

Occlusion Handling in Augmented Reality: Past, Present and Future

Márcio C. F. Macedo and Antônio L. Apolinário Jr.

Abstract—One of the main goals of many augmented reality applications is to provide a seamless integration of a real scene with additional virtual data. To fully achieve that goal, such applications must typically provide high-quality real-world tracking, support real-time performance and handle the mutual occlusion problem, estimating the position of the virtual data into the real scene and rendering the virtual content accordingly. In this survey, we focus on the occlusion handling problem in augmented reality applications and provide a detailed review of 161 papers published in this field between January 1992 and August 2020. To do so, we present a historical overview of the most common strategies employed to determine the depth order between real and virtual objects, to visualize hidden objects in a real scene, and to build occlusion-capable visual displays. Moreover, we look at the state-of-the-art techniques, highlight the recent research trends, discuss the current open problems of occlusion handling in augmented reality, and suggest future directions for research.

Index Terms—Computer Graphics, Augmented Reality, Mutual Occlusion, X-ray Vision, Computational Displays, Depth Maps.

1 INTRODUCTION

UGMENTED Reality (AR) is a technology whose main goal is to augment the real world with additional virtual content, such as an image, object, audio, or other type of non-visual media, while keeping real-time user interactivity. To do so convincingly, a subset of visual AR applications strives to provide a seamless, indistinguishable integration of both real and virtual worlds. However, some requirements must be fulfilled beforehand in order to improve the realism of the augmented scene and achieve that challenging goal. For example: (i) an accurate tracking of the real world must be performed to better register the virtual data into the augmented scene, (ii) the real-world illumination must be estimated in order to photo-realistically render the virtual data in the real scene, and (iii) the mutual occlusion problem must be solved to determine the regions whether the real scene is in front of the virtual one and vice versa, and render the virtual content accordingly.

Occlusion handling is a long-standing problem in AR and has been an active topic of research in the literature for almost 30 years. A review of the literature reveals that researchers and developers have mostly tackled three different aspects of the occlusion problem, that may be mandatory or optional for systems that want to fully solve that problem:

- (**Required**) the **order problem**, to determine the depth order between real and virtual data (top row of Figure 1);
- (**Optional**) the **X-ray vision problem**, to allow, if desired, the visual exploration of a virtual data that represents a structure occluded in the real scene (middle row of Figure 1). Dependent on the solution of the order problem;

M. Macedo conducted this work while with the Department of Computer Science, Federal University of Bahia, Brazil. E-mail: marciocfmacedo@gmail.com

A. Apolinário is with the Department of Computer Science, Federal University of Bahia, Brazil. E-mail: antonio.apolinario@ufba.br

- (**Required**) the **visual display problem**, to add support to the occlusion effect mainly for video see-through (VST) and optical see-through (OST) displays (bottom row of Figure 1). Dependent on the solution of the order problem;

Despite the increasing number of techniques that have been proposed to provide occlusion handling in AR, this problem is still far from being solved in some AR scenarios. We can mention the applications that use OST displays or handheld devices to visualize the augmented scene, or that provide X-ray visualization for outdoor scenarios. In those use cases, both hardware and software technologies still must be matured to cope with the high-quality, real-time constraints of AR, to be used in more practical applications.

The main goal of this survey is to provide a detailed review of 161 papers published between January 1992 and August 2020 in the field of occlusion handling for AR applications. By looking at this field from distinct, but complementary perspectives, our main contribution is to provide a general overview of the main achievements as well as open problems of this field. We believe that such an overview of this challenging field could be useful for researchers and developers, while also serving as an introductory text for students and professionals interested in AR.

The remainder of this survey is organized as follows. Section 2 reviews the previous surveys and state-of-the-art reports that are related to this work. Section 3 shows the major steps that we have performed to select or reject papers for this review. In Sections 4, 5 and 6, we provide a historical review of the main strategies used by distinct techniques to determine the order between real and virtual objects, to handle the occlusion problem for X-ray vision applications and to build occlusion-capable visual displays. Section 7 presents a general analysis of the occlusion handling techniques, with respect to distinct properties related to AR. Section 8 provides a more thorough discussion about

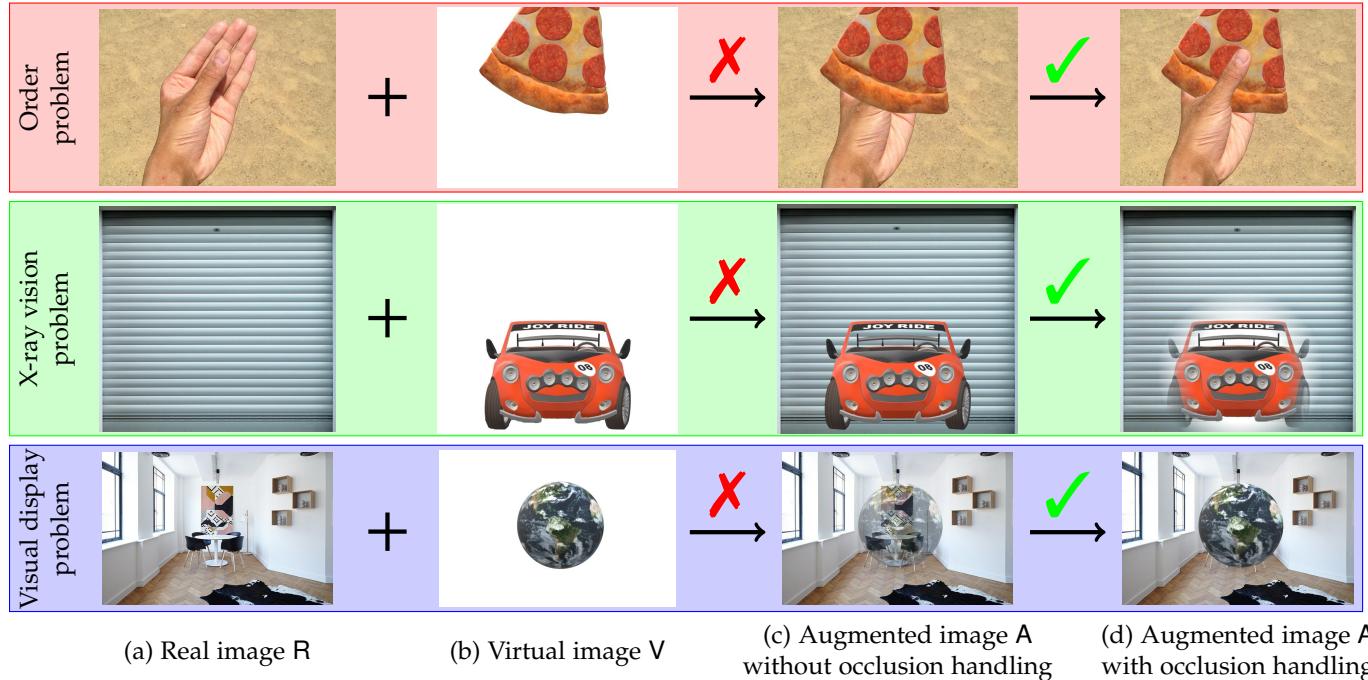


Fig. 1. Schematic view of the different aspects of the occlusion problem in AR. (Top row) Several AR applications just overlay (c) the virtual data (b) onto the real world (a) when creating the augmented image, and that may cause an undesirable effect when the virtual data is partially or fully behind the real world (c). To solve that problem, some techniques aim to determine, for each pixel of the final image, whether virtual data is occluding the real world, to better compose the augmented image (d). (Middle row) For X-ray vision applications in AR, where the virtual content represents an entity that is hidden in the real world (e.g., the virtual car (b) that is in the real garage (a)), the simple overlay of the virtual data over the real world may affect the depth perception of the user (c). Some strategies, such as the use of alpha blending together with a virtual window, may enhance the depth perception and the notion of X-ray vision for the user (d). (Bottom row) Depending on the illumination condition of the real world (a), OST displays may exhibit the augmented scene with some undesired transparency (c). In this case, the goal of occlusion-capable OST displays is to correctly manage the color that will be displayed by the hardware, enhancing the depth order between virtual and real data (d). Images (a) of the top, middle and bottom rows by HNDPTESBC, PIRO-4D and Pexels from Pixabay, respectively.

the advantages and drawbacks of the occlusion handling strategies reviewed in this survey. In Section 9, we present the current open challenges, research trends, and also suggestions for future directions in the field of occlusion handling in AR. Finally, this survey is concluded in Section 10.

2 RELATED SURVEYS

Given the considerable amount of work already proposed in the field of AR, several surveys and state-of-the-art reports about AR [1], [2], [3], [4], [5], [6] have been published during these last 30 years. These works are useful to present to the AR community a historical overview of the whole field, mainly in terms of its applications, characteristics, and current challenges. More recently, books [7], [8] were also written in order to present a more in-depth coverage of the basic theoretical and practical aspects of AR. However, none of these materials provide a detailed analysis of how the occlusion problem has been handled in AR.

As we have stated in Section 1, the occlusion handling in AR can be seen under three orthogonal aspects: as an order problem, as an X-ray vision problem, and as a visual display problem (Figure 1).

Techniques proposed to solve the order problem of occlusion handling want to determine, for a given viewpoint of the augmented scene, whether the real scene is in front of the virtual one and vice versa (Figure 1-top). Traditionally, this problem has been solved using a feature (model) of the

real scene able to separate occluding and occluded regions of the augmented world, or by comparing the depth of the virtual data with live depth data estimated from the real scene. Since real-world depth data estimation is another long standing problem in computer graphics, several strategies have already been proposed in the literature and reviewed in surveys [9], [10], [11], [12], [13]. However, those previous surveys did not focus on how these techniques have been adopted into AR scenarios. Here, we provide not only a review of the usage of depth-based techniques for occlusion handling in AR, but we also review the model-based strategies used to accomplish this task.

Once the virtual data is known to be located behind the real scene, X-ray vision techniques in AR aim to determine how these contents can be visualized, without decreasing the depth perception and the spatial awareness of the user with respect to the rendered scene (Figure 1-middle).

In 2007 and 2008, Elmqvist and Tsigas [14], [15] proposed a taxonomy for 3D occlusion management techniques in computer graphics. In their paper, they derived a set of design patterns, namely Multiple Viewports, Virtual X-Ray, Tour Planner, Volumetric Probe, and Projection Distorter, that were commonly used by the reviewed techniques.

Differently from that survey, the works of Kalkofen *et al.* [16] and Livingston *et al.* [17] published in 2011 and 2013, respectively, provided a detailed analysis of how the Virtual X-Ray and Projection Distorter design patterns were being used specifically in AR. Moreover, those surveys provided

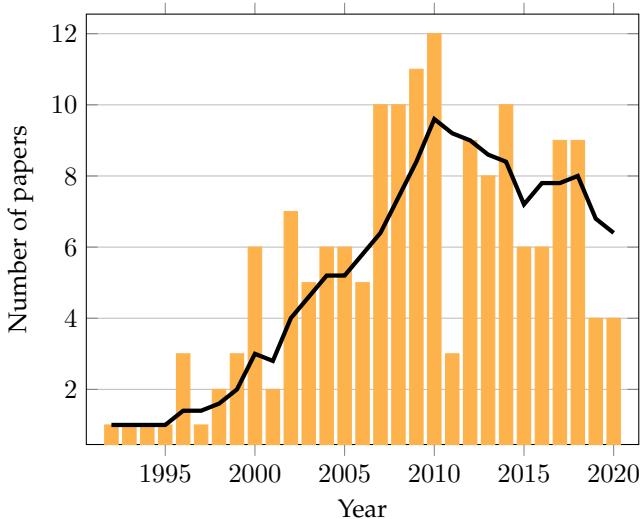


Fig. 2. Bar plot with the number of selected papers per year. Black line shows a simple moving average of the number of selected papers in a window size of five years.

an in-depth theoretical analysis of the factors that must be taken into consideration by X-ray vision applications to enhance the user's depth perception. In this work, we aim to complement those surveys by providing: (i) an updated analysis of the main features used by the X-ray vision applications (Section 5), (ii) an analysis of whether and how the X-ray vision techniques have been used with techniques that estimate the order between real and virtual objects dynamically or in distinct visual displays (Sections 7 and 8), and (iii) updated suggestions for future research on the basis of the most recent research trends (Section 9).

More recently, Zollmann and colleagues [18] have published a survey where they reviewed 67 papers and identified common pipelines and design patterns used by AR visualization techniques. Our survey differs from the work of Zollmann *et al.* because we do not investigate how the reviewed techniques handle more general visualization problems (*e.g.*, information clutter, data filtering and abstraction) in AR. Rather, we focus on how they solve the different aspects of the occlusion problem (Figure 1). We also review the approaches, not included in [18], that handle occlusion for OST displays.

With respect to the visual display problem shown in Figure 1-bottom, the surveys of Kiyokawa [19] and Wetzstein *et al.* [20] presented an overview of the main strategies proposed in the literature to provide occlusion support for OST displays in AR setups.

In our survey, we update and complement the previous surveys by providing an overview of how the occlusion problem has been handled in AR in terms of the order problem, the X-ray vision problem, and the visual display problem. By covering a time span of almost 30 years of research, we show how the field has evolved, the main strategies that have been developed, as well as the recent research trends. Moreover, by reviewing the techniques under different, but interrelated perspectives, we aim to provide new insights to the AR community with respect to the current open challenges that could be further investigated.

3 DATA COLLECTION

In order to select relevant publications that were proposed in the literature to handle the occlusion problem in AR, we first defined a search criteria to gather relevant papers from a large, well-known academic database. Hence, we have used the Elsevier's Scopus database to retrieve papers that:

- 1) Contain the prefix "*occlu*" in the title or abstract AND;
- 2) Contain the exact terms "*augmented reality*" or "*mixed reality*" in the title or abstract;

This search query, ran on August 4, 2020, returned a total of 763 papers. Then, we defined a set of rejection criteria to remove publications that:

- Do not show visual results of the proposed method;
- Simply overlay or blend virtual data with the real world;
- Only propose tracking, or registration, techniques robust to occlusion;
- Perform occlusion handling between virtual objects, without taking into consideration real-world information;
- Solve occlusion using an approach already proposed by related work (*i.e.*, lack novelty);
- Are not written in English;

After this paper rejection step, we ended up with 110 relevant publications. Finally, we have added new relevant papers, not captured by our previous search criteria, by following the references of the initially selected papers and checking whether each one of those references would lie in the scope of this study. Moreover, we have also added a few papers following our expertise. After adding 51 new papers following these criteria, a total of 161 papers were selected for this survey.

In Figure 2, we show a bar plot with the distribution of the total number of selected papers per year. As can be seen from the moving average of Figure 2, since the first paper of Bajura and colleagues was published in 1992 [23], the number of relevant papers has increased considerably until 2010, where the moving average started to decrease slightly.

To help us on the explanation of the behaviour shown in Figure 2 and guide our analysis of how the field of occlusion handling in AR has evolved, we have also built a citation network (Figure 3) with the papers that we have selected in this study. Using only the connectivity between nodes as a basis to control the graph visualization, we can see that the graph nodes are more or less clustered in three groups. A closer look at those groups reveals that the papers associated with the nodes that lie in the same group, generally aim to solve the occlusion problem under the same perspective (Figure 3-(a)). For instance, the light blue nodes in Figure 3-(a) are associated with papers that propose techniques to determine whether the virtual data is in front of the real scene. On the other hand, the dark green nodes are associated with papers that solve the occlusion problem for OST displays, and red nodes are associated with papers that follow the Virtual X-Ray and Projection Distorter design patterns [14], [15] to visualize hidden content.

By visualizing the citation network according to the year that the papers have been published (Figure 3-(b)), we can

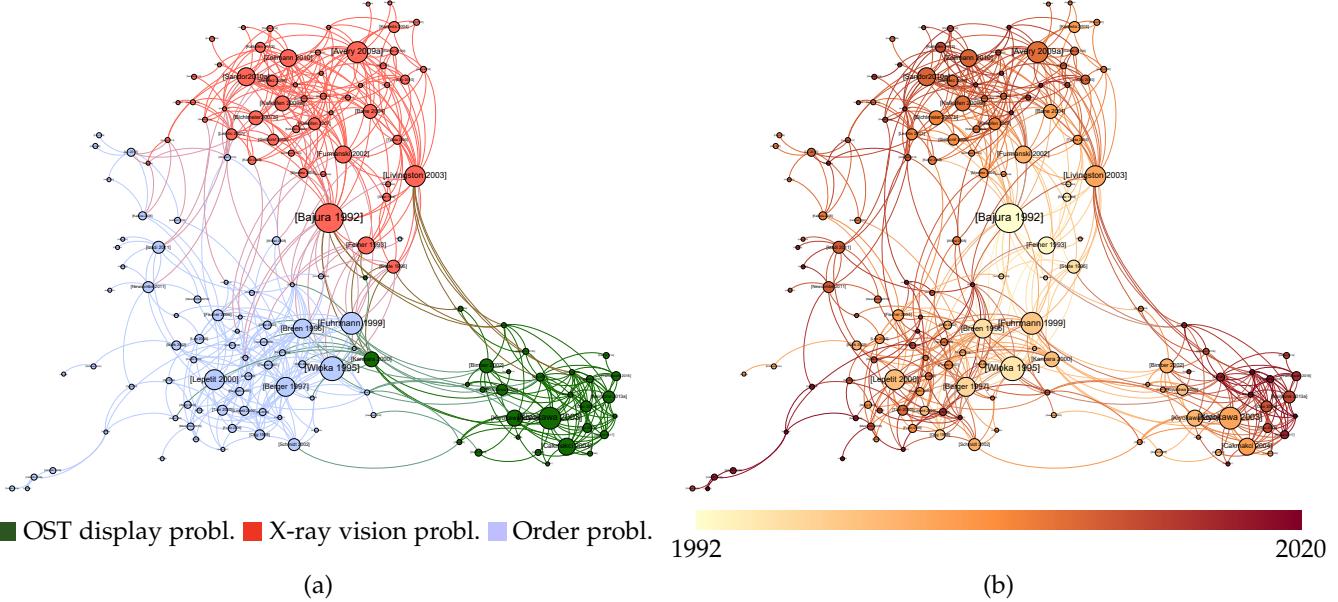


Fig. 3. Citation network of the papers selected in this study. This citation network is a directed graph, where each node represents a paper, identified by the main author's last name and year of publication, each directed edge is a citation that goes from one paper to another one, and each node's size is directly proportional to its degree (*i.e.*, the number of times that a paper was cited by the other ones included in this citation network). In (a), nodes and outgoing edges are colored according to the main aspect of the occlusion problem that was handled in the corresponding papers. Dark green nodes, located at the bottom-right side of the graph, are associated with papers that build occlusion-capable OST displays. Red nodes, located at the top side of the graph, are associated with papers that propose X-ray vision techniques. Light blue nodes, located at the bottom-left side of the graph, are associated with papers that solve the order problem of occlusion handling and do not fall under the previous two categories. In (b), nodes and outgoing edges are colored according to the year of publication of the papers. The oldest papers, more located at the center of the graph, are colored in white. The newest papers, more located at the borders of the graph, are colored in brown. In-between papers are colored in orange. This citation network, visualized according to the group (a) and year of publication (b) of the reviewed techniques, was rendered using the ForceAtlas visualization mode [21] of Gephi [22].

see that all of those three groups have evolved consistently along the years. Hence, in the next sections of this paper, we will present how each one of those groups has evolved between 1992 and 2020 in the field of occlusion handling in AR.

4 ORDER PROBLEM

First, let us assume that, unless stated otherwise, every image \mathbf{l} is a matrix, with m rows and n columns, that stores in each cell $\mathbf{l}(x, y)$, with $x \in [1, n]$ and $y \in [1, m]$, an RGB pixel $\mathbf{l}(x, y) = [\mathbf{l}_r(x, y), \mathbf{l}_g(x, y), \mathbf{l}_b(x, y)]^T$ with red, green and blue color channels, respectively. Then, let us denote \mathbf{R} and \mathbf{V} as images that capture, in that order, real and virtual world information from a specific viewpoint. The main goal of the techniques that want to solve the order problem of occlusion handling is to determine, for every pixel of a final augmented image \mathbf{A} , whether real ($\mathbf{A}(x, y) = \mathbf{R}(x, y)$) or virtual ($\mathbf{A}(x, y) = \mathbf{V}(x, y)$) world information is visible and must be seen at that location.

As we show in the following subsections, many strategies have been proposed in AR to determine dynamically whether real data are in front of virtual data. We use the standard classification criteria as [24], [25], [26], [27], [28] to classify these as model-based or depth-based strategies. Model-based strategies make use of a feature estimated from the real scene to solve the mutual occlusion problem (Section 4.1). On the other hand, depth-based strategies use live depth data generated on the basis of a specific hardware to handle the occlusion problem (Section 4.2).

4.1 Model-based Techniques

In general, model-based techniques rely on the availability of a virtual phantom (Figure 4-(a)), the capture of a background (Figure 4-(b)), the presence of an AR marker (Figure 4-(c)), the definition of a color (Figure 4-(d)), or the extraction of a contour (Figure 4-(e)) to determine the order between real and virtual objects in AR. A bar plot with the distribution of these model-based strategies in the papers selected in this survey is available in Figure 5.

Virtual phantom: The first and most popular (Figure 5) model-based strategy to solve the order problem in AR consists in the use of a geometric virtual phantom that is modelled or acquired previously to represent an object of interest of the real scene. In this case, once the virtual phantom is registered with its real counterpart (Figure 4-(a)), one can generate a single-channel depth map of the virtual data \mathbf{V}^D and another single-channel depth map of the virtual phantom representing the real data \mathbf{R}^D . Then, the augmented image \mathbf{A} can be computed just by checking which data is in front of the other (Figure 4-(f)):

$$\mathbf{A}(x, y) = \begin{cases} \mathbf{R}(x, y), & \text{if } \mathbf{R}^D(x, y) < \mathbf{V}^D(x, y), \\ \mathbf{V}(x, y), & \text{otherwise.} \end{cases} \quad (1)$$

Since the work of Breen *et al.* [24], in 1996, the phantom-based strategy has been used to solve occlusion in distinct AR applications, such as mechanical diagnostics [24], interior design [24], collaboration [32], [33], haptics [34], games [35], medicine [36], [37], [38], archaeology [39], storytelling

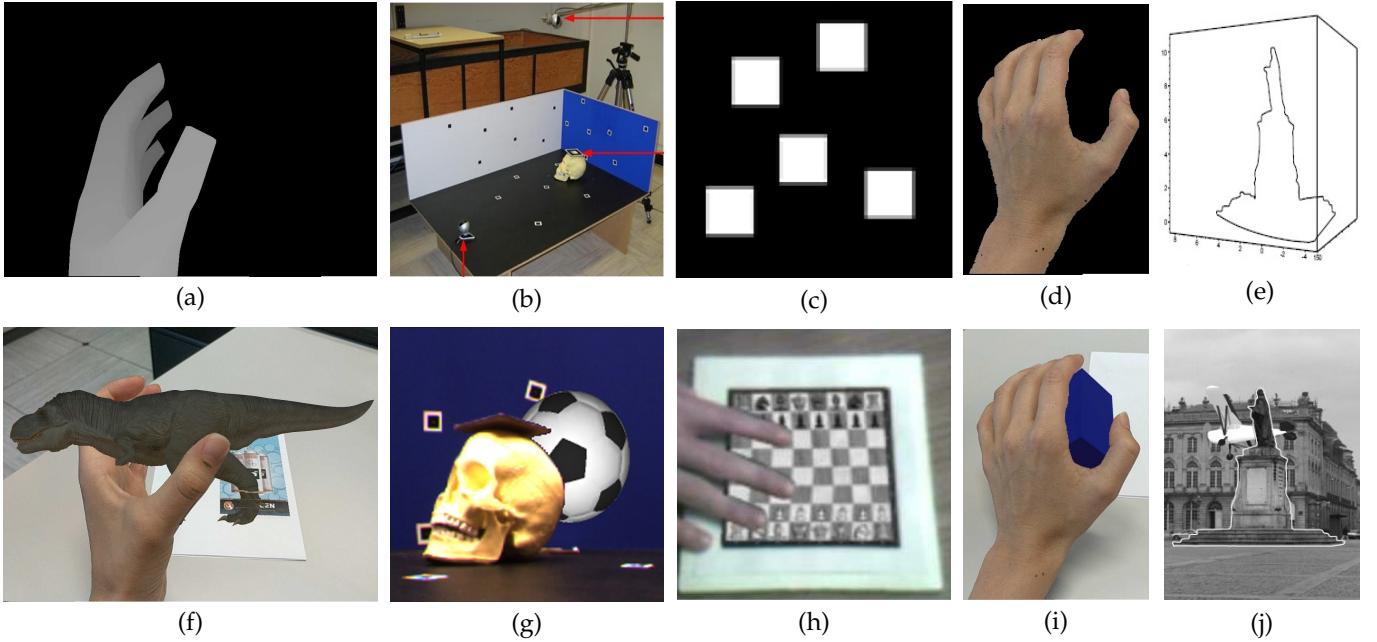


Fig. 4. An overview of the model-based strategies used to solve the order problem in AR. A virtual phantom (a) of the real counterpart (hand in (f)) may be used to determine whether the real scene is in front of the virtual one by the use of a simple depth test (Battisti *et al.* [29] ©[2018] IEEE). The background of a scene (blue wall in (b)) may be captured previously to allow foreground, and, consequently, occluder (skull in (g)) segmentation (Fortin and Hebert [26] ©[2006] IEEE). Likewise, an AR marker (c) may be used as a basis to allow occluder (hand in (h)) segmentation (Malik *et al.* [30] ©[2002] IEEE). A simple color (skin color in (d)) can represent an occluder (hand in (i)) object (Battisti *et al.* [29] ©[2018] IEEE). Finally, contours extracted from the real image (e) may be used to determine the depth order between virtual (plane in (j)) and real objects (Lepetit and Berger [31] ©[2000] IEEE). Red arrows in (b) point to the performance camera (bottom), the real object (middle), and the tracking camera (top). The original images were minimally adapted to fit on this figure.

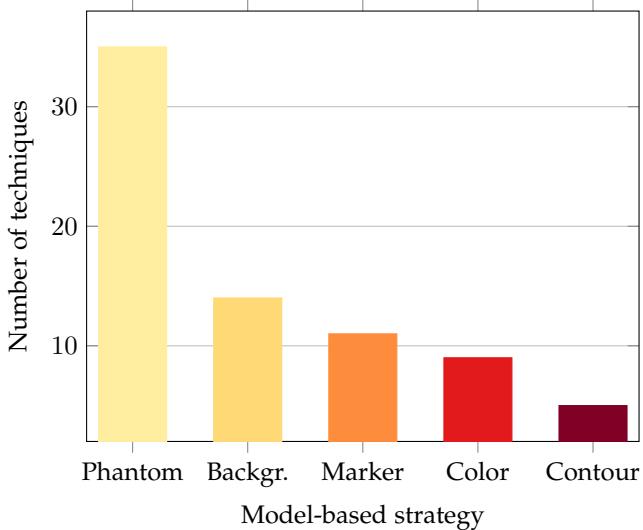


Fig. 5. A bar plot with a distribution of the model-based strategies used in the papers selected in this study. Backgr. refers to Background.

[40], projection mapping [41], navigation [42], driving [43], among others [44], [45], [46], [47], [48].

A variant of the phantom-based strategy that has been used in the early 2000s consists in producing a real counterpart of a virtual object with the use of a retro-reflective material [49], [50]. Then, whenever the retro-reflective object is occluded by another real object, its corresponding virtual data can be rendered as occluded by an OST display.

Rather than relying on the rasterization pipeline to produce the depth maps, Santos *et al.* [51] have already shown that ray tracing is also useful to solve occlusion given the availability of a virtual phantom representing the real data.

Since 2010, a few works [52], [53] made use of data available from geographic information systems to build sparse phantoms of outdoor structures and solve the occlusion problem for outdoor scenarios. Other works [29], [54] tracked a user's hand and deformed a high quality virtual hand model, in order to enable occlusion-aware hand-based interactions in AR applications.

Background: AR applications have also used background images as a way to model the occluded part of the real world (Figure 4-(b)). Let us define that V^M and R^M are binary masks, where $V^M(x, y) = 0$ for background pixels that do not contain virtual data, and $V^M(x, y) = 1$, otherwise. Also, $R^M(x, y) = 0$ whenever a pixel is estimated to be part of the background region of the real image R , meanwhile $R^M(x, y) = 1$ otherwise. In this case, the background image that can be captured statically or computed dynamically from R , is used to aid the estimation of the dynamic foreground objects that will occlude the virtual data. Then, the augmented image A is determined to be (Figure 4-(g)):

$$A(x, y) = \begin{cases} R(x, y), & \text{if } R^M(x, y) = 1 \text{ or } V^M(x, y) = 0, \\ V(x, y), & \text{otherwise.} \end{cases} \quad (2)$$

In general, statistics such as mean, median, standard deviation, covariance, histogram, or Gaussian mixture models [55] computed in a specific color space (*e.g.*, YUV [56], RGB

[57], quantized RGB [58], [59], YCrCb [26]) are estimated from a captured, or dynamically learnt background image. Then, during the live AR step, foreground occluding pixels are detected whenever their values converted to the predefined color space are too distinct to the values estimated to represent the background region of the real scene.

Ventura and Höllerer [60] reconstructed a background image given a set of keyframes, and used per-pixel differences between the live real image and the synthesized background image in the current view to perform foreground segmentation. Ladikos and Navab [61] made use of a color table to correct the color of a live real image to match the color distribution of the original background image, and increased the robustness of the foreground segmentation for distinct illumination conditions. Cordes *et al.* [62] proposed a method that took as input user-annotated foreground pixels, separated an input image into foreground and background regions and then tracked those regions to minimize the occlusion problem when compositing virtual objects in the scene. Kilimann *et al.* [63] handled occlusion in an outdoor environment by segmenting the background sky region in an image. The user selected a sample region of the blue sky in a real image, then the K-means clustering algorithm was applied to segment the sky region in the entire image.

Marker: Since many AR applications make use of fiducial markers (Figure 4-(c)) to achieve real-time tracking, several approaches have been proposed to solve the order problem of occlusion handling for marker-based AR applications. These techniques typically take advantage of the known pattern of the fiducial marker to detect possible foreground parts of the real scene ($R^M(x, y) = 1$) that may be occluding the AR marker, and would consequently occlude the virtual content (Figure 4-(h)).

Malik *et al.* [30] used histogram-based image thresholding and flood filling to detect occlusion over a fiducial marker. McDonald and Roth [64] aligned the image of the fiducial marker, detected in a given frame, with respect to a specific coordinate system, and then performed image subtraction and binary thresholding to detect the foreground occluding object located over the fiducial marker. Sanches and colleagues [65] assigned fiducial markers to occluding and occluded objects in the real scene, performed a foreground extraction to segment those objects and, on the basis of the marker distance to the camera, solved the occlusion problem. AR markers composed of several sub-markers have also been used as a strategy to ease the occlusion handling [66], [67]. The work of Fischer *et al.* [68] handled occlusion for static, non-fiducial markers by means of template matching, to detect non-trivial patterns, and image comparison, to detect occlusions. Other works [69], [70], [71], [72], [73] were even able to handle occlusion for non-rigid markers by the use of non-rigid registration or non-rigid template matching.

Color: One of the less popular, but simpler model-based strategies to solve occlusion in AR consists in the definition of an occluding color C for the real scene. In this strategy, the binary mask $R^M = 1$ if $R(x, y) \approx C$ and (2) can be used to estimate the augmented image A .

Several works [29], [74], [75], [76], [77], [78] have used the skin color (Figure 4-(d)) as a basis to extract the user's hands

from the live AR stream, and use them to occlude the virtual data (Figure 4-(i)). The work of Kutter *et al.* [79] used the blue color defined in the HSV color space to extract gloves that would occlude the virtual medical data. Ventura *et al.* [80] used the color histogram to extract user's hands in the real image. Tang *et al.* [81] used the green color to separate background from user's hands and proposed a deep neural network to learn how to predict an occlusion mask and allow hand-based user interactions in an AR environment.

Contour: Another model-based strategy that is not as popular (Figure 5), relies on the extraction of contours of the occluding region of interest in the real world (Figure 4-(e)). Basically, given the image R of the real world, one can use a user-assisted approach or an automatic contour detection algorithm in order to extract a closed contour of an object of interest in the scene. Assuming that, in this case, the binary mask $R^M(x, y) = 1$ if the pixel in the real image is located inside the area bounded by the closed contours, and $R^M(x, y) = 0$, otherwise, the augmented image A can be estimated exactly as in (2), as exemplified in Figure 4-(j). In other words, the real scene occludes the virtual scene whenever a pixel is found to be in the area bounded by the contours, or the virtual pixel is invalid (*i.e.*, do not contain a virtual data).

The use of contour extraction to determine the order between real and virtual objects was first proposed by Berger [82], in 1997, and later improved by Lepetit and Berger [31], [83], in 2000. The main idea behind those works was to extract the contours of the real objects inside the virtual mask ($V^M(x, y) = 1$), chain those contours with the parts of the real objects outside the virtual mask, and use the temporal motion between frames to build the occluding mask R^M using only the extracted contours found to be in front of the virtual data. In 1998, Ong *et al.* [84] used a user-assisted contour extraction approach to annotate the contours of occluding objects in the real image. Afterwards, a Structure from Motion (SfM) algorithm tracked and estimated sparse 3D points along those contours. Then, those points were used to recover an approximate 3D model of the occluding objects, which allowed the occlusion handling on the basis of (1).

The last relevant contour-based work was proposed in 2010, by Tian *et al.* [27]. In their approach, the user could interactively segment the occluding object in the real image, then the application tracked the object contours and, in the AR step, the segmented object was redrawn after the virtual image was overlaid over the real image.

Model-based techniques for determining the order between real and virtual objects are quite popular because they do not need an additional hardware to be able to solve the occlusion problem. However, the lack of generality of the available strategies make those techniques less useful for general-purpose AR applications that require high-quality occlusion handling.

4.2 Depth-based Techniques

Rather than use a previously modelled virtual phantom to represent the real world and solve (1), one could ease such a task by capturing or estimating a live depth data of the real world on the basis of a specific hardware setup. As

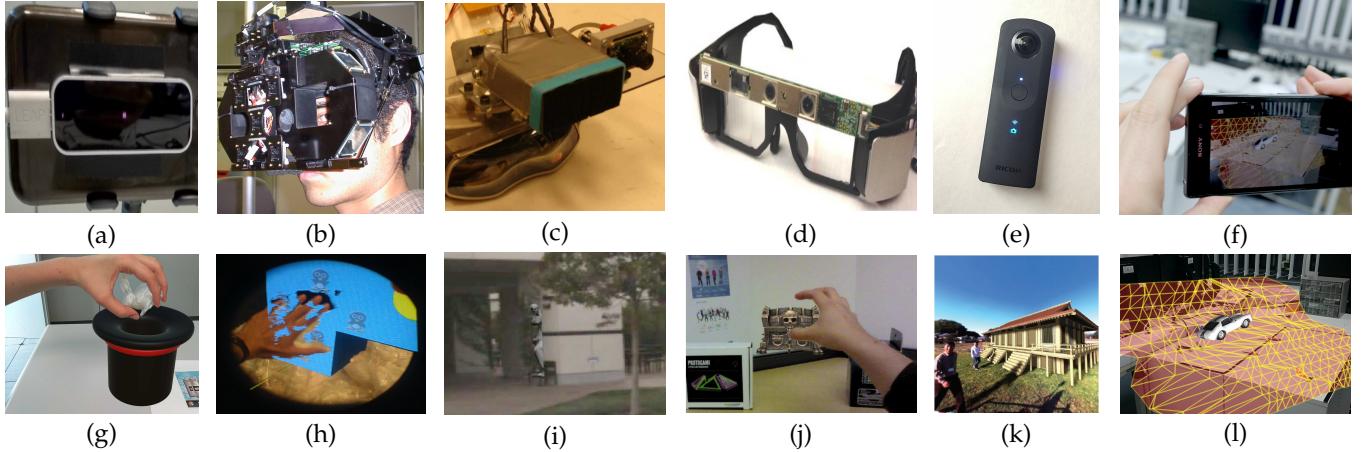


Fig. 6. Live depth data of the real scene may be computed on the basis of distinct hardware setups: two cameras (Leap Motion in (a)) that compose a stereo vision system (Battisti *et al.* [29] ©[2018] IEEE). Five cameras (b) in a multi-view stereo setup (Kiyokawa *et al.* [85] ©[2003] IEEE). Laser rangefinder (c) to provide active depth data (Wither *et al.* [86] ©[2008] IEEE). Structured light depth sensor (Intel RealSense in (d)) mounted over glasses (Du *et al.* [28] ©[2016] IEEE). Spherical camera (e) that allows depth data estimation (Ricoh Theta S released by Flickr user Serendigity in public domain). A single camera (f) provided by a smartphone, where depth data is computed by the means of monocular SfM (Schöps *et al.* [87] ©[2014] IEEE). Corresponding AR applications that allow occlusion-aware hand-based interactions ((g) (Battisti *et al.* [29] ©[2018] IEEE), (h) (Kiyokawa *et al.* [85] ©[2003] IEEE), (i) (Du *et al.* [28] ©[2016] IEEE)) and occlusion-aware virtual object compositing into the real world ((j) (Wither *et al.* [86] ©[2008] IEEE), (k) (Lu *et al.* [88] ©[2010] IEEE), (l) (Schöps *et al.* [87] ©[2014] IEEE)) on the basis of those setups can be seen in the bottom images. The original images were minimally adapted to fit on this figure.

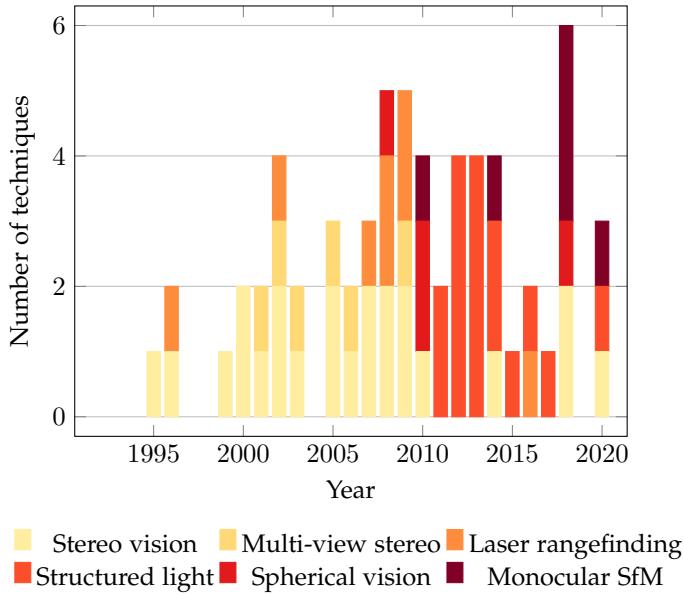


Fig. 7. Stacked bar plot with a distribution of depth-based strategies per year.

can be seen in the bar plot shown in Figure 7, as long as new hardware technologies have evolved or become more popular due to their increased quality or decreased acquisition cost, they have been adopted to solve occlusion in AR.

Stereo vision: In the '90s, the first attempts [24], [89] to estimate live depth data to solve occlusion in AR used the principle of stereo vision to simulate the depth perception provided by the human binocular vision system by the use of two color or intensity cameras. Stereo vision techniques typically perform a stereo matching step, to determine whether two pixels of distinct images correspond to the

same point in the real scene, and then estimate and refine a depth map on the basis of the disparity computed between the matched pixels [12].

Most of the stereo vision techniques used in AR between 1995 and 2003 [25], [74], [75], [89], [90], [91] generated low-resolution depth maps computed on the basis of intensity images. The exception is the work of Kojima *et al.* [76], that extracted color features and located them on both left and right images, in order to estimate the depth data of the user's fingers. The works of Hayashi *et al.* [92] and Li *et al.* [93] restricted the stereo matching step to the boundaries of the moving objects to generate depth maps at a frame rate compatible with an AR application. Zhou *et al.* [94] used a stereo vision system to solve the order problem for an OST display. Techniques such as [26], [58], [59], [78], [80] coupled model-based strategies and stereo vision to perform the stereo matching only for foreground objects extracted from the scene.

As can be seen in Figure 7, the stereo vision strategy was the dominant one to estimate live depth data in AR applications until 2009. Recently, motivated by the increasing popularity of the Leap Motion (Figure 6-(a)), a hardware that uses stereo vision to provide hand and finger tracking (Figure 6-(g)), this kind of technology has become popular again, since 2018, to handle occlusion in AR [29], [54], [95].

Multi-view stereo: A few works [61], [85], [96] that were proposed in the 2000s use the principle of multi-view stereo to perform stereo matching and depth map estimation steps on a set composed of more than two cameras (Figures 6-(b, h)). Their main goal was to improve the accuracy and minimize the presence of holes in the generated depth map, when compared to the stereo vision techniques. Mulder [97], [98], for instance, used a set of cameras to capture the real scene from distinct viewpoints, and a visual hull algorithm to recover a 3D representation of the foreground objects. The depth of this visual hull was used as a basis to handle the

occlusion problem.

A major drawback of passive approaches, such as stereo vision and multi-view stereo, that do not use an emissive light source to aid in the task of depth estimation, is that they may not be robust under low-light scene conditions. Such a problem can be overcome by active approaches that use depth sensors to emit signals in the scene and automatically compute depth data at real-time frame rates.

Laser rangefinding: Laser rangefinders (Figure 6-(c)) have been used sporadically in the literature [46], [86], [99] to provide depth data and assist in the occlusion handling for AR applications (Figure 6-(i)). In Time-of-Flight (ToF) depth sensors [41], [100], [101], [102], [103], for instance, depth is estimated as a function of the time that a signal takes to be emitted and sensed back by a sensor, after being reflected by an object in the scene [9].

Structured light: With the announcement of the Microsoft Kinect in 2009, and the rise of its popularity in 2010 [11], the structured light sensor technology has become the predominant one among depth-based strategies between 2011 and 2017 (Figure 7).

Structured light sensors (*e.g.*, Intel RealSense (Figure 6-(d))) project known stripe patterns in the scene and estimate depth data on the basis of the deformation of those patterns over the objects in the scene.

In 2011, the seminal papers of KinectFusion [104], [105] presented distinct occlusion-aware AR applications that could be developed on the basis of high-quality 3D models reconstructed from the real scene. Even so, several other works [51], [106], [107], [108], [109], [110] have used the raw depth data provided by the Microsoft Kinect sensor to handle the occlusion problem. Yii *et al.* [111] proposed a client-server system that projected the point clouds captured by a fixed Microsoft Kinect sensor to the poses of the moving handheld devices. Leal-Meléndrez *et al.* [112] proposed an inpainting method to fill the holes in the noisy depth maps provided by the Microsoft Kinect sensor. Macedo and Apolinário [113], [114] used the KinectFusion to reconstruct a 3D model of a patient's region of interest as a basis to provide occlusion-aware on-patient medical data visualization. Ha *et al.* [115] proposed the use of two RGB-D sensors attached in a head-worn display to capture near-field and far-field depth data, and used such data to detect and segment the user's hand using depth thresholding, contour extraction and distance transform.

Finally, a few other works [28], [116], [117] enhanced the quality of raw depth data typically captured by an RGB-D sensor (Figure 6-(d)) by the means of contour-based depth filling, to be able to improve the accuracy of the occlusion handling (Figure 6-(j)). The work of Du *et al.* [28] detected depth contours, estimated smooth normals at the depth contour pixels, and followed the normal directions using a dynamic programming algorithm to search for high-gradient pixels in the neighbourhood of the corresponding color image in order to assist in the depth filling process. On the other hand, the works of Hebborn *et al.* [116] and Walton and Steed [117] proposed an alpha matting technique to generate a trimap to detect invalid, unknown, background, and foreground pixels in the raw depth map. Then, the depth at the region with unknown pixels was estimated or refined using a depth propagation algorithm.

Spherical vision: Punctually in 2010 and later in 2018, the use of a single camera for depth map estimation was predominant among the depth-based techniques proposed in those two years (Figure 7). The clear advantage of this strategy is that one may provide depth-based occlusion support even for an AR application that runs in a basic hardware setup, without using an additional camera or depth sensor as a basis to provide the depth estimation. In this sense, spherical cameras (Figure 6-(e)) have eventually been used by a few works [88], [118], [119], [120] mainly to estimate depth data for outdoor applications (Figure 6-(k)).

Monocular SfM: Techniques that use monocular SfM to estimate depth data have the potential to become more ubiquitous in AR, since they may be optimized enough to provide depth order estimation in AR even on smartphones (Figure 6-(f)). In 2010, Newcombe and Davison [121] were the pioneers in the use of monocular SfM for occlusion handling in AR with a single camera. They used a multi-GPU system to provide a static, monocular 3D reconstruction of the scene interactively, taking only the live color stream provided by a single camera as input to the algorithm. In 2014, Schöps *et al.* [87] estimated a coarse plane-based collision mesh from semi-dense depth maps that allowed occlusion handling by means of collision detection (Figure 6-(l)). To run on a smartphone, the proposed technique used two parallel threads to separate tracking from mapping steps. Coarse semi-dense depth maps were estimated from low-resolution 240p images. In 2018, Wang *et al.* [122] proposed a monocular depth map estimation algorithm that relied on a pre-trained indoor depth dataset in order to provide monocular, dense 3D scene reconstructions in real time. Since 2018, a few works [123], [124], [125] have reinforced that smartphones are becoming ready to solve the occlusion problem in mobile AR applications.

Depth-based techniques are useful when providing occlusion handling for AR applications because they do not rely on any priors in the scene, and the most recent techniques are able to generate dense, high-quality depth maps for AR applications. Those depth maps may be used not only to support the process of occlusion handling, but also to allow a better understanding of the real scene (*e.g.*, the 3D shape of the objects, the global illumination condition of the scene).

5 X-RAY VISION PROBLEM

Once the depth order between real and virtual data is known, one needs to determine how the virtual content will be rendered in the augmented scene. The field of computer graphics has plenty of techniques that can provide the real-time rendering of a virtual object with photo-realism or with non-photorealism, if desired. Furthermore, as already revised by Elmqvist and Tsigas [14], [15], several techniques have already been proposed to render occluded objects in a virtual environment.

In the context of AR, X-ray vision techniques aim to use the AR technology to visualize hidden, occluded structures of the real scene. Since humans do not have such an X-ray vision ability, several strategies (see Figures 8 and 9) have been proposed in the literature to compose the visualization of the hidden objects while trying to not decrease the depth

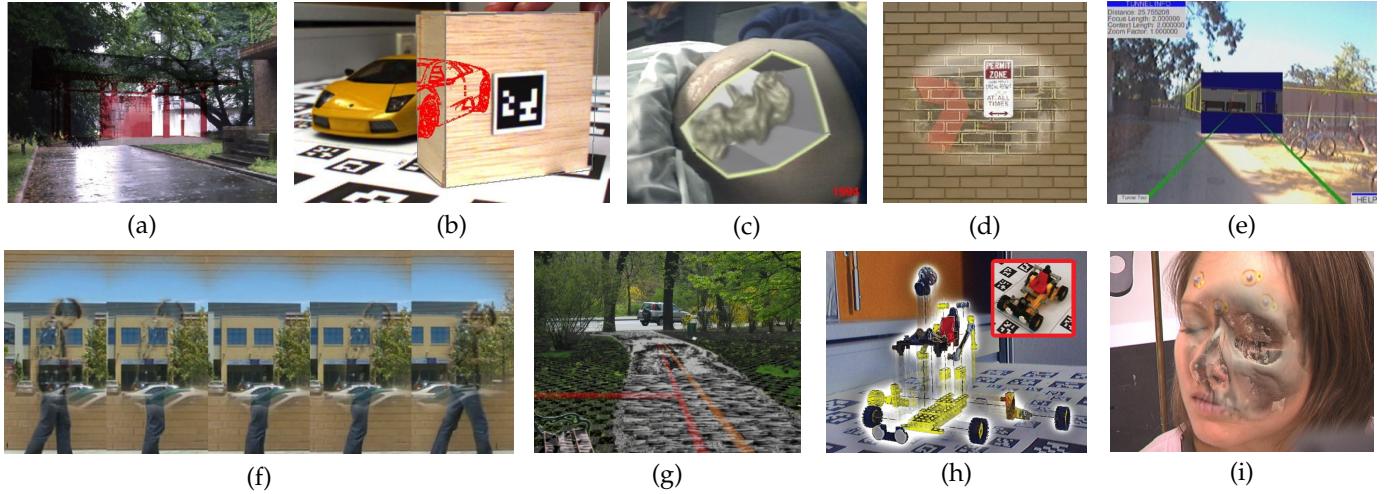


Fig. 8. An overview of the most common features used by the techniques to provide X-ray vision for occluded objects. (a) Alpha blending is commonly used to provide a smooth transition between visible and occluded objects (Fukuiage *et al.* [126] ©[2012] IEEE). (b) Edges extracted from real or virtual objects aid in the visualization of hidden structures. (Kalkofen *et al.* [127] ©[2007] IEEE). (c) A virtual window restricts the visualization of occluded objects (State *et al.* [128] ©[1994] IEEE). (d) Saliency maps computed from real and virtual scenes aid the alpha blending process (Sandor *et al.* [129] ©[2010] IEEE). (e) Perspective lines enhance the depth perception with respect to the position and distance of the occluded objects in the scene (Bane and Höllerer [130] ©[2004] IEEE). (f) Motions may be considered as salient features of the scene, influencing the augmented composition of real and virtual worlds (Sandor *et al.* [129] ©[2010] IEEE). (g) Texture details of the real scene may be extracted and preserved in the final augmented scene (Zollmann *et al.* [131] ©[2010] IEEE). (h) Real data may be spatially manipulated (e.g., warped, distorted) tridimensionally through its virtual phantom, to allow the visualization of hidden structures (Kalkofen *et al.* [132] ©[2009] IEEE). (i) Curvatures could aid the real-virtual composition, preserving the visualization of the most relevant parts of the scene (Bichlmeier *et al.* [37] ©[2007] IEEE). The original images were minimally adapted to fit on this figure.

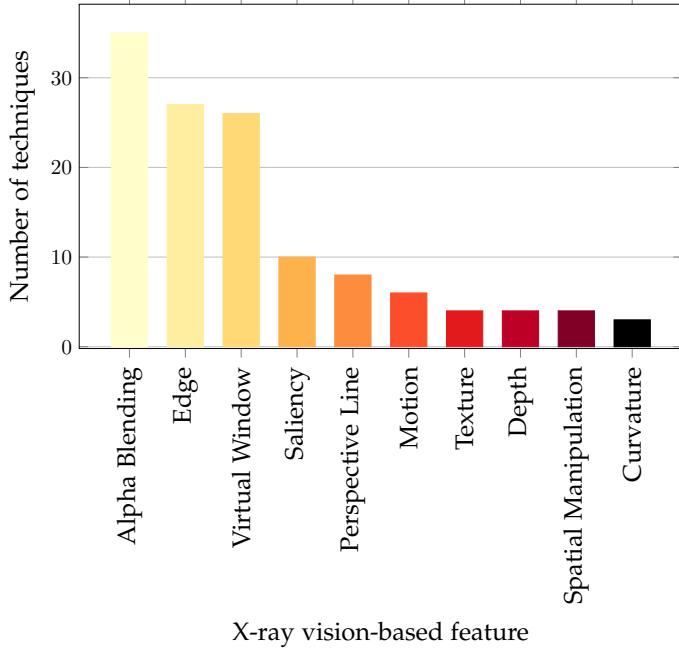


Fig. 9. A bar plot with a distribution of the features used by the AR techniques selected in this study and able to provide X-ray vision.

perception of the user with respect to the augmented scene. In this case, it is worthy to note that, rather than using solely one strategy to simulate the X-ray view, as we show in Figure 10, many of those strategies have been used together in order to enhance the quality and the depth perception of the see-through experiences.

Alpha blending: Rather than simply rendering the vir-

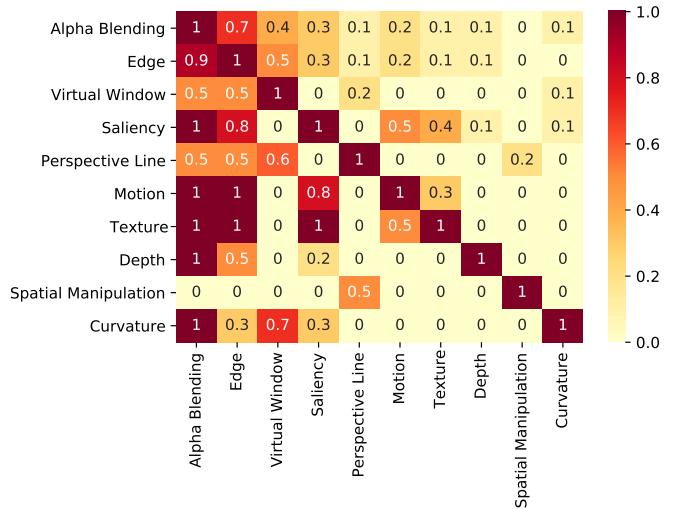


Fig. 10. A relative contingency table between the features used by X-ray vision techniques. Each cell represents how much a feature in a row has been used together with another feature in a column, as seen in the papers reviewed in this study.

tual phantom representing the hidden real structure on top of the real scene, alpha blending is the most common feature (Figure 9) shared by the X-ray vision techniques. The use of alpha blending (Figure 8-(a)) provides a smooth transition between the visualization of the visible and the occluded structures in an X-ray vision application. Kameda *et al.* [133] proposed a method for outdoor see-through visualization based on the use of surveillance cameras to recover the images of occluded regions, and alpha blending to merge the real scene with the virtual, occluded one.

Fukiage *et al.* [126] determined that multiplicative and inverse-multiplicative blending are the best strategies to merge virtual hidden models in an AR scene, once the real image is separated into background and foreground regions. Macedo and Apolinário [113], [114] proposed distinct ways to visualize clipped medical data on the patient. A previous capture of the background scene was required to provide a realistic composition of the see-through effect on the clipped region of interest of the patient.

Alpha blending has been used not only to provide X-ray vision, but also to minimize artifacts caused by an inaccurate object order estimation [27], [33], [35], [77], [116], [117]. In these cases, the noisy boundary regions located at the transition between real and virtual objects are filtered to provide a more pleasant occlusion handling.

Edge: As shown in Figure 8-(b), edges are also commonly extracted from both real and virtual worlds to aid in the visualization of the hidden structures (Figure 9), since edges are generally considered as low-level features that convey the basic structure of the scene [127]. As shown in the relative contingency table of Figure 10, about 90% of the techniques that use edge information to provide the X-ray vision, also use alpha blending to provide a smooth visualization of those edges. The works of Feiner and colleagues [134] and Webster *et al.* [135], for instance, rendered the occluded structures as virtual wireframes, and used alpha blending to fade out the visualization of those structures that were located at the peripheral vision.

Virtual window: One of the uses of AR in medicine is to provide on-patient medical data visualization, as exemplified by the case shown in Figure 8-(c), where a virtual fetus is visualized on top of a pregnant patient. The first strategy to provide such an X-ray vision effect, without decreasing the understanding the scene and the depth perception of the physician, relied on the use of a virtual window to provide a cutaway view of the augmented scene and restrict the visualization of the virtual data, as proposed by Bajura *et al.* in 1992 [23]. Since then, other medical AR works have used the same strategy for distinct medical applications [99], [128], [136], [137], or have incorporated its use with other features such as alpha blending and edge extraction (the two features most correlated to virtual window, as shown in Figure 10), to improve the depth perception of the user [37], [79], [138], [139], [140], [141].

Magic lenses [142], [143] have also been used in AR to separate focus from context regions, enhancing the visualization of the relevant structures in the augmented scene [127], [144], [145], [146], [147]. To capture hidden structures in an outdoor scenario, Barnum *et al.* [148] used two cameras, a reference stationary camera to capture the occluded scene, and a source moving camera to capture the occluding scene, while Maia *et al.* [149] used static data available from Google Street View to represent the occluded streets. Then, both works used edges extracted from the real scene, virtual windows drawn by users, and alpha blending to control the X-ray vision effect.

Perspective line: Since the notion of depth may be lost when merging the visualization of visible and occluded structures in an X-ray vision application, Bane and Höllerer [130] have included the rendering of virtual perspective lines (Figure 8-(e)) going from the camera towards the

occluded, hidden content. Perspective lines are useful, in this case, because they appear to converge at a vanishing point positioned at a maximum distance to the camera. Hence, perspective lines serve as a perceptual cue for the user with respect to the distance of a virtual content.

Spatial manipulation: Although not as popular as other strategies (see Figure 9), spatial manipulation of the real world by means of "explosion" [132] (Figure 8-(h)), distortion [150], [151] or warping [42] of its virtual counterpart is also a way to enable the see-through visualization of occluded objects in AR. Perspective lines have also been used with spatial manipulation techniques to assist the user on how the real world has been distorted in the augmented scene, since the perspective lines will also suffer from the same distortion effect. As shown in the relative contingency table of Figure 10, perspective lines were used in two [150], [151] of the four spatial manipulation papers.

Other features: While edges may be considered as low-level features of a given scene, one can make use of high-level features extracted from a scene to give support on how real and virtual worlds can be composed in an X-ray vision setup. As illustrated in Figure 10, saliency (Figure 8-(d)) [120], [129], [131], [152], [153], [154], [155], motion (Figure 8-(f)) [129], [154], [155], [156], texture (Figure 8-(g)) [131], [154], [155], depth [120], [157], [158], [159] and curvature (Figure 8-(i)) [37], [138], [153] features have been used specifically to increase the visual quality of the alpha blending composition. As shown in the relative contingency table of Figure 10, 100% of the techniques that used one of those high-level features, also used alpha blending to visualize the hidden content (Figure 10).

Different from the techniques presented in Section 4, X-ray vision techniques cannot be evaluated using an accuracy metric. Since humans do not have X-ray vision capability, a ground-truth result is not available to allow such an objective evaluation. In this case, several papers [160], [161], [162], [163], [164], [165], [166], [167], [168], [169], [170] entirely dedicated to evaluate X-ray vision techniques through user studies have also been published in the literature.

Many strategies have been used by AR applications to enable the X-ray vision. Alpha blending is the most common way to compose virtual and real scenes in an X-ray vision setup, and the edges are the most common low-level features used to extract a simplified representation of both worlds. Despite the fact that the field still does not have a single composition of features that will provide the best X-ray vision effects in general AR applications, some specialized fields seem to have found the best features that work well for the majority of their AR applications. Medical AR works, for instance, tend to use virtual windows with alpha blending and edge extraction to provide on-patient medical data visualization.

6 VISUAL DISPLAY PROBLEM

In AR, we generally categorize the visual augmenting displays in two groups: video see-through (VST) and optical see-through (OST) displays [8]. In VST displays, the user does not see the real world directly through the display. Rather, real-world imagery is captured through the use of at least one additional video camera attached to the display

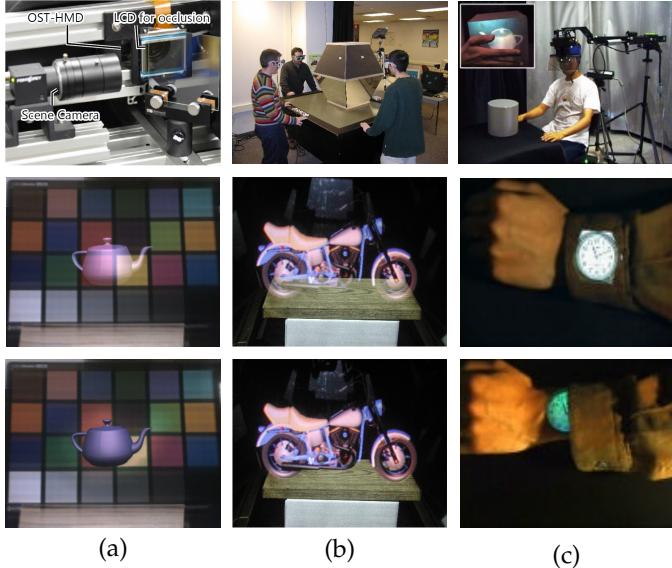


Fig. 11. An overview of the most common strategies proposed to build occlusion-capable OST displays. (top of (a)) Liquid Crystal Displays (LCDs) are typically embedded as spatial light modulators in the OST display to cut off light rays that may add an undesired transparency effect (middle of (a)) for foreground virtual objects that must occlude the real scene (bottom of (a)) (Itoh *et al.* [171] ©[2017] IEEE). The pattern illumination of the real scene can also be controlled by the use of light projectors in an OST setup (top of (b)), in order to project shadows at the regions where the virtual object occludes the real world (middle and bottom of (b)) (Bimber and Fröhlich [45] ©[2002] IEEE). Finally, the production of a real counterpart of a virtual scene with a special material, such as a retro-reflective screen (top of (c)) may be used to provide occlusion handling for OST displays (middle and bottom of (c)) (Inami *et al.* [49] ©[2000] IEEE). The original images were minimally adapted to fit on this figure.

hardware. Then, the augmented image A is presented to the user as a combination of a digital representation of the real world R with a computer-generated image V. In OST displays, the user is able to look at the real world straight through the display, and the AR effect is obtained by the visualization of the real world mixed with a virtual image shown over the display.

Given that the order between real and virtual objects has already been accurately estimated, the task of solving the occlusion problem by correctly choosing whether real data must be displayed in front of virtual data can be easily achieved for applications that make use of VST displays. In this case, knowing that the digital representation of both real and virtual worlds is available to the display technology, the occlusion visualization problem can be reduced to the determination of whether each pixel color to be shown by the display will be taken on the basis of the real or virtual image available. The majority of the previous techniques presented in Sections 4 and 5 make use of VST displays to visualize the augmented scenes.

With the recent increase in popularity and quality of devices such as Microsoft HoloLens, OST displays are becoming even more powerful to support the visualization and the development of AR applications. The most recent commercial OST displays are able to cope with the real-to-virtual occlusion problem, in which an object in the real world occludes a virtual object in the augmented scene.

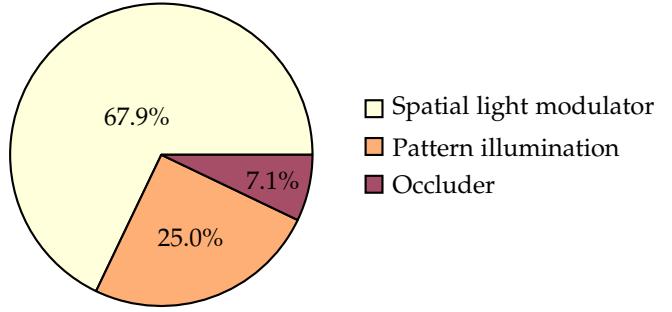


Fig. 12. A pie chart with a relative distribution of the strategies to build occlusion-capable OST displays used in the papers selected in this study.

However, they still are not able to solve the virtual-to-real occlusion problem, where the goal is to render fully opaque virtual objects to occlude the real world. Solving the mutual occlusion problem for OST displays is not so trivial, since those displays cannot easily control how virtual images will be visualized on the display. Some aspects of the real world, such as its illumination condition, typically interfere with the visualization of the augmented scene. Due to those hardware constraints, for instance, virtual objects are still rendered with some undesired transparency (see the middle of Figure 11-(a) for an example).

Since the occlusion problem can be easily solved for VST displays, we will focus the remaining of this paper on the presentation of the main strategies employed by OST displays to provide support for occlusion handling. Kiyokawa [19] stated that the three most common strategies to provide such an occlusion support rely on the embedding of a spatial light modulator on the display (Figure 11-(a)), the controlling of the pattern illumination of the real scene (Figure 11-(b)) or the production and placement of an intrusive occluder in the real world (Figure 11-(c)).

Spatial light modulator: The most popular strategy to add support for the visualization of fully opaque virtual objects that occlude the real world consists in the use of a spatial light modulator embedded in the OST displays (Figure 12). A spatial light modulator is able to block light rays coming from the real world at the position of the occluder virtual objects and going to the user's eyes. Transmissive LCDs [44], [85], [94], [96], [97], [98], [172], such as the one shown at the top of Figure 11-(a), reflective Liquid Crystal on Silicon (LCoS) [173], [174], [175], [176], [177] and single reflective Digital Micromirror Device (DMD) [178], [179], are commonly used to provide such a spatial light modulation. Occlusion support has also been provided for compact OST devices, such as AR glasses [180], devices that used the principles of cloaking optics [181], and recently varifocal occlusion support has also been added for LCD-based spatial light modulators [182], [183].

Pattern illumination: Another option to enable the correct rendering of fully opaque objects in OST displays consists in controlling the illumination of the real world, or at least a subset of it where the virtual objects will be rendered (top of Figure 11-(b)), through the use of light projectors. Those projectors may cast shadows onto real objects to be occluded and OST displays may be used to visualize the vir-

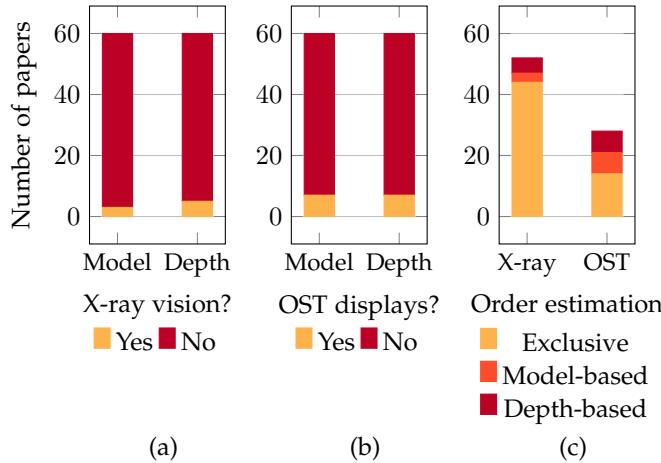


Fig. 13. Stacked bar plots with techniques that solve the order problem of occlusion handling grouped by whether they also solve the occlusion problem for (a) X-ray vision applications and (b) OST displays. Stacked bar plots in (c) show how many papers that handle the occlusion problem for X-ray vision applications and OST displays also determine dynamically the order between real and virtual data. Such an order can be determined using a model-based or depth-based strategy, or assuming a static, exclusive fixed depth order, that does not handle the mutual occlusion problem.

tual objects exactly at those shadow positions (middle and bottom of Figure 11-(b)). This pattern illumination strategy is able to enable the rendering of fully opaque virtual objects in the augmented scene [184], [185].

Bimber and Fröhlich [45] applied this strategy in a light-controllable mirror-based static environment shown at the top of Figure 11-(b). Murase *et al.* [47] relied on a set of stereo projectors to project the virtual object onto a semi-transparent mirror, and light projectors to illuminate the scene. Then, when a virtual object was occluded by a real object, stereo projectors did not render the virtual object at the occluded positions. On the other hand, when a real object was occluded by a virtual object, light projectors did not illuminate the real object at the occluded positions. Kurz *et al.* [39] adapted the pattern illumination strategy for tabletop displays with the use of an additional stereo projector located at the top of the mounted scenario. Maimone *et al.* [110] proposed the use of Microsoft Kinect sensors, OST displays and light projectors to solve occlusion for visual telepresence applications. Avveduto *et al.* [48] controlled the illumination of the real scene to add mutual occlusion support for the Microsoft HoloLens.

Occluder: The less popular strategy to solve occlusion for OST displays relies on the preparation of a real counterpart of the occluder virtual object (Figure 12) with the use of a specific material (*e.g.*, retro-reflective screen [49], [50], top of Figure 11-(c)) to enable the OST display to solve the mutual occlusion problem (middle and bottom of Figure 11-(c)) without controlling the illumination of the scene, nor embedding a spatial light modulator into the OST display.

Out of those three strategies presented in this section able to solve occlusion for OST displays, the spatial light modulation strategy is the most popular and the most promising solution to be embedded in commercial OST displays in the future. That strategy enables the occlusion problem to be solved by the own OST display, without

requiring a custom setup outside the hardware (**pattern illumination**) or the production of a real **occluder** counterpart.

7 DATA ANALYSIS

In this section, we provide a general analysis of the field of occlusion handling in AR following the most relevant facts that we could extract from the data that we have collected for each selected paper. We refer the reader to Tables 1, 2, 3, and 4 of the supplementary document to see all the data collected, that support the generation of the charts and bar plots shown in this manuscript.

In Sections 1 and 2, we stated that the occlusion problem in AR can be seen under three distinct, but complementary perspectives: as an order problem, as an X-ray vision problem and as a visual display problem. To support that statement, we show in Figure 13 how many papers have dealt with the occlusion handling problem in all of these perspectives.

Figure 13-(a) shows that less than 10% of the papers that propose model-based or depth-based techniques also provide support to X-ray vision in AR applications. Considering the total of X-ray vision papers, only 13.7% (Figure 13-(c)) have also solved the dynamic order problem [37], [42], [79], [99], [113], [114], [120]. Most of these techniques were proposed to deal with on-patient medical data visualization [37], [79], [99], [113], [114], and solely the work of Roxas *et al.* [120] could provide, at the same time, X-ray vision and depth-based occlusion handling for an outdoor AR application.

11.7% of the papers that propose model-based or depth-based techniques also solve the mutual occlusion problem for OST displays (Figure 13-(b)). In fact, they represent 50.0% of the total of OST display techniques reviewed in this survey, as shown in Figure 13-(c). Despite that high percentage value, only two [48], [177] out of the eight [48], [171], [172], [177], [178], [181], [182], [183] OST display papers proposed in the last five years have also solved the order problem dynamically. While several papers [24], [26], [29], [41], [42], [46], [51], [54], [58], [59], [61], [74], [75], [76], [78], [80], [118] have proposed the use of both model- and depth-based strategies to solve the order problem of occlusion handling, none of the papers selected in this study have solved both the X-ray vision problem and the rendering of fully opaque virtual objects in OST displays at the same time.

With respect to the environments where the proposed techniques have been evaluated, we can see from Figure 14-(a) that less than 15% of the model-based and depth-based techniques have been tested in or designed for outdoor scenarios. Contour [31], [83], [84], background [62], [63], [118] and phantom [52], [53] were the features used to handle the occlusion in those cases. As already stated in Section 4.2, depth map estimation on the basis of a single camera, through spherical vision [88], [118], [119], [120] or monocular SfM [123], [124], [125], is predominant among the outdoor depth-based techniques. Out of the 41.8% of X-ray vision techniques proposed for outdoor applications, only the work of Roxas *et al.* [120] is also able to solve the order problem in dynamic scenarios. Given the considered "indoor" X-ray vision applications in AR, some of them have

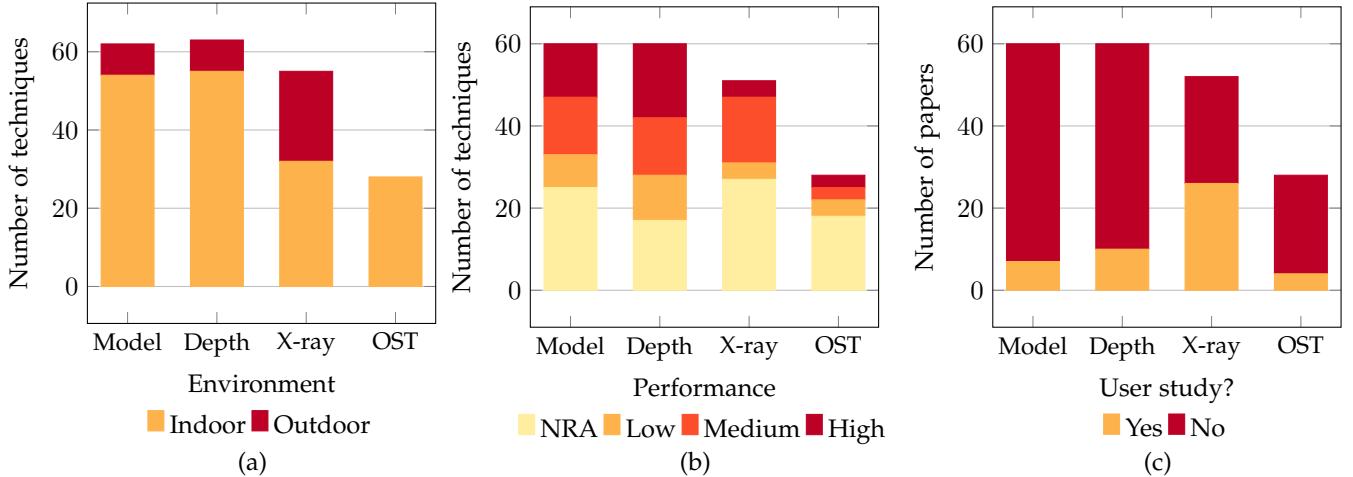


Fig. 14. Stacked bar plots with distinct occlusion handling techniques grouped by whether they solve the order problem (model-based and depth-based techniques), the X-ray vision problem or the OST display problem, respectively and: (a) the environments where they have been evaluated, (b) their performance and (c) whether they were evaluated with a user study, as reported by the reviewed papers. In (a), techniques evaluated for both indoor and outdoor scenarios are counted twice. Inspired by the classification proposed by related work [186], in (b), we label a performance as high (equal or above 30 frames per second (FPS)), medium (between 6 and 30 FPS), low (below 6 FPS) or not reported by the authors (NRA). Techniques that handle more than one aspect of the occlusion problem (e.g., depth-based techniques that also solve the occlusion problem in X-ray vision scenarios) are counted once for each occlusion aspect handled.

been proposed specifically for minimally invasive surgery applications [138], [141], [157], [158], [159], where the distance field is even narrower, and the desired accuracy is higher. Finally, none of the techniques proposed to provide occlusion support for OST displays have been evaluated in outdoor lighting conditions.

In terms of performance, Figure 14-(b) shows that real-time performance has been more present among the strategies that estimated the order between real and virtual objects in AR, being achieved by 21.7% of the model-based techniques and by 30% of the depth-based techniques. Only three papers (10.7% of the total) [47], [85], [179] were reported to achieve real-time frame rates when handling the occlusion problem for OST displays. On the other hand, 7.4% of the X-ray vision techniques [79], [113], [114], [152] could achieve real-time frame rates. It is noteworthy that none of those real-time, X-ray vision techniques or OST strategies to build occlusion-capable displays have been proposed for outdoor scenarios, meanwhile, only the model-based work of Kasperi *et al.* [53] and the depth-based technique of Valentin *et al.* [124] reported real-time results in outdoor AR applications.

In Figure 14-(c), we show how many papers have provided a user study evaluation. As already expected, a low percentage of 11.7% of the model-based [42], [47], [48], [53], [54], [66], [81], 16.7% of the depth-based [42], [54], [85], [95], [96], [106], [107], [108], [109], [115], [120], and 14.3% of the OST display [47], [48], [85], [171] techniques additionally evaluated the proposed techniques using a user study, since the main problems that these techniques aim to solve can be modelled by an objective function. On the other hand, 50% of the X-ray vision papers reviewed in this survey presented a user study evaluation. As mentioned in Section 5, that behaviour is expected because the X-ray vision problem lacks an accuracy metric, since no ground-truth is available to determine the best way to compose the rendering of the hidden, occluded objects into the augmented scene.

8 DISCUSSION

In this section, we provide a critical discussion about some of the main findings that we have obtained after reviewing the 161 papers included in this study.

Model-based strategies: Virtual phantom (Figures 4-(a, f)) is the most common feature used by model-based techniques that solve the order problem of occlusion handling (Figure 5). But, since the virtual phantom is commonly taken as a static prior knowledge of the scene, the corresponding real object may not be changed dynamically during the AR live stream, at least without the availability of another virtual counterpart of the new real object.

Background, marker and color features (Figures 4-(b, c, d, g, h, i)) are useful as a way to provide occlusion handling at real-time frame rates when the occluder is a foreground object that comes from the real world. However, these features are not generally robust enough to solve the mutual occlusion problem in scenarios where a virtual object may be located in front of the foreground real object.

The contour feature (Figures 4-(e, j)) is not as popular as the other features to provide support for model-based object order estimation (Figure 5). Even so, the edges that compose the contours of real and virtual objects are really important to enhance the quality of depth-based techniques [28], [92], [93], [116], [117] (Section 4.2) or to represent low-level features of hidden structures (Figure 8-(b)) in X-ray vision applications (Section 5).

Depth-based strategies: The popularity of low-cost devices such as Leap Motion (Figure 6-(a)), Microsoft Kinect and Intel RealSense (Figure 6-(b)), together with the practicality of the use of two cameras for depth map estimation, has made the usage of stereo vision and structured light technologies predominant among the depth-based techniques used for occlusion handling in AR (Figure 7). Recently, companies are even integrating these technologies into mobile (e.g., Microsoft HoloLens) and handheld

displays (*e.g.*, iPhone 11, iPad Pro). However, despite of this recent industrial trend, some state-of-the-art techniques [123], [124], [125] have shown that it is possible to produce high-quality depth maps, even in real time, for both indoor and outdoor scenarios, on the basis of the live color stream captured by monocular cameras already embedded into handheld devices. That indeed broadens the possibility to solve the order problem of occlusion handling for a plethora of AR applications.

Even with the recent advances in the real-world depth map estimation, there is a demand to further increase the accuracy of the generated depth maps, while keeping the real-time performance and temporal coherence of the depth map generation process. Some techniques blur the transition between real and virtual data in order to minimize some artifacts generated due to an inaccurate occlusion handling (see Section 5). Given that the raw depth data commonly provided by depth sensors are noisy, with shadows and holes distributed along the depth data, some techniques (*e.g.*, [28], [116], [117]) have focused on the enhancement of the quality of those raw depth data by improving the depth map quality mainly at the contours of the captured objects. Since those works take as input raw color and depth data provided by an RGB-D sensor, and give as output a higher quality depth data at real-time frame rates, we believe that they may be coupled as a pre-processing step of other techniques, in order to enhance the accuracy of the occlusion handling.

X-ray vision techniques: Controlling the alpha blending between real and virtual images is the easiest and most common way used in the literature to provide an X-ray view of an AR scene (Figure 9). Although alpha blending is more useful than the simple overlay of the virtual scene over the real one [160], a careless, naïve blending of real and virtual scenes may confuse users with respect to the depth order of the augmented objects, does not improving the depth perception of the AR application [126].

Rather than relying solely on the available color information to determine how real and virtual worlds would be alpha blended in an X-ray vision application, the literature has already shown that distinct low- and high-level features of both worlds can be computed in real time and can be used to control the X-ray view effect. Features such as depth, edges, saliency, texture, motion, and curvature, improve the quality of the alpha blending algorithm, by enhancing the visualization of the most relevant, salient parts of both real and virtual objects. However, it is still unclear which combination of features, if any, provides the best X-ray augmentations for AR, and whether novel features could be useful to enhance the visualization of hidden contents in AR.

In medical AR applications, virtual windows are really popular and useful to restrict the X-ray view of hidden organs to a region of interest on the screen [23], [37], [79], [99], [113], [114], [128], [136], [137], [138], [139], [140], [141], enabling physicians to focus their visualization in the hidden organ, while also keeping them aware of the context of the real scene that is surrounding the virtual rendering.

Other strategies, such as the addition of virtual perspective lines in the AR scene, and the spatial manipulation of the real world to reveal the presence of hidden structures,

have also been proposed in the literature. While perspective lines are not much useful alone, and may be useful to enhance the depth perception of AR applications that use virtual windows to visualize hidden contents, the spatial manipulation was not much exploited by the literature.

Occlusion for OST displays: The literature has already proposed OST Head Mounted Display (HMD) solutions that are encumbrance-free [180], robust to varifocal rendering [182], [183], and that provide model- or depth-based dynamic occlusion handling [85], but that are still prototypes. Current commercial OST-HMDs, such as the Microsoft HoloLens, have also been integrated with depth sensors. In this case, while the provided depth maps in OST-HMDs expand the possibilities of occlusion handling in AR, they are not sufficiently accurate or optimized for some applications, like [48]. Furthermore, most of those commercial OST-HMDs still are not integrated with solutions to minimize the mutual occlusion problem, as reported by related work [48], [178].

Projector-based displays that use the pattern illumination strategy to provide occlusion handling for OST displays have already been integrated in collaborative setups [110] or with current OST-HMDs [48], but the custom setup required in the scene still may prevent their use in more practical scenarios.

9 FUTURE DIRECTIONS

As we could see from this review, after almost 30 years of research, a lot of distinct techniques have been proposed to better handle the occlusion problem in AR applications. So, the big question one may ask is: **what are the remaining open challenges of this field?** In this section, we aim to answer that question, bringing out the key topics that we believe still require a careful attention of the AR community.

Real-time non-rigid registration: Real objects may consist not only of rigid bodies, but also of non-rigid bodies that can be deformed. Therefore, non-rigid virtual phantoms may be used to better represent deformable real objects in AR setups and one may still want to track those deformable objects and determine whether they are in front of other virtual objects in the AR scene. 2D non-rigid registration techniques have already been used for marker-based occlusion handling in AR [69], [70], [71], [72], [73]. Furthermore, the literature has already shown that real-time non-rigid registration can be achieved when RGB-D sensors are used to capture color plus depth streams [187]. However, it is still challenging to provide real-time 3D non-rigid registration for markerless AR applications that rely solely on an RGB stream captured by a monocular camera, mainly due to the high computational costs involved in this process. One possible solution towards this goal, for instance, is to make use of monocular SfM techniques or deep learning strategies to estimate depth from color data, and then convert the monocular camera into an RGB-D sensor. Unfortunately, since only a few monocular depth map estimation techniques [123], [124] could achieve a performance compatible with the tight processing time restriction of AR applications, significant efforts may be required to achieve real-time performance when integrating RGB-D non-rigid registration in this pipeline.



Fig. 15. An original image (left) is distorted (right) to reveal hidden structures located outside of the current view. (Sandor *et al.* [150] ©[2009] IEEE). The original images were minimally adapted to fit on this figure.

X-ray vision: Several techniques have already been proposed to allow an X-ray view of the real scene. However, we believe that this field still needs more studies for five reasons, presented as follows.

First, to solve the X-ray vision problem in scenarios where contents displaced at multiple distances in depth must be seen in a single, augmented final image. Suppose the scenario where a user is located in a room, in a building or other indoor space, and he/she wants to see hidden contents located in multiple rooms, that are positioned in front of him/her, at multiple depths. Another possible scenario is an outdoor AR application where the user may be able to see hidden contents located in distinct streets, which may be near to far away from the user. Hence, the main question is: how can these contents be composed into a single image, while keeping a good spatial awareness for the user? The user study conducted by Livingston *et al.* [161] with eight participants and three depth layers has already shown that the use of a wire and fill drawing style, with opacity and intensity decreasing as far as the object is located, improved the user perception of the augmented scene. Bane and Höllerer [130] separated the multiple depth layers into focus and context regions, and used edges and perspective lines (Figure 8-(e)) to provide a depth cue for the user with respect to the position of the occluded contents. Even so, more studies are needed to check whether these strategies are still valid when features such as motion (Figure 8-(f)) and spatial manipulation (Figure 8-(h)) are used to provide the X-ray vision effect.

Second, to handle the X-ray vision problem in scenarios where one wants to view hidden contents that may be located not only in front of the user, but also at its peripheral view, as illustrated in Figure 15. To the best of our knowledge, only the distortion-based work of Sandor *et al.* [129], [150] has addressed this problem for outdoor scenarios using a static 3D model of the scene, merged with textures captured from Google Street View, and the work of Wu and Popescu [42] has achieved that goal for indoor scenarios using the dynamic color and depth data provided by an RGB-D sensor.

The third reason is to provide context-aware X-ray vision using out-of-view content captured in real time, using an additional camera positioned in the scene, an RGB-D sensor in an indoor environment or surveillance cameras available in the outdoor space. We have seen that a few papers took care of this problem and supported the capture of the dynamic occluded content of the real scene. Bajura *et al.*

[23] and State *et al.* [128] captured dynamic ultrasound data using medical sensors and rendered that in an on-patient medical data visualization application. The works of Avery *et al.* [145], [164] and Barnum *et al.* [148] positioned a colored camera outdoors to capture the occluded region in real time, meanwhile other works [133], [162] used surveillance cameras for that task. The work of Wu and Popescu [42] used an RGB-D sensor to capture dynamically the hidden content.

Fourth, to compare the existing strategies for X-ray vision and determine the best ones through an extensive user study. We stated in Section 5 that at least 11 papers entirely dedicated to provide user studies for X-ray vision in AR have already been written and published in the literature. While these papers have covered distinct AR applications where the X-ray vision effect is desirable, in fields like medicine [163], education [167], navigation [169], some of these user studies provided a comparison between the naïve alpha blending composition and one or two alternative techniques proposed in the literature. As we show in Figure 8, several features have been proposed to improve the visual quality of the X-ray view effect. We believe that a more extensive user study comparing the use of those features, together with an evaluation of their recent X-ray vision techniques would be useful to suggest the possible current best strategies proposed in this field.

The final reason is to integrate X-ray vision and model-/depth-based approaches, in order to avoid the careless overlay of the X-ray view in the real scene. As we detailed in Section 7, only 13.7% of the X-ray vision papers have also solved the order problem of occlusion handling, and the work of Roxas *et al.* [120] was the only X-ray vision technique able to solve the order problem in an outdoor scenario. The dynamic determination of the depth order between real and virtual objects is desirable even for X-ray vision applications. Considering that the hidden scene is rendered at a fixed depth position, which may be, for example, inside a virtual window attached in a part of the real scene, that part of the scene may be occluded by another real object, and the X-ray view should be occluded as well.

OST displays: The lack of mutual occlusion handling for OST displays poses a real problem for AR applications that run on those devices, since, depending on the real-world illumination conditions of the real scene, the rendered virtual objects may appear to be floating in the real world, mainly due to their transparent aspect when visualized on the OST displays. Even so, despite all the efforts and advances already achieved by the literature in the proposition of novel OST displays that are able to handle the occlusion problem with plausible visual quality, we still do not have robust, popular and commercially available OST-HMDs that solve the mutual occlusion problem, while providing a lightweight hardware solution, robust to varifocal rendering and order-based occlusion handling. That fact per se shows that much research still must be conducted to enhance the capabilities of OST-HMDs, advance the state-of-the-art solutions and fill that gap between academy and industry.

Machine / Deep Learning: Recently, machine and deep learning technologies are providing huge advances in several fields of computer science [188], including AR. State-of-the-art techniques are using machine [124] and deep

learning to provide model-based [81] and depth-based [125] order estimation, and even X-ray vision in AR [120].

Every year, newer and more robust machine and deep learning techniques have been proposed in other fields, such as image processing, that are related to AR. In this sense, advances not only in fields such as monocular depth map estimation, but also in other fields such as single-image occlusion estimation [189] and saliency determination [190], could also be integrated into AR applications to provide more tools for researchers and developers to minimize different aspects of the occlusion problem.

Despite the fact that machine learning techniques still could not be integrated to solve the occlusion problem for OST displays, issues such as generalization and real-time performance still must be addressed to enable a broader use of such technologies to handle occlusion in general AR applications.

Temporal Coherence: To improve the accuracy and the consistency of the occlusion handling, some techniques have already explicitly modelled temporal coherence as a part of the problem to be solved. For instance, the work of Luo *et al.* [125] and the depth-based techniques that reconstruct a 3D model of the real scene [104], [105], [122] are generally able to provide temporally coherent depth maps, which result in consistent occlusion handling during the live streaming of the AR application. X-ray vision techniques followed a similar path by incorporating the motion feature (Figure 8-(f)) into their algorithms [129], [154], [155], [156].

Even so, we believe that more techniques should take temporal coherence into consideration, while still being able to provide real-time performance, in order to enhance the visual quality of the occlusion handling and also the user experience with the application.

10 CONCLUDING REMARKS

With this survey, we expect to contribute to the future of AR, by presenting a review and critical analysis of how the occlusion problem has been handled so far in AR, such that reviewers and developers could work on the suggested topics to further bring advances to the state-of-the-art, and to increase the robustness of existing AR applications. We have treated occlusion in AR as an order problem, as an X-ray vision problem and as a visual display problem, and we could provide a detailed presentation and discussion about this topic under these three perspectives.

A recent growing interest in the field of monocular depth estimation has already driven researchers to achieve high-quality results at real-time frame rates, even for images or video streams directly captured from cameras embedded in smartphones. The advances in this field have the potential to make a huge impact in the field of AR, by increasing the popularity, the diversity, and the robustness of several AR applications that already run on handheld devices.

In terms of the X-ray vision techniques in AR, the use of alpha blending to provide a smooth transition between the visualization of visible and hidden objects is the strategy most used in the literature to provide this see-through capability. Virtual windows are also popular to restrict the visualization of the occluded structures inside a region of the augmented scene. Low-level features, such as edges, and

high-level features of the scene, such as motion, texture, curvature, may be computed from both real and virtual objects to compose a saliency map that indicates which are the most salient regions of both real and virtual worlds, and that must be present at the final rendered scene. A few works have rendered perspective lines to enhance the depth perception and spatial awareness of the AR users.

Most of the commercial OST displays still do not provide support for mutual occlusion handling, and the majority of the prototypes proposed in the literature still are too bulky to be employed for real-world applications.

Finally, we could also see that this field of occlusion handling still has room for improvements. A few papers could solve dynamically the order problem of occlusion handling for X-ray vision applications and a few model- and depth-based techniques have been proposed, or evaluated, in outdoor scenarios. No one of the reviewed techniques has performed that evaluation when handling the mutual occlusion for OST displays. A high percentage of papers do not report their performance results. Even so, a few techniques reported real-time performance for X-ray vision applications, and when handling occlusion for OST displays. Nevertheless, we could detect promising trends by reviewing state-of-the-art techniques. The development of real-time monocular depth map estimation algorithms could open new ways of interaction between users and AR applications. The use of machine and deep learning could enhance the accuracy of existing occlusion handling techniques in terms of improved depth map estimation, for instance. Temporal coherence could also be exploited to further boost accuracy and stability of the occlusion handling techniques.

ACKNOWLEDGMENTS

Márcio C. F. Macedo was financially supported by the Post-doctoral National Program of the Coordination for the Improvement of Higher Education Personnel (PNPD/CAPES), grant number 88882.306277/2018-01.

REFERENCES

- [1] R. T. Azuma, "A Survey of Augmented Reality," *Presence: Teleoperators and Virtual Environments*, vol. 6, no. 4, pp. 355–385, 1997.
- [2] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, "Recent advances in augmented reality," *IEEE Computer Graphics and Applications*, vol. 21, no. 6, pp. 34–47, 2001.
- [3] Feng Zhou, H. B. Duh, and M. Billinghurst, "Trends in augmented reality tracking, interaction and display: A review of ten years of ismar," in *ISMAR*, 2008, pp. 193–202.
- [4] D. van Krevelen and R. Poelman, "A Survey of Augmented Reality Technologies, Applications and Limitations," *International Journal of Virtual Reality*, vol. 9, no. 2, pp. 1–20, Jan. 2010.
- [5] M. Billinghurst, A. Clark, and G. Lee, "A Survey of Augmented Reality," *Found. Trends Hum.-Comput. Interact.*, vol. 8, no. 2–3, p. 73–272, Mar. 2015.
- [6] K. Kim, M. Billinghurst, G. Bruder, H. B. Duh, and G. F. Welch, "Revisiting Trends in Augmented Reality Research: A Review of the 2nd Decade of ISMAR (2008–2017)," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 11, pp. 2947–2962, 2018.
- [7] D. Schmalstieg and T. Höllerer, *Augmented Reality - Principles and Practice*. United States: Addison-Wesley Professional, 6 2016.
- [8] S. Aukstakalnis, *Practical Augmented Reality: A Guide to the Technologies, Applications, and Human Factors for AR and VR*, ser. Usability Series. Addison-Wesley, 2017.

- [9] A. Kolb, E. Barth, R. Koch, and R. Larsen, "Time-of-Flight Cameras in Computer Graphics," *Computer Graphics Forum*, vol. 29, no. 1, pp. 141–159, 2010.
- [10] S. Foix, G. Alenya, and C. Torras, "Lock-in Time-of-Flight (ToF) Cameras: A Survey," *IEEE Sensors Journal*, vol. 11, no. 9, pp. 1917–1926, 2011.
- [11] L. Cruz, D. Lucio, and L. Velho, "Kinect and RGBD Images: Challenges and Applications," in *SIBGRAPI Tutorials*, 2012, pp. 36–49.
- [12] R. A. Hamzah and H. Ibrahim, "Literature survey on stereo vision disparity map algorithms," *Journal of Sensors*, vol. 2016, pp. 1–23, 2016.
- [13] M. Zollhöfer, P. Stotko, A. Görlitz, C. Theobalt, M. Nießner, R. Klein, and A. Kolb, "State of the Art on 3D Reconstruction with RGB-D Cameras," *Computer Graphics Forum*, vol. 37, no. 2, pp. 625–652, 2018.
- [14] N. Elmquist and P. Tsigas, "A Taxonomy of 3D Occlusion Management Techniques," in *VR*, 2007, pp. 51–58.
- [15] ———, "A Taxonomy of 3D Occlusion Management for Visualization," *IEEE Trans. Vis. Comput. Graph.*, vol. 14, no. 5, pp. 1095–1109, 2008.
- [16] D. Kalkofen, C. Sandor, S. White, and D. Schmalstieg, "Visualization techniques for augmented reality," in *Handbook of Augmented Reality*. Springer, 2011, pp. 65–98.
- [17] M. A. Livingston, A. Dey, C. Sandor, and B. H. Thomas, *Pursuit of "X-Ray Vision" for Augmented Reality*. Springer, 2013, pp. 67–107.
- [18] S. Zollmann, R. Grasset, T. Langlotz, W. H. Lo, S. Mori, and H. Regenbrecht, "Visualization Techniques in Augmented Reality: A Taxonomy, Methods and Patterns," *IEEE Trans. Vis. Comput. Graph.*, pp. 1–1, 2020.
- [19] K. Kiyokawa, *Occlusion Displays*. Berlin, Heidelberg: Springer, 2012, pp. 2251–2257.
- [20] G. Wetzstein, A. Patney, and Q. Sun, *State of the Art in Perceptual VR Displays*. Cham: Springer, 2020, pp. 221–243.
- [21] M. Jacomy, T. Venturini, S. Heymann, and M. Bastian, "ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software," *PLOS ONE*, vol. 9, no. 6, pp. 1–12, 06 2014.
- [22] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: an open source software for exploring and manipulating networks," in *International AAAI Conference on Weblogs and Social Media*, vol. 8, 2009, pp. 361–362.
- [23] M. Bajura, H. Fuchs, and R. Ohbuchi, "Merging Virtual Objects with the Real World: Seeing Ultrasound Imagery within the Patient," in *SIGGRAPH*. ACM, 1992, p. 203–210.
- [24] D. E. Breen, R. T. Whitaker, E. Rose, and M. Tuceryan, "Interactive occlusion and automatic object placement for augmented reality," *Computer Graphics Forum*, vol. 15, no. 3, pp. 11–22, 1996.
- [25] C. Duchesne and J. . Herve, "A point-based approach to the interposition problem in augmented reality," in *ICPR*, vol. 1, 2000, pp. 261–265.
- [26] P. Fortin and P. Hebert, "Handling Occlusions in Real-time Augmented Reality: Dealing with Movable Real and Virtual Objects," in *CRV*, 2006, pp. 54–54.
- [27] Y. Tian, T. Guan, and C. Wang, "Real-time occlusion handling in augmented reality based on an object tracking approach," *Sensors (Basel, Switzerland)*, vol. 10, no. 4, pp. 2885–2900, 2010.
- [28] C. Du, Y. Chen, M. Ye, and L. Ren, "Edge Snapping-Based Depth Enhancement for Dynamic Occlusion Handling in Augmented Reality," in *ISMAR*, 2016, pp. 54–62.
- [29] C. Battisti, S. Messelodi, and F. Poiesi, "Seamless Bare-Hand Interaction in Mixed Reality," in *ISMAR-Adjunct*, 2018, pp. 198–203.
- [30] S. Malik, C. McDonald, and G. Roth, "Hand tracking for interactive pattern-based augmented reality," in *ISMAR*, 2002, pp. 117–126.
- [31] V. Lepetit and M. . Berger, "A semi-automatic method for resolving occlusion in augmented reality," in *CVPR*, vol. 2, 2000, pp. 225–230.
- [32] H. Schumann, S. Burtescu, and F. Siering, "Applying augmented reality techniques in the field of interactive collaborative design," in *3D Structure from Multiple Images of Large-Scale Environments*. Springer, 1998, pp. 290–303.
- [33] A. Fuhrmann, G. Hesina, F. Faure, and M. Gervautz, "Occlusion in collaborative augmented environments," *Computers & Graphics*, vol. 23, no. 6, pp. 809 – 819, 1999.
- [34] S. Walairacht, K. Yamada, S. Hasegawa, Y. Koike, and M. Sato, "4 + 4 Fingers Manipulating Virtual Objects in Mixed-Reality Environment," *Presence*, vol. 11, no. 2, pp. 134–143, 2002.
- [35] G. Klein and T. Drummond, "Sensor fusion and occlusion refinement for tablet-based ar," in *ISMAR*, 2004, pp. 38–47.
- [36] J. Fischer, D. Bartz, and W. Straßer, "Occlusion Handling for Medical Augmented Reality Using a Volumetric Phantom Model," in *VRST*. New York, NY, USA: ACM, 2004, p. 174–177.
- [37] C. Bichlmeier, F. Wimmer, S. M. Heinig, and N. Navab, "Contextual Anatomic Mimesis Hybrid In-Situ Visualization Method for Improving Multi-Sensory Depth Perception in Medical Augmented Reality," in *ISMAR*, 2007, pp. 129–138.
- [38] R. Frikha, R. Ejbal, and M. Zaied, "Handling occlusion in augmented reality surgical training based instrument tracking," in *ICCSA*, 2016, pp. 1–5.
- [39] D. Kurz, K. Kiyokawa, and H. Takemura, "Mutual Occlusions on Table-Top Displays in Mixed Reality Applications," in *VRST*. New York, NY, USA: ACM, 2008, p. 227–230.
- [40] K. Kim, Y. Park, and W. Woo, "Digilog Miniature: Real-Time, Immersive, and Interactive AR on Miniatures," in *VRCAI*. New York, NY, USA: ACM, 2010, p. 161–168.
- [41] Y. Zhou, S. Xiao, N. Tang, Z. Wei, and X. Chen, "Pmomo: Projection Mapping on Movable 3D Object," in *CHI*. New York, NY, USA: ACM, 2016, p. 781–790.
- [42] M. Wu and V. Popescu, "Efficient VR and AR Navigation Through Multiperspective Occlusion Management," *IEEE Trans. Vis. Comput. Graph.*, vol. 24, no. 12, pp. 3069–3080, 2018.
- [43] J. Gimeno, S. Casas, C. Portolés, and M. Fernández, "Addressing the Occlusion Problem in Augmented Reality Environments with Phantom Hollow Objects," in *ISMAR-Adjunct*, 2018, pp. 21–24.
- [44] K. Kiyokawa, Y. Kurata, and H. Ohno, "An optical see-through display for mutual occlusion of real and virtual environments," in *ISAR*, 2000, pp. 60–67.
- [45] O. Bimber and B. Frohlich, "Occlusion shadows: using projected light to generate realistic occlusion effects for view-dependent optical see-through displays," in *ISMAR*, 2002, pp. 186–319.
- [46] Y. Ohta, Y. Sugaya, H. Igarashi, T. Ohtsuki, and K. Taguchi, "Share-Z: Client/Server Depth Sensing for See-Through Head-Mounted Displays," *Presence*, vol. 11, no. 2, pp. 176–188, 2002.
- [47] K. Murase, T. Ogi, K. Saito, T. Koyama, K. S. Biru, and J. K. S. Biru, "Correct occlusion effect in the optical see-through immersive augmented reality display system," in *ICAT*, 2008.
- [48] G. Avveduto, F. Tecchia, and H. Fuchs, "Real-World Occlusion in Optical See-through AR Displays," in *VRST*. New York, NY, USA: ACM, 2017.
- [49] M. Inami, N. Kawakami, D. Sekiguchi, Y. Yanagida, T. Maeda, and S. Tachi, "Visuo-haptic display using head-mounted projector," in *VR*, 2000, pp. 233–240.
- [50] Hong Hua, Chunyu Gao, L. D. Brown, N. Ahuja, and J. P. Rolland, "A testbed for precise registration, natural occlusion and interaction in an augmented environment using a head-mounted projective display (hmpd)," in *VR*, 2002, pp. 81–89.
- [51] A. L. D. Santos, D. Lemos, J. E. F. Lindoso, and V. Teichrieb, "Real Time Ray Tracing for Augmented Reality," in *SVR*, 2012, pp. 131–140.
- [52] S. Zollmann and G. Reitmayr, "Dense Depth Maps from Sparse Models and Image Coherence for Augmented Reality," in *VRST*. New York, NY, USA: ACM, 2012, p. 53–60.
- [53] J. Kasperi, M. P. Edwardsson, and M. Romero, "Occlusion in Outdoor Augmented Reality Using Geospatial Building Data," in *VRST*. New York, NY, USA: ACM, 2017.
- [54] Q. Feng, H. P. H. Shum, and S. Morishima, "Resolving Occlusion for 3D Object Manipulation with Hands in Mixed Reality," in *VRST*. New York, NY, USA: ACM, 2018.
- [55] J. Pilet, C. Strecha, and P. Fua, "Making Background Subtraction Robust to Sudden Illumination Changes," in *ECCV*. Springer, 2008, pp. 567–580.
- [56] J. Vallino and C. Brown, "Haptics in augmented reality," in *IEEE International Conference on Multimedia Computing and Systems*, vol. 1, 1999, pp. 195–200.
- [57] H. L. Wang, K. Sengupta, P. Kumar, and R. Sharma, "Occlusion Handling in Augmented Reality Using Background-Foreground Segmentation and Projective Geometry," *Presence*, vol. 14, no. 3, pp. 264–277, 2005.
- [58] J. Zhu and Z. Pan, "Occlusion Registration in Video-Based Augmented Reality," in *VRCAI*. ACM, 2008.

- [59] J. Zhu, Z. Pan, C. Sun, and W. Chen, "Handling Occlusions in Video-Based Augmented Reality Using Depth Information," *Comput. Animat. Virtual Worlds*, vol. 21, no. 5, p. 509–521, Sep. 2010.
- [60] J. Ventura and T. Hollerer, "Online environment model estimation for augmented reality," in *ISMAR*, 2009, pp. 103–106.
- [61] A. Ladikos and N. Navab, "Real-time 3d reconstruction for occlusion-aware interactions in mixed reality," in *Advances in Visual Computing*. Springer, 2009, pp. 480–489.
- [62] K. Cordes, B. Scheuermann, B. Rosenthal, and J. Ostermann, "Occlusion Handling for the Integration of Virtual Objects into Video," in *VISAPP*, 2012, pp. 173–180.
- [63] J.-E. Kilmann, D. Heitkamp, and P. Lensing, "An Augmented Reality Application for Mobile Visualization of GIS-Referenced Landscape Planning Projects," in *VRCAI*. ACM, 2019.
- [64] C. McDonald and G. Roth, "Replacing a Mouse with Hand Gesture in a Plane-Based Augmented Reality System," in *International Conference on Vision Interface*, 2003.
- [65] S. R. R. Sanches, D. M. Tokunaga, V. F. Silva, A. C. Sementille, and R. Tori, "Mutual occlusion between real and virtual elements in augmented reality based on fiducial markers," in *WACV*, 2012, pp. 49–54.
- [66] G. A. Lee, M. Billinghurst, and G. J. Kim, "Occlusion Based Interaction Methods for Tangible Augmented Reality Environments," in *VRCAI*. ACM, 2004, p. 419–426.
- [67] S. Garrido-Jurado, R. Muñoz-Salinas, F. Madrid-Cuevas, and M. Marín-Jiménez, "Automatic generation and detection of highly reliable fiducial markers under occlusion," *Pattern Recognition*, vol. 47, no. 6, pp. 2280 – 2292, 2014.
- [68] J. Fischer, H. Regenbrecht, and G. Baratoff, "Detecting Dynamic Occlusion in Front of Static Backgrounds for AR Scenes," in *EGVE*. ACM, 2003, p. 153–161.
- [69] V. Gay-Bellile, A. Bartoli, and P. Sayd, "Deformable Surface Augmentation in Spite of Self-Occlusions," in *ISMAR*, 2007, pp. 235–238.
- [70] J. Pilet, V. Lepetit, and P. Fua, "Retexturing in the Presence of Complex Illumination and Occlusions," in *ISMAR*, 2007, pp. 249–258.
- [71] Y. Fujimoto, G. Yamamoto, T. Taketomi, C. Sandor, and H. Kato, "Pseudo printed fabrics through projection mapping," in *ISMAR*, 2015, pp. 174–175.
- [72] G. Narita, Y. Watanabe, and M. Ishikawa, "Dynamic Projection Mapping onto a Deformable Object with Occlusion Based on High-Speed Tracking of Dot Marker Array," in *VRST*. New York, NY, USA: ACM, 2015, p. 149–152.
- [73] G. Narita, Y. Watanabe, and M. Ishikawa, "Dynamic Projection Mapping onto Deforming Non-Rigid Surface Using Deformable Dot Cluster Marker," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 3, pp. 1235–1248, 2017.
- [74] M. Kanbara, T. Okuma, H. Takemura, and N. Yokoya, "Real-time composition of stereo images for video see-through augmented reality," in *International Conference on Multimedia Computing and Systems*, vol. 1, 1999, pp. 213–219.
- [75] —, "A stereoscopic video see-through augmented reality system based on real-time vision-based registration," in *VR*, 2000, pp. 255–262.
- [76] Y. Kojima, Y. Yasumuro, H. Sasaki, I. Kanaya, O. Oshiro, T. Kuroda, S. Manabe, and K. Chihara, "Hand manipulation of virtual objects in wearable augmented reality," in *VSMM*, 2001, pp. 463–469.
- [77] Woohun Lee and Jun Park, "Augmented foam: a tangible augmented reality for product design," in *ISMAR*, 2005, pp. 106–109.
- [78] A. F. Abate, F. Narducci, and S. Ricciardi, "An Image Based Approach to Hand Occlusions in Mixed Reality Environments," in *Proceedings, Part I, of the 6th International Conference on Virtual, Augmented and Mixed Reality. Designing and Developing Virtual and Augmented Environments - Volume 8525*. Berlin, Heidelberg: Springer-Verlag, 2014, p. 319–328.
- [79] O. Kutter, A. Aichert, C. Bichlmeier, C. Bichlmeier, S. M. Heinig, B. Ockert, E. Euler, and N. Navab, "Real-time Volume Rendering for High Quality Visualization in Augmented Reality," in *AM-ARCS*. New York, USA: MICCAI Society, Sept. 2008.
- [80] J. Ventura and T. Höllerer, "Depth compositing for augmented reality," in *SIGGRAPH Posters*. ACM, 2008.
- [81] X. Tang, X. Hu, C.-W. Fu, and D. Cohen-Or, "GrabAR: Occlusion-aware Grabbing Virtual Objects in AR," *ArXiv*, vol. abs/1912.10637, 2019.
- [82] M. O. Berger, "Resolving occlusion in augmented reality: a contour based approach without 3d reconstruction," in *Proc. of CVPR*, 1997, pp. 91–96.
- [83] V. Lepetit and M. . Berger, "Handling occlusion in augmented reality systems: a semi-automatic method," in *ISAR*, 2000, pp. 137–146.
- [84] K. C. Ong, H. C. Teh, and T. S. Tan, "Resolving occlusion in image sequence made easy," *The Visual Computer*, vol. 14, no. 4, pp. 153–165, Oct 1998.
- [85] K. Kiyokawa, M. Billinghurst, B. Campbell, and E. Woods, "An occlusion capable optical see-through head mount display for supporting co-located collaboration," in *ISMAR*, 2003, pp. 133–141.
- [86] J. Wither, C. Coffin, J. Ventura, and T. Hollerer, "Fast annotation and modeling with a single-point laser range finder," in *ISMAR*, 2008, pp. 65–68.
- [87] T. Schöps, J. Engel, and D. Cremers, "Semi-dense visual odometry for ar on a smartphone," in *ISMAR*, 2014, pp. 145–150.
- [88] B. V. Lu, T. Kakuta, R. Kawakami, T. Oishi, and K. Ikeuchi, "Foreground and shadow occlusion handling for outdoor augmented reality," in *ISMAR*, 2010, pp. 109–118.
- [89] M. M. Wloka and B. G. Anderson, "Resolving occlusion in augmented reality," in *3D*. ACM, 1995, p. 5–12.
- [90] J. Schmidt, H. Niemann, and S. Vogt, "Dense disparity maps in real-time with an application to augmented reality," in *WACV*, 2002, pp. 225–230.
- [91] H. Kim and K. Sohn, "Hierarchical depth estimation for image synthesis in mixed reality," in *Stereoscopic Displays and Virtual Reality Systems X*, vol. 5006. SPIE, 2003, pp. 544 – 553.
- [92] K. Hayashi, H. Kato, and S. Nishida, "Occlusion Detection of Real Objects Using Contour Based Stereo Matching," in *ICAT*. New York, NY, USA: ACM, 2005, p. 180–186.
- [93] L. Li, T. Guan, and B. Ren, "Resolving occlusion between virtual and real scenes for augmented reality applications," in *HCI*. Springer, 2007, pp. 634–642.
- [94] Y. Zhou, J.-T. Ma, Q. Hao, H. Wang, and X.-P. Liu, "A novel optical see-through head-mounted display with occlusion and intensity matching support," in *Technologies for E-Learning and Digital Entertainment*. Springer, 2007, pp. 56–62.
- [95] C.-K. Yang, Y.-H. Chen, T.-J. Chuang, K. Shankhwar, and S. Smith, "An augmented reality-based training system with a natural user interface for manual milling operations," *Virtual Reality*, vol. 24, no. 3, pp. 527–539, Sep 2020.
- [96] K. Kiyokawa, Y. Kurata, and H. Ohno, "An optical see-through display for mutual occlusion with a real-time stereovision system," *Computers & Graphics*, vol. 25, no. 5, pp. 765 – 779, 2001.
- [97] J. D. Mulder, "Realistic occlusion effects in mirror-based co-located augmented reality systems," in *VR*, 2005, pp. 203–208.
- [98] —, "Occlusion in Mirror-Based Co-Located Augmented Reality Systems," *Presence*, vol. 15, no. 1, pp. 93–107, 2006.
- [99] A. State, M. A. Livingston, W. F. Garrett, G. Hirota, M. C. Whittton, E. D. Pisano, and H. Fuchs, "Technologies for Augmented Reality Systems: Realizing Ultrasound-Guided Needle Biopsies," in *SIGGRAPH*. New York, NY, USA: ACM, 1996, p. 439–446.
- [100] J. Fischer, B. Huhle, and A. Schilling, "Using Time-of-Flight Range Data for Occlusion Handling in Augmented Reality," in *EGVE*. Eurographics Association, 2007, p. 109–116.
- [101] B. Bartczak, I. Schiller, C. Beder, and R. Koch, "Integration of a time-of-flight camera into a mixed reality system for handling dynamic scenes, moving viewpoints and occlusions in real-time," in *3DPVT Workshop*, 2008.
- [102] R. Koch, I. Schiller, B. Bartczak, F. Kellner, and K. Köser, "MixIn3D: 3D Mixed Reality with ToF-Camera," in *Dynamic 3D Imaging*. Springer, 2009, pp. 126–141.
- [103] U. Hahne and M. Alexa, "Depth imaging by combining time-of-flight and on-demand stereo," in *Dynamic 3D Imaging*. Springer, 2009, pp. 70–83.
- [104] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *ISMAR*, 2011, pp. 127–136.
- [105] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon, "KinectFusion: Real-Time 3D Reconstruction and Interaction Using a Moving Depth Camera," in *UIST*. New York, NY, USA: ACM, 2011, p. 559–568.

- [106] J. Gimeno, P. Morillo, J. M. Orduña, and M. Fernández, "An Occlusion-Aware AR Authoring Tool for Assembly and Repair Tasks," in *GRAPP*, 2012, pp. 377–386.
- [107] S. Corbett-Davies, A. Dünser, and A. Clark, "An interactive augmented reality system for exposure treatment," in *ISMAR-AMH*, 2012, pp. 95–96.
- [108] S. Corbett-Davies, A. Dunser, R. Green, and A. Clark, "An advanced interaction framework for augmented reality based exposure treatment," in *VR*, 2013, pp. 19–22.
- [109] D. W. Seo and J. Y. Lee, "Direct hand touchable interactions in augmented reality environments for natural and intuitive user experiences," *Expert Systems with Applications*, vol. 40, no. 9, pp. 3784 – 3793, 2013.
- [110] A. Maimone, X. Yang, N. Dierk, A. State, M. Dou, and H. Fuchs, "General-purpose telepresence with head-worn optical see-through displays and projector-based lighting," in *VR*, 2013, pp. 23–26.
- [111] W. Yii, Wai Ho Li, and T. Drummond, "Distributed visual processing for augmented reality," in *ISMAR*, 2012, pp. 41–48.
- [112] J. A. Leal-Meléndrez, L. Altamirano-Robles, and J. A. Gonzalez, "Occlusion handling in video-based augmented reality using the kinect sensor for indoor registration," in *CIARP*. Springer, 2013, pp. 447–454.
- [113] M. C. d. F. Macedo and A. L. A. Junior, "Improving On-Patient Medical Data Visualization in a Markerless Augmented Reality Environment by Volume Clipping," in *SIBGRAPI*, 2014, pp. 149–156.
- [114] M. C. Macedo and A. L. Apolinário, "Focus plus context visualization based on volume clipping for markerless on-patient medical data visualization," *Computers & Graphics*, vol. 53, pp. 196 – 209, 2015.
- [115] T. Ha, S. Feiner, and W. Woo, "WeARHand: Head-worn, RGB-D camera-based, bare-hand user interface with visually enhanced depth perception," in *ISMAR*, 2014, pp. 219–228.
- [116] A. K. Hebborn, N. Höhner, and S. Müller, "Occlusion Matting: Realistic Occlusion Handling for Augmented Reality Applications," in *ISMAR*, 2017, pp. 62–71.
- [117] D. R. Walton and A. Steed, "Accurate Real-Time Occlusion for Mixed Reality," in *VRST*. ACM, 2017.
- [118] T. Kakuta, L. B. Vinh, R. Kawakami, T. Oishi, and K. Ikeuchi, "Detection of Moving Objects and Cast Shadows Using a Spherical Vision Camera for Outdoor Mixed Reality," in *VRST*. New York, NY, USA: ACM, 2008, p. 219–222.
- [119] K. Ikeuchi, T. Oishi, M. Kagesawa, A. Banno, R. Kawakami, T. Kakuta, Y. Okamoto, and B. V. Lu, "Outdoor Gallery and Its Photometric Issues," in *VRCAI*. ACM, 2010, p. 361–364.
- [120] M. Roxas, T. Hori, T. Fukuiage, Y. Okamoto, and T. Oishi, "Occlusion Handling Using Semantic Segmentation and Visibility-Based Rendering for Mixed Reality," in *VRST*. ACM, 2018.
- [121] R. A. Newcombe and A. J. Davison, "Live dense reconstruction with a single moving camera," in *CVPR*, 2010, pp. 1498–1505.
- [122] J. Wang, H. Liu, L. Cong, Z. Xiahou, and L. Wang, "CNN-MonoFusion: Online Monocular Dense Reconstruction Using Learned Depth from Single View," in *ISMAR-Adjunct*, 2018, pp. 57–62.
- [123] A. Holynski and J. Kopf, "Fast Depth Densification for Occlusion-Aware Augmented Reality," *ACM Trans. Graph.*, vol. 37, no. 6, Dec. 2018.
- [124] J. Valentin, A. Kowdle, J. T. Barron, N. Wadhwa, M. Dzitsiuk, M. Schoenberg, V. Verma, A. Csaszar, E. Turner, I. Dryanovski, J. Afonso, J. Pascoal, K. Tsotsos, M. Leung, M. Schmidt, O. Guleryuz, S. Khamis, V. Tankovich, S. Fanello, S. Izadi, and C. Rhemann, "Depth from Motion for Smartphone AR," *ACM Trans. Graph.*, vol. 37, no. 6, Dec. 2018.
- [125] X. Luo, J.-B. Huang, R. Szeliski, K. Matzen, and J. Kopf, "Consistent Video Depth Estimation," *ACM Trans. Graph.*, vol. 39, no. 4, Jul. 2020.
- [126] T. Fukuiage, T. Oishi, and K. Ikeuchi, "Reduction of contradictory partial occlusion in mixed reality by using characteristics of transparency perception," in *ISMAR*, 2012, pp. 129–139.
- [127] D. Kalkofen, E. Mendez, and D. Schmalstieg, "Interactive Focus and Context Visualization for Augmented Reality," in *ISMAR*, 2007, pp. 191–201.
- [128] A. State, D. T. Chen, C. Tector, A. Brandt, Hong Chen, R. Ohbuchi, M. Bajura, and H. Fuchs, "Observing a volume rendered fetus within a pregnant patient," in *VIS*, 1994, pp. 364–368.
- [129] C. Sandor, A. Cunningham, A. Dey, and V. Mattila, "An Augmented Reality X-Ray system based on visual saliency," in *ISMAR*, 2010, pp. 27–36.
- [130] R. Bane and T. Hollerer, "Interactive tools for virtual x-ray vision in mobile augmented reality," in *ISMAR*, 2004, pp. 231–239.
- [131] S. Zollmann, D. Kalkofen, E. Mendez, and G. Reitmayr, "Image-based ghostings for single layer occlusions in augmented reality," in *ISMAR*, 2010, pp. 19–26.
- [132] D. Kalkofen, M. Tatzgern, and D. Schmalstieg, "Explosion Diagrams in Augmented Reality," in *VR*, 2009, pp. 71–78.
- [133] Y. Kameda, T. Takemasa, and Y. Ohta, "Outdoor see-through vision utilizing surveillance cameras," in *ISMAR*, 2004, pp. 151–160.
- [134] S. Feiner, B. MacIntyre, and D. Seligmann, "Knowledge-Based Augmented Reality," *Commun. ACM*, vol. 36, no. 7, p. 53–62, Jul. 1993.
- [135] A. Webster, S. Feiner, B. MacIntyre, W. Massie, and T. Krueger, "Augmented reality in architectural construction, inspection and renovation," in *Third Congress on Computing in Civil Engineering*, vol. 1, 1996, p. 996.
- [136] C. Bichlmeier and N. Navab, "Virtual Window for Improved Depth Perception in Medical AR," in *AMI-ARCS*. Copenhagen, Denmark: MICCAI Society, Oct. 2006.
- [137] C. Bichlmeier, T. Sielhorst, S. M. Heining, and N. Navab, "Improving depth perception in medical ar," in *BVM*. Springer, 2007, pp. 217–221.
- [138] M. Lerotic, A. J. Chung, G. Mylonas, and G.-Z. Yang, "pq-space Based Non-Photorealistic Rendering for Augmented Reality," in *MICCAI*. Springer, 2007, pp. 102–109.
- [139] C. Bichlmeier, M. Kipot, S. Holdstock, S. M. Heining, E. Euler, and N. Navab, "A Practical Approach for Intraoperative Contextual In-Situ Visualization," in *AMI-ARCS*, Sept. 2009.
- [140] M. Kersten-Oertel, I. Gerard, S. Drouin, K. Mok, D. Sirhan, D. S. Sinclair, and D. L. Collins, "Augmented reality in neurovascular surgery: feasibility and first uses in the operating room," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 10, no. 11, pp. 1823–1836, Nov 2015.
- [141] E. Özgür, A. Lafont, and A. Bartoli, "Visualizing In-Organ Tumors in Augmented Monocular Laparoscopy," in *ISMAR-Adjunct*, 2017, pp. 46–51.
- [142] E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose, "Toolglass and Magic Lenses: The See-through Interface," in *SIGGRAPH*. New York, NY, USA: ACM, 1993, p. 73–80.
- [143] J. Viega, M. J. Conway, G. Williams, and R. Pausch, "3D Magic Lenses," in *UIST*. ACM, 1996, p. 51–58.
- [144] E. Mendez, D. Kalkofen, and D. Schmalstieg, "Interactive context-driven visualization tools for augmented reality," in *ISMAR*, 2006, pp. 209–218.
- [145] B. Avery, W. PiekarSKI, and B. H. Thomas, "Visualizing Occluded Physical Objects in Unfamiliar Outdoor Augmented Reality Environments," in *ISMAR*, 2007, pp. 285–286.
- [146] D. Kalkofen, E. Mendez, and D. Schmalstieg, "Comprehensible Visualization for Augmented Reality," *IEEE Trans. Vis. Comput. Graph.*, vol. 15, no. 2, pp. 193–204, 2009.
- [147] B. Avery, C. Sandor, and B. H. Thomas, "Improving Spatial Perception for Augmented Reality X-Ray Vision," in *VR*, 2009, pp. 79–82.
- [148] P. Barnum, Y. Sheikh, A. Datta, and T. Kanade, "Dynamic seethroughs: Synthesizing hidden views of moving objects," in *ISMAR*, 2009, pp. 111–114.
- [149] L. F. Maia, W. Viana, and F. Trinta, "A Real-Time x-Ray Mobile Application Using Augmented Reality and Google Street View," in *VRST*. New York, NY, USA: ACM, 2016, p. 111–119.
- [150] C. Sandor, A. Cunningham, U. Eck, D. Urquhart, G. Jarvis, A. Dey, S. Barbier, M. R. Marner, and S. Rhee, "Egocentric space-distorting visualizations for rapid environment exploration in mobile mixed reality," in *ISMAR*, 2009, pp. 211–212.
- [151] C. Sandor, A. Dey, A. Cunningham, S. Barbier, U. Eck, D. Urquhart, M. R. Marner, G. Jarvis, and S. Rhee, "Egocentric space-distorting visualizations for rapid environment exploration in mobile mixed reality," in *VR*, 2010, pp. 47–50.
- [152] E. Mendez and D. Schmalstieg, "Importance Masks for Revealing Occluded Objects in Augmented Reality," in *VRST*. New York, NY, USA: ACM, 2009, p. 247–248.
- [153] D. Kalkofen, E. Veas, S. Zollmann, M. Steinberger, and D. Schmalstieg, "Adaptive ghosted views for Augmented Reality," in *ISMAR*, 2013, pp. 1–9.

- [154] A. Padilha and V. Teichrieb, "Motion detection based ghosted views for occlusion handling in augmented reality," in *ISMAR*, 2014, pp. 291–292.
- [155] ——, "Motion-Aware Ghosted Views for Single Layer Occlusions in Augmented Reality," in *ISMAR Workshops*, 2015, pp. 60–67.
- [156] J. Chen, X. Granier, N. Lin, and Q. Peng, "On-Line Visualization of Underground Structures Using Context Features," in *VRST*. New York, NY, USA: ACM, 2010, p. 167–170.
- [157] C. Hansen, J. Wieferich, F. Ritter, C. Rieder, and H.-O. Peitgen, "Illustrative visualization of 3d planning models for augmented reality in liver surgery," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 5, no. 2, pp. 133–141, Mar 2010.
- [158] B. Marques, N. Haouchine, R. Plantefève, and S. Cotin, "Improving Depth Perception during Surgical Augmented Reality," in *SIGGRAPH Posters*. New York, NY, USA: ACM, 2015.
- [159] R. Wang, Z. Geng, Z. Zhang, and R. Pei, "Visualization Techniques for Augmented Reality in Endoscopic Surgery," in *Medical Imaging and Augmented Reality*. Springer, 2016, pp. 129–138.
- [160] C. Furmanski, R. Azuma, and M. Daily, "Augmented-reality visualizations guided by cognition: perceptual heuristics for combining visible and obscured information," in *ISMAR*, 2002, pp. 215–320.
- [161] M. A. Livingston, J. E. Swan II, J. L. Gabbard, T. H. Höllerer, D. Hix, S. J. Julier, Y. Baillot, and D. Brown, "Resolving Multiple Occluded Layers in Augmented Reality," in *ISMAR*, 2003, p. 56.
- [162] T. Tsuda, H. Yamamoto, Y. Kameda, and Y. Ohta, "Visualization Methods for Outdoor See-through Vision," in *ICAT*. ACM, 2005, p. 62–69.
- [163] T. Sielhorst, C. Bichlmeier, S. M. Heinig, and N. Navab, "Depth perception – a major issue in medical ar: Evaluation study by twenty surgeons," in *MICCAI*, 2006, pp. 364–372.
- [164] B. Avery, B. H. Thomas, and W. PiekarSKI, "User evaluation of see-through vision for mobile outdoor augmented reality," in *ISMAR*, 2008, pp. 69–72.
- [165] A. Dey, A. Cunningham, and C. Sandor, "Evaluating depth perception of photorealistic mixed reality visualizations for occluded objects in outdoor environments," in *3DUI*, 2010, pp. 127–128.
- [166] A. Dey, G. Jarvis, C. Sandor, A. Wibowo, and V.-V. Mattila, "An Evaluation of Augmented Reality X-Ray Vision for Outdoor Navigation," in *ICAT*, November 2011, pp. 28–32.
- [167] M. E. C. Santos, A. Chen, M. Terawaki, G. Yamamoto, T. Takekomi, J. Miyazaki, and H. Kato, "Augmented Reality X-Ray Interaction in K-12 Education: Theory, Student Perception and Teacher Evaluation," in *ICALT*, 2013, pp. 141–145.
- [168] A. Dey and C. Sandor, "Lessons Learned: Evaluating Visualizations for Occluded Objects in Handheld Augmented Reality," *Int. J. Hum.-Comput. Stud.*, vol. 72, no. 10–11, p. 704–716, Oct. 2014.
- [169] S. Zollmann, R. Grasset, G. Reitmair, and T. Langlotz, "Image-Based X-Ray Visualization Techniques for Spatial Understanding in Outdoor Augmented Reality," in *OzCHI*. ACM, 2014, p. 194–203.
- [170] M. T. Eren and S. Balcisoy, "Evaluation of X-Ray Visualization Techniques for Vertical Depth Judgments in Underground Exploration," *Vis. Comput.*, vol. 34, no. 3, p. 405–416, Mar. 2018.
- [171] Y. Itoh, T. Hamasaki, and M. Sugimoto, "Occlusion Leak Compensation for Optical See-Through Displays Using a Single-Layer Transmissive Spatial Light Modulator," *IEEE Trans. Vis. Comput. Graph.*, vol. 23, no. 11, pp. 2463–2473, 2017.
- [172] Y. Yamaguchi and Y. Takaki, "See-through integral imaging display with background occlusion capability," *Appl. Opt.*, vol. 55, no. 3, pp. A144–A149, Jan 2016.
- [173] O. Cakmakci, Yonggang Ha, and J. P. Rolland, "A compact optical see-through head-worn display with occlusion support," in *ISMAR*, 2004, pp. 16–25.
- [174] O. Cakmakci, Y. Ha, and J. Rolland, "Design of a compact optical see-through head-worn display with mutual occlusion capability," in *Novel Optical Systems Design and Optimization VIII*, vol. 5875. SPIE, 2005, pp. 122 – 127.
- [175] C. Gao, Y. Lin, and H. Hua, "Occlusion capable optical see-through head-mounted display using freeform optics," in *ISMAR*, 2012, pp. 281–282.
- [176] C. Gao, Y. Lin, and H. Hua, "Optical see-through head-mounted display with occlusion capability," in *Head- and Helmet-Mounted Displays XVIII: Design and Applications*, vol. 8735. SPIE, 2013, pp. 107 – 115.
- [177] A. Wilson and H. Hua, "Design and prototype of an augmented reality display with per-pixel mutual occlusion capability," *Opt. Express*, vol. 25, no. 24, pp. 30 539–30 549, Nov 2017.
- [178] B. Krajancich, N. Padmanaban, and G. Wetzstein, "Factored Occlusion: Single Spatial Light Modulator Occlusion-capable Optical See-through Augmented Reality Display," *IEEE Trans. Vis. Comput. Graph.*, vol. 26, no. 5, pp. 1871–1879, 2020.
- [179] Y.-G. Ju, M.-H. Choi, P. Liu, B. Hellman, T. L. Lee, Y. Takashima, and J.-H. Park, "Occlusion-capable optical-see-through near-eye display using a single digital micromirror device," *Opt. Lett.*, vol. 45, no. 13, pp. 3361–3364, Jul 2020.
- [180] A. Maimone and H. Fuchs, "Computational augmented reality eyeglasses," in *ISMAR*, 2013, pp. 29–38.
- [181] I. D. Howlett and Q. Smithwick, "Perspective correct occlusion-capable augmented reality displays using cloaking optics constraints," *J. Soc. Inf. Disp.*, vol. 25, no. 3, pp. 185–193, 2017.
- [182] T. Hamasaki and Y. Itoh, "Varifocal Occlusion for Optical See-Through Head-Mounted Displays using a Slide Occlusion Mask," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 5, pp. 1961–1969, 2019.
- [183] K. Rathinavel, G. Wetzstein, and H. Fuchs, "Varifocal Occlusion-Capable Optical See-through Augmented Reality Display based on Focus-tunable Optics," *IEEE Trans. Vis. Comput. Graph.*, vol. 25, no. 11, pp. 3125–3134, nov 2019.
- [184] H. Kakeya, S. Ishizuka, and Y. Sato, "Realization of an aerial 3D image that occludes the background scenery," *Opt Express*, vol. 22, no. 20, pp. 24 491–24 496, Oct 2014.
- [185] Q. Y. J. Smithwick, D. Reetz, and L. Smoot, "LCD masks for spatial augmented reality," in *Stereoscopic Displays and Applications XXV*, vol. 9011. SPIE, 2014, pp. 202 – 218.
- [186] T. Akenine-Möller, E. Haines, and N. Hoffman, *Real-Time Rendering, Fourth Edition*, 4th ed. USA: A. K. Peters, Ltd., 2018.
- [187] S. Bouaziz, A. Tagliasacchi, H. Li, and M. Pauly, "Modern Techniques and Applications for Real-Time Non-Rigid Registration," in *SIGGRAPH ASIA 2016 Courses*. ACM, 2016.
- [188] S. Pouyanfar, S. Sadiq, Y. Yan, H. Tian, Y. Tao, M. P. Reyes, M.-L. Shyu, S.-C. Chen, and S. S. Iyengar, "A Survey on Deep Learning: Algorithms, Techniques, and Applications," *ACM Comput. Surv.*, vol. 51, no. 5, Sep. 2018.
- [189] R. Lu, F. Xue, M. Zhou, A. Ming, and Y. Zhou, "Occlusion-Shared and Feature-Separated Network for Occlusion Relationship Reasoning," in *ICCV*, 2019, pp. 10 342–10 351.
- [190] A. Borji, "Saliency Prediction in the Deep Learning Era: Successes and Limitations," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2019.



Márcio Macedo received his doctoral degree in Computer Science from the Federal University of Bahia, Brazil, in 2018, and was a Postdoctoral Scholar at the same university from 2018 until 2020. His research interests include nearly everything related to computer graphics, image processing, augmented reality, real-time rendering, and parallel processing.



Antônio Apolinário Jr. is an Associate Professor of Computer Science at the Federal University of Bahia, in Brazil. He received his doctoral degree from the Federal University of Rio de Janeiro, Brazil, in 2004. His research interests include computer graphics, augmented reality, real-time rendering, 3D modeling, and physics-based simulation.