Over- and Under-Exposure Reconstruction of a Single Plenoptic Capture

WEI HU, Peking University, China
MOZHDEH SEIFI, Technicolor, France
ERIK REINHARD, Technicolor, France

Light field images, for example taken with plenoptic cameras, offer interesting post-processing opportunities, including depth-of-field management, depth estimation, viewpoint selection, and 3D image synthesis. Like most capture devices, however, plenoptic cameras have a limited dynamic range, so that over- and under-exposed areas in plenoptic images are commonplace. We therefore present a straightforward and robust plenoptic reconstruction technique based on the observation that vignetting causes peripheral views to receive less light than central views. Thus, corresponding pixels in different views can be used to reconstruct illumination, especially in areas where information missing in one view is present in another. Our algorithm accurately reconstructs under- and over-exposed regions (known as declipping), additionally affording an increase in peak luminance by up to 2 f-stops, and a comparable lowering of the noise floor. The key advantages of this approach are that no hardware modifications are necessary to improve the dynamic range, that no multiple exposure techniques are required, and therefore that no ghosting or other artefacts are introduced.

CCS Concepts: • **Computing methodologies** → **Computational photography**; **Computational photography**; **Image processing**; *Image representations*;

Additional Key Words and Phrases: Light field acquisition, light field reconstruction, declipping, over-exposure, under-exposure, dynamic range

ACM Reference Format:
Wei Hu, Mozhdeh Seifi, and Erik Reinhard. 2018. Over- and Under-Exposure Reconstruction of a Single Plenoptic Capture. *ACM Trans. Multimedia Comput. Commun. Appl.* 1, 1, Article 1 (January 2018), 21 pages.
https://doi.org/0000001.0000001

1 INTRODUCTION

Conventional cameras record only the total sum of light rays striking each point on the sensor. This means that light is integrated over all angles of incidence, as well as over the surface area of each pixel. In comparison, plenoptic cameras, constructed by gluing a sheet of micro-lenses to the sensor, retain angular information to some extent [2, 20, 44], so that they can be used to directly capture light fields [18, 30]. While this concept has been known for more than a century [27], practical implementations have emerged only recently [2, 34, 44]. The additional angular information enables a variety of post-processing applications such as refocusing [26, 44], depth estimation [4, 53, 59, 66, 69], glare

Authors' e-mail addresses: W. Hu, forhuwei@pku.edu.cn; M. Seifi, mozhde.seifi@gmail.com; E. Reinhard, Erik.Reinhard@technicolor.com.

1551-6857/2018/1-ART1 $$15.00
https://doi.org/0000001.0000001

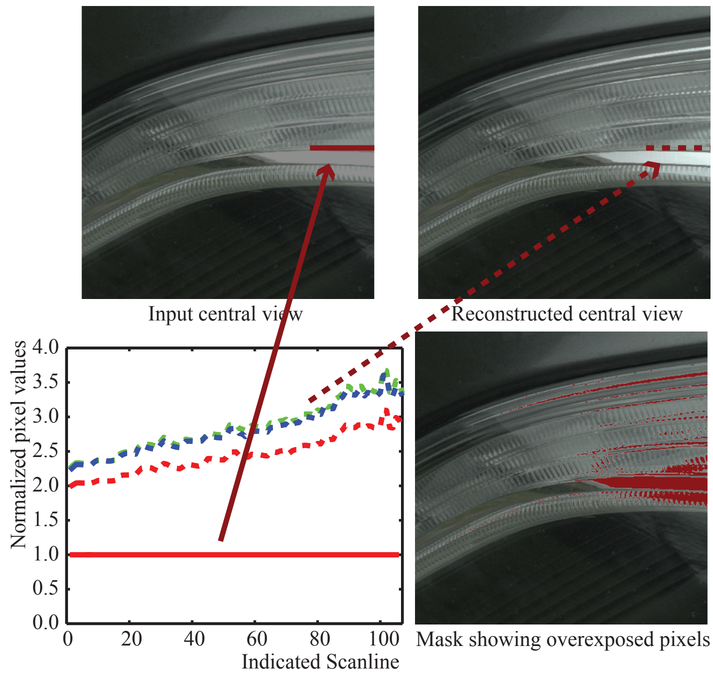

Fig. 1. Redundancy as well as vignetting in plenoptic images can be exploited to reconstruct over- and under-exposed regions (the botton right image shows over-exposed pixels colored red). The marked pixel rows in the input (top left) and reconstructed central view (top right) are plotted for the red, green and blue color channels (bottom left), showing that in this example of a car headlamp the reconstructed image has pixel values more than 3.5 times higher than the input. All input pixels in this row were over-exposed in all three color channels. For visualisation, all images were jointly tonemapped to maintain relative pixel values (Photographic operator [51]).

reduction [47], material recognition [64], intrinsic imaging [14], 3D imaging from a single shot [45], dolly zoom effects [46] and 3D microscopy [31].

Here, more so than in conventional photography, there is a need to maximize the number of sensor elements [33] to accommodate a reasonable angular resolution without unduly sacrificing spatial resolution [17]. Although super-resolution techniques may mitigate resolution limitations [5], this requirement directly competes with the desire to improve a sensor's dynamic range [50] (p. 687), so that fewer areas are under- or over-exposed.

One way to mitigate dynamic range limitations is to pursue high dynamic range (HDR) light field capture, as it would enhance image fidelity, but also opens up the possibility to use light fields as advanced light sources in rendering applications [40, 61]. Standard multiple exposure techniques could be employed to generate high dynamic range plenoptic imagery [10, 38, 42, 52], but this would require a change in photographic practices as a scene would have to be captured multiple times. In addition, these methods have a tendency to introduce ghosting artefacts as a result of camera movement or changes in scene composition, requiring specialized and often complicated ghost removal techniques [13, 19, 28, 56, 67]. Further, ghost removal as well as image alignment may be more complicated for light field data due to its spatio-angular nature.

Alternatively, to capture HDR light field imagery, hardware-based solutions could be considered, including the use of a camera add-on [37], adding a neutral density filter array [15, 43], or adding

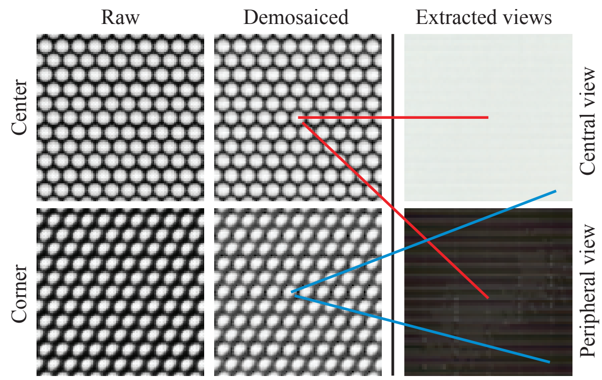

Fig. 2. Micro-lenses cause significant signal attenuation near their edges, as shown here using central and corner sections of a raw plenoptic image of a diffuse white surface (a white image, left). After view extraction, this causes peripheral views to be darker than central views (right). The red and blue lines indicate the mapping between sections of the white image and the extracted views. The images are zoomed in and cropped for the sake of visualization.

apertures of different sizes to each micro-lens [16]. Alternatively, a plenoptic camera may be constructed by replacing the micro-lens array with a pinhole array, and adding a filter array in the pupil plane [21]. The pinhole array would reduce design tolerances on the filter array, while the latter could be designed as neutral density filters such that high dynamic range captures may be realised. Further variants include taking multiple exposures of a grid of mirrored spheres [61], by using a high dynamic range omni-directional camera mounted on a sled [60], or by using a high dynamic range camera in combination with a planar mirror [25].

Without changing photographic practices or requiring hardware modifications, it is possible to reduce the occurrence of under- and over-exposed areas by exploiting redundancy present in images. For example, redundancy in sequences of video frames may be exploited to increase dynamic range using techniques such as generalized mosaicing [54].

In contrast, we propose to leverage the redundancy available in single standard dynamic range (SDR) plenoptic captures to minimize exposure problems (see Figure 1 for an example), in a straightforward post-process that avoids the need for hardware redesigns. It is based on the insight that a given point in a scene is imaged multiple times, albeit under different amounts of light attenuation as a consequence of vignetting. A significant portion of vignetting in plenoptic cameras is due to the micro-lenses (see Figure 2), and is typically removed in a preprocess [3, 6, 9, 12, 53]. We note that our technique is not intended for use with camera designs that do not exhibit vignetting, including those based on coded apertures [55, 62].

Rather than treating vignetting as undesirable, we exploit it prior to its removal to reconstruct and enhance light field captures, notably without approximations normally encountered in declipping [1], inpainting [8], hallucination [63] or other standard data synthesis techniques that could conceivably be employed. Note that our algorithm could be seen as a declipping algorithm for plenoptic captures. A key distinction, however, is that declipping algorithms for conventional images synthesize texture or patterns in over- or under-exposed areas, whereas our algorithm has access to the captured information in a light field, so that the light field is reconstructed, rather than synthesized.

Demosaicing and view reconstruction are necessary pre-processing steps for many tasks, including our algorithms. However, they tend to be less reliable for peripheral views [70]. For this reason, our algorithm includes a signal adaptive filtering step to mitigate their effects. This is achieved

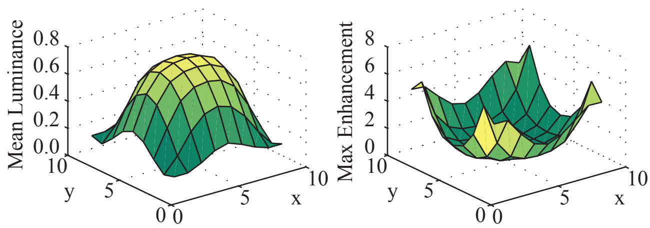

Fig. 3. The mean luminance of each reconstructed view of a white image (left) shows significant vignetting in peripheral views. The ratio between the luminance of the central pixel over the luminance of each peripheral pixel (right) provides the potential ratio of the central pixel enhancement in dynamic range in the case the corresponding peripheral view is used. Therefore, the dynamic range of the central view may be increased by factors of up to 6 in this specific example.

by exploiting recent results from graph signal processing [57], notably the use of a graph-signal smoothness prior [22, 41]. Note that all views may be processed, for both recovering over-exposed pixels, as well as under-exposed pixels, thus producing a full reconstructed 4D light field. In summary, our key contributions are as follows:

- To our knowledge, we are the first to exploit vignetting in plenoptic cameras as a feature to declip a plenoptic light field from a single plenoptic capture.
- We present a novel mathematical formulation for reconstructing poorly exposed pixels in one view from well-exposed corresponding pixels in other views, leveraging a graph-signal smoothness prior based on a graph that encodes inter-pixel similarities of the corresponding pixels. This induces tolerance against noise and propagation of pre-processing artefacts.
- Finally, we developed a computationally efficient algorithm to recover poorly exposed pixels iteratively until convergence.

2 VIGNETTING

Vignetting in conventional photographs is seen as a radial dimming of the image toward the image borders. It is caused by various optical phenomena occuring in the main lens and on the sensor, such as mechanical, optical and pixel vignetting. Specific causes include off-axis illumination fall-off due to the $\cos^4$ law, the change of apparent shape of the lens when viewed from locations further from the optical axis, and light incident upon photo-wells at oblique angles being transduced less efficiently [71].

In plenoptic cameras these effects are strongly exacerbated by the presence of a sheet of micro-lenses (Figure 2, left). The amount of vignetting present determines the level of correction that any algorithm might accomplish. For a first generation Lytro camera, a representative plenoptic capture of a diffuse white surface was analysed for this purpose. Individual views were reconstructed (example views are shown in Figure 2, right) and the resulting mean luminance values for each view are plotted in Figure 3 (left). For the central view, the maximum dynamic range enhancement that may be achieved as function of each peripheral view is shown in Figure 3 (right). For this specific example, the corner views allow a dynamic range increase by a factor of up to 6.3. Further analysis has shown that this value varies as function of focus distance for the first generation Lytro camera. Particularly for very long focus distances we found that this value reduces to 2.2. Note that such long focus distances are uncommon in light-field imaging because the ability to perform meaningful edits to such light-fields becomes minimal due to the very small disparities associated with such focusing. Dependent on focus distance and amount of zoom applied, the potential to reconstruct dynamic range may vary. The enhancement of dynamic range is an indicator of how effective a reconstruction algorithm may be in correcting poorly exposed pixels.

For each pixel location $\mathbf{x} \in \mathbb{N}^2$ and view $\mathbf{k} \in \mathbb{N}^2$, we denote the amount of vignetting/attenuation with $w_\mathbf{k}(\mathbf{x})$. Values of vignetting may be obtained from a model of vignetting [24, 29, 32], or, as used in this paper, an appropriately chosen white image $\mathbf{W}$ (an image taken of a uniform white surface) that encapsulates the sum total of all vignetting effects [6, 9, 12]. Alternatively, the reconstructed views of the input image itself may be analysed to derive vignetting values as well, as outlined in Section 5.

With L_e the irradiance associated with a point in the scene, image irradiance is given by $w_\mathbf{k}(\mathbf{x})\, L_e$. The recorded pixel value $I_\mathbf{k}(\mathbf{x})$ relates to the image irradiance, exposure time Δt as well as the camera response function $f(\cdot)$:

$$I_\mathbf{k}(\mathbf{x}) = f\,(w_\mathbf{k}(\mathbf{x})\, L_e\, \Delta t). \tag{1}$$

By inverting this equation we can solve for the product of luminance and exposure time of the imaged scene point:

$$L_e\, \Delta t = f^{-1}(I_\mathbf{k}(\mathbf{x}))/w_\mathbf{k}(\mathbf{x}). \tag{2}$$

Here, the pixel value $I_\mathbf{k}(\mathbf{x})$ may be over- or under-exposed in one view but may be well-exposed in one or more other views. Assuming that due to disparity the corresponding pixel in view $\mathbf{n}$ is located at position $\mathbf{x_n}$, and that this pixel is well exposed, the value of $I_\mathbf{k}(\mathbf{x})$ can be reconstructed directly:

$$I_\mathbf{k}(\mathbf{x}) = f\left(\frac{w_\mathbf{k}(\mathbf{x})}{w_\mathbf{n}(\mathbf{x_n})}\, f^{-1}\,\left(I_\mathbf{n}(\mathbf{x_n})\right)\right). \tag{3}$$

Such a reconstruction would rely on disparity estimation algorithms to find pixel correspondences [4, 53, 66], perhaps in conjunction with additional neighborhood matching techniques [11, 72]. However, by definition the disparity for pixel $I_\mathbf{k}(\mathbf{x})$ is unknown, because this pixel was under- or over-exposed, meaning that evaluation of Equation (3) may not provide meaningful results. As a consequence, a more advanced algorithm is required to reconstruct under- and over-exposed pixel values. This motivates our solution, which will be presented in the following sections.

3 ALGORITHM OVERVIEW

The inputs to the algorithm are a plenoptic capture and its corresponding white image (the Lytro camera, for instance, is supplied with a library of white images taken under different camera settings). We begin by demosaicing [36], demultiplexing [6] and linearizing both inputs (line 3 in Algorithm 1; note that numbers in parentheses refer to section numbers). This gives us a matrix of views for both the image and the white image.

Next, we detect under- and over-exposed pixels by simple thresholding (line 4 in Algorithm 1). In practice, we find that the thresholds of 0.05 and 0.95 on the normalized image intensity work well. We explain our algorithm in terms of over-exposed pixels only, noting that under-exposed pixels are treated similarly (results are shown for both). We maintain a binary mask M indicating which pixels are over-exposed (and separately, a mask for under-exposed pixels).

For each over-exposed pixel, block matching is performed to find the corresponding blocks in the more strongly vignetted views (Section 4 and lines 7–8 in Algorithm 1). Block matching is a general technique based on the assumption that two pixels are similar if their immediate neighborhoods, defined by blocks centered around these pixels, are similar [72]. In our case, an over-exposed pixel may have well-exposed neighbors [see Figure 4 (left)]. If another block in a view with more vignetting has similar neighbors, it is considered matched [Figure 4 (middle)].

Then, the over-exposed pixel, as well as other over-exposed pixels in the same block, may be reconstructed from the matched block according to the available information in the latter (line 9 in Algorithm 1; see also Figure 4 (right) and Section 5). In its simplest form, this amounts to transferring

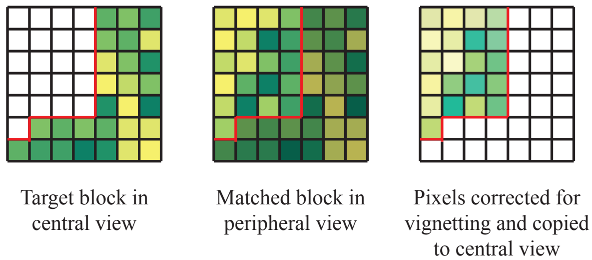

Fig. 4. A block in a central view containing over-exposed pixels (left), a matched block in a peripheral view with more vignetting (middle), and reconstructed pixels in the central view, taken from the well-exposed pixels in the matched block in the peripheral view.

Algorithm 1 Light Field Reconstruction
(sections listed in brackets)

1: **Input**: a plenoptic capture and a white image
2: **Output**: a reconstructed plenoptic capture
3: demosaic, demultiplex and linearize (3)
4: record poorly-exposed pixels $p_{\mathbf{k}}^{\mathbf{x}}$ in mask $\mathbf{M}$ (3)
5: **repeat**
6:     **for all** pixels $p_{\mathbf{k}}^{\mathbf{x}}$ at position $\mathbf{x}$ in view $\mathbf{k}$ **do**
7:         form a block $\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}$ centered around $p_{\mathbf{k}}^{\mathbf{x}}$ (4)
8:         find $\mathbf{b}_{\mathbf{n}}^{\mathbf{x}'}$ using block matching with $\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}$ (4)
9:         reconstruct $\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}$ using $\mathbf{b}_{\mathbf{n}}^{\mathbf{x}'}$ (5 or 6)
10:         update mask $\mathbf{M}$ (3)
11: **until** all poorly-exposed pixels are reconstructed.

the candidate block into the area surrounding the over-exposed pixel, using Equation (3), possibly with the aid of some blending.

This process is iterated for all over-exposed pixels in the views we wish to reconstruct, as long as they have enough well-exposed neighboring pixels to enable block matching. This allows regions of arbitrary size to be reconstructed. The minimum number of pixels that should be well-exposed in the block containing the over-exposed pixel is a non-critical parameter that trades the number of reconstructable pixels against the quality of the reconstruction. In practice, we found that requiring the block to have 50% well-exposed pixels provides a good trade-off.

We find that such a straightforward algorithm performs reasonably well in many cases, albeit that noise may be amplified, and inaccuracies may be introduced due to non-uniform disparity across pixels in the candidate patch. Further, neighboring over-exposed pixels may lead to the selection of candidate blocks that are not fully consistent with each other. Such discrepancies may be resolved by applying a graph-based reconstruction which includes a smoothness prior on the reconstructed view (Section 6).

Either of these two reconstruction algorithms yield results for over-exposed pixels that are within half a block's width from a set of well-exposed pixels. Within reason, larger over-exposed areas can be synthesized by repeated application of either reconstruction algorithms (lines 5, 11 in Algorithm 1). The output of the algorithm is a new matrix of views in which all over-exposed pixels (and under-exposed pixels) are reconstructed.

4 CANDIDATE BLOCK SELECTION

Our algorithm relies on neighborhood matching techniques to determine where in the matrix of views suitable information is available to reconstruct over-exposed pixels. A wide choice of algorithms is available, such as those used in texture synthesis [11, 68], or techniques inspired by template matching [58]. However, to account for the presence of vignetting, some modifications are necessary.

An over-exposed pixel may have corresponding pixels that are well-exposed in some other views as a result of stronger signal attenuation by vignetting. As disparity is limited in plenoptic cameras, a given over-exposed pixel $I_{\mathbf{k}}(\mathbf{x})$ at location $\mathbf{x}$ in a given target view $\mathbf{k}$ will have a corresponding pixel in a different view $\mathbf{n}$ within a rectangular neighborhood $\mathcal{N}_{\mathbf{k},\mathbf{n}}^{\mathbf{x}}$ centered at position $\mathbf{x}$ and of size $(2|\mathbf{n}-\mathbf{k}| \odot \mathbf{d}) + \mathbf{1}$, where $\mathbf{d}$ is a vector of the unsigned maximum horizontal and vertical disparities of the camera system, and $\odot$ denotes the Hadamard product. For a first generation Lytro camera, we find that the disparity is constrained by $\mathbf{d} = (2, 2)$[1].

To determine the corresponding pixel for a given over-exposed pixel, we resort to block matching [72]. The idea is that at least some neighbors of an over-exposed pixel will be well-exposed, and these can be matched in other views that have more vignetting. The well-exposed pixels in the matched blocks are then used to perform the reconstruction. Here, a $t \times t$ pixel block represented in vectorized form by $\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}$ is constructed around the over-exposed pixel at position $\mathbf{x}$ in view $\mathbf{k}$. In our experiments, we set $t = 13$, a value which balances accuracy against computational costs. The notation for indexing a single element in such a vector is given by $b_{\mathbf{k}}^{\mathbf{x}}(i)$.

The block matching algorithm is modified in the following key aspects to account for our specific use case. First, in each of the views $\mathbf{n}$ we only search for matching blocks $\mathbf{b}_{\mathbf{n}}^{\mathbf{x}'}$ where $\mathbf{x}' \in \mathcal{N}_{\mathbf{k},\mathbf{n}}^{\mathbf{x}}$. Second, the blocks $\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}$ and $\mathbf{b}_{\mathbf{n}}^{\mathbf{x}'}$ are different due to both vignetting and the presence of over-exposed pixels, requiring an adaptation of the distance metric used to assess the quality of a block match. To account for vignetting, we subtract the mean pixel luminance of both $\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}$ and $\mathbf{b}_{\mathbf{n}}^{\mathbf{x}'}$ to obtain $\hat{\mathbf{b}}_{\mathbf{k}}^{\mathbf{x}}$ and $\hat{\mathbf{b}}_{\mathbf{n}}^{\mathbf{x}'}$. Third, we apply a weighting operator $\omega(t) = 1 - (2\,t - 1)^{12}$ on elements of $\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}$ to reduce the influence of over-exposed pixels, where the input of ω is assumed to be normalized [49]. Our distance metric between two blocks thus becomes:

$$\left\| \omega(\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}) \odot \left(\hat{\mathbf{b}}_{\mathbf{k}}^{\mathbf{x}} - \hat{\mathbf{b}}_{\mathbf{n}}^{\mathbf{x}'} \right) \right\|_2^2 . \tag{4}$$

Block matching may be carried out in each of the candidate views, selecting the block $\mathbf{b}_{\mathbf{n}}^{\mathbf{x}'}$ for which $\hat{\mathbf{b}}_{\mathbf{n}}^{\mathbf{x}'}$ has the smallest distance to block $\hat{\mathbf{b}}_{\mathbf{k}}^{\mathbf{x}}$. We note that vignetting occurs most strongly in the most peripheral views, so that in principle the list of candidate views may be limited to those views. In Section 7 we will show, however, that in practice the search space can be limited even further without significant loss of quality. The over-exposed pixels in $\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}$ are then recovered using the corresponding pixels in $\mathbf{b}_{\mathbf{n}}^{\mathbf{x}'}$ as discussed in the following section.

5 SIMPLE IMAGE RECONSTRUCTION

The block $\mathbf{b}_{\mathbf{k}}^{\mathbf{x}}$ containing over-exposed pixels and its matched block $\mathbf{b}_{\mathbf{n}}^{\mathbf{x}'}$ form the input to our reconstruction algorithm. Our algorithm consists of two steps, namely correcting the candidate block for vignetting, followed by reconstruction of over-exposed pixels using the corrected candidate block.

[1] For clarity, note that with this definition of disparity, the total pixel displacement between views is the product of disparity d and the angular coordinate difference between the corresponding views.

Denoting the corresponding white image blocks of $\mathbf{b}_{\mathrm{k}}^{\mathbf{x}}$ and $\mathbf{b}_{\mathrm{n}}^{\mathbf{x}'}$ in vectorized form $\mathbf{w}_{\mathrm{k}}^{\mathbf{x}}$ resp. $\mathbf{w}_{\mathrm{n}}^{\mathbf{x}'}$, vignetting correction can be applied to candidate block $\mathbf{b}_{\mathrm{n}}^{\mathbf{x}'}$ as follows:

$$\mathbf{b}_{\mathrm{n,corr}}^{\mathbf{x}'} = \mathbf{b}_{\mathrm{n}}^{\mathbf{x}'} \frac{\sum_i w_{\mathrm{k}}^{\mathbf{x}}(i)}{\sum_i w_{\mathrm{n}}^{\mathbf{x}'}(i)} . \tag{5}$$

This function follows Equation (3), albeit that it is applied pixel-wise on blocks. The summations promote consistency of the results, especially in cases where the white image contains residual noise or demosaicing artefacts.

In an alternative formulation, it is possible to apply vignetting correction to a block without the use of white images. This would be useful for devices for which no library of white images is available. Here, the mean levels of the matched blocks themselves may be analyzed, leading to the following vignetting correction:

$$\mathbf{b}_{\mathrm{n,corr}}^{\mathbf{x}'} = \mathbf{b}_{\mathrm{n}}^{\mathbf{x}'} \frac{\sum_i b_{\mathrm{k}}^{\mathbf{x}}(i)\, \omega\left(b_{\mathrm{k}}^{\mathbf{x}}(i)\right)}{\sum_i b_{\mathrm{n}}^{\mathbf{x}'}(i)\, \omega\left(b_{\mathrm{k}}^{\mathbf{x}}(i)\right)} , \tag{6}$$

with ω similarly defined as in (4). It is applied here to avoid including over- and under-exposed pixels in the computation of the vigneting correction factor.

Candidate blocks for neighboring over-exposed pixels will partialy overlap, so that multiple candidates may be obtained for each over-exposed pixel. Blending or averaging candidate blocks would lead to loss of detail, allowing us to adopt a greedy approach instead whereby an over-exposed pixel is selected, its entire corresponding patch is pasted into the image, and all pixels that have been touched are marked using a mask.

We can now reconstruct block $\mathbf{b}_{\mathrm{k,corr}}^{\mathbf{x}}$, for instance by inserting the block of pixels $\mathbf{b}_{\mathrm{n,corr}}^{\mathbf{x}'}$ directly into the output image, but this may lead to edge artifacts. we therefore use alpha-matting to blend the well-exposed pixels in $\mathbf{b}_{\mathrm{k}}^{\mathbf{x}}$ with the well-exposed pixels in $\mathbf{b}_{\mathrm{n,corr}}^{\mathbf{x}'}$. The alpha value is derived from the binary mask M by first dilating the mask and then applying a Gaussian convolution, leading to an alpha matte. For our 13×13 blocks, we use a dilation of 3 pixels and a filter parameter of $\sigma = 1.5$. Representing the neighborhood corresponding to $\mathbf{b}_{\mathrm{k}}^{\mathbf{x}}$ on the alpha matte as $\hat{\mathbf{m}}_{\mathrm{k}}^{\mathbf{x}}$, the final pixel values are then calculated as follows:

$$\mathbf{b}_{\mathrm{k,final}}^{\mathbf{x}} = (\mathbf{1} - \hat{\mathbf{m}}_{\mathrm{k}}^{\mathbf{x}}) \odot \mathbf{b}_{\mathrm{k}}^{\mathbf{x}} + \hat{\mathbf{m}}_{\mathrm{k}}^{\mathbf{x}} \odot \mathbf{b}_{\mathrm{n,corr}}^{\mathbf{x}'} . \tag{7}$$

We found this algorithm to work well in practice, although the quality of the final result depends on the quality of the pre-processing algorithms (notably demosaicing) for the peripheral views. We therefore introduce a refinement of our algorithm in the next section.

6 GRAPH-BASED IMAGE RECONSTRUCTION

When the captured data is corrupted by noise, e.g. when the ISO value is set high under low light conditions, the above algorithm will copy or even magnify the noise. Further, light field decoding (notably view extraction) may introduce artefacts, e.g. stripping effects, which may find its way into the reconstructed image areas. To address these problems, we introduce a graph-based image reconstruction method, which essentially provides a signal-adaptive filtering.

In particular, a basic reconstructed vector $\mathbf{b}_{\mathrm{k,corr}}^{\mathbf{x}}$ may be calculated by requiring that (i) the reconstructed pixels in $\mathbf{b}_{\mathrm{k,corr}}^{\mathbf{x}}$ are multipliers r of corresponding pixels in the selected matched candidate block $\mathbf{b}_{\mathrm{n}}^{\mathbf{x}'}$, where $r = w_{\mathrm{k}}^{\mathbf{x}}/w_{\mathrm{n}}^{\mathbf{x}'}$ [see Equation (5)], and (ii) the values of well-exposed pixels in the target block remain unchanged, enforced by using $\mathbf{1} - \mathbf{m}_\mathbf{k}^\mathbf{x}$ similar to the first term in Equation (7). Here, $\mathbf{m}_\mathbf{k}^\mathbf{x}$ represents the neighborhood corresponding to $\mathbf{b}_\mathbf{k}^\mathbf{x}$ on the mask $\mathbf{M}$.

In addition, we wish to impose a graph-signal smoothness prior, as in other applications this has shown to help regularize the results, leading to better reconstruction and less noise than the reconstruction technique presented in the preceding section. In our case, we impose an undirected 8-connected graph on the candidate block $\mathbf{b}_\mathbf{n}^{\mathbf{x}'}$, represented by adjacency matrix $\mathbf{A}$ of size $t \times t$, with non negative elements $a_{i,j}$ computed as follows [23, 35]:

$$a_{i,j} = \exp\left(-\frac{\|b_\mathbf{n}^{\mathbf{x}'}(i) - b_\mathbf{n}^{\mathbf{x}'}(j)\|_2^2}{2\sigma_a^2}\right). \tag{8}$$

where the parameter σ_a controls the sensitivity of the similarity measure to the range of the luminance differences. This parameter is empirically set to a scalar multiplied by the maximum luminance difference between neighboring pixels in the block, i.e. $\sigma_a = s \max_{i,j} \|b_\mathbf{n}^{\mathbf{x}'}(i) - b_\mathbf{n}^{\mathbf{x}'}(j)\|_2$, where $s \in [0.1, 0.2]$. Note that $a_{i,j} \approx 1$ if the connecting pixels i and j have similar values.

The previously mentioned requirements can be combined to lead to the following objective function:

$$\begin{aligned}\arg\min_{\mathbf{b}_{\mathbf{k},\mathrm{corr}}^\mathbf{x}} & \|\mathbf{m}_\mathbf{k}^\mathbf{x} \odot (\mathbf{b}_{\mathbf{k},\mathrm{corr}}^\mathbf{x} - r\,\mathbf{b}_\mathbf{n}^{\mathbf{x}'})\|_2^2 \\ & + \|(\mathbf{1} - \mathbf{m}_\mathbf{k}^\mathbf{x}) \odot (\mathbf{b}_{\mathbf{k},\mathrm{corr}}^\mathbf{x} - \mathbf{b}_\mathbf{k}^\mathbf{x})\|_2^2 \\ & + \beta\, \mathbf{b}_{\mathbf{k},\mathrm{corr}}^{\mathbf{x}\,\mathrm{T}}\, \mathbf{L}\, \mathbf{b}_{\mathbf{k},\mathrm{corr}}^\mathbf{x},\end{aligned} \tag{9}$$

where the last term contributes to the enforcement of the graph-signal smoothness prior, and β is a parameter that strikes a balance between data fidelity and the graph-signal smoothness prior ($\beta = 5$ in our experiments). In particular, $\mathbf{L}$ represents the unnormalized combinatorial graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$ [7], with $\mathbf{D}$ being a diagonal matrix where each element $d_{i,i}$ is given by the row sum $\sum_{j=1}^m a_{i,j}$. The mentioned graph-signal smoothness prior ensures that $\mathbf{b}_{\mathbf{k},\mathrm{corr}}^\mathbf{x}$ is smooth with respect to the constructed graph Laplacian $\mathbf{L}$ of the well-exposed block, so as to recover the underlying structure even in the presence of noise in both $\mathbf{b}_\mathbf{k}^\mathbf{x}$ and $\mathbf{b}_\mathbf{n}^{\mathbf{x}'}$.

As Equation (9) is a quadratic problem, it admits the optimal and closed-form solution, leading to (see the supplemenary material for detail):

$$\begin{aligned}\mathbf{b}_{\mathbf{k},\mathrm{corr}}^\mathbf{x} = & (\mathrm{diag}\,(\mathbf{1}) + \beta\,\mathbf{L})^{-1} \\ & \left(r\mathbf{m}_\mathbf{k}^\mathbf{x} \odot \mathbf{b}_\mathbf{n}^{\mathbf{x}'} + (\mathbf{1} - \mathbf{m}_\mathbf{k}^\mathbf{x}) \odot \mathbf{b}_\mathbf{k}^\mathbf{x}\right).\end{aligned} \tag{10}$$

In the final step, the vector $\mathbf{b}_{\mathbf{k},\mathrm{corr}}^\mathbf{x}$ is merged into the target view by replacing $\mathbf{b}_{\mathbf{n},\mathrm{corr}}^{\mathbf{x}'}$ in Equation (7).

7 RESULTS

In the following, we first demonstrate the effect of some of the decisions that went into the design of our algorithm. Second, we compare our algorithm against other techniques that might conceivably be employed to reconstruct poorly exposed areas. Unless noted otherwise, all images used in our experiments were taken with a first generation Lytro camera. All measurements were carried out on linear data, but images included in this paper are tonemapped with Reinhard and Devlin's operator [48], unless otherwise indicated. Note that input and result images were composited first into one larger image before tonemapping, to preserve relative pixel values and therefore enable a meaningful comparison.

Our algorithm is implemented in Matlab, and for the images presented in this paper the processing time varies between around 20 to 200 seconds on a laptop with an Intel Core i5 processor running

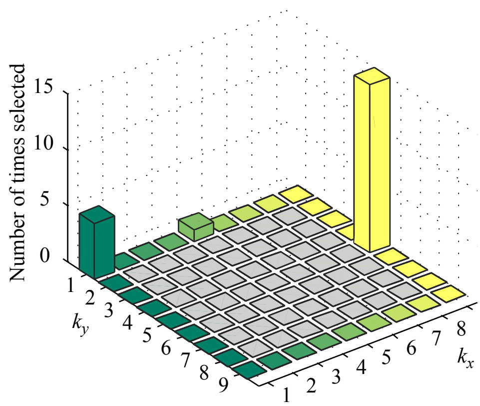

Fig. 5. Histogram of the number of matched blocks as function of location in the matrix of views. Only views located in the extreme periphery are included. Of 21 images tested, the view toward the right was picked 15 times by our block matching algorithm.

at 2.6 GHz, dependent on the size of the reconstructed areas. Compared to typical times for other preprocessing steps such as view reconstruction and disparity estimation, this constitutes only a small overhead. We suggest the reader to investigate the results on the soft copy of the paper.

7.1 Algorithmic Variants

Some interesting observations were made during the development of our algorithm, which we would like to report here. First, Figure 3 shows that we might expect the strongest dynamic range enhancement for views located near the edges, and especially the corners. Figure 2 (bottom right), on the other hand, shows that peripheral views may be reconstructed with low fidelity, which may be expressed as demosaicing artefacts and/or noise. Block-matching would be able to naturally find a good trade-off between dynamic range enhancement and artefact-suppression.

To demonstrate this, we have limited the block matching search space to the views that are in the extreme periphery. We counted across a set of images whose views contained the matches that were used in the reconstructions. Figure 5 shows a histogram of matches per view. It is clear that some views are strongly favored over others independent of the scene content. The highest peak in the histogram for our Lytro camera corresponds to a view right of the central view, along the edge. Corner views are not chosen due to artefacts and view reconstruction errors. As a consequence, we can limit the search space of block matching to a single view at the middle of the right edge ($\mathbf{k} = (9, 5)$), without losing significant loss of visual quality. Figure 6 shows additional analysis for views closer to the central view, confirming the results shown in Figure 5.

It is thought that both the hardware construction of the specific camera used (a first generation Lytro camera for the examples in this paper) as well as the pre-processing software may contribute to the behavior observed in Figures 5 and 6. The placement of the sheet of lenslets relative to the sensor, for example, may exhibit a slight offset and/or rotation, which may cause certain sensor locations to be imaged with higher accuracy. The pre-processing software, which notably includes demosaicing, tends to produce artefacts that are more significant toward the corners of the sensor array. This means that vignetting occurs most in those areas where artefacts are also likely to be most prevalent. As shown in Figures 5 and 6, our algorithm automatically accounts for this by finding the views with the best trade-off between reconstruction capability and presence of artefacts.

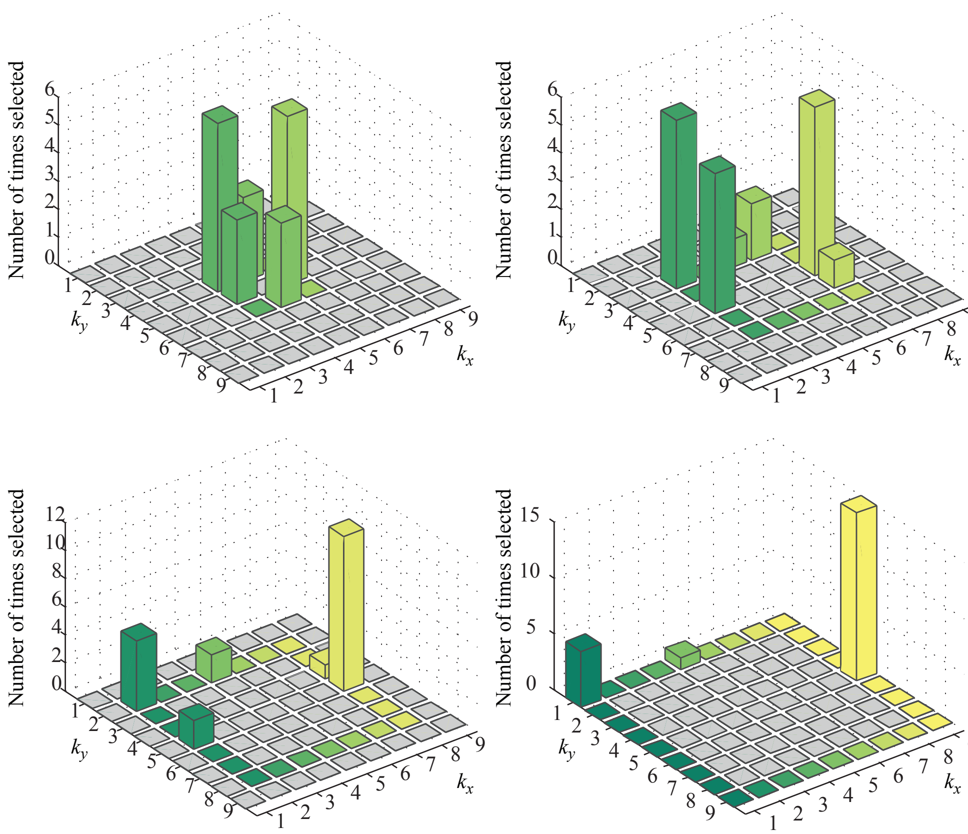

Fig. 6. Histograms of the number of matched blocks as function of views located at different distances to the central view.

Figure 6 allows one further observation, which is that there is benefit in selecting a target view for drawing patches from, instead of relying on 4D patches which would include the angular component. Such 4D patches would essentially cover multiple neighboring views in this figure. For such patches to be reliable, our current view selection method would have to show little variability in the number of times each view is selected. Given that Figure 6 shows that with increasing eccentricity neighboring views start to show significant differences in how often they are selected, we may infer that 4D patches selected toward the edges of the sensor will be less reliable. In some sense this may be expected as artefacts are more prevalent in more eccentric views. Additionally, searching in multiple neighboring views, perhaps in a 3×3 block of views, would incur a significantly higher computational cost.

Figure 7 shows reconstruction results using candidate views (6,5), (7,5), (8,5) and (9,5), demonstrating that the ability to reconstruct over-exposed pixels depends strongly on how far the candidate view is removed from the central view.

Second, we have presented two reconstruction algorithms, one straightforward block copying approach, described in Section 5 which gives us a baseline quality to compare our more advanced solution of Section 6 against. Example results are shown in Figure 8. Note that the graph-based method is able to create softer-looking results due to its ability to adapt to the signal.

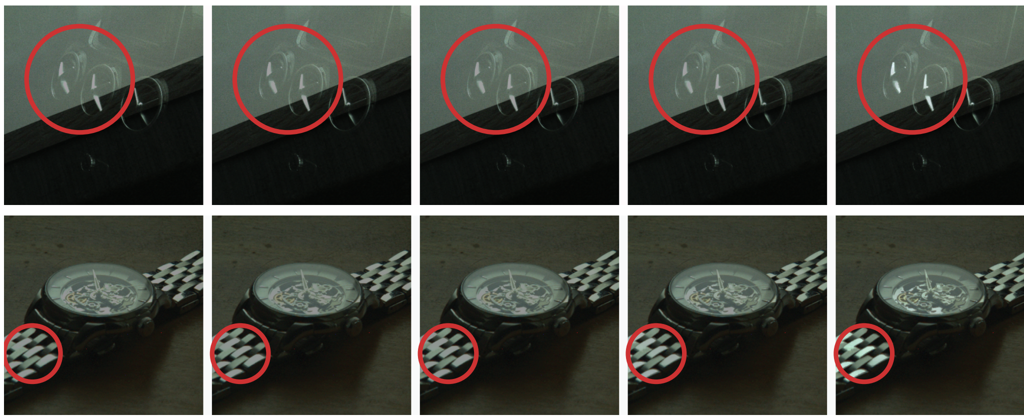

Fig. 7. Reconstruction of the central view from different peripheral views. From left to right: input central view ($\mathbf{k} = (5, 5)$), reconstructed views from views (6,5), (7,5), (8,5) and (9,5). Significant differences are encircled.A Circle is drawn on each image to mark one modified region.

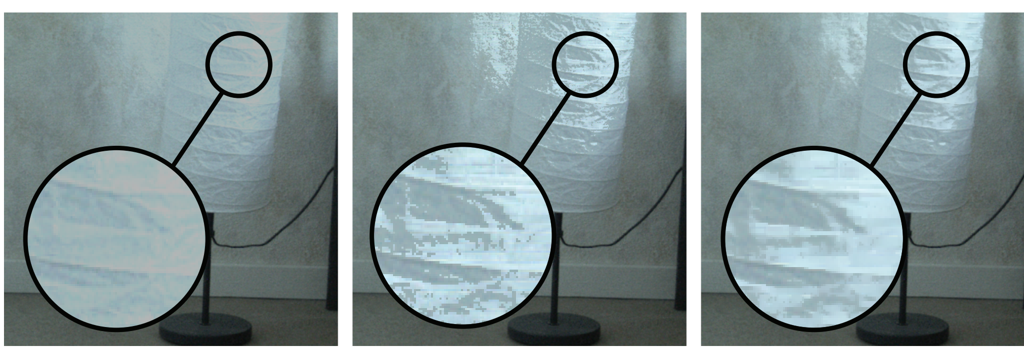

Fig. 8. The input image (left) was processed with our simple reconstruction method (middle) and our graph-based solution (right). Note that our graph-based solution adapts the copied pixels better, leading to higher visual quality than our simple method. The zoom shows an area where the improvement is especially significant.A Circle is drawn on each image to mark one modified region.

Third, vignetting correction may be applied on the basis of white images [Equation (5)], or the input plenoptic capture may be analysed directly [Equation 6]. The latter case turns out to be a less robust method, which is confirmed by the example shown in Figure 9. As such, we would recommend using white images, especially if they are supplied by the camera manufacturer, resorting only to analysis of vignetting factors using the input image if no white image is available.

Finally, we demonstrate the correction of dark areas in peripheral views using well-exposed pixels in the central view in Figure 10. This figure shows that noise may be effectively suppressed by our method.

7.2 Comparisons

To our knowledge, reconstruction of missing data in plenoptic captures was not attempted before. However, it would conceivably be possible to apply declipping methods to the individual views after view reconstruction. Declipping methods are in essence inpainting algorithms for over-exposed areas, and therefore synthesize data where no image information is available. We compare against

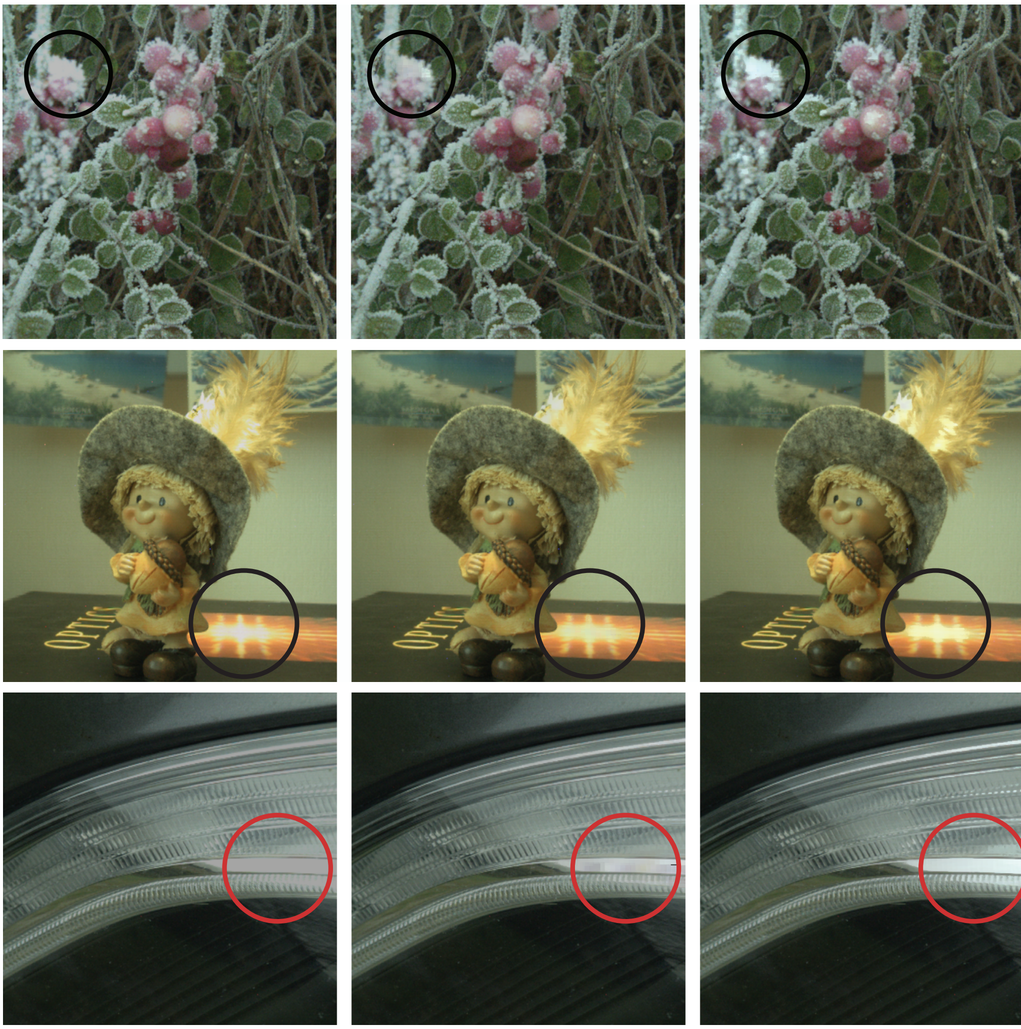

Fig. 9. The input images (left) were corrected using the images themselves to calculate the vignetting correction (middle). The images corrected using a corresponding white image are shown on the right. Note especially the marked regions.A Circle is drawn on each image to mark one modified region.

the methods by Masood et. al. [39] and Abebe et. al. [1] in Figure 11. Our algorithm has the benefit of being able to use redundancy in plenoptic captures, which standard declipping methods do not. For this reason, we expect our reconstruction results to be of higher fidelity than synthesis methods. This is confirmed by comparing the results against the ground truth images in Figure 11, where our method reconstructs details and increases the dynamic range, whereas the two methods we compare with do not reach the same level of reconstruction, both in terms of dynamic range enhancement and color fidelity. Note that the ground truth images are normalized to allow a meaningful comparison of the reconstruction quality of over-exposed details, even if this means that the absolute dynamic range cannot be compared directly in this image. MoreDynamic range comparisons are available in Figures 12- 13. For each of the images in Figure 12, the peak output value generated by our reconstruction algorithm was determined, and a horizontal pixel row centered around this point was

Fig. 10. The darks in peripheral views (left) are reconstructed (right). Note the strong reduction in noise. The images are gamma corrected for display using $\gamma = 2.2$. Significant differences can be seen in the marked regions.A Circle is drawn on each image to mark one modified region.

selected for analysis, as shown in Figure 13. This figure shows that data synthesis techniques such as Masood et. al. [39] and Abebe et. al. [1] do not create significant dynamic range, whereas our reconstruction techniques are able to create a plausible signal in over-exposed areas.

Additionally, we have assessed the relative merit of our algorithm and the two declipping methods by calculating the SSIM metric [65]. To this end we have taken 7 plenoptic captures with a first generation Lytro camera, and for each of these captures we have taken an additional capture with an exposure time 2 f-stops shorter which serves as ground truth because these images have no over-exposed pixels. While we could also have created ground truth images by recording an exposure sequence to construct an HDR light-field, this would involve image alignment, calculation of the camera response and possibly deghosting methods [49]. Each of these techniques introduces uncertainties that may affect the ability to draw conclusions. Instead, all that is required of a ground truth image is that the lightest pixels are not over-exposed. This can be achieved by simply stopping down a sufficient number of f-stops, and then scaling up the recorded image by the same amount.

The SSIM metric is calculated with respect to these ground truth images after multiplying them by a factor of 4. The average SSIM for the input image relative to the ground truth image is 0.70. Our graph-based solution averages to 0.81, while both Masood's and Abebe's algorithms produce an average value of 0.72. The gain of our algorithm is therefore just over 17%, whereas the other two algorithm improve the input by on average 2%.

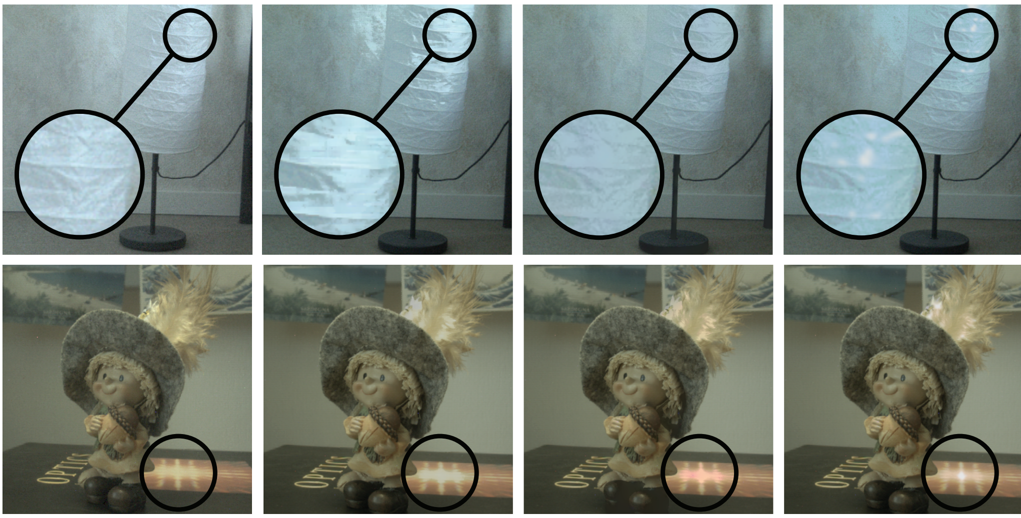

Fig. 11. Comparisons with state-of-the-art declipping methods. Shown here are from left to right: ground truth images, our results, the method of Masood et. al. [39], and the method of Abebe et. al. [1]. Salient differences are visible in the marked regions.A Circle is drawn on each image to mark one modified region.

Finally, we compare our results against ground truth data, which we obtain by taking one well-exposed plenoptic capture, and one plenoptic capture that was under-exposed by 2 f-stop. Its pixel values are therefore a factor of 4 lower. By scaling this ground truth image by a factor of 4, it is therefore possible to assess the quality of reconstruction, as shown in Figure 14. While the ground truth image is scaled, and therefore contains significantly more noise, the bright parts in our image are reconstructed with high fidelity.

Finally, note that plenoptic captures may simultaneously benefit from under- and over-exposure correction in the same capture. Compare, for instance the results presented in Figures 1 and 10.

7.3 Limitations

Our algorithm generally finds good matching blocks, and with the aid of our graph-signal processing approach, they help reconstruct pixels well. However, dependent on the dynamic range in the scene, it may happen that the over-exposed pixels in the central view are also over-exposed in peripheral views. In this case, our algorithm is not able to reconstruct over-exposed pixels, because the required information is not present in any of the views. This is shown in the handle of the milk can in Figure 15, although note that a large area of the milk can was successfully recovered. In such cases, standard declipping methods or inpainting methods may be employed after our reconstruction algorithm is applied.

Further, the method in its current form assumes that a Type 1 light field image is used with near-zero aliasing in the views. In the presence of aliasing in the views, however, this approach might somewhat limit the achievable resolution of refocused images in some refocusing planes. Although left for future work, such cases could be caught by finding the corresponding patches using pattern matching considering all the views to be reconstructed and by including the relative pixel disparities in one integrated cost function.

Fig. 12. Further results showing from left to right: the input image, our simple reconstruction, our graph-based reconstruction, as well as the methods by Masood et. al. [39] and Abebe et. al. [1]. Note, the horizontal lines indicate the pixel rows plotted in Figure 13.

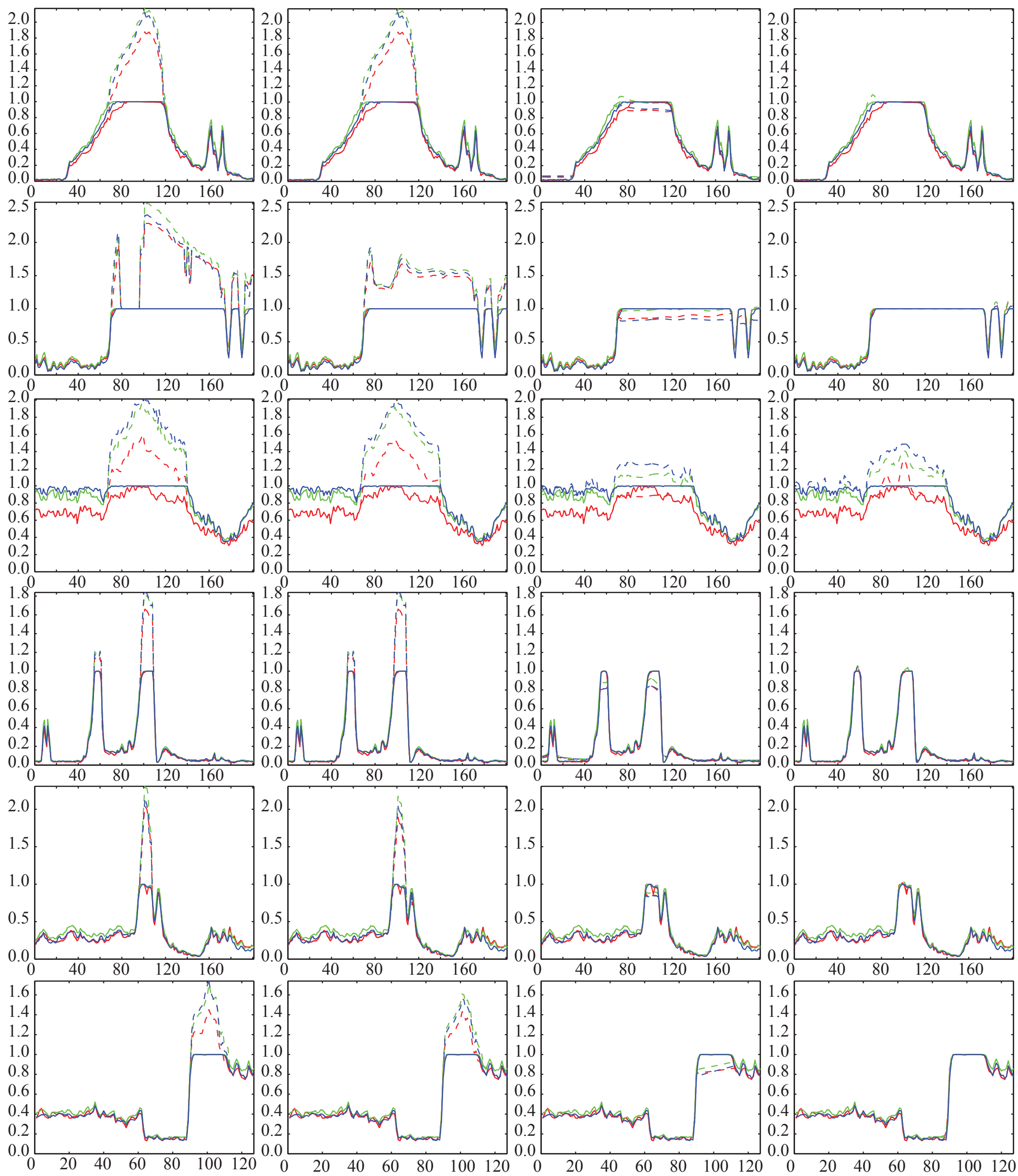

Fig. 13. Enhancement plots for each of the results shown in Figure 12. From left to right: our simple reconstruction, our graph-based reconstruction, as well as the methods by Masood et. al. [39] and Abebe et. al. [1]. On the horizontal axis is the pixel location along a scanline. The vertical axis shows normalized pixel value. The values for the input images are shown as solid lines (one line for each color channel) and never exceed a value of 1.0. The pixel values for the result images are plotted in dashed lines (also one line for each color channel).

Fig. 14. Comparison of our reconstructed center views (middle) against ground truth images (right). The input images are shown on the left. Salient differences are encircled.A Circle is drawn on each image to mark one modified region.

Fig. 15. Reconstruction of an input view (left) may fail if pixels in the peripheral views remain over-exposed, as shown in the handle of the milk can (right).

8 CONCLUSIONS

We exploit the redundancy of plenoptic captures as well as the occurrence of vignetting to reconstruct over- and under-exposed areas, without having to resort to heuristics. All reconstructed views may be enhanced, producing less noise in dark areas and increasing the peak luminance. In bright areas, an increase in dynamic range by a factor of around 4 is demonstrated, dependent on the camera settings. This approaches the maximum achievable for our camera, which was measured to be a factor of around 6. This translates to a number of reconstructed pixels that is highly dependent on the scene and its illumination, as evidenced in the Figures, albeit that on average a significant number of pixels can be recovered.

Our method applies to all lenslet-based plenoptic cameras (including Lytro and Raytrix), and comes only at the cost of a lightly increased pre-processing time.

ACKNOWLEDGMENTS

We would like to warmly thank Neus Sabater, Matthieu Fradet and Arno Schubert for their help, discussions and support during the project. This work was supported by National Natural Science Foundation of China under contract No.U1636206, Alibaba Innovative Research under contract No.XTG201800108 and MSRA Collaborative Research under project ID FY18-Research-Sponsorship-029.

REFERENCES

[1] Mekides Assefa Abebe, Tania Pouli, Jonathan Kervec, and Chaker Larabi. 2015. Color Clipping and Over-exposure Correction. In *Eurographics Symposium on Rendering - Experimental Ideas & Implementations*, Jaakko Lehtinen and Derek Nowrouzezahrai (Eds.). The Eurographics Association.

[2] Edward H Adelson and John Y. A. Wang. 1992. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14, 2 (1992), 99–106.

[3] Kurt Barton Akeley, Brian Cabral, Colvin Pitts, Chia-Kai Liang, Bennett Wilburn, Timothy James Knight, and Yi-Ren Ng. 2014. Compensation for sensor saturation and microlens modulation during light-field image processing. (2014). Patent No. US 9386288 B2, Filed December 15, 2014, Issued July 5, 2016.

[4] Tom E Bishop and Paolo Favaro. 2012. The light field camera: Extended depth of field, aliasing, and superresolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 5 (2012), 972–986.

[5] Tom E Bishop, Sara Zanetti, and Paolo Favaro. 2009. Light Field Superresolution. In *Proceedings of the IEEE International Conference on Computational Photography*. 1–9.

[6] D. Cho, M. Lee, S. Kim, and Y.-W. Tai. 2013. Modeling the calibration pipeline of the Lytro camera for high quality light-field image reconstruction. In *IEEE International Conference on Computer Vision*. 3280–3287.

[7] F R Chung. 1997. *Spectral Graph Theory*. American Mathematical Society.

[8] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. 2004. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing* 13, 9 (2004), 1200–1212.

[9] Donald Dansereau, Oscar Pizarro, and Stefan Williams. 2013. Decoding, calibration and rectification for lenselet-based plenoptic cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1027–1034.

[10] Paul Debevec and Jitendra Malik. 1997. Recovering high dynamic range radiance maps from photographs. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. 369–378.

[11] Alexei A Efros and Thomas K Leung. 1999. Texture synthesis by non-parametric sampling. In *The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999*, Vol. 2. 1033–1038.

[12] Juliet Fiss, Brian Curless, and Richard Szeliski. 2014. Refocussing Plenoptic Images using Depth-Adaptive Splatting. In *Proceedings of the IEEE International Conference on Computational Photography*. 1–9.

[13] Orazio Gallo, N Gelfandz, Wei-Chao Chen, Marius Tico, and Kari Pulli. 2009. Artifact-free high dynamic range imaging. In *Computational Photography (ICCP), 2009 IEEE International Conference on*. IEEE, 1–7.

[14] Elena Garces, Jose I Echevarria, Wen Zhang, Hongzhi Wu, Kun Zhou, and Diego Gutierrez. 2016. Intrinsic Light Fields. *https://arxiv.org/abs/1608.04342* (2016).

[15] Todor Georgiev and Andrew Lumsdaine. 2010. Rich image capture with plenoptic cameras. In *Proceedings of the IEEE International Conference on Computational Photography*. 1–8.

[16] Todor Georgiev, Andrew Lumsdaine, and Sergio Goma. 2009. High Dynamic Range Image Capture with Plenoptic 2.0 Camera. In *Proceedings of Signal Recovery and Synthesis*. Optical Society of America, SWA7P.

[17] Todor Georgiev, Ke Colin Zheng, Brian Curless, David Salesin, Shree Nayar, and Chintan Intwala. 2006. Spatio-Angular Resolution Tradeoff in Integral Photography. In *Eurographics Symposium on Rendering*. 263–272.

[18] Steven J Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F Cohen. 1996. The lumigraph. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. ACM, 43–54.

[19] Miguel Granados, Hans-Peter Seidel, and Hendrik P. A. Lensch. 2008. Background estimation from non-time sequence images. In *Proceedings of graphics interface 2008*. 33–40.

[20] Christopher Hahne, Amar Aggoun, Vladan Velisavljevic, Susanne Fiebig, and Matthias Pesch. 2016. Refocusing Distance of a Standard Plenoptic Camera. *Optics Express* 24, 19 (2016), 21521–21540.

[21] Roarke Horstmeyer, Gary Euliss, Ravindra Athale, and Marc Levoy. 2009. Flexible Multimodal Camera Using a Light Field Architecture. In *Proceedings of the IEEE Conference on Computational Photography*. 1–8.

[22] W. Hu, G. Cheung, and M. Kazui. 2016. Graph-based Dequantization of Block-Compressed Piecewise Smooth Images. *IEEE Signal Processing Letters* 23, 2 (2016), 242–246.

[23] W. Hu, X. Li, G. Cheung, and O. Au. 2013. Depth Map Denoising using Graph-based Transform and Group Sparsity. In *IEEE International Workshop on Multimedia Signal Processing*. Pula, Italy.

[24] Wei Hu, Mozhdeh Seifi, and Erik Reinhard. 2017. Optical Center Estimation for Lenslet-based Plenoptic Cameras. In *19th IEEE International Workshop on Multimedia Signal Processing (MMSP)*.

[25] Ivo Ihrke, Timo Stich, Heiko Gottschlich, Marcus Magnor, and Hans-Peter Seidel. 2008. Fast Incident Light Field Acquisition and Rendering. *Journal of WSCG* 16, 1 (2008), 25–32.

[26] Aaron Isaksen, Leonard McMillan, and Steven J Gortler. 2000. Dynamically Reparameterized Light Fields. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. 297–306.

[27] Frederic E Ives. 1903. Parallax Stereogram and Process of Making Same. (1903). U.S. Patent US725567 A.

[28] EA Khan, AO Akyuz, and E Reinhard. 2006. Ghost removal in high dynamic range images. In *IEEE International Conference on Image Processing*. Washington, DC, USA, 2005–2008.

[29] Seon Joo Kim and Marc Pollefeys. 2008. Robust radiometric calibration and vignetting correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 4 (2008), 562–576.

[30] Marc Levoy and Pat Hanrahan. 1996. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. ACM, 31–42.

[31] Marc Levoy, Ren Ng, Andrew Adams, Matthew Footer, and Mark Horowitz. 2006. Light field microscopy. *ACM Transactions on Graphics (TOG)* 25, 3 (2006), 924–934.

[32] Chia-Kai Liang and Ravi Ramamoorthi. 2015. A Light Transport Framework for Lenslet Light Field Cameras. *ACM Trans. Graph.* 34, 2 (March 2015), 16:1–16:19.

[33] Andrew Lumsdaine and Todor Georgiev. 2008. Full resolution lightfield rendering. *Indiana University and Adobe Systems, Tech. Rep* (2008).

[34] Andrew Lumsdaine and Todor Georgiev. 2009. The focused plenoptic camera. In *IEEE International Conference on Computational Photography (ICCP)*. 1–8.

[35] U. Luxburg. 2007. A Tutorial on Spectral Clustering. In *Statistics and Computing*, Vol. 17, no.4. 395–416.

[36] Henrique S Malvar, Li-Wei He, and Ross Cutler. 2004. High-Quality Linear Interpolation for Demosaicking of Bayer-Patterned Color Images. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 3. iii–485.

[37] Alkhazur Manakov, John Restrepo, Oliver Klehm, Ramon Hegedus, Elmar Eisemann, Hans-Peter Seidel, and Ivo Ihrke. 2013. A reconfigurable camera add-on for high dynamic range, multispectral, polarization, and light-field imaging. *ACM Transactions on Graphics* 32, 4 (2013), 47–1.

[38] S Mann and RW Picard. 1995. On being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures. In *Proceedings of IS&T 48th annual conference*. 422–428.

[39] Syed Z Masood, Jiejie Zhu, and Marshall F Tappen. 2009. Automatic Correction of Saturated Regions in Photographs using Cross-Channel Correlation. *Computer Graphics Forum* 28, 7 (2009), 1861–1869.

[40] Vincent Masselus, Pieter Peers, Philip Dutré, and Yves D Willems. 2003. Relighting with 4D Incident Light Fields. *ACM Transactions on Graphics* 22, 3 (2003), 613–620.

[41] Peyman Milanfar. 2013. A Tour of Modern Image Filtering: New Insights and Methods, both Practical and Theoretical. *IEEE Signal Processing Magazine* 30, 1 (2013), 106–128.

[42] T Mitsunaga and S K Nayar. 1999. Radiometric self calibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1. 374–380.

[43] Shree K Nayar and Srinivasa G Narasimhan. 2002. Assorted Pixels: Multisampled Imaging with Structural Models. In *European Conference on Computer Vision*. 636–652.

[44] Ren Ng, Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz, and Pat Hanrahan. 2005. Light field photography with a hand-held plenoptic camera. *Computer Science Technical Report CSTR* 2, 11 (2005), 1–11.

[45] Christian Perwass and Lennart Wietzke. 2012. Single lens 3D-camera with extended depth-of-field. In *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 829108–829108.

[46] Colvin Pitts, Timothy James Knight, Chia-Kai Liang, and Yi-Ren Ng. 2015. Generating dolly zoom effect using light field image data. (2015). Patent No. US 8971625 B2, Filed June 23, 2014, Issued March 3, 2015.

[47] Ramesh Raskar, Amit Agrawal, Cyrus A. Wilson, and Ashok Veeraraghavan. 2008. Glare Aware Photography: 4D Ray Sampling for Reducing Glare Effects of Camera Lenses. *ACM Transactions on Graphics* 27, 3 (2008), 56:1–56:10.

[48] E Reinhard and K Devlin. 2005. Dynamic range reduction inspired by photoreceptor physiology. *IEEE Transactions on Visualization and Computer Graphics* 11, 1 (2005), 13–24.

[49] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski. 2010. *High Dynamic Range Imaging: Acquisition, Display and Image-Based Lighting* (2nd ed.). Morgan Kaufmann.

[50] Erik Reinhard, Erum Arif Khan, Ahmet Oğuz Akyüz, and Garrett M Johnson. 2008. *Color Imaging: Fundamentals and Applications*. A K Peters, Wellesley.

[51] Erik Reinhard, Michael Stark, Peter Shirley, and James Ferwerda. 2002. Photographic tone reproduction for digital images. *ACM Transactions on Graphics* 21, 3 (2002), 267–276.

[52] MA Robertson, S Borman, and RL Stevenson. 2003. Estimation-Theoretic Approach to Dynamic Range Enhancement using Multiple Exposures. *Journal of Electronic Imaging* 12 (2003), 219–228. Issue 2.

[53] Neus Sabater, Mozhdeh Seifi, Valter Drazic, Gustavo Sandri, and Patrick Pérez. 2014. Accurate disparity estimation for plenoptic images. In *Computer Vision — ECCV 2014 Workshops*. Springer, 548–560.

[54] Yoav Y Schechner and Shree K Nayar. 2003. Generalized Mosaicing: High Dynamic Range in a Wide Field of View. *International Journal of Computer Vision* 53, 3 (2003), 245–267.

[55] David C Schedl, Clemens Birklbauer, and Oliver Bimber. 2014. Coded Exposure HDR Light-Field Video Recording. *Computer Graphics Forum* 33, 2 (2014), 33–42.

[56] Pradeep Sen, Nima Khademi Kalantari, Maziar Yaesoubi, Soheil Darabi, Dan B. Goldman, and Eli Shechtman. 2012. Robust patch-based HDR reconstruction of dynamic scenes. *ACM Transactions on Graphics* 31, 6 (2012), 203:1–203:11.

[57] D I Shuman, S K Narang, P Frossard, A Ortega, and P Vandergheynst. 2013. The Emerging Field of Signal Processing on Graphs. *IEEE Signal Processing Magazine* 30, 3 (2013), 83–98.

[58] Thiow Keng Tan, Choong Seng Boon, and Yuya Suzuki. 2006. Intra prediction by template matching. In *IEEE International Conference on Image Processing*. 1693–1696.

[59] Michael Tao, Sunil Hadap, Jitendra Malik, and Ravi Ramamoorthi. 2013. Depth from combining defocus and correspondence using light-field cameras. In *Proceedings of the IEEE International Conference on Computer Vision*. 673–680.

[60] J Unger, S Gustavson, P Larsson, and A Ynnerman. 2008. Free Form Incident Light Fields. *Computer Graphics Forum* 27, 4 (2008), 1293–1301.

[61] J Unger, A Wenger, T Hawtins, A Gardner, and P Debevec. 2003. Capturing and Rendering with Incident Light Fields. In *Eurographics Symposium on Rendering*. 141–149.

[62] Ashok Veeraraghavan, Ramesh Raskar, Amit Agrawal, Ankit Mohan, and Jack Tumblin. 2007. Dappled Photography: Mask Enhanced Cameras for Hyterodyned Light Fields and Coded Aperture Refocusing. *ACM Transactions on Graphics* 26, 3 (2007), 69.

[63] Lvdi Wang, Li-Yi Wei, Kun Zhou, Baining Guo, and Heung-Yeung Shum. 2007. High dynamic range image hallucination. In *Proceedings of the 18th Eurographics conference on Rendering Techniques*. Eurographics Association, 321–326.

[64] Ting-Chun Wang, Jun-Yan Zhu, Ebi Hiroaki, Manmohan Chandraker, Alexei A Efros, and Ravi Ramamoorthi. 2016. A 4D Light-Field Dataset and CNN Architecture for Material Recognition. In *European Conference on Computer Vision*. 121–138.

[65] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.

[66] Sven Wanner and Bastian Goldluecke. 2014. Variational light field analysis for disparity estimation and super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 3 (2014), 606–619.

[67] Greg Ward. 2003. Fast, robust image registration for compositing high dynamic range photographs from hand-held exposures. *Journal of Graphics Tools* 8, 2 (2003), 17–30.

[68] Li-Yi Wei and Marc Levoy. 2001. Texture synthesis over arbitrary manifold surfaces. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. 355–360.

[69] Z Yu, X Guo, H Ling, A Lumsdaine, and J Yu. 2013. Line Assisted Light Field Triangulation and Stereo Matching. In *Proceedings of the IEEE International Conference on Computer Vision*. 2792–2799.

[70] Zhan Yu, Jingji Yu, Andrew Lumsdaine, and Todor Georgiev. 2012. An Analysis of Color Demosaicing in Plenoptic Cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 901–908.

[71] Yuanjie Zheng, Stephen Lin, Chandra Kambhamettu, Jingyi Yu, and Sing Bing Kang. 2009. Single-image vignetting correction. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 12 (2009), 2243–2256.

[72] Shan Zhu and Kai-Kuang Ma. 2000. A new diamond search algorithm for fast block-matching motion estimation. *IEEE Transactions on Image Processing* 9, 2 (2000), 287–290.

Received August 2017; revised ; accepted