ambience, which is in fact fundamental in our perception and feeling. Visiting a very calm temple with people moving delicately, or visiting a noisy and very active market would be totally different without those feedbacks. So, populating the VR environment was one of the first main needs, especially with real humans *behind* those virtual entities. Secondly, even if such VR immersion gives a good sensation of presence, so of a visit, we're not really visiting the reality. Behind *second life*'s virtual characters, we have people sit down, in front of their computer. What about having a bijection between the *reality* and the *virtuality* ? Seeing virtual entities

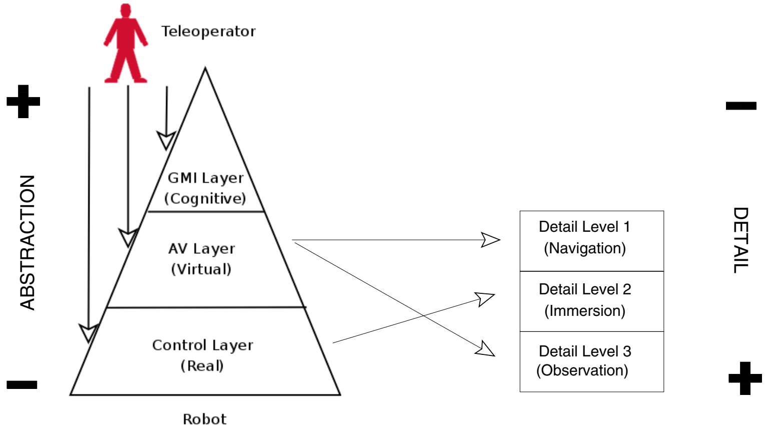

Fig. 11. Different levels of abstraction mapped into different levels of detail

in the VR environment and knowing that behind those entities the reality is hidden, directly increases the feeling of really visiting, being in a place. Especially when we can switch between virtual world and real world.

Following those comments, the proposed system mixes VR and Reality in the same application. The figure 11 represents this mix, its usage, and the adaptation we made of our general framework.

On the left part, we have the degree of immersion, while on the right part, we have the level of details. The degree of immersion is made of the three levels[22]: Group Management Interface, Augmented Virtuality and Control Layer:

- First, the GMI layer, still gives the ability to control several robots. This level could be used by distant visitors, but in the actual design it's mainly used by people from the museum to take a global view on robots when needed, and to supervise what distant visitors are doing in the real museum.
- Second, the AV layer -*Augmented Virtuality*, allows the user to freely navigate in the VR environment. It includes high-definition textures, coming from real high-definition photos of the art-paintings. This level offer different levels of interactions: precise control of the virtual robot and its camera (so as a consequence, the real robot will move in the same way), ability to define targets that the robot will reach autonomously, ability to fly though the 3D camera in the museum, etc.
- Third, the Control layer. At this levels, teleoperators can control directly the robots, in particular the camera previously presented. Users can see directly like if they were located at the robot's location. This level is the *reality* level, the users are immersed in the real distant world where they can act directly. Note that with the current development of large public VR or tracking system like the Nintendo wiimote, the visitors may have an equivalent system as the one we presented in figure 6.

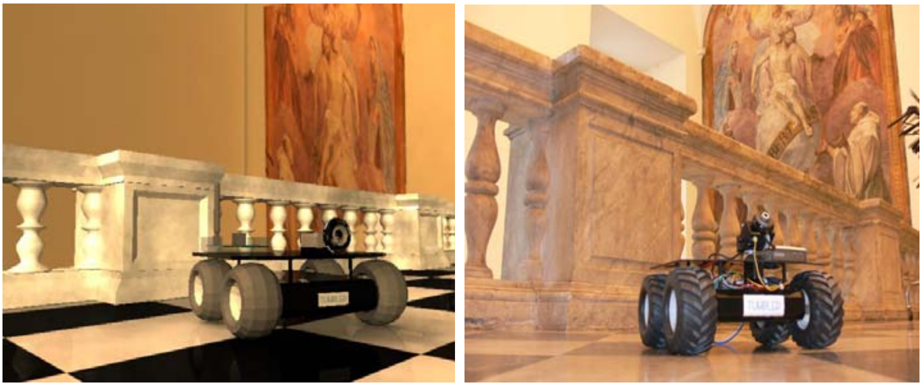

Fig. 12. Detail Level 1 is purely virtual, and is the equivalent of the reality

Fig. 13. Detail Level 3 (high detail) is purely virtual, with high-resolution pictures as textures. This one is used in the scene of the figure 12

On another hand, on the right part of the figure 11, the level of details represent the precision the users perceive of the environment:

- Detail Level 1 represents mainly an overview of the site and robots for navigation. The figure 12 shows the bijection between virtual and real, so the usage that a distant visitor can have of the virtual world as an abstraction of the real word.
- Detail Level 2 represents the reality, seen through the robots' cameras. At this level of details, users are limited by the reality, such as obstacles and cameras' limitations. But they are physically immersed in the real distant world.

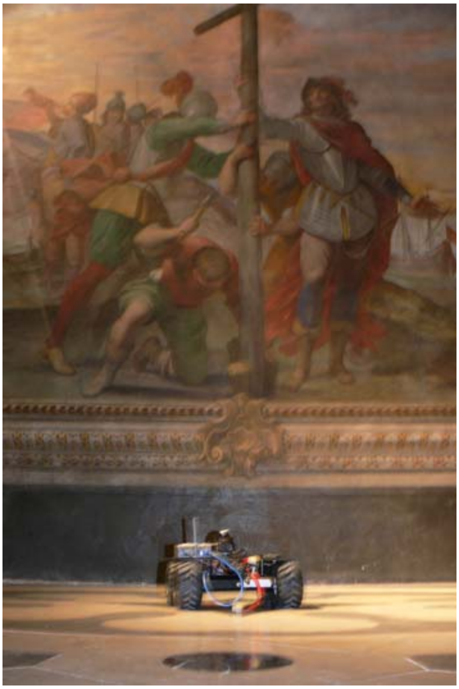

Fig. 14. Detail Level 2 is purely real. A user is observing, through his robot and its camera, a art-painting. This screenshot comes from another robot's camera observing the scene: that's what can be seen by another user, so he can see other *distant* visitors like him.

- Detail Level 3 is used when distant visitors want to see very fine details of the art-paintings for example, or any art-objects that have been digitalized in high-definition. We can see in figure 13 a high-definition texture, that a user can observe in the virtual world when he wants to focus his attention on parts of the art-painting of the figure 12, that could not be accessible with the controlled robots.

When distant visitors want to have an overview of the site, and want to move easily inside, or on the opposite when they want to make a very precise observation of one art-painting for example, they use the two Detail Levels 1 and 3, in the Virtual Environment. With this AV level, they can have the feeling of visiting a populated museum, as they can see other distant visitors represented by other virtual robots, but they do not have to fit with real problems like for example occlusions of the art-painting they want to see in details due to the crowd, or displacement problems due to the same reasons.

On another hand, when visitors want to feel themselves more present in the real museum, they can use the Detail level 2. This is the point where we mix Telerobotics with Virtual Reality in order to improve the immersion feeling. In the figure 14, we can first see a robot observing one art-painting. So, a distant visitor is currently observing through the robot's camera the *real* environment, and in particular the *real* art-painting rather than observing it in the virtual world in high-definition. Moreover, this figure comes in fact from another robot's camera: it means that another visitor is actually observing another distant visitor in front of the painting. We offer here the ability for the visitors to be *physically* present in the distant world with this Telepresence system, and to evolve like if they were really present. As a consequence, they can see the real museum and art-work, but also other visitors, local or distant, as we can see in figure 10.

5.3 Same System, Different Usages

While the most interesting aspect for visitors is to feel individually immersed through this system, it is important to note that the use of the GMI allows two separate tasks for the museum. The first one is, as previously introduced, a supervision of the distant visitors: the deployment of the robots, their states and location are directly accessible in the VACE world, and of course control can be taken on those entities. The second one is that the same system can be used as a security system when the museum is closed, or even to prevent security problems during the opening hours. Robots can work alone, easily controlled through the GMI as a global *group entity*, to observe the environment and to detect any kind of problem. And when precise actions or observations are required, exactly in the same way virtual visitors are doing, it is then possible to take a direct control on individual entities.

6 Conclusion

We presented in this paper an innovative system for an efficient teleoperation between several teleoperators and groups of robots. We introduced in this system our vision and usage different levels of interactions: GMI with a scenario language, AV and direct control. We briefly presented the VACE we developed to model the robots activities and states, an environment where teleoperators can have collaboratively an intermediate level of interaction with the real distant robots by using the virtual ones. We illustrated our project through the integration of real heterogeneous robots, and an immersed vision with head-tracking and firsts AR techniques. We finally presented one deployment of this platform for an innovative artwork perception proposed to distant visitors of a museum. Our project is currently very active and new results come frequently. Actual experiments are turned on human's perception evaluation in the case of complex interactions with groups of robots.

We would like to make some special acknowledgments to Delphine Lefebvre, Baizid Khelifa, Laura Taverna, Lorenzo Rossi and Julien Jenvrin for their contributions in the project and the article. The locations for our platform in the museum application are kindly provided by Palazzo Ducale, Genoa.

References

1. Warwick, K., Kelly, I., Goodhew, I., Keating, D.: Behaviour and learning in completely autonomous mobile robots. In: IEE Colloquium on Design and Development of Autonomous Agents, pp. 7/1–7/4 (November 1995)
2. Lidoris, G., Klasing, K., Bauer, A., Xu, T., Kuhnlenz, K., Wollherr, D., Buss, M.: The autonomous city explorer project: aims and system overview. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007. IROS 2007, October 29- November 2, pp. 560–565 (2007)
3. Glasgow, J., Thomas, G., Pudenz, E., Cabrol, N., Wettergreen, D., Coppin, P.: Optimizing information value: Improving rover sensor data collection. IEEE Transactions on Systems, Man and Cybernetics, Part A 38(3), 593–604 (2008)
4. Saffiotti, A., Broxvall, M., Gritti, M., LeBlanc, K., Lundh, R., Rashid, J., Seo, B., Cho, Y.: The peis-ecology project: Vision and results. In: IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS 2008, September 2008, pp. 2329–2335 (2008)
5. Urbancsek, T., Vajda, F.: Internet telerobotics for multi-agent mobile microrobot systems - a new approach (2003)
6. Elfes, A., Dolan, J., Podnar, G., Mau, S., Bergerman, M.: Safe and efficient robotic space exploration with tele-supervised autonomous robots. In: Proceedings of the AAAI Spring Symposium, March 2006, pp. 104–113 (2006) (to appear)
7. Wertheimer, M.: Experimentelle studien über das sehen von bewegung. Zeitschrift für Psychologie 61, 161–265 (1912)
8. Hickey, S., Manninen, T., Pulli, P.: Telereality - the next step for telepresence. In: Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics (SCI 2000), Florida, vol. 3, pp. 65–70 (2000)
9. Kheddar, A., Tzafestas, C., Blazevic, P., Coiffet, P.: Fitting teleoperation and virtual reality technologies towards teleworking (1998)
10. Tachi, S.: Real-time remote robotics-toward networked telexistence. IEEE Computer Graphics and Applications 18(6), 6–9 (1998)
11. Zhai, S., Milgram, P.: A telerobotic virtual control system. In: Proceedings of SPIE, Boston. Cooperative Intelligent Robotics in Space II, vol. 1612, pp. 311–320 (1991)
12. Yang, X., Chen, Q.: Virtual reality tools for internet-based robotic teleoperation. In: DS-RT 2004: Proceedings of the 8th IEEE International Symposium on Distributed Simulation and Real-Time Applications, Washington, DC, USA, pp. 236–239. IEEE Computer Society, Los Alamitos (2004)
13. Monferrer, A., Bonyuet, D.: Cooperative robot teleoperation through virtual reality interfaces, p. 243. IEEE Computer Society, Los Alamitos (2002)
14. Gerbaud, S., Mollet, N., Ganier, F., Arnaldi, B., Tisseau, J.: Gvt: a platform to create virtual environments for procedural training. In: IEEE VR 2008 (2008)
15. Cremer, J., Kearney, J.: Scenario authoring for virtual environments. In: IMAGE VII Conference, pp. 141–149 (1994)
16. Rickel, J., Johnson, W.: Steve: An animated pedagogical agent for procedural training in virtual environments. In: Intelligent virtual agents, Proceedings of Animated Interface Agents: Making Them Intelligent, pp. 71–76 (1997)
17. Querrec, R., Chevaillier, P.: Virtual storytelling for training: An application to fire-fighting in industrial environment. In: Balet, O., Subsol, G., Torguet, P. (eds.) ICVS 2001. LNCS, vol. 2197, pp. 201–204. Springer, Heidelberg (2001)
18. Devillers, F.: Langage de scenario pour des acteurs semi-autonomes. Ph.D thesis, IRISA Université Rennes1 (2001)

19. Mollet, N., Arnaldi, B.: Storytelling in virtual reality for training. In: Pan, Z., Aylett, R.S., Diener, H., Jin, X., Göbel, S., Li, L. (eds.) Edutainment 2006. LNCS, vol. 3942, pp. 334–347. Springer, Heidelberg (2006)
20. Mollet, N., Chellali, R.: Virtual and augmented reality with head-tracking for efficient teleoperation of groups of robots. In: Cyberworlds, Hangzhou, China (2008)
21. Mollet, N., Brayda, L., Chellali, R., Fontaine, J.: Virtual environments and scenario languages for advanced teleoperation of groups of real robots: Real case application. In: IARIA/ACHI 2009, Cancun (2009)
22. Mollet, N., Brayda, L., Chellali, R., Khelifa, B.: Standardization and integration in robotics: case of virtual reality tools. In: Cyberworlds, Hangzhou, China (2008)
23. Brayda, L., Mollet, N., Chellali, R.: Mixing telerobotics and virtual reality for improving immersion in artwork perception. In: Edutainment 2009, Banff, Canada. LNCS, vol. 5670. Springer, Heidelberg (2009)