CMOS IMAGE SENSOR DESIGN WITH PROGRAMMABLE SPATIAL-TEMPORAL EXPOSURE FOR MACHINE VISION AND COMPUTATIONAL IMAGING APPLICATIONS

by

YI LUO

B.A.Sc, University of Regina, 2012 M.A.Sc., The University of British Columbia, 2015

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

in

THE FACULTY OF GRADUATE AND POSTDOCTORAL STUDIES

(Electrical and Computer Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

October 2020

© Yi Luo, 2020

The following individuals certify that they have read, and recommend to the Faculty of Graduate and Postdoctoral Studies for acceptance, the dissertation entitled:

CMOS Image Sensor Design with Programmable Spatial-Temporal Exposure for Machine Vision and Computational Imaging Applications

submitted by	Yi Luo	in partial fulfillment of the requirements for
the degree of	Doctor of Philosophy	
in	Electrical and Computer Enginee	ring
Examining Committee:		
Prof. Shahriar	Mirabbasi, Electrical and Compute	er Engineering
Supervisor		
Prof. Robert R	ohling, Electrical and Computer E	ngineering
Supervisory C	ommittee Member	
Prof. Mieszko	Lis, Electrical and Computer Engi	neering
Supervisory C	ommittee Member	
Prof. Jim Little	e, Computer Science	
University Exa	aminer	

Prof. Panos Nasiopoulos, Electrical and Computer Engineering University Examiner

Abstract

Compressive sensing, as one of computational imaging techniques, employs exposure encoding of cameras. Currently, as coded exposure is not supported monolithically on image sensors, computational cameras rely on discrete optical modulators to implement compressive sensing. In this thesis, we propose image sensor designs that are capable of per-frame spatial-temporal exposure encoding. We propose merging exposure-programmable pixels, which consist of charge modulators and exposure-code memory, into the imager design. Through pixel-wise exposure manipulation in every frame of image capture, compressive sensing and its related imaging applications are extended to the sensor node with significant benefits of high optical throughput, improved power efficiency, and compact footprint.

In the design of exposure-programmable pixels, four types of pixel architectures are proposed. The capacitive-transimpedance-amplifier-based pixels are advantageous in sensitivity and chargetransfer speed, while the other two which use active-pixel-sensor-based structures offer a more compact size and circuit simplicity. To exploit the full potential of proposed pixel designs, the image sensor architecture is correspondingly modified as compared to the conventional image sensor designs.

To evaluate the feasibility and performance of the proposed designs, two prototype image sensors are fabricated in a CMOS process. From experimental results, both conventional non-intermittent exposure and per-frame spatial-temporal coded exposure are verified. In demonstration of on-chip compressive sensing applications, two examples from high-speed imaging and compressive focal-stack depth sensing are presented. By performing compressive sensing at the sensor level, the CMOS image sensor designs introduced in this work further pave ...

the way to on-chip computational imaging and facilitate implementation of many emerging applications in the machine vision paradigm.

Lay Summary

Computational imaging is an emerging field in machine vision. One of the computational imaging techniques is compressive sensing, which processes scene information with only a few image captures. Compressive sensing is usually implemented through camera coded exposure which is implemented off-chip by an optical modulator device placed in front of the camera lens. For applications where a small camera size is important, however, the use of optical modulator devices is problematic due to their calibration difficulties and large packaging size.

In this research, an on-chip solution for implementing compressive sensing is presented. The proposed image sensor design is capable of per-frame pixel-wise coded exposure, which facilitates implementation of compressive sensing related imaging applications on the image sensor. With a compact device packaging and favorable power efficiency, the image sensor designs presented in this thesis pave the way for on-chip implementation of computational imaging techniques for emerging machine vision applications.

Preface

This research work is conducted in the Department of Electrical and Computer Engineering in the University of British Columbia (UBC), under the supervision of Prof. Shahriar Mirabbasi. The content of this thesis is mostly based on the publications listed below.

Part of Chapter 1 has been published in the following two conference papers:

- Y. Luo, N. Guo, S. Mirabbasi, and D. Ho, "CMOS Computational Pixel for Binary Temporal Optically Encoded High-Speed Imaging," *in proceedings of the IEEE International Conference on Electron Devices and Solid-State Circuits*, Hong Kong, August 2016, pp. 44-47.
- Y. Luo, S. Mirabbasi, and D. Ho, "Bidirectional Multi-Level Spatial Coded Exposure CMOS Capacitance TIA Pixel Design," *in proceedings of the IEEE International Conference on Electron Devices and Solid-State Circuits*, Hong Kong, August 2016, pp. 460-463.

I conceived the idea, conducted the circuit design, performed the simulation, and draft the manuscript. N. Guo provided useful supports on software setup and data collection. D. Ho and S. Mirabbasi assisted in editing of all manuscripts and supervised the project.

Chapter 2 has been published in two conference proceedings and a peer-reviewed journal:

• Y. Luo and S. Mirabbasi, "Always-On CMOS Image Sensor Pixel Design for Pixel-Wise Binary Coded Exposure," *in proceedings of the IEEE International Symposium on Circuits and Systems*, Baltimore, USA, May 2017.

- Y. Luo and S. Mirabbasi, "A CMOS Pixel Design with Binary Space-time Exposure Encoding for Computational Imaging," *in proceedings of the IEEE Custom Integrated Circuits Conference*, Austin, USA, April 2017.
- Y. Luo, D. Ho and S. Mirabbasi, "Exposure-Programmable CMOS Pixel with Selective Charge Storage and Code Memory for Computational Imaging," *IEEE Transactions on Circuits and Systems I: Regular Papers*, Vol. 65(5), pp. 1555-1566, 2018.

I conceived the idea, conducted the circuit design, performed the device-level simulation, tested the fabricated chips, collected experimental data, and draft all manuscripts. D. Ho helped edit the journal manuscript, and S. Mirabbasi assisted to edit all manuscripts and supervised the project.

In Chapter 3, a portion of contents has been published in a journal listed below:

 Y. Luo, J. Jiang, M. Cai and S. Mirabbasi, "CMOS Computational Camera with a Two-Tap Coded Exposure Image Sensor for Single-Shot Spatial-Temporal Compressive Sensing," *Optics Express*, Vol. 27(22), pp. 31475-31489, 2019.

I conceived the idea, conducted the circuit design, performed the device-level simulation, tested the fabricated chips, collected experiment data, and draft all manuscripts. J. Jiang and M. Cai helped with test setup and testing. S. Mirabbasi assisted in editing all the manuscripts and supervised the project.

Most of the material in Chapter 4 is being submitted to a journal.

Table of Contents

Abstractiii
Lay Summaryv
Prefacevi
Table of Contents viii
List of Tables xi
List of Figures xii
List of Abbreviations xviii
Acknowledgementsxx
Dedication xxi
Chapter 1: Introduction1
1.1 Camere coded exposure and the hardware implementation
1.1.1 Types of camera coded exposure
1.1.2 Typical hardware implementation of coded exposure
1.2 Implementation of coded exposure on image sensor chips
1.3 Thesis outline
1.3.1 Objectives
1.3.2 Thesis organization
Chapter 2: Design of exposure-programmable pixels9
2.1 In-pixel charge modulator and exposure-code memory
2.1.1 LEFM and CTIA charge modulators
2.1.2 DRAM and SRAM based exposure-code memory

2.2	Circuit implementation of exposure-programmable pixels	15
2.2.1	CTIA charge modulator and DRAM exposure-code memory	15
2.2.2	LEFM charge modulator and DRAM exposure-code memory	30
2.2.3	CTIA charge modulator and SRAM exposure-code memory	37
2.2.4	LEFM charge modulator and SRAM exposure-code memory	39
2.2 0	Chapter conclusion	43
Chapter 3	: Computational cameras with coded-exposure image sensors	.45
3.1 A	A coded-exposure CIS with 2-tap CTIA+SRAM pixels	45
3.1.1	Circuitry and architecture design of CIS	45
3.1.2	Camera hardware development and experiemental results	54
3.2 A	A coded-exposure CIS with 2-tap LEFM+DRAM pixels	60
3.2.1	Chip architecture of CIS	60
3.2.2	Camera hardware and measurement results	61
3.3 (Chapter conclusion	66
Chapter 4	: CS-inspired machine vision by coded exposure cameras	.68
4.1 0	Compressive high-speed imaging	68
4.1.1	Camera coded exposure and space-time volume reconstruction	69
4.1.2	Compressive high-speed imaging by the prototype coded exposure camera	71
4.2 0	Compressive focal-stack depth sensing	76
4.2.1	Compressive focal stack and depth from defocused images	77
4.2.2	Compressive focal-stack photography by the prototype coded exposure camera	79
4.3 (Chapter conclusion	80
Chapter 5	: Conclusion and future work	.82
		ix

5.1	Conclusion of this work	82
5.2	Future development and technological advances	
5.2.2	1 Multi-level spatial-temporal coded exposure	85
5.2.2	2 System-on-a-chip machine vision	
Bibliogra	aphy	93
Appendi	ix A: CMOS pixel basics	102
A.1	Basic n+/p-sub photodiode	102
A.2	Operation of a pixel	104
A.3	Passive pixel device	105
A.4	Active pixel device	106
A.5	Pixel performance evaluation	110
Appendi	ix B: CMOS Image Sensor Architecture	113
B.1	Sensor-level signal processing	113
B.2	Column-parallel signal processing	115
B.3	Pixel-level signal processing	115

List of Tables

Table 1.1 A comparison between different hardware implementation of coded exposure
Table 3.1 The exposure code arrangements for valve switches M1 and M2 operations
Table 3.2 The exposure code arrangements for valve switches M_1 and M_2 operations of the
proposed pixel
Table 3.3 Comparison between different proposed exposure-programmable pixel designs
Table 4.1 Comparison between two implemented CIS chips and other related CIS designs 67

List of Figures

Figure 1.1 Types of CIS exposure: (a) conventional non-intermittent exposure, (b) temporal coded
exposure, and (c) pixel-wise spatial-temporal coded exposure and its corresponding (d)
time diagram
Figure 1.2 Types of SLM devices and their hardware implementation setups: (a) Working principle
of the transmissive LCD, LCoS, and DMD devices. (b) Cross-sectional view of
computational camera systems using transmissive LCD (up) and LCoS (bottom) 4
Figure 2.1 Block diagrams of pixel architecture of (a) a conventional APS pixel, and (b) the
proposed exposure-programmable pixel 10
Figure 2.2 (a) Floorplan of a 2-tap LEFM pixel. (b) Electrostatic potential diagram for different
control codes applied to both gates
Figure 2.3 (a) Structural diagram of a typical CTIA pixel. (b) Proposed 2-tap CTIA pixel capable
of selective charge integration13
Figure 2.4 (a) Structural diagram of a typical CTIA pixel. (b) Proposed 2-tap CTIA pixel capable
of selective charge integration14
Figure 2.5 Circuit diagram of the proposed two-tap CTIA + DRAM pixel. The CTIA charge
modulator selectively integrate charges. Two DRAM units store the exposure codes
guiding the operation of M_1 and M_2
Figure 2.6 (a) Time diagram of a pixel under spatial-temporal coded exposure in a frame period.
(b) Implemented test chip architecture

Figure 2.7 (a) Layout of the proposed exposure programmable pixel in size of 12.1 μ m×12.2 μ m.
(b) The micrograph of the prototype image sensor showing the test pixel array and
peripheral control circuits
Figure 2.8 Oscilloscope plot of a pixel under non-intermittent exposure. The output voltage level
continuously increases during the exposure period
Figure 2.9 (a) Code control signals applied for temporal coded exposure. (b) Output signals of a
pixel under temporal coded exposure. Output voltages increase intermittently according
to the exposure codes applied
Figure 2.10 Pixel output arrays. (a) V_{out1} array under a dark environment without exposure
encoding. (b) V_{out1} array under non-intermittent exposure. (c) V_{out1} array under temporal
coded exposure. (d) V_{out2} array under temporal codec exposure
Figure 2.11 (a) Control signals for the spatial-temporal coded exposure of a pixel. (b) Outputs of
a pixel from spatial-temporal coded exposure
Figure 2.12 Spatial-temporal coded exposure test with different exposure-code masks. (a) The
spatial exposure-code mask for CTIA1. (b) The spatial exposure-code mask for CTIA2.
The corresponding pixel V_{out1} array (c) and the pixel V_{out2} array (d). (e) 10 grey-scale
exposure-code masks delivered by φ_{code1} . The V _{out1} array (f) and the V _{out2} array (g) from
applying exposure-code masks described in (e). The V_{out1} array (h) and V_{out2} array (i)
from applying 10 pseudo-random binary spatial exposure code masks
Figure 2.13 (a) SNR of the proposed pixel as a function of the illumination intensity. (b) Power
consumption of the pixel array in different types of coded exposure
Figure 2.14 The circuit diagram of proposed two-tap LEFM + DRAM pixel design. Capacitor C_1
and C_2 are for charge storage while two DRAM units store the exposure codes 30
xiii

Figure 2.15 The time diagram of (a) temporal coded exposure and (b) spatial-temporal coded
exposure of the proposed LEFM+DRAM pixel design
Figure 2.16 (a) Implemented test chip architecture. (b) Layout of the proposed LEFM+DRAM
pixel. (c) Chip micrograph of the test image sensor
Figure 2.17 (a) Plot of a pixel under non-intermittent exposure. (b) Plot of a pixel under temporal
coded exposure. Output voltage levels decrease intermittently according to the exposure
codes applied
Figure 2.18 (a) Plot of a pixel under spatial-temporal coded exposure. (b) The resultant pixel array
outputs after applied pseudo-random binary exposure-code masks
Figure 2.19 The circuit diagram of proposed two-tap CTIA + SRAM pixel design. Capacitor C_1
and C_2 are for charge storage while a SRAM cell stores the exposure code
Figure 2.20 Time diagram of spatial-temporal coded exposure of the proposed CTIA+SRAM pixel
design
Figure 2.21 The circuit diagram of proposed two-tap LEFM + SRAM pixel design. Capacitor C_1
and C ₂ are for charge storage while a SRAM cell stores the exposure code 40
Figure 2.22 The time diagram of (a) temporal coded exposure and (b) spatial-temporal coded
exposure of the proposed LEFM+SRAM pixel design
Figure 3.1 Circuit diagram of the implemented exposure-programmable pixel
Figure 3.2 Time diagram of a pixel operation under spatial-temporal coded exposure in a frame
period
Figure 3.3 Block diagram of the first CIS. All coded exposure related blocks are highlighted in
grey

Figure 3.4 Circuit diagram of the coded-exposure related functional blocks
Figure 3.5 Time diagram of the process of exposure-code delivery. Exposure codes are streamed
into de-serializers through 12 channels and shipped to pixels in row-by-row basis 48
Figure 3.6 Circuit diagram of the column-wise signal processing slice
Figure 3.7 Time diagram of the column-wise signal processing slice during a readout period51
Figure 3.8 Conceptual and circuit diagram of the adaptive SS-ADC enabled by the output slope
adjustment of the ramp generator
Figure 3.9 Micrograph of a prototype CIS chip. The die size is 4.5 mm × 4.0 mm including bonding
pads54
Figure 3.10 (a) Block diagram of the camera system for compressive focal-stack imaging tests. (b)
The camera hardware platform which holds the prototype CIS chip and the VCM auto-
focus lens
Figure 3.11 The DNL, INL, and spectrum plots of the SS-ADC
Figure 3.12 Exposure code mask patterns and experimentally captured images using non-
intermittent exposure and spatial-temporal coded exposure
Figure 3.13 (a) Chip power consumption with different code composition and SS-ADC ramp
slopes in a frame period containing 5 exposure code masks. (b) Chip power consumption
with different number of exposure code masks in a frame period
Figure 3.14 The block diagram of chip architecture of the second CIS
Figure 3.15 Chip micrograph. The second chip is in a size of 3 mm \times 3 mm
Figure 3.16 (a) Fabricated CMOS image sensor. (b) Prototype computational camera system 63
Figure 3.17 Camera output images of (a) a non-intermittent exposure test, (b) a spatial-temporal
coded exposure test using 128 column-grey-scale masks
XV

- Figure 4.6 The experimental result of the proposed on-chip compressive focal-stack depth sensing.

Figure 5.5 (a) A potential pixel design accepts both positive and negative exposure codes for multi-
level coded exposure. (b) Time diagram of multi-level coded exposure with different
exposure code polarities
Figure 5.6 An example of potential CIS-SoC design using BSI technology and multi-wafer
stacking for emerging machine vision applications
Figure A.1 A n+/p-sub photodiode design: (a) Structural diagram. (b) Circuit diagram
Figure A.2 (a) A basic CMOS pixel cell. (b) Time diagram of pixel operation 104
Figure A.3 A typical setup of a PPS pixel design
Figure A.4 An example of 3-T APS pixel design
Figure A.5 (a) A CMOS pinned photodiode. (b) Circuit diagram of 4-T APS pixel design. (c)
Charge transfer from the pinned-photodiode to FD
Figure B.1 A CMOS image sensor architecture with sensor-level signal processing. The signal
processing module is implemented either on-chip or off-chip
Figure B.2 A CMOS image sensor architecture with column-parallel signal processing 114
Figure B.3 A CMOS image sensor architecture with pixel-level signal processing 115

List of Abbreviations

3D	Three dimensions
4D	Four dimensions
ADC	Analog-to-digital convertor
AI	Artificial intelligence
APS	Active-pixel sensor
ASIC	Application specific integrated circuit
BSI	Backside illumination
CDS	Correlated double sampling
CMOS	Complementary metal-oxide semiconductor
CCD	Charge-coupled device
CIS	CMOS image sensor
CS	Compressive sensing
CAPD	Current assisted photon demodulator
CPU	Central processing unit
CTIA	Capacitive transimpedance amplifier
DAC	Digital-to-analog convertor
DDR	Discrete double data
DFF	Depth from focus
DFD	Depth from defocus
DMD	Digital micro-mirror
DNL	Differential nonlinearity
DRAM	Dynamic random-access memory
DSP	Digital signal processor
FD	Floating diffusion
ENOB	Effective number of bits
FPGA	Field-programmable gate array
FPN	Fixed pattern noise
GBW	Gain bandwidth
GPU	Graphics processing unit
HDR	High dynamic range
IC	Integrated circuit
ISP	Image signal processor
INL	Integral nonlinearity
LCD	Liquid-crystal display
LED	Light-emitting diode
LEFM	Lateral-electric-field modulator
LCoS	Liquid-crystal on silicon
LSB	Least significant bit
MIM	Metal insulator metal
OTA	Operational transconductance amplifier
PDK	Process design kit
PGA	Programmable gain amplifier

Photonic mixer device
Photodiode
Phase-locked loop
Peak signal-to-noise ratio
Region of interest
Successive-approximation register
Spatial-light modulator
Signal-to-noise and distortion ratio
Storage gate
System on chip
Single slope
Static-random-access memory
Synchronous dynamic random-access memory
Time of flight
Universal serial bus
Vocal-coil motor
Very-large-scale integration

Acknowledgements

I would like to thank my parents for their endless love and enormous support throughout my voyage in the boundless sea of learning. Without them, I could not have this thesis work accomplished in Canada.

I would like to thank my supervisor, Prof. Shahriar Mirabbasi, for his guidance and academic supports throughout my entire Ph.D. program. He is a warm-hearted and knowledgeable professor, and one of the smartest people I admire. I particularly appreciate the research freedom, the research internship in Hong Kong, and the plenty of fabrication opportunities he provided, which had been important experiences to me and will continuously influence my future career.

I acknowledge the University of British Columbia for providing stable hardware supports (office and chip test facilities) throughout my Ph.D. studies.

I also acknowledge the Natural Sciences and Engineering Research Council of Canada (NSERC) and Huawei Canada for their supports for this research.

I also would like to thank my lab mates from the System-on-a-Chip lab, and colleagues from IEEE Solid-State Circuit Society and IEEE Circuit and System Society for their academic advises, paper publication supports, and precious friendship. To my parents for their unconditional love and spiritual solicitude. To those who loved this world and knew friendly company therein

Chapter 1: Introduction

Digital cameras have become ubiquitous in the past decade, primarily due to their widespread adoption in consumer electronic products. Many advances in camera design have been achieved by using high-resolution complementary metal-oxide semiconductor (CMOS) image sensors which result in compact size packaging, low power and low-cost products. To offer advanced imaging capabilities such as high-speed imaging and depth ranging, a new type of technology called "computational cameras" are developed to achieve superior imaging performance that cannot be achieved using conventional image acquisition methods. Unlike conventional off-theshelf digital cameras that perform image capture through non-intermittent exposure of image sensors placed under standard lenses, computational cameras employ optical encoding (modulation) of scene light followed by computational decoding (demodulation). Owing to the rapid advancement in computer vision, recent computational cameras represent the gateway to novel photographic functionalities. For example, light-field cameras (also called plenoptic cameras) record a 4D light field used for image refocus and perspective change [1] - [5]. Catadioptric cameras capture omnidirectional field of view of the scene to produce immersive experience images [6] - [8]. Depth ranging cameras measure object structure and distance to synthesize 3D volume models of the scene [9] - [14]. With the on-going development of optical encoding techniques, the clan of computational cameras expands to offer unprecedented photographic experiences.

One of the most active areas of research in the optical encoding paradigm is compressive sensing (CS). Through exploiting the intrinsic redundancy of a scene, cameras acquire visual information in a compressive approach (far fewer samples than required by the Shannon-Nyquist sampling theorem) to produce images with high fidelity after sparse reconstructions [15]. From recently published reports, CS has been adopted by several computational camera systems developed to achieve imaging capabilities such as high-speed imaging [16] - [19], high-dynamic range (HDR) imaging [20] – [24], and depth sensing [25] – [28]. The most common method to implement CS in a computational camera system is coded exposure. Unlike coded aperture which programs camera's lens placed on the pupil plane [29], coded exposure modulates the exposure of image sensors during the camera's exposure period. In this chapter, an overview of camera coded exposure and several typical hardware setups are presented in Section 1.1. In Section 1.2, some of the state-of-the-art on-chip implementations of coded exposure are summarized and their advantages and practical issues are discussed. This chapter is concluded in Section 1.3, which presents the outline of this thesis.

1.1 Camera Coded Exposure and the Hardware Implementation

1.1.1 Types of Camera Coded Exposure

Coded exposure was initially introduced to resolve images with motion blur created by fastmoving objects [30]. It is also called "flutter shutter" which modulates camera exposure through programming the shutter device placed in front of the image sensor. Through open and close operation of the shutter during the exposure period, high-frequency details in the scene are preserved as the coded exposure acts like a broad-band filter [16]. Currently, the definition and the working principle of camera coded exposure are expanded to space and time domains. Depicted in Fig. 1.1 is a comparison between conventional camera exposure and two types of coded exposure. In conventional image capture approach, the image sensor is continuously exposed (Fig. 1.1(a)). In temporal coded exposure, the scene light reaching the image sensors is temporally modulated (Fig. 1.1(b)). For all pixels in a pixel array, they all experience intermittent exposure according to the same sequence. If exposure programming in the spatial domain is added, the spatial-temporal coded exposure is achieved which is shown in Fig. 1.1(c). The working principle of spatial-temporal coded exposure is pixel-wise exposure manipulation (Fig. 1.1(d)). In every frame (T_{frame}), the exposure period (T_{expo}) is consisted of individual sub-period (T_{chop}) which defines the duration of an exposure code mask. With code pattern differences between exposure code masks, every pixel in the pixel array is exposed by a unique exposure code sequence before reaching the readout period (T_{read}).



Figure 1.1 Types of CIS exposure: (a) conventional non-intermittent exposure, (b) temporal coded exposure, and (c) spatial-temporal coded exposure and its corresponding (d) time diagram.

1.1.2 Typical Hardware Implementation of Coded Exposure

The implementation of coded exposure requires supports from camera hardware. Since on-chip pixel-wise optical encoding control was not available in the off-the-shelf image sensors, coded exposure has been typically realized off-chip by implementing emulated imaging systems using

three types of spatial-light modulators (SLMs) (Fig. 1.2(a)). The transmissive liquid-crystal display (LCD) blocks scene light by controlling pixel transparency [31]. The liquid-crystal on silicon (LCoS) device polarizes the incoming P-polarized light to S-polarized light [32]. The digital micro-mirror (DMD) device consists of reflective micro-mirrors, which reflects incoming natural lights to different directions [33]. Figure 1.2(b) shows typical system setups of computational cameras. For coded exposure, the transmissive LCD is mounted in front of the image sensor. This LCD is controlled by a micro-controller and working as an electrical shutter programed by a binary coded sequence while triggered by a hot-shoe signal from the camera [16]. In contrast, the LCoS device modulates the incoming scene light to S- and P- type polarized light using a light splitter. Unlike the transmissive LCD, the LCoS selectively polarizes the S-polarized light to P-polarized light, which is reflected by the splitter to image senor. The remaining S-polarized light is reflected by the LCoS to the splitter without reaching the sensor [34].



Figure 1.2 Types of SLM devices and their hardware implementation setups: (a) Working principle of the transmissive LCD, LCoS, and DMD devices. (b) Cross-sectional view of computational camera systems using transmissive LCD (up) and LCoS (bottom).

Comparing the abovementioned SLM devices, the transmissive LCD is inexpensive and easy to control, but it has low light throughput and contrast since its driving circuits are located between the liquid crystal pixels. Reflective LCoS devices have higher fill factors and contrast ratio, but they require a polarizer which significantly reduces the light throughput. DMD devices provide high contrast with less loss on light throughput. However, their resolution (the size of pixel array) is much smaller than that of image sensors, resulting in a worse exposure programming in spatial domain. In addition, computational cameras using SLM devices need a combination of optical lens and driving circuits, resulting in complications in calibration and additional power consumption. Their large sizes severely limit their applications to portable, wearable, and biomedical electronics.

To achieve high image quality, low power consumption, and small footprint, emerging and future computational cameras need a practical solution to directly implement coded exposure on image sensor chip. Table 1.1 compares the SLM based computational camera system to an on-chip solution. Integrating coded exposure feature into the image sensor chip brings superior performance in all perspectives. However, the critical enabler is the ability to realize per-pixel exposure manipulation on sensor plane (focal plane).

	Trasmissive LCD	LCoS	DMD	Image Sensor		
Light throughput	< 50%	~50%	> 50%	~100%		
Contrast	Low	Medium	High	High		
Light polarization	No	Yes	No	No		
Pixel-wise control	Difficult	Yes	Yes	Yes		
Cost	Low	Medium	High	Low		
Power	High (need	Low				
Package size	Large (N	Small				
Mechanical robustness	Low (M	High				

Table 1.1	A	comparison	between	different	t solution	ns to l	hard	lware i	mpl	lementati	ions of	cod	led	exposur	·e
-----------	---	------------	---------	-----------	------------	---------	------	---------	-----	-----------	---------	-----	-----	---------	----

1.2 Implementation of Coded Exposure on Image Sensor Chips

With the background stated in the previous section, to enable coded exposure on the sensor plane, several image sensor designs with embedded pixel exposure control techniques have been recently reported by integrated circuit (IC) research groups worldwide. R. Schwarte et al. introduced a photonic mixer device (PMD) [35], which became a cornerstone of many emerging imaging applications like indirect time-of-flight (ToF) depth ranging [36] - [38]. Pixels were functional as current-assisted photon demodulators (CAPDs) which guided photo-current flow to either of two readout diodes. Following such 2-tap charge modulation mechanism, Wan et al. proposed a multibucket pixel design utilizing multiple storage gates (SGs) to store generated charges [39]. Temporal coded exposure was implemented by selectively transferring charges to designated SG units. Since the pixel design contains no storage blocks for SG control signals (exposure codes), spatial exposure encoding has not been supported. Mochizuki et al., also realized temporal exposure encoding with the multi-bucket pixel design. Additionally, they introduced 15 aperture memory cells to store the exposure codes for SG switching [40]. In the recent development, Y. Shirakawa et al. demonstrated an 8-tap multi-bucket pixel design for ToF based three-dimension (3D) sensing applications [41]. However, the aperture memory cells are off-pixel design, making spatial coded exposure impossible as all pixels still share the same exposure codes. Zhang et al. introduced a pixel design that included an in-pixel static-random-access memory (SRAM) unit for exposure codes storage [42]. Each pixel is guided by its unique exposure code during the exposure period, hence the spatial coded exposure is possible. However, due to the lack of in-pixel charge selective integration, the temporal coded exposure cannot be achieved in a frame. When applying a sequence of exposure codes, the temporal exposure encoding requires a pixel readout operation for each exposure code. Thus, these cameras require high readout rates, and the exposure code length is limited by the pixel readout speed.

In conclusion, although abovementioned works are instrumental in advancing computational imaging, none of them offers an efficient solution for the realization of spatial-temporal coded exposure capabilities on image sensors. Therefore, research on this topic continues to create a practical solution to bring spatial-temporal coded exposure and compressive sensing related applications to the sensor node.

1.3 Thesis Outline

1.3.1 Objectives

The primary objective of this research work is to develop an efficient image sensor capable of spatial-temporal coded exposure. Starting from the pixel design, the proposed image sensor architecture includes new circuits and blocks for pixel exposure control and exposure-code delivery. Also, the signal processing block in the image sensor has been optimized to handle pixel outputs generated from either non-intermittent exposure or coded exposure. After chip tape-out, the fabricated image sensors are used to build prototype computational camera systems to validate the effectiveness of spatial-temporal coded exposure. In addition, compressive sensing based imaging applications are also examined by the prototype camera to demonstrate its capability and potential in the machine vision paradigm.

1.3.2 Thesis Organization

The organization of this thesis is as follows: Chapter 2 focuses on pixel design methodology. Four types of pixel structures are proposed to achieve exposure programming with their performance

measured through separate test chips. In Chapter 3, comprehensive image sensor architectures are discussed. For each of the proposed pixel designs, a corresponding image sensor is designed, fabricated, and characterized to verify the feasibility of on-chip coded exposure. Chapter 4 introduces the prototype computational camera hardware equipped with the developed image sensors. The capability of on-chip compressive sensing is demonstrated by applying two computational imaging applications on the prototype camera. Finally, in Chapter 5, the conclusion of this research work and future research outlooks are provided.

Chapter 2: Design of Exposure-Programmable Pixels

In order to achieve spatial-temporal coded exposure on image sensors, the critical enabler is a pixel design to realize both selective charge integration and exposure code storage. As previously stated in Chapter 1, in coded exposure, the exposure of each pixel needs to be electronically switchable. Within an exposure period, the pixel is 'guided-exposed' by a sequence of exposure codes. Therefore, charges generated by the photodetector are selectively transferred to different in-pixel charge storage units before the pixel is read out. When perform coded exposure in spatial domain, pixels also need an additional exposure code storage feature. Each pixel performs its own exposure encoding according to the stored unique exposure codes, which can be distinct from that of other pixels on the array.

The design of exposure-programmable pixel starts with a review of conventional pixel architectures utilized by present image sensors (see Appendix A). Since most of image sensor designs have switched from charge-coupled device (CCD) structures to a CMOS basis, the backbone of pixels are circuit components available in modern CMOS fabrication processes. Depicted in Fig. 2.1 are block diagrams of conventional active-pixel sensor (APS) pixel adopt in classical CMOS image sensors (CISs), and the proposed exposure-programmable pixel capable of selective charge storage and exposure-code memorization. In APS pixels, generated charges are stored on the photodetector's self-capacitor for the entire exposure period (Fig. 2.1(a)). When the readout process starts, charges are transferred to a readout buffer (floating diffusion) and are typically read out using the correlated double sampling (CDS) scheme. Since charge generation and its integration are performed within the photodetector, it is impossible to implement charge selection unless an off-sensor SLM blocks the light path and stops the charge generation process.



Figure 2.1 Block diagrams of pixel architecture of (a) a conventional APS pixel, and (b) the proposed exposureprogrammable pixel.

To enable charge selection, the proposed pixel structure integrates two extra charge storage units as a modulator (Fig. 2.1(b)). Within an exposure period, the photodetector generated charges are selectively transferred to one of the storage units according to the applied exposure code. Once the exposure is done, during the pixel readout process, only the charge storage unit that contains the desired charges is read out. Charges stored in the other charge storage unit are discarded when the pixel is reset. Each charge storage unit is associated with a code-storage memory block. When one applies spatial-temporal coded exposure, the selective charge transfer in each pixel is guided by its unique exposure codes that are stored in the code storage blocks.

For the mentioned selective charge modulator and exposure code memory, they can be implemented by various functional circuits. In the rest of this chapter, combinations of two types of charge modulator and exposure code memory are introduced with detailed analysis.

2.1 In-pixel charge modulator and exposure-code memory

This section discusses candidate circuits for the charge modulator and the exposure-code memory. Among vast amount of analog and mixed-signal circuit designs, two types of charge integration circuits are considered competent to be the charge modulator: lateral-electric-field modulator (LEFM), and capacitive transimpedance amplifier (CTIA). For the exposure-code memory selection, dynamic random-access memory (DRAM) and static random-access memory (SRAM) are suggested duo to their mature designs after decades of development in digital IC.

2.1.1 LEFM and CTIA charge modulators

The idea of LEFM was introduced by S. Kawahito *et al.* to increase time resolution of image sensors in imaging applications like fluorescence lifetime imaging microscopy and indirect ToF [43] – [45]. Based on a classical 4-T APS design, the LEFM structure contains a photodiode (usually a pinned photodiode) connected to multiple charge storage drains through corresponding draining gates. Shown in Fig. 2.2 is a typical design of a 2-tap LEFM pixel. The photodiode is connected to two charge drains through two code-control valve gates (Fig. 2.2(a)). When different 1-bit binary codes are applied, charges generated by the photodiode flow into corresponding charge drain. During pixel readout, charges buffered in both charge drains are converted into voltage levels and streamed out of pixel for further signal processing. To implement selective charge storage, the valve gates receive opposite control codes (complementary binary codes). When a valve gate is applied with code "1", the other one receives code "0". During an exposure period, the binary code signal "1" opens the valve gate and generated charges flow into the charge drain (Fig. 2.2(b)). The code signal "0", on the other hand, closes the valve gate by creating a potential barrier, preventing charges flow to the charge drain. Through programming the sequence

of control codes, charges generated in the exposure period are selectively distributed into different charge drains, completing the charge modulation.



Figure 2.2 (a) Floorplan of a 2-tap LEFM pixel. (b) Electrostatic potential diagram for different control codes applied to both gates.

Another candidate to the charge modulator is CTIA, which is popular in high-speed image sensor designs for applications such as biomedical fluorescent imaging and micro-computed tomography [46] – [48]. Unlike the LEFM pixel which is modified based on an APS backbone, a CTIA pixel relies on an operational transconductance amplifier (OTA). Illustrated in Fig. 2.3(a) is a typical CTIA pixel design consists of an OTA and a capacitor (C_{int}) for charge integration. During an exposure period, the photodiode is held at a constant voltage. All generated charges are sent out of the photodiode and accumulated at C_{int} . By tuning the capacitance of C_{int} , the pixel sensitivity various. Another noticeable advantage of CTIA structure is the fast charge-transfer speed due to the OTA. When under weak illumination, the OTA assists pixels to settle in shorter time.



Figure 2.3 (a) Structural diagram of a typical CTIA pixel. (b) Proposed 2-tap CTIA pixel capable of selective charge integration.

As CTIA pixels are never been employed to practice coded exposure, a modified CTIA pixel architecture is proposed in this research to enable selective charge storage. As shown in Fig. 2.3(b), the proposed CTIA pixel contains two CTIA structures connecting to a photodiode. Like LEFM pixels, generated charges are selectively integrated on either C_{int1} or C_{int2} based on the control codes applied to the valve switches S_1 and S_2 . In comparison to the LEFM pixel design, the proposed pixel owns all benefits from using the CTIA structure. The circuit complication and layout difficulties, however, result in control complexity (needs analog bias) and large pixel pitch.

2.1.2 DRAM and SRAM based exposure-code memory

After decades of development in digital IC, DRAM and SRAM have become mainstream volatile memory solutions to the vast majority of electronic products on the market. In modern IC design flow, DRAM and SRAM are incorporated into process design kit (PDK) as standard cells. A DRAM cell is formed by a capacitor controlled by a transistor. Its structure simplicity is in fever of consuming electronics where low-cost and high-capacity memory are required. SRAM cells, on the other hand, consists of six transistors which is more expensive and occupying larger silicon space than that of DRAM. They are commonly employed in applications where high speed and low power consumption matters the most. To merge DRAM and SRAM into the proposed exposure-programmable pixel, their reading and writing mechanisms are carefully studied. Depicted in Fig. 2.4 are the proposed exposure-code memory solutions using DRAM and SRAM, respectively. The DRAM based solution includes two 1-bit DRAM cells to memorize exposure codes for both valve switches (S₁ and S₂). When the DRAM writing signal is asserted high, exposure codes access to the storage capacitors C₁ and C₂ through switches M₁ and M₂ (Fig. 2.4(a)). After DRAM cells writing/refreshment, the switching operations of S₁ and S₂ are guided by the exposure code signals (Q and Q') stored in corresponding DRAM cells. Noticed that Q and Q' are complementary to each other to ensure charges generated by the photodiode (PD) always flow into either of two charge storage units. In the SRAM solution, by taking the advantage of the internal cross-coupled inverters, only one SRAM cell is needed (Fig. 2.4(b)). The writing operation of the SRAM cell is through closing switches M₃ and M₄. Unlike DRAM cells, there is no need to frequently refresh the SRAM cell to maintain contents stored inside.



Figure 2.4 Structural diagrams of (a) DRAM based exposure-code memory, and (b) SRAM based exposure-code memory.

2.2 Circuit implementation of exposure-programmable pixels

With those abovementioned charge modulators and exposure-code memory designs, four types of Exposure-programmable pixels are proposed through their cross-combinations. In the rest of this section, each type of pixels is introduced with detailed circuit implementation.

2.2.1 CTIA charge modulator and DRAM exposure code memory

In the combination of CTIA charge modulator and DRAM exposure-code memory, two CTIA structures and two DRAM cells are included in the pixel design. Shown in Fig. 2.5 is the circuit diagram of proposed exposure-programmable pixel. Each CTIA consists of a discrete integration capacitor for charge accumulation, and a single-ended cascode amplifier.



Figure 2.5 Circuit diagram of the proposed two-tap CTIA + DRAM pixel. The CTIA charge modulator selectively integrate charges. Two DRAM units store the exposure codes guiding the operation of M₁ and M₂.

The photodiode, PD, is connected to two CTIA blocks via switches M_1 and M_2 . Capacitors C_1 are C_2 are used for charge integration and can be reset by closing switches M_3 and M_4 . The pixel
exposure period is divided into charge transfer time chops (T_{chop}). In each T_{chop} , charges generated by the PD are transferred to either C_1 or C_2 depending on whether M_1 or M_2 is closed. The amplifier, biased by V_{bp} and V_{bn} , holds its input node to prevent the charge integration in PD. Therefore, generated charges solely accumulate on C_1 or C_2 during the exposure period. For the operational transconductance amplifier (OTA) in the CTIA structure to settle within T_{chop} , the minimum requirement on its gain-bandwidth product is defined as [49]:

$$GBW = \frac{5}{\pi T_{chop}} \cdot \frac{C_{int} + C_{pd}}{C_{int}}$$
(2.1)

where C_{int} represents the integration capacitor (C_1 or C_2). C_{pd} donates the photodiode selfcapacitance. To read out the pixel, M_7 and M_8 are closed when signal φ_{sel} is pulled up. The output signals, $V_{out1}(t)$ and $V_{out2}(t)$, reflecting the voltage level change during a time period t, can be calculated from writing the Kirkhoff's current law (KCL) at the input node of the CTIA [46]:

$$I_{pd}(t) = [V_{out(1,2)}(t) - \frac{V_{out(1,2)}(t)}{A_g}] \frac{C_{int}}{t} - \frac{V_{out(1,2)}(t) C_{pd}}{A_g t}$$
(2.2)

which simplifies to:

$$V_{out(1,2)}(t) = \frac{I_{pd}t}{C_{int}(1 - \frac{C_{int} + C_{pd}}{A_{g}C_{int}})}$$
(2.3)

where A_g denotes the amplifier gain. In consideration of OTA offset, dark current, and leakage currents from the reset switch and the pixel selection switch, the offset voltage V_{off} is added to the output:

$$V_{out(1,2)}(t) = \frac{I_{pd}t}{C_{int}(1 - \frac{C_{int} + C_{pd}}{A_g C_{int}})} + V_{off}$$
(2.4)

16

The switching operation of M_1 and M_2 is controlled by the exposure code signals φ_{code1} and φ_{code2} . As summarized in Table 2.1, different φ_{code1} and φ_{code2} combinations switch M_1 and M_2 on/off accordingly and control the charge transfer direction during the temporal exposure encoding. In spatial exposure encoding, as each pixel stores its unique exposure codes, two DRAM-based exposure code memory cells (DRAM1 and DRAM2) are inserted to store φ_{code1} and φ_{code2} signals. Each DRAM cell consists of a switch (M_5 , M_6), and a capacitor (C_3 , C_4). When the control signal φ_{ctr1} is low, M_1 and M_2 maintain their on/off status according to the signals buffered in C_3 and C_4 irrespective of the status of φ_{code1} and φ_{code2} . Due to the leakage current I_{leak} , charges stored in DRAM cells leak out and cause problematic logic shift. It is essential to refresh DRAM cells to maintain their stored logic level. The maximum DRAM refresh period $T_{refresh,max}$ is defined as:

$$T_{\text{refresh,max}} = \frac{\left(V_{\text{DD}} - V_{\text{th}}\right)\left(C_{\text{DRAM}} + C_{\text{gate}} + C_{\text{drain}}\right)}{I_{\text{leak}}}$$
(2.5)

where V_{th} is the NMOS threshold voltage. C_{DRAM} represents $C_{3, 4}$. C_{gate} , and C_{drain} represent gate capacitance of $M_{1, 2}$, and drain capacitance of $M_{5, 6}$. During the spatial exposure encoding, to ensure an effective storage of exposure codes, T_{chop} must be always smaller than $T_{refresh,max}$.

	Øcode1	Øcode2	Øcode1	Øcode2	∅ code1	∅ code2
Exposure Code	1	0	0	1	Х	Х
Øctrl	1		1		0	
\mathbf{M}_{1}	Close		Open		Determined by DRAM1	
M_2	Open		Close		Determined by DRAM2	

Table 2.1 The exposure code arrangements for valve switches M1 and M2 operations.

When perform spatial-temporal coded exposure, as illustrated in Fig. 2.6(a), the pixel is assigned a unique encoding period (T_{code}) in each T_{chop} for its exposure code update. Starting with

pixel reset (T_{rst}), the voltage level in PD, C₁ and C₂ are reset to V_{rst} . During T_{expo} , once the pixel is being updated, φ_{ctrl} pulls up for a period of T_{code} to let φ_{code1} and φ_{code2} refresh the exposure codes stored in DRAM1 and DRAM2. If other pixels are being updated, the pixel operates according to its stored exposure codes for a period of T_{dram} only, regardless of the changes in φ_{code1} and φ_{code2} . In such manner, within a T_{chop} , all pixels accomplish their exposure codes update. Hence, an exposure-code mask is applied to the pixel array. Prior to the next T_{chop} , the pixel array is spatially exposed based on the applied exposure-code mask. The number of exposure-code masks, M, applied within T_{expo} is calculated as (T_{expo}/T_{chop}). As charges are distributed into CTIA1 and CTIA2, the voltage level at node A (V_A) and node B (V_B) increase. Once T_{expo} ends, φ_{code1} and φ_{code2} stop updating. The pixel selection signal (φ_{sel}) is then pulled up to read out the final output V_{out1} and V_{out2} . Within the readout period (T_{read}), V_A and V_B remain unchanged until φ_{rst} is pulled up again to start the next frame.



Figure 2.6 (a) Time diagram of a pixel under spatial-temporal coded exposure in a frame period. (b) Implemented test chip architecture.

To validate the proposed pixel design, a test image sensor containing 10×10 pixels is fabricated in an 8-metal, 1-poly, $0.13 - \mu m$ CMOS process (V_{DD} = 3.3V). Figure 2.6(b) depicts the chip architecture. The image sensor consists of three decoder-based control modules and a readout scanner. Under global shuttering, at the beginning of a frame, all pixels on the image sensor are reset simultaneously by pulling φ_{rst} up. The code control module provides column-based exposure code delivery, and the DRAM control module performs row-based in-pixel DRAM cells control. To apply temporal exposure encoding, the DRAM control module enables all φ_{ctrl} rows during T_{expo} to permit exposure codes transferred from the code control module to all pixels. To implement spatial exposure encoding, the DRAM control module selects a φ_{ctrl} row during each T_{code} . The code control module, in the meantime, delivers exposure codes to all pixels on the selected row. Within a T_{chop}, all ten rows of the pixel array use their unique T_{code} to update their exposure codes. During T_{read}, all pixels in a column share one output route. The row select module selects pixels in a row-by-row basis. Hence, outputs V_{out1} and V_{out2} of each row are separately presented without overlapping with other rows. All Vout1 and Vout2 are read by a readout buffer formed by a simple NMOS-based source follower. The final outputs are scanned to the I/O pads by a column scanner.

Figure 2.7(a) shows the layout of the proposed pixel, which has a size of 12.1 μ m × 12.2 μ m. The n-well/p-sub type PD occupies an area of 7.1 μ m × 6.8 μ m, giving the pixel a fill factor of 33.2%. All capacitors are metal-insulator-metal (MIM) capacitors, and transistors and routing wires are placed underneath them to save area. Both C₁ and C₂ capacitors are designed in a fixed size of 5.3 μ m × 5.3 μ m, and they have a capacitance of 150 fF. The DRAM capacitors C₃ and C₄ are designed with smaller sizes, occupying an area of 3.0 μ m × 2.0 μ m with a capacitance of 75 fF and giving an estimated T_{refresh,max} of 3.0 ms. The valve transistors M_{1,2} and the reset switch M_{3,4} have an aspect ratio of $(0.4 \,\mu\text{m}/0.4 \,\mu\text{m})$. The PMOS transistors in CTIAs have aspect ratio of $(1.2 \,\mu\text{m}/1.2 \,\mu\text{m})$ so that they can drive the output wires. The CTIA bias current is limited to 0.2 nA, and the simulated open-loop gain and GBW product are 52 dB and ~30 MHz, respectively. The image sensor chip is powered by two independently regulated 3.3 V supplies – one for the pixel array and the other one for all peripheral control modules.



Figure 2.7 (a) Layout of the proposed exposure programmable pixel in size of 12.1 μ m×12.2 μ m. (b) The micrograph of the prototype image sensor showing the test pixel array and peripheral control circuits.

Figure 2.7(b) shows the micrograph of the prototype image sensor chip with all the test structures. Chip control signals are provided by a field-programmable-gate-array (FPGA) based microcontroller (Cyclone V from Altera). Pixel outputs are shown using a 500 MHz digital oscilloscope (RTM100, Rohde & Schwarz). Off-chip biases are provided by a separate programmable power supply (DP832, Rigol Technologies). Similar to off-the-shelf CISs, the default frame rate used for chip characterization is set to 60 fps with an exposure period of 13.28 ms. A red light-emitting diode (LED) centered at 630 nm is calibrated by an optical power meter

(Model 1930-C, Newport) and used as a light source to continuously illuminate the prototype chip. The pixel performance is evaluated in terms of non-intermittent exposure, temporal coded exposure, spatial-temporal coded exposure, noise performance, and power consumption.

• Pixel performance in non-intermittent exposure

Pixels equipped in current CISs experience non-intermittent exposure as the camera's mechanical shutter stays open during the entire exposure period. To perform such non-stop exposure, in configuring the prototype chip, both charge selection and exposure code storage features are deactivated.



Figure 2.8 Oscilloscope plot of a pixel under non-intermittent exposure. The output voltage level continuously increases during the exposure period.

During the time period T_{expo} , φ_{code1} and φ_{ctr1} stay high to transfer generated charges to CTIA1. φ_{code2} , in the meantime, is held low to disable CTIA2. In order to observe the output level shifts of V_{out1} , a row of φ_{se1} is pulled up to high at all times. Shown in Fig. 2.8 is the measured raw output V_{out1} of a pixel in row 5. Prior to the start of T_{expo} , φ_{rst} is set to high to reset the pixel for 3.32 ms, and V_{out1} drops to the 566.0 mV reset voltage level. Once φ_{rst} is low, the photodiode generated charges that are integrated onto CTIA1. During T_{expo} , as charges are continuously accumulated on C₁, V_{out1} keeps increasing until it reaches 2.492 V at the end of T_{expo} . As φ_{rst} goes up again, the pixel is reset and V_{out1} returns to 566 mV for the next frame. Due to the leakage on M₂, the output V_{out2} has a slight change (to 0.562 V) from its base level (566 mV) after the reset in every frame. For a large size of pixel array (e.g. in a VGA resolution), the maximum frame rate is limited by ADC conversion rate. In the presented prototype with the pixel array size of 10 × 10, the frame rate is constrained by pixel access time. By setting the exposure period to the minimum OTA settling time (0.15 µs), the maximum frame rate of the prototype CIS will be 6.0 Mfps.

• Pixel performance in temporal coded exposure

Verification of pixel capability in temporal coded exposure is accomplished by programming φ_{code1} and φ_{code2} during T_{expo} . The measured GBW product of the OTA is 32 MHz, which results in the smallest T_{chop} period to be 0.15 μ s. As an example, a sequence of binary exposure codes for high-speed motion deblur from [16] is applied. The exposure code sequence, having a length of 52 bits and consisting of 26 "1"s and 26 "0"s, is presented as following:

To implement the temporal exposure encoding, as illustrated in Fig. 2.9(a), the abovementioned exposure codes sequence is applied as φ_{code1} . T_{expo} is divided into 52 T_{chop} periods with each T_{chop} occupying 256 μ s. Meanwhile, φ_{code2} is synchronized to φ_{code1} , which however represents its one's complement code pattern. Such setup ensures charges are not trapped inside PD and always flow to either CTIA1 or CTIA2. Figure 2.9(b) plots both V_{out1} and V_{out2} signals of a pixel in row 5. As can be seen, during the T_{expo} period, both V_{out1} and V_{out2} increase intermittently

22

according to the variation of φ_{code1} and φ_{code2} . When φ_{code1} is code "1" and φ_{code2} presents code "0", generated charges are transferred to C₁, elevating V_{out1} and keeping V_{out2} constant. If φ_{code1} changes to code "0" and φ_{code2} swifts to code "1", charges are transferred to C₂ to raise V_{out2} and maintain V_{out1}. As charge accumulation are alternated between C₁ and C₂, after all 52 T_{chop} are elapsed, V_{out1} and V_{out2} reach 1.596 V and 1.564 V, respectively. Note that the 32-mV difference between the two outputs is due to the mismatch between their OTA offset and charge storage capacitors (C₁ and C₂) and the leakage of M₁ and M₂.



Figure 2.9 (a) Code control signals applied for temporal coded exposure. (b) Output signals of a pixel under temporal coded exposure. Output voltages increase intermittently according to the exposure codes applied.

In comparison with V_{out1} from the non-intermittent exposure, V_{out1} in this case increased by a half only. This result is expected and corresponds to the equal number of "1"s and "0"s. If the 52-bit binary exposure codes are applied to the pixel array, all pixels experience the binary encoded exposure simultaneously. After a 1.66 ms of reset period, the DRAM control module keeps pulling up all φ_{ctrl} rows to permit exposure codes delivery to every pixel. During the 13.28 ms of exposure, the code control module alters φ_{code1} and φ_{code2} according to the exposure code pattern. Within the 1.66 ms of readout period (T_{read}), the row select module selects each row to let the readout module scan all V_{out1} and V_{out2} . Figure 2.10 depicts the final value of V_{out1} and V_{out2} of the pixel array (presented in grey scale). As a reference, under dark condition, the resultant V_{out1} values are around 550 mV (Fig. 2.10(a)). If the non-intermittent exposure is applied, V_{out1} of all pixels vary around 2.5 V (Fig. 2.10(b)). After applying the temporal exposure encoding, both V_{out1} (Fig. 2.10(c)) and V_{out2} (Fig, 2.10(d)) range around 1.5 V, which, as expected, is in the middle of 0.5 V to 2.5 V. Therefore, by programming the binary pattern of the exposure code sequence, the proposed pixel enables temporally encoded exposure of CISs for imaging applications using temporal computations.



Figure 2.10 Pixel output arrays. (a) V_{out1} array under a dark environment without exposure encoding. (b) V_{out1} array under non-intermittent exposure. (c) V_{out1} array under temporal coded exposure. (d) V_{out2} array under temporal coded exposure.



• Pixel performance in spatial-temporal coded exposure

Figure 2.11 (a) Control signals for the spatial-temporal coded exposure of a pixel. (b) Outputs of a pixel from spatial-temporal coded exposure.

As mentioned previously, the spatial-temporal coded exposure requires storing the exposure codes in DRAM cells. During each T_{chop} , the DRAM control module scans between the rows and interacts with the code control module to update exposure codes stored in each pixel. For this prototype design, after measurement, the maximum effective code storage time of the DRAM cells is 2.8 ms, and thus a minimum number of five T_{chop} periods can be included within the 13.28 ms of T_{expo} . To validate the exposure code storage feature of the proposed design, V_{out1} and V_{out2} of a pixel in row 5 are recorded when its DRAM cells are programmed to store a variety of exposure codes. As shown in Fig. 2.11(a), for convenience, φ_{code1} and φ_{code2} have the same patterns. φ_{ctr1} of a pixel, instead of always being high, is pulled up for 256 μ s (T_{code}) every 1.024 ms (T_{chop}). Therefore, there are 13 T_{chop} periods in total, and the DRAM guided exposure period (T_{dram}) is 768

 μ s. Illustrated in Fig. 2.11(b) are the resultant V_{out1} and V_{out2} outputs. During T_{expo}, both V_{out1} and V_{out2} change when φ_{ctrl} is high. If φ_{ctrl} is low, V_{out1} and V_{out2} maintain their status despite changes in φ_{code1} and φ_{code2} . The exposure codes are stored in the pixel's DRAM cells to guide charge transfer during T_{dram}. Stored exposure codes are updated by φ_{code1} and φ_{code2} only when φ_{ctrl} turns high. At the end of T_{expo}, V_{out1} and V_{out2} reach 1.228 V and 1.932 V, respectively. Comparing to the results from the temporal exposure encoding, as expected, generated charges are unevenly distributed to CTIA1 and CTIA2, even if a sequence with equal number of code "1" and code "0" is applied.



Figure 2.12 Spatial-temporal coded exposure test with various exposure-code masks. (a) The spatial exposurecode mask pattern for CTIA1. (b) The spatial exposure-code mask pattern for CTIA2. The corresponding pixel V_{out1} array (c) and the pixel V_{out2} array (d). (e) 10 grey-scale exposure-code masks delivered by φ_{code1} . The V_{out1} array (f) and the V_{out2} array (g) from applying exposure-code masks described in (e). The V_{out1} array (h) and V_{out2} array (i) from applying 10 pseudo-random binary spatial exposure code masks.

To verify the spatial-temporal code exposure of the pixel array, ten exposure-code masks with the same binary pattern (Fig. 2.12) are applied. During T_{expo} (13.28 ms), φ_{code1} and φ_{code2} update the exposure codes to each pixel with a T_{code} of 10 μ s. As T_{chop} set to 1.328 ms, T_{dram} of a pixel is 1.318 ms and the loaded spatial exposure mask is refreshed for 10 times. The mask delivered by φ_{code1} (Fig. 2.12(a)) is the one's complement of that delivered by φ_{code2} (Fig. 2.12(b)). The output V_{out1} and V_{out2} arrays are shown in Fig. 2.12(c) and Fig. 2.12(d), respectively. Results show that pixels encoded with code "1" produces higher V_{out1} as expected. Pixels encoded with code "0", on the other hand, end up with lower V_{out1}, which indicates charges are merely integrated onto their CTIA1s. Instead of applying the same pattern, the spatial exposure code mask may be formed by different binary patterns in each T_{chop} . Shown in Fig. 2.12 (e) are ten masks formed by grey-scale binary codes. As an example, they are applied to φ_{code1} while their one's complement masks are applied to φ_{code2} . During T_{expo} (13.28 ms), each pixel experiences ten different T_{chop} (1.328 ms) periods. The results of V_{out1} (Fig. 2.12(f)) and V_{out2} (Fig. 2.12(g)) arrays show grey-scale transitions, indicating successful spatial-temporal coded exposure on the pixels. Shown in Fig. 2.12(h) and Fig. 2.12(i) are the resultant V_{out1} and V_{out2} arrays when ten pseudo-random binary masks are applied. The mosaic-like results indicate that each pixel experienced independent coded exposure, producing its unique output different from other pixels in the array. Through custom design of the binary pattern in each mask, a pixel's exposure can be independently programmed.

• Noise performance

The noise in pixels is from multiple sources. Temporal noises such as shot noise and read noise produce temporal variations in pixel outputs. With a constant light source, the temporal noise varies as exposure time and illumination changes. To evaluate the dark current of the proposed pixel, the prototype chip is placed in a dark environment and T_{expo} is set to 1.0 s. Under a non-

intermittent exposure, Vout1 of a pixel reached 573.25 mV from a base level of 566 mV. The pixel dark current is measured as 1.09 fA, while the pixel SNR is measured by changing the illumination intensity of the red LED from 1 nW/cm² to 2 μ W/cm² calibrated by an optical power meter. When operating at 60 fps with an exposure time of 13.28 ms, the SNR is calculated as the ratio of a pixel's V_{out1} over the standard deviation of 1000 frames. The pixel achieves a detection limit as low as 5.0 nW/cm². By increasing the light intensity, as shown in Fig. 2.13(a), the peak SNR at $2 \,\mu \text{W/cm}^2$ is 51.5 dB. For different exposure encoding, with the same number of T_{chop}, we also found lower SNR in the spatial exposure encoding due to the uneven number of digital switching. Under a non-intermittent exposure and using up the pixel's full-well capacity, V_{out1} reaches to 2.685 V with the read noise measured to be 5.2 mV. Therefore, the dynamic range of the proposed pixel is 52 dB. Fixed-pattern-noise (FPN), originated from the mismatch across pixels due to process variation, is manifested as pixel output variations. FPN is measured under the dark condition with a non-intermittent exposure of 13.28 ms for 1000 frames. After computation of the mean of the standard deviation of all pixels in the array, the extracted pixel FPN is 0.15% while the column FPN is 0.18%, attributed to the low mismatch of the in-pixel MIM capacitors.



Figure 2.13 (a) SNR of the proposed pixel as a function of the illumination intensity. (b) Power consumption of the pixel array in different types of coded exposure.

• Power consumption

Pixel power consumption varies with the length of the exposure codes. In temporal coded exposure, the power consumption of an $X \times Y$ pixel array in a frame period can be estimated as:

$$P_{t} = \frac{P_{rst}T_{rst} + (P_{code}T_{chop}M + P_{CTIA}T_{expo})XY + P_{read}T_{read}}{T_{frame}}$$
(2.6)

where P_{code} and P_{CTIA} are the coding and the amplification power consumed by a pixel in each T_{chop} period, P_{read} and P_{rst} are the read and reset power, and M is the total number of T_{chop} applied within T_{expo} . Similarly, in spatial-temporal coded exposure, the power consumption in each frame is calculated as:

$$P_{s} = \frac{P_{rst}T_{rst} + [(P_{code} + P_{DRAM})NT_{chop} + P_{CTIA} T_{expo}]XY + P_{read}T_{read}}{T_{frame}}$$
(2.7)

where P_{DRAM} is the power consumption of the in-pixel DRAM cells within T_{chop} , and N is the total number of exposure-code masks (T_{chop}) applied within T_{expo} . The power dissipation of the pixel array is measured after applying a sequence of pseudo-random binary-code-based masks. As summarized in Fig. 2.13(b), at 60 fps with 13.28 ms of exposure time, the non-intermittent exposure (M = N = 0) consumes approximately 6.32 μ W. When temporal coded exposure is applied, the power consumption rises exponentially. If M is relatively small, the exposure encoding related power ($P_{code} \times X \times Y \times M$) is a relatively small component as compared to the total power dissipation. If N is a larger value, such as M = 5000, the pixel array consumes 57.77 μ W with the dominant power cost being for pixel coded exposure. Similarly, when more masks are included within T_{expo} in the spatial-temporal coded exposure, the power consumption grows rapidly. When applying 5000 exposure-code masks (N = 5000), 53.12 μ W of power is consumed with most of it spent on exposure coding. Note that for the same amount of T_{chop} (M = N), the power consumption of the temporal coded exposure is larger than that of the spatial-temporal coded exposure. Since signal φ_{crtl} stays high during the entire T_{expo} in the temporal coded exposure, the component (P_{code} × X × Y) in each T_{chop} consumes more power, as expected. The total power consumption of peripheral control modules is higher when performing the spatial-temporal coded exposure.

2.2.2 LEFM charge modulator and DRAM exposure code memory

In the combination of LEFM charge modulator and DRAM exposure-code memory, a LEFM and two DRAM cells are included in the pixel design. Shown in Fig. 2.14 is the circuit diagram of proposed two-tap exposure-programmable pixel. Capacitors C_1 and C_2 work as two charge storage units, which store charges transferred from the photodiode (PD) through transistors M_1 and M_2 .



Figure 2.14 The circuit diagram of proposed two-tap LEFM + DRAM pixel design. Capacitor C₁ and C₂ are for charge storage while two DRAM units store the exposure codes.

In coded exposure, charges generated from PD are solely integrated on either C₁ or C₂ by switching on/off M₁ or M₂, respectively. The switching operation of M₁ and M₂ is controlled by binary signals φ_{code1} and φ_{code2} , where φ_{code1} is complementary to φ_{code2} . Two DRAM cells are inserted to the path between $\varphi_{code1}/\varphi_{code2}$ and M₁/M₂. When the DRAM control signal φ_{ctrl} is low, M₁ and M₂ maintain their on/off status according to the φ_{code1} and φ_{code2} stored in DRAM1 and DRAM2. Like the exposure-code arrangement described in Table 2.1, generated charges are integrated on C₁ when φ_{code} is set to "1" while flowing to C₂ if φ_{code} is "0". During readout period, charges accumulated in C₁ are read out. After triggered φ_{tran} , charges in C₁ are transferred to the floating diffusion node (FD), where charges are converted to φ_{out} in voltage levels by a source follower once φ_{sel} is pulled up. Charges in C₂, on the contrary, stay still in the readout period until the initial reset of next frame. By pulling up φ_{rst} , charges in C₁, C₂, and FD are demolished and the pixel is reset for incoming exposure period.

Shown in Fig. 2.15 are the timing diagrams of the pixel operation in temporal and spatialtemporal coded exposure schemes during one frame period, T_{frame} . The temporal coded exposure starts with a reset period T_{rst} to reset the voltage level in C_1 and C_2 to V_{rst} (Fig. 2.15(a)). During the exposure period T_{expo} , φ_{ctrl} stays high to pass exposure codes φ_{code1} and φ_{code2} directly to M_1 and M_2 . φ_{code1} and φ_{code2} are synchronized at a rate of $(1/T_{chop})$, which sets the maximum allowable length of the exposure code sequence to be (T_{expo}/T_{chop}) . When T_{expo} ends, φ_{code1} and φ_{code2} stop updating and φ_{sel} pulls high to read out the final output voltage V_{out} . In spatial-temporal coded exposure, within T_{expo} , every pixel is assigned a specific encoding period T_{code} in each T_{chop} for its unique exposure code update (Fig. 2.15(b)). During each T_{code} , φ_{ctrl} of the corresponding pixel pulls high to allow φ_{code1} and φ_{code2} to update/refresh the exposure codes stored in C_3 and C_4 . When other pixels are being processed, each pixel operates according to its stored exposure codes only for a period of T_{dram} , regardless of the changes of φ_{code1} and φ_{code2} . In such manner, within a T_{chop} , all pixels accomplish their exposure codes update (which is equivalent to applying a spatial exposure mask). Until the next T_{chop} starts, the pixel array exposure is programmed by the applied exposure mask. The maximum number of spatial exposure masks that can be applied during one T_{expo} is thus

Texpo/Tchop.



Figure 2.15 The time diagram of (a) temporal coded exposure and (b) spatial-temporal coded exposure of the proposed LEFM+DRAM pixel design.

To verify the feasibility of the pixel design, a test chip consisting of an array of 10×10 pixels is implemented in an 8-metal, 1-poly, $0.13 \ \mu m$ CMOS process. Figure 2.16(a) shows the block diagram of the chip, which is similar to the chip architecture shown in Fig. 2.6(b). The code control module is binary decoder based and is used to deliver φ_{code1} and φ_{code2} to each pixel. The DRAM control module can select either a specific row only or all rows to enable DRAM units in each pixel. Both the row-select module and the column-scanner module are scanner based. During T_{read}, the row-select module performs row-by-row selection while the column-scanner module scans column-by-column to transfer V_{out} signals out of the chip. The proposed pixel structure has a total area of 10 μ m × 10 μ m, while the n-well/p-sub PD occupies an area of 7.0 μ m × 6.9 μ m, giving a fill factor of 48.3% (Fig. 2.16(b)). All capacitors are MIM based, and the transistors and routing wires are placed underneath the capacitors to save area. Each of C₁ and C₂ occupies an area of 3.5 μ m × 3.5 μ m and has a capacitance of 150 fF. C₃ and C₄ are smaller, each occupying an area of 2.0 μ m × 1.5 μ m with a capacitance of 70.2 fF. The chip is powered by two independently regulated 3.3 V power supplies – one for the pixel array and the other one for all peripheral control modules.



Figure 2.16 (a) Implemented test chip architecture. (b) Layout of the proposed LEFM+DRAM pixel. (c) Chip micrograph of the test image sensor.

Control signals for all peripheral control modules are generated from a FPGA chip. To obtain practical results, the default frame rate in all experiments is set to be 60-fps with an exposure time, T_{expo} , of 13.3ms. A 1 μ W/cm² LED centered at 625-nm is used to illuminate all pixels on the chip. The pixel performance is evaluated in terms of non-intermittent exposure, temporal coded exposure, and spatial-temporal coded exposure.

• Pixel performance in non-intermittent exposure

The non-intermittent pixel exposure applies no exposure codes. During T_{expo} , all charges are continuously transferred to C₁. Shown in Fig. 2.17(a) is the oscilloscope plot of the output V_{out} of a pixel. Prior to the start of T_{expo} , φ_{rst} goes high to reset the photodiode and C₁ to a 3.052 V base level. Once φ_{rst} is pulled low, the photodiode generated charges start accumulating on C₁. As charges continuously accumulate on C₁, V_{out} level decreases until it reaches to 1.086 V at the end of T_{expo} .



Figure 2.17 (a) Plot of a pixel under non-intermittent exposure. (b) Plot of a pixel under temporal coded exposure. Output voltage levels decrease intermittently according to the exposure codes applied.

• Pixel performance in temporal coded exposure

To demonstrate the capability of pixel in temporal coded exposure, the same sequence of exposure codes used in section 2.2.1 is applied. T_{expo} is divided into 52 T_{chop} intervals with each T_{chop} occupying 256 μ s. Figure 2.17(b) illustrates V_{out} of a pixel in the array. When φ_{rst} is active, both C_1 and C_2 are reset. Once T_{expo} starts, φ_{code1} applies the exposure codes while φ_{code2} applies the complimentary exposure codes. As charges are distributed to C_1 and C_2 , V_{out} changes accordingly. When all 52 T_{chop} are elapsed, at the end of T_{expo} , V_{out} reaches to 1.982 V. In comparison with the non-intermittent exposure, V_{out} has decreased by half. Such results are expected as a 50% duty cycle exposure-code sequence is applied.



Figure 2.18 (a) Plot of a pixel under spatial-temporal coded exposure. (b) The resultant pixel array outputs after applied pseudo-random binary exposure-code masks.

• Pixel performance in spatial-temporal coded exposure

The spatial-temporal coded exposure requires enabling DRAM cells for exposure-code storage. For the purpose of comparison, φ_{code1} and φ_{code2} remain the same patterns as above. Shown in Fig. 2.18(a) is the output of a pixel in row 1. During T_{expo} , in this example, φ_{code1} and φ_{code2} begin to transfer exposure codes and φ_{ctrl} goes to high every 4 codes. In such manner, when φ_{ctrl} in row 1 is high for T_{code} (with a duration of 256 μ s), the pixel updates the exposure codes from φ_{code1} and φ_{code2} to its DRAM units. When φ_{ctrl} of row 1 is low, the pixel charge distribution during a period of T_{dram} is based on the codes stored. Within T_{expo} , the pixel experiences 13 T_{chop} intervals. From the V_{out} plot, one can see that the voltage levels change only when stored exposure codes are updated. The changes of φ_{code1} and φ_{code2} do not affect the pixel unless φ_{ctrl} turns to high. At the end of T_{expo}, V_{out} decreases to 2.378 V. Comparing with the results from temporal exposure encoding, the pixel produces different outputs as its exposure codes are sourced from DRAM units. Since each T_{chop} can apply a spatial exposure-code mask with different patterns, a series of ten exposure-code masks are applied within the T_{expo} for test. Shown in Fig. 2.18(b) are the pseudo-random binary-codes based masks. During T_{expo}, ten different exposure-code masks are applied with one mask per T_{chop}. For each pixel in the array, ten different exposure codes are received and stored by the in-pixel DRAM units. From the resulting V_{out} outputs after the ten masks are applied, the mosaic-like result is observed, indicating that the intermittent exposure of each pixel is independently programmed, producing a unique output voltage level compared to any other pixel in the array.

2.2.3 CTIA charge modulator and SRAM exposure code memory

In the combination of CTIA charge modulator and SRAM exposure-code memory, two CTIA structures and a SRAM cells are included in the pixel design. Shown in Fig. 2.19 is the circuit diagram of proposed two-tap exposure-programmable pixel of such combination. Capacitors C_1 and C_2 work as two charge storage units, which store charges transferred from the photodiode (PD) through transistors M_1 and M_2 .



Figure 2.19 The circuit diagram of proposed two-tap CTIA + SRAM pixel design. Capacitor C₁ and C₂ are for charge storage while a SRAM cell stores the exposure code.

The 2-tap CTIA charge modulator (CTIA1/CTIA2) is connected to the photodiode through charge-transfer gates (M₁/M₂). Each CTIA black consists of an OTA and a charge storage capacitor (C₁ or C₂). When signal φ_{sel} is pulled to high, CTIA1 and CTIA2 are read out and the pixel output voltage V_{out1} and V_{out2} are defined as:

$$V_{out} = \frac{(I_{PD} + I_{dark} + I_{leak})MT_{chop}}{\left(1 - \frac{C + C_{PD}}{AC}\right)C} + V_{offset}$$
(2.8)

where A and M are the CTIA open-loop gain and the number of T_{chop} periods with exposure code "1" applied, respectively. C represents the integration capacitor C₁ or C₂. I_{PD} is the photocurrent related to the illumination on the photodiode surface. I_{dark} and I_{leak} donate the photodiode dark current and the leakage currents from M₁ and M₃. V_{offset} is the offset voltage from the OTA and the readout circuit. The 1-bit SRAM code memory holds the exposure code signals designated to control M₁ and M₂. To update SRAM, exposure code signal (φ_{code1}) and the complementary code signal (φ_{code2}) are accessed into the memory core through M₅/M₆ by toggling φ_{ctrl} signal.



Figure 2.20 Time diagram of spatial-temporal coded exposure of the proposed CTIA+SRAM pixel design.

Depicted in Fig. 2.20 is the timing diagram of the proposed pixel in a frame period. At the start of a frame, φ_{rst} pulls up to reset both CTIA blocks. When T_{expo} begins, φ_{rst} turns to low and followed by the start of first T_{chop} period. In each T_{chop} , signal φ_{ctrl} toggles to update control signal

Q for M_1 and its complementary signal Q' for M_2 . As charges gather on C_1/C_2 , voltage level V_A/V_B changes according to the condition of Q/Q'. V_A increases up when Q is high (exposure code "1" stored in SRAM), and remains still if Q is low (exposure code "0" stored in SRAM). During the readout period T_{read} , Q keeps at low to turn off M_1 and signal φ_{sel} toggles when the pixel is selected to output V_{out1} and V_{rst} for further processing. Summarized in Table 2.2 is the exposure code arrangements for different operations of valve switches M_1 and M_2 .

	∅ code1	Øcode2	∅ code1	∅ code2	∅ code1	arphicode2
Exposure Code	1	0	0	1	Х	Х
arphictrl	1		1		0	
M_1	Close		Open		Determined by SRAM	
M ₂	Open		Close		Determined by SRAM	

Table 2.2 The exposure code arrangements for valve switches M1 and M2 operations of the proposed pixel.

Due to the limited chip tape-out resources, there is no stand-alone test chip for this proposed exposure-programmable pixel design. The pixel design feasibility and performance, however, are evaluated in Chapter 3, where the pixel design proposed in this section is implemented by one of the prototype computational CIS chips.

2.2.4 LEFM charge modulator and SRAM exposure code memory

In the combination of LEFM charge modulator and SRAM exposure-code memory, two LEFMs and a SRAM cells are included in the pixel design. Illustrated in Fig. 2.21 is the proposed circuit diagram of the two-tap exposure-programmable pixel in this combination.



Figure 2.21 The circuit diagram of proposed two-tap LEFM + SRAM pixel design. Capacitor C₁ and C₂ are for charge storage while a SRAM cell stores the exposure code.

The 2-tap LEFM charge modulator is connected to the photodiode via charge-transfer valves (M_1/M_2) . Charges generated from PD are accumulated on either C₁ or C₂ by switching on/off M₁ or M₂, respectively. The switching operation of M₁ and M₂ is controlled by binary signals Q and its complementary signal Q', which are provided by the internal core of the SRAM based exposure-code memory cell. The content writing of SRAM cell is through feeding exposure-code signals φ_{code1} and φ_{code2} . When the SRAM control signal φ_{ctr1} is low, M₁ and M₂ maintain their on/off status according to Q and Q' stored in the SRAM cell. Once φ_{ctr1} is asserted to high, Q and Q' get updated by signals φ_{code1} and φ_{code2} . Like the exposure-code arrangement described in Table 2.2, generated charges are integrated on C₁ when set φ_{code} to "1" while flowing to C₂ if configure φ_{code} to "0". During readout period, charges accumulated in C₁ and C₂ are read out. After triggered φ_{tran} , charges in C₁ and C₂ are transferred to corresponding FD, where charges are converted to V_{out1} and V_{out2}

in voltage levels by source followers once φ_{sel} is pulled up. By pulling up φ_{rst} , charges in C₁, C₂, and FD are demolished and the pixel is reset for the next frame period. To be noticed that reading out C₂ is not mandatory. If only V_{out1} is needed, the source follower for V_{out2} can be omitted.



Figure 2.22 The time diagram of (a) temporal coded exposure and (b) spatial-temporal coded exposure of the proposed LEFM+SRAM pixel design.

Depicted in Fig. 2.22 are time diagrams for pixel operations in temporal coded exposure and spatial-temporal coded exposure. In temporal coded exposure (Fig. 2.22(a)), the SRAM writing signal keeps in high to permit exposure-code signals φ_{code1} and φ_{code2} access to all pixels. In each

T_{chop}, φ_{code1} and φ_{code2} toggle to update exposure codes to every pixel. Based on the updated Q and Q', M₁ and M₂ switch accordingly to guide the charge flow towards to either C₁ or C₂. The voltage level at point A (V_A) drops to lower level if M₁ switch is closed (apply code "1" to φ_{code1}). In opposite, the voltage level at point B (V_B) decreases when M₂ switch is closed (apply code "1" to φ_{code2}). During T_{read}, the final voltage level of V_A and V_B are interpreted by source followers and the pixel outputs V_{out1} and V_{out2}. In spatial-temporal coded exposure (Fig. 2.22(b)), each pixel receives unique exposure codes in T_{expo}. In a T_{chop} period, the SRAM writing signal φ_{ctrl} toggles for a period of T_{code} only when the pixel is selected for exposure code update. For the rest of T_{chop}, the pixel operates according to the updated exposure code stored in its SRAM while φ_{code1} and φ_{code2} are updating other pixels. Therefore, the minimum length of T_{chop} is the time where all pixels are refreshed with new exposure codes.



Figure 2.23 (a) Sample layout of the proposed pixel design. (b) Time diagram of CIS operation in performing spatial-temporal coded exposure.

The proposed pixel is implemented using an 8-metal, 1-poly, 0.13- μ m CMOS process. Figure 2.23(a) illustrates an example of pixel layout, which has a total size of 10.8 μ m × 10.5 μ m. The n-42

well/p-sub PD occupies an area of 7.2 μ m × 6.9 μ m, giving a fill factor of 43.8%. To save silicon usage, all capacitors are formed by MIM structures where the transistors and routing wires are placed underneath. In this research work, there is no chip tape-out specifically for this proposed pixel design. However, for a CIS chip employed the proposed pixel to implement spatial-temporal coded exposure, an operation sequence of CIS is suggested. Described in Fig. 2.23(b) is the time diagram of CIS control signals in performing spatial-temporal coded exposure. For a CIS chip containing M × N pixels, signals φ_{rst} , φ_{sel} , and φ_{ctrl} control pixels through M row-wise wires. Exposure-code signals φ_{code1} and φ_{code2} access to pixels via N column-wise lines. Starting with a reset period (T_{rst}), all pixels are reset after toggled φ_{rst} . The exposure period T_{expo} is formed by number of T_{chop} periods. In each T_{chop} , φ_{ctrl} scans row by row to enable SRAM writing of a row of pixels for a period of T_{code}. Meanwhile, φ_{code1} and φ_{code2} in each column wire access to pixels in the selected row and update their exposure codes. After all pixels in the pixel array are refreshed with new exposure codes, an exposure-code mask is armed and maintained for a period of T_{sram}. By circulating such process in each T_{chop}, a total number of K exposure-code masks are implemented within T_{expo} , which is followed by a row-by-row pixel readout process.

2.3 Chapter conclusion

In conclusion, this chapter discussed various pixel designs with exposure-programming capability for the implementation of spatial-temporal coded exposure on CIS chips. Two types of in-pixel charge modulators and exposure-code memory structures are proposed to realize in-pixel exposure-code guided selective charge accumulation. In circuit level implementation, four types of 2-tap pixel architectures are introduced with each in a different combination of charge modulator and exposure-code memory. A comparison between those four types of pixel design is summarized in Table 2.3, where specifications are qualitatively concluded.

Pixel Architecture	CTIA+DRAM	LEFM+DRAM	CTIA+SRAM	LEFM+SRAM
Number of Taps	2	2	2	2
Memory Cells	2 DRAMs	2 DRAMs	1 SRAM	1 SRAM
Pixel Size	Medium $(12.2\mu m \times 12.1\mu m)$	$\frac{\text{Small}}{(10\mu\text{m}\times10\mu\text{m})}$	Large* (12.6µm × 12.6µm)	Medium $(10.8\mu m \times 10.5\mu m)$
Fill Factor	Low (32.7%)	High (48.3%)	Medium* (38.7%)	High (43.8%)
Coding Speed	Slow	Slow	Fast	Fast
Power Consumption	High	Medium	Medium	Low
Circuit Complexity	Medium-High	Low	High	Medium
Control Complexity	Medium	Low	High	Medium-Low

* Based on the implemented pixel design in Chapter 3.

Table 2.3 Comparison between different proposed exposure-programmable pixel designs.

In general, the comparison indicates a tradeoff between pixel design complexity and power efficiency. From the design point of view, the LEFM + DRAM based pixel design has the least number of components (transistors and capacitors), the smallest pixel pitch, and the simplest configuration requirement in trade of pixel coding speed (the exposure-code memory writing speed). For power efficiency, the LEFM + SRAM based pixel design achieves the lowest power consumption while the CTIA + DRAM based design consumes the most due to the OTA and frequent DRAM refreshment.

Chapter 3: Computational Cameras with Coded-Exposure Image Sensors

Based on the exposure-programmable pixel designs discussed in Chapter 2, CIS designs with a spatial-temporal coded exposure feature are deployed to form computational cameras for CS applications. In this chapter, two prototype CS cameras equipped with two types of coded-exposure CISs are presented to evaluate their performance under coded exposure. Section 3.1 and 3.2 give the design details of two CIS chips. The hardware formation and experimental results of prototype cameras are described in Section 3.3.

3.1 A Coded-Exposure CIS with 2-Tap CTIA+SRAM Pixels

The CTIA+SRAM based pixel design proposed in Chapter 2 is selected to form the first CIS chip. As discussed in Section 2.2.3, this type of pixel owns all the advantages provided by CTIA structure and SRAM. However, it has the most sophisticated pixel architecture which brings difficulties to the CIS design.

3.1.1 Circuitry and Architecture Design of CIS

Based on the pixel circuitry suggested in Fig. 2.19, the exposure-programmable pixel implemented in the CIS is illustrated in Fig. 3.1. There are two major modifications in the implemented pixel: the exclusion of V_{out2} and the inclusion of sampling circuitry in the readout block. As for a binary coded exposure, all charges generated during code "0" periods are disposed. Hence, the second pixel output is neglected. In addition, the readout block contains a sampling and readout circuit. After each pixel reset, the reset noise voltage is sampled through a switch (M₇) and held in capacitor C_r , which is read out to column V_{rst} in T_{read} . With such arrangement, CDS operation is enabled when CIS operates in global shuttering scheme.



Figure 3.1 Circuit diagram of the implemented exposure-programmable pixel.



Figure 3.2 Time diagram of a pixel operation under spatial-temporal coded exposure in a frame period.

With the abovementioned modifications, the pixel operation under coded exposure also changed. Shown in Fig. 3.2 is the time diagram of pixel control signals in a frame period. At the beginning of T_{expo} , signal φ_{tran} stays high for a short period to ensure the reset noise level V_{rst} is recorded by C_r. After the reset noise sampling is done, φ_{tran} turns to low and the first T_{chop} period starts. By pulling up $\varphi_{Q_{EN}}$ (the global mask shutter), charges accumulate on C₁ (C₂) while voltage level V_A (V_B) changes according to the condition of Q (Q'). V_A increases when Q is high (exposure code "1" is stored in SRAM), and remains ideally unchanged when Q is low (exposure code "0" is stored in SRAM). During the readout period (T_{read}), φ_{sel} toggles and the pixel outputs the final voltage level V_{out} and V_{rst} into separate wires for further processing.





Figure 3.3 Block diagram of the first CIS. All coded exposure related blocks are highlighted in grey.



Figure 3.4 Circuit diagram of the coded-exposure related functional blocks.



Figure 3.5 Time diagram of the process of exposure-code delivery. Exposure codes are streamed into deserializers through 12 channels and shipped to pixels in row-by-row basis.

The overall CIS architecture is shown in Fig. 3.3. Based on a conventional CIS architecture design (see Appendix B), the pixel array consists of 192×192 pixels is surrounded by imaging related blocks and coded-exposure specific functional blocks (marked in grey). The row logic

block consists of scanners and demultiplexers to feed φ_{rst} , φ_{tran} , and φ_{sel} to each row of pixels. Exposure codes are loaded from off-chip memory by 12 1-to-16 de-serializers and distributed to pixels through 192 shared column lines. As illustrated in Fig. 3.4, the de-serializer drived by clock signal $\varphi_{\text{CODE CLK}}$ receives and buffers exposure codes $(D_1...D_{192})$ from the channel. After 16 clock cycles, clock signal $\varphi_{\text{LOAD CLK}}$ is asserted high and 192 output D flip-flops are triggered simultaneously to stream out loaded exposure codes $(Q_1...Q_{192} \text{ for } \varphi_{\text{code1}} \text{ and } Q'_1...Q'_{192} \text{ for } \varphi_{\text{code2}})$ to column wires. Pixels receive their exposure codes in row-by-row basis (see the timing diagram depicted in Fig. 3.5). The divided clock φ_{LOAD_CLK} also drives an 8-bit binary counter, which guides the row scanner to select one of 192 rows and pulls up signal φ_{ctrl} of all pixels in the row. As signal φ_{ctrl} stays low in unselected rows, exposure code signal φ_{code1} and φ_{code2} loaded on column wires access to the selected row of pixels only. In each T_{chop} period, such row-wise exposure code refreshment lasts for T_{update} until all rows of pixels are updated. The pixel array is then armed with an exposure code mask for a period of T_{stay}. Every pixel operates under the guidance of its own exposure code before update starts in the next T_{chop} period. Theoretically, the duration of T_{stay} can be reduced to zero, which gives the maximum number of T_{chop} (or exposure code mask) included in a T_{expo} period.

During T_{read} period, the pixel array is read out row by row. Pixels on the selected row output their signals (V_{out} and V_{rst}) to the column-wise signal processing slice through shared column wires. As depicted in Fig. 3.6, the column pixel outputs are processed by a two-stage programmable-gain amplifier (PGA) followed by a 10-bit single-slope (SS) analog-to-digital convertor (ADC). The PGA is formed by two capacitive amplifiers. The first amplifier is used to amplify the incoming pixel outputs with a fixed gain. The input common-mode voltage is adjusted by feeding with appropriate offset voltage levels ($V_{offset+}$ and $V_{offset-}$). After the first stage, V_{rst} is subtracted from the pixel output V_{out}, which is converted from a single-ended form to a differential signal with a common-mode voltage for the OTA. The second amplifier has a variable gain determined by the capacitance of input capacitor (C_{a2}). The output differential signal from the first stage is amplified with a selected gain. The OTA consists of common-mode feedback (CMFB) structure and operates in continuous time when it is enabled (triggering φ_{PGA}). After reset (toggling φ_{PGA} -rst), pixel output signals are amplified and reached to the output of PGA directly.



Figure 3.6 Circuit diagram of the column-wise signal processing slice.



Figure 3.7 Time diagram of the column-wise signal processing slice during a readout period.

The comparator in the SS-ADC is designed with three stages and a selective bias (V_{bias}) for power efficiency. By selecting different bias fed to the gate of tail transistor of the first stage, the comparator operates in either high current mode (selecting V_{b_h}) for comparison or low current mode (selecting V_{b_h}) in reset and hold periods. During the comparison period T_{comp} , the comparator compares the PGA output (V_{PGA}) and a ramp wave V_{ramp} while a 10-bit global counter counts down. The A/D conversion is accomplished by storing the counter codes in column D flipflops at the falling edge of the comparator output. After all columns are done conversion, within T_{scan} , the horizontal scanner scans all column ADC outputs and sends out of the chip for further processing. After all rows are selected and processed, at the end of T_{read} , the pixel array is read out and an image is output from the chip.

When perform spatial-temporal coded exposure, every pixel is expected to experience various exposure times to the scene in comparison to the non-intermittent exposure. Therefore, the SS-ADC in the proposed CIS includes an adaptive ramp generation feature which adjusts the ramp slope for pixel output signal from different scene brightness conditions.


Figure 3.8 Conceptual and circuit diagram of the adaptive SS-ADC enabled by the output slope adjustment of the ramp generator.

Fig. 3.8 illustrates the working principle of the adaptive ramp generator. The ramp generator consists of a 10-bit charge scaling digital-to-analog convertor (DAC), which is followed by a

switch-capacitor based ramp-slope adjuster. There are three parallelly placed capacitors (C_{g0} , C_{g1} , and C_{g2}) in the adjuster, giving a capacitance relationship of $C_{g0} = C_{g1} = 0.5C_{g2}$ and their total capacitance equals to C_{comp}. One plate of C_{g0} and C_{g1} are connected to either ground or the ramp signal V_{amp} through switches SWA_0/1 and SWB_0/1. In different combinations of switching states, the output ramp wave V_{ramp} varies from V_{amp} with different slope. In T_{comp} , the voltage level of V_{ramp} drops during the global counter down-counting period T_{slope}. With four different slopes, V_{ramp} reaches to four voltage levels (V_{L1} to V_{L4}) at the end of T_{slope} . Thus, for the voltage level of V_{PGA}, a corresponding slope is configured to adjust the ramp wave adopt in the A/D conversion. The decision of slope selection is made after an evaluation of V_{PGA} from all pixels. Hence, a 2-bit successive-approximation-register (SAR) ADC utilizing the PGA differential outputs (Vo+ and Vo-) is incorporated in each column processing slice. Through appropriate setup on the reference voltage V_{ref_SAR}, 192 2-bit outputs ([B₁, B₀]₁ to [B₁, B₀]₁₉₂) are acquired after processing a row of pixel outputs. As each column may produce different results, 191 data selectors are placed between columns to compare and search for the largest $[B_1, B_0]$ value. The comparison between the biggest $[B_1, B_0]$ of different rows is done by the 192th data selector. The previous comparison result is buffered and used to compare with the largest $[B_1, B_0]$ from the current selected row of pixels. Once all rows are processed, the largest SAR-ADC result of all pixels, $[B_1, B_0]_{max}$, is latched in the buffer and delivered to a control logic block to synthesize corresponding SWA_0/1 and SWB_0/1 control signals for ramp slope adjustment. Such adaptive ramp adjustment requires an operation period of two frames. In the first frame, the adaptive ramp generator searches for $[B_1,$ $B_0]_{max}$ from the processing results generated by column-wise SAR-ADCs. Based on $[B_1, B_0]_{max}$, the adaptive ramp generator produce corresponding ramp wave to be used in T_{slope} of the second

frame period. Such adaptive ramp generation effectively assists the SS-ADC to handle various pixel outputs, especially large pixel outputs due to coded exposure or high scene brightness.



Figure 3.9 Micrograph of a prototype CIS chip. The die size is 4.5mm × 4.0mm including bonding pads.

3.1.2 Camera Hardware Development and Experimental Results

Based on the pixel and CIS architecture described in previous sections, the first prototype CIS chip is design and fabricated in a 0.13- μ m 8-metal, 1-poly, *n*-well CMOS process. Shown in Fig. 3.9 is the chip microphotograph. The die size is 4.5mm × 4.0mm including the bonding pads. Each pixel occupies an area of $12.6 \times 12.6 \mu$ m² with a fill factor of 38.7%. For purpose of saving space, all in-pixel capacitors are implemented in metal-to-metal (MIM) structures shielded with ground wires and placed adjacent to the pinned photodiode. With the integration capacitors C₁ and C₂ of 100 fF, the simulated open-loop gain and the gain-bandwidth product of the CTIA is 56 dB and 36 MHz, respectively. The first stage amplifier of PGA has a unity gain, and a selective gain of ×1 and ×2 on the second stage amplifier. The chip is powered up by two independently regulated power supplies -a 2.5 V power source for the pixel array and the analog circuits in the column processing slice, and a 1.2 V supply for all digital circuits.



Figure 3.10 (a) Block diagram of the camera system for compressive focal-stack imaging tests. (b) The camera hardware platform which holds the prototype CIS chip and the VCM auto-focus lens.

The fabricated CIS chip is packaged in a camera module under a 25.0-mm f/2.4 focal lens driven by a M12-mount vocal-coil motor (VCM) device. Illustrated in Fig. 3.10 is the camera system which consists of two printed-circuit boards (PCBs) which hold the camera module and a processing system based on field-programmable gate array (FPGA), respectively. Through cross-PCB interconnections, the FPGA chip (Cyclone V, Altera) send commands to guide CIS operation. The VCM device (VM18001, SZKJ Motors) is controlled by a DAC based VCM driver (DRV201, Texas Instruments) synchronized with the FPGA to implement designated camera focal sweep. The prepared exposure code masks are buffered in a discrete double data rate synchronous dynamic random-access memory (DDR-SDRAM) on the FPGA PCB. When the CIS is under coded exposure, exposure code masks are extracted by the FPGA and shipped to the sensor. The CIS output images are collected by the FPGA and stream out to a PC through a universal serial bus (USB) 3.0 module.



Figure 3.11 The DNL, INL, and spectrum plots of the SS-ADC.

Prior the experiments of implementing machine vision applications, the prototype CIS is characterized. Operating in non-intermittent exposure mode, the measured sensor sensitivity is $1.7V/lux \cdot s$ under illumination of a light-emitting diode (LED) centered at 630nm. When place the

CIS in a dark environment, the pixel dark current is measured as 1.2 fA and the output Vout reached to 584.3 mV from a base level 573.2 mV after 1.0 s of non-intermittent exposure. By sweeping the LED illumination intensity up to 5.0 μ W/cm², the pixel used up its full-well capacity and outputted the maximum V_{out} of 2.18 V with the measured read noise of 4.2 mV. Hence, the pixel dynamic range is 51.6dB. By tuning the illumination intensity to target ~50% of the pixel saturation (Vout = 1.4 V), the FPN is estimated as the mean of standard deviation of the pixel array. The extracted pixel and column FPN are 0.42 % and 0.26 %, attributed to the well-matched in-pixel MIM capacitors. Through separate analog inputs connected to the column processing slice, the performance of the ADC is evaluated with streaming in external analog test signals. Summarized in Fig. 3.11 are the measured differential nonlinearity (DNL) and integral nonlinearity (INL) of the SS-ADC in a typical column. The sampling rate of the SS-ADC is 97.6 KS/s at 100 MHz clock frequency and the voltage interval between V_{L1} to V_{L4} is set to 0.5 V after calibration. After measurement, the DNL and INL in four ramp slopes are all less than -0.37/+0.32 LSB and -1.13/+1.41 LSB. When apply a 200 KHz sinusoidal waveform with full voltage swing in four ramp slope schemes, as depicted in the spectrum plot in Fig. 3.11, the worst signal-to-noise and distortion ratio (SNDR) and effective number of bits (ENOB) are 62.85 dB and 10.1 bits, respectively.

The evaluation of camera imaging capability starts with the image capture in different exposure modes. The CIS operates at 30 fps with T_{expo} of 23.62 ms. With the frequency of clock signal φ_{CODE_CLK} is set at 100 MHz, the mediatory duration of T_{update} is 30.72 μ s and the maximum allowable number of exposure code masks (or T_{chop}) in a T_{expo} period is 768 (T_{stay} is reduced to 0 s). Illustrated in Fig. 3.12 are the captured images of a resolution chart (QA-72A) with 192 exposure code masks applied in a frame ($T_{chop} = 122.92 \ \mu$ s). The non-intermittent exposure is

implemented by formatting exposure codes in all masks to "1". All generated charges are preserved in C_1 while CTIA2 is excluded during the exposure period. To practice the spatial-temporal coded exposure, each exposure code mask is tailored with a unique code pattern. As a demonstration, a column gradient code pattern is selected and sequenced by 192 coded masks. As expected, the captured image reveals a grey-scale like transparency transition, which implies the effective coded exposure applied by the CIS.

Non-Intermittent Exposure:





Spatial-Temporal Coded Exposure:

192 Exposure Code Masks (Gradient Code Pattern)



CIS Output Image



Figure 3.12 Exposure code mask patterns and experimentally captured images using non-intermittent exposure and spatial-temporal coded exposure.



Figure 3.13 (a) Chip power consumption with different code composition and SS-ADC ramp slopes in a frame period containing 5 exposure code masks. (b) Chip power consumption with different number of exposure code masks in a frame period.

The CIS power consumption varies when apply coded exposure. Two main factors which affect the power dissipation are the composition of the exposure code pattern and the number of exposure code masks included in a frame period. Fig. 3.13(a) is the chip power consumption with different percentages of code "1" contained in pixels' exposure code sequences. Under a constant illumination, 5 identical exposure code masks are applied to the pixel array in a frame. The outputs of pixel array are sampled by the SS-ADC in 4 different ramp slopes, which terms out to discriminations of power dissipation. From the results, it is clear that bigger ramp slopes contribute lower power due to the increasing quantization. As the slope adjuster always selecting the largest

possible ramp slope to accommodate the highest pixel output, chip power consumption is optimized. When more exposure code masks are included in a frame to achieve higher compression ratio, the chip power consumption climbs. As depicted in Fig. 3.13(b), the total power ramps up to 31.5 mW when the number of exposure code masks increased to 768 (with 20% of exposure code "1" per frame and using V_{L4} ramp slope). Analog circuits dominate the power dissipation when the number of exposure code mask is small, while the power consumed by digital blocks start to become dominate once it is higher than 600 due to the high-volume code delivery.

3.2 A Coded-Exposure CIS with 2-Tap LEFM+DRAM Pixels

The second prototype CIS chip is with a different pixel type. As discussed in Chapter 2, the LEFM + DRAM type pixel has superior design simplicity. Therefore, it is selected to form a pixel array in the implementation of the second CIS chip.

3.2.1 Chip Architecture of CIS

The pixel circuitry is the same as depicted in Fig. 2.14, and the overall sensor architecture is illustrated in Fig. 3.14. Similar to the first CIS chip, the pixel array is supported by variety of functional blocks. The row decoder block performs row-by-row scanning to provide signals φ_{rst} , φ_{tran} , and φ_{sel} to the pixel array. The DRAM controller is a row scanner which sequentially selects a row of pixels to enable their exposure codes refreshment. If signal φ_{ctrl} in the chosen row is set to high, then the column exposure code decoder accesses to pixels in the selected row and distributes φ_{code1} and φ_{code2} to their DRAMs. In every T_{chop}, the DRAM controller scans through all rows to ensure the column exposure code decoder updates exposure codes in every pixel. In a T_{read} period, the pixel array is read out in a CDS scheme realized by column-parallel CDS circuits.

Before reaching to a column scanner to output the final image data, pixel output signals are digitized to corresponding digital format through column-parallel 8-bit SS ADCs.



Figure 3.14 The block diagram of chip architecture of the second CIS.

3.2.2 Camera Hardware and Measurement Results

For hardware implementation, the second test image sensor containing 128×128 pixels are fabricated in an 8-metal, 1-poly, 0.13- μ m, *n*-well CMOS process (Fig. 3.15). The chip dimension

is 3.0 mm \times 3.0 mm with a pixel pitch size of 10.2- μ m and a fill factor of 41.5%. All in-pixel capacitors are implemented in MIM structure, which allows routing wires placed underneath to maximize space utilization. A total of 128 column-parallel SS-ADCs are employed to convert pixel output signals in every column into 8-bit digital values before they are scanned and send out of chip. The chip is powered by two separately regulated power sources – a 3.3V power supply for analog circuits and a 1.2V power line for all digital control modules.



Figure 3.15 Chip micrograph. The second chip is in a size of 3 mm \times 3 mm.

Except for the VCM lens, the second prototype camera system is in a similar formation as the first prototype camera shown in Fig. 3.10(a). As illustrated in Fig. 3.16(a) and (b), the test image sensor is bonded on a customized PCB stacked on another PCB housing power management and microcontroller chips. A FPGA chip is employed as the microcontroller to provide and process signals come to/from the test image sensor. The prototype camera communicates to a computer through a USB wire, which is also utilized as the system power supply.



Figure 3.16 (a) Fabricated CMOS image sensor. (b) Prototype computational camera system.

The characterization of the test image sensor starts from pixel level. The pixel performance is evaluated through illumination of a red LED centered at 630 nm. In a dark environment, the pixel dark current is measured as 1.27 fA, while the pixel output φ_{out} decreased to 2.48 V from the reset level ($V_{rst} = 2.56$ V) after 1.0 s of non-intermittent exposure. When sweeping the LED illumination intensity from 1.0 nW/cm² to 2.0 μ W/cm², the lowest achievable detection limit of a pixel is 7 nW/ cm². Under an exposure time of 13.28 ms (at 60 fps), the peak signal-to-noise ratio (PSNR) at $2.0 \,\mu\text{W/cm}^2$ is 34.2 dB and the output dynamic range is 47.3 dB. In order to prevent code lost, the minimum refreshment frequency of the in-pixel DRAM cell is 338.9 Hz, which determines the maximum allowable length of every T_{chop} to be 377.6 ms. As limited by the fabrication process, the maximum speed of the column exposure code decoder is 500 MHz. Therefore, the minimum length for each T_{chop} is calculated as $128 \times 128 \times 1/(500 \text{ MHz}) = 32.76 \,\mu\text{s}$. By calculating the mean of the standard deviation of all pixel outputs, the extracted pixel and column FPN are 0.17 % and 0.22 %, respectively. The imaging capability of the pixel array is validated using a resolution chart (QA-71). With T_{expo} set to 10.49 ms, 128 exposure-code masks are accommodated in a frame with each T_{chop} equals to 81.92 μ s. Figure 3.17(a) displays the output image when the exposure codes 63

in all masks are "1". As φ_{code1} in each T_{chop} was continuously pulling up, pixels preserved all collected charges and the image sensor implemented non-intermittent exposure. The spatial-temporal coded exposure, on the other hand, was verified by utilizing variety of code patterns. Figure 3.17(b) reveals an output image when spatial grey-scale code masks were applied. Each pixel experienced 128 different T_{chop} periods and the resultant image indicated a smooth grey-scale transparency transition, indicating effective spatial-temporal exposure encoding on the pixel array.

(a) Non-intermittent Exposure



Output Image

Figure 3.17 Camera output images of (a) a non-intermittent exposure test, (b) a spatial-temporal coded exposure test using 128 column-grey-scale masks.

The power dissipation of the second prototype camera system is contributed by the CIS chip and its peripheral signal processing devices. In terms of the CIS, which is the most power hungry component in this system, the power consumption depends on the number of exposure-code masks applied during the coded exposure. Summarized in Fig. 3.18(a) is the power consumption of the pixel array and its peripheral blocks measured after applying different number of exposure-code masks in a single shot (one frame period). When the number of mask is 1, which is the minimum number of exposure-code mask applied in T_{expo} , the pixel array consumes 0.76 μ W while its peripheral circuitry spent 12.83 μ W of power (mostly dissipated on SS-ADCs). As the number of mask increases, the power dissipation rises exponentially with most of it spent on exposure-code mask refreshment. While the number of masks is still small, the power consumed by the pixel array is relatively similar to that of its peripheral modules. If it reaches to 30000, which is the maximum applicable number of exposure-code masks in one second (with T_{chop} sets to 32.76 μ s), the pixel array consumes 22.85 mW while ~10 mW of power dissipated on peripheral modules.

The overall power consumption of the CIS, as suggested in Fig. 3.18(b), also climbs rapidly when more exposure-code masks are included in a T_{expo} period. The CIS consumes 23.5 μ W of power when included 1 exposure-code mask per frame. Its power consumption surges up to 32.85-mW when a number of 30000 exposure-code masks are applied. Apparently, a smaller number of exposure-code mask helps reduce the chip power consumption. However, it limits the temporal resolution of camera coded exposure, which directly impacts on the result of CS. Therefore, there is a tradeoff between the power budget and the number of masks applied. Based on the type of scene, one needs to maximize the performance of spatial-temporal coded exposure with an acceptable number of exposure-code masks to meet the power budget requirement.



Figure 3.18 (a) Power consumption of (a) the pixel array and peripheral modules during coded exposure. (b) The overall power consumption of the image sensor in a single shot.

3.3 Chapter Conclusion

In this chapter, the hardware implementations of the proposed computational cameras are presented. Two prototype CIS chips equipped with two types of exposure-programmable pixels are developed to demonstrate the capability of spatial-temporal coded exposure. Summarized in Table 3.1 is a comparison between the two CIS chips introduced in this work and other related CIS designs recently reported from different research groups. Clearly, the CIS presented in this work has a superior per-frame exposure-code mask volume with a relatively high resolution.

	The 1 st CIS	The 2 nd CIS	[50]	[51]	[42]	[40]
Pixel Architecture	CTIA	LEFM	LEFM	CTIA	APS	LEFM
Exposure- Code Memory	1 SRAM	2 DRAMs	2 Flip-Flops	2 DRAMs	1 SRAM	N/A
CMOS Process	0.13 μm	0.13 μm	0.11 μm	0.13 μm	0.18 μm	0.11 μm
Resolution	192×192	128 imes 128	244×162	10×10	127×90	64 imes 108
Pixel Size	H: 12.6 μm W: 12.6 μm	H: 10.2 μm W: 10.2 μm	H: 11.2 μm W: 11.2 μm	H: 12.1 μm W: 12.2 μm	H: 10 μm W: 10 μm	H: 11.2 μm W: 5.6 μm
Fill Factor	38.7%	41.5%	45.3%	33.2%	52%	42%
Frame Rate	30 fps	60 fps	25 fps	60 fps	100 fps	32 fps
Read Noise	4.2 mV	5.4 mV	3.6 mV	5.2 mV	N/A	N/A
Dark Current	1.2 fA	1.3 fA	N/A	1.09 fA	N/A	N/A
Column FPN	0.26%	0.22%	N/A	N/A	1.02%	N/A
Dynamic Range	51.6 dB	47.3 dB	50.5 dB	52 dB	51.2 dB	N/A
On-Chip ADC	10-Bit SS-ADC	8-Bit SS-ADC	N/A	N/A	8-Bit SAR-ADC	1.5-Bit FI/C-ADC
Power FoM*	28.5 nJ**	200.5 nJ	34 nJ	205 nJ	1.14 nJ	7324 nJ
Coded Exposure Type	Spatial- Temporal	Spatial- Temporal	Spatial- Temporal	Spatial- Temporal	Spatial Only	Temporal Only
Number of Taps	2	2	2	2	1	2
Mask Shutter	Global/Rolling	Rolling	Global	Rolling	N/A	N/A
Region of Interest (ROI) Coding	Yes	Yes	Yes	N/A	N/A	No
Code Masks/Frame	768/frame @ 30 fps	500/frame @ 60 fps	7.27/frame @ 25 fps	5000/frame @ 60 fps	1/frame @ 100 fps	15/frame @ 32 fps

*

* Power FoM = Power / [(Total number of pixels) × (Frame rate)]
 ** A maximum number of exposure code mask applied in every frame period.

Table 3.1 Comparison between two implemented CIS chips and other related CIS designs.

Chapter 4: CS-Inspired Machine Vision by Coded Exposure Cameras

In this chapter, two CS based machine vision applications are demonstrated by using the computational cameras presented in Chapter 3. As there are numerous examples of practicing CS in imaging applications, the selection criterion is employing camera coded exposure as the only apparatus to carry out CS. In Section 4.1, compressive high-speed imaging is achieved through spatial-temporal coded exposure of the prototype camera. In Section 4.2, compressive focal-stack photography is implemented by the prototype camera for passive depth sensing.

4.1 Compressive High-Speed Imaging

In conventional digital cameras, the CIS is non-intermittently (continuously) exposed to light. Therefore, there is a fundamental tradeoff between camera spatial resolution and temporal resolution [52]. If increase the camera frame rate (temporal resolution), the resolution of CIS output images (spatial resolution) decreases. One of the main reasons of this problematic tradeoff is the circuitry limitation of CIS. To acquire high-resolution images (e.g. 8K resolution) in a high frame rate (e.g. 120 fps), the image processing circuitry (e.g. image buffers, PGAs, and ADCs) needs to operate at ultra-high speed (>1.0 GHz) with huge power consumption (> 5.0 W). Also, the pixel photodiode suffers from severe noises (e.g. shot noise and thermal noise) due to the short period of exposure time in high frame rate. As a compromise, off-the-shelf digital cameras usually implement tunable high-speed image capture, which increases the frame rate while lower the resolution of output images in a degree that is still acceptable to camera users.

Compressive high-speed imaging is one of many solutions to resolve abovementioned dilemma. Through exploiting the intrinsic redundancy of time-varying appearance of a scene, a

high-speed video is recovered from a single coded image captured by coded-exposure cameras. In the rest of this section, detailed implementation of compressive high-speed imaging on the proposed prototype camera is presented after a briefly discussion of the working principle.



4.1.1 Camera coded exposure and space-time volume reconstruction.

Figure 4.1 Conceptual diagram of compressive high-speed imaging using camera coded exposure.

Depicted in Fig. 4.1 is a conceptual diagram that illustrates the CS pipeline for high-speed image synthesis. Through spatial-temporal coded exposure of the CIS in a frame period (T_{frame}), a coded image is generated by the CIS. The T_{expo} period consists of a number of N sub-periods (T_{chop}) with uniform time duration. As the CIS outputs one image per frame, the scene information encoded in every coded image I(x, y) is described as [52]:

$$\mathbf{I}(x, y) = \sum_{n=1}^{N} \mathbf{M}_{n}(x, y) \odot \mathbf{F}_{n}(x, y)$$
(4.1)

where $\mathbf{F}_n(x, y)$ donates the image from the space-time volume that is generated by using nonintermittent exposure in the *n*th sub-period (T_{chop}). The notation (*x*, *y*) is used to emphasize that each parameter is a 2-D image. $\mathbf{M}_n(x, y)$ represents the exposure-code mask that is applied in the n^{th} sub-period. The operator \odot is the Hadamard product (i.e., element-wise product). For brevity, let us denote Equation (4.1) in the matrix form of: **I=MF**. The target of reconstruction is to estimate the unknown three-dimensional space-time volume **F** from the encoded image $\mathbf{I}(x, y)$ where $\mathbf{F} = {\mathbf{F}_1(x, y), ..., \mathbf{F}_N(x, y)}$. Estimating **F** (space-time volume) from a single image $\mathbf{I}(x, y)$ is an underdetermined linear system, it is a rather challenging problem to solve. Previously reported research in the field of compressive sensing [53] – [55] by relying on sparsity of **F** have shown that this under-determined system can be solved by using the sparse representation $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_k]$ with a dictionary **D** to estimate **F**. The estimated space-time volume model **F** can be written as [52]:

$$\mathbf{F} = \mathbf{D}\boldsymbol{\alpha} = \alpha_1 \mathbf{D}_1 + \alpha_2 \mathbf{D}_2 + \alpha_3 \mathbf{D}_3 + \dots + \alpha_k \mathbf{D}_k$$
(4.2)

where $\alpha_1, \ldots, \alpha_k$ are the sparse coefficients associated with dictionary elements $\mathbf{D}_1, \ldots, \mathbf{D}_k$ where each dictionary element is a space-time volume. The sparse representation can be obtained by solving the following optimization problem [42], [52]:

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\alpha} \|\boldsymbol{\alpha}\|_{0} \quad subject \ to \quad \|\mathbf{M}\mathbf{D}\boldsymbol{\alpha} - \mathbf{I}\|_{2} \le \varepsilon$$

$$(4.3)$$

where $\|\boldsymbol{\alpha}\|_0$ denotes norm 0 of $\boldsymbol{\alpha}$, i.e., total number of non-zero elements in $\boldsymbol{\alpha}$, and ε is the tolerable reconstruction error. Thus, the estimated \mathbf{F} would be $\hat{\mathbf{F}} = \mathbf{D}\hat{\boldsymbol{\alpha}}$. It should be noted that the quality of the reconstructed $\hat{\mathbf{F}}$ is mainly affected by the choice of dictionary \mathbf{D} and sparse representation. In previous works, discrete wavelets (DWT) and discrete cosine transform (DCT) were employed as the transform basis for the sparse coefficients [55] – [56]. Dictionaries trained from a diverse set of videos or generated based on independent and identically distributed entries were also

reported [57]. The patch size of the dictionary is usually constrained in a certain range to optimize the time cost while scene details are still included in the reconstruction result.

4.1.2 Compressive high-speed imaging by the prototype coded exposure camera.

As mentioned in above, the principium of CS in this application is to recover an uncompressed space-time volume $\hat{\mathbf{F}}$ after spatial-temporal coded exposure of a camera. Shown in Fig. 4.2 is an example of image capture using single-shot spatial-temporal coded exposure of the second prototype camera. In this example, a food blender operating at 1000 rpm is captured by the prototype camera operating at 10 fps with T_{expo} and T_{chop} are set to 96.5 ms and 3.01 ms, respectively. Thus, the number of implemented exposure-code masks in a T_{expo} period is 32. Depending on different exposure-code patterns, contents shown in the captured image alter as the result of coded exposure. When a non-intermittent exposure code pattern ($\mathbf{M}_n(x, y) \in \{1\}$) is applied, similar to conventional cameras, the rotating whisk is captured and shown in the output image with severe motion blur. To achieve the best root-mean-squared error and structural similarity performance, as reported in [52], we select pseudo-random binary codes ($\mathbf{M}_n(x, y) \in \{0, \dots, n\}$ 1}) as the code pattern for each exposure-code mask. The resultant coded image reveals mosaiclike patterns, which reveal the coded motion blur of the fast-rotating whisk and confirm the independent coded exposure of each pixel. For space-time volume recovery, we employed a learned over-complete dictionary to estimate the space-time volume model. Since the temporal resolution is 3.01ms, the learned over-complete dictionary was trained by a video collection of moving objects at 320 fps using K-SVD algorithm. Each video consists of a pitch size of 8×8 with rotations in 8 directions and circular replay forward and backward. The output coded image is divided into blocks in size of 8×8 . For each block, a space-time volume of $8 \times 8 \times 32$ is

reconstructed by optimizing Eq. (4.3). After performing block-wise reconstruction, as the result, we recovered a space-time volume of 32 images from the coded image. Each reconstructed image depicts the scene captured in a corresponding T_{chop} period, the rotating whisk is clear to observe with more details and less motion blur.



Figure 4.2 Camera compressive sensing by single-shot spatial-temporal coded exposure. Comparing to a blurry image generated from non-intermittent exposure, a space-time volume of images is reconstructed from a captured coded image.

Image of Static Fan

30fps Video Frames Using Per-Frame Coded Exposure

Reference 30fps Video Using Non-Intermittent Exposure



High-Speed Video Frames from Recovered Space-Time Volumes



Figure 4.3 High frame rate video synthesis using per-frame coded exposure. The prototype CS camera operates at steady frame rate while each output coded frame reconstructs a space-time volume as a part of the final high-speed video.

The single-frame space-time volume recovery is useful in synthesis of high-frame-rate (e.g. 1000 fps) videos while the camera operates at low frame rates (e.g. 30 fps). As a demonstration, we captured four low-frame-rate videos of a 7-blade CPU fan at 30 fps ($T_{frame} = 33.3$ ms) using the prototype CS camera. In this experiment, we render numbers of 8, 16, 24 and 32 exposurecode masks to the CIS, hence the temporal resolution of the reconstructed space-time volume varies from 4.16 ms to 104.2 μ s. Illustrated in Fig. 4.3 are coded frames each from the spatialtemporal coded exposure with N = 8, 16, 24 and 32. Meanwhile, frame images from a reference 30 fps video using non-intermittent exposure is also shown in Fig. 4.3 for comparison. It can be seen in all outputted frame images, fan blades and a symbol "2" written on one of fan blades are blurred out. For those frames from coded exposure, we used the same exposure-code mask patterns and reconstruction procedure. Since the scene frequency is unknown priori, the clarity of highfrequency components shown in the reconstructed frames is determined by the minimum temporal resolution of the recovered space-time volume. Depicted in the bottom-left of Fig. 4.3 are images of four space-time volumes reconstructed from four corresponding coded frames. Through recovering space-time volumes from every coded frame generated by the prototype camera, highspeed videos are synthesized at frame rates of 240 fps, 480 fps, 720 fps and 960 fps, respectively. As expected, in comparison to the reference 30 fps video, it is noticed that severe motion blur is still observable in the 240-fps video but decreases progressively in higher frame-rate videos.

Another benefit offered by the on-sensor spatial-temporal coded exposure is the capability of applying CS on a region of interest (ROI). Through defining a ROI in the exposure-code masks, coded exposure is applied on pixels within the ROI only. As pixels located outside of the ROI experience non-intermittent exposure, the space-time volume recovery concentrates on image data from ROI. Therefore, cameras can efficiently apply CS related techniques on ROI areas and save on decompression power from contents outside of ROI (e.g., static background). Fig. 4.4 depicts an example of ROI based CS on the second prototype camera operating at 30 fps. Shooting on the same scene (the 7-blade CPU fan case), the ROI is defined in each pixel-wise exposure code mask with a size of 70×70 pixels. Note that the exposure codes outside of ROI are set to code "1", which ensure pixels outside of ROI are excluded from coded exposure. After applying 32 exposure code masks in every frame, the generated 30 fps video clearly indicates coded blur in the ROI while areas outside of ROI show typical motion blur. After space-time volume recovery, the reconstructed 960 fps video illustrates motion blur outside of ROI while the rotating fan blades are observable within the ROI, as expected.



Figure 4.4 High-speed video synthesis using region of interest (ROI) based per-frame coded exposure. The size of ROI is 70 pixels by 70 pixels.

4.2 Compressive Focal-Stack Depth Sensing

Depth ranging is another active research topic in machine vision, which has been adopted in various commercial applications such as self-driving cars [58] – [60] and robotics [61]. With the rapid development in 3D imaging technology, coded exposure is also intervened into the process of depth information acquisition. By leveraging CS, computational cameras can draft depth maps in high resolutions with fewer scene measurements.

In optical 3D imaging, based on the physical principle of range computation, depth measurement techniques can be classified into active or passive sensing methodologies. Active sensing employs active light source to illuminate objects in the scene. For instance, a structured light camera projects light patterns on the scene, and object depth geometry is evaluated through careful view alignment and image rectification [62]-[63]. Another example in this category is optical time-of-flight (ToF). By measuring either delayed time or phase shift of the projected light traveling between the camera and scene objects, optical ToF offers a fast-response long-range (up to 6 km) depth sensing using either direct [64] or indirect [37] sensing device. In addition to the power overhead of the active light source, active sensing 3D cameras commonly suffer from the noise from background light [65] and low spatial resolution [66]. Passive range sensing, on the other hand, produces high-resolution depth maps without using active light sources. Currently popular approaches such as stereo vision and depth from focus/defocus (DFF/DFD) use multipoint observation [67] and defocus cues [68], however, they require using multiple cameras (hardware complexity) and costly computations (power overhead).

Compressive focal-stack photography is a passive 3D imaging technique that combines CS and camera focal sweep to reveal a full resolution depth map from a single capture. When under exposure, a camera performs both coded exposure and linear focal sweep. From the output encoded

image, full-resolution scene depth map is acquired through exploitation of sparsity in the coded focal stack. In comparison to other range sensing approaches, as no need of active light source nor multi-position measurements, the proposed 3D imaging solution offers both power efficiency and compact footprint, which are desirable attributes for applications with energy and space constraints, e.g, portable devices. As a demonstration, the rest of section elaborates the implementation of compressive focal-stack imaging on the prototype coded-exposure camera.

4.2.1 Compressive focal stack and depth from defocused images



Figure 4.5 Conceptual diagram of a compressive focal-stack depth sensing system.

The focal stack depth sensing has been developed over the recent decades and has been used in many subfields of scientific imaging [69]. It is a passive depth sensing approach and it is based on the fact that objects distanced from the observer appear in focus for a certain optical power. Scene depth map is usually acquired after estimation of sharpness in the focal stack produced by focal 77

adjustments [70]. To improve accuracy and resolution of the reconstructed depth map, usually a large number of defocused images in the focal stack are required. Therefore, the image sensor should capture a significant number of images, resulting in a slow production of the focal stack and an exponential growth in power spent on image processing. By introducing coded-exposure based CS, the number of images required in the focal stack can be greatly reduced. The compressive focal-stack 3D imaging, as depicted in Fig. 4.5, is described here with results demonstrated in Section V. The implementation procedure is similar to that used in [71] and [72] is summarized in the following steps:

- 1) At the start of a frame period, the CIS chip and camera lens are reset to their initial states.
- 2) Next, during each camera exposure period, the position of the camera lens is swept to produce different focal depths in the range of (z_1, z_2) on the depth coordinate, *z*. Meanwhile, the camera implements coded exposure by utilizing number of N exposure-code masks.
- 3) After the exposure period, pixels on the CIS are read out to output the corresponding coded image.
- 4) By following the method in [71], through exploiting the intrinsic sparsity in the focal stack and using off-chip image processing, a focal stack is reconstructed from the coded image. The focal stack I_n , $n \in \{1, 2, ..., N\}$, contains N defocused images and each of which has the same image size as the coded image.
- 5) For each defocused image in the recovered focal stack, compute the focus metric \mathbf{F}_n (the criterion for sharpness). Similar to [72], the focus metric is calculated by filtering the defocused image with a Laplacian of Gaussian.
- 6) Evaluate the focus metric of each defocused image and generate the index map **M**. Each element of **M** at coordinate (x, y) is the index *n* for which the value of pixel p(x, y) in \mathbf{F}_n is

maximum: $\mathbf{M}(x, y) = \underset{n \in \{1, 2, ..., N\}}{\operatorname{arg max}} \mathbf{F}_n(\mathbf{p}(x, y))$, where $\mathbf{p}(x, y)$ is the pixel in the defocused image

at coordinates (x, y).

- 7) Given the index map, a corresponding optical power map is synthesized by associating every index in $\mathbf{M}(x, y)$ to its corresponding optical power value.
- 8) Finally, from the optical power map, the depth map is extracted by converting the optical power values into depth values on the depth coordinate z.

The set of exposure masks utilized in this CS approach is served as a modulation function to encode light signals before converting to image data. In case of global coded exposure, the CIS experiences spatial-temporal coded exposure.

4.2.2 Compressive focal-stack photography by the prototype coded exposure camera

The experiment of compressive focal-stack imaging is carried out by the first prototype camera introduced in Chapter 3. Through control signal synchronization, the VCM lens performs focal sweep while the CIS is in spatial-temporal coded exposure. During the focal sweep, the optical power varies and scene objects lying within the focal range brought in and out of focus. The depth map is constructed by sorting out the sharpness of images in the recovered focal stack. Depicted in Fig. 4.6 is the image processing pipeline for compressive focal-stack depth ranging. The CIS operates at a speed of 30 fps. In a frame period, the VCM lens sweeps a range of scene depth of 40 cm while 4 exposure-code masks are applied during T_{expo} with code patterns formed by random binary sequences. By following the method reported in [71], a focal stack consisting of 4 defocused images are recovered after 10 iterations of sparse reconstruction on the generated coded image. In each of those defocused images, a specific focal depth is interested and a toy doll is in focus.

Similar to the method introduced in [72], by calculating the focus metric and sorting out maximum sharpness, depth values are associated to pixels and a depth map is synthesized.



Figure 4.6 The experimental result of the proposed on-chip compressive focal-stack depth sensing.

4.3 Chapter Conclusion

In this chapter, two examples of CS-based computational imaging applications are presented in use of two prototype coded-exposure cameras developed in this research. As the on-going development of CS algorithms, novel coded-exposure related machine vision applications will emerge and implemented by the proposed coded-exposure camera. Those imaging innovation related topics, however, are out of the scope of this research.

Through camera spatial-temporal coded exposure followed by image sparse reconstruction, high-speed video frames are synthesized from a single coded image. In comparison to conventional high-speed cameras and other state-of-the-art CS cameras, the proposed on-chip CS solution saves system power consumption and enhances camera's temporal resolution without any compromise on spatial resolution. In association of the spatial-temporal coded exposure with a focal sweep, the prototype camera captures a coded image revealing a defocused focal stack after decompression. Through evaluation of sharpness of defocused images, a depth map is synthesized from the defocused focal stack. In comparison to an equivalent implementation of focal-stack imaging using high-volume image frames in focal-stack synthesis, the proposed on-chip solution introduces CS to greatly reduce the output data rate and power dissipation in a camera system. The realization of on-chip compressive focal-stack imaging facilitates a frame-saving path to passive depth sensing in the machine vision paradigm.

Chapter 5: Conclusion and Future Work

5.1 **Conclusion of this work**

In this thesis, a comprehensive study on the on-chip implementation of CS for machine vision applications is presented. The proposed computational camera hardware includes a proposed CIS design capable of spatial-temporal coded exposure, which has extended CS to sensor nodes and is seamlessly integrated into the reset-exposure-readout based operation flow of the conventional CIS devices. The key contributions of this research are summarized below:

- 1. Development of exposure-programmable pixels: The proposed pixel architectures include an in-pixel charge modulator and exposure-code memory cells. During camera exposure periods, pixels are tunable between a conventional continuous-exposure mode and operations of exposure-code guided intermittent exposure. In detailed circuit implementations, four types of pixel designs are presented and are analyzed and measured. These developments in the pixel design is the corner stone of merging coded-exposure features into a CIS.
- 2. Development of coded exposure CIS: Based on conventional CIS designs, the architecture of the coded exposure CIS adds in multiple control blocks for exposure-code delivery and refreshments on the pixel array. The column-wise signal processing block is also modified to handle pixel output variations in different types of exposure. In circuit implementation, the proposed CIS designs are fabricated in a CMOS process used in many CIS products on the market. Two prototype computational cameras are assembled as test chambers to verify the performance of the fabricated CIS chips as well as their capabilities in spatial-temporal coded exposure. Compared to other related works reported in recent years, experimental

results indicate that the CIS in this research has a superior per-frame exposure-code mask volume and a relatively high resolution.

- 3. Introduction of adaptive image processing in the signal processing blocks of CIS: When the scene brightness changes, the pixel output signal varies. The operation of coded exposure further increases the pixel output variations which makes negative impacts on conversion speed and power consumption of ADCs. The proposed adaptive ADC design includes a pixel output sensing feature to adjust ADCs to work in a proper mode. Such a power-efficient signal processing methodology is tailored for the CISs when the coded exposure is enabled.
- 4. Demonstrations of CS-based computational imaging applications realized by on-chip coded exposure: Two examples are presented to show the practicality of on-chip coded exposure in realization of CS-based imaging applications. In the demonstration of compressive high-speed imaging, by establishing spatial-temporal coded exposure on the CIS, camera's temporal resolution is enhanced while its spatial resolution is maintained. The compressive focal-stack photography is carried out through concurrent operations of coded exposure and camera focal sweep. Through evaluation of reconstructed defocused focal stack, the scene depth information is acquired in a passive way.

As the research in this area is in a fast pace along with the advances in machine vision and artificial intelligence, more achievements in on-chip CS and computational imaging are about to happen over the next few decades. In the rest of this chapter, some perspectives on future developments and potential technological advances are summarized.

5.2 Future Development and Technological Advances

For the future development, potential improvements can be contributed from the following three levels:

- Circuit-level developments: The CIS design presented in this work only accepts binary exposure codes. In other words, the coded exposure practiced on the CIS is in a two-tap basis. Therefore, the target of future developed CISs is enabling multi-level exposureprogramming in pixel designs.
- 2. System/chip-level improvements: The current CIS design, either in off-the-shelf CIS products or the CIS presented in this thesis, relies on off-chip image computation. All CS-inspired machine vision applications require off-chip graphical processors to process images generated by the CIS, resulting in huge costs of power and time. Hence, the developing direction on this topic is to incorporate image processing into an application specific integrated circuit (ASIC) module inside the CIS architecture.
- 3. Application-level innovations: From the research works published over the last decade, CS related applications mainly focus on high-speed imaging, HDR, and enhancement of 3D sensing. As the computation power required for emerging machine vision applications is fast-growing, new applications using the CS approach appear to bring more needs and opportunities to the coded exposure camera.

In the rest of this section, two technological advances on the circuit-level and the chip-level are discussed. The predictions on the innovative imaging applications are expected to be given by computer scientists.

5.2.1 Multi-level spatial-temporal coded exposure

Since the charge modulator in the proposed pixel design is a 2-tap design, the proposed CIS in the prototype camera only accepts binary coded exposure. In other words, the exposure-code mask pattern, which represents the modulation function in CS, is limited to binary codes. To improve the exposure-programmability of the pixel design, both charge modulator and exposure-code memory have to be re-designed to enable multi-level coded exposure. Illustrated in Fig. 5.1 are timing diagrams of the 2-tap binary coded exposure enabled by the proposed pixel design and an example of 3-tap multi-level coded exposure that can be realized for the future pixel designs. During the exposure period T_{expo} , the exposure-code signal (φ_{code}) toggles between a high voltage level and the ground, giving binary exposure codes to guide pixel exposure. The voltage level of the charge modulator (V_c) gradually decrease from V_{rst} to lower levels with an identical slope (Fig. 5.1(a)). In the multi-level coded exposure (Fig. 5.1(b)), on the other hand, φ_{code} is a multi-level signal which toggles between four different voltage levels. Each of φ_{code} level represents a unique exposure code, defining the voltage decreasing slope of V_c . As a result, the final pixel output V_{out} is different than that of binary coded exposure due to the multi-slope charge modulation.



Figure 5.1 Time diagrams of (a) 2-tap binary coded exposure and (b) 3-tap multi-level coded exposure.

The implementation of multi-level coded exposure is more complex than that of the binary version. Firstly, the pixel needs to accept multi-level exposure-code signals. One potential solution is to use pulse-amplitude modulation (PAM) as the formation of φ_{code} . However, it would dramatically increase the complexity of the exposure-code transmitter in each column as φ_{code} requires multi-level voltage encoding. Another potential solution is to formulate φ_{code} as an exposure-code delivery bus, which delivers multi-level exposure codes in binary forms. The biggest disadvantage of this solution would be the large layout area required to place bus wires. However, this could be overcome by employing wafer stacking available in the CIS fabrication processes. Secondly, the multi-level code exposure requires a multi-bit exposure-code memory unit and a multi-level charge modulator. One potential pixel architecture to achieve the goal is shown in Fig. 5.2, where the pixel design includes a number of M charge storage units and exposure code memory cells.



Figure 5.2 Block diagram of a potential pixel design for M-level coded exposure.

The photodetector in this potential design is connected to a charge modulator consisting of M charge storage units. Each charge storage unit accesses to the photodetector through a valve switch $(S_1....S_M)$ controlled by the exposure code stored in the corresponding exposure-code memory. The exposure-code delivery bus streams in exposure codes for each exposure-code memory during a T_{chop} period. Hence, charges generated by the photodetector flow into different charge storage units based on the status of delivered exposure codes, enabling the multi-level coded exposure. In pixel readout period, charges stored in selected charge storage units are read out for further processing while other charge storage units are reset at the beginning of next frame period. The output signal in this potential pixel design can be a single wire delivering a voltage level represents a summation of all charges from selected charge storage units, or an output bus contains multiple outputs generated from different charge storage units.

The circuit implementation of the abovementioned potential design solution for a multi-level coded-exposure pixel is illustrated in Fig. 5.3. In this example, the charge modulator is based on a modified LEFM structure which contains M capacitive charge storage units controlled by exposure codes stored in the exposure-code memory. For different exposure codes, charges generated by the photodiode are transferred through selectively enabled valve switches ($M_{val1}...M_{valM}$). When the exposure period ends, different volume of charges is accumulated in the integrating capacitors ($C_1...C_M$). The type of readout circuit is a classical 4-T APS based source follower, which reads out charges integrated in each charge storage unit and send out of pixels for further processing.

As the exposure-code memory cell controls only one valve switch, the memory structure needs to be modified accordingly. Shown in Fig. 5.4 is a potential exposure-code memory design using a SRAM structure. In this solution, an inverter has been inserted to generate the complementary code signal of φ_{code} , and the memory core is connected to one valve switch only.


Figure 5.3 Circuit diagram of a potential pixel design for M-level coded exposure.



Figure 5.4 Circuit diagram of a potential design of a SRAM based exposure-code memory cell for multi-level exposure-programmable pixels.

In multi-level coded exposure, it is possible that the exposure code is a negative code. In other words, the direction of charge flow between the photodetector and the charge storage unit is manipulatable by applying exposure codes in different polarity. When a negative exposure code is in effective, charges accumulated in the charge storage unit flows out. Therefore, the voltage level variation in the charge storage is opposite to that of applying positive exposure codes. To realize such negative coded exposure, a potential solution is to flip the operation of a photodetector from the photo-sensing mode to the photo-voltaic mode. Depicted in Fig. 5.5(a) is a pixel architecture design which performs coded exposure in both polarities. The exposure-code memory cell stores both multi-level exposure codes and the code polarity. The level select feature is realized by the M-level charge storage unit design described in above. The polarity selection is implemented through working-mode switching of the photodiode. Five transistors are employed to flip the polarity of photodiode, which works as a photodiode when the exposure code is positive (pull up $\varphi_{code(+)}$) and operates as a solar cell once the exposure code is in negative (assert $\varphi_{code(-)}$). Noticed

that a NOR gate is employed to prevent polarity confusions. Shown in Fig. 5.5(b) is a simulation result of applying exposure codes in different levels and polarities. From this experimental result, it is clear to see the voltage level of a charge storage unit increase or decrease based on the code polarity. Also, the voltage-variation slope in both cases is based on the code level applied to the charge storage unit.



Figure 5.5 (a) A potential pixel design accepts both positive and negative exposure codes for multi-level coded exposure. (b) Time diagram of multi-level coded exposure with different exposure code polarities.

5.2.2 System-on-a-chip machine vision

The concept of system on a chip (SoC) was initially inspired by very-large-scale integrated (VLSI) circuits such as central processing unit (CPU) and graphics processing unit (GPU), where multiple sub-systems are integrated to form a multi-task processing platform. In most of the present CIS design, chip functionality remains as its original definition: converting optical signals to electrical signals in an array form. CISs output raw images, which are processed by image signal processors (ISPs) to synthesize readable images. Due to the limited space and CMOS process restrictions on CISs, imaging applications in machine vision are implemented by discrete ISP chips.



Figure 5.6 An example of potential CIS-SoC design using BSI technology and multi-wafer stacking for emerging machine vision applications.

To resolve the bandwidth limitation issue for transferring data between the chips and to reduce power, a desired solution is to integrate features of the ISP chip into the CIS chip. Such CIS-SoC design, however, requires sophisticated on-chip system integration and encounters routing hardships in chip layout. Therefore, recently developed CISs start to use the backside illumination (BSI) technology with multi-wafer stacking in the fabrication process. As shown in Fig. 5.6, three wafer layers are stacked in a sandwich-like structure. The first wafer layer holds the pixel array on one side while all pixel readout circuits are placed on the other side of the layer, giving pixels a fill factor of 100%. The second layer called memory layer is used to buffer pixel outputs waiting for signal processing. The bottom layer, which includes all signal processing systems, is the digital signal processing (DSP) layer. The biggest advantage of using wafer stacking is to enlarge effective silicon usage spaces while maintain chip's 2D dimension. By insertion of a ISP core into the DSP layer, raw image signals are processed directly on a CIS chip. Such ASIC-like implementation makes the CIS-SoC design accepts various imaging processing for applications in machine vision. For example, for the coded-exposure based CS applications, the exposure-code memory cells can be placed on the memory layer and the sparse reconstruction algorithms can be implemented by the ISP core on the DSP layer. The chip outputs, instead of encoded images, are decoded images in standard image format which is directly interpreted by other devices and becomes readable to users.

Bibliography

- M. Levoy and P. Hanrahan, "Light field rendering," *in Proc. SIGGRAPH* 96, 1996, pp. 31–42.
- [2] R. Ng, "Fourier slice photography," ACM Trans. Graph., vol. 24, pp.735–744, July 2005.
- [3] A. Veeraraghavan, R. Raskar, A. Agrawal, A. Mohan, and J. Tumblin, "Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing," in *Proc. SIGGRAPH*, 2007, pp.1–14.
- [4] D. Lanman, R. Raskar, A. Agrawal, and G. Taubin, "Shield fields: Modeling and capturing 3D occluders," in *Proc. SIGGRAPH*, 2008, pp. 1–10.
- [5] C. Liang, T. Lin, B.Wong, C. Liu, and H. Chen, "Programmable aperture photography: Multiplexed light field acquisition," in *Proc. SIGGRAPH*, 2008, pp. 1–55.
- [6] K. Yamazawa, Y. Yagi, and M. Yachida, "Omnidirectional imaging with hyperboloidal projection," in *Proc. Int. Conf. Robots Syst.*, 1993, pp.1029–1034.
- S. Baker and S. K. Nayar, "A theory of single-viewpoint catadioptric image formation," *Int. J. Comput. Vis.*, vol. 35, no. 2, pp. 1–22, 1999.
- [8] S. Peleg, M. Ben-Ezra, and Y. Pritch, "Omnistereo: Panoramic stereo imaging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 3, pp. 279–290, Mar. 2001.
- [9] B. Trifonov, D. Bradley, and W. Heidrich, "Tomographic reconstruction of transparent objects," in *Proc. ACM SIGGRAPH Sketches*, 2006, pp. 1–55.
- [10] D. H. Lee, I. S. Kweon, and R. Cipolla, "Single lens stereo with a biprism," in *Proc. IAPR Int. Workshop Mach. Vis. Appl.*, 1998, pp. 136–139.
- [11] Pentland, "A new sense for depth of field," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 4, pp. 523–531, Jul. 1987.
- [12] M. Subbarao and N. Gurumoorthy, "Depth recovery from blurred edges," in Proc. Conf. Comput. Vis. Pattern Recognit., 1988, pp. 498–503.
- [13] A. Levin, R. Fergus, F. Durand, and W. T. Freeman, "Image and depth from a conventional camera with a coded aperture," in *Proc. SIGGRAPH*, 2007, pp. 1–70.

- [14] C. Zhou, O. Cossairt, and S. K. Nayar, "Depth from diffusion," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1110–1117.
- [15] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information." *IEEE Transactions on information theory*, Vol. 52(2), pp. 489–509, 2006.
- [16] R. Raskar, A. Agrawal, and J. Tumblin, "Coded exposure photography: Motion deblurring using fluttered shutter," in *Proc. ACM SIGGRAPH*, 2006, pp. 795–804.
- [17] B. T. Bosworth, J. R. Stroud, D. N. Tran, T. D. Tran, S. Chin, and M. A. Foster, "High-speed compressed sensing measurement using spectrally-encoded ultrafast laser pulses," in *Proc. IEEE Information Sciences and Systems*, 2015.
- [18] L. Gao, J. Liang, C. Li, and L. V. Wang, "Single-shot compressed ultrafast photography at one hundred billion frames per second," Nature, vol. 516, no. 7529, pp. 74–77, 2014.
- [19] Y. Li, M. Tofighi, J. Geng, V. Monga and Y. C. Eldar, "Efficient and Interpretable Deep Blind Image Deblurring Via Algorithm Unrolling," *IEEE Transactions on Computational Imaging*, vol. 6, pp. 666-681, 2020.
- [20] T. Portz, L. Zhang, and H. Jiang, "Random coded sampling for high-speed HDR video," in Proc. IEEE International Conference on Computational Photography, 2013.
- [21] D. J. Griffiths, and A. Wicks, "High Speed High Dynamic Range Video", Sensors Journal IEEE, vol. 17, no. 8, pp. 2472-2480, 2017.
- [22] H. Nagahara, D. Liu, T. Sonoda, and J. Gu, "Space-Time-Brightness Sampling Using an Adaptive Pixel-Wise Coded Exposure", in *Proc. Computer Vision and Pattern Recognition Workshops (CVPRW) 2018 IEEE/CVF Conference on*, pp. 1915-19158, 2018.
- [23] J. N. P. Martel, L. K. Muller, S. J. Carey, P. Dudek, and G. Wetzstein, "Neural Sensors: Learning Pixel Exposures for HDR Imaging and Video Compressive Sensing with Programmable Sensors", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 42, no. 7, pp. 1642-1653, 2020.
- [24] C. A. Metzler, H. Ikoma, Y. Peng, and G. Wetzstein, "Deep Optics for Single-Shot High-Dynamic-Range Imaging" in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 1375-1385.

- [25] S. Kuthirummal, H. Nagahara, C. Zhou, and S. K. Nayar, "Flexible depth of field photography," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 33(1), pp: 58–71, 2011.
- [26] O. Cossairt, N. Matsuda, and M. Gupta, "Digital refocusing with incoherent holography", in Proc. IEEE International Conference on Computational Photography, 2014, pp. 1-9.
- [27] X. Yuan, "Compressive dynamic range imaging via Bayesian shrinkage dictionary learning", *Optical Engineering*, vol. 55, pp. 123110, 2016.
- [28] M. Sheinin, and Y. Y. Schechner, "Depth from Texture Integration", in *Proc. IEEE International Conference on Computational Photography*, 2019, pp. 1-10.
- [29] C. Zhou and S. K. Nayar, "Computational Cameras: Convergence of Optics and Processing," in *IEEE Transactions on Image Processing*, vol. 20, no. 12, pp. 3322-3340, 2011.
- [30] B. Bascle, A. Blake, and A. Zisserman, "Motion deblurring and super-resolution from an image sequence. in *Proc. European Conference on Computer Vision*, 1996, pp. 573–582.
- [31] S.K. Nayar, V. Branzoi, and T.E. Boult, "Programmable Imaging: Towards a Flexible Camera," *IEEE Int'l J. Computer Vision*, Vol. 70, no. 1, pp. 7-22, 2006.
- [32] D. Reddy, A. Veeraraghavan, and R. Chellappa, "P2C2: Programmable Pixel Compressive Camera for High Speed Imaging," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011, pp. 329-336.
- [33] S. Ri, Y. Matsunaga, M. Fujigaki, T. Matui, and Y. Morimoto, "Development of DMD Reflection-Type CCD Camera for Phase Analysis and Shape Measurement," *J. Robotics and Mechatronics*, Vol. 18, no. 6, pp. 728, 2006.
- [34] H. Nagahara, C. Zhou, T. Watanabe, H. Ishiguro, and S.K. Nayar, "Programmable Aperture Camera Using LCoS," in *Proc. European Conf. Computer Vision*, 2010, pp. 337-350.
- [35] R. Schwarte, H. Heinol, Z. Xu, K. Hartmann, "New active 3D vision system based on rfmodulation interferometry of incoherent light," in *Proc. SPIE Intelligent Robots and Computer Vision XIV: Algorithms, Techniques, Active Vision, and Materials Handling*, 1995.
- [36] O. Atalar, R. V. Laer, C. J. Sarabalis, A. H. Safavi-Naeini, and A. Arbabian, "Time-of-flight imaging based on resonant photoelastic modulation," Appl. Opt., Vol. 58, pp. 2235-2247, 2019.

- [37] C. S. Bamji *et al.*, "A 0.13 μm CMOS System-on-Chip for a 512 × 424 Time-of-Flight Image Sensor With Multi-Frequency Photo-Demodulation up to 130 MHz and 2 GS/s ADC," *IEEE Journal of Solid-State Circuits*, vol. 50, no. 1, pp. 303-319, 2015.
- [38] T. Hsu, T. Liao, N. Lee and C. Hsieh, "A CMOS Time-of-Flight Depth Image Sensor With In-Pixel Background Light Cancellation and Phase Shifting Readout Technique," *IEEE Journal of Solid-State Circuits*, vol. 53, no. 10, pp. 2898-2905, 2018.
- [39] G.Wan, X. Li, G. Agranov, M. Levoy, and M. Horowitz, "CMOS Image Sensor with Multi-Bucket Pixels for Computational Photography," J. Solid-State Circuits, Vol. 47, no.4, pp. 1031-1042, 2012.
- [40] F. Mochizuki, K. Kagawa, S. Okihara, M.W. Seo, B. Zhang, and T. Takasawa, "Single-Shot 200Mfps 5×3-Aperture Compressive CMOS Imager," in *Proc. IEEE ISSCC Dig. Tech. Papers*, 2015, pp. 116–118.
- [41] Y. Shirakawa, K. Yasutomi, K. Kagawa, S. Aoyama, and S. Kawahito, "An 8-Tap CMOS Lock-In Pixel Image Sensor for Short-Pulse Time-of-Flight Measurements," *Sensors*, Vol. 20(4): 1040, 2020.
- [42] J. Zhang, T. Xiong, T. Tran, S. Chin, and R. Etienne-Cummings, "Compact all-CMOS Sptiotemporal Compressive Sensing Video Camera with Pixel-Wise Coded Exposure," *Optics Express*, Vol. 24, no. 8, 2016.
- [43] S. Kawahito, G. Baek, Z. Li, S.M. Han, M.-W. Seo, K. Yasutomi, and K. Kagawa, "CMOS Lock-in Pixel ImageSensors with Lateral Electric Field Control for Time-Resolved Imaing" in *Proc. International Image Sensor Workshop*, 2013.
- [44] S. Han, T. Takasawa, K. Yasutomi, S. Aoyama, K. Kagawa and S. Kawahito, "A Time-of-Flight Range Image Sensor With Background Canceling Lock-in Pixels Based on Lateral Electric Field Charge Modulation," *IEEE Journal of the Electron Devices Society*, Vol. 3, no. 3, pp. 267-275, 2015.
- [45] M. Seo et al., "A 10 ps Time-Resolution CMOS Image Sensor with Two-Tap True-CDS Lock-In Pixels for Fluorescence Lifetime Imaging," *IEEE Journal of Solid-State Circuits*, Vol. 51, no. 1, pp. 141-154, 2016.

- [46] K. Murari, R. Etienne-Cummings, N. V. Thakor and G. Cauwenberghs, "A CMOS In-Pixel CTIA High-Sensitivity Fluorescence Imager," *IEEE Transactions on Biomedical Circuits* and Systems, Vol. 5, no. 5, pp. 449-458, 2011.
- [47] B. Liu and J. Yuan, "A Quantum-Limited Highly Linear Monolithic CMOS Detector for Computed Tomography," *IEEE Transactions on Circuits and Systems I: Regular Papers*, Vol. 59, no. 3, pp. 566-574, 2012.
- [48] R. Xu, B. Liu, and J. Yuan, "A 1500 fps Highly Sensitive 256 × 256 CMOS Imaging Sensor with In-Pixel Calibration", *IEEE Journal of Solid-State Circuits*, Vol. 47, no. 6, pp. 1408-1418, 2012.
- [49] R. Xu, W. C. Ng, J. Yuan, S. Yin, and S. Wei, "A 1/2.5 inch VGA 400 fps CMOS Image Sensor With High Sensitivity for Machine Vision", *IEEE Journal of Solid-State Circuits*, Vol. 49, no. 10, pp. 2342-2351, 2014.
- [50] N. Sarhangnejad *et al.*, "Dual-Tap Computational Photography Image Sensor With Per-Pixel Pipelined Digital Memory for Intra-Frame Coded Multi-Exposure," *IEEE Journal of Solid-State Circuits*, Vol. 54, no. 11, pp. 3191-3202, 2019.
- [51] Y. Luo, D. Ho and S. Mirabbasi, "Exposure-Programmable CMOS Pixel with Selective Charge Storage and Code Memory for Computational Imaging," *IEEE Transactions on Circuits and Systems I: Regular Papers*, Vol. 65, no. 5, pp. 1555-1566, 2018.
- [52] D. Liu, J. Gu, Y. Hitomi, M. Gupta, T. Mitsunaga and S. K. Nayar, "Efficient Space-Time Sampling with Pixel-Wise Coded Exposure for High-Speed Imaging," *IEEE Transactions* on Pattern Analysis and Machine Intelligence, Vol. 36, no. 2, pp. 248-260, 2014.
- [53] D. L. Donoho, M. Elad and V. N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Transactions on Information Theory*, Vol. 52, no. 1, pp. 6-18, 2006.
- [54] E.J. Candes, J. Romberg, and T. Tao, "Stable Signal Recovery from Incomplete and Inaccurate Measurements," *Comm. Pure and Applied Math.*, Vol. 59(8), pp. 1207-1223, 2006.
- [55] M. Elad and M. Aharon, "Image Denoising via Learned Dictionaries and Sparse Representation," in Proc. *IEEE Conf. Computer Vision and Pattern Recognition*, 2006, pp. 895-900.

- [56] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. Kelly, and R. Baraniuk, "Compressive Imaging for Video Representation and Coding," in *Proc. IEEE PCS*, 2006, pp. 1-7.
- [57] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *IEEE Trans. Signal Processing*, Vol. 54(11), pp. 4311-4322, 2006.
- [58] J. Levinson, J. Askeland and J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, "Towards fully autonomous driving: Systems and algorithms," in *Proc. IEEE Intelligent Vehicles Symposium*, 2011, pp. 163–168.
- [59] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J. M. Allen, V. D. Lam, A. Bewley, and A. Shah, "Learning to Drive in a Day", in *Proc. International Conference on Robotics and Automation*, 2019, pp. 8248-8254.
- [60] Y. Lyu, L. Bai, X. Huang, "ChipNet: Real-Time LiDAR Processing for Drivable Region Segmentation on an FPGA", *IEEE Transactions on Circuits and Systems I: Regular Papers*, Vol. 66, no. 5, pp. 1769-1779, 2019.
- [61] K. Gallo and G. Assanto, "Vision based obstacle detection for wheeled robots," in Proc. International Conference on Control, Automation and Systems, 2008, pp. 1587–1592.
- [62] Q. Du, R. Liu, and Y. Pan, "Depth extraction for a structured light system based on mismatched image pair rectification using a virtual camera," *IET Image Process.*, Vol. 11, no. 11, pp. 1086–1093, 2017.
- [63] A.P. Harrison, C.M. Wong, and D. Joseph, "Virtual reflected-light microscopy," *Journal of Microscopy*, Vol.244, pp: 293-304, 2011.
- [64] C. Niclass, M. Soga, H. Matsubara, M. Ogawa, and M. Kagami, "A 0.18-×m CMOS SoC for a 100-m-range 10-frame/s 200 × 96-pixel time-of-flight depth sensor," *IEEE J. Solid-State Circuits*, Vol. 49, no. 1, pp. 315–330, 2014.
- [65] X. Huang *et al.*, "Polarimetric target depth sensing in ambient illumination based on polarization-coded structured light," *Appl. Opt.*, Vol. 56, no. 27, pp. 7741–7748, 2017.
- [66] R. Lange and P. Seitz, "Solid-state time-of-flight range camera," *IEEE J. Quantum Electron.*, Vol. 37(3), pp. 390–397, 2001.

- [67] H.-H. Chen, C.-T. Huang, S.-S. Wu, C.-L. Hung, T.-C. Ma, and L.-G. Chen, "A 1920×1080 30 fps 611 mW five-view depth-estimation processor for light-field applications," in *IEEE Int. Solid-State Circuits Conf.: Dig. Tech. Papers*, 2015, pp. 1–3.
- [68] P. Favaro, "Recovering thin structures via nonlocal-means regularization with application to depth from defocus," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1133-1140.
- [69] M. S. Sigdel, M. Sigdel, S. Dinç, I. Dinc, M. L. Pusey and R. S. Aygün, "FocusALL: Focal Stacking of Microscopic Images Using Modified Harris Corner Response Measure," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, Vol. 13, no. 2, pp. 326-340, 2016.
- [70] S. Nayar and Y. Nakagawa, "Shape from focus," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 16(8), pp. 824–831, 1994.
- [71] X. Lin, J. Suo, G. Wetzstein, Q. Dai, and R. Raskar, "Coded Focal Stack Photography," in Proc. IEEE International Conference on Computational Photography, 2013, pp.1-9.
- [72] J. N. P. Martel, L. K. Müller, S. J. Carey, J. Müller, Y. Sandamirskaya and P. Dudek, "Real-Time Depth From Focus on a Programmable Focal Plane Processor," in *IEEE Transactions* on Circuits and Systems I: Regular Papers, vol. 65, no. 3, pp. 925-934, 2018.
- [73] J. Ohta, Smart CMOS Image Sensors and Applications. Boca Raton, FL, USA: CRC Press, 2008.
- [74] K. Murari, R.E. Cummings, N. Thakor, and G. Cauwenberghs, "Which Photodiode to Use: A Comparison of CMOS-Compatible Structures," *IEEE Sensor Journal*, Vol. 9, no.7, pp. 752-760, 2009.
- [75] B. Fowler, M. D. Godfrey, and S. Mims, "Reset noise reduction in capacitive sensors," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 53, no. 8, pp. 1658–1669, Aug. 2006.
- [76] E. R. Fossum, "CMOS image sensors: electronic camera on a chip," in Proceedings of International Electron Devices Meeting, pp. 17–25.
- [77] J. Choi, S. Park, J. Cho, and E. Yoon, "A 1.36µW adaptive CMOS image sensor with reconfigurable modes of operation from available energy/illumination for distributed wireless sensor network," in *Proc. 2012 IEEE International Solid-State Circuits Conference*, 2012, pp. 112–114.

- [78] B. C. Burkey, W. C. Chang, J. Littlehale, T. H. Lee, T. J. Tredwell, J. P. Lavine, and E. A. Trabka, "The pinned photodiode for an interline- transfer CCD image sensor," in *Proc. Int. Electron Devices Meeting*, 1984, pp. 28–31.
- [79] P. Lee, R. Gee, M. Guidash, T. Lee, and E. R. Fossum, "An active pixel sensor fabricated using CMOS/CCD process technology," in *Proc. IEEE Workshop on CCDs and Advanced Image Sensors*, 1995, pp. 115–119.
- [80] D. Kim, Y. Chae, J. Cho, and G. Han, "A Dual-Capture Wide Dynamic Range CMOS Image Sensor Using Floating-Diffusion Capacitor," *IEEE Trans. Electron Devices*, Vol. 55, no. 10, pp. 2590–2594, 2008.
- [81] O. Skorka and D. Joseph, "CMOS digital pixel sensors: technology and applications," Proceedings of SPIE 9060, Nanosensors, Biosensors, and Info-Tech Sensors and Systems, Vol. 9060, pp: 1-15, 2014.
- [82] A. Boukhayma, A. Peizerat, and C. Enz, "A Sub-0.5 Electron Read Noise VGA Image Sensor in a Standard CMOS Process," *IEEE J. Solid-State Circuits*, Vol. 51, no. 9, pp. 2180–2191, Sep. 2016.
- [83] C. Lotto, P. Seitz, and T. Baechler, "A sub-electron readout noise CMOS image sensor with pixel-level open-loop voltage amplification," in *Proc. IEEE International Solid-State Circuits Conference*, 2011, pp. 402–404.
- [84] D. Kim, Y. Chae, J. Cho, and G. Han, "A Dual-Capture Wide Dynamic Range CMOS Image Sensor Using Floating-Diffusion Capacitor," *IEEE Trans. Electron Devices*, Vol. 55, no. 10, pp. 2590–2594, Oct. 2008.
- [85] S.-W. Han, S.-J. Kim, J. Choi, C.-K. Kim, and E. Yoon, "A High Dynamic Range CMOS Image Sensor with In-Pixel Floating-Node Analog Memory for Pixel Level Integration Time Control," in Proc. Symposium on VLSI Circuits, Digest of Technical Papers, 2006, pp. 25–26.
- [86] S. Kavadias, B. Dierickx, D. Scheffer, A. Alaerts, D. Uwaerts, and J. Bogaerts, "A logarithmic response CMOS image sensor with on-chip calibration," *IEEE J. Solid-State Circuits*, Vol. 35, no. 8, pp. 1146–1152, 2000.
- [87] S. Yoshihara *et al.*, "A 1/1.8-inch 6.4 MPixel 60 frames/s CMOS Image Sensor With Seamless Mode Change," *IEEE J. Solid-State Circuits*, Vol. 41, no. 12, pp. 2998–3006, Dec. 2006.
- [88] S. Lim *et al.*, "A 240-frames/s 2.1-Mpixel CMOS Image Sensor with Column-Shared Cyclic ADCs," *IEEE J. Solid-State Circuits*, Vol. 46, no. 9, pp. 2073–2083, Sep. 2011.

- [89] Y. Chae *et al.*, "A 2.1 M Pixels, 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.
- [90] D. Kim, J. Cho, S. Lim, D. Lee, and G. Han, "A 5000S/s Single-Chip Smart Eye-Tracking Sensor," in Proc. 2008 IEEE International Solid-State Circuits Conference, 2008, pp. 46–47.
- [91] M. Sakakibara *et al.*, "A Back-Illuminated Global-Shutter CMOS Image Sensor with Pixel-Parallel 14b Subthreshold ADC," in *Proc. 2018 IEEE International Solid-State Circuits Conference*, 2018, pp. 80-81.

Appendices

Here lists the supplementary material and background information. All appendices are properly referred in the body of this thesis.

Appendix A : CMOS Pixel Basics

This section provides the background information on the CMOS pixel design. Start from the photodiode, all typical pixel structures in present CIS designs are reviewed.

A.1 Basic n+/p-sub Photodiode



Figure A.1 A n+/p-sub photodiode design: (a) Structural diagram. (b) Circuit diagram.

A photodiode is type of optic-semiconductor device used as the photodetector in most CIS designs. After decades of development, various designs of photodiode are proposed and applied in research and commercial products. Shown in Fig. A.1(a) is an exemplary structure of a p-n photodiode, which is the most basic photodiode type created in CMOS process. In photodiodes fabrication, an n+ layer is implanted in a silicon wafer on top of the p-type substrate. For photo-detection, the photodiode is reversed-biased with its anode and cathode are connected to ground and other circuit 102 components, respectively. When light (photons) strikes on the n+ surface, the reverse biased photodiode forms a depletion region (an electric field) around the *pn*-junction and produces electron-hole pairs. Once the electron and hole are separated and floating out of the depletion region, all electrons are collected in the cathode terminal while holes are drained by the ground. The symbol of photodiode is shown in Fig. A.1(b), where it is reverse biased in a circuit. With a series connection to a power source (batteries), during illumination of light, a photocurrent (I_{ph}) is formed and flowing in reverse direction of the photodiode. As a typical diode, the diode forward current of the photodiode is expressed as [73]:

$$I_{\text{forward}} = I_{\text{diffusion}} \left[\exp\left(\frac{qV}{nkT_{\text{A}}}\right) - 1 \right]$$
(A.1)

with the diffusion current I_{diffusion} defined as:

$$I_{\text{diffusion}} = q A \left(\frac{D_n}{L_n} n_{p0} + \frac{D_p}{L_p} p_{n0} \right)$$
(A.2)

where *n* is a fit factor, A is the photodiode cross-section area, k is Boltzmann constant, T_A is the absolute temperature, $D_{n,p}$, $L_{n,p}$, n_{p0} and p_{n0} denote the diffusion coefficient, diffusion length, minority carrier concentration in the p-type region, and the minority carrier concentration in the *n*-type region, respectively. As photons reach the photodiode n+ surface, electron-hole pairs are generated in the *pn*-junction region if the photon energy is greater than the bandgap of the material $(hv > E_g)$ [74]. Therefore, the current flows through a photodiode can be expressed as:

$$I_{ph} = R_{ph}LA + qA\left(\frac{D_n}{L_n}n_{p0} + \frac{D_p}{L_p}p_{n0}\right) \left[1 - \exp\left(\frac{qV}{nkT_A}\right)\right]$$
(A.3)

where R_{ph} is the photodiode sensitivity, and L indicates the illumination at the photodiode surface. In the absence of light, photodiodes produce dark current (I_{dark}) based on their working temperature and surface leakage:

$$I_{dark} = qA\left(\frac{D_n n_{p0}}{L_n} + \frac{D_p p_{n0}}{L_p}\right) \exp\left(-\frac{E_g}{kT_A}\right) + \frac{1}{2}qn_i S_r A_s$$
(A.4)

where n_i , S_r , and A_s are the intrinsic carrier concentration, the surface recombination rate, and the surface area, respectively. From above expressions, it can be seen that the performance of a photodiode highly depends on the formation of *pn*-junction.

A.2 Operation of a pixel



Figure A.2 (a) A basic CMOS pixel cell. (b) Time diagram of pixel operation.

The operation of a pixel is a conversion process transferring light waves into electric signals. The conversion starts with photon-sensing and photocurrent generation, and ends with charge integration and voltage level variations in CMOS pixels. Fig. A-2 shows a basic CMOS pixel structure using a photodiode as the photodetector. The photodiode is reverse biased to a power source V_{DD} through a reset switch (Fig. A.2(a)). During an operation, a reset signal (φ_{rst}) toggles in the reset period T_{rst} to set the initial condition of the photodiode voltage (Fig. A.2(b)). Due to the switching of the reset transistor, a kT/C noise is added in. This noise is usually called as the 104

reset noise of CMOS pixels [75]. During the pixel exposure period (T_{expo}), charges are integrated in the photodiode self-capacitor C_{PD} , pushing down the voltage level at the pixel output terminal V_{out} until the next reset period. The pixel output signal is the voltage difference between the initial voltage level after reset operation and the final voltage level at the end of the T_{expo} period (indicated as "Signal" in Fig. A.2(b)). The magnitude of the voltage difference directly reflects to the light intensity.

The measurement of the pixel output signal can be accomplished by various pixel readout circuits, which directly define the type of pixel. Two most common types are passive pixel sensor (PPS) and APS. In the next two sections, the working principle of each type is introduced with discussion its advantages and disadvantages.



A.3 Passive pixel device

Figure A.3 A typical setup of a PPS pixel design.

Following the basic pixel structure illustrated in the previous section, Fig. A.3 shows a typical design of PPS pixels with a corresponding column processing slice. A readout switch is inserted to enable pixel outputs. When φ_{read} is pulled up, the voltage level reflecting amount of charge in the photodiode self-capacitor is read out. All pixel outputs are weighted by a charge transfer amplifier based PGA, which is shared by many pixels in a column. With a shared column line, only one of pixel outputs reaches to the PGA at a time while φ_{read} in other pixels remain low. At the beginning of signal amplification, the amplifier is set by its virtual ground voltage (V_{ref}). When a pixel output streams in, the voltage level on the feedback capacitor C_{PGA} varies and the amplifier output voltage is calculated as:

$$V_{PGA} = \frac{\Delta Q}{C_{PGA}} = \frac{C_{PD}}{C_{PGA}} \Delta V_A$$
(A.5)

where ΔQ and V_A are the volume of photodiode generated charges and the voltage across the photodiode, respectively. The biggest of advantage of PPS pixel design is only two transistors are required, offering high pixel fill factor and a compact pixel layout. When the number of pixels in a column increased, however, the column parasitic capacitance (C_{col}) also climbs up. To drive a big C_{col} in the shared column line, PPS pixels suffer from large readout noise, and slow readout speed [76].

A.4 Active pixel device

To resolve abovementioned issues in the PPS pixel design, an improved pixel structure called APS was proposed. Shown in Fig. A.4 is a typical design of a 3-T APS pixel. Comparing to the PPS design, the biggest difference is the readout circuit is replaced by a source follower. The source

follower consists of two transistors and works as an amplifier. The current source of the amplifier (I_{bias}) is implemented on the shared column line to save pixel space and prevent static power consumption when pixels are disabled. The readout switch is closed by pulling up φ_{read} only when the pixel is selected to output its signal. The pixel output signal is enlarged by the source follower and becomes capable of driving a large column capacitance (C_{col}, not show in Fig. A.4).

One of the biggest issues of the 3-T APS design is the reset level randomly varies in different pixels due to the mismatch of threshold voltage of the reset transistor. Such mismatch induces FPN to the CIS output image. Therefore, a technique called double sampling is usually performed to measure the reset noise. The double sampling operation begins with the reset of photodiode. After reset, the voltage across photodiode (V_A) is $V_{DD} + V_{rst}$ where V_{rst} is the additional reset noise. After charge integration on C_{PD} , the voltage level at the end of T_{expo} is then sampled by the source follower. The pixel output signal is expressed as $V_{DD} + V_{rst} - \Delta V_A$. Once the pixel readout is done, the pixel is reset again and sampled by the readout circuit. The newly sampled voltage level is V_{DD} + V_{rst2} . To be noticed that the newly sampled reset noise, V_{rst2} , is different then the previous sampled reset noise V_{rst} . By subtracting the newly sampled reset voltage level from the pixel output signal, a de-noised pixel output can be calculated as:

$$V_{out_ds} = V_{DD} + V_{rst2} - (V_{DD} + V_{rst} - \Delta V_A) = \Delta V_A + \sqrt{V_{rst}^2 + V_{rst2}^2}$$
 (A.6)

From above calculation, it is clear to see that the reset noise is not removed from the pixel output signal due to the un-correlation of two sampled reset noise. Hence, the reset noise levels add up in the 3-T APS pixel design. To completely cancel the reset noise, the two reset noise need to be correlated to each other.



Figure A.4 An example of 3-T APS pixel design.



Figure A.5 (a) A CMOS pinned photodiode. (b) Circuit diagram of 4-T APS pixel design. (c) Charge transfer from the pinned-photodiode to FD.

The reset noise correlation can be realized by using CDS. To achieve CDS operation, however, the pixel is required to own separate voltage conversion and the photon-sensing regions. Therefore, a type of photodiode from CCD [78] called pinned-photodiode is hired in the 4-T APS CMOS pixel design [79]. As shown in Fig. A.5(a), unlike the n+/p-sub photodiode, both implanted p+ and n+ layers are buried in a deep n-well region. The cathode of the pinned-photodiode is connected to the p+ layer while the n+ layer is the anode port. One big advantage of this type of structure is extremely low dark current due to the integration region (n type layer) is shifted away from the wafer surface. With the reduced dark current, CIS provides better performance especially in a low-light environment where the shot noise and dark current become dominant noise sources.

Besides the pinned structure, the photodiode is also connected to an n+ region called a floating diffusion (FD) through a transfer gate (Fig. A.5(b)). This transfer gate is tuned by a control signal φ_{tx} and designated to separate the photo-sensing/charge integration and readout circuits. Charges generated by the photodiode are accumulated on C_{PD}, and then transferred out to FD by closing the transfer gate. Due to the potential difference between the pinned voltage of the photodiode and the reset voltage of FD, all charges are transferred to FD from C_{PD} of the photodiode.

The readout and reset circuits are the same as that of 3-T APS pixel design. However, the separation of FD from C_{PD} enables CDS. The operation of CDS starts from the moment before the charge transfer occurrence (Fig. A.5(c)). FD is reset by pulling up φ_{rst} while the charge transfer gate kept open. The reset voltage of FD (V_{FD}) is then read out by the source follower and expressed as $V_{DD} + V_{rst}$. After the FD reset voltage readout is accomplished, signal φ_{tx} is asserted high to initiate the charge transfer process. When the charge transfer is done, the final voltage level in FD is then read out and calculated as $V_{DD} + V_{rst} - \Delta V_{FD}$. To be noticed that the reset noise in the final

readout is the same reset noise in the reset voltage of FD. Therefore, in the 4-T APS pixel design, the final pixel output signal is reset noise free after performed the noise cancellation:

$$V_{out \ cds} = V_{DD} + V_{rst} - (V_{DD} + V_{rst} - \Delta V_{FD}) = \Delta V_{FD}$$
(A.7)

Except for the low noise benefit, there are two main issues with the 4-T APS pixel design. Due to the charge transfer from the photodiode to FD requires a specific time, the charge-transfer switch must remain closed to prevent image lags caused by the incomplete charge transfer. Also, the fabrication of pinned photodiodes requires deep n-well implantation, which is available only in CIS-specific CMOS process.

A.5 **Pixel performance evaluation**

• Pixel fill factor

In CMOS pixel designs, as depicted in previous sections, a pixel is composed of two major components: a photodiode and affiliated circuits such as reset transistor, transfer transistor, and readout circuitry. Normally, in a standard CMOS fabrication process where BSI technology is not available, all affiliated circuits are covered by metal shield layers to prevent optical exposure. Only the area where a photodiode is placed is left open for exposure. Therefore, the definition of a pixel fill factor is the ratio of open region in the photodiode to the total size of the pixel pitch. With a larger pixel fill factor, pixels can receive more light and become more sensitive. In state-of-the-art CIS deigns, various BSI technologies are applied to increase the pixel fill factor up to 100%.

• Dark current

When place a CIS in a completely dark environment, the photodiode in a pixel still generates charges, which forms a leakage current called a dark current. The dark current varies in every

pixel. Therefore, the dark current will cause FPN and be mixed with the shot noise to deteriorate the overall SNR especially when the CIS is in a low-light environment. To suppress the level of dark current, two methods are commonly used in CIS designs: system temperature control and the use of pinned photodiode. As the photodiode leakage current is temperature related, the level of dark current can be constrained by lowing the temperature of a camera system. The use of pinned photodiode is another useful method to suppress the dark current, as explain in the previous section.

• Full well capacity

The full well capacity defines the maximum number of charges a pixel can hold. When a pixel is saturated, excessive charges will not be accepted. For a typical APS pixel, the full well capacity is limited by the size of FD in 4-T APS pixels, and defined by the self-capacitance of the photodiode in 3-T APS pixels. The total amount of charges held in a pixel can be calculated as:

$$Q_{\text{full-well}} = \frac{\Delta V \times C_{\text{PD/FD}}}{q}$$
(A.8)

where q is the charge of an electron (1.6e-19 C), and ΔV is the voltage swing of a pixel. With large full-well capacity, a pixel has higher dynamic range and can handle image captures in high illumination. As the pixel pitch becomes smaller, high full-well capacity can be achieved by techniques like a dual capturing method [80].

• Sensitivity

Pixel sensitivity is defined as the ratio between the pixel output voltage and the illumination level with a unit of V/lx·s. The sensitivity of a pixel is determined by many factors such as the quantum efficiency, the gain of source follower, and the conversion gain. The definition of quantum

efficiency is the ratio of collected charges to the incident photons. The source follower used in APS designs provides a gain of less than unity. The conversion gain can be increased by reducing the size of FD, however, in sacrifice of the full-well capacity.

• Dynamic range

The dynamic range of a pixel is defined as the maximum achievable signal divided by the noise floor:

Dynamic range =
$$20 \times \log \frac{Q_{\text{full-well}}}{N_{\text{noise}}}$$
 (A.9)

where N_{noise} is the sum of pixel read noise and dark noise expressed as the number of electrons. The dynamic range of a pixel is critical in image capture under low-light as well as in strong illumination. Wide dynamic range imaging capabilities are demanded in industrial and surveillance applications [81]. In a low-light environment, the pixel dark current and the noise of pixel output amplifier are dominant. Therefore, a deign of low-noise readout circuit is important [82]-[83]. When a pixel receives high illumination, the dynamic range can be extended by techniques like duo-capture [84], per-pixel integration time control [85], and non-linear photoresponse [86].

Appendix B : CMOS Image Sensor Architecture

This section provides the background information on some typical architectures utilized in CMOS image sensors. Depends on the implementation of ADCs, the image signal processing pipeline is classified into three categories: Sensor-level signal processing, column-parallel signal processing, and pixel-wise signal processing.

B.1 Sensor-level signal processing

In the CIS architecture using sensor-level signal processing, all pixel outputs are scanned and streamed out to a signal processing block. As illustrated in Fig. B.1, during pixel readout periods, the pixel array is scanned row by row by a row scanner. For every selected row of pixels, outputs are processed by column-parallel CDS and PGA blocks for noise cancellation and signal amplification. After the CDS operation accomplished, signals are scanned by a column scanner and sent to a on-chip/off-chip signal processing unit for further processing.



Figure B.1 A CMOS image sensor architecture with sensor-level signal processing. The signal processing module is implemented either on-chip or off-chip.

The signal processing unit contains an ADC module, which converts the incoming analog signal into its corresponding digital form. Such A/D conversion repeats until all outputs from the pixel array are scanned and processed. With this architecture, the overall speed of CMOS image sensor is determined by the speed of ADC. Therefore, a high-speed ADC is required to achieve high frame rate. As high frame rate and high precision, however, is still a design dilemma to the state-of-the-art ADC. The sensor-level signal processing based CIS architecture becomes less popular in modern digital camera products.



Figure B.2 A CMOS image sensor architecture with column-parallel signal processing.

B.2 Column-parallel signal processing

In a column-parallel signal processing based CIS architecture, the signal processing unit is placed after the column-wise CDS and PGA block. As shown in Fig. B.2, the signal processing unit consists of column-wise ADCs which offer parallel A/D conversion for output signals processed from the selected row of pixels. Comparing to the sensor-level CIS architecture, such column-parallel signal processing achieves higher data throughput, which assists high-resolution CIS to operate in higher frame rates.

With the on-going development in stacked CIS technology, this type of architecture becomes a mainstream in CIS design. Different types of ADC implementation such as single-slope [87], cyclic [88], and delta-sigma [89] are reported in various CIS designs.



Figure B.3 A CMOS image sensor architecture with pixel-level signal processing.

B.3 Pixel-level signal processing

The pixel-level signal processing offers premium image signal throughput, which assists CIS to realize ultra-high-speed operation (e.g. > 1000 fps) [90]. As depicted in Fig. B.3, every pixel in

the pixel array equips a signal processing block consists of CDS and ADC circuits. Such in-pixel design yields a large pixel size and sacrifices pixel fill factor. Even in a latest stacked CIS design [91], this architecture can be implemented in a relatively small resolution due to the layout and time closure complicity. Also, with more circuit components included in each pixel, pixel performance consistency becomes more challenging. Despite of those drawbacks, the pixel-level signal processing provides extremely high-speed and low-power signal processing which cannot be reached by any other CIS architecture introduced in previous sections.