This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2778249, IEEE Transactions on Visualization and Computer Graphics

# Efficient VR and AR Navigation through Multiperspective Occlusion Management

Meng-Lin Wu and Voicu Popescu

**Abstract**—Immersive navigation in virtual reality (VR) and augmented reality (AR) leverages physical locomotion through pose tracking of the head-mounted display. While this navigation modality is intuitive, regions of interest in the scene may suffer from occlusion and require significant viewpoint translation. Moreover, limited physical space and user mobility need to be taken into consideration. Some regions of interest may require viewpoints that are physically unreachable without less intuitive methods such as walking in-place or redirected walking. We propose a novel approach for increasing navigation efficiency in VR and AR using multiperspective visualization. Our approach samples occluded regions of interest from additional perspectives, which are integrated seamlessly into the user's perspective. This approach improves navigation efficiency by bringing simultaneously into view multiple regions of interest, allowing the user to explore more while moving less. We have conducted a user study that shows that our method brings significant performance improvement in VR and AR environments, on tasks that include tracking, matching, searching, and ambushing objects of interest.

**Index Terms**—Augmented reality, Virtual reality, Navigation, Occlusion management, Multiperspective visualization, Depth cues

## 1 Introduction

TRACKED Head-Mounted Displays (HMDs) provide an intuitive interface for exploring virtual and real 3D scenes in virtual reality (VR) and augmented reality (AR) applications. The user naturally selects the desired view by walking and by rotating their head. However, the efficiency of such 3D scene exploration is limited by occlusions. Consider the case of a VR exploration of a city model with the goal of finding a specific street-level region of interest (ROI). Tall buildings occlude the streets and the user has to move considerable amounts to gain a direct line sight to the street currently being examined. If the street proves to be empty, the user proceeds with examining the next street. This sequential exploration is inefficient. In the case of a dynamic ROI, such a sequential exploration might never find the ROI.

Consider the application of surveillance of corridors inside a building. Occlusions prevent the user from seeing beyond the current corridor segment. The user has to walk to each intersection sequentially in order to examine side corridors. Again, the sequential exploration might never find a moving intruder. Even worse, if the intruder is aware of the user's position, the intruder can easily avoid being detected. Another challenge of conventional navigation in VR and AR applications is that some of the viewpoints best suited for alleviating occlusions are unreachable. Consider the VR urban model mapped to a room. Walls and furniture might prevent the user from assuming the viewpoint that establishes a direct line of sight to a ROI. For example, a higher viewpoint is less affected by occlusion from the tall buildings, but a viewpoint higher than the standing height of the user is hard to achieve.

In this paper, we propose to increase VR and AR scene exploration efficiency by enhancing the visualization with additional perspectives. The HMD shows to the user a multiperspective image that captures the scene from multiple viewpoints, yet the image is non-redundant, with no parts of the scene being shown more than once, and continuous, with nearby 3D scene points projecting to nearby image locations. The ROIs occluded from the user's viewpoint are captured from secondary viewpoints and integrated into the original view. The additional perspectives are rendered with correct disparity so the "stereo" multiperspective HMD visualization provides appropriate depth cues to the user. We employ two types of multiperspective visualization, designed to alleviate occlusions in two scenarios.

The first type of multiperspective visualization is designed to overcome occlusions in an urban VR scene (Fig. 1, top). The second type is designed for AR visualization of environments defined by cells connected by portals, such as a building interior with rooms and corridors connected by doors and intersections (Fig. 1, bottom). The visualization supports the simultaneous disocclusion of multiple ROIs (Fig. 2), as well as the integration of RGBD streams acquired with depth cameras (Fig. 3).

In order to quantify any VR and AR navigation benefits brought by the multiperspective visualization, we have conducted a randomized controlled user study in which the subjects were asked to perform four visual and navigational tasks in the environments shown in Fig. 1. Our multiperspective VR and AR reduced viewpoint translation by 45.2% and view direction rotation by 43.2%. We also refer the reader to the accompanying video. To the best of our knowledge, our work is the first investigation of multiperspective occlusion visualization in VR and AR HMD applications.

*The authors are with the Department of Computer Science, Purdue University, West Lafayette, IN 47907.*
*E-mail: wu223, popescu@purdue.edu.*

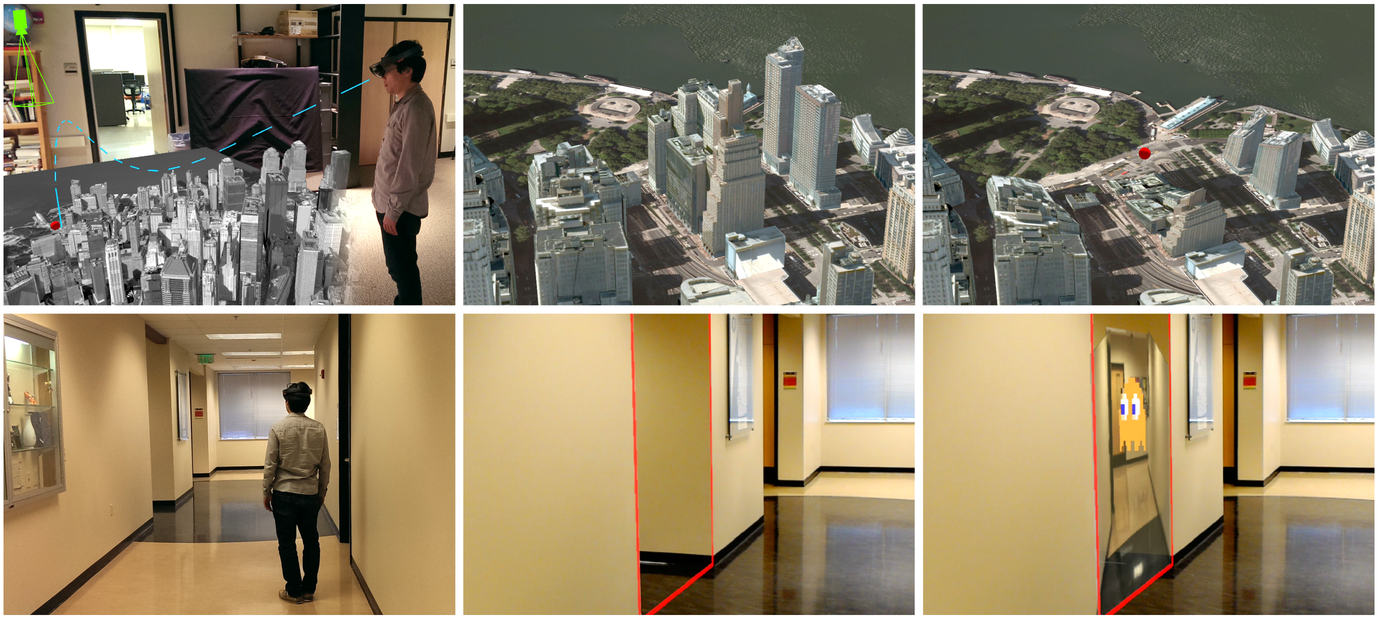
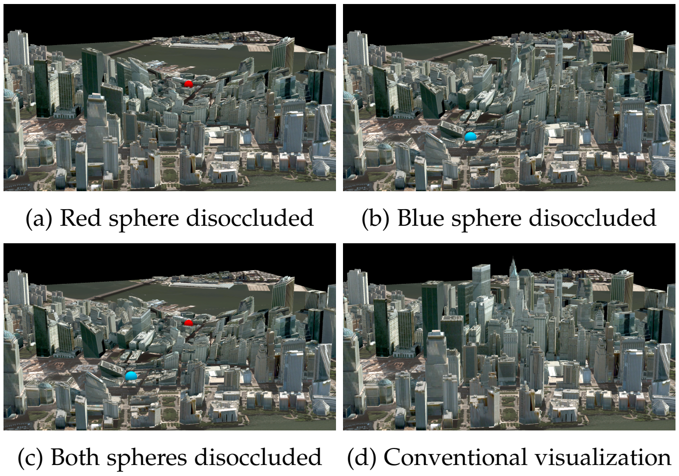


Fig. 1: (Top) Multiperspective disocclusion in a VR exploration of an urban scene: artist rendition of a second-person view (left), conventional VR visualization (middle), and our multiperspective VR visualization (right). The user does not have a direct line of sight to the red sphere, which is occluded in the a conventional image; an additional perspective (green frustum) is used to route rays over the occluding buildings (dashed blue line), which disoccludes the red sphere in the resulting multiperspective image. (Bottom) Multiperspective disocclusion in an AR exploration of a real world indoor scene: second-person image (left), conventional image from user viewpoint (middle), and our AR multiperspective visualization (right). The left side corridor is disoccluded by inserting an additional perspective at the portal (red rectangle), which reveals the target (orange ghost).



(a) Red sphere disoccluded (b) Blue sphere disoccluded

(c) Both spheres disoccluded (d) Conventional visualization

Fig. 2: Simultaneous disocclusion of multiple ROIs through our multiperspective visualization.

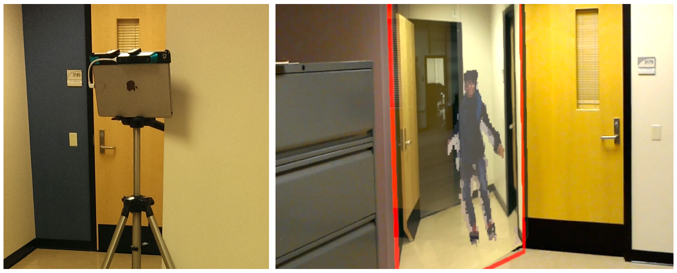


Fig. 3: Left: the RGB camera with depth sensor accessory captures a RGBD video stream that is converted to a 3D point cloud. Right: the point cloud, as seen from the secondary perspective, is integrated into the primary perspective in the multiperspective visualization.

## 2 Prior work

In VR applications, viewpoint navigation in the virtual scene is performed by physical locomotion in a host real world scene. The quality of the navigation experience is essential to the success of the VR application. One challenge stems from a mismatch between the size of the physical world accessible to the user and where the user can be tracked, and the size of the virtual world that the user wants to explore. We discuss prior work that addresses this challenge in Section 2.1. A second challenge stems from the reduction in visualization efficiency caused by occlusions. In conventional VR visualization, the user can only see parts of the scene to which there is a direct line of sight. The user has to explore the scene sequentially, walking to reach viewpoints from where region of potential interest is visible. We discuss existing approaches for occlusion management in Section 2.2, and we compare our proposed technique to prior multiperspective techniques in Section 2.3.

### 2.1 Physical to virtual world mapping

The virtual scene is frequently more expansive than the available physical space. This limitation is worked around using in-place walking systems based on treadmills [1] or inertial sensing hardware [2]. These methods interpret a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2778249, IEEE Transactions on Visualization and Computer Graphics

subset of the user's motion associated with walking, such as feet or head motion, to infer the desired locomotion. The drawback to in-place navigation systems is that the user does not change the viewpoint naturally by walking. Our system allows the user to walk freely, and, when desired, the user can save on the amount of walking by inspecting an occluded area with the help of an intuitively deployed secondary viewpoint. If the occluded area turns out to be a region of interest, the user can walk to it for further intuitive visualization.

When the physical space is less restrictive, locomotion by actual walking is found to be more intuitive for users [3], [4]. The problem of navigating a large virtual scene in a smaller physical space is well established. There exist methods such as redirection [5], [6], variable scene scaling [7], variable translation gain [8], and pose resetting [9]. Another approach is an optimized non-linear mapping from physical to virtual space [10]. Although effective at extending the virtual scene beyond the physically available space, these methods cause persistent discrepancies between perception and reality, of which the user can become aware, making the navigation less intuitive [11], [12]. Our system does not attempt to hide the added supernatural visualization capability, which is deployed at the user's demand, with the secondary viewpoint visualization anchored in the familiar single-perspective visualization from the user's viewpoint. Through the use of the secondary viewpoint, our system provides a limited extension of the virtual world beyond the physical locations that the user can reach.

In AR applications, the graphical annotations must remain registered to the physical scene, therefore any navigation method that does not preserve the identity mapping between virtual and physical spaces is ill-suited to AR. Because the VR methods mentioned improve navigation efficiency by diverging the virtual viewpoint from the user's physical location, they are not applicable to AR.

## 2.2 Occlusion management

Occlusion management is a promising approach to improving the efficiency of viewpoint navigation in both VR and AR, while retaining navigation intuitiveness. Occlusions reduce visualization effectiveness when the line of sight to the ROI is blocked by opaque objects. Conventional approaches to occlusion management fall into categories of X-ray, cutaway, and explosion methods. Multiperspective approaches for managing occlusions are discussed separately, in Section 2.3.

In X-ray visualization, the occluder is rendered semi-transparently so that occluded scene segments are visible through the occluder [13]. However, this visualization technique violates pictorial depth cues because the background is blended with the foreground. To alleviate this problem, the *ghosting* approach emphasizes foreground information at the edges [14], or other salient features [15], [16], by overlaying them as a ghost image. While adding a single processed foreground layer restores missing depth cues [17], scaling with occluder complexity, which requires visualization of multiple transparent layers, remains challenging. In addition, the compromise must still be made between conveying foreground information and limiting visual complexity, due to the overlapping image space position of the occluder and the ROI.

In cutaway visualization, the occluder is completely removed instead of being visualized transparently [18], [19]. The removal of the occluder leaves a hole for the line of sight to pass through and reach the ROI. Cutaway visualization maintains correct depth cues, but extra geometry must be generated for cut surfaces surrounding the hole, and little information about the removed occluders is conveyed.

Explosion methods segment the scene and translates the segments radially away from the ROI to reduce occlusion. The method is widely employed in computer aided design applications where the constituent parts are naturally separable by clear boundaries, in accordance to assembly sequences [20]. Volumetric datasets support explosion visualization when scene segments are annotated [21]. In AR applications, explosion visualization is possible when there exist synthetic 3D models corresponding to the physical objects [22]. In this case, the 3D models are required to visualize scene segments from alternative perspectives as they are translated due to the explosion. Compared to X-ray visualization, which increases visual complexity, and cutaway visualization, which omits foreground information, the explosion approach is able to present the foreground without interfering with the visualization of the ROI. However, explosion methods artificially fragment scene geometry which causes discontinuities across scene segments. Furthermore, the translation of scene segments hinder the user's spatial awareness, which is undesirable in VR and AR applications. Hybrid scene deformation methods seek to minimize disturbance to the scene geometry by combining multiple operations such as scene segment translation, scaling, and viewpoint shifting [23]. However, this approach relies on extensive scene preprocessing to identify scene segments, and optimization of cost functions tailored to these scene segments. The viewpoint shifting operation also interferes with the navigation in VR and AR.

Our method takes the multiperspective visualization approach, and we discuss prior work in this category in detail in the next subsection. X-ray, cutaway, and explosion visualization have the advantage of intuitiveness, as they can be seen as applying a familiar change to the real world: the occluding layer is built out of transparent material, a hole is cut into the occluding layer, or the scene is disassembled into individual parts. There is no familiar real world manipulation that achieves a similar effect to multiperspective visualization. However, multiperspective visualization is a powerful occlusion management technique with unique strengths.

## 2.3 Multiperspective visualization

Multiperspective visualization was first used in the visual arts, such as by Picasso in his cubist paintings. The single viewpoint constraint is relaxed in favor of more expressive images containing multiple integrated views. In imaging research, the study of camera models progressed beyond the traditional pinhole camera to novel cameras such as the push broom [24], the multiple center-of-projection [25], and the general linear camera models [26]. The goal is comprehensive acquisition of real world scenes with powerful

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2778249, IEEE Transactions on Visualization and Computer Graphics

imaging systems that overcome occlusions by capturing rays from multiple viewpoints.

In desktop visualization, multiperspective cameras were developed to increase the information bandwidth of images by alleviating occlusions. Occlusion cameras [27] are a class of multiperspective cameras that generalize the viewpoint to a view region by routing sampling rays around occluders. The curved ray camera routes rays around occlusions through multiple sequential viewpoints with $C^1$ continuity, though it does not support branching to multiple viewpoints in parallel [28]. The graph camera [29] generates a continuous and non-redundant multiperspective image that integrates multiple disparate viewpoints using frustum bending, splitting, and merging operations. The graph camera is literally a graph of conventional planar pinhole cameras. The node cameras were subsequently upgraded to general linear cameras to achieve multiperspective focus+context visualization: ROIs are shown from secondary viewpoints, and the perspective reverts to the main viewpoint outside the ROIs [30].

Several prior multiperspective visualization efforts target specifically urban and terrain scenes and opt for the approach of deforming distant, occluded geometry upward, while pushing near, occluding geometry downward [31], [32], [33]. One system improves spatial awareness through disocclusion in large-scale scenes, by providing the user with a choice of a large number of video feeds acquired from multiple vantage points [31]. The user selects the desired first person view from a third person view of the available feeds, and then resorts to a two-viewpoint multiperspective visualization to disocclude in the first person view. Compared with these techniques, our method has the advantage of not distorting the ground plane, which prevents user disorientation. Furthermore, our technique visualizes the disoccluded ROI exactly where it would be seen in the absence of occluders, which enables building a mental model of the target location. Benefiting from the stable ground geometry, our visualization supports an intuitive user interface where a ROI is directly selected by the user's gaze towards the ground plane, as opposed to manually inputting deformation parameters [31]. When multiple ROIs are identified, our visualization deploys multiple secondary views to disocclude individual ROIs with localized deformation of the geometry, whereas prior work deforms large terrain partitions without low-level granularity of the disocclusion effect [31], [32], [33].

Our visualization is able to augment the main physical view with secondary views in AR. Prior work displays secondary views by inserting virtual billboards into the scene [31], [34]. This has the advantage of allowing the user to teleport to distant viewpoints [31], whereas our method restricts the user to portals within sight. However, the billboards displaying the secondary viewpoints in the overview suffer from visualization discontinuity. Furthermore, the visualization based on monoscopic video lacks depth perception, which our method achieves by rendering the geometric model for each eye. Other prior work inserts deformed 3D geometry into the scene [35], but there is significant discontinuity between the deformed geometry and the physical world. In our AR visualization, secondary views are inserted as 3D geometry that seamlessly integrates with the un-deformed physical geometry.

We choose to build our multiperspective visualizations based on the graph camera [29] and on the multiperspective focus+context [30] frameworks. We extend these prior multiperspective frameworks to achieve an intuitive deployment of the additional viewpoints based on head-tracking, and to achieve stereoscopic multiperspective rendering as needed for the HMD. We measure the navigation efficiency increases in a controlled user study with two VR and two AR tasks.

## 3 Multiperspective VR visualization in urban scenes

In the urban VR scene, ROIs at street level are occluded by tall buildings. Such ROIs become disoccluded from an overhead perspective that has a viewpoint above the ROI and a downward view direction. When the location of the ROI is not known, as is the case for the tracking and matching tasks considered in this paper, we define the overhead perspective interactively under the assumption that the target is at the street-level point of intersection between the user's current view direction and the ground plane. Once the secondary, overhead perspective is defined, the scene is rendered with the resulting multiperspective camera model which integrates the secondary perspective seamlessly into the primary, user perspective. The resulting multiperspective image disoccludes the street-level look-at point without compromising the user's spatial awareness.

### 3.1 Interactive construction of secondary perspective

The user attempts to locate the ROI by scanning the urban model. The goal is to let the user see at street level, free of occlusions. The secondary perspective is constructed to provide an occlusion-free visualization of a 3D focus point defined as the intersection between the user's view direction and the ground plane. To aid the user in selecting the street-level point that should be disoccluded, a cursor is displayed at the focus point. The secondary viewpoint is placed vertically above the focus point at a height equal to the distance, $r$, from the primary viewpoint to the focus point (Fig. 4). The secondary perspective looks directly downwards and avoids occlusion from nearby tall buildings. The secondary perspective is rendered with the same field of view as the primary perspective, so, since the distance to the focus point is in the same in both perspectives, the disoccluded ROI will be shown in the multiperspective image at the same scale at which it would be seen in the primary perspective in the absence of occlusions. We use the same secondary viewpoint for both the left and the right user's eyes, which we compute by defining the primary viewpoint as the midpoint of the segment between the two eyes.

### 3.2 Secondary view integration

The integration of the secondary view into the user's primary view of the scene has to achieve three goals: (1) it has to give preference to the secondary view around the ROI, in order to achieve the desired disocclusion effect, (2) it has to give preference to the primary view away from the ROI, in order to help the user remain aware of their position

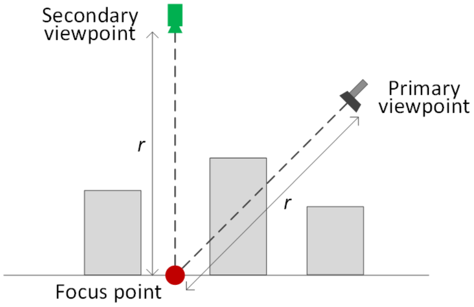


Fig. 4: Secondary viewpoint placement

and orientation in the scene, (3) and it has to display the street-level ROI where the user would see it in the absence of occluders, for the user to get an accurate sense of where the ROI is located in the scene. We achieve the third goal by ”anchoring” the ground plane, i.e. by enforcing that the ground plane is not distorted by the multiperspective visualization.

We render the multiperspective image by transforming the scene vertices with a three step operation. The results of the individual transformation steps are shown in Fig. 6. The intermediate steps are only shown here for illustration purposes; in our implementation, the vertices are projected directly from model space to the multiperspective image. In the first transformation step, a scene vertex $u$ is transformed to the local coordinates $v_1$ and $v_2$ of the primary and secondary perspectives by multiplication with their respective transformation matrices $\mathbf{V}_1$ and $\mathbf{V}_2$. All matrices and vectors are in standard homogeneous coordinates.

$$\begin{aligned} v_1 &= \mathbf{V}_1 \times u \\ v_2 &= \mathbf{V}_2 \times u \end{aligned} \tag{1}$$

The second step anchors the transformed vertex in the secondary perspective to the ground plane in the primary perspective by offsetting $v_2$ by a displacement vector $d$.

$$\begin{aligned} v_2' &= v_2 + d \\ &= v_2 + (\mathbf{V}_1 \times u_{ground} - \mathbf{V}_2 \times u_{ground}) \end{aligned} \tag{2}$$

The 3D point $u_{ground}$ is the ground plane projection of scene vertex $u$. For any vertex $u$ on the ground plane where $u = u_{ground}$, the transformed vertex $v_2'$ is equivalent to transforming $u$ with $\mathbf{V}_1$, ignoring $\mathbf{V}_2$, and Eq. 2 becomes

$$\begin{aligned} v_2' &= v_2 + d \\ &= \mathbf{V}_2 \times u_{ground} + (\mathbf{V}_1 \times u_{ground} - \mathbf{V}_2 \times u_{ground}) \\ &= \mathbf{V}_1 \times u_{ground} \\ &= \mathbf{V}_1 \times u = v_1 \end{aligned} \tag{3}$$

Thus the resulting multiperspective image shows the ground plane the same way the primary perspective does.

In the third and final step, the transformation of the vertex $u$ is finalized as a blend between the anchored transformation and the the primary perspective transformation.

$$v = \mathrm{G}(r,\sigma)v_2' + (1 - \mathrm{G}(r,\sigma))v_1 \tag{4}$$

The blend weights are given by a Gaussian G centered at the center of the secondary perspective image. The distance $r$ is computed in the secondary perspective image as the distance between the center of the image and the projection of the vertex. The root-mean-square width $\sigma$ is set according to the application (Fig. 5). Once the vertex is transformed with Eq. 4, the transformed vertex is then multiplied with a conventional projection matrix, that is the same for both the primary and secondary perspectives.

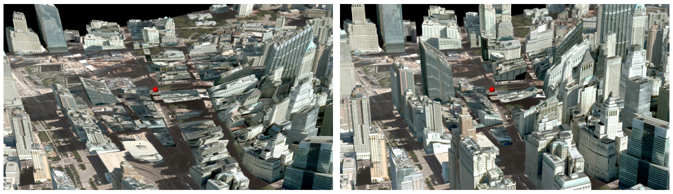


Fig. 5: The amount of screen space taken up by the secondary perspective image depends on the width $\sigma$ of the Gaussian G. Left: $\sigma = 0.8$; right: $\sigma = 0.4$

In the case of multiple ROIs, each ROI defines its own focus point and secondary perspective. Eq. 4 is modified such that the secondary perspectives are first integrated with one another (Eq. 5), and then the result is integrated with the primary perspective (Eq. 6).

$$\bar{v} = \frac{\sum_{i>1} \mathrm{G}(r_i,\sigma)v_i'}{\sum_{i>1} \mathrm{G}(r_i,\sigma)} \tag{5}$$

$$v = \mathrm{G}_{max}\bar{v} + (1 - \mathrm{G}_{max})v_1 \tag{6}$$

$v_i'$ is the vertex transformed according to secondary perspective $i$, and $r_i$ is the screen space distance to the center of the image of secondary perspective $i$. $\mathrm{G}_{max}$ is the largest of the weights $\mathrm{G}(r_i,\sigma)$. This gives preference to the secondary perspective transformation over the primary perspective transformation if a vertex is close to the center of any of the multiple ROIs. Consider a case with 3 ROIs. A vertex at the center of the first ROI and peripheral to the other two ROIs, i.e. which has transformation weights approximately equal to (1, 0, 0), should be transformed according to the secondary perspective of the first ROI. Using the average of the weights $\mathrm{G}(r_i,\sigma)$ in Eq. 6 would incorrectly give the first ROI transformation only a 33% weight.

The multiple secondary perspectives do not have to be disjoint. In the case when two dynamic ROIs become close or coincident, their respective secondary viewpoints also become close or coincident while sharing the identical downward view direction. Therefore, as two ROIs approach each other, their associated secondary perspectives smoothly converge, and the multiperspective visualization remains continuous.

### 3.3 Stereoscopic rendering

The HMD conveys depth cues through a stereoscopic image rendered from the left and the right eye viewpoints, separated by the interpupillary distance. To support stereoscopic rendering in our multiperspective visualization, each vertex $v$ in the coordinates of the monocular primary perspective is transformed to the coordinates of the left and the right perspectives, $v_L$ and $v_R$.

$$\begin{aligned} v_L &= \mathbf{V}_L\mathbf{V}_1^{-1} \times v \\ v_R &= \mathbf{V}_R\mathbf{V}_1^{-1} \times v, \end{aligned} \tag{7}$$

where $\mathbf{V}_L$ and $\mathbf{V}_R$ are transformation matrices of the left and the right perspectives. The resulting multiperspective visualization has the correct disparity over the primary and secondary perspectives, conveying appropriate depth cues to the user.

## 4 Multiperspective indoor AR visualization

Efficiency of exploration of a real-world indoor scene is limited by the tight turns at corridor intersections. In order to inspect the side corridors, the user has to walk to each intersection. Our AR multiperspective visualization leverages additional views of the corridors acquired by cameras placed at the intersections to present a comprehensive visualization to the user, who can see down a side corridor without having to walk up to the intersection. The secondary perspective down the corridor is seamlessly integrated with the user's primary perspective. In this section, we first describe the scene acquisition from additional perspectives. Then we describe the selection and integration of secondary views tailored to the user's current viewpoint.

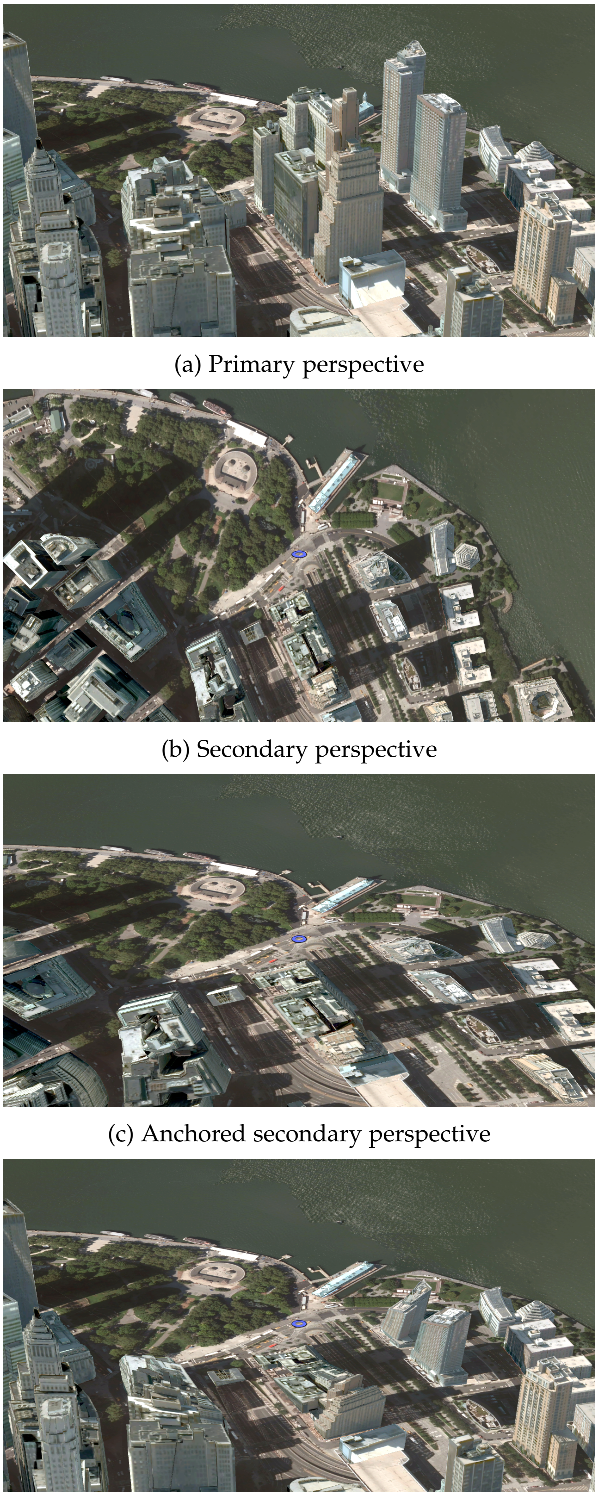
### 4.1 Acquisition

Unlike in the case of VR synthetic scenes, the AR context requires capturing the additional perspectives with physical cameras. Furthermore, the multiperspective visualization requires rendering the additional perspectives from novel viewpoints, so a 2D image is not sufficient, and a geometric model is needed. In the indoor context, the building geometry (i.e. walls, ceiling, floors) is fixed and we model it with a simple geometric model projectively texture mapped with a photograph. For the controlled experiments described in this paper, the content of the corridor is synthetic (i.e. we use a conventional computer graphics model of the ghost seen in Fig. 1, bottom. Dynamic real-world geometry is acquired with a color+depth (RGBD) camera, whose frames are reprojected to the desired viewpoint as a cloud of 3D points with color (Fig. 3).

### 4.2 Selection and integration of the secondary view

When the user sees an intersection, the rectangular portals leading to side corridors are highlighted with a red wireframe. If the user centers and holds the view on a portal, a secondary perspective swings into place, revealing what is visible through the portal (Fig. 1, bottom). The deviation from the conventional view of the scene is confined to the area of the portal. The portal frame acts like a hinge connecting the primary and secondary perspectives with $C_0$ continuity.

When a portal is activated, the scene geometry beyond that portal is visualized by rotating the scene vertices into view. Vertex transformation proceeds similarly to that of VR visualization, where the first step is to transform each scene vertex to coordinates of the primary and secondary perspectives, and the second step is to anchor the geometry to the portal plane for the primary perspective (Fig. 7). However, there is no need for blending between secondary and primary perspectives because they integrate at the boundary of the portal.



(a) Primary perspective

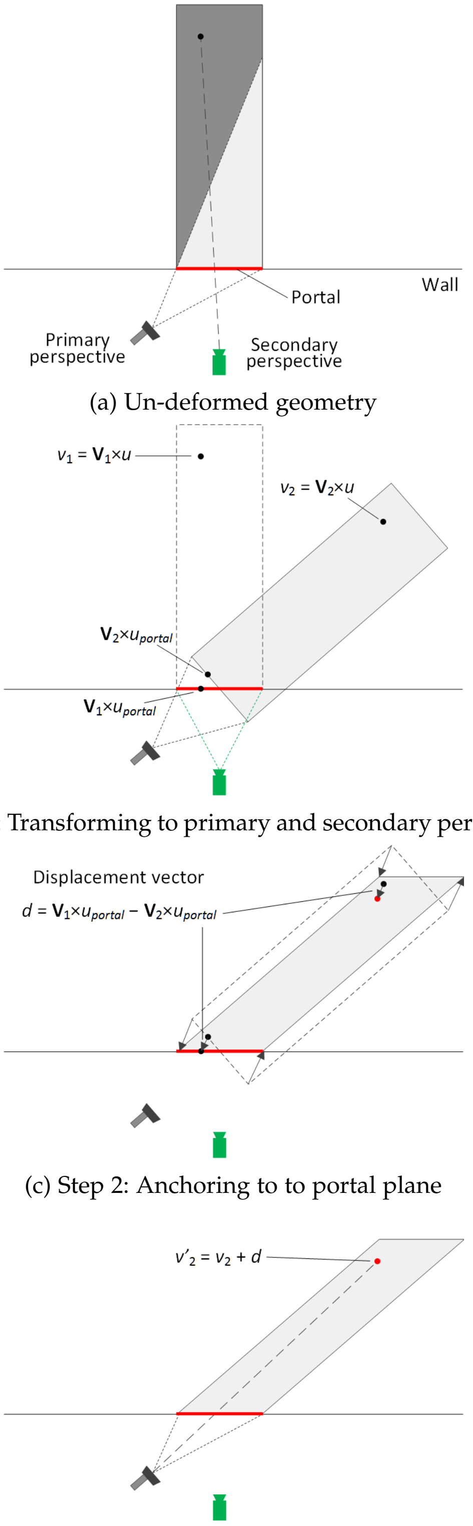
(b) Secondary perspective

(c) Anchored secondary perspective

(d) Integrated perspectives

Fig. 6: Illustration of the multiperspective transformation. Both the primary (a) and secondary (b) perspectives are centered on the ground plane focus point, which is occluded in the primary perspective and disoccluded in the secondary perspective. The secondary perspective is first modified (c) for the ground plane to appear the same way it would appear in the primary perspective in the absence of occlusions. The final multiperspective image (d) transitions smoothly from the primary perspective, at the periphery, to the secondary perspective, at the center.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2778249, IEEE Transactions on Visualization and Computer Graphics


Wall
Portal
Primary perspective
Secondary perspective
$v_1 = \mathbf{V}_1 \times u$
$v_2 = \mathbf{V}_2 \times u$
$\mathbf{V}_2 \times u_{portal}$
$\mathbf{V}_1 \times u_{portal}$
Displacement vector
$d = \mathbf{V}_1 \times u_{portal} - \mathbf{V}_2 \times u_{portal}$
$v'_2 = v_2 + d$

(a) Un-deformed geometry

(b) Step 1: Transforming to primary and secondary perspectives

(c) Step 2: Anchoring to to portal plane

(d) Deformed geometry

Fig. 7: (a): the un-deformed geometry, where the target is occluded along with the shadowed parts of the corridor, although they are visible from the secondary viewpoint. (b): the geometry in coordinates of both primary and secondary perspectives. (c): the geometry anchored to the portal plane. (d): the deformed geometry, where the target and all of the corridor become visible from the primary viewpoint. The depth beyond the portal is preserved for each vertex.

We wish to restrict virtual scene distortion to the 2D horizontal plane, so the primary and secondary perspectives are first restricted to be horizontal and at the same height. This restriction is relaxed at the final step of the rendering pipeline, where the full unrestricted primary perspective is used for rendering the final image.

In the first step, the primary perspective is constructed with the viewpoint at the position of the HMD, and the view direction is towards the center of the portal. Each scene vertex $u$ beyond the portal plane is transformed to both the primary and secondary perspectives using their respective transformation matrices, $\mathrm{V}_1$ and $\mathrm{V}_2$.

$$\begin{aligned} v_1 &= \mathbf{V}_1 \times u \\ v_2 &= \mathbf{V}_2 \times u \end{aligned} \tag{8}$$

The second step anchors the vertex $v_2$ in the coordinates of the secondary perspective to the portal plane by offsetting $v_2$ by a displacement vector $d$.

$$\begin{aligned} v'_2 &= v_2 + d \\ &= v_2 + (\mathbf{V}_1 \times u_{portal} - \mathbf{V}_2 \times u_{portal}), \end{aligned} \tag{9}$$

where $u_{portal}$ is the projection of vertex $u$ to the plane of the portal.

For any vertex $u$ on the portal plane where $u = u_{portal}$, the transformed vertex $v'_2$ is identical to $v_1$, and Eq. 9 becomes

$$\begin{aligned} v'_2 &= v_2 + d \\ &= \mathbf{V}_2 \times u_{portal} + (\mathbf{V}_1 \times u_{portal} - \mathbf{V}_2 \times u_{portal}) \\ &= \mathbf{V}_1 \times u_{portal} \\ &= \mathbf{V}_1 \times u = v_1 \end{aligned} \tag{10}$$

Thus the resulting multiperspective image shows the portal plane without any deformation, and the secondary perspective is anchored to the primary perspective on the portal plane.

A deformed side corridor is shown in Fig. 7. The target, represented by a red circle, is occluded from the primary perspective, but is visible from the secondary perspective. The target becomes visible when the scene geometry is deformed such that the corridor beyond the portal is aligned with the primary perspective. Fig. 1 (right) shows the rendered AR result.

Multiple secondary perspectives can be active simultaneously. Geometry visible through each portal is deformed separately according to each secondary perspective. No blending between secondary perspectives is necessary because the portals are disjoint, and the visualization of the deformed geometry is confined to the area of the portals.

With modified scene geometry, stereoscopic rendering proceeds straightforwardly by projecting scene vertices separately to the left and the right eye coordinates using their respective transformation matrices $\mathrm{V}_L$ and $\mathrm{V}_R$, as described in Section 3.3.

## 5 User Study

We have conducted a controlled randomized user study to evaluate the effectiveness of our VR and AR multiperspective visualization methods. In this first study we evaluate

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2778249, IEEE Transactions on Visualization and Computer Graphics

our method against conventional planar pinhole camera visualization, which is used in the overwhelming majority of VR and AR applications. We implemented our visualization methods for the urban VR and indoor AR scenes using an HMD with SLAM user tracking (i.e. the Microsoft HoloLens). The urban VR scene is a photo-textured model of Manhattan that is mapped to an empty floor space of $4m \times 5m$ (Fig. 1, top). The indoor AR scene covers a $10m \times 15m$ section of the floor plan of our office building (Fig. 1, bottom).

### 5.1 Subjects

We recruited a total of 16 subjects for our study, 12 male and 4 female. The subjects were undergraduate and graduate students between the age of 19 and 42. The subjects were randomly assigned to equal-sized control and experimental groups of 8 subjects each. The subjects assigned to the control group used a conventional visualization that showed only the primary perspective. The other subjects, assigned to the experimental group, used the multiperspective visualization. Each subject performed all four tasks directly, without training and without familiarization with the multiperspective visualization and with the user interface. The control group subjects wore the HMD even for the AR tasks, when the portals were highlighted with the red wireframe, but without the additional perspectives. In addition to age, the demographic information collected from the subjects also covered prior AR and VR experience. From the 16 subjects, 8 had prior experience with VR applications, and 6 had prior experience with AR applications.

### 5.2 Tasks and data collection

Each subject performed four tasks. Two of the tasks are performed in the urban VR scene, and the other two are performed in the indoor AR scene. The tasks require subjects to gain and maintain sight of static and dynamic synthetic objects placed in the scenes. The control and experiment configurations of the tasks are identical except for the visualization method employed. The user's head position and orientation are tracked and saved for data analysis.

*Target tracking task (VR1)* In the first urban VR task the subject is asked to keep a target in view as the target moves on the streets of the city. The target moves for 38s, after which the task is complete.

*Pair matching task (VR2)* In the second urban VR task the subject is asked to find pairs of spheres with the same color pattern. There are six pairs of spheres placed at street level and scattered throughout the scene. The subject selects a sphere by centering the view on it and by clicking a hand-held wireless mouse. In the experimental group, the selected sphere is marked as a ROI and is kept disoccluded even when the subject changes focus. The subject clicks are recorded for analysis of the rate of matching error. When two matching spheres are selected, the spheres are removed from the scene, and the task is completed when no sphere remains.

*Search task (AR1)* In the first AR task, the subject is asked to explore the corridors of the scene in search of stationary targets, implemented as computer graphics ”ghosts” (Fig. 1.) Only one target is available at one time. A target is found and removed when the user moves within 2m of it. Once a target is found, the next target is placed in the scene, which forces the user to revisit the scene in search of the new target. The task is completed after four targets are found. The target locations are the same for the control and experiment conditions.

*Ambush task (AR2)* The second AR task is similar to the first one, except that now the targets move. When the subject is in the direct line of sight of the target, the target moves away from the subject. This requires the subject to ”ambush” the target, by waiting around a corner for a target to move into the intersection where it is ”captured”. The targets move with the speed of 1.2m/s, so the subject can also run after the evading target to catch up with it. When a target is captured, the next target is spawned in a part of the scene far away from the current position of the subject. The subject is informed of the target's evasive motion strategy, but is not advised of a counter strategy.

### 5.3 Results and discussion

We analyze scene navigation performance using several metrics extracted from the subject pose traces recorded during our experiments. We detect and measure benefits of the multiperspective visualization compared to the conventional visualization in two ways: using Cohen's effect size $d$ [36], and using a two-sample t-test. Cohen's $d$ is calculated from measurements performed on the control and experimental groups, and its value is the difference between the mean values divided by the pooled standard deviation. Effect sizes are conventionally qualified as ”small”, ”medium”, and ”large” for the cases where $d > 0.2$, $d > 0.5$, and $d > 0.8$ respectively. Due to the variation of effect sizes observed in our metrics, the qualifiers for effect sizes are expanded to include ”very small”, ”very large”, and ”huge” for $d < 0.01$, $d > 1.2$, and $d > 2.0$ respectively [37]. We investigate the statistical significance of the measured improvements using a two-sample t-test, reporting the probability $p$ (i.e. $p$-value) for the measured improvements to be due to chance. We first discuss metrics used for all four tasks, and then we discuss metrics specific to individual tasks.

#### 5.3.1 Viewpoint translation

One metric common to all tasks is subject viewpoint translation. Our visualization brings in additional perspectives that are invoked and examined with intuitive head motions and without requiring locomotion to assume their corresponding viewpoints. Therefore, for all tasks, we expect a reduction in the total distance traveled by the subjects in the experimental group compared to those in the control group. As shown in Table 1, the total distance traveled by the subjects in the experimental group is significantly shorter. For example, for the VR1 task, 68% of the control group subjects traveled between $13.7m - 3.4m$ and $13.7m + 3.4m$ (i.e. one standard deviation). The difference between the averages is 8.3m, and Cohen's $d$ is 2.3, which corresponds to a huge effect size. The largest benefit of multiperspective visualization is measured for task VR2, where the experimental group subjects learned how to search for targets by staying in place, and where the control group

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2778249, IEEE Transactions on Visualization and Computer Graphics

TABLE 1: Average distance traveled per subject, in meters.

| task | control | experiment | difference | $p$ | $d$ | effect size |
|---|---|---|---|---|---|---|
| VR1 | $13.7 \pm 3.4$ | $5.4 \pm 3.8$ | 8.3 | $< 0.01$ | 2.3 | huge |
| VR2 | $91.7 \pm 47.0$ | $8.2 \pm 7.3$ | 83.4 | $< 0.01$ | 2.5 | huge |
| AR1 | $99.1 \pm 3.0$ | $84.8 \pm 12.3$ | 14.4 | $< 0.01$ | 1.6 | very large |
| AR2 | $109.3 \pm 16.9$ | $93.3 \pm 17.6$ | 16.0 | 0.04 | 0.9 | large |

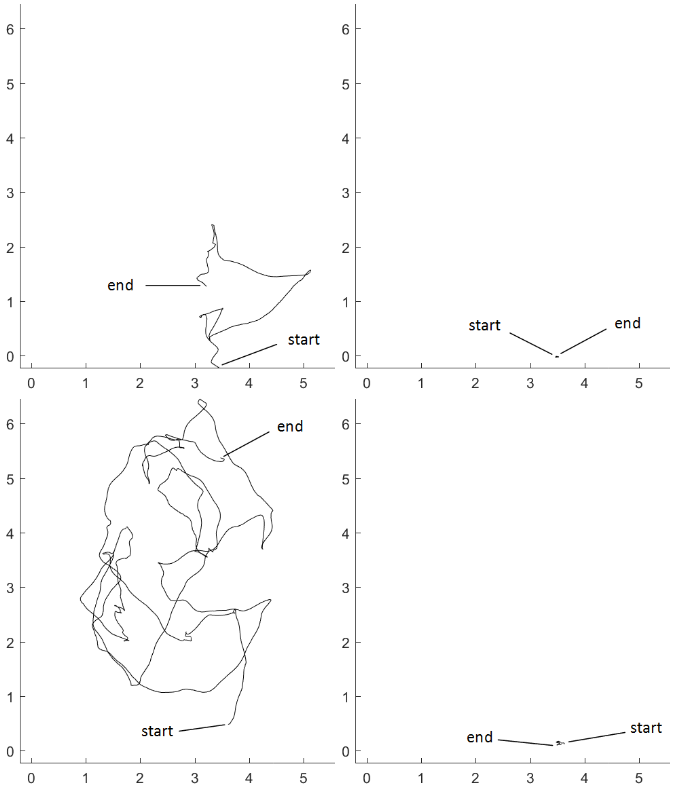


Fig. 8: User viewpoint 2D trajectory visualizations for the urban VR tracking (top) and pair matching (bottom) tasks. When using a conventional visualization (left), the user moves over considerable distances. When using the multiperspective visualization (right), the user can complete the task by standing in one place.

subjects move repeatedly back and forth as they forget the colors of the targets. The smallest, but still large, benefit of multiperspective visualization is measured for task AR2, where the targets are spawned as far away as possible from the subject. For the VR tasks, the improvement brought by multiperspective visualization is statistically significant with $p$-values below 0.01. Due to the variable nature of target spawn locations, measurements for task AR2 also suffer slightly lower statistical significance ($p$-value of 0.04). Overall, the large effect sizes measured for this and other metrics (Tables 2, 3, and 4) justify the number of subjects used in the control and experimental groups, i.e. eight and eight.

Fig. 8 shows the actual trajectories of the control and experimental group subjects who moved the least while completing the urban VR tasks. The control group subject covers a significant part of the scene, whereas the experimental group subject stays within 0.2m of the starting position.

TABLE 2: Total head rotation, in hundreds of degrees.

| task | control | experiment | difference | $p$ | $d$ | effect size |
|---|---|---|---|---|---|---|
| VR1 | $7.8 \pm 1.3$ | $4.4 \pm 2.4$ | 3.3 | $< 0.01$ | 1.8 | very large |
| VR2 | $66.0 \pm 22.5$ | $10.0 \pm 4.8$ | 56.0 | $< 0.01$ | 3.4 | huge |
| AR1 | $41.6 \pm 6.9$ | $32.8 \pm 10.8$ | 8.7 | 0.04 | 1.0 | large |
| AR2 | $43.9 \pm 8.7$ | $33.7 \pm 6.8$ | 10.1 | 0.01 | 1.3 | very large |

TABLE 3: Average head downward rotation, in degrees.

| task | control | experiment | difference | $p$ | $d$ | effect size |
|---|---|---|---|---|---|---|
| VR1 | $57.9 \pm 14.3$ | $40.9 \pm 14.2$ | 17.0 | 0.02 | 1.2 | very large |
| VR2 | $48.5 \pm 4.1$ | $28.4 \pm 5.2$ | 20.0 | $< 0.01$ | 4.3 | huge |

### 5.3.2 View direction rotation

The second metric common to all tasks is subject view direction rotation. The multiperspective visualization performs automatically most of the view direction rotation needed for the user to gain line of sight to the target. Therefore, for all tasks, we also expect a reduction in the total view direction rotation performed by the subjects in the experimental group. As shown in Table 2, the subjects in the experimental group rotated their view direction significantly less. All $p$-values are less than 0.05.

For the target tracking (VR1) and pair matching (VR2) tasks, where subjects tilt their head downwards to look at targets at street level, we analyzed the average downward head rotation (Table 3). The subjects in the experimental group required significantly less downward head rotation ($p < 0.02$), which is an important improvement. When the subject looks down close their feet, the HMD's limited field of view covers only a small part of the scene, which requires moving a lot to search for and to track the target. In addition, seeing only a small part of the scene reduces spatial awareness, which further complicates the tasks. When the view direction is less tilted, the scene is farther away and more of it is visible, which improves task completion efficiency. Finally, viewing the scene with a downward titled view direction is uncomfortable since it leads to an imbalanced distribution of the HMD's weight, as spontaneously reported by two of the subjects.

### 5.3.3 Task completion time

Table 4 gives completion times for the tasks, with the exception of VR1 where the task is complete at the end of the fixed 38 second target trajectory. In the pair matching task (VR2), subjects complete the task in significantly shorter time in the experimental group ($p < 0.01$). This is expected because the subject is able to simultaneously examine two spheres, when one of them is marked as ROI and kept disoccluded, and the other is found and disoccluded by focusing the view on it. In contrast, the subject in the control group is only able to examine the spheres sequentially, with significant physical motion in between. In the search task (AR1), the subject in the experimental group benefits from the multiperspective visualization which shows multiple additional perspectives simultaneously without requiring physical locomotion, while the subject in the control group must translate their viewpoint physically, and is only able to examine side corridors sequentially. Therefore, task AR1 is performed significantly faster in the experimental group than in the control group ($p = 0.01$). In the ambush task

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2017.2778249, IEEE Transactions on Visualization and Computer Graphics

TABLE 4: Task completion time, in seconds.

| task | control | experiment | difference | $p$ | $d$ | effect size |
|---|---|---|---|---|---|---|
| VR2 | $215.5 \pm 86.4$ | $57.8 \pm 26.5$ | 157.7 | $< 0.01$ | 2.5 | huge |
| AR1 | $91.5 \pm 8.0$ | $69.0 \pm 19.5$ | 22.5 | $< 0.01$ | 1.5 | very large |
| AR2 | $75.1 \pm 24.9$ | $73.6 \pm 19.3$ | 1.5 | 0.45 | 0.1 | very small |

(AR2), however, subjects in the experiment and control groups take about the same time to complete the task. The subject in the experimental group is able to leverage the multiperspective visualization to quickly locate the mobile target, but may choose to spend time to wait to ambush. This strategy results in overall less physical locomotion for the experimental group, but not in faster task completion.

### 5.3.4 Task-dependent metrics

For the target tracking task (VR1), we measured for what percentage of the time the subject succeeded at keeping the target in sight. The target sphere is considered visible if any part of it is in the subject's view. In the experimental group, the subjects achieved $97.5\%$ of target visibility compared to the control group, which achieved only $74.1\%$ of target visibility. In other words, even by walking continually, the subject cannot always keep the target in sight. When the target makes a turn, the user has to translate significantly to realign with the street on which the target moves, which takes time. The improvement is 23.3 percentage points with a Cohen's $d$ of 1.4, which corresponds to a very large effect size. This effect is statistically significant ($p = 0.01$).

For the pair matching task (VR2), the subject is asked to select two spheres of the same color in succession. If the second sphere selected is of a different color, then this pair of selections is considered a matching error. The subjects in the experimental group incurred 1.6 erroneous selections on average, while the subjects in the control group incurred a higher rate of 2.5 erroneous selections on average. The improvement is 0.9 selections with a Cohen's $d$ of 0.4, a small effect size. The effect is not statistically significant ($p = 0.22$).

### 5.3.5 Effect of subject prior AR/VR experience on task performance

Due to the absence of familiarization periods allowed before each task, the subjects' prior experiences with VR and AR applications were analyzed for any possible effect on task performance. From the eight subjects with prior VR experience, four were assigned to the control group and four to the experiment group, for each of the tasks VR1 and VR2. For AR1, from the six subjects with prior AR experience, two were assigned to the control group and four were assigned to the experiment group. For AR2, three of the AR experienced subjects were assigned to each of the two groups. Overall, there were no consistent differences between subjects with and without prior experience within single groups, and any difference found was statistically insignificant. Due to the small number of subjects with experience in a group, a dedicated user study is needed to investigate fully the effect of prior experiences when using our visualization methods.

## 6 Conclusions and Future Work

We have presented novel multiperspective visualizations to improve navigation efficiency in AR and VR. Our visualization techniques seamlessly integrate perspectives from multiple secondary viewpoints into the main user perspective. The secondary perspectives are selected by the user intuitively, with minimal or no interface manipulation. The output supports stereoscopic displays by rendering the scene with correct depth cues. We demonstrate the effectiveness of our visualization techniques in a user study where subjects were asked to accomplish tasks in AR and VR using an HMD with tracked pose. Based on the study, we report significant improvement in navigation efficiency while using multiperspective visualization compared with using conventional visualization.

There exist limitations to our visualization techniques. The urban VR visualization assumes that any ROI at street level can be disoccluded from an overhead perspective. This assumption does not hold, for example, when the target object hides underneath a bridge. It is possible to include additional panning control to the construction of secondary views, but this increases the complexity of the user interface. If the target is unreachable, e.g., it hides inside an enclosed space, then even the addition of panning control cannot bring it into view, and we must resort to X-ray or cutaway visualizations. The indoor AR visualization seamlessly swings side corridors into view, but the deformation operation can only guarantee continuity between real and virtual scene parts joined at a planar portal. Supporting non-planar portals is possible at the cost of visual complexity.

Fig. 9 compares our multiperspective visualization approach to handling occlusions to X-ray visualization, for both the urban VR and the indoor AR scenes. For the VR scene, the X-ray visualization does not convey the location of the target (red sphere) with respect to the network of streets. The multiperspective visualization anchors the target at the correct street location, and, unlike a simple overhead view, does also convey the height of the nearby buildings. For the AR scene, the X-ray visualization covers a significant fraction of the nearby occluding wall with the side corridor, which obscures parts of the scene close to the user. Our multiperspective visualization does not blend mismatching colors, which maintains the clarity of the visualization, and confines the depiction of the side corridor to the user's view of the portal. Overall, our visualization presents a different approach to the problem of missing depth cues suffered by X-ray views.

Our user study relies on conventional visualization for the control group. The user manages occlusions by physically navigating the viewpoint to locations from where there is line of sight to potential ROIs. This first study provides a useful baseline for our visualization approach. Future studies could attempt to compare with other approaches to enhance navigation, such as occlusion management based on X-ray, cutaway, and explosion visualization. We foresee that some applications, scenes, and tasks will be better served by one approach over another. Furthermore, hybrid approaches should also be investigated.

Our approach to VR navigation has potential advantages in a multi-user environment. Unlike non-linear VR naviga-

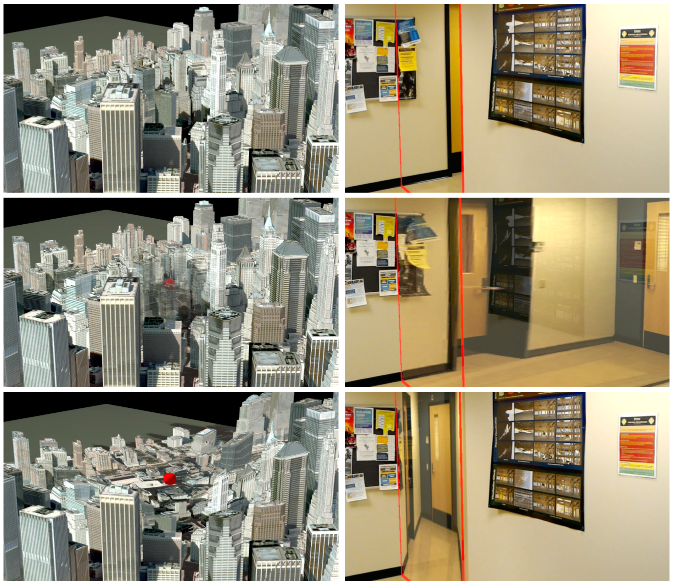


Fig. 9: Comparison between conventional (top row), X-ray (middle row), and multiperspective (bottom row) VR (left) and AR (right) visualizations. The street-level target (left) and the side corridor (right) are occluded in conventional VR and AR visualizations. Unlike the multiperspective visualizations, the X-ray visualizations do not convey the street-level location of the target and the nearby geometry.

tion techniques such as teleportation, redirected walking, or non-linear mapping, our method strictly preserves mapping between physical and virtual spaces at unit scale for the main, user perspective. This opens up possibility for multiple users to simultaneously collaborate in a shared VR scene. Consider the urban VR scene visualized by multiple users. The same target can be disoccluded simultaneously for all users with multiperspective visualizations tailored to each individual user. Each multiperspective visualization shows the target at the same undistorted location, which allows the multiple users to refer to the target consistently.

Our work introduces a paradigm where the user receives visual assistance from additional perspectives, without the disorienting effect of abandoning the main perspective. Our work does not advocate for supplanting all user physical locomotion through the use of multiperspective visualization. Instead, our contribution is a flexible framework where the user can choose how much to walk and how much to rely on the multiperspective visualization. For example, in target tracking, some users relied completely on the multiperspective visualization which all but eliminated the need to walk, and some other users only relied on the multiperspective visualization to simplify their walking trajectories, in order to succeed at keeping the target in sight without having to walk fast and with abrupt turns. Inspired by the significant reduction of physical locomotion for some users in the urban VR scene, future work could study multiperspective visualization in VR or AR applications where the user is seated or almost stationary. Such applications include navigation between multiple virtual displays in a VR desktop, assisted surgical operations with multiperspective augmented annotations, or improved situation awareness during operation of machinery or vehicles.

In the indoor scene, the multiperspective visualization shows what is visible beyond the first turn, and the user still has to walk to be at most one turn away from the target. Future work could examine extending the multiperspective visualization to show what is visible beyond the second and third turns away from the current user position, which will undoubtedly come at the cost of increased visualization complexity. Future studies are needed that will leverage the flexibility of our multiperspective rendering framework to investigate the optimal trade off between multiperspective disocclusion power and visualization eloquence

## References

[1] J. L. Souman, P. R. Giordano, M. Schwaiger, I. Frissen, T. Thümmel, H. Ulbrich, A. D. Luca, H. H. Bülthoff, and M. O. Ernst, "Cyberwalk: Enabling unconstrained omnidirectional walking through virtual environments," *ACM Trans. Appl. Percept.*, vol. 8, no. 4, pp. 25:1–25:22, Dec. 2008. [Online]. Available: http://doi.acm.org/10.1145/2043603.2043607

[2] S. Tregillus and E. Folmer, "Vr-step: Walking-in-place using inertial sensing for hands free navigation in mobile vr environments," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, ser. CHI '16. New York, NY, USA: ACM, 2016, pp. 1250–1255. [Online]. Available: http://doi.acm.org/10.1145/2858036.2858084

[3] M. Usoh, K. Arthur, M. C. Whitton, R. Bastos, A. Steed, M. Slater, and F. P. Brooks, Jr., "Walking > walking-in-place > flying, in virtual environments," in *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '99. New York, NY, USA: ACM Press/Addison-Wesley Publishing Co., 1999, pp. 359–364. [Online]. Available: http://dx.doi.org/10.1145/311535.311589

[4] R. A. Ruddle and S. Lessels, "The benefits of using a walking interface to navigate virtual environments," *ACM Trans. Comput.-Hum. Interact.*, vol. 16, no. 1, pp. 5:1–5:18, Apr. 2009. [Online]. Available: http://doi.acm.org/10.1145/1502800.1502805

[5] S. Razzaque, D. Swapp, M. Slater, M. C. Whitton, and A. Steed, "Redirected walking in place," in *Proceedings of the Workshop on Virtual Environments 2002*, ser. EGVE '02. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2002, pp. 123–130. [Online]. Available: http://dl.acm.org/citation.cfm?id=509709.509729

[6] S. Razzaque, Z. Kohn, and M. C. Whitton, "Redirected Walking," in *Eurographics 2001 - Short Presentations*. Eurographics Association, 2001.

[7] J. J. LaViola, Jr., D. A. Feliz, D. F. Keefe, and R. C. Zeleznik, "Hands-free multi-scale navigation in virtual environments," in *Proceedings of the 2001 Symposium on Interactive 3D Graphics*, ser. I3D '01. New York, NY, USA: ACM, 2001, pp. 9–15. [Online]. Available: http://doi.acm.org/10.1145/364338.364339

[8] X. Xie, Q. Lin, H. Wu, G. Narasimham, T. P. McNamara, J. Rieser, and B. Bodenheimer, "A system for exploring large virtual environments that combines scaled translational gain and interventions," in *Proceedings of the 7th Symposium on Applied Perception in Graphics and Visualization*, ser. APGV '10. New York, NY, USA: ACM, 2010, pp. 65–72. [Online]. Available: http://doi.acm.org/10.1145/1836248.1836260

[9] B. Williams, G. Narasimham, B. Rump, T. P. McNamara, T. H. Carr, J. Rieser, and B. Bodenheimer, "Exploring large virtual environments with an hmd when physical space is limited," in *Proceedings of the 4th Symposium on Applied Perception in Graphics and Visualization*, ser. APGV '07. New York, NY, USA: ACM, 2007, pp. 41–48. [Online]. Available: http://doi.acm.org/10.1145/1272582.1272590

[10] Q. Sun, L.-Y. Wei, and A. Kaufman, "Mapping virtual and physical reality," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 64:1–64:12, Jul. 2016. [Online]. Available: http://doi.acm.org/10.1145/2897824.2925883

[11] F. Steinicke, G. Bruder, J. Jerald, H. Frenz, and M. Lappe, "Analyses of human sensitivity to redirected walking," in *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology*, ser. VRST '08. New York, NY,

USA: ACM, 2008, pp. 149–156. [Online]. Available: http://doi.acm.org/10.1145/1450579.1450611

[12] ——, "Estimation of detection thresholds for redirected walking techniques," *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 1, pp. 17–27, Jan 2010.

[13] Y. Kameda, T. Takemasa, and Y. Ohta, "Outdoor see-through vision utilizing surveillance cameras," in *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*, Nov 2004, pp. 151–160.

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

[15] C. Sandor, A. Cunningham, A. Dey, and V. V. Mattila, "An augmented reality x-ray system based on visual saliency," in *2010 IEEE International Symposium on Mixed and Augmented Reality*, Oct 2010, pp. 27–36.

[16] S. Zollmann, D. Kalkofen, E. Mendez, and G. Reitmayr, "Image-based ghostings for single layer occlusions in augmented reality," in *2010 IEEE International Symposium on Mixed and Augmented Reality*, Oct 2010, pp. 19–26.

[17] S. Zollmann, R. Grasset, G. Reitmayr, and T. Langlotz, "Image-based x-ray visualization techniques for spatial understanding in outdoor augmented reality," in *Proceedings of the 26th Australian Computer-Human Interaction Conference on Designing Futures: The Future of Design*, ser. OzCHI '14. New York, NY, USA: ACM, 2014, pp. 194–203. [Online]. Available: http://doi.acm.org/10.1145/2686612.2686642

[18] M. Burns and A. Finkelstein, "Adaptive cutaways for comprehensible rendering of polygonal scenes," *ACM Trans. Graph.*, vol. 27, no. 5, pp. 154:1–154:7, Dec. 2008. [Online]. Available: http://doi.acm.org/10.1145/1409060.1409107

[19] W. Li, L. Ritter, M. Agrawala, B. Curless, and D. Salesin, "Interactive cutaway illustrations of complex 3d models," *ACM Trans. Graph.*, vol. 26, no. 3, Jul. 2007. [Online]. Available: http://doi.acm.org/10.1145/1276377.1276416

[20] W. Li, M. Agrawala, B. Curless, and D. Salesin, "Automated generation of interactive 3d exploded view diagrams," *ACM Trans. Graph.*, vol. 27, no. 3, pp. 101:1–101:7, Aug. 2008. [Online]. Available: http://doi.acm.org/10.1145/1360612.1360700

[21] S. Bruckner and M. E. Groller, "Exploded views for volume data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 5, pp. 1077–1084, Sept 2006.

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

[23] H. Deng, L. Zhang, X. Mao, and H. Qu, "Interactive urban context-aware visualization via multiple disocclusion operators," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 7, pp. 1862–1874, July 2016.

[24] R. I. Hartley and R. Gupta, *Linear pushbroom cameras*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 555–566. [Online]. Available: http://dx.doi.org/10.1007/3-540-57956-7_63

[25] P. Rademacher and G. Bishop, "Multiple-center-of-projection images," in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '98. New York, NY, USA: ACM, 1998, pp. 199–206. [Online]. Available: http://doi.acm.org/10.1145/280814.280871

[26] J. Yu and L. McMillan, *General Linear Cameras*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 14–27. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-24671-8_2

[27] C. Mei, V. Popescu, and E. Sacks, "The Occlusion Camera," *Computer Graphics Forum*, 2005.

[28] 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*, vol. 16, no. 6, pp. 1235–1242, Nov 2010.

[29] V. Popescu, P. Rosen, and N. Adamo-Villani, "The graph camera," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 158:1–158:8, Dec. 2009. [Online]. Available: http://doi.acm.org/10.1145/1618452.1618504

[30] M. L. Wu and V. Popescu, "Multiperspective focus+context visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 5, pp. 1555–1567, May 2016.

[31] E. Veas, R. Grasset, E. Kruijff, and D. Schmalstieg, "Extended overview techniques for outdoor augmented reality," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 4, pp. 565–572, April 2012.

[32] H. Lorenz, M. Trapp, J. Döllner, and M. Jobst, *Interactive Multi-Perspective Views of Virtual 3D Landscape and City Models*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 301–321. [Online]. Available: https://doi.org/10.1007/978-3-540-78946-8_16

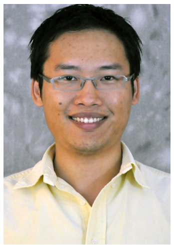
[33] S. Kim and A. K. Dey, "Simulated augmented reality windshield display as a cognitive mapping aid for elder driver navigation," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '09. New York, NY, USA: ACM, 2009, pp. 133–142. [Online]. Available: http://doi.acm.org/10.1145/1518701.1518724

[34] M. Tatzgern, D. Kalkofen, R. Grasset, and D. Schmalstieg, "Embedded virtual views for augmented reality navigation," *International Symposium On Mixed and Augmented Reality - Workshop on Visualization in Mixed Reality Environments*, 2011.

[35] 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 *2010 IEEE Virtual Reality Conference (VR)*, March 2010, pp. 47–50.

[36] J. Cohen, *Statistical power analysis for the behavioral sciences: Jacob Cohen*. Psychology Press, 2009.

[37] S. S. Sawilowsky, "New effect size rules of thumb," *Journal of Modern Applied Statistical Methods*, vol. 8, no. 2, pp. 597–599, 2009.



**Meng-Lin Wu** is a Ph.D. student in the Computer Science Department of Purdue University. He received B.S. and M.S. degrees in physics from National Taiwan University, Taiwan in 2005 and 2007 while participating in experimental high-energy physics at KEK, Japan. Prior to joining Purdue, he programmed game physics at International Games System. His research is concerned with occlusion and visibility management in visualization and augmented reality. His current projects include camera model design, 3D scene acquisition and synthesis, and educational gesturing avatars.

**Voicu Popescu** received a B.S. degree in computer science from the Technical University of Cluj-Napoca, Romania in 1995, and a Ph.D. degree in computer science from the University of North Carolina at Chapel Hill, USA in 2001. He is an associate professor with the Computer Science Department of Purdue University. His research interests lie in the areas of computer graphics, computer vision, and visualization. His current projects include camera model design, perceptual evaluation of rendered imagery and 3D displays, research and development of 3D scene acquisition systems, and research, development, and assessment of next generation distance learning systems.