Focus plus context visualization based on volume clipping for markerless on-patient medical data visualization

Márcio C. F. Macedo*, Antônio L. Apolinário Jr.

Department of Computer Science, Federal University of Bahia (UFBA), Brazil

Abstract

Focus plus context visualization can be used in augmented reality to improve the visual perception of the augmented scene. In the scope of in situ or on-patient medical data visualization, the focus plus context paradigm is used to improve depth perception for physicians showing the patient's anatomy as a focus region in the context of the patient's body. Volume clipping is one technique to realize focus plus context visualization. However, some of the existing methods for focus plus context visualization based on volume clipping do not run in full real time or are prone to artifacts. In this article, we present an extension for two of these techniques to improve performance and image quality of the original approaches. We validate all the techniques in a markerless augmented reality environment. A 3D reference model is tracked by the application, and volumetric medical data are shown to the user at the position of the patient's anatomy. Our technique is able to handle multiple anatomic regions, although the main region of interest used in this article is the face. Moreover, tracking accuracy is improved by the use of a hierarchical approach. From an evaluation of the proposed techniques, the results obtained highlight that all of them are free of artifacts, optimized for real-time performance, and improve the visual quality of the augmented scene.

Keywords:
Volume clipping, Focus plus context visualization, Augmented reality, Volume rendering

1. Introduction

Physicians see medical data, typically images of a patient's anatomic structures, on a monitor and they must analyze and mentally compose what is shown on the screen. This mental model of the patient's anatomy will help the physician provide health care in time-critical situations. Therefore, the physician must have sufficient knowledge of the patient's and general human anatomy to proceed appropriately during any medical procedure (e.g., diagnosis, surgery). With the availability of augmented reality (AR) technology, one can take over this task of mental mapping by transferring it to a computer. Therefore, the physician will be able to visualize, at the same time, the patient and a part of the patient's anatomy. On-patient or in situ medical data visualization can be used to improve surgical planning, training, medical diagnosis, and post-operative examination. This kind of application is desirable in fields such as those involving craniofacial data, in which the visualization of 3D examinations on the patient may help the physician understand the trauma.

AR is a technology which augments the view of a real scene with additional virtual information. Accurate tracking of the real scene, realistic rendering of the virtual data, and real-time user interactivity are the most important technical challenges of AR applications [1]. The face is a part of the body in which depth- or texture-based tracking is easier because of the availability of face detection algorithms and the presence of distinguishable geometric structures. We take advantage of this to focus on the problem of on-patient medical data visualization for patients with craniofacial traumas. The decision to use a markerless AR (MAR) environment for tracking resulted from observations of the current limitations of the techniques proposed in the field of on-patient medical data visualization. Here, we are mainly interested in investigating the possibility of developing an MAR environment for on-patient medical data visualization which supports high-quality on-patient visualization and depth-based tracking (invariant to illumination conditions). Taking advantage of our main motivation to improve the physician's knowledge of the patient with craniofacial trauma, in this work we focused our tests on the patient's head as the region of interest (ROI). Although we have developed a solution for the scenario of craniofacial data visualization, in this article we show how the MAR environment can be adapted for other patient ROI (i.e., torso and pelvis; Section 6). The generality of the proposed work is discussed in this article.

Traditionally, on-patient medical data visualization applications superimpose virtual medical data on the patient. However, in such applications, the virtual content seems to be floating in front of the patient. As stated in previous work [2, 3, 4, 5], a better solution is to show the patient's anatomy as a focus region in the context of the patient's own body. This process is known as focus plus context (F+C) visualization paradigm [6],

*Corresponding author

Email addresses: `marciocfmacedo@gmail.com` (Márcio C. F. Macedo), `apolinario@dcc.ufba.br` (Antônio L. Apolinário Jr.)

and it is known to improve the visual perception of the content being visualized. In the field of volume rendering, one way to improve the understanding and extend the exploration of the medical volume is by use of volume clipping. Therefore, the effect of volume clipping added in an F+C visualization technique is a new tool for the user to explore and understand the augmented scene.

The existing techniques for F+C visualization based on volume clipping are prone to artifacts or do not run in full real time [5]. Such issues decrease the application's visual quality and performance, respectively. One way to solve both problems is by use of an adaptive strategy to mitigate artifacts and shaders to execute the technique in parallel.

In this article, we present improvements in terms of performance and visual quality over the F+C visualization techniques based on volume clipping proposed in [5]. We expand the evaluation of the MAR environment for different ROI in the patient and improve tracking accuracy through the use of a hierarchical algorithm. A more detailed description of the algorithms used in the entire solution (i.e., MAR environment and F+C visualization) and an in-depth analysis of the results obtained and the limitations of the proposed approach are presented as well.

The remainder of this article is organized as follows. Section 2 reviews recent related work on medical AR and F+C visualization applied in AR. Section 3 introduces the MAR environment used in this article for validation of the F+C techniques. Section 4 presents the F+C techniques based on volume clipping for on-patient medical data visualization. Section 5 presents the tests conducted and the experimental results obtained. Section 6 discusses the results obtained and the limitations of this work. In Section 7, a summary of the article and recommendations for future work are presented.

2. Related work

Medical AR systems for on-patient medical data visualization have been driven by different approaches in recent years. In this section, we classify the approaches on the basis of their tracking technology: marker based or markerless.

Over the past decades, many relevant approaches have been proposed for marker-based medical AR, such as those in [3, 4, 7]. Artificial fiducial markers provide fast and accurate tracking because of their shape; however, they are commonly associated with some issues which make this technology unsuitable for on-patient medical data visualization applications:

- They are intrusive, because they are not part of the original scene.
- When the traditional fiducial marker, such as the one used in popular applications such as ARToolKit [8], is not used, the optical tracking system hardware may be too expensive.
- In general, this kind of tracking must operate only on the image space, according to features computed from the pixels. The main drawbacks for this color- or texture-based tracking are the susceptibility to illumination conditions and marker occlusion, which may affect the accuracy of the tracking algorithm.

Recently, systems have been proposed in the field of markerless medical AR. Some of them do not run in real-time (more than 15 frames per second, FPS) [9, 10] and others rely on specific prior knowledge about the ROI to be tracked (see [11, 12, 13] for the body and [14] for the face). To the best of our knowledge, there is only one exception which can be used for general-purpose markerless on-patient medical data visualization: the semiautomatic approach proposed in [15, 16, 17].

The semiautomatic MAR environment uses an RGB-D sensor to reconstruct and track a 3D reference model of the patient's ROI through the AR live stream. Then, after the virtual medical data positioning, it can be displayed for a physician at the location of the patient's real anatomy. Real-time performance is achieved by exploitation of the parallelism provided by the graphics processing unit (GPU).

To validate the F+C visualization techniques, we use a marker-free tracking algorithm because it requires a low processing time and can operate on customer hardware with good accuracy. A first necessary step is to evaluate the performance and visual quality of the proposed approach. In this sense, the semiautomatic MAR environment proposed in [15, 16, 17] is used because it runs in real time and, with some adaptations, its tracking solution can be applied for several ROI in the patient, in contrast to other state-of-the-art solutions. Such adaptations are discussed in Section 6.

An application for on-patient medical data visualization requires special attention to be paid to the composition of the virtual and real entities of the AR environment. Recently, many approaches have been proposed in the field of F+C visualization to dynamically define how this composition should be done. These, also known as ghosting or X-ray vision techniques, share the concept of an importance map, a mask (similar to an alpha mask) which controls how real and virtual entities should be blended.

Sandor et al. [18] designed a method for importance map computation based on the feature regions of both real and virtual objects inspired by three features: luminosity—to preserve regions with high illumination; hue—to preserve strong colors; motion—to preserve moving structures in the final rendering. As stated by Sandor et al. [18], this work was an extension of the work of [19], which is based on edge overlay to improve spatial perception.

Mendez et al. [20] proposed an F+C technique in which the lightness and color contrast of a given image are modified according to the importance map computed from a live color video. By adding subtle changes in the image, they guarantee temporal and spatial coherence between frames. The problem with this approach is its performance, which does not achieve the full 30 FPS even when it is implemented on the GPU.

An adaptive F+C visualization technique was recently introduced by Kalkofen et al. [21]. In their approach, an importance map is computed for the occluder [20] and the occludee is inserted into the scene. Then, another importance map is computed from this resulting image and is then compared against the first map computed. Regions on the first importance map that are not present in the final rendering are then emphasized to be visible. This approach improves the visual quality of the augmented scene and it runs in real time. However, it is not suitable for MAR environments, as it alone requires a processing time of 33 ms. Therefore, this additional time would decrease severely the performance of an MAR application.

F+C rendering was also proposed for visualization of underground structures in street scenes [22, 23, 24]. In these approaches, a method is used to dynamically compute when the underground structures must be rendered in relation to moving objects present in the scene. Although the final visual quality is good, the performance of the existing techniques is not full real time.

Traditional methods which compute the importance maps from live color video of the real scene are prone to errors because they are dependent on the illumination and material properties of the real environment. To overcome these problems, Mendez and Schmalstieg [25] proposed a method to compute an importance mask based on the 3D model of the scene. This task is accomplished by use of techniques such as mesh saliency [26] or through user interaction in a preprocessing step. The problem with this approach is that the importance mask creation requires some processing time. Therefore, the user cannot change interactively the importance mask during an AR live stream.

The methods proposed in the literature for F+C visualization in general AR applications capture the features of the image; however, their importance maps are not accurate enough to be used for medical applications.

F+C visualization has been proposed not only for AR, but also for volume rendering. In this case, it is used to define how the internal structures of the volume (e.g., bone, organ) should be visualized in the context of the soft tissue.

Bruckner et al. [27, 28] proposed a method for context-preserving volume rendering. From factors such as shading intensity, gradient magnitude, distance to the eye point, and previously accumulated opacity, the method allows the user definition of the F+C rendering according to only two parameters which controls these four factors to interactively change the transparency level between internal and external structures of the volume. The technique is easy to implement and runs directly on the shader. An extension of this algorithm was proposed by [29]. It incorporates rotation, scale, position, and mouse click to dynamically select focus and context regions.

Kruger et al. [30] proposed ClearView. Four layers (i.e., focus and context structures, isosurface's normal and curvature) are generated and composed for each frame in order to define the final visualization. The main disadvantages of this method is that it is naturally multi-pass (i.e., one shading pass is required to compute every layer) and the layers must be recomputed for every change of viewpoint. Therefore, the approach has a considerable cost in terms of performance.

Kirmizibayrak et al. [31] proposed a volumetric brush method for interactive definition of focus and context regions for volumetric models. Their approach runs in real time and provides a good alternative for physicians to visualize medical data, especially for applications such as radiation therapy.

F+C solutions have also been proposed in the literature to help in the visualization of complex fiber distributions [32], blood flow [33], structured biomedical data [34] and ultrasound [35].

The main goal of the F+C visualization techniques applied in AR environments is to improve the depth perception of the augmented scene. This is specially important for medical AR applications, in which physicians must have good understanding of the augmented scene to proceed with their tasks appropriately. Despite the number of techniques and applications which have been proposed for medical AR, only a few of them consider the visualization a relevant aspect for the application [36]. We show here some of the visualization techniques proposed specifically for the field of medical AR to achieve the goal of improved depth perception.

Lerotic et al. [37] suggested the use of a *pq* space–based nonphotorealistic rendering method for augmented visualization in minimally invasive surgery . In their approach, the anatomic surface is expressed in terms of a *pq*-space representation, where p and q are the slope of the surface along the x and y axes. These values are used to determine which regions of the suface are more salient and must be emphasized in the final rendering. Pratt et al. [38] extended this technique to run in real time on the GPU. For it to do so, the original algorithm was simplified by use of an intensity gradient filter to highlight anatomic surface details.

Bichlmeier et al. [39] proposed the virtual mirror, a technique which improves not only the depth perception, but also the navigation, visualization, and understanding of the virtual structures positioned into the augmented scene. This can be achieved through the use of a specialized hardware setup and standard techniques in computer graphics for mirror reflection computation.

Kersten-Oertel et al. [40] provided an evaluation of several strategies for improving depth perception (namely, fog, pseudochromadepth, kinetic depth, edge depiction, and stereo) in the medical data visualization. The evaluation was conducted with novice and expert users, and the conclusion was that the fog and pseudochromadepth [41] techniques improve understanding of the medical structures.

One of the first techniques proposed for F+C visualization in the field of on-patient medical data visualization was the contextual anatomic mimesis (CAM) proposed by Bichlmeier et al. [2]. Its importance map is defined by three parameters: the curvature of the patient's skin surface, the angle of incidence (i.e., angle between the normal on the skin surface and a vector pointing from the position of the surface and the eye), and the distance falloff (i.e., the distance between each point on the surface and the intersection point of the line of sight and the skin surface). Differently from the color-based methods mentioned for the F+C techniques applied in the common AR scenario, this one operates directly on the shader and is not dependent on illumination or texture for the importance map definition. Although it provides improved visualization of the 3D medical data in the scene, it does not give special attention to the effect of volume clipping.

Aiming to provide the physician with more tools to improve the visual perception of the augmented scene, previous work has proposed three F+C techniques based on volume clipping: the smooth contours technique, and the visible background on CT and MRI data techniques [5]. Each of them defines a specific region of the volume to be used as a focus or context region. However, the smooth contours technique is not optimized for real-time performance, and the visible background on MRI data technique generates images with visible artifacts and, in fact, it was not evaluated with respect to visual quality, although it showed promising results. In this article, we present an extension of the work of [5] to solve such problems.

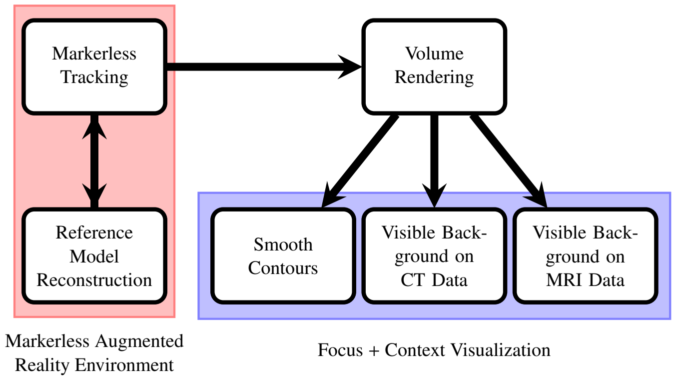

Figure 1: Integrated solution for markerless on-patient medical data visualization based on focus plus context rendering viewed as components and their relationships.

3. Markerless augmented reality environment

In this section, we describe the MAR environment used in this work, which is mostly based on the one proposed in [15, 16, 17]. However, we present modifications to improve tracking accuracy while enabling real-time performance for the MAR environment.

An overview of the proposed solution is given in Figure 1. First, we reconstruct a 3D reference model of the patient's ROI to track it without markers in the AR live stream. Three-dimensional (3D) reference model reconstruction requires markerless tracking to align the different viewpoints acquired from the patient's ROI. In contrast, markerless tracking requires 3D reference model reconstruction to perform live tracking during the on-patient medical data visualization. Because of the recent availability of MAR environments for on-patient medical data visualization, which are based on off-the-shelf hardware and provide good composition of the real and virtual entities in the AR environment [17, 14, 13], they can be used to validate the F+C techniques. From the estimated camera pose (i.e., position and orientation), the medical volume can be rendered and displayed for a physician inside the patient's body at the location of the real anatomy. Volume data are rendered according to standard volume rendering techniques. After volume rendering, F+C visualization techniques (i.e., smooth contours technique, and visible background on CT and MRI data techniques) are used to define which parts of the volume will be visualized in the final augmented scene. Real-time performance is achieved by implementation of the MAR environment (i.e., markerless tracking and 3D reference model reconstruction) on the general-purpose GPU and volume rendering together with the F+C visualization using GLSL shaders.

To track the medical volume in the AR environment without markers, a 3D reference model of the patient's ROI is generated. To reconstruct a single 3D reference model of the patient's ROI, it is necessary to detect it and segment it from the real scene captured by the RGB-D sensor. In this work, F+C visualization techniques were validated in a scenario where the ROI consists mainly of the patient's face. For face detection and segmentation, the Viola-Jones face detector [42] is applied in the color image provided by the RGB-D sensor. Once the ROI has been segmented in the color image, this segmented region is fixed. Then, the user is constrained to move the ROI in this fixed region so that the system can capture the different viewpoints from the same ROI. Through calibration of the color and depth sensors, it is possible to transfer this segmented region of the color image to the depth image. The depth map is denoised with a bilateral filter [43], and then the pyramid algorithm is applied to build low-resolution approximations of the original denoised depth map [44, 45]. To do so, a mean filter is implemented on the GPU to compute only two coarse levels from the denoised depth map. Filtered depth maps are converted into vertex and normal maps. Maps computed from the original depth map are used through all the steps of the algorithm. Coarse vertex and normal maps are used only for tracking. Then, the KinectFusion algorithm [46] is used to reconstruct the reference model of the patient's ROI in real time.

KinectFusion is an algorithm that reconstructs high-quality 3D models from raw, noisy depth data captured from a depth sensor. To do so, for each voxel a 3D grid stores the signed distance to the closest surface and a weight that indicates the uncertainty of the surface measurement. This volumetric representation and integration is based on the VRIP algorithm [47]. One extracts the implicit surface of this representation (i.e., reference model) by detecting zero-crossings (i.e., positions at which the distance sign changes) on the grid through a ray caster. This volumetric representation of KinectFusion is especially useful for the F+C visualization based on the visible background on MRI data technique, where the ray casting algorithm is used to clip the 3D reference model directly from the 3D grid. Moreover, to keep the pyramid framework consistent, a pyramid version of the ray cast data is built for each frame.

As evaluated in [48], the KinectFusion algorithm has maximum accuracy of approximately 10 mm; therefore, it is assumed that its reconstructed models are suitable to be used as reference for tracking and virtual data positioning in MAR applications which do not demand high accuracy. All of the steps described above run in the GPU and are optimized for real-time performance.

The 3D reference model is reconstructed only once, and it is the basis for MAR live tracking. To position the medical volume into the scene, a semiautomatic registration method is used [17]. The virtual data are coarsely aligned with the 3D reference model (which represents the patient's ROI data) in terms of scale, positioning, and orientation. By controlling parameters such as the scale factor, rotation angles, and translation vector, the user is able to make fine adjustments (e.g., rescale the virtual data, change the position of the virtual data, or modify the orientation of the virtual data) over the coarse registration in order to produce a more visually pleasant integration of the medical data into the augmented scene.

After the placement of the medical data into the scene, the markerless tracking is started. In fact, live tracking is done in two steps: during the reconstruction of the 3D reference model, to align the different viewpoints acquired from the patient's ROI, and during the MAR with the patient and the medical data. A real-time variant of the Iterative closest point (ICP) algorithm [49] implemented on the GPU is used to estimate the rigid transformation that aligns the current depth frame captured by the depth sensor with the previous one represented by the 3D reference model. To improve tracking accuracy without too much impact on performance, we use a hierarchical variant of the ICP algorithm, similarly as done in [50]. Hence, we estimate the camera pose starting from the coarsest level to the finest one using the previously computed vertex and normal map pyramid. After each iteration, we update the final camera pose estimated for the current frame. As discussed in the Section 5, by controlling the number of iterations used for each level of the tracking algorithm, we can trade off accuracy and performance of the tracking in the MAR environment.

As stated in [46, 50], the use of the 3D reference model for tracking allows a more consistent rigid registration with less incremental error. However, in the presence of fast rigid motion between frames, the ICP algorithm may fail (i.e., not converge to a valid result). Taking advantage of the fact that the main ROI in this article is a head, we used a real-time head pose estimation [51] to provide a new initial guess to the ICP algorithm to compute correctly the current transformation [52].

4. On-patient medical data visualization based on volume clipping

4.1. Volume rendering

Volume rendering is a field concerned with techniques for synthesizing images from 3D scalar data. This problem of image synthesis is mathematically formulated as a volume rendering integral most commonly based on an emission-absorption optical model [53].

To synthesize the medical image, a single rendering pass ray casting is applied over the bounding box of the medical volume [54]. To improve image quality and performance of the volume rendering, several techniques are used as follows:

- Stochastic jittering (i.e., random ray-start)—to reduce sampling artifacts;
- Fast GPU-Based tricubic filtering—to reduce filtering artifacts [55, 56];
- Empty-space leaping—to skip nonvisible voxels [57];
- Early ray termination—if the opacity accumulated is sufficiently high;
- Preintegrated transfer functions—to capture high frequencies introduced in the transfer functions defined with low sampling rates [58];
- Blinn-Phong shading with on-the-fly gradient computation—to add realism in the final rendering [59];
- GPU tricubic prefilter—to improve tricubic filtering accuracy [60];
- Volume clipping—to extract and emphasize importants parts of the volume [53].

In this work, the volume is clipped according to six planes parallel to the faces of the volume boundind box, although there are several alternative techniques for volume clipping, such as that in [61]. Nevertheless, we emphasize that the F+C techniques can be used regardless of the technique used to crop the volume.

After the volume rendering, medical data must be visualized in the augmented scene. To achieve this goal, F+C visualization is used to show the medical data in a focus region in the context of the patient's body, as described in the next subsection.

4.2. Focus and context visualization

We present improvements over the F+C visualization based on volume clipping proposed in [5]. When one is clipping a volume and rendering its image in an AR environment, there will not be occlusion between the internal region of the volume and the patient's ROI, as shown in left image in Figure 2.

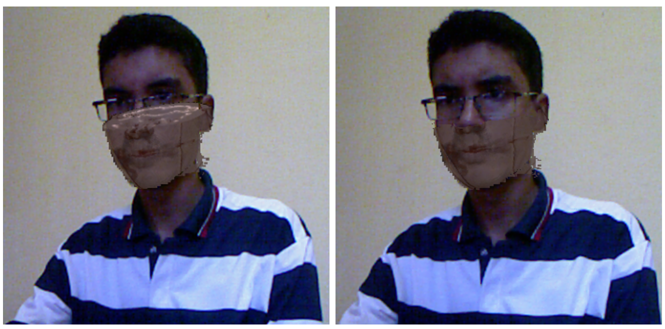

Figure 2: Occlusion between the volume's internal structures and the patient's region of interest. Left image: direct volume rendering with clipping. Right image: volume clipped rendered according to the proposed algorithm.

If desirable, one can solve this issue by changing the single-pass ray casting [53]. We check if the first hit position of the ray cast in the volume is in the clipped region. If it is, the ray stops its traversal and is discarded from rendering. Otherwise, the ray continues its traversal in the volume as normally done in the standard ray casting algorithm. The visual effect of this algorithm can be seen in the right image in Figure 2, where the internal structures of the volume were removed in the final rendering.

4.2.1. Smooth contours

When a volume is clipped, to reveal hidden structures of the medical data, and its image is rendered in an AR environment, edges located at the intersection between the volume and the clipping planes become visible. This visibility of the edges occurs not only in this region, but also for the entire contour of the volume rendered (Figure 3, left image).

Figure 3: Influence of the smooth contours technique in the final rendering. Left image: direct volume rendering with clipping. Right image: volume clipped rendered according to the proposed algorithm. Contours are localized between the yellow shapes.

According to the F+C technique presented in [2], one can improve depth perception by smoothing the transition between the volume in the focus region and the rest of the AR scene. On the basis of this, Macedo and Apolinario [5] proposed a new method for F+C visualization based on the smooth contours technique, an algorithm that adds a smooth transition between the volume rendered and the real scene based on the volume contours.

The smooth contours technique proposed in [5] consists of the following steps: from the medical volume image, which is loaded from the GPU to the CPU, it is converted to grayscale, binarized by use of the threshold computed from Otsu's method [62], contours are extracted from the method proposed by Suzuki [63] and are smoothed by use of a Gaussian blur (kernel size 3×3 pixels). The resulting image is a mask ($\alpha_{smoothCont}$) which weights the blending of the volume and the patient's color images. Also, a factor w_c can be dynamically defined by the user to adjust the level of smoothing of the contours, expanding or compressing the area of operation of the algorithm (Equation 1). It ranges from 0, where the contours are rendered, to $+\infty$, where the contour area is expanded, contours are smoothed, and then suppressed in the final rendering because to the high level of smoothness required.

Instead of the technique running entirely on the CPU, we propose an alternative method for the technique to run entirely on the shader, improving performance and achieving the same quality of the final rendering. To achieve this goal, the pipeline is changed as follows (Figure 4, top part): the medical volume image is binarized by use of a predefined threshold t_b, which operates on the gray intensity of each pixel (empirically we have found $t_b = 0.1$ a good threshold for such a task), and the binary image is blurred by one iteration of a two-pass Gaussian blur (kernel size 3×3 pixels). Instead of explicitly computing the contours by using Suzuki's method, we just apply the Gaussian blur directly over the binary image. In practice, we have not found a perceptual difference between these two approaches. Moreover, as discussed in Section 5, with our new algorithm we improved the performance of the original approach, as we remove the need to transfer data from the GPU to the CPU, which is a time-consuming step. Furthermore, because of the separability of the Gaussian functions, the use of a two-pass approach to convolve the binary image reduces the processing time required by the filter while maintaining the same visual result.

As can be seen in Figure 3, the smooth contours technique softens the transition between the medical volume image and the real scene. Furthermore, this method can be easily integrated with other existing solutions, such as the CAM technique [2]. An example of the result of such integration can be seen in Figure 5. In the top image in Figure 5, a circular mask is defined over the window to select which parts of the medical volume must be rendered into the augmented scene. With the CAM method, there is no clear handling of the contours which result from the clipping the volume. By using the smooth contours technique (Figure 5, bottom image), we can solve this problem by smoothing the contours inside the focus window.

Two methods for F+C visualization that take advantage of the clipping effect and the concept of a visible background were proposed. They take advantage of the type of scanning technology (CT or MRI) to enable new ways for physicians to visualize and explore the medical data on the patient.

4.2.2. Visible background on CT data

In volume rendering, CT data can be used to enable the visualization of internal structures of the patient such as bones. By designing an appropriate transfer function, one can visualize the bone apart from the soft tissue of the volume. On the basis of the color values associated with the soft tissue, the virtual background used for rendering can be seen. In this case, it is desirable to replace this virtual background by the real one, enhancing the visual perception of the scene. Moreover, by use of this strategy, the visualization of the soft tissue is deemphasized in the final rendering, emphasizing rather the focus region of the visualization, the bone structure. The F+C visualization based on the visible background on CT data technique can be applied to enable this kind of visualization. An overview of this method can be seen in the middle part of Figure 4.

The background scene is captured and stored in memory. Next, the image of the volume after clipping is binarized and sent to the shader as a foreground subtraction mask $I_{subtraction}$. This mask identifies the region where the background can be visualized on the basis of the gray intensity of the volume. Then, a user-defined threshold $w_{grayLevel}$ operates on the gray level of

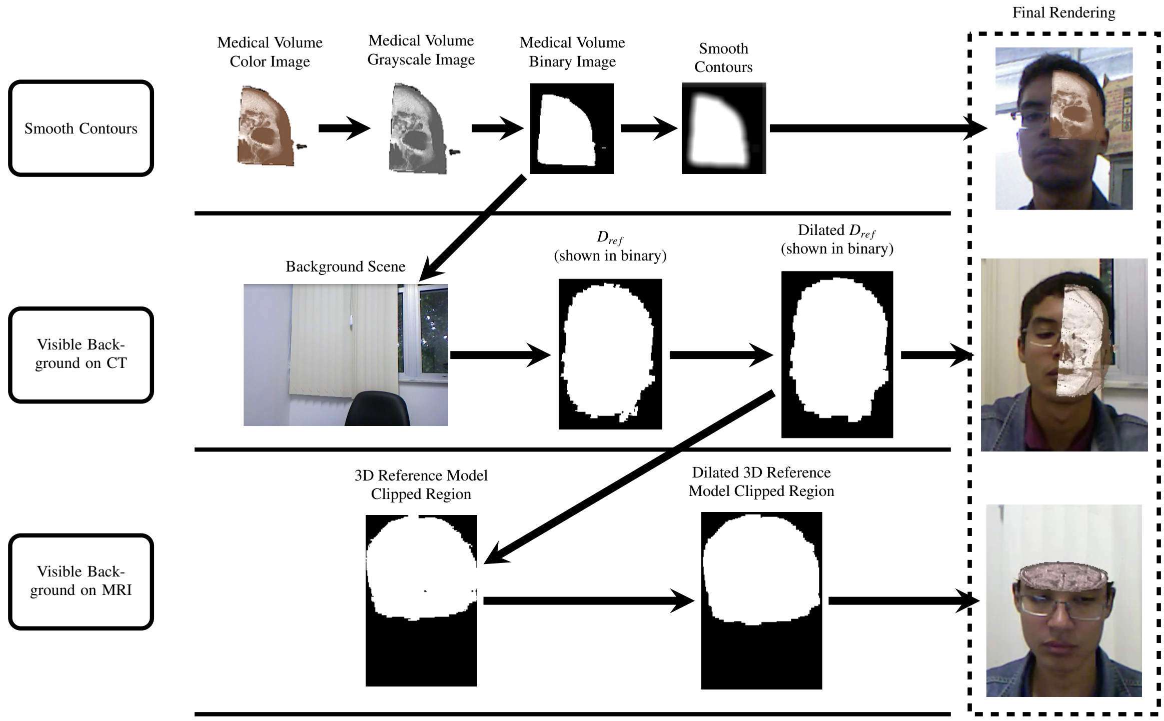

Figure 4: A schematic view of the proposed techniques. Focus plus context (F+C) visualization based on the smooth contours technique (top part): the blurred version of the binary image of the medical volume is used as a mask that smooths the transition between the medical data and the real scene on the final rendering. F+C visualization based on visible background on CT data technique (middle part): from the binary image of the medical volume, the dilated image of the 3D reference model, and the background scene, the soft tissue of the medical data can be displayed merged with the background, emphasizing the visualization of the bone structure. F+C visualization based on the visible background on MRI data technique (bottom part): by rendering a clipped image of the 3D reference model, the organs of the medical data can be displayed in the context of the patient's region of interest.

the volume and separates bone and soft tissue regions, indicating where the background scene must be rendered.

In our case, D_{ref}, the depth map of the 3D reference model, does not overlap perfectly with the patient's ROI. To avoid the presence of artifacts in the final rendering, D_{ref} is dilated only on its contours to preserve the original depth (which is used for occlusion computation) and sent to the shader to represent the patient's ROI.

4.2.3. *Visible background on MRI data*

In volume rendering, MRI data can be used to enable the visualization of internal structures of the patient's anatomy such as organs. In an AR environment, the best way to visualize data of this kind is by clipping not only the medical volume but also the corresponding region of the patient's color image. In this scenario, it is desirable to see the background scene in the region clipped, which is the main goal of the visible background on MRI data technique. An overview of this technique is given in the bottom part of Figure 4.

The technique originally proposed in [5] is similar to the one used for CT data. The background scene is saved. Next, taking advantage of the volumetric representation of KinectFusion, which stores the 3D reference model as an implicit surface in a 3D grid, one can clip the patient's ROI in real time. The algorithm to render an image from the 3D clipped reference model is given in **Algorithm 1**. This algorithm is an extension of the pseudocode presented in [46]. We ray-cast the 3D grid, and when the ray traverses a zero-crossing position (i.e., the silhouette of the 3D reference model stored in the volume) and it is in the clipped region, the voxel's corresponding pixel is rendered in the output image. The medical volume is clipped separately and sent to the shader. The output image from this algorithm is $I_{subtraction}$, which is used with the same objective as described for the visible background on CT data technique. Both $I_{subtraction}$ and D_{ref} are dilated because of the problem of overlapping described before.

The algorithm proposed in [5] is subject to the presence of artifacts at the intersection between the clipping plane and the 3D reference model. To mitigate their effects, we use adaptive sampling to reduce the step size of the ray when it is near the zero-crossing position. We check this proximity by using a specific threshold (t_{prox}) over the truncated signed distance function stored at the voxel g being traversed (g_{tsdf}). When near the zero-crossing, the step size of the ray cast is reduced to the value w_s to perform a more accurate traversal. From empirical tests, we have set $t_{prox} = 0.5$ and w_s equals to one fourth of the original step size. As shown in Section 5, by using this new algorithm, we improved the visual quality of the method while maintaining almost the same performance.

4.3. *Final rendering*

After the volume rendering, the color frame buffer is sent to the shader for blending with the patient's color data coming from the RGB-D sensor. For the CAM and smooth contours

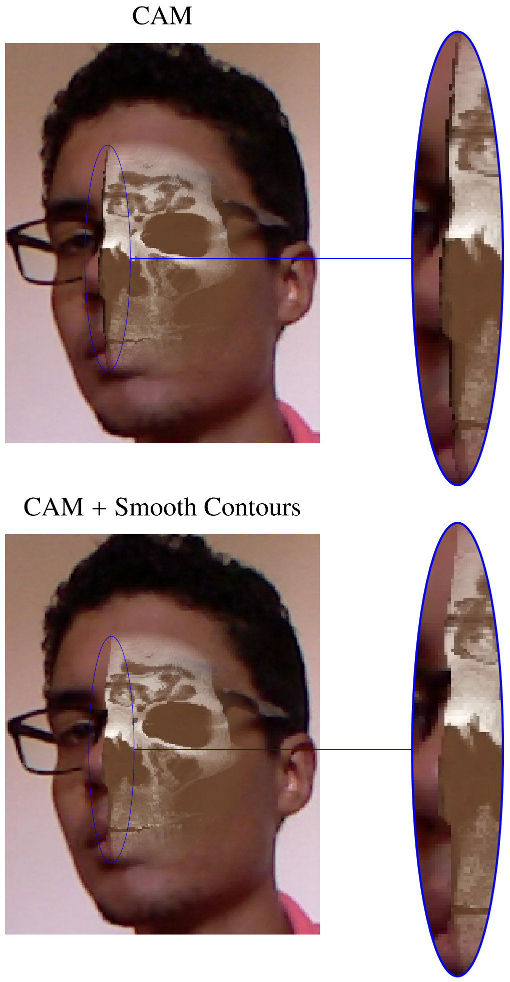

Figure 5: Focus plus context visualization based on the contextual anatomic mimesis (CAM) algorithm (top image) and its extension with the smooth contours technique (bottom image). By using the smooth contours technique, we can suppress the black border resulting from the clipping of the medical data, allowing a more seamless transition between real and virtual images.

techniques, blending is done by the following linear interpolation:

$$I_{final} = \beta I_{real} + (1 - \beta) I_{medical} \tag{1}$$

where I_{real} is the image captured by the sensor, $I_{medical}$ is the image corresponding to the medical volume, and I_{final} is the resulting augmented image. In our approach, β is defined dynamically, for every fragment/pixel, by the F+C visualization techniques mentioned before, according to the following equation:

$$\beta = clamp(max(w_c(1.0 - \alpha_{smoothCont}), \alpha_{CAM})) \tag{2}$$

where *clamp* is a function that clamps the input parameter to the interval $[0, 1]$.

For the visible background–based F+C techniques, the shader listed in **Algorithm 2** is used instead of Equation 1, because this Equation does not include the background rendering.

Algorithm 1 Ray casting the clipped region of the 3D reference model volume

1: **for** each pixel $\mathbf{u} \in$ output image $I_{subtraction}$ **in parallel do**
2: $\quad I_{subtraction}(u) \leftarrow 0$;
3: $\quad ray^{start} \leftarrow$ back project $[\mathbf{u}, 0]$; convert to grid position
4: $\quad ray^{next} \leftarrow$ back project $[\mathbf{u}, 1]$; convert to grid position
5: $\quad ray^{dir} \leftarrow$ normalize $(ray^{next} - ray^{start})$
6: $\quad ray^{len} \leftarrow 0$
7: $\quad g \leftarrow$ first voxel along ray^{dir}
8: $\quad$ **while** voxel g within volume bounds **do**
9: $\quad\quad ray^{len} \leftarrow ray^{len} + stepsize$
10: $\quad\quad g^{prev} \leftarrow g$
11: $\quad\quad g \leftarrow$ traverse next voxel along ray^{dir}
12: $\quad\quad$ **if** $g_{tsdf} < t_{prox}$ **then**
13: $\quad\quad\quad stepsize \leftarrow w_s$
14: $\quad\quad$ **end if**
15: $\quad\quad$ **if** zero crossing from g to g^{prev} and g is in the clipped region **then**
16: $\quad\quad\quad I_{subtraction}(u) \leftarrow 255$;
17: $\quad\quad$ **end if**
18: $\quad$ **end while**
19: **end for**

The algorithm for the visible background on CT data technique can be seen in lines 1-15 and 22-24. The color image captured from the RGB-D sensor is rendered in the region that does not represent the patient's ROI (i.e., where the depth of the 3D reference object is zero, as it was not reconstructed) (lines 2-4). The captured color image is also rendered when the volume is occluded and the occludee has depth (i.e., it is not in a hole region) (lines 5-7). Next, if the fragment is in the subtraction mask region, the volume or the background scene is rendered. Otherwise, the fragment is in the clipped region and the real color image is rendered (lines 23-24). Gray intensity is computed from the volume (by the *gray* function) and assigned to β. Considering that the bone is rendered with a gray level greater than the soft tissue's and than $w_{grayLevel}$, it is rendered without the background scene. Assuming that bone and soft tissue have different gray intensities, $w_{grayLevel}$ can be adjusted to render the bone with its original color and the soft tissue can be linearly interpolated with the background scene (lines 8-15).

The algorithm for the visible background on MRI data technique is shown in lines 1-8 and 16-24. The color image captured from the RGB-D sensor is rendered in the same way as described for the visible background on CT data technique. The main difference here is that if the subtraction mask is active (i.e., the patient's ROI is clipped) and if there are medical data to be visualized, they are rendered. Otherwise, the background image is rendered.

In an AR environment, it is desirable to solve the problem of occlusion between virtual and real data. For a specific viewpoint, depth images of the patient's 3D reference model D_{ref} and the 3D object coming from the sensor's live stream D_{live} are used to solve this issue. If the depth from D_{live} is lower than that from D_{ref}, the object captured by the sensor is in front of the reference object and the medical volume is the ocludee, otherwise, the medical volume is the occluder.

Algorithm 2 Focus plus context visualization based on the visible background

for in parallel do
  if $D_{ref} == 0.0$ **then**
    return I_{real};
  end if
  if $D_{live} < D_{ref}$ and $D_{live} \mathrel{!}= 0.0$ **then**
    return I_{real};
  end if
  if $I_{subtraction} == 1.0$ **then**
    if CT data **then**
      $grayLevel \leftarrow gray(I_{medical})$;
      $\beta \leftarrow grayLevel$;
      if $grayLevel < w_{grayLevel}$ **then**
        return $\beta I_{background} + (1-\beta) I_{medical}$;
      end if
      return $I_{medical}$;
    else
      if $I_{medical} == 0.0$ **then**
        return $I_{background}$;
      end if
      return $I_{medical}$;
    end if
  end if
  return I_{real};
end for

5. Experimental results

In this section, the performance and visual quality of the F+C visualization techniques based on volume clipping are evaluated.

5.1. Experimental setup

For all tests, as the computer we used an Intel Core™ i7-3770K CPU (3.50 GHz), 8GB RAM, and a NVIDIA GeForce GTX 660 graphics card. For 3D reference model reconstruction, we used the open-source C++ implementation of KinectFusion released by the Point Cloud Library project [64].

We use a Microsoft Kinect device as a low-cost, accessible, and versatile RGB-D sensor [65]. The medical dataset used was a CT volumetric dataset of a head released by the Visible Human Project [66] of resolution $128 \times 256 \times 256$, an MRI volumetric dataset of a head from MRI Head available in Volume Library [67] of resolution 256^3, and an MRI dataset of a knee of resolution $400 \times 400 \times 250$ and a CT dataset of a torso of resolution $512 \times 512 \times 288$, both available in OsiriX [68]. The 3D reference models were reconstructed with KinectFusion with a grid with resolution of 512^3.

5.2. Performance evaluation

In our preprocessing computation, the 3D reference model was reconstructed at 40 FPS. From empirical tests, the user takes less than 10 s to place the volume into the scene and adjust it. The markerless live tracking and volume rendering techniques together run at 45 FPS. These performance results are the same as those reported in previous work [5]; however, they were computed without taking into consideration the depth sensor's performance[1].

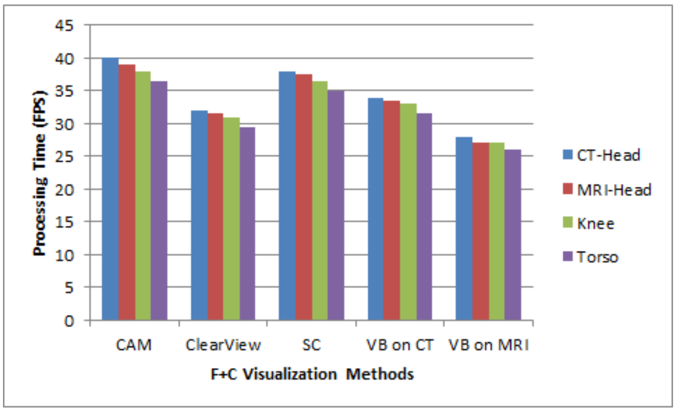

Figure 6: Performance results measured in frames per second (FPS) for each one of the focus plus context (F+C) visualization techniques discussed in this article. CAM, contextual anatomic mimesis; SC - smooth contours; VB - visible background.

The performance of the on-patient medical data visualization for the F+C visualization techniques used in this work can be seen in Figure 6. The performance was evaluated for all four medical datasets described in Section 5.1.

From Figure 6, we see that the CAM technique provides the best performance, which is expected since this technique does small computations directly on the shader. For ClearView, an F+C visualization technique proposed specifically for volume rendering [30], we generated only one context layer and one focus layer in our application. In the context layer, we computed three layers (i.e., position, normal and curvature) for the medical volume. For the focus layer, we rendered an isosurface from the medical volume, according to a user-defined iso-value. The layers were recalculated for every change in viewpoint. In our application, ClearView requires approximately 6.25 ms to compute and render these layers, which are composed according to the distance-based importance shader [30], decreasing the application's performance to a frame rate even lower than that provided by the smooth contours and visible background on CT data techniques. For the smooth contours technique, by transferring all the pipeline to the GPU, we obtained a huge improvement over the original technique proposed in [5], which achieved 20 FPS (for CT-Head in Figure 6) on the same hardware. The visible background on CT data technique runs in full real time because it operates mostly on the shader. Dilation applied on D_{ref} decreases the application's performance slightly. The visible background on MRI data technique is slower than the other techniques because of the ray casting performed on the 3D reference model to render the clipped patient's ROI.

[1]The Kinect sensor acquires depth data at 30 FPS; hence, this limits the maximum performance of the application.

However, differently from [5], we added an adaptive sampling scheme to improve the visual quality of the approach. This adaptive approach allowed us to obtain the same performance as the original technique. Moreover, all of the techniques run at more than 25 FPS, therefore in real time, even for the medical dataset of highest resolution (Torso in Figure 6).

5.3. *Visual quality evaluation*

By the use of the shader proposed in **Algorithm 2**, occlusion is supported by our application, as can be seen in Figure 7.

Figure 7: Occlusion support is achieved by comparing depth values from current and previous depth frames.

For the F+C visualization based on the smooth contours technique, the level of smoothness can be controlled by the parameter w_c. As can be seen in Figure 8, the transition between the volume and the real scene becomes smoother as w_c increases. At the same time, the volume contours become less visible.

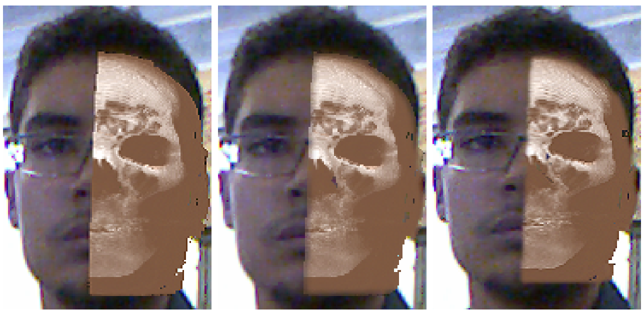

Figure 8: Influence of the parameter w_c in the smooth contours technique. Left image: $w_c = 0$. Middle image: $w_c = 2$. Right image: $w_c = 4$.

For the F+C visualization based on the visible background on CT data technique, bone and soft tissue structures can be separated with use of $w_{grayLevel}$. From Figure 9, it can be seen that by changing this parameter, we can render the volume without the background scene, with the soft tissue linearly interpolated with the background scene or almost completely invisible.

For the F+C visualization based on the visible background on MRI data technique, we have proposed an improvement to mitigate artifacts resulting from clipping of the patient's ROI [5]. A

Figure 9: Influence of $w_{grayLevel}$ in the visible background on CT data technique. Top-left image: $w_{grayLevel} = 0$. Top-right image $w_{grayLevel} = 0.5$. Bottom-left image: $w_{grayLevel} = 0.75$. Bottom-right image: $w_{grayLevel} = 1$.

visual comparison between our approach and the one proposed in [5] can be seen in Figure 10. As the artifacts become more visible during the user's movement, this figure shows the patient's ROI in different positions and the presence of artifacts in these scenarios. Moreover, regions around the contours of the clipped data are zoomed to enable a clear visualization of the problems of related work [5] in comparison with the improvements proposed here. Artifacts at the intersection between the patient's ROI and the clipping plane are more visible when the user rotates his or her head in front of the sensor [5]. From column I iin Figure 10, we can see artifacts arising at the contours. By use of our approach (Figure 10, column II), artifacts are mitigated and the results are comparable to a scenario (Figure 10, column III) where the ray is cast in a uniform sampling way and the step size of the ray is too small to render the clipped data in an AR application. In this case, our method has better performance than the best visual quality scenario, as ours runs at 28 FPS, whereas because of its use of ray casting with a small step size, the ground-truth approach runs at only 9 FPS, which does not provide performance that is enough for an interactive application [69].

Our MAR environment supports not only rendering of the head, but also rendering of other ROI in the patient's body. The on-patient visualization of the torso and knee datasets with the CAM technique is shown in Figure 11, and can be found in the

Figure 10: Different schemes for focus plus context visualization based on the visible background on MRI Data technique. Column I results from the application of the original technique proposed in [5], column II refers to the adaptive scheme proposed in this article, and column III represents a ground-truth scenario where the ray casting performs uniform sampling and the step size of the ray is too small to render the clipped data in an interactive application. Our adaptive approach (column II) is three times faster than the ground-truth scenario (column III), while achieving almost the same visual quality. For each image, we zoom in on the contours of the clipped data to highlight the differences between the different approaches. Furthermore, red arrows are used to show regions where the visual difference is apparent. The presence of alias in the zoomed images is due to the digital zoom.

supplementary video. Even with different ROI, our MAR environment tracks the 3D reference model and shows the medical data at the position of the patient's anatomy. As the torso comprises mainly the abdomen and the pelvis, we have found it useful to show them separately.

The process to augment these structures over the patient's body is almost the same as the one described in the previous sections. The small adaptations required to make the augmentation of other structures possible are discussed in Section 6.

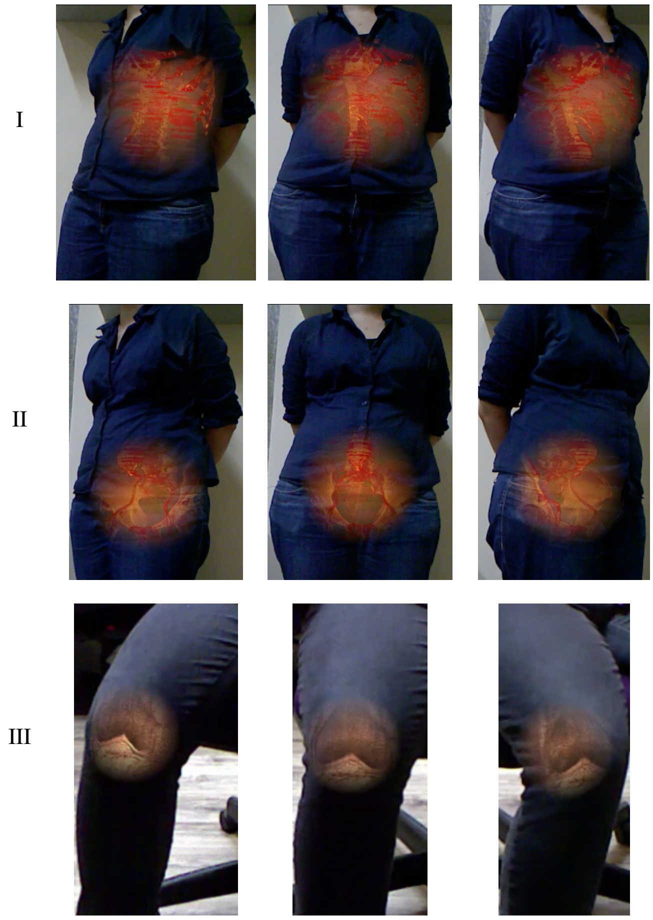

Figure 11: On-patient visualization of CT torso (rows I and II) and MRI knee (row III) datasets. To highlight the features of the CT torso dataset, we show the abdomen (row I) and pelvis (row II) separately.

Depending on the transfer function and/or the structure of the medical dataset, the ray casting technique may render holes in the final image (Figure 12, left image). In this case, the F+C techniques handle the holes in different ways, achieving different results for the final rendering. An example of this can be seen in the right image in Figure 12, where the smooth contours technique replaces the virtual background color of the hole by the color captured by the RGB-D sensor and smooths the contours around the hole region. The visible background on CT data technique replaces the virtual background color of the hole by the background color of the real scene, which was captured previously.

5.4. Accuracy evaluation

In all the experiments, the patient's ROI is augmented with a generic volumetric dataset. The use of a generic volume does not affect our visual quality evaluation since the volume is scaled and positioned semiautomatically with the user's fine adjustments [17]. In this way, the accuracy of the registration between the medical data and the patient's ROI depends on the quality of the user's fine adjustment because of the use of a generic volume.

Related to the accuracy of the MAR environment, 3D reconstruction has accuracy of approximately 10 mm [48], and by using the hierarchical ICP algorithm, we improved the live tracking accuracy from approximately 3 mm [5] to approximately 2 mm, according to the point-to-plane error metric [49]. In this environment, tracking error does not accumulate between frames.

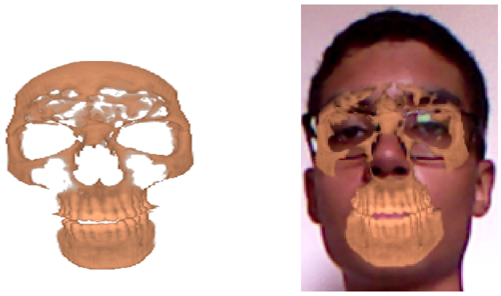

Figure 12: Medical volume with holes (left) rendered into the augmented scene with the smooth contours plus contextual anatomic mimesis (CAM) technique (right). For this figure, CAM's fall-off weight was set to 0.

6. Discussion

As mentioned in Section 1, inspired by the field of on-patient craniofacial data visualization, we evaluated performance and visual quality of the proposed techniques in a scenario where the patient's ROI consists of the patient's head. In other contexts, where the ROI can be another part of the body, such as the abdomen, pelvis and knee (Figure 11), one can still use the solution presented in this article with minor adaptations. However, the MAR environment may still require some additional changes to reconstruct and track the poses. The main problems related to these adaptations rely on the segmentation, tracking and reconstruction of the patient's ROI.

To segment the patient's ROI in the scene, we propose the use of a classification algorithm to detect and segment it from the color image. This solution is desirable for a few ROI, such as the head and hand. For others, which do not have classification algorithms available to perform such a task, one solution is to position the ROI relatively distant from the background scene and segment it in the depth image with background segmentation through z-axis thresholding. For the scenarios shown in Figure 11, we used this strategy based on depth segmentation to remove the background scene.

Depth-based tracking algorithms (e.g., ICP) are dependent on the presence of geometric information on the scene [70]. Some ROI, such as the arm and leg, do not have much variation in the depth values captured by the sensor between different viewpoints. In this case, a texture-based tracking algorithm which operates according to the features of the color image [71, 72] can be used to improve tracking accuracy. Furthermore, markerless tracking uses geometric data of part of the real scene as a natural marker. The natural marker (i.e., in our case the patient's ROI) may suffer nonrigid motion if it is a deformable object. For the patient's face or hand, for instance, it is desirable for the tracking algorithm to support non-rigid interaction between the patient and the application. Despite the complexity of nonrigid registration, there are some methods which provide real-time performance [73, 74] and can be used together with the markerless rigid tracking used in this work to improve accuracy and robustness for tracking of deformable structures. In all tests reported in this article, we used only the ICP algorithm for tracking.

For the hand and foot, ROI which contain smaller structures (e.g. fingers), more accurate 3D reconstruction algorithms [75, 73] may be required to reconstruct a 3D reference model which captures the finest details of the patient's ROI, hence enabling high-quality tracking and occlusion handling even in these smaller structures. In this situation, the KinectFusion algorithm is able to reconstruct acceptable 3D reference models for such ROI, although showing some artifacts which can have some impact on tracking accuracy. In this work, we used only the KinectFusion algorithm to reconstruct the 3D reference models.

As can be seen in the supplementary video, the use of a 3D reference model as a basis for markerless registration allows tracking of the medical data not only when the center of rotation is located at the position of the patient's anatomy. For knee visualization (Figure 11), the center of rotation is located mainly in the torso of the user, which is translated in relation to the knee region. Even in this case, our MAR environment supports the tracking of the medical data into the augmented scene.

By using the hierarchical tracking algorithm, we improved tracking accuracy, as mentioned in Section 5. The advantage of this improvement is twofold: for the reconstruction of the 3D reference model, in which the viewpoints captured by the depth sensor are rigidly aligned with more accuracy, resulting in a more accurate 3D reference model reconstruction; for the AR tracking, giving more tracking stability and less misalignment between real and virtual objects.

As already known in the field of AR, tracking technologies may suffer from jittering. As can be seen in the supplementary video, even when we used the hierarchical ICP algorithm together with the head pose estimation solution to improve tracking accuracy and robustness, the MAR environment is still prone to jittering when the user moves his or her ROI in front of the depth sensor. To minimize the jittering, one can increase the number of ICP iterations to trade off tracking accuracy and performance or change the tracking algorithm for another one which can explicitly handle such a problem. Hence, we empirically have found it useful to use only three iterations of the hierarchical ICP algorithm (i.e., one for each level of the pyramid), prioritizing performance over accuracy.

Markerless tracking solutions are not as accurate as some commercial marker-based solutions. When developing a medical AR application, one must decide carefully which of these technologies to use. Although being intrusive in the scene, marker-based solutions may ease the positioning of the medical data into the scene and the tracking of the medical data with high accuracy, being recommend for medical applications that deal with surgery, as done in [2]. Markerless solutions are not very accurate, and are therefore recommended for applications which demand visually appealing results for the composition of real and virtual data, typically medical applications developed for training or visualization purposes [7].

For on-patient medical data visualization, the proposed application supports CT and MRI data, but can be easily extended to support other scanning data as well. Medical data with resolution higher than 512^3 can be used for a cadaver or phantom study. For such scenarios, the only difference is that the ROI is static in the scene and it is the sensor that must be moved to capture different viewpoints and reconstruct a single 3D reference model. We have evaluated the F+C techniques only for an in-vivo study with different users as a patient.

In AR applications, one must pay attention to the way in which the virtual content will be visualized in the augmented scene. In this article, we have described three techniques to improve the depth perception in medical AR scenarios. The artifacts present in the visible background on MRI data technique proposed in [5] decrease the quality of the final image in the region of the clipped medical data. This problem is even severer because of the high spatial and temporal discontinuity of the artifacts. Through the use of an adaptive scheme where the ray casting samples more voxels only at the location of the clipping plane, we achieved high-quality images (Figure 10, column II), almost indistinguishable from the ground-truth images shown in column III in Figure 10.

The techniques for F+C visualization based on the visible background do not support the visualization of real dynamic background scenes. In this case, we cannot use the color camera of the RGB-D sensor because the patient occludes part of the background being captured. A multiview approach, in which an additional webcam is used to capture the real background scene, may solve this problem.

7. Conclusion and future Work

We have presented improvements for on-patient medical data visualization by using F+C visualization and volume clipping. The performance and visual quality of the proposed techniques were evaluated, and from the tests conducted, we conclude that they are capable of running in real time and improve the visual quality of the final scene. To further enhance the quality of the integration of the virtual data into the augmented scene, occlusion is handled and tracking accuracy is improved. Finally, we have shown that our approach is versatile such that it can be used for different ROI of the patient.

In future work, we intend to evaluate the full solution (MAR environment and F+C techniques) in a real medical training environment, where high accuracy is not required for the application. Further, an in-depth study must be conducted to improve accuracy for scenarios where the medical dataset of the patient must be used for the on-patient visualization.

With feedback from specialists, we will be able to improve or adapt the methods where needed or even to collect a database of craniofacial data to further improve future tests and evaluations of our approach.

For all the F+C visualization techniques proposed in this article, quantitative evaluation and extensive user study must be conducted to validate the proposed techniques from the perspective of the final users.

For the AR environment, we used a conventional display to show the augmented scene. Multiview solutions based on AR glasses or portable solutions based on mobile devices can be used, where the proposed approach is performed on a server and the visualization of the augmented content is transferred to those alternative hardware devices, allowing a seamless visualization of the virtual content on the real scene.

The markerless tracking algorithm fails if the patient's ROI is not visible in the view of the RGB-D sensor and the algorithm does not support relocalization nor nonrigid registration of the 3D reference model. These features must be supported to further enhance the accuracy and robustness of the tracking.

Acknowledgments

We are grateful to the Point Cloud Library project for providing the open-source implementation of the KinectFusion algorithm. We are also grateful to Gabriele Fanelli for providing the open-source implementation of her real-time head pose estimation algorithm. We thank Rafaela Souza for being the model in some experiments of this study. This research is financially supported by Fundação de Amparo à Pesquisa do Estado da Bahia (FAPESB) and Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES).

[1] Azuma R, Baillot Y, Behringer R, Feiner S, Julier S, MacIntyre B. Recent advances in augmented reality. IEEE Comput Graph Appl 2001;21(6):34–47. doi:10.1109/38.963459.

[2] Bichlmeier C, Wimmer F, Heining SM, Navab N. Contextual anatomic mimesis hybrid in-situ visualization method for improving multi-sensory depth perception in medical augmented reality. ISMAR '07; Washington, DC, USA: IEEE. ISBN 978-1-4244-1749-0; 2007, p. 1–10.

[3] Kutter O, Aichert A, Bichlmeier C, Traub J, Heining SM, Ockert B, et al. Real-time Volume Rendering for High Quality Visualization in Augmented Reality. In: AMI-ARCS 2008. New York, USA: MICCAI Society; 2008,.

[4] Wieczorek M, Aichert A, Kutter O, Bichlmeier C, Landes J, Heining SM, et al. GPU-accelerated Rendering for Medical Augmented Reality in Minimally-Invasive Procedures. In: BVM 2010. Springer; 2010,.

[5] Macedo MCdF, Apolinario AL. Improving on-patient medical data visualization in a markerless augmented reality environment by volume clipping. In: Graphics, Patterns and Images (SIBGRAPI), 2014 27th SIBGRAPI Conference on. 2014, p. 149–56. doi:10.1109/SIBGRAPI.2014.33.

[6] Card SK, Mackinlay JD, Shneiderman B, editors. Readings in Information Visualization: Using Vision to Think. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1999. ISBN 1-55860-533-9.

[7] Debarba HG, Grandi J, Maciel A, Zanchet D. Anatomic hepatectomy planning through mobile display visualization and interaction. In: MMVR; vol. 173. IOS Press. ISBN 978-1-61499-022-2; 2012, p. 111–5.

[8] Kato H, Billinghurst M. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In: Augmented Reality, 1999. (IWAR '99) Proceedings. 2nd IEEE and ACM International Workshop on. 1999, p. 85 –94. doi:10.1109/IWAR.1999.803809.

[9] Maier-Hein L, Franz AM, Fangerau M, Schmidt M, Seitel A, Mersmann S, et al. Towards mobile augmented reality for on-patient visualization of medical images. In: Bildverarbeitung für die Medizin. Informatik Aktuell; Springer. ISBN 978-3-540-19335-4; 2011, p. 389–93.

[10] Suenaga H, Hoang Tran H, Liao H, Masamune K, Dohi T, Hoshi K, et al. Real-time in situ three-dimensional integral videography and surgical navigation using augmented reality: a pilot study. International Journal of Oral Science 2013;(2):98–102.

[11] Blum T, Kleeberger V, Bichlmeier C, Navab N. mirracle: Augmented reality in-situ visualization of human anatomy using a magic mirror. In: VR, 2012 IEEE. 2012, p. 169–70. doi:10.1109/VR.2012.6180909.

[12] Meng M, Fallavollita P, Blum T, Eck U, Sandor C, Weidert S, et al. Kinect for interactive ar anatomy learning. In: Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on. 2013, p. 277–8. doi:10.1109/ISMAR.2013.6671803.

[13] Stefan P, Wucherer P, Oyamada Y, Ma M, Schoch A, Kanegae M, et al. An ar edutainment system supporting bone anatomy learning. In: Virtual Reality (VR), 2014 iEEE. 2014, p. 113–4. doi:10.1109/VR.2014.6802077.

[14] Mercier-Ganady J, Lotte F, Loup-escande E, Marchal M, Lecuyer A. The mind-mirror: See your brain in action in your head using eeg and augmented reality. In: Virtual Reality (VR), 2014 iEEE. 2014, p. 33–8. doi:10.1109/VR.2014.6802047.

[15] Macedo MC, Apolinario AL, Souza AC, Giraldi GA. A semi-automatic markerless augmented reality approach for on-patient volumetric medical data visualization. In: Virtual and Augmented Reality (SVR), 2014 XVI Symposium on. 2014, p. 63–70. doi:10.1109/SVR.2014.29.

[16] Macedo M, Almeida C, Souza A, Silva J, Apolinario A, Giraldi G. A Markerless Augmented Reality Environment for Medical Data Visualization. In: Workshop de Informatica Medica. WIM; Brazil; 2014,.

[17] Macedo M, Apolinario A, Souza AC, Giraldi GA. High-Quality On-Patient Medical Data Visualization in a Markerless Augmented Reality Environment. Journal on 3D Interactive Systems 2014;5:41–52.

[18] Sandor C, Cunningham A, Dey A, Mattila VV. An augmented reality x-ray system based on visual saliency. In: ISMAR. IEEE Computer Society; 2010, p. 27–36. doi:10.1109/ISMAR.2010.5643547.

[19] Avery B, Sandor C, Thomas B. Improving spatial perception for augmented reality x-ray vision. In: Virtual Reality Conference, 2009. VR 2009. IEEE. 2009, p. 79–82. doi:10.1109/VR.2009.4811002.

[20] Mendez E, Feiner S, Schmalstieg D. Focus and context in mixed reality by modulating first order salient features. SG'10; Berlin, Heidelberg: Springer-Verlag. ISBN 3-642-13543-9, 978-3-642-13543-9; 2010, p. 232–43.

[21] Kalkofen D, Veas E, Zollmann S, Steinberger M, Schmalstieg D. Adaptive ghosted views for augmented reality. In: Mixed and Augmented Reality (ISMAR), 2013 IEEE International Symposium on. Adelaide, SA, Australia: IEEE Computer Society; 2013, p. 1–9. doi:10.1109/ISMAR.2013.6671758.

[22] Schall G, Mendez E, Kruijff E, Veas E, Junghanns S, Reitinger B, et al. Handheld augmented reality for underground infrastructure visualization. Personal Ubiquitous Comput 2009;13(4):281–91. doi:10.1007/s00779-008-0204-5.

[23] Chen J, Granier X, Lin N, Peng Q. On-line visualization of underground structures using context features. In: Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology. VRST '10; New York, NY, USA: ACM. ISBN 978-1-4503-0441-2; 2010, p. 167–70. doi:10.1145/1889863.1889898.

[24] Padilha A, Rolim C, Teichrieb V. The ghosting technique applied to augmented reality visualization. In: Virtual and Augmented Reality (SVR), 2013 XV Symposium on. 2013, p. 159–66. doi:10.1109/SVR.2013.36.

[25] Mendez E, Schmalstieg D. Importance masks for revealing occluded objects in augmented reality. In: Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology. VRST '09; New York, NY, USA: ACM. ISBN 978-1-60558-869-8; 2009, p. 247–8.

[26] Lee CH, Varshney A, Jacobs DW. Mesh saliency. ACM Trans Graph 2005;24(3):659–66. doi:10.1145/1073204.1073244.

[27] Bruckner S, Grimm S, Kanitsar A, Gröller ME. Illustrative context-preserving volume rendering. In: Proceedings of the Seventh Joint Eurographics / IEEE VGTC Conference on Visualization. EUROVIS'05; Aire-la-Ville, Switzerland, Switzerland: Eurographics Association. ISBN 3-905673-19-3; 2005, p. 69–76. doi:10.2312/VisSym/EuroVis05/069-076.

[28] Bruckner S, Grimm S, Kanitsar A, Groller M. Illustrative context-preserving exploration of volume data. Visualization and

Computer Graphics, IEEE Transactions on 2006;12(6):1559–69. doi:10.1109/TVCG.2006.96.

[29] Sikachev P, Rautek P, Bruckner S, Gröller ME. Dynamic focus+context for volume rendering. In: Proceedings of Vision, Modeling and Visualization 2010. University of Siegen, Siegen, Germany; 2010, p. 331–8.

[30] Kruger J, Schneider J, Westermann R. Clearview: An interactive context preserving hotspot visualization technique. Visualization and Computer Graphics, IEEE Transactions on 2006;12(5):941–8. doi:10.1109/TVCG.2006.124.

[31] Kirmizibayrak C, Wakid M, Yim Y, Hristov D, Hahn JK. Interactive focus + context medical data exploration and editing. Computer Animation and Virtual Worlds 2014;25(2):129–41. doi:10.1002/cav.1538.

[32] Röttger D, Merhof D, Müller S. The BundleExplorer: A Focus and Context Rendering Framework for Complex Fiber Distributions. In: Proceedings of the Eurographics Conference on Visual Computing for Biology and Medicine. EG VCBM'12; 2012, p. 1–8. doi:10.2312/VCBM/VCBM12/001-008.

[33] Gasteiger R, Neugebauer M, Beuing O, Preim B. The flowlens: A focus-and-context visualization approach for exploration of blood flow in cerebral aneurysms. Visualization and Computer Graphics, IEEE Transactions on 2011;17(12):2183–92. doi:10.1109/TVCG.2011.243.

[34] Abellan P, Puig A, Ayala D. Focus + context rendering of structured biomedical data. In: VCBM'08. 2008, p. 109–16.

[35] Schulte zu Berge C, Baust M, Kapoor A, Navab N. Predicate-based focus-and-context visualization for 3d ultrasound. Visualization and Computer Graphics, IEEE Transactions on 2014;PP(99):1–. doi:10.1109/TVCG.2014.2346317.

[36] Kersten-Oertel M, Jannin P, Collins D. Dvv: A taxonomy for mixed reality visualization in image guided surgery. Visualization and Computer Graphics, IEEE Transactions on 2012;18(2):332–52. doi:10.1109/TVCG.2011.50.

[37] Lerotic M, Chung AJ, Mylonas G, Yang GZ. Pq-space based non-photorealistic rendering for augmented reality. In: Proceedings of the 10th International Conference on Medical Image Computing and Computer-assisted Intervention. MICCAI'07; Berlin, Heidelberg: Springer-Verlag. ISBN 3-540-75758-9, 978-3-540-75758-0; 2007, p. 102–9.

[38] Pratt P, Mayer E, Vale J, Cohen D, Edwards E, Darzi A, et al. An effective visualisation and registration system for image-guided robotic partial nephrectomy. Journal of Robotic Surgery 2012;6(1):23–31. doi:10.1007/s11701-011-0334-z.

[39] Bichlmeier C, Heining S, Feuerstein M, Navab N. The virtual mirror: A new interaction paradigm for augmented reality environments. Medical Imaging, IEEE Transactions on 2009;28(9):1498–510. doi:10.1109/TMI.2009.2018622.

[40] Kersten-Oertel M, Chen SS, Collins D. An evaluation of depth enhancing perceptual cues for vascular volume visualization in neurosurgery. Visualization and Computer Graphics, IEEE Transactions on 2014;20(3):391–403. doi:10.1109/TVCG.2013.240.

[41] Steenblik RA. The chromostereoscopic process: A novel single image stereoscopic process. vol. 0761. 1987, p. 27–34. URL: `http://dx.doi.org/10.1117/12.940117`. doi:10.1117/12.940117.

[42] Viola P, Jones MJ. Robust real-time face detection. Int J Comput Vision 2004;57(2):137–54. doi:10.1023/B:VISI.0000013087.49260.fb.

[43] Tomasi C, Manduchi R. Bilateral filtering for gray and color images. In: Computer Vision, 1998. Sixth International Conference on. 1998, p. 839 –46. doi:10.1109/ICCV.1998.710815.

[44] Burt PJ. Fast filter transform for image processing. Computer Graphics and Image Processing 1981;16(1):20 – 51. doi:http://dx.doi.org/10.1016/0146-664X(81)90092-7.

[45] Adelson EH, Anderson CH, Bergen JR, Burt PJ, Ogden JM. Pyramid methods in image processing. RCA Engineer 1984;29(6):33–41.

[46] Izadi S, Kim D, Hilliges O, Molyneaux D, Newcombe R, Kohli P, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology. UIST '11; USA: ACM. ISBN 978-1-4503-0716-1; 2011, p. 559–68. doi:10.1145/2047196.2047270.

[47] Curless B, Levoy M. A volumetric method for building complex models from range images. In: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '96; New York, NY, USA: ACM. ISBN 0-89791-746-4; 1996, p. 303–12. doi:10.1145/237170.237269.

[48] Meister S, Izadi S, Kohli P, Hämmerle M, Rother C, Kondermann D. When can we use kinectfusion for ground truth acquisition? In: IROS. IEEE Computer Society; 2012,.

[49] Rusinkiewicz S, Levoy M. Efficient variants of the icp algorithm. In: Third International Conference on 3D Digital Imaging and Modeling (3DIM). 2001, p. 145–52.

[50] Newcombe RA, Izadi S, Hilliges O, Molyneaux D, Kim D, Davison AJ, et al. Kinectfusion: Real-time dense surface mapping and tracking. In: Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on. 2011, p. 127–36. doi:10.1109/ISMAR.2011.6092378.

[51] Fanelli G, Weise T, Gall J, Gool LV. Real time head pose estimation from consumer depth cameras. In: Proceedings of the 33rd International Conference on Pattern Recognition. DAGM'11; Berlin, Heidelberg: Springer-Verlag. ISBN 978-3-642-23122-3; 2011, p. 101–10.

[52] Macedo M, Apolinario A, Souza A. A robust real-time face tracking using head pose estimation for a markerless ar system. In: Virtual and Augmented Reality (SVR), 2013 XV Symposium on. 2013, p. 224–7. doi:10.1109/SVR.2013.12.

[53] Hadwiger M, Kniss JM, Rezk-salama C, Weiskopf D, Engel K. Real-time Volume Graphics. USA: A. K. Peters, Ltd.; 2006. ISBN 1568812663.

[54] Hadwiger M, Ljung P, Salama CR, Ropinski T. Advanced illumination techniques for gpu-based volume raycasting. SIGGRAPH '09; New York, NY, USA: ACM; 2009, p. 2:1–2:166.

[55] Sigg C, Hadwiger M. Fast third-order texture filtering. In: Pharr M, editor. GPU Gems 2. Addison-Wesley; 2005, p. 313–29.

[56] Ruijters D, ter Haar Romeny BM, Suetens P. Efficient gpu-based texture interpolation using uniform b-splines. J Graphics Tools 2008;13(4):61–9.

[57] Li W, Mueller K, Kaufman A. Empty space skipping and occlusion clipping for texture-based volume rendering. In: VIS 2003. IEEE Computer Society; 2003, p. 317–24.

[58] Engel K, Kraus M, Ertl T. High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. HWWS '01; New York, NY, USA: ACM. ISBN 1-58113-407-X; 2001, p. 9–16.

[59] Blinn JF. Models of light reflection for computer synthesized pictures. SIGGRAPH '77; New York, NY, USA: ACM; 1977, p. 192–8.

[60] Ruijters D, Thévenaz P. Gpu prefilter for accurate cubic b-spline interpolation. Comput J 2012;55(1):15–20. doi:10.1093/comjnl/bxq086.

[61] Wu ST, Yauri Vidalon J, Souza Loos W, Coan A. Query tools for interactive exploration of 3d neuroimages: Cropping, probe and lens. In: Graphics, Patterns and Images (SIBGRAPI), 2013 26th SIBGRAPI - Conference on. 2013, p. 250–7. doi:10.1109/SIBGRAPI.2013.42.

[62] Otsu N. A Threshold Selection Method from Gray-level Histograms. SMC 1979;9(1):62–6. doi:10.1109/tsmc.1979.4310076.

[63] Suzuki S. Topological structural analysis of digitized binary images by border following. CVGIP 1985;30(1):32 – 46. doi:http://dx.doi.org/10.1016/0734-189X(85)90016-7.

[64] Rusu R, Cousins S. 3d is here: Point cloud library (pcl). In: ICRA. IEEE Computer Society; 2011, p. 1 –4. doi:10.1109/ICRA.2011.5980567.

[65] Cruz L, Lucio D, Velho L. Kinect and rgbd images: Challenges and applications. In: Graphics, Patterns and Images Tutorials (SIBGRAPI-T), 2012 25th SIBGRAPI Conference on. 2012, p. 36–49. doi:10.1109/SIBGRAPI-T.2012.13.

[66] Visible human data. `http://www.nlm.nih.gov/research/visible/`; 2014. Accessed 22 September 2014.

[67] Volume library. `http://www9.informatik.uni-erlangen.de/External/vollib/`; 2014. Acessed 22 September 2014.

[68] Osirix. `http://www.osirix-viewer.com/datasets/`; 2015. Acessed 26 May 2015.

[69] Akenine-Moller T, Moller T, Haines E. Real-Time Rendering. 2nd ed.; Natick, MA, USA: A. K. Peters, Ltd.; 2002. ISBN 1568811829.

[70] Peasley B, Birchfield S. Replacing projective data association with lucas-kanade for kinectfusion. In: Robotics and Automation (ICRA), 2013 IEEE International Conference on. 2013, p. 638–45. doi:10.1109/ICRA.2013.6630640.

[71] Lucas BD, Kanade T. An iterative image registration technique with an application to stereo vision. In: Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2. IJCAI'81; San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1981, p. 674–9.

[72] Horn BK, Schunck BG. Determining optical flow. Artificial Intel-

ligence 1981;17(1–3):185 – 203. doi:http://dx.doi.org/10.1016/0004-3702(81)90024-2.

[73] Zollhöfer M, Nießner M, Izadi S, Rhemann C, Zach C, Fisher M, et al. Real-time non-rigid reconstruction using an rgb-d camera. ACM Transactions on Graphics (TOG) 2014;33(4).

[74] Souza ACS, Macedo MCF, Apolinário Jr. AL. Multi-frame adaptive non-rigid registration for markerless augmented reality. In: Proceedings of the 13th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry. VRCAI '14; New York, NY, USA: ACM. ISBN 978-1-4503-3254-5; 2014, p. 7–16. doi:10.1145/2670473.2670486.

[75] Nießner M, Zollhöfer M, Izadi S, Stamminger M. Real-time 3d reconstruction at scale using voxel hashing. ACM Transactions on Graphics (TOG) 2013;.