STAG: Smart Tools and Applications in Graphics (2017)
A. Giachetti, P. Pingi and F. Stanco (Editors)

A Seamless Pipeline for the Acquisition of the Body Shape: the Virtuoso Case Study

M. Saba[1], F. Sorrentino[1], A. Muntoni[1], S. Casti[1], G. Cherchi[1], A. Carcangiu[1], F. Corda[1], A. Murru[1], L. D. Spano[1], R. Scateni[1], I. Vitali[2], O. Salvetti[3], M. Magrini[3], A. Villa[3], A. Carboni[3] and M. A. Pascali[3]

[1]Department of Mathematics and Computer Science, University of Cagliari, Italy
[2]Intecs, Ospedaletto, Pisa, Italy
[3] ISTI-CNR, Pisa, Italy

Abstract

In this paper, we describe the design and the implementation of the demonstrator for the Virtuoso project, which aims at creating seamless support for fitness and wellness activities in touristic resort. We define the objectives of the user interface, the hardware and software setup, showing how we combined and exploited consumer-level devices for supporting 3D body scan, contact-less acquisition of physical parameters, exercise guidance and operator support.

1. Introduction

The Virtuoso project was designed as the result of a specific call for projects by the Sardinian Regional Government (Italy) requiring to propose the design and implementation of an innovative multimedia systems able to improve the health of tourists during their vacations. The main idea behind the proposal was to acquire several parameters of the user upon the arrival at the vacation resort and, eventually, to compare them with the same parameters acquired at the end of the vacation, to give an idea of how the vacations could change the *wellness* of the single individual.

The project evolved in realizing a seamless pipeline of semi-automatic acquisition of the shape characteristics of the human body, where the user-interaction was carefully designed with the aim of being usable even by a non specialist of the field, like the operator of the resort.

The most interesting contributions of this work can be summarized as follows:

- The hardware used for building the prototype is off-the-shelf and relatively cheap; the most relevant piece of hardware is the rotating board, on top of it we just use a normal PC with camera and depth camera.
- We integrate in the project the mixed competences of our research group, which is active in both fields of computer-human interaction and geometry processing; this allows to have a deep insight on the shape acquisition and manipulation mixed up with the scientific design of the interaction.
- We report the implementation details, showing which problems we faced and how we solved them for creating the working system. This may be reused by both researchers and practitioners for creating similar interaction environments.
- We were able to perform a large acquisition campaign made in a public environment, much more challenging than a user test conducted in a lab, and even more challenging than the final deployment site; more than one hundred full body scans were performed in three days and we have been able to give an immediate feedback to all the subjects.

2. Related work

In this section, we provide an overview of systems and research work presenting interaction similarities with our work. We will introduce first applications based on the smart mirror metaphor exploiting motion sensing devices. Then, we discuss the approaches for acquiring body measures without contact or with a low intrusiveness for users.

2.1. The mirror metaphor

The availability of body tracking devices such as the Microsoft Kinect, the Leap Motion or the Creative Senz3D fostered the design of different applications based on the mirror metaphor: the user's image is "reflected" by a screen and, through a set of gestures, s/he can control the interface. The designed interaction follow three main styles: *command-style*, *augmented reality* and *virtual reality*[†].

In the first style (command), the user navigates the information

[†] In this paper, we include in the definition of Virtual Reality the environments where the user controls an avatar through his/her movements and interacts with a completely synthetic world, since it provides a certain level of immersion even without a head-mounted display.

DOI: 10.2312/stag.20171229

displayed in the mirror with a set of gestures interpreted by the interface as commands such as e.g., swipe for next or previous, thumb up for confirmation etc. The interface reacts updating its status but the displayed information has no spatial relation with the user: it has a dedicated portion of the interface and it can be activated independently from the user's position. Vera et al. [VGCF11] applied such style in what they call the "control scenario", where the user navigates the information through commands.

The augmented reality style inserts virtual objects inside the reflected mirror image. The user maps the position of the virtual objects inside the real world through the feedback of his/her reflected image. This style allows making the virtual object interactive and using them for both express selection and commands, overlaying additional information on real objects or for simulating a real-world behaviour of the virtual objects through physics.

Such approach is one of the most used in the literature. Vera et al. [VGCF11] used it in the *augmented scenario* for mixing reality and virtual objects for mirror-based entertaining games. Meng et al. [MFB*13,MFS*15] proposed an augmented mirror for teaching anatomy. The application tracks the user's skeleton using Microsoft Kinect and provides an in-situ overlay of organs and bones models the user can explore and interact with. Hoermann et al. [HDSM*15] exploited the augmented reality style for providing exercise guidance in rehabilitation.

The virtual reality style does not provide a realistic reflection of the user's image. It displays an avatar that mimics the user's movement. Whit respect to the augmented reality approach, the user does not see his/her proper reflection, but having a complete virtual representation of the world avoids the combination of real and virtual objects in the mirror, allowing for a more controlled environment. Such style is mainly used for game-like experiences with different objectives. Henriksen et al. [HNS*16] used it for treating phantom limb pain. Won et al. [WBLL15] applied such metaphor for studying how users can control bodies different from their own.

2.2. Contactless body measures

The mirror metaphor has been combined with the acquisition of physiological measures in the "Medical Mirror" [PMP11], where a real mirror was placed on top of a normal screen equipped with a webcam. The system is able to detect faces and, through an analysis of the RGB image, measures the heartbeat in 15s. The "Wize Mirror" [CCG*15] exploits different devices for detecting the user's physical state, such as a breathalyser, but also contactless equipment such as RGB and multispectral cameras a depth sensor. It detects and monitors semeiotic face signs, for reducing the cardio-metabolic risk, and it encourages improving the user's lifestyle. We combine the idea of acquiring physiological measures through contactless devices, including the information navigation and physical exercises.

Different user's data acquisition procedures rely on Microsoft Kinect. Xu et al. [XYZ*13] proposed an accurate method for estimating body parameters of moving users, such as arm and leg length, neck-hip distance and waist and hip girth. Araujo et al. [AGA13] acquire similar information for identifying users from the way they walk. Velardo et al. [VD11] discuss a solution for estimating people's weight from the Kinect data, while Domingues et al. [DBP*16] compared the precision of measures estimated with the Kinect against the Qualisys validation system, obtaining a higher precision.

3. Supported Interaction

The application has been designed for supporting a tourist arriving at his or her chosen resort, to enjoy a relaxing week or fortnight of vacation. Most probably, he or she does not like to undergo a set of medical exams to figure out what is his or her state of health. We support the acquisition with a rapid and unintrusive scanning procedure. The tourist enters wearing a swimsuit into a funny room with a board on the ground and, simply completing a spin, the system estimates a set of physical parameters for suggesting activities inside the resort. The parameters require the acquisition of a 3D model of the user and his/her weight.

The interface is based on a mirror metaphor, which opportunistically exploits different visualisation and interaction techniques for presenting information and collecting input from the user. It presents the results of the analysis through tables, graphs and other graphical elements, but also overlaying the measures on the user's reflected image. Finally, it supports exercise guidance, tracking the user's movements and helping him or her to execute them correctly.

The mirror senses finger touches on its surface, but it tracks also gestures for the interaction from a distance.

4. Hardware setup

Supporting the interaction scenario requires different hardware devices, which must be coordinated by the interface software components. Figure 1 shows the selected hardware setup. It consists of the following components:

1. A large monitor, representing the smart mirror.
2. A depth sensor for tracking the user's movement and for supporting the 3D body scan.
3. A rotating board for automatically turning the user during the acquisition
4. A RGB camera for "reflecting" the user's image.
5. A board for evaluating the user's weight and posture.
6. A PC for controlling the user's interface
7. A PC (complete with monitor) for controlling the operator's interface
8. One or more mobile devices for accessing and updating the user's data remotely.

The market offers different devices that fit the setup requirements. In general, they vary according to their cost and the precision for both the interaction and the physical parameters estimation. For each one of the components we briefly discuss the selected device, and the motivation of its preference over the existing alternatives. Our main goal was maintaining the support for the scenario at a reasonable cost, which would enable the adoption of the application in real resorts.

Mirror. For the implementation of the smart mirror we need

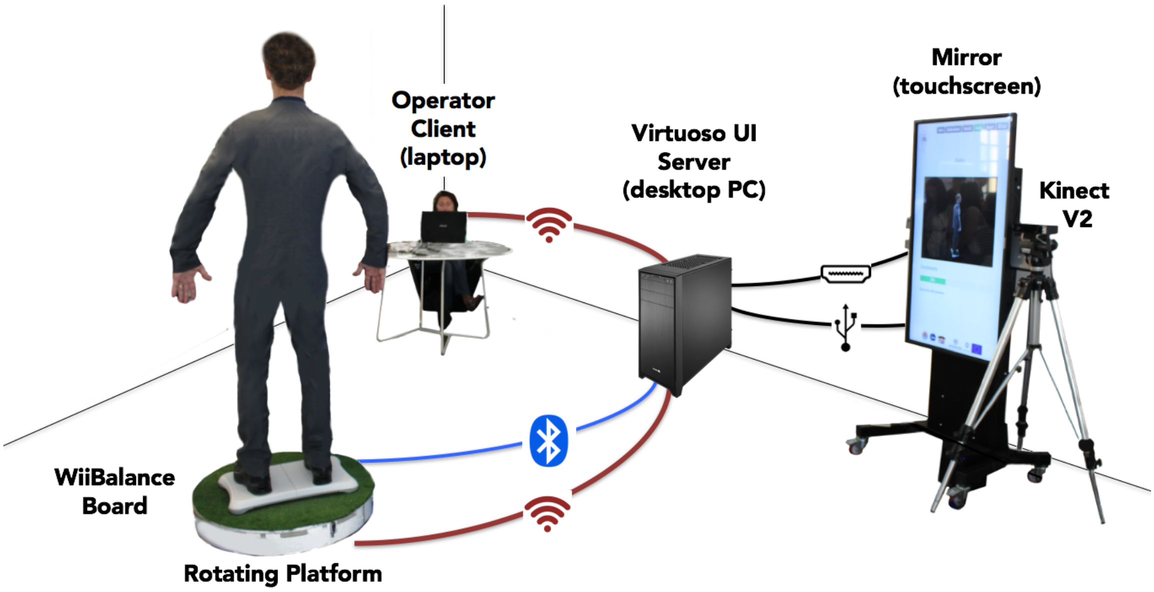


Figure 1: *Virtuoso hardware setup*

a large screen supporting multitouch interaction. Many producers provide screens with built-in multitouch support, but large version (e.g., more than 42") are quite expensive. Other solutions consists of two hardware components: the screen panel and an external device for sensing the user's touch. The latter usually consists of a frame equipped with a touch-sensible glass, that seamlessly adds interaction capabilities to standard screens[‡]. Another solution is applying films on the screen surface, designed for adding interaction capabilities to e.g., shop windows and showcases[§]. In our development setup, we used a touchscreen with built-in support, but resorts may select less expensive solutions. However, from the implementation point of view, a change in the selection of the screen and touch device does not require writing additional management code, since all differences are handled at the operating system level.

Depth sensor. The most promising candidate for supporting gestural interaction was the Microsoft Kinect V2. Considering the success of the first version, especially in the interaction research field, such device offers different advantages that are relevant for our project. First of all, it is cheap (it costs about 100€) and it offers a higher resolution for both the RGB camera and the depth sensor. In addition, an advanced SDK[¶] provides many features that are required for this project, such as body tracking for gestural interaction. In addition, it supports 3D object scanning through an implementation of the Fusion algorithm [IKH*11]. Finally, it provides also a full-HD RGB camera for implementing the mirror "reflection".

Rotating board. For supporting a 360^o 3D body scan using a single depth sensor, we need to either rotate it around the user or to rotate the user around him/herself keeping the sensor in the same position. Obviously, the second solution is much simpler, since the sensor is wired and would require special sliding guides. In order to obtain a good result from the scanning procedure, the rotation must be smooth and controllable by the application. The solution we adopted consists of a rotating board, similar to those used for creating 360^o photo galleries for e-commerce products. We needed a board whose diameter may contain a standing person and able to withstand the weight of a person. We opted for one produced by an Italian company[‖], having a diameter of 70cm and a maximum load of 300kg. The board contains internally an Arduino board for sending rotation commands via WiFi, specifying the angle and the speed.

Balance board. For acquiring information on the user's standing posture and his/her weight we used the Nintendo Balance Board[**], containing four pressure sensors (two for each foot) and supporting Bluetooth connection with a PC, using third party libraries. More advanced devices offer a precise stabilometric analysis, but they are targeted for professional use and they have much higher costs.

Operator and mobile devices. Besides the configuration of the mirror interface, the application has two additional clients. The first

[‡] For instance, see http://www.touchwindow.it/en/overlay-multi-touch.php

[§] For instance, see http://www.scenes.it/touch_screens.html

[¶] https://developer.microsoft.com/en-us/windows/kinect

[‖] http://automationforfashion.it/prodotti/pedane-girevoli/

[**] https://www.nintendo.com/consumer/downloads/wiiBalanceBoard.pdf

one is a control interface for the resort or wellness centre operators. It allows triggering and monitoring the 3D scanning process through a dedicated PC or tablet. In addition, users can access their personal data and maintain an activity log through their mobile device. Both devices are not used exclusively for interacting with the Virtuoso application, both interfaces should be accessible from the higher number of devices possible.

The cost of our test setting is dominated by the large multitouch screen (about 3000€) and by the rotating platform (about 2000€). However, for a real installation in a resort, both may be replaced with cheaper solutions, such as e.g., a large screen (about 800€) and a multitouch frame (300€) with a smaller platform with a wired connection (1000€). The interactive devices (both Kinect and WiiBalance Board) cost about 150€. In addition, a computer capable of running the Kinect SDK is required. At the time of writing, it costs about 700€. In summary, the described setup may be replicated at the cost of about 3000€.

5. Software Setup

In this section we discuss the design and the implementation of the software solution to manage the complexity of the hardware configuration described in section 4 and supporting the envisioned interaction.

Figure 2 shows an overview of the software components in the Virtuoso project. The application logic consists of two main packages: the *RemoteServer*, and the *LocalServer*. The former maintains the user's data requiring a ubiquitous access and the mobile interface. The latter is more complex. It contains the components for managing the interface for the tourist and the operator both at the client and server side (respectively *VirtuosoUIClient* and *VirtuosoUIServer*, *OperatorUIClient* and *OperatorUIServer*). In addition, it provides the wrappers for accessing the Kinect, the WiiBalanceBoard and the rotating platform (*KinectServer*, *WiiBalanceServer*, *RotatingPlatformServer*). Each wrapper communicates with the device exploiting its SDK or driver. The system distributes the updates coming from the devices with dedicated events distributed through websockets. The other components register to sources they require (e.g., the *KinectSocket* for receiving updates on user's movements) and may send configuration commands through them for filtering notifications no longer needed (e.g., the Kinect RGB stream) at any time.

In the rest of the section, we discuss more in detail each component, discussing the design choices.

5.1. User interface toolkit

The interface we designed for interacting with the Virtuoso system exploits different interaction and presentation styles, according to the specific input needed and user task. For instance, we support the navigation of the different application functionalities through both touch and body gestures, the exercise games require a gestural avateering control, while the authentication or registration requires standard forms.

We considered different candidates for the UI implementation. The first candidate was Microsoft Windows Presentation Foundation (WPF) [Cha06], which easily integrates with the Kinect SDK, it has a good GUI widget library and extensions for supporting advanced data visualisation. The main limit in selecting this technology is the support offered in the exercise interface, which requires abstractions on the 3D API similar to those available in games. In addition, it would tie the operator control interface to Windows devices or it would require a different technology for supporting that part of the project.

Proper game engines, such as Unity [††], have the opposite problem: they are good for implementing the exercise part but they offer less support for more conventional widget-based interaction. We considered also splitting the interface in two applications, but this would result in a cumbersome interaction.

For integrating all the interface parts and getting an adequate support for all of them, we opted for a browser-based solution, implementing the interface in HTML, CSS and Javascript. In this solution, all devices access the UIs via HTTP, decoupling the machine that handles the devices from the one that manages the interface. This provides many advantages. The first one is an advanced support for the 2D interaction that, considering all frameworks ease the layout development for web applications (e.g. Bootstrap[‡‡]), speeds-up the development of high-quality UIs. The second advantage is related to the management of 3D graphics, that coexists easily with the other parts of the interface. Indeed, there is an increasing support for WebGL [JG17] in all major browsers. Moreover, many Javascript-based game engines exist (e.g., Three.js[§§] or Babylon.js[¶¶]). They control the rendering inside a dedicated element (a canvas) and provide means for loading and animating 3D models in a virtual world. Obviously, they provide a lower graphics quality, but it is sufficient for our exercise game.

The main drawback in this approach is the communication with the input devices and the rotating board. While touch-based interaction has a good support in all browsers, there is no standard support for exploiting the Kinect V2, the WiiBalanceBoard or the rotating board in web applications. Therefore, the cost of this choice was the implementation of an intermediate layer for wrapping the information coming from the input devices and making them available in the Javascript environment.

5.2. Device input management

In order to enable the communication between the browser-based UI implementation and the input devices, we wrapped their information adding a websocket layer. We associate each device with a dedicated server-socket endpoint, running on the PC directly connected with them (see Figure 2).

The socket-client is a singleton Javascript object, exposing the device events. We hide the websocket connection inside the reusable object implementation. From the UI development point

[††] https://unity3d.com/

[‡‡] http://getbootstrap.com/

[§§] https://threejs.org/

[¶¶] https://www.babylonjs.com/

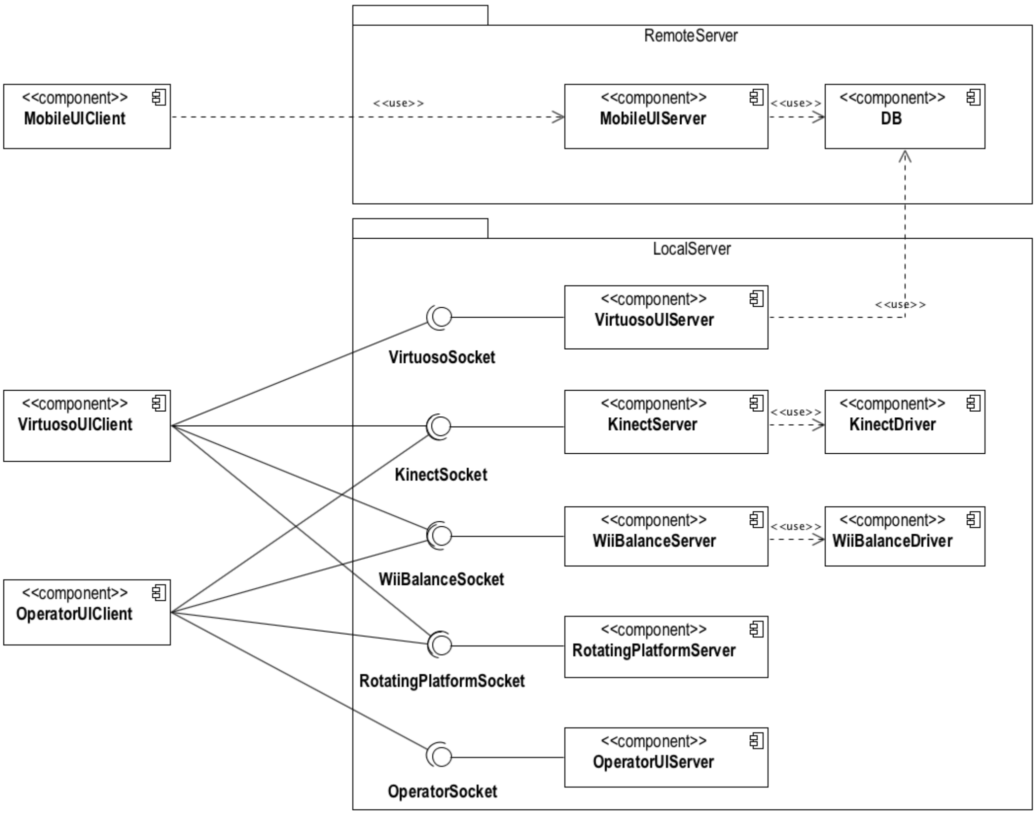

RemoteServer
<<component>> MobileUIClient
<<use>>
<<component>> MobileUIServer
<<use>>
<<component>> DB
LocalServer
<<component>> VirtuosoUIServer
<<use>>
VirtuosoSocket
<<component>> VirtuosoUIClient
<<component>> KinectServer
<<use>>
<<component>> KinectDriver
KinectSocket
<<component>> WiiBalanceServer
<<use>>
<<component>> WiiBalanceDriver
WiiBalanceSocket
<<component>> OperatorUIClient
<<component>> RotatingPlatformServer
RotatingPlatformSocket
<<component>> OperatorUIServer
OperatorSocket

Figure 2: *Overview of the Virtuoso software components.*

of view, the device object is a simple event emitter, while the websocket layer supports receiving data from other devices in the same network and even more than one client at the same time. This is useful for supporting the 3D scan management: the input is handled both by the mirror interface and the operator UI.

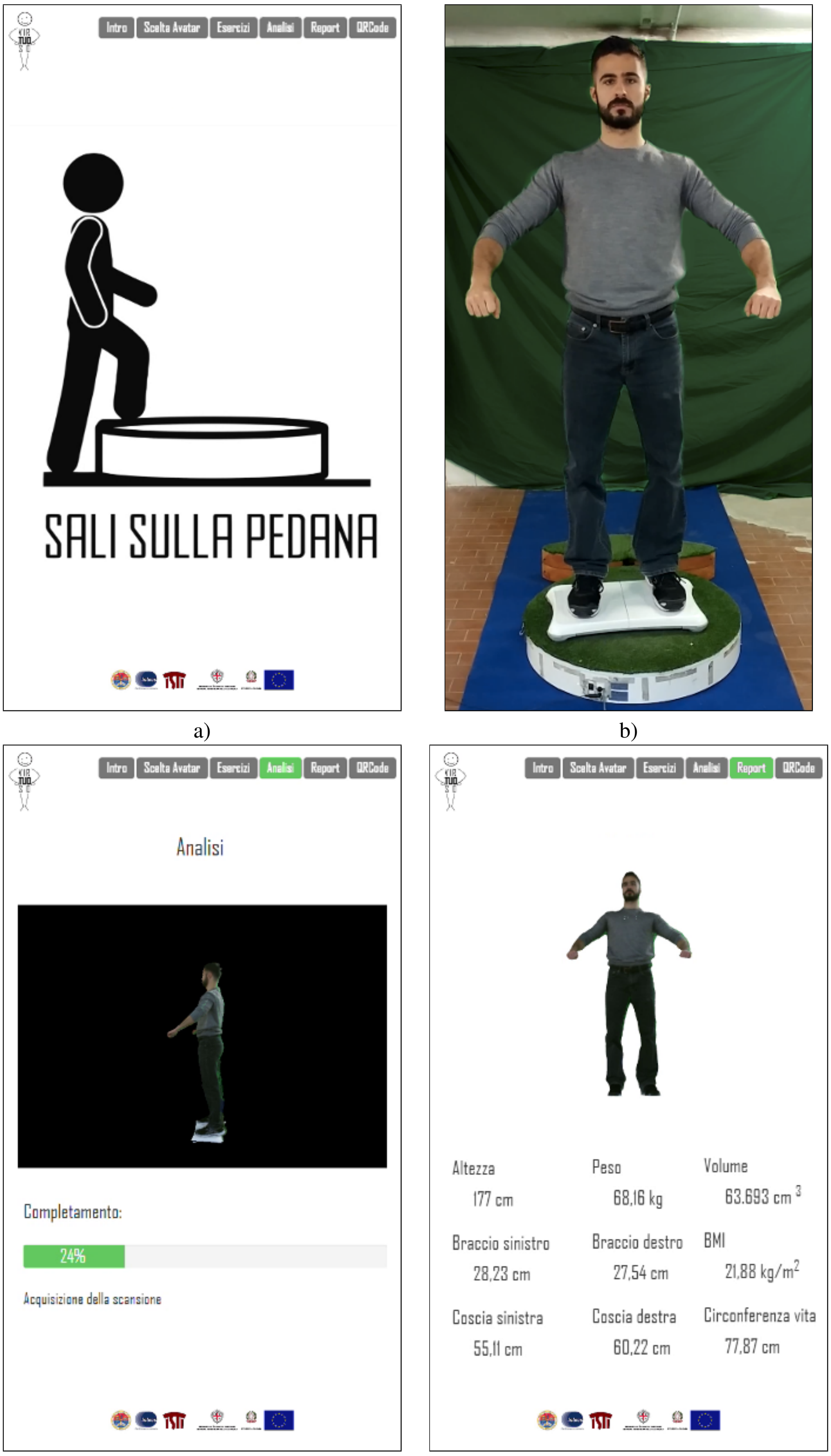
Kinect. The Kinect SDK produces an amount of information that is difficult to stream directly through the websocket layer. It includes: i) four camera images, namely RGB (1980x1080 px), depth (512x424 px), infrared (512x424) and user-index map (512x424); ii) an animation skeleton for each tracked user (max 6), including joints position, rotation in 3D and the state of both hands (open, closed, lasso gesture); iii) information on the Kinect frustum and ground plane definition. All these structures have a high update rate (30 Hz), and usually the interface do not exploit them all: it may use the skeleton information for animating a character, or use it in conjunction with the RGB image, without considering the other ones.

Packing this information for sending it to the Javascript client via websockets adds both marshalling and transmission delays to the update. Therefore, by default the Javascript object receives only the skeleton updates, while other information sources must be explicitly requested to the websocket endpoints. The visualization of all camera streams is implemented encoding directly the data in the `src` attribute of an `img` element, shifting the update from the Javascript interpreter to the browser rendered for speeding-up the visualization and keeping the video fluid.

We serialise the skeleton information in a binary format that is converted into a hierarchy of Javascript object having the same structure of the C# counterpart.

It is worth pointing out that, for keeping the interface responsive to the user's input, the Javascript procedure must only control the visualization of the information coming from the Kinect, without executing complex procedures on images or on skeleton data.

Kinect Fusion. Besides the raw data coming from the sensor, we wrapped with a websocket endpoint the Kinect Fusion SDK, that allows reconstructing 3D environments freely moving the camera in the space [IKH*11]. Obviously, the websocket server endpoint performs all the reconstruction work, saving the generated 3D mesh in a shared folder, accessible by the UI client via HTTP. In addition, it provides the client part with means for modifying the reconstruction parameters: minimum and maximum depth threshold, voxels resolution on all axes, volume max integration weight and whether it should map the colour image on the mesh or not. Finally, the endpoint provides with an image preview of the mesh, pre-rendered at server side, whose point of view may be changed through websocket messages. In this way, it is possible to support realtime previews of the final result directly in the browser.



SALI SULLA PEDANA

a)

b)

Analisi

Completamento:

24%

Acquisizione della scansione

Altezza 177 cm

Peso 68,16 kg

Volume 63.693 cm ^{3}

Braccio sinistro 28,23 cm

Braccio destro 27,54 cm

BMI 21,88 kg/m^2

Coscia sinistra 55,11 cm

Coscia destra 60,22 cm

Circonferenza vita 77,87 cm

c)

d)

Figure 3: *Virtuoso interface*

WiiBalanceBoard. Wrapping the WiiBalanceBoard with a websocket layer has been much easier than the Kinect: it sends five floating point corresponding to the weight measured by the four pressure sensors and the total sensed weight at 40 Hz. In order to read the updates, we used the WiiDeviceLibrary[||||] in a C# environment.

Rotating platform. The rotation platform connection is supported by another websocket endpoint, which can be configured for communicating with the Arduino board controlling the rotation either via WiFi or through wired USB connection. The problem with the wireless connection is mainly due to producer settings of the board, that create a dedicated network rather than connecting to an existing one, requiring the PC to exploit two networks adapters. The USB connection solves this limitation, but it requires a cable between the PC and the platform. In any case, the websocket allows receiving messages for receiving the settings (speed and rotation angle) and notifies when the rotation is complete with an event.

6. Interaction with the system

As described introducing the scenario, the interface supports the interaction with both the touchscreen and with gestures. The implemented touch interaction is straightforward: the user can activate each element in the interface supporting selection, click, etc. with the standard touch interaction associated to the widget.

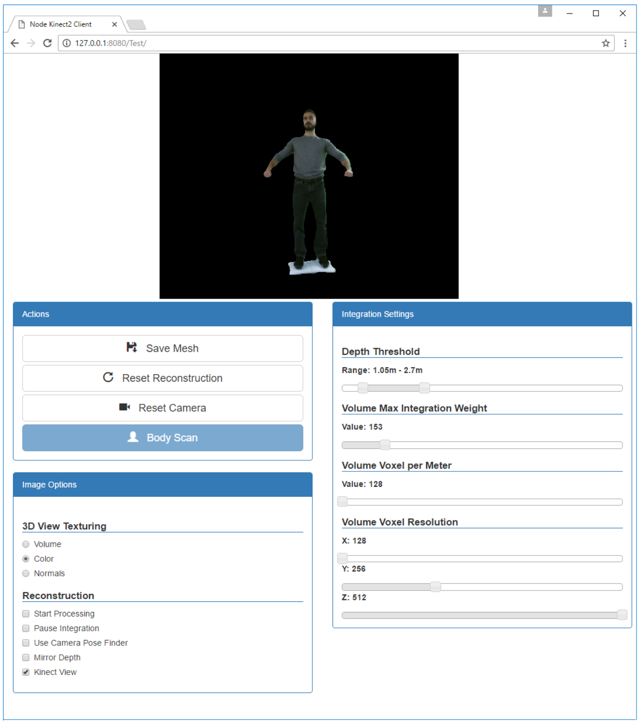
The gestural interaction supports the navigation and the activation of the different functionalities from a distance. We adopted two gestural styles in the interface, depending on the user's activity. The information navigation is supported with deictic gestures, through pointing and selecting object with specific hand postures. The exercise games use pantomimes gestures, where the user completely controls his/her avatar that replicates the movements.

The interaction with the system contains different unusual elements for the user (i.e. gestural interaction, avatar control etc.) that requires appropriate guidelines. We included an embodied agent that provides help to the user in natural language, describing the exercises, the procedures and the actions s/he has to do in the different situations.

6.1. 3D body scan

We begin the evaluation of the physical state of the resort guest performing a 3D body scan. This procedure is carried out with the help of a resort operator. First of all, the user steps on the WiiBalanceBoard, placed on top of the rotating platform. The user assumes a modified T pose, with forearms pointing down depicted in Figure 3-b. This allows separating the arms from the rest of the body, while maintaining a low volume of the integration box.

Then, the resort operator checks the scanning algorithm parameters with the interface in Figure 4. In particular, s/he may adjust the depth threshold, in order to both avoid noise in the reconstruction even if other people walk behind the scan area without cutting outside user's body parts. These parameters need a fine tuning e.g., according to the user's height.



3D View Texturing
○ Volume
● Color
○ Normals

Reconstruction
□ Start Processing
□ Pause Integration
□ Use Camera Pose Finder
□ Mirror Depth
☑ Kinect View

Depth Threshold
Range: 1.05m - 2.7m

Volume Max Integration Weight
Value: 153

Volume Voxel per Meter
Value: 128

Volume Voxel Resolution
X: 128
Y: 256
Z: 512

Figure 4: *Virtuoso operator UI for controlling the 3D body scan process.*

The operator sees a real time preview of the integration process on the dedicated UI. Using the sliders, s/he can modify the Kinect Fusion algorithm [IKH*11] parameters. Even without a complete understanding of the parameter effects on the algorithm, the real time preview allows a trial-and-error process for finding the delta between the standard configuration and one that is acceptable for the current user. We plan in future work to investigate more the interaction with the parameter setting panel, in order to better support people without technical knowledge through help and guided actions.

Once all parameters are set, the operator asks the user to stay still while the platform rotates and presses the activation button on the control UI. The 3D scanning process starts and the mirror shows the reconstruction preview together with a completion bar (Figure 3-c). Once completed the 360^o rotation, the user can relax him/herself while the system analyses the reconstructed mesh, estimating the physical parameters as described in section 6.2.

Finally, the interface shows both the parameter estimations and the acquired 3D model, which can be explored with touch and gestural interaction (Figure 3-d).

6.2. Morphometric measures

The data acquired through the Kinect Fusion (skeleton coordinates and 3D mesh) are used to perform a morphological analysis of the

[||||] http://wiibrew.org/wiki/Wii_Device_Library body: the proposed algorithm is able to automatically locate landmark points (armpits, crotch point), extract the relevant geometrical features and measures, and to compute the major indices for body fat estimation.

Obesity is an established risk factor for increased morbidity and mortality. The assessment of obesity is primarily based on the body mass index (BMI), but BMI is a crude measure, not distinguishing between lean and fat mass. However, recent studies have shown that body fat distribution and in particular abdominal obesity is closely associated with risk of morbidity and mortality. In [HMZ*10, PBH*08] the authors propose the waist circumference (WC) and the waist-to-hip ratio (WHR) as indicators of abdominal obesity, resulting to be better predictors of risk than BMI for several diseases, including cardiovascular disease Another index, designed to be a potential replacement to BMI, and called body volume index (BVI) has been proposed in [TBB*08] to evaluate obesity.

In our approach, we used both the skeleton data and the point clouds. The skeleton data are useful for driving the segmentation of the human body figure represented in the point cloud. We first remove outliers and floor points from the point cloud, exploiting the information provided by the Kinect SDK (floor plane and user's skeleton); then we perform a remeshing to ensure a uniform quality using the Poisson method [KBH06]. These first steps are implemented using Meshlab scripting [CCC*08]. Then, the algorithm proposed proceeds as described in the following six steps. Please note that the entire process is provided as a service called directly by the UI components, without the need of any manual procedure.

Platform Detection: A direct analysis of the triangulated mesh is performed in order to find the platform (level "0") of the system.

Extraction of skeleton joints points: The spatial coordinates of the joints, given by the sensor, are matched to the correspondent triangles in the mesh.

Mesh slicing: This is a key step: the triangulated mesh is cut into a large set of slices, and the system computes an approximation of the intersection curve between the mesh and a given plane (Figure 5).

Best Fitting of Geometrical Ellipse: The slice containing the hips is identified; hence, the system analyzes its geometrical properties in order to find the best candidate for the hip circumference and for the crotch point.

Underarm Points: The right and left shoulders' points are used to define the slicing of the upper part of the torso and to detect the armpits.

Torso Surface and Volume: Given the hip and underarm circumferences (respectively the blue and yellow curve in Figure 6), the body in between the two curves is divided into n regions and for each one the system computes its volume and surface.

Hence, the system can compute some relevant indexes related to human wellness: in particular, we focused on the Barix [SG09] and the WHR wait-to-hip ratio, defined by the following expressions

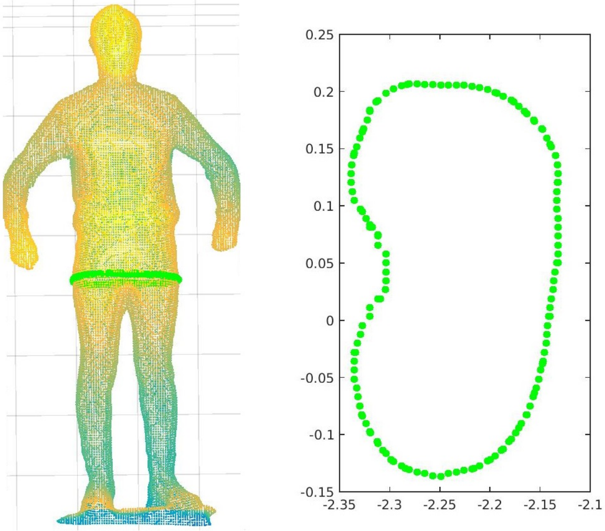


Figure 5: *Mesh slicing at hips level*

$$\mathbf{Barix} = \frac{\text{Torso height} \cdot \text{Torso surface area}}{\text{Torso Volume}}$$
$$\mathbf{WHR} = \frac{\text{Waist length}}{\text{Hip length}} \tag{1}$$

Also, we believe that the analysis of the area-height graph (Figure 6, right) over the whole torso could be useful: people in healthy condition seem to have two peaks and a regular pattern, while people with excess body fat have a more jagged pattern. We could use the graph given by area- in order to localize the adiposity in excess, and eventually guide a targeted intervention. For example, in Figure 6 we can note a peak at the level of the waist. A machine learning approach of the analysis of this area-height function is currently under development.

6.3. Mirror interface

Besides the classic table visualization of the physical parameters, the system supports an Augmented Reality (AR) layer overlay of the values on the mirror screen "reflection". As already discussed in section 4, we obtain the image from the Kinect RGB camera. We anchor the information layer on the user's skeleton joints received from the Kinect websocket endpoint, as shown in Figure 7-c.

The interface supports the navigation of the information through three main gestures, all provided with a feedback guidance:

1. *Hand closure*. This gesture switches the AR layer on and off. In order to avoid false activations for input flickering, the user must keep the position in Figure 7-a for at two seconds. We show an animated arc that increases its angle until it completes a full circle as feedback during the activation.

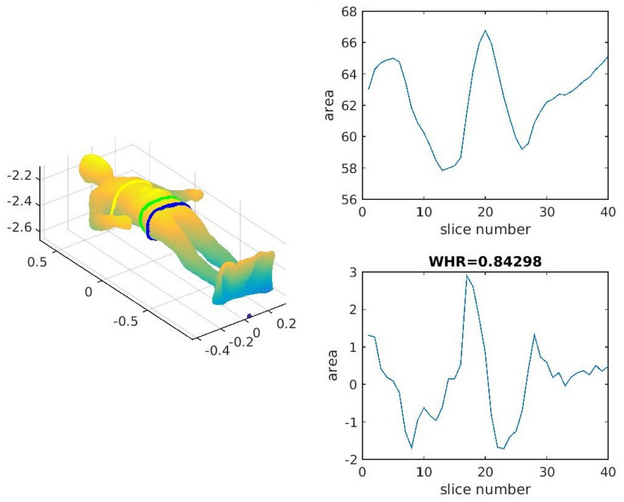
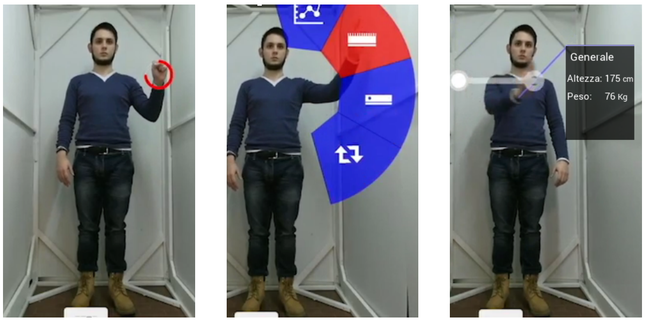


Figure 6: *Mesh slicing, area-height and its derivative*

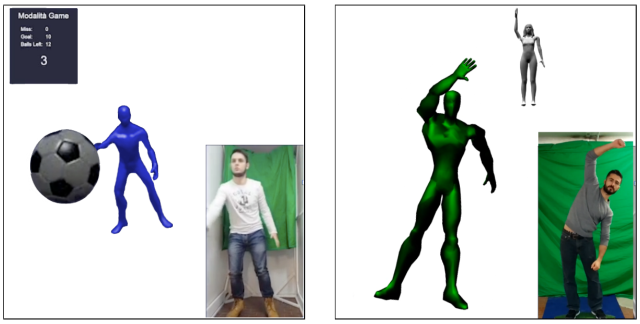


a) b) c)

Figure 7: *Gestural interaction with the mirror interface*

2. *Pointing*. Once the AR-layer is available, the user can select an option from a radial menu (see Figure 7-b), positioning his/her dominant hand over the option and confirming with a lasso hand pose (index and middle finger stretched, the others closed) for a second.
3. *Swipe*. The overlayed information exploits pagination for avoiding a cluttered visualization. The user can change the piece currently visualised through a hand swipe (see Figure 7-c), from left to right for showing next page and from right to left for showing the previous one. The interface shows a line feedback for guiding the user during the performance.

The sample interaction in Figure 7 shows the navigation of the physical parameters in the AR mode. The user first activates the radial menu (part a) and selects the measurement icon (part b). The interface shows a callout positioned on his chest, showing general body information such as height and weight (part c). Through a set of left to right swipes the interface shows the acquired parameters related to other specific body parts.



a) b)

Figure 8: *Interaction with the VR interface*

6.4. Exercise games

Virtuoso supports guided physical exercises through a game-like interface. An avatar replicates the user's movements, allowing the introduction of virtual object for creating more engaging exercises. For instance, Figure 8-a shows an interactive game where the user has to hit virtual balls falling from the sky. The upper-left part of the interface shows the points s/he collected. Figure 8-b shows another simple game, where the user imitates the movements of a virtual coach performing an exercise. The avatar changes is colour to green each time the user performs the exercise correctly.

6.5. Mobile interface

We provide users with a mobile interface for logging their personal activity and for visualizing data summaries. In addition, the interface supports the visualization of the 3D body scan and the querying the physical parameters directly through the 3D visualization.

7. Conclusion and Future Work

In this paper, we discussed the design and the implementation of the Virtuoso project demonstrator. The system allows a quick evaluation of the user's physical state analysing his/her 3D model acquired through a scanning process. The interface is based on a mirror metaphor, which exploits different styles according to the task supported: 2D tables and charts for data summaries, augmented reality for physical parameters navigation and presentation, virtual reality for exercise guidance.

We detailed the hardware requirements and setup, together with the software components defined for building the solution. In addition, we report on the design and the technical implementation choices we made for the user interface, for combining the different styles in a single application.

Finally, we discussed how we analyse the 3D scan for obtaining morphometric measures for evaluating the user's physical state.

In future work, we want to increase the number of the morphometric measures supported by the system and to link them with reference models for building a precise recommendation system, considering both the user's dimensions and the resort scheduled activities.

In addition, we will run an evaluation study for assessing the usability of the entire system, with a long term experimentation in a real resort. We already deployed an instance of the system at a technology fair for three days. About one hundred persons tried the system and provided positive feedback on both measures and the supported interaction. Given the target audience of the project, a long-term study will highlight the user's satisfaction and the perceived usefulness for both resort owners and customers.

8. Acknowledgments

The Virtuoso project is funded by Sardinia Regional Government (CUP code F78C13000530002) with the funds of Regional Law 7/07, year 2013, "Invito a presentare progetti di ricerca fondamentale o di base orientata a temi di carattere specifico di stretto interesse regionale".

References

[AGA13] Araujo R. M., Graña G., Andersson V.: Towards skeleton biometric identification using the microsoft kinect sensor. In *Proceedings of SAP 2013* (2013), ACM, pp. 21–26. 2

[CCC*08] Cignoni P., Callieri M., Corsini M., Dellepiane M., Ganovelli F., Ranzuglia G.: MeshLab: an Open-Source Mesh Processing Tool. In *Eurographics Italian Chapter Conference* (2008), Scarano V., Chiara R. D., Erra U., (Eds.), The Eurographics Association. doi:10.2312/LocalChapterEvents/ItalChap/ItalianChapConf2008/129-136. 8

[CCG*15] Colantonio S., Coppini G., Germanese D., Giorgi D., Magrini M., Marraccini P., Martinelli M., Morales M. A., Pascali M. A., Raccichini G., et al.: A smart mirror to promote a healthy lifestyle. *Biosystems Engineering 138* (2015), 33–43. 2

[Cha06] Chappell D.: Introducing windows presentation foundation. https://msdn.microsoft.com/en-us/library/aa663364.aspx, 2006. Accessed: 2017-05-22. 4

[DBP*16] Domingues A., Barbosa F., Pereira E. M., Santos M. B., Seixas A., Vilas-Boas J., Gabriel J., Vardasca R.: Towards a detailed anthropometric body characterization using the microsoft kinect. *Technology and Health Care 24*, 2 (2016), 251–265. 2

[HDSM*15] Hoermann S., Dos Santos L. F., Morkisch N., Jettkowski K., Sillis M., Cutfield N., Schmidt H., Hale L., Kruger J., Regenbrecht H., et al.: Computerized mirror therapy with augmented reflection technology for stroke rehabilitation: A feasibility study in a rehabilitation center. In *Proceedings of ICVR 2015* (2015), IEEE, pp. 199–206. 2

[HMZ*10] Huxley R., Mendis S., Zheleznyakov E., Reddy S., Chan J.: Body mass index, waist circumference and waist: hip ratio as predictors of cardiovascular riskâĂŤa review of the literature. *European journal of clinical nutrition 64*, 1 (2010), 16–22. 8

[HNS*16] Henriksen B., Nielsen R., Szabo L., Evers N., Kraus M., Geng B.: A virtual reality system for treatment of phantom limb pain using game training and motion tracking. *International Journal of Virtual Reality 16*, 1 (2016). 2

[IKH*11] Izadi S., Kim D., Hilliges O., Molyneaux D., Newcombe R., Kohli P., Shotton J., Hodges S., Freeman D., Davison A., Fitzgibbon A.: Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of UIST 2011* (New York, NY, USA, 2011), ACM, pp. 559–568. doi:10.1145/2047196.2047270. 3, 5, 7

[JG17] Jackson D., Gilbert J.: *WebGL 2 Specification*. Editor's draft, Kronos Group, Jan. 2017. https://www.khronos.org/registry/webgl/specs/latest/2.0/. 4

[KBH06] Kazhdan M. M., Bolitho M., Hoppe H.: Poisson surface reconstruction. In *Proceedings of SGP 2006* (2006), The Eurographics Association, pp. 61–70. doi:10.2312/SGP/SGP06/061-070. 8

[MFB*13] Meng M., Fallavollita P., Blum T., Eck U., Sandor C., Weidert S., Waschke J., Navab N.: Kinect for interactive ar anatomy learning. In *Proceedings of ISMAR 2013* (2013), IEEE, pp. 277–278. 2

[MFS*15] Ma M., Fallavollita P., Seelbach I., Heide A. M., Euler E., Waschke J., Navab N.: Personalized augmented reality for anatomy education. *Clinical Anatomy* (2015). 2

[PBH*08] Pischon T., Boeing H., Hoffmann K., Bergmann M., Schulze M. B., Overvad K., Van Der Schouw Y., Spencer E., Moons K., Tjønneland A., et al.: General and abdominal adiposity and risk of death in europe. *New England Journal of Medicine 359*, 20 (2008), 2105–2120. 8

[PMP11] Poh M.-Z., McDuff D., Picard R.: A medical mirror for non-contact health monitoring. In *ACM SIGGRAPH 2011 Emerging Technologies* (2011), ACM, p. 2. 2

[SG09] Stefan D. B., Gilbert D. A.: Method for determining a subject's degree of adiposity, Oct. 6 2009. US Patent 7,599,537. 8

[TBB*08] Tahrani A., Boelaert K., Barnes R., Palin S., Field A., Redmayne H., Aytok L., Rahim A.: Body volume index: time to replace body mass index? 8

[VD11] Velardo C., Dugelay J.-L.: Real time extraction of body soft biometric from 3d videos. In *Proceedings of MM 2011* (2011), ACM, pp. 781–782. 2

[VGCF11] Vera L., Gimeno J., Coma I., Fernández M.: Augmented mirror: interactive augmented reality system based on kinect. In *Proceedings of INTERACT 2011* (2011), Springer, pp. 483–486. 2

[WBLL15] Won A. S., Bailenson J., Lee J., Lanier J.: Homuncular flexibility in virtual reality. *Journal of Computer-Mediated Communication 20*, 3 (2015), 241–259. 2

[XYZ*13] Xu H., Yu Y., Zhou Y., Li Y., Du S.: Measuring accurate body parameters of dressed humans with large-scale motion using a kinect sensor. *Sensors 13*, 9 (2013), 11362–11384. 2