VR Exploration Assistance through Automatic Occlusion Removal

Lili Wang, Jian Wu, Xuefeng Yang, and Voicu Popescu

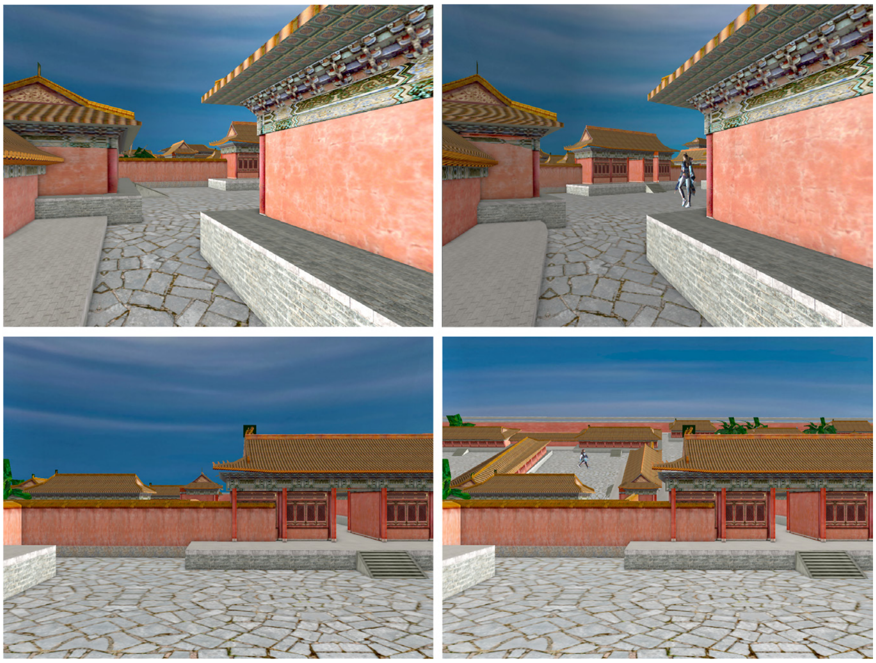


Fig. 1. Conventional (left) and multiperspective (right) visualization. Occlusions are alleviated with a left-right (top) and a top-down (bottom) disocclusion effect.

Abstract— Virtual Reality (VR) applications allow a user to explore a scene intuitively through a tracked head-mounted display (HMD). However, in complex scenes, occlusions make scene exploration inefficient, as the user has to navigate around occluders to gain line of sight to potential regions of interest. When a scene region proves to be of no interest, the user has to retrace their path, and such a sequential scene exploration implies significant amounts of wasted navigation. Furthermore, as the virtual world is typically much larger than the tracked physical space hosting the VR application, the intuitive one-to-one mapping between the virtual and real space has to be temporarily suspended for the user to teleport or redirect in order to conform to the physical space constraints. In this paper we introduce a method for improving VR exploration efficiency by automatically constructing a multiperspective visualization that removes occlusions. For each frame, the scene is first rendered conventionally, the z-buffer is analyzed to detect horizontal and vertical depth discontinuities, the discontinuities are used to define disocclusion portals which are 3D scene rectangles for routing rays around occluders, and the disocclusion portals are used to render a multiperpsective image that alleviates occlusions. The user controls the multiperspective disocclusion effect, deploying and retracting it with small head translations. We have quantified the VR exploration efficiency brought by our occlusion removal method in a study where participants searched for a stationary target, and chased a dynamic target. Our method showed an advantage over conventional VR exploration in terms of reducing the navigation distance, the view direction rotation, the number of redirections, and the task completion time. These advantages did not come at the cost of a reduction in depth perception or situational awareness, or of an increase in simulator sickness.

Index Terms—VR exploration, occlusion removal, disocclusion portal, multiperspective visualization

1 Introduction

When tracked with six degrees of freedom, a head-mounted display (HMD) does not only immerse the user into the visualization of the virtual environment, but also allows the user to specify the desired view naturally, by walking and by rotating their head. However, such exploration in virtual reality (VR) can be inefficient. In a complex scene occlusions abound, and the user has to navigate the virtual viewpoint around many occluders to inspect the parts of the scene that they hide. When a part of the scene turns out to be of no interest, the user has to retrace their path. Such sequential navigation can lead to a lot of wasted time and physical effort, and also to disorientation due to the many back and forths. Furthermore, physical limitations such as a small tracked area or obstacles in the real world hosting the VR

- *Lili Wang, Jian Wu and Xuefeng Yang are with State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beijing Advanced Innovation Center for Biomedical Engineering, Beihang University, Beijing 100191, China. E-mail: wanglily,lanayawj,yangxuefeng@buaa.edu.cn.*
- *Voicu Popescu is with Purdue University, West Lafayette, IN 47907. E-mail: popescu@purdue.edu.*

Manuscript received 10 Sept. 2018; accepted 7 Feb. 2019.
Date of publication 17 Feb. 2019; date of current version 27 Mar. 2019.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2019.2898782

application such as walls or pieces of furniture, make it impossible for the user to assume all the viewpoints needed for VR exploration, and the naturalistic interface has to be suspended temporarily for the user to teleport beyond these physical limitations. However, teleportation introduces visualization discontinuity, which leads to confusion, a mismatch between acceleration and visuals, which leads to simulator sickness, and a switch from first to second or third person, which perturbs the sense of immersion in the VR scene.

Occlusion removal is a well studied paradigm for increasing 3D scene exploration efficiency in non-immersive visualization, and researchers have begun adapting some of the techniques developed to the context of HMD visualization. One approach for occlusion removal is X-ray visualization [1], which lets the user see through occluders. The challenge of the approach is poor scalability with the number of occluding layers. Another approach is cut-away visualization [4], which simply removes the occluders, at the cost of a poor in-context localization of the disoccluded region. A third approach for mitigating occlusions is based on an exploded visualization that moves the occluding parts radially away from the region of interest (ROI) [17], an approach that works best in the context of CAD applications where there is a clear partition of the scene into parts, and a clear assembly order. Top-view visualization is an approach of conveying to the user the overall scene layout and their current location through a map rendered with a vertical view direction, which is either always shown in the corner of the main visualization, or invoked by the user as needed [19]. In the context of HMD visualization, the top-view approach has the challenge of a small screen real estate, which implies that the map is shown at low resolution, and of switching to a third person view, which breaks the sense of immersion.

A promising approach for alleviating occlusions is multiperspective visualization (MPV), which relies on expressive images that integrate samples captured from multiple viewpoints [40]. Consider a scene with two rooms connected by an open door. A multiperspective visualization can let a user located in the first room explore the second room without actually moving to the second room. The MPV pipeline starts by defining the secondary viewpoints needed to gain line of sight to all ROIs, then a general camera model is defined by routing non-linear rays according to the secondary viewpoints, and then the scene is rendered with the general camera to obtain the multiperspective image. In the context of VR HMD scene exploration, MPV has the challenge of limited display resolution, which limits the number of potential ROIs that can be disoccluded simultaneously. Furthermore, the VR user should not be encumbered with having to define the disocclusion effect through the user interface. Finally, the disocclusion effect should be deployed and retracted through intuitive interface manipulations.

In this paper we propose a method to improve VR HMD scene exploration efficiency by providing the user with occlusion removal assistance. The disocclusion effect is constructed automatically, in real time. The scene is first rendered conventionally from the current viewpoint, the resulting depth buffer is analyzed, disocclusion portals are built from depth discontinuities, and the availability of the disocclusion portals is indicated to the user through a highlight of the portal frame. The user can select a disocclusion portal with their view direction, and engage the disocclusion effect with the controller. Once engaged, secondary viewpoints are deployed in front of the disocclusion portals in sync with small lateral user head translations. The secondary viewpoints are used to build a graph camera with piece-wise linear rays, and the scene is rendered with the graph camera to obtain a multiperspective visualization frame that also shows samples gathered from the secondary viewpoints, alleviating occlusions.

In Fig. 1, top, a lateral disocclusion effect is defined automatically based on the two nearby buildings and it reveals the person in the courtyard, which is not visible in a conventional visualization. The disocclusion effect is deployed by the user by translating their head 10cm to the left. In the bottom row, a vertical disocclusion effect is defined automatically based on the wall and the gate roof, and it reveals the courtyard beyond the wall. The disocclusion effect is deployed by the user with a 10cm upward translation of their viewpoint, i.e. by getting up on their tiptoes. In both cases, the geometry close to the user is rendered conventionally, and the disocclusion effect ”superpower” is confined to the distant part of the scene. To the best of our knowledge, our method is the first to construct the disocclusion effect automatically, in real time. Prior methods require for the disocclusion portals to be defined by hand, off-line, before the visualization sessions begins; furthermore, prior methods rely on special scene properties, such as the presence of physical portals defined by corridor intersections, door ways, or balcony ledges [26, 38, 39].

The advantage of defining the disocclusion effect in real time is particularly important in the case of dynamic scenes, where occlusions change over time and their removal cannot be pre-programmed. In Fig. 2, a horse-drawn carriage moves through the scene. The disocclusion portals (green contour lines) are based on the current position of the carriage to achieve a disocclusion effect that is appropriate for the current scene configuration.

We tested our VR HMD disocclusion method in the context of a user study where participants (N=16) performed a static target searching task and a dynamic target chasing task. Half of the participants relied on our MPV, and half relied on conventional visualization, serving as control. The quantitative metrics comparing the two conditions were the total viewpoint translation distance, the total viewpoint rotation angle, the number of redirections caused by reaching the physical boundary of tracked space, and the completion time. Our MPV had a significant advantage for all these metrics. We have also measured spatial awareness, using a separate pointing task, depth perception using a separate distance estimation task, and simulator sickness using a standard questionnaire [18]. Despite the spatial distortion introduced by the MPV, our method did not perform significantly worse than conventional visualization on these perceptual metrics. We also refer the reader to the accompanying video which further illustrates our MPV method and the tasks from the user study.

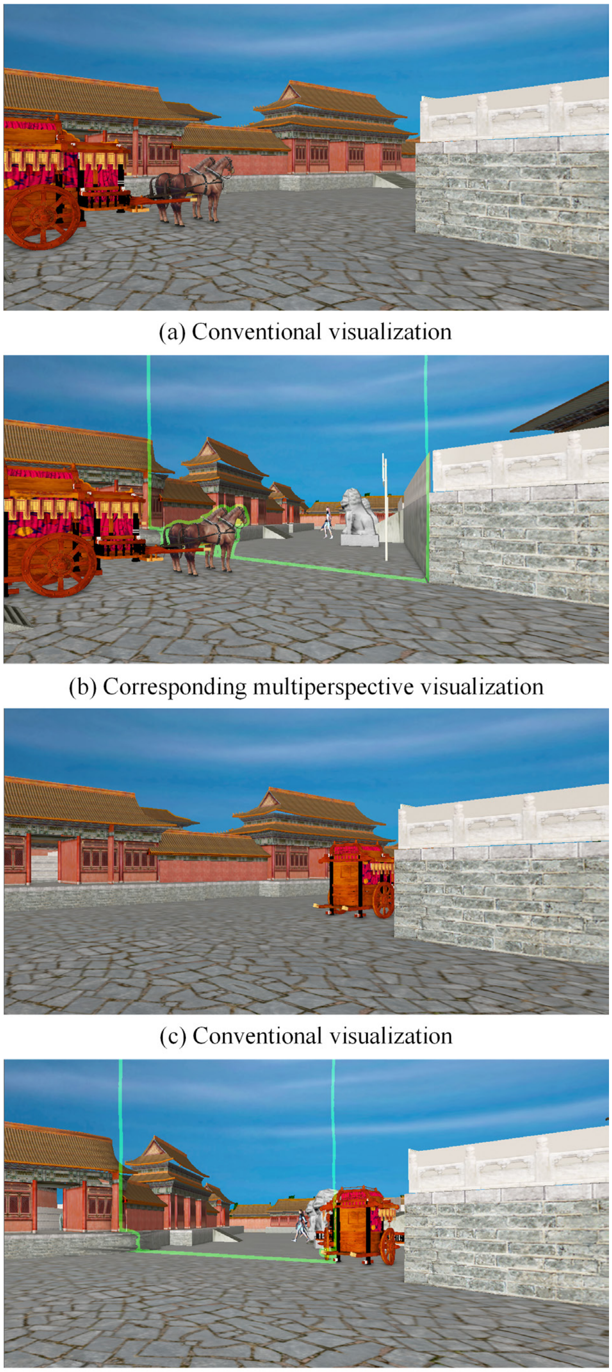
2 Prior Work

A considerable amount of previous research focused on VR scene navigation and exploration, targeting two main challenges. The first challenge is to allow the user to explore a large virtual environment, while the VR application is hosted in a limited physical space. The second challenge is to allow the user to explore the environment quickly, effectively, and comfortably, i.e. without disorientation or simulator-induced sickness. We discuss prior work in VR exploration first. Our VR navigation method is based on alleviating occlusions, and we discuss prior work in occlusion removal second.

2.1 Virtual Reality exploration methods

One approach for making the exploration of a large virtual environment tractable in the context of physical space limitations is redirected walking. The approach packs the user's virtual environment path in the confined physical space by resorting to path distortions. One goal is for the path distortions to not create geometry anomalies [33]. Another goal is to hide the path manipulations from the user, by making them as close to imperceptible as possible [8]. Subsequent work augments pre-programmed smooth path distortions with real-time abrupt changes synchronized with time periods when the user is less perceptive of such changes, for example during user head rotations, saccades, or blinks ([20], [32]). They relocated and redirect users when they rotated their head, blinked or made saccades.

The traditional approach for giving the user access to the entire virtual environment relies on teleportation, which temporarily suspends the one to one mapping between the virtual and physical spaces. For example, in point-based teleportation users can jump to locations point to with a handheld controller [3]. The challenge of teleportation is that it breaks the sense of immersion. When the teleportation is gradual, i.e. when the user is flown to the destination, the mismatch between visuals and acceleration induces simulator sickness. When the teleportation is abrupt, the visualization discontinuity is disorienting. Rapid continuous node-based locomotion was proposed to alleviate simulator sickness [16]. The user sees a network of floating ”footprint” icons that it can use to move from one node to another. The mobious transition is a



(a) Conventional visualization

(b) Corresponding multiperspective visualization

(c) Conventional visualization

(d) Corresponding multiperspective visualization

Fig. 2. Conventional (a and c) and corresponding multiperspective visualization (b and d, respectively) in a dynamic scene. The disocclusion effect is computed automatically based on the current position of the moving carriage.

teleportatio mechanism developed specifically for navigation of VR environment modeled with 360 degree videos [22].

Another approach is to let the user control the viewpoint translation through body movements that are, to a greater or a lesser degree, evocative of the actual locomotion. One classic line of work is a treadmill on which the user walks to translate in the virtual environment without translating in the physical environment [13]. More recently, the LazyNav system allows the user to tailor navigation through tracked body elements [15]. Another solution is to let the user indicate the direction of travel through head tilt [35].

Different ideas to improve the efficiency of VR exploration include the amplification of the user's head rotations to assume scene coverage with minimal view direction changes [11, 27]. A different line of work focuses on prescribing to the user locations that maximize scene sampling quality, based on predefined scene region importance scores and on real-time accounting of the exploration status of scene regions [12]. In previous work we explored the benefit of deviating from the natural field of view of VR and AR HMD's [37]. The resulting free field of view HMD visualization does not aid with occlusions, as the viewpoint remains in the same place; however, it bypasses unnecessary navigation to gain a comprehensive view when the field of view is too small, or unnecessary forward viewpoint translation just to achieve the magnification of the visualization of a region of interest, as both effects can be trivially achieved with a wider/narrower field of view.

2.2 Disocclusion methods

Occlusion is an important factor that limits 3D scene exploration efficiency. Occlusions have been addressed with a series of approaches in conventional, non-immersive visualization, and some of these approaches have been ported to the context of HMD visualization.

A top view map, rendered from above with a downwards view direction, can help users locate their current position in the virtual scene, and plan paths to reach distant parts of the scene. However, using the map in the context of HMD visualization is challenging due to the small resolution of the HMD and to the break of immersion when the user switches from the first to the third person visualization.

Another classic approach for occlusion removal is to render occluders transparently [1,7,10]. The challenge of such X-ray methods is poor scalability with the number of occluding layers. Cutaway techniques remove parts of occluders, to reveal the parts of the scene behind [4], which comes at the cost of losing contextual information provided by the occluder, which disorients the user. For example, if the user sees a distant object of interest by removing all walls of a maze, the user will have hard time finding a path to the object once the maze walls are reinstated. Spatial context is better preserved in ”radial distortion and melting techniques [29], which introduce a barrel distortion to enlarge the field of view at the corners of the image, while melting occluding layers away at the center of the image.

Multiperspective visualization methods (MPV) render seamless composite images that show the scene from multiple viewpoints. MPV maintains context while showing the parts of the scene that are behind occluders. Earlier research focused on relaxing the single center of projection constraint, but the sampling rays remained linear [14]. More recent work introduced sampling rays that are piecewise linear or even curved to provide the flexibility needed to route rays around occluders, to reach distant ROIs.

Several multiperspective methods target specific use cases, for example to better visualize 3D landscape and urban environments [21]. The method composites a scene overview into the conventional first person view used for navigation. The hybrid perspective visualization is achieved through a scene deformation. In order to avoid changing multi-perspective camera configurations manually, a multiscale visualization method is proposed [25], which interpolates the different global deformation automatically with considered of the viewing angle of the virtual camera. Other prior multiperspective visualization techniques opt to design the disocclusion effect in camera space, with the advantage of better control of the disocclusion effect. The family of occlusion cameras perturb the rays of a conventional planar pinhole cameras to capture not only samples visible from the user viewpoint, but also samples that are barely hidden, i.e. samples that become visible with minor viewpoint translations [23]. Occlusion cameras have the merit of considering visibility on a continuum, which allows generalizing the viewpoint to a view region. The family of graph cameras allows disocclusion deep into the scene, by integrating samples from a sparse set of viewpoints [26]. The graph camera is literally an acyclical graph of conventional cameras, constructed recursively through frustum bending, splitting, and merging operations. In this work, we adopt the graph camera model for its flexibility.

AR researchers have pursued multiperpsective visualization to augment the user's view of the real world with samples captured with cameras from additional viewpoints. The multi-view AR technique integrates a video feed captured from a higher viewpoint [36], and the variable perspective technique lets the user select the additional viewpoints from a set of available viewpoints that are visualized [36].

In our prior work, we have begun exploring the advantages of multiperspective visualization in VR and AR HMD scene exploration [38]. The disocclusion effects are constructed by manually pre-defining portals in the VR or AR environment. The user activates a disocclusion effect with the handheld controller. Multiperspective visualization proves to have significant advantages over conventional visualization in terms of scene exploration efficiency, as measured in searching, matching, and chasing tasks. In subsequent work [39], we have replaced the controller activated deployment of the disocclusion effect with an interactive deployment based on amplified user head translation. While multiperspective visualization reduces the need for teleportation, teleportation is still needed. Our subsequent work [39] also contributes a teleportation mechanism based on multiperspective visualization, which keeps the user anchored at all times to avoid simulator sickness.

The current work builds on our previous efforts, by adopting the graph camera implementation of the disocclusion effect and by adopting the translation amplification for an intuitive deployment of the disocclusion effect. The current work contributes the automatic construction of the disocclusion effect, which replaces the manual, off-line design of the disocclusion effect. Furthermore, the current work constructs the disocclusion effect in real-time, which extends our technique to dynamic scenes, where occlusion-disocclusion relationships appear, change, and disappear over time.

3 Automatic Occlusion Removal

Multiperspective visualization methods provide a good way to remove occlusions in complex scenes without losing the context of the current view. However, in prior multiperspective rendering methods, the secondary viewpoints needed to achieve the disocclusion effect are set manually [6, 26, 38]. Our method constructs the disocclusion effect automatically, in real time. Given a 3D scene modeled with triangles and the user's current view, the goal is to build a multiperspective visualization that alleviates the occlusions in the user's current view. Our method proceeds according to the pipeline shown in Fig. 3. The overall approach is to automatically route piece-wise linear rays around occluders to gain line of sight to parts of the scene not visible from the current view.

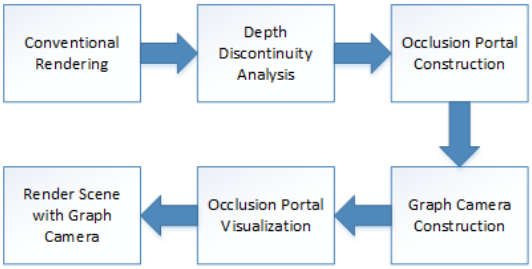

Conventional Rendering
Depth Discontinuity Analysis
Occlusion Portal Construction
Graph Camera Construction
Occlusion Portal Visualization
Render Scene with Graph Camera

Fig. 3. Overview of our automatic occlusion removal pipeline.

First, the scene is rendered conventionally from the user's view to obtain a depth buffer. Then the depth buffer is analyzed to extract horizontal and vertical depth discontinuities, as described in Sect. 3.1. The depth discontinuities are used to build disocclusion portals, as described in Sect. 3.2. The disocclusion portals are used to build a graph camera, which also takes into account the current position of the user's head to deploy or to retract the secondary perspectives accordingly (Sect. 3.3). Finally, the scene is rendered with the graph camera by projection followed by rasterization [26].

3.1 Depth discontinuity analysis

The first step is to find silhouette edges of occluders, which will guide the construction of the rays. For this, we render the scene from the user's viewpoint and detect depth discontinuities in the resulting depth buffer D. Depth discontinuity detection proceeds along the horizontal and the vertical center lines of D according to Algorithm 1.

Algorithm 1 Depth discontinuity analysis

```
Input: depth buffer $D$, center line length $n$, parameters $\varepsilon_1, \varepsilon_2, \varepsilon_3$
Output: center line depth discontinuity sets $SNF$, $SFN$, and $BIP$
 1: for $i$ in $n$ do
 2:     $d_{i-1} = D_i - D_{i-1}$, $d_i = D_{i+1} - D_i$, $d_{i+1} = D_{i+2} - D_{i+1}$
 3:     if $d_i - d_{i-1} < \varepsilon_1$ then
 4:         continue
 5:     end if
 6:     if $i < n/2$ and $|d_i/d_{i-1}| > \varepsilon_2$ then
 7:         $SNF = SNF \cup \{i\}$
 8:         continue
 9:     end if
10:     if $i > n/2$ and $|d_i/d_{i+1}| > \varepsilon_2$ then
11:         $SFN = SFN \cup \{i\}$
12:         continue
13:     end if
14:     if $|d_{i+1} - d_{i-1}|/max(|d_{i+1}|, |d_{i-1}|) > \varepsilon_3$ then
15:         $BIP = BIP \cup \{i\}$
16:         continue
17:     end if
18: end for
19: return $SNF$, $SFN$, $BIP$
```

D stores a measure of depth proportional to $1/z$, as shown in Equation 1, where z_n is the depth of the near plane, z_f is the depth of the far plane, and z_i is the z value at the current pixel.

$$D_i = (1/z_n - 1/z_i)/(1/z_n - 1/z_f) \tag{1}$$

For a pixel to be a depth discontinuity, we first threshold the second order depth difference (line 2). Since depth is proportional to $1/z_i$, the second order difference is exactly zero for any plane, no matter its orientation relative to the user view.

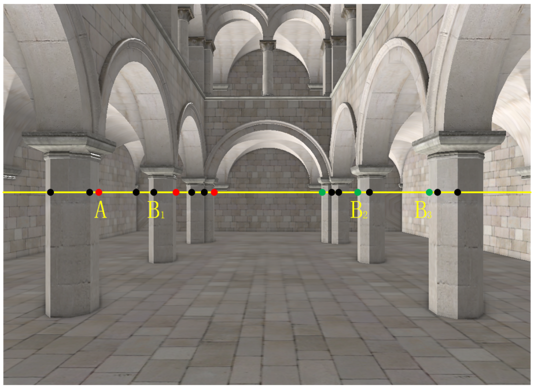


Fig. 4. Inflection (black), near to far (red), and far to near (green) depth discontinuity points on the horizontal center line.

Then the algorithm distinguishes between depth discontinuities that are silhouette points (red and green in Fig. 4, lines 6-13) and inflection points (black in Fig. 4, lines 14-17). Red silhouette points are near to far depth discontinuities, in left to right order. To keep the disocclusion effect simple, we only detect red silhouette points in the left half of the image (line 6), where disocclusion will be enacted with a small head translation to the right. In Fig. 4, the user will disocclude the columns in the left half of the image by looking around them through the right side. Similarly, green silhouette points are far to near depth discontinuities, in left to right order. Again, we detect green silhouette points only in the right half of the image for disocclusion effect simplicity.

3.2 Disocclusion portal construction

Disocclusion portals are planar contours that serve two purposes: first, they help route the piece-wise linear graph camera rays to reach occluded parts of the scene (Sect. 3.3), and second, they indicate to the user the availability of a disocclusion effect (Sect. 3.4). The disocclusion portals are computed as shown in Algorithm 2. The input to the algorithm are the depth discontinuity points computed as described above, together with the user view.

Algorithm 2 Construct disocclusion portals

```
Input: user viewpoint $V$, user view matrix $M$, depth buffer $D$, set of
    near to far silhouette points $SNF$, set of far to near silhouette points
    $SFN$, set of inflection points $BIP$
Output: Set of disocclusion portals $PRT$
for $i$ in $SNF$ do
    for $j$ in $SFN \cup BIP$ with $j > i$ do
        maxpq = 0
        $P_i$ = Unproject($D_i$, $M$)
        $P_j$ = Unproject($D_j$, $M$)
        $C = (P_i + P_j)/2$
        $N = (P_j - P_i) \times (0,1,0)$
        $\beta = angle(VC, N)$
        $pq = 45° - |\beta - 45°|$
        if $pq > maxpq$ then
            $portal_i = (P_i, P_j, N)$
            $maxpq = pq$
        end if
    end for
    $PRT = PRT \cup \{portal_i\}$
end for
return $PRT$
```

The algorithm constructs one portal for each of the silhouette points. Here we only give the algorithm for constructing portals for the near to far (red) silhouette points. The algorithm for the far to near (green) silhouette points is similar.

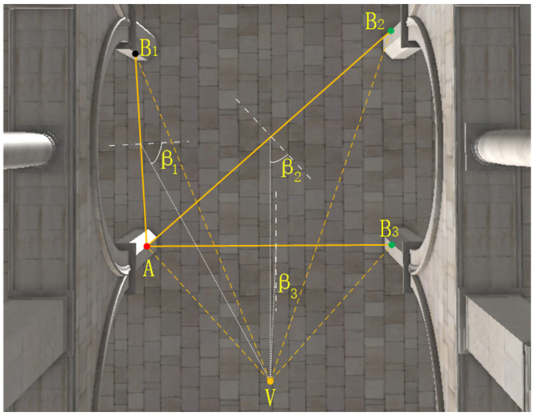


Fig. 5. Selection of portal for point A from Fig. 4. Three candidates are considered: AB_1, AB_2, and AB_3.

For each near to far silhouette point p_i, we consider a set of candidate portals. A candidate portal is defined by either a far to near silhouette

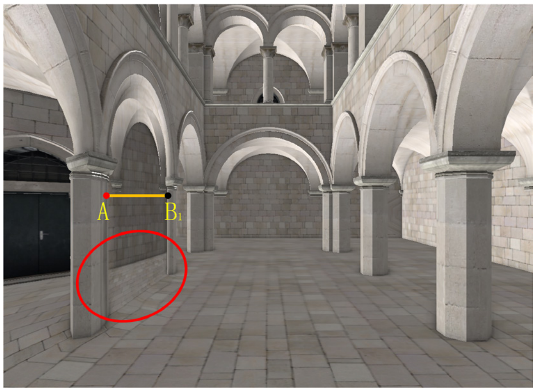


Fig. 6. Disocclusion effect based on candidate portal AB_1 from Figs. 4 and 5. The portal footprint is small so the disoccluded area is visualized at low resolution (red ellipse).

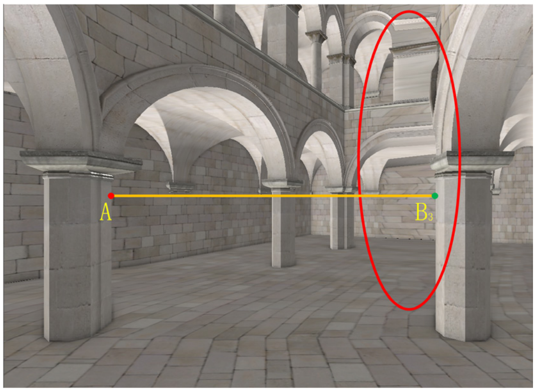


Fig. 7. Disocclusion effect based on candidate portal AB_3 from Figs. 4 and 5. The disoccluded area is distorted significantly (red ellipse).

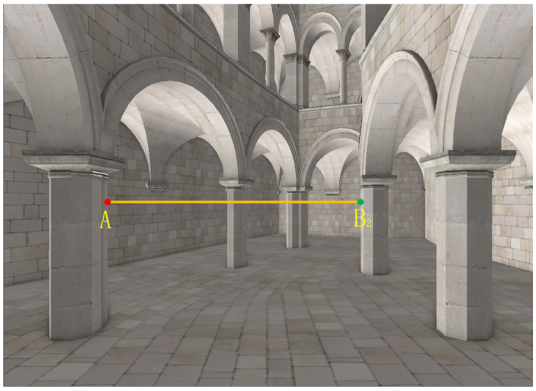


Fig. 8. Disocclusion effect based on candidate portal AB_2 from Figs. 4 and 5. The visualization looks like a conventional visualization of a plausible 3D scene, which indicates that the portal achieves the right balance between resolution and scene distortion.

point, or by an inflection point. Only points p_j to the right of p_i are considered (line 2). A portal quality score pq is computed for each candidate portal, and the candidate with the highest quality score $maxpq$ is selected. In order to compute the portal quality for the current candidate, the portal is first unprojected to 3D, using the user view matrix M and the portal pixels, including their depth (lines 4, 5). Then the portal normal N is computed using the up vector (line 7). The quality of the candidate portal depends on the angle $beta$ between the portal normal and the user ray through the 3D center C of the portal (line 9). The more similar beta is to 45°, the higher the portal quality.

Fig. 5 illustrates three candidate portals for the near to far silhouette point A of Fig. 4: AB_1, AB_2, AB_3. The candidate portal with the highest quality score is AB_2. The parts of the scene discoccluded by a portal will be shown on the projection of the portal. For AB_1, the footprint of the projection of the portal is too small to show the disoccluded part of the scene at adequate resolution (Fig. 6). For AB_3, the footprint of the portal is large, but achieving a meaningful disocclusion effect requires translating the secondary viewpoint substantially, which introduces substantial distortions (Fig. 7). AB_2, which is neither sampled tangentially, nor perpendicularly, offers a good compromise of these two antagonistic requirements (Fig. 8).

3.3 Graph camera construction

The disocclusion effect is implemented by rendering the scene with a graph camera, which is constructed based on the disocclusion portals. Fig. 9 shows the construction of a graph camera based on portal AB. The user view frustum (green) bends at the portal plane AB. Geometry beyond AB (green) is rendered from the secondary viewpoint V', which disoccludes the scene region covered by angle θ. The amount of disocclusion is controlled by the placement of V'. In Fig. 10 the amount of disocclusion increases as the secondary viewpoint moves from V to V_1 and then to V_2. The secondary viewpoint is placed on a circle centered at the center of the portal. The location on the circle is defined based on the amount of lateral translation of the user's head. Here, a translation to the right of 20cm moves the secondary viewpoint from V to V_2, amplifying the translation to 7.62m. The motion of the secondary viewpoint on the circle is capped to V_2 to avoid sampling the portal at extreme angles. Of course, the disocclusion effect implies not sampling parts of the scene that were sampled from the original viewpoint V, i.e. encompassed by ϕ_1 for V_1, and by ϕ_2 for V_2, but these parts were already visualized by the user before the disocclusion effect is deployed.

Once the graph camera is constructed, the scene is rendered by projecting vertices with the graph camera. Vertices inside the secondary frustum (blue in Fig. 9) are projected with a combined projection and transformation matrix for V and V'. Vertices in the primary frustum (green in Fig. 9) are projected conventionally with the user view matrix. Triangle edges that are completely inside the secondary frustum will map to 2D graph camera image edges. The graph camera essentially takes a picture from V of an image of the scene rendered from V'. Consequently, conventional rasterization produces correct results. However, triangle edges that cross from one subfrustum to the other, i.e. by intersecting the portal plane AB, will have a piece-wise linear projection. We use conventional rasterization and we control the approximation error by scene subdivision.

3.4 Disocclusion portal visualization

We provide two modes for disocclusion based on the graph camera constructed based on the disocclusion portals. In one mode, the user simply engages the disocclusion effect using a controller button, after which small lateral and vertical head translations are amplified and applied to all secondary viewpoints to reveal the hidden parts of the scene. For the example shown in Fig. 4, after the disocclusion effect is engaged, small head translations to the left disoccludes based on the portals on the right side of the image, constructed from the green silhouette points, and vice versa.

In a second mode, the disocclusion portals are visualized for the user, who selects one or several disocclusion portals, which are then used to provide the disocclusion in sync with head translations. This

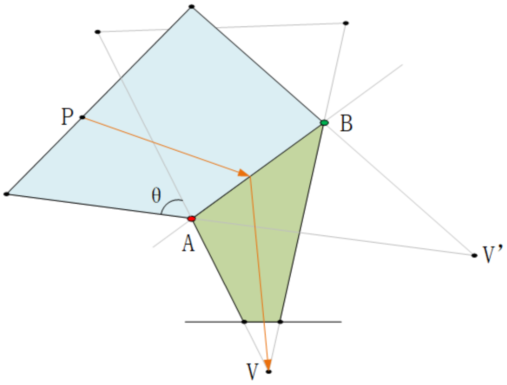


Fig. 9. Graph camera construction by bending the frustum at the portal plane AB. The disocclusion effect brought by the secondary viewpoint V' is encompassed by angle θ. The resulting graph camera has two subfrusta, the green sub-frustum, rendered from the primary user perspective, and the blue sub-frustum, rendered from the combined secondary and primary perspectives.

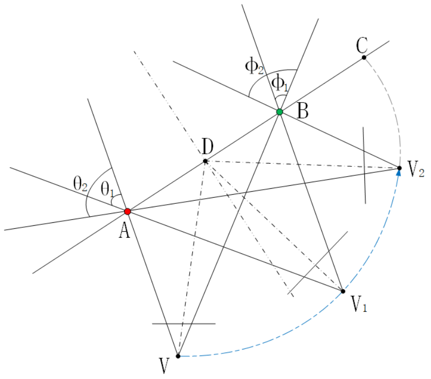


Fig. 10. Progressive deployment of disocclusion effect by revolving the secondary viewpoint around the center D of the portal AB, from V to V_1 and to V_2. The disocclusion effect magnitude increases from 0 to θ_1 and to θ_2.

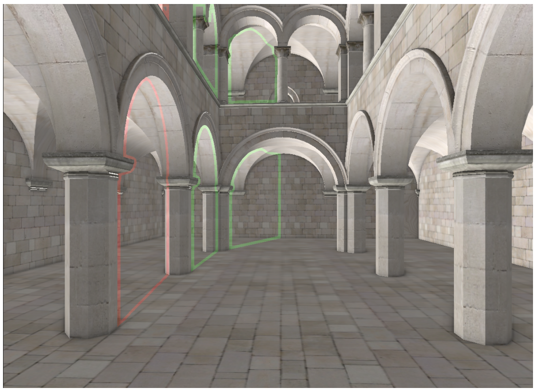


Fig. 11. Portal visualization with contour lines. Here the user selects one of the three portals manually (red).

interactive mode of engaging the available disocclusion effects requires a good visualization of the disocclusion portals. Algorithm 2 describes a portal with a horizontal segment and a normal, which defines a vertical rectangle with no top or bottom boundaries. The visualization highlights the contour of the portal based on the actual scene geometry (Fig. 11). The contour is drawn with a pass over the image. The current pixel is highlighted as a contour pixel if its depth is deeper than that of the portal rectangle, and if not all of its immediate neighbors have a depth that is deeper than that of the portal rectangle. The half size of the region defining the immediate neighbors gives the thickness of the highlight. We use a 17×17 region.

The user can select multiple portals, with the restriction that they be on the same side of the image. This restriction ensures that the disocclusion effect of the multiple portals selected can be merged into one bigger disocclusion effect. The disocclusion effects are merged by replacing the multiple secondary viewpoints of the individual portals (i.e. V' in Fig. 9) with their center of mass, and the resulting single secondary viewpoint controls the merged disocclusion effect. Fig. 12 shows the merged disocclusion effect of three portals.

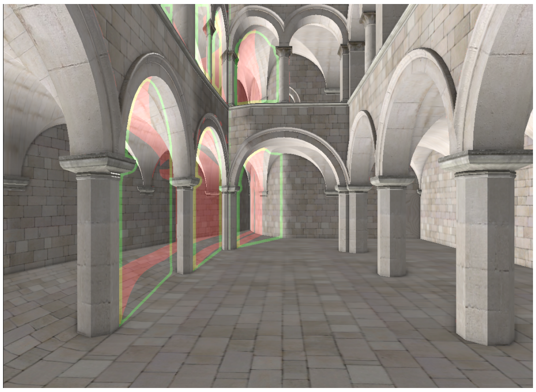


Fig. 12. The disocclusion effect achieved by combining several portals.

4 User Study

In order to evaluate the effect of our VR exploration assistance method, we conducted controlled user studies in which participants (Sect. 4.1) were asked to find a stationary target in a Desert Village VR environment and to find a dynamic target in a Chinese Garden VR environment (Sect. 4.2). Furthermore, participants took part spatial awareness perceptual experiments, in addition to the main tasks (Sect. 4.3). The performance of the experimental group using our multiperspective visualization method was compared to that of a control group that used conventional visualization. In addition, participants completed a simulator sickness questionnaire (SSQ) [[18]] before and after the two VR exploration tasks. Results are presented and discussed in Sect. 4.4.

We used an HTC Vive system which has a tracked HMD, an external tracker, and a wireless hand-held controller. The HMD is tethered to a desktop PC (Intel i7 processor, 16GB RAM, and NVIDIA 1080-ti graphics card). The virtual environments were rendered at 30fps for each eye.The tracked physical space hosting the VR applications is 4m × 4m. When a participant reaches the boundary of the tracked space, the participant can trigger a 180 degree rotation of the virtual environment [[28]]. This walking redirection mechanism allows folding a much larger virtual environment in the relatively small tracked area. The walking redirection mechanism is abrupt, and the redirection is clearly perceived by the participant. However, this mechanism has the advantage that all user viewpoint translation corresponds to actual locomotion, without teleportation or flying.

4.1 Participants

Our participants were 16 graduate students, 12 male and 4 female, between 20 and 30 years old. Fourteen of our participants had used immersive VR applications before. Furthermore, participants had normal or corrected vision, and none reported vision or balance disorders. The participants were randomly assigned to two groups of eight. The participants in the control group (CG) used conventional visualization. The participants in the experimental group (EG) had the option of using the disocclusion effect provided by our method. In our studies we relied exclusively on the disocclusion mode that, once engaged, provided disocclusion based on the portals in the current view. Furthermore, we limited the number of portals to at most one disocclusion portal to each side of the image, i.e. left-right, or top-down.

4.2 VR exploration tasks

All participants performed two VR exploration tasks, with at least one day and at most three days of rest in between tasks. The CG and EG assignment was the same for both tasks. For the first task (VR1), participants were asked to find three target vehicles, one at the time, in our Desert village virtual environment (Fig. 13, Fig. 14). The environment covers a 75m × 75m area. Before the experiments start, participants in both groups were given three minutes to familiarize themselves with the environment and with the redirected walking mechanism. EG participants also familiarized themselves with the disocclusion effect provided by our method. The participant can engage the effect at any time, and the subsequent small head translations deploy and retract the disocclusion effect.

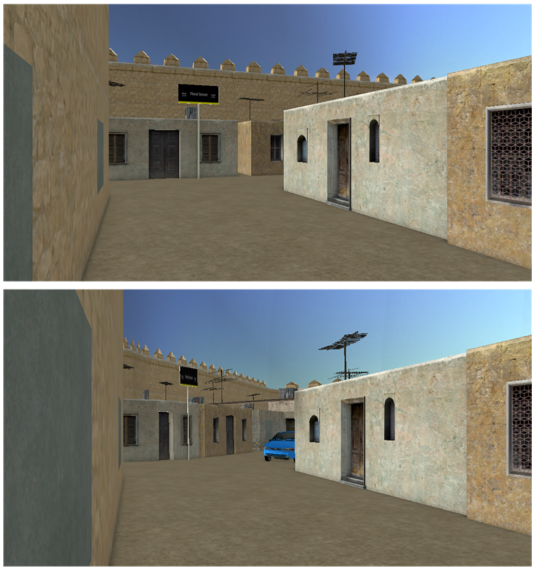


Fig. 13. Conventional (top) and multiperspective (bottom) visualization in our Desert Village scene. The lateral disocclusion effect reveals the blue sedan parked around the corner.

The participant was informed by the experimenter which vehicle to search for, e.g. ”a blue sedan”. A target vehicle is found when the participant’s position is within 1.5m of the vehicle. This means that it was not sufficient for a participant to see the vehicle, but they had to actually walk close to it.

For the second task (VR2), participants were asked to catch up with a target walking person in our Chinese Garden virtual environment (Fig. 1). The Chinese Garden covers a 130m × 140m area. The task is considered complete when the participant gets within one meter of the target person.

Task performance was measured with the following metrics:

- total viewpoint translation, in meters, computed as the sum of all frame to frame head translations.
- total view direction rotation, in multiples of full 360° rotations, computed as the sum of all frame to frame view direction rotation; as our environments are fairly flat, and as the participant does not roll their view, this metric essentially aggregates the frame to frame pan angles.

- number of boundary walking redirections, in times.
- task completion time, in seconds.

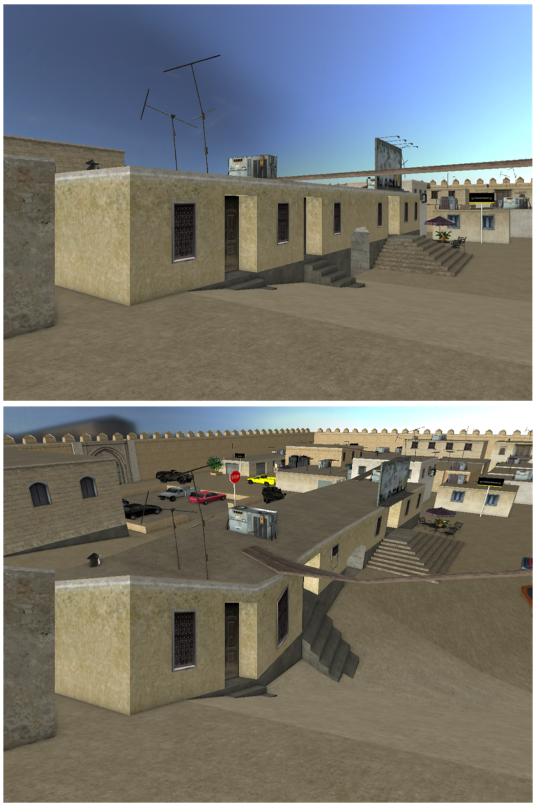


Fig. 14. Conventional (top) and multiperpsective (bottom) visualization. The vertical disocclusion effect reveals the yellow truck in the courtyard, which is occluded by buildings in the conventional visualization.

4.3 Perceptual experiments

In order to evaluate the impact of our method on spatial awareness, we tested our participants in additional depth estimation and pointing direction tasks, performed in the Sponza virtual environment (Fig. 15).

The depth estimation task asked a participant to decide which of a green and a red ball are farther from the participant's current position (Fig. 15). Both balls are visible in the conventional visualization, which is provided to CG participants (Fig. 15, left). For the EG participants, neither ball is visible in the conventional visualization, and they become visible once the disocclusion effect is deployed. In both cases, the distance between the participant and the balls is approximately 12m, and the ball that is farther is farther by approximately 2m. This ensures that task difficulty is similar for the two groups. Note that it is not possible to use the exact same ball positions for the two groups, since seeing the balls before deploying the disocclusion effect would contaminate EG participants. Furthermore, simply showing to EG participants the multiperspective image once the disocclusion effect is deployed (Fig. 15, right), would deprive these participants from important visual clues provided by the gradual transition from conventional to multiperspective visualization. The depth estimation task was repeated five times, with different positions for the two balls. We count the number of depth estimation errors, which is between 0 and 5.

Pointing tasks, in which the participant is asked to indicate the direction to a previously seen object that is no longer visible have been used to gauge participants' spatial awareness, in environments such as building interiors [34] and large outdoor spaces [24]. A pointing task has also been used to evaluate a participant's ability to maintain spatial orientation while navigating through virtual corridors [2]. Based on this prior work, we rely on a pointing task to further gauge the spatial understanding of our participants.

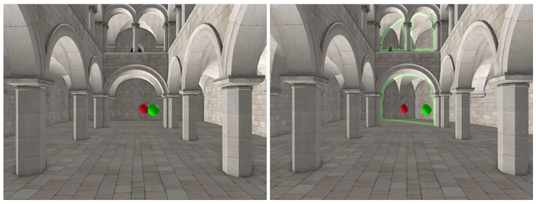


Fig. 15. Conventional (left) and multiperpsective (right) visualizations used in the depth estimation experiment with CG and EG participants, respectively.

Table 1. Total viewpoint translation, in meters

Task	Group	Avg± std.dev.	Red.	p	Cohen's d	Effect size
VR1	CG EG	291 ± 58.5 156 ± 23	46%	0.0051	3.0	Huge
VR2	CG EG	485 ± 107 126 ± 74.1	74%	0.0015	3.9	Huge

A green ball was placed in the Sponza environment such that it is not visible from a reference location V_0 (blue ball in Fig. 16). CG participants were instructed to walk to a secondary location V_1 (red ball in Fig. 16), from where the green ball is visible, and then to return to V_0. EG participants stayed at V_0 and disoccluded the green ball using our technique. Once the CG participant returned to V_0 and the EG participant retracted the disocclusion effect, the participant was asked to point in the direction of the green ball with a virtual laser ray emitted by the controller. This pointing task was repeated five times, with different positions of the green ball. The metric quantifying pointing task performance inaccuracy is the sum of the angles between the correct direction to the green ball, and the direction provided by the participant.

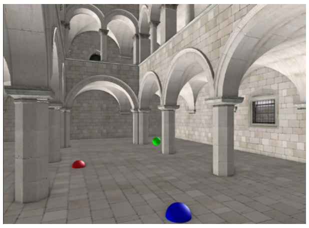


Fig. 16. Scene used for the pointing task. Participants had to point to the memorized position of a green ball, which is not visible from the reference participant location (blue). CG participants see the green ball by walking to a secondary location (red), and EG participants see it from the reference location using a disocclusion effect.

4.4 Results and discussion

We analyze the values of our metrics using the p value of a t-test [31,41], and we quantify effect size using Cohen's d [5,30]. Table 1 gives the total viewpoint translation results for the two VR exploration tasks. The average per-participant total translation is reduced by 46% and 74% by our disocclusion method. The difference is statistically significant (i.e. $p < 0.05$), and the effect size is ”Huge” [5,30]. As expected, the ability to see behind occluders without actual viewpoint translation reduces the total viewpoint translation substantially. The larger Chinese Garden virtual environment translated to an even larger advantage of our method over conventional visualization. In addition to the larger virtual environment, chasing the dynamic target in VR2 is more problematic for conventional visualization, which has to rely on luck for the user to be in the right place at the right time.

The total amount of view direction rotation is given in Table 2. As in the case of viewpoint translation, the disocclusion effect reduces the

Table 2. Total view direction rotation, as a multiple of full 360° rotations

Task	Group	Avg± std.dev.	Red.	p	Cohen's d	Effect size
VR1	CG EG	61 ± 7.8 30.7 ± 2.6	50%	0.0003	5.2	Huge
VR2	CG EG	62 ± 7.3 19 ± 8.8	70%	< 0.0001	5.4	Huge

Table 3. Tracked boundary redirections, in times

Task	Group	Avg± std.dev.	Red.	p	Cohen's d	Effect size
VR1	CG EG	99 ± 19 55 ± 8	44%	0.011	3.0	Huge
VR2	CG EG	62 ± 19 14 ± 9	77%	0.004	3.2	Huge

total rotation significantly, and more so for the VR2 task. With the back and forth sequential exploration of the scene afforded by conventional visualization, a participant has to pan the view by 180 degrees every time they retrace their steps.

Table 3 gives the number of walking redirections caused by reaching the boundary of the tracked physical space. The EG participants had significantly fewer redirections, a consequence of the reduced viewpoint translation. As these abrupt redirections contribute to user disorientation, avoiding them is an important benefit of our multiperspective visualization method.

Table 4 gives the task completion times. EG participants finish the tasks significantly faster than CG participants. Combining Table 1 and Table 4, our method reduces for VR2 the average translation speed from 0.88m/s to 0.58m/s, which, like the reduced view direction rotation and the reduced number of redirections, have the potential to reduce participant disorientation. Table 1, Table 2, and Table 4 also show that overall participant effort is reduced significantly by our method, which leads to improved VR exploration ergonomics.

Table 5 gives the errors for the perceptual tasks, for both participant groups. For distance estimation, no CG participant made any error. EG participants estimated depth incorrectly on average 1.25 times out of the five trials, but the difference to CG is not significant. We attribute this difference to the scene deformation induced by our method, which affects the visualization of the distance between scene points and the participant viewpoint. Indeed, in Fig. 15, right, the red and green balls appear side by side in the multiperspective visualization, and not one after the other. For pointing, the error is slightly larger for CG, but the difference is again not significant. We attribute this to the disorientation incurred by having to return from the secondary to the reference position.

Table 6 gives the result of simulator sickness test. Each participant completed the SSQ before and after each of the two VR exploration tasks. We analyze the SSQ data using the Total Severity score (TS) [18]. The TS increase from pre- to post- exposure are below 70, which indicates that simulator sickness is not a factor for either group of participants [9]. This confirms the design choice of restricting the disocclusion effect to the distant part of the scene. The near part of the scene is rendered with the primary participant perspective, which anchors the user and avoids simulator sickness.

Table 4. Task completion time, in seconds

Task	Group	Avg± std. dev.	Red.	p	Cohen's d	Effect size
VR1	CG EG	824 ± 81 454 ± 84	45%	0.0007	4.5	Huge
VR2	CG EG	550 ± 88 216 ± 118	61%	0.0039	3.2	Huge

Table 5. Distance estimation error (times) and pointing angle error (degrees)

Task	Group	Avg± std. dev.	p
Distance est.	CG EG	0.00 ± 0.00 0.25 ± 0.43	0.29
Pointing	CG EG	24.5 ± 7.58 19.0 ± 7.66	0.35

Table 6. Total Severity score (TS) for pre and post SSQ

Task	Group	pre± std. dev.	post± std. dev.
VR1	CG EG	1.87 ± 3.24 12.16 ± 11.64	7.48 ± 0.00 26.18 ± 16.09
VR2	CG EG	1.87 ± 3.24 8.42 ± 12.51	7.48 ± 0.00 15.90 ± 10.7

5 Conclusions, Limitations, and Future Work

We have presented an approach for alleviating occlusions in the HMD exploration of virtual environments. The central idea of our approach is to allow the user to preview, from the current viewpoint, scene regions to which there is no line of sight. The disocclusion effect is constructed automatically based on depth discontinuities in a conventional rendering of the scene, as these mark places where occlusions occur. The disocclusion effect is deployed and retracted intuitively by overloading the six degree-of-freedom tracking of the HMD–small user head translations are amplified to swing secondary viewpoints to locations from where they can remove occlusions. As the user sees more of the scene compared to using a conventional visualization, our approach translates to more efficient scene exploration. Our approach combines the advantages of sequential, interactive exploration with those of parallel visualization from multiple viewpoints. The benefits of our approach do not come at the cost of a significant decrease in spatial awareness, or of simulator sickness.

Our approach relies on depth discontinuities found on the horizontal and the vertical center lines. In the case of HMD visualization this is a reasonable choice as the user will aim their view direction towards the part of the scene that they want to focus on. Of course, complex scenes suffer from occlusions in many parts of the image, and future work could look into devising disocclusion effects for discontinuities away from the two center lines. The graph camera that underlies our MPV visualization supports any number of recursive frustum bending, splitting, and merging operations. As always, the tradeoff to consider is that between visualization complexity and disocclusion payload.

Another limitation of the current work is that we target large scale occlusions, as those that arise in indoor and outdoor architectural environments. The present work does not address small occlusions, as, for example, those created by the leafs of a tree, or a complex car engine. We foresee that future work aimed at addressing this problem will have to abandon the disocclusion portal graph camera construction, and simply focus on detecting the scene sub-volumes where these fine grain occlusions occur, which should then be visualized with a secondary viewpoint that revolves around such sub-volumes.

Our work increases navigation efficiency without trying to hide it from the user, which also means that the user's sense of immersion in the virtual world is reduced. Future work should quantify the effect of multiperpsective visualization on the sense of immersion. Either way, our method provides superpowers to the user, which the user quickly learns how to invoke in order to improve exploration efficiency. The visualization is altered away from the user, while nearby geometry is rendered according to the unmodified pose parameters read from the HMD tracker, which minimizes disorientation and avoids simulator sickness.

The current work provides a greedy solution to the problem of occlusions, by defining a few disocclusion portals that let the user see around occluders. Another possible direction of future work is to quantify the amount of occlusion in a conventional visualization frame, and to define disocclusion effects that guarantee the reduction of occlusions below an application chosen threshold.

Acknowledgments

We would like to thank Meng-Lin Wu for his inspiring work on occlusion removal for HMD visualization for VR and AR. We are also thankful to Antong Cao, Zhichao Li, and Hui Zhang for their help with conducting the user study. This work was supported in part by the National Natural Science Foundation of China through Project 61272349 and the Beijing Natural Science Foundation L182016.

References

[1] B. Avery, C. Sandor, and B. H. Thomas. Improving spatial perception for augmented reality x-ray vision. In *2009 IEEE Virtual Reality Conference*, pp. 79–82. IEEE, 2009.

[2] D. A. Bowman, E. T. Davis, L. F. Hodges, and A. N. Badre. Maintaining spatial orientation during travel in an immersive virtual environment. *Presence*, 8(6):618–631, 1999.

[3] E. Bozgeyikli, A. Raij, S. Katkoori, and R. Dubey. Point & teleport locomotion technique for virtual reality. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play*, pp. 205–216. ACM, 2016.

[4] C. Coffin and T. Hollerer. Interactive perspective cut-away views for general 3d scenes. In *3D User Interfaces, 2006. 3DUI 2006. IEEE Symposium on*, pp. 25–28. IEEE, 2006.

[5] J. Cohen. Statistical power analysis for the behavioral sciences. 1988, hillsdale, nj: L. *Lawrence Earlbaum Associates*, 2, 1988.

[6] J. Cui, P. Rosen, V. Popescu, and C. Hoffmann. A curved ray camera for handling occlusions through continuous multiperspective visualization. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1235–1242, 2010.

[7] A. Dey, G. Jarvis, C. Sandor, A. Wibowo, and V. V. Mattila. An evaluation of augmented reality x-ray vision for outdoor navigation. In *International Conference on Artificial Reality and Telexistence*, pp. 28–32, 2011.

[8] Z.-C. Dong, X.-M. Fu, C. Zhang, K. Wu, and L. Liu. Smooth assembled mappings for large-scale real walking. *ACM Transactions on Graphics (TOG)*, 36(6):211, 2017.

[9] J. A. Ehrlich and E. M. Kolasinski. A comparison of sickness symptoms between dropout and finishing participants in virtual environment studies. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 42, pp. 1466–1470. SAGE Publications Sage CA: Los Angeles, CA, 1998.

[10] O. Erat, W. A. Isop, D. Kalkofen, and D. Schmalstieg. Drone-augmented human vision: Exocentric control for drones exploring hidden areas. *IEEE transactions on visualization and computer graphics*, 24(4):1437–1446, 2018.

[11] S. Freitag, B. Weyers, and T. W. Kuhlen. Examining rotation gain in cave-like virtual environments. *IEEE transactions on visualization and computer graphics*, 22(4):1462–1471, 2016.

[12] S. Freitag, B. Weyers, and T. W. Kuhlen. Interactive exploration assistance for immersive virtual environments based on object visibility and viewpoint quality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 355–362. IEEE, 2018.

[13] J. Fung, C. L. Richards, F. Malouin, B. J. McFadyen, and A. Lamontagne. A treadmill and motion coupled virtual reality system for gait training post-stroke. *CyberPsychology & behavior*, 9(2):157–162, 2006.

[14] R. Gupta and R. I. Hartley. Linear pushbroom cameras. *IEEE Transactions on pattern analysis and machine intelligence*, 19(9):963–975, 1997.

[15] E. Guy, P. Punpongsanon, D. Iwai, K. Sato, and T. Boubekeur. Lazynav: 3d ground navigation with non-critical body parts. In *3D User Interfaces (3DUI), 2015 IEEE Symposium on*, pp. 43–50. IEEE, 2015.

[16] M. P. J. Habgood, D. Moore, D. Wilson, and S. Alapont. Rapid, continuous movement between nodes as an accessible virtual reality locomotion technique. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 371–378, 2018.

[17] D. Kalkofen, M. Tatzgern, and D. Schmalstieg. Explosion diagrams in augmented reality. In *Virtual Reality Conference, 2009. VR 2009. IEEE*, pp. 71–78. IEEE, 2009.

[18] R. S. Kennedy, N. E. Lane, K. S. Berbaum, and M. G. Lilienthal. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The international journal of aviation psychology*, 3(3):203–220, 1993.

[19] B. J. Kuipers and T. S. Levitt. Navigation and mapping in large scale space. *AI magazine*, 9(2):25, 1988.

[20] E. Langbehn, F. Steinicke, M. Lappe, G. F. Welch, and G. Bruder. In the blink of an eye: leveraging blink-induced suppression for imperceptible position and orientation redirection in virtual reality. *ACM Transactions on Graphics (TOG)*, 37(4):66, 2018.

[21] H. Lorenz, M. Trapp, J. Döllner, and M. Jobst. Interactive multiperspective views of virtual 3d landscape and city models. In *Proc. 11th International Conference on GI Science*, pp. 301–321, 2008.

[22] A. MacQuarrie and A. Steed. The effect of transition type in multi-view 360 media. *IEEE transactions on visualization and computer graphics*, 24(4):1564–1573, 2018.

[23] C. Mei, V. Popescu, and E. Sacks. The occlusion camera. In *Computer Graphics Forum*, vol. 24, pp. 335–342. Wiley Online Library, 2005.

[24] A. Okabe, K. Aoki, and W. Hamamoto. Distance and direction judgment in a large-scale natural environment: Effects of a slope and winding trail. *Environment and Behavior*, 18(6):755–772, 1986.

[25] S. Pasewaldt, M. Trapp, and J. Döllner. Multiscale visualization of 3d geovirtual environments using view-dependent multi-perspective views. *Journal of WSCG*, 19:111–118, 2011.

[26] V. Popescu, P. Rosen, and N. Adamo-Villani. The graph camera. In *ACM Transactions on Graphics (TOG)*, vol. 28, p. 158. ACM, 2009.

[27] E. D. Ragan, S. Scerbo, F. Bacim, and D. A. Bowman. Amplified head rotation in virtual reality and the effects on 3d search, training transfer, and spatial orientation. *IEEE transactions on visualization and computer graphics*, 23(8):1880–1895, 2017.

[28] S. Razzaque, Z. Kohn, and M. C. Whitton. Redirected walking. In *Proceedings of EUROGRAPHICS*, vol. 9, pp. 105–106. Citeseer, 2001.

[29] 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 *Proc. 8th IEEE International Symposium on Mixed and Augmented Reality*, pp. 211–220, 2009.

[30] S. S. Sawilowsky. New effect size rules of thumb. *Journal of Modern Applied Statistical Methods*, 8(2):26, 2009.

[31] Student. The probable error of a mean. *Biometrika*, pp. 1–25, 1908.

[32] Q. Sun, A. Patney, L.-Y. Wei, O. Shapira, J. Lu, P. Asente, S. Zhu, M. McGuire, D. Luebke, and A. Kaufman. Towards virtual reality infinite walking: dynamic saccadic redirection. *ACM Transactions on Graphics (TOG)*, 37(4):67, 2018.

[33] Q. Sun, L.-Y. Wei, and A. Kaufman. Mapping virtual and physical reality. *ACM Transactions on Graphics (TOG)*, 35(4):64, 2016.

[34] P. W. Thorndyke and B. Hayes-Roth. Differences in spatial knowledge acquired from maps and navigation. *Cognitive psychology*, 14(4):560–589, 1982.

[35] S. Tregillus, M. Al Zayer, and E. Folmer. Handsfree omnidirectional vr navigation using head tilt. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 4063–4068. ACM, 2017.

[36] E. Veas, R. Grasset, E. Kruijff, and D. Schmalstieg. Extended overview techniques for outdoor augmented reality. *IEEE transactions on visualization and computer graphics*, 18(4):565–572, 2012.

[37] L. Wang, A. Cao, Z. Li, X. Yang, and V. Popescu. Effective free field of view scene exploration in vr and ar. In *Adjunct Proceedings of the IEEE International Symposium for Mixed and Augmented Reality 2018 (To appear)*, 2018.

[38] M.-L. Wu and V. Popescu. Efficient vr and ar navigation through multiperspective occlusion management. *IEEE transactions on visualization and computer graphics*, 2017.

[39] M.-L. Wu and V. Popescu. Multiperspective visualization for efficient vr navigation. In *Proc. 15th Annual EuroVR Conference.*, pp. 1–20. EuroVR Association, London, 2018.

[40] J. Yu, L. McMillan, and P. Sturm. Multiperspective modeling, rendering, and imaging. In *ACM SIGGRAPH ASIA 2008 courses*, p. 14. ACM, 2008.

[41] D. W. Zimmerman. Teacher's corner: A note on interpretation of the paired-samples t test. *Journal of Educational and Behavioral Statistics*, 22(3):349–360, 1997.