CMOS Image Sensor With Per-Column $\Sigma\Delta$ ADC and Programmable Compressed Sensing

Yusuke Oike, *Member, IEEE*, and Abbas El Gamal, *Fellow, IEEE*

***Abstract*—A CMOS image sensor architecture with built-in single-shot compressed sensing is described. The image sensor employs a conventional 4-T pixel and per-column $\Sigma\Delta$ ADCs. The compressed sensing measurements are obtained via a column multiplexer that sequentially applies randomly selected pixel values to the input of each $\Sigma\Delta$ modulator. At the end of readout, each ADC outputs a quantized value of the average of the pixel values applied to its input. The image is recovered from the random linear measurements off-chip using numerical optimization algorithms. To demonstrate this architecture, a 256x256 pixel CMOS image sensor is fabricated in 0.15 μm CIS process. The sensor can operate in compressed sensing mode with compression ratio 1/4, 1/8, or 1/16 at 480, 960, or 1920 fps, respectively, or in normal capture mode with no compressed sensing at a maximum frame rate of 120 fps. Measurement results demonstrate capture in compressed sensing mode at roughly the same readout noise of 351 μV_{rms} and power consumption of 96.2 mW of normal capture at 120 fps. This performance is achieved with only 1.8% die area overhead. Image reconstruction shows modest quality loss relative to normal capture and significantly higher image quality than downsampling.**

***Index Terms*—$\Sigma\Delta$ ADC, CMOS image sensor, compressed/compressive sensing.**

I. Introduction

CMOS image sensor resolution and frame rate have been steadily increasing in recent years [1]–[4]. Since most of these sensors are used in mobile devices, power consumption is a primary concern. Studies have shown that the dominant component of power consumption in CMOS image sensors with column-parallel ADCs is A/D conversion followed by output readout [3]–[5]. As such, power consumption in CMOS image sensors increases at least linearly in resolution and frame rate. On-chip image compression can help to reduce the readout rate [6]–[8], hence the power consumption, but does not reduce the A/D conversion power consumption, since it is performed after all pixel values have been converted to the digital domain.

Compressed sensing (CS) [9]–[11] is a recently developed sampling theory that holds the promise to significantly reduce the number of captured measurements, hence the number of A/D conversions performed, without adversely affecting signal recoverability. The basic premise of this new theory is that if a signal can be efficiently represented using only a few coefficients in a transform domain, such as wavelet transform, it can be recovered from a number of random linear measurements that is much smaller than the number of samples dictated by the Nyquist rate. Since its inception a decade ago, there have been significant efforts in applying CS to different areas in data compression, error correcting codes, inverse problems, and analog-to-information acquisition. The most successful of these applications to date has been Magnetic Resonance Imaging (MRI) [12] in which CS is used to significantly reduce acquisition time—a crucial requirement for scanning children. Although the first attempt at applying CS was in visible range image acquisition [13]–[16], it has yet to be applied to any commercial image sensors. This is because previous implementations of compressed sensing in imaging [17]–[19] have suffered from several limitations, including limited scalability, the inability to perform single-shot imaging, and low SNR.

In this paper, we describe a CMOS image sensor with integrated compressed sensing that addresses the shortcomings of previous CS implementations. Our sensor architecture [20] is based on the idea that if multiple pixel values are applied sequentially to the input of a $\Sigma\Delta$ modulator and then the entire output sequence is decimated, the output of the ADC represents a quantized version of the sum (average) of the applied pixel values. Our main contribution is to show that this idea can be implemented in a CMOS image sensor that employs a conventional 4-T pixel design and per-column ADC with very small chip area overhead, allowing for single-shot imaging with no SNR degradation due to signal readout or A/D conversion relative to standard sensor readout. Maintaining such high SNR is important because it helps to reduce the degradation in image quality incurred by image compression. To allow for imaging of different types of scenes, the compression ratio (CR), that is, the ratio of the number of captured samples to the total number of pixels, can be programmed to 1 (no compressed sensing), 1/4, 1/8, or 1/16. Alternatively, the sensor frame rate can be proportionally boosted by a factor of 4, 8, or 16 with power consumption close to that for normal capture with no compression.

The rest of the paper is organized as follows. In the following section we provide a brief background on compressed sensing and previous CS implementations in imaging. In Section III, we present the basic principle on which our implementation is based. In Section IV, we describe the architecture and operation of our sensor. In Section V, we describe a 256 × 256 pixel prototype image sensor fabricated in 0.15 μm CMOS technology and present measured results.

Manuscript received April 27, 2012; revised August 03, 2012; accepted August 06, 2012. Date of publication October 04, 2012; date of current version December 31, 2012. This paper was approved by Guest Editor Maurits Ortmanns. This work was supported in part by Sony Corporation.

Y. Oike is with Sony Corporation, Atsugi-shi, Kanagawa, 243-0014, Japan (e-mail: Yusuke.Oike@jp.sony.com).

A. El Gamal is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: abbas@ee.stanford.edu).

Digital Object Identifier 10.1109/JSSC.2012.2214851

0018-9200/$31.00 © 2012 IEEE

II. Background and Previous Work

The Shannon–Nyquist theorem states that a bandlimited signal can be recovered from a sequence of samples if the sampling rate is higher than twice the bandwidth of the original signal. This is the principle on which all signal processing acquisition systems are based today. In many applications, however, the Nyquist sampling rate is much higher than the signal "information rate." Conventional digital image compression [6]–[8] cannot solve the high sampling rate problem because it is performed after sampling and A/D conversion are performed, hence can only reduce storage and transmission rates. In [21], an analog on-chip compression using a two-dimensional basis transform is reported. Although the variable bit resolution of this technique reduces the total A/D conversion power consumption, the implementation requires precise analog circuits and does not reduce the number of A/D conversions needed since the number of transform coefficients is the same as the number of pixels in the image sensor.

CS asserts that signals and images can be recovered from fewer samples than dictated by the Shannon-Nyquist theorem if they are *sparse*, that is, if their information rate is lower than the Nyquist rate. In the following we provide a brief introduction to CS. For a more detailed introduction, see for example [11].

Assume a discrete-time signal $X = [x_1 \ x_2 \ \ldots \ x_n]^T$ that may represent the pixel values of an image. Let $\Psi = [\psi_1 \ \psi_2 \ \ldots \ \psi_n]^T$, where $\psi_1, \psi_2, \ldots, \psi_n$ are n-dimensional orthonormal basis vectors, be a transform matrix such as for a wavelet transform. Using this transform matrix, the signal can be represented as $X = \sum_{i=1}^{n} s_i \psi_i$ and there is a one-to-one correspondence between X and its transform $S = [s_1 \ s_2 \ \ldots \ s_n]^T$. The signal X is said to be k-sparse if it has only k non-zero s_i coefficients. This is the property exploited in transform coding, such as JPEG and MPEG, in which only the largest k coefficients are kept and then losslessly compressed. However, in transform coding, we first acquire all the n samples x_i, quantize them, compute all the transform coefficients s_i, and keep only the k largest ones and their locations so that the signal can be reconstructed.

CS provides a much better way to exploit sparsity [9]–[11]. Instead of acquiring all n samples, we acquire a much smaller number of linear measurements $Y = [y_1 \ y_2 \ \ldots \ y_m]$, $m \ll n$, using an $m \times n$ measurement matrix $\Phi = [\phi_1 \ \phi_1 \ \ldots \ \phi_m]$, that is, we acquire

$$Y = \Phi X = \Phi \Psi S = \Theta S.$$

However, since $m \ll n$, this system of linear equations is highly underdetermined. It can be shown that if Φ is *incoherent* with Ψ, that is, $\max_{i,j} \phi_j^T \psi_i, i = 1, \ldots, n, j = 1, \ldots, m$ is small (akin to the well-known time-frequency duality in Fourier transform), then S is recoverable. The first key contribution of CS theory is to show that a *randomly* generated Φ matrix is incoherent with Ψ *with high probability*—in fact even a random ± 1 or binary matrix can be used. Note that in addition to reducing the number of samples acquired, CS has the advantage of being *non-adaptive* (much like uniform sampling)—the same measurement matrix Φ can be used as long as the signal sparsity is less than k.

The second contribution of CS theory is to show that S can be recovered with high probability using only $m = ck \log k$ measurements for some constant c by solving the L_1-norm minimization problem:

$$\min \ \|\tilde{S}\|_1 \quad \text{subject to} \quad Y = \Theta \tilde{S}.$$

This problem can be solved efficiently using linear programming (also referred to as *basic pursuit*). The basic pursuit algorithm can be extended to the more realistic noisy measurement case, $Y = \Theta S + Z$, where Z is the noise vector, by replacing the above constraint with one that requires Y to be close to $\Theta \tilde{S}$ in the L_2-norm sense. There is still a great deal of ongoing research on finding computationally efficient recovery algorithms for real-world signals; see for example [22] for a survey of recovery algorithms.

In the following, we briefly review previous different implementations of CS in image sensors and discuss the limitations that have precluded their successful commercialization.

A. Previous Work

To implement CS in an image sensor, one needs to find a way to read out not the pixel values themselves but a set of random linear combinations of pixel values. In previous work, this random linear sampling was implemented either in the optical domain [13]–[16] or on-chip in the circuit domain [17]–[19].

Optical domain implementations include [15] and [16]. In [15], the random measurement matrix is implemented using a random phase mask placed at the image's Fourier plane as depicted in Fig. 1(a). The modulated intensity image is then sampled using a low resolution imager to obtain the linear measurement vector. This idea has been demonstrated in IR imaging in which a significant part of the system cost is due to the focal plane array pixel resolution. This implementation uses a conventional low resolution imager and the image is captured in a single shot. However, it suffers from several limitations, especially in visible range imaging; (i) the optical mask degrades sensitivity, (ii) precise alignment is needed, and (iii) it is difficult to scale the system resolution. The second optical implementation of CS in imaging is the celebrated single pixel camera in [16] (see Fig. 1(b)). In this approach, the measurements are acquired sequentially using a single photodiode. The incident light from the scene is reflected off a digital micromirror device (DMD) and the reflected light is collected by a single photodiode. Each mirror can be independently oriented either towards the photodiode (corresponding to a 1) or away from it (corresponding to a 0). To acquire a measurement, a randomly generated 0/1 vector is used to set the mirror orientation. The output of the pixel represents the sum of the reflected light from the mirrors oriented towards the photodetector. This CS implementation, however, is attractive only when the cost of the photodetector is the dominant component of the total system cost, which is not the case in visible range imaging. It also suffers from several limitations; (i) the image is captured using multiple shots, which makes it unsuited for imaging a moving target, (ii) it is difficult to scale the system resolution, and (iii) reflections from the mirrors result in loss of sensitivity. In summary, existing optical domain implementation of CS are not well-suited to mainstream visible range imaging.

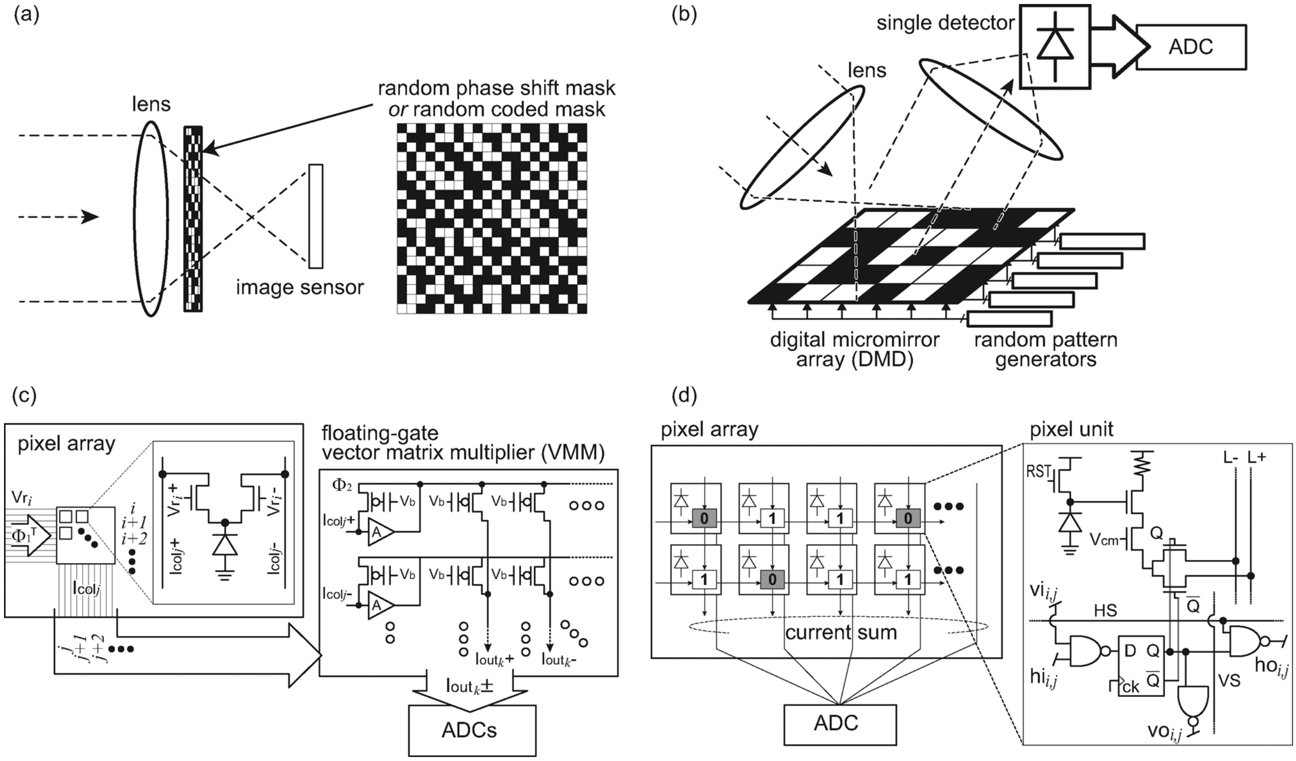


Fig. 1. Previous CS implementations: (a) camera with random phase shift mask [15], (b) single pixel camera [16], (c) CMOS image sensor with analog VMM [17], (d) CMOS image sensor with in-pixel random sequencer [18].

TABLE I
COMPARISON OF PREVIOUS CS IMPLEMENTATIONS

	Optical domain		On-chip electronics domain		
Architecture	[15]	[16]	[17]	[18]	This work
Single/Multi shot(s)	Single	Multi(sequential)	Multi(sequential)	Multi(sequential)	Single
Pixel access	-	-	Current sum	Current sum	Active pixel
Pinned PD	Available	Available	Not available	Not available	Available
#Tr./pixel	-	-	2	>30	4
Sensitivity	Loss	Loss	No loss	No loss	No loss
Alignment Calibration	Required	None	None	None	None

On-chip circuit domain implementation of CS avoid the loss of light and alignment issues of optical domain implementations. In [17], the measurements are captured one pixel block at a time using separable transform matrices. The first computation is performed at the focal plane using an array of computational pixels and column current summing as depicted in Fig. 1(c). The second computation is performed in an analog vector matrix multiplier (VMM) implemented using floating-gate transistors. This CS implementation requires multiple shot image capture, is difficult to scale to large pixel arrays, and suffers from low SNR due to the use of a passive pixel sensor and analog summing operations. In [18], [19], CS is implemented by shifting a random digital pattern representing a ± 1 measurement matrix via a shift register distributed over the pixel array (see Fig. 1(d)). The currents from the pixels with the $+1$ pattern in the same column are summed over one line while the currents from the pixels with the -1 pattern are summed over a second column line. The total weighted sum is then performed again in analog at the chip level. This implementation requires multiple shot image capture, is not scalable due to the large pixel size, and suffers from low SNR due to pixel design and analog summation.

Table I summarizes and compares these previous CS implementations as well as our implementation. As we will see, the sensor architecture described in this paper addresses the limitations of the aforementioned approaches. It employs a standard 4-T active pixel and achieves single shot imaging with no SNR degradation and with very small die area overhead. We will also see that power consumption is reduced by almost the same factor as the compression ratio.

III. USING $\Sigma\Delta$ ADC TO OBTAIN LINEAR MEASUREMENTS

Our image sensor implementation of CS uses a random binary measurement matrix. The main idea is to simultaneously perform summation (averaging) and quantization of the randomly selected pixel values via a $\Sigma\Delta$ ADC. To obtain a measurement,

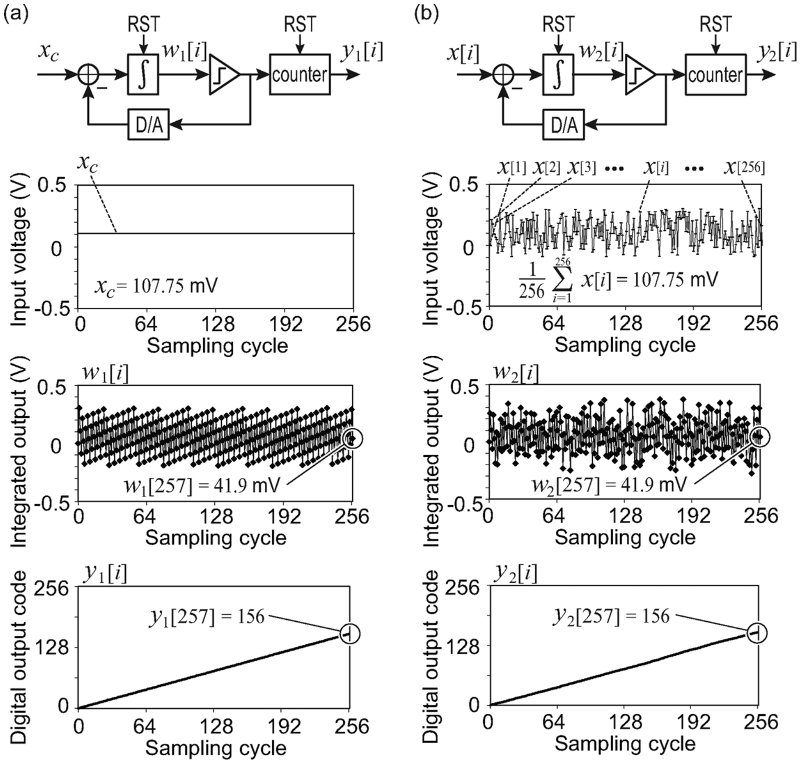


Fig. 2. (a) Incremental ΣΔ AD operation, (b) averaging and quantizing pixel values using ΣΔ ADC.

the selected pixel values are applied sequentially to the input of the modulator as depicted in Fig. 2(b). The output from the modulator is then decimated using a counter to obtain a digital representation of the average pixel values. Fig. 2 shows simulation results for an ideal first-order ΣΔ ADC (with infinite amplifier gain, no temporal noise, etc.) with 256 sampling cycles corresponding to 8-bit quantization. Fig. 2(a) shows the output of the ΣΔ modulator and its decimated value for a constant input voltage $x_c = 107.75$ mV and Fig. 2(b) shows the output when a different input $x[i]$ is applied to the ΣΔ ADC input in each cycle $i = 1, \ldots, 256$ such that the average of these inputs is equal to x_c, i.e.,

$$\frac{1}{256}\sum_{i=1}^{256} x[i] = x_c = 107.75\ \text{mV}.$$

Note that the final internal integrator voltages $w_1[256]$ and $w_2[256]$ and their respective decimated value $y_1[256]$ and $y_2[256]$ are equal in both cases, which shows that the ADC performs simultaneous averaging and quantization when multiple input values are sequentially applied to it. In Section V, we present measured DNL and INL results from a fabricated ΣΔ with multiple pixel values applied to its input.

Fig. 3 illustrates the implementation of random linear sampling for a 4 × 4 pixel image sensor with per-column ΣΔ ADC and compression ration $\text{CR} = 1/4$, i.e., capturing four random linear samples from the 16 pixels, using a 4 × 16 measurement matrix Φ. As shown in the figure, the four samples are acquired simultaneously using a multiplexer controlled by the column values of the Φ matrix. The pixel values are sequentially applied to the input of the multiplexer in a raster scan fashion. In clock cycle $j = 1, 2, \ldots, 16$, the j-th pixel value is applied to ADC $i = 1, \ldots, 4$ if $\phi_{ij} = 1$, and a reference voltage V_g is applied to it if $\phi_{ij} = 0$. The modulators may continue sampling after all pixel values have been applied to obtain a higher resolution quantized value of each sample average.

Performing random sampling on an entire pixel array, however, is not scalable both in die size and in the computational complexity of image recovery. Hence, in our implementation, the pixel array is partitioned into blocks and sampling is performed on each block using the same measurement matrix as detailed in the next section. An important question that arises here is how large does the pixel block needs to be in order to achieve high compression ratio while maintaining good image quality. To answer this question, we varied the pixel block sizes from 8 × 8 to 64 × 64 and performed simulations with different types of images. Fig. 4 plots the PSNR of the reconstructed image versus the pixel block size for different compression ratios. In this simulation, the A/D conversion resolution was set to 16 bits. The number of pixels averaged was half the number of pixels in the block. For example, for the 16 × 16 pixel block, 128 pixels were averaged. Note that PSNR degrades slowly with block size in this range. As such, we decided to use a 16 × 16 block size in our implementation.

A second important parameter to choose is the ADC bit resolution. We again performed simulations to compare image quality as measured by PSNR as a function of ADC resolution. In this simulation, the block size was set to 16 × 16 and the number of averaged pixel values was 128. Fig. 5, shows that PSNR improves markedly when the resolution is increased from 8 to 11 bits, but no noticeable improvement is observed beyond 12 bits. Based on these results, we decided on 12-bit

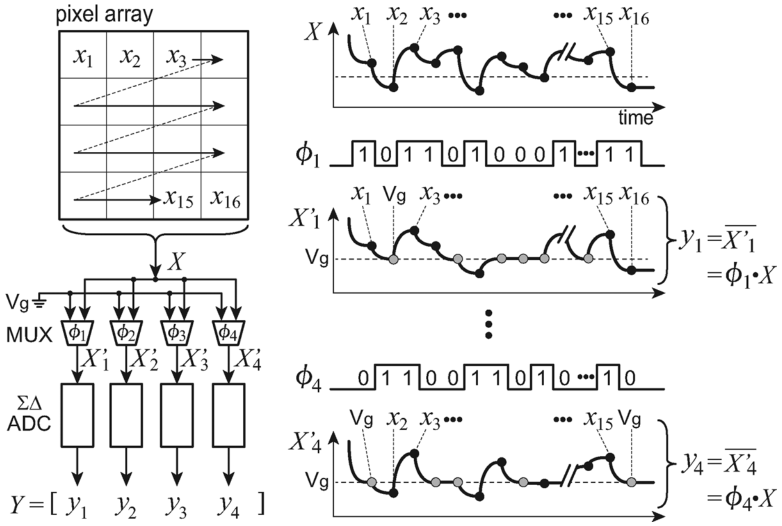

pixel array
x_1 x_2 x_3
x_{15} x_{16}
X
Vg
MUX
ϕ_1 ϕ_2 ϕ_3 ϕ_4
X'_1 X'_2 X'_3 X'_4
ΣΔ ADC
$Y=[\ y_1\ \ y_2\ \ y_3\ \ y_4\ ]$
X
x_1 x_2 x_3 ••• ••• x_{15} x_{16}
time
ϕ_1 1 0 1 1 0 1 0 0 0 1 ••• 1 1
X'_1
x_1 Vg x_3 ••• ••• x_{15} x_{16}
Vg
$y_1=\overline{X'_1}$
$=\phi_1 \bullet X$
ϕ_4 0 1 1 0 0 1 1 0 1 0 ••• 1 0
X'_4
Vg x_2 x_3 ••• ••• x_{15} Vg
Vg
$y_4=\overline{X'_4}$
$=\phi_4 \bullet X$

Fig. 3. Random linear sampling performed at per-column ΣΔ ADC.

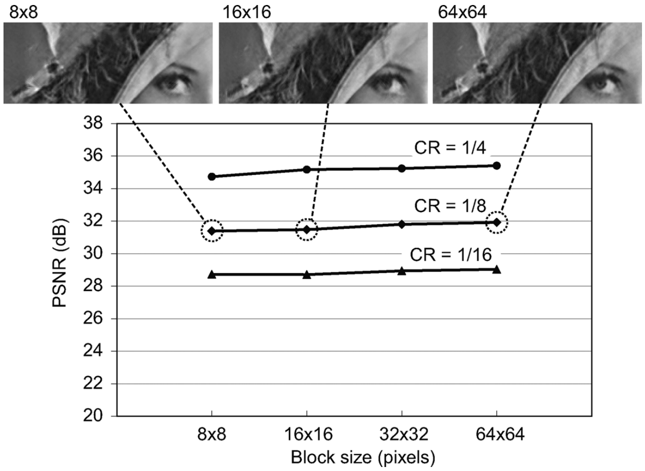


Fig. 4. Recovered image quality measured by PSNR as a function of pixel block size.

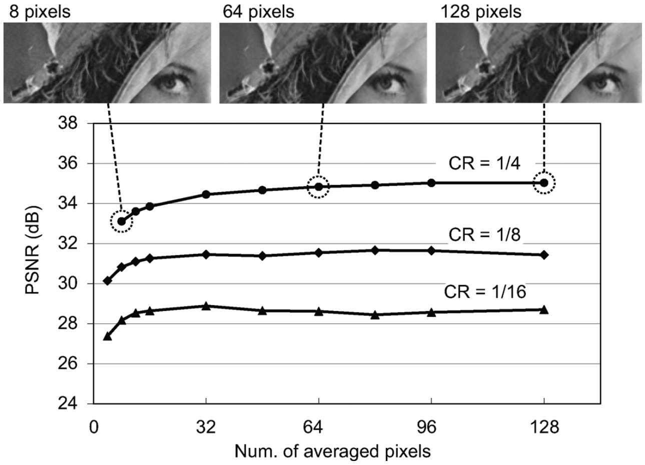


Fig. 6. Recovered image quality measured by PSNR as a function of the number of averaged pixels per sample.

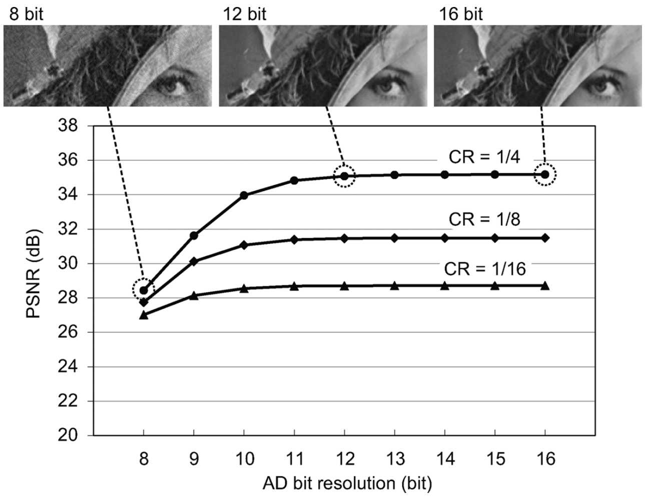


Fig. 5. Recovered image quality measured by PSNR as a function of AD bit resolution.

ADC resolution. The implementation of our ADC is detailed in the next section.

We also performed simulations to decide on the number of pixels to be averaged in each sample, that is, the density of ones in the measurement matrix. This is an important parameter because if Φ has too few or too many ones, the needed incoherence with a transform matrix is lost resulting in poor image recovery. Fig. 6 plots the PSNR of the reconstructed image versus the number of averaged pixel values. In this simulation, the bock size and the ADC resolution were set at 16×16 pixels and 16 bits, respectively. Based on these simulation results and implementation considerations to be discussed in the next section, we decided on having 64 pixel values averaged in each measurement.

Remark: The simulated PSNR values in Figs. 4–6 are also obtained under ideal ΣΔ circuit model.

IV. Chip Architecture and Detailed Operation

Fig. 7 depicts the architecture of our CMOS image sensor. It comprises a 256 × 256 pixel array with row block/pixel selectors, column block selectors, compressed sensing multiplexer (CS-MUX) controlled by a pseudo-random pattern generator

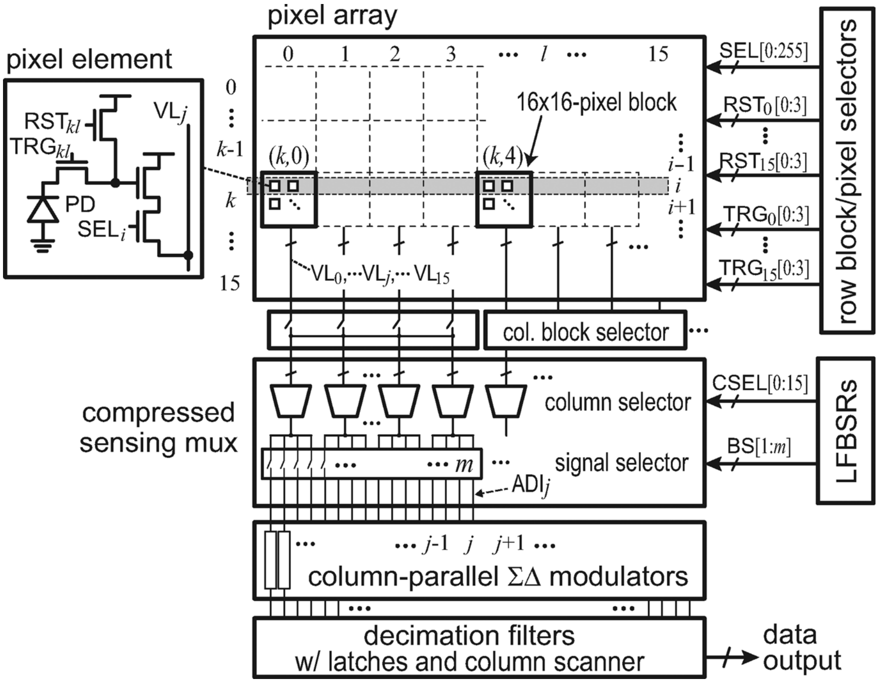


Fig. 7. Image sensor architecture configured for compression ratio $\mathrm{CR} = 1/4$.

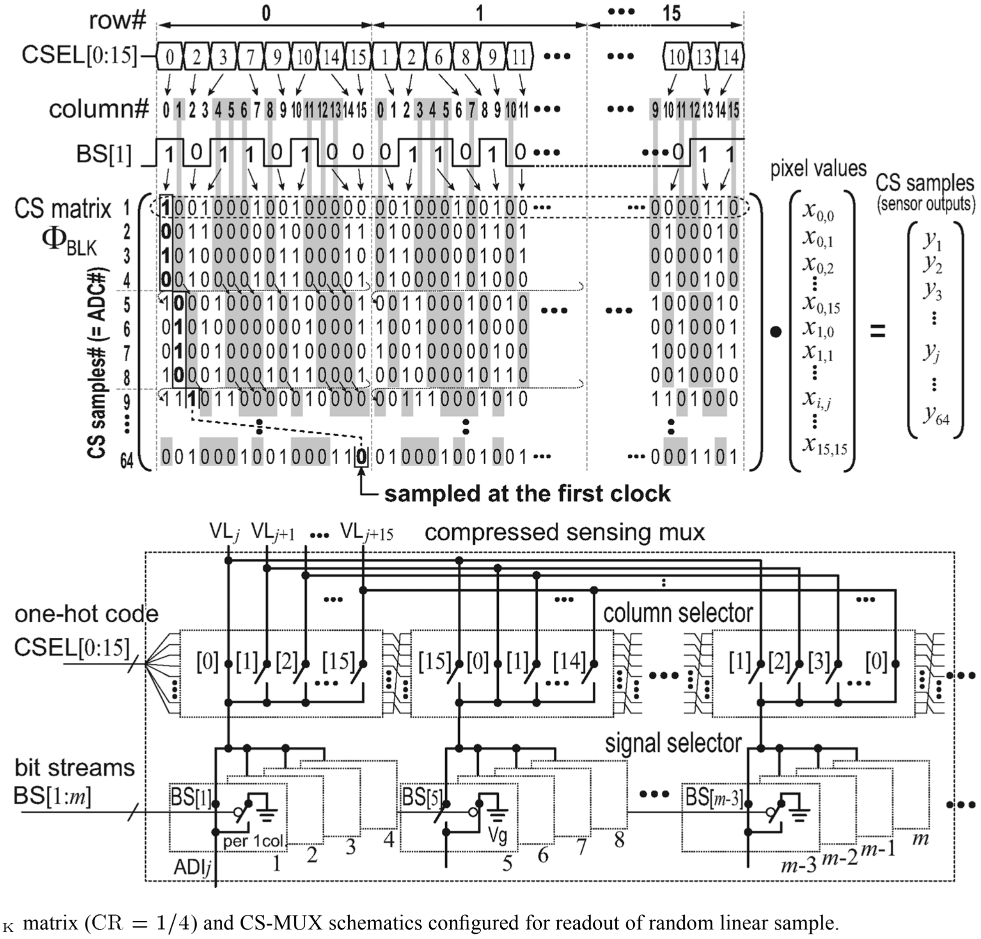

row#
CSEL[0:15]
column#
BS[1]
CS matrix $\Phi_{\rm BLK}$
CS samples# (= ADC#)
pixel values
CS samples (sensor outputs)
$x_{0,0}$, $x_{0,1}$, $x_{0,2}$, $x_{0,15}$, $x_{1,0}$, $x_{1,1}$, $x_{i,j}$, $x_{15,15}$
y_1, y_2, y_3, y_j, y_{64}
sampled at the first clock
VL$_j$ VL$_{j+1}$ ··· VL$_{j+15}$
compressed sensing mux
one-hot code CSEL[0:15]
column selector
signal selector
bit streams BS[1:m]
per 1col.
Vg
ADIj

Fig. 8. Example $\Phi_{\rm BLK}$ matrix ($\mathrm{CR} = 1/4$) and CS-MUX schematics configured for readout of random linear sample.

(LFBSRs), column-parallel ΣΔ ADCs with decimation filters, and a column output scanner. The pixel array is segmented into 16×16 pixel blocks and CS is performed over each block using the same $m\times$ 256 random binary CS matrix $\Phi_{\rm BLK}$ specified by the 16-bit one-hot code CSEL[0:15] and the 64-bit bit stream BS[1:m], where $m = 64$, 32, or 16 is the number of random

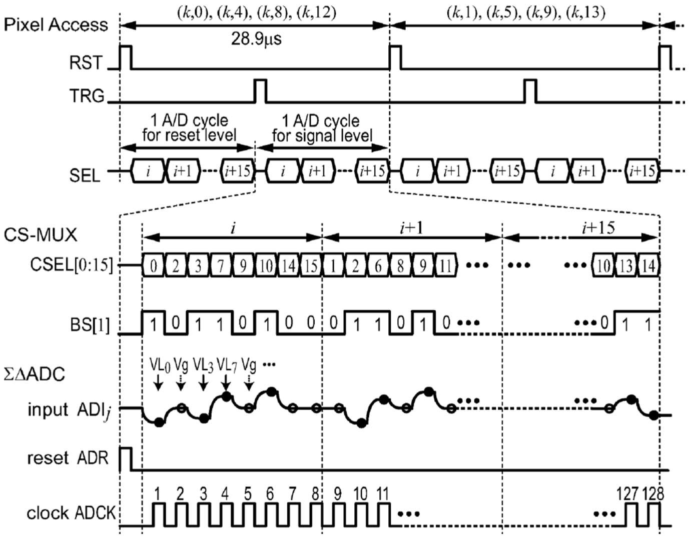


Fig. 9. Readout timing diagram for row block k at compression ratio $\mathrm{CR} = 1/4$.

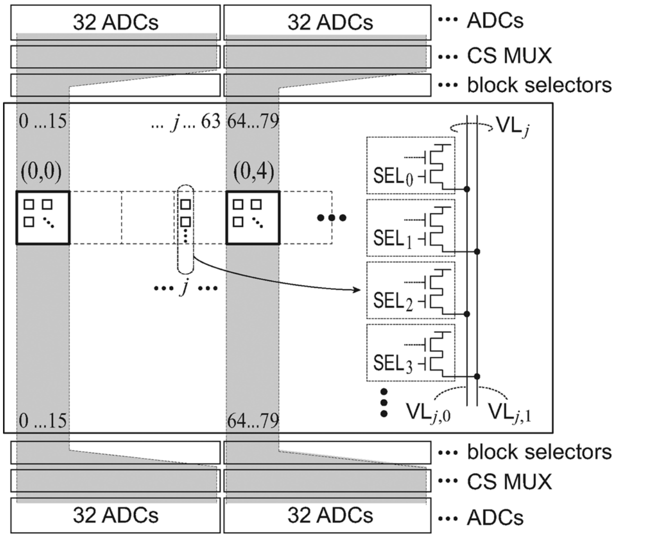


Fig. 10. Details of column circuit top-bottom split configured for compression ratio of 1/4.

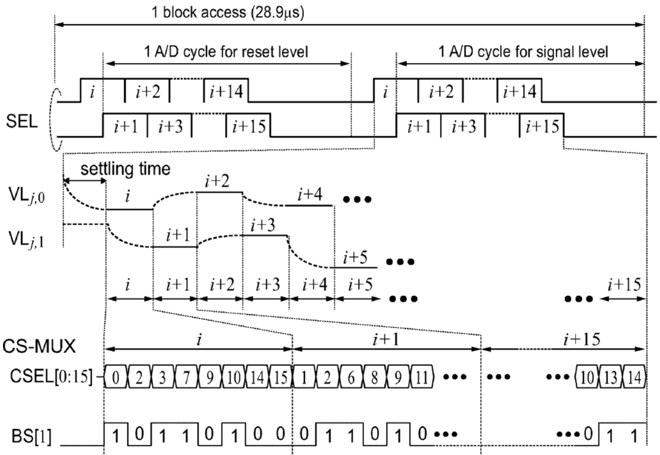


Fig. 11. Timing diagram of interleaved and overlapped pixel access.

linear samples obtained from each block depending on the selected CR. Alternatively, conventional sensor operation can be performed by bypassing the CS-MUX and directly reading out the pixel values themselves. The pixel comprises a pinned photodiode and 4 transistors. The transfer and reset gates in each pixel are controlled by TRGk_l and RST_l, respectively, in a per-block fashion, while the select gate is controlled by SEL_i in a per-row-fashion.

Chip readout operation is described with the help of Figs. 8 and 9 for $\mathrm{CR} = 1/4$ using the example block CS matrix in Fig. 8. As shown in the top three waveforms in Fig. 9, readout is performed simultaneously for each 4-block group (k, b), $(k, b+4)$, $(k, b+8)$, $(k, b+12)$ within block row $k = 0, \ldots, 15$ for $b \in \{0, 1, 2, 3\}$. During this readout, $m = 64$ consecutive ADCs are dedicated to each of the four blocks (for $\mathrm{CR} = 1/8$, eight blocks are read out at the same time and 32 ADCs are dedicated to each block, etc.). Digital correlated double sampling is performed by first acquiring random linear samples of the pixel reset values according to Φ_{BLK} and then acquiring the corresponding random linear samples of the pixel signal values using the same matrix. The operation in each phase is performed one pixel row at a time (see bottom parts of Figs. 8 and 9). For each row $i = 0, \ldots, 15$, and simultaneously for the 4-block group, 8 out of the 16 pixel values are sequentially selected by each column selector. This selection is performed using a different shift of the 16-bit one-hot code CSEL to control each of the four column selectors. The purpose of this selection step is to reduce the oversampling ratio and the load on each pixel column output. The selected pixel value at the output of each column selector is then either applied to each of four ADC inputs ADI_j or not via a signal selector controlled by the random bit stream BS. When a pixel is not selected, a reference voltage V_g is applied to ADI_j. The same procedure is repeated for each row of pixels within the selected 4-block group until all random linear samples from these blocks have been read out. The entire readout process is then repeated for the other 4-block groups within the block row. The next block row is then read out in the same manner and the process is repeated until all 64×256 linear samples of the current frame are acquired (for $\mathrm{CR} = 1/4$). The segmentation of the CS-MUX can be programmed to implement other compression ratios. For example, to achieve $\mathrm{CR} = 1/8$, eight blocks are chosen at a time and 32 ADCs are dedicated to each block. Note that in this CS operation, the random linear measurements for each frame are acquired after only one exposure period. This single-shot imaging makes it possible to capture a moving target with CS. The motion artifacts caused by the per-block electronic shutter is no more severe than that for a conventional CMOS image sensor with a rolling shutter. The single-shot operation also makes it possible to employ a global-shutter such as [23] to completely avoid motion artifacts.

Details of the column circuit top-bottom split are shown in Fig. 10. Two vertical signal lines are alternately connected to even and odd pixel rows. The pixel readout operation initiated by SEL_i is interleaved so that the pixel readout settling time overlaps with the A/D conversion of the previous pixel row as shown in Fig. 11.

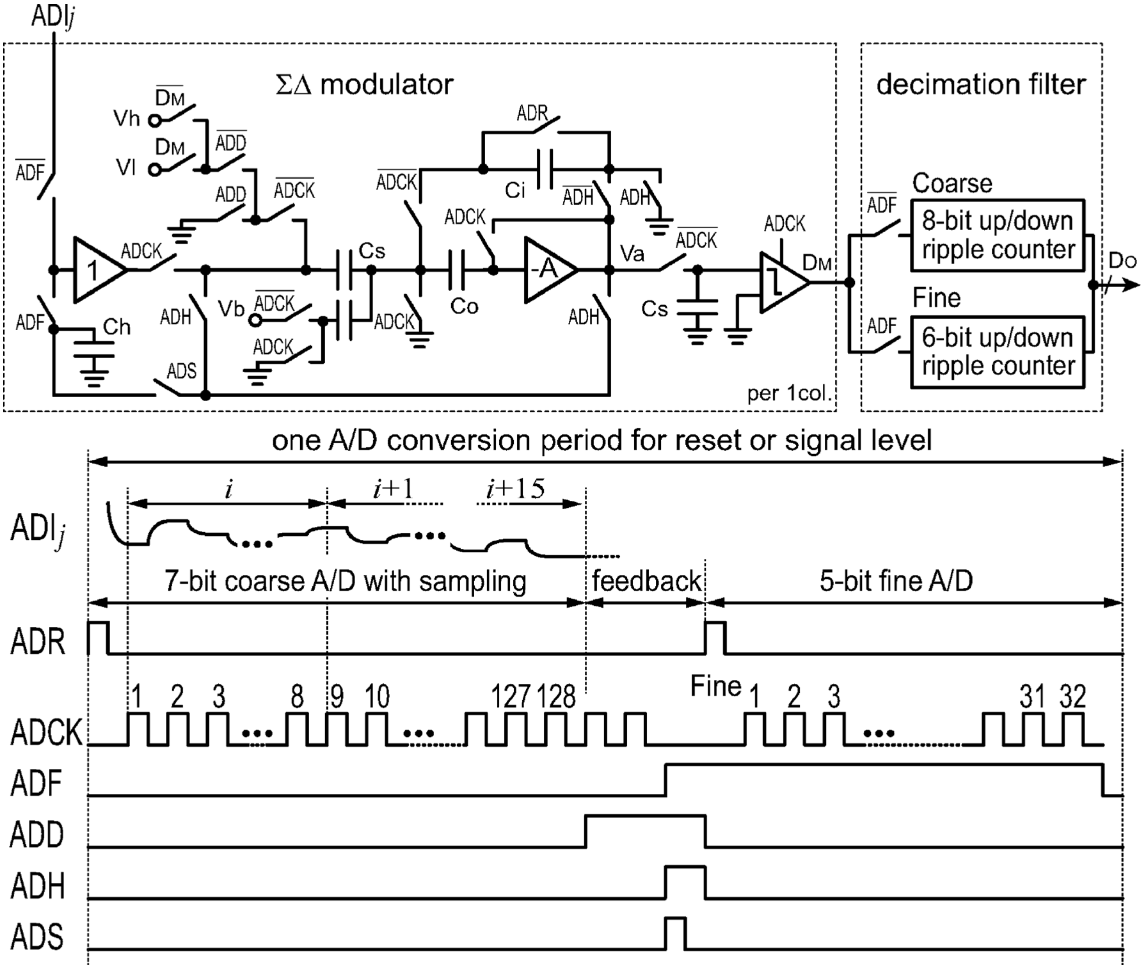


Fig. 12. Circuit schematic and timing diagram of the first-order algorithmic ΣΔ ADC.

Fig. 12 shows the circuit diagram and operation of the per-column first-order algorithmic ΣΔ ADC [24]. The ADC operates in incremental mode in which the modulator and decimation filter are reset before each A/D conversion, hence it does not suffer from the problem of idle tones [25], [26]. The ΣΔ modulator samples the ADI signal generated by the CS-MUX and outputs a 128-bit stream corresponding to a 7-bit coarse quantization of the average of the selected pixel values. The residual voltage of the integrator is then fed back to the input for an additional 32-bit stream corresponding to 5-bit fine quantization. The decimation filter is implemented using an 8-bit and a 6-bit up/down ripple counters with digital CDS capability. In our prototype implementation, the first-order modulator is based on a circuit topology with an improved virtual ground for offset cancellation [4]. The integrator amplifier is a cascoded common-source amplifier and achieves 73 dB gain at room temperature.

Remark: Although the CS image sensor architecture we described is for monochromatic imaging, it can be readily extended to color imaging. Using a color filter array (CFA) such as a Bayer filter, the CS-MUX is used to separate the color channels, and then random linear measurements are acquired from each color channel using a different set of column ADCs as described above. Each color channel is then separately recovered off-chip from its random linear measurements.

V. Prototype and Experimental Results

To demonstrate our architecture, we fabricated a prototype image sensor. Fig. 13 depicts the prototype chip microphotograph and Table II lists the main chip characteristics. As can be seen, the column readout circuits are split between the top and the bottom of the array to allow for a wider ADC pitch. The image sensor is fabricated in a 1P6M 0.15 μm CMOS process with MIM capacitors and pinned photodiodes in a 2.9×3.5 mm^2 die. The pixel pitch is 5.5 μm and the column selectors/ADC pitch is 11 μm. The CS overhead, that is, the CS-MUX and the random pattern generators, occupy only 1.8% of the total the chip area.

The prototype has been fully tested and characterized. Table III summarizes the chip characteristics. The measured readout noise is 368 μV_{rms} in the normal mode at 120 fps with conversion gain of 20 $\mu\text{V/e-}$ and saturation level of 25,000 e-. In CS mode with $\text{CR} = 1/4$, the readout noise is 351 μV_{rms}. The ADC noise measured at the CS reference voltage $V_g = 700$ mV is 339 μV_{rms}. Fig. 14 shows measured DNL and INL results for a single per-column ADC. The mea-

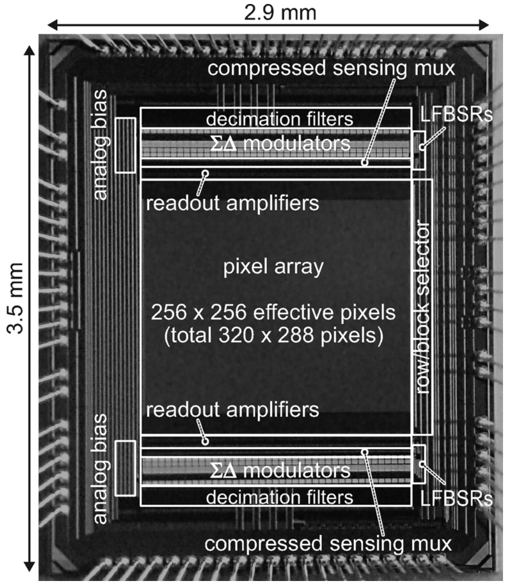


Fig. 13. Chip microphotograph of the prototype image sensor.

TABLE II
CHIP SPECIFICATIONS

Technology	0.15 μm 1P6M CIS process
Chip size	2.9 mm (H) $\times$ 3.5 mm (V)
Power supplies	3.3 V / 2.0 V / 1.8 V
Num. of total pixels	320 (H) $\times$ 288 (V)
Num. of effective pixels	256 (H) $\times$ 256 (V)
Pixel size	5.5 μm $\times$ 5.5 μm
Pixel type	Pinned photodiode with 4 transistors
ADC	12 bit 11μm-pitch $\Sigma\Delta$ ADCs

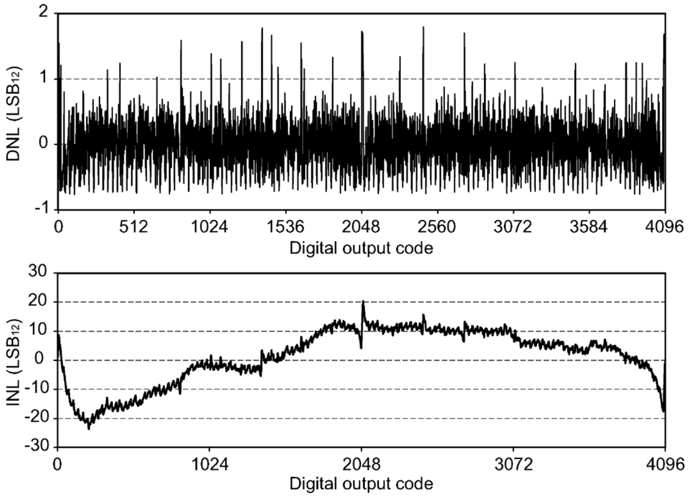


Fig. 14. Measured $\Sigma\Delta$ ADC linearity. (a) DNL. (b) INL.

TABLE III
CHIP CHARACTERISTICS

Conversion gain	20 μV/e-
Saturation level	25,000 e-
Read noise	368 $\mu V_{\rm rms}$ (normal mode)
	351 $\mu V_{\rm rms}$ (CS mode, CR=1/4)
Frame rate	120 fps (no compression)
	480, 960, 1920 fps (CR=1/4, 1/8, 1/16)
Power consumption	93.1 mW at 120 fps (normal mode)
	96.2 mW at 1920 fps (CR=1/16)
A/D resolution	12 bit, 242 μV/LSB
DNL	-0.76/+1.80 LSB
INL	$< \pm 0.6\%$
Column FPN	0.57 $\rm LSB_{rms}$ at dark level

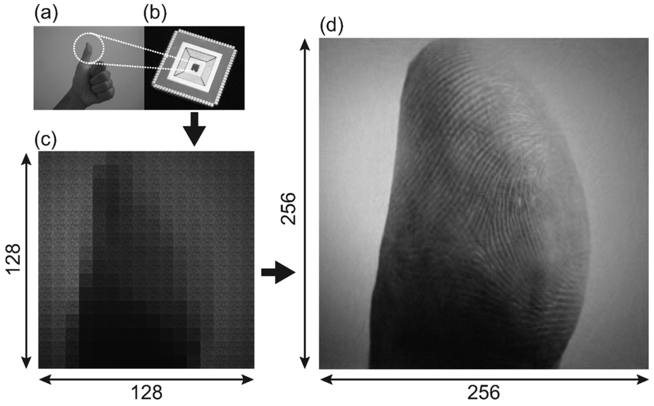


Fig. 15. Captured and reconstructed images: (a) target object, (b) packaged prototype chip, (c) sensor output at $\mathrm{CR} = 1/4$, (d) reconstructed image.

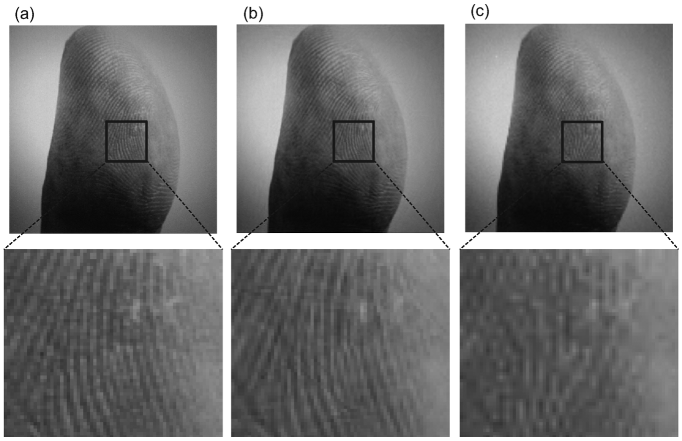


Fig. 16. Sample images captured in: (a) normal mode at 120 fps, (b) compressed sensing at $\mathrm{CR} = 1/4$ and 480 fps, (c) downsampling at 1/4 ratio.

sured 12 b DNL is within -0.76 to $+1.80$ LSB and the INL is $< \pm 0.6\%$. The DNL is higher than expected because of the gain error during the fine A/D conversion phase. The feedback signal from the residual voltage of the integrator is multiplied by Ci/Cs. This gain is set by connecting the capacitors Cs and Ci in the opposite way of the integration phase. The measured column FPN rms at dark level is 0.57 LSB.

Fig. 15 depicts an example of a captured image and its reconstruction at $\mathrm{CR} = 1/4$. The captured image (Fig. 15(c)) has 128×128 measurements while the recovered image (Fig. 15(d)) has 256×256 pixels. The image is recovered using a block-based algorithm involving a two-step iterative curvelet thresholding [27], [28] running on a PC. Fig. 16 shows three close-ups that compare: (a) the image captured using the normal mode

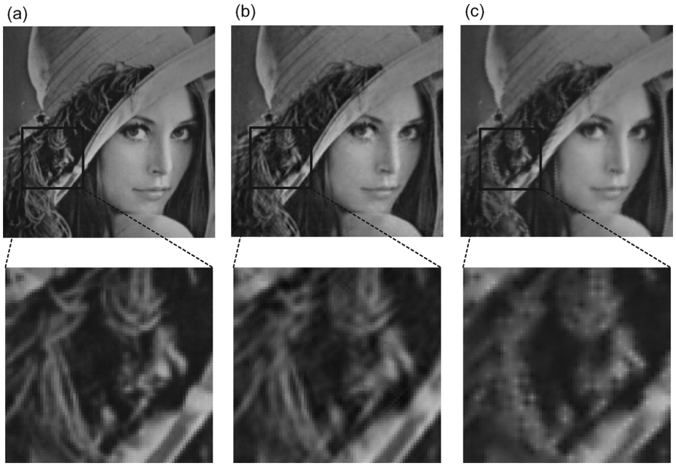


Fig. 17. Sample images captured in: (a) normal mode at 120 fps, (b) compressed sensing at $\mathrm{CR} = 1/8$ and 960 fps, (c) downsampling at 1/8 ratio.

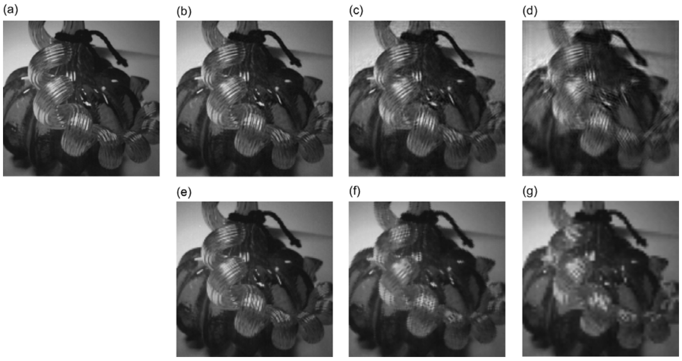


Fig. 18. Sample images captured in: (a) normal mode, (b)–(d) compressed sensing at $\mathrm{CR} = 1/4$, 1/8 and 1/16, respectively, (e)–(g) downsampling at 1/4, 1/8, and 1/16 ratio, respectively.

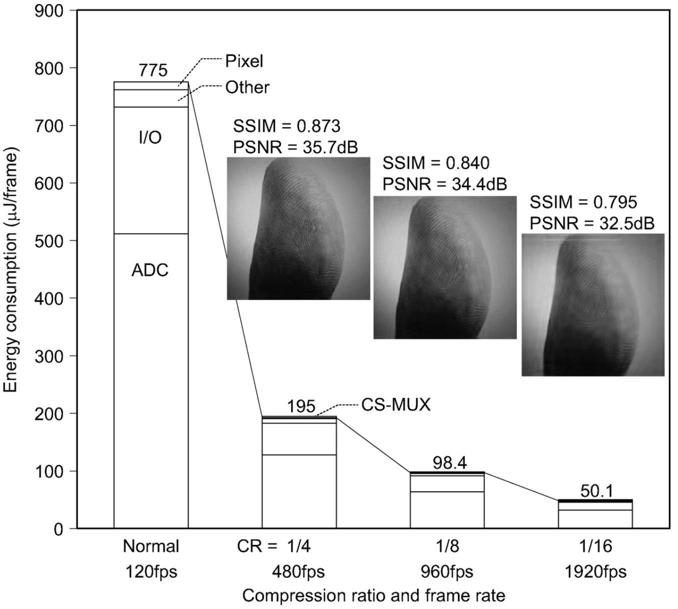


Fig. 19. Energy consumption per frame, SSIM, and PSNR versus compression ratio CR.

with no compressed sensing, (b) the image captured with compressed sensing at $\mathrm{CR} = 1/4$, and (c) an image captured by conventional downsampling at 1/4 ratio. Note that the image with compressed sensing has more detailed texture than the downsampled image and is very close to the image with no compression, which is captured at 1/4th the frame rate of the compressed sensing image.

Fig. 17 shows another sample image taken at $\mathrm{CR} = 1/8$ with comparisons to the same image captured using the normal mode and via downsampling at 1/8 ratio. Fig. 18 shows a third captured image using different compression ratios. As can be seen, image quality gracefully degrades as compression ratio is decreased. However, in all cases, the compressed sensing image contains better texture details than its downsampled image counterpart.

Fig. 19 compares the energy consumption per frame, Structural Similarity (SSIM) [29], and PSNR for images captured at different compression ratios to the normal capture image. Note that the reduction in energy consumption per frame is almost the same as the compression ratio, with only 3.4% increase in overall power consumption using $\mathrm{CR} = 1/16$ relative to normal capture. Also, note that both SSIM and PSNR drop only marginally with the compression ratio.

VI. Conclusion

We presented the first image sensor architecture with single-shot compressed sensing. Per-block compressed sensing is programmably implemented simultaneously with A/D conversion via per-column $\Sigma\Delta$ ADC and a column multiplexer for random pixel selections. The architecture employs an off-the-shelf pixel design and can be implemented with very small area overhead. To demonstrate this architecture, we designed and fabricated a 256×256 image sensor in 0.15 μm CIS process. Measured results show no loss in SNR or sensitivity relative to normal capture and close to linear reduction in energy consumption per frame with CS compression ratio. We also showed that image quality degrades gracefully with compression ratio and is significantly higher than downsampling with the same readout rate.

Much work remains to be done to demonstrate the usefulness of compressed sensing in visible range image sensors. Current CS recovery algorithms do not perform uniformly well over different types of images and require significant computation time, making them unsuited for mobile system implementation. We demonstrated significant reduction in power consumption and/or increase in frame rate using our architecture, but with increase in off-chip image recovery complexity.

Acknowledgment

The authors wish to thank Prof. J. Ma, Dr. J. E. Farrell, Dr. S. Kavusi, and Dr. K. Ghosh for helpful discussions. The CS recovery algorithm used in this study was implemented with support from Prof. J. Ma. The chip used in this study was fabricated in the LFoundry GmbH, Landshut, Germany.

REFERENCES

[1] T. Toyama, K. Mishina, H. Tsuchiya, T. Ichikawa, H. Iwaki, Y. Gendai, H. Murakami, K. Takamiya, H. Shiroshita, Y. Muramatsu, and T. Furusawa, "A 17.7 Mpixel 120 fps CMOS image sensor with 34.8 Gb/s readout," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2011, pp. 420–422.

[2] S. Matsuo, T. J. Bales, M. Shoda, S. Osawa, K. Kawamura, A. Andersson, H. Munirul, H. Honda, B. Almond, Y. Mo, J. Gleason, T. Chow, and I. Takayanagi, "8.9-megapixel video image sensor with 14-b column-parallel SA-ADC," *IEEE Trans. Electron Devices*, vol. 56, no. 11, pp. 2380–2389, Nov. 2009.

[3] J.-H. Park, S. Aoyama, T. Watanabe, K. Isobe, and S. Kawahito, "A high-speed low-noise CMOS image sensor with 13-b column-parallel single-ended cyclic ADCs," *IEEE Trans. Electron Devices*, vol. 56, no. 11, pp. 2414–2422, Nov. 2009.

[4] Y. Chae, J. Cheon, S. Lim, M. Kwon, K. Yoo, W. Jung, D.-H. Lee, S. Ham, and G. Han, "A 2.1 Mpixel 120 frame/s CMOS image sensor with column-parallel ΔΣ ADC architecture," *IEEE J. Solid-State Circuits*, vol. 46, no. 1, pp. 236–247, Jan. 2011.

[5] T. Watabe, K. Kitamura, T. Sawamoto, T. Kosugi, T. Akahori, T. Iida, K. Isobe, T. Watanabe, H. Shimamoto, H. Ohtake, S. Aoyama, S. Kawahito, and N. Egami, "A 33 mpixel 120 fps cmos image sensor using 12 b column-parallel pipelined cyclic adcs," in *IEEE ISSCC Dig. Tech. Papers*, 2012, pp. 388–389.

[6] Y. Nishikawa, S. Kawahito, M. Furuta, and T. Tamura, "A high-speed CMOS image sensor with on-chip parallel image compression circuits," in *Proc. IEEE Custom Integrated Circuits Conf. (CICC)*, 2007, pp. 833–836.

[7] A. Olyaei and R. Genov, "Focal-plane spatially oversampling CMOS image compression sensor," *IEEE Trans. Circuits Syst. I*, vol. 54, no. 1, pp. 26–34, Jan. 2007.

[8] A. Nilchi, J. Aziz, and R. Genov, "Focal-plane algorithmically-multiplying CMOS computational image sensor," *IEEE J. Solid-State Circuits*, vol. 44, no. 6, pp. 1829–1839, Jun. 2009.

[9] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[10] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[11] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[12] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MRI imaging," *ISMRM Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, Dec. 2007.

[13] S. D. Babacan, R. Ansorge, M. Luessi, R. Molina, and A. K. Katsaggelos, "Compressive sensing on light fields," in *Proc. IEEE Int. Conf. Image Processing*, Nov. 2009, pp. 2337–2340.

[14] A. Stern, Y. Rivenson, and B. Javidi, "Optically compressed image sensing using random aperture coding," in *Proc. SPIE 6975*, May 2008, pp. 69 750D-1–10.

[15] R. F. Marcia and R. M. Willett, "Compressive coded aperture superresolution image reconstruction," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2008, pp. 833–836.

[16] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 83–91, Mar. 2008.

[17] R. Robucci, J. D. Gray, L. K. Chiu, J. Romberg, and P. Hasler, "Compressive sensing on a CMOS separable-transform image sensor," *Proc. IEEE*, vol. 98, no. 6, pp. 1089–1101, Jun. 2010.

[18] L. Jacques, P. Vandergheynst, A. Bibet, V. Majidzadeh, A. Schmid, and Y. Leblebici, "CMOS compressed imaging by random convolution," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2009, pp. 1113–1116.

[19] V. Majidzadeh, L. Jacques, A. Schmid, P. Vandergheynst, and Y. Leblebici, "A 256x256 pixel 76.7 mW CMOS imager/compressor based on real-time in-pixel compressive sensing," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS)*, 2010, pp. 2956–2959.

[20] Y. Oike and A. El Gamal, "A 256x256 CMOS image sensor with ΔΣ-based single-shot compressed sensing," in *IEEE ISSCC Dig. Tech. Papers*, 2012, pp. 386–387.

[21] S. Kawahito, M. Yoshida, M. Sasaki, K. Umehara, D. Miyazaki, Y. Tadokoro, K. Murata, S. Doushou, and A. Matsuzawa, "A CMOS image sensor with analog two-dimensional DCT-based compression circuits for one-chip camera," *IEEE J. Solid-State Circuits*, vol. 32, no. 12, pp. 2030–2041, Dec. 1997.

[22] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, Feb. 2009.

[23] M. Sakakibara, Y. Oike, T. Takatsuka, A. Kato, K. Honda, T. Taura, T. Machida, J. Okuno, A. Ando, T. Fukuro, T. Asatsuma, S. Endo, J. Yamamoto, Y. Nakano, T. Kaneshige, I. Yamamura, T. Ezaki, and T. Hirayama, "An 83 dB-dynamic range single-exposure global-shutter CMOS image sensor with in-pixel dual storage," in *IEEE ISSCC Dig. Tech. Papers*, 2012, pp. 380–381.

[24] G. Mulliken, F. Adil, G. Cauwenberghs, and R. Genov, "Delta-sigma algorithmic analog-to-digital conversion," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS)*, 2002, vol. 4, pp. 687–690.

[25] J. Markus, J. Silva, and G. C. Temes, "Theory and applications of incremental $\delta\sigma$ converters," *IEEE Trans. Circuits Syst. I*, vol. 51, no. 4, pp. 678–690, Apr. 2004.

[26] S. Kavusi, H. Kakavand, and A. El Gamal, "On incremental sigma-delta modulation with optimal filtering," *IEEE Trans. Circuits Syst. I*, vol. 53, no. 5, pp. 1004–1015, May 2006.

[27] J. Ma, "Improved iterative curvelet thresholding for compressed sensing and measurement," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 1, pp. 126–136, Jan. 2011.

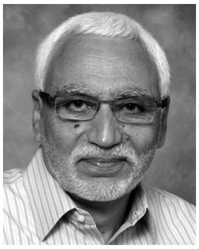
[28] J. Ma and M. Y. Hussaini, "Extensions of compressed imaging: Flying sensor, coded mask, and fast decoding," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 9, pp. 3128–3139, Sep. 2011.

[29] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

Yusuke Oike (S'00–M'05) received the B.S., M.S., and Ph.D. degrees in electronic engineering from the University of Tokyo, Tokyo, Japan, in 2000, 2002, and 2005, respectively.

In 2005, he joined Sony Corporation, Japan, where he has been engaged in research and development of architectures, circuits and devices for CMOS image sensors. From 2010 to 2011, he was a Visiting Scholar at Stanford University, Stanford, CA. His research interests include architecture and mixed-signal circuit design for image sensors and image processing algorithms.

Dr. Oike serves on the technical program committee of the IEEE ISSCC. He is a member of the Institute of Image Information and Television Engineers in Japan (ITEJ).



Abbas El Gamal (S'71–M'73–SM'83–F'00) received the B.Sc. (honors) degree in electrical engineering from Cairo University, Cairo, Egypt, in 1972 and the M.S. degree in statistics and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 1977 and 1978, respectively.

From 1978 to 1980, he was an Assistant Professor in the Department of Electrical Engineering at the University of Southern California (USC), Los Angeles. He has been on the Stanford faculty since 1981, where he is currently the Hitachi America Professor in the School of Engineering and Chairman of the Department of Electrical Engineering. He has played key roles in several semiconductor, EDA, and biotechnology startup companies. His research interest and contributions are in the areas of network information theory, imaging sensors and systems, and configurable integrated circuits. He has authored or coauthored over 200 papers and 30 patents in these areas. He is coauthor of the book *Network Information Theory* (Cambridge University Press, 2011).

Prof. El Gamal has won several honors and awards, including the 2012 Claude E. Shannon Award and the 2004 Infocom Best Paper Award. He has served on the technical program committees of many international conferences, including ISSCC, ISIT, and the International Symposium on FPGAs. He has been on the Board of Governors of the IT Society and currently serves as Vice President.