PROCEEDINGS

IS&T / SPIE
Electronic
Imaging
SCIENCE AND TECHNOLOGY

Digital Photography X

Nitin Sampat
Radka Tezaur
Sebastiano Battiato
Boyd A. Fowler
Todor G. Georgiev
Francisco H. Imai
Andrew Lumsdaine
Kevin J. Matherson
Dietmar Wüller
Editors

3–5 February 2014
San Francisco, California, United States

Sponsored by
IS&T—The Society for Imaging Science and Technology
SPIE

Cosponsored by
Google (United States)
Fairchild Imaging (United States)
Canon (United States)

Published by
SPIE

Volume 9023

Proceedings of SPIE 0277-786X, v. 9023

Digital Photography X, edited by Nitin Sampat, Radka Tezaur, et. al. Proc. of SPIE-IS&T
Electronic Imaging, SPIE Vol. 9023, 902301 · © 2014 SPIE-IS&T
CCC code: 0277-786X/14/$18 · doi: 10.1117/12.2063416

The papers included in this volume were part of the technical conference cited on the cover and title page. Papers were selected and subject to review by the editors and conference program committee. Some conference presentations may not be available for publication. The papers published in these proceedings reflect the work and thoughts of the authors and are published herein as submitted. The publishers are not responsible for the validity of the information or for any outcomes resulting from reliance thereon.

Please use the following format to cite material from this book:
Author(s), "Title of Paper," in *Digital Photography X*, edited by Nitin Sampat, Radka Tezaur, et al., Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 9023, Article CID Number (2014)

ISSN: 0277-786X
ISBN: 9780819499400

Copublished by
SPIE
P.O. Box 10, Bellingham, Washington 98227-0010 USA
Telephone +1 360 676 3290 (Pacific Time) · Fax +1 360 647 1445
SPIE.org
and
IS&T—The Society for Imaging Science and Technology
7003 Kilworth Lane, Springfield, Virginia, 22151 USA
Telephone +1 703 642 9090 (Eastern Time) · Fax +1 703 642 9094
imaging.org

Printed in the United States of America.

Paper Numbering: Proceedings of SPIE follow an e-First publication model, with papers published first online and then in print and on CD-ROM. Papers are published as they are submitted and meet publication criteria. A unique, consistent, permanent citation identifier (CID) number is assigned to each article at the time of the first publication. Utilization of CIDs allows articles to be fully citable as soon as they are published online, and connects the same identifier to all online, print, and electronic versions of the publication. SPIE uses a six-digit CID article numbering system in which:

- The first four digits correspond to the SPIE volume number.
- The last two digits indicate publication order within the volume using a Base 36 numbering system employing both numerals and letters. These two-number sets start with 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B … 0Z, followed by 10-1Z, 20-2Z, etc.

The CID Number appears on each page of the manuscript. The complete citation is used on the first page, and an abbreviated version on subsequent pages. Numbers in the index correspond to the last two digits of the six-digit CID Number.

Contents

Conference Committee

Symposium Chair

Sergio R. Goma, Qualcomm Inc. (United States)

Symposium Co-chair

Sheila S. Hemami, Northeastern University (United States)

Conference Chairs

Nitin Sampat, Rochester Institute of Technology (United States)
Radka Tezaur, Nikon Research Corporation of America (United States)

Conference Co-chairs

Sebastiano Battiato, Università degli Studi di Catania (Italy)
Boyd A. Fowler, Google (United States)
Todor G. Georgiev, Qualcomm Inc. (United States)
Francisco H. Imai, Canon U.S.A., Inc. (United States)
Andrew Lumsdaine, Indiana University (United States)
Kevin J. Matherson, Microsoft Corporation (United States)
Dietmar Wüller, Image Engineering GmbH & Co. KG (Germany)

Conference Program Committee

Erhardt Barth, Universität zu Lübeck (Germany)
Donald J. Baxter, STMicroelectronics Ltd. (United Kingdom)
Kathrin Berkner, Ricoh Innovations, Inc. (United States)
Ajit S. Bopardikar, Samsung Electronics, India Software Operations Ltd. (India)
Frédéric Cao, DxO Laboratories (France)
Peter B. Catrysse, Stanford University (United States)
Jeff Chien, Adobe Systems Inc. (United States)
Lauren A. Christopher, Indiana Univ.-Purdue University Indianapolis (United States)
Jeffrey M. DiCarlo, Intuitive Surgical, Inc. (United States)
Henry G. Dietz, University of Kentucky (United States)
Alexandru F. Drimbarean, Tessera (FotoNation) Ireland Ltd. (Ireland)
Joyce E. Farrell, Stanford University (United States)
Paolo Favaro, Universität der Künste Berlin (Germany)
Robert D. Fiete, ITT Exelis (United States)
Sergio R. Goma, Qualcomm Inc. (United States)

Mirko Guarnera, STMicroelectronics (Italy)
Bahadir K. Gunturk, Louisiana State University (United States)
Li Hong, Nikon Research Corporation of America (United States)
Paul M. Hubel, Apple Inc. (United States)
Xiaoyun Jiang, Qualcomm Inc. (United States)
George John, Microsoft Corporation (United States)
Michael A. Kriss, MAK Consultants (United States)
Jiangtao Kuang, OmniVision Technologies, Inc. (United States)
Feng Li, Apple Inc. (United States)
Jingqiang Dylan Li, Lifesize Communications, Inc. (United States)
Manuel Martinez, Universidad de València (Spain)
Lingfei Meng, Ricoh Innovations, Inc. (United States)
Jon S. McElvain, Dolby Laboratories, Inc. (United States)
Bo Mu, BAE Systems (United States)
Seishi Ohmori, Nikon Corporation (Japan)
Joni Oja, Nokia Research Center (Finland)
Shmuel Peleg, The Hebrew University of Jerusalem (Israel)
Kari A. Pulli, NVIDIA Corporation (United States)
John R. Reinert-Nash, Lifetouch, Inc. (United States)
Brian G. Rodricks, Image Engineering GmBH & Co. KG (United States)
Jackson Roland, Imatest LLC (United States)
Mårten Sjöström, Mid Sweden Universitet (Sweden)
Filippo D. Stanco, Università degli Studi di Catania (Italy)
Qun Sun, GalaxyCore, Inc. (United States)
Sabine Süsstrunk, Ecole Polytechnique Fédérale de Lausanne (Switzerland)
Touraj Tajbakhsh, Apple Inc. (United States)
Zhan Yu, University of Delaware (United States)
Jingyi Yu, University of Delaware (United States)
Ashok Veeraraghavan, Rice University (United States)
Thomas Vogelsang, Rambus Inc. (United States)
Michael Wang, Intel Corporation (United States)
Weihua Xiong, OmniVision Technologies, Inc. (United States)
Alireza Yasan, Foveon Inc. (United States)
Lei Zhang, The Hong Kong Polytechnic University (Hong Kong, China)

Session Chairs

1 Computational Photography
Andrew Lumsdaine, Indiana University (United States)

2 Mobile Photography
Sebastiano Battiato, Università degli Studi di Catania (Italy)

3 Image Quality Evaluation Methods/Standards for Mobile and Digital Photography: Joint Session with Conferences 9016 and 9023
Dietmar Wüller, Image Engineering GmbH & Co. KG (Germany)
Sophie Triantaphillidou, University of Westminster (United Kingdom)
Robin B. Jenkin, Aptina Imaging Corporation (United States)

4 Blur
Radka Tezaur, Nikon Research Corporation of America (United States)

5 Image Processing Pipeline and Camera Characterization
Nitin Sampat, Rochester Institute of Technology (United States)

6 Computer Vision and Applications
Todor G. Georgiev, Qualcomm Inc. (United States)

7 Color
Francisco H. Imai, Canon U.S.A., Inc. (United States)

8 HDR
Kevin J. Matherson, Microsoft Corporation (United States)

Special Presentations from the *Journal of Electronic Imaging*

In addition to the usual conference presentations, the 2014 Digital Photography X conference included a "Focal Track" of peer-reviewed papers that have been published in a special section of the *Journal of Electronic Imaging*. The JEI articles can also be found on the SPIE Digital Library at the following locations:

1. Vogelsang, T., Stork, D. G., Guidash, "A hardware validated unified model of multi-bit temporally and spatially oversampled image sensors with conditional reset," J. Electron. Imag. **23**(1), 013021 (2014). http://dx.doi.org/10.1117/1.JEI.23.1.01302

2. Wang, Q., Zhan Y., Rasmussen, C., Jingyi, Y., "Stereo vision-based depth of field rendering on a mobile device," J. Electron. Imag. **23**(2), 023009 (2014). http://dx.doi.org/10.1117/1.JEI.23.2.023009

Hardware validated unified model of multibit temporally and spatially oversampled image sensors with conditional reset

Thomas Vogelsang,* David G. Stork, and Michael Guidash
Rambus Inc., 1050 Enterprise Way, Suite 700, Sunnyvale, California 94089

Abstract. We describe a photon statistics-based theoretical model of the response to incident light of an image sensor and show that conditional reset and multibit temporal oversampling increase the dynamic range significantly. This photon-based modeling approach describes the full image sensor design space of temporal and spatial oversampling either with a binary comparison or with a multibit read of each sample. We find excellent quantitative agreement between measurements on custom hardware and our theoretical predictions. We then use this model to show what improvements in dynamic range and low-light response can be achieved by oversampling and what the limits of improvement caused by pixel size and lens parameters are. [DOI: 10.1117/1.JEI.23.1.013021]

Keywords: image sensors; image reconstruction; photons; noise.

Paper 13310SS received May 31, 2013; revised manuscript received Dec. 4, 2013; accepted for publication Jan. 14, 2014; published online Feb. 13, 2014.

1 Introduction

Over the past decade, CMOS image sensors have replaced both photographic film and CCD image sensors in nearly all imaging applications. During the same time, the advancement of silicon process technology according to Moore's law has led to smaller and smaller pixels in many of these applications. In a conventional image sensor, a pixel is sampled only once per exposure. The high end of the dynamic range is therefore limited by the full well capacity of the pixel. The low end of the dynamic range is determined by the minimum amount of light required to generate a signal that can be distinguished from the combined photon noise and sensor read noise. Modern image sensors have achieved sensitivities of a few photo electrons.[1]

The method proposed by us and other work discussed below apply oversampling to extend the dynamic range. In contrast to a conventional image sensor, an oversampling sensor combines multiple measurements of light intensity into a single pixel value of the final image. These multiple measurements can be distributed over space or over time or both.

To overcome the limitation of the high end of the dynamic range given by the reduced full well capacity and to make better use of highly sensitive pixels smaller than diffraction limit and with high sensitivity, Fossum[2,3] and Sbaiz et al.[4] proposed oversampling the incident light in space and time, making a binary decision at each sampling event by comparing the number of detected photons against a threshold. The total number of photons can then be reconstructed from the results (0 or 1) of multiple such binary samplings. Yang et al.[5] derived the theoretical limits of such binary oversampling based on photon statistics. All these proposals assume samplings equidistant in time and pixel reset after each sampling event and require a very small pixel with close to single-photon sensitivity. In the remainder of this paper, we will follow Ref. 3 by naming the binary sampled element "jot," reserving "image pixel" or simply "pixel" for the aggregate that is used to form the final image. Vogelsang and Stork[6] expanded this binary oversampling approach to be usable with less sensitive and conventional pixels and to provide more control over the sensor characteristics by the introduction of conditional reset and the variation of sampling thresholds and sampling interval durations. This new method fully resets the pixel instead of proportional to the sampled signal, and the threshold comparison and conditional reset is done at fixed times independent of the time when the threshold is reached. The sensor response is therefore different from $\Sigma\Delta$ approaches (Refs. 7 and 8) to binary oversampling.

Multibit sampling differs from these binary oversampling approaches and has been shown to extend dynamic range as well. Many of today's cameras have a high dynamic range (HDR) mode where multiple exposures with different exposure times are taken and afterward combined into a final image with extended dynamic range according to the proposal by Debevec and Malik.[9] Extension of dynamic range in a single exposure can be achieved either by circuit techniques in the pixel that modify the effective full well capacity or by multiple samplings during light accumulation. Yang and El Gamal compared some of these approaches in Ref. 10. Multibit sampling at exponentially spaced time intervals employing a pixel-level analog-to-digital converter (ADC) has been further explored by Yang et al.[11] The sequence of effective sampling durations is monotonically increasing since there is no reset during an exposure. This limits the possibilities to shape the sensor response.

The conditional and selective per-pixel full reset of our proposed method allows sampling of each pixel with the optimum sample interval duration for the given illumination

*Address all correspondence to: Thomas Vogelsang, E-mail: tvogelsang@rambus.com

level without the need to add per-pixel decision circuitry or a pixel-level ADC. The only addition to a pixel is one transistor to enable column control of the reset in addition to the usual row control. The method is, therefore, well suited for sensors with small pixels.

Vogelsang et al.[12] have shown a fundamental equivalence between multibit oversampling of pixels and binary oversampling using virtual jots that have thresholds at the steps of the ADC. As such, the same mathematical description applies to these two apparently different approaches. The mathematical representation of these approaches can be used to optimize the design of oversampled image sensors both for the expected light conditions and for the hardware properties of the pixel technology that is available to manufacture the sensor.

The work presented here is organized as follows. Section 2 describes our photon-based sensor model. The analytical model combining the theory of binary sampling first presented in Ref. 6 and multibit sampling first presented in Ref. 12 is described in Sec. 2.1, and the Monte Carlo approach that is used when noise sources other than photon shot noise need to be considered is described in Sec. 2.2. Different sampling policies (sampling schedules and threshold settings) are compared in Sec. 2.3, and an experimental hardware validation of the model is shown in Sec. 2.4. We then discuss in Sec. 3 how the photon-based sensor model can be related to imaging situations in the real world by connecting scene illumination and camera parameters to the image sensor parameters (Sec. 3.1) and use this relationship to compare low light and dynamic range capabilities of conventional digital cameras and cameras using the proposed oversampling sensor (Sec. 3.2). Section 4 summarizes our work.

2 Photon Statistics-Based Sensor Model

2.1 *Analytical Model*

The light intensity incident on a pixel of an image sensor is, in general, represented as a digital number of a certain bit depth. In a conventional image sensor, this bit depth is the bit depth of the ADC used to sample the photodetector response. The binary oversampling sensors discussed above achieve their total bit depth through the number of spatial and temporal binary samplings that are combined to form the signal in an image pixel. Multiexposure HDR derives most of its resolution through ADC bit depth but has some temporal oversampling (unconditional or hard reset between samples). The total image sensor design space can, therefore, be viewed as a three-dimensional space with the bit depth of the ADC and the amount of temporal and spatial oversampling as the axes. Figure 1 illustrates this concept and shows planes of constant total bit depth as well as the design space used by different sensors. Our theory accurately models all combinations of temporal and spatial oversampling as well as ADC bit depth.

2.1.1 *Binary sampling*

Sensor operation. Each image pixel comprises a number of binary sampled jots. At each sampling event, each jot produces a single binary output (a 1) if its integrated exposure exceeds a threshold θ and a 0 otherwise. If the jot produces a 1, then its integrated exposure is reset to 0; otherwise its integrated photon signal is not reset (a nondestructive read), as shown in Fig. 2. The threshold in each binary jot can be varied in space or time to improve image quality.

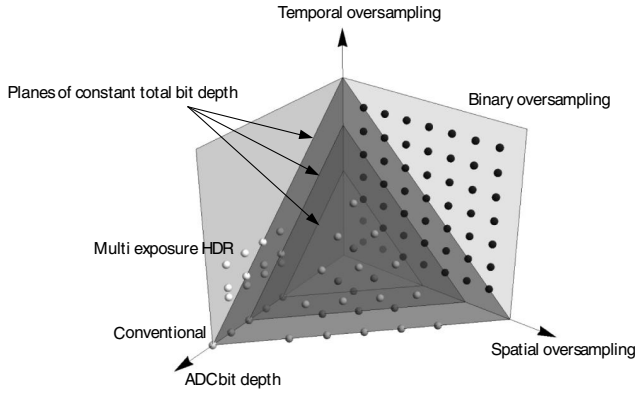

Fig. 1 Image sensor design space. Binary oversampling methods lie on the plane spanned by the temporal and spatial oversampling axes, while conventional image sensors lie on the analog-to-digital converter (ADC) bit depth axis. Multiexposure high dynamic range lies on the plane spanned by the temporal oversampling and ADC bit depth axes. Our theory describes the full three-dimensional design space.

Forward response model. The mathematical theory of operation of this sensor architecture is based on repeated conditional sampling from Poisson distributions.

An image pixel consists of S jots that are oversampled N times within one exposure. The image pixel response Y is the sum of the jot sampling results and, therefore, a number between 0 and $N \cdot S$ (other models could be used as well). Jots are grouped by type where jots of a given type all have the same area and the same threshold. The basic relations between these variables are $S = \sum_{i=1}^{n_T} s_i$, $t_{\exp} = \sum_{m=1}^{N} t_m$, and $A = \sum_{i=1}^{n_T} s_i a_i$, where S is the spatial oversampling, i.e., the number of jots in an image pixel, n_T is the number of types of jots (different types have different thresholds or area or both), s_i is the number of jots of type i in an image pixel, $t_{\exp}$ is the exposure time, N is the temporal oversampling, i.e., the number of readouts during exposure time $t_{\exp}$, t_m is the duration of sampling interval m (denoted as τ when constant), A is the area of an image pixel, and a_i is the area of jot of type i.

Photons impacting the image sensor are distributed according to a Poisson distribution. The probability of observing θ or more photons in a jot given an average incident photon number λ is therefore

$$Q(\lambda, \theta) \equiv \Pr[k \geq \theta; \lambda] = 1 - e^{-\lambda} \sum_{k=0}^{\theta-1} \frac{\lambda^k}{k!}. \quad (1)$$

Key to the calculation of the expected sensor response is the probability $p_{i,m}$ that a jot of type i will be at or above threshold in sampling interval m. This probability can be computed calculating forward from the first to the last sampling interval.

$$p_{i,1} = Q(\lambda_{i,1}, \theta_{i,1}), \quad (2)$$

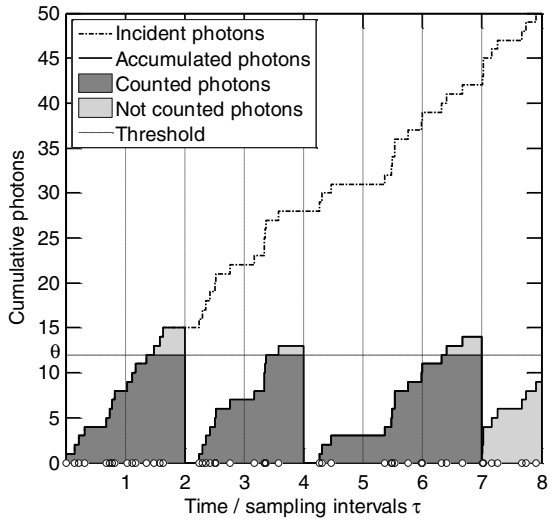

Fig. 2 The dots on the x axis indicate random photon strikes in a jot and the dot-dash line is their cumulative histogram. The vertical dashed lines show the sampling intervals of constant period τ. In this example ($\theta = 12$), the conditional reset occurs three times —$t = 2\tau$, 4τ, and 7τ, and thus the digital output is $y = 3$.

$$p_{i,2} = r_{i,1} Q(\lambda_{i,2}, \theta_{i,2}) + \sum_{n=0}^{\theta_{i,2}-1} \frac{\lambda_{i,1}^n e^{-\lambda_{i,1}}}{n!} Q(\lambda_{i,2}, \theta_{i,2} - n) + \sum_{n=\theta_{i,2}}^{\theta_{i,1}-1} \frac{\lambda_{i,1}^n e^{-\lambda_{i,1}}}{n!}, \tag{3}$$

$$p_{i,m\geq 3} = r_{i,m-1} Q(\lambda_{i,m}, \theta_{i,m}) + \sum_{n=0}^{\theta_{i,m}-1} P_{1,m-1}^{(i,n)} Q(\lambda_{i,m}, \theta_{i,m} - n) + \sum_{j=1}^{m-2} r_{i,j} \sum_{n=0}^{\theta_{i,m}-1} P_{j+1,m-1}^{(i,n)} Q(\lambda_{i,m}, \theta_{i,m} - n) + \sum_{n=\theta_{i,m}}^{\theta_{i,m}-1} P_{1,m-1}^{(i,n)} + \sum_{j=1}^{m-2} r_{i,j} \sum_{n=\theta_{i,m}}^{\theta_{i,m}-1} P_{j+1,m-1}^{(i,n)}. \tag{4}$$

Here, $p_{i,m}$ is the probability to sample at or above threshold at a jot of type i at sampling interval m, $r_{i,m}$ is the probability to reset a jot of type i at sampling interval m, $\lambda_{i,m}$ is the average number of photons impacting a jot of type i during sampling interval m, and $\theta_{i,m}$ is the sampling threshold of jot of type i in sampling interval m. The terms in Eqs. (3) and (4) denote the jots' sampling and reset history. The first term is the probability that the jot has been reset in the directly preceding interval, so that at least the threshold number of photons are needed to accumulate again to sample at or above threshold in the following interval. The other terms denote sequences where no reset has occurred in the directly preceding interval. They are summarized in Table 1. Each term needs to be multiplied with the probability of its occurrence and summed over the combinatorial possibilities of photon combinations to reach it.

The term $P_{a,b}^{(i,n)}$ is the probability of a photon sequence of total n photons distributed over the sampling intervals a through b in a way that the threshold is not reached in any sampling interval a through b and that the sensor is not reset after sampling. The intervals a and b denote any pair of sampling intervals with a being less equal to b. There are different equations to calculate $P_{a,b}^{(i,n)}$ depending on the sequence of thresholds and the reset operation (see below).

The expected value of the image pixel response is

$$\mathbb{E}[Y] = \sum_{i=1}^{n_T} \left(s_i \sum_{m=1}^{N} p_{i,m} \right). \tag{5}$$

Unconditional reset. In the case of unconditional reset (cf., Refs. 2 to 5), after each sampling $r_{i,m} = 1$ and $P_{a,b}^{(i,n)} = 0$. Equation (5), therefore, becomes

$$\mathbb{E}[Y] = \sum_{i=1}^{n_T} \left[s_i \sum_{m=1}^{N} Q(\lambda_{i,m}, \theta_{i,m}) \right]. \tag{6}$$

Conditional reset with thresholds constant over time. If the pixels are conditionally reset only if they are sampled at or above threshold (cf. Fig. 2), then $r_{i,m} = p_{i,m}$ and the full equations need to be used. As long as the thresholds do not vary in time, it is, however, possible to find a simplified expression for $P_{a,b}^{(i,n)}$ as only one threshold needs to be considered.

The third term of Eq. (3) and the fourth and fifth terms of Eq. (4) vanish since there is no change in threshold. The range of the sum over n of the second term of Eq. (3) and the second and third terms of Eq. (4) from 0 to threshold

Table 1 Sampling sequence types in a jot.

$Q(\lambda_{i,m}, \theta_{i,m})$	Pixel has been reset at sampling $m-1$ and threshold is reached at sampling m.
$P_{1,m-1}^{(i,n)} Q(\lambda_{i,m}, \theta_{i,m} - n)$	Pixel has never been reset before sampling m and threshold is reached at interval m by adding n photons in interval m.
$P_{j+1,m-1}^{(i,n)} Q(\lambda_{i,m}, \theta_{i,m} - n)$	Pixel has been reset at sampling $j < m-1$ and threshold is reached at interval m by adding n photons in interval m.
$P_{1,m-1}^{(i,n)}$	Pixel has never been reset before sampling m and threshold is reached at interval m without adding photons in interval m because the threshold of sampling m is n photons lower than the number of photons at the end of sampling $m-1$.
$P_{j+1,m-1}^{(i,n)}$	Pixel has been reset at sampling $j < m-1$ and threshold is reached at interval m without adding photons in interval m because the threshold of sampling m is n photons lower than the number of photons at the end of sampling $m-1$.

minus one makes sure that n is never at or above threshold. The probability of n photons below threshold in the sampling intervals a to b becomes, therefore, simply the Poissonian probability of n photons. Since the sum of Poissonian probabilities over expected photon numbers is the Poissonian probability of the sum over these expected photon numbers, the sampling sequence probability becomes

$$P_{a,b}^{(i,n)} = \frac{(\sum_{k=a}^{b} \lambda_{i,k})^n e^{-\sum_{k=a}^{b} \lambda_{i,k}}}{n!}. \tag{7}$$

Conditional reset with temporally variable threshold. The most complex calculation is for a sensor with thresholds varying over time where the pixels are conditionally reset only when they have been sampled above threshold. In this case, $P_{a,b}^{(i,n)}$ needs to be evaluated by examining the photon sequences in detail. The set $\Xi_{a,b}^{(i,n)}$ of photon sequences $\{\varphi_a \cdots \varphi_b\}$ is the subset of all possible sequences of total n photons that do not exceed the threshold at any sampling between intervals a through b. The elements φ of the sequence are the photon numbers reaching the sensor at each interval of the sequence. For computational simplification, all thresholds between sampling event a and b can be replaced by a monotonic sequence giving the sequence $\theta'_{i,a} \cdots \theta'_{i,b}$ with $\theta'_{i,a} \le \theta'_{i,b}$. Such a replacement is not necessary to calculate the sampling probabilities, but the time required to do the computation depends strongly on the number of photon sequences, and this simplification, therefore, reduces computation time. The sequence can be made monotonic since if a lower threshold would follow a higher threshold and the number of photons in that sequence would be between the high and low threshold, it would be above the low threshold and the sequence would therefore not be a sequence that satisfies the condition that the number of photons is below the threshold for all intervals from a to b. This allows replacing nonmonotonic θ with monotonic θ' by replacing high thresholds with following low thresholds. The sequence is then reduced to one entry per threshold value to the sequence $\theta^*_{i,a} \cdots \theta^*_{i,b}$ with $\theta^*_{i,a} < \theta^*_{i,b}$, and the sampling sequence probability becomes

$$P_{a,b}^{(i,n)} = \sum_{\{\varphi_a \cdots \varphi_{b^*}\} \in \Xi_{a,b^*}^{*(i,n)}} \prod_{l=a}^{b^*} \frac{\lambda_{i,l}^{*\varphi_l} e^{-\lambda^*_{i,l}}}{\varphi_l!}. \tag{8}$$

The set $\Xi_{a,b^*}^{*(i,n)}$ of photon sequences $\{\varphi_a \cdots \varphi_{b^*}\}$ is the subset of all possible photon sequences that fulfill the conditions $\sum_{l=a}^{b^*} \varphi_l = n$ and $\sum_{l=a}^{k} \varphi_l < \theta^*_{i,k}$ $\forall\ k \in [a, b^*]$. The list of effective thresholds $\theta^*_{i,k}$ fulfills the conditions $\theta^*_{i,k} < \theta^*_{i,k+1}$ $\forall\ k \in [a, b^* - 1]$, $\theta^*_{i,b^*} = \theta_{i,b}$, $\theta^*_{i,\mathrm{k}} \in \underbrace{\{\theta_{i,r}\}}_{a \le r \le b}$ $\forall\ k \in [a, b^*]$, $\theta_{i,r} \ge \theta_{i,r+1}$ $\forall\ r \in [a, b]$, and $\theta^*_{i,\mathrm{k}} = \theta_{i,b}$. The photon count in the modified intervals is determined as $\lambda^*_{i,l} = \sum_{r=a}^{b} \lambda_{i,r}$.

2.1.2 *Multibit sampling*

Equivalence of multibit and binary sampling. The equivalence between binary oversampling and multibit oversampling can be shown by examining the probabilities of the ADC to return a specific data number d. If the ADC sampling a pixel has as output a number between 0 and n_T in a sampling interval, then the expected pixel response in that interval is the sum over all possible ADC output values multiplied with their probability

$$\mathbb{E}[Y_m] = \sum_{i=1}^{n_T} i \cdot \Pr_{\mathrm{ADC}}[i; \lambda_m]. \tag{9}$$

Here, Y_m is the image pixel response at sampling interval m.

If the ADC has a step size of d_m in the temporal oversampling interval m, then the probability $\Pr_{\mathrm{ADC}}[i; \lambda_m]$ to return data number i in that interval given an average number of photons λ_m is the probability to sample $i \cdot d_m$ or more photons minus the probability to sample $(i+1) \cdot d_m$ or more photons.

$$\Pr_{\mathrm{ADC}}[i; \lambda_m] = \Pr[k \ge i \cdot d_m; \lambda_m] - \Pr[k \ge (i+1) \cdot d_m; \lambda_m]. \tag{10}$$

Assuming n_T virtual jots with thresholds at multiples of the ADC step size d_m in sampling interval m, one can set the threshold $\theta_{i,m}$ of the i'th virtual jot to

$$\theta_{i,m} = i \cdot d_m. \tag{11}$$

It follows then from the definition of $p_{i,m}$ as the probability to sample at or above threshold at a jot of type i at sampling interval m together with Eq. (9).

$$\begin{aligned}\mathbb{E}[Y_m] &= \sum_{i=1}^{n_T-1} i[p_{i,m}(\lambda_m, i \cdot d_m) - p_{i+1,m}(\lambda_m, (i+1) \cdot d_m)] \\ &\quad + n_T p_{n_T,m}(\lambda_m, n_T \cdot d_m) \\ &= \sum_{i=1}^{n_T-1} i p_{i,m}(\lambda_m, i \cdot d_m) - \sum_{i=2}^{n_T} (i-1) p_{i,m}(\lambda_m, i \cdot d_m) \\ &\quad + n_T p_{n_T,m}(\lambda_m, n_T \cdot d_m).\end{aligned} \tag{12}$$

Grouping the sums cancels the terms proportional to i and leaves

$$\mathbb{E}[Y_m] = \sum_{i=1}^{n_T} p_{i,m}(\lambda_m, i \cdot d_m) = \sum_{i=1}^{n_T} p_{i,m}(\lambda_m, \theta_{i,m}). \tag{13}$$

This is the expected value in sampling interval m of a pixel consisting of n_T binary sampled jots having thresholds according to Eq. (11), each jot receiving the average light intensity λ_m corresponding to the light intensity impacting the multibit sampled pixel. Multibit oversampling is, therefore, equivalent to binary oversampling with virtual jots having thresholds at the steps of the ADC.

There are, however, important differences between virtual and real jots that need to be considered. Real jots need to be placed in different spatial positions, while the virtual jots of the multibit sampling all occupy the area of the image pixel. Hence, virtual jots are larger than real jots for any given pixel size. Because the virtual jot has a larger photoactive area, a multibit oversampled image sensor will have better low-light response than a spatially oversampled binary image sensor if

all other factors such as sensitivity are held constant. Such a spatially oversampled image sensor can achieve the same low-light response only if it has jots that are more light-sensitive by a factor that compensates for the jot area reduction. Also, either all or none of the virtual jots that correspond to a pixel have to be reset, while a pixel using real jots and conditional reset would reset the jots above the threshold but not below the threshold. In Eqs. (3) and (4), a common reset threshold $r_{i,m} = \theta_{rst,m}$ has to be used when calculating reset probabilities. To calculate the final pixel response, ADC results are captured and summed up when the ADC is above the threshold and at a final residue read at the end of the exposure. The expected pixel response then becomes

$$\mathbb{E}[Y] = \sum_{i=1}^{n_T} \sum_{m=1}^{N-1} \frac{p_{\theta_{rst,m}}}{p_{1,m}} p_{i,m} + \sum_{i=1}^{n_T} p_{i,N}. \quad (14)$$

2.1.3 *Sampling policies*

The analytical model discussed above describes the sensor response as function of the selected oversampling type (binary or multibit, spatial, temporal, or both), the sequence of thresholds, the duration of temporal oversampling intervals, the area of the jots and pixels, and the reset conditions. We use the term "sampling policy" to describe the set of sampling periods, thresholds, and spatial areas of pixels comprising an image pixel. Different such policies will yield different response curves and noise characteristics. In an actual hardware sensor, some of the parameters can be varied in use while others are fixed at manufacturing. The selection of the right sampling policy defines the exposure setting of a sensor according to our proposal, similar to the selection of exposure time and ISO in a conventional sensor.

The achievable low end of the dynamic range is defined by the total exposure time, light sensitivity, and the noise level of the sensor. At very low light levels, no conditional reset will occur, and the sensor response will be the signal measured at the end of the exposure time proportional to all photons that have struck the pixel during that time. In multibit oversampling, a signal different from 0 will be reached if the ADC output is at least 1 data number. In spatial binary oversampling, the required condition is that at least one jot has reached or exceeded the threshold. The high end of the dynamic range on the other hand is defined by the duration of the shortest oversampling interval and the respective threshold and ADC step size used when sampling that interval. The measured signal will be meaningful, i.e., below saturation, if in the case of multibit oversampling the ADC output is at least 1 data number below its saturation or full well value and in the case of binary spatial oversampling if at least one jot has not reached the threshold.

2.2 *Monte Carlo Model and Sensor Noise*

2.2.1 *Monte Carlo model description*

The analytical model described above includes the full effect of photon shot noise since it is based on the Poisson statistics of the incident photons. A real image sensor also has intrinsic temporal and spatial noise sources that influence its response: read noise from the pixel read-out path, ADC noise, amplifier noise, and reset noise. Models for all these additional noise sources can be included in a Monte Carlo model of the sensor response.

The Monte Carlo model described here follows the approach of the analytical model by simulating Poisson distributed photons impacting the sensor, but it has models for the other noise sources listed in Table 2 added. Figure 3 shows the flow diagram of the Monte Carlo program. In the case of only temporal oversampling (either binary or multibit), there is only one real jot per pixel, so the expression "per real jot" in Fig. 3 can be read as "per pixel."

2.2.2 *Modeled noise sources*

An image sensor has a number of noise sources that have to be included when one wants to accurately simulate its

Table 2 Sensor noise in Monte Carlo program.

Noise	Cause	Model
Fixed pattern noise	Random deviations of circuitry, e.g., threshold voltage variation. Can be per pixel or per circuit that is shared by a number of pixels, e.g., an analog-to-digital converter (ADC).	Number of photo electrons has the same adder each time it is sampled. The adder is done as random number from a normal distribution around 0 either per pixel or for a group of pixels. The fit parameter is the standard deviation.
Temporal threshold noise	Analog comparison to a threshold, e.g., by using a sense-amplifier, is subject to noise that can be different for different time intervals.	The threshold is modified according to a normal distribution around 0. The fit parameter is the standard deviation.
Pixel response nonuniformity	Pixel-to-pixel gain variation. Noise is proportional to light intensity, i.e., number of photo electrons.	The number of photo electrons to be sampled is multiplied by a random number according to a normal distribution around 1. The fit parameter is the standard deviation.
Read noise	Along the analog read path from pixel to ADC noise can change the sampled value.	The number of photo electrons is modified according to a normal distribution around 0. The fit parameter is the standard deviation.
Reset noise	Switching the reset transistor off changes the reset level (thermal switching noise)	The reset level is modified according to a normal distribution around 0. The fit parameter is the standard deviation.

response. In the Monte Carlo model used in this work, we model the noise sources described in Table 2.

2.2.3 *Linearization of response*

Image processing chains of typical imaging systems expect that the sensor output is a linear function of light intensity when they perform functions like color demosaicking or white balance. If the binary above-threshold counts or the ADC outputs at each above-threshold event are directly added to each other, then the sum is generally not linear. Linearization can be achieved by a number of methods. One method is to precalculate the expected response as a function of light intensity and then to use a lookup table to get the light intensity from a measured pixel response. Another method is to use a weighted sum that linearizes the response, instead of the simple sum over individual samples. If each response that is above the threshold and becomes part of the final response is weighted by the ratio of the time since the previous reset to the total exposure time, then the response becomes linear. Only nonsaturated ADC outputs can be used in this method. If all ADC outputs are saturated, then the response becomes the total exposure time divided by the shortest interval duration. Let T_i denote the time since last reset before Y_i, and the linear response becomes

$$Y_{\text{linear}} = \begin{cases} \dfrac{t_{\text{exp}}}{\min\limits_{i \in 1 \cdots N} t_i} Y_{\text{sat}} & Y_i = Y_{\text{sat}} \quad \forall\, i \in 1 \cdots N \\ \dfrac{t_{\text{exp}}}{\sum_{i \text{ s.t. } Y_{\text{sat}} > Y_i \geq \theta_{rst}} T_i} \left(\sum\limits_{i \text{ s.t. } Y_{\text{sat}} > Y_i \geq \theta_{rst}} Y_i \right) & \exists i \quad \text{s.t. } 0 < Y_i < Y_{\text{sat}} \\ 0 & Y_i = 0 \quad \forall\, i \in 1 \cdots N \end{cases} . \tag{15}$$

Since the maximum possible result of Eq. (15) is larger than the saturated ADC output by a factor of the ratio of the total exposure time to the duration of the shortest sampling interval, while the direct sum has the saturated ADC output multiplied with the number of sampling intervals as maximum, the linearized response spans a wider numerical range than the direct sum if the sampling intervals are not of the same duration.

2.3 *Comparison of Sampling Policies*

2.3.1 *Spatial, temporal, and multibit oversampling*

Figures 4 (unconditional reset) and 5 (conditional reset) compare the different approaches. Lines denote the analytical model; symbols denote Monte Carlo simulation in the top graphs. The signal-to-noise ratio (SNR) shown in the bottom graphs is derived from Monte Carlo simulation. In all cases, the total bit depth is 8, and the various policies with temporal oversampling have variable sampling interval duration with the longest interval 128 times longer than the shortest to extend the dynamic range without increasing the total bit depth. The dotted black curve is the response of a conventional sensor with an 8-bit ADC and 20 electrons per data number saturating at 5100 electrons. The other curves are oversampling sensors, adding the result of the individual samplings. Gray curves are binary oversampling sensors with a threshold of 20 electrons. The dashed gray curve is of a sensor that oversamples only in time (256 times), while the solid gray curve oversamples both in space (16 jots) and time (16 times). The black dot-dash curve is a sensor oversampling in time (16 times) with a 4-bit ADC.

All sampling policies shown in Fig. 4 reset the pixel after each sampling. Without conditional reset, binary temporal and mixed temporal and spatial oversampling is equivalent (gray curves in Fig. 4). The bright-light response is extended for the binary sampling approach compared to that of the conventional sensor readout. The low-light response of the binary pixel without conditional reset is much worse than the conventional (black dotted curve) approach since the number of photons per jot and sampling interval is lower. The multibit temporal oversampled pixel (dot-dash black curve) has improved low-light response since the spatial oversampling is less.

Figure 5 shows the improved low-light response when conditional reset is used. The colors and line styles of the different curves correspond to the same sampling policies as in Fig. 4. Unlike the case shown in Fig. 4, pixels are reset only if they are sampled at or above the threshold (binary oversampled gray curves) or if the data number returned by the ADC is not zero (multibit oversampled black dot-dash curve). The only curve that shows reduced low-light response corresponds to the approach with spatial oversampling (solid gray curve). The temporally oversampled approaches keep the extended bright-light response. As a result of retaining low-light response and extending the bright-light response, the sensor dynamic range is extended. In this example, the dynamic range is extended by a factor of 20 compared to a conventional sensor, corresponding to an increase of effective full well capacity from 5100 electrons to over 100,000.

2.3.2 *Threshold sequencing*

Figure 6 shows a comparison of threshold sequences. The ascending binary oversampled threshold sequence of Fig. 6(a) will have a reduced low-light response compared to the descending sequence of Fig. 6(b) but an increased high end of the dynamic range. The reason is that the low-light limit of the dynamic range is reached when no intermediate conditional reset has occurred and the light collected over the full exposure time is assessed with the last threshold and readout. Since that last threshold is higher for an ascending sequence, such a sequence will have a reduced low-light response. The high end of the range is determined by a combination of threshold and sampling interval duration. In Fig. 6(a), the shortest interval has also the highest threshold and will, therefore, have the highest end of the dynamic range. Figure 6(c) shows the virtual jots when using multibit oversampling. Since the ADC steps are always present, both low and high thresholds are always present as well, and thus the wide dynamic range leads to

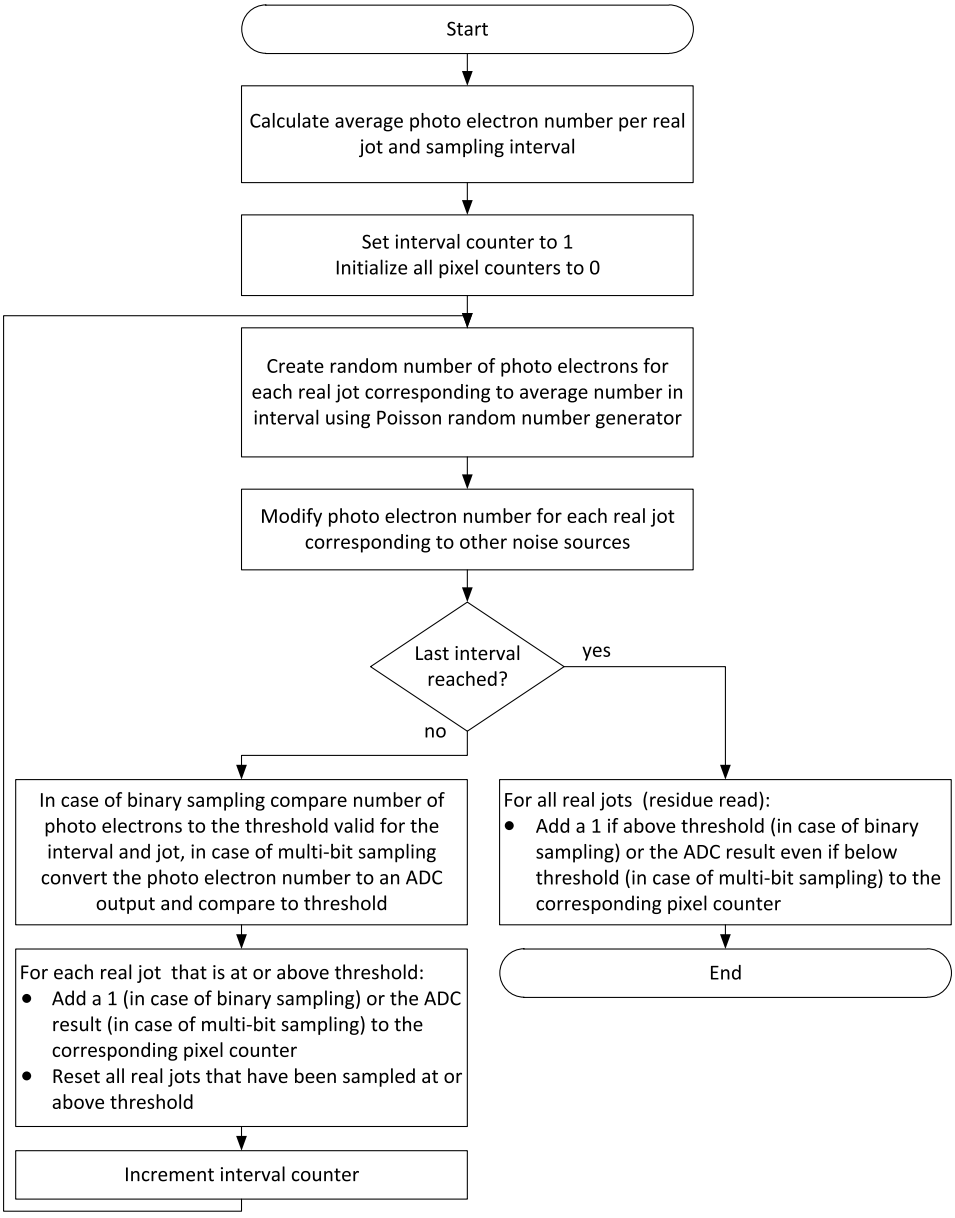

Fig. 3 Flow diagram of Monte Carlo model.

both a good low-light response and a high limit in bright illumination.

2.3.3 *Variation of interval duration*

Figure 7 compares three different sampling policies. The response is shown on top and the SNR on the bottom. All policies use multibit oversampling with a 4-bit ADC and a full well capacity of 300 electrons. The first two policies temporally oversample 16 times for a total bit depth of 8 bits. The first curve [(a), red dashes] varies the sampling interval duration logarithmically from 1 to 65 relative to each other so that the shortest interval is 1/261 of the total exposure time. This number was picked to be as close as possible to the equidistant ratio of the third curve with logarithmically spaced integer thresholds. The second curve [(b), solid blue] has intervals of equal length, each interval being $1/16$ of the total exposure time. The third curve [(c), black dot-dash) again uses equidistant temporal oversampling, however, with 256 samples for a total bit depth of 12 bits. The dynamic range of the policies with similar length of the shortest interval [(a) and (c)] is nearly the same, while the dynamic range of the approach with longer intervals (b) is much less. The SNR curves show that the price of achieving high dynamic range with fewer samplings is a reduced SNR at the high end. The response with equal sampling intervals stays close to linear and the SNR follows the photon shot noise limit, while the approach with varying interval duration has a response that shows quasi-logarithmic behavior and the SNR saturates.

2.4 *Hardware Verification*

We compared our sensor model to hardware on a small test chip and presented initial results in Vogelsang et al.[12] The results shown here are from later measurements of the

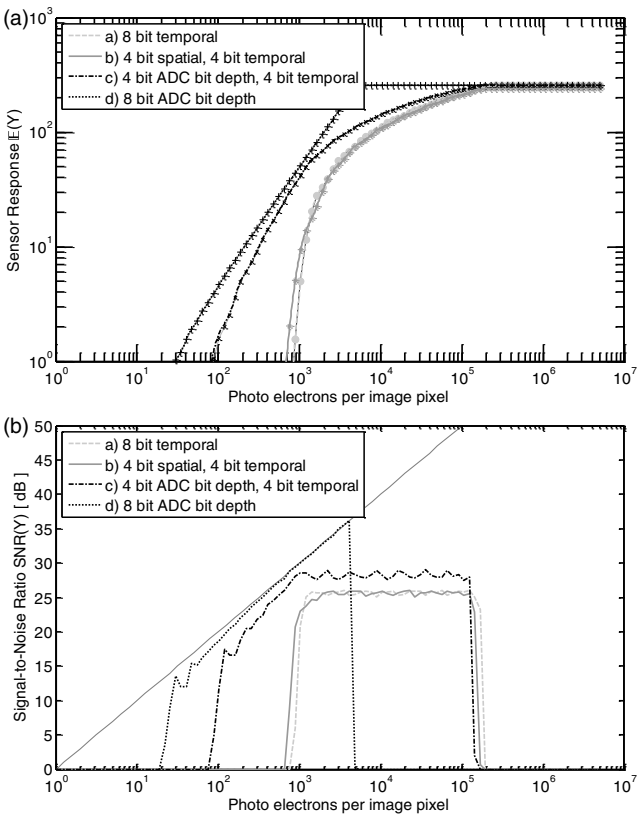

Fig. 4 Response (a) and signal-to-noise ratio (b) when using unconditional reset. All curves have a total bit depth of 8. The binary oversampled methods have the worst low-light response, followed by the temporally oversampled multibit method. None of the oversampled methods reaches the low-light response of the conventional sensor, but all extend the dynamic range at the high end.

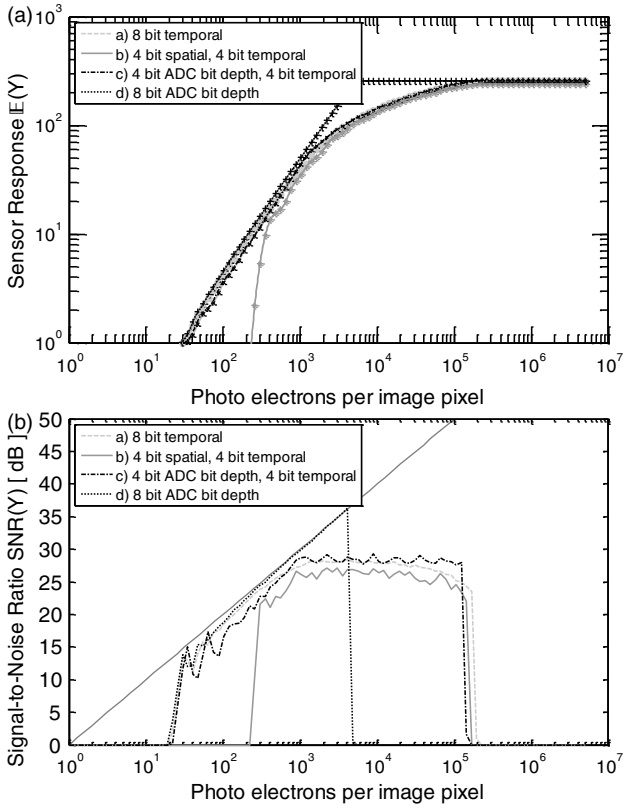

Fig. 5 Response (a) and signal-to-noise ratio (b) when using conditional reset. All curves have a total bit depth of 8. The temporally oversampled methods have the same low-light response as the conventional sensor, but extend the dynamic range at the high end. Only the spatially oversampled method has less low-light response.

same hardware. The pixel is based on a conventional 4T-pixel to which an additional transistor is added to provide column control in addition to row control for pixel reset. The test chip was built in a 180 nm CMOS image sensor technology using a fully pinned photodiode and a pixel pitch of 7.2 μm. The measured conversion gain is 83 $\mu V/e^-$. The read noise was fitted as 10 e^- and the photoresponse nonuniformity as 1.5%. The pixel was oversampled four times with a 10-bit ADC, and the ratio of the shortest interval to the total exposure time was 1:13.

Figure 8 shows a comparison of the simulated and measured response, and Fig. 9 shows the same comparison for the SNR. Linearization was done according to Eq. (15). The dashed and solid lines denote the simulated curves for conventional and oversampled operations, respectively, while the asterisks and circles, respectively, denote the corresponding measurement results. The agreement between measurement and simulation is very good. At an SNR of 0 dB, the dynamic range is 58 dB for the conventional operation and 79 dB for the oversampled operation. At an SNR of 20 dB, the dynamic range is 35 and 56 dB, respectively. The sampling policy chosen for this example did, therefore, expand the dynamic range by 21 dB or a factor of 11.

The SNR of the oversampled sensor shows a clearly visible dip in the extended dynamic range part of the curve. This dip is caused by not having identical duration of all subframes. In the measurement and simulation shown here, the first subframe had one quarter of the duration of the other three subframes. When subframes of similar duration are combined with oversampling with conditional reset, the dynamic range is extended with a smooth continuation of the SNR curve (here from $\sim 10^4$ photo electrons to $\sim 3 \cdot 10^4$ photo electrons). The shorter subframe, however, creates an SNR curve that is shifted according to the duration ratio. Combining this short subframe with the other three subframes extends the dynamic range further to $\sim 1.2 \cdot 10^5$ photo electrons, but the SNR is lower when there are so many photons that the longer subframes saturate. When designing the sampling policy for the sensor, it is important to make sure that such a dip does not extend below a desired SNR in order to not visibly degrade the image. In this example, the lowest point of the dip is over 30 dB, which will still give a very good image quality.

3 Camera Parameters and Sensor Modeling

3.1 *Scene Illuminance and Photons Per Pixel*

The sensor model described so far relies on the knowledge of the number of photo electrons per pixel to calculate the sensor response. When one wants to predict the sensor response when taking images with a camera, the illuminance of the scene has to be translated into the number of photo electrons sampled by each pixel.

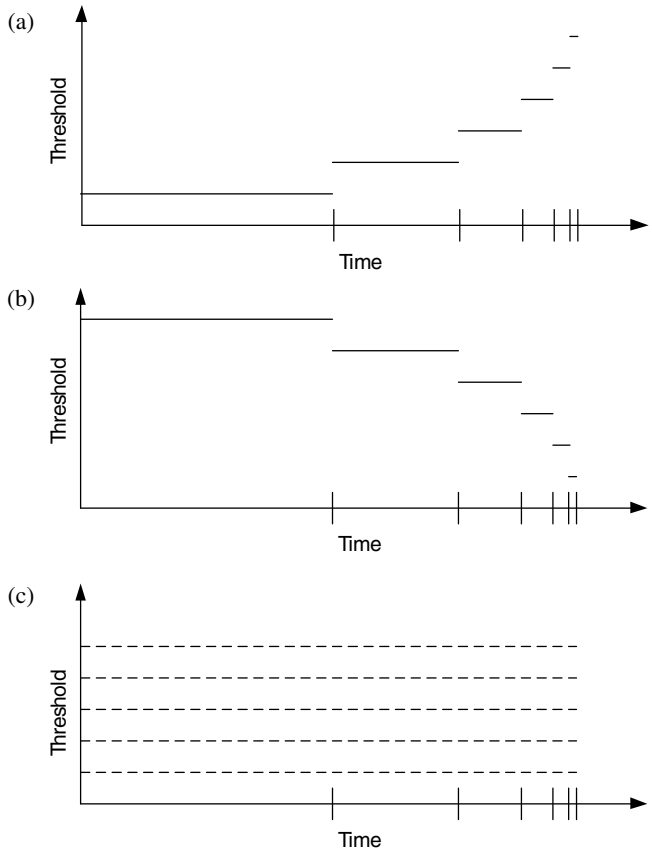

Fig. 6 Sampling policies with different threshold sequences. The dynamic range of the binary oversampled ascending sequence (a) will be shifted to higher light intensities compared to the descending sequence (b) since the bright-light response is determined by the threshold in the shortest interval, while the low-light response is determined by the threshold at the end of the exposure time when conditional reset is used. When using multibit oversampling, all thresholds are present in the virtual jots (c) at all intervals, and the response spans the widest range.

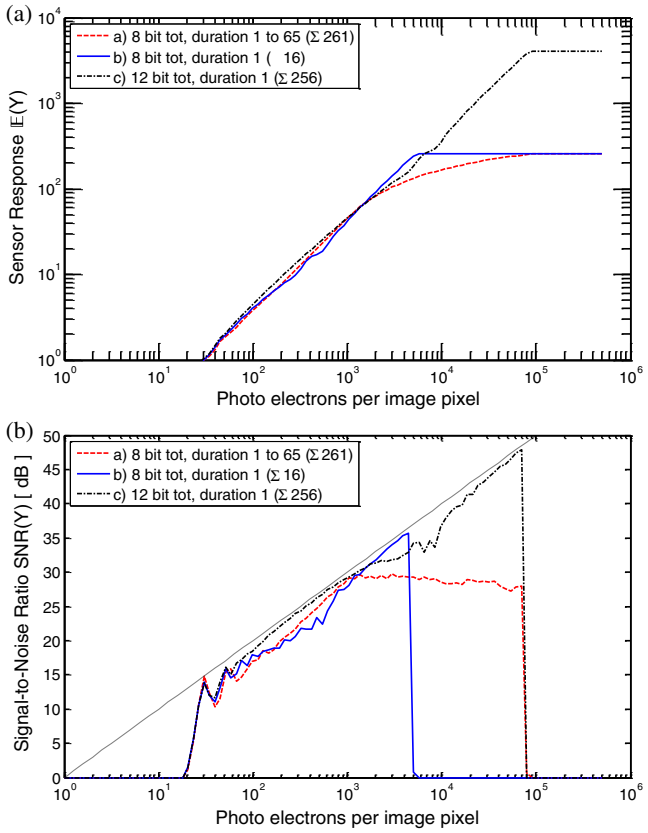

Fig. 7 Response (a) and signal-to-noise ratio (b) of policies with constant and variable interval duration. Variable interval duration can extend the dynamic range to be as large as when using a much higher total bit depth at the cost of a reduced signal-to-noise ratio at the high end.

3.1.1 *Photon energy and illumination*

To derive the number of photons impacting a pixel, it is first necessary to calculate the number of photons impacting a target area of which a camera is taking a picture of as a function of the specified illumination of that area.

The CIE spectral power density and luminosity curves[13] can be used to calculate the number of photons for a given illuminant. The luminous flux per Watt for the spectrum of the illuminant is the product of the luminosity function L and the spectral power density D_P integrated over the visible spectrum. By integrating the spectral power density of the illuminant over the wave length, average photon energy for the illuminant can be derived as well. Together these calculations give the number of photons that impact a given area a_{target} during the exposure time at a given illuminance E_V as

$$\tilde{\Lambda} = \gamma E_V t_{\text{exp}} a_{\text{target}}. \tag{16}$$

Here, $\tilde{\Lambda}$ is the average number of photons incident on an image pixel during the exposure time t_{exp} and γ is the photon density.

The constant γ that gives the number of photons per luminous energy and area is dependent on the illuminant. For the standard daylight spectrum CIE-D65, this constant is

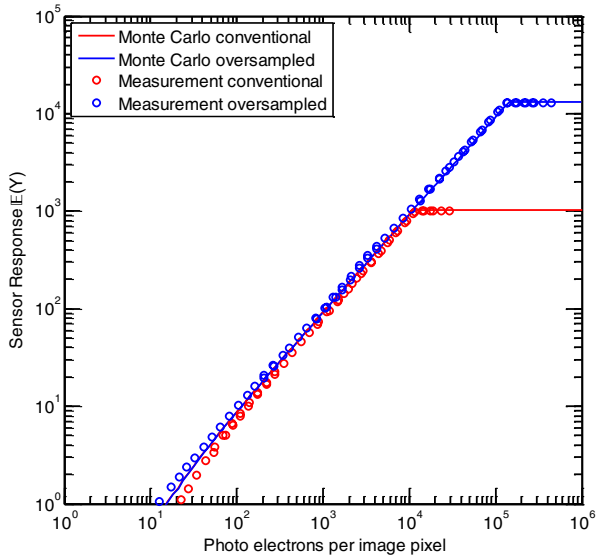

Fig. 8 Model-to-hardware comparison of the response of an image sensor operated conventionally, respectively, oversampled with conditional reset. The response of the oversampled sensor was linearized using the weighted sum of Eq. (15).

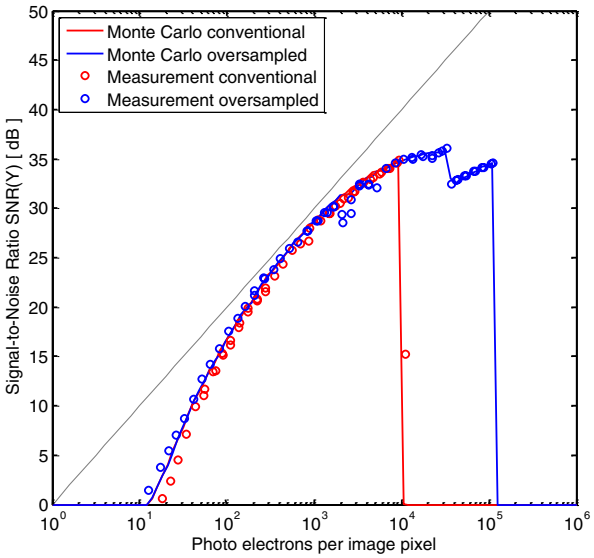

Fig. 9 Model-to-hardware comparison of the signal-to-noise ratio of the linearized response of an image sensor operated conventionally, respectively, oversampled with conditional reset. The dynamic range is extended by 21 dB.

$$\gamma_{D65} = 12612\ \mathrm{lx}^{-1}\,\mu\mathrm{m}^{-2}\,\mathrm{s}^{-1}. \tag{17}$$

For a spectrum that is shifted to longer wavelengths, the number of photons is higher for two reasons: (1) such a spectrum is not centered on the peak eye sensitivity; there is, therefore, more power needed for one lumen and (2) long-wavelength photons are less energetic, and therefore, more such photons are needed to provide a given power. For the standard incandescent spectrum CIE-A, the constant is

$$\gamma_A = 19892\ \mathrm{lx}^{-1}\,\mu\mathrm{m}^{-2}\,\mathrm{s}^{-1}. \tag{18}$$

3.1.2 *Lenses and pixel size*

The previous section related the illuminance of the target to the number of photons impacting that target. As a next step, the fraction of these photons that reach a pixel of the image sensor needs to be calculated. Assuming Lambertian reflection of the target and a lens focused at infinity, the illuminance of the sensor can be calculated from the illuminance of the target.

Multiplying with the quantum efficiency as the factor between the number of photons and the number of converted photo electrons gives an expression similar to the one derived in Ref. 14 that allows calculation of the number of photons sensed by an image pixel as a function of target illuminance, target reflectivity, the f-number of the lens, the exposure time, and the pixel area with

$$\Lambda = QE \cdot \gamma \frac{1}{4}\frac{\rho}{F^2} E_V^{\mathrm{target}} t_{\mathrm{exp}} A. \tag{19}$$

Here, Λ is the average number of photons sensed in an image pixel during the exposure time t_{exp}, QE is the quantum efficiency, ρ is the reflectivity, and F is the f-number.

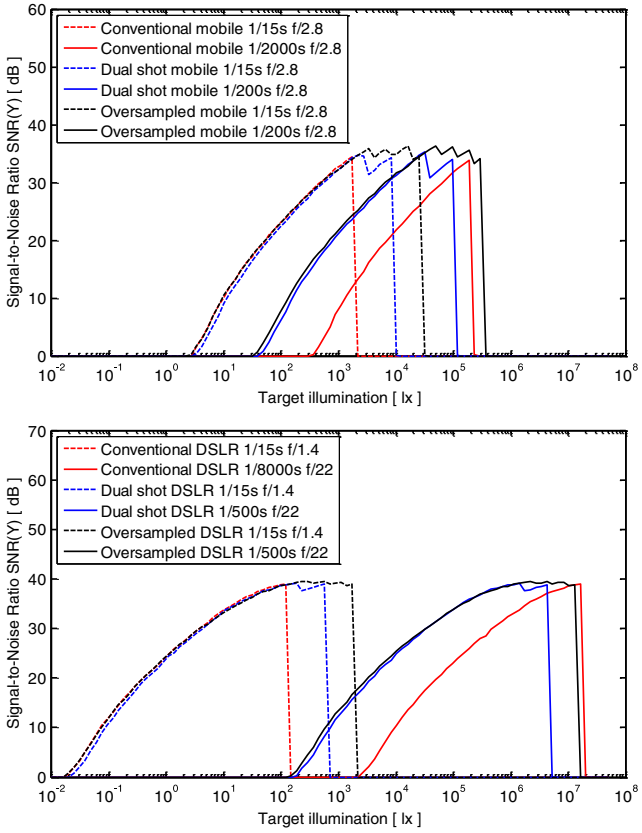

Fig. 10 Signal-to-noise ratio as function of scene illumination: (a) mobile sensor and (b) DSLR sensor. In both cases, a comparison at the low-light and at the bright-light end is shown. At the low-light end, the same total exposure time is used for the conventional and oversampled operation to match the low-light response, while at the high end, a longer exposure time is used for the oversampled operation to extend the dynamic range to the low end while matching the bright-light response.

3.2 *Low-Light and Dynamic Range Capabilities*

3.2.1 *Influence of pixel size and lens*

The simulations whose results are shown in Fig. 10 apply Eq. (19) to generate the number of photons as input for the sensor model based on target illumination. Compared are three different sensing schemes: the first is the conventional single-shot approach, the second is a dual-shot HDR approach, where the ratio between the short and long exposure is 1:4, and third is the oversampled approach proposed in this work. In all oversampled curves, the incident light is oversampled four times with intervals of relative sampling durations 1, 2, 4, and 8. At the lowest light intensity, no conditional reset occurs and the response is that of a conventional sensor, while at highest intensity, the pixel is above the threshold and, therefore, resets every time, and the shortest sampling interval becomes 1/15 of the exposure time.

The spectrum of the illumination used in the simulation was D65. If a color filter array were present, one would integrate the spectral density function of the light source with the photopic response, the quantum efficiency, and the spectral response of each of the three color filters. To simplify the task, we used here a combined quantum efficiency of 40% for the pixel and color filter together with the integral over spectral density function and photopic response. It was possible to use this approach as we only want to compare the

impact of pixel parameters and optics on the light sensitivity, not simulate a specific hardware solution. We used a target reflectivity of 18% for the simulations.

Figure 10(a) shows the simulated SNR of a small-pixel sensor typical for a mobile system. We assumed a pixel pitch of 1.1 μm and a fixed f-number of f/2.8. The full well capacity was assumed to be 5000 electrons and a 10-bit ADC was used. The exposure time of the conventional sensor (red curves) in a low-light situation (dashed lines) was assumed to be 1/15 s, close to the limit that can be done with a handheld camera. At the high end (solid lines), 1/2000 s of exposure time gives a high end of the dynamic range that is sufficient to take images in bright sunlit outdoor scenes. The oversampled sensor (black curves) uses the same exposure 1/15 s setting for the low-light situation, but increases the total exposure time for the bright-light situation to 1/200 s. The dynamic range is extended by 24 dB in both situations. At the low-light situation, this extension occurs at the high-intensity end. At the bright-light situation, our choice of exposure parameters extends the dynamic range mostly to the lower end.

Figure 10(b) shows similar results for a digital single lens reflex camera (DSLR)-type sensor. The assumed pixel pitch was 6.3 μm, the f-number was variable between f/1.4 and f/22, the full well capacity was 50,000, and a 14-bit ADC was used. The extension of the dynamic range is similarly 24 dB between a conventional and our proposed oversampled sensor since the sampling policy is the same. As in the example of the mobile sensor, we selected an increased exposure time for the bright situation to extend the dynamic range to the lower end.

Figures 10(a) and 10(b) clearly show that the DSLR can cover a much wider total dynamic range than the mobile sensor. The extension of the dynamic range at the low end comes mainly from the large pixel area that accepts many more photons into a pixel according to Eq. (19). The larger aperture that is possible with DSLR lenses contributes as well. At the high end, the ability to make the aperture very small together with a shorter minimum exposure time extends the range.

It is also clear that the multibit oversampling with conditional reset approach provides a small pixel mobile camera system with very wide dynamic range and wide exposure latitude. A mobile camera system using this approach can expose for the low-light regions of the scene while retaining all of the bright-light information and detail.

Both SNR curves show a nonmonotonic behavior at high intensities. This is caused by the varying duration of the sampling intervals similar to Fig. 9. Figure 9 had only two different durations and, therefore, only one visible dip, while in Fig. 10, all four durations are different and there are three visible dips. Both for the mobile sensor and the DSLR, the dips occur above 30 dB SNR and will, therefore, not be visible on the final image.

The ability of the dual-shot approach (blue curves) to capture a wide dynamic range is as expected between the single-shot approach and our method. The nonmonotonic dip of the SNR curve in the dual-shot approach is more pronounced than in our approach since there are only two possible exposure times available, while in our approach, the conditional reset makes available all combinations of interval durations between the shortest interval length and the full exposure time. The dual-shot approach shows a slightly reduced SNR at the low-light end since we assumed that the sum of the short and long exposures is the same for all three methods compared, thereby making the long exposure shorter than both the total exposure time in our approach and the exposure time of a single shot.

3.2.2 *Optimization of low-light response*

Figure 11 shows the response and SNR of the small-pixel sensor of Fig. 10(a) for different full well capacity and read noise. Since the dynamic range of the oversampled method is so much higher, the full well capacity can be reduced, in this example, from 5000 to 1250 electrons to shift the response curve to lower light intensities while still increasing the dynamic range at high light intensity. If the reduced full well capacity can be used to reduce the read noise as well, e.g., by a higher conversion gain, then the SNR at low light intensities can be improved compared to the conventional sensor.

Figure 12 demonstrates this using the example of an image with 96 dB dynamic range. The top set of images is false color, comparing the quality of the reconstruction by showing the relative difference between the reconstructed linearized response and the original input data. The bottom set of images shows tone mapped simulation output. Identical tone mapping has been applied to all simulations. The small-pixel conventional sensor simulated in Fig. 10 is

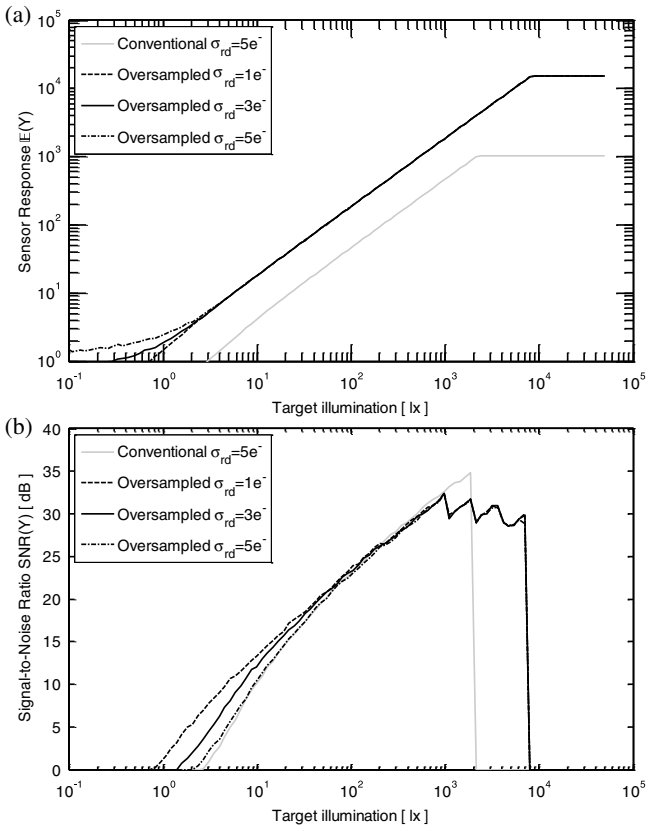

Fig. 11 Comparison of response (a) and signal-to-noise ratio (b) of conventional and oversampled image sensors. The full well capacity of the conventional sensor is 5000 electrons. The oversampled sensor has 1250 electrons full well capacity and is oversampled four times with relative sample interval durations of 1, 2, 4, and 8. If the reduced full well capacity can be used to reduce the read noise, then the low-light response can be improved.

xxiii

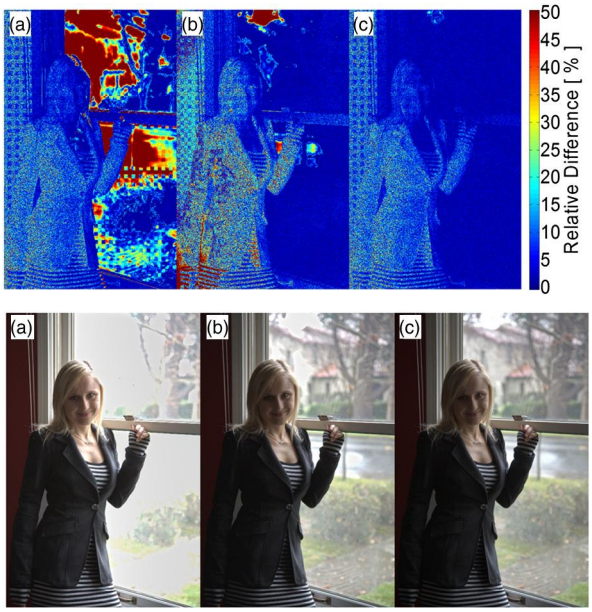

Fig. 12 False color (top) and tone mapped (bottom) output of an image simulation comparing conventional and oversampled sensors. The false color image shows the relative difference between the original and reconstructed linear response. The two left images (a) and (b) are calculated from the simulated response of the small-pixel conventional sensor used in Fig. 10; the right image is calculated from the simulated response of the oversampled sensor with reduced full well capacity and read noise. Different from Fig. 1, the sensor has been oversampled six times with relative sample interval durations of 1, 2, 4, 8, 16, and 32. Images (a) and (c) are exposed at the same exposure value, while image (b) is exposed at a two stops higher exposure value. The conventional sensor has either blown out highlights (a) or increased noise in the dark areas (b), while the oversampled sensor has high-quality reconstruction over the full dynamic range.

shown with two exposures, one optimized for the darker parts of the image (a) and one for the brighter parts of the image (b). The difference between the two exposures is two stops. The first exposure has overexposed highlights, while the second has significantly increased noise in the dark regions. The oversampled image (c) has reduced full well capacity (from 5000 to 1250 electrons) and sensor read noise (from 5 to 1 electron). The sequence of relative sampling durations was chosen to be 1, 2, 4, 8, 16, and 32 to match the large dynamic range of the input image. The total exposure is the same as for image (a). The oversampled approach gives even better bright-light reconstruction as the short exposure with the conventional sensor and excellent reconstruction of the dark parts of the image as well.

Figure 13 is another example comparing different HDR approaches. In this case, the four images compare a conventional single-shot exposure, the line-interleaved HDR approach in which alternating pairs of rows are exposed with different exposure times, dual-shot HDR blending two images with different exposure time, and the oversampling approach with conditional reset proposed in this work. The relative sample interval durations of our approach are 12, 1, 1, 1 in this example. Again, exposure values have been selected to get the best overall image quality in all cases. The exposure value of the line-interleaved and dual-shot HDR are, therefore, one stop higher than single shot, and the image taken with our approach is exposed three stops more. The line-interleaved image has a ratio of 2:1 between the long and short exposures, while the dual-shot image has a ratio of 4:1. A comparison of the images clearly shows the benefit of our approach. The single-shot image is worst with saturated highlights and significant noise in the dark. The line-interleaved approach has better highlights, but the noise in the dark gets even worse due to the necessary interpolation. Dual shot does not have this problem and has both better exposed highlights and less noise, but it is still significantly noisier than our approach.

4 Conclusions

We have developed a theoretical model that describes light capture of a photosensor based on photon statistics, thereby incorporating photon shot noise directly. This model describes the sampling of photons as a series of binary comparisons with a threshold. We showed in previous work that multibit sampling with an ADC is mathematically equivalent

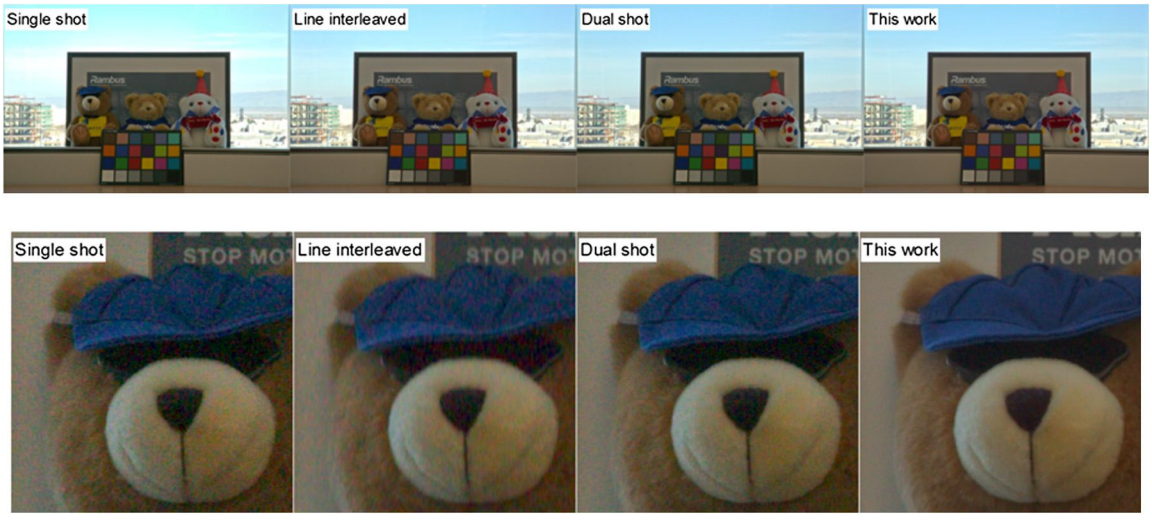

Fig. 13 Full image (top) and zoom into dark part (bottom) of a high dynamic range image. The exposure has been adjusted to take advantage of wider dynamic range; line interleaved and dual shot are therefore exposed one stop more than the single-shot image, and the oversampling with conditional reset of this work is exposed three stops more than the single-shot image.

to spatially oversampling the pixel with virtual jots that are sampled with thresholds at the steps of the ADC. Our sensor model can, therefore, be used to predict and optimize the light response of any binary oversampling sensor, conventional single-sample multibit sensors, and multibit oversampling sensors. The sensor response can be linearized either by a lookup table or by a weighted sum of the results of the individual samplings. We verified this model on hardware using a small test chip. Using the model, we demonstrated that sampling policies that use only temporal oversampling (binary or multibit) and reset the pixel only conditionally when a threshold has been reached have better low-light response than sampling policies with unconditional reset or spatial oversampling. By calculating the number of photons on the sensor based on target illumination and camera parameters, we were able to compare exposure settings for low-light and bright-light settings of conventional sensors with oversampled sensors both for sensors typical for mobile devices as for DSLR sensors. A significant increase of dynamic range of ∼24 dB in our example can be seen in all cases. In a typical camera application, the dynamic range would be extended to the high end in a low-light situation and to the low end in a bright-light situation. The dynamic range of an oversampled mobile camera can be as large as the range of a conventional DSLR in medium- or bright-light situations. Such a matchup is not possible either at very low light situations where the pixel size is important to collect as many photons as possible or at very bright light situations where the aperture needs to be changed to let less light on the sensor. While the high end can be further extended in all cases by more oversampling, at the low end, an improvement is only possible when more photons can be collected by having a larger pixel area, higher pixel sensitivity, or a combination of these approaches. We expect that the pixel can be designed to achieve higher sensitivity when using our approach as there is no need to have a large full well capacity to handle brightly lit parts of the scene. More generally, low-light response can be improved in camera systems employing sensors having multibit oversampling with conditional reset by exposing for the low-light regions of the scene while retaining all of the bright-light information and detail.

References

1. P. Seitz and A. Theuwissen, *Single-Photon Imaging*, Springer Series in Optical Sciences, Springer, Berlin, Heidelberg (2011).
2. E. Fossum, "What to do with sub-diffraction-limit (SDL) pixels," in *IEEE Workshop on Charge-Coupled Devices and Advanced Image Sensors*, pp. **214**–217, IEEE (2005).
3. E. Fossum, "The quanta image sensor (QIS): concepts and challenges," presented at *2011 OSA Topical Meeting on Optical Sensing and Imaging*, Toronto, Canada, Optical Society of America (10–14 July 2011).
4. L. Sbaiz et al., "The gigavision camera," in *IEEE Conf. on Acoustics, Speech and Signal Processing*, pp. 1093–1096, IEEE (2009).
5. F. Yang et al., "Bits from photons: oversampled image acquisition using binary Poisson statistics," *IEEE Trans. Image Process.* **21**(4), 1421–1436 (2012).
6. T. Vogelsang and D. G. Stork, "High-dynamic-range binary pixel processing using non-destructive reads and variable oversampling and thresholds," in *IEEE Sensors*, pp. 1–4, IEEE (2012).
7. S. Kavusi, H. Kakavand, and A. El Gamal, "Quantitative study of high-dynamic range SigmaDelta-based focal plane array architectures," *Proc. SPIE* **5406**, 341–350 (2004).
8. Z. Ignjatovic, D. Maricic, and M. F. Bocko, "Low power, high dynamic range CMOS image sensor employing pixel level oversampling—analog-to-digital conversion," *IEEE Sensors J.* **12**(2012), 737–746 (2012).
9. P. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *Proc. of 24th Annual Conf. on Computer Graphics and Interactive Techniques*, pp. 369–378, ACM Press/Addison-Wesley Publishing Co, New York, NY (1997).
10. D. Yang and A. El Gamal, "Comparative analysis of SNR for image sensors with enhanced dynamic range," *Proc. SPIE* **3649**, 197–211 (1999).
11. D. Yang et al., "A 640 x 512 CMOS image sensor with ultrawide dynamic range floating-point pixel-level ADC," *IEEE J. Solid-State Circuits* **34**(12), 1821–1834 (1999).
12. T. Vogelsang, M. Guidash, and S. Xue, "Overcoming the full well capacity limit: high dynamic range imaging using multi-bit temporal oversampling and conditional reset," in *International Image Sensors Workshop*, Snowbird, Utah, International Image Sensor Society (12–16 June 2013).
13. "CIE standard colorimetric observer data and CIE standard illuminant data," http://www.cis.rit.edu/mcsl/online/cie.php.
14. J. Alakarhu, "Image sensors and image quality in mobile phones," in *Int. Image Sensors Workshop*, International Image Sensor Society, Ogunquit, Maine (June 2007).

Thomas Vogelsang received his diploma and doctoral degree in physics from the Technical University Munich. He worked on DRAM from 1994 to 2009 at Siemens, Infineon, Qimonda, and Rambus. Since 2010 he has been a member of the Computational Sensing and Imaging group of Rambus Labs and leading the work on binary pixels. He is a senior member of the IEEE, and author or co-author of 14 publications, 20 patents and over 30 pending patent applications.

David G. Stork is Rambus Fellow and leads the Computational Sensing and Imaging Group within Rambus Labs. A graduate in physics from MIT and the University of Maryland, he has held faculty positions in eight disciplines in several leading liberal arts and research universities. His published eight books/proceedings volumes variously translated into four languages. He holds 43 patents and is a fellow of the International Association for Pattern Recognition and SPIE.

Michael Guidash received his BS in electrical engineering from the University of Delaware, and MS in electrical engineering from Rochester Institute of Technology. Michael worked at Kodak from 1981 to 2011 in the fields of CCD and CMOS image sensors. He led the R&D and product development of CMOS sensors. He is now consulting in CMOS sensor technology. He is an author or co-author of 15 publications, 69 patents and over 20 patent applications.

Stereo vision–based depth of field rendering on a mobile device

Qiaosong Wang,* Zhan Yu, Christopher Rasmussen, and Jingyi Yu
University of Delaware, Newark, Delaware 19716

Abstract. The depth of field (DoF) effect is a useful tool in photography and cinematography because of its aesthetic value. However, capturing and displaying dynamic DoF effect were until recently a quality unique to expensive and bulky movie cameras. A computational approach to generate realistic DoF effects for mobile devices such as tablets is proposed. We first calibrate the rear-facing stereo cameras and rectify the stereo image pairs through FCam API, then generate a low-res disparity map using graph cuts stereo matching and subsequently upsample it via joint bilateral upsampling. Next, we generate a synthetic light field by warping the raw color image to nearby viewpoints, according to the corresponding values in the upsampled high-resolution disparity map. Finally, we render dynamic DoF effect on the tablet screen with light field rendering. The user can easily capture and generate desired DoF effects with arbitrary aperture sizes or focal depths using the tablet only, with no additional hardware or software required. The system has been examined in a variety of environments with satisfactory results, according to the subjective evaluation tests. *© 2014 SPIE and IS&T* [DOI: 10.1117/1.JEI.23.2.023009]

Keywords: depth of field; programmable cameras; joint bilateral upsampling; light field.

Paper 13493SSP received Sep. 4, 2013; revised manuscript received Jan. 31, 2014; accepted for publication Feb. 26, 2014; published online Mar. 19, 2014.

1 Introduction

Dynamic depth of field (DoF) effect is a useful tool in photography and cinematography because of its aesthetic value. Capturing and displaying dynamic DoF effect were until recently a quality unique to expensive and bulky movie cameras. Problems such as radial distortion may also arise if the lens system is not setup properly.

Recent advances in computational photography enable the user to refocus an image at any desired depth after it has been taken. The hand-held plenoptic camera[1] places a microlens array behind the main lens, so that each microlens image captures the scene from a slightly different viewpoint. By fusing these images together, one can generate photographs focusing at different depths. However, due to the spatial-angular tradeoff[2] of the light field camera, the resolution of the final rendered image is greatly reduced. To overcome this problem, Georgiev and Lumsdaine[3] introduced the focused plenoptic camera and significantly increased spatial resolution near the main lens focal plane. However, angular resolution is reduced and may introduce aliasing effects to the rendered image.

Despite recent advances in computational light field imaging, the costs of plenoptic cameras are still high due to the complicated lens structures. Also, this complicated structure makes it difficult and expensive to integrate light field cameras into small hand-held devices like smartphones or tablets. Moreover, the huge amount of data generated by the plenoptic camera prohibits it from performing light field rendering on video streams.

To address this problem, we develop a light field rendering algorithm on mobile platforms. Because our algorithm works on regular stereo camera systems, it can be directly applied to existing consumer products such as three-dimensional (3-D)-enabled mobile phones, tablets, and portable game consoles. We also consider the current status of mobile computing devices in our software system design and make it less platform dependent by using common libraries such as OpenCV, OpenGL ES, and FCam API. We start by using two cameras provided by the NVIDIA Tegra 3 prototype tablet to capture stereo image pairs. We subsequently recover the high-resolution disparity maps of the scene through graph cuts (GCs)[4] and then generate a synthesized light field for dynamic DoF effect rendering. Once the disparity map is generated, we synthesize a virtual light field by warping the raw color image to nearby viewpoints. Finally, dynamic DoF effects are obtained via light field rendering. The overall pipeline of our system is shown in Fig. 1. We implement our algorithm on the NVIDIA Tegra 3 prototype tablet under the FCam architecture.[5] Experiments show that our system can successfully handle both indoor and outdoor scenes with various depth ranges.

2 Related Work

Light field imaging opens up many new possibilities for computational photography, because it captures full four-dimensional radiance information about the scene. The captured light information can later be used for applications like dynamic DoF rendering and 3-D reconstruction. Since conventional imaging systems are only two-dimensional (2-D), a variety of methods have been developed for capturing and storing light fields in a 2-D form. Lippmann[6] was the first to propose a prototype camera to capture light fields. The Stanford multicamera array[7] is composed of 128 synchronized CMOS firewire cameras and streams, capturing data to four PC hosts for processing. Because of the excessive data

*Address all correspondence to: Qiaosong Wang, E-mail: qiaosong@udel.edu

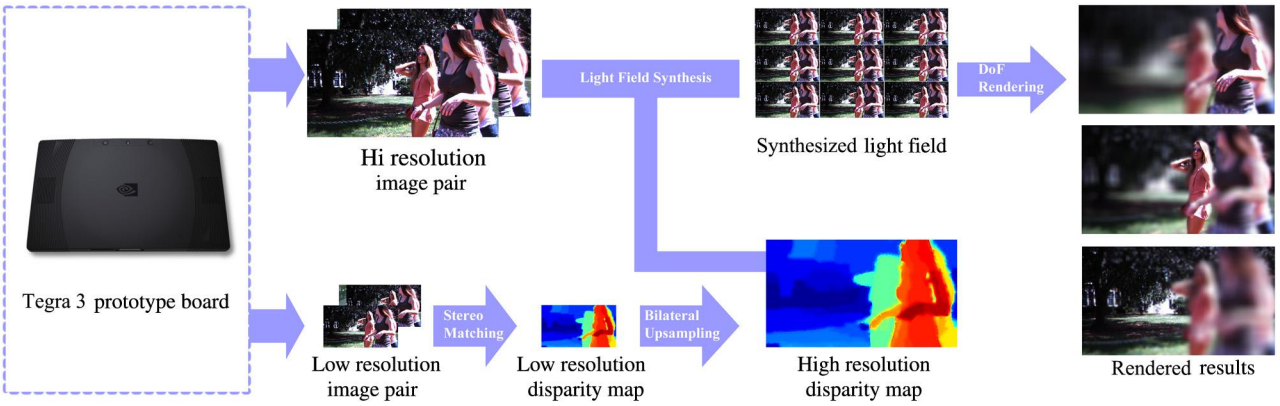

Fig. 1 The NVIDIA Tegra 3 prototype tablet and the processing pipeline of our software system. All modules are implemented on the Android 4.1 operating system.

volume, DoF effects are rendered offline. The Massachusetts Institute of Technology light field camera array[8] uses 64 usb webcams and is capable of performing real-time rendering of DoF effects. However, these camera systems are bulky and hard to build. Recently, Ng et al.[1] have introduced a new camera design by placing a microlens array in front of the sensor with distance equals microlens focal length, wherein each microlens captures a perspective view in the scene from a slightly different position. However, the spatial resolution near the microlens array plane is close to the number of microlenses. To overcome this limitation, Georgiev and Lumsdaine[3] introduced the focused plenoptic camera which trades angular resolution for spatial resolution. An alternative approach is to integrate light-modulating masks to conventional cameras and multiplex the radiance in the frequency domain.[9] This design enables the camera sensor to capture both spatial and angular frequency components, but reduces light efficiency.

As the rapid research and development provide great opportunities, hand-held plenoptic camera has been proven practical and quickly progressed into markets. The Lytro camera[10,11] is the first implementation of a consumer-level plenoptic camera. Recently, Pelican Imaging[12] announced a 16-lens mobile plenoptic camera system and scheduled to implement it to new smartphones in 2014.

Our work is inspired by the algorithms proposed by Yu et al.[13,14] However, the system proposed in these two papers is bulky and expensive, and the algorithm is highly dependent on the GPU performance, making it hard to transfer the proposed method to small hand-held devices such as cellphones and compact size cameras. The system used by Yu et al.[13] is composed of a desktop workstation and a customized stereo camera system. The desktop is equipped with a 3.2 GHz Intel Core i7 970 6-core CPU and a NVIDIA Geforce GTX 480 Graphic Card with 1.5 GB memory. Actually, very few laptops on the market can reach the same level of performance, let alone tablets or cellphones. Also, this system connects to two Point Grey Flea 2 cameras via a Firewire link. The retail price for two Flea cameras is around $1500, and the camera itself requires external power source and professional software for functionalities such as auto exposure, white balancing, and stereo synchronization, which is almost impractical for general users without a computer vision background. In addition, most scenes in this article are indoor scenes with controlled lighting, and the user is required to tune different parameters on a GUI in order to obtain a good-looking disparity map in different scenes. In contrast, our software system works directly on an off-the-shelf tablet, which costs less than $400. Since our algorithm is implemented under the Android operating system using highly optimized CPU-only functions from OpenCV4Android SDK, it can be easily ported to other hand-held Android devices with limited computational power. Besides, we conducted extensive experiments to obtain parameters that generate optimal results. Therefore, it is easy to install and use our software, no hardware setup or parameter adjustment is required. Furthermore, our system uses GCs[15] instead of belief propagation (BP)[16] for stereo matching and is tested working under complex illumination conditions. According to the tests carried out by Tappen and Freeman,[17] GCs generate smoother solutions compared with BP and consistently perform better than BP in all quality metrics for the Middlebury[18] Tsukuba benchmark image pair. To conclude, we made the following contributions:

- We propose light field rendering as a possible solution to generate dynamic DoF effects. We also discussed why our method is good at reducing boundary discontinuity and intensity leakage artifacts compared with depth-based image blurring schemes.
- We implemented the entire system on an off-the-shelf Android tablet using highly optimized CPU-only functions from OpenCV4Android SDK. The system can be easily ported to other mobile photography devices with limited computational power.
- We conducted extensive experiments to obtain the optimal combination of methods and parameters under the Tegra 3 T30 prototype device. As a result, there is no need for parameter adjustment and it is easy for the user to install and use our application.
- We experimented with GCs for disparity map calculation, and the system is capable of working with a variety of scene structures and illumination conditions.

3 Overview

In this article, we demonstrate that the DoF effects can be rendered using low-cost stereo vision sensors on mobile devices. We first capture stereo image pairs by using

the FCam API and then apply the GCs stereo-matching algorithm to obtain low-resolution disparity maps. Next, we take raw color images as guide images and upsample the low-resolution disparity maps via joint bilateral upsampling. Once the high-resolution disparity maps are generated, we can synthesize light fields by warping the raw color images from the original viewing position to nearby viewpoints. We then render dynamic DoF effects by using the synthetic light fields and visualize the results on the tablet screen. We evaluate a variety of real-time stereo-matching and edge-preserving upsampling algorithms for the tablet platform. Experimental results show that our approach provides a good tradeoff between expected depth-recovering quality and running time. All aforementioned processing algorithms are implemented to the Android operating system and tested on the Tegra 3 T30 prototype tablet. The user can easily install the software and capture and generate desired DoF effects using the tablet only, with no additional hardware or software required. The system has been tested in a variety of environments with satisfactory results.

4 Programmable Stereo Camera

4.1 *Development Environment*

The Tegra 3 T30 prototype tablet is equipped with a 1.5 GHz quad-core ARM Cortex-A9 CPU and a 520 MHz GPU. It has three sensors. The rear main sensor and secondary sensor are identical with a 6-cm baseline. The third sensor is on the same side of the multitouch screen facing the user. The raw image resolution is 640×360 (16:9).

Our software is running under Android 4.1 (Jelly Bean) operating system. We use the Tegra Android Developer Pack (TADP) for building and debugging the application. This software toolkit integrates Android SDK features to Eclipse IDE by using the Android Development Tools (ADT) Plugin. The ADT extends the capabilities of Eclipse and enables the user to design graphic UI, debug the application using SDK tools, and deploy APK files to physical or virtual devices. Since typical Android applications are written in Java and compiled for the Dalvik Virtual Machine, there is another toolset called Android Native Development Kit (NDT) for the user to implement part of the application in native code languages such as C and C++. However, using the NDT brings certain drawbacks. First, the developer has to use the NDT to compile native code, which hardly integrates with the Java code, so the complexity of the application is increased. Besides, using native code on Android system generally does not result in a noticeable improvement in performance. For our application, since we need to use the FCam API for capturing stereo pairs and OpenCV and OpenGL ES for image processing and visualization, we implemented most of the code in C++ and run the code inside the Android application by using the Java Native Interface (JNI). The JNI is a vendor–neutral interface that permits the Java code to interact with the underlying native code or load dynamic-shared libraries. By using the TADP, our workflow is greatly simplified. We first send commands to the camera using the FCam API, then convert raw stereo image pairs to *cv::Mat* format, and use OpenCV for rectification, stereo matching, joint bilateral upsampling, and DoF rendering. The final results are visualized on the screen using OpenGL ES.

4.2 *FCam API*

Many computational photography applications follow the general pattern of capturing multiple images with changing parameters and combining them into a new picture. However, implementing these algorithms on a consumer-level tablet has been hampered by a number of factors. One fundamental impediment is the lack of open software architecture for controlling the camera parameters. The Frankencamera[5] proposed by Adams et al. is the first architecture to address this problem. Two implementations of this concept are a custom-built F2 camera and a Nokia N900 smartphone running on a modified software stack to include the FCam API. Troccoli et al. extended the implementation of FCam API to support multiple cameras[19] and enabled the NVIDIA Tegra 3 prototype tablet to trigger stereo captures.

4.3 *Calibration, Synchronization, and Autofocus*

Since the two sensors are not perfectly aligned, we calibrated the stereo pair using a planar checker board pattern outlined by Zhang.[20] We computed the calibration parameters and saved them to the tablet hard drive as a configuration file. Once the user starts the application, it automatically loads the calibration parameters to memory for real-time rectification. This reduces distortion caused by the optical lens and improves stereo-matching results. Even though we obtained rectified image pairs, we still need to synchronize the sensors since we cannot rectify over time for dynamic scenes. The main mechanism for synchronizing multiple sensors in FCam API is by extending the basic *sensor* component to a *sensor array*.[19] A new abstract class called *SynchronizationObject* is also derived from the *Device* class with members *release* and *wait* for software synchronization. When the request queue for the camera sensors is being processed, if a *wait* is found and a certain condition is not satisfied, the *sensor* will halt until this condition is satisfied. On the other hand, if a *release* is found and the condition is satisfied, the halted *sensor* will be allowed to proceed. The FCam API also provides classes such as *Fence*, *MultiSensor*, *MultiShot*, *MultiImage*, and *MultiFrame* for the user to control the stereo sensor with desired request parameters.

In our application, we use the rear main camera to continuously detect the best focusing position and to send updates to the other sensor. First, we ask the rear main lens to start sweeping the lens. We then get each frame with its focusing location. Next, we sum up all the values of the sharpness map attached to the frame and send updates to the autofocus function. The autofocus routine will move the lens in a relatively slower speed to refine the best focal depth. Once this process is done, we trigger a stereo capture with identical aperture, exposure, and gain parameters for both sensors. The overall image quality is satisfactory, considering the fact that the size of the sensor is very small and the cost is much lower than research stereo camera systems such as Pointgreys Bumblebee. Figure 2 shows how our software system interacts with the imaging hardware.

5 Disparity Map Generation

Computing depth information from stereo camera systems is one of the core problems in computer vision. Stereo algorithms based on local correspondences[21,22] are usually fast but introduces inaccurate boundaries or even bleeding artifacts. Global stereo estimation methods, such as GCs[15]

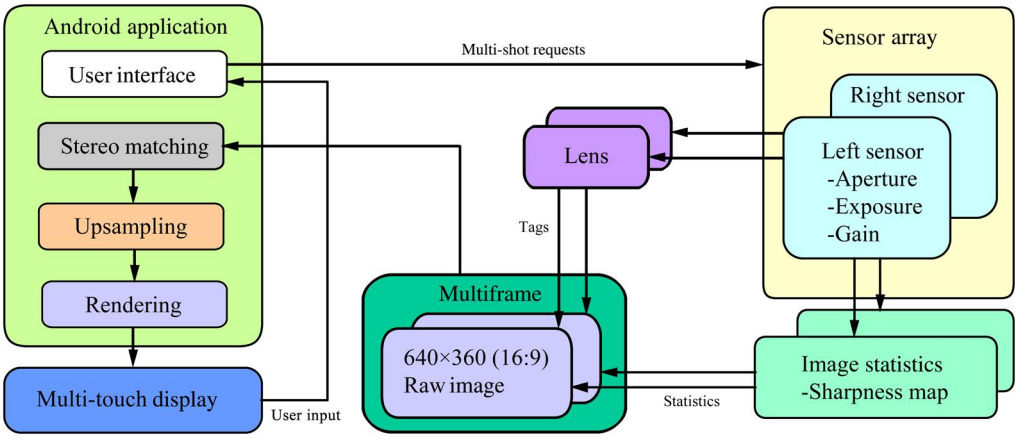

Fig. 2 This diagram shows our system architecture. Our application accepts user input from the multi-touch screen, sends multishot requests to the sensors with desired parameters, and then transfers the raw stereo image pairs to the stereo-matching module. We then upsample the low-resolution disparity map and synthesize a light field image array. Finally, we render DoF effects on the screen of the tablet. We compute the best focal plane by using image statistics information tagged with the raw image frame.

and BP,[16] have shown good results on complex scenes with occlusions, textureless regions, and large depth changes.[18] However, running these algorithms on full-resolution (1 MP) image pairs is still expensive and hence impractical for mobile devices. Therefore, we first downsample the raw input image pair and recover a low-resolution disparity map via GCs. Next, we take each raw color image as the guidance image and upsample the disparity map via joint bilateral upsampling.[23]

5.1 *GCs Stereo Matching*

In order to efficiently generate a high-resolution disparity map with detailed information about the scene, we propose a two-step approach. We first recover a low-resolution disparity map on downsampled image pairs with the size of 160×90. Given the low-resolution image pairs, the goal is to find labeling of pixels indicating their disparities. Suppose $f(p)$ is the label of pixel p; $D_p(x)$ is the data term, reflecting how well pixel p fits its counterpart pixel p shifted by x in the other image; $V_{p,q}(y,z)$ is the smoothness term indicating the penalty of assigning disparity y to pixel p and disparity z to pixel q; and N is the set of neighboring pixels, the correspondence problem can be formulated as minimizing the following energy function:

$$E(f) = \arg\min_f \left\{ \sum_{p \in P} D_p[f(p)] + \sum_{\{p,q\} \in N} V_{p,q}[f(p), f(q)] \right\}. \tag{1}$$

The local minimization of Eq. (1) can be efficiently approximated using the alpha expansion presented in Ref. 15. In our implementation, we set the number of disparities to be 16 and run the algorithm for five iterations. If the algorithm cannot find a valid alpha expansion that decreases the overall energy function value, then it may also terminate in less than five iterations. The performance of GCs on the Tegra 3 tablet platform can be found in Table 1.

To evaluate our scheme, we performed experiments on various stereo image datasets. The stereo-matching methods used here are block matching (BM), semi-global BM (SGBM),[21] efficient large-scale stereo (ELAS),[22] and GCs.[15] Table 1 shows the running time of these algorithms on the Tegra 3 tablet, and Fig. 3 shows the calculated disparity map results. According to our experiments, BM is faster than SGBM and ELAS on any given dataset but requires an adequate choice of the window size. Smaller window sizes may lead to a larger bad pixel percentage, whereas larger window sizes may cause inaccuracy problems on the boundary. Besides, the overall accuracy of disparity values generated by BM is not very high. As can be seen from Fig. 3, we can still identify the curved surface area of the cones from the results generated by SGBM and ELAS, but the same area looks almost flat in BM. SGBM and ELAS are the two very popular stereo-matching algorithms with near real-time performance. According to our experiments on the tablet, they are very similar to each other in terms of running time and accuracy. From Table 1 and Fig. 3, we can see that ELAS generates better boundaries than SGBM on the cones dataset, but takes more processing time and produces more border bleeding artifacts. The GCs gives smooth transitions between regions of different disparity values. According to Table 2, the GCs algorithm outperforms other algorithms in most of the quality metrics on the Middlebury datasets. For our application, since the quality of upsampled result is highly dependent on the precision of boundary values in low-resolution disparity maps, we choose to use GCs rather

Table 1 Comparing running time (ms) of different stereo-matching methods on the Tegra 3 tablet, using the Middlebury Cones dataset. The longer edge is set to 160 pixels, and the number of disparities is set to 16.

Datasets	BM	SGBM	ELAS	GCs
Tsukuba	15	28	51	189
Venus	13	30	97	234
Cones	19	42	124	321

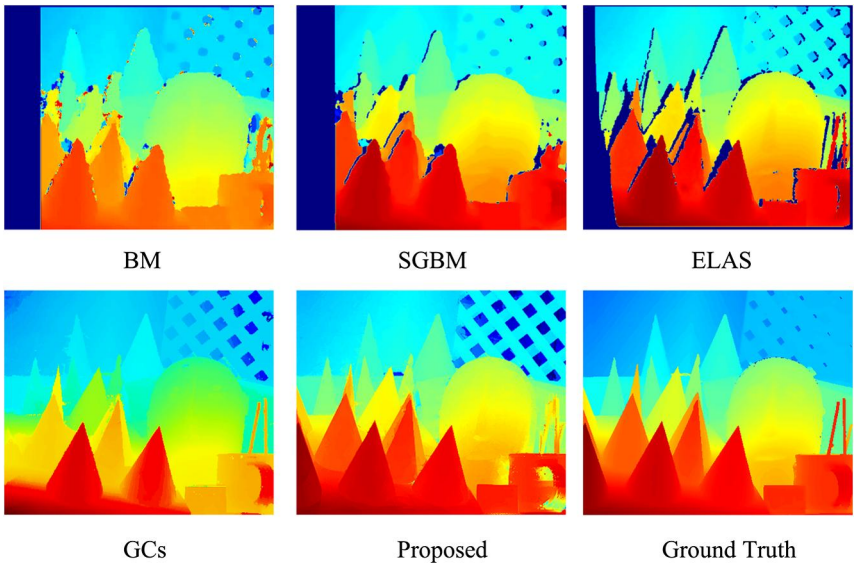

Fig. 3 Comparison of our approach and other popular stereo-matching algorithms.

than other methods which runs faster. Another reason is that we are running the GCs algorithm on low-resolution imagery. According to Table 1, the running time is around 250 ms, which is still acceptable compared with ELAS (around 100 ms). In return, noisy and invalid object boundaries are well optimized and the resulting disparity map is ideal for refinement filters such as joint bilateral upsampling.

5.2 *Joint Bilateral Upsampling*

Because the stereo-matching process is performed on low-resolution stereo image pairs, the resulting disparity map cannot be directly used for DoF synthesis. We need to upsample the disparity map while keeping important edge information.

Bilateral filtering proposed by Tomasi and Manduchi[24] is a simple, noniterative scheme for edge preserving smoothing, which uses both a spatial kernel and a range kernel. However, for low signal-to-noise ratio images, this algorithm cannot keep the edge information very well. A variant called joint bilateral filter introduced by Kopf et al.[23] addresses this problem by adding the original RGB image as a guidance image. More formally, let p and q be two pixels on the full-resolution color image I; $p_\downarrow$ and $q_\downarrow$ denote the corresponding coordinates in the low-resolution disparity map D'; f is the spatial filter kernel, g is the range filter kernel, W is the spatial support of kernel f, and K_p is the normalizing factor. The upsampled solution D_p can be obtained as

$$D_p = \frac{1}{K_p} \sum_{q_\downarrow \in W} D'_{q\downarrow} f(\|p_\downarrow - q_\downarrow\|) g(\|I_p - I_q\|). \tag{2}$$

This method uses RGB values from the color image to create the range filter kernel and combines high-frequency components from the color image and low-frequency components from the disparity map. As a result, color edges are integrated with depth edges in the final upsampled disparity map. Since depth discontinuities typically correspond with color edges, this method can remove small noises. However, it may bring some unwanted effects. First, blurring

Table 2 Evaluation of different stereo-matching methods on the Middlebury stereo datasets cite in bad pixel percentage (%). The method shown in the last row applies five iterations of joint bilateral upsampling to the downsampled results (half of the original size) of GCs, using the full-resolution color image as the guidance image. The resolutions of the four datasets (Tsukuba, Venus, Teddy, and Cones) are 384×288, 434×383, 450×375, and 450×375, respectively. If not specified, raw image size of each individual dataset will be the same for the remainder of this article.

	Tsukuba			Venus			Teddy			Cones		
	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
BM	10.3	11.9	21.5	12.4	13.9	21.6	16.7	23.1	27.3	7.46	17.2	23.8
SGBM	3.26	3.96	12.8	1.00	1.57	11.3	6.02	12.2	16.3	3.06	9.75	8.90
ELAS	3.96	5.42	17.9	1.82	2.78	20.9	7.92	14.5	22.8	6.81	14.9	17.2
GCs	1.94	4.12	9.39	1.79	3.44	8.75	16.5	25.0	24.9	7.70	18.2	15.3
Proposed	1.01	2.83	5.42	0.18	0.59	1.99	6.57	11.2	15.1	3.06	9.70	8.92

Note: nonocc, bad pixel percentage in nonoccluded regions; all, bad pixel percentage in all regions; disc, bad pixel percentage in regions near-depth discontinuities.
xxx

and aliasing effects caused by the optical lens are transferred to the disparity map. Besides, the filtering process may change disparity values in occlusion boundaries, according to the high-frequency components in the color image, and thus causing the disparity map to be inaccurate. We address this problem by iteratively refining the disparity map after the upsampling process is done. As a result, the output image of the previous stage becomes the input of the next stage.

Figure 4 shows the results after different numbers of iterations. The initial disparity map [see Fig. 4(a)] is noisy and inaccurate, because it is generated on low-resolution image pairs. However, if too many iterations are applied to the input image [Fig. 4(d)], the boundaries of the cup handle start to bleed into the background, which is a result of over-smoothing. Also, more iterations add to the complexity and processing overhead of the entire application. According to Fig. 5, the quality of the disparity map can be improved during the first five or six iterations. This is because joint bilateral upsampling can preserve edges while removing noises in the disparity map. However, if the refining process contains too many iterations, then the disparities of one side of edges start to bleed into the other side, causing the bad pixel percentage to go up, especially in regions near depth discontinuities (refer to the increase of disc values in Fig. 5). Therefore, a compromise number of iterations must be chosen. In our application, the number is set to 5. Since the Middlebury datasets contain both simple scenes like Venus and complex scenes such as Teddy and Cones, we assume that five iterations should return good results under a variety of scene structures. Generally, it takes around 40 ms to finish the five iteration steps on the tablet. Figure 6 illustrates the detailed view of our result compared with other standard upsampling methods. Because DoF effects are most apparent around the depth edges, it is very important to recover detailed boundaries in the high-resolution disparity map. According to Table 3, our method outperforms other methods in all quality metrics and generates better boundary regions (refer to disc values in Table 3) by using the fine details from the high-resolution color image.

6 DoF Rendering

Once we obtained the high-resolution disparity map, we can proceed to synthesize dynamic DoF effects. Previous studies suggested that the real-time DoF effects can be obtained by applying a spatially varying blur on the color image and using the disparity value to determine the size of the blur kernel.[25,26] However, this method suffers from strong intensity leakage and boundary bleeding artifacts. Other methods such as distributed ray tracing[27] and accumulation buffer[28] give more accurate results. However, these methods are computationally expensive and therefore can only provide a limited frame rate.

6.1 *Synthesized Light Field Generation*

In this article, we use a similar approach to Ref. 29 by generating a synthetic light field on the fly. The main idea is to get the light field image array by warping the raw color image to nearby viewpoints, according to the corresponding values in the upsampled high-resolution disparity map. The light field array can then be used to represent rays in the scene. Each ray in the light field can be indexed by an integer 4-tuple (s, t, u, v), where (s, t) is the image index and (u, v) is the pixel index within an image. Next, we set the rear main camera as the reference camera and use the high-resolution color image and disparity map for reference view R_{00}. We then compute all rays passing through a spatial point X with shifted disparity γ from the reference view. Suppose X is projected to pixel (u_0, v_0) in the reference camera, we can compute its image pixel coordinate in any other light field camera view R_{st} as

$$(u, v) = (u_0, v_0) + (s, t) \cdot \gamma. \tag{3}$$

However, this algorithm may introduce holes in warped views, and this artifact becomes more severe when the synthesized baseline increases. To resolve this issue, we start from the boundary of the holes and iteratively take nearby pixels to fill the holes. Note that this module is only used for generating pleasing individual views for the user to interactively shift the perspective. In the final rendering process, missing rays are simply discarded and the filled pixels are not used. Figure 7 shows the warped views of an indoor scene using the aforementioned warping and hole-filling algorithms.

Since the image formed by a thin lens is proportional to the irradiance at pixel a,[30] if we use $L_{\text{out}}(s, t, u, v)$ to represent the out-of-lens light field and $L_{\text{in}}(s, t, u, v)$ to represent the in-lens light field, the pixels in this image can be obtained as a weighted integral of all incoming radiances through the lens

$$a(x, y) \simeq \sum_{(s,t)} L_{\text{in}}(s, t, u, v) \cdot \cos^4\phi. \tag{4}$$

To compute the out-of-lens light field, we simply remap the pixel $a(x, y)$ to pixel $(u_0, v_0) = (w - x, h - y)$ in the reference view R_{00}. Therefore, we can focus at any scene

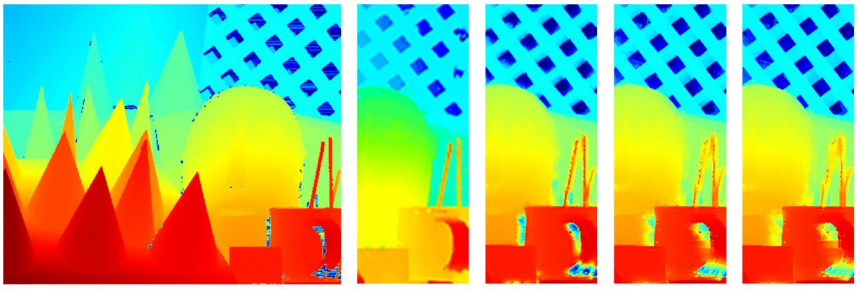

Fig. 4 Comparison of results using different numbers of iterations. Panels (a), (b), (c), (d) are obtained using 0, 5, 10, 20 iterations, respectively.

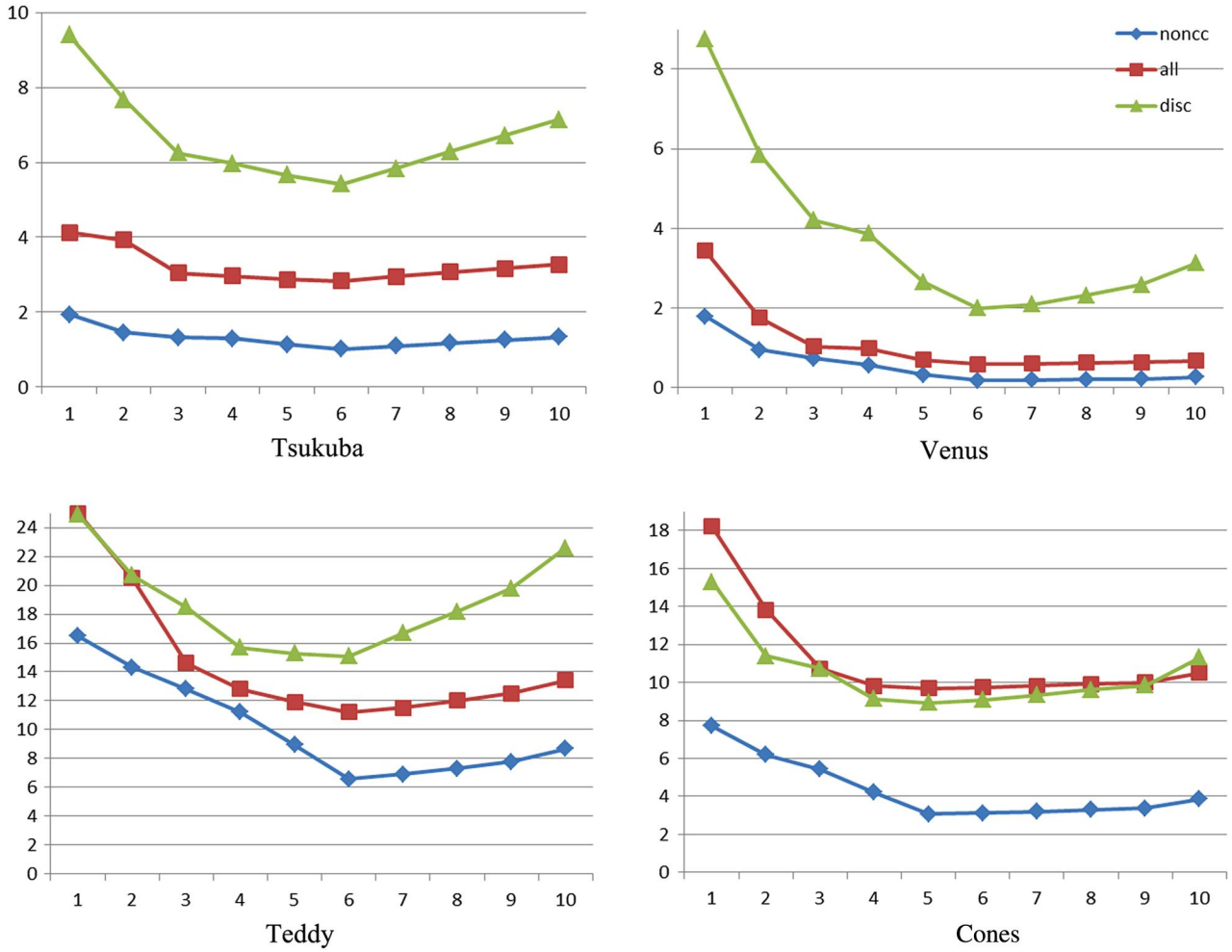

Fig. 5 Evaluation of the disparity maps using different numbers of joint bilateral upsampling iterations on the Middlebury stereo dataset. The horizontal axis shows the number of iterations and the vertical axis shows the bad pixel percentage.

depth with corresponding disparity γ_f by finding the pixel index in camera R_{st} using Eq. (3). Since the direction for each ray is $(s,t,1)$, we can approximate the attenuation term $\cos^4\phi$ as $\frac{1}{(s^2+t^2+1)^2}$, and the irradiance at a can be calculated as

$$a(x,y) \simeq \sum_{(s,t)} \frac{L_{\text{out}}(s,t,u_0+s\cdot\gamma_f, v_0+t\cdot\gamma_f)}{(s^2+t^2+1)^2}. \quad (5)$$

Figure 8 shows the details of the rendered image by using different sizes of the synthesized light field array. Since aliasing artifacts are related to scene depth and sampling frequency,[31] we can reduce aliasing in the rendered image by increasing the size of the synthesized light field array.

6.2 *Comparison of Our Method and Single-Image Blurring*

Reducing boundary artifacts is very important as DoF effects are apparent near the occlusion boundaries. Comparing with single-image blurring methods,[25,26] our light field–based analysis is good at reducing two types of boundary artifacts: the boundary discontinuity and intensity leakage artifacts. We summarize four types of boundary artifacts and analyze them separately. A detailed illustration of the four cases can

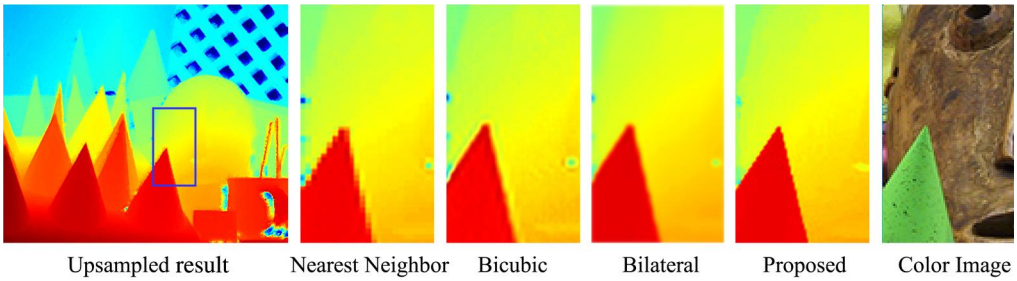

Fig. 6 Comparison of our approach and other upsampling algorithms on the Middlebury cones dataset.

Table 3 Evaluation of various upsampling methods on the Middlebury stereo datasets in bad pixel percentage (%). We run these methods on downsampled ground truth data (half of the original size), and then try to recover the disparity maps at original size and measure the error percentage.

	Tsukuba			Venus			Teddy			Cones		
	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
Nearest neighbor	5.55	6.65	18.3	0.47	1.02	6.56	8.65	9.77	28.2	7.98	9.62	23.7
Bicubic	4.97	5.69	18.7	0.67	0.93	9.32	4.89	5.61	17.8	6.81	7.59	20.6
Bilateral	4.59	5.04	10.8	0.41	0.60	5.75	4.52	5.12	16.3	6.85	8.41	20.5
Proposed	3.08	3.34	7.54	0.25	0.33	3.47	2.41	2.89	8.76	3.45	3.96	10.5

Note: nonocc, bad pixel percentage in nonoccluded regions; all, bad pixel percentage in all regions; disc, bad pixel percentage in regions near-depth discontinuities.

be found at Fig. 9. In practice, the four cases can occur at the same time within a single scene.

Our analysis is based on the real-world scene shown in Fig. 9. Consider a woman in a black dress walking in front of a white building. When we conduct the DoF analysis, the camera is either focused at the foreground (the woman) or at the background (the building). For Figs. 9(a) and 9(b), we assume that the camera to be focused at the background, and for Figs. 9(c) and 9(d), we assume that the camera is focused at the foreground. For each case, a comparison of results using different methods is shown at the right side of the images.

Now consider the first two cases shown in Figs. 9(a) and 9(b). Suppose P_b is a point on the background building and

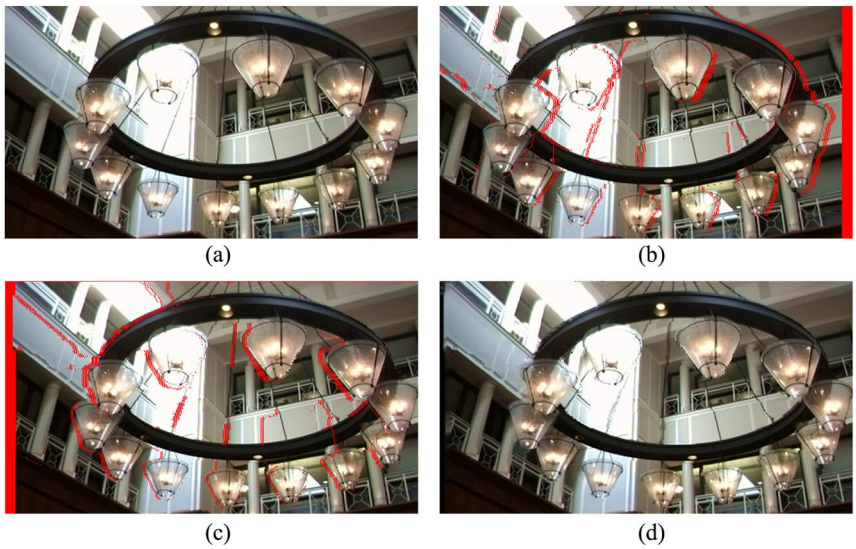

Fig. 7 Synthesized light field view, missing pixels are marked in red. (a) Input image, (b) warped left side view, (c) warped right side view, and (d) resulting image using our hole-filling algorithm, taking (c) as the input.

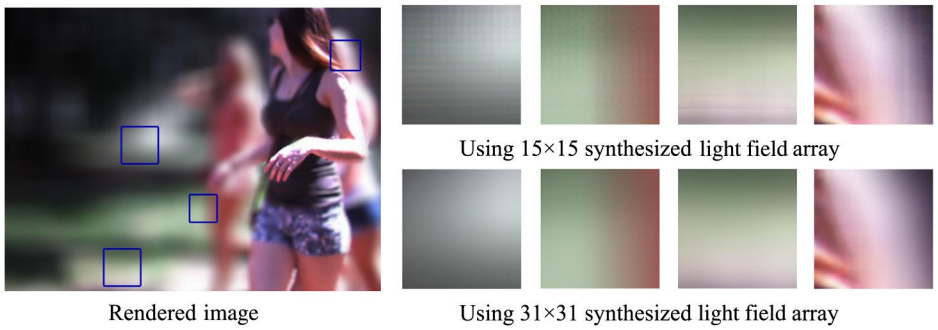

Fig. 8 Comparing rendering results with different sizes of the synthesized light field array.

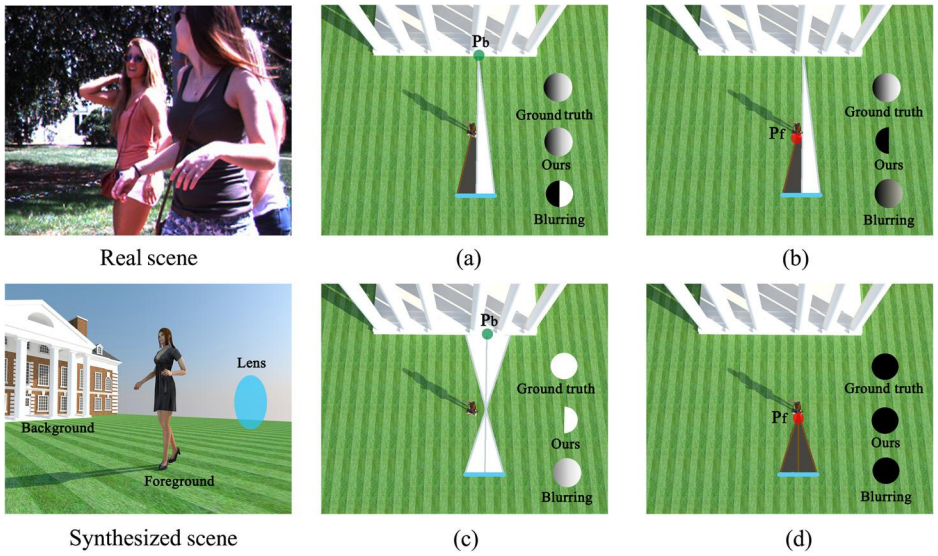

Fig. 9 Causes of different boundary artifacts (see Sec. 6.2 for details). In (a) and (b) the camera is focused at the background. In (c) and (d), the camera is focused at the foreground.

its image I_b in the camera is right next to the foreground as shown in Fig. 9(a). The ground truth result should blend both foreground and background points for calculating I_b to make the transition natural and smooth. However, single-image blurring methods would consider P_b in focus and directly use its color as the value of I_b. This will result in a boundary discontinuity artifact because of the abrupt jump between foreground and background pixel values. Our method, however, takes advantage of the synthesized light field, attempts to use rays originating from both foreground and background to calculate the pixel value of I_b, and hence generates correct results for this scenario. Similarly, for a foreground point P_f shown in Fig. 9(b), the ground truth result should blend its neighboring foreground pixels and a single in-focus background point. The single-image blurring methods will use a large kernel to blend a group of foreground and background pixels, producing the intensity leakage artifact. In contrast, our method only takes rays needed to get the value of P_f and is free of intensity leakage artifacts. However, due to occlusion, some background pixels may be missing. In this case, our method will blend foreground rays and accessible background rays together. Since the missing rays only occupy a small portion of all background rays, our method produces reasonable approximations.

For the other two cases [Figs. 9(c) and 9(d)], assume that the camera is focused at the foreground. As shown in Fig. 9(c), the ground truth result should only blend background pixels. However, because of the blur kernel, the single-image blurring method blends both foreground and background pixels and thus causing intensity leakage problems. Our method, on the other hand, only attempts to blend background rays. Similar to the previous case, some rays are occluded by the foreground. We simply discard these rays and by blending existing rays together, we are able to reach

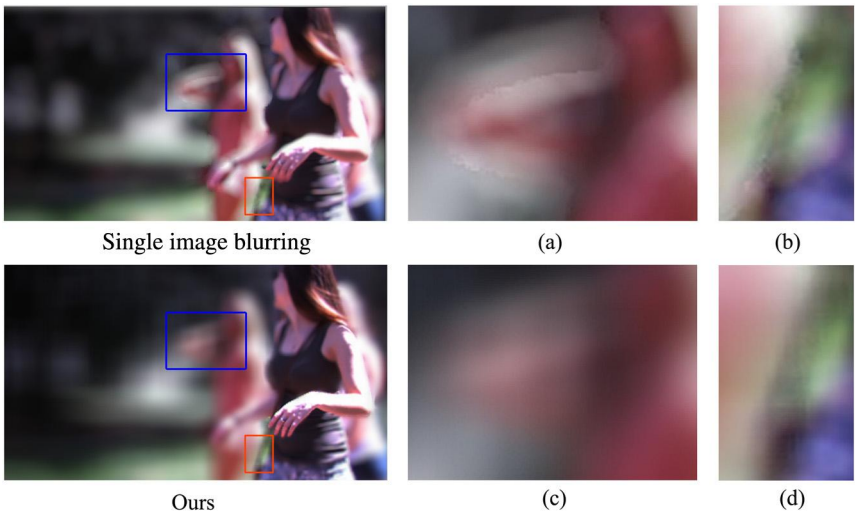

Fig. 10 Comparison between our method and single-image blurring. Single-image blurring methods suffer from intensity leakage (a) and boundary discontinuity (b) artifacts. Our method (c and d) reduces these artifacts.

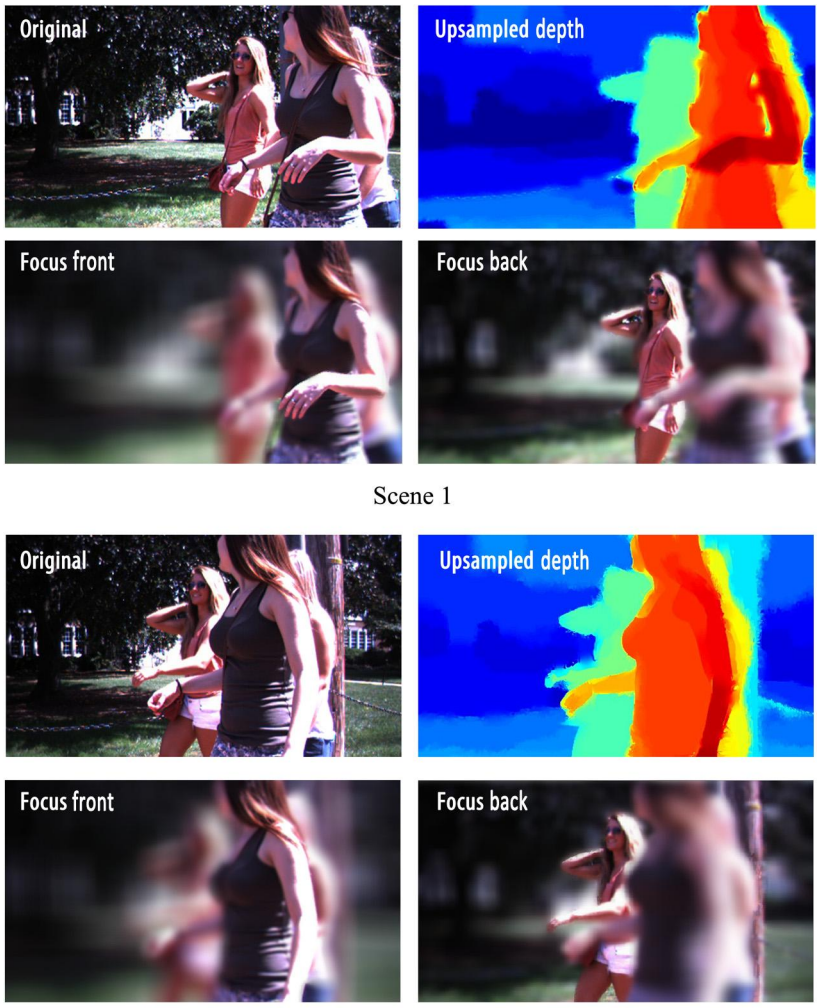

Fig. 11 Input disparity map and rendered images of our system on two frames from the same stereo video sequence.

reasonable approximations of the ground truth. For the last case, consider a point P_f on the foreground, as shown in Fig. 9(d). Since this pixel is considered to be in focus, the single-image blurring method will directly use its color and produces the correct result. Our method collects all rays coming from P_f, and these rays are all accessible. Therefore, our method is also able to get the correct result.

Figure 10 shows the results of our method and single-image blurring on an outdoor scene. As mentioned before, our method reduces artifacts on boundary regions compared with single-image blurring approaches. In fact, our method will not cause any intensity leakage problems. When examining the single-image blurring result [Fig. 10(a)], it is very easy to find intensity leakage artifacts along the boundary, whereas our technique prevents such leakage [Fig. 10(c)]. Also, our method provides smooth transitions from the handbag strips to the background [Fig. 10(d)], whereas single-image blurring method exhibits multiple discontinuous jumps in intensity values.

7 Results and Analysis

We conducted extensive experiments on both indoor and outdoor scenes. Figures 11 and 12 show the results generated by our system under different scene structures and illumination conditions. Scenes 1 and 2 demonstrate our system's ability of handling real-time dynamic scenes; Scene 3 shows the result on an outdoor scene with strong illumination and shadows; Scene 4 displays the result on an indoor scene with transparent and textureless regions.

The processing speed of different frames varies from less than a second to several hundred seconds depending on the parameters such as number of stereo-matching iterations, number of bilateral upsampling iterations, and the size of the synthesized light field array. The user can keep taking pictures while the processing takes place in the background. Considering the performance of current mobile device processors, rendering real-time DoF effects on HD video streams is still not practical. However, this does not prevent users from taking consecutive video frames and rendering them offline, as can be seen in scenes 1 and 2 of Fig. 11. Also, since in general the stereo cameras on mobile devices have a small baseline, the disparity values of pixels in the downsampled images have certain max/min thresholds. We can reduce the number of disparity labels in the GCs algorithm and further improve the processing speed without introducing much performance penalty.

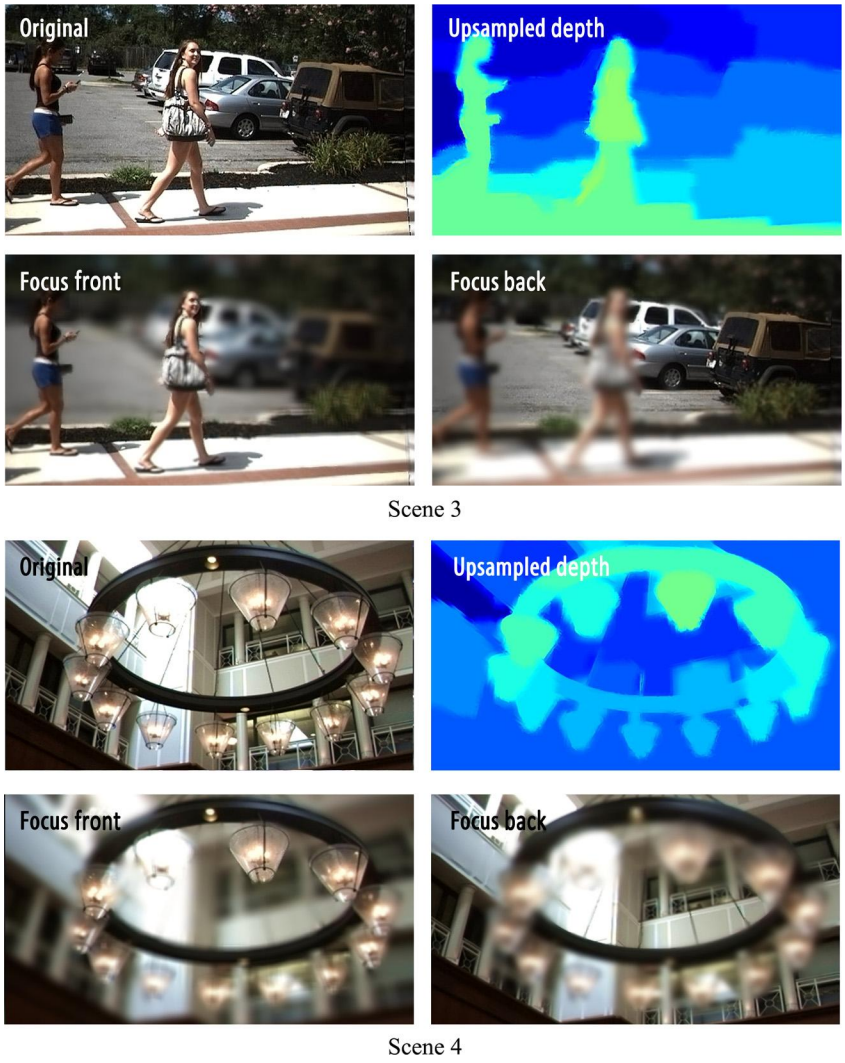

Fig. 12 Input disparity map and rendered images of our system on two real scenes with the same arrangement as in Fig. 11.

We first demonstrate our system in dynamic outdoor scenes. Figure 11 shows the results of two frames from the same video sequence. Since we currently do not have any auto-exposure or high-dynamic range (HDR) modules implemented, some parts of the photo are over-exposed. As shown in the photograph, many texture details are lost in the over-exposed regions, making it challenging for the stereo-matching algorithm to recover accurate disparity values. Moreover, the background lawn contains noticeable shadows and large portions of the building wall are textureless. This adds to the difficulty of finding pixel to pixel correspondences. Notwithstanding, our algorithm generates visually good-looking disparity maps. The edges of the woman's hand and arm are preserved when they are in focus, and objects outside of the focal plane are blurred smoothly.

Scene 3 of Fig. 12 displays a scene of two women walking in front of a parking lot. Typically the working range of the tablet sensor is from half a meter to 5 m. As a result, the cars in the parking lot are already approaching the maximum working distance of the sensor. This, however, does not affect the overall refocusing result as the cars with similar disparity values are either all in focus [Fig. 12, row 2, column 2] or blurred [Fig. 12, row 2, column 1]. The sidewalk in front of the parking lot has a lot of textureless areas, making it difficult to achieve coherent disparity values. As a result, the left and right parts of the sidewalk are blurred slightly differently although they are on the same plane [Fig. 12, row 2, column 2]. Also, because the women in scene 3 are farther away from the camera compared with the women in scenes 1 and 2, the boundaries of women in scene 3 are coarser and fine details on the bodies are lost. Therefore, foregrounds in scene 3 are more uniformly blurred compared with scenes 1 and 2.

Indoor scenes have controllable environments and undoubtedly aid the performance of our system. For example, most structures from an indoor scene are within the working range of our system and typically indoor lighting would not cause problems such as over-exposure or shadows. Scene 4

Table 4 Results of subjective quality rating tests.

User ♯	1	2	3	4	5	6	7	8	9	10	Average
Nonexperts	7	9	9	8	9	8	7	7	9	8	8.1
Experts	5	3	7	6	8	7	1	5	5	6	5.3

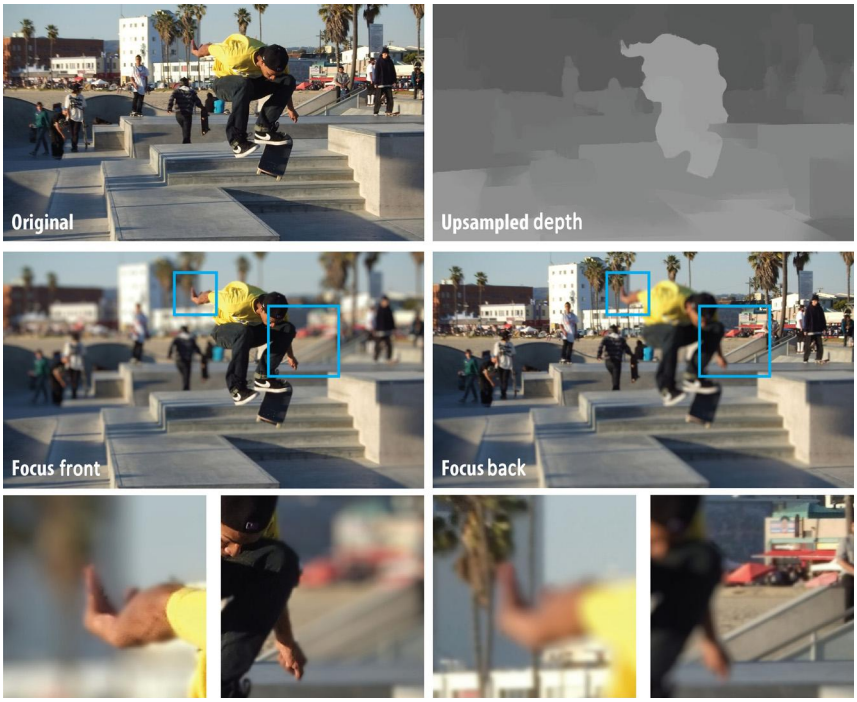

Fig. 13 Our result on a skateboard scene at 6 MP captured by Fujifilm FinePix Real 3-D camera (courtesy of Design-Design).[32]

of Fig. 12 shows the results on an indoor scene with transparent objects and textureless regions. Since our algorithm effectively fills holes and corrects bad pixels on the disparity map by using the guide color image, the resulting disparity map looks clean and disparity edges of the chandelier are well preserved [Fig. 12, row 3, column 2]. The upper left part of the wall surface is over-exposed and the light bulb in the foreground almost merged into the background. However, the disparity map still recovers edges correctly. As can be seen in Fig. 12, row 4, column 2, the defocus blur fades correctly from the out-of-focus light bulb regions into the in-focus wall regions, despite the fact that they are both white and do not have clear boundaries in between.

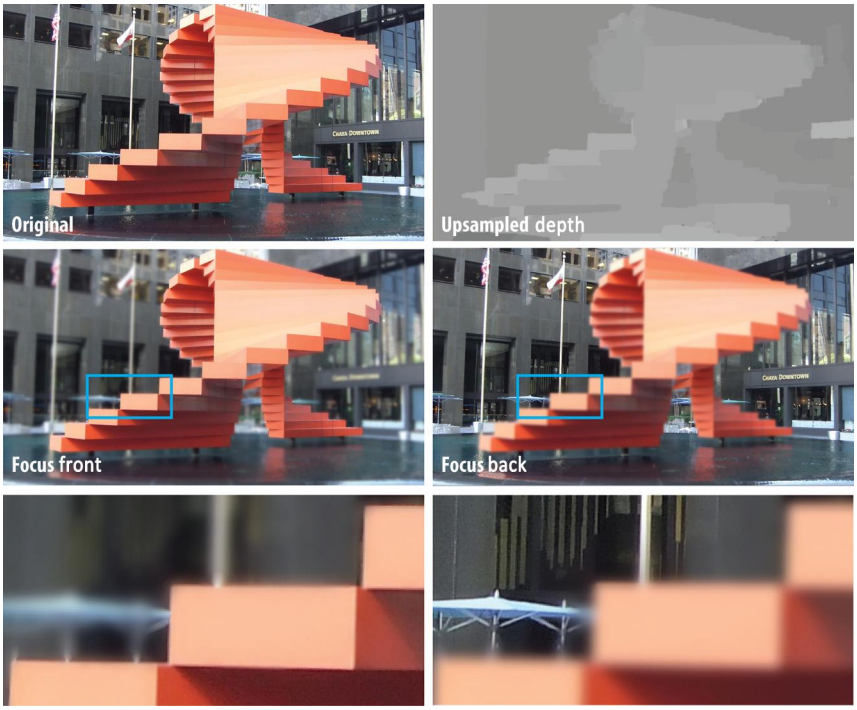

Fig. 14 Our result on a sculpture scene at 6 MP captured by Fujifilm FinePix Real 3-D camera (courtesy of Design-Design).[32]

The discussion here is based on our own captured data, and it is hard to evaluate rendered results because of the lack of ground truth. To address this problem, we conducted subjective rating tests with 20 people. Among these people, 10 have a computer vision or graphics background and the remaining have no expertize in the related field. For convenience and clarity, the rating is done on a 0 to 9 scale for measuring the quality of rendered results. We define the rating as follows: 0 (not acceptable), 1 (acceptable), 3 (good, but needs improvement), 5 (satisfactory), 7 (very good), and 9 (excellent). The test results can be found in Table 4. The average rating of the nonexpert group is 8.1, and the average rating from the experts is 5.3. Therefore, the overall quality of the rendered results can be concluded as satisfactory.

According to Table 2, our method returns the best disparity map results in terms of overall bad pixels percentage. Also, our system correctly handles complex scene structures with real-world illumination conditions. Last but not least, according to the resulting images in Fig. 8, we reduce aliasing artifacts in out-of-focus regions by blending multiple synthesized light field views together.

Finally, to demonstrate that our algorithm is also capable of generating high-quality DoF effects using high-resolution stereo input, we leverage mobile devices Fujifilm FinePix Real 3-D camera to capture a set of stereo images and to generate the shallow DoF images with refocus capabilities at 6-MP resolution, as shown in Figs. 13 and 14. Current light field cameras are not capable of generating such high-resolution images. Figure 13 shows the scene of a person playing with a skateboard. Our algorithm is able to preserve most of the depth discontinuities in the scene such as the edges of the hand, the skateboard, and the leg. Note that the background between the legs is marked as the foreground, leaving artifacts in the final rendering. This is due to the unsuccessful depth estimation of the GCs algorithm, and our current depth upsampling is largely relying on the initial estimation. In the future, we plan to employ the depth error correction into our upsampling scheme. Figure 14 shows a scene of a sculpture in a shopping mall. Despite the complex occlusion conditions in the scene, our algorithm is still able to synthesize shallow DoF effects with little artifacts such as fussy edges on the stairs.

8 Conclusion

We have presented an affordable solution for producing dynamic DoF effects on mobile devices. The whole system runs on an off-the-shelf tablet, which costs less than $400. We compare the performance of popular stereo-matching algorithms and design a hybrid resolution approach, which tries to improve both speed and accuracy. Also, we generate the synthesized light field by using a disparity warping scheme and render the high-quality DoF effects. Finally, we map all processing stages onto the Android system and control the computational imaging device by using the FCam architecture. Our system efficiently renders dynamic DoF effects with arbitrary aperture sizes and focal lengths in a variety of indoor and outdoor scenes.

Our future efforts include adding modules such as auto-exposure or HDR to improve the imaging quality. We would also like to explore the possibility of implementing our approach to sparse camera arrays with limited number of views.

Acknowledgments

We thank NVIDIA Corporation for providing the Tegra prototype devices. This project is supported by the National Science Foundation under grant IIS-1319598.

References

1. R. Ng et al., "Light field photography with a hand-held plenoptic camera," *Comput. Sci. Tech. Rep.* **2**(11), 1–11 (2005).
2. T. Georgiev et al., "Spatio-angular resolution tradeoffs in integral photography," in *Proc. of the 17th Eurographics Conf. on Rendering Techniques, EGSR'06*, Nicosia, Cyprus, pp. 263–272, Eurographics Association, Aire-la-Ville, Switzerland (2006).
3. T. Georgiev and A. Lumsdaine, "Focused plenoptic camera and rendering," *J. Electron. Imaging* **19**(2), 021106 (2010).
4. V. Kolmogorov and R. Zabih, "Computing visual correspondence with occlusions using graph cuts," in *Proc. Eighth IEEE Int. Conf. on Computer Vision, 2001. ICCV 2001*, Vol. 2, pp. 508–515, IEEE (2001).
5. A. Adams et al., "The frankencamera: an experimental platform for computational photography," *ACM Trans. Graph.* **29**(4), 1–12 (2010).
6. G. Lippmann, "La photographie intgrale," *Comp. Rend. Acad. Sci.* **146**, 446–451 (1908).
7. B. Wilburn et al., "High performance imaging using large camera arrays," *ACM Trans. Graph.* **24**(3), 765–776 (2005).
8. J. C. Yang et al., "A real-time distributed light field camera," in *Proc. 13th Eurographics Workshop on Rendering*, pp. 77–86, Eurographics Association, Aire-la-Ville, Switzerland (2002).
9. A. Veeraraghavan et al., "Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing," *ACM Trans. Graph.* **26**(3), 69 (2007).
10. Lytro, "Lytro Camera," https://www.lytro.com (March 2014).
11. T. Georgiev et al., "Lytro camera technology: theory, algorithms, performance analysis," *Proc. SPIE* **8667**, 86671J (2013).
12. PelicanImaging, "Pelican Imaging Array Camera," http://www.pelicanimaging.com (March 2014).
13. Z. Yu et al., "Racking focus and tracking focus on live video streams: a stereo solution," *Visual Comput.* **30**(1), 45–58 (2013).
14. Z. Yu et al., "Dynamic depth of field on live video streams: a stereo solution," in *Proc. of the 2011 Computer Graphics Int. Conf., CGI 2011*, Ottawa, Ontario, Canada, ACM (2011).
15. Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(11), 1222–1239 (2001).
16. J. Sun, N.-N. Zheng, and H.-Y. Shum, "Stereo matching using belief propagation," *IEEE Trans. Pattern Anal. Mach. Intell.* **25**(7), 787–800 (2003).
17. M. F. Tappen and W. T. Freeman, "Comparison of graph cuts with belief propagation for stereo, using identical MRF parameters," in *Proc. Ninth IEEE Int. Conf. Computer Vision*, pp. 900–906, IEEE (2003).
18. D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *Int. J. Comput. Vision* **47**(1–3), 7–42 (2002).
19. A. Troccoli, D. Pajak, and K. Pulli, "Fcam for multiple cameras," *Proc. SPIE* **8304**, 830404 (2012).
20. Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(11), 1330–1334 (2000).
21. H. Hirschmuller, "Accurate and efficient stereo processing by semi-global matching and mutual information," in *IEEE Computer Society Conf. Computer Vision and Pattern Recognition, 2005. CVPR 2005*, Vol. 2, pp. 807–814, IEEE (2005).
22. A. Geiger, M. Roser, and R. Urtasun, "Efficient large-scale stereo matching," in *Proc. of the 10th Asian Conf. on Computer Vision—Volume Part I, ACCV'10*, Queenstown, New Zealand, pp. 25–38, Springer-Verlag, Berlin, Heidelberg (2011).
23. J. Kopf et al., "Joint bilateral upsampling," *ACM Trans. Graph.* **26**(3), 96 (2007).
24. C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Sixth Int. Conf. Computer Vision, 1998*, pp. 839–846, IEEE (1998).
25. S. Lee, E. Eisemann, and H.-P. Seidel, "Depth-of-field rendering with multiview synthesis," *ACM Trans. Graph.* **28**(5), 1–6 (2009).
26. S. Lee, E. Eisemann, and H.-P. Seidel, "Real-time lens blur effects and focus control," *ACM Trans. Graph.* **29**(4), 1–7 (2010).
27. R. L. Cook, T. Porter, and L. Carpenter, "Distributed ray tracing," *ACM SIGGRAPH Comput. Graph.* **18**(3), 137–145 (1984).
28. P. Haeberli and K. Akeley, "The accumulation buffer: hardware support for high-quality rendering," *ACM SIGGRAPH Comput. Graph.* **24**(4), 309–318 (1990).
29. X. Yu, R. Wang, and J. Yu, "Real-time depth of field rendering via dynamic light field generation and filtering," *Comput. Graph. Forum* **29**(7), 2099–2107 (2010).
30. L. Stroebel et al., *Photographic Materials and Processes*, Focal Press, Boston/London (1986).

31. J.-X. Chai et al., "Plenoptic sampling," in *Proc. 27th Annual Conf. Computer Graphics and Interactive Techniques*, pp. 307–318, ACM Press/Addison-Wesley Publishing Co., New York, NY (2000).
32. P. Simcoe, "Design–Design Sample 3D Images," http://www.design-design.co.uk/sample-mpo-3d-images-for-television-display (March 2014).

Qiaosong Wang received his BEng degree from the Department of Automation Science and Technology, Xi'an Jiaotong University, in 2011. He is now a PhD student at the Department of Computer and Information Sciences, University of Delaware. His research interests include computer vision and mobile robotic perception.

Zhan Yu has been a research scientist at Adobe Systems Inc. since December 2013. Before that, he received his PhD degree in computer science from the University of Delaware and a BS degree in software engineering from Xiamen University. His research interests include computational photography, computer graphics, and computer vision.

Christopher Rasmussen is an associate professor in the Department of Computer & Information Sciences at the University of Delaware. He received his PhD degree in computer science from Yale University in 2000 and an AB degree from Harvard University in 1993. His research interests are in mobile robot perception and field robotics. He is the recipient of an NSF CAREER Award.

Jingyi Yu is an associate professor in the Department of Computer & Information Sciences and the Department of Electrical & Computer Engineering at the University of Delaware. He received his BS degree from Caltech in 2000 and a PhD degree from MIT in 2005. His research interests span a range of topics in computer vision and graphics, especially on computational cameras and displays. He is a recipient of both an NSF CAREER Award and the AFOSR YIP Award.