Wide field of view compressive light field display using a multilayer architecture and tracked viewers

Renjie Chen
Andrew Maimone
Henry Fuchs
Ramesh Raskar
Gordon Wetzstein

Abstract — In this paper, we discuss an intuitive extension to compressive multilayer light field displays that greatly extends their field of view and depth of field. Rather than optimizing these displays to create a moderately narrow field of view at the center of the display, we constrain optimization to create narrow view cones that are directed to a few viewers' eyes, allowing the available display bandwidth to be utilized more efficiently. These narrow view cones follow the viewers, creating a wide apparent field of view. Imagery is also recalculated for the viewers' exact eye positions, creating a greater depth of field. The view cones can be scaled to match the positional error and latency of the tracking system. Using more efficient optimization and commodity tracking hardware and software, we demonstrate a real-time, glasses-free 3D display that offers a 100° × 40° field of view.

Keywords — *wide field of view, compressive display, light field display, multilayer.*

DOI # 10.1002/jsid.285

1 Introduction

Since the invention of the stereoscope by Sir Charles Wheatstone in 1838, the public has been fascinated by the idea of viewing photographs, cartoons, or movies in 3D. Although a vast range of 3D displays have been invented in the last 150 years,[1] 3D technologies have had limited commercial success thus far despite the ubiquity of 2D displays among smartphones, tablets, computer monitors, and televisions. Part of the reason for this failure may be that most existing 3D displays require additional eyewear, such as polarizing glasses, which is impractical for everyday use. Light field or glasses-free 3D displays are an alternative, but usually sacrifice image resolution for 3D capabilities.[2] Unfortunately, the human visual system is extremely sensitive to resolution. Degrading overall image quality to support 3D presentation does not increase the overall viewing experience – rather, it appears to decrease it. With a critical flicker fusion threshold of about 60 Hz, however, the human visual system is rather insensitive to fast motions. Time-multiplexed 3D displays offering high-image resolution and 3D capabilities simultaneously therefore seem like the most promising avenue of practical 3D displays (e.g.,[3,4]). One of the major challenges of time-multiplexed displays, however, is that most readily available display hardware, such as liquid crystal displays (LCDs), support only limited refresh rates of 60-240 Hz. The number of time-multiplexed images is therefore limited to only a few sub-frames, making these technologies suitable for stereoscopic but not multiview or light field displays.

Over the last few years, a new generation of displays has started to emerge: compressive light field displays. By combining unconventional optical setups, such as multilayer LCDs[5–9] or directional backlights,[10,11] with compressive computation, these types of displays support unprecedented image resolution and 3D capabilities using commodity hardware. The key idea behind all of these displays is to directly exploit the compressibility of the presented light field image content. However, as opposed to conventional 2D image compression, compressive light field displays employ a joint optical and computational approach to presenting compressed content that allows the human visual system to act as a decoder. Despite the efficiency gains made by compressive displays, however, these displays still have limited display bandwidth: they are unable to present glasses-free imagery over a wide field of view.

In this paper, we explore the combination of head tracking and compressive light field displays. Whereas previously-described light field displays support limited fields of view of about 10° to 20°, we demonstrate that head tracking can significantly increase the field of view of a compressive light field display as well as its depth of field.

2 Related work

Stereoscopic and multiview 3D displays have been an active area of research for more than a century.[1] With the emergence of high-speed displays, time-multiplexed 3D image presentation has become one of the most promising directions.[3,4] These technologies can be made more efficient by exploiting the compressibility of the displayed multiview

Received 08/7/14; accepted 03/9/15
R. Chen, A. Maimone and H. Fuchs is with University of North Carolina at Chapel Hill, Chapel Hill, NC, USA; e-mail: renjie.c@gmail.com.
R. Raskar is with MIT Media Lab, Cambridge, MA, USA
G. Wetzstein Stanford University, CA, USA

image content. In particular, compressive light field displays[6,11] use modern matrix and tensor factorization algorithms to enhance image brightness and reduce the requirements on display refresh rates as compared with conventional time-multiplexing techniques. Whereas most existing 3D displays are limited to binocular depth cues, compressive light field displays have also demonstrated support for eye accommodation.[10]

One of the primary challenges for all existing light field display technologies is support for a wide field of view. Current display designs dictate a tradeoff on achievable field of view: image resolution must be degraded[2] or display refresh rates must be increased[3,4] to widen the field of view. Whereas most compressive light field display architectures have thus far only demonstrated limited fields of view,[5–7,9,11] head and eye tracking has been shown to allow for more flexible display modes in some configurations.[10,12] However, these existing tracked works[10,12] have not demonstrated a real-time, wide field of view display system that operates in high resolution.

In this paper, we explore the promise of head-tracked compressive light field displays. By flexibly steering a 'view cone' with a small instantaneous field of view towards the direction of the observer, we demonstrate that the observed field of view of a dual-LCD light field display can be significantly increased. This methodology applies to a variety of other light field displays as well.

3 Compressive image synthesis

This section briefly reviews image formation, rank deficiencies of multilayer light field displays, and strategies for efficient image synthesis. Our basic description follows Lanman et al.[6] and Wetzstein et al;[11] please see these works for full details.

3.1 Rank deficiency of parallax barriers

We employ an *absolute two-plane* parameterization for the light field emitted by a dual-layer display. The discrete light field $\boldsymbol{L}$ produced by two stacked attenuating layers $\boldsymbol{f}[i]$ and $\boldsymbol{g}[k]$, for instance, LCDs, can be represented as the outer product:

$$L[i,k] = f[i] \otimes g[k] = f[i]g^T[k]$$

In this analysis, we show 1D masks and a 2D light field for simplicity, but this generalizes to 4D by reordering the 4D light field into a 2D matrix.

From the aforementioned equation, it is clear that any pair of 2D masks can only produce a rank-1 approximation of a 4D light field. Here, rank can be understood to indicate the *degrees of freedom* available, thus a rank-1 approximation of a light field will be of poor quality: the field of view is small and the depth of field is shallow. Experiments shown in past work[6] indicate that a rank-3 reconstruction should be sufficient to obtain a PSNR (peak signal-noise ratio) of 30 dB.

To achieve a higher rank reconstruction, we take advantage of a perceptual effect in the human visual system: flicker fusion. A rapid sequence of images will be perceived as their temporal average. Thus, a series of rank-1 matrices can be integrated to achieve a higher rank matrix. Any sequence of T 1D mask pairs creates (at most) a rank-T decomposition of a 2D light field matrix such that

$$L[i,k] = \frac{1}{T}\sum\nolimits_{t=1}^{T} f_t[i] \otimes g_t[k] = \frac{1}{T}\sum\nolimits_{t=1}^{T} f_t[i]g_t^T[k],$$

where $\boldsymbol{f}_t[i]$ and $\boldsymbol{g}_t[k]$ denote the rear and front masks for frame t, respectively. The light field matrix must be decomposed as the matrix product

$$L = FG^T,$$

where $\boldsymbol{F}$ and $\boldsymbol{G}$ are $N_i \times T$ and $N_k \times T$ matrices, respectively. Column t of $\boldsymbol{F}$ and column t of $\boldsymbol{G}$ are the masks displayed on the rear and front LCD panels during frame t.

3.2 Optimizing pixel states

Mask pairs $\{\boldsymbol{f_t}[i,j], \boldsymbol{g_t}[k,l]\}$ must be non-negative because they represent the pixel states of the LCDs. We aim for a light field factorization $\tilde{L}=FG^T$, which minimizes the weighted Euclidean distance to the target light field $\boldsymbol{L}$, under the necessary non-negativity constraints such that

$$\underset{F,G}{argmin}\,\frac{1}{2}\left\|\beta L - FG^T\right\|_W^2, \quad for \quad 0 \le F, G \le 1$$

The field of view can be adapted to multiple viewers by specifying elements of the weight matrix $\boldsymbol{W}$, a binary valued matrix that indicates which rays are constrained. The weight matrix ensures that a low-rank approximation obtains higher accuracy by artificially reducing the rank of the target light field. Brightness scaling factor β is used to control the tradeoff between image quality and brightness.

This expression can be solved using non-negative matrix factorization (NMF). We use the weighted multiplicative update rule, as described in past work.[6] Figure 1 represents typical mask pairs found by the NMF approach outlined here. Note that these masks are significantly more transmissive than pinhole arrays. In fact, although we can predict their structure, we cannot obtain them analytically. The masks create localized 1D parallax barriers, which follow the contours of the angular gradient of the light field.

3.3 Extending field of view through head tracking

The main challenge of dual-layer compressive light field displays is that only a small field of view and shallow depth of field can be achieved. Although these shortcomings can be alleviated using multiple stacked layers or directional backlighting,[10,11] this is either optically less efficient (stacks of LCDs block a lot of light) or more challenging to engineer. In this paper, we propose a simple technique to significantly increase the perceived field of view of a dual-layer display for a few (≤4) observers by flexibly steering a small field of view directly towards the observer(s).

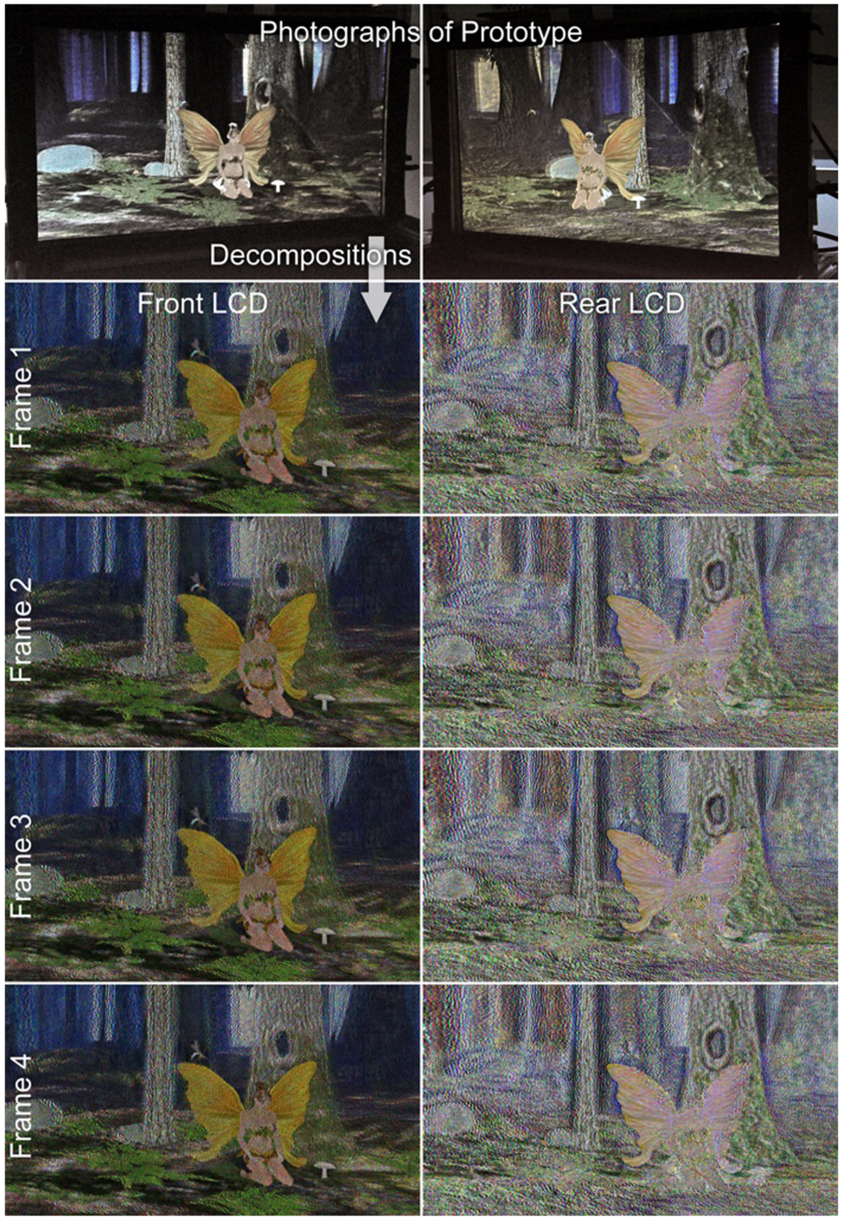


FIGURE 1 — Photographs of two perspectives of the 'fairy' scene (top). Each of the perspectives is dynamically decomposed into a set of four patterns for each liquid crystal display (LCD). These patterns are displayed at high speed on the respective LCD layers.

In other words, we constrain the optimization only for rays that reach a small area around the eyes of the viewer(s); all other rays are unconstrained. This is implemented by setting the weight matrix $\boldsymbol{W}$ (Section 3.2) to indicate only this set of rays. $\boldsymbol{W}$ is computed according to the eye positions of the viewers.

Figure 2 shows how this sparse binary matrix looks for two viewers, in comparison with that of the conventional light field display. The x-axis and y-axis represent the pixels on the front and the rear layer accordingly, that have been concatenated in the horizontal direction to form a long vector. Each blue point corresponds to a constraint for the light field decomposition, while white means there is no constraint for those combination of front and rear pixel pairs. In other words, each blue point indicates the (inner) product of the pair of pixels on the front, and rear masks should be constrained to match the corresponding ray in the target light field. To avoid clutter in the plot, we show the weight matrix for a light field with spatial resolution of 100×100. For the conventional display, we set its field of view to be 30° in both horizontal and vertical directions. As is evident from the figure, our tracked display has much sparser $\boldsymbol{W}$, which implies there are far less constraints on the light field decomposition. This is due to the simple fact that, for conventional light field, there is a constraint for every possible ray (inside the field of view), while for our tracked display, only those rays that are in the narrow view cones of the viewers' eyes need to be constrained.

As shown in Figure 3, this allows us to increase the observed field of view from about $10° \times 10°$ to $100° \times 40°$ for

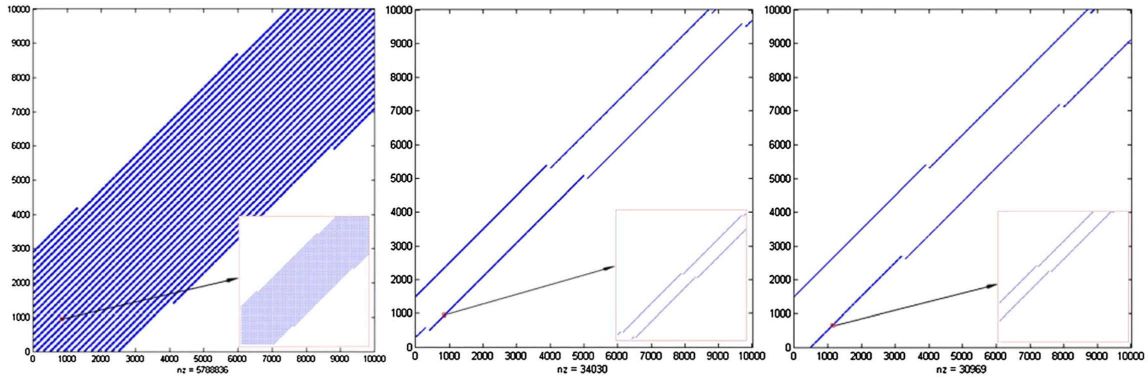


FIGURE 2 — Weight matrix W for the conventional light field display with 40×40 views (left) and our tracked multilayer display with four views at two sets of different view positions(middle and right). The number of nonzero is denoted by nz in the matrix, that is the number of constraints.

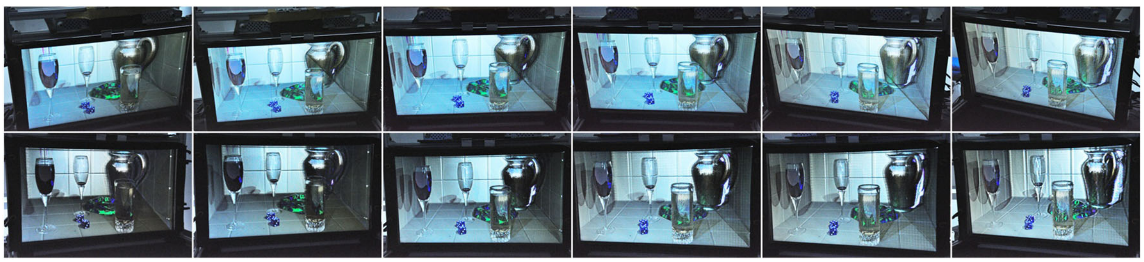
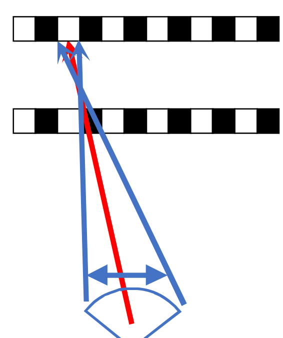


FIGURE 3 — Wide field of view compressive light field display. We extend the instantaneous field of view of a dual-layer light field display using head tracking. Whereas the instantaneous field of view provided by the dual-layer display is only about $10° \times 10°$, this can be flexibly steered into the direction of the observer without any hardware modifications. For this experiment, we achieve a perceived field of view of $100° \times 40°$ (please refer to the text regarding how the angles are measured) using the proposed techniques. Three dimensional model credit: Gilles Tran.

this experiment. We measure the field of view as the angle formed by two rays most oblique to the display from opposite directions (horizontally or vertically) that can be seen by two viewers simultaneously. The same eye tracking technique is applicable to other multilayer light field displays and is indifferent to the underlying head tracking mechanism. A Microsoft Kinect and vision-based tracking is used to find the 3D positions of the eyes. Although the field of view of our current system is limited by the (Field of View) FOV of a single Kinect, it can be trivially extended by using multiple Kinects. For results in this article that extend beyond the Kinect's field of view, we placed the camera capturing the display at a known position.

When tracking the eye, an important consideration is the size of the 'view cone' or region that the eye can move while still seeing the intended image. A larger view cone helps to address error and latency in the tracked eye position. In this article, we use a 'point' view cone for each view; that is, assuming we know the exact positions of the viewers' eyes, we constrain to the rays shooting from each of these exact positions. However, in practice, the discrete pixels of the display already equips us with a larger view cone for 'free', as one need to move a small amount in order to see different pixels on the rear layer through the same pixels on the front layer. The inset figure shows such a 'free' view cone. The size of the view cone can be estimated as $\frac{pp \times z}{d}$, where pp is the pixel pitch, z is the distance from the viewer to the display, and d is the layer spacing. For the experiments conducted in this paper, this gives a view cone of 0.7 cm width. In our experience, this view cone is big enough to cancel the error in the eye tracking system with moderate head movement speed. We can further increase the size of the view cones by adding constraints for rays shooting from more points around each view position. Of course, this will decrease the sparsity of the weight matrix $\boldsymbol{W}$ and therefore increase the complexity of layer factorization and reduce performance.



4 Implementation

In this section, we describe software and hardware implementation details of the proposed compressive display.

4.1 Hardware

Our prototype display consists of two stacked spatial light modulator layers and a tracking camera mounted on a metal frame. The spatial light modulator layers are LCD panels obtained from 27" Asus VG278H monitors which operate at 144 Hz with a resolution of 1920×1080. The LCD panels were removed from their housing and mounted on a frame with a spacing of 3.3 cm. The rear polarizer of the front LCD was removed, and the front polarizer of the front LCD was replaced with a clear polarizer so that there was an alternating set of crossed polarizers about the LCD stack. A mildly diffusing film was placed in front of the rear LCD panel to reduce the effects of moiré between the two panels. A Microsoft Kinect color plus depth camera was mounted above the LCD layers to provide 3D eye tracking of the viewer. The prototype display is illustrated in Fig. 4.

4.2 Software

Our prototype system is capable of real-time multilayer optimization that allows the display to be observed over a wide field of view by an eye-tracked user. The input light field for the optimization process can be pre-rendered or rendered in real-time on the GPU; Fig. 3 shows a pre-rendered scene raytraced using POV-Ray, while Fig. 1 shows a scene rendered in real-time using OpenGL.

4.2.1 Nonnegative matrix factorization

We implemented the factorization on GPU using OpenGL and the OpenGL shading language. To maximize the performance, we used several of the fast GPU rendering techniques, such as multiple rendering targets, rendering to texture, frame buffer object, and array textures. For each layer, all the time-multiplexed frames are stored in an array texture, and each layer of the array texture is bound to a rendering attachment of a frame buffer object. The multiplicative matrix update rules[6] can be easily implemented in OpenGL shading language. Appendix A shows the shader program used in our experiments. Because OpenGL does not allow reading and writing to the same texture simultaneously, we use the standard Ping-Pong scheme to update the time-multiplexed frames. For each layer, a single OpenGL pass is used to update all the time-multiplexed frames, so the OpenGL driver overhead can be minimized. Additionally, as we have multiple viewpoints, we perform the projective texture mapping inside the shader according to the viewpoints. Without further algorithmic optimization, we achieved an average of 4.44 ms per iteration, which allows us to drive the display at its fastest refresh rate (144 Hz) without multithreading.

At each frame, the initial state of the layer masks is seeded with the state of the previous frame before performing optimization. Because of the small change in the scene between frames, this allows the optimization to converge with few iterations. In most cases, the images do not show significant improvement after five iterations. Because multiple iterations/frames are needed for the optimization to converge, some artifacts are expected to show in the display; however, in practice, these artifacts are very subtle in our experiments.

Optimization was performed with four time-multiplexed frames and a brightness scaling factor of $\beta = 0.5$ (Section 3.2), while the weight matrix W was set to consider only the viewpoints around the viewer's eyes during optimization.

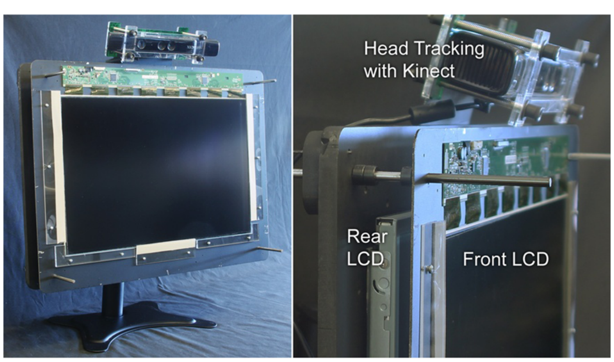


FIGURE 4 — Prototype compressive light field display. The device comprises two stacked, off-the-shelf liquid crystal displays and a Microsoft Kinect sensor that is used to estimate the position of the observer's eyes.

4.2.2 Eye tracking

To obtain the user's 3D eye positions, eyes were tracked in 2D using the FaceTracker library,[1] and the depth coordinate was obtained using the depth map from the Kinect depth camera. For the results shown in this paper, we used a printed face – recognized as a real face by the tracking system – to obtain the 3D positions at several locations in front of the display where a camera was placed to photograph the display. For the photographs, we constrained the optimization to only the exact position of a camera. However, for moving tracked user, the optimization can be constrained to a small view *cone* around the user's eyes, allowing the correct image to be seen in the presence of tracker error and latency. Using this technique, we were able to use the display with human viewers during informal testing.

4.2.3 Viewing angle dependent Gamma correction

The LCD panels have brightness and contrast that vary with viewing angle. Unfortunately, this effect is very pronounced with the high-speed Twisted Nematic-type LCD panels used in our system. The angular variation of the panels is further compounded when multiple layers are stacked. To alleviate this behavior, we adaptively change the brightness of each pixel according to the viewing angles. In our implementation,

[1]https://github.com/kylemcdonald/FaceTracker

we simply set the brightness scaling factor β using the following formula:

$$\beta = \beta_0 e^{\frac{\theta^2}{a^2} + \frac{\varphi^2}{b^2}}$$

where β_0 controls the overall brightness; θ and φ are the horizontal and vertical viewing angles, of either the whole screen or a single pixel, they can be derived from the positions of the viewpoints; a and b are parameters obtained by calibrating the screen from different viewing angles. Fig. 5 shows a single panel before and after the Gamma correction. This per-pixel gamma correction makes it more difficult to reduce the factorization energy, and our simulations show that it causes the PSNR of the factorization results to degrade as much as 0.6 dB.

5 Results

Figure 3 shows photographs of our prototype with the ‘glasses’ scene. A small instantaneous field of view showing the scene from the observer’s perspective is rendered and pixel states of the display optimized accordingly. The total observed field of view of this scene is 100° × 40° – a significant improvement over previously reported fields of view for light field displays. Fig. 1 shows photographs of the prototype for an additional scene. We also show the decomposed patterns displayed on the front and rear LCD for one of the perspectives. Whereas these patterns almost look like noise, when optically overlaid on the screen and displayed at a high refresh rate, the viewer perceives a consistent, high-resolution 3D image of the target scene.

Figure 6(a) shows the PSNR of the simulated views of our tracked display under different configurations for the ‘glasses’ scene. The NMF solver behaves similarly for our tracked display as in the conventional display: the image quality improves as rank increases, but the improvement is marginal when rank >6. Note that our tracked display gives good image quality (PSNR > 36) for two views, when rank >3. In Fig. 6(b), we compare with the conventional multilayer display. In this comparison, we only vary the field of view in the horizontal direction. For the conventional display, the number of views increases linearly from 8 for FOV = 5° to 216 for FOV = 110°. For both the conventional display and our tracked display, we set the distance from the viewer to the display to be 75 cm and β the brightness scaling to be 0.5. From Fig. 6(b), we can see that the image quality of the conventional display quickly deteriorates as the field of view increases, while our tracked display provides the same image quality for wider fields of view. In fact, the main challenge for us to get a wider field of view (>110) is the angular variation issue of the LCD panels that we use.

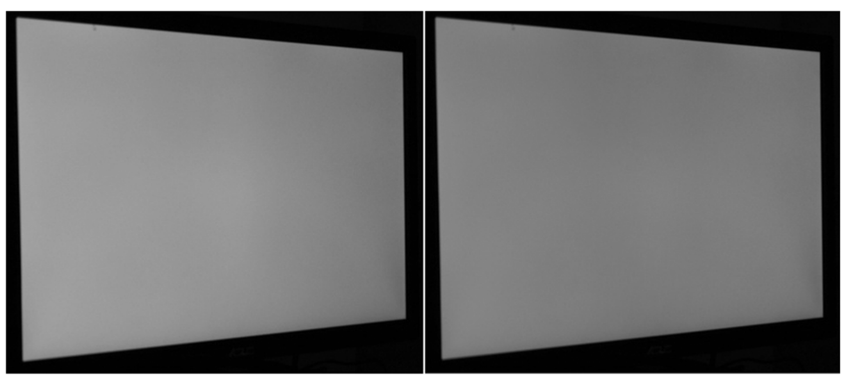


FIGURE 5 — Photographs of a liquid crystal display screen before and after viewing angle dependent Gamma correction.

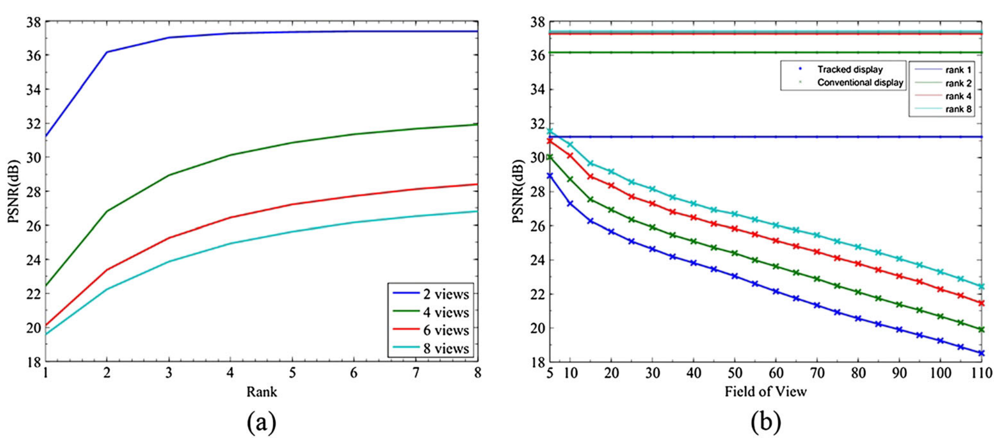
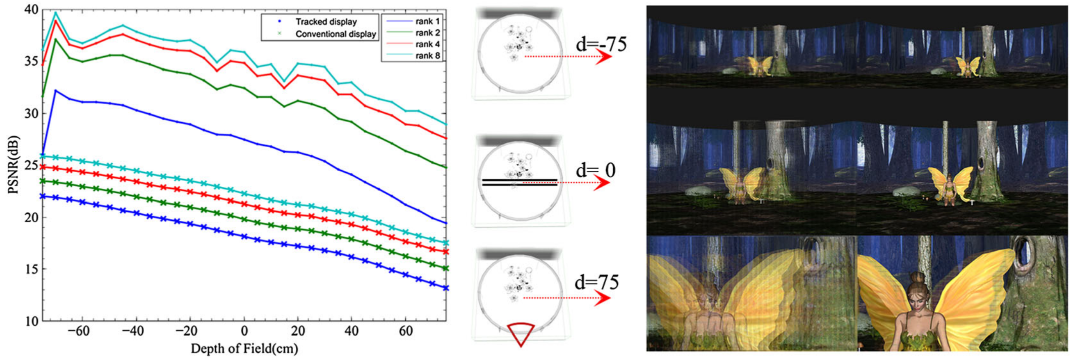


FIGURE 6 — (a) PSNR measure of simulated views of our tracked display under different configurations; (b) comparison of PSNR-FOV (Field of View) relationship in conventional and tracked display configurations.



(a) (b) (c) (d)

FIGURE 7 — Comparison of our tracked display and the conventional multilayer display for the 'forest' scene under different depth ranges: (a) shows the PSNR measure of the simulated views of conventional display and tracked display with different depth of field and different ranks; (b) show how the virtual scene is positioned w.r.t the display; (c) and (d) show the simulated views of conventional display and our tracked display at three different depth accordingly.

In Fig. 7, we compare the image quality of our tracked display and a conventional display for the 'fairy' scene under different depth ranges to measure the displays' depths of field. The virtual scene can be placed anywhere as long as it is visible to the camera. We measure the depths of the field as the distance from the scene center to the display center. In Fig. 7(a), we show the PSNR measure for varying depth ranges for both conventional multilayer and our tracked displays. In Fig. 7(b), we show how we place the virtual scene with respect to the display. In Fig. 7(c) and (d), we show the simulated views of the conventional display and our tracked display with rank = 4, and the scene center is placed at −75 cm, 0 cm and 75 cm away from the center of the display accordingly. The distance between the viewer and the display is 75 cm. Like the conventional display, the image quality of our tracked display decreases as scene depth increases. However, with sufficient rank (≥4), our tracked display has generally acceptable image quality (>30 dB) for the tested depth range.

6 Discussion

In summary, we explore the combination of compressive light field displays and head tracking. By steering a small instantaneous field of view dynamically into the direction of a single tracked observer, we demonstrate how the perceived field of view of the display is significantly improved.

Our current prototype exhibits artifacts, which can mostly be attributed to limitations in the precision of LCD panel alignment and the limitation of the current NMF solver, that it is essentially gradient descent and thus can be easily trapped to local optimums. The display is also rather dim due to the brightness scaling, which could be enhanced by a strong uniform backlighting. The depth of field of each perspective is limited by the panel refresh rate. Artifacts observed for objects that extend from the physical display plane, such as the jug on the right-hand side of the 'glasses' scene in Fig. 3, could be mitigated by using faster LCD panels or other high-speed light modulators, or a better NMF solver.

In the future, we would like to explore display settings that facilitate wide fields of view for multiple observers. Currently, the most practical solution for this problem would be high-speed light modulators.

Acknowledgments

Gordon Wetzstein was supported by a NSERC PDF scholarship. This research was supported in part by the BeingThere Centre, collaboration between UNC Chapel Hill, ETH Zurich, and NTU Singapore, supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the Interactive Digital Media Programme Office. This project was also supported by Samsung's Global Research Outreach program.

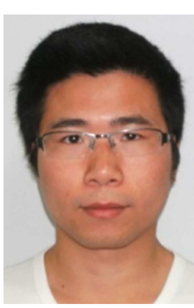
Appendix A. GLSL shader for Nonnegative Matrix Factorization

```
const float halfLayerSpacing =     3.3f/2 // half of the spacing between the 2 layers, in cm
const float pixelPitch =       0.0311f; // size of pixels in cm
const ivec2 NMFSize =            ivec2(1920, 1080);   // size of the input images, in pixels
const int maxViews = 8;

uniform float gainMul; // 0.5f, adjust the overall brightness of the reconstructed light field, for better image quality
uniform int nFrame; // 4, time-multiplexer
uniform int nViews; // 4, number of views

uniform sampler2DArray LayerF;        // time multiplexed patterns on the front layer
uniform sampler2DArray LayerG;          // time multiplexed patterns on the rear layer
uniform sampler2DArray LayerPrev; // LayerF/LayerG from last iteration
uniform sampler2DArray srcimages; // input to NMF
uniform sampler2DArray recimages; // reconstructed images
uniform vec3 eyepositions[maxViews];

vec2 mapView2LayerF(ivec2 p, vec3 epos) {return p - halfLayerSpacing*(p-NMFSize/2.-epos.xy/pixelPitch)/epos.z;}
vec2 mapLayerF2View(ivec2 p, vec3 epos)     {return p + halfLayerSpacing*(p-NMFSize/2.-epos.xy/pixelPitch)/(epos.z-hbarrierSep);}
vec2 mapLayerG2LayerF(ivec2 p, vec3 epos) {return p + 2*halfLayerSpacing*(p-NMFSize/2.-epos.xy/pixelPitch)/(epos.z-hbarrierSep);}
vec2 mapView2LayerG(ivec2 p, vec3 epos) {return p + halfLayerSpacing*(p-NMFSize/2.-epos.xy/pixelPitch)/epos.z;}
vec2 mapLayerG2View(ivec2 p, vec3 epos) {return p - halfLayerSpacing*(p-NMFSize/2.-epos.xy/pixelPitch)/(epos.z + hbarrierSep);}
vec2 mapLayerF2LayerG(ivec2 p, vec3 epos) {return p - 2*halfLayerSpacing*(p-NMFSize/2.-epos.xy/pixelPitch)/(epos.z + hbarrierSep);}
vec3 gain(vec3 c)       {return c*gainMul;      }
vec3 ungain(vec3 c)  {return c/gainMul;      }

// update the reconstructed images
void update_F_Times_G(){
    ivec2 pos = ivec2(gl_FragCoord.xy);
    for(int i = 0; i < nViews; i++){
        vec2 fpos = (mapView2LayerF(pos, eyepositions[i]) + 0.5)/NMFSize;
        vec2 gpos = (mapView2LayerG(pos, eyepositions[i]) + 0.5)/NMFSize;
        vec3 col = vec3(0);
        for(int j = 0; j < nFrame; j++)
            col + = texture(LayerF, vec3(fpos, j)).rgb*texture(LayerG, vec3(gpos, j)).rgb;
        gl_FragData[i].rgb = ungain(col);
    }
}

void update_Layer_F(){
    ivec2 pos = ivec2(gl_FragCoord.xy);
    for (int r = 0; r < nFrame; r++){
    vec3 s = vec3(0), t = vec3(0);
    for (int i = 0; i < nViews; i++){
        vec2 ppos1 = (mapLayerG2LayerF(pos, eyepositions[i]) + 0.5)/NMFSize;
        vec2 ppos2 = (mapLayerF2View(pos, eyepositions[i]) + 0.5)/NMFSize;
        vec3 col = texture(LayerG, vec3((mapLayerG2LayerF(pos, eyepositions[i]) + 0.5)/NMFSize, r)).rgb;
        s + = col * gain(texture(srcimages, vec3(ppos2, i)).rgb);
        t + = col * gain(texture(recimages, vec3(ppos2, i)).rgb);
      }
      gl_FragData[r].rgb = min(vec3(1,1,1), texture(LayerPrev, vec3(vec2(pos + 0.5)/NMFSize, r)).rgb*s / t);
    }
}
```

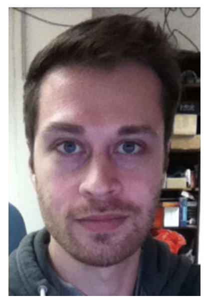
```
void update_Layer_G(){
    ivec2 pos = ivec2(gl_FragCoord.xy);
    for (int r = 0; r < nFrame; r++){
      vec3 s = vec3(0), t = vec3(0);
      for (int i = 0; i < nViews; i++){
        vec2 ppos1 = (mapLayerF2LayerG(pos, eyepositions[i]) + 0.5) / NMFSize;
        vec2 ppos2 = (mapLayerG2View(pos, eyepositions[i]) + 0.5) / NMFSize;
        vec3 col = texture(LayerF, vec3((mapLayerF2LayerG(pos, eyepositions[i]) + 0.5)/NMFSize, r)).rgb;
        s + = col * gain(texture(srcimages, vec3(ppos2, i)).rgb);
        t + = col * gain(texture(recimages, vec3(ppos2, i)).rgb);
      }
      gl_FragData[r].rgb = min(vec3(1,1,1), texture(LayerPrev, vec3(vec2(pos + 0.5)/NMFSize, r)).rgb*s / t);
    }
}
```

References

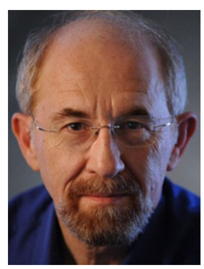
1 H. Urey *et al.*, "State of the Art in stereoscopic and autostereoscopic displays," *Proc. IEEE* **99**, No. 4, 540–555 (2011).
2 D. Fattal et al., "A multi-directional backlight for a wide-angle, glasses-free three-dimensional display," *Nature* **495**, 348–351 (2013).
3 Y. Kim *et al.*, "Point light source integral imaging with improved resolution and viewing angle by the use of electrically movable pinhole array," *Opt. Express* **15**, No. 26, 18253–18267 (2007).
4 A. Travis, "Autostereoscopic 3-D display," *Appl. Opt.* **29**, 4341–4342 (1990).
5 H. Gotoda, "Design of time-multiplexed auto-stereoscopic displays based on virtual stacking of multi-layer panels," Proc. SPIE 8648 (2013).
6 D. Lanman *et al.*, "Content-adaptive parallax barriers: optimizing dual-layer 3D displays using low-rank light field factorization," *ACM Trans. Graph. (SIGGRAPH Asia)* **29**, 163:1–163:10 (2010).
7 D. Lanman *et al.*, "Polarization fields: dynamic light field display using multi-layer LCDs," *ACM Trans. Graph. (SIGGRAPH Asia)* **30**, 1–9 (2011).
8 N. Ranieri *et al.*, "Light-field approximation using basic display layer primitives," *Proc. SID* **44**, No. 1, 408–411 (2013).
9 G. Wetzstein *et al.*, "Layered 3D: tomographic image synthesis for attenuation-based light field and high dynamic range displays," *ACM Trans. Graph. (SIGGRAPH)* **30**, 1–11 (2011).
10 A. Maimone *et al.*, "Focus 3D: compressive accommodation display," *ACM Trans. Graph.* **32**, No. 153, 1–13 (2013).
11 G. Wetzstein *et al.*, "Tensor displays: compressive light field synthesis using multilayer displays with directional backlighting," *ACM Trans. Graph. (SIGGRAPH)* **31**, 1–11 (2012).
12 A. Nashel and H. Fuchs "Random Hole Display: A non-uniform barrier autostereoscopic display". 3DTV Conference, (2009).



Renjie Chen is a postdoctoral research associate at the Department of Computer Science of the University of North Carolina at Chapel Hill. He is a key researcher at the BeingThere international research centre, Singapore. His current research focuses on high-quality real-time glasses-free 3D display. He received his BSc and PhD in Applied Mathematics from Zhejiang University, China, in 2005 and 2010, respectively. His research interests includes computer graphics, glasses-free 3D display, geometry processing, computational geometry and geometric modeling.



Andrew Maimone is a fifth year PhD student studying near-eye and 3D displays under Prof Henry Fuchs at the University of North Carolina at Chapel Hill. He is a member of the BeingThere International Research Centre group. His current research focuses on the development of new displays which use simple optics and powerful and flexible software to obtain new functionality, with emphasis on the development of augmented reality eyeglasses and 3D displays. Andrew obtained his MS from the University of North Carolina in 2012, his BS from the Rochester Institute of Technology in 2006, and formerly worked at Boeing Space and Intelligence Systems.



Henry Fuchs is the Federico Gil Distinguished Professor of Computer Science and Adjunct Professor of Biomedical Engineering at UNC Chapel Hill. He has been active in computer graphics since the early 1970s, with rendering algorithms (BSP Trees), hardware (Pixel-Planes and PixelFlow), virtual environments, tele-immersion systems, and medical applications. He received his PhD in 1975 from the University of Utah. From 1975 to 1978, he was an assistant professor at the University of Texas at Dallas. Since 1978, he has been on the faculty at UNC Chapel Hill. He is a member of the National Academy of Engineering, a fellow of the American Academy of Arts and Sciences, the recipient of the 1992 ACM-SIGGRAPH Achievement Award, the 1992 Academic Award of the National Computer Graphics Association, the 1997 Satava Award of the Medicine Meets Virtual Reality Conference, and the 2013 IEEE-VGTC Virtual Reality Career Award.

Ramesh Raskar joined the Media Lab from Mitsubishi Electric Research Laboratories in 2008 as head of the Lab's Camera Culture research group. His research interests span the fields of computational photography, inverse problems in imaging, and human-computer interaction. Recent projects and inventions include transient imaging to look around a corner, a next generation CAT-Scan machine, imperceptible markers for motion capture (Prakash), long distance barcodes (Bokode), touch + hover 3D interaction displays (BiDi screen), low-cost eye care devices (Netra,Catra), new theoretical models to augment light fields to represent wave phenomena and algebraic rank constraints for 3D displays (HR3D). In 2004, Raskar received the TR100 Award from Technology Review, which recognizes top young innovators under the age of 35 years, and in 2003, the Global Indus Technovator Award, instituted at MIT to recognize the top 20 Indian technology innovators worldwide. In 2009, he was awarded a Sloan Research Fellowship. In 2010, he received the Darpa Young Faculty award. Other awards include Marr Prize honorable mention 2009, LAUNCH Health Innovation Award, presented by NASA, USAID, US State Dept and NIKE, 2010, Vodafone Wireless Innovation Project Award (first place), 2011. He holds over 40 US patents and has received four Mitsubishi Electric Invention Awards. He is currently co-authoring a book on Computational Photography.

Gordon Wetzstein Prior to joining Stanford University's Electrical Engineering Department as an Assistant Professor in 2014, Gordon Wetzstein was a Research Scientist in the Camera Culture Group at the MIT Media Lab. His research focuses on computational imaging and display systems as well as computational light transport. At the intersection of computer graphics, machine vision, optics, scientific computing, and perception, this research has a wide range of applications in next-generation consumer electronics, scientific imaging, human-computer interaction, remote sensing, and many other areas. Gordon's cross-disciplinary approach to research has been funded by the Defense Advanced Research Projects Agency, National Science Foundation, Samsung, Intel, and other grants from industry sponsors and research councils. In 2006, Gordon graduated with Honors from the Bauhaus in Weimar, Germany, and he received a Ph.D. in Computer Science from the University of British Columbia in 2011. His doctoral dissertation focuses on computational light modulation for image acquisition and display and won the Alain Fournier Ph.D. Dissertation Annual Award. He organized the IEEE 2012 and 2013 International Workshops on Computational Cameras and Displays, founded displayblocks.org as a forum for sharing computational display design instructions with the DIY community, and presented a number of courses on Computational Displays and Computational Photography at ACM SIGGRAPH. Gordon won best paper awards at the International Conference on Computational Photography in 2011 and 2014 as well as a Laval Virtual Award in 2005.