Multi-planar plenoptic displays

Nicola Ranieri (SID Student Member)
Simon Heinzle
Peter Barnum
Wojciech Matusik
Markus Gross

Abstract — Multi-planar plenoptic displays consist of multiple spatially varying light-emitting and light-modulating planes. In this work, we introduce a framework to display light field data on this new type of display device. First, we present a mathematical notation that describes each of the layers in terms of the corresponding light transport operators. Next, we explain an algorithm that renders a light field with depth into a given multi-planar plenoptic display and analyze the approximation error. We show two different physical prototypes that we have designed and built: The first design uses a dynamic parallax barrier and a number of bi-state (translucent/opaque) screens. The second design uses a beam splitter to co-locate two pairs of parallax barriers and static image projection screens. We evaluate both designs on a number of different 3D scenes. Finally, we present simulated and real results for different display configurations.

Keywords — *multi-planar plenoptic display, light field decomposition.*

DOI # 10.1002/jsid.201

1 Introduction

Glasses-free 3D displays have experienced a major renaissance in the past few years. In general, these displays can be divided into two main categories: parallax-based displays and volumetric displays. Parallax-based approaches such as integral imaging[1] and parallax barriers[2] redirect spatially varying pixels onto different viewing directions. These approaches trade off spatial resolution in favor of angular resolution, which directly relates to the depth range that can be displayed without aliasing.[3] Although being capable of view-dependent effects and proper occlusions, these devices often exhibit low spatial or angular resolution, and lack correct accommodation cues. Volumetric displays,[4–6] on the other hand, physically deploy light-emitting voxels in 3D space and provide a direct and natural approximation of the input scene. They provide correct accommodation cues but are, with few exceptions, not capable of providing proper occlusions or view-dependent effects. A comprehensive overview of volumetric displays is given in the work of Favalora.[7]

As alternative, multi-planar displays[8–12] have been suggested to provide natural ways to show 3D scenes at nearly correct accommodation cues with increased display bandwidth and hence higher angular and spatial resolution. In essence, these displays combine parallax and volumetric displays and draw benefits from both. In our work, we generalize these concepts to multi-planar plenoptic displays by defining basic display layer primitives, consisting of emissive and modulating layers. We provide a mathematical framework to describe light transport through any combination of such layers. Based on this framework, we then provide a method to distribute an input light field according to a given display configuration. Furthermore, a quantitative error analysis for different layer configurations is provided. Finally, we present two physical prototypes, capable of rendering volumetric content with view-dependent effects, proper occlusions, and better accommodation cues; and we show results for both of them.

2 Mathematical framework

Our model assumes co-planar display layers, which are aligned with the xy plane. Each of the layers can either be emitting or modulating, performing a certain operation on the overall light transport, which will be described in the following.

Our model is using a similar notation and concepts as presented by Durand *et al.*[13] In general, the light field ℓ describes the radiance of light rays passing through points (x,y) and (u,v) at distance z from the xy plane and is denoted as $\ell(x,y,z,u,v)$. For simplicity, we will only consider light rays traveling along the positive z direction, as our displays will only be viewed from the front.

2.1 Light transport in free space

The basic light transport is illustrated in Fig. 1 (middle). A ray starting at position (x,y,z) passing through (u,v) traverses in free space to

$$x' = x + \Delta z \cdot u, y' = y + \Delta z \cdot y$$

As the ray moves in depth, its position will change to $(x',y', z + \Delta z)$ while keeping its original traveling direction (u,v).

Received 07/29/13; accepted 12/11/13.
N. Ranieri and M. Gross are with Computer Graphics Laboratory, ETH Zürich, Zürich, Switzerland.
S. Heinzle, P. Barnum and M. Gross are with Disney Research Zürich, Zürich, Switzerland.
W. Matusik is with the CSAIL, MIT, Cambridge, MA, USA; e-mail: ranieri@inf.ethz.ch.

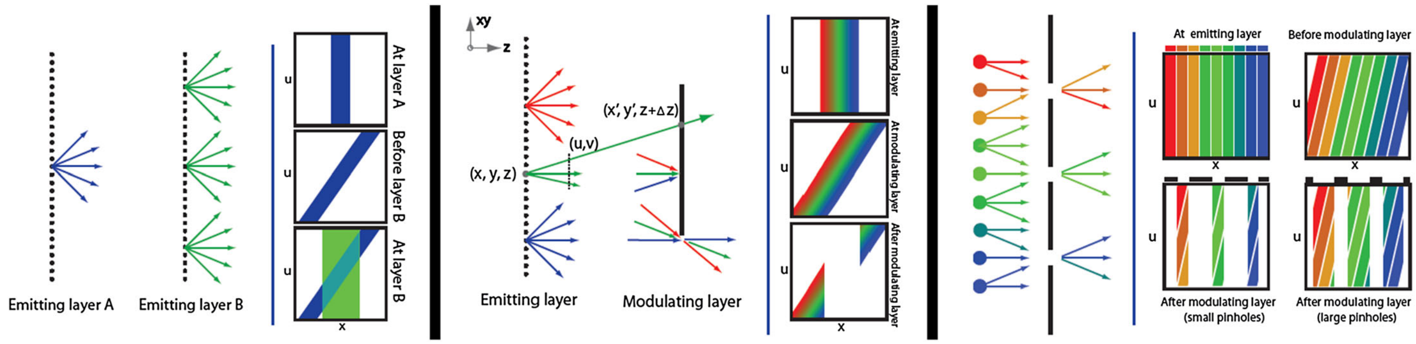

FIGURE 1 — The three basic display layer primitives: the emissive layer, modulating layer, and the parallax barrier primitive. The effect of two emissive layers is shown on the left: each layer consists of multiple diffuse pixels that emit light uniformly across all directions *u*. Our model assumes transparent, non-blocking pixels, and the ray content of all layers is blended additively. The impact of a modulating layer combined with an emissive layer is shown in the middle: the uniformly emitted rays from the emissive layer are blocked by the modulating layer in all directions *u*, at the respective spatial positions *x*. The principle of a parallax barrier layer is shown on the right: Different pixels from an emissive layer are multiplexed to different directions.

2.2 Display layer primitives

Many of the display layer types deployed in modern systems can be generalized to two categories. We describe both categories as basic display primitive and provide a mathematical notation for the light transport operator. Together with the notation of light transport in free space, more complex systems made of any combination of such display primitive layers can be described.

2.3 Emissive layer

An emissive layer E acts like an array of point light sources, emitting constant spherical light. We use the notation $Ez(x,y)$ as the light portion at x and y on the plane at depth z, radiating into all directions (u,v). The emissive layer adds light to an input light field ℓi and yields the output light field ℓo, as illustrated in Fig. 1 on the left:

$$\ell o(x,y,z,u,v) = \ell i(x,y,z,u,v) + Ez(x,y)$$

Opaque emissive layers can be found in any 2D display consisting of, for example, a backlight with a modulating color liquid crystal display (LCD). However, to optically overlay them with other display layer primitives, transparent emitters are preferred. This can be implemented using the upcoming transparent organic light-emitting diode technology, by transparent back-projection foils or, as in our prototype, polymer dispersed liquid crystal (PDLC) layers in combination with a projector.

2.4 Modulating layer

A spatial modulating plane M will gradually attenuate all rays (u,v) passing through a certain pixel (x,y). The modulating layer $Mz(x,y)$ is therefore represented as scalar between zero and one, and the output light field can be described as follows:

$$\ell o(x,y,z,u,v) = \ell i(x,y,z,u,v) \cdot Mz(x,y)$$

The operation is shown in Fig. 1 in the middle. Modulating layers can be implemented by (grayscale) LCDs, as the polarization rotation capability of twisted nematic liquid crystals can be used to block light when combined with two polarizers.

2.5 Parallax barrier layer

Parallax barrier displays are basically a combination of an emissive layer and a modulating layer with a special modulating pattern and small spacing Δz between the two layers. Therefore, the same light transport operators as for above layers are used. As this pairing is fundamental, we define it as third basic primitive shown in Fig. 1 on the right. The modulating layer is used to achieve ray separation by displaying a vertical slit, diagonal slit, or pinhole pattern, while the emissive layer displays the different rays that pass through the slits/pinholes. As a consequence, an observer will see different rays from different directions. N pixels on the emitter plane can be partitioned into any number of spatial and angular samples (x,y,u,v), such that $N > x \cdot y \cdot u \cdot v$. In practice, such displays trade a substantial reduction of spatial resolution for a relatively small amount of rays and apparent depth. The work of Levin *et al.*[14] provides a good background on this subject.

3 Light field decomposition

Based on the analysis in the previous section, we will describe an algorithm that approximates an input light field $\ell i(x,y,z,u,v)$ as an output light field $\ell o(x,y,z,u,v)$ targeted for a given multi-layer plenoptic display Di. Our algorithm decomposes the input light field into a number of components. Each component is then displayed on one or multiple display primitives. In order to aid the decomposition process, we assume that for all rays (x,y,z,u,v) of the input light field, we know the depth z of the closest object, the diffuse component $RD \subseteq \ell i$, and the specular/glossy component $RS \subseteq \ell i$.

More specifically, the algorithm separates the light field data into planar components. First, volumetric components ℓE, which are visible from at least one viewing angle, are extracted from the light field and assigned to the closest emissive layer. Next, partial occlusions ℓM between elements of ℓE are computed and assigned to a proper modulating layer. Finally, the remaining light field $\ell residue$ is extracted for rendering with parallax barrier layers. Algorithm 1 gives a high level overview of how to generate $\ell o(x,y,z,u,v)$ from an input light field $\ell i(x,y,z,u,v)$ for a given display setup *Di*. Example decomposition is given in Fig. 2, and exemplary components assigned to the different layers are shown in Fig. 3.

Elements shown on different emissive layers can overlap from certain viewing angles. As described in the previous section, the light contribution of each emissive layer would add up, leading to a ghost-like see-through effect as visible in the first two images of Fig. 4. Hence, in a second step shown in Algorithm 3, all such overlaps are detected, and proper occlusion masks are computed. Optimally, each emissive layer is preceded by a modulating layer to provide perfect occlusions. However, as in practice modulating layers often absorb much light also in their transparent state, a fewer number of modulating layers are desired. For each emissive pixel, overlaps for all emissive layers in front are detected, and the modulating layer closest to but behind the emissive layer in front is used for masking. The occlusion mask is retrieved by intersecting the ray from the emissive pixel to its overlapping pixel with the chosen modulating layer. This step creates black borders (illustrated in the third image in Fig. 4) because occlusions are detected conservatively over the whole viewing angle. Therefore, such "over-occlusions" have to be added back in a later step and are thus subtracted from ℓE.

Algorithm 1: High level overview of the rendering algorithm

ℓE ← assignEmissiveElements(ℓi, Di)
ℓM ← assignOcclusions(ℓE, Di)
ℓresidue ←assignResidue(ℓi, ℓE, Di)
*ℓo ← ℓE*ℓM+ ℓresidue*

In a first step, all diffuse components that are visible from at least one viewing angle are extracted from the light field. The extracted components are then distributed to the available emitting layers. Only components that are spatially close enough to the emitting layers are considered for display, to minimize the re-projection error. More specifically, each part of *RD* that is within a distance *zthresh* from any layer is assigned to the nearest layer in the display setup *Di*. Assignment is performed by parallel projection. The parts of *RD* that are further than *zthresh* from all layers are not processed and thus left as residue for the automultiscopic display layers.

Algorithm 2: *ℓE ← assignEmissiveElements(ℓi, Di)*

for ∀emissiveLayer ϵ *Di*
  for ∀x, ∀y, ∀z ϵ ℓi
    dz ← distance([x, y, z], emissiveLayer)
    //check if the element is opaque, close enough and visible from any viewpoint
    if ℓi.RD(x, y, z) ≠ 0 **and** dz < zthresh **and** isVisible(x, y, z)
      //assign the diffuse color to the layer
      emissiveLayer[x, y] ← ℓi.RD(x, y, z)
      //remember the color and the original depth for later mapping of specular components
      ℓ*E*[x, y, emissiveLayer.z].add(ℓi.RD, z])

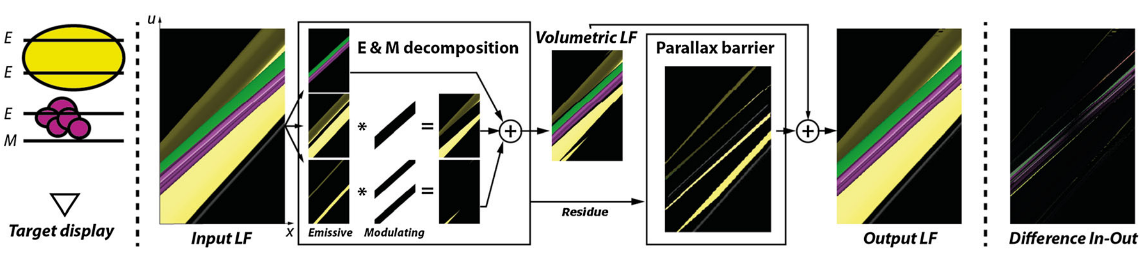

FIGURE 2 — Illustration of the decomposition algorithm for the display layer configuration shown on the left (frontmost modulating layer followed by three emissive layers). First, the volumetric emissive elements are assigned, and then the modulating occlusions are computed, and finally the residue is added using parallax barrier rendering. In this display configuration, the content of each emissive layer is shown using time multiplexing, as all three layers share one modulating layer. The parallax barrier layer also uses the frontmost modulating and emissive layer, and thus it requires its own time slot too.

Algorithm 3: *ℓM ← assignOcclusions(ℓE, Di)*

```
ℓE'= ℓE
for ∀x, ∀y, ∀z ϵ ℓE
  for ∀x', ∀y', ∀z' ϵ ℓE
    //check if one element is in front of the other, both are opaque and the front one occludes the back one
    if z' > z and ℓE.RD(x, y, z) ≠ 0 and ℓE.RD(x', y', z') ≠ 0 and occludes([x', y', z'],[x, y, z])
      //get the closest modulator behind the front element
      modulatingLayer ← getClosestModulatorBehind([x', y', z'], Di)
      //intersect the ray defined by the two elements with the modulator and mask the intersected pixel black
      mx,my ← modulatingLayer.intersect([x, y, z], [x', y', z'])
      modulatingLayer[mx, my] ← 0
      //remove the element in the back so it can be added a later step
      ℓE'[x, y, z].remove()
      ℓM[mx, my, modulatingLayer.z].add()
ℓE= ℓE'
```

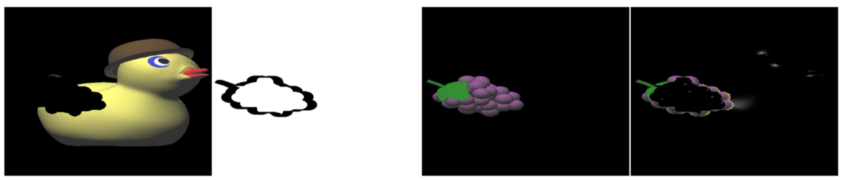

FIGURE 3 — The images to be displayed on each display layer primitive, as computed by our algorithm for a layer configuration of two emissive layers, followed by a modulating layer (similar to Fig. 8). The modulating layer and the backmost emissive layer are further used as parallax barrier primitive. The content of the individual primitives is shown from left to right: backmost emissive layer, frontmost modulating layer, center emissive layer, and parallax barrier layer. The emissive and modulating layers show the view-independent content with view-dependent, approximate occlusion. The parallax barrier layer then adds view-dependent content (residue) back into the image. The parallax barrier shares the same emissive and modulating layers, and thus uses time multiplexing. Note that the modulating layer is multiplicative where a black color denotes full light-blocking, whereas white denotes no modulation. The modulation pattern for the barrier is omitted here.

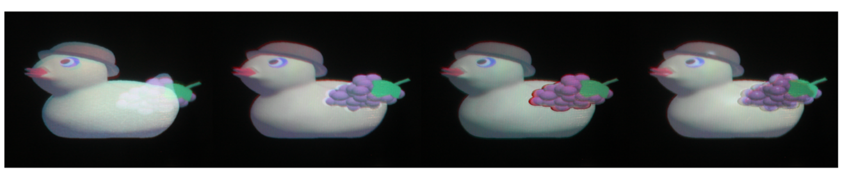

FIGURE 4 — Illustration of the steps of our light field decomposition: The first image shows results using volumetric rendering (using the emissive layers only) without any occlusion handling. The second image illustrates volumetric rendering with approximate occlusion handling. In this mode, only diffuse components that are visible from at least one angle are displayed. The third image shows our conservative occlusion culling for overlapping regions using both emissive and modulating layer. In the last image, the light field residue including specular highlights is overlaid using a parallax type layer. Photographs are taken from our multi-layered display prototype.

In a last step, the residue has to be rendered, filling the black borders and adding other light field portions as specular highlights. Because of the planar mapping to the emissive layers, the holes cannot be filled naively. Our solution stretches the occluded residual to match the gap borders as illustrated on the left-hand side of Fig. 5, which corresponds to a scale in depth as shown on the right-hand side. The same has to be considered for the glossy parts: They are mapped to the same plane as the diffuse part before being rendered. All components of this third step shown in Algorithm 4 are rendered using the closest parallax barrier layer and have to be filtered accordingly using existing approaches.[15] A result of the complete algorithm is shown on the right of Fig. 4.

4 Decomposition analysis

We analyze our system with respect to the projective error, the error introduced by the sampling in the light field layer, and a quantitative error measurement.

Algorithm 4: *ℓresidue ← assignResidue(ℓi, ℓE, Di)*

```
for ∀x, ∀y, ∀z, ∀u, ∀v є ℓi
    //check if the diffuse component is already assigned to an emissive layer
    if ℓE.RD(x, y, z) ≠ 0
        //map the specular component to the same depth as its diffuse part in ℓE
        x', y', z', u', v' ← mapTo(ℓi.RS(x, y, z, u, v), ℓi.RS(x, y, ℓE.z(x, y, z), u, v))
        ℓresidue[x', y', z', u', v'].add(0, ℓi.RS(x, y, z, u, v))
    else
        //fill holes according to Figure 5
        x', y', z', u', v' ← stretchAccordingToGapBorder(x, y, z, u, v)
        ℓresidue[x', y', z', u', v'].add(ℓi.RD(x, y, z), ℓi.RS(x, y, z, u, v))
for ∀x, ∀y, ∀z, ∀u, ∀v є ℓresidue
    lightfieldLayer ←getClosestLightFieldLayer(z)
    lightfieldLayer[x, y, u, v] ← ℓresidue.RD(x, y, z) + ℓresidue.RS(x, y, z, u, v)
```

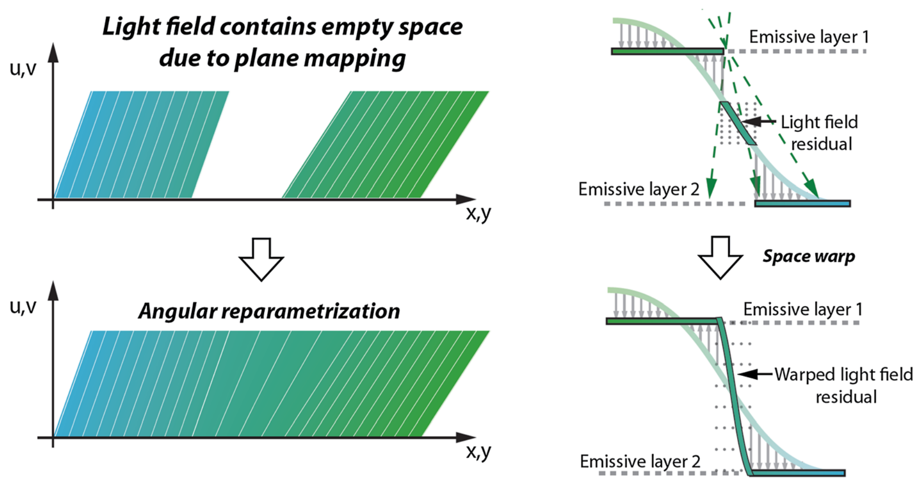

FIGURE 5 — Adaptive hole filling between emissive layers.

4.1 Projective error

The projection onto the planar emitters inherently produces an approximation of the motion parallax. The motion parallax produced by an object at distance z to a viewer with focal length f that moves along a baseline at a distance b can be expressed as

$$d = -f \cdot b/z$$

We express the relative projective approximation error of an object at distance z projected on a plane at distance $z0$ by

$$e(z, z0) = |1/z0 - 1/z|$$

The emissive layers should therefore ideally be placed near dense occurrences of objects in depth, and the occluding layers should be placed as near as possible to the respective layers that need occlusion. Furthermore, fewer display elements are needed the farther the scene is with respect to the viewer's position. This error can be used to determine the optimal display configuration using a suited optimization method, in cases where a display configuration is optimized for a given scene.

4.2 Light field sampling error

Light field layers usually trade off spatial against angular resolution, and the approximation error is directly proportional to the loss in spatial resolution. However, additional errors are introduced if aliasing occurs when the angular frequencies are too high. These problems can be overcome by either using time multiplexing for the parallax barrier display or combining multiple parallax barrier displays superimposed onto the same optical path. The latter approach has further the advantage, that the required bandwidth for each light field layer is decreased by $1/n^2$, with n the number of layers, resulting in a total required bandwidth of $1/n$, as described in the work of Ranieri *et al*.[10]

4.3 Quantitative error

To analyze the impact of number of layer primitives, we compare the resulting re-projection errors using our software simulation. We simulated two different scenes: a duck scene containing two objects at different depths with occlusion and a bust scene depicting a continuous surface, shown in Fig. 6. Both scenes contain a small amount of specular highlights. The simulated results are compared to a perfect rendering, and the mean square error between the simulated and perfect images is computed for a number of views in a field of view of 15°. The resulting error plots are shown in Fig. 7.

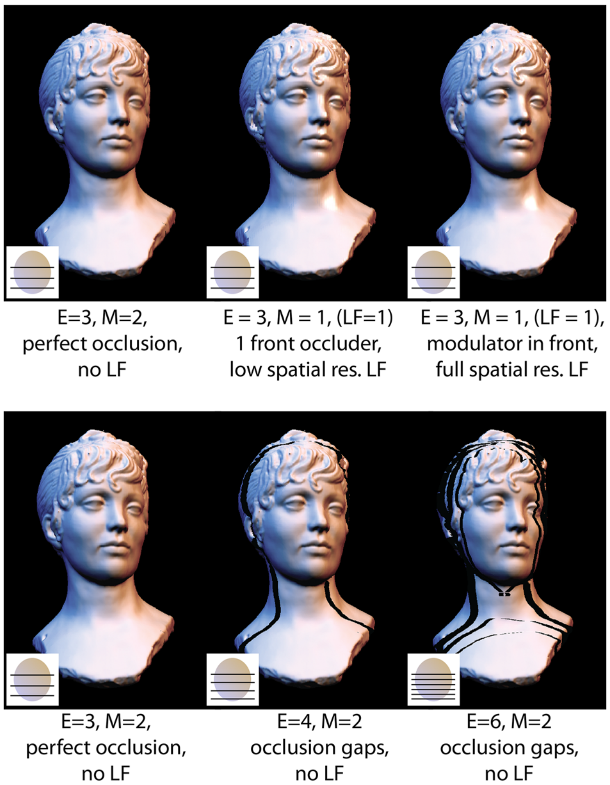

FIGURE 6 — Simulated results used for the quantitative error analysis. Layer configurations are shown in the insets. Modulating layers are added behind and adjacent to the emissive layers, starting from front to back with increasing numbers of modulators.

In the left plot, the impact of an increasing number of emissive layers is depicted for three cases: In the first case, each emissive layer is preceded by a modulating layer, providing perfect occlusions for scene content in the back (red plots). In the second and third cases, only one modulating layer is deployed frontmost, and view-dependent content is added by low resolution (green plots) or full resolution (blue plots) light field rendering. The error decreases fast when adding the first few layers; however, adding more than four layers does not reduce the error as drastically anymore.

In the right plot, the impact of an increasing number of modulating layers is assessed for a fix number of six (red plots), four (green plots), and three (blue plots) emissive layers. For this analysis, the modulating layers are placed after each emissive layer, starting from the frontmost layer. An increasing number of modulating layers helps to reduce the error significantly. The plots furthermore show that increasing the number of emissive layers without increasing the number of modulating layers leads to significant high errors due to incorrect occlusions, perceived as black gaps as shown in Fig. 6 on the bottom.

5 Implementation

Based on the analysis presented in the previous section, we implemented two types of multi-layer plenoptic displays. Our two setups are shown in Figs. 8 and 9. The first prototype uses temporal multiplexing to superimpose the different layers, while the second uses spatial multiplexing. Temporal multiplexing is performed by combining a projector with multiple bi-state scattering planes. Spatial multiplexing is performed by combining two automultiscopic displays using a beam splitter.

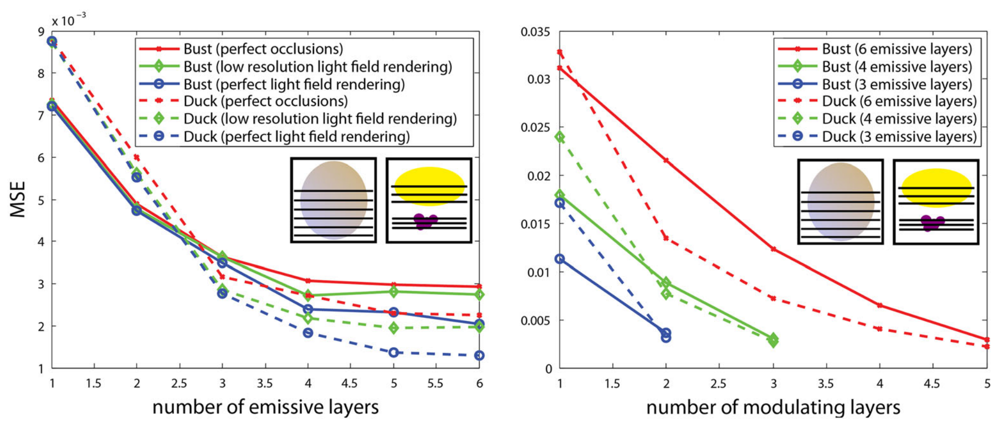

FIGURE 7 — Quantitative error analysis using simulated results. Impact of an increasing number of emissive layers (left) as well as modulating layers (right) is evaluated. The display geometry for six emissive layers is shown in the insets. Modulating layers are added successively, starting with the frontmost layer. For perfect light field rendering, the same angular resolution as the input light field is used, whereas in the low resolution rendering, only 12 angular images are used over the whole field of view.

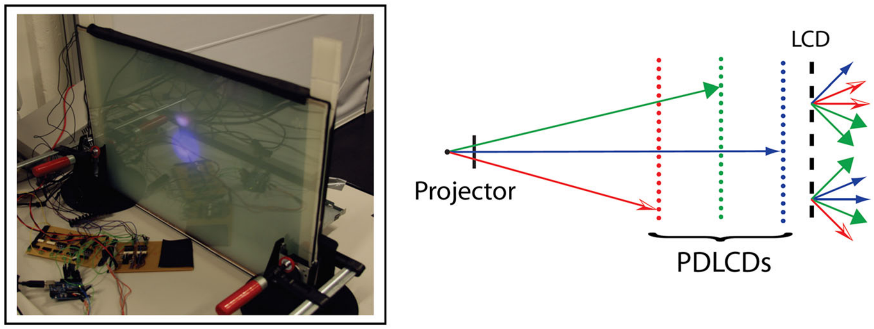

FIGURE 8 — Our temporally multiplexed prototype consists of three emissive layers and a frontmost modulating layer.

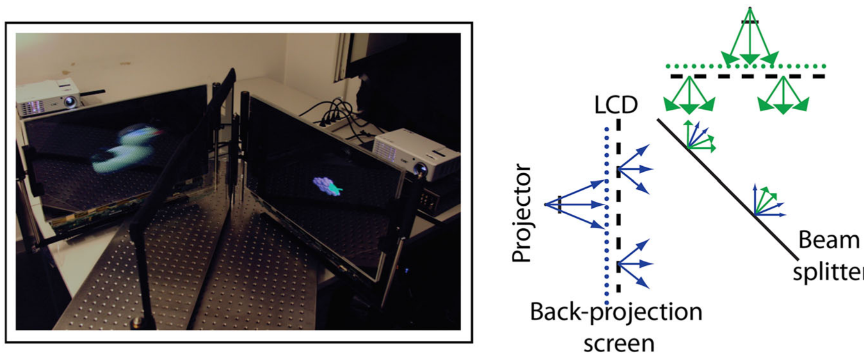

FIGURE 9 — Our spatially multiplexed prototype uses a beamsplitter two overlay two parallax barrier displays.

5.1 Display prototype using temporal multiplexing

We provide an example configuration in a first, temporally multiplexed prototype. It deploys three PDLC layers in combination with a projector to approximate the emissive layer primitive and an LCD in the front as modulating layer. In each frame, one of the PDLC layers is opaque and diffuses incoming light, while the others are transparent. This allows to show different images on different layers in time-multiplexed manner, similar to those of Sullivan.[6] The PDLC layers are driven by a circuitry that creates an alternating square wave function, preserving damages to the liquid crystal structures. The circuitry is synchronized with the v-sync signal of the projector such that the opaque layer is switched with each newly projected frame. As modulating layer, we employ an LCD with non-diffusing polarizing films. This layer is frontmost and used both to provide approximated occlusion and to render the light field portion of our decomposition algorithm.

The PDLC layers are spaced at 4, 10, and 16 mm from the front LCD. The projector has XGA resolution and a refresh rate of 60 Hz, matching the refresh rate of the used PDLC layers. The front LCD renders 12 views in a 10° field of view when used as parallax barrier layer. The effective refresh rate of the display is 15 Hz, because the frontmost emissive layer is used both for volumetric rendering and for parallax barrier rendering. The complete setup is shown in Fig. 8.

Multi-planar plenoptic displays with homogeneous and well-aligned optical elements do not require calibration, as pixels are stacked directly behind each other in a one-to-one correspondence. However, in our setup, we combine heterogeneous elements such as the projector and the LCD, which makes software calibration necessary. We propose a variant of the calibration scheme proposed in the work of Annen *et al*.[8] and perform homography estimation based on photographs of projected checkerboard patterns.

5.2 Display prototype using spatial multiplexing

Our second prototype combines two automultiscopic parallax barrier displays using spatial multiplexing. The parallax barriers are only used for 3D light fields, that is, the barriers only provide distinct rays aligned with the horizontal plane. Both displays are combined onto the same optical path using a beam-splitter mirror. Each of the displays is placed at a different distance from the beam splitter and is then used to display different parts of the light field to achieve increased depth range. Each parallax barrier display is composed of a 120-Hz projector paired with a diffuse back-projection layer for the emissive primitive and a twisted nematic LCD displaying a parallax barrier pattern as modulating primitive. Parallax barrier displays usually require big spacing between the barrier stripes to achieve an acceptable angular resolution. They therefore result in spatially under-sampled images that additionally lack a considerable amount of brightness because of the pinhole nature of the barrier. We therefore employ temporal multiplexing for each barrier display: Multiple spatially offset barrier patterns are projected in short sequence, with the respective light field content on the emissive primitive to achieve higher perceived spatial resolution.

Both projectors and displays have a native resolution of 1920×1080, and the pixel spacing is approximately 0.27 mm. The parallax barrier and the emissive plane are at a distance of 10 mm. The spacing between the parallax barrier slits is between 9 and 12 pixels for one frame and is adjusted according to the displayed scene. Three consecutive frames are used for temporal multiplexing, resulting in perceived parallax barrier spacing of three to four pixels. The corresponding 9–12 views of each parallax barrier consequently can be used to achieve a field of view of 7–9.5°. The virtual distances between the two parallax barrier displays are 100 and 200 mm.

6 Results

6.1 Examples

We demonstrate several examples for both the spatially and temporally multiplexed setups. Figure 4 shows all steps of our algorithm captured on the temporally multiplexed prototype. Figure 6 shows some of the simulation results used for our quantitative analysis. Figure 10 shows real results displayed on the spatially multiplexed prototype with visible parallax between the two views. Our displays are quite unique because an observer is provided with accommodation cues, visible in Fig. 11, as well as binocular cues and motion parallax. In addition to this, view-dependent effects can be observed clearly, and they add to the depth perception. We believe that the superimposed spatial modulator does not significantly influence the accommodation cues.

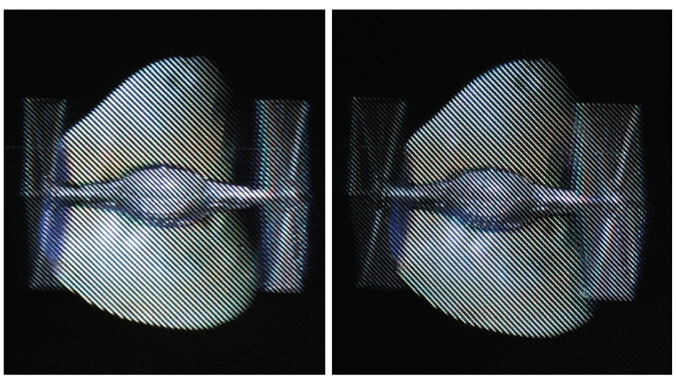

FIGURE 10 — Two views captured on our spatially multiplexed prototype with clearly visible parallax and proper occlusions. The spaceship is rendered on the front layer, whereas the moon is displayed by the layer in the back.

6.2 Component limitations

Our beam-splitter setup as well as the time-multiplexed setup is limited in terms of possible size and scalability. As shown by Barnum *et al.*,[9] the maximum possible number of layers is limited by the finite switching speed of the layers and the projector refresh rate. In addition, the dynamic range of the display is limited. In order to achieve constant brightness across the displayed light field, the lines in parallax patterns must be brighter than the parts rendered using the emissive layers only. Furthermore, the PDLCD panels are maximally 80% transparent and can only switch at 60 Hz. These could be replaced by more transparent and faster switching panels, as shown by Sullivan.[6] In addition, common LCD panels take several milliseconds to switch from white to black, causing shadowing. Finally, transmission of LCD layers is typically less than 10%, which makes the stacking of many LCD layers impractical. Much faster switching panels exist, such as the π-cell, and there is research on large-size, sub-millisecond switching modulators.

6.3 Limitations of the algorithm

Our light field decomposition assumes knowledge of the scene depth and is so far restricted to synthetic scenes. Our algorithm could work well with pre-recorded light fields, as long as a sufficiently accurate and dense depth map can be computed. Furthermore, we cannot currently handle transparent or semi-transparent scene elements. Finally, the spatially varying modulators assign fully blocked or transparent states only. Taking advantage of the intermediate states could, in principle, improve the overall bandwidth.

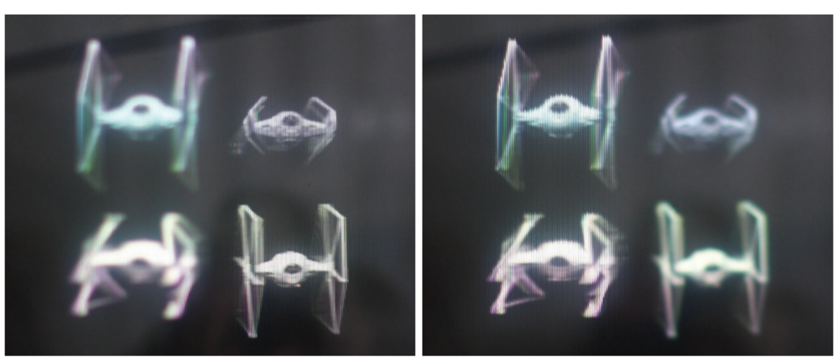

FIGURE 11 — Multi-planar plenoptic displays provide good accommodation cues as visible in these images, captured on our spatially multiplexed prototype with different camera focus.

7 Conclusion

We have introduced a general concept of multi-planar plenoptic displays. These display systems fuse multiple emissive and spatially modulating layers, providing a viewer with nearly correct accommodation cues, binocular cues, motion parallax, and properly handled occlusions, as in contrast to conventional displays. First, we have presented a mathematical framework to analyze light transport for these displays. Second, we have described a rendering algorithm that takes as input a 3D scene and drives a given multi-planar system. Next, we have conducted an error analysis for multi-planar plenoptic displays. We then discussed practical issues of designing and building different display configurations. We demonstrate examples both in a simulation and on our two physical prototypes. Figure 4 shows all steps of our algorithm recorded on our multi-planar plenoptic display, each step decreasing the difference to the input light field. Figure 6 shows some of the simulated results that are used in our quantitative error analysis, supporting our insights about required number of layers. Although limitations are imposed by current available hardware, we believe our approach to suit a certain range of applications and will get more involved with upcoming technologies.

References

1 G. M. Lippmann. “La photographie integrale,” *Comptes-Rendus* **146**, 446–451 (1908).
2 F. Ives. “Parallax stereogram and process for making same”. US Patent No. **725**, 567 (1903).
3 J.-X. Chai *et al*., “Plenoptic sampling”. In Proceedings of SIGGRAPH’00, p. 307–318 (2000).
4 O. S. Coissart *et al*., “Occlusion-capable multiview volumetric three-dimensional display”. *Appl. Optics* **46**, 1244–1250 (2007).
5 Jones, A. *et al*., “Rendering for an interactive 360 light field display,” *ACM Transactions on Graphics* **26**, 3 (2007).
6 A. Sullivan. “DepthCube solid-state 3D volumetric display,” *In SPIE Stereoscopic Displays and Virtual Reality Systems* **5291**, 279–284 (2004).
7 G. E. Favalora. “Volumetric 3D displays and application infrastructure,” *In Computer* **38**, No. 8, 37–44 (2005).
8 T. Annen *et al*., “Distributed rendering for multiview parallax displays”. In Stereoscopic Displays and Applications, Proceedings of SPIE Vol. 6055 (2006).
9 P. C. Barnum *et al*., “A multi-layered display with water drops”. In SIGGRAPH (2010).
10 R. Ranieri *et al*., “Multi-layered automultiscopic displays,” *Computer Graphics Forum* **31**, No. 7, 2135–2143 (2012).
11 S. Tamura and K. Tanaka. “Multilayer 3-D display by multidirectional beam splitter,” *Appl. Optics* **21**, 3659–3663 (1982).
12 G. Wetzstein *et al*., “Tensor displays: compressive light field synthesis using multilayer displays with directional backlighting,” *ACM Transactions on Graphics* **31**, 4, (2012).
13 F. Durand *et al*., “A frequency analysis of light transport”. In SIGGRAPH, p. 1115–1126 (2005).
14 A. Levin and F. Durand. “Linear view synthesis using a dimensionality gap light field prior”. In CVPR, p. 1831–1838 (2010).
15 M. Zwicker *et al*., “Antialiasing for automultiscopic 3D displays”. In Eurographics Symposium on Rendering (2006).

Nicola Ranieri received his master’s degree from the Swiss Federal Institute of Technology Zürich (ETH) in 2011 and is now a PhD candidate in the Computer Graphics Laboratory at the Institute for Visual Computing. His research interests include computer graphics, computer vision, 3D display technology, and interaction with such devices as well as virtual reality.

Dr. Simon Heinzle is currently at ETH Zurich. Before returning to ETH, Simon was a post-doctoral researcher at Disney Research Zurich until 2013 and was working in the area of advanced video technology. Before joining Disney, Simon received his master’s degree from ETH in 2006 and his Doctor of Sciences from ETH in 2010. Simon’s research interests span both computer graphics and vision. In particular, he is interested in video acquisition systems, video and light field processing, stereoscopy, novel display devices, and ASIC/FPGA design for video processing and computer graphics.

Peter Barnum defended his PhD in May 2011, and is now part of the Texas Instruments Embedded Vision R&D group. He is interested in using computer vision to solve real -world problems, for both for human assistive devices and, as well as autonomous robotic applications. Specifically, he is currently focused on scene understanding and systems that assist drivers.

Wojciech Matusik is an Associate Professor of Electrical Engineering and Computer Science at the Computer Science and Artificial Intelligence Laboratory at MIT, where he is a member of the Computer Graphics Group. Before coming to MIT, he worked at Mitsubishi Electric Research Laboratories, Adobe Systems, and Disney Research Zurich. He studied computer graphics at MIT and received his PhD in 2003. His research interests are in direct digital manufacturing and computer graphics, particularly in modeling and physical reproduction of materials, computational photography, and novel display systems.

Markus Gross is a Professor of Computer Science at the Swiss Federal Institute of Technology Zürich (ETH), head of the Computer Graphics Laboratory, and the Director of Disney Research, Zürich. He joined the ETH Computer Science faculty in 1994. His research interests include physically based modeling, computer animation, immersive displays, and video technology. Before joining Disney, Gross was director of the Institute of Computational Sciences at ETH. He received a master of science in electrical and computer engineering and a PhD in computer graphics and image analysis, both from Saarland University in Germany in 1986 and 1989.