Chapter 10
Enhancing User Role in Augmented Reality Interactive Simulations

Pier Paolo Valentini

1 Introduction

Scientific literature reports an increasing interest for the development of applications of augmented reality (AR) in many different fields [1–3]. The AR has been used in entertainment [4–8], education [9–12], medicine [13–15], military field [16, 17], implant and components maintenance [18, 19], robotics [20], engineering [21–28] and archeology [29, 30]. Some recent developments about mobile augmented reality applications have been discussed in Chaps. 6 and 7. The most of all these applications deals with the merging in the real world of objects, scenes and animations which have been modeled and simulated outside the system. It means that the user perceives a real scene augmented with pre-computed objects. For these reasons, his interaction with the augmented scene is often limited to visual and acoustic exploration.

In 1999 the International Standard Organization (ISO) provided a definition of an *interactive system* as: "An interactive system is a combination of hardware and software components that receive input from, and communicate output to, a human user in order to support his or her performance of a task". The recent improvements of both hardware and software performances fuelled the development of innovative methodologies in order to increase of the interaction between the user and the scene [31, 32]. The purpose of these enhancements is to change the user role from spectator to actor. The main idea to achieve this objective is to use innovative approaches for going beyond a mere visual or acoustical experience of pre-computed contents, including the capability of real-time modifying and updating the contents of the scene and the two-ways interaction.

P.P. Valentini (✉)
Department of Industrial Engineering, University of Rome "Tor Vergata",
via del Politecnico 1, 00133 Rome, Italy
e-mail: valentini@ing.uniroma2.it

W. Huang et al. (eds.), *Human Factors in Augmented Reality Environments*,
DOI 10.1007/978-1-4614-4205-9_10, © Springer Science+Business Media New York 2012

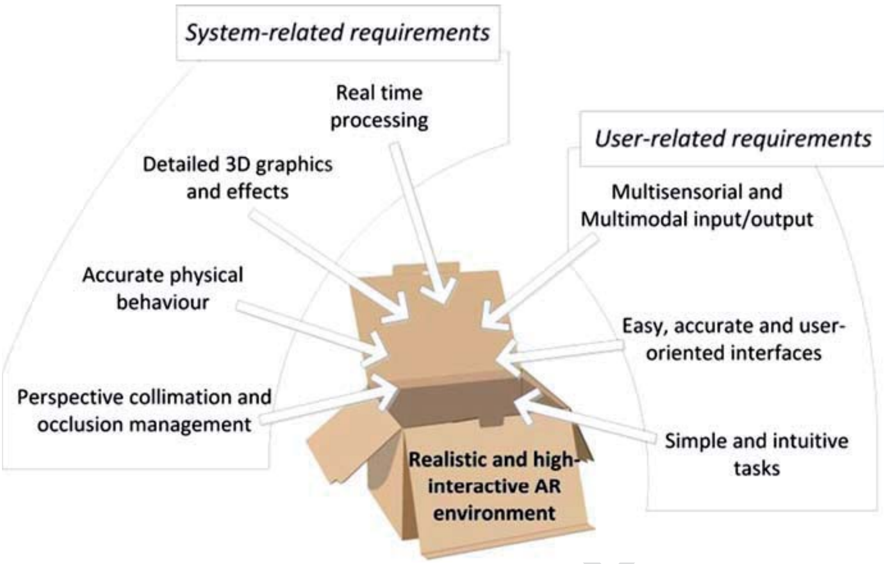

Fig. 10.1 System-related and user-related requirements for the implementation of an high interactive and realistic AR environment

Generally speaking, an augmented environment can be implemented with different levels of interaction. Interaction concerns with users tasks that can be classified according to Gabbard [33] and Esposito [34] that organized them in navigation, object selection and object manipulation, modification and querying.

An high interactive and realistic augmented reality environment needs both system-related and user-related requirements to be successfully implemented (see Fig. 10.1) [35, 36]. System-related requirements are concerned with the architecture and implementation of the processing engine (hardware and software). The user-related aspects are concerned with the way the user interact (input and output) with the environment, taking into account cognitive aspects as discussed in the Chap. 5.

In a first and basic level of interaction, the user can only reviewed pre-computed virtual contents. Following this approach, animations and graphics are prepared outside from the system and they are projected to the user in the right moment and context. For the superimposed geometries, the level of details of the augmented scene has to be very realistic and the registration between real world and virtual contents has to be accurate in order to give the illusion of a unique real world. On the other hands, textual information has to be clearly visible and readable in the scene.

An intermediate level of interaction is concerned with the possibility of relating to virtual objects and information in the scene. With this type of integration, the user is active in the scene and can change the augmented contents by picking, pushing and moving objects and controlling the provided information. The interaction is carried out with advanced input/output devices involving different sensorial channels (sight, hear, touch, etc.) in an integrated way [37]. In particular, in order to interacts with digital information through the physical environment, the system can be provided of tangible user interfaces (TUIs) [38–42]. The TUIs are more suitable than the graphic user interfaces (GUIs) to work as communicators between the user and the augmented system because they are based on physical entities that can be grabbed, moved, pushed, etc.

With an higher level of interaction, the user can modify the contents of the scene and the virtual objects in the environment behave according to realistic physics laws (dynamic simulation, deformation, etc.). In general, the interaction can be provided by specific TUIs whose design and features are suitable for an enhanced communication with the scene.

The highest level of interaction includes the reaction of the virtual objects on the user (action–reaction, force feedback, etc.) as well. In this case, the TUIs have to be able to produce sensorial feedback and their characteristic is a two-way communication (scene ↔ user).

The design and optimization of the tangible user interfaces involve an accurate attention to the related human factors and communication requirements. Human factors are concerned with anything that affects the performance of system operators whether hardware, software, or liveware [43, 44]. They include the study and application of principles of ergonomic design to equipment and operating procedures and in the scientific selection and training of operators.

On the one hand, the interfaces have to be able to track the user in the scene with adequate precision and robustness and acquire his intent. On the other hand, they have to be light and small enough to be minimally invasive and be used without difficulties in order to achieve the best possible performance within machine design limitations. A user interface designer is challenged by choosing the most appropriate way of acquiring and presenting information by adequate media and modalities.

With reference to Fig. 10.2, the standard implementation of an interactive augmented reality can be described as follows. First of all, an image stream of the real world has to be acquired. One or two RGB camera(s) are used for acquiring a mono or stereo vision of the scene, respectively. Then, the user is able to interact with communication devices in order to participate in the scene. This role is played by tangible or multimodal interfaces which can be different depending on the type of implementation. Their design has to take into account the specific human factors and simulated tasks. A device (external monitor or head mounted display) has to be also present in order to ensure the portable projection to the user of the augmented scene.

The pieces of information coming from the video acquisition and the user interface have to be processed in order to estimate the perspective transformation of the camera point of view, interpret the intent of the user and compute all the virtual objects to be added. At the end of the computation an augmented video stream is rendered taking into account also special effects as occlusions, congruent illumination, etc. and projected back to the user. In the applications which provide the two-way interaction with the user, a feedback has to be also sent back to the user via the interface devices.

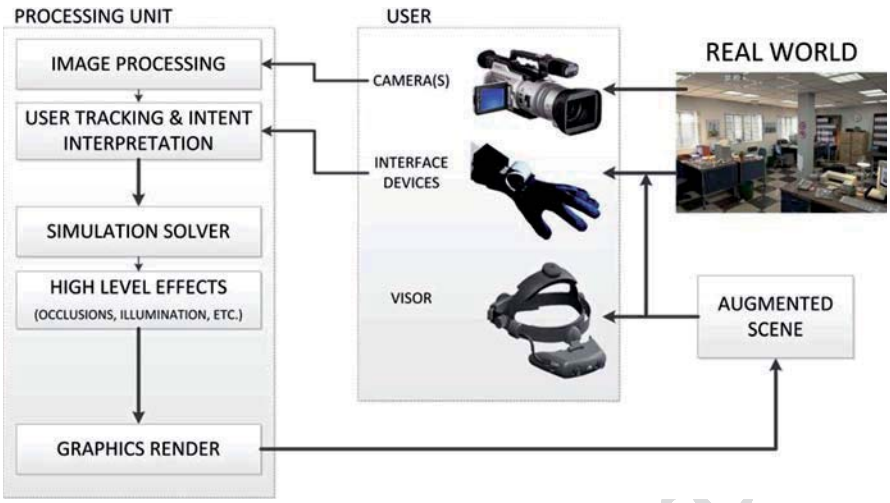

Fig. 10.2 Processing scheme for a generic augmented reality interactive implementation

Starting from this schematic representation, three main requirements for achieving an interactive augmented reality simulation can be considered.

The first one is the *realism*. The real scene and the virtual objects have to be properly integrated giving the illusion to the user to a mere real world. This means that the graphics of the objects and their illumination has to be detailed. Moreover, the AR system has to be able to manage occlusions between real objects and virtual ones, in order to avoid the perception of simple superimposition. A scene which is not able to include occlusion may produce unrealistic feeling to the user and vanish all the efforts toward the building of a realistic environment. The physical behavior of virtual objects is another important feature to achieve realism in the scene. For this purpose, the movement of all the virtual objects has to be consistent to physical laws. It means that objects cannot interpenetrate but collide, are subjected to gravity force, etc. From the user's point of view, all these features are important to increase the feeling at ease in the scene and perceive a familiar and harmonized world as an unique real environment. The presence of well-designed TUIs may surely improve this feeling.

The second requirement for an interactive AR simulation is about the *real-time processing* of the scene. It means that all the computations (image processing, user tracking, intent interpretation, physical behavior and scene updating and rendering) have to be performed in real-time (or better synchronously to the scene acquisition) in order to achieve fluidity and enhancing the illusion of a natural scene. This specification requires the development of specific simulation strategies and the use of dedicated solver and processor. The most challenging implementation is about the simulation of physical behavior of the environment including gravity, impenetrability, contact and impact dynamics, etc. The current level of hardware performances allows the use of standard computer architectures for achieving this result.

The third requirement for an interactive AR simulation is about the implementation of adequate *interaction devices and methodologies*. In order to interact with the scene, the user has to communicate to the AR system. From this point of view, the TUIs can be considered as a fundamental requirement in order to implement an interactive augmented reality environment. On the other hand, the only TUIs are not sufficient to ensure interactivity, but specific methodologies for interpreting the user's intent and his relationship with the augmented environment have to be studied and implemented. These devices and methodologies have to be integrated to the computational routines for simulating a congruent behavior of the overall system. Scientific literature reports several contributions dealing with possible methodologies achieving this interaction which are discussed in the next section of the chapter.

The chapter is organized as follows. In the first part a brief overview of the state-of-the-art methodologies for achieving interactive simulation in augmented reality environment is presented, focusing to the system-related and user-related aspects. In a second part the emerging concept of natural interface in augmented reality is introduced and discussed. In the last part of the chapter some details of implementation and examples are presented and discussed.

2 User Interaction in Augmented Reality Scenarios

Among the requirements to achieve an interactive AR simulation, the most important user-related issue is concerned with the development of devices and methodologies for achieving a robust, simple and comprehensive interface. In the very low level augmented reality implementations, the interaction is limited to graphics and (in some cases) to acoustics outputs. In basic implementations, the interaction is extended by using the mouse and the keyboard as in a standard computer application. This arrangement has the advantage that the user is already familiar to the devices and he does not need training or particular skills to use the interfaces. On the other hand, the interaction is limited to very simple operations (2D or 3D pointing and clicking).

In the intermediate-level implementations the communication between the user and the scene can be achieved using patterned makers. They are used for both computing the perspective transformation between the camera and the real world and for transferring information from the user to the scene. They are considered as communicators and the interaction is based on the computation of the their relative position and attitude with respect to the camera and the other markers reference frames. They can be rigidly mounted on real objects in order to build tangible user interfaces with 6° of freedom (three translations and three rotations). Following this approach, the advantage is that the image processing for marker detection in the acquired video stream is performed only once but it is useful for both perspective collimation and user tracking. The disadvantages of these methodologies are mainly two. First of all, their precision is low, because the position and attitude of the markers are computed by the processing of standard resolution images using segmentation and

Fig. 10.3 Marker-based tracking for implementing an interactive AR procedure for supporting cable harnessing

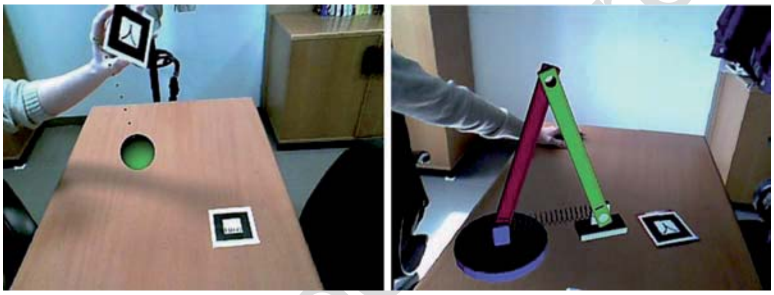

Fig. 10.4 Marker-based tracking for implementing interactive AR simulation of dynamic systems: launching a bouncing ball (on the *left*) and moving a slider-crank mechanism with a spring damper element

correlation algorithms. Secondly, in order to be tracked, the markers need to be always visible to the camera and this limits the capture volume and suffers occlusion phenomena. Figure 10.3 shows and example of this tracking methodology used for implementing an interactive procedure for supporting cable harnessing in augmented reality [24]. In this application five patterned markers are placed on a cube at the end of a stick in order to implement a traceable pen. The pen is used to sketch and modify the path of a virtual cable interactively routed in the scene. Although only one marker is sufficient to track the position of a rigid body in space (requiring 6° of freedom), redundant markers can be used to ensure a continuous visibility and more accurate tracking.

Marker-based interaction has been used also in interactive dynamic simulations. Figure 10.4 reports two examples of this implementation. In this case the markers are directly grabbed by the user in order to interactively set and modify the initial conditions of the motion simulations [45] of a collection of rigid bodies.

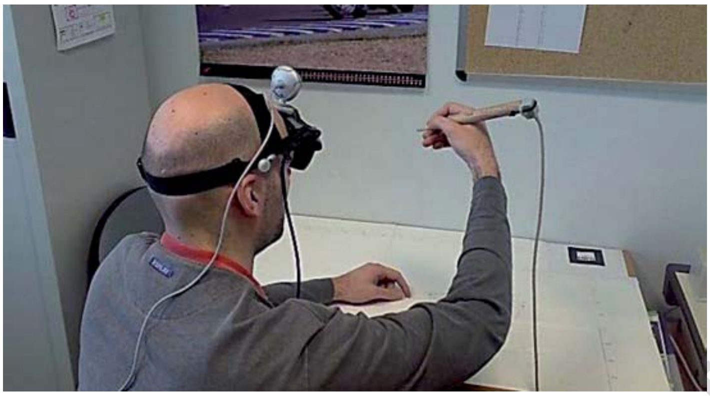

Fig. 10.5 Magnetic device-based tracking for implementing interactive AR geometric modeling environment

Other methodologies introduced the use of different sensors for tracking the position of the user in the scene and interpreting his intent. Some of them are concerned with the use of optical tracking systems [46]. These implementations make often use of reflective markers (usually spheres for improving visibility) or pattern of markers whose position in the scene can be recognized by photogrammetric analysis using multiple cameras which can be different from those used for real world acquisition and perspective collimation. Since the reflective markers can be rigidly placed on almost every object, they can be used to implement tangible user interfaces or for simple user main body parts tracking. These methodologies can be more precise than the previous ones because the optical tracking is performed using a dedicated system. On the other hand, the presence of several markers may be uncomfortable for the user and, as for the other optical systems, their precision is affected by the resolution of the cameras and highly suffers occlusion phenomena.

Other acquisition methodologies are based on the use of magnetic trackers [47]. In the common embodiments, these devices are comprised of an emitter and a receiver. The emitter generates a magnetic field which is captured by the receiver. The changing of the acquired signal is converted to information about the position and attitude of the receiver. Due to its small size, the receiver can be easy put on by the user or attached to a graspable stick in order to perform user tracking or build tangible user interfaces. In general, magnetic trackers are more precise than the optical ones, but their performance is tremendously influenced by electromagnetic perturbations caused by metallic parts in the scene and the capture volume is dependent on the strength of the magnetic field generated by the emitter. Figure 10.5 shows an example of the use of a magnetic tracking system for implementing a augmented reality system for interactive sketching and modeling [22]. In this application the magnetic sensor (Flock of Birds by Acension) is placed on at the end of a stick in order to implement a traceable pen and help the user in interactive operations.

In order to achieve more precise and robust tracking, mechanical (or mechatronic) devices can be used [48]. They commonly use a multi degree-of-freedom this figure will be printed in b/w

Fig. 10.6 Mechanical device-based sketching for implementing interactive AR reverse engineering modeling tool

this figure will be printed in b/w

Fig. 10.7 Mechanical device-based tracking for implementing interactive AR engineering simulation of motion: ten pendula simulation (on the *left*) and flexible slender beam (on the *right*)

linkage to compute the position in the space of a end-effector which can be grabbed by the user. By this way the devices can be directly considered as the tangible interface. Their precision is high (<0.2 mm), they are not affected by occlusions and perturbations, but their capture volume is still limited by the dimension of the linkage. Thanks to all these advantages, they are suitable for accurate interaction involving technical and engineering aspects (interactive sketching, reverse engineering, measurement, etc.).

Figure 10.6 shows an example of a mechanical tracker (Microscribe GX2 by RevWare) used to perform an interactive reverse engineering tool in augmented reality.

Figure 10.7 shows and another example dealing with two simulations of movement performed using the same mechanical tracker for defining the boundary conditions for both rigid [48] and deformable bodies [49] simulation.

Fig. 10.8 An example of integration between a pattern-based tracking system and a data glove for hand position and gesture acquisition in an augmented reality environment

In order to increase the sensorial feedback, haptic output can be added to the tangible user interfaces. Haptics comes from a Greek word *ηαπτεστηαι* meaning "grasping" or "the science of touch". In recent years, its meaning extended to the scientific study for applying tactile and force feedback sensations of humans into the computer-generated world.

All the above mentioned tracking systems can be used in addition to other specific devices in order to enhance the communication properties between the user and the augmented scene. One of the possible implementation is concerned with the use of data gloves. These wearable devices can be used for acquiring the hand gesture of the user, interpreting his intent of indexing, picking, etc. Since they provide only gesture assessment, they have to be used in addition to other tracking devices, as optical or magnetic systems. They have the advantage to enhance the possibility of interaction, interpreting an extended range of user's intent.

Figure 10.8 shows an example of integration between a pattern-based tracking system and a data glove. The system is able to acquire user's hand position and gesture and has been used for implementing a virtual assembly procedure in an augmented reality environment [50].

The described methodologies for enhancing the interaction between the user and the augmented scene can be compared in terms of system performance and user-related factors.

Concerning with the performance the main characteristic of the solutions are the cost, the precision, the capture volume and the suitable applications. Table 10.1

Table 10.1 Performance comparison among tracking devices for interactive augmented reality implementations

Interaction method	Capture volume	Precision	Cost	Preferred applications
Visual/acoustic only	n.a.	n.a.	Low	Entertainment; augmented navigation; augmented maintenance; military applications
Mouse and keyboard	n.a.	Very low	Low	Entertainment; gaming; augmented navigation
Optical patterned markers	Limited, suffer occlusion	Low	Very low	Design or model review; collaborative design; simple interactive simulation; development of simple tangible user interface for virtual prototyping
Optical reflective marker-based systems	Limited, suffer occlusion	Medium–low	Medium	Interactive and immersive simulations; interactive exploration of wide and complex scenarios; interactive sketching
Magnetic systems	Limited, suffer metallic objects	Medium	Medium	Interactive sketching and modeling; developing of efficient tangible user interface for virtual prototyping
Mechanical systems	Limited	High	Medium–high	Precise interactive sketching and modeling. Accurate interactive simulations, engineering applications
Mechanical with force feedback systems	Very limited	High	Medium–high	Precise interactive sketching and modeling. Accurate interactive simulations, engineering and surgery applications
Optical or magnetic + local sensor interfaces	Limited	Low to medium	Medium–high	Interactive simulation including hand gesture

Table 10.2 User related factors comparison among tracking devices for interactive augmented reality implementations

Interaction method	Wearability	User friendliness	Training	Invasiveness in the scene
Visual/acoustic only	Yes	Good	Very low; training required for stereoscopic projection	Low
Mouse and keyboard	No	Very good	No	Very low
Optical patterned markers	Yes	Good	No	Low
Optical reflective marker-based systems	Yes	Good	Low	Medium
Magnetic systems	Yes	Good	Low	Medium–low
Mechanical systems	No	Good	Medium	Medium
Mechanical with force feedback systems	No	Discrete	Medium–high	Medium
Optical or magnetic + local sensor interfaces	Yes	Good	Medium	Medium–low

reports a comparison among the different typologies of the interaction solutions. It can be noted that the optical system based on marker recognition are suitable for the developing of basic interactive augmented scenarios, especially for entertainment, gaming, design reviews, conceptual technical applications requiring low accuracy. The mechanical devices are very suitable for accurate interaction and for the implementation of precise engineering and surgery simulations. Due to their architecture, they are also suitable for the implementation of force-feedback sensors and immersive simulation.

Concerning with the user-related factors the main characteristic of the solutions are the wearability, the user friendliness, the necessity of dedicated training and the invasiveness in the scene. Table 10.2 reports a comparison among the different typologies of the interaction solutions. All the presented devices can be arranged to be worn by the user, except the mechanical trackers. In general, small markers (simple sphere of patterned ones) can be easily managed by the user and they are less invasive in the scene. On the other side, they need to be always visible to the cameras and require a little training to be properly used. Both optical, magnetic and mechanical systems allow the arrangement of tangible user interfaces similar to a pen, which enhances the friendliness and reveal to be familiar to the user.

3 The Concept of Natural Interface

According to the considerations in the review presented in the previous section, two different aspects have to be underlined. First of all, in order to interact with the scene, the environment has to include interfaces which are implemented by using devices and methodologies for tracking user's position, interpreting his intent and transferring information to the simulation engine. On the other hand, the development of such interfaces may involve the use of complex and bulky devices. In too many case, they miss the point to produce a realistic scene because they are considered external, unrealistic and cumbersome by the user.

In order to overcome these problems, the idea is to avoid the use of specific interface devices and try to track the user and interpret his intent just observing the scene as it happens in real life in interpersonal communication. By this way, the user is the interface or better he uses a natural interface which is his body posture and attitude.

3.1 Implementation Details

As introduced in the first part of the chapter, most of the augmented reality applications performs the acquisition of the real world using a single camera or, for stereoscopic projection, two cameras. The role of these devices is to produce one or two RGB image(s) that can be used for both image processing and for the definition of the background image of the final augmented scene projection.

In order to implement the concept of the natural interface, tracking the user in the scene and interpreting his intent, simple cameras are not sufficient because they produce two dimensional images only. In order to have continuous three dimensional information about the acquired scene, a compound of an RGB camera, an IR projector and a IR depth camera can be used. For the specific purposes of the study, the author has tested the Microsoft Kinect Sensor which contains such arrangement in a compact bundle. The use of the Kinect Sensor allows the synchronized acquisition of an RGB image and a depth map of the same real scene. The two streams are always collimated by the fixed location in the bundle. An RGB image is a data structure containing color information of each acquired point (pixel). A depth map is a data structure containing the distance from the sensor of each pixel along a direction perpendicular to the image plane. In order to acquire and process the data coming from the Kinect sensor the Prime Sense drivers has been used. They are suitable for C++ programming language implementation and can be freely downloaded at https://github.com/PrimeSense/Sensor.

According to its architecture, there are two data streams coming from the Kinect Sensor that have to be managed. The first one, as in a traditional augmented reality application, is the RGB video stream. Each RGB frame can be processed in order to recognize the presence of patterned markers in the scene and to compute the perspective transformations between the camera and each marker. The processing of the depth map stream allows to include several enhancements useful for increasing the realism of the augmented scene and the lever of interaction. Two are the main processes involving the depth map. The first one is concerned with the computation of the environmental mesh which is a geometrical representation of the real world three-dimensional geometry. Starting from the knowledge of the 3D coordinates of each point observed by the depth camera, it is possible to build a structured polygonal mesh describing the geometry of the surrounding environment. This mesh can be textured with the color information coming from the RGB camera in order to achieve a complete 3D reconstruction of the augmented environment.

The second important use of the depth camera stream is the possibility of tracking the users in the scene and implementing the natural interface concept in a very smart way as described in the following section.

3.2 User Tracking

The processing of the depth map information allows the real-time tracking of the user. For the tested implementation involving the Kinect Sensor, the tracking is implemented using the OpenNI programming library freely downloadable at https://github.com/OpenNI/OpenNI. The OpenNI is a collection of C++ routines for direct accessing and processing data from Kinect Sensor and includes numerical procedures for achieving a robust and precise tracking of user's body main landmarks. Although the exact implementation of these algorithms is not open access, some useful information can be extracted from the related patent application [50].

According to this approach, the tracking of the user is performed by processing the depth map in order to recognize the spatial position of the user's main body joints in the real scene. The OpenNi algorithm allows the real time recognition and tracking of the following 16 joints (see Fig. 10.9):

- Center of the head
- Center of the neck
- Right and left shoulder joints
- Right and left elbow joints
- Center of right and left hand
- Center of the chest
- Center of the abdomen
- Right and left hip joints
- Right and left knee joints
- Center of the right and left feet

The algorithm allows the tracking of several users at the same time.

The spatial position of the above mentioned 16 body joints are sufficient to interpret the pose of a human body. By this way, the intent of the user can be interpreted by comparing his pose to a collection of preset posture (Fig. 10.10). This assessment is very fast because requires the comparison of only a small set of 3D points.

The recognition of the body pose can be useful for activating commands and updating the scene contents.

The most important body joints to be tracker are the hands because they represent the main human interface to the physical (and virtual) world. According to the scientific literature and practical evidence almost all interaction methodologies are

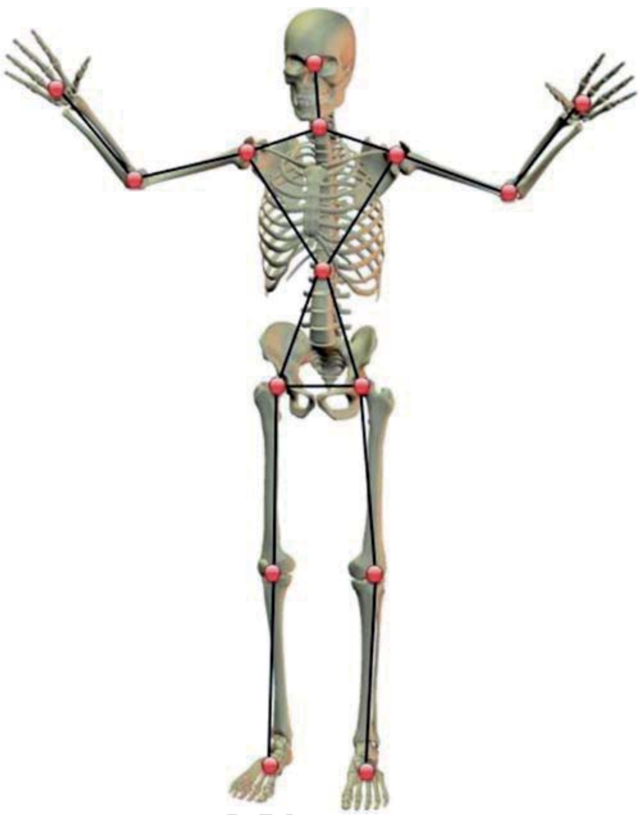

Fig. 10.9 Traceable body landmarks using numerical libraries

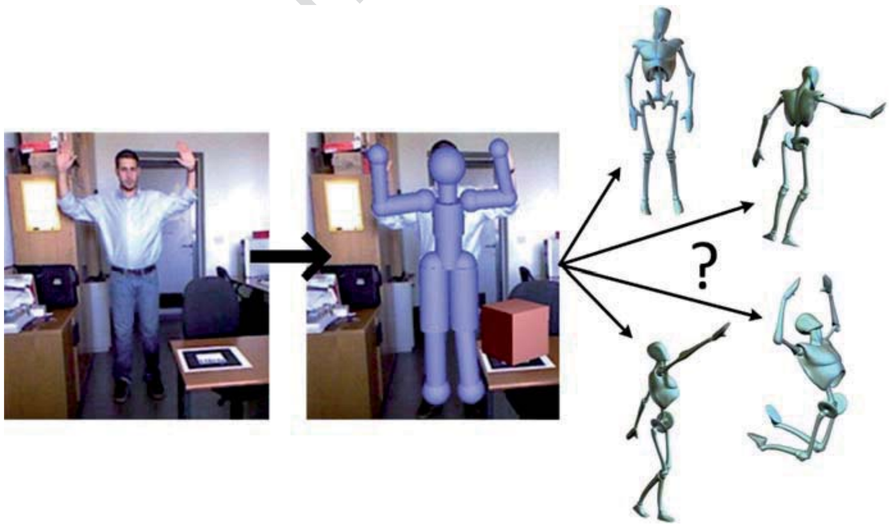

Fig. 10.10 Tracking user body and recognizing his pose

based on the tracking of the user's hands. In fact, indexing, picking, grabbing and pushing are all activities that involve the use of one or both hand. The recognition and the tracking of their position in the scene is therefore crucial.

3.3 Realism and Occlusion

As discussed in the introduction, the presence of correct occlusions between real and virtual objects in the augmented environment is a very important topic for enhancing the realism of the scene. Absent or wrong occlusion management may mine the overall quality of the environment and the user may perceive an unreal and disturbing environment.

A correct interpretation of occlusions is one of the most challenging topics in augmented reality applications [52–56]. According to some authors, the correct occlusion management is one of the most important requirement for a realistic implementation. Unfortunately, dealing with occlusions is quite complicated.

Standard augmented reality implementations usually neglect occlusions between real and virtual objects and the acquired image from the real world is considered as a simple background texture on which virtual objects and information are superimposed. On the other hand, occlusions involving only virtual objects can be easily computed by using the z-depth comparison which is a widely used technique in computer graphics. According to this approach, all the entities to be rendered are arranged in a list (called z-buffer) starting from the farthest up to the closest with respect to the point of view and along the direction normal to the image plane. Then they are rendered respecting the computed order and by this way the farther geometries are rendered after the nearer ones producing automatic occlusions.

The use of an IR projector/camera system to acquire a depth map of the environment can also help the managing of occlusion of real objects with respect to the virtual ones. As described in the subsection dealing with the implementation of the system, the Kinect Sensor produces a 3D geometric (polygonal) description of the acquired scene. Starting from this collection of data, it is possible to compute the 3D coordinates for each point of the real objects acquired by the sensor. Following a similar approach, the information coming from the depth camera can be processed in order to compute the z-coordinate (the distance from the image plane) for each pixel of the environmental mesh. The information about this mesh can be used for including the real objects in the scene in the z-depth comparison together with the other 3D virtual entities.

The processing of the environmental mesh is suitable for a real time computation and an example of application is reported in Fig. 10.11. It can be noticed that in the depicted augmented environment there are two virtual objects: a cube (placed on a real table) and a cylinder (in the right side, behind a real chair). With the proposed approach it is possible to compute the occlusion between the user body and the two objects and between the other real objects in the scene and the two virtual objects. It has to be underlined that the managing of the occlusions has some small imprecisions this figure will be printed in b/w

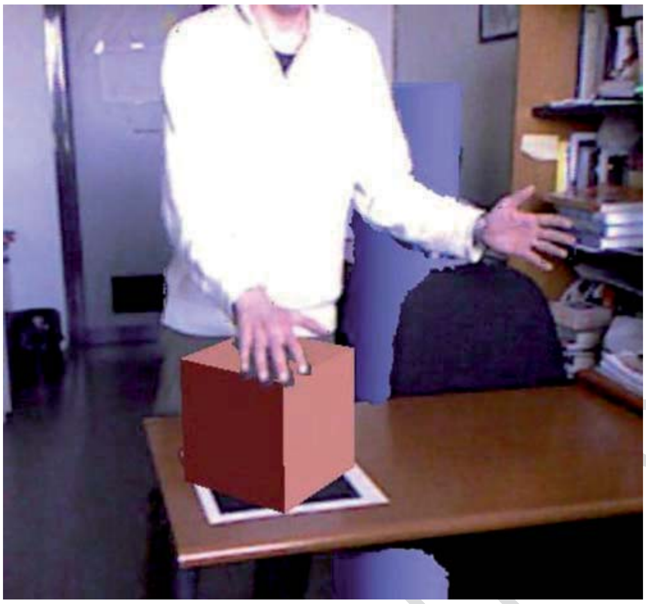

Fig. 10.11 Occlusion management using depth map acquisition and computation

(the edges of the objects are often irregular) but the detail is sufficient to enhance the realism of the environment and avoid the perception an unreal and wrong (or even impossible) scene.

4 Simulating Physical Behavior

One of the other important requirements of an interactive environment is the achieving of accurate simulation of the objects behavior according to the actual physical laws. A correct mimic of the real world is very crucial for giving to the user the illusion of a consistent scene [57]. Moreover, a correct simulation respecting physical laws can be useful for implementing not only entertainment and gaming applications, but also technical and engineering scenarios in which the interpretation of the results can be used for improving product design and related performances [45].

Many augmented reality implementations make use of animation in order to transfer information to the user by using appealing moving graphics. These animations are studied and prepared outside from the running system and then are projected in the right place and context during exploration. However, performing simulations is different from simply animating. An animation concerns with the movement of the objects according to some specific predefined schemes and sequences. By this way, the animated movement can be convenient and didactical, but can be unreal and inconsistent. This solution may be an advantage for some

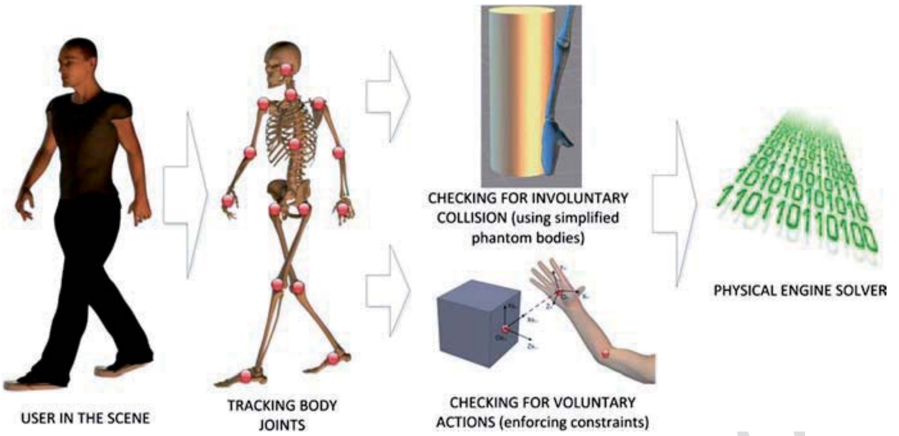

Fig. 10.12 Managing user presence for implementing accurate and realistic physical simulation

implementations, but may produce highly unreal scenarios. On the contrary, the simulation is the replication of a behavior which is consistent to the presence of physical law. A simulated virtual object behaves exactly (or quite exactly) in the same way as it would be real.

Introducing correct physical behavior in an high-interactive environment implies that the user can actively take part in the simulation. This participation has two different aspects: involuntary and voluntary ones. On the one hand, the presence of the user can affect the environment without being involved in a specific action. The collision between body limb and a virtual object is an example of this involuntary participation. On the other hand, the user can also voluntary affect the environment by picking, moving, throwing virtual objects. These two different kinds of interaction has to be taken into account in the simulation (see Fig. 10.12). And then, of course, all the virtual objects take part in the simulation and may interact among them.

The interaction between the user and the simulated environment requires the tracking of the body main joints and so it can be managed by the use of the Kinect Sensor as well. The positions of body joints are real-time computed during all the simulation. Phantom geometries (cylinders, cones and spheres) can be attached to these joints in order to check if collisions occur and manage involuntary contact between the user and the virtual objects in the scene (see Fig. 10.10). This approach can be implemented without any additional sensor to be attached to the user, respecting the purpose of a natural interface to the augmented environment.

The voluntary picking and moving of the objects can be also implemented starting from the tracking of the body main joints, but it required a more complicated approach. In particular, it is sufficient to track the position of the hands to check if the user is about to pick an object and then impose a grabbing constraint. Mathematical formulations and strategies to impose this kind of constraint goes beyond the scope of this chapter and an interested reader can find additional details in referenced papers [48, 58, 59].

The voluntary interaction can concern with both the definition of boundary conditions and initial parameters and the real-time control of the simulation.

The main difficulty in the practical implementation of the physical simulation of the virtual objects behavior is that all these computations have to be processed in real time. Three are the main problems of such processing. First of all, the scene may include many virtual objects whose dynamic behavior has to be computed, detecting and taking into account multiple collisions at the same time. Secondly, there are many different events in the simulation that require an updating of the topology of the system and a rearranging of the mathematical equations. This highly nonlinear behavior makes many standard integrators unsuitable for the purpose. Thirdly, the simulation has to be continuously performed for a long time needing robust integration and producing accurate and fluid results. For all these purposes, specific strategies have to be implemented in order to deduce and solve the equations of motion of the simulated system in a smart way.

Previous publications about the integration of dynamics simulation in augmented reality applications [48, 60–63] have revealed the interesting capability of the sequential impulse solvers. One of the most used is the *Bullet Physics Engine* which is an open source simulator with efficient real-time collision detection algorithms [64]. It is used in games, visual effects in movies and can be freely downloaded at http://bulletphysics.org/wordpress/.

The sequential impulse solver strategy allows a quick, stable and accurate simulation even in presence of all the above mentioned difficulties. According to this approach, the solution of the dynamics equations is based on the following steps. Firstly, the equations of motion are tentatively solved considering elastic and external forces but neglecting all the kinematic constraints and contact overlapping. This choice produces a solution that is only approximated. In a second step, a sequence of impulses are applied to each body in the collection in order to correct their velocities according to the limitation imposed by all the physical constraints. This second step is iterative but quite fast. It means that a series of impulse is applied to all the bodies until the constraint equations are fulfilled within a specific tolerance. Again, the detailed description of this methodology goes beyond the scope of the chapter and further details can be found in the referenced papers.

4.1 An Example of Implementation

Let us discuss some details of the implementation of the dynamic solver in the augmented reality interactive environment with an example. The scenario is about a simple interactive game in which a user can grab a ball and can throw it towards a stack of boxes. Both the ball and the boxes are virtual objects. The scene is acquired by a Kinect Sensor and processed by a DELL Precision M4400 laptop provided with an Intel Centrino 2 vPro (dual-core processor), 4 Gb RAM and a NVidia Quadro FX770M graphic card. No additional sensors have been used for tracking

Fig. 10.13 Six snapshots of the discussed example

the user in the scene. Figure 10.13 shows a sequence of six snapshots taken during the simulation.

The grabbing of the ball and the throwing are both voluntary actions. The user can freely move in the scene and the system tracks his body in real-time. When the user puts his hand near the ball the system recognizes the action which has to be confirmed by the user (snapshot A). From this moment, the movement of the ball is constrained to that of the hand. It means that the position, velocity and acceleration of the ball are dependent from those of the hand. Then, he can decide to remove the connection releasing the ball which is thrown with specific kinematic initial conditions as position and velocity (snapshot B). From this moment the ball moves subjected only to gravity till it hits the stack of boxes (snapshot C). All the collisions are managed by physics engine and the equations of motion are solved by means of the sequential impulse strategy (using Bullet Physics libraries) and continues throughout the simulation (snapshots D, E and F). Occlusions between real and virtual objects are detected during all the simulation (i.e. between the ball and the user's hand) and managed accordingly (see snapshots A, D and E). All the acquisition, computation and rendering are performed in real-time achieving a continuous and fluid output stream.

5 Discussion and Conclusion

Focusing on the user role is fundamental in order to develop augmented reality interactive environments. The user has to be considered the starting point for developing methodologies and devices. From this point of view, the interface design has to take into account both user factors as ergonomics, invasiveness, friendliness and system factors as accuracy, precision, robustness and reliability. This chapter focused on the emerging concept of natural interface. The idea is to avoid the use of additional sensors in order to implement the interface between the user and the environment. The user body is the interface as it happens in everyday communication and an intent can be expressed using posture and gesture. This approach requires the real-time tracking of the user body main joints and the interpretation of his pose. A possible solution involves the use of an infrared projector and an infrared camera which are able to produce a three dimensional depth map of the acquired scene. By the interpretation of this map, it is possible to recognize the user body limbs and track their joint positions. By this way, the tracking of the user is performed without any sensor to be mounted on the user body (like markers or magnetic transducers) and without the use of external devices.

The natural interface methodology can be integrated in complex systems including occlusion management, collision detection and physical behavior simulation. These complex scenarios enhance the realism of the scene and make the user perceived the augmented environment very close to the real one.

The discussed example and many others developed for testing different aspects of the whole methodology have underlined that the natural interface approach is suitable for real-time processing also using standard computer architectures and simulating complex scenarios. The achieved results are very promising, the tracking of the user body is very robust and the physical simulation is accurate.

Considering system-related aspects, in comparison to other standard methodologies the natural interface has a greater capture volume but a lower precision. The capture volume is influenced by the IR projector and camera properties. The precision is influenced by the approximated algorithm for estimating the position of the body main joints. On the other hand, the cost is very low.

Considering user-related aspects, the natural interface has many advantages because the sensor can be quite easily worn, it is simple to use, it requires a very short training and it is not invasive in the scene.

The methodology has also been tested on a set of 30 users of different gender and age and without any experience of augmented or virtual environments. All the 30 testers reported a very interesting experience and were surprised by the easiness in achieving the interaction with the system.

The achievement of both realistic scenario and minimally invasive interaction allow the use of this methodology for many different purposes. Moreover, the natural interface requires a reduced time for training in order to be confident with the augmented environment. The combination between augmented visualization, high interaction and simulation can be a solid base for developing specific computer-aided tools for supporting different activities from simple entertainment and gaming to technical implementations in medicine, architecture and engineering.

References

1. R.T. Azuma, Y. Baillot R. Behringer, S. Feiner, S. Julier and B. MacIntyre. Recent advances in augmented reality. IEEE Computer Graphics, vol. 21(6), pp. 34–47, 2001.
2. J. Carmigniani, B. Furht, M. Anisetti, P. Ceravolo, E. Damiani, M. Ivkovic. Augmented reality technologies, systems and applications. Multimedia Tools and Applications, vol. 51, pp. 341–377, 2011.
3. O. Bimber, R. Raskar. Spatial Augmented Reality: Merging Real and Virtual Worlds. A K Peters, Ltd, 2005.
4. J.L. Lugrin, M. Cavazza. Towards AR Game Engines. Proceedings of SEARIS - 3rd Workshop on Software Engineering and Architecture of Realtime Interactive Systems, Waltham, MA, USA, 2010.
5. L.T. De Paolis, G. Aloisio, M. Pulimeno. A Simulation of a Billiards Game Based on Marker Detection. Proceedings of the 2nd international Conferences on Advances in Computer-Human interactions, IEEE Computer Society, Washington, DC, USA, 2009.
6. O. Oda, L.J. Lister, S. White, S. Feiner. Developing an augmented reality racing game. Proceedings of the 2nd international Conference on Intelligent Technologies For interactive Entertainment, Cancun, Mexico, 2008.
7. C. Matysczok, R. Radkowski, J. Berssenbruegge. ARBowling: Immersive and Realistic Game Play in Real Environments Using Augmented Reality. of the 3rd IEEE/ACM international Symposium on Mixed and Augmented Reality, IEEE Computer Society, Washington, DC, USA, 2004.
8. F. Liarokapis. An augmented reality interface for visualizing and interacting with virtual content. Virtual Reality, vol. 11, pp. 23–43, 2007.
9. P. Buchanan, H. Seichter, M. Billinghurst M, R. Grasset. Augmented reality and rigid body simulation for edutainment: the interesting mechanism - an AR puzzle to teach Newton physics. Proceedings of the 2008 international Conference on Advances in Computer Entertainment Technology, Yokohama, Japan, 2008.
10. S. Irawati, S. Hong, J Kim, H Ko. 3D edutainment environment: learning physics through VR/AR experiences. Proceedings of the International Conference on Advances in Computer Entertainment Technology, New York, NY, USA, 2008.
11. F. Liarokapis, P. Petridis, P.F. Lister, M. White. Multimedia augmented reality interface for E-learning (MARIE). World Transactions on Engineering and Technology Education, vol. 1(2), pp. 173–176, 2002.
12. Z. Pan, A.D. Cheok, H. Yang, J. Zhu, J. Shi. Virtual reality and mixed reality for virtual learning environments. Computers & Graphics, vol. 30, pp. 20–28, 2006.

13. E. Samset, A. Talsma, O. Elle, L. Aurdal, H. Hirschberg, E. Fosse. A virtual environment for surgical image guidance in intraoperative MRI. Computer Aided Surgery, vol. 7(4), pp. 187–196, 2002.
14. P. Edwards, D. Hawkes, D. Hill, D. Jewell, R. Spink, A. Strong, M. Gleeson. Augmented reality in the stereo operating microscope for otolaryngology and neurological guidance. Computer Assisted Surgery, vol. 1(3), pp. 172–178, 1995.
15. M. Rosenthal, A. State, J. Lee, G. Hirota, J. Ackerman, K. Keller, E.D. Pisano, M. Jiroutek, K. Muller, H. Fuchs. Augmented reality guidance for needle biopsies: A randomized, controlled trial in phantoms. Lecture Notes in Computer Science: Medical Image Computing and Computer- Assisted Interventions (MICCAI), vol. 2208, pp. 240–248, 2001.
16. M.A. Livingston, L.J. Rosenblum, S.J. Julier, D. Brown, Y. Baillot, J.E. Swan, J.L. Gabbard, D. Hix. An augmented reality system for military operations in urban terrain. Proceedings of Interservice/Industry Training, Simulation, and Education Conference, Orlando, FL, USA, 2002.
17. M.A. Livingston, J.E. Swan, J.L. Gabbard, T.H. Höllerer, D. Hix, S.J. Julier, Y. Baillot, D. Brown. Resolving multiple occluded layers in augmented reality. Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR2003), Tokyo, Japan, 2003.
18. G. Klinker, O. Creighton et al. Augmented maintenance of powerplants: A prototyping case study of a mobile AR system. Proceedings of ISAR '01 - The Second IEEE and ACM International Symposium on Augmented Reality, New York, NY, USA, 2001.
19. B. Schwald, J. Figue, E. Chauvineau et al. STARMATE: Using Augmented Reality Technology for Computer Guided Maintenance of Complex Mechanical Elements. E-work and Ecommerce, vol. 1, pp. 196–202, 2001.
20. M. Stilman, P. Michel, J. Chestnutt, K. Nishiwaki, S. Kagami, J.J. Kuffner. Augmented Reality for Robot Development and Experimentation. Tech. Report CMU-RI-TR-05-55, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA, 2005.
21. S.K. Ong, Y. Pang, A.Y.C. Nee. Augmented Reality Aided Assembly Design and Planning. Annals of the CIRP, vol. 56(1), pp. 49–52, 2007.
22. P.P. Valentini, E. Pezzuti, D. Gattamelata. Virtual engineering in Augmented Reality. Chapter in Computer Animation, Nova Science Publisher Inc., New York, 2010.
23. Y. Pang, A.Y.C. Nee, S.K. Ong, M.L. Yuan, K. Youcef-Toumi. Assembly Feature Design in an Augmented Reality Environment. Assembly Automation, vol. 26(1), pp. 34–43, 2006.
24. P.P. Valentini. Interactive cable harnessing in Augmented Reality. International Journal on Interactive Design and Manufacturing, vol. 5, pp. 45–53, 2011.
25. P.P. Valentini. Reverse Engineering Tools in Augmented Reality to Support Acquisition, Processing and Interactive Study of Cultural and Archaeological Heritage. Chapter in Virtual Reality, Nova Science Publisher Inc., New York, 2011.
26. W. Dangelmaier, M. Fischer, J. Gausemeier, M. Grafe, C. Matysczok, B. Mueck. Virtual and augmented reality support for discrete manufacturing system simulation. Computers in Industry, vol. 56, pp. 371–383, 2005.
27. S.K. Ong, A.Y.C. Nee. Virtual and Augmented Reality Applications in Manufacturing. Springer, London, United Kingdom, 2004.
28. R. Reif, D. Walch. Augmented & Virtual Reality applications in the field of logistics. Visual Computing, vol. 24, pp. 987–994, 2008.
29. S. Kirchner, P. Jablonka. Virtual Archaeology - VR based knowledge management and marketing in archaeology first results - nexts steps. Proceedings of the 2001 Conference on Virtual Reality, Archeology and Cultural Heritage Glyfada, Greece, 2001.
30. Liarokapis F, Sylaiou S et al. (2004) An interactive visualization interface for virtual museum. Proceedings of the 5th international symposium on Virtual Reality, Archaeology- Cultural Heritage, Brussels and Oudenaarde, Belgium
31. J. Vallino. Interactive augmented reality. PhD thesis, Department of Computer Science, University of Rochester, USA, 1998.
32. S. Kim, A.K. Dey. AR interfacing with prototype 3D applications based on user-centered interactivity. Computer-Aided Design, vol. 42, pp. 373–386, 2010.

33. J. Gabbard, D. Hix. A Taxonomy of Usability Characteristics in Virtual Environments, Virginia Polytechnic Institute and State University, VA, USA, 1997.
34. C. Esposito. User interfaces for virtual reality systems. Human Factors in Computing Systems, CHI96 Conference Tutorial Notes, 1996.
35. F. Merienne. Human factors consideration in the interaction process with virtual environment. International Journal of Interactive Design and Manufacturing, vol. 4(2), pp. 83–86, 2010.
36. D. Bowmann, E. Kruijf, J. La Viola, I. Poupyrev. 3D User Interfaces: Theory and Practice. Addison-Wesley, 2005.
37. A. Jaimes, N. Sebe. Multimodal human-computer interaction: A survey. Computer vision and image understanding, vol. 108(1–2), pp. 116–134, 2007.
38. K. Fishkin. A taxonomy for and analysis of tangible interfaces. Personal Ubiquitous Computing, vol. 8(5), pp. 347–358, 2004.
39. B. Ullmer, H. Ishii. Emerging frameworks for tangible user interfaces. IBM Systems Journal, vol. 39(3–4), pp. 915–931, 2000.
40. M. Fiorentino, G. Monno, A.E. Uva. Tangible Interfaces for Augmented Engineering Data Management. Chapter in Augmented Reality, Intech, Croatia, 2010.
41. M. Akamatsu, I.S. MacKenzie, T. Hasbrouc. A Comparison of Tactile, Auditory, and Visual Feedback in a Pointing Task using a Mouse-type Device. Ergonomics, vol. 38, pp. 816–827, 1995.
42. H. Slay, B. Thomas, R. Vernik. Tangible user interaction using augmented reality. Proceedings of the 3rd Australasian Conference on User interfaces, Melbourne, Victoria, Australia, 2002.
43. D. Bowmann, E. Kruijf, J. La Viola, I. Poupyrev. An Introduction to 3-D User Interface Design. Presence: Teleoperators and Virtual Environments, vol. 10(1), pp. 96–108, 2001.
44. G.A. Lee, C. Nelles, M. Billinghurst, G.J. Kim. Immersive Authoring of Tangible Augmented Reality Applications. Proceedings of the 3rd IEEE/ACM international Symposium on Mixed and Augmented Reality, IEEE Computer Society, Washington, DC, USA, 2004.
45. P.P. Valentini, E. Pezzuti. Interactive Multibody Simulation in Augmented Reality. Journal of Theoretical and Applied Mechanics, vol. 48(3), pp. 733–750, 2010.
46. G. Guerra-Filho. Optical motion capture: theory and implementation. Journal of Theoretical and Applied Informatics, vol. 12(2), pp. 61–89, 2005.
47. D. Roetenberg. Inertial and magnetic sensing of human motion, Ph.D. dissertation, University of Twente, Twente, The Netherlands, 2006.
48. P.P. Valentini, L. Mariti. Improving interactive multibody simulation using precise tracking and sequential impulse solver. Proceedings of ECCOMAS Multibody Dynamics Congress, Bruxelles, Belgium, 2011.
49. P.P. Valentini, E. Pezzuti. Dynamic Splines for interactive simulation of elastic beams in Augmented Reality, Proc. of IMPROVE 2011 International congress, Venice, Italy, 2011.
50. P.P. Valentini. Interactive virtual assembling in augmented reality. International Journal on Interactive Design and Manufacturing, vol. 3, pp. 109–119, 2009.
51. T. Berliner et al. Modelling of Humanoid forms from depth maps. U.S. Patent Application Publication n. US 2010/0034457 A1, 2010.
52. D.E. Breen, R.T. Whitaker, E. Rose, M. Tuceryan. Interactive Occlusion and Automatic Object Placement for Augmented Reality. Proceedings of Computer Graphics Forum EUROGRAPHICS'96, Poitiers, France, 1996.
53. J. Fischer, D. Bartz, W. Straßer. Occlusion Handling for Medical Augmented Reality using a Volumetric Phantom Model. Proceedings of ACM Symposium on Virtual Reality Software and Technology, Hong Kong, Cina, 2004.
54. K. Hayashi, H. Kato, S. Nishida. Occlusion detection of real objects using contour based stereo matching. Proceedings of ICAT, Christchurch, New Zealand, 2005.
55. Y. Tian, T. Tao Guan, C. Wang. Real-Time Occlusion Handling in Augmented Reality Based on an Object Tracking Approach. Sensors, vol. 10, pp. 2885–2900, 2010.
56. J. Fischer, B. Huhle, A. Schilling. Using Time-of-Flight Range Data for Occlusion Handling in Augmented Reality. Proceedings of Eurographics Symposium on Virtual Environments (EGVE), Weimar, Germany, 2007.

57. C. Chae, K. Ko. Introduction of Physics Simulation in Augmented Reality. Proceedings of the International Symposium on Ubiquitous Virtual Reality, ISUVR. IEEE Computer Society, Washington, DC, USA, 2008.
58. W. Huagen, G. Shuming, P. Qunsheng. Virtual Grasping for Virtual Assembly Tasks. Proceedings of the 3rd International Conference on Image and Graphics, Hong Kong, China, 2004.
59. D.A. Bowmann, L.F. Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments, Proceeding of the Symposium on Interactive 3D Graphics, Providence, RI, USA, 1997.
60. B. Mac Namee, D. Beaney, Q. Dong. Motion in augmented reality games: an engine for creating plausible physical interactions in augmented reality games. Internaional Journal of Computer Games Technology, Article ID 979235, 2010.
61. B.V. Mirtich. Impulse-based dynamic simulation of rigid body systems. PhD thesis, University of California, Berkeley, CA, USA, 1996.
62. A. Schmitt, J. Bender, H. Prautzsch. On the convergence and correctness of impulse-based dynamic simulation. Internal report, n. 17, Institut für Betriebs und Dialogsysteme, Karlsruhe, Germany, 2005.
63. A. Schmitt, J. Bender. Impulse-based dynamic simulation of multibody systems: numerical comparison with standard methods. Proceedings of Automation of Discrete Production Engineering, Sozopol, Bulgaria, 2005.
64. E. Coumans. Bullet 2.74 Physics SDK Manual, 2009. [Online]. Available: http://bulletphysics.com [accessed may 2011].