Restoring the Awareness in the Occluded Visual Field for Optical See-Through Head-Mounted Displays

Long Qian, *Student Member, IEEE*, Alexander Plopski, *Member, IEEE*,
Nassir Navab, *Member, IEEE*, and Peter Kazanzides, *Member, IEEE*

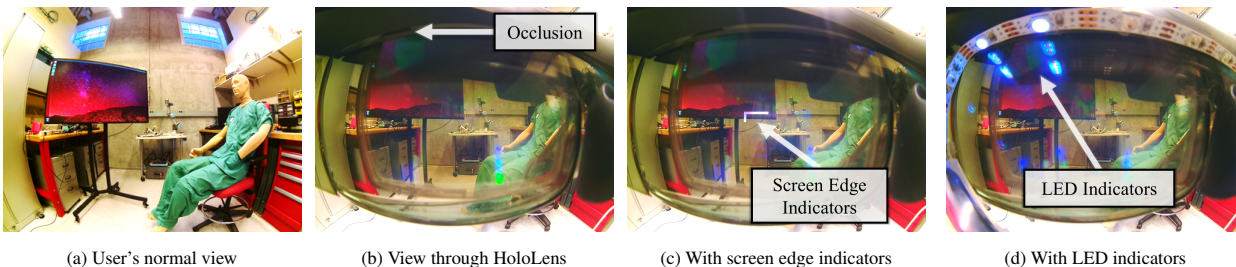| (a) User's normal view | (b) View through HoloLens | (c) With screen edge indicators | (d) With LED indicators |

Fig. 1: When wearing an OST-HMD, the natural vision of a user (a) becomes restricted by the structure of the OST-HMD (b). The occlusion causes the loss of awareness of the environment, which is dangerous in many situations, e.g., driving and manufacturing. In this paper, we propose to compensate for the loss of awareness with screen edge indicators on the display (c) or with LED indicators attached to the OST-HMD (d).

Abstract—
Recent technical advancements support the application of Optical See-Through Head-Mounted Displays (OST-HMDs) in critical situations like navigation and manufacturing. However, while the form-factor of an OST-HMD occupies less of the user's visual field than in the past, it can still result in critical oversights, e.g., missing a pedestrian while driving a car. In this paper, we design and compare two methods to compensate for the loss of awareness due to the occlusion caused by OST-HMDs. Instead of presenting the occluded content to the user, we detect motion that is not visible to the user and highlight its direction either on the edge of the HMD screen, or by activating LEDs placed in the user's peripheral vision. The methods involve an offline stage, where the occluded visual field and location of each indicator and its associated occluded region of interest (*OROI*) are determined, and an online stage, where an enhanced optical flow algorithm tracks the motion in the occluded visual field. We have implemented both methods on a Microsoft HoloLens and an ODG R-9. Our prototype systems achieved success rates of 100% in an objective evaluation, and 98.90% in a pilot user study. Our methods are able to compensate for the loss of safety-critical information in the occluded visual field for state-of-the-art OST-HMDs and can be extended for their future generations.

Index Terms—View Expansion, Prototype, Optical See-Through Head-Mounted Display

---

1 INTRODUCTION

Head-mounted displays for augmented reality (AR) can be categorized into video see-through head-mounted displays (VST-HMD) and optical see-through head-mounted displays (OST-HMD) [49]. AR presented on HMDs has been applied in a variety of fields, such as aiding manufacturing [13], medical interventions [44], and education [4].

Compared to VST-HMDs, OST-HMDs offer the advantage that they do not block the user's direct vision. They are thus failsafe, i.e., the user can still see the real world if the device fails, and do not falsify the user's view by non-ideal settings of the external view camera. Although OST-HMDs do not directly intercept the user's vision as VST-HMDs do, they still introduce additional interference, e.g., distortion [27] and occlusion [63]. The distortion is due to the refraction of the light by the optical elements in front of the user's eyes. It can be estimated by an offline calibration procedure [27, 32]. However, this calibration does not account for the occlusion of the user's peripheral view by the HMD frame [63] (Fig. 1b). The occluded peripheral view is critical for safe and efficient mobility [35]. The occlusion caused by the frame of the OST-HMD is thus a significant security risk, e.g., a pedestrian may not

see a car coming from the side or a worker may miss a moving robot arm because it is occluded by the HMD frame. With the increasing number of commercial OST-HMDs entering the market, and more AR applications built on OST-HMD platforms, it is important to face the issue of the incomplete awareness and alleviate the potential danger.

An ideal solution would be to use contact-lens type displays, however such displays are in the initial stages of research and it is not clear when they will reach the consumer market. A common approach for VST-HMDs is to capture the invisible area with a camera and then map it onto the display [3, 41]. This solution is not viable for OST-HMDs, as the user can still see the background. A mapping of the invisible areas into the user's field-of-view would inevitably create artifacts.

We present two ways to attract the user's attention to the occluded content. The first method displays indicated areas of interest as a contour in the augmentable area (Fig. 1c). This method takes advantage of the area that the user is most likely focusing on. By visualizing information in the edge area of the display, the user can easily understand in what direction occluded motion is detected and direct his/her attention towards it, if necessary. The second method takes advantage of the anatomy of the eye. As humans are more sensitive to brightness and contrast changes in their periphery [1, 55, 62], brightness changes in the area occluded by the OST-HMD frame will naturally attract the user's attention. To adjust the brightness in these areas, we attached LED lights to the OST-HMD frame and adjust their brightness to indicate detected motion (Fig. 1d).

The main contributions of our paper are:

- *Long Qian and Peter Kazanzides are with Johns Hopkins University. E-mail: {long.qian, pkaz}@jhu.edu.*
- *Alexander Plopski is with Nara Institute of Science and Technology. E-mail: plopski@is.naist.jp.*
- *Nassir Navab is with Technical University of Munich and Johns Hopkins University. E-mail: nassir.navab@tum.de.*
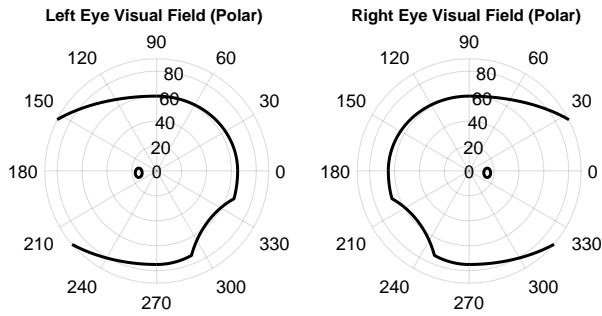
Fig. 2: Sample of synthesized visual field for human left and right eye (V_{EL} and V_{ER}), represented in polar coordinate system.

- We describe how to estimate the occlusion caused by OST-HMDs on a user's visual field during an offline calibration step, and present two methods (screen edge and LED indicators) to restore the user's awareness of the occluded areas during operation.
- We build prototypes of both methods integrated into state-of-the-art OST-HMDs (Microsoft HoloLens and ODG R-9).
- We evaluate and compare the prototypes in objective experiments and a pilot user study.

2 BACKGROUND AND LITERATURE REVIEW

Our work considers the user's field-of-view (FOV) to restore the vision loss caused by the frame of the HMD. In this section, we first review the anatomy of the eye and the user's field of vision. We then discuss previous studies that investigated the effects of diminished field of vision, as well as previous work on extending the user's FOV in HMDs.

2.1 Human visual field

The field of vision is that portion of space in which objects are visible during gaze fixations [53]. A region in the visual field where no target can be seen is called absolute scotoma. Areas where some targets can be seen, but smaller ones are not visible are referred to as relative scotoma [63]. Fig. 2 shows a synthesized visual field in a polar coordinate system that captures the main features of the normal human visual field (typically 100° lateral, 60° medial, 60° upward and 75° downward span of visual field [53]; scotoma at the blind spot and nose; interpolation with elliptical curves). The blind spot is a portion of the human retina without any photoreceptors [20].

The human retina contains three types of photoreceptor cells that facilitate our vision: rod, cone and non-image-forming photosensitive ganglion cell [12, 52]. Cones are responsible for the eye's color sensitivity. They are concentrated in the fovea centralis and there is only a small number of cones in the peripheral area of the eye. Rods are more sensitive to brightness than cones, but insensitive to colors. Their distribution is contrary to the rods, with only a few rods in the fovea centralis, and comparably many in the peripheral area. Therefore, while humans have very accurate vision of the focused area, we become less sensitive to details in the periphery. In fact, our ability to detect the color of objects in our periphery depends on the size of the stimuli [50].

2.2 Occlusion of Peripheral Vision and Danger

The hardware of an OST-HMD causes both absolute scotoma and relative scotoma to the user's visual field [63]. The occlusion is usually on the user's peripheral vision because the center part is designed for graphics overlays and direct (see-through) vision of the environment. Peripheral vision is critical for safe and efficient mobility [35]. Johnson et al. [29] studied the loss of visual field and its relationship to driving performance. They screened 10,000 drivers and found that drivers with binocular visual field loss had accident and conviction rates twice as high as those with normal visual fields. Szlyk et al. [57] found that driving performance of glaucoma patients correlates with peripheral visual field loss. Apart from keeping the person safe, peripheral vision also contributes to form vision [55] and scene gist recognition by resolving lower spatial frequencies [34]. Researchers have also proposed using peripheral perception for conveying information [16, 33].

Therefore, when an AR application on an OST-HMD is used in a mobile scenario, or requires the user to pay attention to the surroundings, developers and designers must seriously consider the occlusion due to the OST-HMD.

2.3 View Expansion

The loss of the visual field is a concern in VR and AR applications. Methods to expand the user's FOV to improve game experience or to facilitate localization of objects have been studied in the context of normal monitors [30], 3D monitors [8], mobile devices [59], heads-up displays [58], skier's helmet [40] and HMDs. For HMDs, different methods have been developed to expand the FOV of the virtual and real environments. We show the taxonomy in Tab. 1.

Expanding the FOV in the virtual environment is an off-screen visualization technique. The system is aware of the location of off-screen objects and provides hints to the user about their existence. Common approaches involve presenting the user with a world in miniature view [5, 9, 54] or guiding the user's attention towards areas of interest through indication on the display [5, 7, 9, 25, 47, 56]. Xiao et al. [64] proposed the concept of sparse peripheral display, and integrated sparse LED lights into both a VST-HMD and an OST-HMD in order to facilitate object search and reduce motion sickness.

To expand the user's FOV in the real environment, Ardouin et al. [3] captured 360° FOV images and displayed them to the user on a VST-HMD. Fan et al. [17] proposed SpiderVision to analyze the back-view image and overlay it on the front-view image with a VST-HMD. Miyaki et al. [38] proposed LiDARMAN, where the user sees a third-person view of himself/herself in a point cloud of the environment on a VST-HMD. Orlosky et al. [41] proposed Fisheye Vision to obtain more peripheral FOV. A dynamic view expansion method is also implemented to facilitate visual search with a VST-HMD [65].

In addition to providing information about the environment, similar visualization technologies have also been used to display notifications [11, 36]. Luyten et al. [37] explored how signs presented on displays in the peripheral vision can be used to show information to the user. They also found that motion supported perception and comprehension of the presented information. Researchers have also investigated optical designs to expand the FOV of the display [10, 45, 46], while our solution is built upon existing commercial products.

Overall, [40, 60] appear closest to our work. Vargas et al. [60] extract contours from images captured by a front-facing camera. They found that patients who suffered from a restricted FOV could comprehend their surroundings from minified versions of these contour images displayed on an HMD. However, cluttered background and cluttered virtual content could significantly affect the user's understanding of what he sees [19]. This could also significantly interfere with any AR content displayed on the OST-HMD. Our goal is thus to develop a system that intrudes as little as possible into the augmented area. Our idea is also related to the work of Niforatos et. al. [40]. They activated LEDs attached to the frame of a skier's helmet to notify them of other skiers coming from behind them. Although this concept is very similar to ours, they do not need to consider the user's natural FOV and how much the helmet occludes it, as their focus is on the area behind the user. We are also considering how the augmentable area of the OST-HMD could be used to restore the user's awareness of the occluded areas.

3 METHODS

Our goal is to compensate for the loss of vision in the occluded areas of the OST-HMD. To do so, it is necessary to determine the area that is invisible to a user wearing an OST-HMD, and to define a scheme to compensate for it. In this section, we introduce our approach. We record the area surrounding the user with a wide-angle camera CC that is attached to the OST-HMD. First, we explain the offline calibration

Tab. 1: Literature about view expansion on HMDs

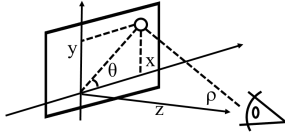| Literature	VST-HMD	OST-HMD
FOV of virtual env.	[6], [14], [21], [22], [54], [56], [64]	[23], [64]
FOV of real env.	[3], [17], [38], [41], [60], [65]	[60]

Fig. 3: Transformation between polar coordinate system of visual field and Cartesian coordinate system

Fig. 4: Sample visual field for human left and right eye (V_{EL} and V_{ER}) in Cartesian coordinate system.

process that determines what area of the image captured by *CC* is invisible to the user due to the occlusion by the OST-HMD. Second, we describe two methods that use two types of indicator to compensate the information loss in the occluded region of interest (*OROI*). We conclude with an explanation of how we visualize the information in an *OROI* using the indicator.

3.1 Notation

We denote some extensively used objects in this paper: left eye (*EL*), right eye (*ER*), center camera (*CC*), camera simulating left eye (*CL*), camera simulating right eye (*CR*). We refer to properties of these objects as the visual field (*V*), occlusion in visual field (*O*), camera intrinsic matrix (*K*), camera distortion function (*D*(·)), indicator (*I*) and its associated occluded region of interest (*OROI*). We refer to properties of a particular object by writing the said object as a subscript of the property. For example, the visual field of the left eye is V_{EL}. When we project information from one object to another, we denote the object it is projected into as a superscript. For example, V_{EL}^{CC} refers to the visual field of the left eye projected onto the visual field of the center camera.

3.2 Determine the occluded visual field: Where to restore the awareness?

Before we can compensate the information occluded by the frame of the OST-HMD, we have to first define the occluded area. To do so, we divide the question "where to restore the awareness" into three sub-problems:

*Q*1: Which part of V_{CC} can the user see normally?
*Q*2: Which part of V_{CC} is occluded by the OST-HMD?
*Q*3: Which part of V_{CC} should the system compensate for the user?

3.2.1 Human visual field projected on camera visual field

To answer *Q*1, we need to project the normal human visual field onto the camera's visual field V_{CC}. The human visual field is usually measured by a perimeter [39], and the results are presented in a polar coordinate system, as in Fig. 2. We first transform the polar representation to a Cartesian coordinate system and then project it to the visual field of the camera. To simplify the calculations we assume that:

*A*1: The user's eyes and the camera *CC* are co-located.
*A*2: The user's viewing direction coincides with *CC*.

*A*1 is equivalent to assuming that all light rays come from infinity. We discuss the error introduced by this assumption in Sec. 5. Assumption *A*2 could be removed with the help of eye-tracking methods that detect the eye's rotation in real-time [31]. Under *A*1 and *A*2, the coordinate systems of *EL*, *ER*, and *CC* are identical. It is thus sufficient to determine what pixel $p(i, j)$ in the image captured by *CC* corresponds to a given angle $p_{polar}(\rho, \theta)$, $0° \leq \rho < 90°$ of V_{EL} and V_{ER}. To determine p, we convert p_{polar} into the Cartesian coordinate system.

Assume a plane at a distance z in front of the eye. The light ray $\vec{L}(x, y, z)$ corresponding to $p_{polar}(\rho, \theta)$ is given by:

$$u = x/z = tan(\rho) \cdot cos(\theta), \quad v = y/z = tan(\rho) \cdot sin(\theta)$$
$$\vec{L}(x, y, z) = \vec{L}(u \cdot z, v \cdot z, z) \quad (1)$$

where z is an arbitrary scaling factor. Fig. 4 shows the human visual field in Fig. 2 in the Cartesian coordinate system. As *EL*, *ER*, and *CC* coincide, $\vec{L}(x, y, z)$ projects onto V_{CC} as:
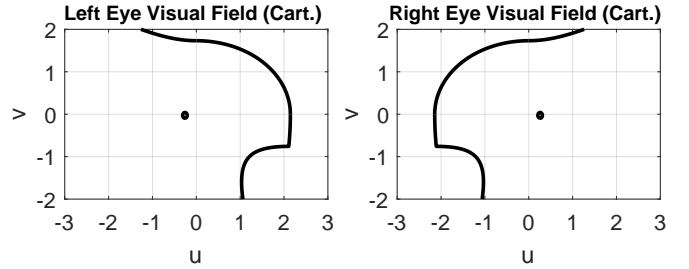

Fig. 5: Human visual field projected on the camera visual field: V_{EL}^{CC} and V_{ER}^{CC}, demonstrated with a sample image.

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = K_{CC} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = K_{CC} \begin{bmatrix} z \cdot tan(\rho) \cdot cos(\theta) \\ z \cdot tan(\rho) \cdot sin(\theta) \\ z \end{bmatrix}$$
$$i' = x'/z', \quad j' = y'/z' \quad (2)$$
$$p(i, j) = D_{CC}(i', j')$$

where K_{CC} is the 3×3 intrinsic matrix of camera *CC*, and function $D_{CC}(\cdot)$ represents distortion. With Eq. 1 and Eq. 2, we can map the human visual field (V_{EL} or V_{ER}) into the camera visual field V_{CC}. The projected visual fields for the left and right eye are denoted V_{EL}^{CC} and V_{ER}^{CC}. Fig. 5 shows an example of V_{EL}^{CC} and V_{ER}^{CC}.

3.2.2 Segmenting occlusion caused by OST-HMD

After we determine what portion of V_{CC} would normally be visible to the user, we need to determine what area is occluded by the OST-HMD. We address *Q*2 by proposing a generic method that is able to segment the inactive area of a camera image frame. We use a pair of cameras with wide-angle lens (*CL* and *CR*) to simulate the user's eyes. We first segment the occluded area in the left and right cameras' own visual field ($O_{CL} \subseteq V_{CL}$ and $O_{CR} \subseteq V_{CR}$), and then project it into the visual field of the center camera $\left(O_{CL}^{CC} \text{ and } O_{CR}^{CC}\right)$.

Ideally, a pixel $(i, j) \in V_{CL}$ captures the frame of the OST-HMD if it is in the occluded area, otherwise, it displays the content of the background. However, the border becomes ambiguous due to reflection and refraction caused by the optics of the OST-HMD. To resolve this ambiguity, we define a function over the image frame that finds the responsiveness score of each pixel with respect to background changes. Based on the belief that reflection or refraction area has lower responsiveness than the direct see-through area, we threshold the responsiveness scores with a threshold value T to filter out the reflection and refraction areas: $Resp_L : V_{CL} \rightarrow \mathbb{R}^+$. Overall, we define the occluded area O_{CL} as

$$O_{CL} = \{(i, j) \mid Resp_L(i, j) < T, (i, j) \in V_{CL}\} \quad (3)$$

The segmentation of occlusion is in the offline stage, so the background can be manually altered to evaluate the responsiveness of pixels. In our method, we display red, green, and blue images on a screen placed in the background.. The camera captures an image for each background condition (IM_1, IM_2 and IM_3) . We define the responsiveness function of a pixel (i, j) as the area of the triangle formed by the RGB values of the three images (illustrated in Fig. 6):

$$(r_k, g_k, b_k) = IM_k(i, j), \quad k = 1, 2, 3$$
$$Resp_L(i, j) = \frac{1}{2} \left| det \begin{pmatrix} r_1 - r_3 & g_1 - g_3 & b_1 - b_3 \\ r_2 - r_3 & g_2 - g_3 & b_2 - b_3 \\ 1 & 1 & 1 \end{pmatrix} \right| \quad (4)$$
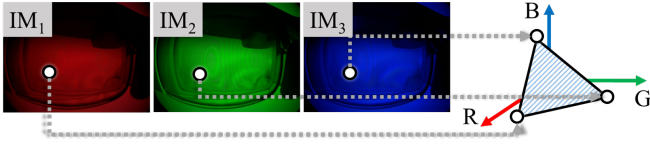
Fig. 6: Each pixel has three (r,g,b) values in the three images of different backgrounds. The responsiveness function of the pixel is defined as the area of the triangle formed by the three color vectors.
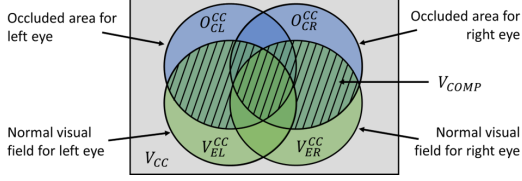

Fig. 7: A Venn diagram for calculating V_{COMP} (the area of human visual field occluded by the frame of OST-HMD) formulated in Eq. 6 (green: human visual field; blue: occlusion of OST-HMD; gray: image frame of the center camera).

Next, we project the occlusion from the left camera's own visual field to that of the center camera, with assumptions $A1$ and $A2$. Given the camera intrinsic matrix and distortion parameters, each pixel $(i,j) \in V_{CL}$ can be mapped to a pixel $(u,v) \in V_{CC}$ by:

$$V_{CL}^{CC} = \{(u,v) \mid (u,v) = D_{CC}(K_{CC}K_{CL}^{-1}D_{CL}^{-1}(i,j)) , (i,j) \in V_{CL}\} \quad (5)$$

If the cameras share the same intrinsic matrices and distortion parameters, Eq. 5 can be reduced to $V_{CL}^{CC} = V_{CL}$. Fig. 10a and Fig. 11a show the segmentation results of HoloLens and ODG R-9. Our segmentation method is able to determine the occluded area, and is applicable to different OST-HMDs.

3.2.3 The loss of visual field

So far, we determined the portion of V_{CC} that the user normally is able to see, and what portion of V_{CC} would be invisible to the user due to the occlusion by the OST-HMD. In this section, we address $Q3$: determine the area of V_{CC} that the user is normally able to see but is not visible when wearing an OST-HMD. This is the area that the system needs to compensate for.

The visibility of a pixel $(i,j) \in V_{CC}$ is defined as:

- if $(i,j) \in V_{EL}^{CC} \cup V_{ER}^{CC}$, the user can see it without an OST-HMD.
- if $(i,j) \in V_{EL}^{CC} \setminus O_{CL}^{CC}$, the user can see it with the left eye when wearing the OST-HMD.
- if $(i,j) \in V_{ER}^{CC} \setminus O_{CR}^{CC}$, user can see it with the right eye when wearing the OST-HMD.
- if $(i,j) \in \left(V_{EL}^{CC} \setminus O_{CL}^{CC}\right) \cup \left(V_{ER}^{CC} \setminus O_{CR}^{CC}\right)$, the user can see it with at least one eye when wearing the OST-HMD.

Given the above definitions, the lost visual field when wearing an OST-HMD can be formally written as:

$$V_{COMP} = \left(V_{EL}^{CC} \cup V_{ER}^{CC}\right) \setminus \left[\left(V_{EL}^{CC} \setminus O_{CL}^{CC}\right) \cup \left(V_{ER}^{CC} \setminus O_{CR}^{CC}\right)\right] \quad (6)$$

We show a Venn diagram of the above equation in Fig. 7. In the case where the visual field of CC is smaller than the human's binocular visual field $\left(V_{EL}^{CC} \cup V_{ER}^{CC} = V_{CC}\right)$, Eq. 6 can be reduced to:

$$V_{COMP} = \left(V_{EL}^{CC} \setminus O_{CL}^{CC}\right)^C \cap \left(V_{ER}^{CC} \setminus O_{CR}^{CC}\right)^C \quad (7)$$

where the superscript C is the complement of a set.

In summary, Sec. 3.2.1, Sec. 3.2.2, and Sec. 3.2.3 addressed $Q1$, $Q2$, and $Q3$ individually. Combining them, we are able to determine the occluded visual field of a user wearing an OST-HMD.

(a) Screen edge indicators on HoloLens   (b) LED indicators on HoloLens

Fig. 8: We propose two methods to visualize the information in the occluded visual field: (a) with the edge of the display on the OST-HMD and (b) with an additional array of LED lights.

3.3 Visualization in the occluded visual field: How to restore the awareness?

We propose to use two types of indicators to highlight the direction of noticeable information in the occluded visual field: with the edge of the display on the OST-HMD and with an additional array of LED lights attached to the frame of the OST-HMD. Fig. 8 demonstrates the user's view when they are activated on a Microsoft HoloLens. Each indicator has its associated occluded region of interest (*OROI*).

3.3.1 Screen edge indicators

Visualization with the screen edge indicators is a compact and portable solution that does not require any additional hardware. It is an off-screen visualization technique [22]. The screen edge indicators are the outer contour of the display, so if the user is focusing on the graphics content on the HMD, he/she is able to see these indicators close to his/her central vision. Once the screen edge indicators are activated, the user will notice the changes and direct his/her attention to the highlighted direction. There are many pixels available for control, so that versatile information can be shown to the user. However, this method reduces the effective area for AR applications. There may also exist a large gap between the location of indicators and the occluded visual field, depending on the FOV of the OST-HMD.

In this method, the edge of the screen of w pixels width is used as indicators. The edge area is discretized into N_S individual indicators. Each indicator appears on the visual field as $I_{Screen,n}^{CC}$, and has an *OROI* that is denoted as $OROI_{Screen,n}^{CC}$. The *OROI* for each indicator is a part of the entire occluded visual field.

The *OROI*s combined for all screen edge indicators cover the entire occluded visual field to be compensated for. We use the left side of the left screen to indicate information for the left half of the visual field, and the right side of the right screen to indicate the right half of the visual field. There is no overlap between multiple *OROI*s. The relationship between the *OROI*s for screen edge indicators are:

$$\bigcup_{n=1}^{N_S} OROI_{Screen,n}^{CC} = V_{COMP} \quad , \quad \bigcap_{n=1}^{N_S} OROI_{Scree,n}^{CC} = \emptyset \quad (8)$$

3.3.2 LED indicators

For our second method, we attached an array of LED lights to the frame of the OST-HMD, as shown in Fig. 8b. A similar setup was integrated with virtual reality or normal glasses in [24,43,64]. The LED indicators are placed at the peripheral vision of the user so they do not interfere with the graphics content on the display and are able to closely reflect the information in the occluded area. It is "as if" the user is seeing one LED light through the frame of the OST-HMD. If the OST-HMD does not offer an interface for custom hardware, an additional wired connection is necessary to power and control the LEDs.

The total number of LED indicators is denoted by N_L. Each LED indicator is a visualization unit that appears in the visual field at $I_{LED,n}^{CC}$, and has an associated *OROI* which is denoted as $OROI_{LED,n}^{CC}$. The *OROI*s of different LEDs overlap. For example, when the system intends to indicate changes at the top-left, the top-left LEDs for the left eye and right eye will both be illuminated. Tab. 2 summarizes the advantages and disadvantages for both types of indicators.

Data: Camera intrinsics and distortion parameters for CC, CL and CR. The number of indicators N_S. Threshold for angular distance θ_{thres}.

Result: The illuminated area of each indicator $I^{CC}_{Screen,n}$. The occluded region of interest for each indicator $OROI^{CC}_{Screen,n}$

begin

  Set $OROI^{CC}_{Screen,n}$ to empty for $n = 1, 2, ..., N_S$;

  for $n = 1$ **to** N_S **do**

    Display *BLACK* in all indicators.;

    foreach *color* in $\{R, G, B\}$ **do**

      Display *color* on indicator i.;

      Capture the view of camera CL and CR as $IM_{L,color}$ and $IM_{R,color}$.;

      Project $IM_{L,color}$ to center camera $IM^{CC}_{L,color}$;

      Project $IM_{R,color}$ to center camera $IM^{CC}_{R,color}$;

      Fuse the simulated binocular image $IM_{color} = (IM^{CC}_{L,color} + IM^{CC}_{R,color})/2$;

    end

    Compute the area of RGB triangle as $Resp(i, j)$ from $\{IM^{CC}_R, IM^{CC}_G, IM^{CC}_B\}$, $\forall (i, j) \in V_{CC}$;

    Threshold and segment the illuminated area $I^{CC}_{Screen,n} = \{(i, j) \mid Resp(i, j) > T, (i, j) \in V_{CC}\}$;

    Find the centroid of $I^{CC}_{Screen,n}$ as $p_n(i_n, j_n)$;

    The single pixel $p_n(i_n, j_n)$ represents the centroid of the nth indicator.;

  end

  forall $(i, j) \in V_{COMP}$ **do**

    Compute angular distance between (i, j) and $p_n(i_n, j_n)$ as θ_n for $n = 1, 2, ..., N_S$;

    Find the smallest $\theta_k \in \{\theta_n \mid n = 1, 2, ..., N_S\}$;

    if $abs(\theta_k) < \theta_{thres}$ **then**

      Add (i, j) to $OROI^{CC}_{Screen,k}$;

    end

  end

end

Alg. 1: The algorithm to determine the occluded region of interest (*OROI*) for each screen edge indicator

3.3.3 Determine *OROI* for indicators

Each indicator is responsible for a portion of the occluded visual field as $OROI^{CC}_{Screen,n}$ or $OROI^{CC}_{LED,n}$. The *OROI* for each indicator is calculated by Alg. 1. This procedure is conducted offline with cameras CL and CR simulating the eyes.

First, the illuminated area of the indicator $I^{CC}_{Screen,n}$ is segmented using the method described in Sec. 3.2.2. Each indicator is manually controlled to display red, green and blue sequentially. Because $I^{CC}_{Screen,n}$ is an area, we use the centroid of the area (a single pixel p_n) to represent each indicator. For the pixel $(i, j) \in V_{COMP}$, we find its closest indicator p_k in terms of angular distance. If this angular distance is smaller than a threshold, this pixel belongs to the *OROI* of p_k. The algorithm can be applied to LED indicators in the same manner as screen edge indicators. As an example, Fig. 10e and Fig. 10f show the indicators and their *OROI*s for HoloLens. Fig. 11c and Fig. 11d show the indicators and their *OROI*s for ODG R-9.

Tab. 2: Comparison between screen edge indicators and LED indicators

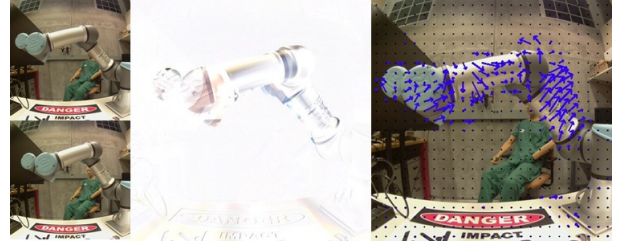| Criteria	Screen	LED
1. Number of indicators	High	Low
2. Distance between indicator and 'incident'	Far	Close
3. Interference with the content on display	Yes	No
4. Complexity of setup	Easy	Hard
5. Additional wired connection	No	Yes
6. Overlap in *OROI*s	No	Yes
7. Reflection and refraction artifacts	No	Yes

Fig. 9: Left: the two frames captured with motion of the robot arm. Center: the absolute difference of the left images. Right: dense optical flow calculated with [18]. Each grid point has an associated arrow showing the vector of optical flow. (The scale of the vector is multiplied by 3 for better visualization.)

3.4 Information processing of the *OROI*s

During runtime we need to determine what information to compensate for in the *OROI*s. One intuitive approach is to average the color of all pixels in $OROI^{CC}_{Screen,n}$ as the color of the indicator:

$$Color\left(I^{CC}_{Screen,n}\right) \leftarrow Avg\left(Color(i, j)\right), \forall (i, j) \in OROI^{CC}_{Screen,n} \quad (9)$$

This approach visualizes the state of the environment. It almost always assigns some color for each indicator, which might be distracting for the user. We propose to visualize the change of environment instead of the state of the environment, by calculating the optical flow [26]. The indicator is activated to display a white color when significant motion is detected in its *OROI*. The brightness value to display is dependent on the extent of change.

Our approach resembles how human attention is attracted by stimuli in the peripheral vision. The human eye is sensitive to motion and contrast in the peripheral vision [1]. Therefore, we map the optical flow of the environment to the brightness changes of the indicators as stimuli. Another factor to consider is the egocentric motion of the user's head. When the user's head is turning, the peripheral vision is constantly changing and the human visual system is already accustomed to this. In this case, an ideal algorithm should be able to distinguish between egocentric head motion and motion of other objects at the periphery.

To detect motion we compute a dense optical flow $Flow(t)$ between the current visual field $V_{CC}(t)$ and the previous visual field $V_{CC}(t-1)$, using Gunner Farneback's algorithm [18]. The optical flow is a per pixel motion vector across the two image frames. If pixel $(i_t, j_t) \in V_{CC}(t)$ is corresponding to the pixel $(i_{t-1}, j_{t-1}) \in V_{CC}(t-1)$, then the ideal optical flow is represented as: $\overrightarrow{Flow}(i, j, t) = (i_t - i_{t-1}, j_t - j_{t-1})$.

After the optical flow is calculated, the brightness value of each indicator at the current frame is assigned to be:

$$Brightness\left(I^{CC}_{Screen,n}\right) \leftarrow \lambda \cdot Avg\left(\|\overrightarrow{Flow}(i, j, t)\|\right)$$
$$for \quad \|\overrightarrow{Flow}(i, j, t)\| > F_{thres}, \quad \forall (i, j) \in OROI^{CC}_{Screen,n} \quad (10)$$

where λ is a constant coefficient that tunes the overall brightness, and F_{thres} is the threshold for minimum flow intensity to filter out noise. In addition, we added one more condition to Eq. 10: a simple eccentricity metric for the values of all indicators. The indicators are only activated when the maximum brightness value is significantly larger than the average brightness value. Otherwise, we treat it as motion of the entire frame due to the user's head motion.

With the indicators and their associated *OROI*s determined in the offline stage, and the online algorithm to process the visual field of camera CC, the real-time control loop for the indicators is complete.

4 IMPLEMENTATION AND SETUP

We integrated both indicators for a Microsoft HoloLens and an ODG R-9.

4.1 Experimental setup for offline stage

Fig. 12a shows the system setup with Microsoft HoloLens in the offline stage. We use SanSmart 5MP Mini cameras as CC, CL, CR. Each
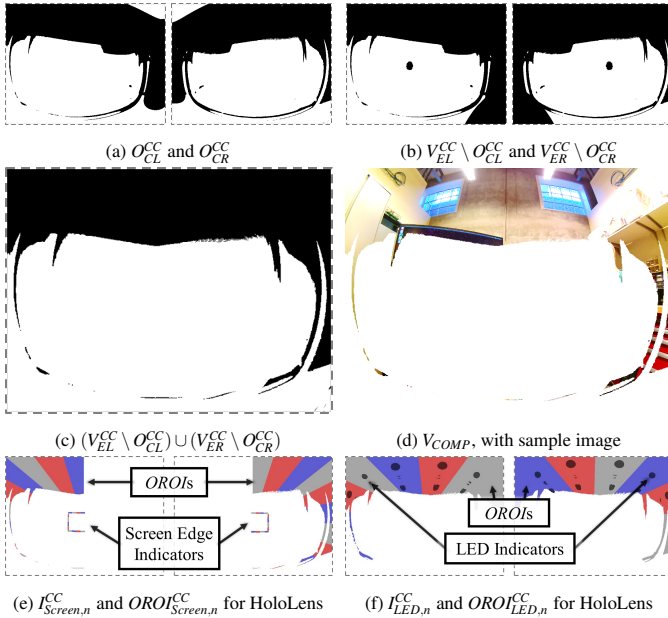
(a) O_{CL}^{CC} and O_{CR}^{CC}

(b) $V_{EL}^{CC} \setminus O_{CL}^{CC}$ and $V_{ER}^{CC} \setminus O_{CR}^{CC}$

(c) $(V_{EL}^{CC} \setminus O_{CL}^{CC}) \cup (V_{ER}^{CC} \setminus O_{CR}^{CC})$

(d) V_{COMP}, with sample image

(e) $I_{Screen,n}^{CC}$ and $OROI_{Screen,n}^{CC}$ for HoloLens

(f) $I_{LED,n}^{CC}$ and $OROI_{LED,n}^{CC}$ for HoloLens

Fig. 10: Offline stage results for Microsoft HoloLens

(a) O_{CL}^{CC}, O_{CR}^{CC}

(b) V_{COMP} with sample image

(c) $I_{Screen,n}^{CC}$ and $OROI_{Screen,n}^{CC}$ for ODG R-9

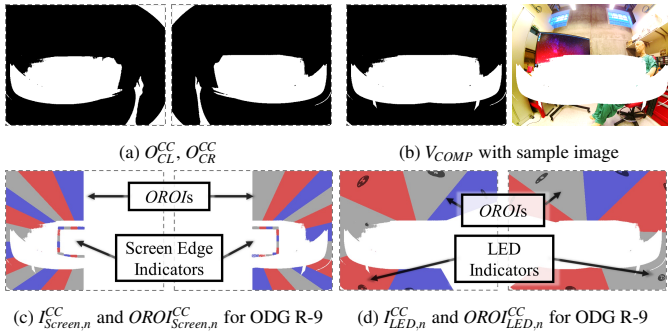(d) $I_{LED,n}^{CC}$ and $OROI_{LED,n}^{CC}$ for ODG R-9

Fig. 11: Offline stage results for ODG R-9

camera is controlled by a Raspberry Pi Model 3b. The cameras CL and CR are mounted on a 3D-printed mount with IPD of $64\,mm$. The Raspberry Pi that controls CC (Raspberry Pi 1) is also in charge of controlling the LED strip: Adafruit Mini Skinny NeoPixel strip of $60\,unit/m$. Three wires are required for the LEDs: control signal, GND and $+3.3V$. A desktop computer with an Intel Core $i5$-$2500@3.3GHz$ CPU and $7.7GB$ memory is responsible for the computation. The background monitor (see Fig. 1) is a Samsung DE55A 55", with a resolution of 1920×1080.

In terms of software, the Raspberry Pis run Ubuntu Mate 16.04, and the PC runs Ubuntu 16.04 LTS. The camera videos ($640 \times 480, 15\,fps$) are encoded and streamed from Raspberry Pi to PC using raspivid and netcat. The exposure parameters of the cameras are controlled manually. The PC accesses the video stream via netcat, decodes the video via libx264 and FFmpeg. We use a Python library rpi_ws281x to interface the LED strip. Programs for Sec. 3.2.1 and Sec. 3.2.2 are implemented based on OpenCV 3.4. The cameras are calibrated using the OpenCV Fisheye model. The calibration results show that they have a horizontal FOV of $142.74°$ and a vertical FOV of $131.60°$.

4.2 Experimental setup for online stage

Fig. 12b shows the experimental setup with HoloLens for the online stage. Raspberry Pi 1 is still used to drive the center camera CC and LEDs. It streams the frames to the PC. The implementation of dense optical flow is from an OpenCV 3.4 extra module. In the method with screen edge indicators, the PC sends serialized brightness values to the OST-HMD via TCP/IP. The application on the OST-HMD includes a TCP client to receive the brightness values. The applications on both OST-HMDs are implemented with Unity 5.6.0f3. In the case where LED indicators are used, the PC sends the packet to Raspberry Pi 1

which then sets the control signal to its IO pin.

4.3 Microsoft HoloLens vs. ODG R-9

The setup for ODG R-9 is slightly different from Microsoft HoloLens due to the different hardware properties. Some key features and differences in their setup are listed in Tab. 3. Fig. 12c shows the LED setup for both devices. The results for offline calibration for Microsoft HoloLens and ODG R-9 are shown in Fig. 10 and Fig. 11.

4.4 System Performance

We measured the performance of our system. The end-to-end video streaming latency from Raspberry Pi to PC is $127\,ms$ and the per-frame computation on the PC takes $73.76\,ms$. The average framerate of the compensation loop is $13.16\,fps$. When screen edge indicators are used, the Unity application on HoloLens runs at $32.76\,fps$ and the application for ODG R-9 runs at $52.14\,fps$.

5 EVALUATION

We conducted an objective experiment and a subjective pilot user study to evaluate the systems. In addition, we evaluated the error introduced by assumption $A1$ (the co-location of eyes and center camera CC), and the accuracy of the segmentation algorithm of Sec. 3.2.2.

5.1 Objective Evaluation

For objective evaluation, we re-use the setup from Sec. 3.2.2 (Fig. 12a) and simulate the user's perspective with the cameras CL and CR. We evaluate four scenarios: HS (HoloLens with screen edge indicators), HL (HoloLens with LED indicators), OS (ODG with screen edge indicators) and OL (ODG with LED indicators). To present controlled motion for our objective evaluation, we display a target moving in a rectangular pattern on a monitor placed in front of the setup. Overall we observe 36 targets (6 different monitor locations × 6 targets). At each location we display a checkerboard on the monitor before the experiment begins to compute its pose relative to the HMD. We show the poses and trajectories of the 36 targets in Fig. 13. The targets for HoloLens and ODG R-9 are not the same.

For each target, we conducted the following steps:
- Disable both indicators, record video with CC, CL, CR. We denote them as VD_0, VD_1 and VD_2.
- Enable screen edge indicators, record video with CL, CR. We denote them as VD_3 and VD_4.
- Enable LED indicators, record video with CL, CR. We denote them as VD_5 and VD_6.

For each of the videos, the first image frame is subtracted from all subsequent frames in order to filter out static global luminance. Low-brightness pixels (< 20, max: 255) are thresholded to 0 for noise removal. We use VD_0 as ground truth, and fuse VD_1 and VD_2 into VD_{Eye} by alpha blending. Similarly, VD_3 and VD_4 are blended into VD_{Screen}, and VD_5 and VD_6 are blended into VD_{LED}. Then, we compute two values for VD_0, VD_{Eye}, VD_{Screen} and VD_{LED}: the average brightness value of pixels of all frames (BR: a floating point number), and the centroid of brightness of all frames (p_{BR}: in pixel coordinates). Therefore, each video is represented by BR and p_{BR}.

We threshold the brightness value of VD_{Eye} to classify the visibility of the target into: visible, partially visible, and invisible. Then, we determine if the indicators are activated by comparing the brightness values BR of VD_{Screen} and VD_{LED} with VD_{Eye}. If the indicators are

Tab. 3: Setup comparison for HoloLens and ODG R-9.
($\star$ 12 of 14 LED indicators are used. The other 2 LED indicators are not in the visual field of the cameras, so we are not able to determine their locations and associated $OROI$.)

Comparison	HoloLens	ODG R-9
Display resolution	$1268 \times 720, \times 2$	$1920 \times 1080, \times 2$
Display refresh rate	$60\,fps$	$60\,fps$
See-through transparency	High	Low
Number of LED indicators N_L	12	$14\star$
Number of screen edge indicators N_S	24	24
Width of edge pixel w	50 pixels	100 pixels

(a) Microsoft HoloLens setup for offline stage   (b) Microsoft HoloLens setup for online stage   (c) LEDs attachment
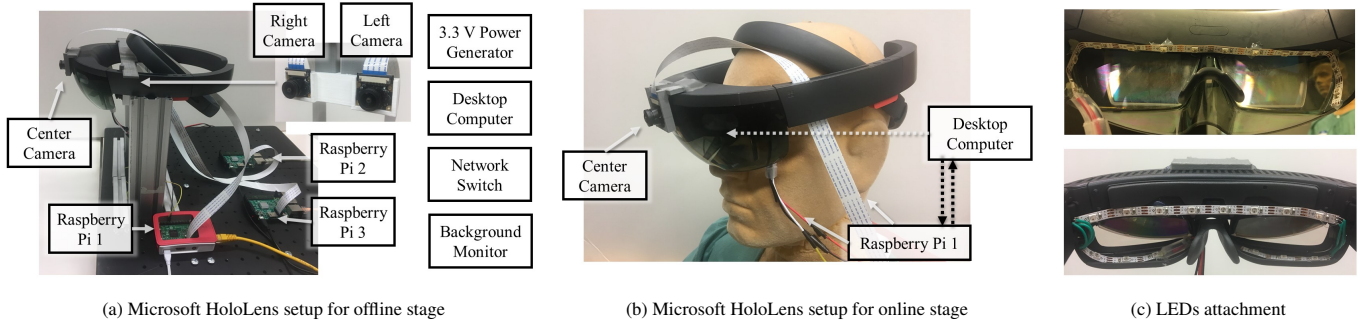
Fig. 12: Experimental setup for a) offline stage, b) online stage, and c) LED attachments on HoloLens (top) and ODG R-9 (bottom).
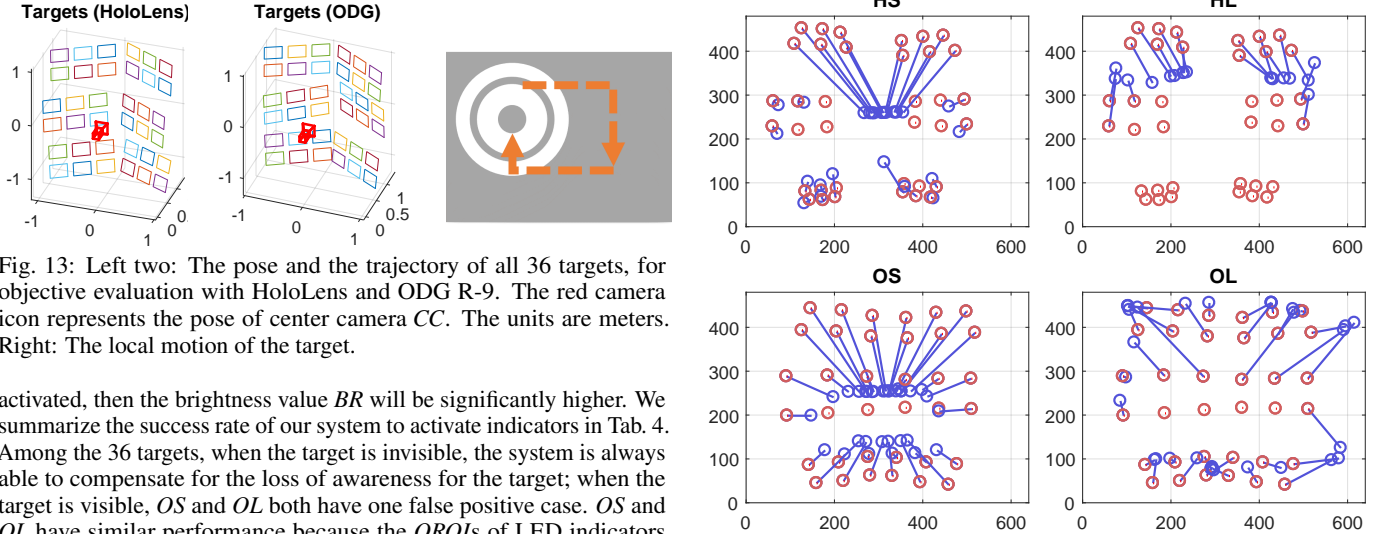

Fig. 13: Left two: The pose and the trajectory of all 36 targets, for objective evaluation with HoloLens and ODG R-9. The red camera icon represents the pose of center camera CC. The units are meters. Right: The local motion of the target.
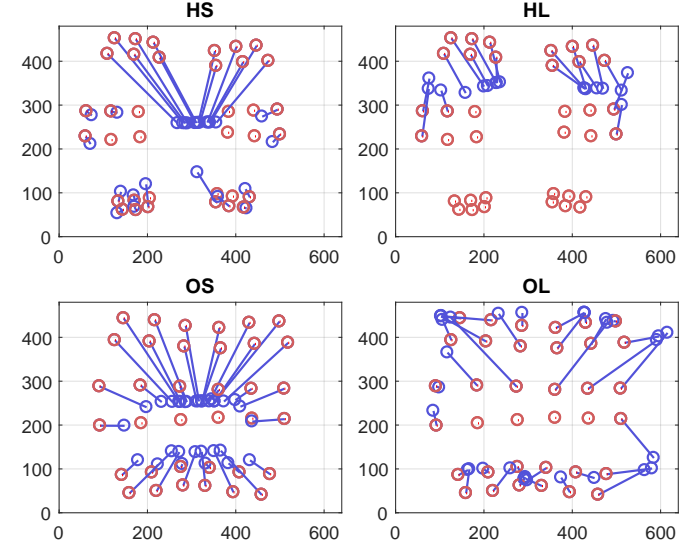

Fig. 14: Centroid of brightness p_{BR} for all targets and the four scenarios. Red circles indicate the ground truth obtained with the center camera CC, and blue circles indicate the perceived brightness location on the image frame when the indicators are activated. (Better seen in color)

activated, then the brightness value BR will be significantly higher. We summarize the success rate of our system to activate indicators in Tab. 4. Among the 36 targets, when the target is invisible, the system is always able to compensate for the loss of awareness for the target; when the target is visible, OS and OL both have one false positive case. OS and OL have similar performance because the $OROI$s of LED indicators for ODG R-9 also completely surround the total area-to-compensate ($\bigcup_{n=1}^{N_S} V_{Screen,n}^{CC} = \bigcup_{n=1}^{N_L} V_{LED,n}^{CC} = V_{COMP}$). However, the HL fails to compensate in a few cases when the target is partially visible. This is because the union of all $OROI$s for LED indicators on HoloLens do not span the total V_{COMP}, e.g., the thin occluded area at the bottom.

Next, we compare the indicated direction of the target with the ground truth. Fig. 14 shows the calculated centroid of brightness p_{BR} for the four scenarios and 36 targets. The red circles represent the p_{BR} calculated from VD_0, as ground truth. The blue circles represent the p_{BR} calculated from VD_{Screen} and VD_{LED} when the indicators are activated. We consider these to be the perceived location of the target, which are the projection of the indicated direction on the image frame. In HS and OS, the perceived locations are closer to the image center, but with higher angular precision because there are more indicators on the screen edge than LEDs. For screen edge indicators, there are also cases where the blue circles are close to the red dot; this is due to the fact that when the object is partially visible, the p_{BR} is a weighted average of both the visible part of the target and the activated indicators.

We analyze the 2D angular error between the red circles and the blue circles. The mean and standard deviation of the 2D angular errors are shown in Fig. 15a. The error is $4.87° \pm 5.62°$ for HS, $17.80° \pm 9.63°$ for HL, $2.99° \pm 2.34°$ for OS, and $8.07° \pm 6.08°$ for OL. With an independent two-sample t-test, we find that the 2D angular error of HS is significantly smaller than HL ($p = 1.76 \times 10^{-6}$), and that of OS is significantly smaller than OL ($p = 3.76 \times 10^{-5}$).

The 3-dimensional angular distance ($\Delta \theta$) is more related to the user's real perception of the target. We compute them by back-projecting the pixel locations into 3D vectors using the camera intrinsic matrix and distortion parameters:

$$\vec{L}_{\{1,2\}} = K_{CC}^{-1} D_{CC}^{-1} \left(P_{BR\{1,2\}} \right)$$

$$\Delta \theta = \left| cos^{-1} \left(\frac{\vec{L}_1 \cdot \vec{L}_2}{\|\vec{L}_1\| \cdot \|\vec{L}_2\|} \right) \right| \quad (11)$$

The mean and standard deviation of 3D angular error are shown in Fig. 15b. The error is $22.16° \pm 19.17°$ for HS, $18.60° \pm 4.97°$ for HL, $23.80° \pm 14.56°$ for OS, and $16.39° \pm 12.42°$ for OL. With independent two-sample t-test, we find that there is no significant difference between HS and HL ($p = 0.46$), and interestingly, the 3D angular error of OS is even statistically significantly larger than OL ($p = 3.23 \times 10^{-2}$). When analyzed in 3D space, the angular error of screen edge indicators becomes much larger due to the large gap between the indicators and the occluded area. There is an increase of standard deviation for HS and OS. Taking HS as an example, for occluded targets (the upper 12), P_{BR} is very close to the central vision and therefore introduces large error when analyzed in 3D. However, for partially occluded targets (e.g., the lower 12), P_{BR} is contributed by both the target itself and the indicators, and thus appears closer to the target. The standard deviation increases drastically in consequence.

5.2 Pilot user study

We conducted a pilot study of the HS and HL scenarios, with three experienced OST-HMD users, to gain more insight into the performance of our indication methods in an actual application scenario and to get feedback on the acceptability of the two indication methods. All the

Tab. 4: Success rate for HS, HL, OS, and OL.

Total (36)	HS	HL		Total (36)	OS	OL
Visible (10)	0	0		Visible (5)	1	1
Partial (14)	14	5		Partial (13)	13	13
Invisible (12)	12	12		Invisible (18)	18	18

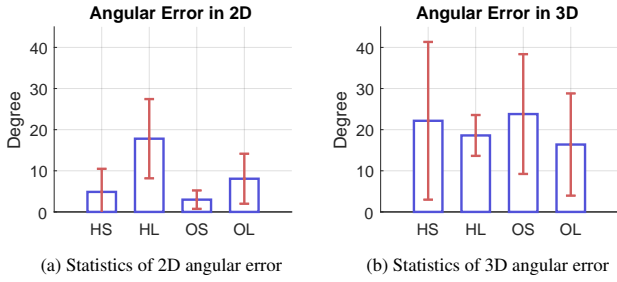(a) Statistics of 2D angular error   (b) Statistics of 3D angular error

Fig. 15: The mean and standard deviation of angular error in 2D and 3D for the four scenarios: *HS*, *HL*, *OS* and *OL*. 2D angle is also the azimuth angle on the image plane, while the 3D angle is calculated by back-projection in Eq. 11.

users reported that they had normal vision. The synthesized normal visual field (Fig. 2) was used for all users.

We adopt the popular experiment paradigm of attention research: the participants are required to attend to a primary signal and a secondary signal simultaneously [1]. We implement a reading application on HoloLens where random sentences are displayed at the screen center. At the same time, a background monitor is placed in front of the user, randomly showing words in random locations. The monitor is placed quite close to the user ($\sim 0.5m$) in order to cover a large FOV, and placed higher than the user because we already know that the majority of the occlusion on HoloLens is in the upper area. We explain the functionality of the system to the user, e.g., the flash at the screen edge and LEDs indicates the direction of a word on the background monitor. During the study, the user reads the sentences on HoloLens and also reads the words that are shown on the background monitor. We measure the success rate of the user noticing words on the background monitor. When the user is notified by our system but finds that there is no word on the background monitor, we record it as a *False Positive*.

The success rate and the number of false positive are presented in Tab. 5. Only 2 targets are missed for a total of 198 targets, and 5 false positives occurred. We collected feedback about their experience with the two systems and their preference. User #2 preferred *HL* and he mentioned that with LEDs at the periphery, he followed the indication intuitively and successfully found the words, however with *HS*, he had to mentally calculate a direction that he should move towards. On the other hand, user #1 preferred *HS* because he thought that the LED indicators were too bright and alarming, which might be appropriate for some cases but not for the reading task. User #3 reported no preference between the two methods. User #1 also mentioned that the indicators were still activated when he was turning his head back, but he was able to understand this behavior and was adapted to it. In our current implementation, the direction of optical flow is not distinguished between moving towards the user and moving away from the user. Therefore, the system treats both kinds of motion identically. None of the users reported uncomfortable situations.

In the pilot user study, we tested whether our systems (*HS* and *HL*) are able to correctly restore the user's awareness to look for changes (motions) in the occluded visual field. The results show that the success rate of our systems is high, despite a few false positive cases.

5.3 Co-Location Assumption

In Sec. 3.2, we assume that both eyes and the center camera *CC* are co-located (*A1*). This is equivalent to assuming that the scene is at infinity. With this assumption, we are able to project visual field of *EL*,

Tab. 5: Results of pilot user study for *HS* and *HL*. *FP* refers to the number of *False Positives* that occurred during the user study.

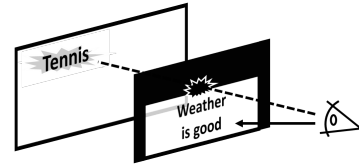|	HS		HL	
User	Success Rate	FP	Success Rate	FP
#1	92.6% (25/27)	1	100% (35/35)	0
#2	100% (26/26)	1	100% (31/31)	0
#3	100% (40/40)	2	100% (39/39)	1

Fig. 16: Illustration of the pilot user study on HoloLens: the user focuses on a reading task, but our system enables them to notice words on the background monitor that would otherwise be occluded by HoloLens.
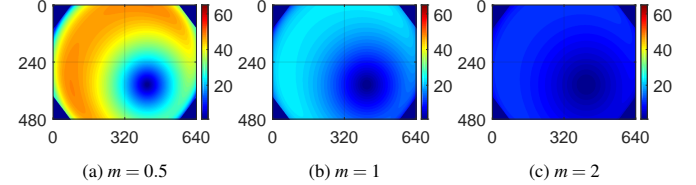

(a) $m = 0.5$ (b) $m = 1$ (c) $m = 2$

Fig. 17: Evaluation of the error introduced by the co-location assumption *A1*. Pixel error is plotted with different pixel depths: 0.5, 1 or 2 meters. (Better seen in color)

ER, *CL*, *CR* to the visual field of *CC* without estimating the depth of every pixel. Next we evaluate the error introduced by this assumption.

For camera *CL* and pixel $(i_L, j_L) \in V_{CL}$, with assumption *A1*, the pixel is projected to $(i_C, j_C) \in V_{CC}$ using Eq. 5. To drop assumption *A1*, we denote the location of *CC* as $\vec{d}(x_0, y_0, z_0)$ in the coordinate system of *CL*. If the 3D location of pixel (i_L, j_L) is at *m* meters away from *CL*, then the pixel is projected to (i'_C, j'_C) as follows:

$$\vec{L} = K_{CL}^{-1} D_{CL}^{-1}(i_L, j_L)$$
$$\vec{P} = m \cdot \vec{L}/\|\vec{L}\| \tag{12}$$
$$(i'_C, j'_C) = D_{CC}\left(K_{CC}(\vec{P} - \vec{d})\right)$$

We use our camera calibration results of the wide-angle cameras, take $d = (0.032m, -0.030m, 0.080m)$ which is approximately the displacement between *CL* and *CC* in the HoloLens setup, and compare the introduced error for different distances $m \in \{0.5m, 1m, 2m, 4m\}$. We visualize the pixel distance error $\|(i_L - i'_L, j_L - j'_L)\|$ in Fig. 17. The corners of the image frame are excluded due to inconsistent behavior of the *cv::fisheye::distortPoints()* function in OpenCV. The means and standard deviations of the pixel errors introduced by the co-location assumption are 36.05 ± 10.71 pixels, 17.51 ± 5.48 pixels, 8.61 ± 2.77 pixels, and 4.26 ± 1.39 pixels for pixel depths at $0.5m$, $1m$, $2m$, and $4m$, respectively. As expected, the error introduced by assumption *A1* decreases when the pixel depth increases.

The displacement between the virtual eye and real eye causes perceptual issues with a VST-HMD [7]. In our setup with an OST-HMD, the displacement does not directly affect the major part of the user's vision. However, the displacement still introduces errors to the desired direction of indication. To completely eliminate assumption *A1*, it is necessary to determine the position of the eyes, e.g., by [42], and the pixel depth in real time, either by RGBD sensors or by software reconstruction methods like [15].

5.4 Segmentation with Responsiveness Function

To evaluate our segmentation algorithm, we manually segment the occluded area by HoloLens as ground truth. Fig. 10a shows the segmentation results of our algorithm, while Fig. 18a shows the corresponding ground truth. Note that in the manual segmentation, all relative scotoma are removed, e.g., the sharp angles on the glass surface. Our algorithm is dependent on the threshold *T* of the responsiveness function $Resp(\cdot)$ of all pixels. The area that should be part of the occlusion but is not correctly segmented is *False Negative*. The area that should not be part of the occlusion but is wrongly segmented is *False Positive*. Fig. 18b and Fig. 18c show the percentage of false positive and false negative for the segmentation of HoloLens. The minimum combined error rate is 2.99% for O_{CL}^{CC} and 5.66% for O_{CR}^{CC}. A threshold of $240 \leq T \leq 4170$
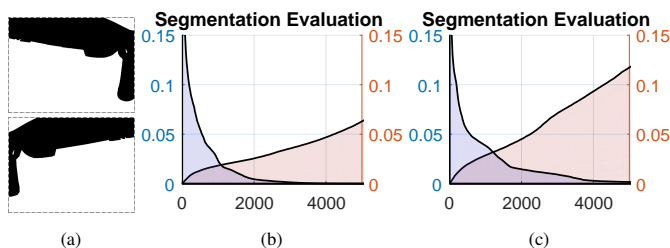
Fig. 18: (a): Manual segmentation of O_{CL}^{CC} and O_{CR}^{CC} (of Fig. 10a) for HoloLens, used as ground truth. (b)(c): The evaluation results of (a). Horizontal axis is the threshold T for the responsiveness function. Left vertical axis: false positive rate. Right vertical axis: false negative rate.

will be able to generate segmentation results with a combined error rate of less than 10% for both images.

The performance of our system is highly dependent on the segmentation result: O_{CL}^{CC} and O_{CR}^{CC}. With a high false positive rate, the system will indicate the motion that is already naturally seen by the user, which results in a more alarming or more disturbing system, depending on the application. A segmentation with a high false negative rate will fail to indicate activity that happens in the wrongly segmented area. Alternative algorithms, e.g. morphological filter [61] and watershed transformation [48], can be applied for segmentation as well.

6 DISCUSSION AND LIMITATIONS

6.1 Screen Edge and LED Indicators

From the evaluations of our prototype, the screen edge indicator and LED indicator both yield a good success rate, but show distinct characteristics. The number of screen edge indicators is higher than the number of LEDs for HoloLens, leading to on average fewer pixels in *OROI*. As a result, Eq. 10 yields a higher value and therefore makes the screen edge indicators more 'sensitive' than LED indicators.This is supported by the higher number of false positive results (14 versus 5 in objective evaluation, 4 versus 1 in user study). A more 'sensitive' indication will benefit applications that involve rare but safety-critical events, e.g., collaborative manufacturing with a robot. As mentioned by user #2, the LEDs placed at the periphery provide intuitive indications on HoloLens. On the contrary, limited by the FOV of HoloLens, the screen edge indicators are much closer to the central vision, and thus introduce a larger 3D angular error. The locations of both indicators are highly dependent on the FOV and structure of the OST-HMD.

Meta 2, a new OST-HMD that features a $90°$ diagonal field-of-view display recently became available on the market. However, its panel still occludes the user's view in the upper area. For Meta 2, the additional LED indicators may be less necessary since the screen edge indicators are already located in the periphery.

When designing our indicators, our goal was to display as little content as possible onto the OST-HMD. However, contours displayed in the user's view, as was explored in [60], could also restore the user's awareness of their surroundings. In the future, it is thus important to perform a formal study that compares the different visualization techniques and their effect on the user's awareness.

In conclusion, the choice between the indicating methods should be decided considering the specific application and in the context of a specific OST-HMD. In this paper, we implemented and evaluated two indication methods for out-of-view events. However, it is necessary to conduct a more thorough user study that compares them with other methods, e.g., [60], in different environments, HMDs, and applications.

6.2 Expansion of the Awareness

In fact, if the camera CC has a larger visual field than the human binocular visual field $V_{EL}^{CC} \cup V_{ER}^{CC} \subset V_{CC}$, it is possible to expand the normal human visual experience instead of only compensating for the occlusion: $V_{EXPAND} = V_{CC} \setminus \left[\left(V_{EL}^{CC} \setminus O_{CL}^{CC} \right) \cup \left(V_{ER}^{CC} \setminus O_{CR}^{CC} \right) \right]$.

Note that when the center camera CC has a FOV larger than $180°$, the projection matrix is not able to cover the entire FOV. In this case, it is more suitable to represent the visual fields in the polar coordinate

system where ρ could be larger than $90°$. When notifying user about events from behind, similar to [40], it is worth studying the effects of spatial audio cues, as users react faster to audio signals [28], or combining the indication techniques discussed in this paper.

6.3 Personalized Visual Field

The human visual system is complex and a person's visual field is dependent on the age and health of the eyes. The measurement of the human visual field also depends on the size, color and luminance of the target [50]. The nominal human visual fields we use in the experiments are synthesized with typical features of normal human visual fields. A mismatch between the nominal visual field and actual visual field can cause false positives or false negatives. If a measured visual field of the user is available, it can be seamlessly incorporated into our workflow by substituting it for the nominal V_{EL} and V_{ER}.

6.4 Optimization of the Implementation

In our current prototype, the latency for video streaming and computation is $127\,ms$ and $73.76\,ms$, respectively, which is not ideal for real-life applications, especially in potentially dangerous situations. The streaming latency can be reduced by using hardware accelerated codecs, or even be eliminated by migrating the computation to the Raspberry Pi or HMD onboard processor. The computation can be accelerated with the support of a GPU. A camera with higher shutter speed can also be used to reduce the overall latency.

The Raspberry Pi and wirelessly connected PC limit the mobility of the user because both HoloLens and ODG R-9 are untethered devices. This design choice is partly due to the lack of custom hardware interface of current OST-HMDs. Ideally, if the OST-HMD offers a wide-angle camera, a wide field-of-view for display (or periphery display for off-screen visualization), and sufficient computational power, the restoration of awareness can be enabled purely with software.

In terms of the algorithm for online processing, we use optical flow to parse the information in the *OROI*s. More context understanding can be built into our framework by substituting the online image processing algorithm, such as egocentric object tracking [2] or SLAM++ [51], depending on the specific requirement for applications.

6.5 Vision for the Future

The ultimate OST-HMD will be similar to normal glasses that do not cause any occlusion. At that time, our method for restoring the awareness will hopefully be retired, and possibly find usage in heads-up displays integrated into cars to mitigate the occlusion caused by the A-pillar. However, until then the occlusion caused by OST-HMDs will remain an issue. Our vision is that, in the near future, OST-HMD manufacturers will use more see-through materials for the hardware structure or even embed micro displays where the occlusion is unavoidable. We also want to raise the awareness to the potential danger due to the occluded visual field and hope designers will consider it when creating AR applications.

7 CONCLUSION

In this paper, we study the issue of occlusion caused by OST-HMDs. The occlusion causes partial loss of peripheral vision, however, peripheral vision is essential for mobility and safety. We calculate the area that is occluded by the OST-HMD in an offline calibration stage and match it with the image captured by a wide-angle scene camera. We detect motion in the occluded area through an optical flow algorithm. We propose two methods to indicate the detected motion to the user to restore his/her awareness of the surroundings. We implement our methods on two state-of-art OST-HMDs, and evaluate them with both an objective experiment and a pilot user study.

The results show that both methods are viable for compensating the loss of safety-critical information in the occluded areas. However, participants reported different preferences towards the two indicating methods in our subjective evaluation. We plan to conduct a more thorough user study in the future. It is also an interesting future direction to investigate the effects of a personalized visual field, and context understanding algorithms with our proposed methods. We also plan

to investigate how other indication methods, e.g., [60], could assist expanding the user's awareness of the surroundings.

REFERENCES

[1] R. W. Adams. *Peripheral vision and visual attention*. PhD thesis, Iowa State University, 1971.

[2] S. Alletto, G. Serra, and R. Cucchiara. Egocentric Object Tracking: An Odometry-Based Solution. In *Proceedings of the International Conference on Image Analysis and Processing*, pp. 687–696, 2015.

[3] J. Ardouin, A. Lécuyer, M. Marchal, C. Riant, and E. Marchand. FlyVIZ: A Novel Display Device to Provide Humans with 360 Vision by Coupling Catadioptric Camera with HMD. In *Proceedings of the ACM symposium on Virtual Reality Software and Technology*, pp. 41–44, 2012.

[4] E. Azimi, A. Winkler, E. Tucker, L. Qian, M. Sharma, J. Doswell, N. Navab, and P. Kazanzides. Evaluation of optical see-through head-mounted displays in training for critical care and trauma. In *Proceedings of the IEEE Virtual Reality*, pp. 1–2, 2018.

[5] P. Baudisch and R. Rosenholtz. Halo: A Technique for Visualizing Off-Screen Objects. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 481–488, 2003.

[6] F. Biocca, A. Tang, C. Owen, and F. Xiao. Attention Funnel: Omnidirectional 3D Cursor for Mobile Augmented Reality Platforms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 1115–1122, 2006.

[7] F. A. Biocca and J. P. Rolland. Virtual Eyes Can Rearrange Your Body: Adaptation to Visual Displacement in See-Through, Head-Mounted Displays. *Presence*, 7(3):262–277, 1998.

[8] S. Burigat and L. Chittaro. Navigation in 3D Virtual Environments: Effects of User Experience and Location-Pointing Navigation Aids. *International Journal of Human-Computer Studies*, 65(11):945–958, 2007.

[9] S. Burigat, L. Chittaro, and S. Gabrielli. Visualizing Locations of Off-Screen Objects on Mobile Devices: A Comparative Evaluation of Three Approaches. In *Proceedings of the Conference on Human-Computer Interaction with Mobile Devices and Services*, pp. 239–246, 2006.

[10] C. B. Chen. Wide Field of View, Wide Spectral Band Off-Axis Helmet-Mounted Display Optical Design. In *Proceedings of the International Optical Design Conference*, vol. 4832, pp. 61–67, 2002.

[11] E. Costanza, S. A. Inverso, E. Pavlov, R. Allen, and P. Maes. Eye-q: Eye-glass peripheral display for subtle intimate notifications. In *Proceedings of the 8th conference on Human-computer interaction with mobile devices and services*, pp. 211–218. ACM, 2006.

[12] C. A. Curcio, K. R. Sloan, R. E. Kalina, and A. E. Hendrickson. Human Photoreceptor Topography. *Journal of Comparative Neurology*, 292(4):497–523, 1990.

[13] M. Dalle Mura, G. Dini, and F. Failli. An Integrated Environment Based on Augmented Reality and Sensing Device for Manual Assembly Workstations. *Procedia CIRP*, 41:340–345, 2016.

[14] F. Danieau, A. Guillo, and R. Doré. Attention guidance for immersive video content in head-mounted displays. In *Virtual Reality (VR), 2017 IEEE*, pp. 205–206. IEEE, 2017.

[15] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse. Monoslam: Real-time single camera slam. *IEEE transactions on pattern analysis and machine intelligence*, 29(6):1052–1067, 2007.

[16] E. S. De Guzman, M. Yau, A. Gagliano, A. Park, and A. K. Dey. Exploring the design and use of peripheral displays of awareness information. In *CHI'04 extended abstracts on Human factors in computing systems*, pp. 1247–1250. ACM, 2004.

[17] K. Fan, J. Huber, S. Nanayakkara, and M. Inami. SpiderVision: Extending the Human Field of View for Augmented Awareness. In *Proceedings of the Augmented Human International Conference*, pp. 49:1–49:8, 2014.

[18] G. Farnebäck. Two-Frame Motion Estimation Based on Polynomial Expansion. In *Proceedings of the Scandinavian Conference on Image Analysis*, pp. 363–370, 2003.

[19] V. Ferrer, Y. Yang, A. Perdomo, and J. Quarles. Consider Your Clutter: Perception of Virtual Object Motion in AR. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, pp. 1–6, Oct 2013.

[20] R. Gregory and P. Cavanagh. The Blind Spot. *Scholarpedia*, 6(10):9618, 2011.

[21] U. Gruenefeld, A. E. Ali, W. Heuten, and S. Boll. Visualizing Out-of-View Objects in Head-Mounted Augmented Reality. In *Proceedings of the International Conference on Human-Computer Interaction with Mobile Devices and Services*, pp. 81:1–81:7, 2017.

[22] U. Gruenefeld, D. Ennenga, A. E. Ali, W. Heuten, and S. Boll. EyeSee360: Designing a Visualization Technique for Out-of-View Objects in Head-Mounted Augmented Reality. In *Proceedings of the ACM Symposium on Spatial User Interaction*, pp. 109–118, 2017.

[23] U. Gruenefeld, D. Hsiao, W. Heuten, and S. Boll. EyeSee: Beyond Reality with Microsoft HoloLens. In *Proceedings of the ACM Symposium on Spatial User Interaction*, pp. 148–148, 2017.

[24] U. Gruenefeld, T. C. Stratmann, W. Heuten, and S. Boll. PeriMR: A Prototyping Tool for Head-Mounted Peripheral Light Displays in Mixed Reality. In *Proceedings of the International Conference on Human-Computer Interaction with Mobile Devices and Services*, pp. 51:1–51:6, 2017.

[25] S. Gustafson, P. Baudisch, C. Gutwin, and P. Irani. Wedge: Clutter-Free Visualization of Off-screen Locations. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 787–796, 2008.

[26] B. K. Horn and B. G. Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981.

[27] Y. Itoh and G. Klinker. Light-Field Correction for spatial Calibration of Optical See-Through Head-Mounted Displays. *IEEE Transactions on Visualization and Computer Graphics*, 21(4):471–480, 2015.

[28] A. Jain, R. Bansal, A. Kumar, and K. Singh. A Comparative Study of Visual and Auditory Reaction Times on the Basis of Gender and Physical Activity Levels of Medical First Year Students. *International Journal of Applied and Basic Medical Research*, 5(2):124, 2015.

[29] C. A. Johnson and J. L. Keltner. Incidence of Visual Field Loss in 20,000 Eyes and its Relationship to Driving Performance. *Archives of Ophthalmology*, 101(3):371–375, 1983.

[30] B. R. Jones, H. Benko, E. Ofek, and A. D. Wilson. Illumiroom: peripheral projected illusions for interactive experiences. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 869–878. ACM, 2013.

[31] M. Kassner, W. Patera, and A. Bulling. Pupil: an open source platform for pervasive eye tracking and mobile gaze-based interaction. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing: Adjunct publication*, pp. 1151–1160. ACM, 2014.

[32] M. Klemm, F. Seebacher, and H. Hoppe. Non-parametric Camera-Based Calibration of Optical See-Through Glasses for AR Applications. In *Proceedings of the International Conference on Cyberworlds*, pp. 33–40, 2016.

[33] F. L. Kooi and M. Mosch. Peripheral motion displays: tapping the potential of the visual periphery. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, pp. 1604–1608. SAGE Publications Sage CA: Los Angeles, CA, 2006.

[34] A. M. Larson and L. C. Loschky. The Contributions of Central Versus Peripheral Vision to Scene Gist Recognition. *Journal of Vision*, 9(10):6:1–6:16, 2009.

[35] J. E. Lovie-Kitchin, J. C. Mainstone, J. Robinson, and B. Brown. What Areas of of the visual Field are Important for Mobility in Low Vision Patients. *Clinical Vision Sciences*, 5(3):249–263, 1990.

[36] A. Lucero and A. Vetek. Notifeye: using interactive glasses to deal with notifications while walking in public. In *Proceedings of the 11th Conference on Advances in Computer Entertainment Technology*, p. 17. ACM, 2014.

[37] K. Luyten, D. Degraen, G. Rovelo Ruiz, S. Coppers, and D. Vanacken. Hidden in plain sight: an exploration of a visual language for near-eye out-of-focus displays in the peripheral view. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 487–497. ACM, 2016.

[38] T. Miyaki and J. Rekimoto. LiDARMAN: Reprogramming Reality with Egocentric Laser Depth Scanning. In *ACM SIGGRAPH 2016 Emerging Technologies*, pp. 15:1–15:2, 2016.

[39] S. Niederhauser and D. S. Mojon. Normal Isopter Position in the Peripheral Visual Field in Goldmann Kinetic Perimetry. *Ophthalmologica*, 216(6):406–408, 2002.

[40] E. Niforatos, A. Fedosov, I. Elhart, and M. Langheinrich. Augmenting skiers' peripheral perception. In *Proceedings of the 2017 ACM International Symposium on Wearable Computers*, pp. 114–121. ACM, 2017.

[41] J. Orlosky, Q. Wu, K. Kiyokawa, H. Takemura, and C. Nitschke. Fisheye Vision: Peripheral Spatial Compression for Improved Field of View in Head Mounted Displays. In *Proceedings of the ACM Symposium on Spatial User Interaction*, pp. 54–61, 2014.

[42] A. Plopski, C. Nitschke, K. Kiyokawa, D. Schmalstieg, and H. Takemura. Hybrid Eye Tracking: Combining Iris Contour and Corneal Imaging. In *Proceedings of the International Conference on Artificial Reality and Telexistence and Eurographics Symposium on Virtual Environments*, pp. 183–190, October 2015.

[43] B. Poppinga, N. Henze, J. Fortmann, W. Heuten, and S. Boll. Ambiglasses-information in the periphery of the visual field. In *Mensch & Computer*, pp. 153–162, 2012.

[44] L. Qian, A. Barthel, A. Johnson, G. Osgood, P. Kazanzides, N. Navab, and B. Fuerst. Comparison of Optical See-Through Head-Mounted Displays for Surgical Interventions with Object-Anchored 2D-Display. *International Journal of Computer Assisted Radiology and Surgery*, 12(6):901–910, 2017.

[45] I. Rakkolainen, R. Raisamo, M. Turk, and T. Höllerer. Field-of-view Extension for VR Viewers. In *Proceedings of the International Academic Mindtrek Conference*, pp. 227–230, 2017.

[46] I. Rakkolainen, M. Turk, and T. Höllerer. A Superwide-FOV Optical Design for Head-Mounted Displays. In *Proceedings of the International Conference on Artificial Reality and Telexistence and the Eurographics Symposium on Virtual Environments*, pp. 45–48, 2016.

[47] P. Renner and T. Pfeiffer. Attention guiding techniques using peripheral vision and eye tracking for feedback in augmented-reality-based assistance systems. In *3D User Interfaces (3DUI), 2017 IEEE Symposium on*, pp. 186–194. IEEE, 2017.

[48] J. B. Roerdink and A. Meijster. The watershed transform: Definitions, algorithms and parallelization strategies. *Fundamenta informaticae*, 41(1, 2):187–228, 2000.

[49] J. P. Rolland, R. L. Holloway, and H. Fuchs. Comparison of Optical and Video See-Through, Head-Mounted Displays. In *Telemanipulator and Telepresence Technologies*, vol. 2351, pp. 293–308, 1995.

[50] R. Rosenholtz. Capabilities and Limitations of Peripheral Vision. *Annual Review of Vision Science*, 2(1):437–457, 2016.

[51] R. F. Salas-Moreno, R. A. Newcombe, H. Strasdat, P. H. J. Kelly, and A. J. Davison. Slam++: Simultaneous Localisation and Mapping at the Level of Objects. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1352–1359, 2013.

[52] T. M. Schmidt, S.-K. Chen, and S. Hattar. Intrinsically Photosensitive Retinal Ganglion Cells: Many Subtypes, Diverse Functions. *Trends in Neurosciences*, 34(11):572–580, 2011.

[53] R. H. Spector. Visual Fields. In H. K. Walker, W. D. Hall, and J. W. Hurst, eds., *Clinical Methods: The History, Physical, and Laboratory Examinations*, chap. 116, pp. 565–572. Boston: Butterworths, 1990.

[54] R. Stoakley, M. J. Conway, and R. Pausch. Virtual Reality on a WIM: Interactive Worlds in Miniature. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 265–272, 1995.

[55] H. Strasburger, I. Rentschler, and M. Jüttner. Peripheral Vision and Pattern Recognition: A Review. *Journal of vision*, 11(5):13:1–13:82, 2011.

[56] M. Sukan, C. Elvezio, O. Oda, S. Feiner, and B. Tversky. Parafrustum: Visualization Techniques for Guiding a User to a Constrained Set of Viewing Positions and Orientations. In *Proceedings of the Annual ACM symposium on User Interface Software and Technology*, pp. 331–340, 2014.

[57] J. P. Szlyk, C. L. Mahler, W. Seiple, D. P. Edward, and J. T. Wilensky. Driving Performance of Glaucoma Patients Correlates with Peripheral Visual Field Loss. *Journal of glaucoma*, 14(2):145–150, 2005.

[58] M. Tonnis and G. Klinker. Effective Control of a Car Driver's Attention for Visual and Acoustic Guidance Towards the Direction of Imminent Dangers. In *Proceedings of the IEEE and ACM International Symposium on Mixed and Augmented Reality*, pp. 13–22, 2006.

[59] M. Trapp, L. Schneider, C. Lehmann, N. Holz, and J. Döllner. Strategies for Visualising 3D Points-of-Interest on Mobile Devices. *Journal of Location Based Services*, 5(2):79–99, 2011.

[60] F. Vargas-Martin, E. Peli, et al. Augmented-View for Restricted Visual Field: Multiple Device Implementations. *Optometry and Vision Science*, 79(11):715–723, 2002.

[61] L. Vincent. Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. *IEEE transactions on image processing*, 2(2):176–201, 1993.

[62] B. A. Wandell. *Foundations of vision*, vol. 8. sinauer Associates Sunderland, MA, 1995.

[63] R. L. Woods, I. Fetchenheuer, F. Vargas-Martín, and E. Peli. The Impact of Non-Immersive Head-Mounted Displays (HMDs) on the Visual Field. *Journal of the Society for Information Display*, 11(1):191–198, 2003.

[64] R. Xiao and H. Benko. Augmenting the field-of-view of head-mounted displays with sparse peripheral displays. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 1221–1232. ACM, 2016.

[65] Y. Yano, J. Orlosky, K. Kiyokawa, and H. Takemura. Dynamic View Expansion for Improving Visual Search in Video See-Through AR. In *Proceedings of the International Conference on Artificial Reality and Telexistence and the Eurographics Symposium on Virtual Environments*, pp. 57–60, December 2016.