



Design of a virtual reality training system for human–robot collaboration in manufacturing tasks

Elias Matsas · George-Christopher Vosniakos

Received: 9 October 2014 / Accepted: 29 January 2015 / Published online: 5 February 2015
© Springer-Verlag France 2015

Abstract This paper presents a highly interactive and immersive Virtual Reality Training System (VRTS) (“beWear of the Robot”) in terms of a serious game that simulates in real-time the cooperation between industrial robotic manipulators and humans, executing simple manufacturing tasks. The scenario presented refers to collaborative handling in tape-laying for building aerospace composite parts. The tools, models and techniques developed and used to build the “beWear of the Robot” application are described. System setup and configuration are presented in detail, as well as user tracking and navigation issues. Special emphasis is given to the interaction techniques used to facilitate implementation of virtual human–robot (HR) collaboration. Safety issues, such as contacts and collisions are mainly tackled through “emergencies”, i.e. warning signals in terms of visual stimuli and sound alarms. Mental safety is of utmost priority and the user is provided augmented situational awareness and enhanced perception of the robot’s motion due to immersion and real-time interaction offered by the VRTS as well as by special warning stimuli. The short-term goal of the research was to investigate users’ enhanced experience and behaviour inside the virtual world while cooperating with the robot and positive pertinent preliminary findings are presented and briefly discussed. In the longer term, the system can be used to investigate acceptability of H–R collaboration and, ultimately, serve as a platform for programming collaborative H–R manufacturing cells.

Keywords Virtual reality · Human–robot collaboration · Safety · Manufacturing training · Interaction · Serious game

1 Introduction

In modern manufacturing systems an emerging need for cooperation (or collaboration) and work-space sharing of industrial robots and humans to execute manufacturing tasks has arisen. The goal of this coexistence and collaboration is to improve quality and productivity by enhancing workers’ sensory-motor abilities, knowledge and dexterity in manually performed tasks with the strength, accuracy and repeatability of robots. In this case, rather than replacing the human’s work with robots in repetitive and alienating tasks, collaborative assistance of robots can lead workers to perform “value-added work” [1]. Robots are already used for teaming up with, or assisting workers in assembly lines, especially in automotive and aerospace industry. For instance a new manipulator arm structure is presented in [2] consisting of a mixture of SCARA manipulator and a parallelogram structure that allows the operator to move inside its workspace and handle very heavy loads with an accuracy of few millimeters. In [1], a mobile robotic assistant vis-à-vis human assistant is evaluated in terms of performance for H–R collaboration in fetch-and-deliver tasks. In the automotive industry a slow-moving robot is reported to have been introduced, which collaborates in proximity with a human worker to insulate and water-seal vehicle doors [3]. Furthermore, a two-armed collaborative manufacturing robot has been designed and produced that automatically adapts to changes in its environment, such as obstacles, accidental contacts, dropped objects, or variable conveyor speeds. It can work close to workers without the need of safety cages, since it contains sensors

E. Matsas · G.-C. Vosniakos (✉)
School of Mechanical Engineering, National Technical University of Athens, Heron Polytechniou 9, 15780 Zografou-Athens, Greece
e-mail: vosniak@central.ntua.gr

E. Matsas
e-mail: imatsas@central.ntua.gr

and Artificial Intelligence software that enable it to “see” objects, “feel” feedback forces and “understand” tasks [4].

In a different application realm, it seems that collaborative robots are destined to provide help, offer personalized services and generally improve quality of life to the elderly, e.g. ASTRO [5] and the Giraff [6] robots. However, safety concerns keep human–robot (H–R) collaboration in industry still under-exploited, despite its seemingly offering a good balance between productivity, quality, flexibility and initial as well as running cost. For a long time now, safety of the human interacting with industrial robots is addressed by segregation between humans and robots in space and/or in time. The most critical safety aspects are the moving robot arm and the perceptibility of arm movements [7]. Modern safety techniques in H–R collaboration integrate infra-red light curtains, robotic vision with cameras, emergency-stop buttons and alarms, structured either in a stop-and-go fashion, or to slow down the robot movements. Yet, most of the above techniques usually stop the entire production when used in production lines and not in smaller cells. Collaborative robots are developing so quickly that international standards are having trouble to keep up. Although the International Organization for Standardization (ISO) has revised the ISO 10218 standard Part II, which allows cooperation of robot with personnel, it has yet to work out safety standards for collaborative robots, such as how much force a robot can safely apply to different parts of a human worker’s body [8,9].

However, no matter how safe (physically) collaborative robots are, they may not be welcomed by human workers. To assure acceptability of such a collaboration between humans and robots, all important physical and “mental” safety issues that arise must be successfully dealt with. Physical safety means that the robot does not physically injure humans. Mental safety, on the other hand, can be defined as the enhanced users’ vigilance and awareness of the robot motion, that will not cause any unpleasantness such as fear, shock or surprise [9,10].

Recent H–R interaction studies have shown that a fluent collaboration (beyond stop-and-go interaction type) requires awareness and anticipation of intention by both human and robotic agents [11]. Situation awareness can be defined as “the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning and the projection of their status in the near future” [12]. To enable and facilitate this awareness and anticipation, it is important that both humans and robots communicate their status, location and intent, either explicitly through certain cues such as audio/visual signals or implicitly via their actions and motion paths [1]. Serious games and highly interactive and immersive Virtual, Augmented or Mixed Reality training applications are preferentially deployed in such cases, since they can provide enhanced training in all three

levels of situation awareness (perception, comprehension and projection).

In terms of functionality, Virtual Reality (VR) allows users to be extracted from physical reality in order to virtually change time, space, and (or) interaction type [13]. Therefore, the three main features of VR are: Interaction, Immersion and Imagination. Virtual Reality-based Training Systems (VRTSs) are advanced computer-assisted, interactive training systems using VR technology, e.g. allowing trainees to properly test and operate new equipment before it is actually installed [14]. VRTSs typically provide trainees important perceptual cues and multi-modal feedback (e.g., visual, auditory, and haptic), enabling effective transfer of virtually acquired knowledge to real-world operation skills. The significance of VR in manufacturing training and in H–R collaboration is widely pointed out in the literature, indicatively [14–18]. Furthermore, VRTSs need to have the necessary physical fidelity to mimic the resolution of the physical world in order to be effective for tasks that are characterized by a significant perceptual and/or motor component [19]. All VRTSs can be decomposed into three distinct functional parts: (i) input devices for interaction, such as sensors, trackers, mice (ii) output devices (mainly for immersion, such as Head Mounted Displays, large 3D displays and force feedback arms), and (iii) a VR engine including data and models that constitute the virtual scene, interaction models, and a graphical representation of the user (avatar) [20].

This paper presents a highly interactive and immersive VRTS (“beWare of the Robot”) in terms of a serious game that simulates in real-time the cooperation between industrial robotic manipulators and humans, executing simple manufacturing tasks. The scenario presented includes collaborative handling in tape-laying for building aerospace composite parts. Safety issues, such as contacts and collisions are mainly tackled through “emergencies”, i.e. warning signals in terms of visual stimuli and sound alarms. Mental safety is of upmost priority and the approach followed is that warning stimuli inside a VRTS that offers immersion and real-time interaction can provide to the user augmented situational awareness and enhanced perception of the robot’s motion. The short-term goal of the research is to investigate users’ enhanced experience and behaviour inside the virtual world, while cooperating with a robot, whilst the overall long-term goal is to investigate the acceptability of H–R collaboration and enhance the relevant terms by means of such an environment.

In Sect. 2, related work on VEs for H–R Collaboration is presented. Sect. 3 provides an overview of the system and details of the VE, whilst Sect. 4 explains implementation-specifics relating to interaction techniques and the 3D user interfaces for tracking and interaction. Preliminary findings concerning user’s experience are presented and briefly dis-

cussed in Sect. 5. Finally, in Sect. 6, conclusions and future work are summarized.

2 Related work

In the literature, a deictic mode of sensory-motor control with symbolic gestures and a Virtual Research Head Mounted Display (HMD) for teleoperation and teleassistance, was first proposed in [21], that can be useful for H–R interaction and control. Morioka and Sakakibara [22] developed a new cell production assembly system, in which physical and information supports are provided to the human operators to increase productivity. A human and a mobile robot cooperate in a shared workspace without stopping with the use of vision and speed restriction techniques. Krüger et al. [23] gives a survey about H–R cooperation in assembly, safety systems, and all available technologies that support the cooperation. Giesler et al. [24] presented an Augmented Reality (AR)-based approach wherein human and robot mostly share skills rather than physically collaborate. Corrales et al. [25] present the implementation of real-time proximity queries between humans and robotic manipulators, used by safety strategies / algorithms for avoiding collisions. From another point of view, Arai et al. [26] deal with mental strains and stress of human operators in H–R cell production systems and propose metrics (distance, speed) for a physiologically comfortable collaboration, concluding that to reduce mental strains human operators need to be notified of robot motions before they happen. Charoenseange and Tonggoed [27] present an implementation of H–R collaboration in shared workspace for virtual assembly tasks, using AR techniques to provide necessary information to the human operator. An AR interface for interactive robot path planning is also shown in [28]. Several H–R interaction methods and a Euclidean distance-based method are implemented, to assist the user in path planning, within the AR environment. In [17], a H–R collaboration experiment is conducted in an immersive Virtual Environment (CAVE), to evaluate the latter with respect to real-world applications. A markerless interface and a Kinect™-based interaction method that allows a human operator to communicate his movements to a dual robotic manipulator during an object manipulation task is presented in [29]. Wang et al. [30] report a novel, vision-guided method of real-time active collision avoidance for H–R collaboration in an augmented environment, where the system can alert the operator, stop or modify the robot's trajectory away from an approaching operator. Qiu et al. [31] propose a real-time driving model for personalized Virtual Human modeling, and integrate the complete model into a VR environment for interactive assembly. Zaeh and Roesel [32] present a safety-based smart H–R system, which perceives its environment with multiple sensors and integrates a robot with reactive behavior and an oper-

ator. Neuhoefer et al. [33] developed an immersive VR/AR simulation system for H–R cooperation, with a virtual robot and real-time physics simulation capabilities. Usability evaluation between the AR and the VR configuration showed users' preference towards the former. Weistroffer et al. [34] presented an immersive VR simulating environment to study how people perceive robots during collaboration tasks, as well as the acceptability of H–R collaboration. To summarize, a large number of authors have thoroughly tackled H–R collaboration safety aspects, as well as tools and methods for safe collaboration. Many authors have also studied VR or AR environments for teleoperation and teleassistance. Very few studies focus on mental safety issues and acceptance of H–R collaboration using immersive and interactive VR systems.

3 VRTS overview

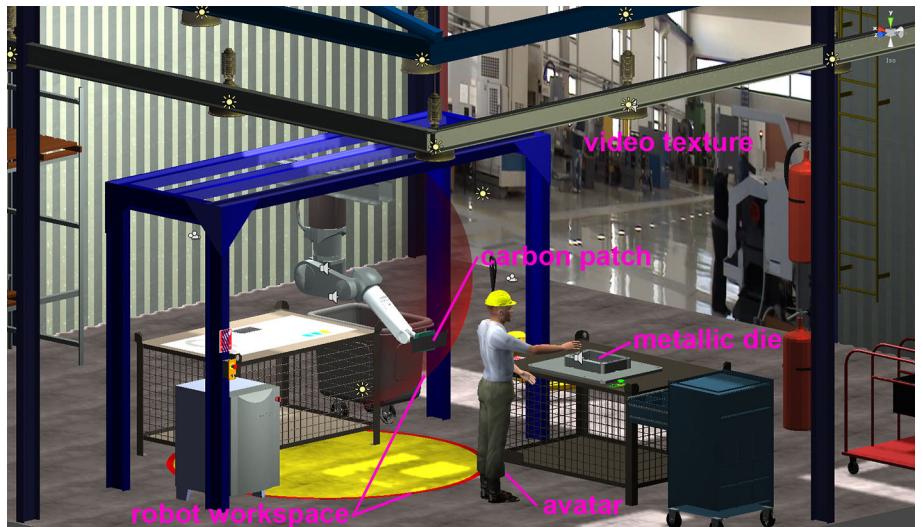
In this section, the tools, models and techniques exploited in building the “beWear of the Robot” application are described. System setup and configuration are presented in detail, as well as user tracking and navigation issues. Special emphasis is given to the interaction techniques used to facilitate virtual H–R collaboration implementation.

3.1 Use case

A use case scenario was developed, representing H–R collaborative tape laying for building aerospace fabric reinforced composite parts. Conventional tape laying typically involves carbon fiber reinforcement in the form of profiled fabric layers (patches/cloths) being stacked manually by an operator inside a die successively, one on top of the other, until the desired thickness is reached. In the H–R collaborative scenario, the robotic manipulator is assigned the picking and the transfer of the patches towards the human. Once the user takes the patch from the robot and places it in the right position inside the metallic die located in front of him, the robot proceeds to feed the next patch. The process is repeated until all different patches are properly placed in the die by the human, hereby represented by the avatar's hands. These direct H–R collaborative tasks are kept as simple as possible, but they involve close proximity between human and robot and they are performed in parallel, as a “hand-to-hand” collaborative manipulation scenario. The latter was selected to represent typical “hand-to-hand” H–R collaborative manufacturing scenarios, to ensure further exploitation in the future.

The robot arm is attached upside-down to a metallic structure next to the user and the workbench on which the die lies, see Fig. 1. The robotic arm takes a carbon patch (cloth) from a workbench on which the patches are stacked, and handles it towards the user. When the robot moves, the user is capable of both virtually seeing and hearing it. In addition, the

Fig. 1 The shop-floor environment and the main VE components as seen from a third person shooter inside Unity 3dTM



robot's work envelope is always visible by the user as a semi-transparent, red, toroidal type surface. The latter is an exact representation of the real robot workspace, and its projection on the floor is also depicted in Fig. 1. When the user enters the robot's workspace, the red surface is blinking and a sound alarm is turned on, to warn the user of the robot's proximity and a possible collision. The use of these visual and audio warning stimuli might be argued to be imprecise or unrealistic and even to compromise physical fidelity, however such exaggerated triggers can prove very effective for tasks that are characterized by a significant perceptual component [19], and can provide the user enhanced situational awareness and alertness of a potentially hazardous workspace [35]. Moreover, according to Bowman et al. [36], 3D user interfaces should deliver only as much realism as is needed for the particular application.

3.2 System setup and configuration

“BeWear of the robot” VRTS is a standalone build developed on the Unity 3dTM game engine platform that can run on a typical PC with WindowsTM operating system. The system platform consists of: the “beWear of the robot” application, a PC running Windows XPTM equipped with an nVidia QuadroTM FX1700 graphics card, an eMagin Z800 3DVisorTM HMD with stereo ear buds, a Microsoft KinectTM sensor, keyboard and mouse. Furthermore, in order to be able to reproduce what the user sees through the HMD and to record the user with a video camera, a projector is normally used, cloning the displays of the HMD on a wall behind the user. KinectTM sensor and headtracker (input) data are transmitted to the PC over USB cables. Communication between Unity 3dTM and the KinectTM sensor is implemented with the OpenNI framework. Figure 2 shows a diagram of the system and data flow between its main components.

Shop-floor environment and its components were developed using RhinocerosTM and 3ds MaxTM software, for 3D part design, mesh creation and editing, and avatar's biped design. The skinned model of the avatar was created online in the Evolver website [37]. 3D parts and models are exported in “.fbx” file format for increased compatibility with Unity 3dTM software. Unity 3dTM game engine was used for assembly, rendering, lighting, shading, physics and collision modelling, simulation, programming and building the compilation.

The VRTS incorporates (i) the assembly of original 3d models forming the virtual model of a composites hand layout work-cell, (ii) the model of a StäubliTM RX90L robotic manipulator, (iii) the skinned model of an avatar with a biped attached to it, (iv) real-time data exchange with the tracking devices, (v) interaction scripts in C# and Unity's JavaScript mainly concerning collision and ray-casting, child/parenting functions and skeletal tracking of avatar joints, (vi) real-time shadows, rendering and lighting, and (vii) image, video and audio textures from real working spaces. The virtual scene of the shop-floor, its main components 3d models and the avatar are depicted in Fig. 1.

3.3 The virtual scene—simulating environment

During execution and testing of the “beWear of the robot” application, the user is asked to wear the HMD and to stand straight at a distance of 2–3 m towards the KinectTM sensor. The VE is rendered in real time in frame-sequential stereoscopic vision in the HMD, and a monoscopic duplicate of it is shown typically through a wall projector behind the user. When the system starts the user sees the initial screen of the VE and a calibration prompt, see Fig. 3.

Calibration of the user against the avatar skeleton is achieved by the user raising hands with both elbows bent at

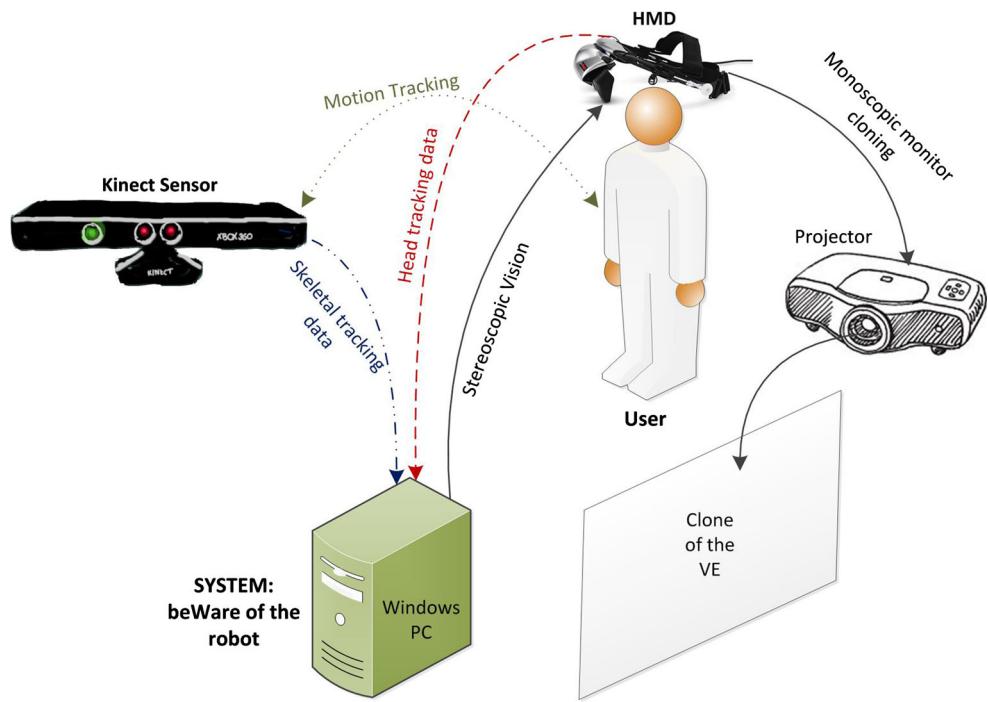


Fig. 2 System setup



Fig. 3 The initial view of the VE as seen from the user (first person shooter/egocentric view)

90° thereby standing in “Y” posture towards the Kinect sensor, see Fig. 4a. After calibrating, users are usually allowed some time to familiarise themselves with system’s tracking, navigating and travelling capabilities and to explore the different objects of the VE as well as interaction with them. When the user moves or turns in real space, the avatar’s body and the virtual viewpoint change accordingly. When the user walks around in the real space, the avatar follows the same path in the VE, though with some spatial restrictions and limits, defined by the HMD cables length and the sensor tracking range. When the user turns his head, the avatar’s head and the first person camera attached to it respond accordingly. The user can also collide or interact with rigid bodies in the virtual scene, bend his body in every direction, move his

hands and handle selected objects, see Fig. 5, noting that the clone of the VE projected on the background wall was for the benefit of the researchers. Stereoscopic vision offered by the HMD, combined with body and head tracking capabilities, provide the user an enhanced feeling of immersion and involvement. Moreover, immersion and presence feelings are enriched with 3d sound sources, i.e. robot motion sound, alarm, shop-floor sounds, real-time shadows, and the video of a real manufacturing work-cell which is reproduced on the background wall, in form of a video texture, see Fig. 1.

Once immersed in the virtual scene and familiarized with navigation and tracking, the user has to push a green button, located on the workbench, to start the training, see Fig. 4b. Apart from the green (start) button, a red, emergency stop button is also located on the workbench. When the user pushes the green start button, a collision is detected between the capsule colliders of the avatar hands and the sphere collider that envelops the button. Colliders are invisible components of the Unity physics library, defining the shape of an object for the purpose of physical collisions, elaborated on further in Sect. 4.2 below. This collision makes the robotic arm start moving from its default position towards the workbench with the carbon cloths (patches) stacked in the desired order. When the robot’s end-effector approaches the first patch in the right distance and orientation, their ray-casting colliders interact with each-other and the end-effector becomes parent of the cloth, i.e. the cloth is virtually attached to it and follows its movements. The robot then moves again and hands the patch over to the user (avatar) with the proper orientation.

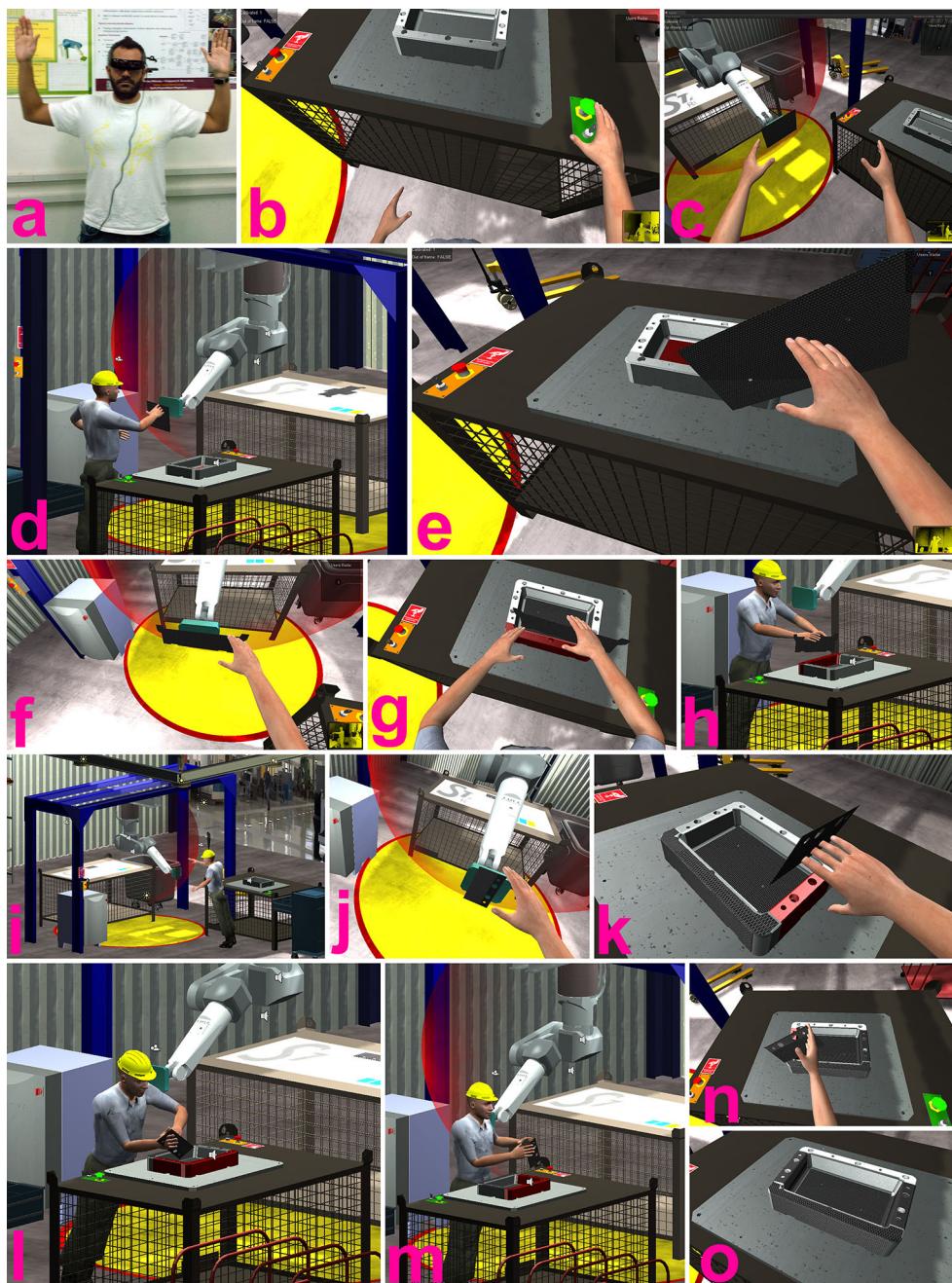


Fig. 4 A storyboard visualising the different tasks of the use-case scenario

At the same time, the user has to approach the robotic arm, see Fig. 4c, preferably with only one arm extended, and take the patch from the end-effector, minding not to collide with the robot's body. A ray-casting collider is attached to the avatar's index fingers, allowing the user to take the patch from the robot, when the fingers touch the patch, see Fig. 4d. The avatar fingers are now parents of the patch. If the user enters the robot's workspace when trying to grab the patch, the red surface representing the workspace starts flashing and an alarm sound is turned on, to warn the user to be aware of the robot's proximity and of a potential hazard.

Afterwards, the user has to move towards the workbench where the metallic die lies and place the cloth on the highlighted (rendered with red color) surface of the die, see Fig. 4e. The appropriate, different part of the die is highlighted, depending on the patch being handled, in order to guide the user as to the position, boundary and orientation of the patch.

The patch is not released from the avatar's hand until it is properly placed in the indicated area of the die. After that it is released (destroyed) and appears laid in its final position on the die, see Fig. 4f–n. Thereafter, the robotic arm moves

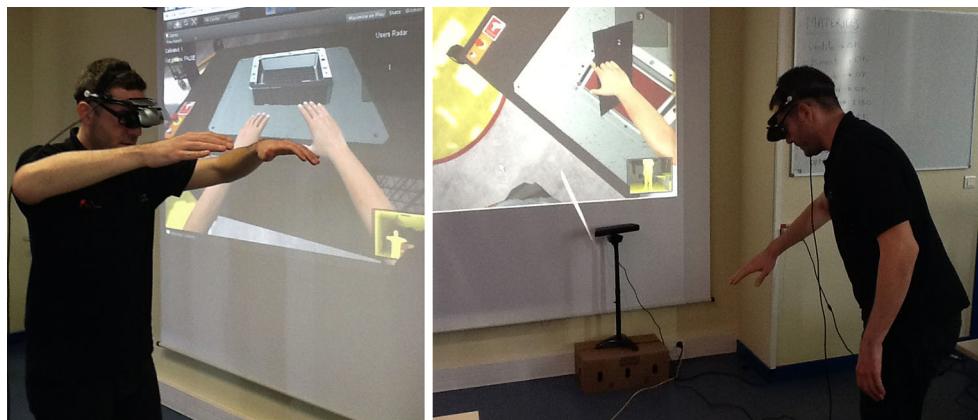


Fig. 5 Hands tracking example-avatar's hands following the user's arm movements (*left*), and avatar's hand moving a patch and placing it onto the *red-coloured surface* of the die (*right*)

again to the workbench with the carbon cloths, in order to fetch the next patch to the user. This process is repeated until all different patches are properly placed in the die (Fig. 4o) and there are no patches left on the workbench.

The completion of the above tasks, including calibration, for four different patches normally takes about 6–8 minutes. A graphical representation of the system activities workflow is given in the UML activity diagram in Fig. 6. In the latter, task-sharing between the user, the tracker and the robot is presented in three vertical swim-lanes respectively.

3.4 Avatar

An avatar is very seldom used in H–R collaboration applications reported in literature irrespective of how immersive or interactive the latter are. Most of them use either a fixed camera, or a simple first person shooter to abstractly simulate the user's vision and navigation. Aiming at a more fluent and realistic collaboration between the user and the robot, it was decided to involve a skinned and fully functional male avatar model. The inclusion of a realistic and functional virtual body may enhance the sense of presence, which in turn has a positive effect on spatial knowledge acquisition and usage [36,38]. Furthermore, Usoh et al. [38] showed the importance of user association with the virtual body. Thus, a highly detailed avatar was created consisting of 11,000 polygons. In addition, when the user is looking down, or is simply facing his arms, he can see a realistic representation of his tracked virtual hands and arms. This is considered to be essential for the sense of presence in such a first-person shooter application, where the user is wearing a HMD and the real world is completely blocked from his view.

The avatar shown in Fig. 7 was created as a textured mesh model, with a custom face (clone) and a body animation rig. A biped was attached to the avatar mesh model, making it kinematically functional, i.e. the different body parts mesh,

the skin and cloth textures follow the biped's kinematics. The representation of the avatar object in Unity game engine has a straightforward tree hierarchy, with the mesh model as a parent and 3D Transforms ordered as children according to the biped's kinematic chain.

The main camera is attached to the avatar's head as a child of the head transform, at eye height and with $\pm 60^\circ$ field of view about the local vertical axis. A first person shooter is created in this way, offering to the user the perspective of the avatar, and a view of his hands, if combined with head and body tracking. The first person perspective was preferred to the third person perspective because it arguably offers an augmented presence, realism and immersion feeling; every action taking place in the VE can be experienced by the user as if performed by him/her.

A capsule collider bounds the avatar's torso and two smaller capsule colliders are attached to its hands. The colliders are used by the physics engine of Unity game engine for collision detection and for event triggering as explained in Sect. 3.3. Rigid body physics is also added in the avatar's hands, head and torso. The 11 avatar tracking points (Unity Transforms) that are used for skeletal tracking of the user are depicted in Fig. 7 on the right.

3.5 Robot

The industrial robot manipulator arm used in the H–R collaboration scenario is a Stäubli™ RX 90L with six rotational joints. It can lift objects with a maximum weight of 12 kg and a maximum reach of 1185 mm. The virtual robot end effector is a simplified vacuum based custom design for fast handling of the lightweight carbon patches.

The virtual robot arm is suspended from a structure next to the workbench, so that it can easily collaborate with the user, as shown in Fig. 8. The robot object in Unity has a straightforward tree hierarchy, allowing proper robot motion according

Activity Diagram: Swims

beWear of the Robot workflow of the main tasks

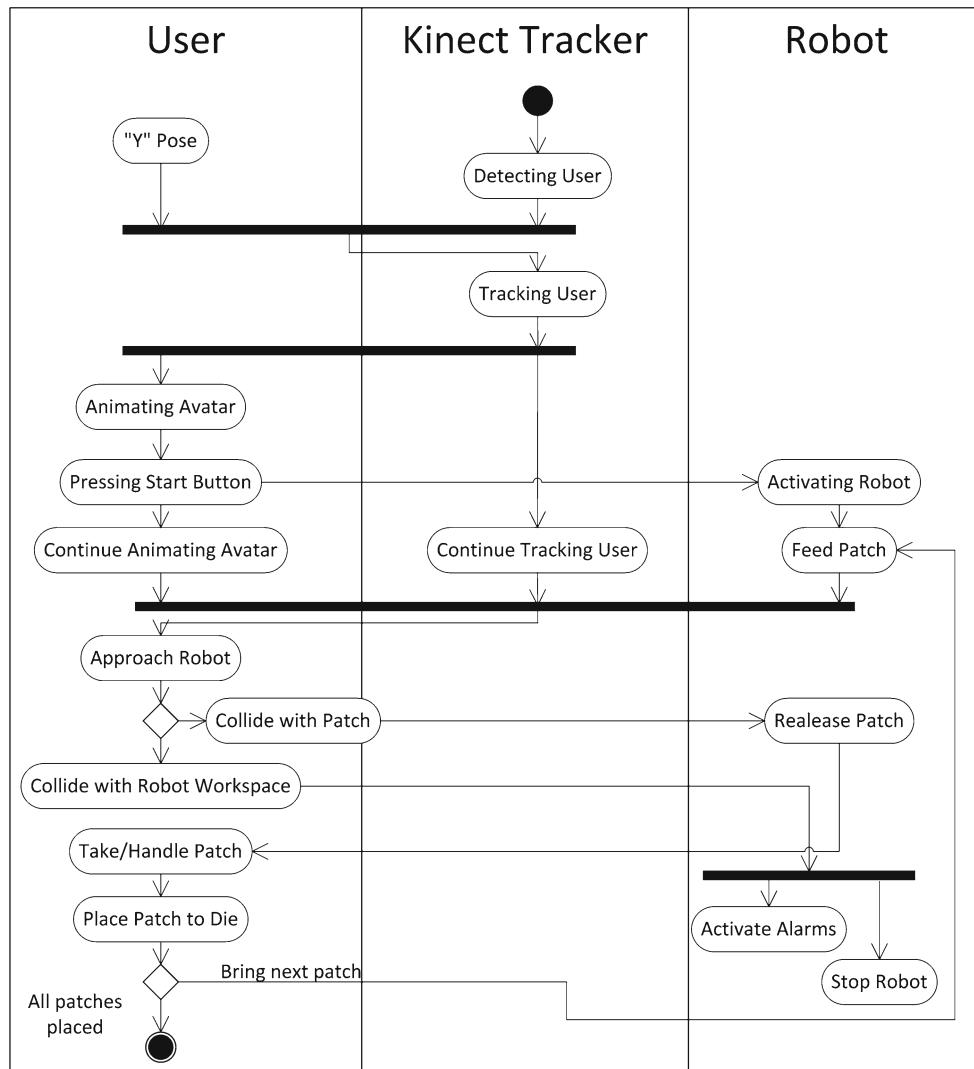


Fig. 6 UML activity diagram for the system workflow

to robot open kinematic chain, starting from the base joint (J1) being just a parent and successively moving to the last one (J6) being just a child. In the VRTS, the robotic arm is controlled by forward kinematics scripts only, i.e. joint values are only used to control the robot motion and determined position and orientation of the end-effector.

A combination of mesh and primitive-shape colliders is used in the robot structure to increase execution speed; primitive-shape colliders such as box, sphere or capsule colliders are faster in collision detection calculations, compared to complex mesh colliders. Capsule colliders are attached to the base, the arm and the forearm of the robot, and box colliders to the shoulder and the elbow. In the end-effector, though, where extra accuracy is needed to avoid false-collisions with the avatar's hand, a mesh collider was preferred.

The robot work envelope is depicted as a penetrable, semi-transparent toroidal-like surface form, an exact copy of the real robot workspace, with a mesh collider and the alarm audio source attached to it.

4 Implementation

4.1 Unity VR engine

Unity is a platform for game creation including a game engine and an Integrated Development Environment (IDE). All the necessary resources (models, textures, sounds etc.) are imported in a Unity project as assets, which are included in GameObjects (classes) [39]. Components and scripts are

Fig. 7 The skinned avatar inside the VE with the capsule collider attached to its *left hand* (*middle*), and, the 11 skeletal tracking points in the avatar's biped (*right*)

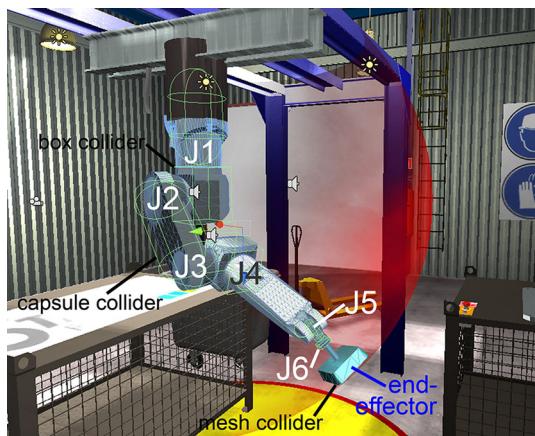


Fig. 8 The robotic arm and the different colliders attached to it

then attached to GameObjects to control their behaviour. Components, or combination of components, usually add functionality to GameObjects, and they can be transforms, cameras, colliders, particles, rays, rigid bodies, audio sources etc.

Scripts are essential elements of every VR application, and they are treated by the game engine as original components, written by the developer. Scripts can be used to respond to inputs, trigger events or wait for events to run, or even to create special physical or kinematical behaviours. Three programming languages are supported in Unity: (i) C#, (ii) UnityScript which is modelled after JavaScript, and, (iii) Boo, a language similar to Python. All original scripts developed for the “beWare of the Robot” application are written in C#. Two functions are defined inside every class in Unity scripts: (i) the Start function, which is called once by Unity, before the play mode begins, and where all properties are set

to their initial values, and, (ii) the Update function, which is handled during the frame update in play mode, and which usually includes movements, actions triggering, responses to user actions, and whatever needs to be handled over time.

A very useful Unity asset type is the Prefab. Prefabs are instances of a GameObject, cloning the complete original one with its components and its properties. Any changes made to a prefab are immediately applied to all its instances. The advantage of Prefabs is that they are re-usable and they can be easily instantiated or destroyed throughout the runtime, with a few lines of code. In the scenario implemented composite patches creation, selection and placement are implemented with prefabs.

Transforms are the essential kinematic components in Unity; transforms are the equivalent components to 3D frames of other VR engines. Every GameObject has a transform attached to it, determining its position, rotation and scaling, relatively to the transform’s parent.

4.2 Interaction techniques

In VR, an interaction technique is a means or a method including both hardware (I/O devices) and software components for accomplishing the desired interaction task, via a user interface (e.g. manipulating, moving, selecting etc) [36]. The most preferable, successful and tried-and-true interaction techniques attempt to simulate, or adapt from, the real, physical world, referring to real behavioural schemes. If these schemes are difficult to implement, 3D interaction metaphors can prove very effective [13]. Such metaphors or a combination of them, form the fundamental mental model of a technique, a symbolic image and a perceptual manifestation of what users can do by using these techniques [36].

The typical generic interaction tasks in VEs are the following: (i) selection, (ii) manipulation, (iii) navigation (travel and way-finding), and, (iv) system control [36]. All of the above tasks are performed during the runtime of ‘BeWare of the Robot’ application, and they are accomplished with the use of a wide variety of natural interaction techniques (schemes). No metaphors or “magic” (i.e. overcoming human or real world limitations) techniques are employed. Many of the techniques used for some tasks are used in combination with interaction techniques for other tasks not directly related to them (cross-task techniques), e.g. the ray-casting technique is used for selection and for system control, or, the direct virtual hand technique is used for manipulation and activation of controls.

Ray-casting and collision-based pointing techniques are used for selection, and direct-virtual-hand, real hand gestures, collision, and child/parenting techniques are used for manipulation. A ray casting script is attached to the avatar’s index fingers and the robot end-effector, which casts a small, invisible ray in a specific direction against all colliders, and returns a true Boolean value when the object with the desired tag is hit. If this value is true during H-R collaboration, the cloth is attached to the avatar’s hand, i.e. it becomes a child of the hand’s transform. According to the parenting technique, the child inherits the movement and rotation of its parent. Although there are two interfering rays hitting the patch (when the user tries to grab the patch from the robot), the user is favoured because the ray cast from the avatar’s fingers is longer than the ray cast from the end-effector. The user then has to properly position and lay the cloth on the red-coloured area of the die. This task is accomplished in the VE with a combination of interaction techniques. First, the box collider of the patch collides with the red-coloured part of the die and a trigger event is sent destroying the patch object; the patch thus appears released from the avatar’s hand. At the same time, a prefab of the laid patch, which was not visible before, is instantiated and rendered, appearing as if it were automatically placed, laid and shaped in its final position of the die.

Collision detection and triggering is the main interaction technique, and it is mostly implemented using sphere, box and cylinder covering techniques. The use of primitive shape colliders (or combination of them) is preferred to the use of complex mesh colliders, for increased execution speed and for maintaining a steady frame rate. During typical tests runtime, a steady frame rate, greater than 25 frames per second is obtained, which balances affordability and quality. Collision detection can be used either for solid object collision behaviour, and thus, initiate rigid body physics actions (`onCollisionEnter` function), or just as a trigger event (`onTriggerEnter` function) that allows other colliders to pass through, and simply triggers an event or calls another function. For example, when a game object tagged as “avatar” collides with the trig-

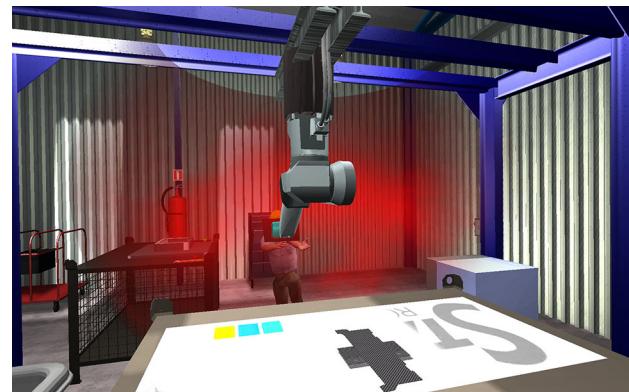


Fig. 9 The visual alarm is enabled (flashing), warning the user that he/she has entered the robot workspace

ger mesh collider of the robot workspace, the `OnTriggerEnter` function enables the audio alarm to start playing, and makes the renderer of the robot workspace start flashing, see Fig. 9.

Navigation and travel are implemented with simple physical motions and (real) walking techniques, since head and body tracking is used. Navigation techniques based on tracking and walking are natural and instinctive, allowing therefore the user to carry out other, more important tasks without being distracted even by thinking about how to move around. Another advantage is that the hands are not used at all for navigation, thus being entirely free to be devoted to the purposes for which they are used in reality. In addition, walking technique provides to the user vestibular cues that help him understand the size of the environment [36]. However, walking is not always feasible due to real space limitations; in the current case real walking is nevertheless an appropriate and effective technique, as the size of the active working environment is equal to the range of the Kinect tracker, and the cabling length of the I/O devices allows freedom of movement in the tracked area.

The system control task is crucial since it allows the user to control the interaction flow between other tasks of the application; a command is issued to request the system to perform a function, or to change the system state. In this case, neither Graphical User Interfaces (GUIs: buttons, menus etc), nor physical tools are used for system control. The main technique used is the tracking-driven virtual hand that interacts (selects, manipulates) with specific objects, and that in turn triggers an event or a change in the system state/interaction mode. For example, when the user’s virtual hand interacts with the green button, e.g. the user walks to, points with his hand to, approaches and finally touches/collides with it, a function is triggered that moves the robotic arm and makes it fetch the first patch. Similarly, patch selection, grabbing, placement and release, call a function and change the system state or request it to perform an action (parenting, render, change colour etc). However, some conventional system con-

trol techniques are also used, such as the postural commands technique during the calibration process. Finally, mouse and keyboard are still used, but for minor system control tasks, such as the exit-on-escape-button function, which terminates the application, and parameter value typing in the settings window at the beginning of the application.

4.3 3D user interfaces for interaction and tracking

3D User Interfaces (UIs) involve 3D interaction, in which user tasks are performed directly in a 3D spatial context [36]. In “beWare of the Robot” application, two main 3D UIs are employed: an eMagin Z800 3DVisor HMD and a Microsoft Kinect™ sensor.

The HMD is used as an output device for visual display, stereoscopic vision and 3D sound, and as an input device for head motion tracking. The Kinect sensor is used as an input device for skeletal tracking and positioning of the user.

The Z800 HMD uses two OLED microdisplays to accomplish the stereoscopic effect, when a frame-sequential 3D signal is transmitted by the graphics card. Although the Z800 HMD has three integrated gyroscopes and accelerometers to detect head motion along the X, Y and Z axis, HMD head tracking is limited to the head’s pitch angle, i.e. head turning up or down (vertical head tracking) in the range $\pm 60^\circ$. Data from the other two gyroscopes are not exploited, since a tracking conflict arose between the tracking data of the Kinect on the one hand, and the data of the HMD gyroscope on the other hand. More specifically, the conflict was caused by the simultaneous horizontal tracking of the head (yaw rotation angle) by the HMD, and of the horizontal skeletal tracking of the shoulders and the neck by the Kinect sensor; for example, the avatar (camera) exhibits intensely and irregular trembling when the shoulders turn in one direction and the head turns in the opposite direction, thus making the immersion experience unbearable. This is the reason for not using the Kinect tracking data of the upper body (head and neck joints).

The HMD headtracker, which comes bundled with a mouse-emulating software, translates the detected motion into smooth and accurate mouse motion on the screen. Specialized settings in the bundled software and the character controller settings in Unity are also used to fine-tune mouse emulation. In Unity VR engine, the main camera is attached to the head transform, which is child of the character controller and the avatar. A “Mouse Look” script is attached to the main camera, mapping the mouse translational move to rotational move of the camera.

3D skeletal tracking and positioning using Microsoft Kinect™ is conducted by continuous processing of RGB and depth images of the user [29], captured by the sensor that is fixed in front of him. The sensor contains two cameras (one infrared and one standard video camera) and an

infrared projector that produces a grid of dots (structured IR light at 830 nm) that measure the distance of objects from the Kinect, and compose a “depth map” of the image [40]. The output images are downsampled and transmitted to the host device via the USB cable, at a frame rate of 9–30 Hz. Kinect software takes the computed depth map and infers body position using machine learning and “known skeletons”, thereby transforming the body parts image into a skeleton.

The OpenNI framework was chosen as the middleware for Kinect, both for skeletal tracking, as well as a basic driver. Tracking data of twenty body joints were available. However, data corresponding to just eleven of the joints were sufficient and were therefore matched to the correspondent biped control points, see Fig. 7. In particular, head, neck and lower leg tracking, including feet, is not carried out. As mentioned above, head tracking is performed by the HMD. In addition, the absence of neck, leg and feet tracking by the Kinect™ sensor does not affect the overall tracking quality or the immersion feeling in such a first person shooter application, since the overall response of the biped to the movements of the tracked joints was accurate enough. On the contrary, it was noticed that lack of feet tracking made the avatar’s movement smoother and steadier in the vertical direction. When the user looks down he/she could see his/her tracked virtual hands and arms, and probably the untracked feet. However, given the fact that the feet are always oriented in the hips direction, feet tracking absence is not noticeable and was not described by any user as a distraction.

As far as Kinect’s™ tracking limitations and problems are concerned, observations in [35] revealed that almost half of the users sometimes “lost control” of their hands, even instantaneously. That is because experiment tasks required user body rotation towards the sensor, and at the extremes hands tracking is confusing: Kinect™ sensor cannot easily distinguish the left from the right hand. In addition when users turn their body more than 90° to the Kinect sensor axis, skeletal tracking is sometimes lost, and avatar remains “frozen” for a while in the last detected posture, although the user may continue moving.

5 User experience and discussion

In interactive VEs, the human user becomes a real participant in the virtual world, interacting and manipulating virtual objects. Human performance in training VEs highly depends on the degree of presence that the user experiences, i.e. the subjective experience of being in one place or environment even when one is situated in another. Indeed, presence is the most important measure for situation awareness in an interactive VE [41]. Prothero et al. [42] claim that “presence” and “situation awareness” are overlapping constructs. It is generally agreed that the two necessary conditions to experience presence are immersion and involvement. This work

Table 1 Test results (SD: strongly disagree, D: strongly agree, N: neutral, A: agree, SA: strongly agree)

Question	SD	D	N	A	SA
I feel like I was really moving in the scene	0	1	3	5	21
Wearing the HMD I feel present and involved in the virtual activity	2	1	5	15	7
I lost my concentration, even instantaneously, during the experiment	28	0	0	2	0
Wearing the HMD I feel immersed in the virtual environment	1	0	7	13	9
I feel like I was really moving an object with my hand, despite the fact that the object did not have physical mass	1	7	8	12	2
I did not easily perceive the red transparent surface that represented the robot workspace	29	0	0	0	1
Audiovisual stimuli (alarms) helped me being aware of a potentially hazardous workspace	0	1	1	8	20
I did not encounter any difficulties during the initial calibration process with the Kinect sensor	0	3	3	6	18
I feel like the avatar followed my body and head movements precisely	1	2	12	14	1
It was easy to take the patches from the robot's end-effector	0	0	9	13	8
It was easy to navigate/move in the virtual world (ease of movement, restraint)	0	3	12	12	3
I feel like my behavior in the virtual world didn't change, compared to my behavior in the real world	3	10	9	8	0
I feel more like I was participating in an amusing game	0	2	5	10	13
After the experiment I went through, I believe that H–R training tasks can be more attractive with the use of "serious games"	1	0	2	8	19

focuses more on “presence by involvement” [43], i.e. both involvement and immersion are thought to be necessary for experiencing presence.

BeWare of the Robot was tested by a group of 30 pre-final year mechanical engineering students and the detailed findings are published in [35]. A summary of the main results is shown in Table 1.

The results revealed that the system scored great in “presence by involvement” and in navigation issues; a vast majority of subjects were feeling as if they were really moving in the scene, “involved and present” in the virtual activity, without losing their concentration at all during the test. It was also found that participants felt immersed in the VE when wearing the HMD and a large number of them felt as if they were really manipulating an object with their hands, despite the fact that the object did not have physical mass.

During the H–R collaboration procedure almost all subjects answered that they easily perceived the red transparent surface that represented the robot's workspace, and 70 % of subjects managed to complete the H–R collaborative tasks without entering into the workspace (which was the potentially hazardous area). Most of the subjects (93 %) favourably accepted the use of visual and auditory stimuli (alarms) for the robot's workspace awareness. Note that, as already mentioned in Sect. 3.1, the employment of the red surface (robot workspace) and of the visual and audio warning stimuli might seem contradictory with fidelity and realism of VEs, but realism and level-of-detail are not always the ultimate goal.

In VRTSs learning procedure for example, exaggeration or deformation of real events is “authorized” for understanding complex situation [44], for enhancing situation awareness [36], and for alerting in a potentially hazardous workspace [35].

Concerning usability and tracking quality, very few subjects reported encountering any difficulties during the initial detection and calibration process by the Kinect sensor. Most of the participants considered that the avatar was following their (tracked) movements precisely. On the other hand, 43 % of the subjects felt that their movements were altered compared with the real ones, and 23 % of them pointed out a slight “vibration” in the avatar's hands movements. Nevertheless, 70 % of the subjects answered that they easily managed to pick-up the cloth patches from the robot's end-effector, while the majority said that it was easy to navigate (travel) in the virtual world.

According to [38] immersion requires that there is a match between the user's proprioceptive feedback about body movements and the information generated on the displays of the HMD. This is obviously favoured by a detailed body mapping. In this work, where tracking is implemented seamlessly, contactlessly, and without time delay, the consistency between proprioceptive information and sensory feedback is high.

As to user motivation, 76 % of the participants replied that during the experiment they were feeling more as if they were participating in an amusing game, and 90 % of the subjects

believe that training tasks requiring H–R collaboration can be more attractive with the use of interactive “serious games”.

Eventually, the results suggest a positive prospect for the use of VR for training on H–R collaboration. Users experience a high level of involvement and immersion, and therefore, presence. The inclusion of a realistic avatar, that is the user’s own kinematic representation, strengthens the sense of presence. Stereoscopic vision combined with the body and head tracking capabilities provide the user an enhanced immersion feeling and have a positive effect on spatial knowledge. Moreover, first-person perspective was preferred to the third-person perspective, because it offers an augmented presence feeling; every action taking place in the VE can be experienced by the user as if performed by him/her. In addition, moving around the scene and using real gestures (skeletal tracking) to manipulate objects and to complete the tasks makes learning more active and impressive for learners than conventional, non-interactive training methods. Ultimately, the more control the user has in interacting with the VE and over the system, the higher his/her experience of presence and awareness, and the greater the success of the VRTS.

6 Conclusion and future work

A novel interactive VRTS named “beWare of the robot” has been presented, in terms of a serious game for H–R collaboration. This short-term aim of the research was to construct this system as a platform for experimentation. Thus, this paper presented an extended overview of the system, functional details of the Virtual Environment, interaction techniques and tools used, as well as 3D user interfaces for interaction, and especially for tracking.

The medium-term aim of the research concerns the acceptability of H–R collaboration, in regard to safety issues that arise during the execution of collaborative manufacturing tasks in a shared workspace. To establish the acceptability of the collaboration, mental safety issues, i.e. human’s awareness and vigilance of the moving robot, have to be tackled. The line adopted in this work is that immersive serious games simulating H–R collaboration are the most pertinent tool to facilitate situation awareness, since they can provide enhanced training at all three levels of situation awareness (perception, comprehension and projection).

H–R collaborative tape-laying for composite parts was implemented as a relatively simple use case, in which the virtual world includes a shop floor environment, an industrial robot and an avatar, a metallic die, carbon fibre patches, and several auxiliary machine shop objects. Both the VE model and the training scenario were designed to have the necessary fidelity and to be as close as possible to reality.

What distinguishes this work is primarily the wide “blending” of different interaction techniques that were used. More

than 25 original scripts were deployed to accomplish all kinds of interaction tasks: selection, manipulation, navigation, and, system control. The main interaction techniques that were used are: ray-casting, collision detection, child/parenting, tracking-driven virtual hand, and, real walking technique. Using markerless body and head tracking, human-system interaction is natural, replicating real and inherent movements. A second novelty of this work is the simultaneous employment of the Kinect™ sensor together with the HMD, for skeletal and head tracking respectively. The HMD is used as an output device for 3D visual display and as an input device for head tracking. Head-tracking quality and presence feeling are therefore higher, because by using one device for both activities, the data generated on the output displays correspond precisely to the input tracking data, and the match between proprioception and sensory feedback is considerable. A further novelty is the inclusion of a high quality avatar with the first person shooter, allowing the user to observe his/her tracked hands, and to experience any activity as if it was really performed by him. This first-person enhanced interactive experience is difficult, or even dangerous to be naturally performed in the real world involving a real robot. Consequently, the VRTS can be seen as a tool to obtain knowledge, and transferring (returning) this knowledge back to the real world.

Overall, testing results are positive and suggest a good prospect in the use of such applications for H–R collaboration simulation or acceptability testing. It would be difficult to claim, of course, that results can be generically applicable to all aspects of H–R collaboration; the findings however suggest that the immersive experience combined with warning stimuli, can help the user to experience the feel of presence as a situation awareness phenomenon, which is, eventually, the intent of the research.

The long term-aim of the research is to consider task-based or even safety-based robot motion programming according to situation awareness and anticipation of intention by both human and robot. For instance, the human could wait for the robot to finish its task, move away or move with caution, or finally perform an alternative task. Conversely, the robot would change its posture, work on another possible task (patch), slow down or move with priority to a specific joint.

Concerning tracking quality and experience newer I/O devices would be considered, such as the Kinect™ for Xbox One sensor for tracking, the Leap Motion™ device for precise hand/finger tracking as well as immersion devices offering wider field of view and extensive headtracking (e.g. the Oculus Rift™).

Acknowledgements This research has been co-financed by the European Union (European Social Fund—ESF) and Greek national funds through the Operational Program “Education and Lifelong Learning” of

the National Strategic Reference Framework (NSRF)—Research Funding Program: Heracleitus II. Investing in knowledge society through the European Social Fund. The authors would also like to thank Dimitrios Batras, MSc, for his generous help in programming, his motivation and his helpful comments during the development of the application.

References

- Unhelkar, V.V., Siu, H.C., Shah, J.A.: Comparative performance of human and mobile robotic assistants in collaborative fetch-and-deliver tasks. In: Proc. 2014 ACM/IEEE Int. Conf. Human-robot Interact—HRI '14, pp. 82–89 (2014). doi:[10.1145/2559636.2559655](https://doi.org/10.1145/2559636.2559655)
- de Santos, P.G., Garcia, E., Sarria, J., Ponticelli, R., Reviejo, J.: A new manipulator structure for power-assist devices. Ind. Robot. An. Int. J. **37**(5), 452–458 (2010). doi:[10.1108/01439911011063272](https://doi.org/10.1108/01439911011063272)
- Knight, W.: Smart Robots Can Now Work Right Next to Auto Workers. <http://goo.gl/EUbRNX> (2013). Accessed 22 April 2014
- Rethink robotics: baxter robot overview. <http://goo.gl/fRUqou> (2014). Accessed 22 April 2014
- Coradeschi, S., Cesta, A., Cortellessa, G., Coraci, L., Gonzalez, J., Karlsson, L., Furfari, F., Loutfi, A., Orlandini, A., Palumbo, F., Pecora, F., von Rump, S., Stimec, A., Ullberg, J., Otslund, B.: Giraff-Plus: combining social interaction and long term monitoring for promoting independent living, 2013. In: 6th Int. Conf. Hum. Syst. Interact, pp. 578–585 (2013). doi:[10.1109/HSI.2013.6577883](https://doi.org/10.1109/HSI.2013.6577883)
- Cavallo, F., Aquilano, M., Bonaccorsi, M., Mannari, I., Carrozza, M.C., Dario, P.: Multidisciplinary approach for developing a new robotic system for domiciliary assistance to elderly people. In: Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. IEEE Eng. Med. Biol. Soc. Annu. Conf., pp. 5327–5330 (2011). doi:[10.1109/HSI.2013.6577883](https://doi.org/10.1109/HSI.2013.6577883)
- Helander, M.G.: Ergonomics and safety considerations in the design of robotics workplaces: a review and some priorities for research. Int. J. Ind. Ergon. **6**(2), 127–149 (1990). doi:[10.1016/0169-8141\(90\)90018-W](https://doi.org/10.1016/0169-8141(90)90018-W)
- Working with robots: our friends electric. <http://goo.gl/v8DhAK> (2013). Accessed 22 April 2014
- Ogorodnikova, O.: Human Robot Interaction: The safety challenge (an integrated frame work for human safety). PhD dissertation, Budapest University of Technology and Economics, Faculty of Mechanical Engineering, Budapest (2010)
- Alami, R., Bicchi, A., Bischoff, R., Chatila, R., De Luca, A., De Santis, A.: Safe and dependable physical human–robot interaction in anthropic domains: State of the art and challenges. In: Bicchi, A., De Luca, A. (ed.) Proceedings IROS Workshop on pHRI—Physical Human–Robot Interaction in Anthropic Domains (2006)
- Hoffman, G., Breazeal, C.: Effects of anticipatory action on human–robot teamwork efficiency, fluency, and perception of team. Proceeding ACM/IEEE Int. Conf. Human-robot Interact. - HRI '07, 1–8 (2007). doi:[10.1145/1228716.1228718](https://doi.org/10.1145/1228716.1228718)
- Wickens, C.D.: Situation awareness: review of Mica Endsley's 1995 articles on situation awareness theory and measurement. Hum. Factors. **50**(3), 397–403 (2008). doi:[10.1518/001872008X288420](https://doi.org/10.1518/001872008X288420)
- Fuchs, P., Moreau, G., Moreau, G., Burkhardt, J.-M., Coquillart, S.: Le traité de la réalité virtuelle, vol. 2. L'interfaçage, l'immersion et l'interaction en environnement virtuel. Les Presses de l'Ecole des Mines de Paris, Paris (2006)
- Lin, F., Ye, L., Duffy, V., Su, C.-J.: Developing virtual environments for industrial training. Inf. Sci. (Ny) **140**(1–2), 153–170 (2002). doi:[10.1016/S0020-0255\(01\)00185-2](https://doi.org/10.1016/S0020-0255(01)00185-2)
- Shukla, C., Vazquez, M., Frank Chen, F.: Virtual manufacturing: an overview. Comput. Ind. Eng. **31**(1–2), 79–82 (1996). doi:[10.1016/0360-8352\(96\)00083-6](https://doi.org/10.1016/0360-8352(96)00083-6)
- Burdea, G., Coiffet, P.: Virtual Reality Technology, 2nd edn. Wiley-Interscience, New York (2003)
- Duguleana, M., Barbuceanu, F.G., Mogan, G.: Virtual and mixed reality—new trends. In: Shumaker, R. (ed.) Virtual and mixed reality—new trends. Springer, Berlin, pp. 164–173 (2011). doi:[10.1007/978-3-642-22021-0_19](https://doi.org/10.1007/978-3-642-22021-0_19)
- Zhu, J., Ong, S.K., Nee, A.Y.C.: A context-aware augmented reality system to assist the maintenance operators. Int. J. Interact. Des. Manuf. (2014). doi:[10.1007/s12008-013-0199-7](https://doi.org/10.1007/s12008-013-0199-7)
- Nathanael, D., Vosniakos, G.-C., Mosialos, S.: Cognitive task analysis for virtual reality training. In: Proceedings of the 28th Annual European Conference on Cognitive Ergonomics—ECCE '10. ACM Press, New York, p. 241 (2010). doi:[10.1145/1962300.1962350](https://doi.org/10.1145/1962300.1962350)
- Burkhardt, J.-M., Bardy, B., Lourdeaux, D.: Immersion, Réalisme et Présence dans la conception et l'évaluation des Environnements Virtuels. Psychol. Française. **48**(2), 35–42 (2003)
- Pook, P.K., Ballard, D.H.: Deictic human/robot interaction. Rob. Auton. Syst. **18**, 259–269 (1996). doi:[10.1016/0921-8890\(95\)00080-1](https://doi.org/10.1016/0921-8890(95)00080-1)
- Morioka, M., Sakakibara, S.: A new cell production assembly system with human–robot cooperation. CIRP Ann. Manuf. Technol. **59**, 9–12 (2010). doi:[10.1016/j.cirp.2010.03.044](https://doi.org/10.1016/j.cirp.2010.03.044)
- Krüger, J., Lien, T.K., Verl, A.: Cooperation of human and machines in assembly lines. CIRP Ann. Manuf. Technol. **58**, 628–646 (2009). doi:[10.1016/j.cirp.2009.09.009](https://doi.org/10.1016/j.cirp.2009.09.009)
- Giesler, B., Steinhaus, P., Walther, M., Dillmann, R.: Sharing skills: using augmented reality for human–robot collaboration. In: Woods, A.J., Merritt, J.O., Benton, S.A., Bolas, M.T. (eds.) Proc. SPIE 5291, Stereoscopic Displays and Virtual Reality Systems XI, pp. 446–453 (2004). doi:[10.1117/12.526689](https://doi.org/10.1117/12.526689)
- Corrales, Ja., Candelas, Fa., Torres, F.: Safe human–robot interaction based on dynamic sphere-swept line bounding volumes. Robot. Comput. Integr. Manuf. **27**(1), 177–185 (2011). doi:[10.1016/j.rcim.2010.07.005](https://doi.org/10.1016/j.rcim.2010.07.005)
- Arai, T., Kato, R., Fujita, M.: Assessment of operator stress induced by robot collaboration in assembly. CIRP Ann. Manuf. Technol. **59**(1), 5–8 (2010). doi:[10.1016/j.cirp.2010.03.043](https://doi.org/10.1016/j.cirp.2010.03.043)
- Charoenseang, S., Tonggoed, T.: Human–robot collaboration with augmented reality. In: Stephanidis, C. (ed.) HCI International 2011—Posters' Extended Abstracts. Springer, Berlin, pp. 93–97 (2011). doi:[10.1007/978-3-642-22095-1_19](https://doi.org/10.1007/978-3-642-22095-1_19)
- Fang, H.C., Ong, S.K., Nee, A.Y.C.: A novel augmented reality-based interface for robot path planning. Int. J. Interact. Des. Manuf. **8**(1), 33–42 (2014). doi:[10.1007/s12008-013-0191-2](https://doi.org/10.1007/s12008-013-0191-2)
- Du, G., Zhang, P.: Markerless human–robot interface for dual robot manipulators using Kinect sensor. Robot. Comput. Integr. Manuf. **30**(2), 150–159 (2014). doi:[10.1016/j.rcim.2013.09.003](https://doi.org/10.1016/j.rcim.2013.09.003)
- Wang, L., Schmidt, B., Nee, A.Y.C.: Vision-guided active collision avoidance for human–robot collaborations. Manuf. Lett. **1**(1), 5–8 (2013). doi:[10.1016/j.mfglet.2013.08.001](https://doi.org/10.1016/j.mfglet.2013.08.001)
- Qiu, S., Fan, X., Wu, D., He, Q., Zhou, D.: Virtual human modeling for interactive assembly and disassembly operation in virtual reality environment. Int. J. Adv. Manuf. Technol. **69**(9–12), 2355–2372 (2013). doi:[10.1007/s00170-013-5207-3](https://doi.org/10.1007/s00170-013-5207-3)
- Zaeh, M., Roesel, W.: Safety aspects in a human–robot interaction scenario: a human worker is co-operating with an industrial robot. In: Kim, J.-H., et al. (eds.) Progress in robotics. Springer, Berlin, pp. 53–62 (2009). doi:[10.1007/978-3-642-03986-7_7](https://doi.org/10.1007/978-3-642-03986-7_7)
- Neuhoefer, J.A., Bernhard, K., Schlick, C.M.: Embedded augmented reality training system for dynamic human–robot cooperation. NATO RTO-MP-HFM-169. RTO Human Factors and

- Medicine Panel (HFM) Workshop, RDECOM, pp. 1–14, Orlando, Florida, USA (2009)
- 34. Weistroffer, V., Paljic, A., Callebert, L., Fuchs, P.: A methodology to assess the acceptability of human-robot collaboration using virtual reality. In: Proceedings of the 19th ACM Symposium on Virtual Reality Software and Technology—VRST '13. ACM Press, New York, p. 39 (2013). doi:[10.1145/2503713.2503726](https://doi.org/10.1145/2503713.2503726)
 - 35. Matsas, E., Batras, D., Vosniakos, G.: Beware of the robot: a highly interactive and immersive virtual reality training application in robotic manufacturing systems. In: Emmanouilidis, C., Taisch, M., Kiritis, D. (eds.) Advances in Production Management Systems. Competitive Manufacturing for Innovative Products and Services. Springer, Berlin, pp. 606–613 (2013). doi:[10.1007/978-3-642-40352-1_76](https://doi.org/10.1007/978-3-642-40352-1_76)
 - 36. Bowman, D.A., Kruijff, E., LaViola, J.J., Poupyrev, I.: 3D User Interfaces: Theory and Practice. Addison Wesley Longman Publishing Co., Inc, Redwood City (2004)
 - 37. Evolver 3D digital avatar website. www.evolver.com. Accessed 23 January 2012
 - 38. Usoh, M., Arthur, K., Whitton, M.C., Bastos, R., Steed, A., Slater, M., Brooks, F.P.: Walking, walking-in-place, flying, in virtual environments. In: Proc. 26th Annu. Conf. Comput. Graph. Interact. Tech.—SIGGRAPH '99, pp. 359–364 (1999). doi:[10.1145/311535.311589](https://doi.org/10.1145/311535.311589)
 - 39. Unity online manual. <http://goo.gl/kH1Hv8>. Accessed 25 July 2014
 - 40. Miles, R.: Start here! Learn microsoft kinect API. O'Reilly Media Inc, Sebastopol (2012)
 - 41. Vora, J., Nair, S., Gramopadhye, A.K., Duchowski, A.T., Melloy, B.J., Kanki, B.: Using virtual reality technology for aircraft visual inspection training: presence and comparison studies. *Appl. Ergon.* **33**(6), 559–70 (2002). doi:[10.1145/311535.311589](https://doi.org/10.1145/311535.311589)
 - 42. Prothero, J.D., Parker, D.E., Furness III, T.A., Wells, M.J.: Towards a robust, quantitative measure for presence. In: Garland, D.J., Endsley, M.R. (eds.) Conference on Experimental Analysis and Measurement of Situational Awareness. Embry-Riddle Aeronautical University Press, Daytona Beach, pp. 359–366 (1995)
 - 43. Schuemie, M.J., van der Straaten, P., Krijn, M., van der Mast, C.A.: Research on presence in virtual reality: a survey. *Cyberpsychol. Behav.* **4**(2), 183–201 (2001). doi:[10.1089/109493101300117884](https://doi.org/10.1089/109493101300117884)
 - 44. Burkhardt, J.-M., Lourdeaux, D., Fuchs, P.: Conception d'un système de RV pour la formation des agents de conduite aux opérations en milieu ferroviaire. In: Richards, A., Grumbach, N. (eds.) Colloque Réalité Virtuelle et Cognition, pp. 123–132. ENST Paris, France (1999)