

ThinVR: Heterogeneous microlens arrays for compact, 180 degree FOV VR near-eye displays

Joshua Ratcliff, Alexey Supikov, *Member, IEEE*, Santiago Alfaro, and Ronald Azuma, *Fellow, IEEE*

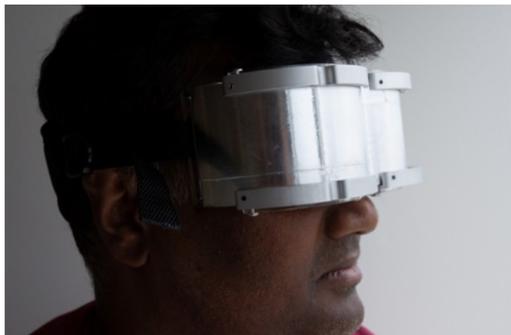


Fig. 1. Left: ThinVR prototype. Right: Photo taken in actual ThinVR prototype, covering the full FOV for one eye (about 130° horizontal). Lens: Samyang 12mm f/2.8 fisheye. Scene credit: Agent 327 (Blender Cloud) [CC-BY-4.0].

Abstract—Today’s Virtual Reality (VR) displays are dramatically better than the head-worn displays offered 30 years ago, but today’s displays remain nearly as bulky as their predecessors in the 1980’s. Also, almost all consumer VR displays today provide 90-110 degrees field of view (FOV), which is much smaller than the human visual system’s FOV which extends beyond 180 degrees horizontally. In this paper, we propose ThinVR as a new approach to simultaneously address the bulk and limited FOV of head-worn VR displays. ThinVR enables a head-worn VR display to provide 180 degrees horizontal FOV in a thin, compact form factor. Our approach is to replace traditional large optics with a curved microlens array of custom-designed heterogeneous lenslets and place these in front of a curved display. We found that heterogeneous optics were crucial to make this approach work, since over a wide FOV, many lenslets are viewed off the central axis. We developed a custom optimizer for designing custom heterogeneous lenslets to ensure a sufficient eyebox while reducing distortions. The contribution includes an analysis of the design space for curved microlens arrays, implementation of physical prototypes, and an assessment of the image quality, eyebox, FOV, reduction in volume and pupil swim distortion. To our knowledge, this is the first work to demonstrate and analyze the potential for curved, heterogeneous microlens arrays to enable compact, wide FOV head-worn VR displays.

Index Terms—Computational display, lenslets, wide field of view, head-worn display



1 MOTIVATION

Virtual Reality (VR) head-worn displays have become viable consumer products, with millions sold to consumers in the past few years. However, major technical problems limit the acceptance of VR displays. Two problems are the bulk and field-of-view (FOV) of such displays. First, the volume occupied by modern consumer VR head-worn displays is nearly the same as the volume of such displays in the 1980’s. Why? The fundamental reason is the distance between the optics and the display. Since the optics are large and F numbers below 1 are impractical, the focal length (which determines the distance between the optics and the display) is constrained to be at least ~40-50mm. Second, most VR displays today provide about 90-110 degrees FOV, partly because supporting wide fields of view with traditional optics requires even larger optical elements, further increasing bulk. However, 90-110 degrees is not ideal because presence and immersion increase with

larger FOV [18]. Some real world tasks, such as driving a vehicle, are impaired if the user’s FOV is restricted. Imagine trying to drive a car if your FOV was equivalent to being only able to see out the front window while looking straight ahead. A method to provide wide FOV in a thin form factor would increase acceptance of VR near-eye displays.

2 APPROACH AND CONTRIBUTION

Lanman and Luebke [16] placed arrays of homogeneous lenslets in front of microdisplays to build a near-eye light-field display. This enables the viewer to accommodate to different distances, at the cost of very low spatial resolution. But the most striking characteristic is that the display is exceptionally thin. Because the lenslets have small diameters, the distance between the optics and the microdisplay is only a few millimeters. We exploit this property by modifying their computational display approach to build a compact, wide FOV stereo VR head-worn display, rather than a near-eye light-field display.

Our approach is substantially different from previous work because we custom designed *heterogeneous* microlens arrays to make a large FOV feasible, and this was a difficult task. Lanman and Luebke used off-the-shelf homogeneous 1mm microlens arrays to cover a small FOV. But in a large FOV display, many lenslets are not viewed along their central axis, but rather at angles far from the central axis (Fig. 2). Lenses are typically designed to be viewed along the central axis, and

- All authors are with Intel Labs. E-mail: {joshua.j.ratcliff| alexei.soupikov| santiago.alfaro| ronald.t.azuma}@intel.com.

Manuscript received 10 Sept. 2019; accepted 5 Feb. 2020.
Date of publication 18 Feb. 2020; date of current version 27 Mar. 2020.
Digital Object Identifier no. 10.1109/TVCG.2020.2973064

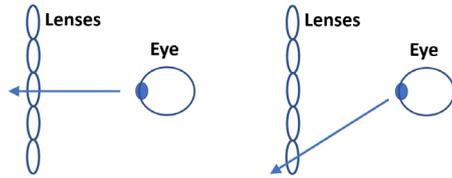


Fig. 2. Left: Lenslet in center is viewed along the central axis. Right: Lenslet in periphery is viewed off the central axis.

when viewed off-axis, they produce large distortions and aberrations, as shown in section 6.1. Therefore, to make a wide FOV possible, we had to design heterogeneous microlens arrays, where lenslets above and below the central horizontal row are optimized to be viewed off-axis.

Besides being heterogeneous, our microlens array also differs in lenslet aperture and curvature. Larger lenslets increase the percentage of the display visible at any given viewing position, thus increasing the spatial resolution. However, our use of larger lenslets also means that our display does not provide many overlapping views, so this is a stereo VR display rather than a light-field display. Furthermore, we designed a *curved* microlens array, instead of the flat array in Lanman and Luebke's system. By curving the optics and placing them in front of curved OLED (organic light emitting diode) displays, we can achieve 180 degrees horizontal FOV in a form factor that more closely fits the viewer's face, reducing the bulk. The minimum distance from the surface of our microlens array and the display surface is 11mm, substantially less than the ~40-50mm separation in traditional VR headsets. Fig. 1 shows a physical prototype demonstrating the potential to enable compact, wide FOV displays.

The contributions of this paper are:

- To our knowledge, this is the first work that combines curved, heterogeneous microlens arrays and curved displays to produce a compact, 180 degree horizontal FOV VR near-eye display.
- We discuss the design space for curved microlens arrays and how that drove the parameters of our prototypes.
- Successfully designing heterogeneous microlens arrays with a sufficient eyebox and acceptable pupil swim distortions required developing custom simulation and optimization tools specifically tuned to design heterogeneous microlens arrays.
- We demonstrate the viability of our approach through implemented prototypes.
- Finally, we evaluate our design. Our assessment of image quality, eyebox, FOV, volume reduction and pupil swim distortion reveals both the advantages and limitations of our approach.

3 RELATED WORK

Kress and Starner [15] provide an overview of a wide range of optical approaches for head-worn displays. Our work is a variation of the near-eye lenslet approach [16], and in the previous section we described how we differ by designing curved, custom heterogeneous optics. Huang and Hua [11] describe methods for designing optics for near-eye light-field displays that use homogeneous microlens arrays.

A few commercial head-worn VR displays provide 180 degrees or more horizontal FOV. Examples include the StarVR One [31], Pimax 5K Plus [27], and Xtal [33]. While these provide wide FOV, they use large optics and are therefore bulky. In section 6.2 we compare the size of our prototype against the Pimax.

The Pinlight display [21] used multiple point light sources to enable a compact, wide FOV near-eye display. Since this is based on flat displays and optical elements, this approach does not scale to 180 degrees horizontal FOV. Diffraction limits the image quality feasible through this approach.

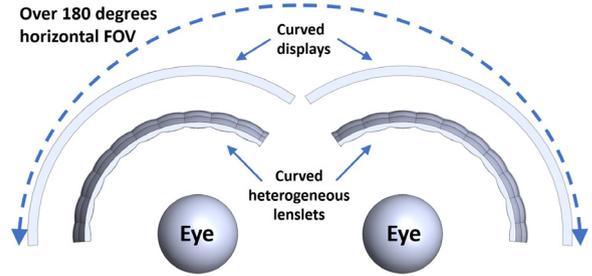


Fig. 3. Scale diagram showing how two cylindrical displays provide 180 degrees horizontal FOV.

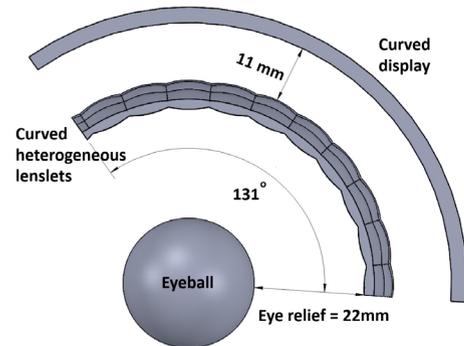


Fig. 4. Scale diagram of cylindrical lenslets and display for right eye.

The use of curved displays has been previously proposed. For example, Samsung filed a patent on a near-eye display with a curved display [8]. Our contribution is not on curving the display per se, but rather in the custom heterogeneous microlens arrays combined with curved displays, along with demonstrating and evaluating our approach.

Homogenous curved lenslet arrays have been used to enable non-head-worn integral displays [14] [29].

Pancake optics are a type of compound optics that bounce light multiple times forwards and backwards through the same optics, thus shrinking the distance between the optics and the display [17]. This comes at the cost of losing brightness and causing ghost imagery. The compound optics required are generally expensive, whereas our lenslet arrays are potentially inexpensive because they can be manufactured out of acrylic as a single object produced from a mold. DLODLO displays may use pancake optics [3].

Freeform optics have the potential to provide large FOV with a small number of optical elements. Limbak has produced a freeform design with two elements per eye and this was demonstrated in a Fraunhofer near-eye display [34]. It is unclear if this approach scales to 180 degrees FOV. Panasonic demonstrated a head-worn VR prototype display that provides 220 degrees FOV with two fused optics and two tiled displays per eye [9].

Head-mounted projective displays [10] can provide very wide FOV by using high power projectors, but these generally require covering the surrounding environment with retro-reflective material.

Massof et. al. [22] described a head-worn display that achieved 150 degrees horizontal FOV by using 16 tiled displays and 16 Fresnel optical elements per eye. The paper does not provide any depiction of what the viewer actually saw, nor does it provide a detailed analysis of the optical performance or the eyebox.

Holographic displays offer the long term potential to provide compact, wide FOV displays [20, 28], although there are serious challenges in computation, image quality, and in simultaneously providing both a wide FOV and an acceptable eyebox.

Finally, metasurfaces [2] offer the long-term potential of complete wavefront control through thin, diffractive optics. Although metasur-

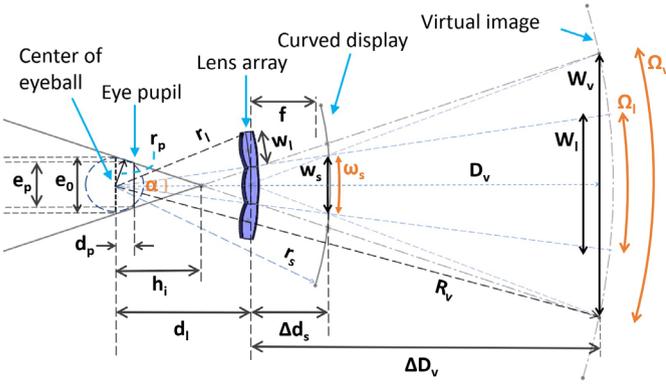


Fig. 5. Figure for eyebox and resolution estimates. Figure is not to scale.

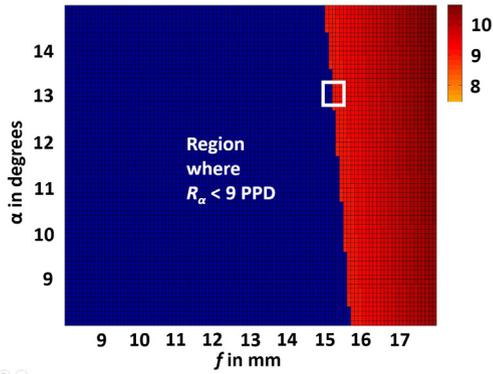


Fig. 6. R_α , the spatial resolution in PPD.

faces might eventually enable thin, low F number optics that provide a new way to build compact VR displays, they have many limitations, including chromatic aberrations and non-uniform diffraction efficiencies.

4 DESIGN SPACE

The first topic to consider in the design space is the shape of the display and optics. In theory, we could use any 3D shape for both the display and optics, to be as compact as possible and to maximize the stereo overlap. In practice, we are constrained by two limitations. First, to procure curved displays we had to extract flexible OLED display panels from phones (section 5.3). These flexible displays can bend along one axis without squashing or stretching, but not along two axes simultaneously. For example, we can curve them into a cylindrical shape but not a toroidal shape. Second, we needed to minimize the amount of optical design required to validate this approach. We could not afford to design dozens or hundreds of unique lenses.

These two constraints drove the basic geometries of the curved display and optics. We chose to curve both the display and the microlens array along matching cylindrical shapes, centered at the center of the eyeball. Two cylindrical displays cover a full 180 degree horizontal FOV (see Fig. 3 and Fig. 4). By using a cylindrical geometry, we reduced the optical design effort to one vertical column, where that column is replicated along the curve (Fig. 10). Due to symmetry, we were able to design half a vertical column and mirror image the top lenslets to form the bottom lenslets.

The remaining key parameters to determine in the design space were the angular width of the lenslets and the focal length, which in turn specify the lenslet pitch and the spacing between lenslets and the display. Our goal was to maximize spatial resolution while providing an adequate eyebox and keeping the system compact so it was physically possible to place two displays close enough to form a stereo VR display. We analyze the horizontal geometry (Fig. 5) which differs from [16] due to the cylindrical optics, display and virtual image. The vertical ge-

ometry follows the analysis of [16] with eyebox corrections if freeform optimization tilts or shifts the off-center lens aperture. Our cylindrical system analysis uses the thin lens approximation [23], treating lenses as magnifiers imaging flat regions w_s to W_v , as a proxy for freeform lenses that image cylindrical regions ω_s to Ω_v . Therefore, the formulas listed below are an approximation identifying a good starting point in the design space but by themselves aren't the actual optical design (see section 5.1 for those details).

First, we develop the equations that estimate the eyebox. e_0 is the width of the eyebox at the center of the eyeball, for one lenslet:

$$e_0 = \frac{r_l^2}{f} \sin(\alpha) \quad (1)$$

Where r_l is the lens replication radius (the distance from the center of the eyeball to the edge of the lenslet), f is the focal length of the lenslet, and α is the angle subtended by that lenslet from the center of the eyeball. Interestingly, e_0 does not depend on the spacing between the lenslets and the display because in the cylindrical design, α limits the maximum elemental image size.

However, what we actually need is e_p , the width of the eyebox at the eye pupil, rather than at the center of the eyeball:

$$e_p = e_0 - \frac{d_p}{d_l} (w_l + e_0) \quad (2)$$

Where d_p is the distance from the center of the eyeball to the eye pupil, d_l is the distance from the center of the eyeball to the center of the lenslet, and w_l is the width of the lenslet (the linear pitch). Given that a typical eyeball has a radius of 13mm and the eye pupil lies 4mm underneath the surface of the eye, d_p is 9mm. If we had only one lenslet, then e_p alone is a sufficient metric. But instead we have a cylindrical array of lenslets. Therefore we also need to factor in a conservative lower bound estimating the width of the overall eyebox for all the lenslets. We compute r_p , the lower bound of the overall eyebox width at the center of the eyeball:

$$r_p = \frac{e_0 h_i}{\sqrt{e_0^2 + 4h_i^2}} \quad \text{where} \quad h_i = \frac{e_0 d_l}{w_l + e_0} \quad (3)$$

r_p should be larger than d_p since r_p is computed at the center of the eyeball and we want the eye pupil to be within the compound eyebox estimate. So we seek to maximize e_p subject to the constraint that $r_p > d_p$.

Note that using a curved microlens array constrains the possible eyebox because the viewpoint needs to be near the center of curvature for the cylindrical geometries. This is a significant difference from the flat microlens array in [16] where there is freedom to adjust the elemental image regions assigned to each lenslet based upon the viewpoint. In a cylindrical design, the lenslets map cylindrical elemental images ω_s to cylindrical virtual images Ω_v .

Next, we estimate spatial resolution:

$$R_p = \frac{\Omega_l}{\Omega_v} N_p \quad \text{where} \quad N_p = \frac{\omega_s}{\Delta p} \quad \text{and} \quad \omega_s = \frac{d_l + \Delta d_s}{\cos \frac{\alpha}{2}} \alpha \quad (4)$$

R_p , the number of pixels visible in one lenslet, depends on Ω_l (the observed arclength through one lenslet at the virtual image), Ω_v (the arclength for the lenslet's elemental image at the virtual image) and N_p (the number of pixels for an elemental image). N_p depends on ω_s (the arclength of the elemental image at the curved display) and Δp (the pixel pitch, which is known given the display). Δd_s is the spacing between the lenslets and curved display.

We want to determine R_p as a function of f , α and set or derived values. Ω_l is obvious from α and R_v , so that is already a function of α . Ω_v is trivially found from ΔD_v , d_l and R_v . N_p depends on Δd_s .

Certain values are set or constrained. We set the virtual image distance R_v to 1m. r_l is constrained by human anatomy. If r_l is too small, the display won't fit around a face, but if r_l is too large the display is bulky. Based on ergonomic measurements of typical faces, we set r_l to ~ 36 mm. Given r_l and α we can compute d_l .

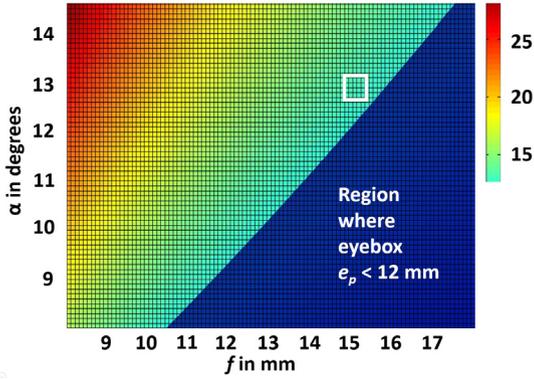


Fig. 7. e_p , the eyebox width at eye pupil, in mm.

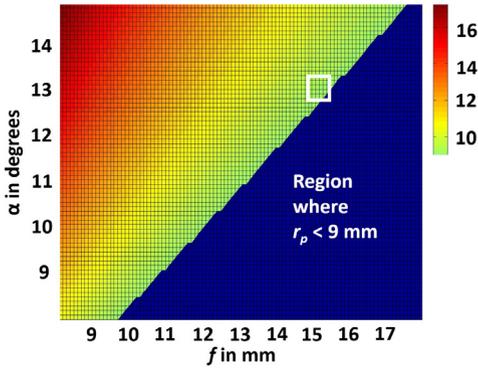


Fig. 8. r_p , lower bound of the compound eyebox at eyeball center.

Therefore, finding Ω_v and N_p requires finding unknowns ΔD_v and Δd_s . Space limits prevent showing the full derivation, but we can express ΔD_v and Δd_s in terms of f and known or derived values through careful substitutions and algebraic manipulation of the following four equations:

$$\frac{W_v^2}{4} + (d_l + \Delta D_v)^2 = R_v^2 \quad \frac{W_v}{w_s} = \frac{\Delta D_v}{\Delta d_s} \quad (5)$$

$$\frac{w_s}{w_l} = \frac{\Delta d_s + d_l}{d_l} \quad \frac{1}{\Delta d_s} - \frac{1}{\Delta D_v} = \frac{1}{f} \quad (6)$$

Where the first equation is based on a right triangle, the next two are due to similar triangles, and the last is the thin lens equation.

Finally, R_α , the spatial resolution in pixels per degree (PPD), is simply R_p/α .

Now we can plot three charts, for R_α (the spatial resolution, Fig. 6), e_p (the eyebox width at the pupil, Fig. 7), and r_p (the conservative estimate of the compound eyebox width at the center of the eyeball, Fig. 8). Each chart is a function of f and α , where r_l is set to 36.38 mm. We set targets of $e_p \geq 12$ mm and $R_\alpha \geq 9$ PPD with an underlying display resolution of 800 PPI (pixels per inch).

The three charts show the tradeoff that as f increases, resolution increases but the eyebox shrinks. Increasing α increases both eyebox and resolution, but that also increases lenslet width and thickness. The basis for this entire approach is using a microlens array rather than a single large lens or a few large lenses. Since an F number less than 1 is impractical, a large lens forces f to also be large. Therefore we want to avoid making α large. The hollow white square represents the point selected for our prototype, where α is 13 degrees (corresponding to a lenslet pitch of 9mm) and f is 15.255 mm.

5 IMPLEMENTATION

This section describes the optimization approach we used to design the heterogeneous optics, followed by a discussion of the other system components (rendering, curved display, manufacturing the lenslets) needed to build working prototypes.

5.1 Optical Design

Multiple attempts to use traditional optical design tools to specify the lenslets were unsuccessful in providing a unified eyebox and controlling pupil swim distortions as the eye moved within the eyebox. A major reason for this was the limitation on the number of rays we could trace and still finish optimization within a reasonable time. Therefore, we switched to a different approach based on implementing a real-time simulation of heterogeneous microlens arrays, which in turn enabled the development of a custom optimizer specifically tuned to design heterogeneous microlens arrays for near-eye displays. Near-real-time simulation meant our optimizer could more thoroughly explore the parameter space than we could with generic optical design tools.

We built our own ThinVR simulator by using Embree [4] to trace rays from a viewpoint or from an aperture, refract them through the lenslets, and then intersect them with the curved display. Given the curves that describe the lenslet geometry, we tessellate those into polygonal surfaces with side length of $50 \mu\text{m}$ and perform ray intersections with that polygonal geometry. We verified the accuracy of the simulator by tracing one million rays through the same lens geometries in both our simulator and in Zemax OpticStudio [36]. On average, the ray intersections at the display matched within $1 \mu\text{m}$. To further verify the simulator, we took pictures through a double convex lens with 20mm diameter and 30mm focal length and verified that the simulations matched those real images. Our simulator rapidly generates images showing what a person would see through the lenslet array. It generates images from a specific viewpoint or as viewed from an aperture, such as an eye pupil, and it simulates the effect of chromatic aberrations. Our simulator generates images at real-time rates for a single viewpoint and in 1-2 seconds when simulating a typical eye pupil aperture of 4mm.

This real-time simulation enabled us to build a better optimizer that performs a more thorough search of the lenslet design space than was feasible with traditional optical design tools that were not built to design heterogeneous lenslet arrays. The key aspects that the optimizer had to focus on were eyebox and pupil swim distortion. The eyeboxes of all the lenslets had to be aligned to produce an overall eyebox where the entire scene is visible. Pupil swim distortion is a more serious problem in a lenslet-based approach than in a normal approach that uses one large lens for each eye. With one large lens, pupil swim causes distorted imagery but the overall scene stays coherent. With multiple, heterogeneous optics, such distortions could cause the scene to become disjointed at the lenslet boundaries, which is a more disturbing artifact.

Therefore, our optimization constraint was designed to control pupil swim distortion and enforce an eyebox. Fig. 9 illustrates the ideal mapping between a point on the display, a lenslet, and the corresponding point on a virtual image. If such a relationship existed for every pixel in the elemental image assigned to that lenslet, that would minimize pupil swim because we would have a fixed distortion pattern even as the eye position changed. This would also guarantee the existence of an eyebox. We set the virtual image distance to 1m so that it is far away enough for accommodation but close enough to maximize the geometric strength of this relationship. The optimizer searches for lenslet parameters to make this relationship true for as many pixels as possible within the elemental image. We model each freeform lenslet as two classic aspherical surfaces, which are each specified by a radius of curvature, a conic coefficient and a Chebyshev polynomial [19] that performs fine adjustments to the surface. We use such polynomials because they are orthogonal, which means that for any given surface there is a unique set of coefficients that describe that surface, and these coefficients map nicely to optical aberrations such as spherical aberration and coma.

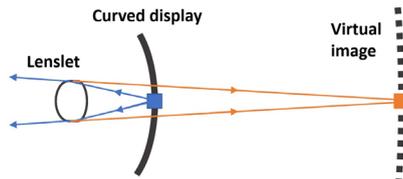


Fig. 9. Idealized relationship between a point on the curved display and its corresponding point on virtual image. This is used both in rendering and in optimization. Figure is not to scale.

Each surface is represented by the following formula:

$$z(x, y) = \frac{x^2 + (y + dy)^2}{R(1 + \sqrt{1 - \frac{(1+k)(x^2 + (y+dy)^2)}{R^2}})} + \sum_{i=0}^{10} \sum_{j=0}^{10} c_{ij} T_i\left(\frac{x}{N_x}\right) T_j\left(\frac{y+dy}{N_y}\right) \quad (7)$$

where $z(x, y)$ is the height of the lens surface, R is the radius of curvature, k is the conic coefficient, dy is a vertical offset, T_i and T_j are Chebyshev polynomials and c_{ij} are coefficients weighting the polynomials. The parameters to optimize are R , k , dy and the c_{ij} values. N_x and N_y are normalization constants that represent the largest magnitudes of x and y , to ensure that $\frac{x}{N_x}$ and $\frac{y+dy}{N_y}$ remain within the range -1 to 1.

The cost function for a single pixel is $u_0 D_{LS}^2 + u_1 S$ where the first term enforces fusion and the second enforces sharpness. u_0 and u_1 are manually set weights. Because it is more geometrically stable to find the closest intersection of a bundle of rays at the display rather than at the virtual image, we compute the least-squares intersection of the bundle of rays near the display and define D_{LS} as the distance from that intersection point to the display. S is the spot size of the bundle of rays at the display surface. The overall cost is the weighted sum of the cost function for 1000 individual pixels evenly sampled within the elemental image assigned to the lenslet being optimized, subject to three constraints: 1) $D_{LS} \leq 14\text{mm}$. 2) $S \leq \text{Pixel pitch of display}$. 3) Minimum lens thickness $\geq 0.8\text{mm}$ (for manufacturing). The weights change dynamically to penalize spot size increases due to a uniform distribution of the spot size budget across the pixels.

The core optimization procedure is an augmented Lagrangian method with nonlinear conjugate gradients, where violating a constraint incurs a severe quadratic penalty and modifies the gradient to push the solution towards meeting that constraint [24].

We first use the simulator to find a good initial guess of R , k , dy , c_{00} , c_{01} and c_{02} for both surfaces, by manually changing these parameters and observing the results in the simulator, which is feasible since the simulation runs in real time. Then the optimizer searches for parameters to minimize the cost function and meet the constraints listed previously. The gradient-based optimization approach requires derivatives, and we have to compute those derivatives numerically. Since we don't have closed form expressions of the first or second derivatives, we do not use optimization approaches that require second derivatives. Despite that, in practice this optimization converges quickly, requiring around 200 iterations, completing in 6 minutes on a 3 GHz Intel i7-5960X.

Fig. 10 shows a rendering of one column of lenslets and a photo of a manufactured heterogeneous microlens array. There are three unique lenslet designs (lenses 0, 1, and 2). Section 9 provides the dimensions and coefficients specifying each lens design.

5.2 Rendering

Lanman and Luebke [16] explain that near-eye displays that use lenslets require rendering small, low-resolution views of the scene from slightly different perspectives. Each individual rendered view is called an elemental image and is mapped to a particular lenslet. The elemental image size and position are computed using the thin lens model and geometrical considerations, subject to further corrections using ray tracing through the actual lens design. In this section we describe how we set the pixels in each elemental image.

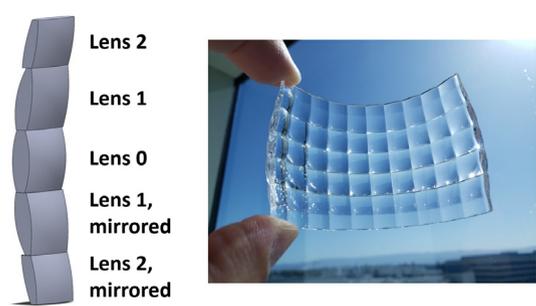


Fig. 10. Left: Rendering of one column of lenslets. Right: Photo of manufactured lenslet array.



Fig. 11. Image taken through microscope showing a magnified view of one manufactured lenslet (Lens 0). The surface is smooth in the center but inaccurate in the crevices between lenslets.

Assume we have rendered the 3D scene into a cylindrical panoramic image that exists on a virtual image, 1m away from the viewer. Fig. 9 shows how a pixel on the curved display maps to a point on the virtual image. The elemental image region the pixel is part of determines which lenslet to use. We trace a bundle of rays from the point on the curved display through the lenslet. These form diverging rays on the viewer's side of the lenslet (i.e., virtual imaging mode) because the spacing between the lenslet and curved display is less than the focal distance. We then trace these rays in the opposite direction, away from the viewer, to find the point where they intersect at the virtual image. Given this mapping, we build a lookup table that maps every pixel on the display to a point on the cylindrical panoramic image. An advantage of this approach is that it automatically incorporates corrections for geometric distortions in the optics.

The rendering task then becomes the task of mapping pixels from the cylindrical panoramic image to the screen space display buffer. Setting each pixel requires two texture queries, where one texture is the map that identifies the coordinates of the corresponding pixel in the other texture, which holds the cylindrical panoramic image. We implement this operation as a simple shader program that runs in parallel on the GPU. Avoiding aliasing would require a multi-sampled lookup operation to determine the final mipmap [35] level in the panoramic texture. Lookup tables can be represented compactly through polynomial fitting, enabling direct computation of sampling coordinates and their derivatives on systems where indirect texture lookups are expensive.

This approach requires rendering a scene as stereo cylindrical panoramic images. Some content already comes in that format. But if we need to render starting from 3D models, we can render multiple views of the scene from the two stereo viewpoints along different viewing angles and stitch or warp the results into stereo panoramas.



Fig. 12. ThinVR prototype with Galaxy S9 phone displays and electronics. Displays were set to a blue background to make the lenslets easy to see.

Rendering occurs at 60 Hz on two Samsung Galaxy S9 phones, one for each display. We use the Google Daydream API to activate low-persistence display mode to reduce the effects of latency, and we use the orientation tracking from that API on one phone and send the computed orientation to the other phone so that both phones display a synchronized scene rendered in stereo.

5.3 Curved Display

We acquired flexible 570 PPI OLED displays, with 2960 by 1440 resolution, by extracting them from Samsung Galaxy S9 phones. This was a difficult task because the displays are not designed to be separated from their glass covers. A company called TianShifu Phone Repair [32] had specialized equipment to expose the displays and covers to very low temperatures, enabling a clean separation. We then carefully mounted the fragile, flexible displays to conform to the necessary cylindrical shape.

5.4 Manufacturing Lenslets

The traditional way to manufacture lenslets is to create a mold, then use that mold to stamp out lenslets from acrylic or other materials. This can result in high-quality optics, and in mass quantities the per unit cost of each lenslet array is low, but this process is not compatible with low volume rapid prototyping. Each mold can cost tens of thousands of dollars and take months to manufacture.

Instead, we had lenslets manufactured through direct computer numerical controlled (CNC) machining of a block of acrylic. This process takes a few weeks, making it suitable for rapid prototyping necessary in a research project. A company called XCentric Mold charged \$735 to manufacture each lenslet array. The surface is smooth in the central parts of the lenslets, but the crevices where adjacent lenslets meet are inaccurate because the milling tool cannot cut deep enough in those areas (see Fig. 11). Incorrect geometries in the crevices can cause visible boundaries between lenslets in the scene. Still, we used this manufacturing technique to quickly and inexpensively build prototypes to evaluate the ThinVR approach.

5.5 Prototypes

We built two types of physical prototypes. First, we built ThinVR prototypes with flexible Galaxy S9 phone displays. These render scenes in real time in response to the wearer's head motions. However, these phone displays can only be driven by the Galaxy S9 electronics, so we had to mount those electronics and batteries onto the head-worn display, making our prototype much bulkier than it fundamentally has to be. Fig. 12 shows a prototype with phone displays.

Second, to more accurately depict the potential of ThinVR to enable compact head-worn VR displays, we also built prototypes with static prints instead of phone displays. We used 2032 PPI light valve technology (LVT) transparencies illuminated by cylindrical backlights. These depict a static scene and do not respond to head motion, but these prototypes show fundamentally how thin this approach could be. By using 2032 PPI prints we also illustrate the image quality expected when such high resolution displays become available. Fig. 1 shows

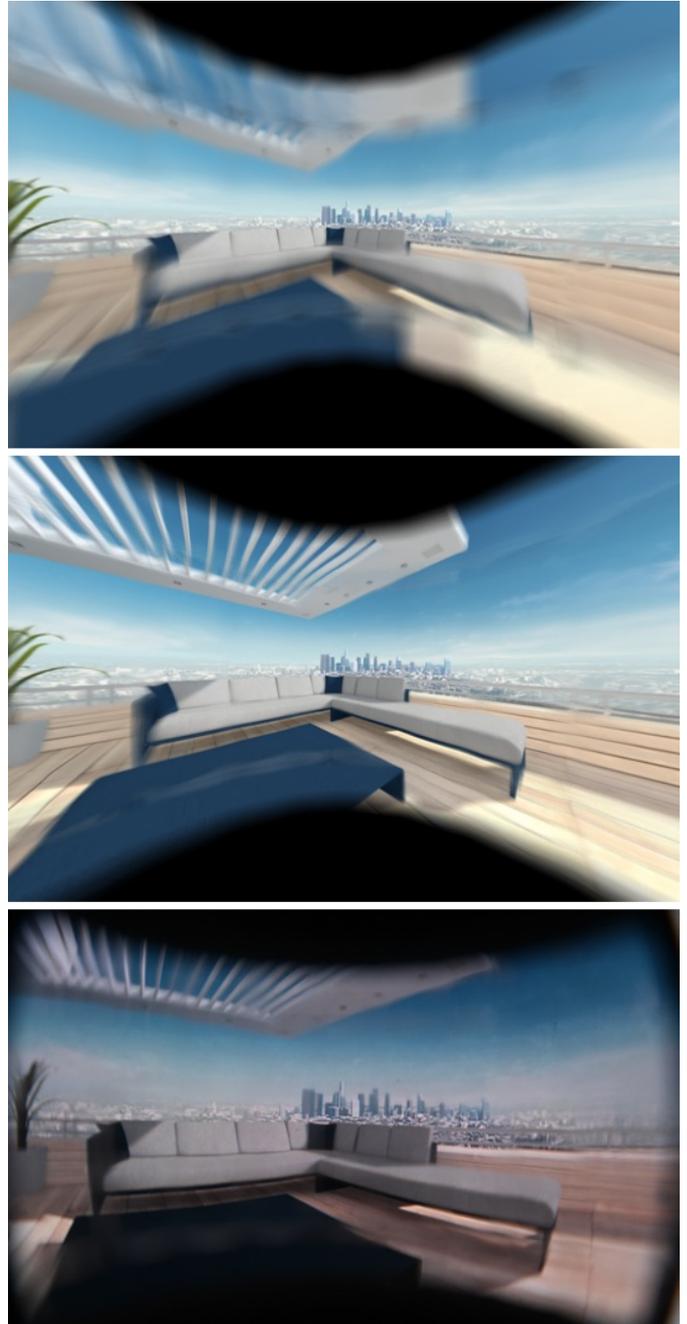


Fig. 13. Top: Simulated rendering of scene through homogeneous array of lenslets where lens 0 is used for all lenses. Middle: Simulated rendering of same scene with our heterogeneous lenslet array. Bottom: Photograph of the same scene taken through ThinVR prototype using static LVT print as the display. All images cover the full FOV visible in one eye, about 130 degrees horizontal. Note that the distortion in the simulated images, which are rendered with rectilinear projection, is not the same as the distortion visible in captured image. We did not attempt to match the distortion in the photo recorded through a wide FOV lens. Lens: Samyang 12mm f/2.8 fisheye, providing an aperture of 4.28mm which matches the typical diameter of the eye pupil. Scene credit: Hazelwood Loft from Polybox [Unity Asset Store license].

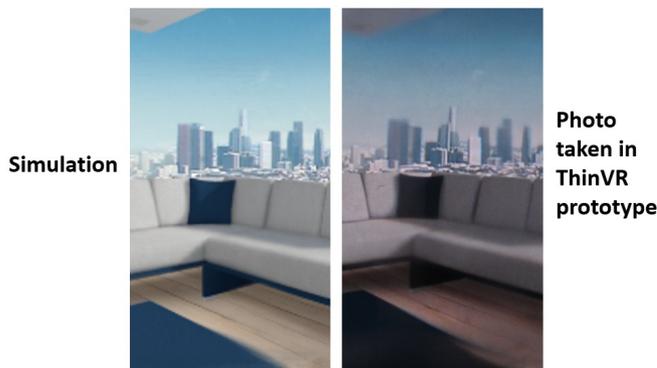


Fig. 14. Zoomed in view of matching parts of the middle and bottom images in Fig. 13. Left image is from the simulation of heterogeneous lenslets, and right image is from the photo taken through a physical ThinVR prototype.

an external view of a ThinVR prototype with printed displays, along with a photo of what a user sees in that prototype. The captured photo shown in Fig. 13 and Fig. 14 was also taken from a prototype using a 2032 PPI print display.

We discovered that precise separation between the lenslets and the curved displays is important. To improve the accuracy of the spacing, we designed and manufactured custom mounts to hold the cylindrical displays and lenslet arrays.

6 ASSESSMENT

In this section we analyze and evaluate the performance of the ThinVR approach, describing both its advantages and weaknesses compared against other approaches. Specifically, we examine image quality and resolution, volume, eyebox, FOV, and pupil swim distortion.

6.1 Image Quality and Resolution

Fig. 13 depicts a scene as seen through microlens arrays. The top image shows a simulation of the scene as viewed through a homogeneous lenslet array where all the lenses use the lens 0 design. The middle image shows a simulation of the same scene viewed through our ThinVR heterogeneous optics, using three lens designs. In both cases, we generated the appropriate elemental images customized to each lenslet array, with geometric distortion correction. Clearly, the homogeneous lenslet array has much worse image quality in the top and bottom areas, demonstrating the advantage of heterogeneous lenslet designs where the peripheral lenses are designed to be viewed off-axis (Fig. 2).

Fig. 15 quantifies this performance difference by plotting MTF (Modulation Transfer Function) [5] [25] charts for the homogeneous and heterogeneous microlens arrays, at three different vertical viewing angles where the eye rotates to look through the centers of lenses 0, 1 and 2. We used Zemax OpticStudio [36] to compute the MTF charts. Modulation values above 0.5 are generally considered good, between 0.3 - 0.5 as passable, and below 0.3 is poor. The homogeneous microlens array performance is poor for lenses 1 and 2. For the heterogeneous array it is acceptable except for lens 2 above 25 lines/mm (or 635 lines per inch), which in practice is tolerable since 25 lines/mm is high resolution.

The bottom image in Fig. 13 shows the same scene that was simulated in the middle image but the bottom image was captured with a DSLR camera photographing an actual ThinVR prototype using a LVT print with a backlight as the display. The two images are substantially similar, although the distortion is different as we did not attempt to match the distortion in the fisheye lens. In the bottom image, artifacts are faintly visible along the boundaries between lenslets. The lenslet boundaries are less apparent in Fig. 1, which is a much darker scene, suggesting this artifact is content dependent. A major cause of artifacts at lenslet boundaries is inaccurate geometries in the crevices between adjacent lenslets. We used an inexpensive manufacturing technique and

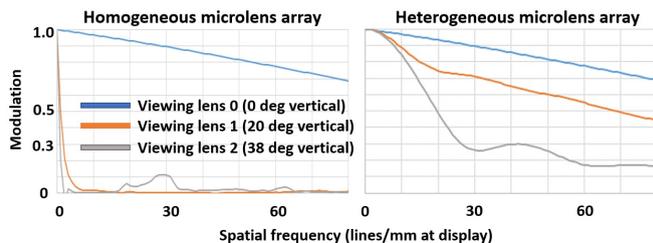


Fig. 15. MTF charts for the homogeneous and heterogeneous lenslet arrays, clearly showing the benefit of heterogeneous lenslets.



Fig. 16. Volume reduction: comparing the core optics and display components of our ThinVR prototype (bottom) and the Pimax (top).

the CNC milling cannot drill accurately and smoothly in the crevices, as seen in Fig. 11. The simulation suggests that this is not a fundamental limitation because such artifacts are not strongly apparent in the middle image, so a more accurate manufacturing process should help significantly. Another cause of artifacts are ledges in the optical design, which are 220 microns between lens 0 and 1 and 490 microns between lens 1 and 2. It is also possible that near the lenslet boundaries, light enters the back of one lens and then refracts to exit the front of a different lens, causing distortions. A revised optical design is needed to tackle those two sources of artifacts. Also, inaccuracies in the spacing and positioning of all the components adversely affect the image quality in the actual prototype. In the simulated image in the middle, all geometries are perfect but in the real prototype we are limited to the accuracy we can achieve with our 3D printed holders that mount the optics and display. Still, the close match between simulation and reality suggests this approach is viable, given a sufficiently precise implementation.

Fig. 14 shows zoomed in views of the middle and bottom images of Fig. 13, comparing the simulated image against what we recorded from an actual ThinVR prototype, near the center of the FOV.

We don't plot the MTF of a large lens from a traditional VR display because we don't have models of those optics. However, we can compare the spatial resolution. The achievable resolution, as measured in the center of lens 0, is 6 PPD when using the Galaxy S9 displays and 21.4 PPD with LVT prints providing a display with 2032 PPI. We computed this in the simulator by counting pixels in the rendered image when simulating different display resolutions. This roughly matches the design goal of 9 PPD with a 800 PPI display (section 4). The resolution we can achieve with Galaxy S9 phone displays is lower than the spatial resolution offered by the first generation (8-10 PPD) and current generation (15 PPD) of consumer VR HMDs, although with very high resolution displays the overall resolution becomes more acceptable. What this illustrates is a fundamental tradeoff of lenslet-based approaches vs. the traditional approach of using a single large lens. The lenslet approach provides flexibility, which can generate multiple overlapping views to support accommodation [16] or, as seen in this paper, the ability to provide wide FOV in a compact form factor. However, at any instant, the viewer sees a smaller number of pixels than he or she would with a single large lens. The pixels that are not viewed are not wasted, because different subsets of the elemental images are seen as the eye moves within the eyebox. But it does mean

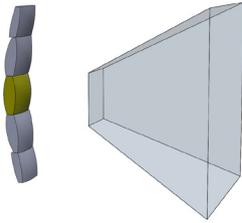


Fig. 17. Viewing volume for a single lenslet.

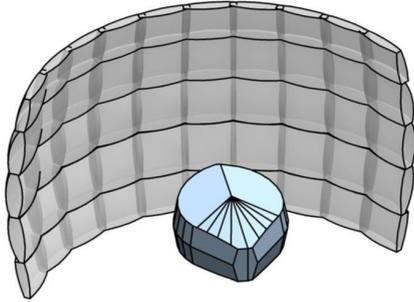


Fig. 18. 3D shape of intersection of all viewing volumes for all lenslets for a single eye. This is the eyebox, shown relative to the microlens array.

that lenslet-based approaches provide lower spatial resolution than a traditional single large lens approach, given the same resolution of the underlying display.

6.2 Volume

What fundamentally limits how compact VR near-eye displays can become are the optics, the displays and the space between them. All other elements, such as electronics and batteries, do not have to be positioned directly in front of the viewer's eyes. Therefore, a fair comparison focuses solely on the optics and display elements.

Fig. 16 shows just those components for our ThinVR prototype and the Pimax. The Pimax is impressively compact. The optics are large but the distance between the middle of the optics and the display surface is only 40mm. We computed the volumes by building 3D models of both in SOLIDWORKS [30]. The total volume of the Pimax is 598.7 cm³ and our ThinVR prototype has a volume of 328.5 cm³. Therefore the ThinVR approach can reduce the volume by almost half.

6.3 Eyebox

Our optical configuration provides a smaller eyebox than traditional VR optics do. The Lagrange Invariant [6] states that the product of the display size and the numerical aperture of the optics equals the product of the eyebox and FOV. A traditional design using a large lens (40-50mm) and 2.5 - 3.5 inch displays can easily provide both a large FOV and large eyebox. However, in our lenslet-based design, each lenslet is small and observes a small patch of the display. While the FOV each lenslet needs to provide is also small, the reduced product constrains the eyebox.

Fig. 17 shows the volume for a single lenslet from which the proper elemental image region is seen through that lenslet. We compute this by shooting rays from all the pixels in the elemental image through the lenslet, then intersecting those with two planes placed in front of and behind the distance where the pupil should be. We then compute the silhouettes around all the ray intersections in each plane, and then create a 3D viewing volume that connects the two silhouettes.

The overall eyebox is the 3D intersection of all the viewing volumes of all the lenslets, which requires careful design to ensure these volumes are aligned with each other. We generated the individual view volumes for all lenslets, then used SOLIDWORKS [30] to compute the 3D volume that is the intersection of all the viewing volumes. Fig. 18 shows a 3D rendering of this intersection, which is the eyebox.

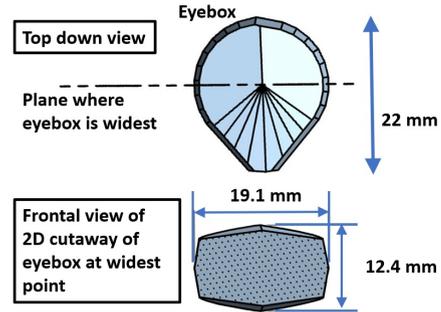


Fig. 19. 2D cutaway view of the 3D eyebox at its widest point.

Fig. 19 shows a 2D cutaway of the eyebox volume from Fig. 18 at its widest point. It is 19.1mm wide by 12.4mm tall. The maximum depth of the 3D eyebox is 22mm. Note that this eyebox represents the region within which we see the correct elemental image for all lenslets. If the eye pupil moves outside this volume, then the eye will see an incorrect elemental image region in at least one lenslet. Therefore the entire eye pupil needs to stay within this volume, so the range of translation is smaller than the width of the eyebox. For example, a 4mm eye pupil could move horizontally 15mm within a 19mm eyebox. The eyebox we achieved is acceptable if the interpupillary distance is adjustable so that both eyes are centered in the eyeboxes (see note in section 7).

6.4 FOV

Given that we have shown that there is an eyebox from which the viewer can see the scene through all lenslets, determining FOV is a simple matter of adding up the FOV contributions of the adjacent lenslets. Each lenslet provides 13.1 degrees horizontally, so the total horizontal FOV for each eye is slightly over 130 degrees. Fig. 3 shows how two displays provide over 180 degrees horizontal FOV. The binocular overlap region is 71.4 degrees. A restricted binocular overlap region is common in very wide FOV HMDs [27, 31, 33]. As viewed from the front of the eyebox, lenses 0 and 1 provide a vertical FOV of almost 18 degrees and lens 2 provides a vertical FOV of almost 14 degrees, for an overall vertical FOV of 81.7 degrees.

6.5 Pupil Swim Distortion

Pupil swim distortion is the change in optical distortion as the viewpoint position changes. This distortion exists in traditional VR displays with large lenses. However, as previously mentioned in section 5.1, such distortion can cause more severe artifacts in microlens arrays than in a single large lens. With one large lens, the viewed image may become warped and distorted but it will still be coherent, whereas with a microlens array, the distortions may differ between adjacent lenslets, causing lines and features to disconnect. Because of the importance of minimizing pupil swim distortion, we used the relationship in Fig. 9 as the optimization criteria.

To evaluate the pupil swim distortion in our prototype, we generated elemental images that, when viewed through the microlens array, create a curved 2D checkerboard pattern. Note that what is rendered onto the display itself is not a simple 2D checkerboard pattern but rather many views of the scene from different positions and angles, set via the procedure in section 5.2. We generated 49 images in the simulator from 49 viewpoints sampled via a Halton Sequence [7] across a 12mm aperture disk. We then shifted each image to align them with each other to compensate for the slightly different viewpoints and then averaged all the pixels to generate the image in Fig. 20. If no pupil swim distortion existed, then this image would show a perfectly sharp checkerboard pattern. Blurry regions in the image identify areas with pupil swim distortion, where regions with more blur have more distortion.

Fig. 20 shows the center of the viewer's FOV (marked by a red cross) has little pupil swim distortion. This is the spot the viewer's fovea sees. Pupil swim distortion increases away from this spot. In natural scenes, pupil swim distortion causes blur and since the resolution of

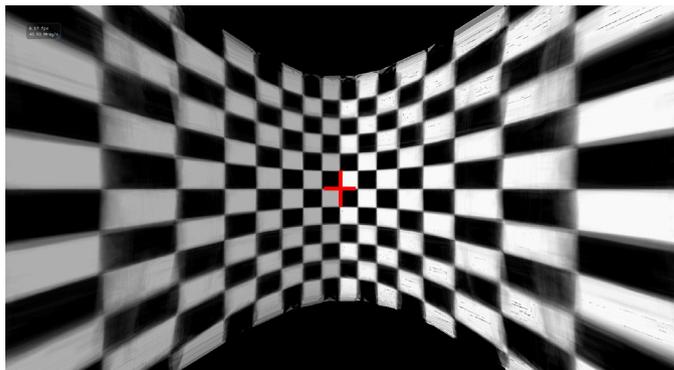


Fig. 20. Depiction of pupil swim distortion by averaging 49 simulated images of a scene that should be a curved 2D checkerboard. Blurry regions indicate areas that have pupil swim distortion. The red cross was added to indicate the center of the viewer's FOV.

the human eye decreases sharply away from the fovea, we expect that this distortion does not cause disturbing artifacts in most scenes. This is not a guarantee, however, as the human visual system is complex and there could be scenes where the pupil swim distortion results in artifacts that the human visual system is sensitive to.

As the viewer's eye rotates horizontally, we will see the same distortion pattern except that the minimum distortion will be at the spot the viewer is fixating, i.e. the region the fovea sees. However, vertical eye rotations do steer the fovea to regions with larger pupil swim distortion. We do observe visible artifacts in our prototypes when rotating our eyes toward the top or bottom of the display. However, in almost all VR displays the optical performance is visibly worse when the viewer chooses to gaze at the periphery rather than the center, so this is not a characteristic unique to this approach.

7 LIMITATIONS AND ISSUES

Our approach relies upon curved displays. Flexible OLED displays exist in some mobile phones, but drivers are not available for such displays. Therefore, such displays can only be driven by the phone that came with the display. Thus, we must attach the phone electronics and batteries to our dynamic head-worn display prototypes, one for each display. This makes our dynamic prototypes bulky, but this is not a fundamental limitation. With a display driver, it is possible to remove those electronics and batteries from the front of the viewer's face.

Since this approach generates a smaller eyebox than what a traditional VR display with large optics provides, we need the ability to adjust the interpupillary distance (IPD) between the displays, in hardware. Traditional VR displays can support such a large eyebox that they often do not require mechanical IPD adjustments.

A compact, form fitting display does not have enough room for a user to wear prescription glasses underneath. It should be possible to provide prescription correction either through custom inserts that follow the curved display or by designing curved Alvarez lenses [1] that the viewer can adjust.

8 FUTURE WORK AND CONCLUSION

We outline several possible approaches to improve ThinVR:

- Reliable eye position tracking would improve the performance of the ThinVR approach. Recent commercial VR and AR displays are incorporating eye gaze tracking natively, and some recent research in head-worn displays relies upon eye tracking to enable dynamic eyeboxes [12, 13, 26]. We could redesign the optics to reduce the eyebox and dynamically adjust which parts of the display map to which lenslets, based upon the eye position. Despite a smaller instantaneous eyebox, the range of supported eye positions would increase due to the dynamic adjustment of elemental image regions. Reducing the eyebox will increase the spatial resolution (Section 4).

- Optical performance could improve significantly if we designed and used two separate lenslet arrays, where the second array would be inserted in between the current array and the display. However, this increases cost and system complexity and requires careful alignment between two sets of optics and the display.
- Finally, instead of the cylindrical design, the ThinVR display could be curved into a shape that better follows the contours of a human face, providing a more compact and ergonomic display. This requires custom designing many more lenslets, which is a substantial but worthwhile optical design effort for a mass-produced consumer product.

In conclusion: We demonstrated that the ThinVR approach can provide 180 degrees horizontal FOV in a head-worn VR display in a compact form factor, simultaneously addressing two crucial parameters in VR displays: bulk and limited FOV. We hope this work stimulates new approaches for solving the remaining problems in VR displays and accelerates the day when VR becomes ubiquitously accepted.

9 APPENDIX: CHEBYSHEV POLYNOMIAL COEFFICIENTS FOR OUR LENSLETS

We list the nonzero coefficients specifying each lenslet. Applying these coefficients to equation (7) defines the front and back surfaces of each lenslet. The front surface faces the viewer's eye and the back surface faces the display. Lens 0 and 1 both have an aperture of 9mm by 9mm, while for lens 2 it is 9mm by 7mm.

Lens 0 front surface coefficients: $R = -11$, $k = -2.5$, $c_{20} = 0.04398116097$, $c_{40} = 0.022$, $c_{10,0} = -0.0027$, $c_{02} = 0.03282507509$, $c_{22} = 0.002768074162$, $c_{42} = -0.003980249166$, $c_{82} = -0.00085$, $c_{04} = 0.01724025235$, $c_{24} = -0.001995130908$, $c_{06} = -0.003260227852$

Lens 0 back surface coefficients: $R = 13.27203274$, $k = -1$, $c_{20} = -0.03866756707$, $c_{40} = 0.03076090291$, $c_{60} = 0.003825364169$, $c_{02} = -0.03255109489$, $c_{22} = -0.004231499042$, $c_{42} = -0.001427617623$, $c_{04} = 0.02790133283$, $c_{24} = -0.0007210691692$, $c_{06} = 0.0007621544064$

Lens 1 front surface coefficients: $R = -13$, $k = -7.17$, $dy = -0.05$, $c_{01} = 1.891272187$, $c_{42} = 0.019750014$, $c_{62} = 0.003$, $c_{03} = 0.03385$, $c_{24} = 0.00105$, $c_{05} = -0.0054$, $c_{26} = 0.0015$, $c_{07} = -0.0074$, $c_{92} = -0.00025$

Lens 1 back surface coefficients: $R = 11.55284405$, $k = -0.93$, $dy = 0.0101$, $c_{00} = -0.1$, $c_{01} = 4.0$, $c_{22} = -0.01975845173$, $c_{42} = 0.003$, $c_{23} = 0.01475$, $c_{24} = 0.0017$

Lens 2 front surface coefficients: $R = -14$, $k = -7.35$, $dy = -0.5$, $c_{01} = 2.01$, $c_{21} = 0.02098057978$, $c_{41} = 0.02411588468$, $c_{81} = -0.001$, $c_{22} = 0.01890855841$, $c_{03} = -0.01$, $c_{23} = -0.00025$, $c_{25} = -0.00285$

Lens 2 back surface coefficients: $R = 13.19037628$, $k = -8.560012817$, $dy = 2.1$, $c_{01} = 2.5$, $c_{21} = -0.07$, $c_{41} = 0.025$, $c_{22} = 0.00611$, $c_{03} = -0.01045$, $c_{23} = 0.0012$, $c_{43} = -0.00015$, $c_{04} = 0.0001$, $c_{64} = 0.00015$, $c_{26} = 0.00005$

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their valuable comments and constructive criticism. We thank Mario Palumbo, Jeff Solari, and Ginni Grover for their reviews and comments. Basel Salahieh performed the initial investigations in designing heterogeneous multilens arrays. Tuotuo Li found the company that removed the glass from the Samsung Galaxy S9 phone displays (Section 5.3). Avinash Kumar wears the ThinVR prototype in Fig. 1. Ginni Grover helped with the MTF analysis.

REFERENCES

- [1] S. Barbero. The Alvarez and Lohmann refractive lenses revisited. *Opt. Express*, 17(11):9376–9390, May 2009.
- [2] S. Chang, X. Guo, and X. Ni. Optical metasurfaces: Progress and applications. *Annual Review of Materials Research*, 48(1):279–302, 2018.
- [3] DLODLO, 2019. <https://www.dlodlo.com/en/index>, last accessed August 19, 2019.
- [4] Intel Embree: High Performance Ray Tracing Kernels, 2019. <https://www.embree.org>, last accessed August 19, 2019.
- [5] J. W. Goodman. *Introduction to Fourier Optics*. W. H. Freeman, New York, NY, USA, 4th edition, 2017. pp. 197-204.

- [6] J. E. Greivenkamp. In *Field Guide to Geometrical Optics*, volume FG01, chapter 1, page 28. SPIE Press, 2004.
- [7] J. H. Halton. Algorithm 247: Radical-inverse quasi-random point sequence. *Commun. ACM*, 7(12):701–702, Dec. 1964.
- [8] S.-S. Han, M.-J. Choi, J.-W. Ko, and T.-K. Kim. Head-mounted electronic device. US Patent Application Publication, US 2019/0026871, Jan. 24, 2019.
- [9] F. He. Panasonic's 220 degree VR headset uses crazy fused lenses and 4 screens, Jan. 2017. <https://www.roadtovr.com/panasonics-220-degree-vr-headset-uses-4-screens-crazy-fused-lenses/>, dated Jan. 12, 2017.
- [10] H. Hua, C. Gao, J. P. Rolland, and F. Biocca. An ultra-light and compact design and implementation of head-mounted projective displays. In *Proceedings of the Virtual Reality 2001 Conference (VR'01)*, VR '01, pages 175–182, Washington, DC, USA, 2001. IEEE Computer Society.
- [11] H. Huang and H. Hua. Generalized methods and strategies for modeling and optimizing the optics of 3D head-mounted light field displays. *Opt. Express*, 27(18):25154–25171, Sep 2019.
- [12] C. Jang, K. Bang, S. Moon, J. Kim, S. Lee, and B. Lee. Retinal 3D: Augmented reality near-eye display via pupil-tracked light field projection on retina. *ACM Trans. Graph.*, 36(6):190:1–190:13, Nov. 2017.
- [13] J. Kim, Y. Jeong, M. Stengel, K. Akşit, R. Albert, B. Boudaoud, T. Greer, J. Kim, W. Lopes, Z. Majercik, P. Shirley, J. Spjut, M. McGuire, and D. Luebke. Foveated AR: Dynamically-foveated augmented reality display. *ACM Trans. Graph.*, 38(4):99:1–99:15, July 2019.
- [14] Y. Kim, J.-H. Park, H. Choi, S. Jung, S.-W. Min, and B. Lee. Viewing-angle-enhanced integral imaging system using a curved lens array. *Opt. Express*, 12(3):421–429, Feb 2004.
- [15] B. Kress and T. Starner. A review of head-mounted displays (HMD) technologies and applications for consumer electronics. In *Proceedings of SPIE 8720, Photonic Applications for Aerospace, Commercial and Harsh Environments IV*. SPIE, 2013.
- [16] D. Lanman and D. Luebke. Near-eye light field displays. *ACM Trans. Graph.*, 32(6):220:1–220:10, Nov. 2013.
- [17] J. A. LaRussa and A. T. Gill. The holographic pancake window TM. In *SPIE Visual Simulation and Image Realism*, volume 0162, pages 120–129. SPIE, 1978.
- [18] J. J.-W. Lin, H. B. Duh, D. E. Parker, H. Abi-Rached, and T. A. Furness. Effects of field of view on presence, enjoyment, memory, and simulator sickness in a virtual environment. In *Proceedings IEEE Virtual Reality 2002*, pages 164–171. IEEE, 2002.
- [19] F. Liu, B. M. Robinson, P. Reardon, and J. M. Geary. Analyzing optics test data on rectangular apertures using 2-D Chebyshev polynomials. *Optical Engineering*, 50(4):1–9–9, 2011.
- [20] A. Maimone, A. Georgiou, and J. S. Kollin. Holographic near-eye displays for virtual and augmented reality. *ACM Trans. Graph.*, 36(4):85:1–85:16, July 2017.
- [21] A. Maimone, D. Lanman, K. Rathinavel, K. Keller, D. Luebke, and H. Fuchs. Pinlight displays: Wide field of view augmented reality eyeglasses using defocused point light sources. *ACM Trans. Graph.*, 33(4):89:1–89:11, July 2014.
- [22] R. W. Massof, L. G. Brown, M. D. Shapiro, G. D. Barnett, F. H. Baker, and F. Kurosawa. 37.1: Invited Paper: Full-Field High-Resolution Binocular HMD. *SID Symposium Digest of Technical Papers*, 34(1):1145–1147, 2003.
- [23] P. Mouroulis and J. Macdonald. *Geometrical Optics and Optical Design (Oxford Series in Optical and Imaging Sciences)*. Oxford University Press, Oxford, England, UK, 1st edition, 1996. pp. 44–45.
- [24] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer-Verlag, New York, NY, 2nd edition, 2006.
- [25] Optical transfer function - Wikipedia, 2019. https://en.wikipedia.org/wiki/Optical_transfer_function, last accessed December 30, 2019.
- [26] J.-H. Park and S.-B. Kim. Optical see-through holographic near-eye-display with eyebox steering and depth of field control. *Opt. Express*, 26(21):27076–27088, Oct 2018.
- [27] Pimax, 2019. <https://www.pimaxvr.com>, last accessed August 19, 2019.
- [28] L. Shi, F.-C. Huang, W. Lopes, W. Matusik, and D. Luebke. Near-eye light field holographic rendering with spherical waves for wide field of view interactive 3D computer graphics. *ACM Trans. Graph.*, 36(6):236:1–236:17, Nov. 2017.
- [29] D.-H. Shin, B. Lee, and E.-S. Kim. Multidirectional curved integral imaging with large depth by additional use of a large-aperture lens. *Appl. Opt.*, 45(28):7375–7381, Oct 2006.
- [30] 3D CAD Design Software SOLIDWORKS, 2019. <https://www.solidworks.com>, last accessed August 22, 2019.
- [31] StarVR, 2019. <https://www.starvr.com>, last accessed August 19, 2019.
- [32] TianShifu Phone Repair, 2019. https://item.taobao.com/item.htm?spm=a1z09.2.0.0.49502e8dY9U4T5&id=558084516850&_u=h2fbk9jb8aa7, last accessed December 30, 2019.
- [33] vrgineers, 2019. <https://vrgineers.com>, last accessed August 19, 2019.
- [34] P. Wartenberg, M. Buljan, B. Richter, G. Haas, S. Brenner, M. Thieme, U. Vogel, and P. Benitez. 40-5: Invited paper: High frame-rate 1" WUXGA OLED microdisplay and advanced free-form optics for ultra-compact VR headsets. *SID Symposium Digest of Technical Papers*, 49(1):514–517, 2018.
- [35] L. Williams. Pyramidal parametrics. *SIGGRAPH Comput. Graph.*, 17(3):1–11, July 1983.
- [36] Zemax: Leading optical product design software for engineering teams, 2019. <https://www.zemax.com>, last accessed August 19, 2019.