Adaptive Image Synthesis for Compressive Displays

Felix Heide[1]   Gordon Wetzstein[2]   Ramesh Raskar[2]   Wolfgang Heidrich[1]

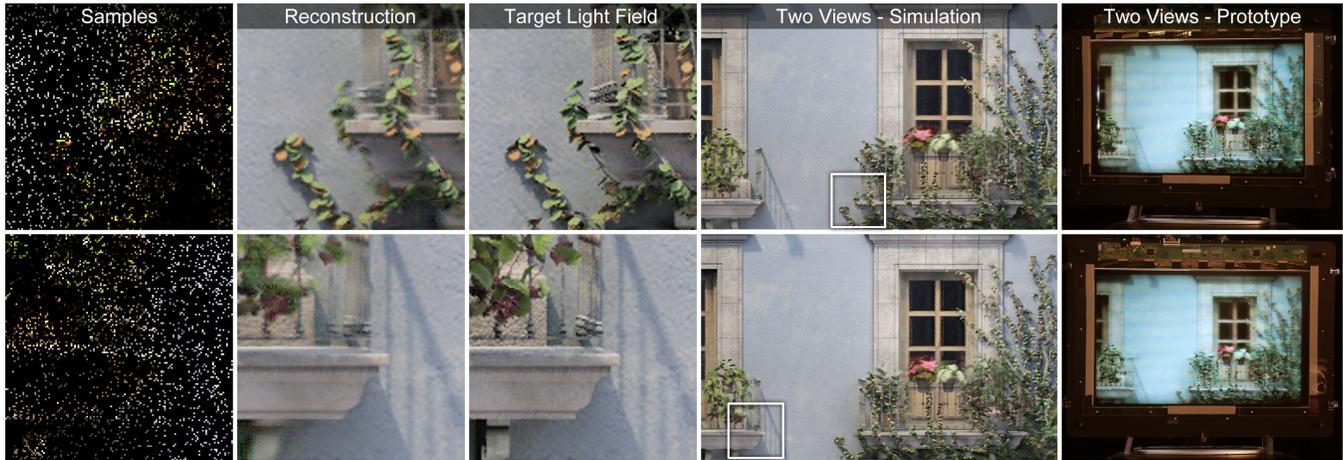[1]University of British Columbia   [2]MIT Media Lab

Figure 1: *Adaptive light field synthesis for a dual-layer compressive display. By combining sampling, rendering, and display-specific optimization into a single framework, the proposed algorithm facilitates light field synthesis with significantly reduced computational resources. Redundancy in the light field as well as limitations of display hardware are exploited to generate high-quality reconstructions (center left column) for a high-resolution target light field of 85×21 views with 840×525 pixels each (center). Our adaptive reconstruction uses only 3.82% of the rays in the full target light field (left column), thus providing significant savings both during rendering and during the computation of the display parameters. The proposed framework allows for higher-resolution light fields, better 3D effects, and perceptually correct animations to be presented on emerging compressive displays (right columns).*

Abstract

Recent years have seen proposals for exciting new computational display technologies that are *compressive* in the sense that they generate high resolution images or light fields with relatively few display parameters. Image synthesis for these types of displays involves two major tasks: sampling and rendering high-dimensional target imagery, such as light fields or time-varying light fields, as well as optimizing the display parameters to provide a good approximation of the target content.

In this paper, we introduce an adaptive optimization framework for compressive displays that generates high quality images and light fields using only a fraction of the total plenoptic samples. We demonstrate the framework for a large set of display technologies, including several types of auto-stereoscopic displays, high dynamic range displays, and high-resolution displays. We achieve significant performance gains, and in some cases are able to process data that would be infeasible with existing methods.

CR Categories: I.3.3 [Computer Graphics]: Three-Dimensional Graphics and Realism—Display Algorithms

Keywords: computational displays, image synthesis

Links: ◆DL ⬛PDF ⬛WEB

1 Introduction

Display technology is currently undergoing major transformations. The ability to include significant computing power directly in the display hardware gives rise to *computational displays*, in which the image formation is a symbiosis of novel hardware designs and innovative computational algorithms. An example of an early commercial success for this approach are high contrast or high dynamic range displays based on low-resolution local backlight dimming [Seetzen et al. 2004].

Many of the recently proposed display designs are not only *computational*, but also *compressive* in the sense that the display hardware has insufficient degrees of freedom to *exactly* represent the target content, and instead relies on an optimization process to determine a perceptually acceptable approximation. In addition to high dynamic range displays, other display technologies exhibiting compressibility in the parameter space include high-resolution projectors using optical pixel sharing [Sajadi et al. 2012], as well as compressive light field displays using either tomographic [Wetzstein et al. 2011; Gotoda 2011; Lanman et al. 2011] or low-rank image formations [Lanman et al. 2010; Wetzstein et al. 2012]. Many of these display technologies show promise to be incorporated in next-generation consumer technology.

The major bottleneck for these display technologies is the increasing demand on computational resources. Consider the example of a high-quality light field display with 100×100 viewpoints, each having HD resolution, streamed at 60 Hz. More than one trillion light rays have to be rendered per second requiring more than 100 Terabytes of floating point RGB data to be stored and processed. Just considering a single frame of that stream, the underlying optimization would require a problem with ten billion observations to be solved in less than $1/60^{th}$ of a second. Clearly, a conventional approach attempting to render all data and subsequently process it is infeasible.
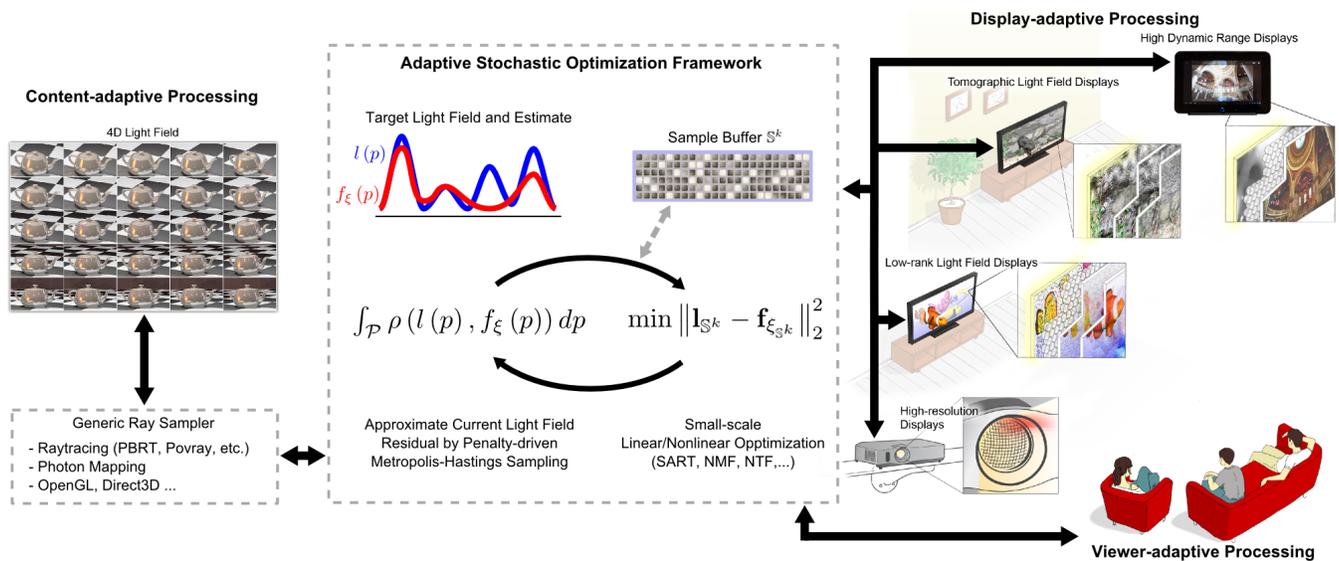
Figure 2: *Illustration of adaptive image synthesis for compressive displays. The proposed framework unifies rendering, optimization, and display to provide high-quality viewing experiences for a variety of emerging compressive display technologies. Processing is adapted to the content and the characteristics of display hardware; limitations of the human visual system are accounted for by the image formation.*

In this paper, we propose a new framework for image synthesis targeting compressive displays. Our framework directly utilizes the compressive nature of the display parameters to reduce the computational cost of both the rendering and the parameter optimization steps. To this end, we propose an adaptive image synthesis framework that interlinks rendering and display-specific optimization. We demonstrate that only a fraction of the light rays actually need to be rendered to achieve high-quality display; similarly, solving smaller optimization problems in an iterative fashion further reduces the demand on computational resources. Figure 1 demonstrates how very few light field samples allow for high-quality light field display, whereas Figure 2 provides an overview of the proposed framework.

Contribution   A key contribution of our work is to identify compressive displays as a distinct class of computational displays that have shared needs for image generation. We then introduce a framework for adaptive image synthesis for such displays. The proposed framework combines the following characteristics:

- rendering (evaluating the radiance function at sampled locations) and optimization of display degrees of freedom are combined into single, adaptive algorithm;

- the sample generation in the rendering stage is steered by the residual of the optimization procedure—new samples are generated where they help most in improving the display parameters;

- in this way, the sampling becomes adaptive to the scene content and the display capabilities;

- the framework enables image generation for compressive displays using only a fraction of the light field samples that would be required using the existing brute-force approaches. This not only lowers rendering times and processing speed, but also bandwidth requirements.

We demonstrate our framework by showing how it can be applied to a range of recently proposed compressive display technologies.

2   Related Work

Our image generation framework for compressive displays draws on work from a number of areas as discussed in the following.

Computational Displays   leverage the co-design of optics and computation to overcome fundamental limits of purely optical designs. Recently, it has been shown that such displays can increase the resolution [Sajadi et al. 2012] and depth of field [Grosse et al. 2010] of projectors and the dynamic range of monitors or TVs [Seetzen et al. 2004]. Autostereoscopic 3D displays have come a long way since the invention of integral imaging [Lippmann 1908] and parallax barriers [Ives 1903]. Recent proposals include volumetric displays using mechanically moving parts [Cossairt et al. 2007; Jones et al. 2007], tomographic multilayer displays [Wetzstein et al. 2011; Lanman et al. 2011], and low-rank light field displays [Lanman et al. 2010; Wetzstein et al. 2012].

While display-adaptive rendering was originally described in the context of devices that have a limited color gamut [Glassner et al. 1995], we are particularly interested in *compressive* displays, in which the display hardware has fewer degrees of freedom than the target image (e.g. [Seetzen et al. 2004; Sajadi et al. 2012; Wetzstein et al. 2011; Gotoda 2011; Lanman et al. 2011; Lanman et al. 2010; Wetzstein et al. 2012]). Compressive displays are attractive means of image generation, because they can make use of perceptual limitations of the human visual system to produce approximations of the target that are almost indistinguishable from the ground truth, at much lower hardware complexity and cost. On the other hand, this compressed representation mandates that the degrees of freedom of the display hardware are obtained from the target image via an optimization procedure. Current solutions start from a dense representation of the target content and optimize the display parameters accordingly. In contrast, we propose an adaptive sampling solution that can determine the full display parameters with only a small fraction of the sampled target.

Stochastic Optimization   is commonly used in the scientific computing community. Approaches to gradient sampling such

as [Widrow and Stearns 1985; Bertsekas 1997; Friedlander and Schmidt 2012], can be employed to mitigate memory usage and compute times in least-squares optimization problems. Friedlander and Schmidt's method [2012] is most closely related to our approach. In contrast to gradient sampling, the observations or plenoptic samples of our optimization problem are assumed to be unknown—they can be sampled, albeit at a significant cost.

Most recently, stochastic tomography has been proposed for the application of capturing mixing fluids [Gregson et al. 2012]. Their method follows a traditional pipeline approach, where all observations are measured first and then the full-sized inverse problem is solved with the help of sampling techniques. In contrast, our approach combines the process of drawing samples from the observations and updating the unknowns of an inverse subproblem in an adaptive and iterative manner.

Sampling in Graphics has a long tradition [Cook et al. 1984] and is a standard tool in rendering global illumination effects [Veach and Guibas 1997; Lehtinen et al. 2012], shadows [Egan et al. 2011] as well as depth of field and motion blur [Soler et al. 2009; Egan et al. 2009; Lehtinen et al. 2011; Li et al. 2012]. Our algorithm uses a simple Markov chain to select new sample positions; more sophisticated sampling strategies, such as the above, could improve performance. However, rendering times grow linearly with the number of traced rays, whereas the bottleneck of the proposed method is the inverse problems in the optimization step (see Sec. 4). These usually exhibit superlinear, quadratic, or worse growth w.r.t. the size of the input data. The highest performance gain for display-adaptive rendering can therefore be achieved by the co-design of rendering and optimization, as introduced in this paper.

Most recently, researchers have considered the problem of *compressive rendering*. In these methods, a small number of randomly selected light rays are rendered in 2D screen space [Sen and Darabi 2011] or higher-dimensional plenoptic space [Sen et al. 2011]. Assuming that the image or plenoptic function is sparse in some transform domain, it is subsequently reconstructed using sparse coding techniques. Unlike compressive sensing approaches, for instance in computational photography [Marwah et al. 2013], compressive rendering is an inpainting problem that employs sparse coding to fill in missing data. While the basic idea of rendering only a small subset of all light rays is similar to ours, the methods target different applications and differ algorithmically. Our framework is an adaptive feedback loop that iterates between generating and rendering plenoptic samples and solving small-scale, display-adaptive inverse problems, whereas compressive rendering samples in a non-adaptive fashion and uses sparse coding to inpaint missing data.

3   Compressive Displays

In this section, we formally introduce *compressive* displays as a special class of computational displays. In particular, a compressive display exposes a relatively small set of programmable parameters ξ, which can be adjusted in order to emit a temporally-varying light field $\tilde{l}$ based on a an image formation process f_ξ:

$$\tilde{l}(x, \nu, t) = f_\xi(x, \nu, t), \qquad (1)$$

where x, ν, and t are the spatial, directional, and temporal dimensions of the plenoptic function [Adelson and Bergen 1991]. Since our framework targets a range of technologies from 2D displays to animated light field displays, we use the terms plenoptic function and light fields interchangeably to describe the target imagery. The function f_ξ maps display state parameters to an emitted light field as a linear or nonlinear process.

To consider some specific examples, in a layered 3D display ξ corresponds to the set of pixel values in the individual layers and f_ξ describes either the light attenuation [Wetzstein et al. 2011] or polarization rotation [Lanman et al. 2011] as rays optically interact with display layers. As another example, in an HDR display [Seetzen et al. 2004] ξ corresponds to the set of backlight LED intensities as well as the pixel values of the front LCD panel, while f_ξ describes the optical blurring of the LED illumination as well as the multiplication with the transparency values of the LCD pixels. In Section 5 we discuss a number of additional compressive display technologies that can be described by this model.

In a compressive display, the degrees of freedom in the parameter vector ξ are typically much lower than the desired resolution of the target light field $l(x, \nu, t)$. Hence, an approximation must be found by solving an optimization problem:

$$\begin{aligned} \underset{\{\xi\}}{\arg\min} \quad & \int_\mathcal{T} \int_\mathcal{V} \int_\mathcal{X} \rho\big(l(x,\nu,t), f_\xi(x,\nu,t)\big)\, dx\, d\nu\, dt + \Gamma(\xi) \\ \text{subject to} \quad & c(\xi) = 0 \end{aligned}$$

$$(2)$$

The high-dimensional integral represents the approximation error, in all plenoptic dimensions, between emitted and target light field. This error is measured using a penalty function $\rho : \mathbb{R} \to \mathbb{R}$. In the easiest case, the penalty function yields a least-squared error, i.e. $\rho(\alpha, \beta) = (\alpha - \beta)^2$, although we show in Section 5 that other penalty terms occur in practice. For full generality, Equation 2 also includes a regularization term on the display elements $\Gamma(\xi)$ as well as constraints on them $c(\xi)$. While the constraints enforce physical limitations, for example optically feasible pixel transmission values, the regularization term promotes desirable properties such as smoothness or sparseness in the pixel states.

3.1   Image Generation for Compressive Displays

The brute-force approach to image synthesis for compressive displays described in the literature is a sequential pipeline: the target light field l is densely sampled and rendered, then the full optimization problem is solved using all rendered light rays. This approach is outlined in Algorithm 1.

Algorithm 1 Conventional Compressive Display Optimization

1: $\mathbb{S}^0 \leftarrow \emptyset, \xi^0 \leftarrow 0$ // initialization
2: $\mathbb{S} \leftarrow uniform()$ // uniform sampling
3: $\xi \leftarrow \min_{\{\xi\}} \|\mathbb{I}_\mathbb{S} - \mathbf{f}_\xi\|_2^2 + \Gamma(\xi)$ // optimization

---

A conventional pipeline approach, however, does not scale to the increasing computational demands of emerging compressive displays. Hence, previously proposed compressive displays either operate on low resolutions (e.g. [Wetzstein et al. 2011; Lanman et al. 2011]) or employ greedy heuristic algorithms (e.g. [Seetzen et al. 2004]).

3.2   Adaptive Sampling Framework

We instead propose an adaptive stochastic sampling framework, as outlined in Figure 2 and Algorithm 2. A ray sampling module generates a sparse set of light field samples that are subsequently rendered. A corresponding, small-scale optimization problem is then solved to determine the display parameters ξ that best represent this sample set. The resulting light field $\tilde{l}$ generated on the display defines a residual function over the plenoptic domain, which can be efficiently sampled using a Markov Chain Monte Carlo approach to determine a new set of locations for the sampling module. This process iterates until the desired approximation quality is achieved.

The proposed framework combines a number of desirable properties:

Computational Efficiency.   High-quality computational displays may require ultra-high resolutions to be processed. Our adaptive sampling framework can generate images on compressive displays using only a fraction of the samples required by brute-force algorithms. This significantly reduces rendering time, memory consumption, and the sizes of optimization problems.

Display Adaptivity.   A display has physical limitations in resolution, refresh rate, dynamic range, and, for light field displays, depth of field. To maximize computational efficiency, these display-specific limitations must be taken into consideration in the sampling process—sampling outside the provided bandwidth is redundant. Our adaptive framework incorporates these considerations through the image formation model f_ξ and in the form of constraints.

Content Adaptivity.   Natural images exhibit characteristics that can be well-modeled by statistical priors. Natural light fields are no different; redundancies in a high-dimensional target light field are exploited by computing optimized, lower-dimensional decompositions for a particular display. Furthermore, adaptive sampling of the residual function quickly hones in on edges and other important detail of target light fields (see Sections 4 and 5).

Viewer Adaptivity.   The human visual system is a complex mechanism and modeling it in detail is an active area of research. With our framework, we mostly exploit its limited temporal resolution by displaying high-speed patterns that are perceptually averaged. Incorporating more complex aspects of human perception, such as sensitivity to contrast [Mantiuk et al. 2011] and disparity [Didyk et al. 2011], is an interesting avenue of future research.

Seamless Integration with Lower-level Architectures.   Finally, our framework integrates well with currently available and emerging low-level hardware and software systems. In particular, it is compatible with, but mostly independent of, a variety of existing graphics systems such as raytracers, hardware-accelerated systems (e.g., CUDA) as well as content creation tools.

4   Adaptive Stochastic Optimization

This section discusses the individual steps of Algorithm 2 in detail.

4.1   Adaptive Light Field Residual Sampling

The first component of the proposed framework is an operator $sample(\cdot)$ that adaptively probes the light field residual in order to

---

Algorithm 2 Adaptive Sampling and Optimization

1: $\mathbb{S}^0 \leftarrow \emptyset, \mathbf{l}_{\mathbb{S}^0} \leftarrow \emptyset, \xi^0 \leftarrow 0$ // initialization
2: **for** $k \leftarrow 1$ to K **do**
3:         // adaptively add samples
4:         $(\mathbb{S}^k, \mathbf{l}_{\mathbb{S}^k}) \leftarrow (\mathbb{S}^{k-1}, \mathbf{l}_{\mathbb{S}^{k-1}}) \cup sample\left(\xi^{k-1}\right)$
5:         // optionally find display parameters affected by $\mathbb{S}^k$
6:         $\{\xi_{\mathbb{S}^k}\} \leftarrow find\left(\mathbb{S}^k\right)$
7:         // optimize display parameters
8:         $\xi^k \leftarrow \min_\xi \left\| \mathbf{l}_{\mathbb{S}^k} - \mathbf{f}_{\xi_{\mathbb{S}^k}} \right\|_2^2 + \Gamma(\xi)$
9: **end for**

---

locate the samples that most significantly contribute to the overall error. To simplify notation, we define the high-dimensional integral over the plenoptic residual as

$$\int_{\mathcal{P}} P_\xi(p)\, dp \equiv \int_{\mathcal{T}} \int_{\mathcal{V}} \int_{\mathcal{X}} \rho\big(l(x,\nu,t), f_\xi(x,\nu,t)\big)\, dx\, d\nu\, dt, \quad (3)$$

where $\mathcal{P}$ is the plenoptic domain, including spatial, angular, and temporal variation. The function P_ξ returns the residual ρ at a specific plenoptic sample $p \in \mathcal{P}$.

The first step in each iteration k of Algorithm 2 is a sampling stage, where we fix the current estimate ξ^{k-1} of display parameters, which in turn fixes the residual function $P_{\xi^{k-1}}$. The goal of this sampling step is to draw an i.i.d. set of light field samples $\mathbb{S}$ from a target probability distribution $b \propto P_{\xi^{k-1}}(p)$ so that new samples concentrate in regions with high residual values. The subsequent optimization step in Algorithm 2 then reduces the residual in these regions.

Consider the example of a two-layered, low-rank light field display, where the target light field is approximated as a time sequence of rank-1 light fields that are in turn represented as the product of a front and a back layer 2D display [Lanman et al. 2010]. The parameters ξ of such a display are the time sequences of the front and back layer images that represent the light field. In iteration $k-1$ of Algorithm 2 the current estimate ξ^{k-1} of these parameters is determined using only a subset $\mathbb{S}^{k-1}$ of the total ray space (see Sections 4.2 and 5.2 for details on the optimization). The residual function $P_{\xi^{k-1}}$ is then the difference between the light field emitted by the display using these estimated layers, and the (not yet fully rendered) target light field. This residual function is difficult to sample and expensive to evaluate since this evolves rendering new light field rays. However, we expect a significant amount of local coherence in the residual function. For this reason, we turn to the Metropolis-Hastings algorithm [Metropolis et al. 1953; Hastings 1970] to implement the sampling process. A Markov Chain of samples is generated, as shown in Algorithm 3, using a proposal distribution $q(p^*|p)$ to generate a new candidate sample p^* given the previous sample p. The chain then moves towards p^* with probability

$$a = \min \left\{ 1, \frac{b(p^*) \cdot q(p^*|p)}{b(p) \cdot q(p|p^*)} \right\} = \min \left\{ 1, \frac{P_{\xi^{k-1}}(p^*)}{P_{\xi^{k-1}}(p)} \right\}. \quad (4)$$

In order to exploit local coherence in the plenoptic domain, we use a multivariate Gaussian proposal distribution q. Since the Gaussian strategy is symmetric, i.e. $(q(p^*|p) = q(p|p^*))$, and the target distribution was chosen to be proportional to the residual, we obtain the simplified acceptance condition outlined on the right of Equation 4. The transition probability requires evaluating the residual for a proposed location p^*, and therefore each proposed sample is rendered immediately, for instance using a ray-tracer. Samples with a residual below a threshold ϵ do not contribute significantly to improving the selection of display parameters in the optimization step, and are hence dropped from the sample set. The initial chain sample is drawn from a uniform distribution.

For simplicity, Algorithm 3 only shows a single Markov Chain. In practice, our implementation runs many chains in parallel. While we observe significant performance benefits from such a parallelization of the sampling stage, a more detailed theoretical analysis of its effects on ergodicity and convergence are left for future investigation.

4.2   Updating Pixel Values through Optimization

The output of the sampling stage in each iteration of Algorithm 2 is a small set of additional sampling locations. These light field

Algorithm 3 Metropolis-Hasting penalty sampling (*sample*-operator in Algorithm 2)

1: $\mathbb{S}^k \leftarrow \emptyset, \mathbf{l}_{\mathbb{S}^k} \leftarrow \emptyset$ // Initialize new recorded sample set
2: $P_p \leftarrow P_{\xi^{k-1}}(p)$
3: **for** $i \leftarrow 1$ to I **do**
4:      $p^* \leftarrow gaussian(p, \vec{\sigma})$
5:      $\mathbf{l}^* \leftarrow render(p^*)$
6:      $P_{p^*} \leftarrow P_{\xi^{k-1}}(p^*)$
7:      $a = \min(1, P_{p^*}/P_p)$
8:      **if** $uniform() \leq a \ \wedge \ P_{p^*} > \epsilon$ **then**
9:          // Record new sample
10:          $\mathbb{S}^k \leftarrow \mathbb{S}^k \cup p^*$
11:          $\mathbf{l}_{\mathbb{S}^k} \leftarrow \mathbf{l}_{\mathbb{S}^k} \cup \mathbf{l}^*$
12:          $p \leftarrow p^*$
13:      **end if**
14: **end for**
15: **return** $(\mathbb{S}^k, \mathbf{l}_{\mathbb{S}^k})$

samples are rendered on the fly and added to the full set of sampled and rendered light rays $\mathbf{l}_{\mathbb{S}^k}$, which includes both the newly generated samples as well as those obtained in previous iterations. Given this set of rays, the display parameters ξ are updated by solving an optimization problem

$$\begin{array}{cc} \underset{\{\xi\}}{\arg\min} & \left\| \mathbf{l}_{\mathbb{S}^k} - \mathbf{f}_{\xi_{\mathbb{S}^k}} \right\|_2^2 + \Gamma(\xi) \\ \text{subject to} & c(\xi) = 0 \end{array} . \qquad (5)$$

Further accelerations can be achieved by restricting the data term in Equation 5 to those display parameters $\xi_{\mathbb{S}^k} \subseteq \xi$ that are directly affected by the sample set $\mathbf{l}_{\mathbb{S}^k}$ (see Step 6 of Algorithm 2). The remaining display parameters are filled in using only the regularization term. However, in order to obtain good image quality, the sampling density will eventually have to be dense enough to cover most of the display parameters ξ, so that the performance gain of this strategy is limited to the first few iterations of the algorithm.

Returning to the previous example of a low-rank 3D display, the time sequence of layers for different sub-frames in the display can be determined using a non-negative matrix factorization [Lanman et al. 2010], which can be expressed as an optimization problem of the form of Equation 5 (see Section 5.2 for a detailed derivation). In our adaptive sampling framework, iteration k of Algorithm 2, involves a matrix factorization that only consists of the rays in the sample set $\mathbb{S}^k$.

At every iteration of the algorithm, Equation 5 minimizes the light field residual where it matters most. However, the optimized display elements $\xi_{\mathbb{S}}$ will in turn affect the residual at locations that were not part of the sampling set $\mathbb{S}$. To account for this, we allow Markov chains to continue throughout multiple iterations of the process. Once the acceptance rate of a chain either drops below some threshold, indicating that the residual in the neighborhood is sufficiently reduced, the chain is discarded and reinitialized at a random location. Ergodicity is ensured in this manner.

While the problem in Equation 5 increases in size throughout the iterative process, we demonstrate in Figures 3, 4 and in Section 5 that only a fraction of all samples is necessary to converge to a high-quality solution. Therefore, we are solving a much smaller problem than Equation 2 and obtain significant savings on the optimization subproblem as well as the rendering subproblem. In fact, since optimizations such as non-negative matrix factorization or tomography usually have high algorithmic complexity, solving the fully sampled optimization is oftentimes infeasible (without large cluster-hardware), see Section 5. The small-scale optimization problem
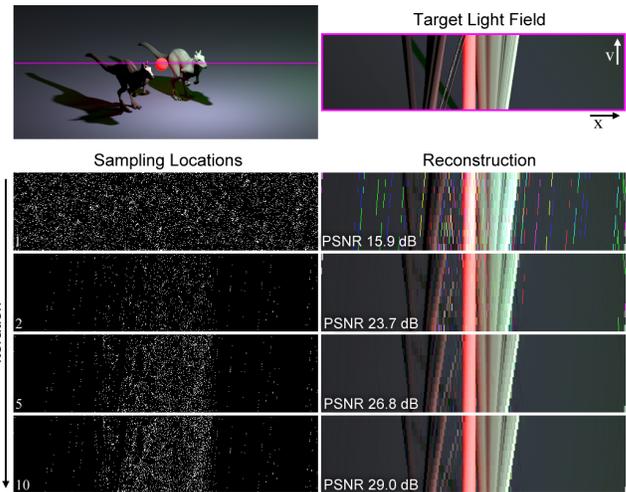

Figure 3: *Low-rank light field synthesis using adaptive stochastic optimization (Alg. 2). A 2D target light field (top row) is decomposed into a set of attenuation layers (see Section 5.2); intermediate sampling locations and reconstructions are shown for several iterations of the proposed algorithm.*
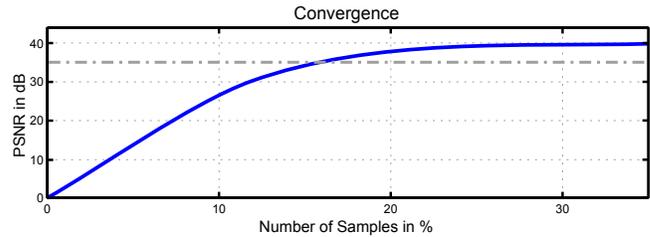

Figure 4: *Convergence of experiment in Figure 3. For this example, the adaptive algorithm converges to a PNSR of 35 dB with about 15% of the light field samples using significantly less computational resources than conventional methods.*

(Eq. 5), can be solved efficiently with conventional linear or nonlinear inversion methods, depending on the targeted display technology. Specific examples for a variety of displays are discussed in Section 5.

4.3 Discussion and Insights

As an illustration of the proposed algorithm, we show a simple 2D experiment in Figure 3. A spatio-angular slice of a light field is decomposed into a set of patterns for two 1D high-speed layers that create a low-rank light field approximation (see Sec. 5.2 for details). Intermediate reconstructions for the iterative process are shown in the right column of Figure 3. After 10 iterations, the algorithm achieves a reconstruction quality of a little less than 30 dB compared to the target light field, which provides a close match to the solution of the full problem. The left column shows the sampling locations at the corresponding iterations. We observe that samples are concentrated around content-dependent structures, such as high-dimensional edges, where the residual is highest.

Figure 4 plots the convergence for this experiment. We see that the algorithm quickly converges after sampling and rendering only about 15% of the light rays. As we shall see in Section 5, higher-dimensional datasets provide even more savings. Although we do not formally proof the convergence rate, convergence is guaranteed

# Optimization Iters (Alg. 2, line 8)	# New Samples per Iteration (I in Alg. 3)				
	200	6000	10000	20000	60000
10	3.72	20.68	24.81	81.07	178.17
100	6.27	1.74	2.93	8.89	32.09
200	5.99	2.28	2.39	3.88	18.51
500	6.72	2.81	4.37	2.18	8.86
1000	6.71	2.94	2.59	2.56	6.13

Table 1: *Compute times (in seconds) of the experiment in Figure 3 for a varying number of new samples I in Algorithm 3 as well as different numbers of iterations in each matrix factorization step (optimization in Alg. 2, line 8). This table explores the tradeoff between increasing compute efforts in the sampling stage (left to right) vs. the optimization stage (top to bottom) of our framework. All parameter settings are run until a target PSNR of 35 dB compared to the ground truth target light field is achieved.*

because the sampling process reaches each light field ray with finite probability, meaning that our approach converges to the full optimization problem in the limit.

Finally, Table 1 shows the compute times for varying parameters in the proposed adaptive optimization. Each column represents a different amount of rendering effort spent in the sampling stage of the algorithm (measured in the number of samples added in each iteration), while the rows represent different amounts of effort spent on the optimization stage (measured in the number of iterations within each optimization subproblem). Therefore, the lower left corner of the table corresponds to adding very few samples in each iteration but solving the resulting optimization problem very accurately, while the settings in the top right corner generate lots of additional samples, but only generate very approximate solutions in each optimization step. The best convergence is achieved with moderate values for both parameters, although the specific values need to be adjusted with problem size.

Two key insights facilitate adaptive image synthesis. First, the residual integrand is often not uniformly distributed, and second, it depends on the unknowns. By exploiting these two properties with a stochastic optimization approach, our method is able to significantly reduce both the number of rendered light field samples, as well as the size of display-specific optimization subproblems.

In the scientific computing community, two approaches to gradient descent exist: full gradient methods and incremental gradient methods (e.g., [Bertsekas 1997]). The former exhibit fast linear convergence but require an iteration cost proportional to the number of observations, while incremental gradient methods, such as the Widrow-Hoff LMS method [Widrow and Stearns 1985], sample the gradient only with respect to a single observation at a time. The cost per iteration is significantly reduced, but more iterations are required for the methods to converge. Our method combines the benefits of both worlds, similar in spirit to Friedlander and Schmidt [2012]. However, in contrast to their method, ours does not use a uniform sampling in the residual, and instead adapts to content, display degrees of freedom, and (through the image formation model of the display technology) to the limitations of the human visual system.

5 Applications and Results

This section demonstrates how the adaptive framework introduced in Section 4 facilitates higher resolutions and better 3D effects for a variety of compressive display technologies.

We have implemented the framework introduced in Section 4 using PBRT [Pharr and Humphreys 2010] as a rendering engine. For all of the examples, we use a fairly large number of around $100,000$ parallel sampling chains, each of length $I = 20$ (which makes

the sampling part of our algorithm easy to parallelize). We use a residual threshold of $\epsilon = 0.05$ (line 8 in Algorithm 3). The mutation strategies are zero-mean multivariate Gaussians with $\sigma = 3$ in each dimension. For future research we envision adaptive MCMC strategies. For rendering each light field sample, a small number of ≤ 25 samples was used in each result. Thus, the light field samples are not completely noiseless (e.g., most papers such as Lehtinen et al. [2012] mention 256-512 for noiseless pixels). The reconstruction of a light field sample from all its individual samples is done using a simple box-filter.

The display-specific image formation models and optimization algorithms discussed below are all implemented in Matlab.

5.1 Tomographic Light Field Displays

Recently, tomographic light field displays have been introduced. These can be constructed from stacks of light-attenuating layers [Wetzstein et al. 2011; Gotoda 2011] such as printed transparencies, or from polarization-rotating liquid crystal panels [Lanman et al. 2011]. While the optical image formation process for both display types is nonlinear, it was shown that the respective display parameter selection problem can be formulated as a linear optimization problem that is closely related to X-ray computed tomography (CT). In the notation introduced in the last section (Eq. 1), the pixel states of a stack of N light attenuating layers $\Xi^{(n)}$ at depths d_n can be mapped to an emitted light field as

$$f_{\mathrm{T}}(x, \nu) = \prod_{n=1}^{N} \Xi^{(n)}(x + d_n \nu). \quad (6)$$

Assuming a uniform backlight, each emitted light ray $\tilde{l}(x, \nu) = f_{\mathrm{T}}(x, \nu)$ is given as the product of the attenuation coefficients at its point of intersection with each layer. Following conventional computed tomography approaches, a linear formulation is achieved in the logarithmic domain:

$$\log f_{\mathrm{T}}(x, \nu) = \sum_{n=1}^{N} \log \left(\Xi^{(n)}(x + d_n \nu) \right). \quad (7)$$

Finding the pixel states that best approximate a target light field is achieved through constrained least-squares methods such as the simultaneous algebraic reconstruction technique (SART). This means that for this type of display, $\rho(l(x, \nu), f_{\mathrm{T}}(x, \nu)) = (\log l(x, \nu) - \log f_{\mathrm{T}}(x, \nu))^2$. Polarization state rotating layers can be formulated in a very similar fashion [Lanman et al. 2011].

We use a modified SART algorithm for optimizing the pixel states in the display layers. Standard SART iterates over different viewing directions ν of the volume, and in each iteration performs a volume rendering and a backprojection step for that viewing direction. In our framework, the sample set $\mathbb{S}^k$ contains only a subset of the pixels x for each view ν, so the volume rendering and backprojection are performed for only those samples in $\mathbb{S}^k$. Since SART already smoothes (i.e. regularizes) along projection directions, we do not employ an additional regularizer in this application.

The major bottleneck of any attempt to solve the tomographic problem above is the size of the problem. High-quality target light fields may require billions of light rays to be rendered and an optimization problem of the same size must be solved. Clearly, there are limits in commonly available computational resources, especially memory, that restrict feasible resolution. These restrictions are particularly severe for on-board display processing units, which are commonly not very powerful, yet have to deliver high-quality viewing experiences for the observers.

Reconstruction & Samples

Figure 5: *Adaptive tomographic light field decomposition. We show intermediate reconstructions for one view of the light field throughout the iterative algorithm. The closeups illustrate the cumulative light field samples of all previous iterations for two regions in this view. We observe that an extremely sparse set of plenoptic samples (close-ups are mostly black) is sufficient to generate high-quality reconstructions.*
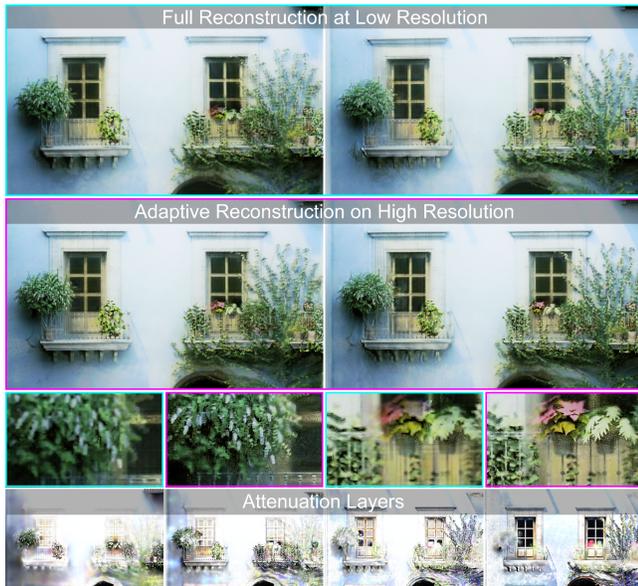

Figure 6: *Photographs of two tomographic light field display prototypes. We decompose a target light field into a set of four attenuation layers (bottom), print them on transparencies and stack them using clear acrylic spacers. Two views are shown for results matching the resolution previously achieved in the literature [Wetzstein et al. 2011] (top row) and at the significantly improved resolution enabled by our adaptive framework (center row). Zoomed regions and the optimized layers are shown at the bottom.*

Figure 5 shows renderings of the central view for a light field of the San Miguel scene, using a spatial resolution of 1680×1050 pixels and 25×25 views. Even after as few as six iterations and only sampling a fraction of the light rays, a high quality light field is synthesized. The method converges after 51 iterations of rendering an extra 2.4 million rays each, for a total of 122.4 million rendered light field rays (11.1% of the total light field). Of these rendered rays, around 40% are rejected in Step 8 of the sampling stage (Algorithm 3), leaving only 71.6 million rays (6.5% of the light field) to be used in the final optimization step. The total computational cost was $357,000$ seconds for rendering using PBRT, and $3,555$ seconds for optimization. While we saved more than a factor of 9 in rendering time using our adaptive approach, the biggest benefits arise in the optimization step, which would have been intractable for the full problem.

Figure 6 shows two photographs of a fabricated prototype taken from different perspectives. We compare our high-resolution results (top) to a full solution at a resolution of 512×384 using only 7×7 views, which is the resolution that was previously used for optimization based on the full-resolution target light field [Wetzstein et al. 2011]. We can see a clear quality improvement from

using higher resolution target imagery, which is made feasible by our adaptive framework.

5.2 Low-rank Light Field Displays

To provide high-quality viewing experiences, compressive light field displays rely on both the limitations of the human visual system (HVS) and specific structures of the presented content to compensate for the lack of degrees of freedom of the display hardware. In particular, display designs, such as duallayer displays [Lanman et al. 2010] or multilayer displays with directional backlighting [Wetzstein et al. 2012], provide an optical basis in which target light fields have been shown to be compressible. Non-negative matrix and tensor factorizations are employed to decompose light fields into a set of patterns presented to the viewer at a display refresh rate that is faster than the critical flicker frequency (CFF) of the HVS. The visual system perceptually averages over the patterns seen throughout the "exposure time" of the eye. Depending on the adaptation luminance, the temporal integration of photoreceptors is generally approximated as 40-60 Hz (see e.g. [Didyk et al. 2010]).

Mathematically, the light field emitted by a duallayer display is modeled as the outer product of both layers [Lanman et al. 2010]. Any outer vector product, however, only creates a rank-1 matrix approximation. A perceptual average over multiple high-speed frames overcomes this limitation in rank, providing a rank-M approximation of a target light field as

$$
\begin{aligned}
f_{\mathrm{LR}_2}(x, \nu) &= \frac{1}{M} \sum_{m=1}^{M} \left(\Xi_m^{(f)}(x + d_1\nu) \cdot \Xi_m^{(b)}(x + d_2\nu) \right) \\
&= \frac{1}{M} \mathbf{F}\mathbf{G}^T,
\end{aligned}
\tag{8}
$$

where $\mathbf{F} = (\Xi_m^{(f)})_m$ is a matrix whose columns represent the pixel values of the front layer for a single subframe, and $\mathbf{G} = (\Xi_m^{(b)})_m$ is the analogous matrix for the back layer.

Whereas low-rank light field displays can employ similar layered display designs as tomographic light field displays, only differing in refresh rates, the combination of optical light modulation (multiplication) and perceptual integration (summation) prevents linear solutions in the log-domain. Nonlinear light field decompositions can be found using matrix and tensor factorization. Compared to tomographic 3D displays, low-rank displays achieve significantly wider fields of view. Unfortunately, this optical benefit increases the resolution demands even more, especially in the angular domain. Without a dense-enough angular sampling, that is sufficiently many light field views, aliasing is observed that degrades 2D and 3D image quality (Fig. 7, top row). However, we can adapt the proposed framework to solve the inverse problem of Equation 8 by initializing all pixel states with random values and then using the following multiplicative update rules in the optimization step of the
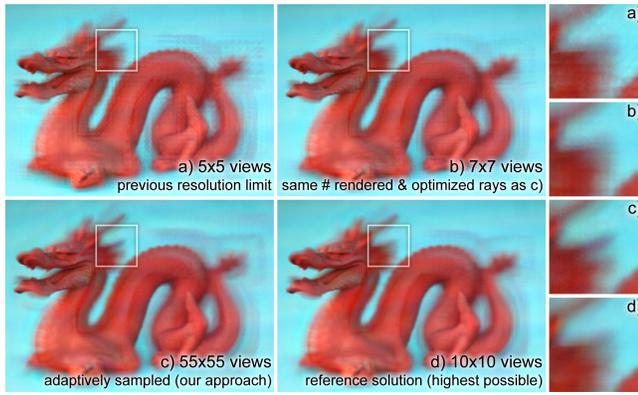
a) 5x5 views
previous resolution limit

b) 7x7 views
same # rendered & optimized rays as c)

c) 55x55 views
adaptively sampled (our approach)

d) 10x10 views
reference solution (highest possible)

Figure 7: *Photographs of dual-layer prototype. Brute-force optimization using the full light field is only feasible with a limited number of target views, as shown for previous resolution limits (a) and the highest resolution we could process (d). Using the proposed framework, it becomes feasible to use very high angular resolutions by sampling the target light field adaptively (c). This allows for smoother reconstructions, enhancing 2D and 3D image quality, as compared to a brute-force solution with the same number of rendered and optimized rays (b).*

algorithm:

$$\mathbf{F}_{\mathbb{S}^k} \leftarrow \mathbf{F}_{\mathbb{S}^k} \circ ((\mathbf{W}_{\mathbb{S}^k} \circ \mathbf{L}_{\mathbb{S}^k})\mathbf{G}_{\mathbb{S}^k} - \lambda \mathbf{F}_{\mathbb{S}^k}) /\!\!/ ((\mathbf{W}_{\mathbb{S}^k} \circ (\mathbf{F}_{\mathbb{S}^k}\mathbf{G}_{\mathbb{S}^k}^T))\mathbf{G}_{\mathbb{S}^k})$$
$$\mathbf{G}_{\mathbb{S}^k} \leftarrow \mathbf{G}_{\mathbb{S}^k} \circ (\mathbf{F}_{\mathbb{S}^k}^T(\mathbf{W}_{\mathbb{S}^k} \circ \mathbf{L}_{\mathbb{S}^k}) - \lambda \mathbf{G}_{\mathbb{S}^k}) /\!\!/ (\mathbf{F}_{\mathbb{S}^k}^T(\mathbf{W}_{\mathbb{S}^k} \circ (\mathbf{F}_{\mathbb{S}^k}\mathbf{G}_{\mathbb{S}^k}^T)))$$

$$(9)$$

where here $\circ$ and $/\!\!/$ are element-wise multiplication and division. These update rules are adaptations of Equation 11 of [Lanman et al. 2010], where $\mathbf{F}_{\mathbb{S}^k}$ and $\mathbf{G}_{\mathbb{S}^k}$ are full-sized matrices representing the pixel values of the front and back layer optimized only for the current sample set, while $\mathbf{W}_{\mathbb{S}^k}$ is a binary mask selecting only those entries directly affected by that sample set.

Note that we also employ a weak Tikhonov-regularizer, i.e. $\Gamma = \lambda\|\mathbf{F}\|_2^2 + \lambda\|\mathbf{G}\|_2^2$, to the pixel states. Similar update rules can be derived for the full tensor model including displays with more than two layers or directional backlighting [Wetzstein et al. 2012].

Using Equation 9 and our adaptive framework allows us to work with very high resolutions or even continuous sampling for the angular domain. Figure 1 shows a light field of the San Miguel scene with 85×21 angular views with a spatial resolution of 840×525. The dragon scene in Figure 7 (c) has 55×55 views of the same spatial resolution. In both examples, we render 2.4 million new sample rays per iteration. For Figure 1, after 21 iterations 30 million ray samples (3.82%) are accepted. Figure 7 only requires 16 iterations, after which 22 million rays are accepted (1.62%), hence actually rendered. The total number of sampled locations, including those that were rejected (not rendered) by Algorithm 3, corresponds to 6.40% and 2.87% of the total light field, respectively. These numbers illustrate significant savings on the rendering side. Since the angular resolution in these examples is even higher than in the example from Section 5.1, and since the low-rank approximation is computationally more expensive than the SART algorithm, solving a low-rank optimization problem on the full target light field would have been impossible on the compute hardware available to us.

5.3 Animations in Low-rank Light Field Displays

Low-rank light field approximations are created by displaying patterns at speeds beyond the critical flicker fusion threshold of the ob-

server. Whereas each of the displayed patterns represents a rank-1 light field, their perceptual average creates a higher-rank approximation. These techniques have been successfully applied to static targets in the past [Lanman et al. 2010; Wetzstein et al. 2012]. First attempts to create animated light fields were also shown. However to generate these, each animation frame was processed separately. This approach models the temporal response of the human visual system as a stop-motion system that is perfectly synchronized with the displayed animation—unfortunately, this is not the case in practice. Visual artifacts occur when the last patterns of one animation frame are perceptually merged with the first patterns of the next frame, because their interaction is not accounted for in the image synthesis (see Fig. 8, left).

To overcome this limitation, the order of high-speed patterns must be considered and interactions between patterns of neighboring animation frames accounted for. This can be done by modeling the perception of such patterns as a temporal low-pass filter linking consecutive animation frames. Extending Equation 8 to include a low-pass filter $g(t)$, via a convolution operator $\otimes$, yields

$$f_{\text{LR}_2}(x, \nu, t) = g(t) \otimes \left(\Xi^{(1)}(x + d_1\nu, t) \cdot \Xi^{(2)}(x + d_2\nu, t)\right).$$
$$(10)$$

This problem ties together all frames in a very large optimization problem making it intractable in both memory and compute time. However, we can make the problem more tractable by smoothing only backwards in time. To this end, we use the results of the optimization for the previous frame, and compute the impact of its subframes to the current frame. We then subtract out this contribution, and use the approach from Section 5.2 to solve for the difference. This simple approach significantly reduces the temporal artifacts produced by low-rank display devices, while the sparseness of our sampling framework makes the method feasible.

Figure 8 compares photographs of several successive frames of an animation displayed on a duallayer light field display prototype. The observer perceptually averages over three high-speed frames for each of the animation frames. Processing these separately (left column) results in visual artifacts for perceived images that are in between the animation frames, especially around depth discontinuities. By enforcing the temporal consistency as described above, we can mitigate these artifacts (right column) and generate temporally-consistent light field animations on low-rank displays. Remaining artifacts are due to the rank-3 light field approximation and could be removed with higher-speed LCD panels.

5.4 High Dynamic Range Displays

The proposed framework is also applicable to 2D HDR displays [Seetzen et al. 2004], which are composed of a high-resolution LCD panel and a low-resolution LED backlight that is optically blurred to produce a smooth light distribution. Similar to layered light field displays, the image formation can be modeled as a multiplication of two layers, in this case with an additional convolution of the backlight

$$f_{\text{HDR}}(x) = \Xi^{(f)}(x) \cdot \left(h(x) \otimes \Xi^{(b)}(x)\right),$$
$$(11)$$

where $h(x)$ is the point spread function (PSF) of the LED backlight $\Xi^{(b)}$. Since the full optimization problem is considered too expensive [Trentacoste et al. 2007], it is usually approximated with a greedy heuristic, where the LED values are determined first using simple image processing operators, and the LCD image is then determined by per-pixel division. However, this greedy approach is known to produce a number of artifacts, especially in regions with high spatial frequencies. A full optimization was deemed too expensive in the past, but can be approached with our framework.
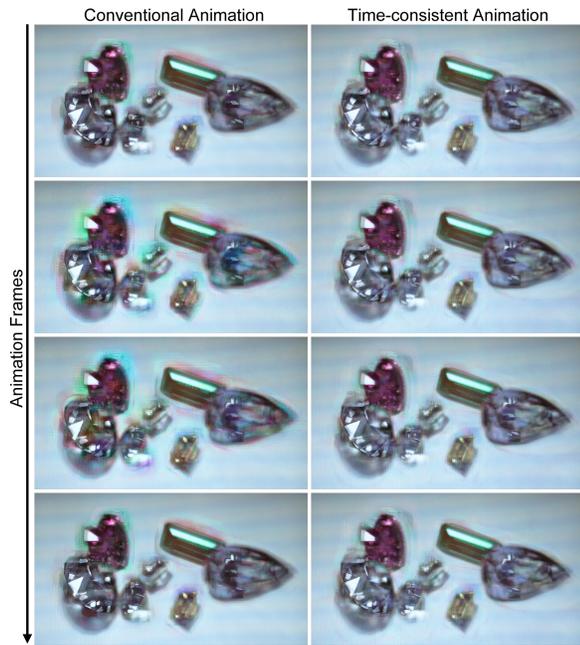
Figure 8: *Animations in low-rank light field displays. A low-rank approximation of a target light field is created by perceptually averaging over a set of high-speed patterns. Processing successive animation frames independently leads to visual artifacts (left), because a perceptual average of patterns in between animation frames is not accounted for. The proposed framework incorporates this perceptual effect and facilitates temporally-consistent low-rank light field animations. See supplemental video for details.*

Like low-rank displays, HDR displays may also suffer from temporal artifacts when animations are processed. In the case of HDR displays, one disturbing artifact known as the "walking LED" problem occurs for small, bright moving objects such as the comet in Figure 9: in order to reach the desired brightness for the bright object, a relatively large region of LEDs may have to be switched on, which results in a faint halo around the object. For static scenes this halo is not very noticeable, and cannot easily be distinguished from glare in the human visual system. For moving objects, however, the slightly changing shape of the halo as different LED are switched on and off may become noticeable and distracting (see Figure 9, second column).

To account for the temporal changes in the LED patterns, we extend the PSF $h(x)$ so that it models a spatio-temporal blur $h(x,t)$. The final optimization problem for determining the display parameters is then given as

$$\begin{array}{cc} \underset{\{\xi_f, \xi_b\}}{\arg\min} & \left\| \mathbf{1} - \boldsymbol{\xi}_f \circ \mathbf{H}\boldsymbol{\xi}_b \right\|_2^2 + \Gamma\left(\boldsymbol{\xi}_b\right) \\ \text{subject to} & 0 \le \boldsymbol{\xi}_f, \boldsymbol{\xi}_b \le 1 \end{array}, \qquad (12)$$

where $\mathbf{H}$ is a matrix incorporating the spatio-temporal blur. The regularizer may be included for spatial smoothness or to bias the solution towards lower LED values, for instance to save power.

Unlike the previously mentioned display technologies, a full sample set of the HDR target image is required in this application, since the LCD panel operates at the full image resolution. We therefore assume that the HDR image is fully available up-front, so that we do not improve the rendering times with the proposed framework. However, as in the earlier examples, the optimization problem can be solved with a small fraction of total pixel values which makes
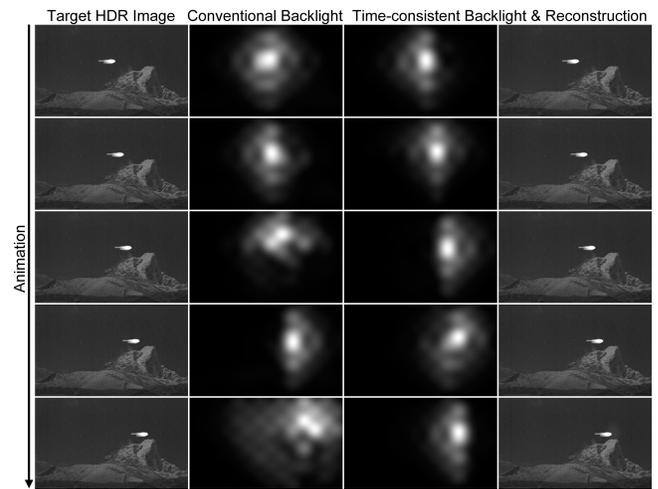

Figure 9: *Temporally-consistent image synthesis for high dynamic range displays. Five frames of an animation are decomposed into the patterns displayed on a low-resolution LED backlight (center columns) and a high-resolution LCD image (not shown). Using a conventional approach (center left), the backlight exhibits temporal discontinuities. These artifacts are removed with our adaptive optimization, resulting in temporally-consistent HDR animations.*

it not only feasible to use a full iterative solver for each static image, but also to tackle the walking LED problem for the first time. Smooth animations, without temporal discontinuities of the backlight, are achieved as shown in Figure 9 (right columns) and in the supplemental video.

5.5 Optical Pixel Sharing

The final compressive display technology we consider here are super-resolution projectors based on Optical Pixel Sharing [Sajadi et al. 2012]. Projectors based on this technology use two low-resolution "layers", such as LCD panels, to generate a super-resolution image as a sum of two sequential time steps. In the first time step, a low-resolution image is generated using only a "front" layer $\Xi^{(f,1)}$. In the second step, an edge image is projected by replicating a second, "back" layer image $\Xi^{(b,2)}$ over the front layer using an optical pixel sharing operator, or *jumbling function* h. This replicated image is optically multiplied with the content of the front layer $\Xi^{(f,2)}$. The image formation model thus becomes

$$\begin{aligned} f_{\text{OPS}}(x) &= \left(u(x) \otimes \Xi^{(f,1)}(x) \right) \\ &+ \left(u(x) \otimes \Xi^{(f,2)}(x) \right) \cdot \left(h(x) \otimes \Xi^{(b,2)}(x) \right), \end{aligned} \qquad (13)$$

where u is a suitable upsampling operator that maps the low-resolution image to the full target image resolution. Since the jumbling function h replicates the pixels of the back layer in different regions of the front panel, it does not have a spatially compact support, but it is still a convolution kernel with a relatively small number of non-zero values.

Note that the first term in Equation 13 can be accounted for by computing a low-resolution version of the image and subtracting it from the high-resolution target image. In the original work by Sajadi et al. [2012], the remaining optimization problem is then solved for binary values of the second front layer image $\Xi^{(f,2)}$, which acts purely as a blocker. This requires a combinatorial problem to be solved, as described in their paper. However, consistent with the
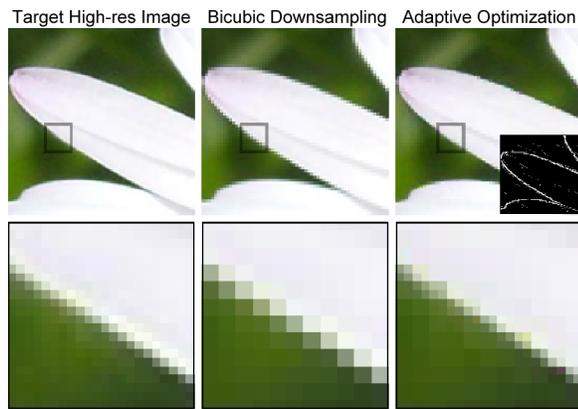
Figure 10: *High-resolution display through optical pixel sharing. A high-resolution target image (left) is downsampled for presentation on a display with half the resolution in both width and height (center). Optical pixel sharing allows for higher-resolution display (right); the patterns are computed with the proposed algorithm using a sparse set of samples of the high-resolution target image (inset, upper right).*

actual hardware capabilities, we can relax this requirement by allowing for a semi-transparent front layer, and optimize the resulting image in a least-squares sense. This yields an objective function with a mathematical structure that is identical to the case of HDR displays (Eq. 12) and can therefore be solved in a similar fashion.

Figure 10 shows a simulated result for a pixel sharing hardware with 2×2 super-resolution. We see that the adaptive sampling framework quickly hones in on strong edges in the low-resolution image and manages to improve sharpness of these edge regions significantly using only a fraction of the full-resolution image pixels.

Our adaptive optimization framework for compressive displays suggests that an alternative image formation model could be developed for the Optical Pixel Sharing hardware. In particular, by moving away from the constraint that the first projected sub-image must be a low-resolution version of the target image, one ends up with an optimization problem where both the front and the back panel are optimized for each sub-frame. Such an image formation model could potentially be approached with methods similar to the low-rank matrix factorization used for light field displays (Section 5.2). We leave this idea as an interesting topic for future work.

6   Discussion

In this paper, we introduce an adaptive image synthesis framework that is tailored to emerging compressive displays. This framework unifies sampling, rendering, and display-specific optimizations. Markov Chain Monte Carlo methods are used to sample high-dimensional light fields. While such methods have been used in rendering applications for years, we derive a new algorithm that alternates between sampling and optimization to adapt to limitations of the underlying display hardware, characteristics of the human visual system, and content-dependent light field structures. Using this framework, we significantly lower the computational resources required for synthesizing content for compressive displays and thereby facilitate higher resolutions and better 3D effects than previously possible. We demonstrate the first solution for generating perceptually correct animations for time-multiplexed low-rank light field displays and show that high dynamic range as well as high-resolution displays can be driven more efficiently and with a higher quality.

The stochastic optimization framework introduced in this paper maps well to parallel processors, such as GPUs. Due to the use of parallel sampling chains, the sampling stage is highly parallel. In many cases, the optimization can also be parallelized efficiently; algebraic tomographic reconstructions and matrix factorization problems, for instance, directly benefit from sparse matrix-vector multiplication routines implemented by most modern GPU interfaces. In all examples, we use enough chains ($> 50{,}000$) to fully utilize the parallel processing capabilityes of modern GPUs. We therefore believe that a parallel implementation of our algorithm would give large benefits. In this paper, however, we do not explore or provide such a fully-optimized reference implementation.

The proposed framework unifies rendering, data processing, and display. While we demonstrate how this is useful for many different compressive displays, we do not propose a new display technology but rather unlock the potential of existing designs and provide the means to generate high-quality content for them. Our stochastic framework may be useful for pure rendering and computational photography, but exploring these applications in detail is outside the scope of this paper.

In the future, we would like to integrate more sophisticated models for the human visual system into our system, for instance sensitivity to contrast [Mantiuk et al. 2011] and disparity [Didyk et al. 2011]. Combined with sparse coding techniques, the proposed framework may find applications in compressive rendering [Sen and Darabi 2011] or rendering in general. Finally, we believe that our algorithm will be useful for a variety of adaptive computational photography and medical imaging techniques, were a minimal number of measured observations is desirable, for instance to reduce a patient's exposure to radiation in X-ray computed tomography.

7   Conclusion

Compressive display technologies leverage the co-design of display optics and compressive computation to overcome fundamental limitations of purely optical designs. While a variety of compressive displays has been proposed to improve characteristics such as dynamic range, resolution, and glasses-free 3D, the problem of efficiently generating content for them is not solved. We believe that future display technologies will blur the boundaries between optical design, numerical optimization, computational perception, and rendering. With the methods presented in this paper, we take a significant step towards a new graphics pipeline that incorporates all of these aspects in an adaptive manner.

Acknowledgements

References

ADELSON, E. H., AND BERGEN, J. R. 1991. The plenoptic function and the elements of early vision. In *Comp. Models of Visual Processing*, 3–20.

BERTSEKAS, D. 1997. A New Class of Incremental Gradient Methods for Least Squares Problems. *SIAM Journal on Optimization 7*, 4, 913–926.

COOK, R., PORTER, T., AND CARPENTER, L. 1984. Distributed ray tracing. In *Proc. SIGGRAPH*, 137–145.

COSSAIRT, O. S., NAPOLI, J., HILL, S. L., DORVAL, R. K., AND FAVALORA, G. E. 2007. Occlusion-Capable Multiview Volumetric Three-Dimensional Display. *Applied Optics 46*, 8, 1244–1250.

DIDYK, P., EISEMANN, E., RITSCHEL, T., MYSZKOWSKI, K., AND SEIDEL, H.-P. 2010. Apparent Display Resolution Enhancement for Moving Images. *ACM Trans. Graph. (SIGGRAPH) 29*, 4, 113:1–113:8.

DIDYK, P., RITSCHEL, T., EISEMANN, E., MYSZKOWSKI, K., AND SEIDEL, H.-P. 2011. A Perceptual Model for Disparity. *ACM Trans. Graph. (SIGGRAPH) 30*, 4, 96:1–96:10.

EGAN, K., TSENG, Y.-T., HOLZSCHUCH, N., DURAND, F., AND RAMAMOORTHI, R. 2009. Frequency Analysis and Sheared Reconstruction for Rendering Motion Blur. *ACM Trans. Graph. (SIGGRAPH) 28*, 3, 93:1–93:13.

EGAN, K., HECHT, F., DURAND, F., AND RAMAMOORTHI, R. 2011. Frequency Analysis and Sheared Filtering for Shadow Light Fields of Complex Occluders. *ACM Trans. Graph. 30*, 2, 9:1–9:13.

FRIEDLANDER, M., AND SCHMIDT, M. 2012. Hybrid deterministic-stochastic methods for data fitting. *SIAM Journal on Scientific Computing 34*, 3, 1380–1405.

GLASSNER, A. S., FISHKIN, K. P., MARIMONT, D. H., AND STONE, M. C. 1995. Device-Directed Rendering. *ACM Trans. Graph. 14*, 1, 58–76.

GOTODA, H. 2011. Reduction of Image Blurring in an Autostereoscopic Multilayer Liquid Crystal Display. In *Proc. SPIE Stereoscopic Displays and Applications XXII*, vol. 7863, 21:1–21:7.

GREGSON, J., KRIMERMAN, M., HULLIN, M. B., AND HEIDRICH, W. 2012. Stochastic Tomography and its Applications in 3D Imaging of Mixing Fluids. *ACM Trans. Graph. (SIGGRAPH) 31*, 4, 52:1–52:10.

GROSSE, M., WETZSTEIN, G., GRUNDHÖFER, A., AND BIMBER, O. 2010. Coded Aperture Projection. *ACM Trans. Graph. 29*, 3, 22:1–22:12.

HASTINGS, W. 1970. Monte carlo sampling methods using markov chains and their applications. *Biometrika 57*, 1, 97–109.

IVES, F. E., 1903. Parallax Stereogram and Process of Making Same. U.S. Patent 725,567.

JONES, A., MCDOWALL, I., YAMADA, H., BOLAS, M., AND DEBEVEC, P. 2007. Rendering for an interactive 360° light field display. *ACM Trans. Graph. (SIGGRAPH) 26*, 40:1–40:10.

LANMAN, D., HIRSCH, M., KIM, Y., AND RASKAR, R. 2010. 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.

LANMAN, D., WETZSTEIN, G., HIRSCH, M., HEIDRICH, W., AND RASKAR, R. 2011. Polarization Fields: Dynamic Light Field Display using Multi-Layer LCDs. *ACM Trans. Graph. (SIGGRAPH Asia) 30*, 186:1–186:9.

LEHTINEN, J., AILA, T., CHEN, J., LAINE, S., AND DURAND, F. 2011. Temporal Light Field Reconstruction for Rendering Distribution Effects. *ACM Trans. Graph. (SIGGRAPH) 30*, 4, 55:1–55:12.

LEHTINEN, J., AILA, T., LAINE, S., AND DURAND, F. 2012. Reconstructing the Indirect Light Field for Global Illumination. *ACM Trans. Graph. (SIGGRAPH) 31*, 4, 51:1–51:10.

LI, T.-M., WU, Y.-T., AND CHUANG, Y.-Y. 2012. SURE-based Optimization for Adaptive Sampling and Reconstruction. *ACM Trans. Graph. (SIGGRAPH Asia) 31*, 6, 194:1–194:9.

LIPPMANN, G. 1908. Épreuves réversibles donnant la sensation du relief. *Journal of Physics 7*, 4, 821–825.

MANTIUK, R., KIM, K. J., REMPEL, A. G., AND HEIDRICH, W. 2011. HDR-VDP-2: A Calibrated Visual Metric for Visibility and Quality Predictions in all Luminance Conditions. *ACM Trans. Graph. (SIGGRAPH) 30*, 4, 40:1–40:14.

MARWAH, K., WETZSTEIN, G., BANDO, Y., AND RASKAR, R. 2013. Compressive Ligth Field Photography using Overcomplete Dictionaries and Optimized Projections. *ACM Trans. Graph. (SIGGRAPH) 32*, 4, 1–11.

METROPOLIS, N., ROSENBLUTH, A., ROSENBLUTH, M., TELLER, A., AND TELLER, E. 1953. Equation of state calculations by fast computing machines. *The journal of chemical physics 21*, 1087–1092.

PHARR, M., AND HUMPHREYS, G. 2010. *Physically based rendering: From theory to implementation*. Morgan Kaufmann.

SAJADI, B., GOPI, M., AND MAJUMDER, A. 2012. Edge-guided Resolution Enhancement in Projectors via Optical Pixel Sharing. *ACM Trans. Graph. (SIGGRAPH) 31*, 4, 79:1–79:122.

SEETZEN, H., HEIDRICH, W., STUERZLINGER, W., WARD, G., WHITEHEAD, L., TRENTACOSTE, M., GHOSH, A., AND VOROZCOVS, A. 2004. High Dynamic Range Display Systems. *ACM Trans. Graph. (SIGGRAPH) 23*, 3, 760–768.

SEN, P., AND DARABI, S. 2011. Compressive Rendering: A Rendering Application of Compressed Sensing. *IEEE TVCG 17*, 4, 487–499.

SEN, P., DARABI, S., AND XIAO, L. 2011. Compressive Rendering of Multidimensional Scenes. In *LNCS "Video Processing and Computational Video"*, 152–183.

SOLER, C., SUBR, K., DURAND, F., HOLZSCHUCH, N., AND SILLION, F. 2009. Fourier Depth of Field. *ACM Trans. Graph. 28*, 2, 18:1–18:12.

TRENTACOSTE, M., HEIDRICH, W., WHITEHEAD, L., SEETZEN, H., AND WARD, G. 2007. Photometric Image Processing for High Dynamic Range Displays. *JVCIR 18*, 5, 439–451.

VEACH, E., AND GUIBAS, L. J. 1997. Metropolis Light Transport. In *Proc. SIGGRAPH*, 65–76.

WETZSTEIN, G., LANMAN, D., HEIDRICH, W., AND RASKAR, R. 2011. Layered 3D: Tomographic Image Synthesis for Attenuation-based Light Field and High Dynamic Range Displays. *ACM Trans. Graph. (SIGGRAPH) 30*, 95:1–95:11.

WETZSTEIN, G., LANMAN, D., HIRSCH, M., AND RASKAR, R. 2012. Tensor Displays: Compressive Light Field Synthesis using Multilayer Displays with Directional Backlighting. *ACM Trans. Graph. (SIGGRAPH) 31*, 80:1–80:11.

WIDROW, B., AND STEARNS, S. 1985. *Adaptive signal processing*, vol. 1. Prentice-Hall.