

On-Chip Spatial Image Processing with CMOS Active Pixel Sensors

By
Canaan Sungkuk Hong

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2001

© Canaan S. Hong 2001



**National Library
of Canada**

**Acquisitions and
Bibliographic Services**

**395 Wellington Street
Ottawa ON K1A 0N4
Canada**

**Bibliothèque nationale
du Canada**

**Acquisitions et
services bibliographiques**

**395, rue Wellington
Ottawa ON K1A 0N4
Canada**

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-65248-3

Canada

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

ABSTRACT

Output images from the sensors more likely are not optimal results for display or further processing mainly because of noise, blurriness and poor contrast. In order to prevent these problems, image processors typically accompany the image sensors as a part of the whole camera system. Typically, two separated chips for sensing and processing are integrated onto the same printed circuit board connected by printed wires. The integration of image sensors and processing circuits on a single monolithic chip, called smart sensing, is done to obtain better performance from sensors and make the sensing and processing system more compact. It has become a popular idea. The integration of image acquisition and processing on the same focal plane has potential advantages through low fabrication cost, low power, compact size, and fast processing frequency. Noise and cross-talk can also be reduced through monolithic connections instead of off-chip wires, which are the only transfer medium between two separated chips.

In this thesis, we propose system-level architectures and design methodology for integrating image processing with CMOS active pixel sensors on a single chip. Conventional approaches to the integration categorized by circuit density of processing elements are not sufficient to achieve optimal design with power, speed, cost, and processing frequency. This thesis observes the nature of image processing algorithms and categorizes them in order to find out adequate design architecture for real time smart sensing. The algorithms can be divided in terms of signal type, operational domain, and regions of operation. We narrow these down into analog/low bit digital operation in spatial domain, and then subdivide the algorithms into point, local, and global operational regions. For each region of operation, we look at examples of processing algorithms and then subdivide them again according to on-chip implementation methodology. Here, we propose system-level architecture and on-chip design methodology for these categorized algorithms.

Four prototype chips, in this thesis, were designed and fabricated for the demonstration of smart sensing: One is a multi-camera system which is the inspiration for the smart sensing research, and the other three are demonstration imagers for each region of operation: point, local and global. These prototype chips are 64x64 photodiode arrays with on-chip image processing fabricated in standard 0.35 μm CMOS technology with 3.3V power supply. Each

chip contains different functional processing and operates at different performances. We have successfully tested the chips with different testing performances and characteristics.

This thesis reports implementation architectures and design methodologies of on-chip processing with image sensors, its analysis along with operational performance and experimental results. These implementations demonstrate the advantages of the single chip solution and contribute as a milestone so designers and researchers can have a better understanding of smart sensing.

ACKNOWLEDGEMENTS

I would like to acknowledge my supervisor, Prof. Richard Hornsey, for his constant support, encouragement, motivation, research direction, and belief in me throughout the duration of this research.

I am also deeply thankful to Dr. Paul Thomas at Topaz Technology Inc., for his support and encouragement in the early research projects.

I would also like to thank all my colleagues at the University of Waterloo for their valuable discussions, suggestions, and supports in helping me become familiar with many of the hardships associated with being a graduate student.

I am grateful for the research support and funding from Natural Science and Engineering Council of Canada and the Center for Research in Earth and Space Technology.

I would also like to thank the Canadian Microelectronics Corporation for permission to access their processing technology and fabrication of chips.

Finally, my profound gratitude goes to my parents, my wife Seungeun, and my son Joseph, for their patience, understanding, unconditional love and support.

TABLE OF CONTENTS

Chapter I.....	1
1. Introduction.....	1
Chapter II.....	5
2. Basic Operation and Structure of CMOS Image Sensors.....	5
2.1. Solid-State Image Sensors.....	5
2.2. History of Image Sensors at Visible Spectrum.....	6
2.3. CCD and CIS for Smart Sensors.....	8
2.4. Fundamentals of CMOS Image Sensors.....	12
2.4.1. Optical Absorption and Photo-Generation.....	12
2.4.2. Photon Collection (Quantum Efficiency).....	13
2.4.3. CMOS Photodetectors.....	16
2.4.4. Active Buffer in Pixel.....	19
2.4.5. Operation of Active Pixel Sensor with Photodiode.....	21
2.4.6. Readout Control.....	25
2.4.6. Sample and Hold (S/H).....	27
2.4.7. Basic Structure of CIS APS array.....	28
2.5. Future Research Focuses of CMOS Image Sensor.....	31
Chapter III.....	33
3. MOSAIC Multi-Camera Imager System with CMOS Image Sensors.....	33
3.1. Introduction.....	33
3.2. Single Chip verse Multi-chip Systems.....	34
3.3. Previous MOSAIC Implementations.....	35
3.4. Design of MOSAIC.....	37
3.4.1. Integrated Bus Interface with CMOS Image Sensor.....	37
3.4.2. Circuits and Layouts.....	39
3.4.3. Demonstration and Tests.....	42
3.5. Conclusions for MOSAIC: Single Chip Camera Modules.....	53
Chapter IV.....	54
4. Spatial Image Processing Integrated with CMOS Image Sensor.....	54
4.1. Introduction.....	54

4.2. Smart Sensors (Vision Chips): Why Smart Sensors?.....	56
4.3. On-chip Early Image Processing: What on Smart Sensors?.....	58
4.4. Architectures for On-chip Processing Integration: How to Implement Smart Sensors?	62
4.4.1. Previous Work.....	63
4.4.2. Types of Hardware Implementation.....	64
4.4.3. Design Issues of Hardware Implementation.....	69
4.4.4. Types of Image processing Algorithms.....	75
Chapter V.....	80
5. Point Operation.....	80
5.1. Introduction.....	80
5.2. Comparisons between On-chip Implementations for Point Operation.....	85
5.3. Design of In-pixel Contrast Stretching.....	88
5.3.1. Introduction.....	88
5.3.2. Intensity Transformation Function.....	90
5.3.3. Previous Work on Pixel Level Processing.....	96
5.3.4. Designs of CMOS Active Pixel Sensor with In-pixel Intensity Transformer.....	96
5.3.5. Tests and Performances.....	103
5.3.6. Summary and Conclusions.....	118
Chapter VI.....	120
6. Local Operation.....	120
6.1. Introduction.....	120
6.1.1. Smoothing Filters.....	121
6.1.2. Sharpening Filters.....	122
6.1.3. Derivative Filters (edge detection).....	124
6.2. Proposed Structure for Local Operation.....	124
6.2.1. Implementations of 3x3 Local Mask Filters.....	127
6.2.2. Implementation of Bigger Masks than 3x3.....	133
6.3. Spatial On-Chip Binary Image Processing.....	135
6.3.1. Fundamental Operation in Binary Image Processing.....	135
6.3.2. Previous Works on Binary Image Processing.....	140

6.3.3. Design of CMOS Active Pixel Sensor with On-Chip Binary Image Processing	141
6.3.4. Tests and Performance.....	150
6.3.5. Summary and Conclusions	164
Chapter VII	167
7. Global Operation.....	167
7.1. Introduction	167
7.2. Structure of Global Processing.....	171
7.3. 2-D Object Positioning System (OPS).....	174
7.3.1. Chip Design	176
7.3.2. Demonstration and Tests	180
7.3.3. Summary and Conclusions	182
Chapter VIII.....	184
8. Summary and Conclusions	184
Appendix A: Inverted Logarithmic Pixel Sensors with Current Readout	189
A.1. Introduction	189
A.2. Inverted Logarithmic Pixel Sensors	189
A.3. Testing and Measurements.....	193
A.4. Conclusions	203
Appendix B: Basic Procedures for Image Capture Test.....	204
Appendix C: Image Sensor Characteristics	205
C.1. Basic Measurements	205
C.2. Imager Characteristics Extraction and Calculation	206
C.3. Image Sensor Characteristics	210
References.....	213

LIST OF FIGURES

Figure 2.1. Solid-state image sensors over wide spectral range	5
Figure 2.2. History of MOS, CCD and CMOS image sensors	6
Figure 2.3. Absorption coefficient and penetration depth of silicon at different wavelengths of incidental light	12
Figure 2.4. Photo-generation and collection of photon-generated electron-hole pairs	14
Figure 2.5. Drift and minority diffusion in collection of photogenerated charge	15
Figure 2.6. CMOS photodetectors	17
Figure 2.7. Active pixel sensor with photodiode and active buffer	19
Figure 2.8. Active buffers in CMOS APS with photodiode	20
Figure 2.9. Cross sectional view of the photodiode	22
Figure 2.10. Schematic view of photodiode pixel sensor with parasitic capacitance	24
Figure 2.11. Configuration of a source follower as a gate buffer and current source	25
Figure 2.12. Shift register, using flip-flops	26
Figure 2.13. Shift register with two inverters in each processing element	26
Figure 2.14. Typical sample and hold circuit	27
Figure 2.15. Sample and hold with PMOS source follower	27
Figure 2.16. An advanced sample and hold	28
Figure 2.17. Schematic of CMOS APS	29
Figure 2.18. A simplified timing control	30
Figure 2.19. Overall structure of CMOS APS array	30
Figure 3.1. MOSAIC multi-camera system with a central controller	33
Figure 3.2 Single chip and multi-chip system for MOSAIC system	35
Figure 3.3. Systematic connection of MOSAIC imager	38
Figure 3.4. Chip photo of MOSAIC chip	39
Figure 3.5. Array structure of ideal MOSAIC image sensor with an integrated bus interface	40
Figure 3.6. Active Pixel Sensor with photodiode and active buffer in integration mode	41
Figure 3.7. Schematic of S/H	41
Figure 3.8. Shift register is implemented for readout circuitry, using two inverters and switches	42

Figure 3.9. Readout circuitry is integrated with switches enabled by Bus Grant signal	42
Figure 3.10. Test board with MOSAIC chip and lens mounted	43
Figure 3.11. Characteristics of single image sensor	44
Figure 3.12. Photosensitivity of single chip of MOSAIC	45
Figure 3.13. Dark current measurement in single chip of MOSAIC	45
Figure 3.14. Images with different Vbiasn	47
Figure 3.15. Images with different S/H Vbiasp	48
Figure 3.16. Images with different sampling rate	49
Figure 3.17. Testing setups for three MOSAIC chips' connection	50
Figure 3.18. Panorama images captured by the MOSAIC system	51
Figure 3.19. Test results of MOSAIC imager	52
Figure 4.1. Optical image system in human	59
Figure 4.2. General machine vision/image processing operational stages of image analysis	60
Figure 4.3. Structures of focal plane implementations with image sensors	67
Figure 4.4. Number of transistors per pixel as a function of process technology	69
Figure 4.5. Fill factor for different number of transistors in a pixel	70
Figure 4.6. Maximum processing time available for the processing element for different sizes of array	71
Figure 4.7. Total power consumption only for image sensor array	73
Figure 4.8. Total power consumption (not including image acquisition) of the different array size for different processing levels	73
Figure 4.9. Image operation divided by regions of operation: point operation, local operation and global operation	79
Figure 5.1. Image processing of image negative	81
Figure 5.2. Contrast stretching technique	82
Figure 5.3. Image compression	82
Figure 5.4. Gray level slicing	83
Figure 5.5. Gray-level intensity transformation function for contrast enhancement	89
Figure 5.6. Original image of Matlab simulations on intensity transformer with its histogram and intensity transformation function	91

Figure 5.7. Matlab simulations on intensity transformer (mapping function) with contrast stretching technique	92
Figure 5.8. Matlab simulations on intensity transformer (mapping function) with brightness adjustment technique	93
Figure 5.9. Matlab simulations on intensity transformer (mapping function) with gamma correction technique	95
Figure 5.10. Die photograph of the prototype chip. The total area is 16 mm²	97
Figure 5.11. Schematics of common source follower consisting of a transformer with enhanced mode NMOS active load	98
Figure 5.12. Voltage response of a common source amplifier with enhanced mode NMOS active load	99
Figure 5.13. Response of a common source amplifier with voltage output of photodiode as its input	99
Figure 5.14. Schematic of intensity transformer	101
Figure 5.15. HSPICE simulations on an intensity transformer	102
Figure 5.16. Overall structure of the chip	104
Figure 5.17. Schematics of main components of intensity transformer chip	105
Figure 5.18. Photoresponse of in-pixel intensity transformer	107
Figure 5.19. Sample images captured in real time by the chip in normal mode	109
Figure 5.20. Pattern noise can be reduced by subtracting white background image form the raw image	109
Figure 5.21. Characteristics of single chip	111
Figure 5.22. Sample images and histograms of three output modes	112
Figure 5.23. Original images captured in normal mode with different illuminations	113
Figure 5.24. Effects of biasing voltage (V_{biasp})	115
Figure 5.25. Effects of reference voltage (V_{ref})	116
Figure 5.26. Mismatches in three output modes	117
Figure 6.1. Matlab simulations on smoothing filters	122
Figure 6.2. Matlab simulations on sharpening filters	123
Figure 6.3. Matlab simulations on edge detection filters	125
Figure 6.4. Local masks with different sizes	126

Figure 6.5. Local masks with different connectivity	126
Figure 6.6. Pixel processing for 3x3 local mask operation	128
Figure 6.7. Column processing for 3x3 local mask operation	129
Figure 6.8. Chip processing for 3x3 local mask operation	130
Figure 6.9. Hybrid method (column + chip processing) for 3x3 local mask operation	131
Figure 6.10. Frame memory processing for 3x3 local mask operation	132
Figure 6.11. Concept of pipelined local masking	134
Figure 6.12. Basic structure of pipelined implementation for large local masks	134
Figure 6.13. Binary Image Processing with various functionalities	139
Figure 6.14. Die photograph of the prototype chip. The total area is 3.2x3.2 mm²	142
Figure 6.15. Overall structure of Binary Image Processing	144
Figure 6.16. Schematic of major components in on-chip binary image processing	145
Figure 6.17. Detailed structure of On-chip Binary Image Processor	145
Figure 6.18. Schematic of Voltage Comparator	146
Figure 6.19. Logic design and schematics of the switches	149
Figure 6.20. Real time images captured by the chip in normal mode operation	151
Figure 6.21. Effects of frame rate in normal mode operation	153
Figure 6.22. Defect of white spot in normal mode	154
Figure 6.23. Removing the defects	154
Figure 6.24. Sample images of on-chip binary image processing	156
Figure 6.25. Demonstrations of binary image processing	157
Figure 6.26. Binary image processing from the shape of the objects	158
Figure 6.27. The effects of reference voltage	161
Figure 6.28. Connectivity	163
Figure 7.1. Transfer function of different types of low pass filters	168
Figure 7.2. Transfer function of ideal high pass filter	169
Figure 7.3. Transfer functions of high frequency emphasis filters	170
Figure 7.4. Structure of 2-D Object Positioning System and its basic operation	175
Figure 7.5. Structure of global OR gate	175
Figure 7.6. Overall structure of CIS array with object positioning systems	176
Figure 7.7. Die photo of Object Positioning Chip	177

Figure 7.8. Schematic of a pixel for 2-D Object Positioning System	178
Figure 7.9. Schematics of a pixel and event detection latch	179
Figure 7.10. Sample images of the 2D object positioning chip	181
Figure 7.11. When multiple balls exist in the input image	181
Figure 7.12. Test results of 2-D OPS imager	182
Figure A.1. Structures of logarithmic pixel sensors	190
Figure A.2. Simulated effect of lithographic deviation on a regular logarithmic pixel sensor	191
Figure A.3. Simulated effect of lithographic deviation on an inverted logarithmic pixel sensor	191
Figure A.4. Schematic view of the sensor structure	192
Figure A.5. Structures of photodiode used for the inverted logarithmic pixel sensors	193
Figure A.6. Variation of the photoresponse of the inverted logarithmic pixel with number of load transistors	194
Figure A.7. Sample image captured by inverted logarithmic pixel sensors	195
Figure A.8. Photograph of the image sensor die. Total die area is 16 mm²	196
Figure A.9. Variation of rms pattern noise with illumination	197
Figure A.10. Effect of image sensor V_{DD} on image quality	198
Figure A.11. Effect of transresistance amplifier reference voltage on image quality	200
Figure A.12. Effect of data sampling rate on image quality	202

LIST OF TABLES

Table 1. Major differences in process between CCDs and CISs	9
Table 2. Present status of CMOS image sensors	31
Table 3. General descriptions and comparisons on hardware implementation structures, with their advantages and disadvantages	68
Table 4. Numerical comparisons of hardware implementation structures for an MxN array with S frames/second	71
Table 5. General descriptions and comparisons of point operation implementations, for different types of the point operation	88
Table 6. Single chip characteristics in normal mode and contrast mode	110
Table 7. General descriptions and comparisons of local operation implementations, for different size of local masks	136
Table 8. Characteristics of single chip	151
Table 9. General descriptions and comparisons of global operation, for different operation domain	173
Table 10. Characteristics of chip tests	180
Table 11. Summary of on-chip implementation methodology for image processing algorithms	186
Table A.1. Electrical and Optical characteristics of the Inverted Logarithmic Sensor chip	197

Chapter I

1. Introduction

There are many kinds of electronic camera available on today's market with various applications such as document and film scanning, video imaging, still-image capture, machine vision, infrared and x-ray imaging, astronomy and microscopy. Despite the wide variety of applications, all digital cameras have the same basic functional components, which consist of optical collection of photons (e.g. a lens), wavelength discrimination of photons (e.g. filters), detector (e.g. solid state sensors), timing, control and drive electronics for the sensors, signal processing electronics for correlated double sampling, colour processing, analog-to-digital conversion and interface electronics [1].

A core component of an electronic camera is the solid-state image sensor that converts light into electrical form and, further may process and convert it into an appropriate signal (e.g. digital signal). For many years, silicon based image sensors have been extensively investigated since silicon has a good light absorption characteristics over the visible spectrum and has a mature technology in its processes and VLSI circuits. Over the visible spectral range, there are two main silicon-based image sensor technologies, Charge Coupled Devices (CCDs) and CMOS Image Sensors (CISs). Although these technologies use the same silicon as substrate, they are quite distinct in their photo-characteristics and functional operation.

CCDs have been the dominant technology for electronic image sensors for several decades due to their low dark current, high photosensitivity, low fixed pattern noise, small pixel size and structure. However, in the last decade, CMOS image sensors have gained attention from

many researchers and industries due to their low power, low fabrication cost, compatibility with VLSI integration, and radiation hardness. Many researchers are attracted by its low power, low weight and radiation hardness for deep-space applications. Custom markets are interested in CISs for their low fabrication cost and the compatibility of VLSI circuits with image sensors.

This thesis focuses on the VLSI compatibility of CISs and more particularly, on integration of image processing algorithms on the same focal plane with CISs, so called smart sensors or vision chips. This thesis discusses why the integration of the smart sensors is advantageous and what should exist on the smart sensors, and how to integrate image processing algorithms with CISs (i.e. how to implement the smart sensors). The thesis includes recommendations on system-level architectures, applications and limitations of the implementation of smart sensors, which are categorized by the nature of image processing algorithms.

The main contributions and objectives of this thesis are summarized as follows: (i) to give milestones of designs for integration of image processors with CMOS image sensors, where designers and engineers can start their initial implementations, and to give a better understanding of the integration to give designers and researchers guidance to improve the implementation techniques for smart sensors, (ii) to determine the feasibility of the integration of image processors with image sensors in standard CMOS 0.35 μm technology, (iii) to demonstrate scalability of design with technology, (iv) to forecast possible design and implementation issues of the integration in advance, and lastly (v) to suggest future research directions, for smart sensor implementations.

In Chapter 2, a brief description and applications of solid-state image sensor is outlined and the history of developments in CCD and CIS is reported. The advantages and disadvantages in functional operation and processes of CCD and CIS are compared. Then, the basic operation of CIS is discussed along with their basic functional components and structural layout. The future expectations and applications of CIS are also included.

In Chapter 3, a concept of MOSAIC (Matrix of Semi-Autonomous Imaging Cameras) is proposed for large field of view. The definition and applications of the MOSAIC system are also discussed in this chapter. A simple MOSAIC chip with CIS array and bus interface was

designed and fabricated for a demonstration of the multi-camera concept. The detailed designs of the MOSAIC chip and its test results are explained. The conclusions and suggestions for the MOSAIC concept are also discussed at the end of the chapter.

Based on the conclusions of the MOSAIC chapter, the main focus of the rest of the research is on effective integration architectures for image processing algorithms with CIS. In Chapter 4, the background of image processing integration with CIS is outlined, including why, what and how to employ smart sensors with CIS. This includes previous implementations of processing integration with CIS and their relations to this thesis. It explains the sequence of image processing analysis and its relation with smart sensors by discussing the structural implementation of focal plane integration with CIS. For effective integration architectures, we categorize the types of image processing algorithms in terms of signals, domains and spatial regions of operation.

In Chapter 5, the advantages and disadvantages of integration architecture of image processing algorithms for point operations are investigated. The definition of a point operation is discussed along with examples of this operation. The merits and drawbacks of point operation in different implementation structures of pixel, column, chip and frame memory processing are compared. The optimal architectures for the integration are also proposed according to general characteristics of sensor applications. A CIS chip with in-pixel contrast stretching, also known as an intensity mapping function, was designed and fabricated as a demonstration of point operation at the pixel level. This chapter includes the detailed design and fabrication of the chip and its test results.

Chapter 6 investigates the architecture of image processing integration for local operation. The definitions of local operation are discussed along with advantages and disadvantages of this technique. Local operational image processing algorithms are divided into 3x3, and larger spatial mask implementations according to size of the local mask. Local operation in pixel, column, chip and frame memory processing are compared for implementing smart sensors, leading to the optimal system-level architectures according to the size of the local mask. A CIS chip with on-chip binary image processing was designed and fabricated as a demonstration of 3x3 local operation at the column level. The detailed designs of the local operation chip and its test results are also included.

In Chapter 7, the architecture of image processing integration for global operations is investigated in terms of operational domain, namely frequency and spatial domains. A definition of global operation is discussed with examples of such operations. In this chapter, global operation at pixel, chip and frame memory processing levels are compared listing their merits and drawbacks and, thereby, possible implementations are proposed according to the operational domain. A CIS chip with an object positioning system was designed and fabricated as a demonstration of global operation. The detailed designs of this chip and its test results are included, along with the discussion of its optimization.

Chapter 8 summarizes the work of the research and presents the conclusions derived from this research along with directions for further work.

Appendix A contains design and test results of a chip with inverted logarithmic pixel sensors. An inverted logarithmic pixel sensor is a modified pixel structure that has advantages of low pattern noise and continuous current readout over conventional logarithmic sensors. This chapter also discusses the potential advantages and disadvantages of current mode operations, and their applications. The detailed concept and design of the pixel sensor are discussed and sample images of the sensor array are demonstrated with their advantages and disadvantages.

Appendix B discusses the basic procedure for image acquisition in the image sensor test. It describes how to test the image acquisition of the image sensor chip for the first time.

Appendix C explains image sensor characteristics in the image sensor test. It discusses basic measurement methods, calculations of optical characteristics and makes comparisons with commercial sensors.

Chapter II

2. Basic Operation and Structure of CMOS Image Sensors

2.1. Solid-State Image Sensors

Solid-state image sensors are integrated circuits (usually silicon-based) that contain a number of photosensitive sensors in typically a 2-dimensional or 1-dimensional array for the purpose of converting an optical image projected onto the device to an electrical output (usually a voltage or current). Compared to conventional camera films, the solid-state image sensors are computer friendly where films need a scanner in order to input images to computers. In addition, solid-state image sensors can save time because they do not require developing time that film inevitably requires, which makes real time operation possible.

As seen in Figure 2.1, there are many kinds of solid-state image sensors (not only silicon-based devices) with very different characteristics over a wide spectral range. Devices

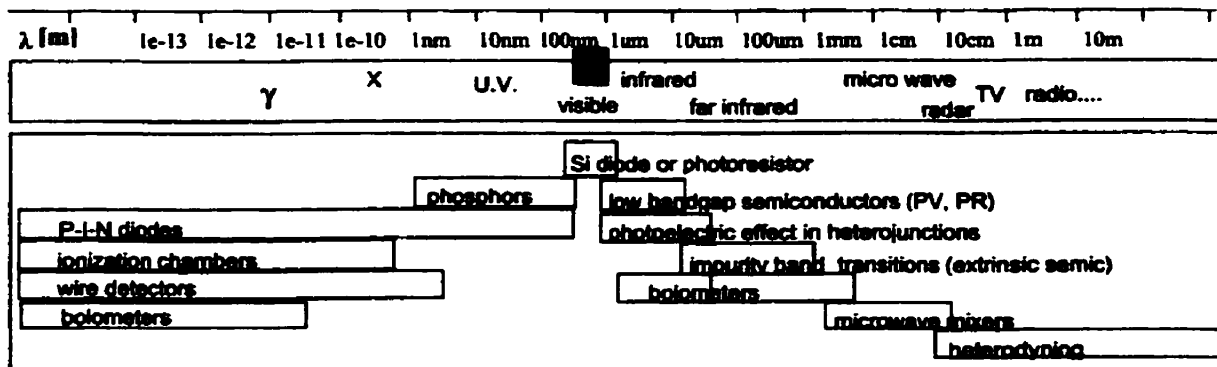


Figure 2.1. Solid-state image sensors over a wide spectral range [86].

may have sensitivity to wavelengths from the γ -ray spectrum to radio frequency spectrum. Yet, a great interest of commercial electronic image sensors resides in the visible spectral range simply because most of applications are for the visible spectrum. This thesis focuses on visible imaging.

2.2. History of Image Sensors at Visible Spectrum

For visible spectral range, charge-coupled devices (CCDs) and complementary metal oxide semiconductor (CMOS) active pixel sensors (APSs) are currently dominant technologies for image sensors. A brief history of the solid-state image sensors for CCDs and CISs (Figure 2.2) is well described by Fossum [1] and can be summarized as follows.

At the beginning stage of solid-state image sensor development, there was a form of MOS image sensors before CMOS APS and before CCD. In the 1960's there were numerous groups working on solid-state image sensors with varying degrees of success using NMOS, PMOS, and bipolar processes. In 1963, Morrison reported a structure of computational sensor that allowed determination of a light spot's position using the photoconductivity effect

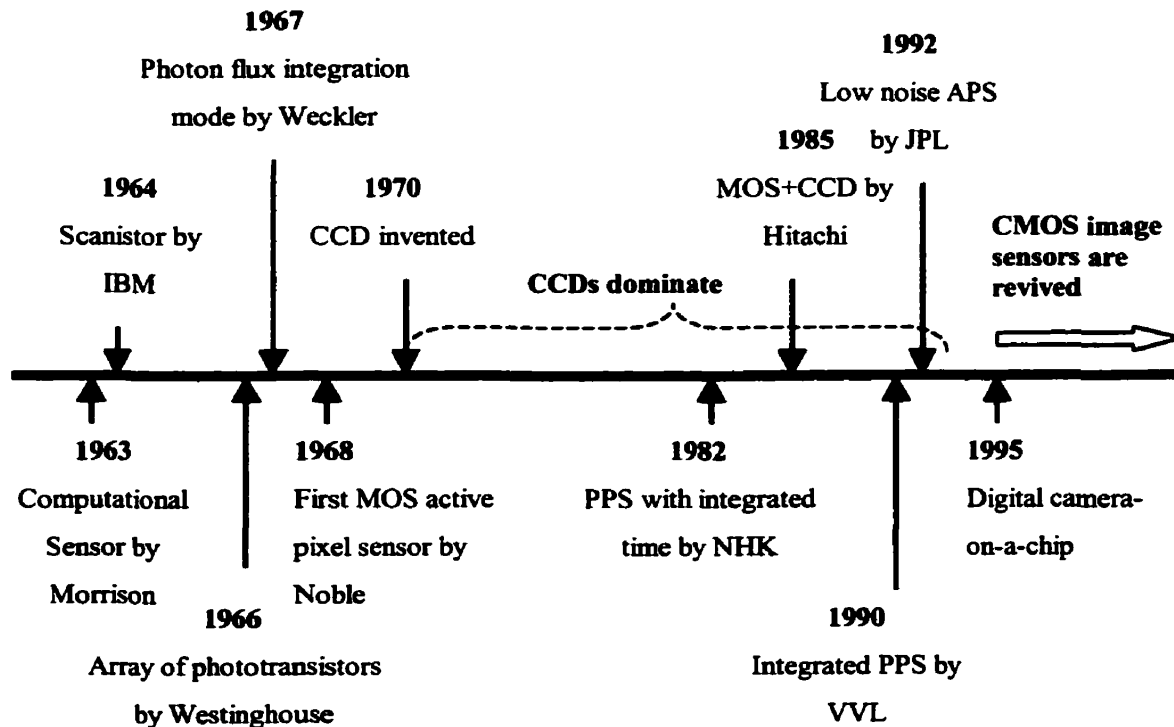


Figure 2.2. History of MOS, CCD and CMOS image sensors.

[2]. In 1964, IBM reported the scanistor that used an array of n-p-n junctions addressed through a resistive network to produce an output pulse proportional to the local incident light intensity [3]. In 1966, Westinghouse reported a 50x50 element monolithic array of phototransistors [4]. Since none of these sensors performed any intentional integration of the optical signal, their sensitivity was low and thereby, often they required some form of signal amplification. In 1967, Weckler from Fairchild suggested operating p-n junctions in a photon flux-integrating mode [5]. A 100x100 element array of photodiodes was reported in 1968 [6]. Weckler later called the device a reticon and formed Reticon to commercialize the sensor. In 1968, Noble reported the first MOS active pixel sensor [7]. Noble discussed a charge integration amplifier for readout, similar to that used later by others. Here, the first use of a MOS source-follower transistor in the pixel for readout buffering was reported.

In 1970, when the CCD was first reported [8], its relatively low Fixed Pattern Noise (FPN: pattern noise in dark room) was one of the major reasons for its adoption over the many other forms of solid-state image sensors. The smaller pixel size afforded by the simplicity of the CCD pixel also contributed to its embrace by industry and it continued until MOS image sensors were resurrected in the late 1980s. While a large effort was made for the development of the CCD in the 1970s and 1980s, MOS image sensors were only periodically investigated and compared unfavorably to CCDs with respect to the above performance criteria [9].

In the late 1970s and early 1980s Hitachi and Matsushita continued the development of MOS image sensors [10], [11] for camcorder-type applications where high-speed operation with relatively low resolutions were focused on. In 1982, NHK successfully integrated timing control with passive pixel sensors. Temporal noise in MOS sensors started to lag behind the noise achieved in CCDs. By 1985, Hitachi combined the MOS sensor with a CCD horizontal shift register [12]. However, perhaps due to residual temporal noise, especially important in low light conditions, Hitachi later abandoned its MOS approach to sensors.

In the early 1990's, the University of Edinburgh (later forming VLSI Vision Ltd.) created highly functional single-chip imaging systems where low cost was the main factor. In 1990, the VVL reported an integrated Passive Pixel Sensor (PPS) array [13]. However, due to large capacitive column bus loads, the use of PPS was limited to small to medium array sizes and

slow to medium readout speed. By comparison with CCDs, noise and mismatch effects limited the quality. However, low power operation and integration demonstrated viability of single chip cameras and integrated sensor-processors. Although, in 1968, Noble demonstrated the first MOS buffer amplifier in a pixel, relatively little active pixel sensor (APS) research was carried out for another 10 years, and it took 20 years for major interest to be renewed when NASA JPL group began research on low noise APS in 1992 [14]. CMOS based image sensors offer the potential to integrate a significant amount of VLSI electronics on-chip and reduce component and packaging cost. Around 1995, after a successful demonstration of low noise CMOS APS, CMOS image sensors took off due to their easy integration with VLSI circuits, low power consumption, low fabrication cost, and radiation hardness. Recently, commercial products using CMOS image sensors have become available and increasingly popular, including PC camera, cellular phone cameras, PDA, toys, etc.

2.3. CCD and CIS for Smart Sensors

Through the 1970s and 1980s, CCD technology was strong and it still survives in digital camera and camcorder markets, simply because it outperforms any other solid-state image sensors in the visible spectrum. The good image quality of CCD is based mainly on low noise and low dark current. CCD has low noise level, typically less than 50 noise electrons. FPN (Fixed Pattern Noise) of the CCD is less than 1% V_{pp} of its saturation level with a good PRNU (Photo-Response Non-Uniformity) of 1 ~ 10% V_{pp} . In addition, very low dark current, typically less than 10 pA/cm², is achieved by this technology. The CCD process itself is optimized for optical detection and therefore, the optical absorption and quantum efficiency outperforms CIS. Since CCDs can share the same area for optical detection and charge transfer, it does not require any special transistors to transfer photon-generated charges, resulting in a high fill factor. However, due to detection and transfer mechanism, CCD is limited to serial scanning with complicated driving and interfacing. CCD is also a specialized technology that is relatively expensive and therefore, many companies cannot afford their own fabrication laboratory. Besides, because CCD is not easily compatible with logic, so on-chip ADC and other on-chip processing circuitry seldom exist on a focal plane with image sensors, but rather exist in separate chips.

For the main theme of this thesis (integration of smart sensors), integration feasibility of technologies is of interest. Here, focusing on smart sensors, CCD and CIS are compared. With aspects of smart sensor implementations, the comparisons of CCD and CIS are very well described and summarized in “*Vision Chips*” [15] by Alireza Moini. The following comparisons are adapted from this reference, emphasizing CMOS compatibility for smart sensors. Although CCD has good image quality, CCD is rarely used for smart sensors, mainly due to VLSI incompatibility with logic and memory. Other major drawbacks of CCD with respect to CMOS are as the follows:

- **Input Control Clock:** A large number of clocks are required in order to trigger all pixels in imager array. At least two clock phases (or more) are required to read out all the pixels.
- **VLSI integration:** CCD is optimized for charge transfer (deep diffusions, thick gate oxide, etc) and it is therefore difficult to develop logic and memory with the technology. CCD is hard to integrate CMOS. The Table 1 shows major differences in their processes. From the table, it is quite obvious why these two technologies are rarely integrated together. Even if they were to be integrated, the integration cost would be very high.

Parameters	CCD	CMOS
Gate Oxide Thickness	800 Å	50 Å
Well depth	P-well depth > 2.5 μm	Well depth ~ 0.5 μm
Channel Stop Depth	~ 1 μm	
Channel Depth	~ 0.8 μm	
Source/Drain Implants		~ 0.1 μm
Operating Voltage	≥ 10 V	≤ 3.3 V
Poly	Several poly-Si and inter-poly dielectrics needed	Digital process has 1 poly, analog has 2 polys

Table 1. Major differences in process between CCDs and CISs.

- **Fabrication cost:** Since CCD technology requires a specialized process, its fabrication cost is very high, compared to very standardized CMOS technology.
- **Power consumption:** CCD typically requires high voltage supply to clock the large capacitive gates of CCD array. Therefore, CCD consumes a large power.

There have been attempts to integrate CCD and CMOS logic [21]. However, due to incompatibility of the two technologies, these attempts were not generally successful. Even if these two technologies are successfully integrated, they never achieve both CCD-like image quality and CMOS-like flexible logic. In fact, the optimization for one degrades the performance of the other. Besides, the integration of CCD and CMOS often requires over 30 masks, which is not cost effective.

In order to effectively implement processing components with the image sensors, designers need a technology beyond CCD, in order to increase functionalities of the smart sensor even if this means sacrificing image quality of the image sensors. Although CMOS technology has been and remains the dominant technology in almost all VLSI design areas, CMOS image sensors did not take off in imaging device fields until the mid 1990s. After a demonstration of the active pixel sensors of CMOS image sensors, they gained attention from researchers and industries because the CMOS technology offers the following advantages [15].

- **Mature technology:** CMOS processes have been available for long period of time. CMOS processes are well developed and well established. Many engineers and researchers have characterized and optimized the technology.
- **Design resources:** Many design libraries for circuit and logic are supported by various research groups and industries. A large number of circuits and layouts are already built in. Designers can save time and effort in simulation and custom layouts.
- **Accessibility:** There are many fabrication facilities around the world, which are willing to fabricate prototype designs at low prices. Engineers and researchers are now able to fabricate their designs without having their own fabrication.
- **Fabrication cost:** Because CMOS process is standardized, the fabrication of CMOS designs is very cheap, compared to other process technologies.
- **Power consumption:** As CMOS technology scales down, the downscaling of the power supply follows a similar trend, resulting in lower power consumption. In fact, CMOS technology is optimized for low power.
- **Compatibility with VLSI circuits:** Since CMOS technology is already optimized for logic and memory, it is easy to integrate VLSI circuits with CMOS image sensors.

- **Radiation hardness:** CMOS image sensor technology is more hardened against the radiation defects than CCD technology. Therefore, the CMOS technology is often used for aerospace applications.

For smart sensors, CMOS becomes a good candidate for the image sensing and integration of processing logic. However, there are a number of disadvantages when CMOS technology is implemented, particularly for CMOS active pixel sensors. According to [15], the major disadvantages for implementing smart sensors are as follows:

- **Analog circuit:** CMOS technology is typically developed for digital logic and memory. They are not well characterized and not optimized for analog circuits. However, some leading edge technology like RF CMOS brings people's attention to this analog characterization.
- **Photodetectors:** Because image-sensing field is relatively a new era for CMOS standard process technology, the photodetector structures are not well characterized. Even in recent years, although many companies optimized their fabrication processing or sometimes modify the processes from the standard ones for CMOS image sensors, still characteristics of photodetectors need to be assured by the designers. It is the designers' responsibility to assure that the photodetectors function as desired.
- **Second order effects:** In CMOS process technology, especially for logic and memory, some second order device characteristics, such as subthreshold operation, are usually ignored or paid less attention. However, sometimes these second order effects play critical roles such as conversion gain, pattern noise, etc, in image sensing designs. Therefore, CMOS technology is sometimes difficult to optimize these image sensing behaviours.
- **V_t and Lithographic Mismatches:** Mismatch in CMOS devices is relatively high, which jeopardizes the image quality in CMOS active pixel sensors. Mismatch in CMOS devices often leads a poor quality of spatial noise or pattern noise in CMOS active pixel sensors, which becomes one of main challenges in CMOS image sensor array design.

CIS suffers from relatively poor image quality compared to CCD. However, as the CMOS technology becomes mature, as well as its optical characteristics in specialized process, its attraction and quality expectation get higher. For smart sensors, where proper balance between image quality and processing circuitry is important, CMOS will be the most suitable technology in the future. As the image quality of the CMOS active pixel sensors improve, it will be exciting to see what smart sensors (beyond only image sensors) become in the next few decades.

2.4. Fundamentals of CMOS Image Sensors

2.4.1. Optical Absorption and Photo-Generation

Photon detection happens through the excitation of a bound electron to an unbound state. The energy of a photon can be transferred to an electron in the valence band of a semiconductor. Then, if the photon energy is larger than the bandgap energy E_g , the electron in the valence band is brought to the conduction band. This is how the photon is absorbed in a semiconductor material and how an electron-hole pair is generated. Photons with energy smaller than E_g , however, cannot be absorbed and thus, the semiconductor is transparent for light with wavelengths longer than $\lambda_c = hc_0/E_g$ (where λ_c is cut-off frequency, h is Planck's constant and c_0 is the velocity of light in vacuum). For example, for Si, $E_g = 1.12$ eV and λ_c is $1.11 \mu\text{m}$ whereas for Ge $E_g = 0.66$ eV and the corresponding $\lambda_c = 1.87 \mu\text{m}$.

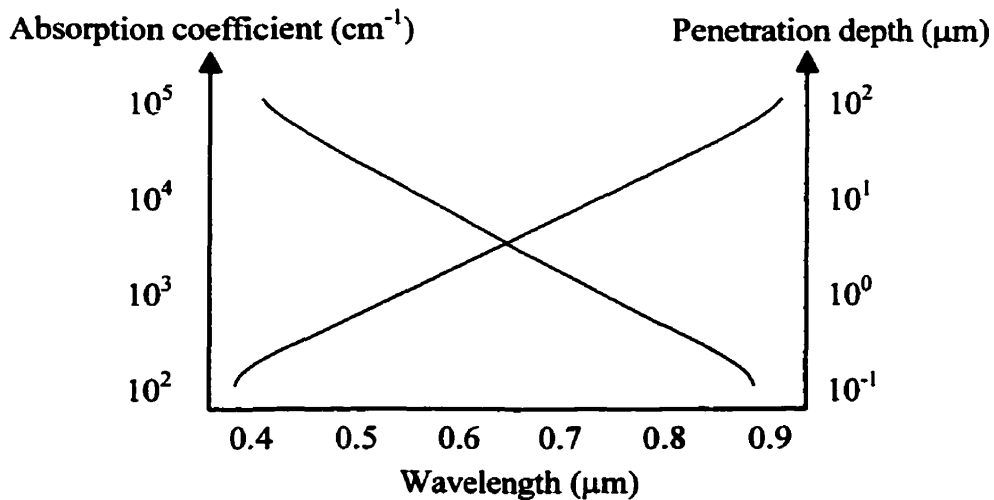


Figure 2.3. Absorption coefficient and penetration depth of silicon at different wavelengths of incidental light.

The optical absorption coefficient α plays an important role in photodetectors. The absorption coefficient, α , indicates what fraction of light a given material absorbs at a given wavelength. Therefore, the absorption of photons in a photodetector, to produce electron-hole pairs and thus a photocurrent, depends on the absorption coefficient α for the given wavelength of the light in the semiconductor. The absorption coefficient also determines the penetration depth ($1/\alpha$) of the light in the semiconductor material according to Lambert-Beer's law:

$$I(y) = I_0 e^{-\alpha y} \quad \text{Equation 2.4.1}$$

Here, I_0 is the light intensity at the surface and y is the depth under the surface. The penetration depth of the light is at the location where the light intensity becomes $1/e$ (63%) of the surface light intensity, I_0 , whose relation with the absorption coefficient is shown in Figure 2.3. Absorption coefficients strongly depend on the wavelength of the light. The slope of the onset of absorption depends on the type of band-band transition. Therefore, this slope is large for direct band-band transition as found in GaAs, InP, Ge and $\text{In}_{0.53}\text{Ga}_{0.47}\text{As}$ because these materials have higher probability for electrons to transfer from valence band to conduction band with less energy, compared to indirect transition materials [85]. For Si, Ge and wide bandgap material 6H-SiC with indirect band-band transition, the slope of the onset of absorption is relatively small. However, silicon detectors are appropriate for the visible and near infrared spectral range. The absorption coefficient of Si is one to two orders of magnitude lower than that of the direct semiconductors in the visible spectral range. Therefore, a much thicker absorption zone is needed than for the direct semiconductors. This is a reason why amorphous silicon can have much thinner films for sensing than silicon materials. However, silicon is economically the most important semiconductor and thus silicon-based imaging devices and integrated circuits are popular in spite of the non-optimum optical absorption.

2.4.2. Photon Collection (Quantum Efficiency)

We have seen how the photons penetrate through materials and how these materials absorb the incoming photons according to its bandgap energy. Here we will look how these penetrated and absorbed photons are collected and transferred in the silicon-based materials.

All carriers that are photo-generated (generated by absorbed photons) in drift regions (also called depletion regions or space-charge regions) contribute to the photocurrent. In other words, all electron-hole pairs generated in depletion regions are collected by its internal electric field (recombination can be neglected due to the fast drift speed). All the carriers photogenerated outside of the depletion region are collected by diffusion rather than the drift mechanism. In the highly doped region (1) of Figure 2.4 and Figure 2.5, the carrier lifetime is reduced significantly due to the high doping density, resulting in a high recombination rate. This considerably reduces the ratio of collected electrons to incident photons, also known as quantum efficiency (QE), for short wavelengths, because a large portion of the short wavelength light is absorbed in region (1).

Light with long wavelengths penetrates deep into the silicon and diffuses in all directions, not only towards the depletion region; overall QE is reduced due to this lower collection efficiency. Since minority carrier diffusion in conventional semiconductor materials is much slower than the carrier drift, collection of photogenerated carriers in region (2) is much slower than that in the depletion region. Therefore, the recombination of photogenerated carriers in N⁺ (region 1) and P (region 2), due to the relative slow diffusion speed, reduces the quantum efficiency. In the high dynamic case, carriers being photogenerated in region (1) and especially in region (2) may not have enough time to diffuse to the depletion or drift

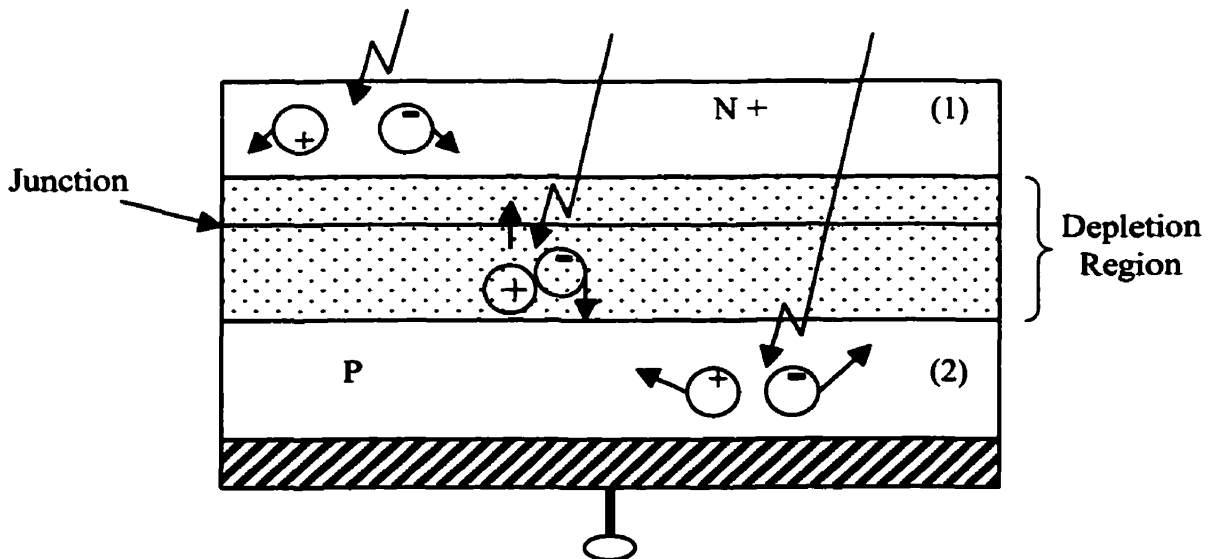


Figure 2.4. Photo-generation and collection of photon-generated electron-hole pairs in an n^+p photodiode.

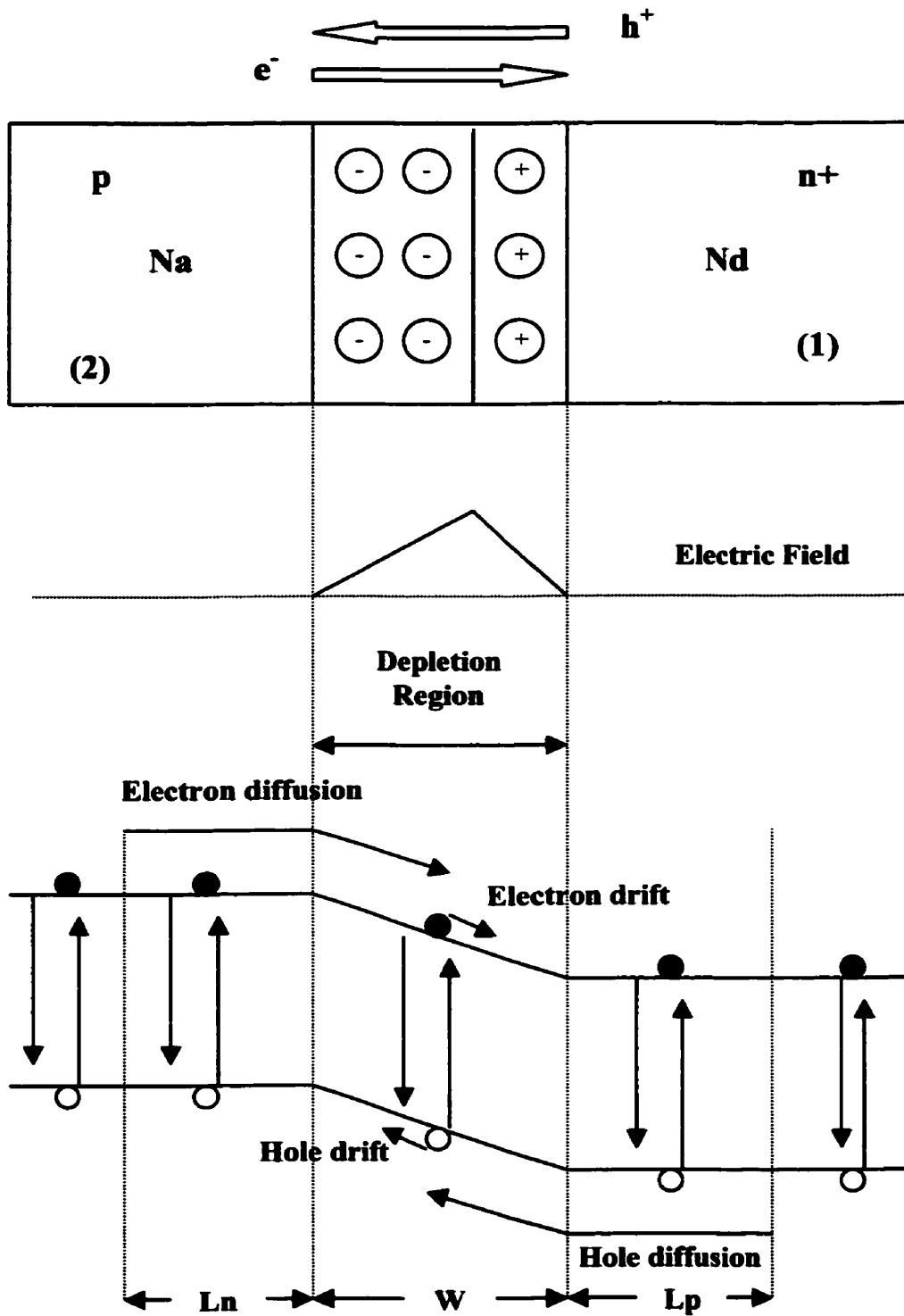


Figure 2.5. Drift and minority diffusion in collection of photogenerated charge.

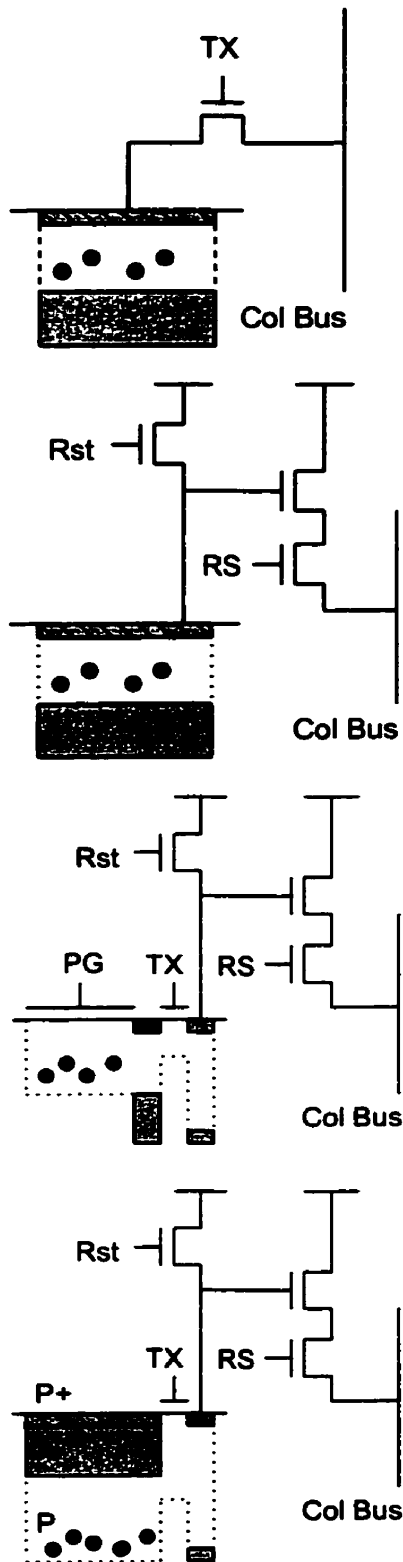
region before the light intensity is reduced again. The dynamical quantum efficiency, therefore, depends on the frequency or data rate. The higher both of these are, the smaller the dynamical quantum efficiency becomes [85].

In addition, the recombination of photogenerated carriers in region (2) can still reduce the quantum efficiency. The recombination of photogenerated carriers in region (1), however, is not so important for long wavelengths due to the large penetration depth and the relatively small portion of photogenerated carriers absorbed in region (1).

2.4.3. CMOS Photodetectors

Based on the fundamental mechanisms in the absorption and collection of photogenerated electron-hole pairs, we continue our investigation on different forms of CMOS photodetectors. The detailed descriptions and comparisons of major CMOS photodetectors are well arranged in Fossum's paper [1]. The following comparisons are adapted from the Fossum's paper, with the addition of another significant photodetector structure, the pinned photodiode. Figure 2.6 shows main photodetector types of CMOS image sensors. These can be divided mainly into two types: passive pixel sensors (PPS) and active pixel sensors (APS). The PPS consists of a photodiode and a select transistor. A charge integration amplifier (CIA) readout circuit is located at the bottom of the column bus to keep the voltage on the column bus constant. With a given pixel size, it has the highest design fill factor because it has only one transistor for the readout. QE (quantum efficiency) can be quite high due to the large fill factor and absence of an overlying layer of polysilicon as found in CCDs. The passive pixel structure has the major problems of their readout speed and noise level due to large capacitive load. Since the large bus is directly connected to each pixel while it is read out, the RC time constant is very high and therefore, the readout speed is slow. In addition, due to the large capacitive load, a passive pixel's readout noise is typically high, with the order of 250 electrons rms, compared to commercial CCDs with less than 10 electrons rms of read noise. Therefore, the passive pixel does not scale well to larger array sizes or faster pixel readout rates.

When the passive pixel sensor was introduced by Weckler in 1967 [5], the problems of the passive pixel were quickly realized and a sensor with an active amplifier within each pixel, called an active pixel sensor, was proposed. The CMOS APS trades pixel fill factor for



CMOS Passive Pixel Sensor (PPS)
 Maximized fill factor
 Smaller pixel size as technology scales
 1 transistor, 2 lines
 High yield due to its simplicity
 High QE due to few overlaying device
 Slow readout and high noise due to high bus capacitance

Photodiode CMOS APS
 Pixel consists of a floating reverse biased p-n junction
 3 transistors, 4 lines per pixel
 Sense node and integration node are same
 Noise and full-well trade against each other
 Moderately high Quantum Efficiency (QE)

Photogate CMOS APS
 Pixel consists of a MOS capacitor coupled to a floating reverse biased p-n junction
 5 transistors, 6 lines per pixel
 Sense node and integration node are separate
 Low noise, small full-well
 Low QE
 Difficult to implement in advanced sub-micron process

Pinned Photodiode CMOS APS
 Pixel consists of pinned diode (p⁺-n-p)
 4 transistors, 5 lines per pixel
 Sense node and integration node are separate
 Low noise, very small full-well
 QE lower than that of PD
 Difficult to implement in advanced sub-micron process

Figure 2.6. CMOS photodetectors.

improved performance compared to passive pixels by using the voltage buffer (source follower) within a pixel. Typically, the pixels have a fill factor of 20~30% [1]. Due to the loss in fill factor, the photon-generated signal is reduced. However, the reduced capacitance in each pixel leads to lower read noise level of the array, and therefore the dynamic range and SNR increases. Main types of the active pixel sensors can be subdivided further into photodiode, photogate and pinned photodiode (see Figure 2.6).

Photodiode APS: Pixel array has on-chip timing, control, correlated double sampling and fixed pattern noise (FPN) suppression circuitry. It has three transistors in each pixel with a typical pixel pitch of 15x minimum size of the technology [1]. The photodiode APS has higher QE than the photogate pixels (Figure 2.6) because there is no overlying polysilicon which is required for photogate. The output photodiode signal is supposedly independent of detector size because a decrease in detector size is compensated by an increase in conversion gain with less pixel capacitance. However, peripheral capacitances from the perimeters of the detector increase the total capacitance of the sensing node and thus, decrease the conversion gain. Despite of the reduction of the capacitance in the pixels, read noise is limited by the reset noise on the photodiode since correlated double sampling is not truly correlated without frame memory. As the pixel size scales down, photosensitivity decreases and the reset noise scales as $C^{1/2}$, where C is the photodiode capacitance. Therefore, the tradeoff can be made in designing pixel fill factor (photodiode area), dynamic range, Signal-to-Noise Ratio (SNR) and conversion gain ($\mu\text{V}/e^-$).

Photogate APS: The basic idea of photogate pixel comes from CCD. While photon-generated charge is integrated under a photogate with high potential well, the output floating node is reset and the corresponding voltage is read out to one of S/H in CDS. When the integration is done, the charge is transferred to the output floating node by pulsing signal on the photogate. Then the corresponding voltage from the integrated charge is read by the source follower to the second S/H of the CDS. The CDS outputs the difference between the reset voltage level and the photo-voltage level. The correlated double sampling can suppress reset noise, $1/f$ noise, and FPN due to V_t and lithographic variations in the array. Therefore, the main noise of the photogate is photon shot noise that cannot be suppressed by any means. The photogate has a pixel pitch typically equal to 20x the minimum size of the technology due to five transistors in each pixel. The floating diffusion capacitance is typically made with

a small capacitance of the order of 10 fF yielding a conversion gain of 10-20 $\mu\text{V}/e^-$ and $2 e^-$ reset noise. However, due to the overlaying polysilicon, there is a reduction in quantum efficiency, particularly in the blue. However, the reduction of noise level increases the total dynamic range and SNR.

Pinned photodiode APS: The pixel consists of pinned diode (p^+n-p), where photon collection area is dragged away from the surface in order to reduce surface defect noise such as dark current. Photon-generated charge is integrated under a pinned diode and transferred to the output floating diffusion for the readout. Similar to the photogate, sense node and integration node are separated so as to optimize the noise. However, the main difference from the photogate is that the potential well for the charge collection is generated by buried intrinsic layer (or n type layer) instead of pulsed gate voltage in the photogate. Each pixel has four transistors and five control lines, resulting in fill factor, which is higher than photogate, but lower than photodiode. In addition, due to a small photon collection area of pinned diode, it has a very small full well for photon-generated charge collection with lower QE, compared to the photodiode.

2.4.4. Active Buffer in Pixel

A definite difference between active pixel sensors and passive pixel sensors is the inclusion of an active buffer into the pixel. The passive pixel sensors suffer from low data rate and high

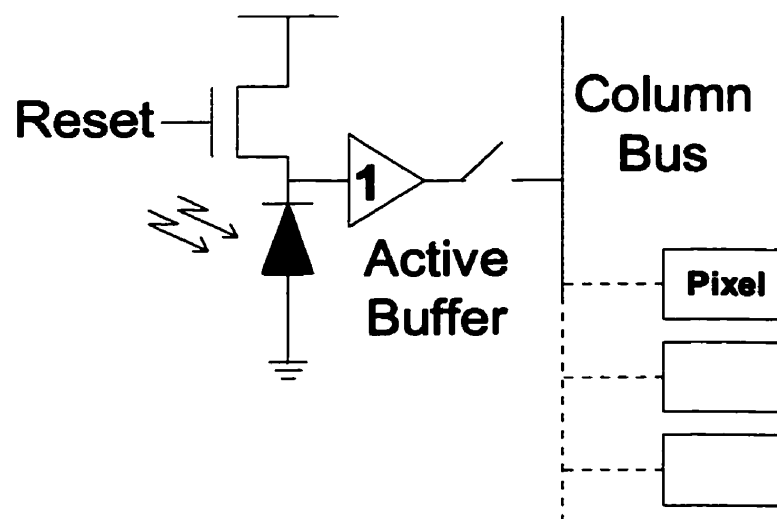


Figure 2.7. Active pixel sensor with photodiode and active buffer.

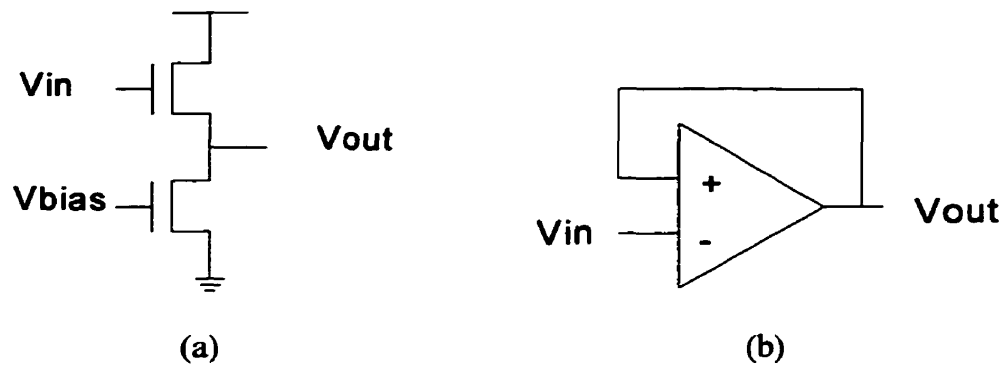


Figure 2.8. Active buffers in CMOS APS with photodiode: (a) NMOS source follower (b) Unity gain amplifier.

readout noise due to the large capacitive loads that are directly connected to photodetection area. In each active pixel, an active buffer is placed, connecting it to the column bus line, as seen in Figure 2.7. By adding the buffer, the charge integration area of the pixel is isolated from the column bus, and instead connected to the gate of the active buffer, whose capacitance is much smaller than that of the bus line. The smaller capacitance of the integration and conversion node of the pixel allows a faster data rate and a lower readout noise. Types of active buffers are source follower, unity gain amplifier and others, as shown in Figure 2.8.

Source Follower: Source follower, typically a NMOS source follower, is a common choice for APS arrays because of its simplicity and small number of transistors. Source followers, however, suffer from lithographical mismatches and V_t deviations, resulting in significant pattern noise in the image sensor array.

Unity Gain Amplifier (UGA): It has a feedback between input and output, remaining at a steady gain of 1 despite of the lithographical and V_t mismatches. However, due to complexity of circuits and a relatively large number of transistors for the OPAMP, the UGA cannot find a practical fit in a pixel. Instead, the UGA is located per column where the implementation area is flexible in the vertical direction. Photon Vision Systems Inc. produced a clever way to implement UGA per column with CMOS image sensors, so called Active Column Sensor (ACS), claiming reduced FPN of less than 0.1 % [16].

Others: There are many different kinds of active buffers implemented with CMOS image sensors, such as adaptive pixel sensors, pixels with feedback for low FPN and pixels with current amplifier. These pixels are for special uses with various applications, different from those of standard voltage buffers. In addition, the complexity of the circuit and the number of transistors are often so large that they cannot be easily implemented in pixels for practical applications.

2.4.5. Operation of Active Pixel Sensor with Photodiode

We have come to understand basic structures of active pixel sensors and their operation. Here, a more detailed mathematical analysis of these operations, particularly for the photodiode, is illustrated. There are three stages of the operation in photodiode with integration mode: (1) photocurrent generation, (2) photocurrent integration and conversion and (3) photo-voltage readout [87]. The mathematical analysis is based on these stages of the operation.

First, photocurrent generation in a vertical n-p photodiode consists of drift current and diffusion current. This is written in Equation 2.4.2.

$$\mathbf{J_{tot} = J_{drift} + J_{diff}} \qquad \mathbf{Equation\ 2.4.2}$$

Under the assumptions that the n-layer (Figure 2.9) is thin enough to cause negligible absorption and that thermal generation (dark current) can be ignored and all the incoming light is absorbed ($\eta=1$, 100% of quantum efficiency), optical generation rate can be written as

$$\mathbf{G(x) = I_0 \alpha \exp(-\alpha x)} \qquad \mathbf{Equation\ 2.4.3}$$

here I_0 is the light intensity at the surface and α is the absorption coefficient. The drift current is therefore,

$$\mathbf{J_{gen} = -q \int_0^W G(x) dx = qI_0 [1 - \exp(-\alpha W)]} \qquad \mathbf{Equation\ 2.4.4}$$

here W is the width of the depletion layer and x is the depth from the surface. For $x > W$ in the p-type, a diffusion equation can be written as

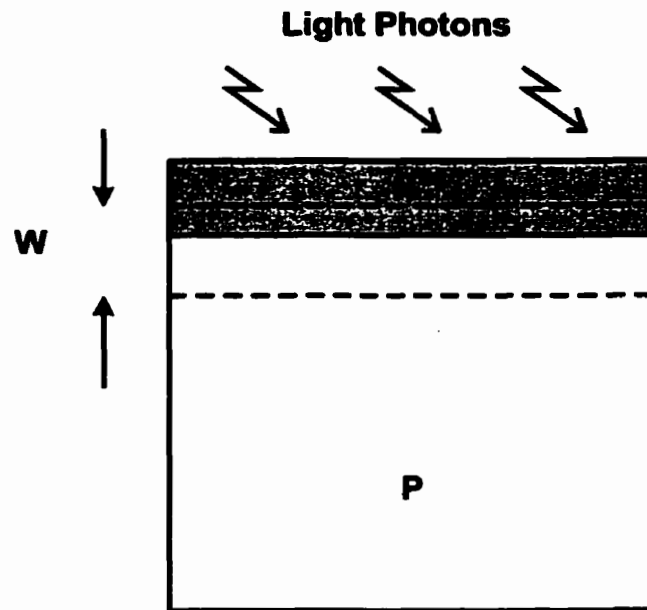


Figure 2.9. Cross sectional view of the photodiode.

$$D_n \frac{\partial^2 N_p}{\partial x^2} - \frac{N_p - N_{p0}}{\tau_n} + G(x) = 0 \quad \text{Equation 2.4.5}$$

here D_n is the diffusion coefficient for electrons, τ_n is the minority carrier lifetime, and p_{n0} is the equilibrium minority carrier concentration. With the boundary conditions for the above equation of

$$P_n = P_{n0} @ x = \infty \quad \text{Equation 2.4.6}$$

$$P_n = P_{n0} @ x = 0$$

equation 2.4.5 can be solved as

$$N_p = N_{p0} - [N_{p0} + C_1 \exp(-\alpha x)] \exp\left[\frac{(W-x)L_p}{L_p}\right] + C_1 \exp(-\alpha x) \quad \text{Equation 2.4.7}$$

where

$$L_n = \sqrt{D_n \tau_n}$$

and

$$C_1 = \left(\frac{I_0}{D_n} \right) \frac{\alpha L_n^2}{1 - \alpha L_n^2}$$

Therefore, the current density of diffusion is given by

Equation 2.4.8

$$J_{diff} = -qD_n \left(\frac{\partial N_p}{\partial x} \right)_{x=W} = qI_0 \frac{\alpha L_n}{1 + \alpha L_n} \exp(-\alpha W) + qN_{p0} \frac{D_n}{L_n}$$

and so the total current density of the photocurrent is

$$J_{tot} = qI_0 \left[1 - \frac{\exp(-\alpha W)}{1 + \alpha L_n} \right] + qN_{p0} \frac{D_n}{L_n} \quad \text{Equation 2.4.9}$$

Therefore, the total current density of photocurrent is linearly proportional to incident light density, as shown in equation 2.4.9.

The second stage of the APS photodiode is the charge integration mode. After the photodiode is reset, the capacitor (Figure 2.10) is discharged by the photocurrent. Therefore, the output voltage of the photodiode is a function of time after the photodiode has been reset. Since the photodiode is isolated, the current in the capacitor must be equal and opposite to the photocurrent (ignoring leakage currents). Hence, the photocurrent can be expressed as

$$C(V) \frac{dV(t)}{dt} = -i_{photo} \quad \text{Equation 2.4.10}$$

For an n^+p photodiode, the capacitance is

$$C_j(V) = \frac{A}{2} \left[\frac{2q\epsilon_{Si} N_A}{V(t)} \right]^{1/2} \quad \text{Equation 2.4.11}$$

where A is the diode area, ϵ_{Si} is dielectric constant of silicon and N_A is the acceptor concentration in the substrate.

When equations 2.4.10 and 2.4.11 are solved, we find

$$\frac{A}{2} (2q\epsilon_{Si} N_A)^{1/2} \left[2\sqrt{V} \right]_{V_{reset}}^{(t) V_0} = -i_{photo} t \quad \text{Equation 2.4.12}$$

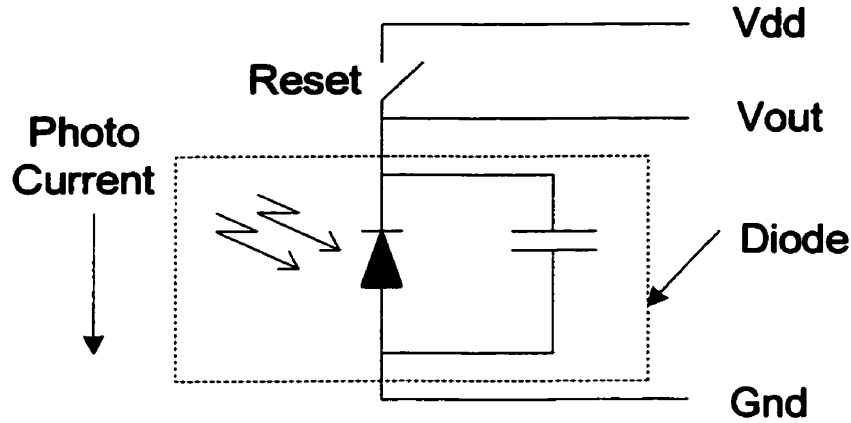


Figure 2.10. Schematic view of photodiode pixel sensor with associated capacitance.

where V_0 is the diode built in voltage, and V_{reset} is the reset reverse bias.

Thus,

$$V(t) = \left[V_{\text{Reset}}^{1/2} - \frac{i_{\text{photo}} t}{A(2q\epsilon_{\text{Si}} N_A)^{1/2}} \right]^2 \quad \text{Equation 2.4.13}$$

Interestingly, this expression includes a term of A , the photodiode area. However, this is cancelled out because $i_{\text{photo}} \propto I_0 A$ where I_0 is the incident flux of photons. Therefore, the collected voltage is independent of the diode area for a given photon flux. In reality, due to the peripheral capacitance of the photodiode and other sources of capacitance not proportional to area, the diode area does have some degree of impact on the total capacitance and thus the output voltage. If we calculate $V(t)$ as a function of time with practical parameters, the voltage drop is almost linear for short times, which is the linearity we want.

The last stage of the APS photodiode is the integrated voltage readout through the active buffer. Provided that $V_{\text{out}} > V_{\text{bias}} - V_{\text{TL}}$, L (Figure 2.11) is in saturation and can be idealized by a current source, I . Then the source follower in the active buffer can be restructured. For transistor M ,

$$I = K [V_{\text{GS}} - V_{\text{TM}}]^2 = K [V_{\text{diode}} - V_{\text{out}} - V_{\text{TM}}]^2 \quad \text{Equation 2.4.14}$$

where $K = 1/2\mu C_{\text{ox}}(W/L)$.

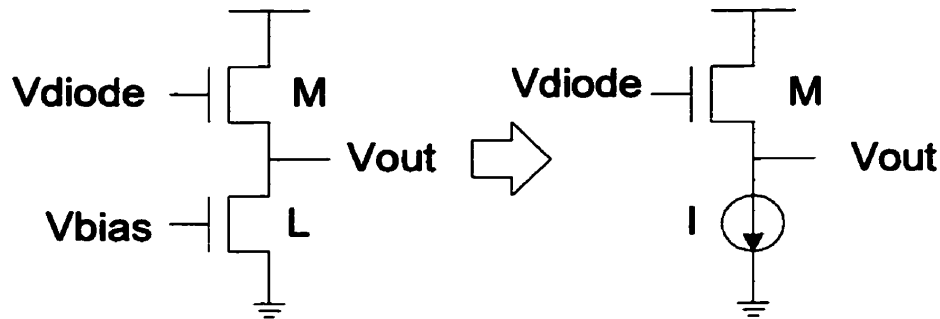


Figure 2.11. Configuration of a source follower as a gate buffer and current source.

Rearranging gives

$$V_{out} = V_{diode} - \left(V_{TM} + \sqrt{\frac{I}{K}} \right) \quad \text{Equation 2.4.15}$$

Where I is the current through the current source. The maximum possible $V_{out} = V_{diode} - V_{TM}$ or including the reset voltage,

$$V_{out} < V_{DD} - (V_{TM} + V_{TR}) \quad \text{Equation 2.4.16}$$

Therefore, the maximum practical output swing is

$$V_{bias} - V_{TL} < V_{out} < V_{DD} - (V_{TM} + V_{TR}) \quad \text{Equation 2.4.17}$$

2.4.6. Readout Control

In addition to active pixel structure, another essential component of CMOS APS array is the readout control circuits, controlling the image readout sequence of the array. Two main structures of the readout control circuit are a decoder and a shift register (SR). The decoder can be used for true random access readout controls because the sequence of the outputs can be selected by the input of the decoders. With RAM, the sequence of the inputs (therefore, outputs) can be programmed in advance.

In contrast, shift registers cannot be programmed for random access readouts because the shift registers produce only sequential outputs from the first element to the last one. Shift registers (SRs) are relatively easier to implement and use fewer transistors than decoders, and SR is easier to expand. This thesis uses two designs of SR: flip-flop structure (FF SR) and two-inverter structure (INV SR). The flip-flop shift register consists of flip-flops (FFs) in

series connected from a flip-flops output to the input of the next one (Figure 2.12). The FF SR transfers its content to the next one by input clock pulse. Since the layout of various FFs can be found easily in design libraries of design packages, the design and implementation of the FF SR is relatively simple. However, because the pre-built FFs have fixed dimensions ($\sim 23 \mu\text{m}$ in our case), it becomes harder to fit the design into a narrower column width, as the pixel size gets smaller. Therefore, a custom design of an FF is required eventually. Another SR structure consists of two inverters in each processing element (Figure 2.13), which holds the input pulse and transfers through control clocks. The INV SR needs two control clocks and thus, its input control becomes harder than for the FF SR. However, due to this small number of transistors, this INV SR can fit into a column width easily. With $0.35 \mu\text{m}$ CMOS technology, we were able to design an INV SR with a $7 \mu\text{m}$ pitch.

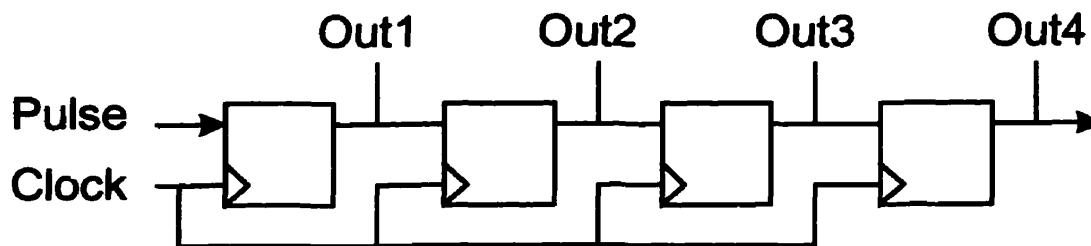


Figure 2.12. Shift register, using flip-flops.

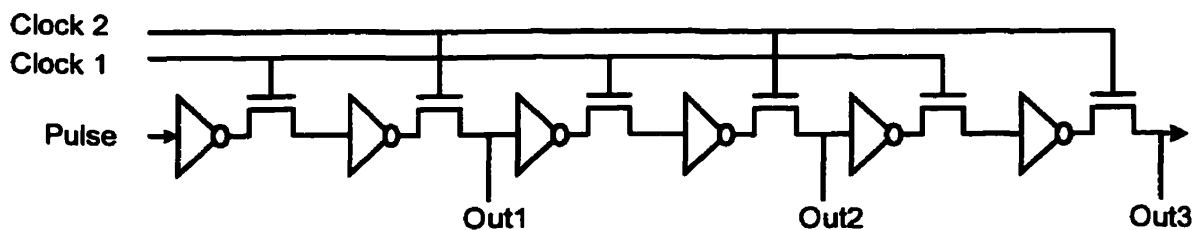


Figure 2.13. Shift register with two inverters in each processing element.

2.4.6. Sample and Hold (S/H)

At some point, unless the array outputs the data in parallel with a same number of channels as columns, an imager array transfers its images to a serial output. Typically whole rows are dumped into storage buffers and then transferred one by one in series to the output. Hence, the array needs storage for the analog image data until all the data of one row are transmitted out. The storage is referred to as a sample and hold (S/H). A standard S/H is shown in Figure 2.14. S/H for CMOS APS typically uses a PMOS source follower (PMOS SF) (as shown in Figure 2.15), as an active buffer because the PMOS SF can compensate for the V_t drop from the NMOS SF in CMOS APS.

Although V_t of NMOS is different from that of PMOS, PMOS SF does the level shifting, positioning output voltage to approximately the same voltage as the photon-sensing node.

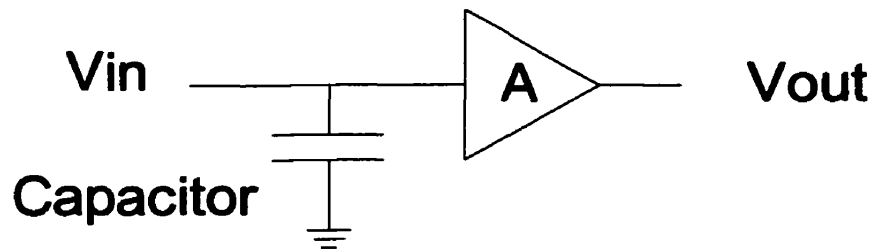


Figure 2.14. Typical sample and hold circuit.

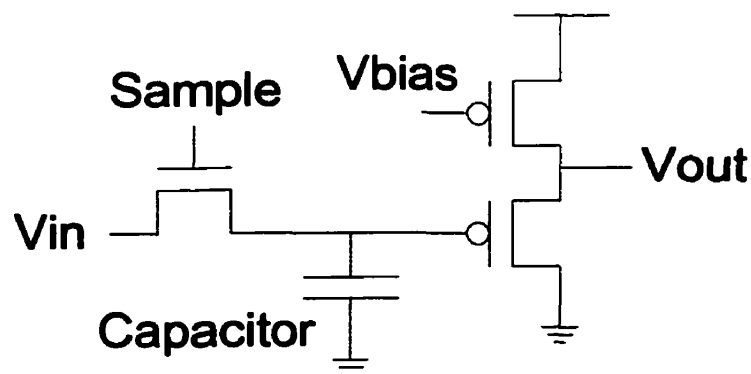


Figure 2.15. Sample and hold with PMOS source follower, used in a typical CMOS APS array.

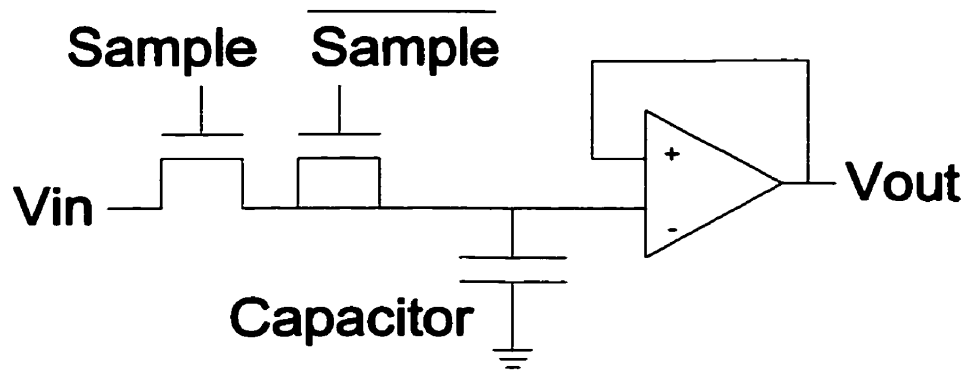


Figure 2.16. An advanced sample and hold.

An advanced S/H is illustrated here in Figure 2.16, with an anti-feedthrough dummy switch and unity gain amplifier. Capacitive feedthrough of the clock happens due to the presence of a capacitive voltage divider between the gate-drain (source) and the load capacitance when the original switch is off. By placing a dummy switch after the switch, half the channel charge is injected toward the dummy switch, matching with charge that would be in capacitive voltage divider. However, it is significant only when a capacitor in S/H is relatively small and becomes comparable to the gate-drain oxide parasitic capacitance. In addition to a dummy switch, a unity gain amplifier can be used. The UGA does not do level shifting like the PMOS SF. With a constant unity gain, it can reduce column pattern noise caused by V_t and lithographical mismatches in the column circuits.

2.4.7. Basic Structure of CIS APS array

We have seen basic components of the CMOS APS. In order to construct a complete CMOS APS array, we need to put them in proper order and in their proper locations. Here, a simple photodiode APS of integration mode is taken as an example (Figure 2.17). Each pixel consists of a photodiode, a reset transistor, a row select transistor and a source follower without bias transistor. The rest of the circuits are located in column. Since the reset transistor and the row select transistor use a NMOS switch, an active high shift register is used for reset and row readout controller. Since PMOS switch is used for a buffer in S/H, an active low shift register is used for column readout controller.

First, the sense/integration (or floating diffusion) node is reset to $V_{DD}-V_T$ and after an integration time (upto one frame readout time), the row select is turned on, dumping the

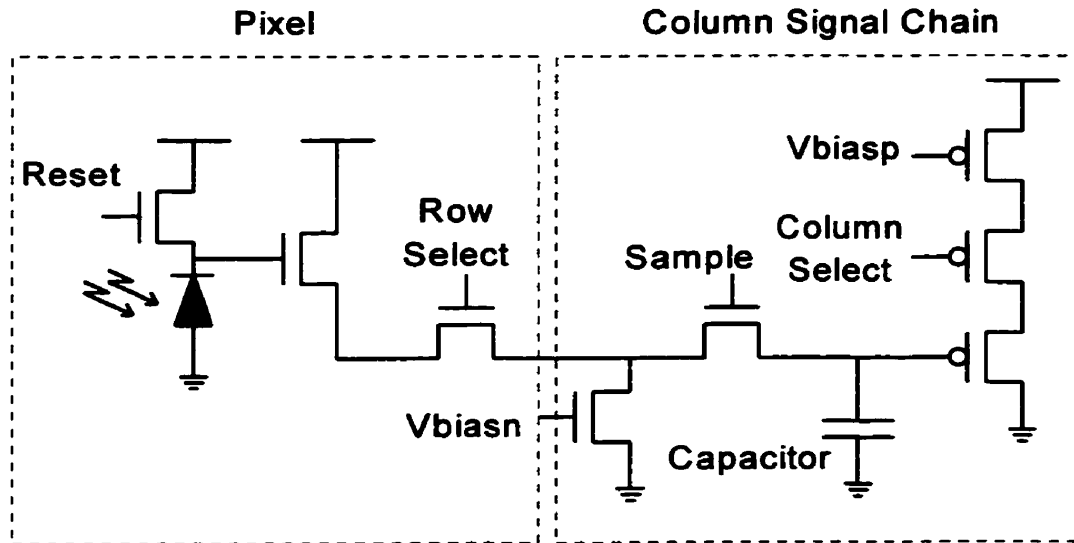


Figure 2.17. Schematic of CMOS APS, including photodiode, active buffer, S/H and output buffer.

image voltage to the S/H. Since the row select is turned on while the reset for the pixel is turned off and the reset for another pixel may be turned on, two separate SRs should run concurrently with different input pulsed, but the same clocks. Once the S/H stores concurrent row images, the column select is turned on one by one until all the columns are read out, as shown in the simplified timing control of Figure 2.18. The sample switch is typically needed because while the column images are read out, row select is still on, dumping image voltages to the S/H continuously.

Therefore, because of the readout time difference between the first column and the last one, the column images will not be concurrent values, potentially causing artifacts. In order to prevent this artifact, the sample switch is activated after the row select is on. Once one row is read out, the next row follows the same procedure and this procedure is repeated until all the values of the array are read out. In order to use the operational time effectively, the reset switch is typically turned on for a short period of time right after the row select is turned off. By doing so, the photodiodes in the row are in integration mode, discharging the floating diffusion by photocurrent while the rest of the rows in the array are read out. Figure 2.19 shows the core structure of CMOS APS array. The reset and row select shift registers are located in both sides of the sensor array. The column select shift register is located at the

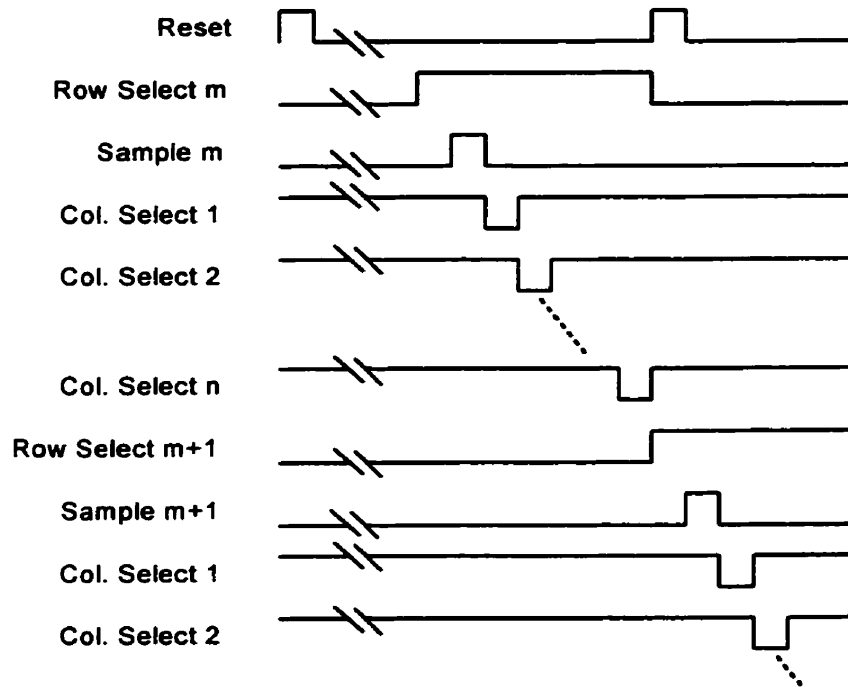


Figure 2.18. A simplified timing control of the photodiode array with integration operational mode.

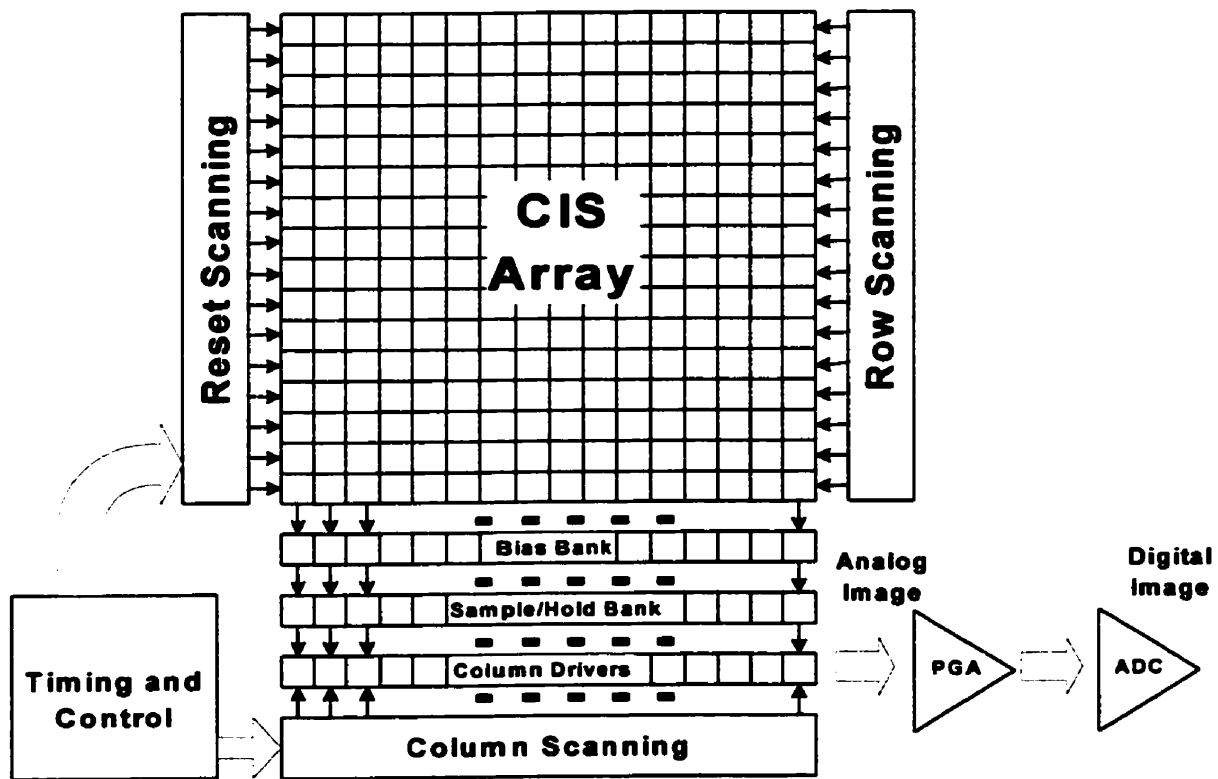


Figure 2.19. Overall structure of CMOS APS array.

bottom of the array, connected to the output buffers in the S/H. The bias transistors are placed away from the pixels for high fill factor, in the columns, so this bias transistor bank is located right below the sensor array and the S/H with output buffer is placed below. The timing and control can be on the same focal plane with the sensor array or off the chip.

2.5. Future Research Focuses of CMOS Image Sensor

Using standard CMOS technology, various image sensor arrays have already been demonstrated by numerous research groups including NASA's Jet Propulsion Laboratory, Lucent technology, IMEC, VLSI vision Ltd., IBM, Hyundai and many other companies, as shown in Table 2.

Company	Pixel Size	Format	Responsivity	Power	Frame rate	Dark rate	Output
Hyundai	8 μm	800x600	3.5 V/s	63 mW	30 FPS	20 pA/cm ²	10 bits
Conexant	5.6 μm	1280x1024		350 mW	27 FPS		10 bits
Photobit	7.9 μm	640x480	1.6 V/s	300 mW	40 FPS		8 bits
Photobit	10 μm	1024x1024	0.5 V/s	400 mW	500 FPS		8 bits
Toshiba	5.6 μm	640x480		100 mW	30 FPS	50 pA/cm ²	
Motorola	7.8 μm	640x480	3.0 V/s	400 mW	25 FPS	2 nA/cm ²	10 bits
Omnivision	7.6 μm	640x480		120 mW	60 FPS		8 bits
Agilent	9 μm	640x480	2V/s	200 mW	30 FPS	3 nA/cm ²	9 bits
Fill-factory	7 μm	1280x1024	1V/s	300 mW	8 FPS	0.25 nA/cm ²	
STM/VVL	7.5 μm	640x480		80 mW	30 FPS		10 bits

Table 2. Present status of CMOS image sensors from several companies [86].

The next generation of CMOS imaging technology is expected to develop in two directions. The first effort is for highly miniaturized, low-power, high quality imaging systems. Such imaging systems are driven by performance, not cost. This effort is led by the U.S. Jet Propulsion Laboratory (JPL) for next-generation deep-space exploration. CMOS image sensor is a suitable technology because of its relative radiation hardness for space applications. In addition, because CMOS consumes low power, the weight of the battery can be drastically reduced. However, since CMOS APS still suffers from high dark current and noise, leading to relatively poor image quality, this high performance research typically focuses on low noise, high image quality, low power, high speed and high resolution.

The second effort is to create highly functional single-chip imaging systems where low cost, and not performance is the driving factor. Although CCD technology is highly optimized for image sensing applications, its cost will probably not be significantly reduced in the future and applications will require multiple chip systems. For many researchers, the advantage in developing CMOS imaging technology is the complete integration of image sensor with low cost, analog-to-digital converters, driving and control circuitry, and sophisticated interfaces - all convenient to addressing the technical challenges posed by digital imaging applications. In addition, the integration of image processors with CMOS image sensors remains as an attractive opportunity, with recent great successes in various applications, such as digital still camera, video cellphones, surveillance, medicine and dentistry, aerospace, machine vision and automobile industry.

In CMOS image sensor technology, these two research directions are not win-or-lose situations, rather they are two distinct future research fields. Despite the aggressive developments of CMOS image sensor performance, there are debates whether CCDs will defend their position as the dominant image sensor technology in the future and never give up its mainstream market to the CMOS counterpart. However, CMOS image sensors will find their places for imaging systems and applications, for example, for space applications and for portable devices like videophone and PDA. In the long term, the ability to integrate complete CMOS imaging systems on a single chip will be one of the driving focuses in developing the next generation of multimedia imaging systems.

Chapter III

3. MOSAIC Multi-Camera Imager System with CMOS Image Sensors

3.1. Introduction

As a part of the future expectations of CMOS image sensors, a method of achieving high resolution over a wide field of view is investigated. An integrated smart sensor, MOSAIC (Matrix of Semi-Autonomous Imaging Cameras), for large field of view is proposed in this thesis.

A MOSAIC imager design is described for a distributed sensor consisting of $10^2 - 10^3$ identical detection modules linked by a serial bus to a central controller, seen in Figure 3.1. Since smaller single chips are used in the MOSAIC imager, relatively high yield,

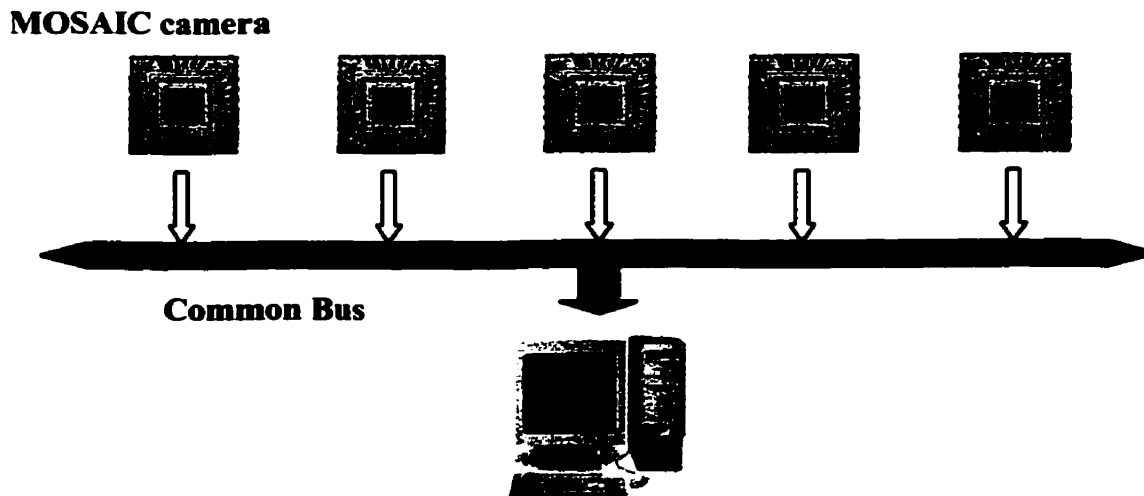


Figure 3.1. MOSAIC multi-camera system with a central controller.

high resolution and low cost can be achieved. The MOSAIC concept can be applied to various applications such as airborne remote sensing, the filling of the focal plane of a large telescope, monitoring of the sky for meteors, monitoring of ships at sea, inter-satellite data sharing, and perimeter surveillance.

One of the focuses of the present MOSAIC system is the development of an efficient communication mechanism, achieved by integrating the CMOS image sensor and bus interface module on the same chip. The integrated bus interface module increases performance of the bus connections by a zero-wait state design that does not require operation time for address over-head. MOSAIC imagers increase the field of view and fabrication cost effectively, by connecting single-chip cameras in a coordinated manner equivalent to a large array of sensors. Components that would have conventionally been in separate chips can be integrated on the same focal plane by using CMOS image sensors. Here, a MOSAIC imaging system is constructed using CIS connected through a bus line (called the image-bus) which shares common input controls and output(s), and enables additional cameras to be inserted with little system modification. The MOSAIC system consumes relatively low power by employing intelligent power control techniques. However, the bandwidth of the bus is still expected to limit the number of camera modules that can be connected in the MOSAIC array. Hence, signal-processing components, such as data reduction and encoding, will be needed on-chip in order to achieve high readout speeds (these will be addressed in Chapters 5, 6, 7). Basic modules for a single-chip camera are proposed for efficient data transfer and power control in MOSAIC imager.

In this thesis, the MOSAIC smart image chip, corresponding to the scheme described above, is implemented using a CMOS 0.35 μm double poly technology with 3.3 V power supply. The implementation demonstrates the advantages of the single chip solution for the MOSAIC imager in terms of area, power, speed, and fabrication cost. The thesis describes the design and performance results of the chip, along with their background algorithms. In addition, the design of the intelligent bus interface and the architecture of the system are addressed.

3.2. Single Chip verse Multi-chip Systems

Large-format and MOSAIC imagers for astronomical, surveillance and other applications require high spatial resolution, coverage of a large area, effective cost and efficient image

update rate. One solution for large format applications is a single monolithic chip, made with either a large array of pixels or an array of large-sized pixels. A large pixel (optical) area, Figure 3.2 (a), leads to low resolution that is often not desirable, while increasing the number of pixels in the array, Figure 3.2 (b), leads to a high complexity of circuits and consequently a high noise floor. In addition, large single chips have relatively low yield, resulting in a high fabrication cost. Another solution for the large-format image sensor applications is a MOSAIC system containing many individual sensor chips, as shown in Figure 3.2 (c).

3.3. Previous MOSAIC Implementations

There have been several attempts to implement the MOSAIC concept into image acquisition applications. This thesis takes three examples where the MOSAIC concept has been applied: machine vision [17], astronomical telescope [18] and medical tele-pathology [19].

There are several previous designs for machine vision such as the DRIFT bus and Improved Integrated Smart Sensor (I^2S^2) bus [17]. These are efficient, high performance bus structures in machine vision. The buses are used for communication between image processors and memory modules or other peripheral modules, not as direct connections between image sensor modules. These bus structures focus more on communications between image sensors and peripheral devices, compared to our MOSAIC system where communication among image sensor chips is emphasized. Also, because the bus connection and its handling modules are separately located from the image acquisition modules, the system fabrication cost will be relatively high.

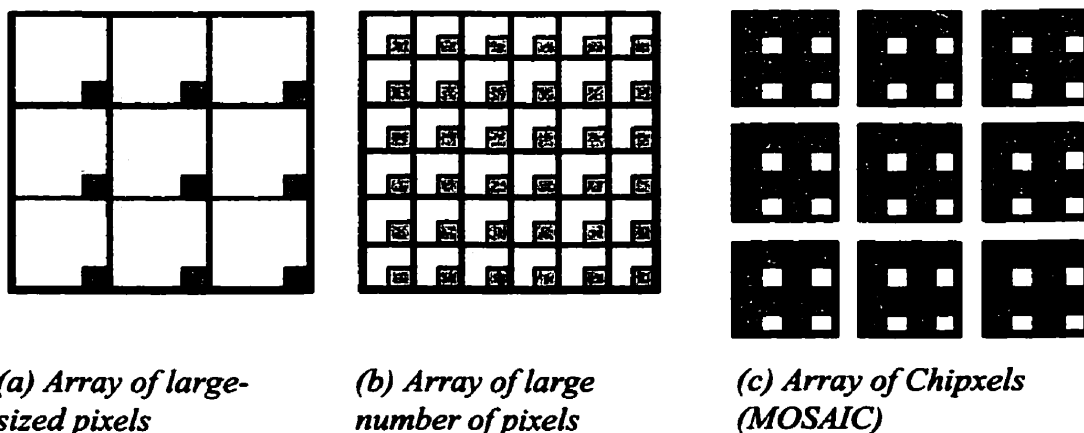


Figure 3.2 Single chip and multi-chip system for MOSAIC system.

Secondly, there is an example of MOSAIC concepts used in astronomy telescope, called NOAO Mosaic Data Handling System [18]. The system takes data from a mosaic of CCDs and decodes, records, archives, displays, and processes the data. The NANO Mosaic CCD Camera consists of 8 CCDs producing an 8K x 8K format. Unlike CMOS cameras, CCD cameras do not contain significant combinational logic, hence communication between the components is handled through a software intensive facility, called a message bus. Also, the use of multiple CCDs requires that data be read out simultaneously from all CCDs, hence the raw data is interleaved as it arrives from the detector and must be “unscrambled” before being written to disk or displayed. Therefore, a powerful computer system and efficient software are required to be able to handle such large formats in the data handling system.

Telemedicine and tele-pathology delivering medical diagnoses and health care to distant patients is another MOSAIC concept implementation [19]. This technology covers the entire view of the patient site with several frames of images, and automatically composes a wide field of view and high-resolution image of patient from these frames by using the computer techniques for generating digital image mosaics. The patient image capturing equipment consists of several high-resolution video cameras, and their connections are made through ISDN network or communication satellites. Therefore, the system may require a higher communication cost because of the greater amount of transmitted information, communication network and computer power. It also emphasizes image interpolation in software, rather than an efficient data transfer mechanism in hardware.

The previous implementations of the MOSAIC concept are shown above to be rather complicated and require intensive integration of expensive software. Often, post-processing mechanisms are required to produce a suitable image quality. These works also need several different functional modules in physically separated forms: camera, processing components, interface modules and bus connections. Therefore, the manufacturing cost is relatively high. In addition, the previous systems focus on problems of software-based image alignment rather than implementation of connection in their image acquisition system because the cameras are not perfectly aligned and have gaps between the cameras, requiring interpolation, image combination and dithering.

A simple and cost effective implementation is suggested in this thesis. A single chip solution of MOSAIC system integrated with low-level hardware pre-processing units is proposed to improve its communication, cost, speed and computing power. The suggested implementation of integrated bus interface, called a “chipxel (chip + pixel)”, is more focused on the low-level hardware design with effective fabrication cost and simple systematic connections. Consequently, the chipxel emphasizes the method of connecting the multi-cameras efficiently, rather than how to interpolate the images from the ordinary cameras in software. The chipxel is unique, compared to the previous works, implementing MOSAIC concept as a single chip solution. Since the optimization of image sensor connections is emphasized in the single bus line, the integrated image camera with processing and bus interface units is proposed here for MOSAIC applications, with considerations for speed, fabrication cost and complexity of the design.

3.4. Design of MOSAIC

3.4.1. Integrated Bus Interface with CMOS Image Sensor

The systematic connections of MOSAIC imager systems can be divided into three different categories as shown in Figure 3.3: multiple inputs to the controller with one output from each camera, one input to the controller through a hub connecting multiple cameras, and one input to the controller connecting multiple cameras through a bus line. In a controller with the multiple inputs, Figure 3.3 (a), the output of each camera is connected to a controller and the controller arbitrates the incoming outputs of the cameras and multiplexes/encodes into one data stream. This connection potentially suffers from high fabrication cost and slow frame rate because the controller needs multiplexer/encoder to combine the multiple streams of data into one stream for further processing. In addition, as the number of cameras in the system increases, the complexity of the controller will increase. When more cameras are added into the system, the controller has to be redesigned to create more channels for the additional cameras and the multiplexer/encoder should be implemented with the new channels. Therefore, the system is less flexible to the inclusion of additional cameras.

In the second architecture, Figure 3.3 (b), the multiplexer/encoder which exists in the controller of the first system is now separated from the controller and replaced with a hub, connecting multiple cameras and streaming one output to the controller. However,

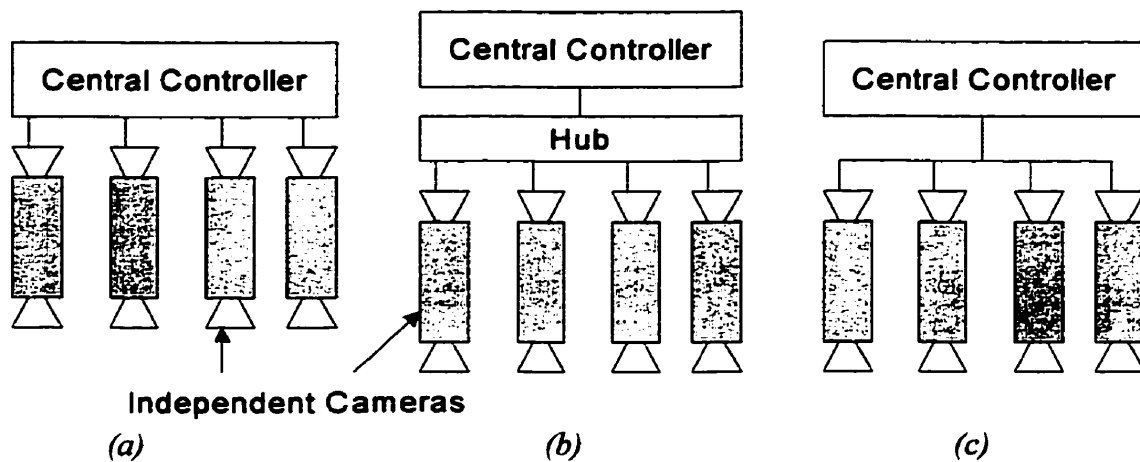


Figure 3.3. Systematic connection of mosaic imager can be categorized into (a) multiple outputs from cameras to controller, (b) multiple outputs from cameras and single input through hub to the controller and (c) single input to the controller with integrated bus interface in cameras.

because there are a limited number of channels from cameras that a hub can take, the fabrication cost and complexity are again relatively high. Whenever additional cameras are connected to the system, extra hubs are required. Now, intelligent cameras of chipxel, each unit with an integrated bus interface, are proposed here. The multiplexer/encoder is taken away from the controller and integrated into each camera. The output data from the distributed cameras are streamed into the controller by the integrated bus interface through a common bus line, as shown in Figure 3.3 (c). Therefore, it is easy to integrate additional cameras into the system with little modification to the central controller or to the connections. In addition, when the MOSAIC system needs independent processing such as event detection, a bus interface is a complementary component in each camera because each camera should be capable of indicating when it detects events and when it needs the bus line. The integrated bus interface therefore significantly increases the flexibility of the system; it requires neither many communication lines nor an expensive hub. In addition, because the signal does not go through many units, the noise level is relatively low and the communication speed is relatively high. In summary, the integrated bus interface in each camera has the advantages of low fabrication cost and high flexibility over the other systems.

3.4.2. Circuits and Layouts

A standard CMOS image sensor array was implemented with the chipxel to demonstrate continuous data transfer in the MOSAIC imager system. A MOSAIC chipxel, whose photo is shown in Figure 3.4, was designed and fabricated with 0.35 μm double poly technology with 3.3 V power supply. The structure in Figure 3.5 includes an image sensor array with pixel readout circuitry, shift registers, sample/hold and bus interface. All the components except for the bus interface are used widely in CMOS image sensor designs. Here, a bus interface was integrated with CMOS image sensor array for the MOSAIC imager connections.

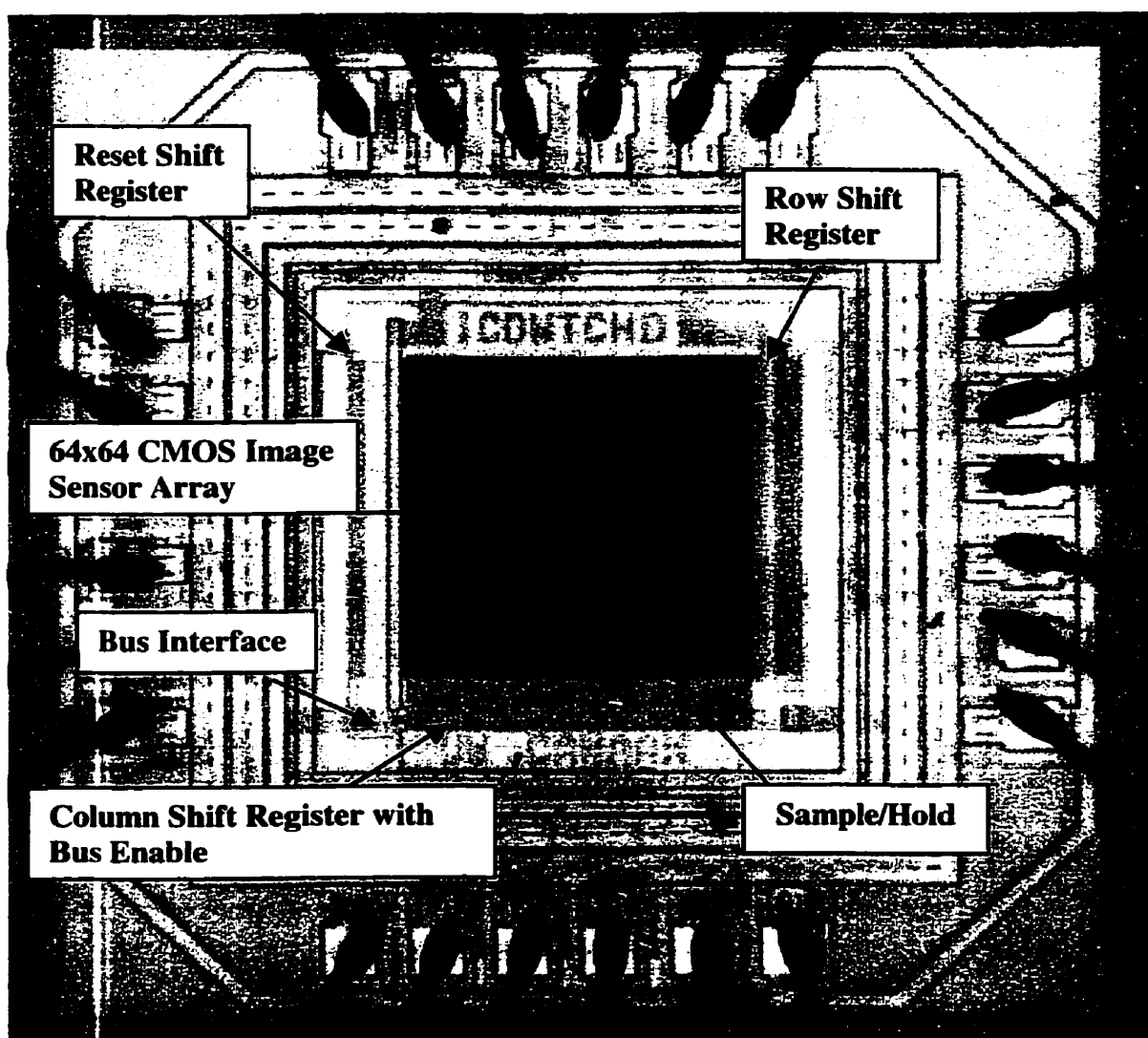


Figure 3.4. Chip photo of MOSAIC chip.

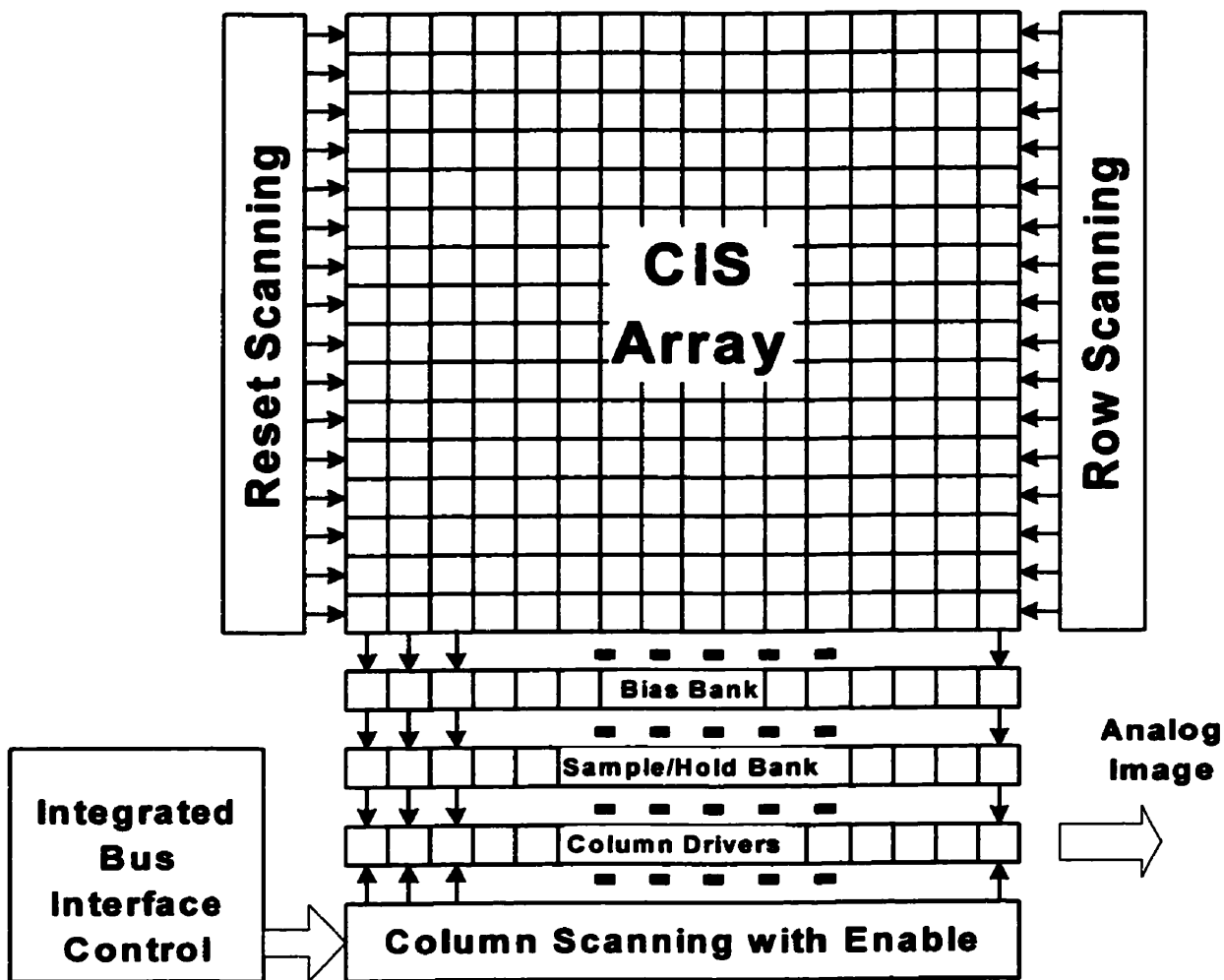


Figure 3.5. Array structure of ideal MOSAIC image sensor with an integrated bus interface.

Each pixel in the CMOS image sensor array consists of a photodetector and readout circuitry, seen in Figure 3.6. The photodetector uses an n+p photodiode, one of the simplest sensor structures in CMOS image sensor technology. A simple source-follower is used for the readout circuitry of the pixel, which blocks capacitor loading from column line of the array. The maximum output voltage of the array, due to the voltage drop of the source follower, is V_t lower than the actual photo-generated voltage, unless further processing occurs (where V_t is the threshold voltage of the source-follower transistor). The second part of the system is for the generation of input control signals. The generation of readout input signals in general can be performed by a shift register, taking less area with a simple design structure, for reset, row and column readout controls in this design. In addition, because the size of shift register

can easily be aligned with each column of the array, the shift register is more suitable for column structure based implementation than the decoder (see Figure 2.4.6). The shift register in the chipxel uses two inverters and two switches with two control clock signals, shown in Figure 3.8.

The sample/hold (S/H) is a storage place for image to be transferred to the outside. Since V_t is lost by the source follower in the pixel readout, the sample/hold uses a PMOS source follower, shown in Figure 3.7, so that the lost V_t voltage can be recovered by V_t rise of the PMOS. Because the source follower in the sample/hold is the off-chip driver, where a large loading exists, the sample/hold should use a large MOSEFT in its source follower. As the size of the driver increases, the driving power of the driver increases, thus speeding up the signal readout on the large external loading. However, the large size also causes larger power consumption. Therefore, an appropriate size of the driver is determined in a given design specification for both speed and power consumption such that the product of the speed and power is at minimum.

There are two mainstream bus interface schemes available for the chipxel chip: independent request and grant (RG) and daisy chain methods. The independent RG sends a bus request signal to the controller whenever it needs to transfer data using its own designated control lines, which is similar to the star configuration in network theory. Therefore, it needs many control lines and the complexity of the design will be high. In contrast, the daisy chain method enables the chip to send its image to the controller whenever it receives the bus grant signal through the daisy chain connection. Hence, the daisy-chain method is relatively slow, but the design is simple and the overall fabrication cost is low.

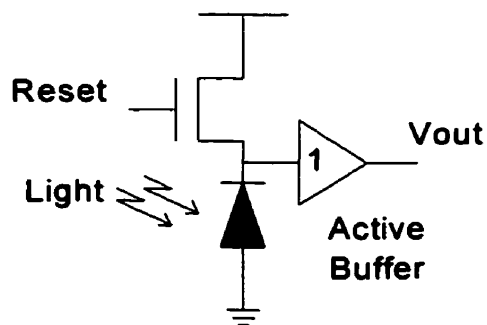


Figure 3.6. Active Pixel Sensor with photodiode and active buffer in integration mode.

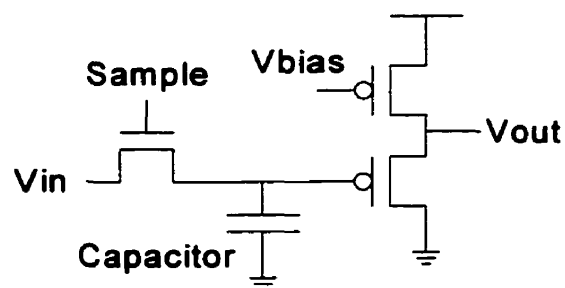


Figure 3.7. Schematic of S/H. A simple S/H is implemented with PMOS source follower for the analog buffer.

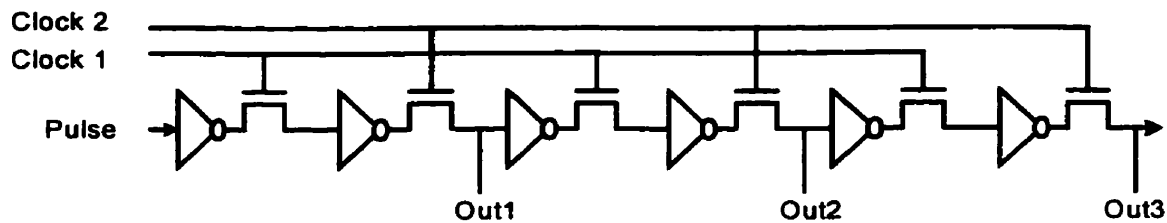


Figure 3.8. Shift register is implemented for readout circuitry, using two inverters and switches.

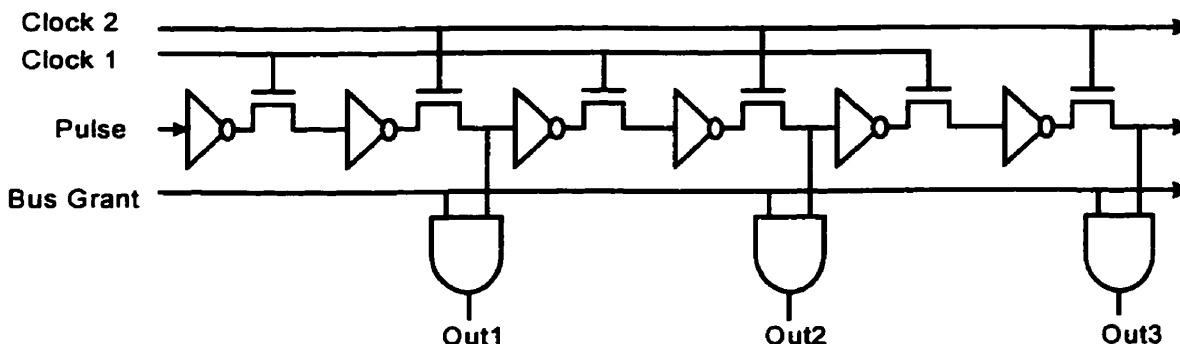


Figure 3.9. Readout circuitry is integrated with switches enabled by Bus Grant signal. The integrated bus interface passes the grant signal, gated by AND function to the sample-and-hold switches.

In addition, the daisy chain does not use any time for address and over-header, which we call the “zero-wait state”, because the images captured by each camera are displayed in sequence. Therefore, the daisy-chain method is chosen for the prototype of the MOSAIC chip. Whenever the Bus Grant (BG) signal comes to a chip, the chip holds the BG signal, enabling the column shift registers and sending out the image, like Figure 3.9. After the chip transfers its frame of image, the BG signal is released to the next chip. The BG signal is once generated by the controller and circulated through the daisy chain until all the images of the system are transferred.

3.4.3. Demonstration and Tests

The first test was to capture an image of the best quality possibly with the chip, verifying test board connections, control input patterns, image display software setup and, most importantly, the design of the chip. The basic procedures of the test for capturing images are

discussed in Appendix B. Figure 3.10 shows the testing setup for the single chip; the testing board contains the chip, lens, lens mount, wire connections for power supplies and biases, and ribbon cables for control input patterns. The input patterns are generated by a software called “GageBit” from Gage Applied Inc. Pulse waveforms are manually drawn in the software and with a appropriate clock rate, it outputs sequence of control patterns. Also, a digital oscilloscope, called “CompuScope” from the same company is used for the data acquisition and image display. With Labview interface, the CompuScope can be programmed to display the image signal into an intensity graph where the signals are displayed as pictures in real time.

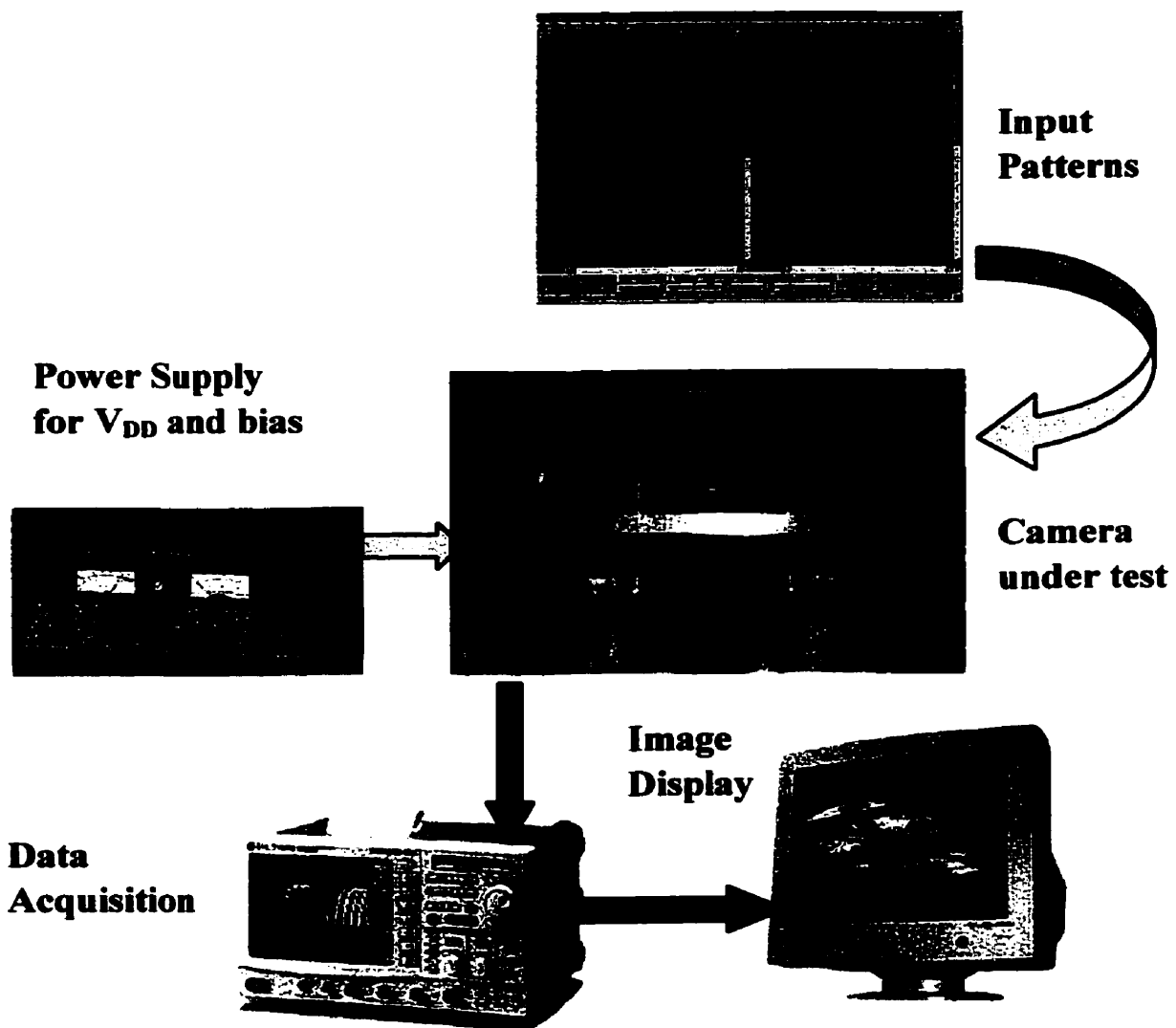


Figure 3.10. Test board with MOSAIC chip and lens mounted. Power supplies and bias voltage lines are shown in the left side of the board. The ribbon cable for control input patterns are connected to the right side of the board.



Figure 3.11. Above: A raw image of Audrey Hepburn, captured by a single image-bus chip. Right: Characteristics of single image sensor.

Technology	0.35 μm CMOS, double poly
V_{DD}	3.3 V
Chip size	1.91 mm x 1.91 mm
Array size	64 x 64
Pixel size	10 μm x 10 μm
Fill factor	46 %
Max. Frame rate	24 frames/sec.
Nominal power	1.46 mW at 5 frames/sec.
Photosensitivity	0.57547 V/lux*sec (with 1 W/m ² ~ 70 lux) 33 mV/($\mu\text{W}/\text{cm}^2$)
FPN	16 mV rms (1.3 % of sat.)
Saturation	1.2 V
Dark signal	0.3 V/sec
Conversion	1.05 $\mu\text{V}/\text{e}^-$
Efficiency	

The characteristics of the single chip, including image sensors, are summarized in Figure 3.11, along with an example of an individual raw image. The characteristics of the chip can be measured and calculated, based on the measurements. According to these measurements, the technology for this chip was not optimal for image sensors. Commercially available CIS chips typically achieve 5~10 $\mu\text{V}/\text{e}^-$ for their conversion efficiency. However, our chip has as estimated conversion efficiency of 1.05 $\mu\text{V}/\text{e}^-$, which is a relatively low result. This estimated conversion gain is calculated from the measurements of the image sensor chip. The detailed calculations and measurements are discussed in the Appendix C. Such a low photosensitivity leads to long integration times and high dark signal, thus degrading image quality. Since the technology is optimized for logic and memory, but not for image sensors, the chip is not expected to display a high performance.

To briefly talk about the tests for characterization of the chip, there are three essential measurements: (1) Measure and save image files at a fixed wavelength and at a fixed frame rate (sampling rate) while changing the illumination (light power or intensity) from 0 to until the output voltage is saturated or an equivalent test by changing the integration time. In addition, the wavelength and frame rate can be varied. This measurement can be directly used for extraction of photosensitivity, PRNU and saturation level. Also, it can be used for

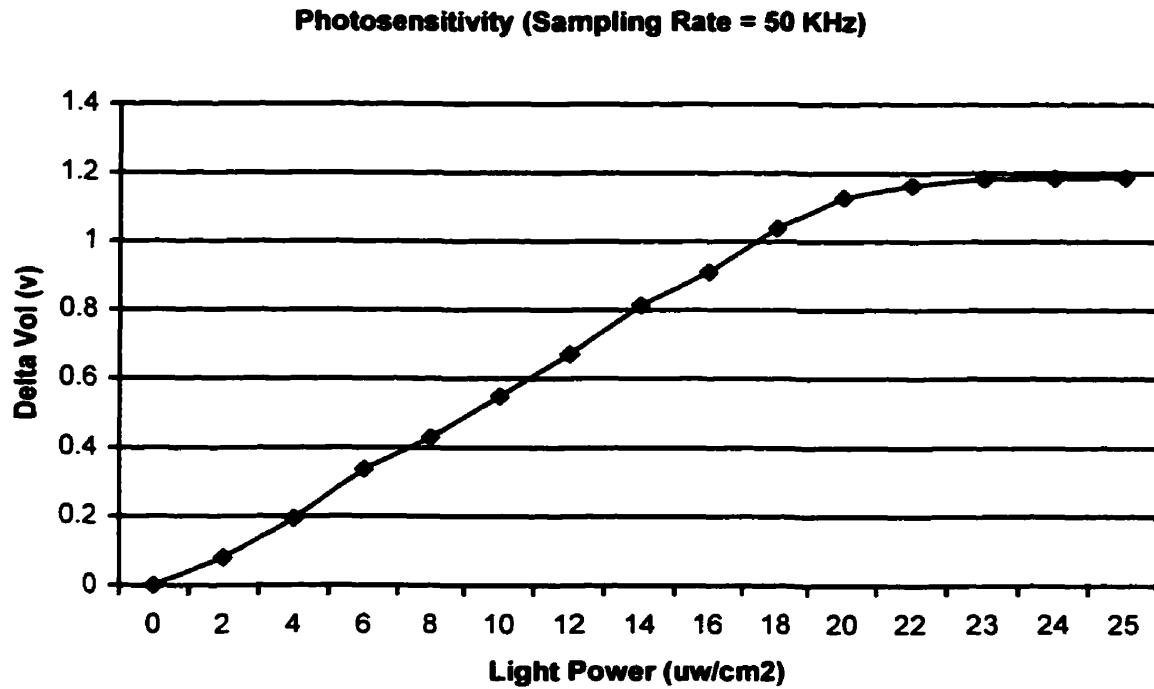


Figure 3.12. Photosensitivity of single chip of MOSAIC. Saturation level of 1.2 V is shown in this diagram at 50 KHz (12 frames/sec).

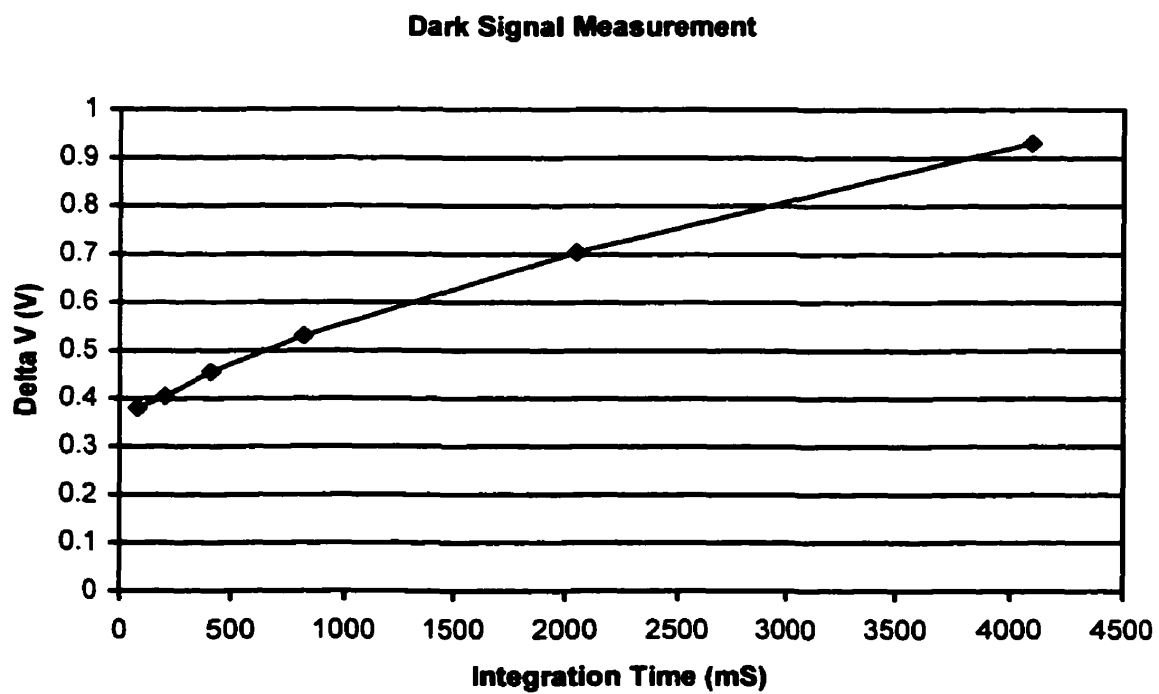


Figure 3.13. Dark current measurement in single chip of MOSAIC.

calculation of conversion efficiency. Figure 3.12 shows the photosensitivity of the MOSAIC chip. From the slope of the graph, the photosensitivity is calculated to be about 0.57547 V/lux*second which is slightly lower than commercial CIS chips. (2) Measure and save image files at fixed illumination (light power) and a fixed frame rate as changing wavelength of the incident light. The illumination and frame rate can be varied. This measurement can be used for spectral response. (3) Measure and save image files in a dark room while changing integration time (sampling rate). This measurement can be used directly for FPN and dark current. Figure 3.13 shows the dark current measurement of the MOSAIC chip. With this chip, there are three variables controlled by users: Vbiasp, Vbiasn (see Figure 2.17) and sampling rate (control patterns' clock rate). Vbiasp and Vbiasn are bias voltages that do not have direct effects on the output images, but only shift saturation level of output voltages. However, when Vbiasp and Vbiasn are out of operational range, the top level of the saturation range hits the V_{DD} of 3.3V, and then the output images are degraded. Figure 3.14 and Figure 3.15 illustrate this phenomenon. The output image of Audrey Hepburn is not affected much between Vbiasn = 0.4 V and 0.6 V. However, when Vbiasn becomes around 0.65 V in Figure 3.14, the output image shows some degradation. Similarly Figure 3.15 also shows no effects on the images between Vbiasp = 2.45 V and 2.70 V. However, when Vbiasp becomes 2.75 V, some degradation appears on the image.

The sampling rate or data rate is directly related to input control patterns' clock rate and to the sensor integration time. It is also related to power consumption and output voltage swing of the chip. Due to the direct relation between sampling rate and integration time, sampling rate affects the quality of the output images. As sampling rate increases, the integration time decreases because the faster sampling rate reduces readout time of one image frame (typically, the maximum integration time of the image sensors is the readout time of one image frame), and thus photon integrating time of the image sensors gets smaller. As seen in Figure 3.16, as sampling rate gets higher, some degradation appears on the output image. When the sampling rate is over 100 KHz, the output image is hardly recognizable. The main limitation of such a low sampling rate is due to the poor photosensitivity of the image sensors; longer integration time is needed to produce a good image quality with a poor photosensitivity.

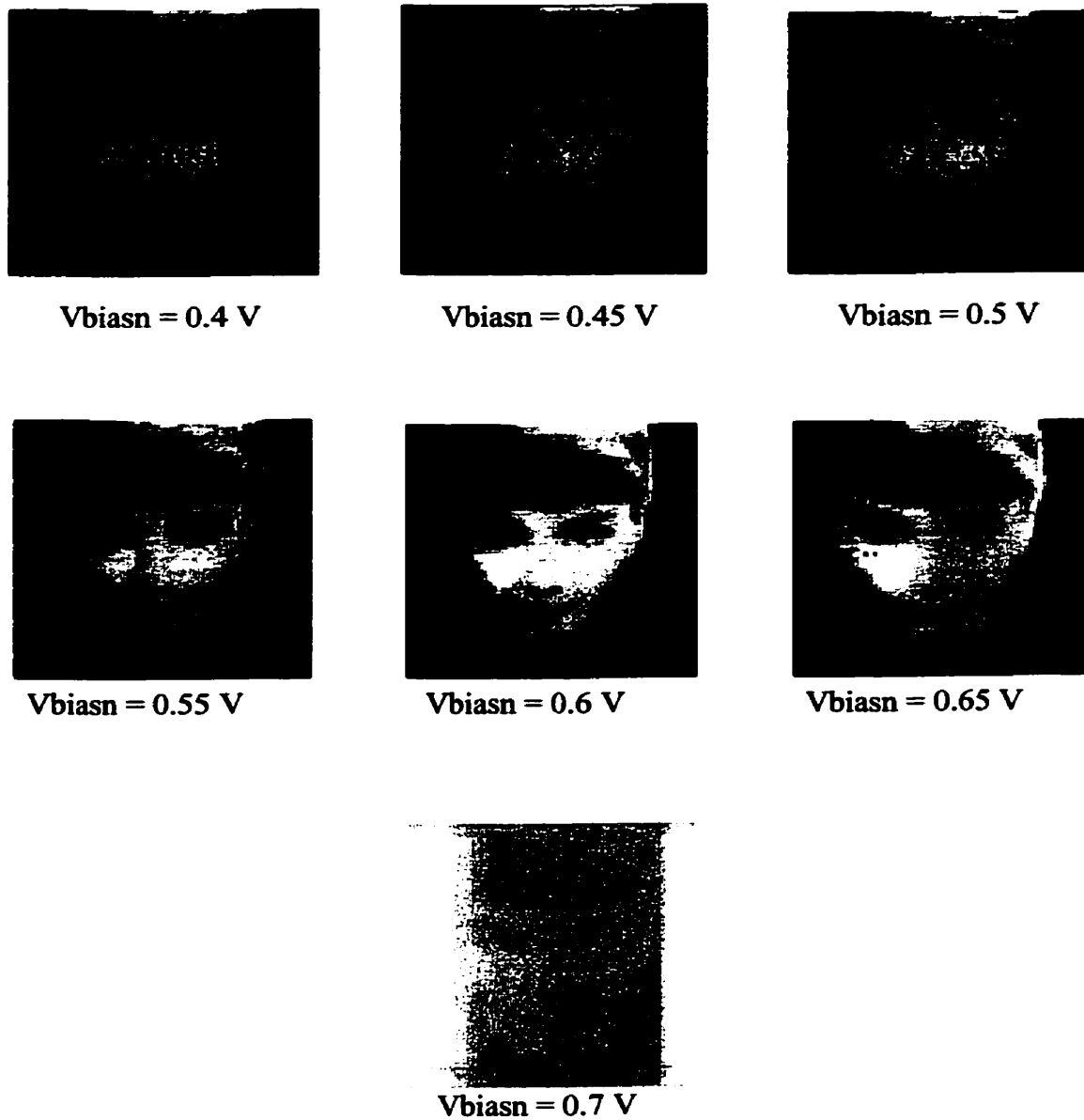


Figure 3.14. Images with different V_{biasn} : They should be same unless it reaches its saturation level. These sample images are captured under same setups of $V_{biasp} = 2.55 \text{ V}$ and Sampling rate = 20 KHz, but at different V_{biasn} .

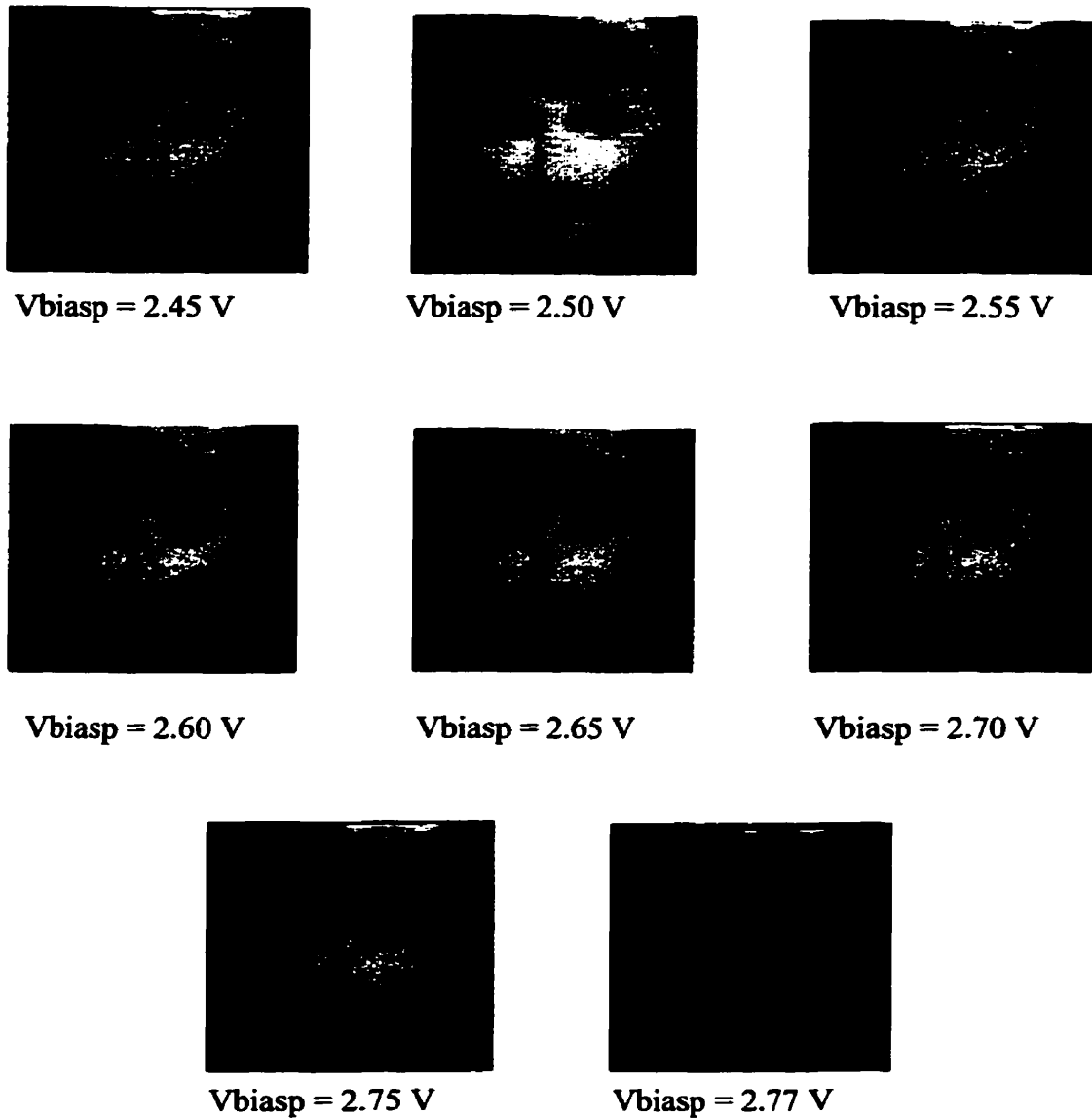


Figure 3.15. Images with different S/H Vbiasp: They should be same unless it reaches its saturation level. These sample images are captured under same setups of Vbiasn = 0.5 V and Sampling rate = 20 KHz, but at different Vbiasp.

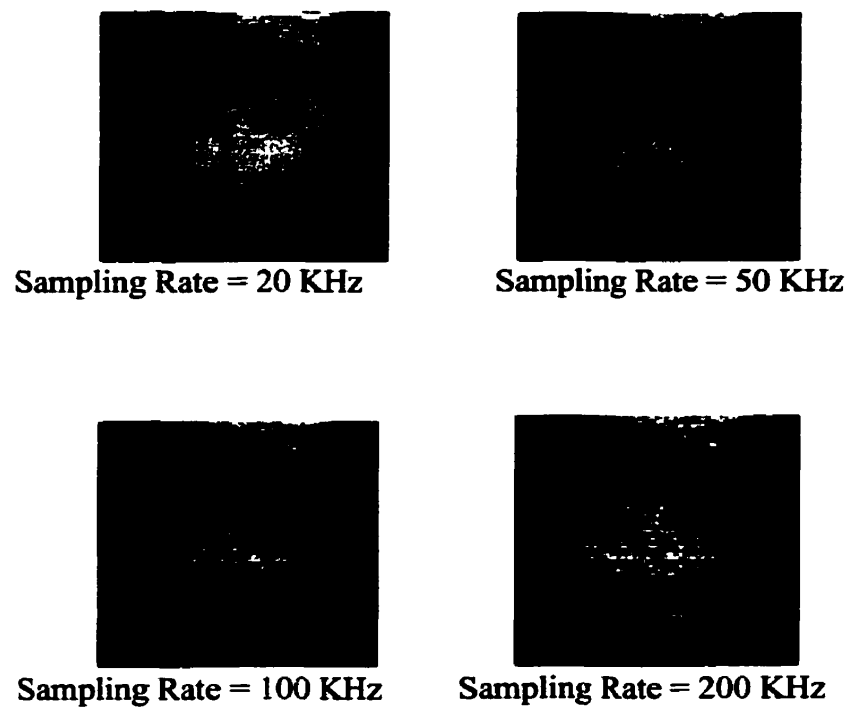


Figure 3.16. Images with different sampling rate. These sample images are captured under same setups of $V_{biasp} = 2.55 V$ and $V_{biasn} = 0.5 V$, but at different sampling rate or data rate.



Figure 3.17. Testing setup for three MOSAIC chips' connection. The three independent cameras are connected together through a common bus line.

For demonstrating multiple image capture, three independent cameras are connected together through a common bus line, as illustrated in Figure 3.17. Each camera captures its input image and transfers its image signals to the controller in sequence through the daisy chain. After the signals are transmitted to the controller, the frame grabber and display module are programmed to capture and display three different images into one panorama, as shown in Figure 3.18. The integrated bus interface operates successfully for multiple images in real time mode.

As the number of chips in the system increases (up to four cameras in our experiments), power consumption and time delay are carefully measured. The power is measured in the dark, rather than under illumination because the power consumption can be affected by the images that the chips capture. For the single chip operation, the chip consumes 1 mW nominally. Interestingly, as the number of chips in the system increases, the power consumption does not increment by the power consumed by the single chip. Rather, for each additional chip, the power increases by about 20% of the single chip power, as shown in Figure 3.19(a). When a chip does not have the bus grant signal, its bus interface disables the shift registers, preventing current from flowing through the PMOS transistors in the S/H. Since a large portion of the power is consumed by the PMOS transistors in the S/H, about 70~80% of the total power [20], the disabling mechanism saves power of the system as a power control method. The overall power consumption of the system can be saved by such power control methods, especially when a large number of chips are connected.

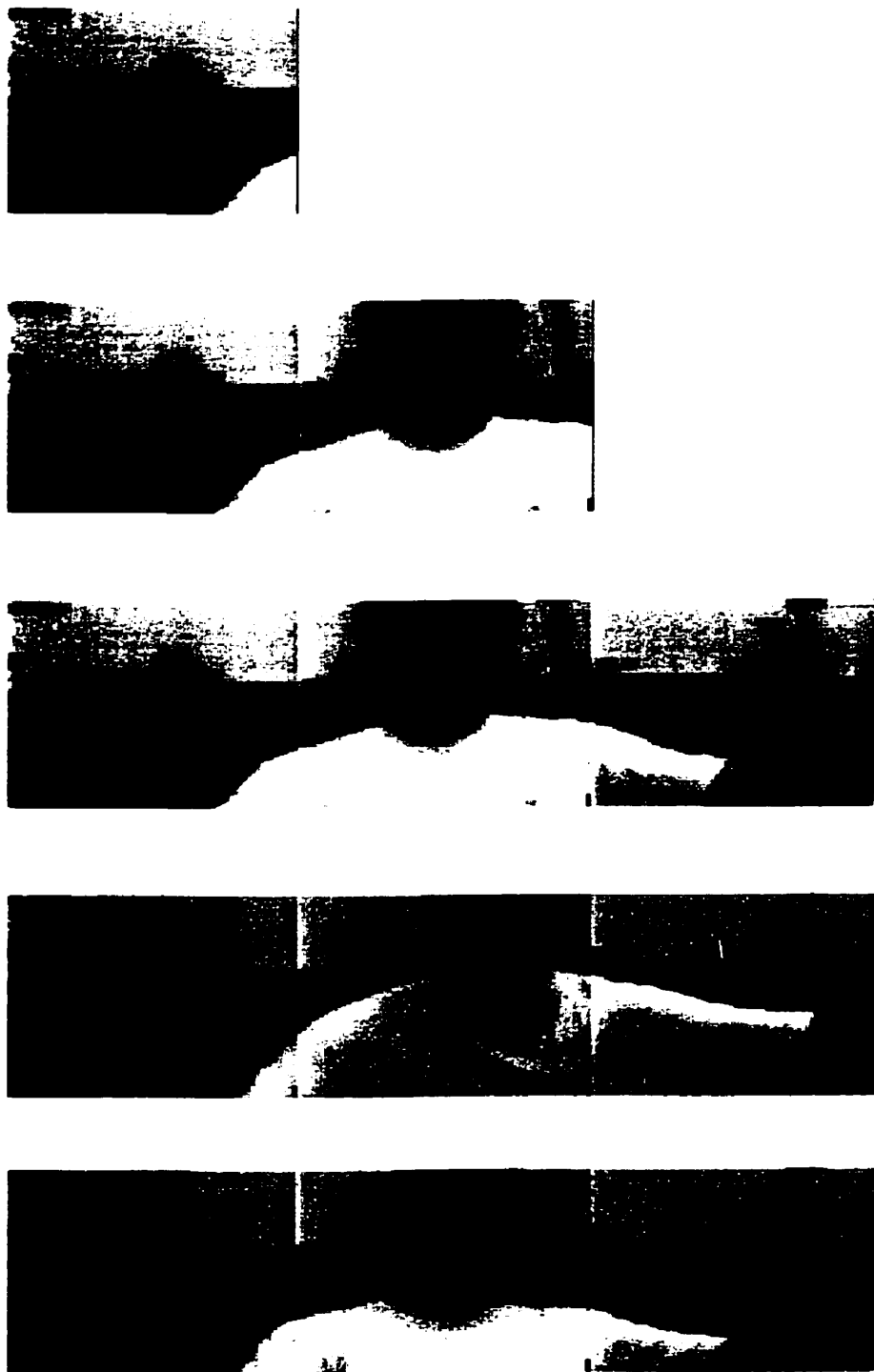


Figure 3.18. Panorama images captured by the MOSAIC system. Three single chip cameras of the mosaic imager are linked together through a common bus line. This is a still image, a part of video images captured in real time mode. These sensors do not include pattern noise correction.

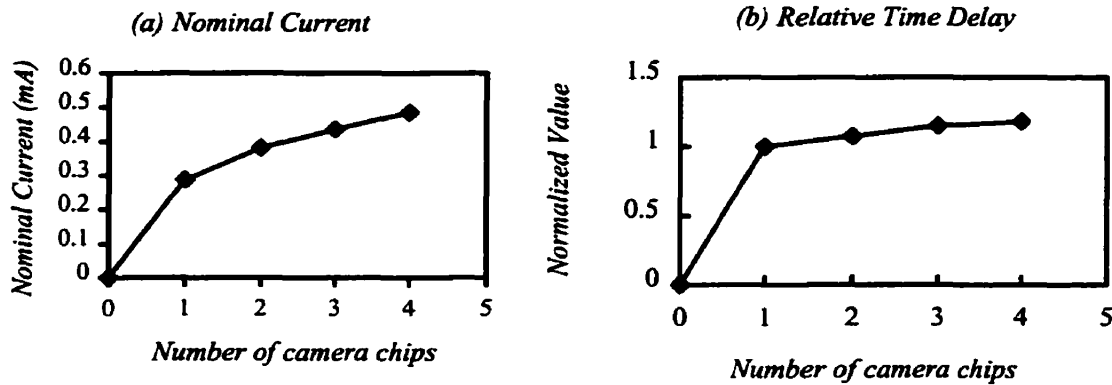


Figure 3.19. Test results of mosaic imager. As the number of chips increases in the mosaic system, the power and time delay are measured.

In order to measure the relative time delays with different numbers of chips, the minimum charging/discharging time of a fixed pixel is consistently measured with the same background image. As the number of chips on the bus line increases, the minimum time delay of charging/discharging also increases, as shown in Figure 3.19 (b). Similar to the power consumption, the RC time delays for the additional chips do not increase by the time delay of the single chip system. When the time delay of the single chip system is normalized, an additional chip to the system experiences only about 7.5% increment. Since the loading of the bus line is mainly caused by the bus line, probe contacts and external connections, the extra loading of the additional chips is relatively small. However, it is evident that as the number of chips to the MOSAIC system increases, the output loading to the system increases, thus slowing down the image transfer speed. Especially for a large field of view, when a large number of chips are connected, the inevitable heavy loading to the MOSAIC imager will be a primary implementation issue.

In order to enhance the frame update rate in the MOSAIC system, six different methods can be proposed. Firstly, multiple output channels will increase the frame update rate. Instead of one output channel, the output data can be transmitted through several different channels in parallel. One shift register (or decoder) can be placed per output channel, dividing the array into blocks by column. Secondly, large drivers increase the frame update rate. The output driving power in our CMOS photodiode array is generated from the source follower in the S/H, where a PMOS source follower is used. The larger the transistor size of the driver is, the more current (driving power) the driver has. Thirdly, a shorter RC charging/discharging

range could be used for output transmission similar to that used in random access memory. Since the voltage swing is small, the time for charging/discharging is reduced, allowing a faster update rate. However, such a small voltage swing potentially suffers from high noise, especially from off-chip connections. Therefore, digital signal transmission is proposed for noise immunity. Even with small voltage swing, the digital transmission of output is relatively immune to noise compared to its analog counterpart. The digital transmission does not necessarily increase the frame update rate, instead it protects the output transmission from noise sources. In addition, efficient bus arbitration algorithm (bus interface that arbitrates the bus ownership so that at a given time, only one module which is connected to the bus has the control of the bus) can enhance the frame update rate. There are many different bus arbitration methods, each suitable for particular applications and systems, so choosing a proper bus arbitration can increase the speed. Lastly, data reduction strategies are of great importance for high speed. Since large volumes of output data slow down the frame rate, a reduction of the data transmitted from on-chip to off-chip will increase the frame speed. The data or image could be compressed after the acquisition of the image. Alternatively, objects or events of interest in the image can be extracted and encoded. Either data compression or data extraction will reduce the amount of output data, thus increasing frame update rate.

3.5. Conclusions for MOSAIC: Single Chip Camera Modules

The integrated bus interface module increases the performance of the bus connections by providing proper structure and arbitration methods. In this thesis, the integrated bus interface demonstrates its effectiveness in terms of fabrication cost and flexibility of operation. Since a common bus line is used for an image transfer to the controller, the number of connection lines is reduced. Also the bus arbitration is managed in each camera, so the system is very flexible for additional cameras. Moreover, by an intelligent power control method of the system, low power operation can be achieved. However, even with efficient on-chip bus interface, large data flow and slow frame update rates are still potential design issues for systems with large numbers of camera modules, due to the output loading to the bus line. Therefore, it is concluded that the implementation of the high frame update rate is necessary for further implementations of MOSAIC system. A smart sensor with on-chip processing is of great importance as an additional technique to increase the frame rate of the MOSAIC.

Chapter IV

4. Spatial Image Processing Integrated with CMOS Image Sensor

4.1. Introduction

Solid-state image sensor technology is based on the inherent photoconversion properties of semiconductors with the advanced silicon processing technology driven by the VLSI industry to achieve high performance and reasonable cost. As mentioned previously, the focus of future CMOS image sensor technology is expected to be in two research eras: cost and performance. The performance refers to the good image quality produced by image sensors with low temporal and spatial noises, low dark current and high dynamic range. The cost rather refers to processing component integration with image sensors for automated controls and enhanced functionality. The integration of processing circuits on the same focal plane with CMOS image sensors will reduce overall fabrication cost, mainly saving wafer area for pads and power supplies. The cost can also be saved from packages, circuit boards, wire connections and assembly.

The main reason for high integration of CMOS image sensors is its compatibility of processes between circuits and image sensors. While CMOS technology requires relatively thin gate oxide thickness, shallow well depths and low power supplies, CCD requires relatively thick gate oxide thickness, deep well, deep channel depth, and high power supplies. Obviously, it is difficult to integrate the two technologies due to these significant differences in their process steps. Essentially, a full-featured combination would require almost all the

stages from both processes, which means probably over 30 masks processing steps.

There have been some efforts to combine good image quality of CCD and logic of CMOS technology. The reduced yield and increased costs has not made a combined CMOS/CCD process viable. The combined process is neither standard CMOS nor standard CCD, and so requires high development expense, and the frequent result is that neither part will work particularly well. Several processes have been reported which claim to preserve the quality of each technology [21][91][92]. However, despite the demonstrated feasibility of CMOS/CCD hybrids, the idea has not yet taken off possibly because few places have access to both sets of fabrication facilities and the design experience [1].

CMOS image sensors use the same technology of CMOS logic/memory processes, and therefore, expensive extra process steps are not needed. Also, the process for CMOS image sensors can be enhanced, mainly by increasing the depth of the epitaxial layer, which is predetermined wafer selection rather than process steps. Therefore, the fabrication of CMOS image sensors with processing circuits such as on-chip ADC, logic, memory, and even processing elements is relatively simple and cheap, without much loss in optical performance.

In this thesis, the interest in integrating image processing with a CMOS image sensor was initiated by the MOSAIC system for large field of view, particularly with reference to data reduction mechanisms. Therefore, the remainder of the thesis is based on system-level architecture and design methodology issues, trying to answer the following questions:

- Why we want to integrate vision algorithms (image processing algorithms) with image sensors (CMOS image sensors in this thesis)?
- What algorithms and processing components should we put with the sensors?
- How we will integrate these processing algorithms?
- What structures are the best for what image processing algorithms?

The first two questions are answered in this chapter and the last two questions are answered in next few chapters, leading to the basis of the main concept of the thesis.

4.2. Smart Sensors (Vision Chips): Why Smart Sensors?

Here, we are trying to answer why we want to integrate image processing (vision) algorithms with image sensors, or to implement smart sensors. Comparisons between smart sensors and camera plus processors are investigated to determine their advantages and disadvantages.

The integration of image sensors and processing circuits on a single chip, for obtaining better performance from sensors and processors, or for making the sensing and processing system more compact, is not a new idea. There are various reports on on-chip signal processing elements with CMOS image sensors, such as correlated double sampling (CDS), delta-difference sampling (DDS), programmable amplification, multiresolution imaging, dynamic range enhancement, and on-chip clock generation. These processing circuits are signal processing to improve the performance of the CMOS image sensors, but not to increase functionality of the imager chip.

A smart sensor is well defined in "*Vision Chips*" by Moini [15]. Moini quotes that "the smart sensors refer to those devices in which the sensors and circuits co-exist, and their relationship with each other and with higher-level processing layers goes beyond the meaning of transmission. Smart sensors are information sensors, not transducers and signal processing elements". In this thesis, the meaning of smart sensor is further narrowed down to the devices in which image sensors and image processing circuits (beyond signal processing) co-exist, and they interact with each other in order to increase functionality of the imager chip.

Traditional photodetectors often require further signal and image processing after the image acquisition to increase quality of imaging in terms of noise, resolution and speed. In contrast, in smart sensors the main interest is the functionality of processing or quality of processing. The important qualities of processing in the smart sensors are the contents of outputs from the smart sensors, algorithms integrated with the sensors, and applications the smart sensors are targeted for. Sometimes, some imaging characteristics, such as resolution, frame rate and power, could be sacrificed to enhance the functionality of processing.

When compared to a vision processing system consisting of a camera and a digital/analog processor, a smart sensor provides many advantages. Although the main advantages are to reduce bandwidth and subsequent stages of computations, there are many other advantages,

well described in [15]. These are the major reasons why smart sensors are better than a combination of a camera and a processor in separate chips.

- **Processing speed:** The processing speed of smart sensors is faster than that of combination of image sensor and processor. In the image sensor and processor combination, the information transfer occurs in a series between the image sensors and the processors, while in smart sensor data between different layers of processing can be processed and transferred in parallel.
- **Single chip integration:** A single chip implementation of smart sensors contains image acquisition, low and high-level analog/digital image processing circuits on a same focal plane. For example, a tiny sized chip can do the equivalent work as a camera-processor system.
- **Adaptation:** In many smart sensors, photocircuits can be located up front with the photodetectors for local and global adaptation capabilities that further enhance their dynamic range. Conventional cameras at best have global automatic gain control with offset at the end of the output data channel in the chip.
- **Power dissipation:** Smart sensors often use analog circuits that operate in sub-threshold region. In addition, a large portion of the total power spent in image sensors is due to output drivers to heavy output loadings of bonding wires, pads at high frequency and off-chip interconnections. By placing image sensors and processors without a separate packaging, the design of the large drivers is avoidable, which reduce the power consumption in operation.
- **Size and Cost:** Single chip implementation of image sensors and a processor can reduce a system size dramatically, mainly saving wafer area for pads and power supplies. The compact size of the chip is directly related to the fabrication cost. Therefore, the integration of processing circuits on the same focal plane with the image sensors will reduce overall fabrication cost.

Although designing single-chip smart sensors is an attractive idea, it faces several limitations and disadvantages:

- **Processing reliability:** Processing circuits of smart sensors often use unconventional analog circuits which are not well characterized and understood in many technologies. Therefore, the processing circuits have low precision on their operation, which is affected by many uncontrollable factors. As a result, if the smart sensor does not account for these inaccuracies, the processing reliability is severely affected.
- **Custom designs:** Unconventional analog circuits are often used in implementation of smart sensors. Therefore, circuits from design libraries cannot be used, but many new analog circuits have to be developed from a scratch. Therefore, smart sensors are always full custom designed, which is known to be time consuming and error-prone.
- **Programmability:** Many smart sensors are not general-purpose devices, and are typically not programmable to perform different vision tasks. They are rather application specific designs. This lack of programmability is undesirable especially during the development of a vision system when various simulations are required. However, it is not necessarily a serious drawback of smart sensors because many applications of the smart sensors are for particular tasks with limited programmability.

Even with these disadvantages of the integration, smart sensors are still attractive mainly because of its effective cost, size and speed with various on-chip functionalities. Simply there are the benefits when a camera and a computer system are converted into a thumbnail sized camera chip.

4.3. On-chip Early Image Processing: What on Smart Sensors?

The basis of the smart sensor concept is that analog VLSI systems with low precision are sufficient for implementing many low-level vision (image processing) algorithms, often for application-specific tasks. Conventionally, smart sensors are not general-purpose devices, but everything in a smart sensor is specifically designed for the application targeted. Yet, in this thesis, we do not wish to limit implementations to application-specific tasks, but to allow for general-purpose applications such as DSP-like image processors with programmability. The idea is based on the fact that some of early level image processing in the general-purpose chips are commonly shared with many image processors, which do not require programmability on their operation. As shown in Figure 4.1, human eyes, not associated with the brain, perform basic image operation in a human such as image filtering, brightness

High Level Processing:
Object Recognition
Pattern Segmentation
Object Interpretation
Image Representation

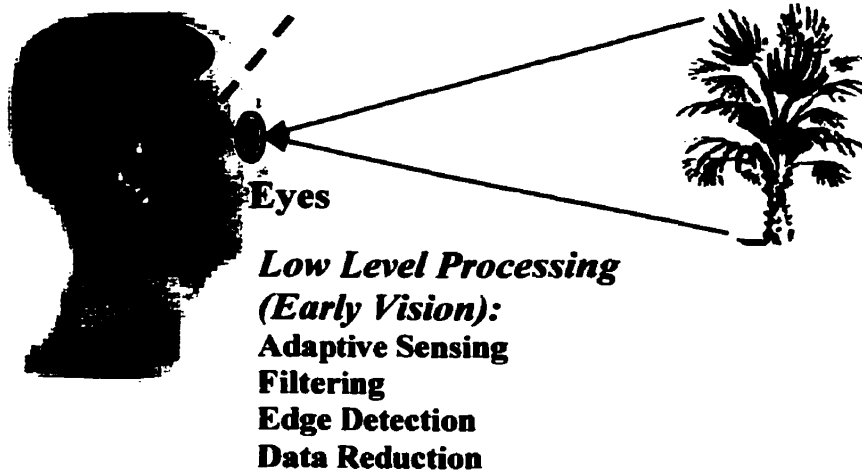


Figure 4.1. Optical image system in human: low level processing such as brightness adaptation and image filtering can be done at eye level, without much interaction with brain.

adaptation, edge extraction and motion detection [22]. These early level image processing algorithms, from the point of views of on-chip implementation, are rather pre-determined and fixed, where their low precision can be compensated later by back-end processing. Here, we will investigate what early image processing algorithms can be integrated on smart sensors as a part of early vision sequences and we will discuss their merits and the issues that designers should consider in advance.

General image processing consists of several image analysis processing steps as shown in Figure 4.2: image acquisition, preprocessing, segmentation, representation and description, and recognition and interpretation. The order of this image analysis can vary for different applications, and stages of the processes can be omitted. In image processing, the image acquisition is used to capture raw images from its input scene, through the use of video camera, scanners and, in the case of smart sensors, the solid-state arrays.

Preprocessing stage is used to perform initial processing that makes the primary task of the image analysis easier. Preprocessing is a stage where the requirements are typically obvious and straightforward, such as removing artifacts from images or eliminating image

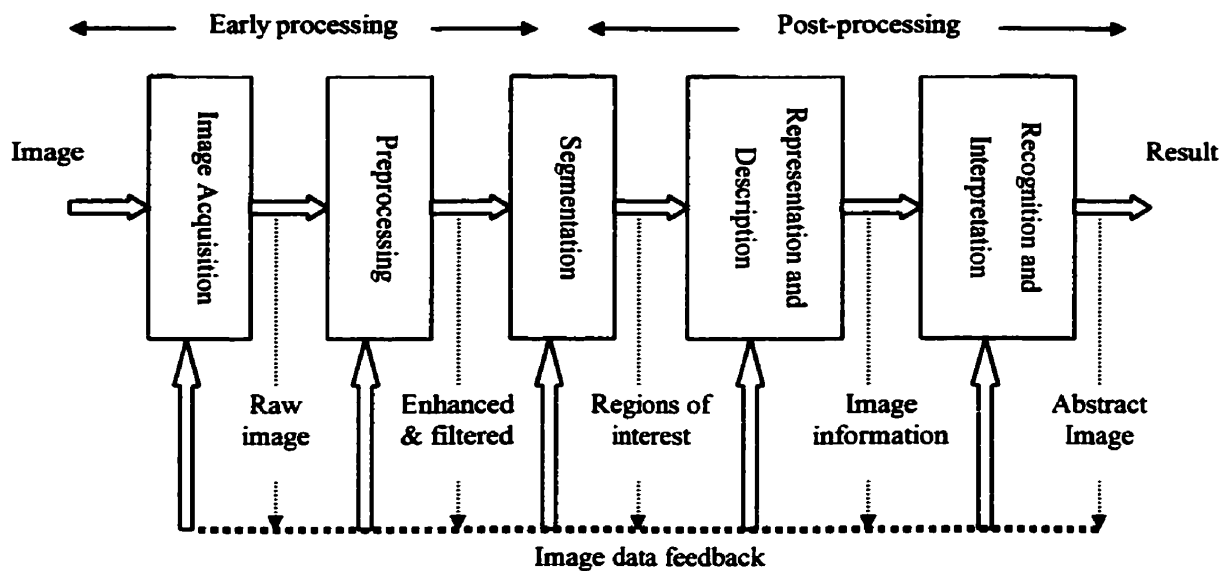


Figure 4.2. General machine vision/image processing operational stages of image analysis.

information unnecessary for the application. It includes basic algebraic operation such as image averaging and subtraction, feature enhancements, contrast stretching, bit slicing, and data reduction of image information. It is mainly subdivided into three different operations: image enhancement, image restoration and image compression. The image enhancement processes an image so that the result is more suitable for a specific application. For example, image smoothing and sharpening filters improve image quality of input raw images. Image restoration is a process that attempts to reconstruct or recover a degraded image with a prior knowledge of the degradation phenomenon. Image restoration is quite similar to image enhancement, but one big difference is the prior knowledge of the degradation. Due to the prior knowledge of the degradation, the recovery of damaged images is relatively easier. Lastly, image compression is another form of data reduction between raw input images and encoded output images. Image compression is a highly recommended preprocessing operation, particularly for high volume communications like multimedia applications.

At the third stage of the image processing, image segmentation is important in many computer vision and image processing applications. The goal of image segmentation is to find regions that represent objects or meaningful parts of objects. The segmentation subdivides an image into its constituent parts or objects. It should stop when the objects of

interest in an application have been isolated [74]. Image segmentation generally follows two methods of detection: detection of discontinuity and detection of similarity. In the first category, the approach is to partition an image by abrupt changes in gray level. The principal areas of interest within this category are the detection of isolated points and the detection of lines and edges in an image. The approaches in the second category, which is detection of similarity, are based on thresholding, region growing, and region splitting and merging.

At the next level of the processing, the resultant data of segmented pixels usually are represented and described in a form suitable for further computer processing. Representation and description is an image processing operation that follows the image segmentation. Basically, representing a region involves two choices: representation of regions in terms of its external characteristics (its boundary), and representation in terms of its internal characteristics (the pixels comprising the region). Therefore, this stage of the processing refines images or image information more adequate for high-level image processing.

At the last stage of image processing, recognition and interpretation is a process of the understanding patterns. This is a stage where understanding patterns that are related to the image processing takes a place. Therefore, it requires large computational power as well as large memory.

We have seen general process stages of image processing and image analysis, popularly used in machine vision. These stages are not necessary operation for all the image analysis. It is rather dependent on the applications that it is used for. The order of the stages can be changed and some of the stages can be omitted for particular applications. For instance, edge detection with CMOS image sensor uses images captured by CMOS image sensor and performs image segmentation on the image, skipping image enhancement or filtering.

Based on the processing stages of the image analysis, on-chip image processing with CMOS image sensors is focused on here. Ideally, on-chip image processing contains all the processing stages of image analysis. However, it is not possible or necessary to design and integrate all the processing circuits of the operation on a single chip. In order to understand clearly what image processing operation are needed and how much image processing task is necessary for the smart sensors, understanding and classification of these image analysis stages are highly recommended. After all, choosing an appropriate algorithm for less power,

less area and faster speed is essentially important for the integration of the CMOS image sensor.

Although few, if any, of the vision chips are general-purpose [93] and many vision chips are not programmable to perform different vision tasks, there are primary image processing tasks needed for many applications. For example, image processing beyond image enhancement and some of segmentations require large computational power and memory to store the data. Also they are applications oriented processing. However, image enhancement and filtering are essential for many other image processing operation. Therefore, image enhancement and filtering implementation should be included in the early level image processing commonly shared by general-purpose image processors.

In summary, on-chip image processing with CMOS image sensors is expected to follow in these two implementation directions: application specific operation, and primary tasks for general-purpose processing such as image enhancement and filtering. Image enhancement, filtering, and sometimes image segmentation, can also be applied to performance improvements of image sensors, which are commonly shared by general-purpose image processors. However, application-specific on-chip image processing is likely to be the dominant use for CMOS image sensors, because of the wide variety of applications and the large number of different design choices for the integration.

4.4. Architectures for On-chip Processing Integration: How to Implement Smart Sensors?

Now, we will investigate efficient architectures for implementing on-chip image processing with CMOS image sensors. In the next few sections, we will first look into the structures available for any signal processing integration on a single chip with image sensors. Then, we explore the nature of image processing algorithms in terms of image signals, processing domain and operational region.

We have seen vision algorithms of on-chip image processing with CMOS image sensors such as image enhancement, segmentation, feature extraction and pattern classification. These algorithms are frequently used in software-based operation, where structural implementation in hardware is not considered. Here, the main research interest focuses on how to integrate image processing (vision) algorithms with CIS or how to implement smart sensors in

hardware, in terms of its system-level architectures and design methodologies. Here, we will first look at previous designs and implementations, focusing on their design structures and methodologies.

4.4.1. Previous Work

There have been many reports involving the sensing and image processing on a single silicon chip, such as smoothing, edge detection, stereo processing, contrast enhancement, motion detection, video compression, discrete cosine transform and neural networks. These works are great efforts and fine works, some of which include revolutionary ideas. Because these works are application-specific designs, the architectural and circuit level designs are often application oriented, and they do not have general applicability.

Some researchers report papers on the implementation of image processing and image sensors. The first successful attempt to perform a low-level image algorithm, convolution by Gaussian filter, on a chip was carried out at Lincoln Laboratory in 1984, based on the control of the charge transferring mechanism [24]. Soon after this, updated and more powerful versions of this algorithm and circuit were presented [25] [26]. After the initial attempts, the detailed design and implementation of a CCD-based image processor, performing two-dimensional filtering operation with programmable 8-bit digital spatial filters, occurred [27]. This system represents a hybrid analog-digital architecture. Derived from the original implementations, a more effective parallel-pipelined architecture of on-chip processing is described in 1991 [28]. It was implemented for an edge detection algorithm and a boundary-preserving image filter. A radial geometry, called log retina, was introduced in early 1980s. This retina, based on the logarithmic mapping between the retina and cortex in mammals, consists of concentric circles with each circle having image sensors, with the pixel size of the imager increasing linearly with eccentricity. The central part of the imager has a constant resolution. Such imager architecture has a number of advantages, such as emphasizing the central part of the image and certain invariance for pattern recognition and motion processing. Other examples of image processing demonstrated in CMOS image sensors include motion detection, spatial local filters, multiresolution, video compression, and neuronMOSFETs.

These on-chip image processing implementations are systematically designed for specific applications, but do not provide an overview description of their limitations and

implementation boundaries. The overview article by Fossum in 1989 [29] provides a comprehensive treatment of solid-state imagers using analog CCD circuitry. Low, medium, and high density detector arrays are discussed in terms of their implementation architectures, and a pipeline-vector-pixel processor is described. Also, the potential of on-chip read/write analog frame memory for image transformation and frame-to-frame processing is addressed. However, the architectural implementation by circuit density (number of transistors per unit area for a processing element) is not sufficient to provide the detailed and general partition because the circuit density is not the only design specification the designers should account for.

Here, in this thesis, more generalized partitions for architectural implementation of on-chip image processing with CMOS image sensors are proposed. The partition includes not only the circuit density, but also the nature of image processing algorithms and the applications for its focal plane integration with the sensors. We will look into the existing architectures of focal plane integrations and its feasibility with CMOS image sensors. We will also explore the nature of image processing algorithms, including operation of the algorithms and their feasibility with imager focal plane implementation.

4.4.2. Types of Hardware Implementation

General architectures for signal processing, not necessarily image processing, on a single chip with the image sensors are examined. It should be noted that this is a general implementation structure of any signal processing for image sensors, such as on-chip ADC, CDS and amplification. The basic components of CMOS imager array, such as photodiodes, shift registers, S/H and output buffers, are assumed to be independent of implementation structures. Architectures of focal plane integration are mainly divided into four different processing structures: pixel, column, chip and memory frame processing. Location of the signal-processing unit, as known as a Processing Element (PE), becomes the dividing factor of these implementation structures, as shown in Figure 4.3.

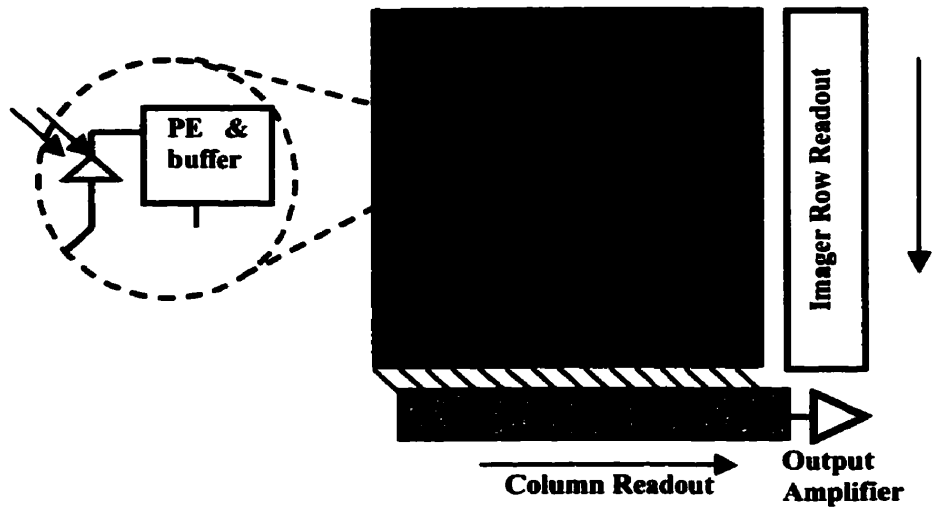
The pixel processing consists of one processing element (PE) per image sensor pixel, shown in Figure 4.3 (a). Each pixel typically consists of a photodetector, an active buffer and a signal-processing element. The pixel-level processing promises many significant advantages, including high SNR, low power, as well as the ability to adapt image capture and processing

to different environments with processing during light integration. However, the popular use of the design has been blocked by the severe limitations on pixel size, low fill factor and restricted number of transistors in PE.

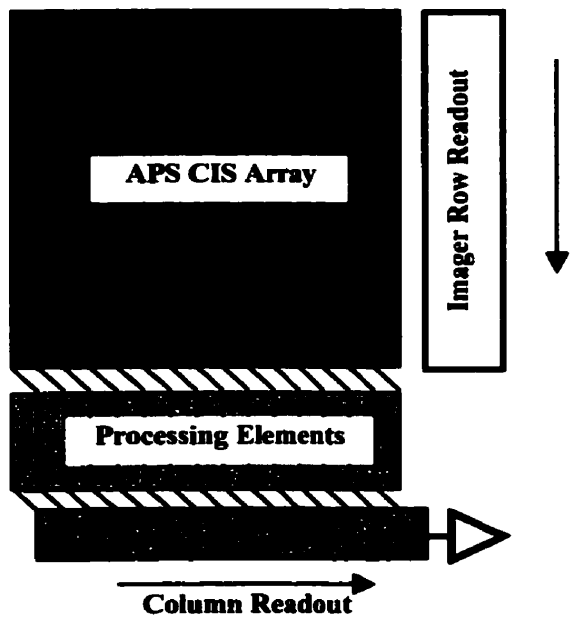
In the column-level processing, shown in Figure 4.3 (b), a PE is located at every column of the imager array. Since images of the array are read row by row, the whole row is dumped into S/H concurrently and then transferred to the output in series pixel by pixel. With this typical readout mechanism of CMOS image sensor array, the column processing offers advantages of parallel processing that permits low frequency processing and thus low power consumption. Compared to pixel processing, the pixel suffers less from low fill factor because the PE is taken out to the column, which increases the photosensitivity of the sensor. Although there is restriction on implementation area, particularly column width, the implementation is relatively flexible because of the freedom in vertical direction of the columns. Still, due to the narrow column width, particularly as the pixel size shrinks, designers cannot have full flexibility of processing circuits area.

The chip-level processing is one of the obvious integration methods due to its conceptual simplicity and flexibility of design area. Each PE is located at the serial output channel at the end of the chip, shown in Figure 4.3 (c). There are fewer restrictions on the implementation area of the PE, leading to a high fill factor of the pixel and a more flexible design. However, the bottleneck of the processing speed of the chip becomes the operational speed of the PE, and therefore, a fast PE is essentially required. The fast speed of the PE results potentially in high complexity of design and the high power consumption of the chip. Therefore, many designers try to avoid using this structure unless the chip requires high complexity of design.

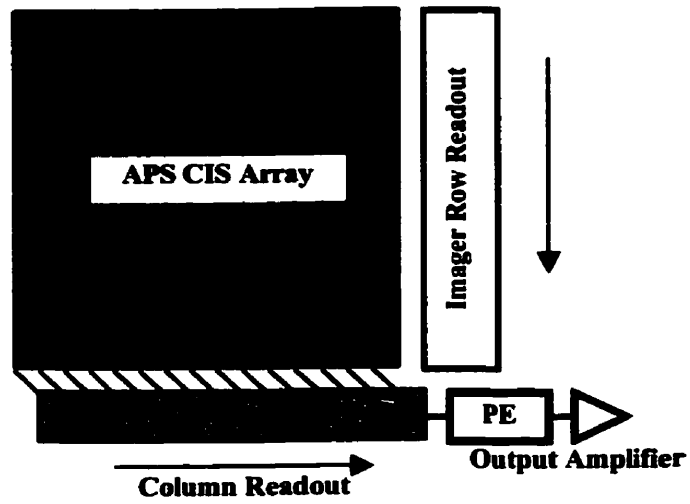
Another structure of the implementation is frame memory processing. As shown in Figure 4.3 (d), a memory array with the same number of elements as the sensor is located below the imager array. Typically, the image memory is analog frame memory that requires less complexity of design, area, and processing time [30]. However, this structure consumes a large area, large power and high fabrication cost. In addition, the processed images have latency of a frame to the output. Structures other than frame memory face difficulty in implementing temporal storage. The frame memory is the most adequate structure that permits iterative operation and frame operation, critical for some image processing



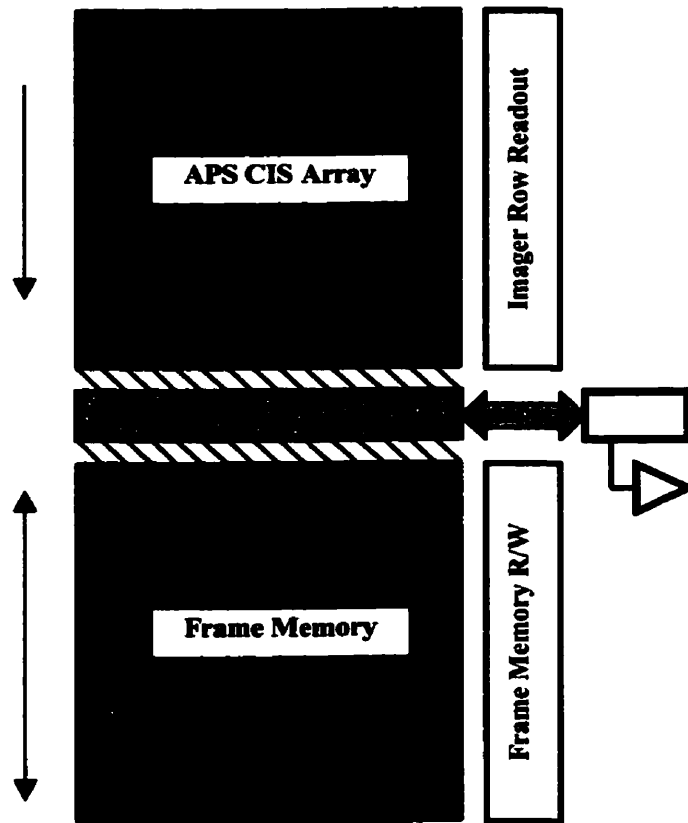
(a) Pixel Processing



(b) Column Processing



(c) Chip Processing



(d) Frame Memory Processing

Figure 4.3. Structures of focal plane implementations with image sensors: pixel, column, chip and frame memory processing.

	Advantages	Disadvantages
Pixel Processing	<ul style="list-style-type: none"> Parallel processing Processing during integration High SNR Slow processing, thus low power Low processing frequency Easy implementation of global and local adaptation Minimized parasitic effects 	<ul style="list-style-type: none"> Low fill factor Restricted size of PE Limited number of transistors in PE Limited programmability and precision Poor uniformity of PE Dark current and cross-talk
Column Processing	<ul style="list-style-type: none"> Flexible implementation in vertical directions Semi-parallel processing Low processing frequency thus low power High fill factor Less non-uniformity than pixel level implementation 	<ul style="list-style-type: none"> Restricted area of column width Limited size of mask (3x3) Higher mismatch than chip structure Higher power than pixel structure Low uniformity of PE's in columns
Chip Processing	<ul style="list-style-type: none"> Small chip area No limitations on PE design area High fill factor High uniformity 	<ul style="list-style-type: none"> Fast PE (High speed) is required High complexity of PE High power No parallel processing Chip speed dependency
Frame Memory Processing	<ul style="list-style-type: none"> Flexible operation High fill factor Image storage 	<ul style="list-style-type: none"> Large chip area Latency of a frame Medium power High fabrication cost Signal degradation in memory

Table 3. General descriptions and comparisons on hardware implementation structures, with their advantages and disadvantages.

algorithms. As a summary, Table 3 illustrates the general descriptions and comparisons of the hardware on-chip implementations with their advantages and disadvantages.

4.4.3. Design Issues of Hardware Implementation

A particular implementation structure cannot be optimal for every implementation, but instead will be application-dependent, where there is one optimal structure for a given application and specification. Here, we suggest specifications and design issues that should be accounted for when we approach a decision of on-chip hardware implementation structure for a given image processing application. These design issues include fill factor, processing time, power, design area, speed, uniformity, dark current and cross-talk.

- Fill Factor:** Since, in the column, chip and frame memory level structures, processing elements are separated from pixels in the array, circuit density is not a limiting factor. However, circuit density plays an important role in pixel level structures because it is inversely proportional to the fill factor that is closely related to the photosensitivity of the image sensors. Therefore, it is important to choose a simple processing element with reasonable precision in the pixel processing structures. However, as technology scales down, the number of transistors that can be implemented in a pixel increases rapidly, according to the estimation of Figure 4.4.

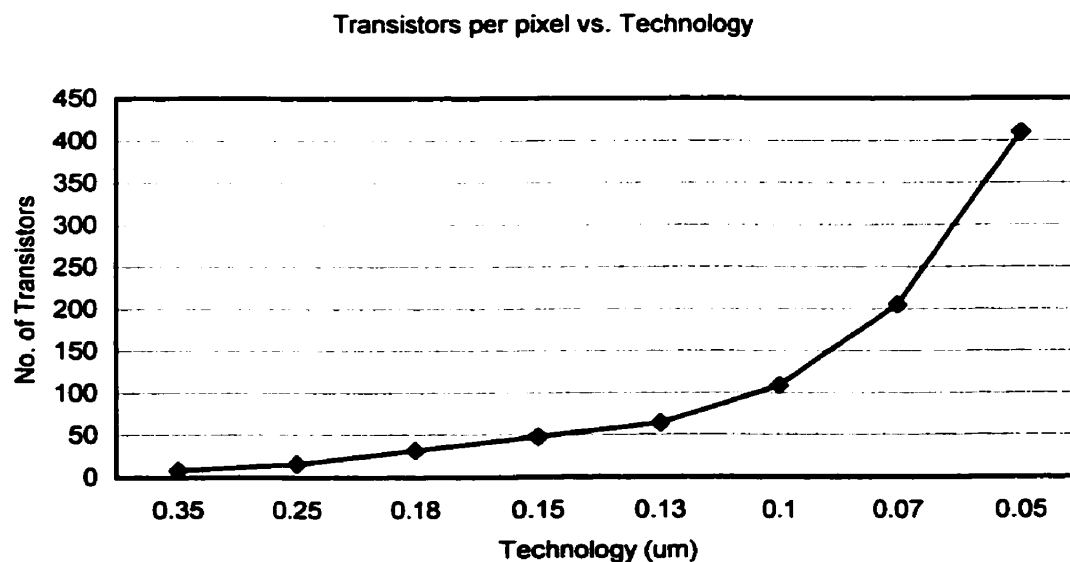


Figure 4.4. Number of transistors per pixel as a function of process technology. These estimates are based on [33]. This figure plots the estimated number of transistors per pixel with minimum transistor size (typically for digital) as technology scales, assuming a $5 \mu\text{m}$ pixel with a constant fill factor of 30%.

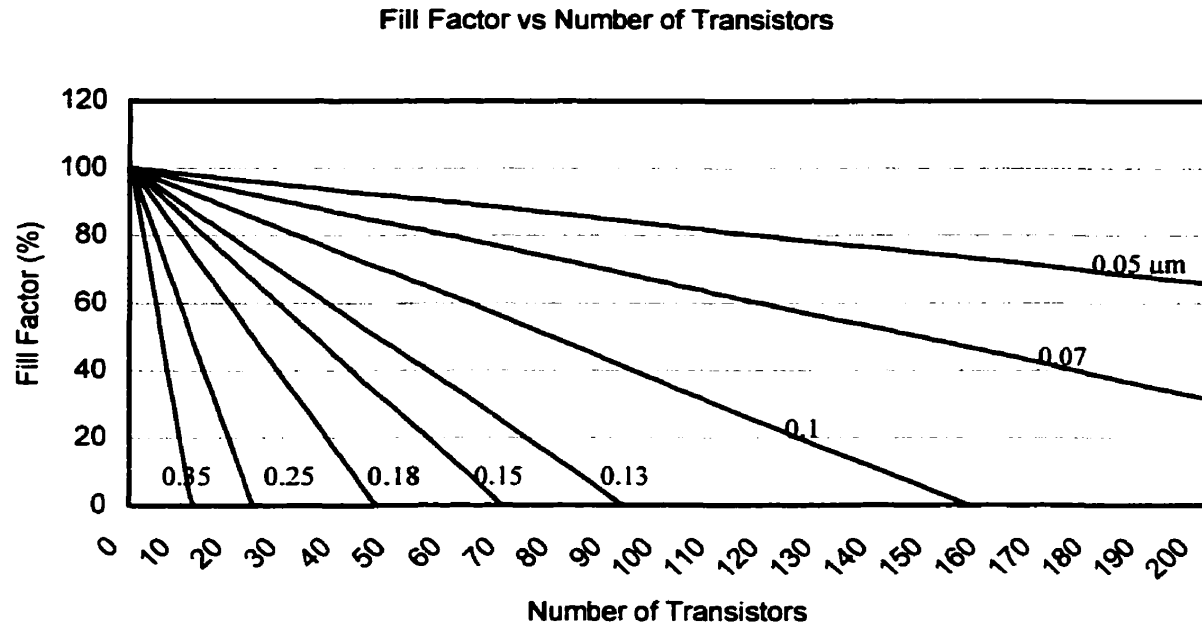


Figure 4.5. Fill factor for different number of transistors in a pixel with different process technologies. The plot is estimated from Figure 4.4.

Figure 4.5 shows the relation between the fill factor and the number of transistors in a pixel, which predicts the number of transistors with a reasonable fill factor for a given process technology. As technology scales down, there is more space available for processing circuitry in a pixel, which encourages the pixel level implementation.

- Processing Time:** Each implementation structure has a different processing time requirement for processing element, from integration time to data sampling rate. The processing time is directly related to the power consumption of the components and typically associated with the design complexity. As longer processing time is allowed for a processing element, the complexity of the element decreases because the circuit has looser speed requirement. When $M \times N$ array is operating at S frames/second, each structure has different maximum processing time allowed. With chip level structures, the processing element should run at or less than the sampling (data) rate, which here is equal to $1/(S \times M \times N)$ seconds. In the column level structures, the maximum processing time is equal to $1/(S \times M)$ seconds that is N times longer than the chip level structure. Meanwhile, the pixel level structures have $1/S$ seconds of the maximum processing time. The frame memory level structures can have the same

	Max. Processing Time (second)	Power Consumption per Processing Element (W)	Total Power Consumption (W)
Chip-based	$1/(S*M*N)$ [33 η]	$\propto (S*M*N)^2 C^2$ [$9 \times 10^{14} C^2$]	$\propto 1*(S*M*N)^2 C^2$ [$9 \times 10^{14} C^2$]
Column-based	$1/(S*M)$ [33 μ]	$\propto (S*M)^2 C^2$ [$9 \times 10^8 C^2$]	$\propto N*(S*M)^2 C^2$ [$9 \times 10^{11} C^2$]
Pixel and Frame Memory	$1/S$ [33 m]	$\propto S^2 C^2$ [$9 \times 10^2 C^2$]	$\propto M*N*S^2 C^2$ [$9 \times 10^8 C^2$]

Table 4. Numerical comparisons of hardware implementation structures for $M \times N$ array with S frames/second. [] are values, based on a 1000×1000 array operating at 30 frames/second and α is assumed to be 2 for the worst case.

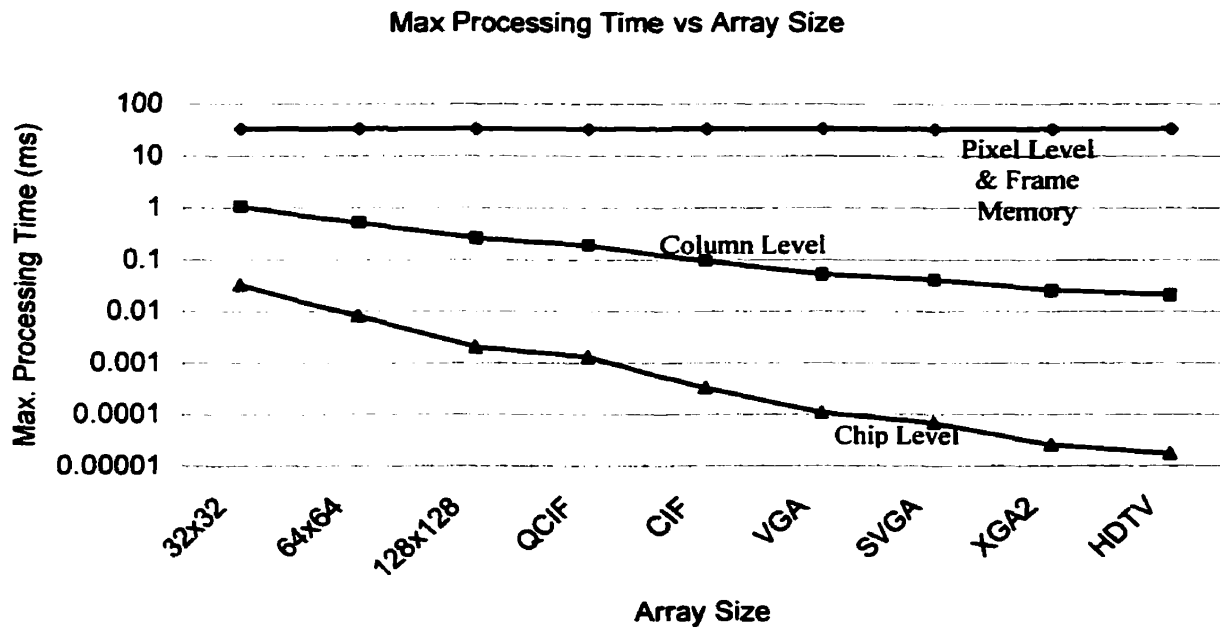


Figure 4.6. Maximum processing time available for the processing element for different sizes of array, assuming 30 frames/second frame rate for the image sensor arrays.

maximum processing time as pixel level processing, yet with a necessary latency of one image frame. An example of the comparisons for MxN array with S frames/second is shown in Table 4. Also, Figure 4.6 shows maximum processing time of a processing element for different sizes of array. As the size of the array increases, the difference in the processing time for different processing levels are clearly illustrated in the Figure 4.6; the maximum processing time for pixel level and frame memory implementation remains constant, but those for column and chip level structures decrease rapidly. No matter what size the array format has, the pixel level implementation always give a constant and relatively long processing time while the time requirements for the column and chip levels get tighter with the increase of the array size.

- **Power:** The power consumption of the processing elements is directly related to the maximum processing frequency. Unlike their digital cousins where typical power consumption is linearly proportional to its operating frequency, analog circuits follow:

$$\text{Power} \propto (\text{Capacitance} \cdot \text{frequency})^\alpha$$

Where α is around 1.5 ~ 2

With chip level structures, the power consumption for each processing element is proportional to $(C \cdot S \cdot M \cdot N)^\alpha$. With column level structures, it is proportional to $(C \cdot S \cdot M)^\alpha$. $(C \cdot S)^\alpha$ is for pixel level and the frame memory level structures. With counts of the number of the processing elements in the chip, the total power consumption will be a product of the power at each element and the total number of the elements in the chip. Therefore, the total power consumption of the pixel level and the frame memory structure is proportional to $(C \cdot S)^\alpha \cdot M \cdot N$. It should be noted that the calculation is based only on the processing element, not including image acquisition. Typically the power consumption of image acquisition is proportional to the product of $(\text{number of pixels})^\alpha$ and $(\text{number of columns})^\alpha$, as shown in Figure 4.7. Therefore, as the array size increases the total power consumption of the chip increases drastically due to the processing elements and image acquisition. The power of the column level processing structure is $(C \cdot S \cdot M)^\alpha \cdot N$. The chip level

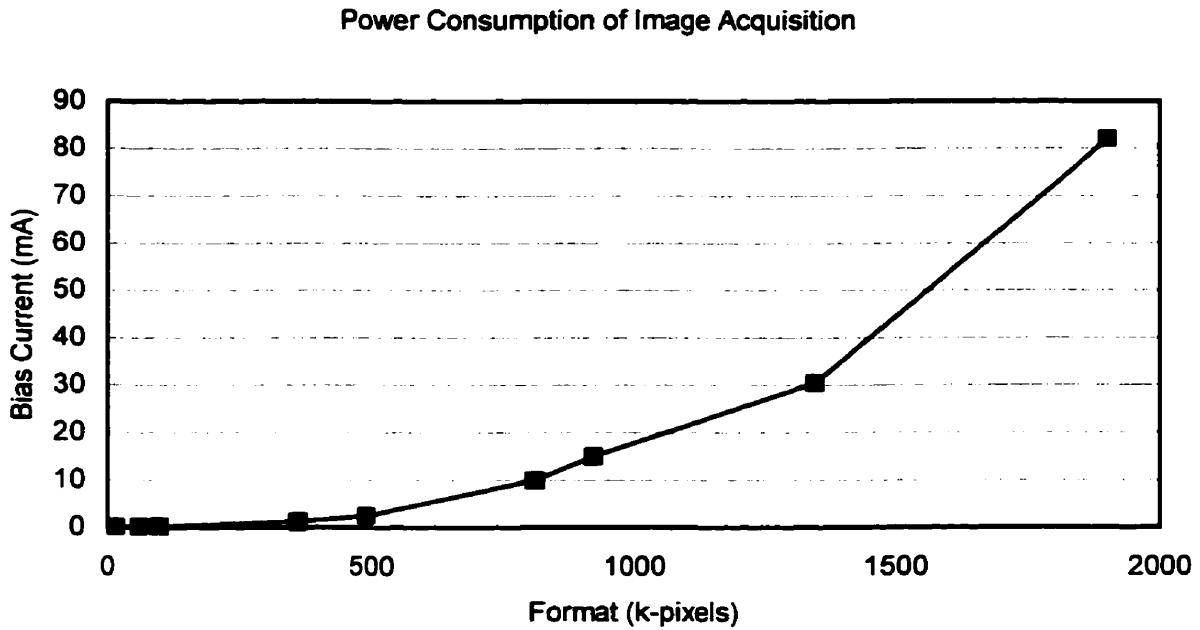


Figure 4.7. Power consumption (excluding processing element) of the different array size. This power consumption is only for image acquisition.

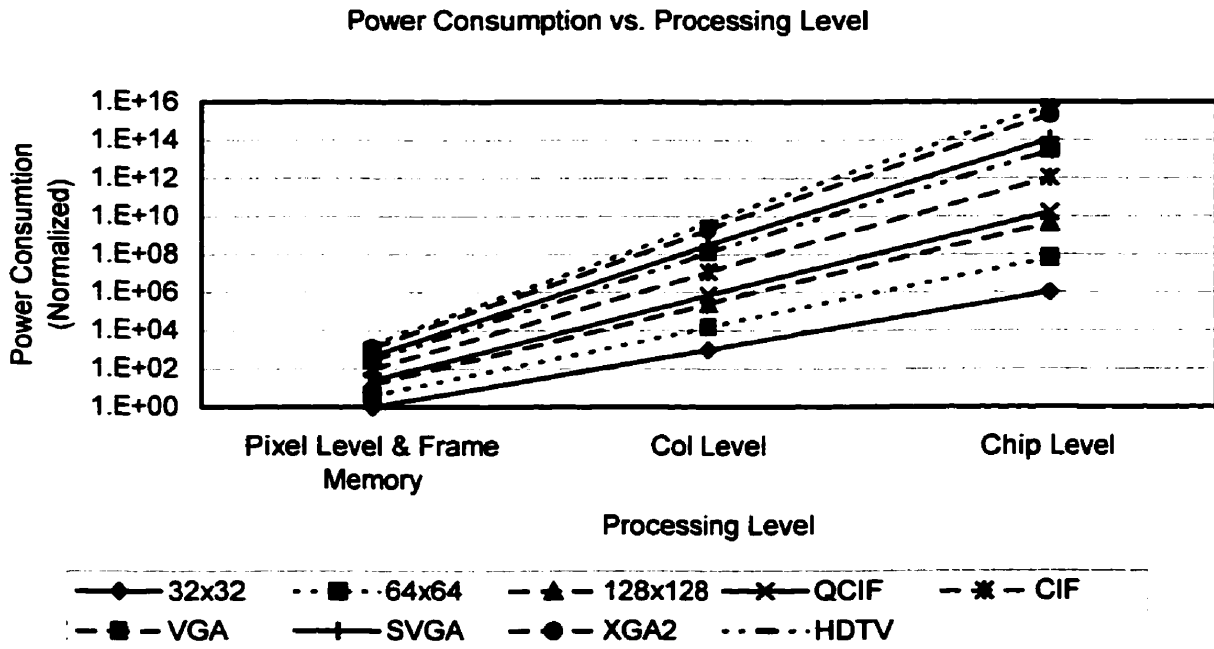


Figure 4.8. Power consumption (excluding image acquisition) of the different array size for different processing levels, assuming 30 frames/second frame rate for the image sensor arrays.

structure has the same total power as that of one processing element because it has only one processing element in the chip. Figure 4.8 shows the total power of the system (not including image acquisition) with the different sizes of the array, for the different processing levels. As the size of the array increases, the total power consumption increases drastically because of the non-linear relationship with the array size. It is clear that the pixel and column level implementations save power consumption as the array size increases, compared to the chip level implementation.

- **Design Area:** Total design area of the chip becomes an important issue because it is closely related to the fabrication cost. The frame memory consumes the largest design area because of the separated storage for a frame of image in the chip, where the chip level structure typically consumes the least area by one relatively big and complex processing element. Below the imager array, the column level structure has the same number of long narrow processing elements per column of the array, with only a slight increase on the chip size. The pixel level structure, under the assumption that same size of processing element is used for all other structures, has the second largest area consumption following the frame memory structure. Yet, because the processing elements in the pixel level structure are relatively small due to the long processing time, and unless the element is small, the photosensitive area of the pixel is drastically reduced. The typical size of the processing element in the pixel level structure is small, resulting in relatively small increase in the chip size.
- **Speed dependency:** The speed of the imager chip is determined by the slowest component in the data path (bottleneck of the output channel). In most cases, the output amplifiers are the bottleneck in the output data path because of the heavy output loads. Because, in the pixel, column and frame memory level structures, the processing elements have relatively a longer processing time than the data output rate, the output amplifiers are more likely to be the bottleneck of the chip speed. In contrast, because the processing element in the chip level structure should have the same processing speed as the output data rate, the output amplifier might not be the bottleneck of the chip speed. Instead, the processing element becomes the bottleneck. Therefore, a design of high-speed processing elements with reasonable power consumption becomes critical in the chip level structures.

- **Uniformity:** As processing elements are spread all over the image sensor array, uniformity of the processing elements becomes important design issue, especially for pixel level and frame memory implementations. As FPN is a critical design factor for the regular image sensor arrays, the uniformity will be an important factor for smart sensors. Even for column level implementations, the uniformity cannot be neglected because of the non-uniformity through the columns. However, chip level structure will not suffer from the non-uniformity of processing elements. As technology scales down, uniformity is expected to increase due to the reduction of body effect coefficient (γ) [86].
- **Dark Current and Crosstalk:** Similar to the uniformity, dark current and crosstalk will be greater for pixel and frame memory than column and chip level implementations. However, these can be reduced by careful circuit designs such as guard ring and separate power supplies, and advanced process technology with low dark current.

4.4.4. Types of Image processing Algorithms

Conventional approaches to hardware implementation of on-chip image processing are accomplished by the density of the circuit [29]. In addition to circuit density, designers should consider the nature of the image processing (vision) algorithms for the on-chip implementations. Often, for on-chip image processing (smart sensors), the nature of the vision (image processing) algorithms is overwhelmed by the circuit density, mainly due to the reduction of fill factor and reduction of photosensitivity and resolution. However, it is sometimes necessary and reasonable to sacrifice fill factor to gain operational performance for given vision algorithms. After all, both the circuit density and the nature of the processing algorithm should be considered for integrating smart sensors. Here, we will investigate and discuss the nature of image processing (vision) algorithms which can be integrated on the smart sensors. The nature of image processing algorithms can be categorized in terms of signal type, processing domain and operational regions.

A. Signal Types: Analog vs. Digital Processing Elements

Broadly, any signals can be divided into analog and digital, including the image signals. The smart sensors focus on analog VLSI implementations even though hardware implementation

of image processing algorithms typically refers to digital implementations. It is because the on-chip analog VLSI implementation of the image processing algorithms for smart sensors has the following advantages, including:

- **No ADC (Analog-to-Digital Converter):** An obvious advantage of analog implementation is that there is no need for ADC. Without ADC, analog implementation can save area, power, and processing time.

- **Size:** Analog implementations of image processing algorithms require compact area that is a crucial design issue for smart sensors. While a simple computation of large digital bit consumes a large area for the component design, a simple analog component with compact size can typically compute the equivalent operation.

- **Speed:** By the parallel nature of analog components, the processing speed can be enhanced; parallel operations between image acquisition and processing without digital sampling and quantization.

- **Continuous mode:** The continuous operating mode of analog circuits is well suited to analog sensory data since they do not suffer from temporal aliasing problems. Since image sensors operate in analog mode, the processing components operate better in analog and become compatible with the image sensors.

- **Power:** There are two different debates on the power consumption. One debate is that because analog circuits run in sub-threshold domain, where negligible current is flowing, the power consumption of the analog circuits is minimal. In contrast, the other argues, since digital operation is based on switching (On & Off) while the analog circuits are always in "On" mode, where current is flowing all the time, digital operation consumes less power. Nevertheless, power consumption depends on the systematic operation and required processing speed, and especially the mode in which the analog circuits are running.

Although the integration of image sensing and analog processing has proven to be very attractive, it also has some limitations. These limitations of analog circuits in smart sensors are well described and well argued in [77]. The limitations are as follows:

- **Programmability (Flexibility):** The analog circuits are designed to perform very specific tasks, unlike digital computers (and DSPs) that can be programmed to perform any logical or numerical operation. On the other hand, for many applications where only specific tasks are of interest, the excessive and expensive digital computers (and DSPs) with good programmability are not needed. Even for high-level processing, a combination of a smart sensor without high programmability, and DSP is recommended because the smart sensors can reduce many stages of (time and power-consuming) computations in the algorithm processing, which would have otherwise been computed by the digital computers. Besides, digital computers and DSPs are preferable for developing and evaluating new image processing (vision) algorithms.
- **Precision:** Analog circuits often suffer from fabrication inhomogeneties, offset currents, lithographic mismatches and other factors that lower the precision. Therefore, the analog smart sensors will have lower precision than the digital cousins. Typically analog circuits have only 7 ~ 8 bits precision where digital counterparts are 12 ~ 16 bit. However, biological systems such as human vision system, only process data with at most 100 levels of gray level, which can be covered with less than 7 bits. Yet with such low accuracy, human can obtain amazing performance.

B. Operational Domain: Frequency vs. Spatial Domain

Often, image processing algorithms transfer the processing domain of the input image from spatial to frequency for easier manipulation and calculations. The foundation of frequency domain techniques is the convolution theorem. Many image processing algorithms, especially localized image processing operation, use convolution in the spatial domain, and are later transferred to multiplication in the frequency domain by Fourier transform, where the multiplication is relatively easier to manipulate and implement than convolution. Operations in the frequency domain are more effective and easier to understand. However, image processing in the frequency domain definitely requires Fourier transform elements that are typically complex circuit designs. Particularly, on-chip image processing in frequency domain should contain ADC, Fourier transform and digital processor with CMOS image sensors, resulting in a large area and high complexity of designs. It is one of the reasons why

general-purpose on-chip digital image processing chip plus image sensors rarely exist yet. Rather, the processing domain of on-chip image processing is restricted to the spatial domain because of its relative ease of implementation and no use of expensive Fourier transform. In this thesis, therefore the focus of the implementation rests on analog on-chip image processing in the spatial domain.

C. Operational Region: Point, Local and Global Operation

Now, the image processing algorithms are separated in terms of neighboring pixels' interconnectivity. The interconnectivity (regions of operation) in the spatial domain plays an important role for implementation of on-chip image processing because the connection routing to the neighboring pixels is sometimes more crucial than the circuit density of the processing element. Therefore, the implementation of smart sensors should consider the neighbors' connectivity.

The type of image processing techniques, by connectivity to the neighboring pixels, can again be separated into point operation, local operation and global operation, as shown in Figure 4.9. Point operation is an image processing method that is based only on the intensity of single pixels. It modifies the gray level of a pixel independently of the nature of its neighbors; each pixel is modified according to a particular equation that is not dependent on other pixel values. In local operation, each pixel is modified according to the values of the pixel's neighbors (typically using convolution masks). Spatial filters typically use local operation of convolution masks. Global operation is a type of image processing where all the pixel values in the image are taken into consideration for the determination of the final value.

Spatial domain processing methods include all three types, but all the frequency domain operations, by nature of the frequency (and sequence) transforms, are global operations. Of course, frequency domain operation can become local operation, based only on a local neighborhood, by performing transform on small image blocks instead of the entire image. However, this is a special case since the frequency domain operation needs Fourier transform that is already considered as global operation.

In the following chapters, the natures of the image processing algorithms are investigated in terms of their interconnectivity to neighboring pixels, and corresponding structures for implementing the algorithms are proposed. Effective architectures of on-chip image

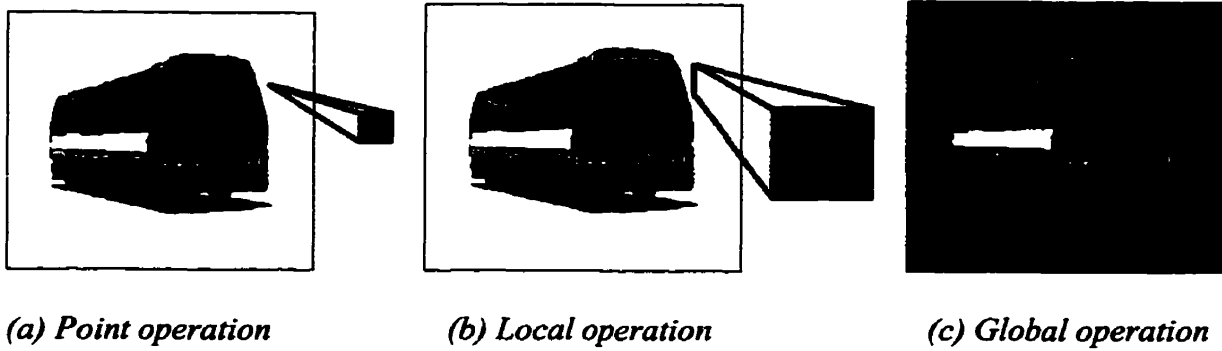


Figure 4.9. Image operation divided by regions of operation: point operation, local operation and global operation.

processing with CMOS image sensors will be studied, particularly analog image processing in spatial domain. Furthermore, the nature of on-chip image processing and architectural implementations will be investigated in terms of the operational regions (interconnectivity) described above. The vision algorithms under the same interconnectivity are subdivided by implementation design and functional operation. Thus, an effective architecture is proposed for each subdivided algorithm. Now, the characteristics of image processing operation and their adequate architectural implementations for the image processing integration will be investigated in detail into in terms of interconnectivity.

Chapter V

5. Point Operation

5.1. Introduction

Among the simplest of all image enhancement techniques some fairly straightforward, yet powerful, processing approaches can be formulated with light intensity (gray level) transformations alone. Because enhancement at any point in an image depends only on the gray level at that point, the techniques in this category are often referred to as point operations. The final output value is spatially independent of other pixel values, but only dependent on that pixel value, typically the gray level, at that point.

In aspects of on-chip integration with image sensors, point operations can give a number of advantages, such as parallel processing during integration, real time operation, slow processing elements, low power consumption, simplicity of design, and small silicon area. In addition, because the point operation is feasible for pixel-level implementations, high SNR, low power and concurrent adaptive processing can be easily achieved with the pixel level implementation. However, there is a limitation on the number of transistors inside the pixel due to a restricted size of pixel with a reasonable fill factor (see Figure 4.5). Point operations are still low-level image processing, and thus it is assumed that further signal and image processing stages can acquire the image output and process it.

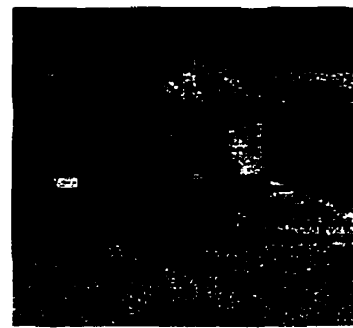
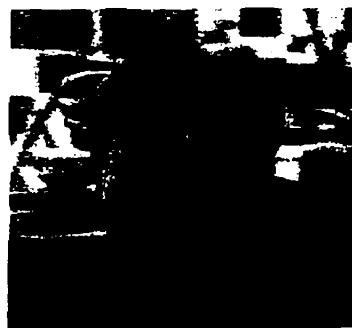
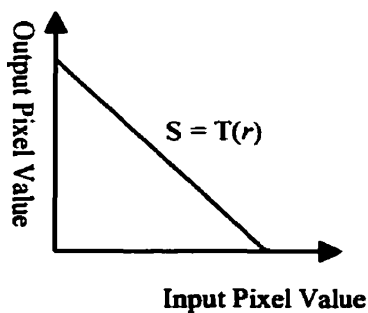
In order to understand the nature of the point operation and to find relationship between algorithms and system-level architecture, we will look into major algorithms of point operation and divide these operations by similarity of the functional processing. These point operations are categorized by their operational nature into three major groups: concurrent

pixel processing (intensity transformation), histogram processing, and inter-frame processing. Examples of the point operation algorithms will be described shortly. These examples include major algorithms for each operation, but do not contain all the possible algorithms in the category.

A. Concurrent Pixel Processing

Concurrent processing is an image-processing algorithm, which operates on only a particular pixel value, independent of any other pixels. Not only is it spatially independent of other pixel values, it is also temporally independent of its own pixel value, which means the present value of a pixel is not affected by the previous or the future values of the pixel. Because all the processing in these operations modifies/transfers light intensity of the input image, keeping a constant relationship between inputs and outputs for the whole array, this process is also called intensity transformation. The output value of a pixel is determined by the input value of the pixel, according to the intensity transfer (response) function, $S = T(\gamma)$. The operation may be processed concurrently during its light integration. Examples of the point operation algorithm, well described in [74][75], include:

Image negatives: The technique is to reverse the order of light intensity values so that the intensity of the output image increases as the intensity of the input decreases, shown in Figure 5.1.



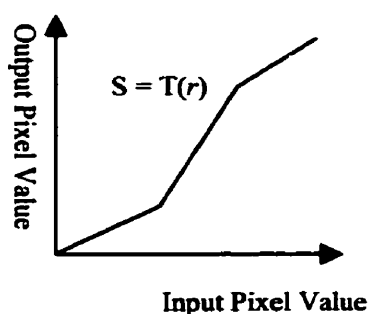
(a) *Intensity Transfer (response) Function*

(b) *Original image*

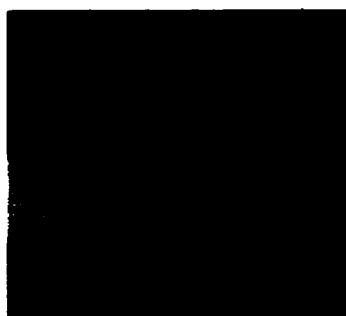
(c) *Image negatives, simulated with LviewPro*

Figure 5.1. Image processing of image negative is to reverse the order from black to white. Intensity response and software (Lview Pro) simulated sample images are illustrated.

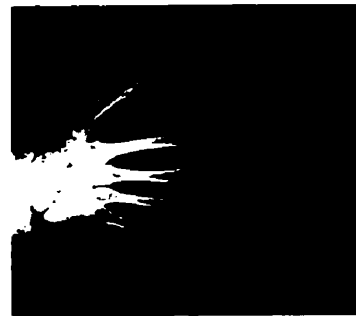
Contrast Stretching: Poor illumination environment and settings often cause low contrast images. The resulting narrowly distributed pixel values of low-contrast images can be expanded into wide intensity distribution, increasing the dynamic range and thus the contrast of the images. One example of typical contrast stretching transformation is shown in Figure 5.2. By increasing the slope of the intensity transfer function, where a large portion of the pixel value distribution is located, the contrast of the input image can be increased.



(a) *Intensity Transfer (response) Function*



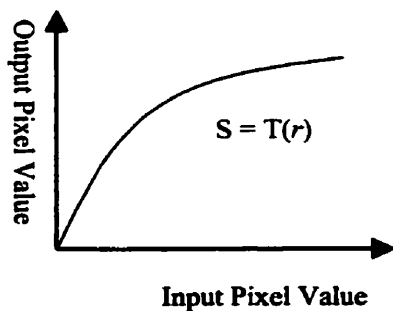
(b) *Original image*



(c) *Contrast stretching, simulated with LviewPro*

Figure 5.2. Contrast stretching technique stretches intensity response line so that the slope of response line in region of interest gets steeper and appearance of interest in an image is emphasized with a higher contrast.

Compression of dynamic range: With a given range of output pixel values, a transfer function can increase range of input pixel values, thus increasing the dynamic range of the light intensity. An effective way to compress the dynamic range of pixel values is to



(a) *Intensity Transfer (response) Function*



(b) *Original image*



(c) *Image compression, simulated with LviewPro*

Figure 5.3. With a given range of output pixel values, output values can have wider range of input pixel values by compression of the pixel values.

perform the logarithmic intensity transformation with the following transfer function:

$$S = c \log (1 + |r|),$$

where c is a scaling constant.

Gray level slicing: This is often used to highlight a specific range of light intensity in an image. One technique is to put a high value for all gray levels in the range of interest and a low value or unity value for all other gray levels, as shown in Figure 5.4.

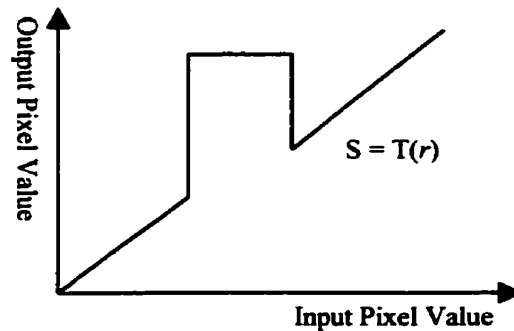


Figure 5.4. Gray level slicing is a technique highlighting a specific range of gray levels in an image by displaying high values for region of interest and low values for all other gray levels.

Bit-plane slicing: Instead of highlighting intensity ranges, the highlighting specific bits might be desired in order to discriminate contribution of individual bits to total image appearance. In 8 bit images, only the five highest order bits contain visually significant data. The other bit planes contribute to more subtle details in the image [74]. Depending on what data is emphasized, the individual bit can be selected and highlighted with bit-plane slicing.

B. Histogram Processing

The second type of point operation is histogram processing. Histogram processing techniques are based on modifying the output images by modifying the histogram of its gray levels through the transformation function. The gray-level histogram of an image is the distribution of the gray levels in an image. In general, a histogram with a small spread has low contrast, and a histogram with a wide spread has high contrast, whereas an image with its histogram clustered at the low end of the range is dark, while a histogram with values clustered at the high end of the range corresponds to a bright image [75]. Histogram processing can vary

from simple mapping functions, which can stretch, shrink (compress), or slide the histogram, to more complicated algorithms that require detailed analysis of its probability density functions such as histogram equalization and histogram specification.

Histogram equalization (linearization): Histogram equalization is a popular technique for improving the appearance of a poor image. It is similar to a histogram stretch but it generates more effective outputs of an input image. This technique is based on obtaining a uniform histogram where the histogram of the resultant image is as flat as possible. The theoretical basis for histogram equalization involves probability theory, where the histogram is treated as the probability distribution of the gray levels [74].

Histogram specification: Since histogram equalization is capable of generating only one result (an approximation to a uniform histogram), it is not an interactive image enhancement application. The histogram specification is to specify particular histogram shapes, highlighting certain gray-level ranges in an image. Because it has a flexibility of selecting a certain gray-level ranges, it can generate more visually appealing appearance of an image and become superior to histogram equalization.

C. Inter-frame Processing

The third type of point operation is inter-frame processing where the intensity level of a pixel is modified independently in space, but not in time. In order to calculate the final values of the pixels, the processing needs multiple frames of images, with at least two frames of input images, containing time dependency. The examples of the processing include:

Image subtraction: The difference between two images is computed as the difference between all pairs of corresponding pixels from the two images. This image subtraction is often used in motion detection, radiography, feature extraction and background subtraction.

Image averaging (multi-image averaging): Under assumptions that the noise is uncorrelated and has zero average value, averaging multiple images reduces the noise of the image, by \sqrt{N} , where N is number of frames. By storing pixel values of previous images in a frame memory, the average values of several images can be computed with lower noise level.

5.2. Comparisons between On-chip Implementations for Point Operation

We have divided point operations into three different types in terms of their processing characteristics: concurrent pixel processing (intensity transformation), histogram processing and inter-frame processing. When these point operations are integrated with CMOS image sensors on a single chip, these characteristics of the processing should be taken into consideration for system-level architecture and circuit designs of on-chip processing integration. Here, we study on-chip implementations of the point operation. The three types of point operation are investigated at different implementation levels of on-chip processing: pixel, column, chip and frame memory. General system-level architectures are discussed and different integration methodologies for each type are compared.

Concurrent Pixel Processing;

Concurrent processing (intensity transformation) can comprise an intensity transformer (linear/non-linear amplifier) with controllability in pixel, column, chip and frame memory processing. The concurrent processing, compared to histogram processing and inter-frame processing, has a wide choice of implementations. Although the concept of the design seems to be simple, the actual design of an amplifier with good controllability, or programmability is not straightforward.

Concurrent processing, integrated at pixel level, provides a number of advantages: parallel processing, processing during integration, high SNR, low frequency processing, low power, and adaptation of image signals and processing. Its main attraction is parallel processing during integration. Parallel processing during integration gives great flexibility of operation as well as local and global adaptation. Parallel processing permits more time for processing because typically integration time for input image (light) is much longer than the processing time. This slow processing frequency results in low power consumption, particularly for analog-intensive designs, because the power in analog processing is typically proportional to capacitance and the operational frequency squared. However, because a pixel requires a reasonable fill factor for good photosensitivity, only a small portion of the pixel area is preserved for processing circuits. Therefore, implementations at pixel level have severe limitations on pixel size and the number of transistors that can be practically used in a pixel.

Concurrent processing at the column level has more freedom on design area than at pixel level. Still, column level implementations have restrictions on column width, but typically not in the vertical direction. Therefore, more flexible circuit designs can be implemented and more programmability can be added. Column level implementation maintain parallel processing, thus resulting in low processing frequency and low power consumption (but higher than pixel level implementations).

Concurrent processing at the chip level, where an intensity transformer is located at the final serial output channel, consumes the smallest area and has the highest flexibility in circuit design and control. However, this requires a fast processing speed with a high bandwidth, and often results in high power consumption.

Concurrent processing with frame memory (typically analog memory) locates all the processing circuits apart from the image sensor array (to below the image sensor array) and results in higher fill factor. However, because the concurrent processing is independent of spatial and temporal differences in the pixels (e.g. it does not need multiple frames of images), analog memory becomes an unpractical implementation, often causing large power, large area, and high fabrication cost. Therefore, analog memory implementation for concurrent processing is not recommended unless special applications are needed.

From the above comparisons, pixel and column level implementations are recommended for system-level architectures and circuit designs in concurrent processing. Particularly because the point operation does not have any interconnections to the neighboring pixels, pixel level implementation is strongly recommended if a small number of transistors can embrace the necessary operation. Pixel and column level implementations, with the benefit of parallel processing, can save power and have flexible designs in processing circuits.

Histogram Processing;

The implementation of histogram processing consists of a histogram generator at chip level and intensity transformers with pixel, column, chip or analog memory implementations. Because the intensity transfer function of histogram processing is generated according to the histograms of input images, the histogram generator becomes an important component, located at a common output channel to collect all the pixel values. Therefore, histogram generation is perhaps strictly a global operation, but closely related to point operation. The

histogram generator in the histogram processing constitutes a major difference from simple concurrent processing where intensity is transformed in concurrent processing, predetermined or manually programmed. Histogram processing uses a data-derived programmed intensity transformer from a histogram generator. Histogram processing has many similarities to concurrent processing due to its intensity transformer that can be implemented at pixel, column, chip or frame memory levels. Therefore, histogram processing has the same architectural design benefits and drawbacks as concurrent processing.

Inter-frame Processing;

The last type of point operation, inter-frame processing, needs present pixel values and the pixel values of the previous frame at the same time. This processing has independency in area, but not in time. The inter-frame processing, by the nature of the operation, has correlations with pixel values in time. Therefore, it requires storage (typically analog storage) of pixel values for at least one frame interval. During the integration of a frame of image, the previous frame of image should be stored until the present image is captured and necessary operation are completed on these frames of the images. Because design of frame memory at column and chip level faces severe difficulty in its implementation, pixel level and analog frame memory structures are recommended for the inter-frame processing. However, for pixel processing, the storage (typically capacitance for analog memory) easily takes a large portion of the pixel area, reducing its fill factor and thus, the photodetector photosensitivity. Analog memory structure has a large storage area without affecting the fill factor of the photosensitive pixels. However, this structure may have high power consumption, high fabrication cost and more likely high complexity of design. Therefore, the choice for the implementation for inter-frame processing depends on the applications and user-defined specifications. If the specifications focus is on low power and low fabrication cost, the pixel level implementation is recommended. If functionality and programmability of the chip is to be more emphasized, the analog memory structure is proposed for the basis of the implementations. As a summary, the general descriptions and comparisons of point operation implementations are summarized in Table 5.

	Concurrent Operation	Histogram Operation	Inter-frame Operation
Pixel Processing	Good performance (high SNR, low power, adaptation, etc), but limited by pixel size		Each pixel should have its own memory in pixel, thus limited by pixel size
Column Processing	Slow processing Low power Flexible design in vertical directions, but still limited by column width		Not feasible because each pixel should be allocated with its own memory (storage) area
Chip Processing	High flexibility on design area and functionality of PE, but high speed requirement, high power. No parallel processing	Since information is extracted from all pixels, each pixel passes through a global processing element unit eventually	
Frame Memory	Iterative operation, but sacrificed with area, power, speed		Because of concurrent image capture, a frame is stored outside, not limited by pixel size

Table 5. General descriptions and comparisons of point operation implementations, for different types of the point operation.

5.3. Design of In-pixel Contrast Stretching

5.3.1. Introduction

Low-contrast images can result from poor illumination, lack of dynamic range in the imaging sensors, or even the wrong setting of a lens aperture during image acquisition. Since contrast

plays a critical role in overall quality of images, it is necessary to assure that the output image contains the appropriate contrast. In this thesis, as a part of image enhancement methods, especially for portable devices such as video cellphones, PDA and toys, a simple, but effective design of image contrast enhancement is investigated and designed. Since this smart sensor is intended to be embedded with portable devices, low power and low weight battery operation are focused as well as the effective operation of contrast enhancement.

The idea behind contrast enhancement is to increase the dynamic range of the gray levels in the image being processed and to obtain widely spread distribution of image histogram. The contrast enhancement techniques have a wide variety of processing methods from a simple contrast stretching to complex histogram equalization. Contrast stretching is a simple, yet powerful contrast enhancement technique, which will be dealt with in this thesis. This technique is used to produce an image of higher contrast than the original with a transfer function like one in Figure 5.5. The output ranges of r below an input of m , shown in Figure 5.5, are compressed, making this output range smaller than the original. The output ranges around m are stretched, which makes the output ranges larger than the original. The output ranges above m are compressed like the ranges below m . Interestingly, in an extreme case of contrast stretching, the transformation function produces a two-level (binary) image.

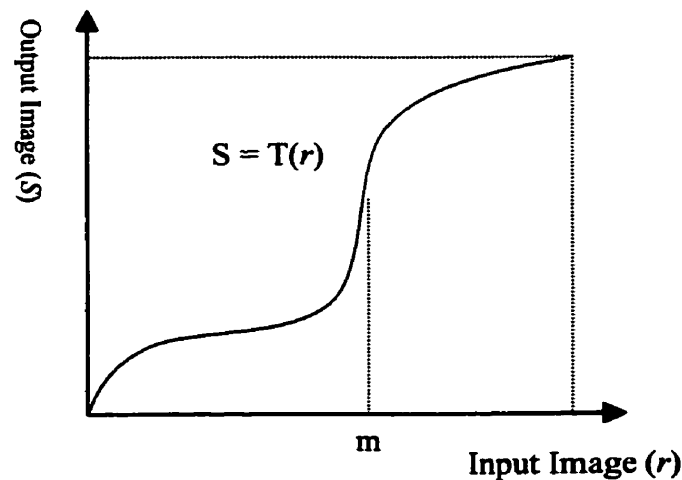


Figure 5.5. Gray-level intensity transformation function for contrast enhancement.

Histogram equalization is a more complex and powerful contrast enhancement technique than the contrast stretching. The histogram equalization modifies the histogram of an image to make the histogram as flat as possible, by multiplying all the pixels with the probability density function of the image. In terms of contrast enhancement, this technique increases the dynamic range of the image, which has a considerable effect in the appearance of the image. The histogram equalization needs two main computations: cumulative density calculation and transformation function generation.

A common component of these contrast enhancement techniques is the transfer function. Indeed, the transformation function is a common processing component in almost every point operation. The transformation function, also called the gray-level mapping function, is typically linear (nonlinear equations can be modeled by piecewise linear models) and maps the original gray-level values to other specified values. Now, we will study how we can use this intensity mapping functions for contrast enhancement operation.

5.3.2. Intensity Transformation Function

In the previous section, we have briefly discussed the operation of intensity transfer functions. The detailed operation of the function will be discussed in this section. Image processing functions in the spatial domain may be expressed as

$$g(x, y) = T [f(x, y)] \quad \text{Equation 5.3.1}$$

where $f(x, y)$ is the input image, $g(x, y)$ is the processed image, and T is an operation on f . In the intensity computation, the transformation function takes the simplest form of

$$s = T(r) \quad \text{Equation 5.3.2}$$

where, for simplicity in notation, r and s are variables denoting the gray level of $f(x, y)$ and $g(x, y)$ at any point (x, y) . This simple transformation function becomes a basis of point operation in image processing. The mapping function represents not only the relation between input and output images, but also has a considerable effect in the appearance of an image by three main operations: contrast adjustment, brightness adjustment and gamma adjustment. Here, the operation of the intensity transformer along with its simulations on appearance of images will be discussed.

The contrast of an image is adjusted by changing the slope of the mapping function. Figure 5.7 illustrates how the slope of the mapping function affects the appearance of an image as well as its histogram. The original image has a linear relation of unity between the input intensity and output intensity, as shown in Figure 5.6. As the slope of the mapping function gets steeper, the contrast of the image gets higher as shown in Figure 5.7. After all, when the slope becomes significantly high, the appearance of the image becomes more like a binary image, shown in Figure 5.7 (c). As the slope gets steeper, the histogram of the image gains more spread in its distribution, which indicates higher contrast. It is a global description that the histogram with a narrow shape indicates little dynamic range and thus corresponds to an image having low contrast. The histogram with a significant spread corresponds to an image with high contrast.

The brightness of an image can also be adjusted by the minimum or maximum value of the output intensity in the transformation function. For example, in Figure 5.8, the minimum value of the outputs in the mapping function increases, the appearance of the image gets brighter, becoming an almost white image in Figure 5.8 (c). The obvious observation from this histogram is that the minimum value of the histogram distribution increases as the minimum value of mapping function increases.

The last operation of the mapping function is to adjust the gamma function of an image, shown in Figure 5.9. The gamma adjustment is for non-linear behavior of many of its elements in the image-transmission chain. The relationship of the gamma correction can be

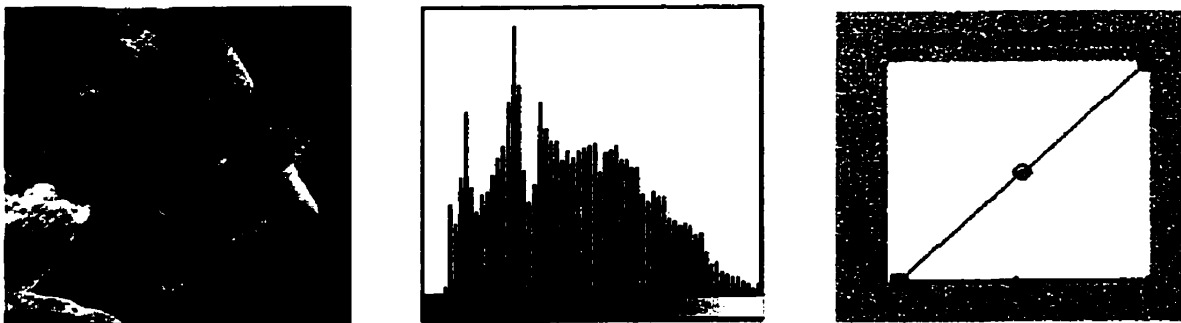
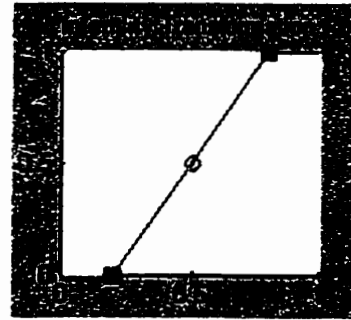
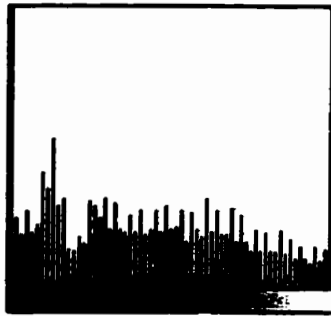
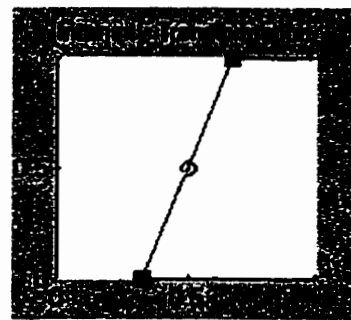


Figure 5.6. Original image for Matlab simulations on intensity transformer showing histogram and intensity transformation function.

(a) Contrast 1



(b) Contrast 2



(c) Contrast 3

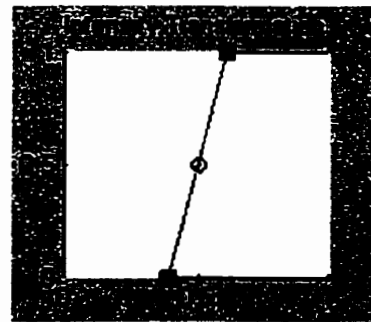
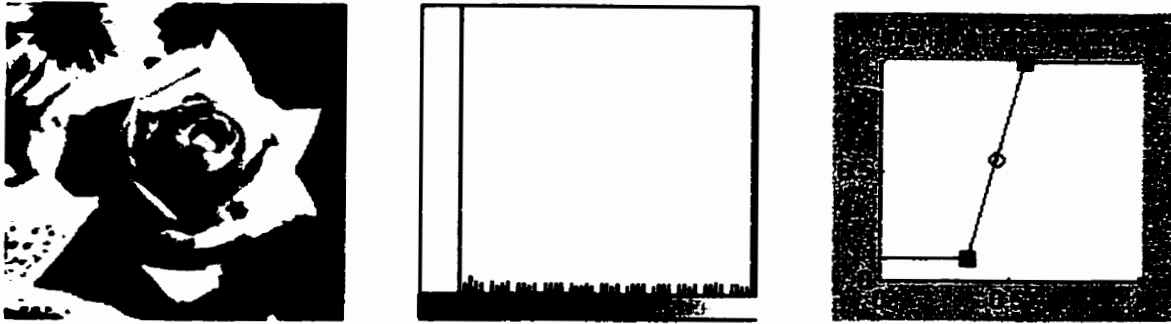


Figure 5.7. Matlab simulations on intensity transformer (mapping function) showing contrast stretching technique. As the slope of the linear line gets steeper, distribution of the histogram spreads out, gaining higher contrast.

(a) Contrast 3 and Vref1



(b) Contrast 3 and Vref2



(c) Contrast3 and Vref3



Figure 5.8. Matlab simulations on intensity transformer (mapping function) showing brightness adjustment technique. As the minimum values of the transformation response, the minimum value of the histogram increases, gaining higher brightness.

expressed in the form:

$$s = c r^\alpha \quad \text{in} \quad s = T(r), \quad \text{Equation 5.3.3}$$

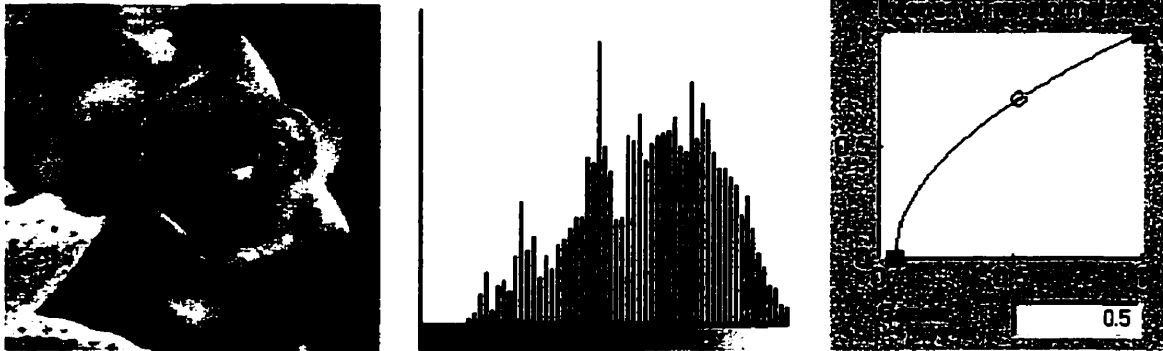
where c and α are constants and the exponent α (referred to the gamma of the device) takes a value between 0.5 and 3. To make sure that the perceived gray scale in the displayed images is correct (to compensate non-linearity of the components in processing), it is usually necessary to insert a gamma correction. The figure obtained by multiplying all the device gammas from the camera through to the display (but not including the eye) is known as the system gamma. If the conditions of viewing at the scene and at the display are the same (they often are not), the system gamma needs to be unity.

The design of an intensity transformer with contrast, brightness and gamma adjustment, requires a good programmability and a reasonable precision. Particularly, for contrast enhancement applications, because the dynamic range plays an important role in contrast, a design of the intensity transformer with high dynamic range is an essential design requirement. Also, it requires a high precision design that is frequently a drawback of analog circuits.

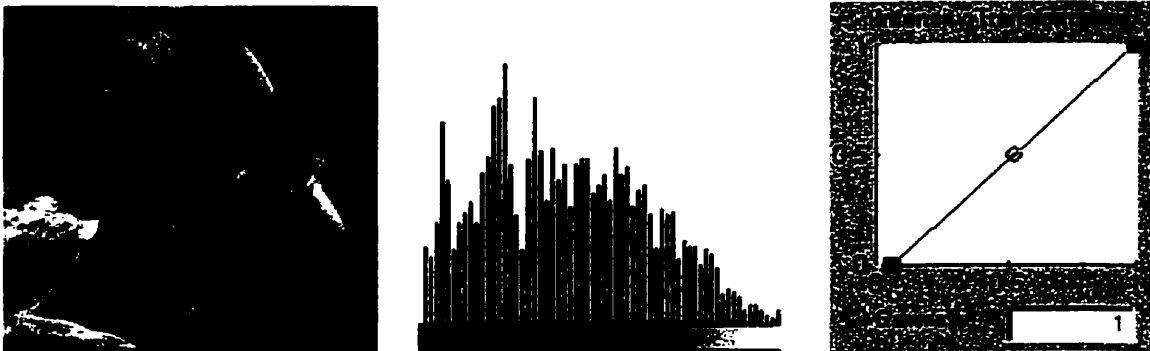
Often, the intensity transformer can be easily realized with a global gain amplifier with an offset, of which circuit designs are already well established. However, these designs are typically manipulated for high precision and high speed with large size. Therefore, these are not suitable for low power operation in the portable devices.

Therefore, this thesis proposes the implementation of an intensity transformer at pixel level, typically consuming lower power. Yet, for the pixel level implementations, the designs of processing elements require high fill factor for its photosensitive area, and thus they need to have a small number of transistors, which is one of the main design challenges in this thesis. In addition, good programmability is often important for the design of an intensity transformer, which is very difficult to achieve with in-pixel implementation. Therefore, in this thesis, the design focuses on an in-pixel intensity mapping function with a small number of transistors, and with reasonable programmability, and with low power for portable devices of interest here.

(a) $\Gamma = 0.5$



(b) $\Gamma = 1$



(c) $\Gamma = 2$

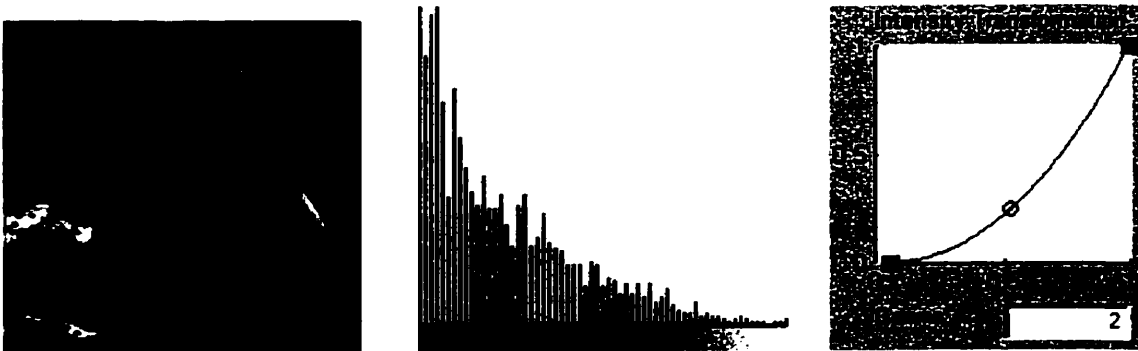


Figure 5.9. Matlab simulations on intensity transformer (mapping function) showing gamma correction technique.

5.3.3. Previous Work on Pixel Level Processing

Many researchers are attracted by potentially outstanding performance of in-pixel processing operation. The general performance and applications of pixel level processing are described with structural implementations of ADC on CMOS image sensors, in [31] focusing on pixel-level and in [32] column-level. Another relevant work concerns the size limitation of the pixel: how small the pixel should be in image sensors [33].

One of the well known works on pixel level processing is a floating point pixel-level ADC implemented by the Stanford ISL group [34]. Using the same design concept and circuit designs, the work also demonstrates a new way to increase dynamic range of the image sensors. Another research project on pixel processing is on-sensor image compression [35]. This proposes a novel integration of image compression and sensing on the same focal plane. The proposed image compression technique uses a conditional replenishment, which detects and encodes only moving areas. While the overall architecture and circuit designs are not directly related to pixel processing designs, the conditional replenishment implementation in analog at the pixel-level is interesting research. Other examples of pixel-level processing demonstrated include motion detection [36], individual pixel reset [37], pattern matching [38], and fingerprint detection [41]. Continuous improvement in pixel processing performance and functionality is expected. However, these are application-specific designs. An interesting in-pixel processing for general-purpose applications can be found in [93]. Here, we will investigate generalized system-level architecture and design methods for point operation, by demonstrating design and manipulation of on-chip light intensity transformer.

5.3.4. Designs of CMOS Active Pixel Sensor with In-pixel Intensity Transformer

We have designed and fabricated a prototype chip comprising a 64 x 64 array of in-pixel intensity transformer circuits with photodiode pixels, in standard 0.35 μm CMOS technology with 3.3V power supply. A die photograph is shown in Figure 5.10. Each pixel is 30 μm square including the in-pixel light intensity transformation circuit and it has a fill factor of 66%. The main objectives of this chip are (i) to demonstrate the feasibility of point operation with CMOS image sensors, (ii) to demonstrate the scalability of in-pixel processing integration with 0.35 μm technology, (iii) to achieve in-pixel processing with low power and real-time operation, and (iv) to address limitations and future directions of in-pixel

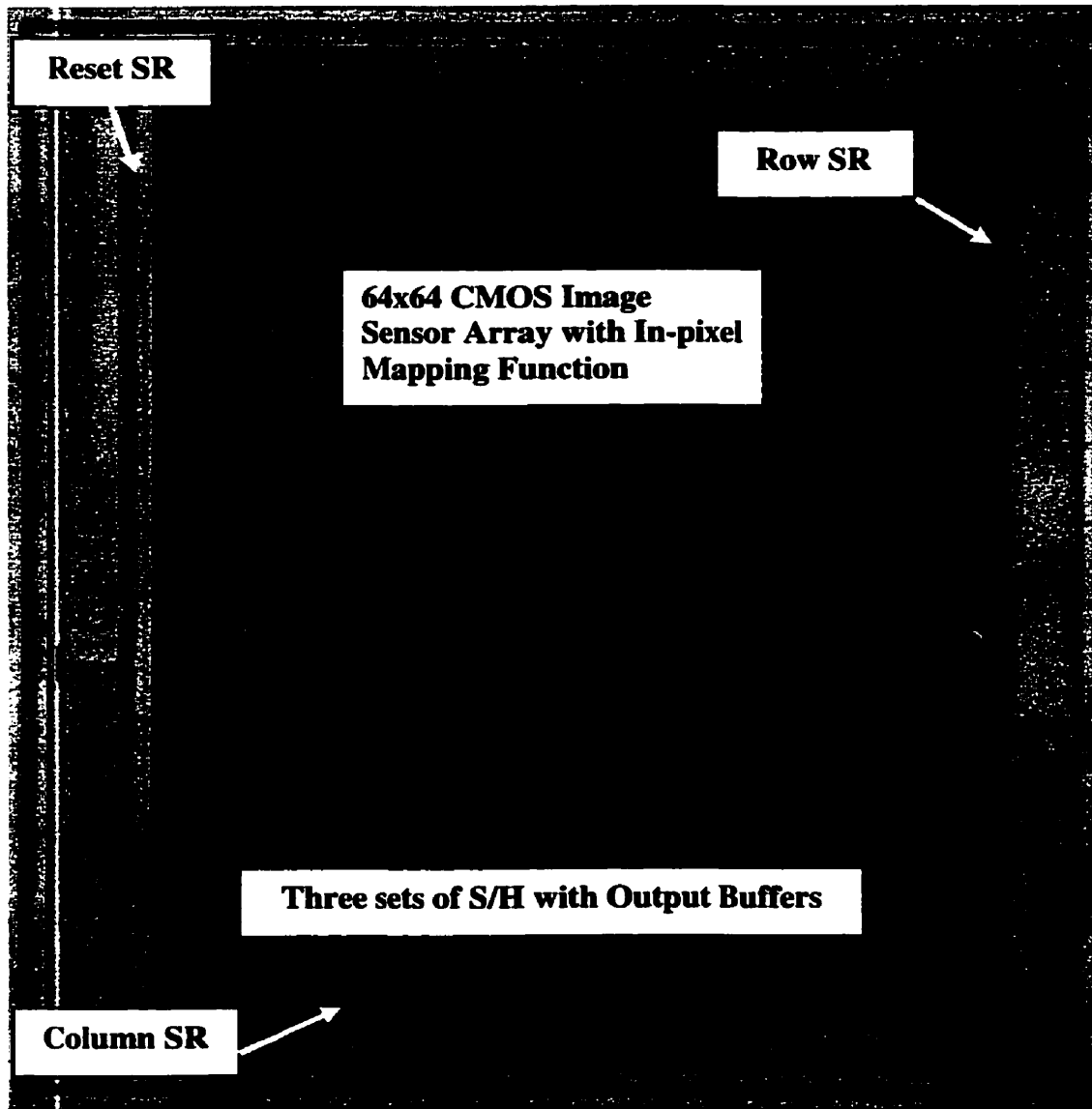


Figure 5.10. Die photograph of the prototype contrast stretching chip. The total area is 16 mm².

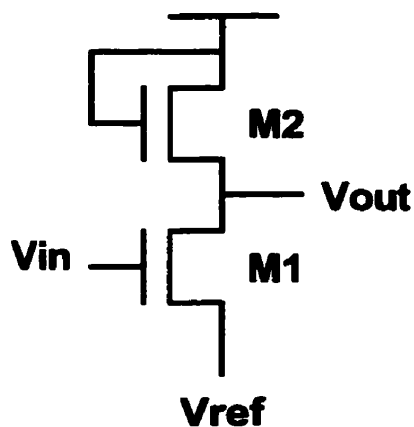


Figure 5.11. Schematic of common source follower consisting of a transformer with enhanced-mode NMOS active load.

processing with CMOS image sensors. The main challenges of this chip are to design a simple circuit with a small number of transistors in restricted pixel area, and to achieve reasonable precision of the circuit.

The main component of the chip is the pixel based intensity transformer, whose circuit schematic is shown in Figure 5.11. The basis of the circuit is a CMOS common-source amplifier with the source connected to input control voltage instead of ground, and an active load instead of a passive load. The transfer function of the common source amplifier is shown in Figure 5.12. The transfer characteristic displays three well-defined regions. In region I of Figure 5.12, the driving transistor M1 is off, since $V_{in} < V_{ref} + V_t$. Nevertheless, M2 is in the saturation region and is conducting a negligible current, thus the voltage across M2 is equal to V_{DD} , and hence the output voltage is $V_{DD} - V_{DD}$. In region II, M1 is conducting and is operating in saturation, and the transfer curve in region II is linear, which is useful for the amplifier operation. Finally, in region III, M1 leaves the saturation region and enters the triode region and the curve flattens out.

The analytical derivation of equation describing the transfer curve will be shown below. The derivation is done under the assumption that both devices (M1 and M2) have infinite output resistance (that is, horizontal characteristic lines) in saturation. Furthermore, the two devices will be assumed to have equal threshold voltages, V_t , but different values of K (K_1 and K_2).

When M1 is in saturation we have

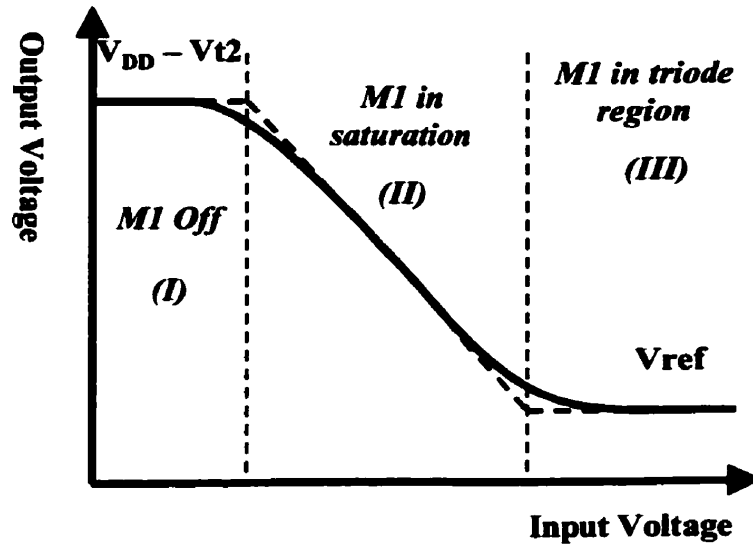


Figure 5.12. Voltage response of a common source amplifier with enhanced mode NMOS active load.

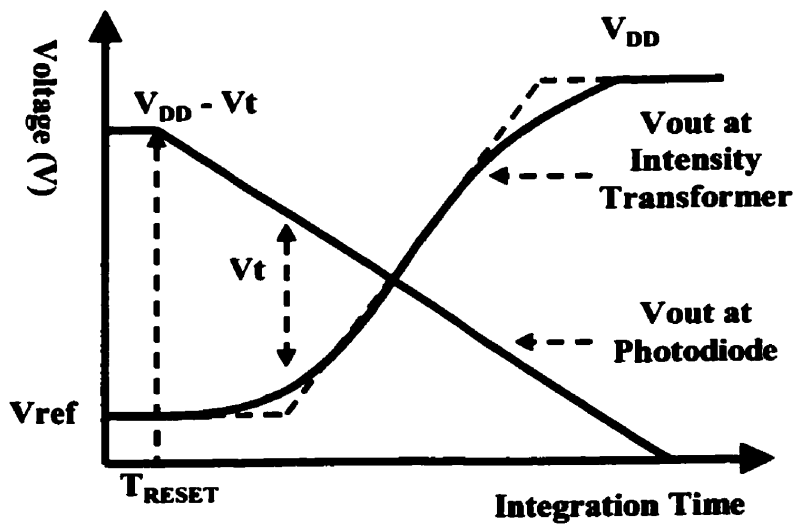


Figure 5.13. Response of a common source amplifier with voltage output of photodiode as its input.

$$I_{D1} = K_1(V_{GS1} - V_D)^2 \quad \text{Equation 5.3.4}$$

Since $I_D = I_{D1} = I_{D2}$ and $V_{GS1} = V_{in} - V_{ref}$, this equation can be rewritten

$$I_{D1} = K_1(V_{in} - V_{ref} - V_D)^2 \quad \text{Equation 5.3.5}$$

The operation of M2 is described by

$$I_{D2} = K_2(V_{GS2} - V_D)^2 \quad \text{Equation 5.3.6}$$

Since $V_{GS2} = V_{DD} - V_{out}$, the equation can be rewritten

$$I_{D2} = K_2(V_{DD} - V_{out} - V_D)^2 \quad \text{Equation 5.3.7}$$

Combining Eqs. (5.3.5) and (5.3.7) and with some simple manipulation we obtain

$$V_{out} = (V_{DD} - V_t + \sqrt{K_1/K_2}V_{ref} + \sqrt{K_1/K_2}V_D) - \sqrt{K_1/K_2}V_{in}$$

$$\text{Equation 5.3.8}$$

which is a linear equation between V_{out} and V_{in} . This is the equation of the straight-line portion of the transfer characteristic (region II) of Figure 5.12. This particular design of in-pixel intensity transformer has controllability on two operations: contrast and brightness. Gamma adjustment cannot be achieved with this design without making the design of the transformer more complex, taking too much area in pixel. In addition, because gamma correction is typically located at the end of processing stages in order to compensate the non-linearity of the components, it loses its value if it is placed at the front stage of image capture.

In Figure 5.13, the relationship between the photodiode input and the transformer is shown. The floating diffusion of the photodiode is placed at the input to the transformer. After a reset time, T_{RESET} , when a pixel is reset to $V_{DD} - V_t$, the voltage at the floating diffusion (photodiode node) decrements due to the photo-generated leakage current, and the transformer is off until the photodiode voltage becomes comparable to $V_{ref} + V_t$ ($V_{FD} = V_{ref} + V_t$). When the photodiode voltage becomes larger than $V_{ref} + V_t$, the transformer is in the linear region with a gain of $\sqrt{K_1/K_2}$ until the driving transistor gets into triode region around V_{DD} . In this analysis, the slope of the response function that is equivalent to the

contrast of the image can be changed by the resistance of the active load. Also, the minimum voltage value that is equivalent to the brightness can be changed by V_{ref} (control voltage connected to the source). With this given property of the common source amplifier, we are able to design a simple intensity transformer with a small number of transistors.

In this particular design of intensity transformer at the pixel level, shown in Figure 5.14, a PMOS active load is used instead of an enhanced mode active load for programmability and output swing. For contrast adjustment, the slope of the transformer should be controllable by an input signal. The enhancement mode NMOS load cannot be programmed, always having a fixed slope determined by the physical dimensions of the transistors. Using a PMOS active load with its gate controlled by input bias voltage allows different slopes according to the bias voltages. In addition, the enhancement mode load has an output voltage range from V_{ref} to $V_{DD} - V_t$ because V_{GS} should be greater than V_t in order for the transistor, M2, to stay on. The PMOS active load has an output range from V_{ref} to V_{DD} , gaining V_t from that of the NMOS load. HSPICE simulations on the transformer with PMOS load are shown in Figure 5.15. The simulation results demonstrate good behavioral performance and good controllability: V_{bias} for contrast adjustment and V_{ref} for brightness adjustment.

A standard source follower is placed right beside the transformer for normal mode image capturing (see Figure 5.17), so the prototype chip has three different outputs; normal, contrast stretched and binary mode. Thus, the array has three different sets of S/H's for each output

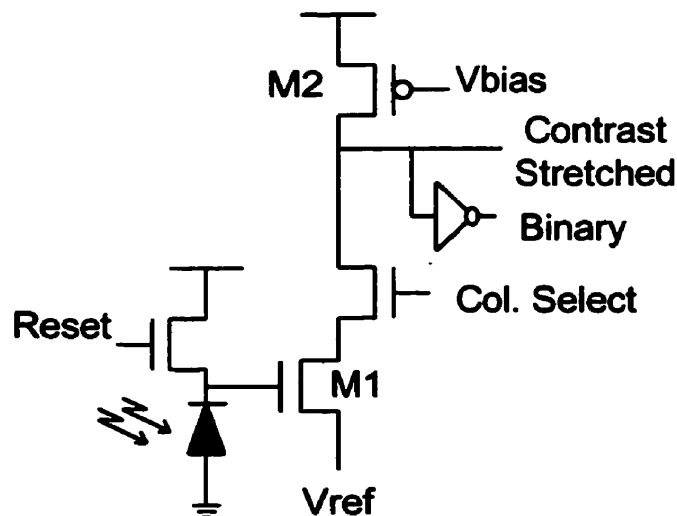


Figure 5.14. Schematic of intensity transformer implemented in design of the chip. The transformer consists of a common source amplifier with PMOS active load.

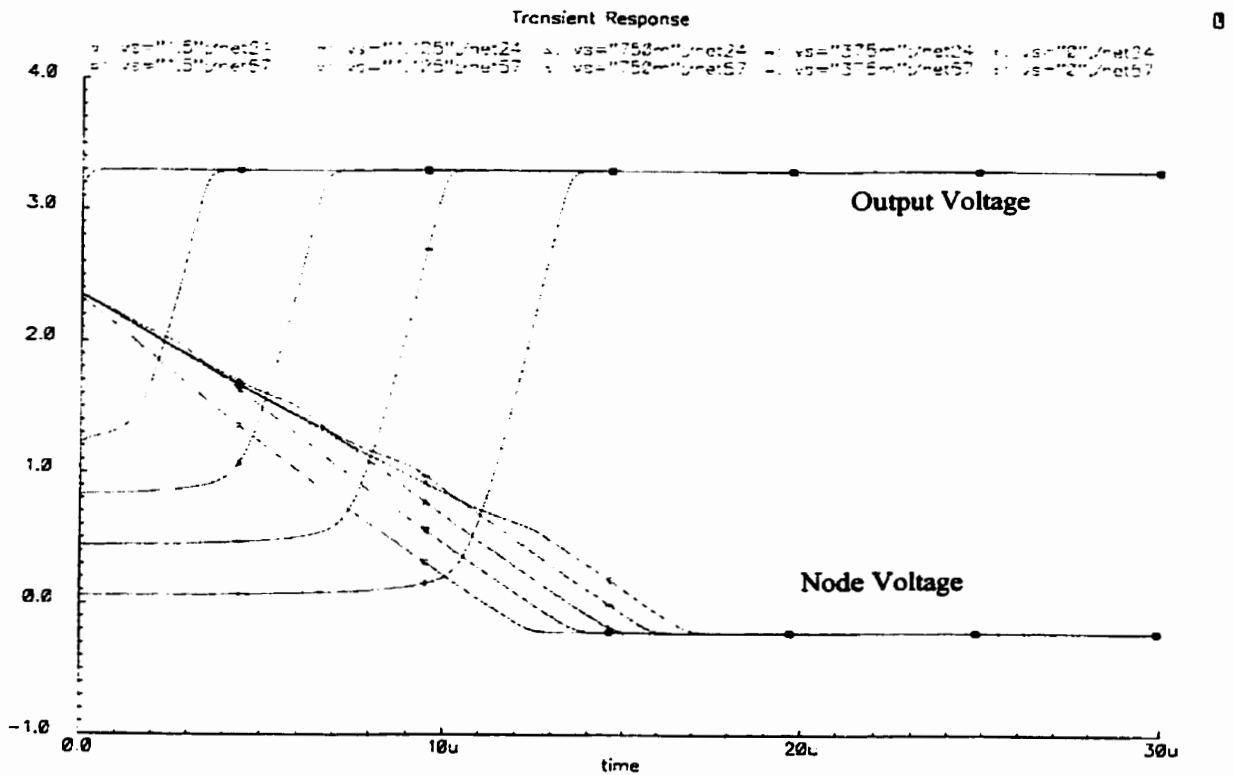
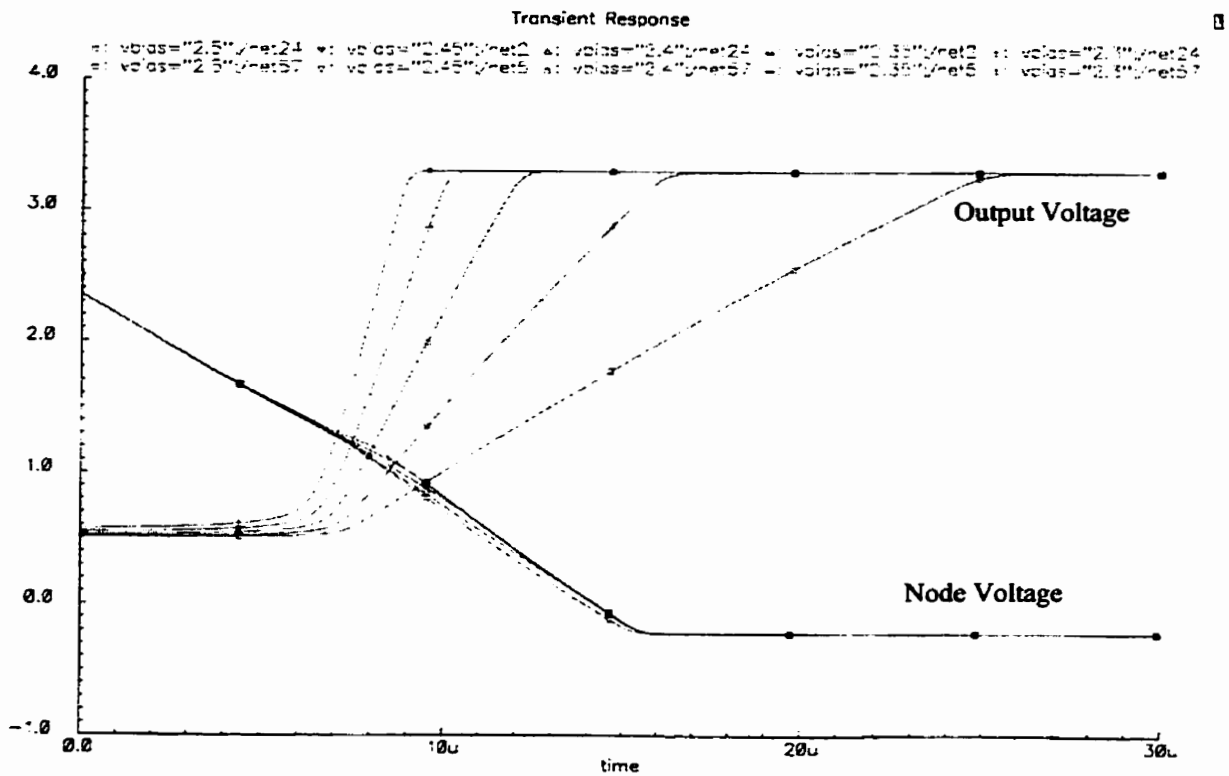


Figure 5.15. HSPICE simulations on an intensity transformer with a PMOS active load with (a) different biasing voltages (V_{biasp}) and (b) different reference voltages (V_{ref}).

channels, shown in the overall structure of the array in Figure 5.16. This structure is similar to the standard structure of CMOS image sensor array: image sensors, shift registers, bias bank and S/H's. Figure 5.16 shows 64x64 image sensors, reset and row shift registers, readout components for normal, contrast and binary mode at the bottom of the array.

Schematics of the major components are shown in Figure 5.17. CMOS image sensors use a photodiode with n^+ diffusion structure for its simplicity in layout. For image capture in normal mode, a standard active buffer with source follower is used in every pixel. S/Hs for the normal mode also use double poly capacitors with PMOS output buffers for the level shifting, as explained in Chapter 2. Different from standard CMOS image sensor techniques, contrast stretched mode and binary mode use common source amplifier structure for the intensity transfer function. In addition, S/Hs for these modes use NMOS output drivers instead of PMOS, because the intensity transformer does not have any voltage drops, unlike V_t drop in the source follower of normal mode, and therefore, the output driver does not need to compensate for any voltage drops. The output swing range of the transformer is from V_{ref} to V_{DD} , and when PMOS output buffers are used, the output of the driver goes from $V_{ref} + V_t$ to V_{DD} with a range of $V_{DD} - V_{ref} - V_t$. When NMOS output buffers are used, the output goes from $V_{ref} - V_t$ to $V_{DD} - V_t$ with a range of $V_{DD} - V_{ref}$. Therefore, the NMOS output drivers have an output range larger, by V_t , compared to the PMOS drivers. However, the use of NMOS and PMOS drivers does not matter in binary mode because of V_t loss in both drivers. Therefore, the reset and row selects are generated by active high shift registers with two inverters. The column selects are generated by active low shift registers.

5.3.5. Tests and Performances

The testing the prototype chip consists both of individual pixel test structures and the whole image sensor array. The tests on the individual pixel test structure are to verify performance of the intensity transformer with photodiodes, and the tests on the image sensor array are to demonstrate the effects of the transformer on the appearance of images.

Signal Responses of Individual Intensity Transformer

The first test is based on the signal response of individual pixel test structures with photodiodes. There are three variables affecting the response of the transformer: light intensity, biasing voltage (V_{biasp}) and reference voltage (V_{ref}). The light intensity is the

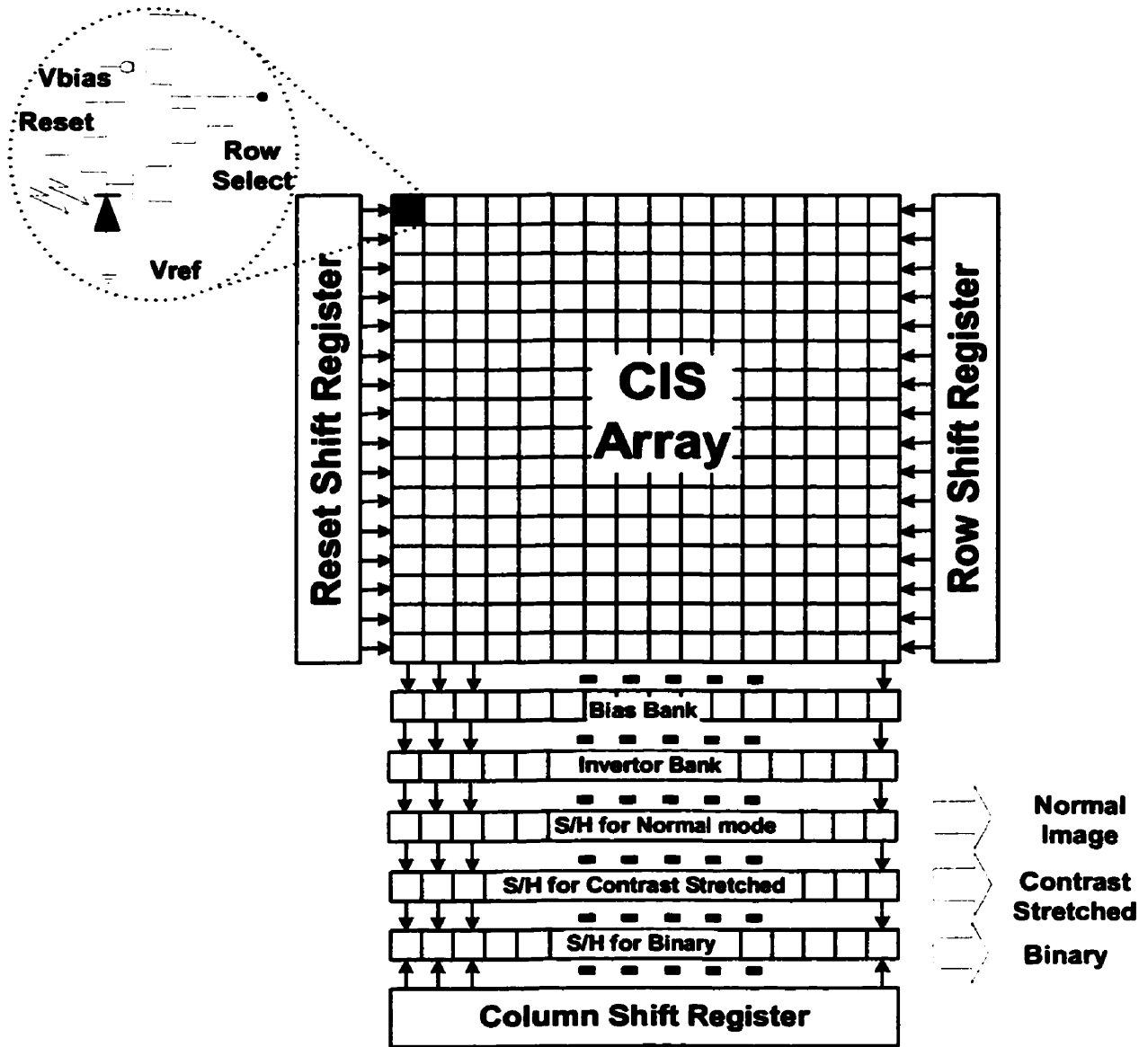


Figure 5.16. Overall structure of the chip, consisting of CMOS image sensors array and readout control circuits. The chip has three different output modes: normal, contrast and binary mode.

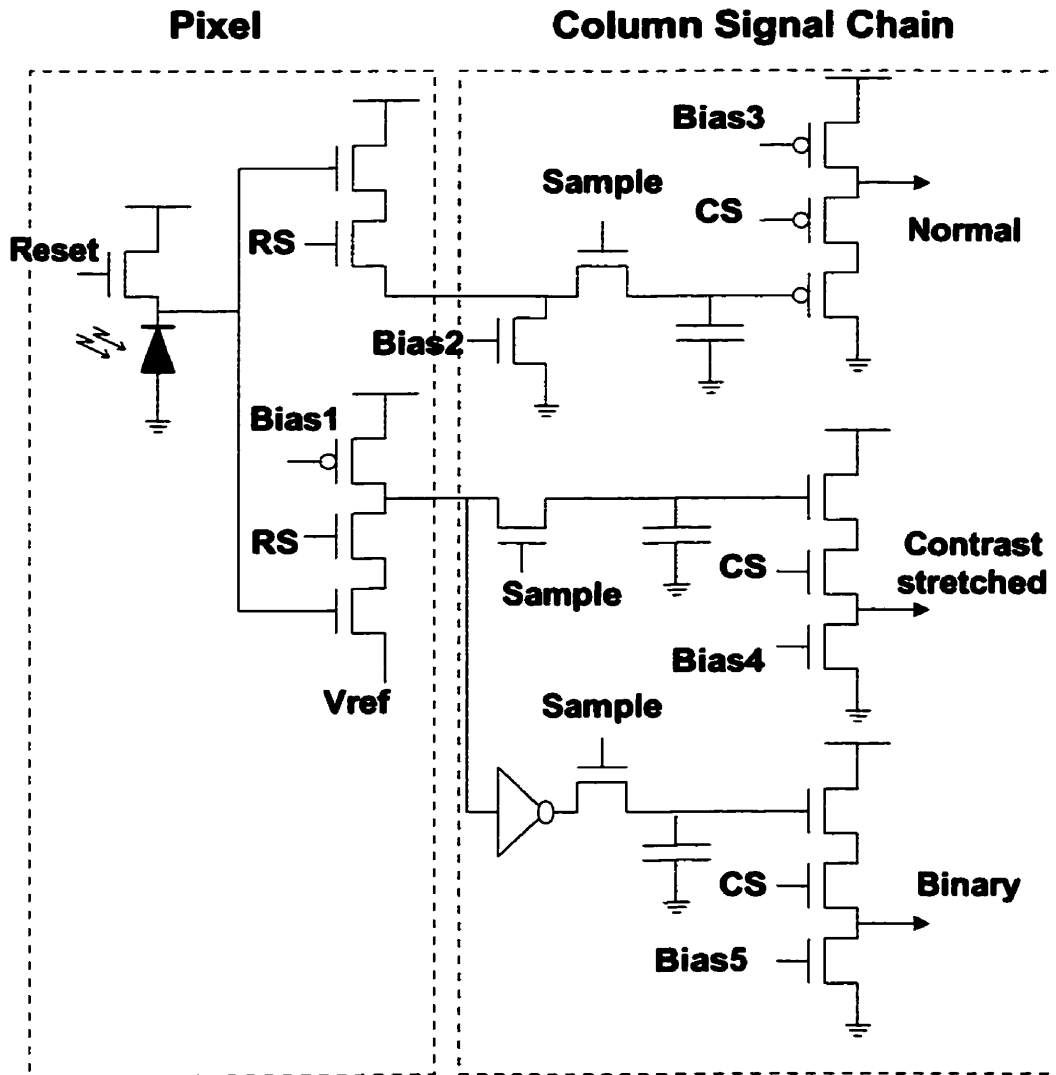


Figure 5.17. Schematics of main components in intensity transformer chip. It contains readout buffers and S/H's for three output modes.

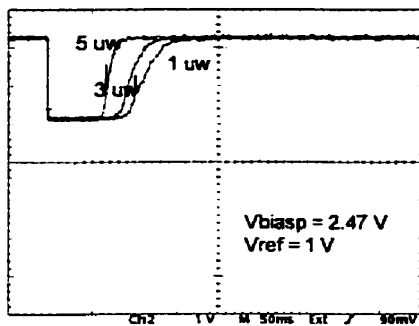
actual input to the transformer. The different light intensities affect the slope of the decrement at the photodiode node, and thus change the slope of the linear region of the transformer as well as the intercept of the off region and linear region. Figure 5.18 (a)-II shows the output of the binary mode (output of the inverter) and Figure 5.18 (a)-I shows the output response of the contrast stretched mode. As the light intensity increases at a fixed V_{biasp} and V_{ref} , the slope of the contrast mode response, Figure 5.18 (a)-I, gets steeper. The intercept of the off region and linear region also starts earlier. The early starting of the linear region with faster slope switches binary response faster. The highest intensity has the fastest switch-to-high, shown in Figure 5.18 (a)-II.

With different V_{biasp} , we observe the changes in the slope of the linear region; the more current (the smaller V_{biasp}) goes through the PMOS transistor, the steeper the slope becomes, shown in Figure 5.18 (b)-I. As discussed in the HSPICE simulations of the previous section, the V_{biasp} changes the slope of the transformer, thus changing the contrast appearance of an image. The steeper slope of the linear region (the faster response) typically generates an image with a higher contrast, which will be demonstrated in the next sections. The steeper slope of the response also leads to the faster switching of the outputs of the binary mode, shown in Figure 5.18 (b)-II.

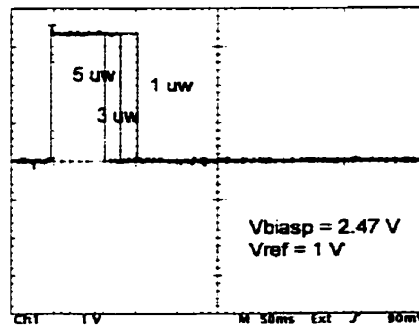
As V_{ref} increases, the minimum output voltage of the response increases, shown in Figure 5.18 (c)-I, because the minimum value is theoretically equal to the V_{ref} . The changes of the minimum voltage directly affect the brightness of the image. Since the higher V_{ref} turns the driving transistor to the linear region faster, the output of the binary mode switches faster with the higher V_{ref} in Figure 5.18 (c)-II.

The variations in the photoresponse of an individual pixel with light intensity, biasing voltages, and reference voltages have been tested and well demonstrated for circuit operation. These test results demonstrate that the response of the pixel to different light intensities and control voltages allows good control of intensity transformation. The tests on the individual pixel test structures verify operational performance for individual intensity transformers with a photoreceptor, which are analytically understood with the HSPICE simulations. These tests are well matched to the simulations and encourage further tests of their effects on images.

(a) Varying Light Intensity

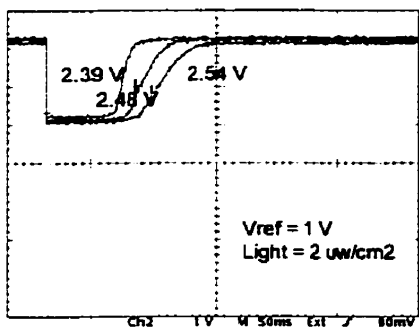


(I)

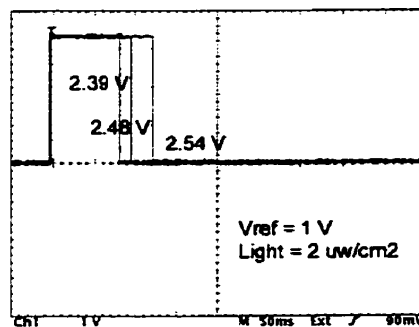


(II)

(b) Varying Vbiasp (Contrast)

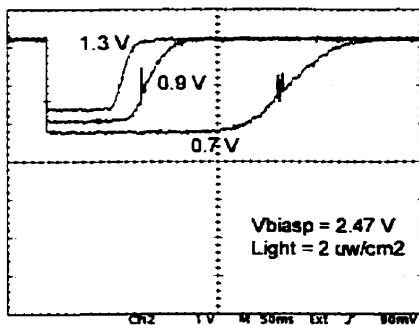


(I)

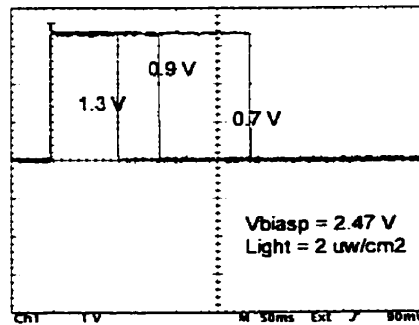


(II)

(c) Varying Vref (Brightness)



(I)



(II)

Figure 5.18. Photoresponse from the “stretch” output (top row) and inverter output (bottom row) of a pixel with in-pixel contrast stretch at various light intensities, bias voltages and reference voltages.

Image Capture in Normal Mode with Characteristics of Image Sensors

The first and the most important test of the image sensor array is to capture an image in real time mode. Here, we are able to demonstrate operation of image capture successfully. Some sample images are illustrated in Figure 5.19. As expected, the quality of images captured by the chip is not high, due to the fact that the chip process technology we used is not optimized for image sensors, but instead for logic and memory. However, the subtraction of a white background image enhances the quality of the captured images by reducing fixed pattern noise. Figure 5.20 (a) is a raw image and Figure 5.20 (b) is a fixed pattern noise subtracted image. There are some noticeable differences in their image quality; the processed image is cleaner and has a higher contrast.

The characteristics of the single chip, including image sensors, are summarized in Table 6. The conversion efficiency of the chip is only $0.1 \mu\text{V}/e^-$ (refer to Appendix C), which is very small, compared to commercially available CIS chips (typically $5\sim 10 \mu\text{V}/e^-$ for their conversion efficiency). Some of this unexpectedly poor performance may be because of the cross-talk between normal mode operation and contrast/binary mode operation, and the increased capacitance of the photodiode node. The global photoresponse of the prototype array in each of the three modes is presented in Figure 5.21 (a) for uniform illumination at wavelength of 540 nm. In a uniform dark room, the dark current is measured with varying integration time (sampling rate), as shown in Figure 5.21 (b). The characteristics of the chip are measured and calculated, based on the measurements and many other tests. The characteristics chart of the chips shows photoresponses including photosensitivity, spatial pattern noise, temporal noise, SNR and dark current. The physical parameters, including pixel size, fill factors and chip size are also included.

The chip consumes power of 14.85 mW, (typical power consumption of commercial CMOS image sensors is around $50 \sim 100$ mW for VGA format). It includes concurrent operation of normal, contrast stretched and binary mode in real time operation at 24 frames/second. When contrast and binary modes are turned off and power with only normal mode on is measured, the power consumption is around 6 mW,

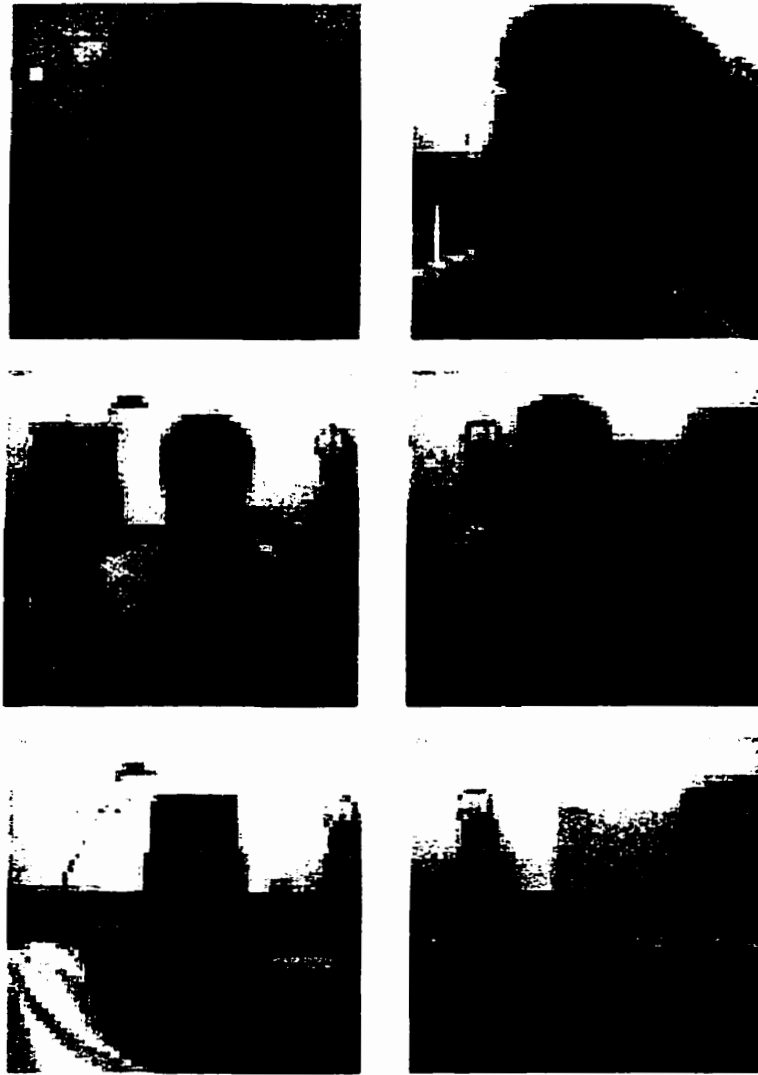


Figure 5.19. Sample images captured in real time by the chip in normal mode.



(a) Raw image

(b) Processed image

Figure 5.20. Pattern noise can be reduced by subtracting white background image from the raw image.

Quantity	Normal	Contrast Stretching	Binary
Technology	0.35 μm CMOS technology with double poly and 3 metal layers		
Chip Size	4 x 4 mm^2		
Pixel Size	30 x 30 μm^2		
Format of Array	64 x 64		
Fill Factor	66.01 %		
Vdd	3.3 V		
Output format	2 analog outputs and 1 digital output		
Frame Rate or Integration Time	5 frames/sec	5 frames/sec	5 frames/sec
Power	3.3x1.84 =6.072 mW at 24 frame rate	3.3 x (4.5 – 1.84)=8.778 mW	
Light Lux	150 ~ 200 lux		
Photo-sensitivity	2.094 mV / (uW/cm^2)	11.341 mV / (uW/cm^2) for range of 5 to 6.5 uW/cm^2	
Gain		5.42 ($\Delta V_{\text{contrast}}/\Delta V_{\text{normal}}$) in the range of 5 to 6.5 uW/cm^2	
Conversion Efficiency	0.1 uV/e^-		
Saturation Range	1.38 V	2.04 V	
Fixed Pattern Noise	50 mV (3.6% of saturation level)	210 mV	
Temporal Noise (N)	23 mV	30 mV	
Signal to Noise Ratio	35.56 dB	36.65 dB	
Dark Signal	0.03 V/sec		

Table 6. Single chip characteristics in normal mode and contrast mode.

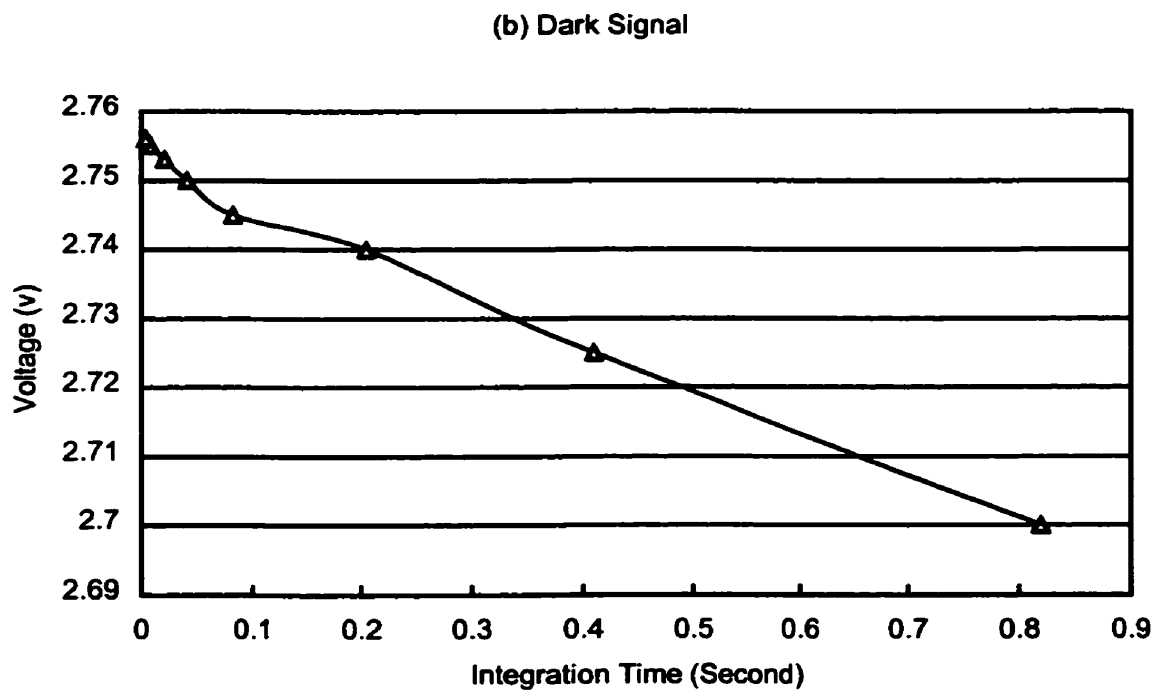
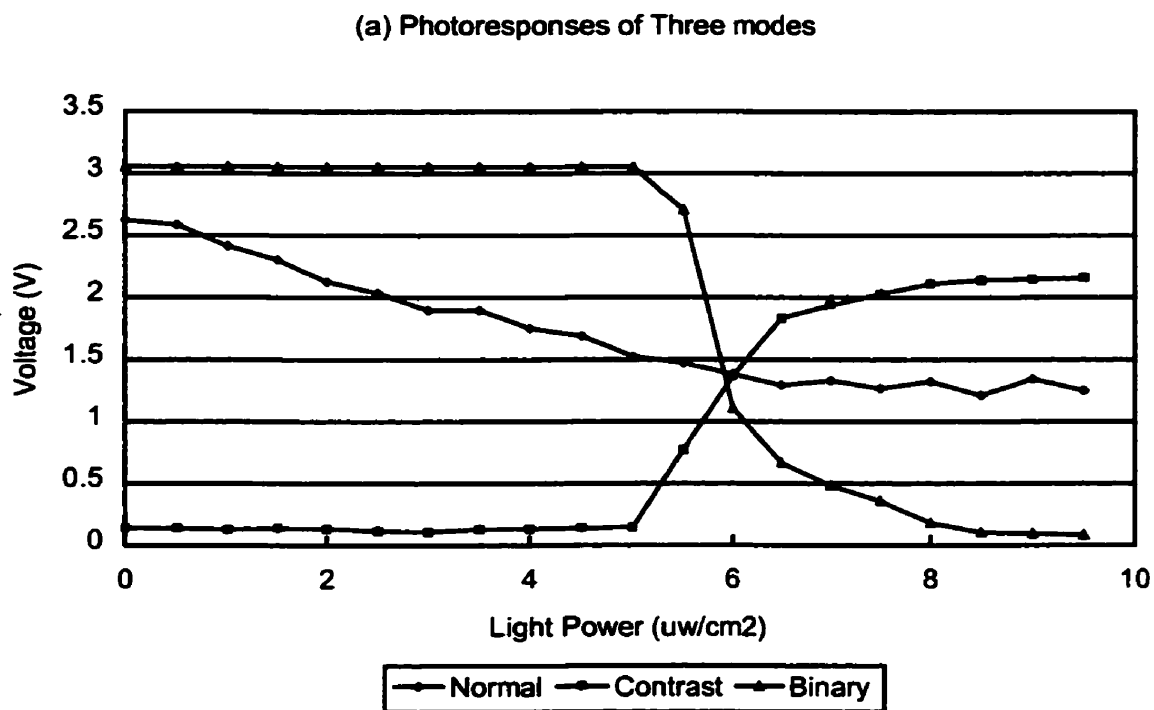


Figure 5.21. Characteristics of single chip. (a) photoresponse of three output modes and (b) dark signal measurements.

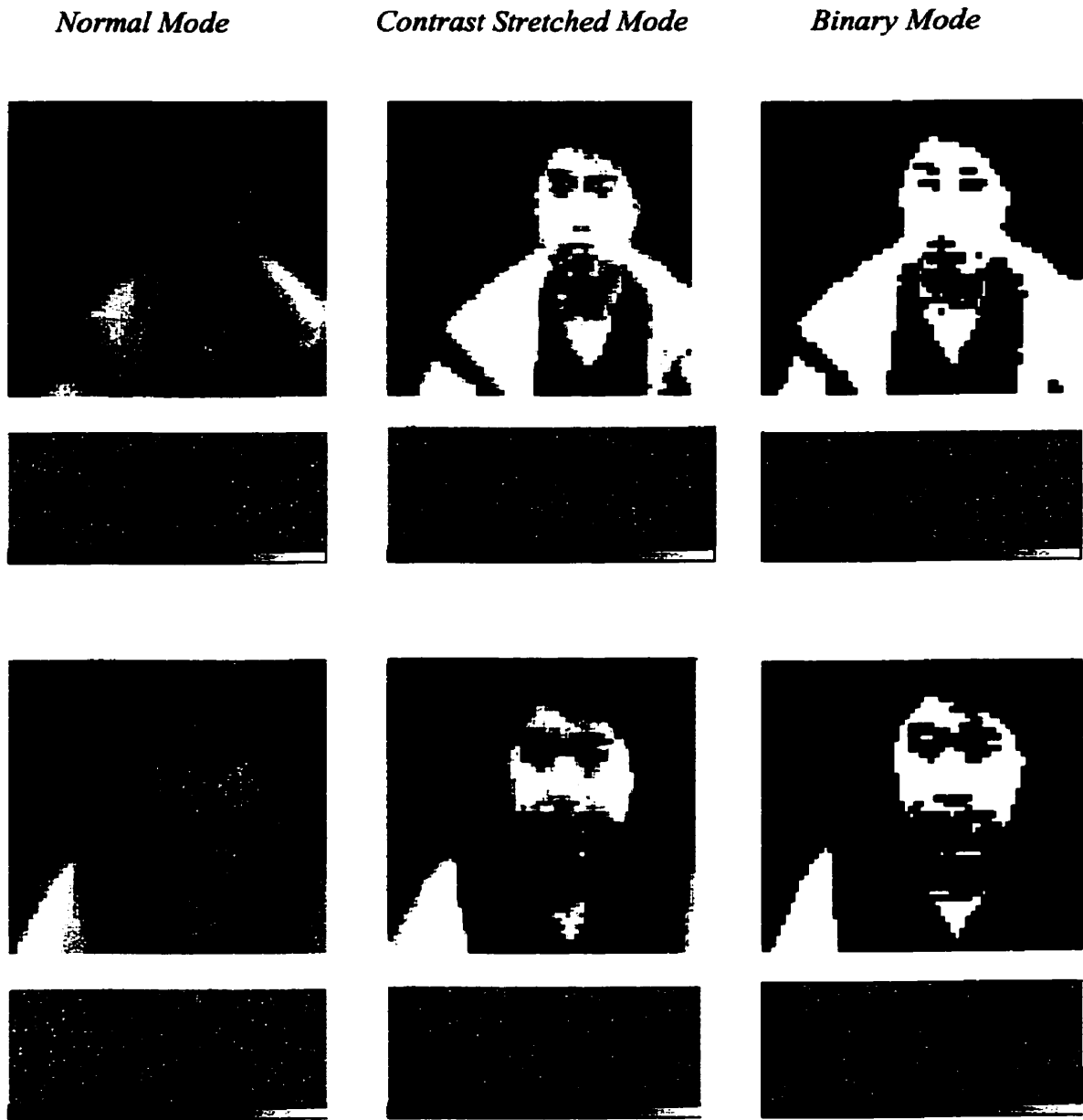


Figure 5.22. Sample images and histograms of three output modes.

which leads to the calculation that power consumption in contrast and binary modes is about 8 mW. This higher power is due to the large capacitance loads of PMOS, connected to the column lines. However, by extracting the active load transistors out of the array and inserting row transistors between, the processing power could be reduced because the CS amplifier is only on when the pixels are read out.

Intensity Transformer in Contrast Stretched Mode and Binary Mode

Images captured by the prototype sensor in the three operational modes are compared in Figure 5.22, along with calculated histograms showing the distribution of pixel values in the image. A normal mode image with poor contrast is shown with its narrowly distributed histogram. With an appropriate values for V_{biasp} and V_{ref} , the contrast stretched mode shows enhancement of the contrast by spreading out the histogram distribution. The binary mode converts the grayscale image to one bit binary image and therefore the histogram contains only two values of black and white. Two sets of three output modes are shown in

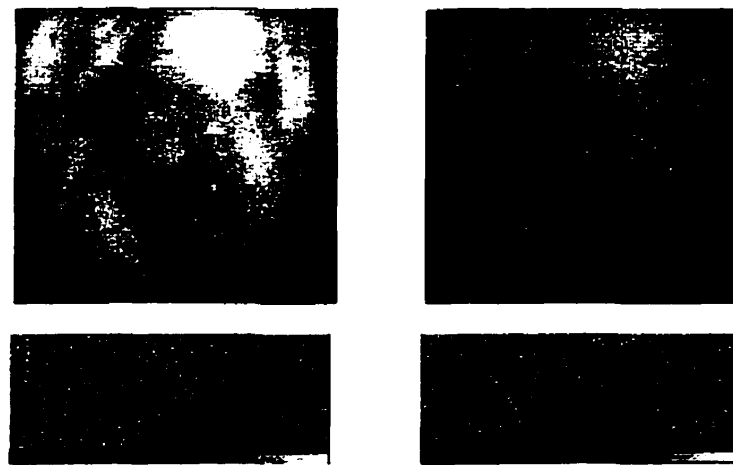


Figure 5.23. Original images captured in normal mode with different illumination (approximately, (a) is under 170 lux and (b) under 130 lux. The image of (a) is captured under a brighter illumination than the image of (b), showing overall distribution of the histogram shifts to the right (brighter grayscale).

Figure 5.22. The first set of images has a better contrast in terms of histogram distribution than the second set. With different combinations of V_{biasp} and V_{ref} applied to each of the images, the contrast stretched modes of these images have approximately the same histogram distributions, and thus same contrast. The binary modes of the images always have the two values of grayscale, with different number of black and white pixels.

The contrast stretched mode enables the modulation of the contrast of the image. As shown in Figure 5.24, biasing voltage (V_{biasp}) changes the distribution of the histogram of the original image in Figure 5.23, retaining the original maximum and minimum values of the histogram. As V_{biasp} decreases, the distribution of the histogram is spread flat.

The decrement of V_{biasp} to the PMOS transistor allows more current to flow through the driving transistor of the transformer and thus, the slope of the linear region response becomes steeper. However, because the original image of Figure 5.23 has a wide spread distribution of the histogram, the effects of V_{biasp} on the appearance of the image are not well demonstrated. Instead, observations on the distribution of the histogram give a good illustration on the effects of the biasing voltage to the contrast of the image. The contrast stretched mode enables modulation of not only the contrast of the image, but also its brightness. The V_{biasp} changes the distribution of the histogram, retaining the original maximum and minimum values of the histogram. In contrast, V_{ref} does not change the basic distribution of the histogram, but modulates the minimum value, thereby changing the brightness. Figure 5.25 shows different image captured with different reference voltages. As the V_{ref} increases, the minimum pixel value of the array increases, and thus the minimum value in the distribution of the histogram increases. The noticeable changes in the brightness of the image are shown, along with different V_{ref} . In addition, the increment of the minimum value of the histogram is observed with the brighter images with a higher V_{ref} .

Mismatches in In-pixel Intensity Transformers

One of the reasons for the degraded appearance in the contrast stretched mode and binary mode is the V_t and lithographical mismatch in the transistors of the transformer. Under the same input voltages of V_{biasp} and V_{ref} , the physical sizes of the transistors determine the output voltage of the transformer. Particularly, when the driving transistor in the intensity transformer is in linear region, the mismatch affects the output most, due to different gains of

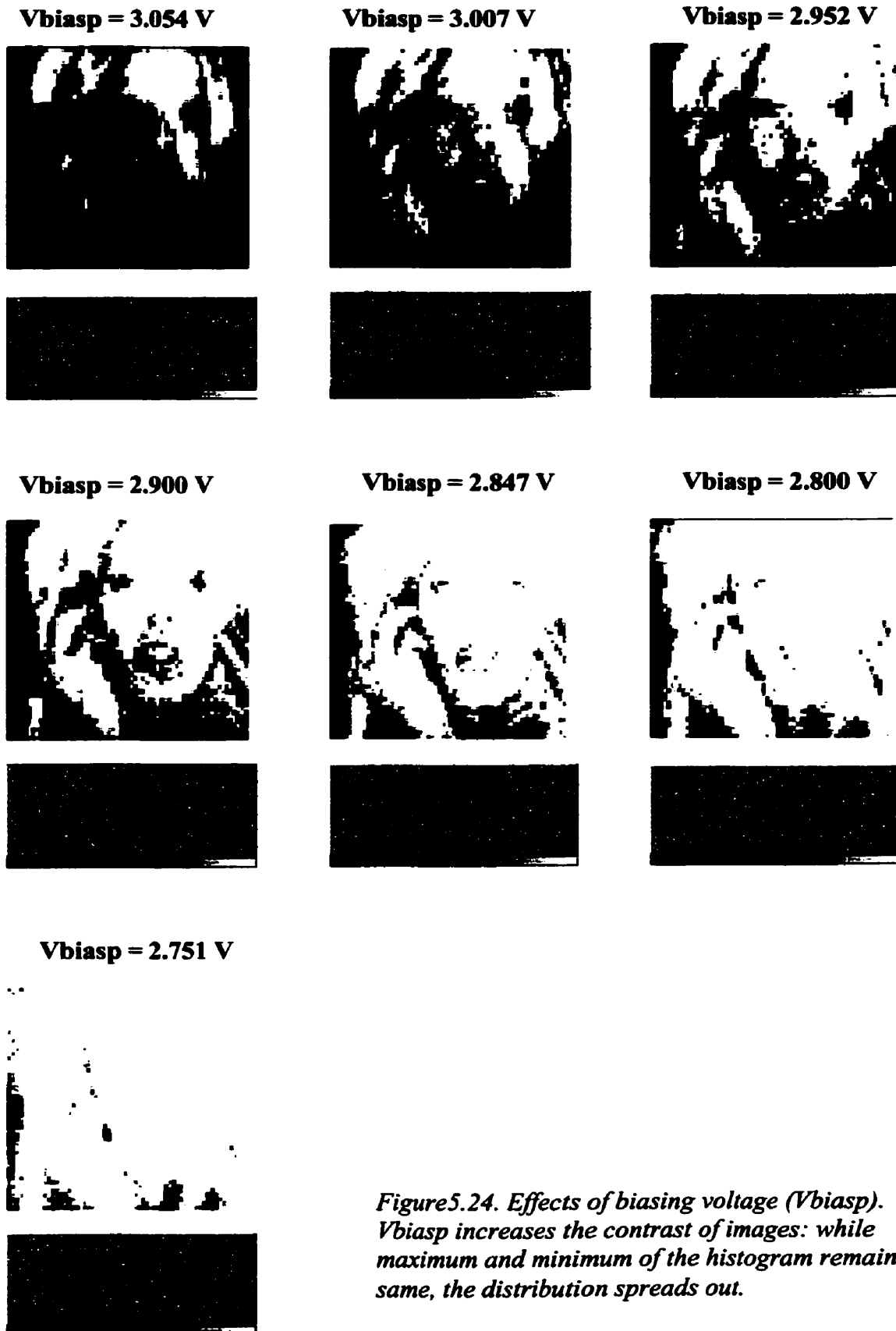


Figure 5.24. Effects of biasing voltage (V_{biasp}). V_{biasp} increases the contrast of images: while maximum and minimum of the histogram remains same, the distribution spreads out.

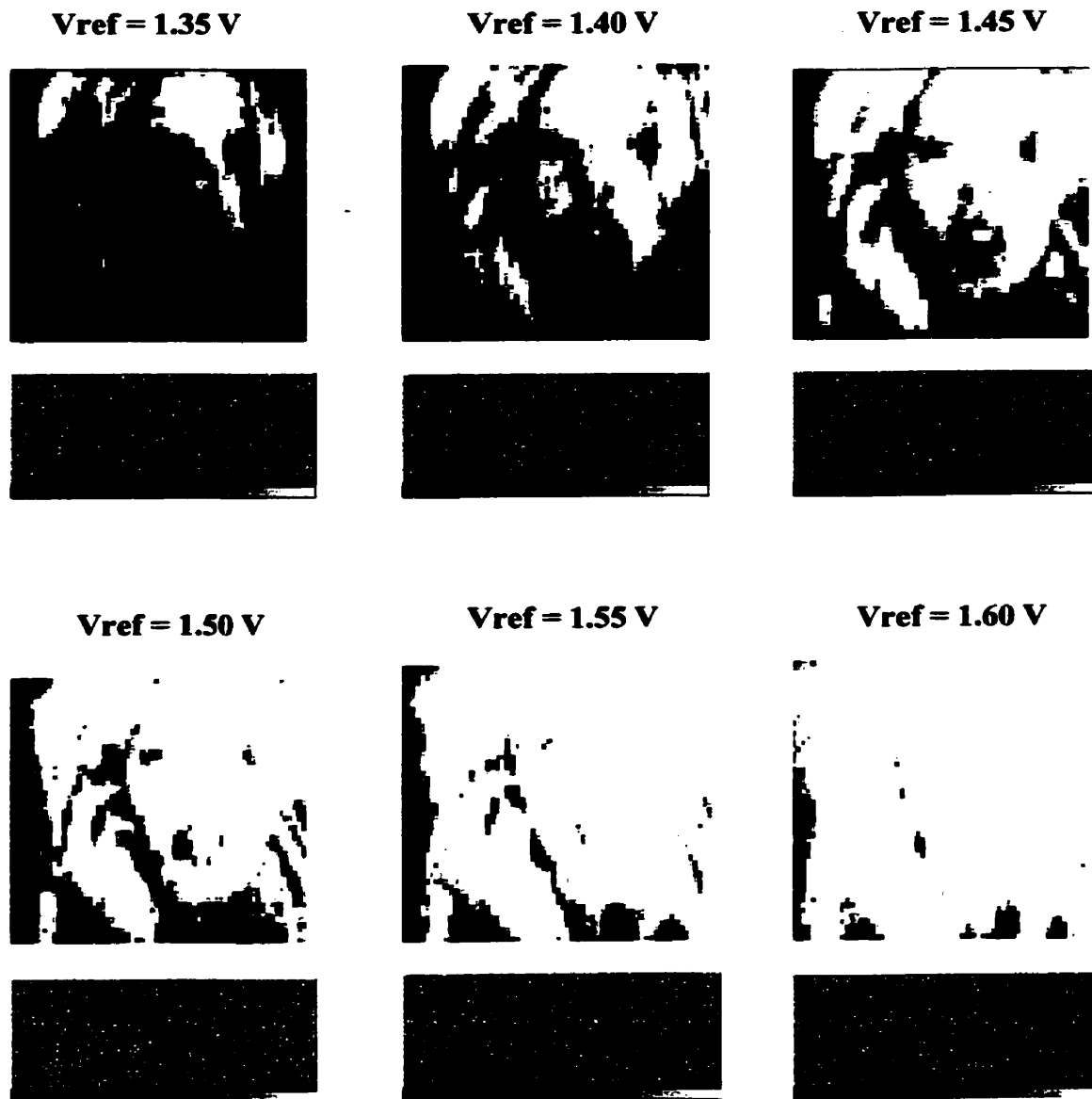


Figure 5.25. Effects of reference voltage (V_{ref}). V_{ref} increases the brightness of image; while distribution of histogram remains same, the minimum value increases as V_{ref} increases.

the transformer in the linear region. The pattern noise, due to the mismatch, is amplified by the gain with which the image signals are amplified. Figure 5.26 shows the tested measurements of the image sensors array in the prototype chip.

As light power (light intensity) increases, the pattern noise of outputs in the normal mode increases slightly. Different from the normal mode, the pattern noises of contrast stretched mode and binary mode remain at roughly constant values until the light power becomes around $5 \mu\text{W}/\text{cm}^2$, where the driving transistor of the transformer goes into saturation region. In the linear region of the transformer, due to the gain, the mismatch (pattern noise) gets higher. As the light power increases, V_{gs} of the driving transistor increases, hence the gain increases along with the pattern noise.

When the light power becomes around $6 \mu\text{W}/\text{cm}^2$, the pattern noises of contrast stretched mode and binary mode reach their peak values of 0.93V and 1.508 V respectively. This is when the driving transistor goes into its triode region. As V_{gs} increases further, the transistor

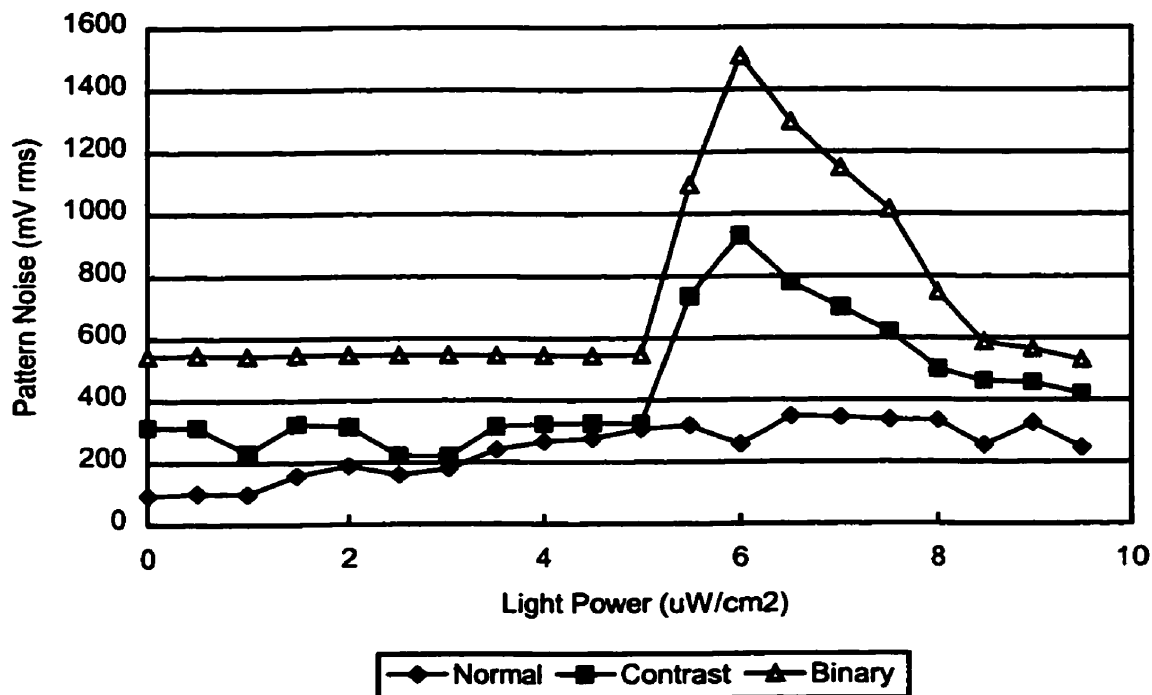


Figure 5.26. Mismatches in three output modes. Due to the lithographical mismatches, appearances of images in contrast stretched mode and binary mode are affected by pattern noise.

goes deeper into the triode region, and the amplification gets smaller. The pattern noise decreases with the decreased amplification. Thus, it is concluded that the effects of V_t and lithographic mismatches on contrast-stretched and binary mode images become more significant than for normal mode operation.

5.3.6. Summary and Conclusions

In summary, this chapter describes the design of an in-pixel intensity transformer and its analysis, along with operational performance and experimental results. A simple intensity transformer is designed with controllability (programmability) of contrast and brightness. Each pixel for the in-pixel mapping function is designed with 3 transistors and demonstrated successfully. The intensity transformer, with common source amplifier structure, is so simple that pixel-level processing implementation is possible and feasible for on-chip integration with CMOS image sensors. Also, the design with a small number of transistors encourages the in-pixel integration.

Full dynamic range of the allowed voltage swing between V_{DD} and ground is still not used in the transformer, due to the necessary V_t difference for switching the driving transistor. During testing, we experienced some degradation of images in normal mode when all the three output modes (normal, contrast and binary mode) are turned on concurrently. This is attributed to cross-talk due to the short physical distance between the source follower and the transformer in the same pixel. In addition, switching in binary mode causes spikes in the contrast stretched mode for a short period of time. However, the effects on the appearance of the image are rarely noticed. Also, the effects of V_t and lithographic mismatches become important, especially for contrast stretched and binary mode images. Thus, it becomes necessary to have an on-chip pattern noise reduction mechanism (e.g. using a feedback system). Also the precision of our analog intensity transformer is questionable. There should be much more effort to improve the precision of this analog implementation (or to design a new implementation of higher precision) in order to use it practically.

Although the intensity transformer has reasonable controllability by altering the biasing voltage and the reference voltage, it does not have DSP-like full programmability for contrast and brightness adjustments. However, with tradeoffs in programmability and precision, the in-pixel intensity transformer is able to achieve low power and real time operation with pixel

processing, which is perfectly suitable for portable and wearable devices. Therefore, this design of the intensity transformer is for low-level image processing applications where low power and pixel level programmability, for further automated contrast optimization, are emphasized.

With design and fabrication of the in-pixel intensity transformer chip, the main purposes of this study, to explore the feasibility of on-chip in-pixel integration, and to gain a better understanding of the design issues needed for high quality contrast enhancement and automated contrast optimization, are successfully achieved. The main issues of designing in-pixel processing with CMOS APS are circuit density of processing elements (that easily erodes photosensitive area and thus reduces fill factor) and massive interconnections between neighboring pixels. In point operation where the massive interconnections are not required, and in-pixel processing is the best design methodology as long as the circuit density of the processing element does not take too much space in the pixel. In this particular design of in-pixel light intensity transformer with 0.35 μm technology, the performance of the chip was not optimal, due to poor photosensitivity and low precision of the processing circuit. As the technology scales down, retaining the same minimum size of pixel (4~5 μm), the space of in-pixel processing circuit will get higher (see Figure 4.5) and thus higher precision of the circuit will be achievable. Therefore, point operation in pixel processing is a promising research area for the near future.

Chapter VI

6. Local Operation

6.1. Introduction

Local operation is also called mask operation where each pixel is modified according to the values of the pixel's neighbors (typically using convolution masks). Local operation is spatially dependent on other pixels around the processed pixel: the final value of the processed pixel is affected by its neighboring pixels in the finite sized masks. The basic approach of the operation, convolution, is to sum products of the mask coefficients and the intensities of the pixels under the mask, at a specific location in the image. Denoting the gray levels of pixels under the mask (3 x 3 mask in this example) at any location by z_1, z_2, \dots, z_9 , the response of a linear mask is

$$R = w_1z_1 + w_2z_2 + \dots + w_9z_9$$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

The gray level of the pixel located at (x, y) is replaced by R if the center of the mask is at location (x, y) in the image. This computation is repeated as the mask is moved to the next pixel location in the image until all the pixels in the array are covered. Linear spatial filters are defined such that the final pixel value, R , can be computed as a weighted sum of convolution mask (non-linear filters cannot be implemented in this way). In the above case, 3x3 local mask was taken as an example for the convolution mask. However, the size of

convolution mask is not restricted to 3x3, but can be expanded to 5x5, 7x7, 9x9, and larger, depending on what precision the final value is required to have.

In aspects of on-chip integration with image sensors, local operations provide advantages of real time operation in image acquisition and processing, such as implementations of many practical linear spatial image filters and image enhancement algorithms. In addition, because the local operation is feasible for column structure implementations, low frequency processing is enabled and thus low power consumption is expected. However, since the local operations are based on a technique where local memory stores pixel values of the neighbours and processes them concurrently, implementation of the operation must contain some type of storage, potentially requiring a large design area. Applications of local operation typically use an iterative technique for advanced image enhancement algorithms, which cannot practically be implemented on-chip. Nevertheless, in the case of column structure implementations, local operation still has a limitation on design area because of the restricted column width, even with flexible design area in the vertical direction. Therefore, in order to overcome these limitations, careful designs and system plans are required for the on-chip implementations.

In order to understand the nature of local operation and to find a relationship between algorithms and architectural on-chip implementations, we will look into the main local operation algorithms, grouped according to similarity of functional processing. With many different local operations in image processing algorithms, these local operations are categorized into three major groups: smoothing filters, sharpening filters and edge detection filters. Examples of the local operation algorithms are described in [74], [75], [76], and summarized as follows.

6.1.1. Smoothing Filters

Smoothing filters (Figure 6.1) are used for blurring and noise reduction. Blurring removes small details from an image and bridges small gaps and holes in lines or curves, often used in preprocessing stages prior to object extraction and segmentation. In addition, blurring can reduce spatial noise by smearing pattern noise in an image. Noise reduction can be accomplished by blurring with a linear filter and also by nonlinear filtering. Smoothing filters consist of four main types, namely order filters, mean filters, order/mean filters and adaptive

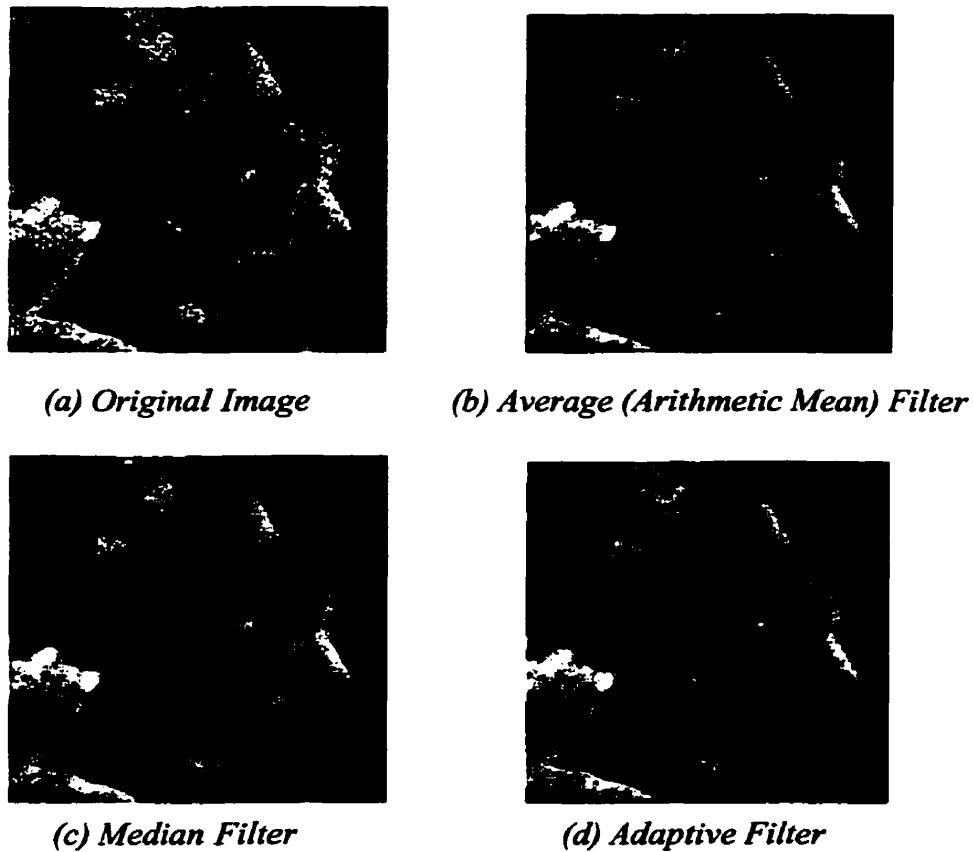


Figure 6.1. Matlab simulations on smoothing filters. The image of the flower is added and degraded with Gaussian noise. The size of local mask is 3 x 3.

filters. Each type of filter has its own characteristics and applications. The detailed description of each filter is omitted since it is out of this thesis' scope.

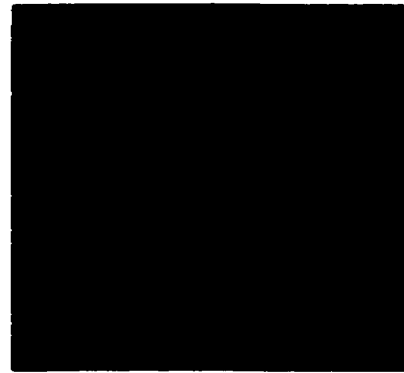
6.1.2. Sharpening Filters

The second type of local operation is sharpening filter. Image sharpening deals with enhancing detail information in an image, as shown in Figure 6.2. Because the high spatial frequency components of the image typically contain the detail information of the image, the sharpening filters should have some form of high-pass filtering. The detail information includes edges and boundaries of objects, which corresponds to image features that are spatially small. This information is visually important because it outlines object and feature, thus increasing the contrast of the image. The sharpening filters are again subdivided into two groups: high pass and high boost filters.

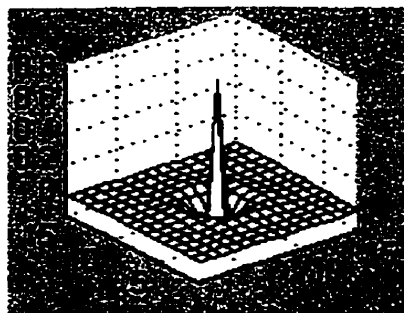
A highpass (sharpening) spatial filter contains positive coefficients near its centered pixel, and negative coefficients in the outer peripheral pixels of the local mask. However, because negative coefficients in the mask remains strongly in the final image output, this high-pass filtering for image enhancement typically requires an extra step of post-processing, such as histogram equalization, to display an acceptable image. High boost filters are more advanced than the highpass filters. With the highpass filtering, edges and high spatial frequency variances in the image will get enhanced, but a large portion of the visual information of the image is lost because the filter attenuates low spatial frequency components even though they are important for the appearance of the final image. The high boost filter solves this problem by adding low frequency offset to the filter function.



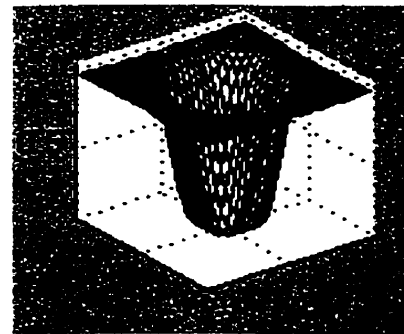
(a) Original image before the sharpening



(b) Sharpened image



(c) Spatial coefficients response



(d) Frequency response

Figure 6.2. Matlab simulations on sharpening filters. The size of local mask is 3 x 3.

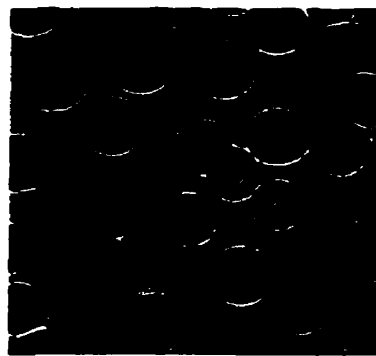
6.1.3. Derivative Filters (edge detection)

Opposite to integration that is analogous to averaging or smoothing, differentiation can be expected to sharpen an image extremely, leaving only boundary lines and edges of the objects. This is an extreme case of high pass filters. The most common methods of differentiation in image processing applications are first difference, gradient and laplacian operator whose Matlab software simulated images are shown in Figure 6.3. The difference filter is the simplest form of the differentiation with subtracting adjacent pixels from the centered pixel in different directions. The gradient filters represent the gradients of the neighboring pixels (image differentiation) in forms of matrices. Such gradient approaches and their mask implementations are represented with various methods: Roberts, Prewitt, Sobel, Kirsch and Robinson. Laplacian is another differentiation method for edge detection. The Laplacian of an image is a second-order derivative of 2D function, which enhances abrupt changes and edges in the image.

6.2. Proposed Structure for Local Operation

In the previous section, local operations are categorized by processing characteristic, into three different types: smoothing filters, sharpening filters and edge detection filters. The operation of these processes is based on a local (typically convolution) mask. The difference between these local filters is the different coefficient values used for the mask and the different sizes of the mask. Depending on the coefficients of the convolution mask, the operation can be smoothing filters, sharpening filters or even edge detection filters. Therefore, in terms of on-chip implementation architecture, it is convenient to divide the local operation by the size of the local masks: 3x3 local mask (the smallest mask size and the simplest for on-chip implementation) and bigger than 3x3 mask.

First, it is better to have a good understanding of the types of local mask in terms of size and connectivity. The local masks can have different sizes such as 3x3, 5x5, 7x7 and so on, of which the center pixel is the processed pixel being affected by the neighbors in the mask. Because the processed pixel is at the center of the mask, the size of the mask goes with odd numbers. The masks do not have to be square, but they are typically squares because of the simplicity of the design and the operation. Figure 6.4 shows different sizes of the local mask with shaded pixel at the center. Typically, as the size of the mask increases, the effect of the



(a) Original image



(b) Roberts



(c) Sobel



(d) Prewitt



(e) Laplacian

Figure 6.3. Matlab simulations on edge detection filters. The different edge detection algorithm vectors produce different effects on appearance of an image. The size of local mask is 3x3.

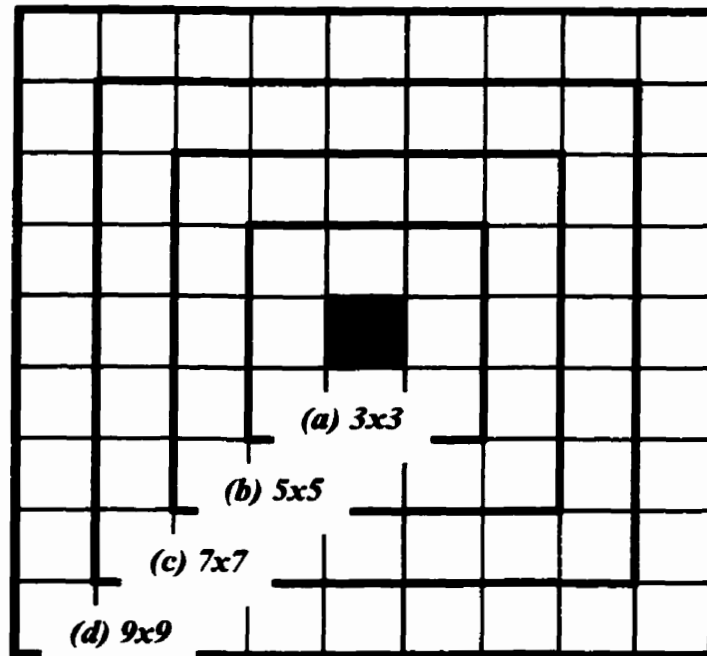
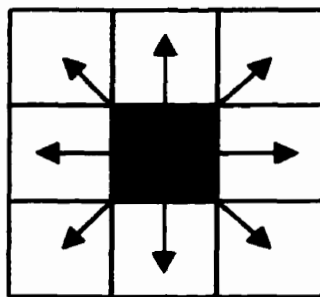
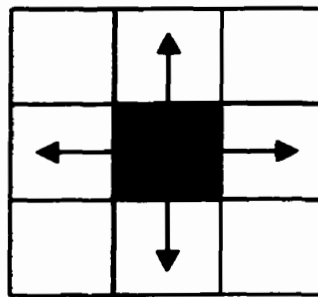


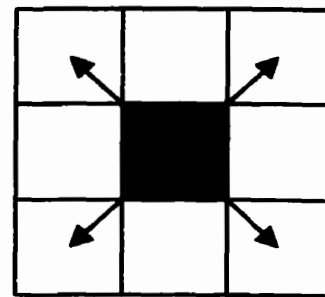
Figure 6.4. Local masks with different sizes.



(a) 8-connectivity



(b) 4-connectivity in cross



(c) 4-connectivity in diagonal

Figure 6.5. Local masks with different connectivity.

processing on the image is more apparent. As a matter of fact, the image quality becomes better with the larger masks under a given operation. However, due to the limited design area, long processing time and complexity of the design, the implementation of the large masks is often impractical.

Connectivity of the local mask refers to the way in which the central pixel is connected to its neighboring pixels. The centered pixel, in a 3x3 mask, has eight possible neighbors: two horizontal neighbors, two vertical neighbors, and four diagonal neighbors. As shown in Figure 6.5, we can define three different connectivity: (1) 8 connectivity, (2) 4 connectivity-cross and (3) 4-connectivity-diagonal. Similar to the size of the mask, as the connectivity in the mask increases, the effect of the processing becomes more apparent and the image quality becomes improved.

When these local operations are integrated with CMOS image sensors on a single chip, these characteristics of the processing mask (size of the mask and interconnectivity) should be taken into consideration for system-level architecture and circuit designs because these characteristics are directly related to design complexity and chip area. Here, we study on-chip implementations for the local operation. The implementations of different sized local masks (3x3 masks and larger masks) are investigated at different implementation levels of on-chip processing: pixel, column, chip and frame memory levels. General structural implementations are discussed and different architectural integrations for each operational type are compared with its merits and drawbacks.

6.2.1. Implementations of 3x3 Local Mask Filters

The implementation of 3x3 local masks can be performed at the pixel, column, chip and frame memory levels. Because 3x3 local operations are the smallest possible masks, their implementation is relatively easy and there is a relatively large choice of architectural implementations. However, due to interconnections between neighboring pixels and to complexity of processing elements, there are many challenges and difficulties in design and implementation of even such a simple mask. Here, it is assumed that these local masks have full connectivity to every neighboring pixel, giving 8-connectivity in a 3x3 mask.

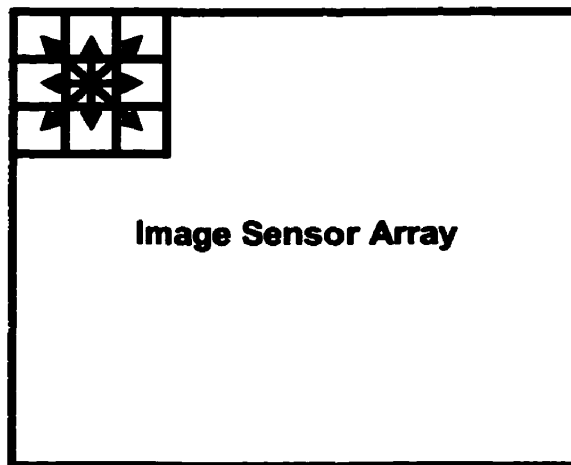


Figure 6.6. Pixel processing for 3x3 local mask operation.

First, implementation of a 3x3 local mask with in-pixel processing structures is an attractive design where each pixel has a photodetector and a processing element, connected to its neighboring pixels in the array, shown in Figure 6.6. The connections to the neighboring pixels are defined by the local mask. In a case of 8 connected neighbors, a photodetector has eight outputs to the processing elements of its neighboring pixels. In addition, a processing element of a pixel has eight inputs from photodetectors of its neighboring pixels. Therefore, an obvious disadvantage of pixel level implementation of local mask is the heavy connections among pixels and processing elements. Due to the interconnections, the pixel loses its fill factor. Not only the interconnections, but also the processing element and the storage take area in a pixel, thus the photosensitive area is further reduced (but microlens can overcome loss of fill factor). In cases when the storage is an analog memory, the leakage (charge retention time) of the memory should be considered carefully to assure that there is not much voltage drop in the memory. The larger memory (e.g. capacitor) is the longer the holding time is. However, large memory typically reduces the fill factor. Also, there should be shield for the storage to block any incident light. Another disadvantage comes from the readout mechanism. Due to the concurrent processing on neighboring pixels, progressive scanning techniques of conventional CIS arrays will not work, unless each pixel has its own memory, which would increase pixel size significantly. Therefore, a new design of peripheral readout component may be required in the pixel processing implementation.

If intelligent connections among pixels and simple processing elements are developed, the pixel level implementation is very attractive due to parallel processing, of which advantages

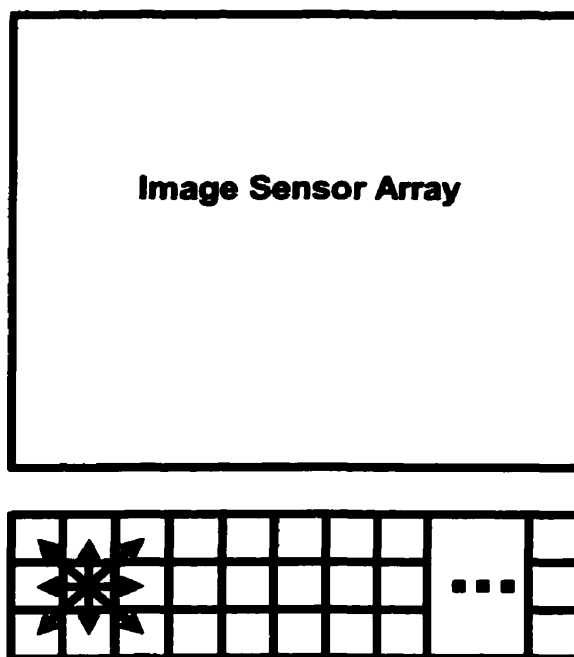


Figure 6.7. Column processing for 3x3 local mask operation.

include low frequency processing, low power consumption and adaptation to the local environment. However, since the pixel level implementation still faces severe limitations on pixel size, the feasibility for given applications should be carefully examined and planned.

The second method of 3x3 local mask implementation is based on column level processing structures with local memory. At the bottom of the imager array, three sets of linear arrays with local storage and processing elements are placed with the same number of the columns in the image array, as shown in Figure 6.7. Since the processing elements are separated from the pixels, a progressive scanning technique of conventional CIS array can be used here. In the progressive scanning method, when pixel values of the image sensor array are read out row by row, pixel values of one row are dumped into the first row of the processing array and stored until next image data come. When the next pixel values come, the previous values of the processing element array are shifted to the second row and then the third row. This repeats until all the rows of the image go through the processing array. Each time a new row is dumped, the operation of the local mask should be done before the transfer to the next row.

The column structure implementation of local operation offers a number of advantages such as column-parallel processing, flexibility of implementation in vertical direction, low frequency processing and low power consumption. Because the column level processing

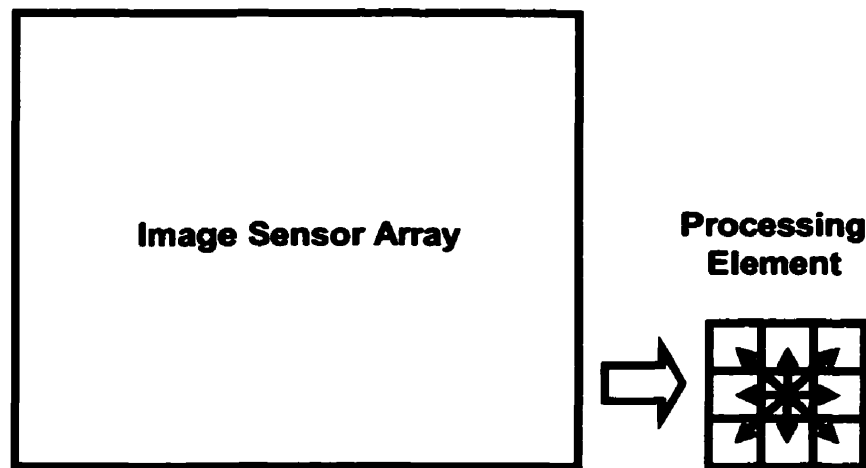


Figure 6.8. Chip processing for 3x3 local mask operation.

structure has added space for its design of processing elements in the vertical direction, the restrictions on design area and the number of transistors are relaxed, compared to the pixel level processing implementations. However, there are still limitations on the column width and thus, a careful design and implementation of processing elements is recommended.

Among the choices of the implementations, the simplest method of structural implementation is chip level implementation where a processing element is located at the end of the output channel in the image sensor array, as shown in Figure 6.8. Because this method does not have any limitations of pixel size nor column width, it has freedom of design area and therefore, it is feasible to use circuits of high complexity and functionality. Even with complex design of processing elements, the chip processing implementation is expected to have the smallest design area. However, it requires a very fast processing frequency, equal to the image data rate of the imager array (~10-100 MHz). High-speed readout typically causes high power consumption that is not desired in many applications, and increases the design complexity of the processing element to protect it from noise and crosstalk. Similar to pixel level implementations, a new scanning method other than progressive or interleaved technique is desired for the chip level implementation because of the concurrent readout and processing operation on neighboring pixels. Also this must be non-destructive because we need to reuse the pixel values.

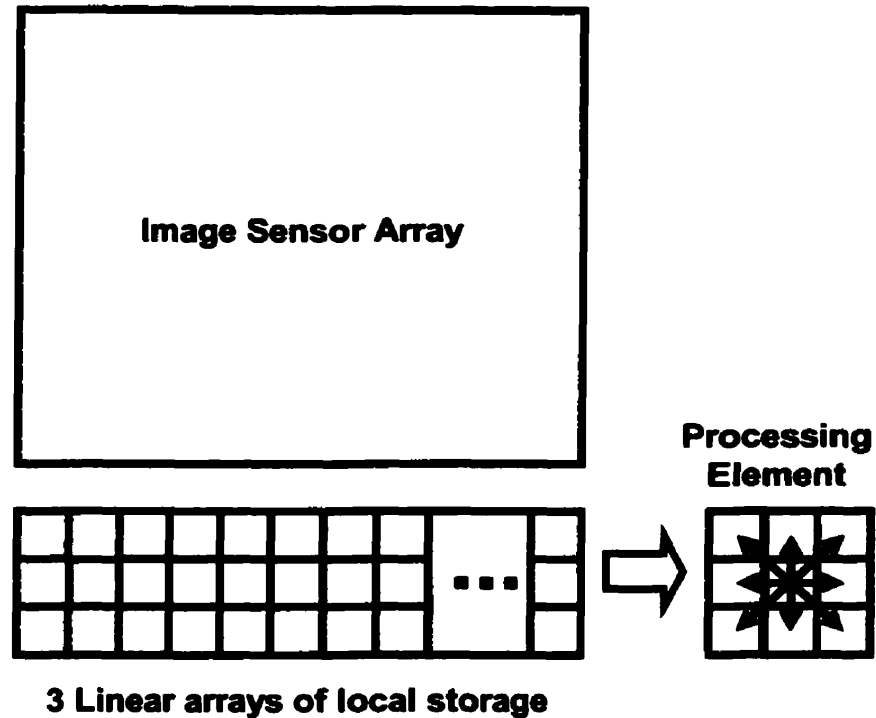


Figure 6.9. Hybrid method (column + chip processing) for 3x3 local mask operation.

A modified structure of chip level implementation is shown in Figure 6.9. Using a progressive scanning method of the column level structure, when the three rows of the local storage array contains valid image data from the image sensor array, image data of all three rows of the storages are shifted in series to the processing element. As the image data come from the local storage array, the processing element operates on the image at very high speed (same as the data output rate or output sampling rate). Still, the method operates at very high speed with high power consumption, but a simple progressive scanning method can be used with a trade-off on a larger area of local storage array.

The last option for on-chip implementation is frame memory level structure, shown in Figure 6.10. All the pixel values of the image sensor array are shifted to the frame memory once photodetectors integrate incoming light and capture an image. Each pixel of the frame memory consists of storage and a processing element, very similar to the pixel processing implementation except that there are no photodetectors in the frame memory. Therefore, the pixels of image sensor array do not lose any fill factor for the processing elements and storages. With the gain of fill factor in the photodetector pixel, the overall chip size increases

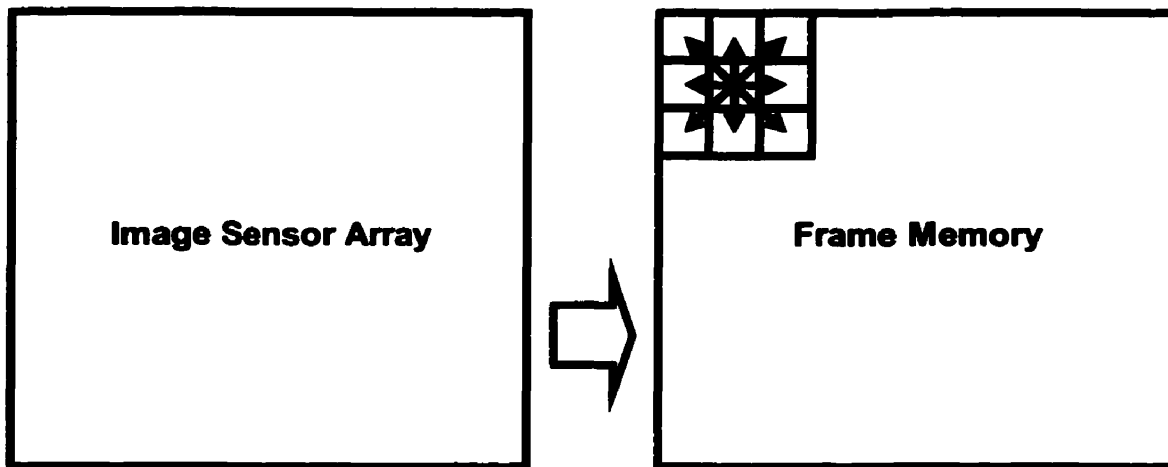


Figure 6.10. Frame memory processing for 3x3 local mask operation.

with frame memory and interface circuits, thus increasing fabrication cost. Because images captured by the image sensor array always go to the frame memory for the local processing, the output images of the chip experience latency of one image frame: the present output of the chip is captured one frame before. Also, because the sensor array is not centered in the package, it needs a special care to align the lens with the package. Despite the high fabrication cost and complexity of memory design, the implementation with the frame memory has the potential advantages of parallel processing, low processing power consumption and flexibility of processing circuit design.

We have seen different architectural implementations for local operation with 3x3 local masks. Each type of implementation has its own advantages and disadvantages. After careful investigation of these implementations, we recommend the column level structure for the implementation method for 3x3 local mask operation because of column-parallel processing, low power consumption, and feasibility of implementation. Currently pixel-level implementation is less feasible, from a practical point of view, due to its extensive interconnections and severe increase of pixel size by the processing element and storage. However, when the CMOS technology scales down further, in-pixel processing may become a practical implementation in the near future. Chip level processing and frame memory

implementations lose their interest in design due to their probable high power consumption and complexity of circuit design.

6.2.2. Implementation of Bigger Masks than 3x3

The on-chip implementation of masks bigger than 3x3 such as 5x5 (24 interconnects), 7x7 (48 interconnects), 9x9 (80 interconnects) and even larger, is very difficult. Simply because of the mask size and the large number of routings, the implementation of these masks requires extreme caution on interconnection routings between neighboring pixels. Pixel, column and frame memory implementations, where, in some ways, limitations on the design area exist, are not suitable for these masks. Even chip processing implementation is not a good choice because of its complicated scanning method. Therefore, there should be some modifications on these implementation methods in order to accomplish the design of the larger masks.

One possible design for the larger masks is a hybrid method, combining column and chip level implementations, introduced in the previous chapter (see Figure 6.9). Because a processing element is located at chip level (one processing element per chip), there are no limitations on design space for the processing elements. In addition, because the three or more linear arrays of local storages (the number of linear arrays is equal to the mask size) are used, a conventional progressive scanning technique can be applied for the readout, reducing design complexity of peripheral readout circuits. However, since the processing of operation is done at chip level, high processing speed (equivalent to the pixel rate) at the processing element is still required. This processing speed eventually determines the data output rate. The high processing speed also consumes high power, which is the main trade-off of the chip processing implementation.

Another implementation method for the larger masks is a pipelined structure. The pipelined structure is based on a concept that some 2 dimensional matrices ($N \times N$) can be represented as products of two linear arrays (product of a linear ($N \times 1$) and a linear array ($1 \times N$)) if the matrix is separable, as shown in Figure 6.11. The computation of the product on linear arrays is relatively easier than that of the 2 dimensional arrays (Figure 6.12) where a 1x3 linear array computation is done with pipelined structure. This computation method may seem to be trivial, but when the mask size gets bigger than 5x5, this method will be highly effective.

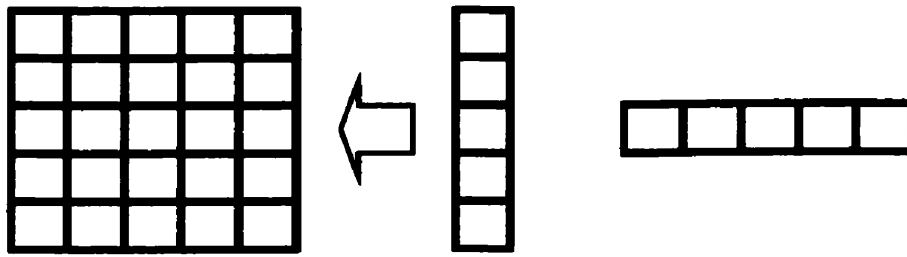


Figure 6.11. Concept of pipelined local masking. A 2-dimensional matrix can be realized by the product of two 1-dimensional (linear) arrays.

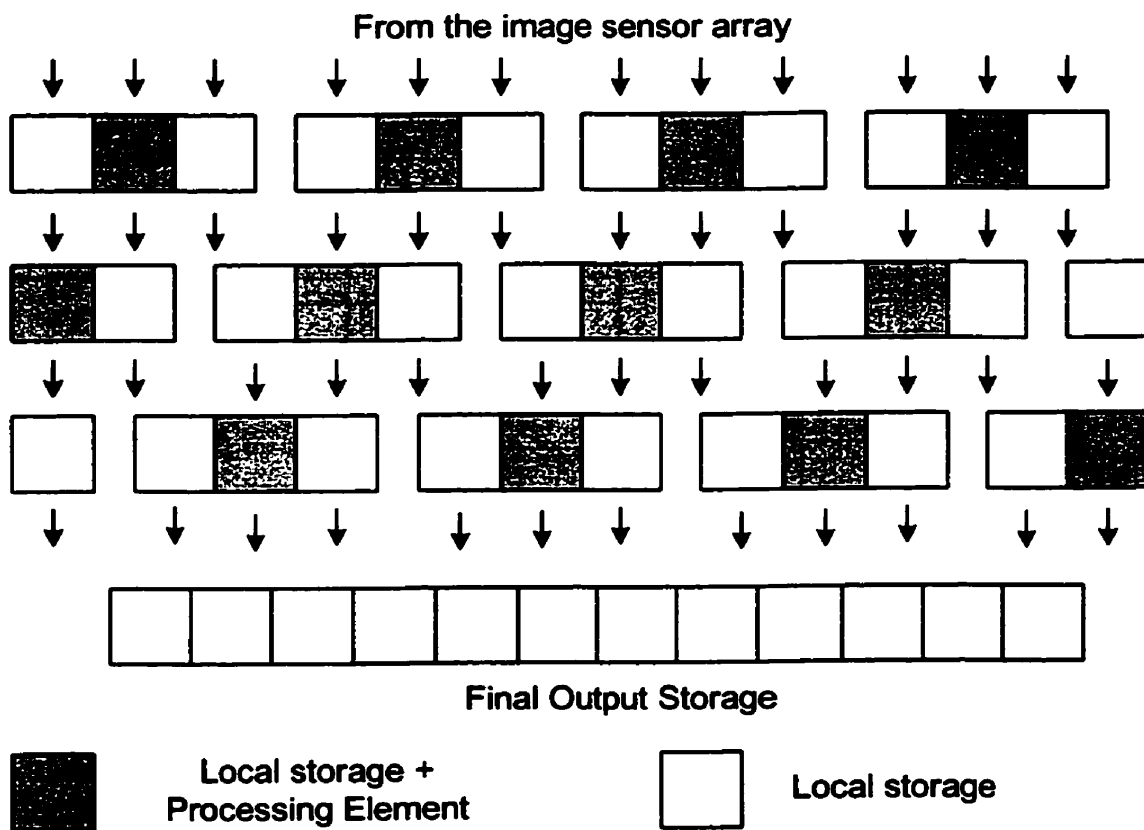


Figure 6.12. Basic structure of pipelined implementation for large local masks.

With the pipelined computation, column level implementation is possible for the larger masks, hence allowing a slow processing frequency and low power consumption.

Because a whole row of the array is computed at a same time, N different linear arrays are needed to compute the product of one linear array, shown in Figure 6.12. After this operation, another similar computation should be done for a horizontal linear array in order to complete the 2 dimensional matrices. Therefore, the circuit design for the computations gets complex and consumes a large area. In addition, the coefficients of the product matrices of two linear arrays are correlated, and thus any changes on a coefficient of the product matrix may affect the other coefficients in the matrix. Because the coefficients of the local masks should be not correlated, but independent form each other, the pipelined structure has limitations on contents of the coefficients. Therefore, matrix must be separable for this pipelined structure.

Although the pipelined structure can take advantage of column processing implementation, providing column parallel processing and low power, its complexity of computations, difficulty in input controls, and large design area limit the use for practical designs. Therefore, for the design of the large convolution masks (larger than 3×3 mask), the hybrid design with column and chip level implementation is highly recommended for its relatively simple design and easy operational control, at the price of high power. However, these designs do not have flexibility on the mask size; the mask size is predetermined and pre-fixed before the chip fabrication. Also, iterative operations cannot be implemented on chip, thus limiting its applications to low level preprocessing. The general description and comparison of local operation are summarized in Table 7.

6.3. Spatial On-Chip Binary Image Processing

6.3.1. Fundamental Operation in Binary Image Processing

Binary image processing is of special interest, since an image in binary format can be processed with very fast logical (Boolean) operators. Each gray level is represented by several bits. In a binary image, only one bit is assigned to each pixel ($B = 1$), implying two possible gray-level values, 0 and 1. These values might indicate the absence or presence of some image property in an associated gray-level image, where 1 indicates the presence of the property at that coordinate in the image, and 0 otherwise. This image property commonly includes the brightness at the pixel. However, more abstract properties such as presence or

	3x3 Masks	Larger than 3x3 Masks
Pixel Processing	Large number of neighbor connections and processing element area sacrifice fill factor and pixel size	Impractical to implement, due to a large number of interconnections to neighboring pixels
Column Processing	High flexibility of implementation with low power	Complex implementation due to large number of connections. If implemented, special architecture like pipelined structure is desired
Chip Processing	High speed and power Combined structure of column row storage and chip level PE with local storage or Chip level PE with a special image scanning method	High speed and power Combined structure of column row storage and chip level PE with local storage or Chip level PE with a special image scanning method
Frame Memory	High fill factor because neighbor connections and local storage are at outside of sensor array, but too much degradation on area, power and speed	High fill factor because neighbor connections and local storage are at outside of sensor array, but difficulty remains in interconnections between pixels

Table 7. General descriptions and comparisons of local operation implementations, for different sizes of local masks.

absence of certain objects might be indicated. Often, a binary image has been obtained by extracting information from a gray-level image, such as object location, object boundaries, the presence of some image property. Here, in this thesis, we restrict the binary image processing obtained by an image property of the brightness (light intensity at a pixel) for further operations.

A much broader and more powerful class of binary image processing operation is binary image morphology, also called morphological image processing. Morphology relates to the structure for forms of objects. Morphological filtering simplifies a binary image to assist the search for objects of interest. This is done by smoothing out object outlines, filling small holes, eliminating small projections, and using other similar techniques. Even though our focus of morphological image processing is for binary images, the extension of the concepts can be applied to gray-level images [42-52].

The two principal morphological operations are dilation and erosion. Dilation expands objects, thus potentially filling in small holes and connecting disjoint objects. Erosion shrinks objects by etching away (eroding) their boundaries. These operations can be varied for an application by the proper selection of the structuring element, which defines the neighbors in the operations, and thus determines exactly how the objects will be dilated or eroded. The neighborhood for a dilation or erosion operation can be of arbitrary shape and size (it can be 4-connected or 8-connected, or even with different sizes of the structuring elements of 3x3, 5x5 or larger). A structuring element is a matrix consisting of only 0's and 1's. The center pixel in the structuring element represents the pixel of interest, while the elements in the matrix that are on (= 1) define the neighborhood.

The dilation and erosion processes are performed by laying the structuring element on the image and sliding it across the image in a manner similar to convolution. The difference between dilation and erosion is the operation performed. The algorithms of these processes are as follows.

- For dilation, if the center of the structuring element coincides with a '1' in the image, or if any pixel in the input pixel's neighborhood (defined by '1' in the structuring element) is on ('1'), the output pixel is on. Otherwise, the output pixel is off ('0').

- For erosion, if the center of the structuring element coincides with a '1' in the image and if all pixels in the input pixel's neighborhood are on ('1'), the output pixel is on. Otherwise, the output pixel is off ('0').

With a dilation operation, all the '1' pixels in the original image will be retained, any boundaries will be expanded, and small holes will be filled. The erosion process is similar to dilation, but it turns pixels to '0', not '1'. All the boundaries of the objects are etched away, and some small objects disappear from the original image. A simulated example is shown in Figure 6.13. After an original gray-level image of stars (Figure 6.13 (a)) is extracted to a binary image (Figure 6.13 (b)), erosion and dilation operations are applied to the binary image. In Figure 6.13 (c), the erosion makes white spots of the stars smaller and even relatively very small stars to disappear from the original image of Figure 6.13 (a). Meanwhile, the dilation of Figure 6.13 (d) makes all the stars larger than the original size.

There are many other types of morphological operation in addition to dilation and erosion. However, many of these operations are just modified forms of dilation or erosion, or combinations of dilation and erosion. The most useful operations for morphological filtering are called opening and closing. Opening consists of an erosion operation followed by dilation with the same structuring element. It can be used to eliminate all pixels in regions that are too small to contain the structuring element while keeping large objects the same sizes, shown in Figure 6.13 (e). A related operation, closing, is the reverse of the opening, consisting of dilation followed by erosion. It can be used to fill in holes and small gaps.

Another interesting and useful operation is to determine the perimeter pixels of the objects in a binary image. This perimeter detection is quite similar to edge detection in gray-level images, but with simpler computations. A similar processing as dilation or erosion is performed on the perimeter detection: we lay the structuring element on the image and slide it across the image. The difference is in the operation with the structuring element. As shown in Figure 6.13 (f), a pixel is considered a perimeter pixel if it satisfies both of these criteria:

- It is an on (=1) pixel
- One (or more) of the pixels in its neighborhood is off (=0)

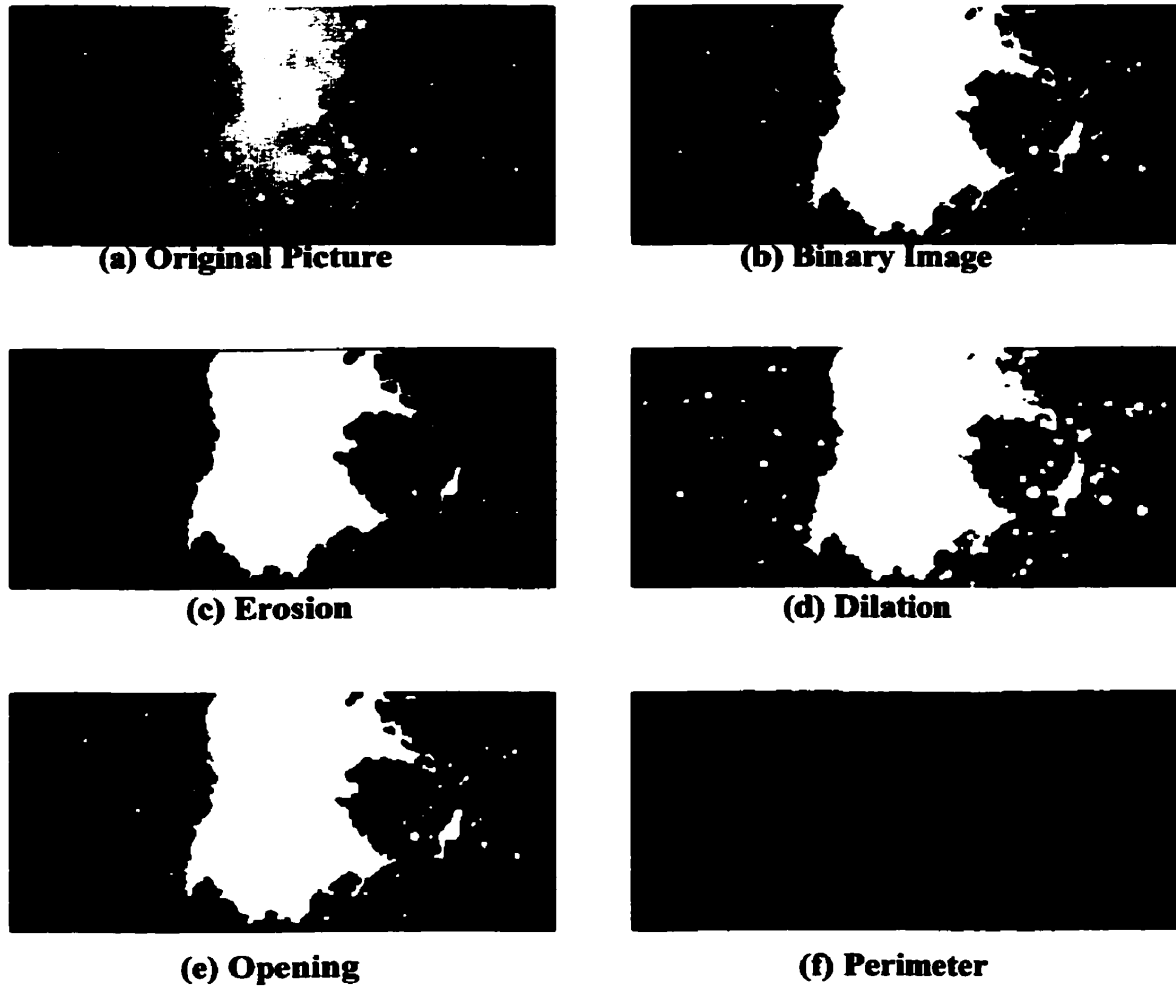


Figure 6.13. Binary Image Processing with various functionalities of erosion, dilation, opening and perimeter detection.

At first glance, perimeter detection may seem trivial, since the perimeter points can be simply defined as the transition from 1 to 0 (and vice versa). However, perimeter detection is quite useful and powerful, particularly for image segmentation and pattern recognition.

Because of Boolean operators and the simplicity of their circuit design, on-chip implementation of binary image processing is relatively straightforward. Here, we try to implement on-chip binary image processing with CIS as a demonstration of on-chip local operation. Although binary image processing is different from gray-level image processing, it has many similarities in operation, but with much less complicated operational computations. As a column processing implementation was proposed for local operation in the previous chapter, a column processing structure is implemented for the binary image processing. In addition to parallel processing and low power in the column processing implementations, the binary processing offers a number of other advantages.

- The processing element is relatively simple compared to other analog processing circuits that need high-levels of complexity in their design.
- The algorithm is powerful enough to be applicable to many low-level processing applications.
- There is no need for high accuracy ADC
- The local storage is relatively simple.

Here, we designed and implemented on-chip binary image processing with CIS to investigate the feasibility of the column structure implementation for local operation and its performance.

6.3.2. Previous Works on Binary Image Processing

The morphological analysis of black-and-white images was initiated by George Matheron in the late 1960s. His early work is described in the publication in 1975 of “Random Sets and Integral Geometry”[41]. Since 1975, the use of the fundamental morphological operation, absent of any significant statistical interpretation, has found a fast-growing field of applications. The developments of binary and morphological image processing algorithms were accelerated. These algorithms include noise reduction [48] [49], image sharpening [47], edge detection [44], image compression [57] and many other morphological filters

[42][43][45][46][50][51]. Large numbers of image processing software packages and hardware peripherals that include morphological operation such as dilation and erosion.

Hardware implementations of morphological processors include not only basic operation of dilation and erosion, but also more complicated image processing on binary images [52-57]. In addition to the binary image processors, there have been attempts to integrate binary image processing with image sensors, aiming for real-time operation of image capturing and processing. The on-chip binary image processing has variety of applications such as motion detection and analysis [59] [61], fingerprint sensing [60], and skeletonization [62]. On-chip binary processors with CMOS image sensors were also implemented for high programmability and flexibility of operation [58][63]. These on-chip binary processors are based on pixel processing implementations, which contain a photodetector and a binary processing element in the same pixel. Therefore, due to the high density of processing circuits in the pixel, only small sizes of arrays, less than 32x32, were implemented and therefore, the applications are restricted to low resolutions.

Because binary image processing uses a structuring element, which indicates relations between the center pixel and its neighboring pixels, the column processing structure is a good fit to the implementation of the binary image processing. Some previous studies have focused on the implementation of image processing, not only binary image processing but also general image signal processing, in column processing structures [64-70]. Also, the basic concepts of hybrid methods are also discussed: pipelined structure [71] as well as the combined structure of column and chip processing [72]. However, these are not for on-chip binary image processing.

Here, we designed and fabricated on-chip binary image processing with CMOS APS in column processing implementation.

6.3.3. Design of CMOS Active Pixel Sensor with On-Chip Binary Image Processing

We have designed and fabricated a prototype chip comprising a 64 x 64 array in standard 0.35 μm CMOS technology with 3.3 V power supply. A die photograph is shown in Figure 6.14. Each pixel is 30 μm square with n^+p photodiode, and it has a fill factor of 82%. The main objectives of this chip are (i) to explore the feasibility of local operation integrated with CMOS image sensors, (ii) to demonstrate the scalability of column processing

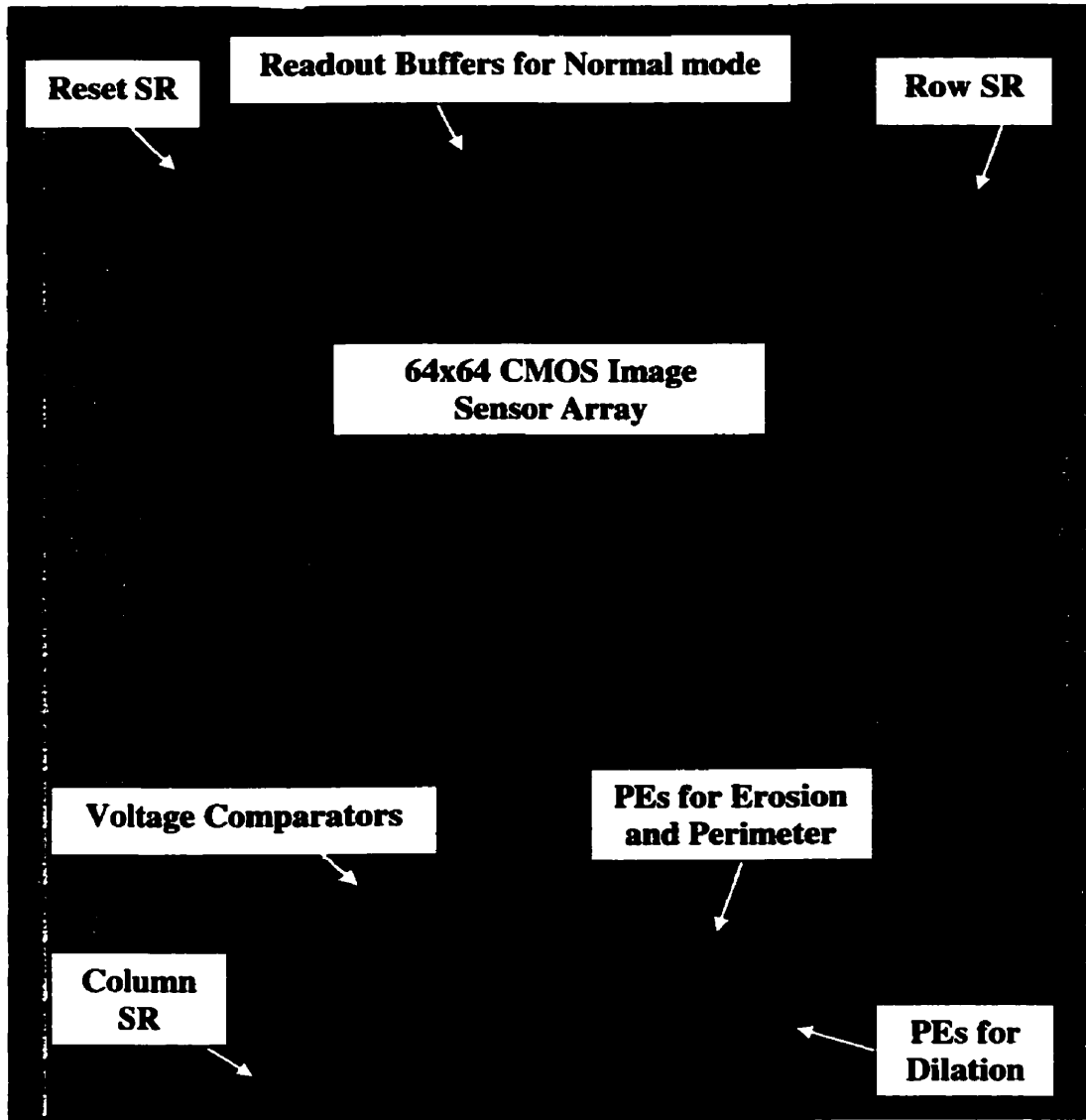


Figure 6.14. Die photograph of the prototype binary image processing chip. The total area is $3.2 \times 3.2 \text{ mm}^2$.

implementation with 0.35 μm technology, where processing elements are fit to the column pitch of the image sensor, (iii) to demonstrate on-chip binary image processing in real time mode, with low power consumption, (iv) to demonstrate feasibility of high resolution implementation, and (iv) to address the benefits and future research direction of on-chip local processing with CMOS APS.

The chip has one analog output and four different 1 bit digital outputs. The analog output is for raw images captured in the normal mode operation, without any signal modifications. The four 1 bit digital signals consist of: Binary image, erosion, opening and perimeter. The overall operational structure of the chip is shown in Figure 6.15. Since the binary image processing is performed by column-based processing components, the compact design of the processing circuits is easily found at the bottom of the chip (see the dark portion at the bottom of Figure 6.14). The chip consists of two main basic portions: one for normal mode operation at the top of the photodiode array, and the other for binary image processing at the bottom of the array. The normal mode of the chip follows the standard operation of CIS: the image is captured by photodiode with integration mode and the image data is transferred in parallel through source followers to the S/H's by row select shift registers, and then transmitted out in series by output buffers. Since basic operation and designs of photodiodes, shift registers and S/H's are discussed in the Chapter 2, the description of these components are omitted here.

In contrast, the binary image processing whose overall schematic is shown in Figure 6.16 consists of voltage comparators, local latches, processing elements, column storage and column readout circuits (shift registers). More detailed structure of the chip is shown in Figure 6.17. Once the image is captured by the same photodiode as used for the normal mode, it is buffered and stored in the S/H for the voltage comparators, the schematic of which is shown in Figure 6.18. The voltage comparator compares the image with the reference voltage to generate 1 bit binary signals (0 or 1), which are stored in the local latches and shifted row by row. Since the CIS array reads out the image data row by row, the shifting rate of the local latches should be the same as the clock rate of the row shift register for the CMOS imager array. This also means that all the necessary processing should be done within one cycle of this clock. In this particular design of binary image processing, 3x3 structuring element (local mask) is used to define the connectivity of neighboring pixels.

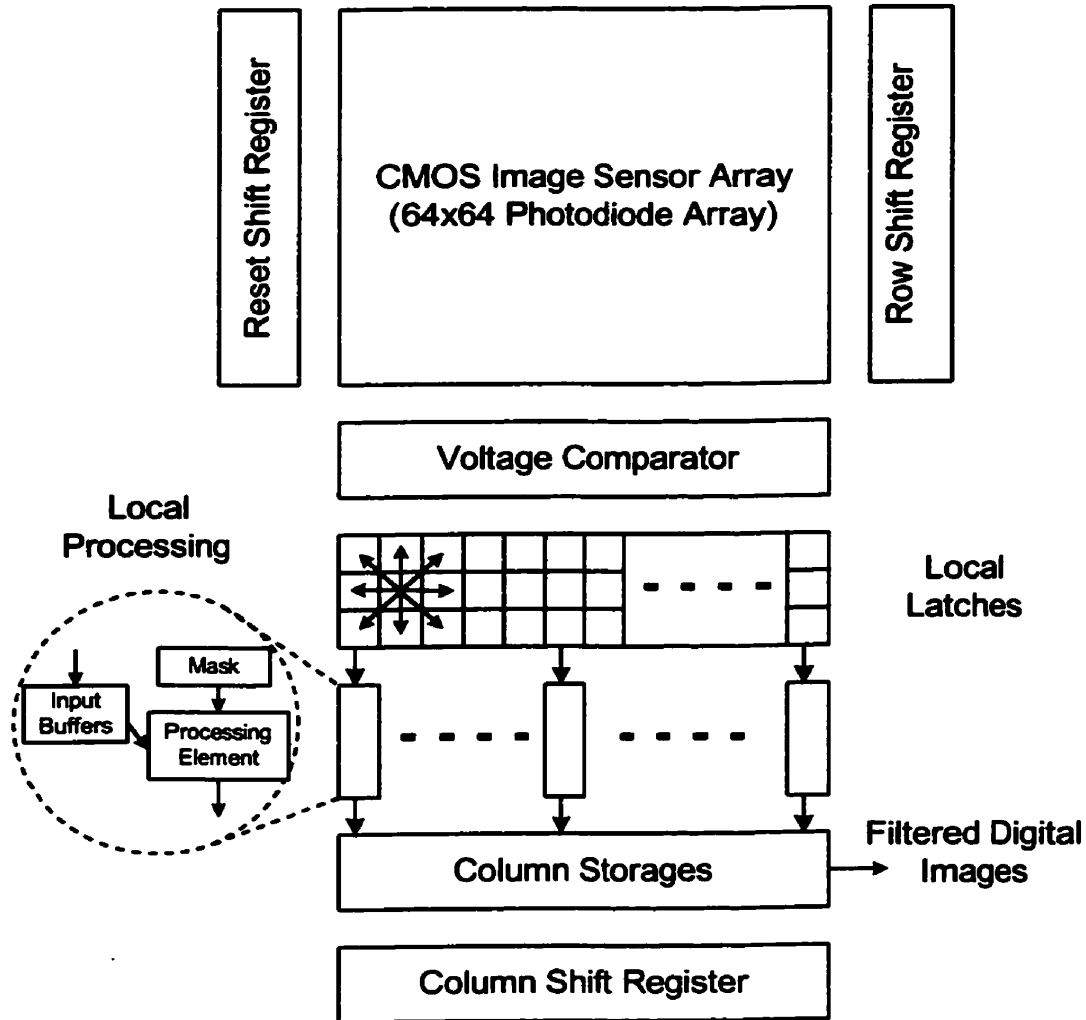


Figure 6.15. Overall Operational Structure of Binary Image Processing.

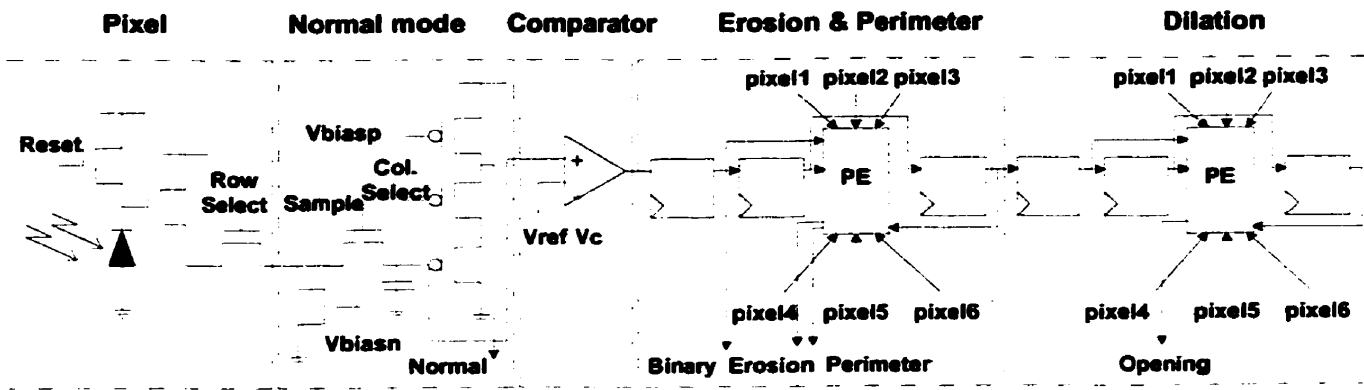


Figure 6.16. Schematic of major components in on-chip binary image processing.

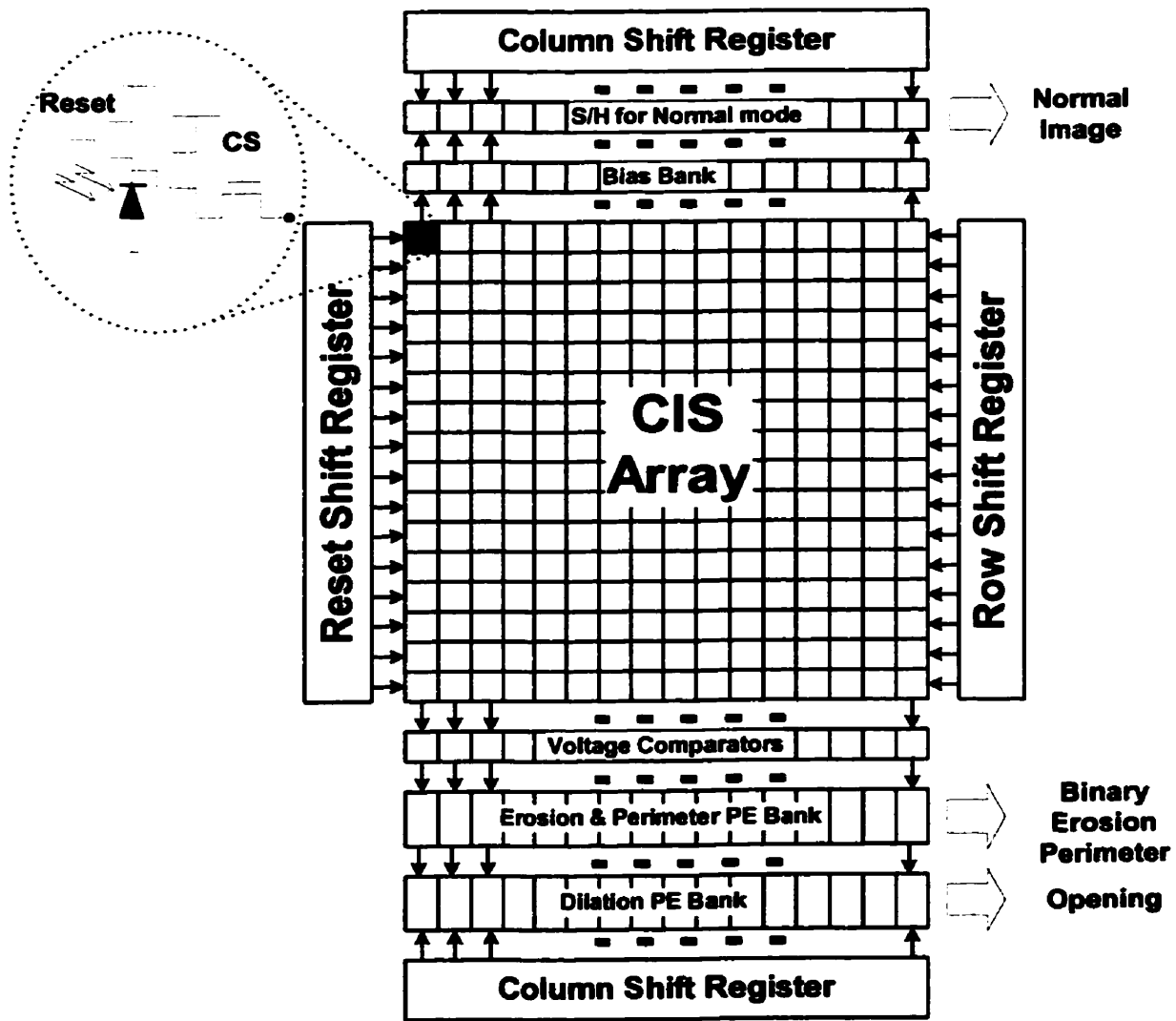


Figure 6.17. Detailed structure of On-chip Binary Image Processor with CMOS image sensor array.

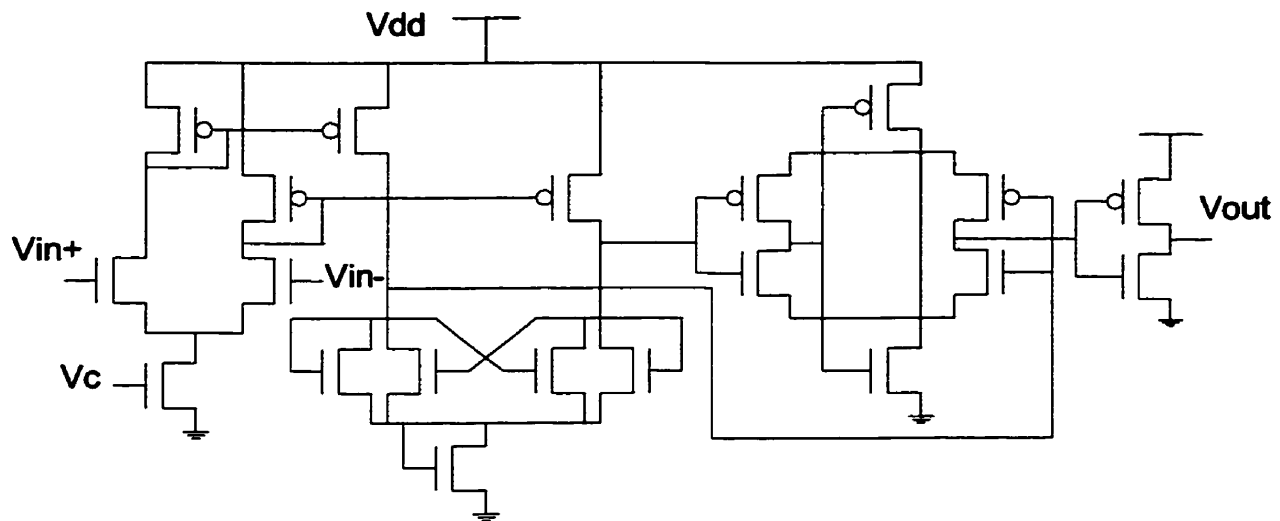
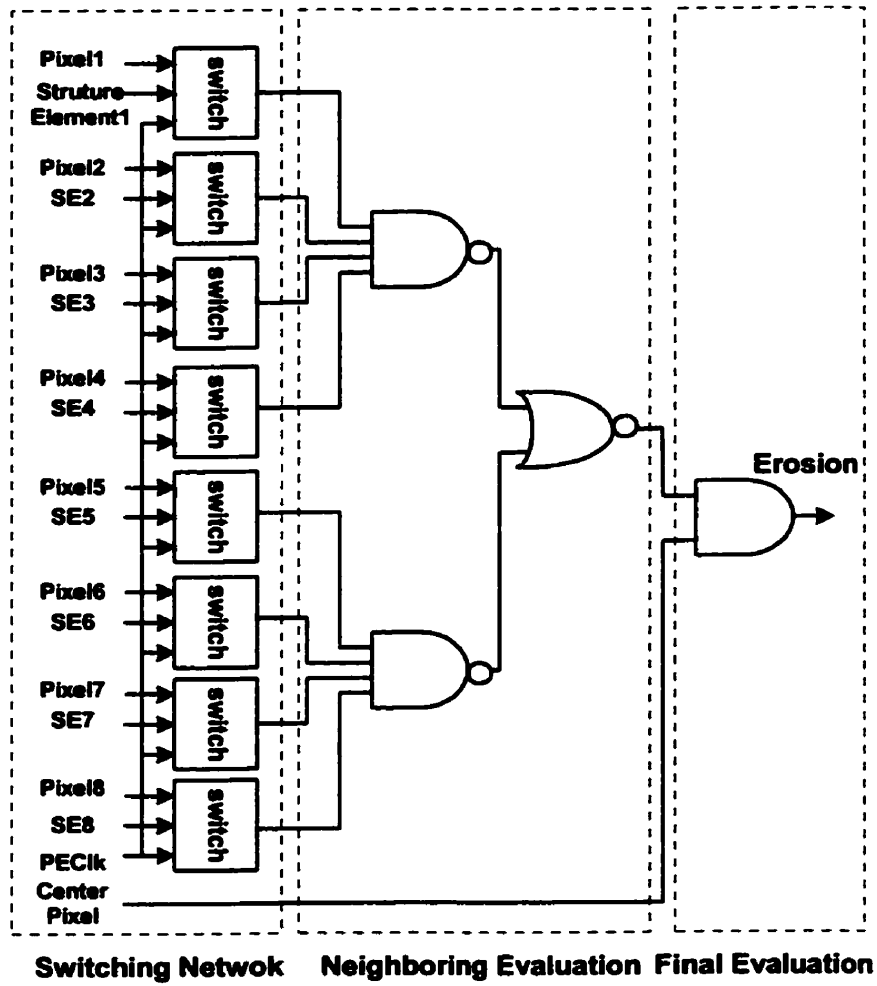


Figure 6.18. Schematic of Voltage Comparator [88].

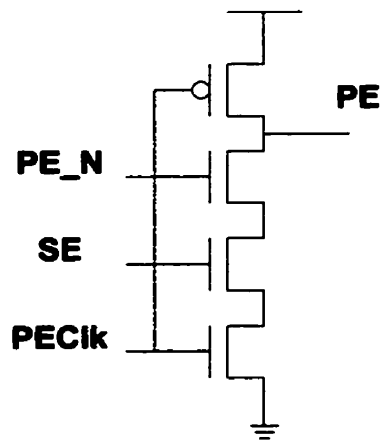
After the voltage comparator, there are three linear arrays of the local latches with the same number of columns as the imager array, followed by an array of processing elements. The circuit design of the processing element depends on which operation is implemented, such as erosion, dilation and perimeter detection. Since the operation of opening is based on the dilation after the erosion, there is another set of local latches and processing elements after the first erosion processing array which takes input images from the erosion and computes dilation operation on the eroded image, as shown in Figure 6.17.

Each output of the binary operation of binarization, erosion, perimeter detection and opening, needs its own output readout storage (column storage in Figure 6.17) for the serial data-out because different binary operations transmit the outputs independently through physically separate channels. Therefore, there are five different column storage elements in the chip, including the normal mode operation. In the chip, despite the different column storages, only two column readout controls (column shift registers) are used: one for the normal mode and the other for binary operation.

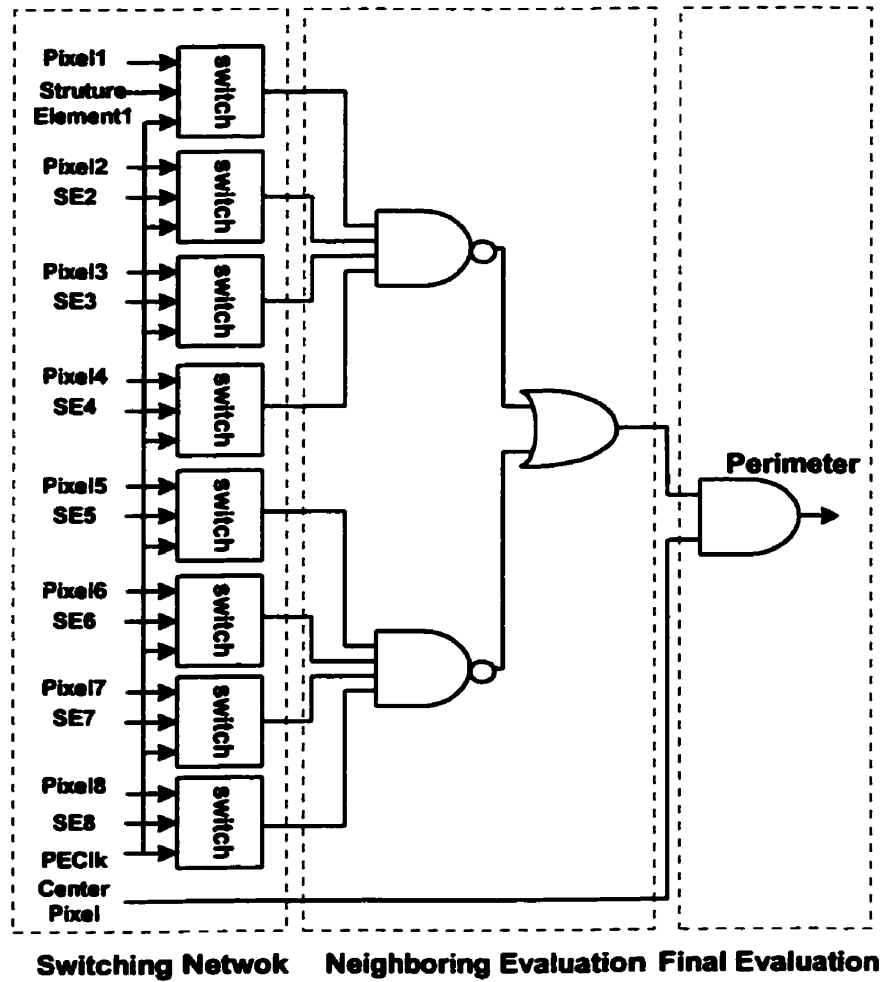
The algorithms of erosion, dilation and perimeter detection are implemented with logic (Boolean) gates. The algorithm of erosion is implemented with AND logic gate, shown in Figure 6.19 (a). Due to the neighborhood selection of the structuring element, the processing logic should be able to discriminate output value of the processing element. In a case where no processing elements are selected by the structuring element, the default values of the



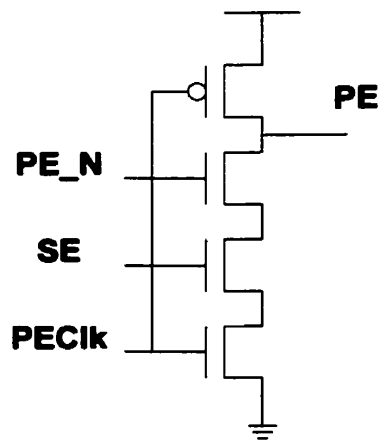
(a) Logic gates for erosion



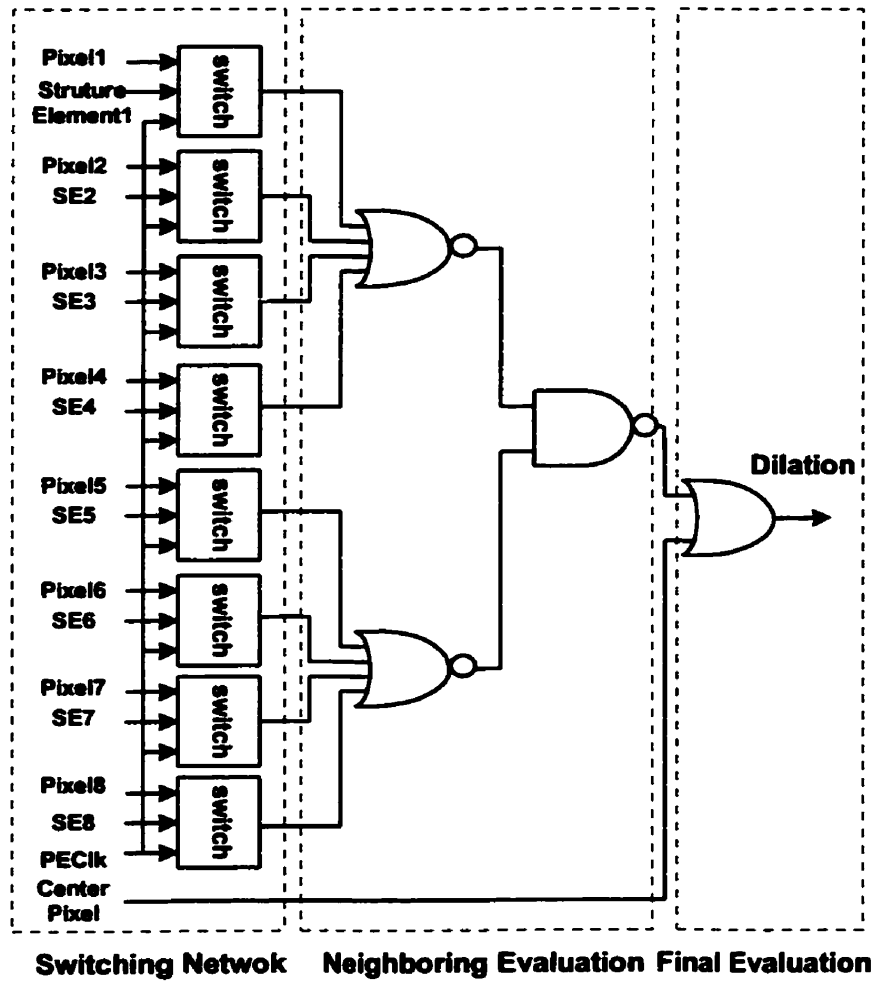
(b) Switch for erosion



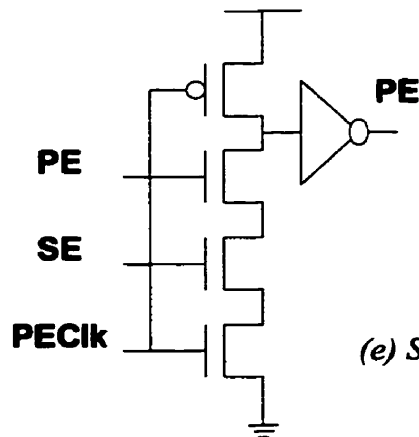
(c) Logic gates for perimeter detection



(d) Switch for perimeter detection



(e) Logic gates for dilation



(e) Switch for dilation

Figure 6.19. Logic design and schematics of the switches: (a) Logic gates for erosion, (b) Switch for erosion and perimeter detection, (c) Logic gates for perimeter detection, (d) Switch for perimeter detection, (e) Logic gates for dilation, (f) Switch for dilation.

inputs to the AND gate should be '1', thus leading to a special design of the switch, shown in Figure 6.19 (b). The switch selects the incoming processing element if the corresponding coefficient of the structuring element is high at a trigger of the PEClk. Otherwise, the output retains its default value of '1'. The design of the perimeter detection is similar to the erosion. The difference is in the logic gate for the neighboring pixels: for the erosion, AND gate is used and for the perimeter detection, NAND gate is used for the selection of the neighboring pixels, shown in Figure 6.19 (c). Also, since the default value of the switch is the same as that of the erosion, the same design of the switch is used for the perimeter detection.

The operation of the dilation is significantly different from the erosion and perimeter detection due to the OR logic operation. With a similar structural design, but different logic gates, the dilation consists of two OR gates and 8 different switches, shown in Figure 6.19 (d). Due to the different default value of the switch, the switch for the dilation is redesigned with some modifications from that of the erosion and perimeter detection, shown in Figure 6.19 (e).

6.3.4. Tests and Performance

Since the tests on the imager characteristics were dealt with frequently in the previous chapter, the detailed descriptions of the optical characteristics of this chip are not repeated. Only basic performance tests such as power consumption, frame rate and physical parameters are discussed here. Rather, the test focuses on the performance of binary operation and their effects on the appearance of the images.

Single Chip Characteristics and Normal Mode Operation

Here, we are able to verify operation successfully on image capture with some sample images, illustrated in Figure 6.20. As expected, the quality of the images captured by the chip is not high, partially due to the fact that the chip process technology is not optimized for image sensors, but instead for logic and memory. However, the subtraction of the white background image at the same illumination when the images are captured (see Figure 6.20 (c)) enhances the quality of the captured images by reducing pattern noise. Figure 6.20 (a) is a raw image and Figure 6.20 (b) is a pattern noise subtracted image. There are some noticeable differences in their image quality: the processed image is cleaner and has a higher

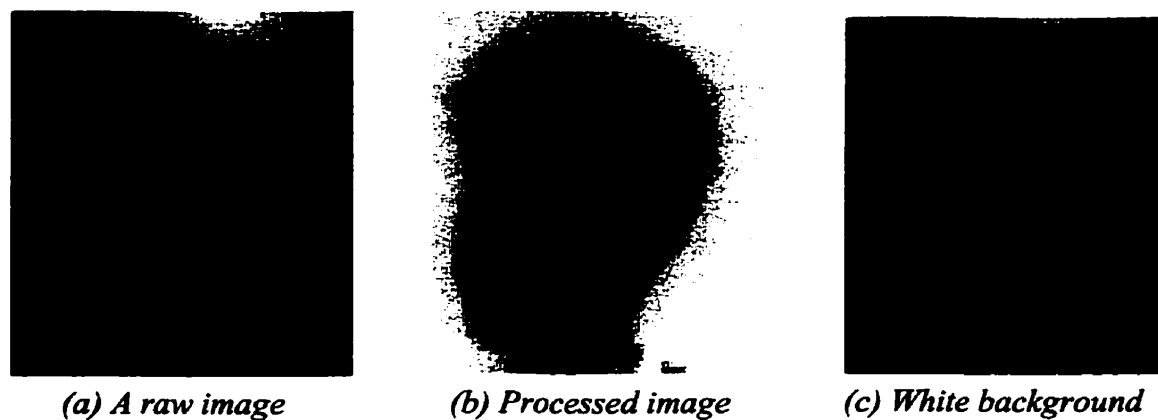


Figure 6.20. Real time images captured by the chip in normal mode operation. (a) Raw image, (b) Processed image after the subtraction of white background from the raw image, (c) White background image.

Technology	0.35 μm CMOS
V _{DD} Power Supply	3.3 V
Output	1 analog output and four 1 bit digital outputs
Package	68 PGA
Chip size	3204.5x3204.5 μm^2
Pixel size	30.8x30.8 μm^2
Format of array	64x64
Fill factor	82.65%
Maximum frame rate	100 KHz (24 frames/s)
Power	3.05 mA x 3.3 V = 10.065 mW at 50 KHz sampling rate
Light lux	Room light (150 ~ 200 lux)

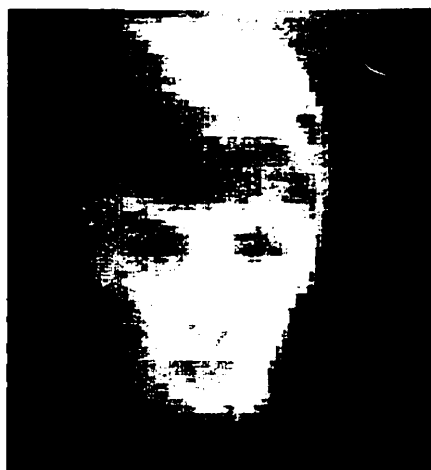
Table 8. Characteristics of single chip.

contrast. The characteristics of a single chip are summarized in Table 8, including basic characteristics of the chip.

The power consumption of the chip is about 10 mW at a frame rate of 12 frames/sec. This includes both the normal operation and binary image processing with 4 different outputs. The pixel size is $30.8 \times 30.8 \mu\text{m}^2$ which is relatively large. This is due to the interconnections and the processing elements in the columns. Since pre-built digital components such as flipflops and logic gates are used from a standard library, the minimum size of the design area ($\sim 20 \mu\text{m}$) cannot be changed. Custom layouts for these components will, however, optimize the column width and thus reduce the pixel size. In addition, the large pixel size with the large fill factor of the chip is necessary to increase the photosensitivity of the photodetectors, already degraded by the poor optical characteristics of the process technology.

This poor photosensitivity also affects the frame rate of the chip. As shown in Figure 6.21, as the frame rate increases, the quality of the images captured degrades rapidly. When the frame rate reaches around 100 KHz of data rate, it is noticeable that the image has become degraded with pattern noise and poor contrast. Therefore, the binary image processing typically operates at around 20 KHz and 50 KHz, which is relatively low compared to commercial high performance chip with a data rate around 20 ~ 40 MHz.

In normal mode, there is a defect on the image sensor array. Even under uniform illumination, the image displays a white half circle at the top of the image, shown in Figure 6.22 (a). Also, this white half circle appears on the binary image which have been filtered with a threshold. Figure 6.22 (b) shows an image of edges in the binary image, illustrating half-circle boundaries at its top of the image. This seems to be due to the cross talk and noise, which are generated by the normal mode readout circuits located at the top. It can be verified by observing a binary image in edge detection mode while the normal mode is turned off. When the normal mode is turned off, the edges (boundaries) of the half-circle cannot be found, shown in Figure 6.23. It is concluded that the readout peripheral circuits of the normal at the top of the array generate the unwanted defects on the normal operation. The optimization of the cross-talk (by putting a ground ring between the array and the readout circuits) is expected to eliminate this defect.



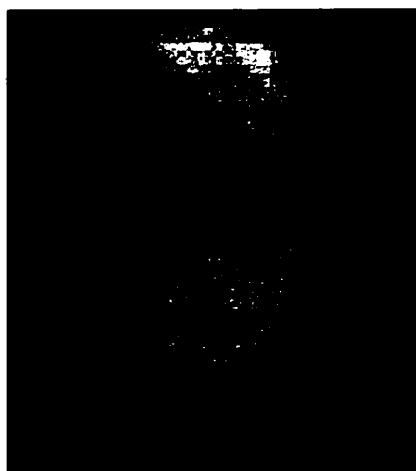
(a) 10 KHz



(b) 20 KHz



(c) 50 KHz



(d) 100 KHz

Figure 6.21. Effects of frame rate in normal mode operation.

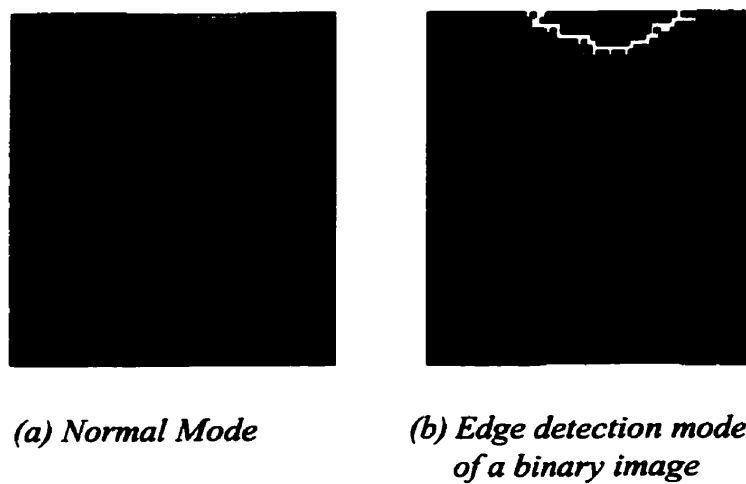


Figure 6.22. When both normal mode and binary operation are on, a defect of white spot can be found at the top of the image in both images.

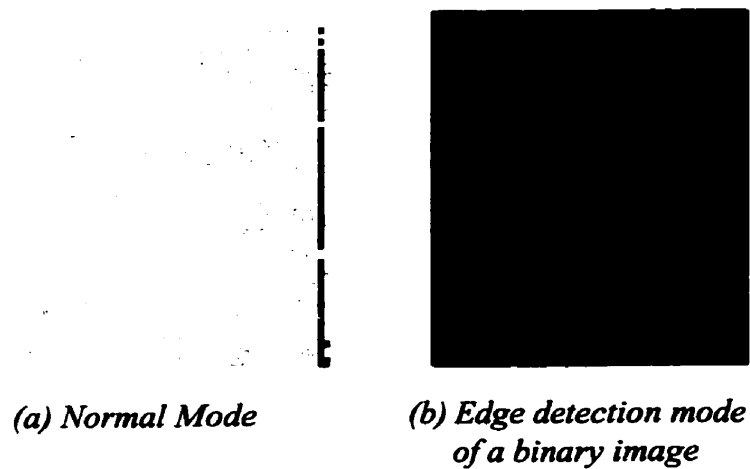


Figure 6.23. In processing only mode, the defect in Figure 6.2.3 disappears regardless of V_{ref} .

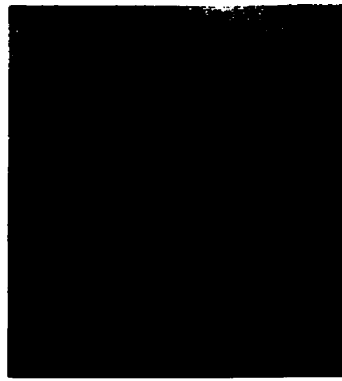
Operation of On-Chip Binary Image Processing

On-chip binary image processing consists of four different operations; thresholding (binarization), erosion, dilation (later combined with erosion, it becomes an operation of opening), and perimeter detection. Figure 6.24 shows sample images of these binary processing, captured by the chip in real time operation mode. Because the outputs could not be displayed at the same time with our testing equipment (we have only two probes), some of the images have time differences, although the chip outputs all images in parallel.

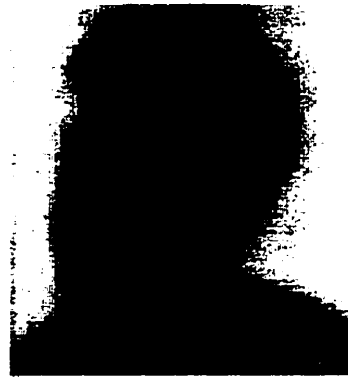
A good demonstration of the binary processing is also shown in Figure 6.25. After capturing the image, the raw image of Figure 6.25 (a) is sent to the voltage comparator to generate the binary image, Figure 6.25 (c). Through the first set of linear arrays for the local storage and processing elements, operation of erosion and perimeter detection are performed on the binary images. Through the operation of erosion, boundaries of objects are etched away and disappear. Large white spots are etched away, becoming smaller and some small spots disappear from the image, shown in Figure 6.25 (d). Another operation with the same local latches is the perimeter detection. In this particular design of my chip, the perimeter detection is applied to the binary images (see Figure 6.25 (e)), not on the images after the erosion. The last operation of the binary image processing is the opening that eliminates all pixels in regions that are too small to contain the structuring element. As shown in Figure 6.25 (f), the small white spots of the binary image, Figure 6.25 (c), were disappeared after the opening, but the original shapes are maintained for large spots. This process can be used in object discrimination and spatial noise reduction.

The performance of the binary image processing integrated on the chip should be independent of shape of the objects in the input image. Figure 6.26 shows the operation of the chip on different shapes of the input objects; circle, triangle and rectangle, demonstrating that the chip operates independent of the shape of the objects.

Interestingly, and perhaps obviously, the binary image processing is very dependent on the conversion from a raw image to a binary image. The conversion is accomplished by the voltage comparator; when the input voltage is lower than the reference voltage (bright image), the output is '1', otherwise, the output is '0'. As noted, choosing a proper reference voltage is



(a) Raw image



(b) Processed image



(c) Binary image



(d) Erosion



(e) Perimeters



(f) Opening

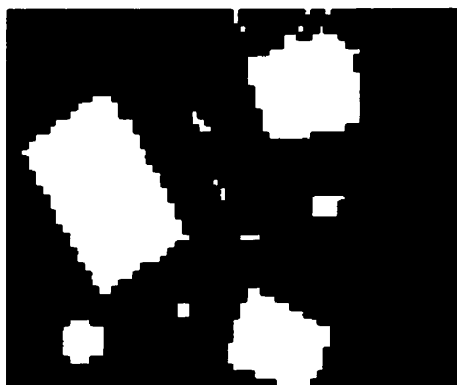
Figure 6.24. Sample images of CMOS Active pixel sensor with on-chip binary image processing.



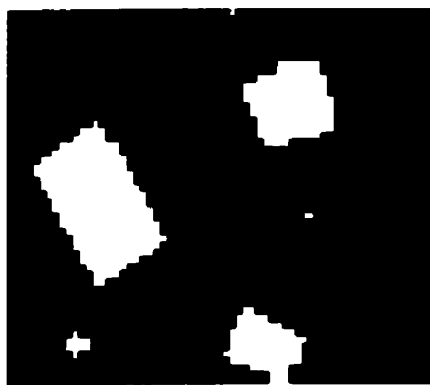
(a) Raw image



(b) Processed image



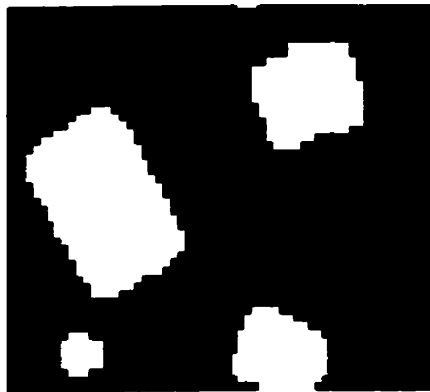
(c) Binary image



(d) Erosion



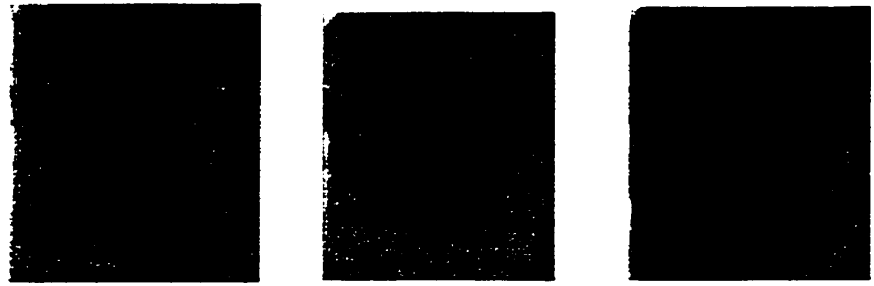
(e) Perimeters



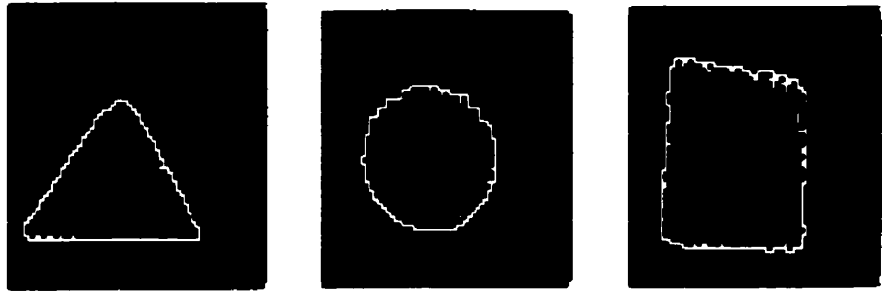
(f) Opening

Figure 6.25. Demonstrations of binary image processing. All the images are captured by the prototype chip in real time mode.

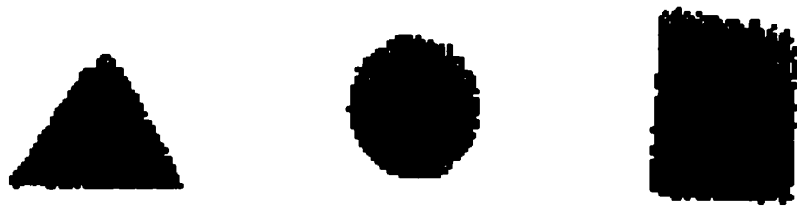
(a) Gray level image



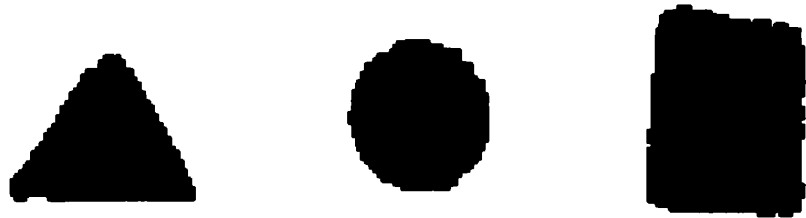
(b) Perimeters



(c) Binary image



(d) Erosion



(f) Opening

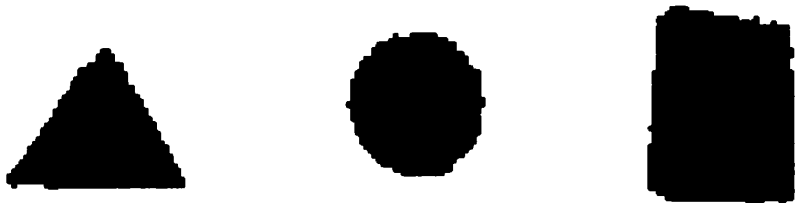
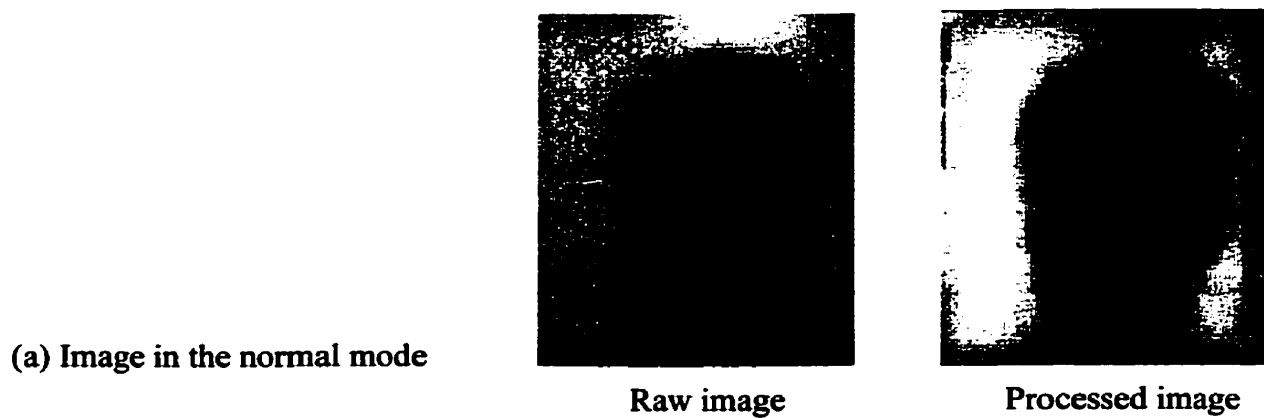


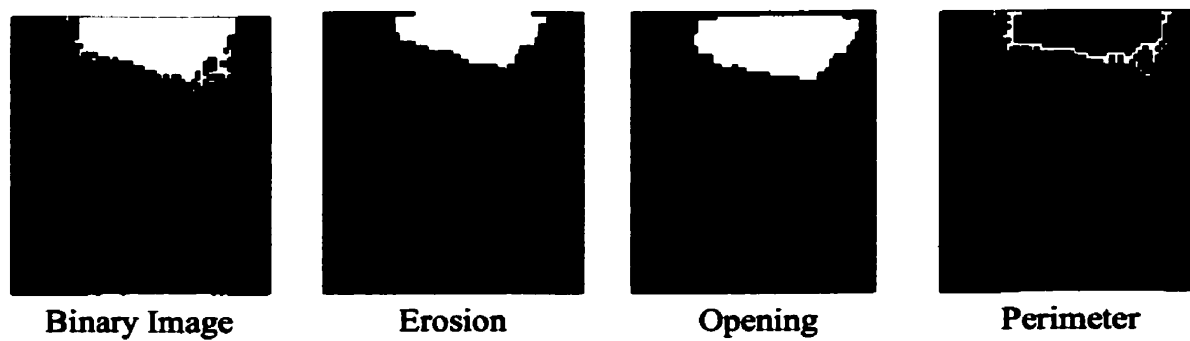
Figure 6.26. Independent operation of binary image processing from the shape of the objects.

quite an important process for the binary image processing. This processing is also called “thresholding” in the image processing field. Although there have been many studies and demonstrations [73] [74] [75] [94], thresholding is not an obvious and straightforward subject. Figure 6.27 illustrates the effects of the different reference voltages to the comparator on the binary operation. It demonstrates the importance of the reference voltage to the binary image processing. For example, the reference voltage of 1.56 V gives the best results on binary images and their operation, at this particular input image and under a particular illumination (environment). However, this reference voltage does not give the best results all the time, rather the most appropriate voltage should be chosen carefully for different environment and input images. When the reference voltage is low compared to the average pixel values of the input image, the output image mainly consists of ‘0’, corresponding to a black image, where objects cannot be recognized and boundaries of the objects are meaningless, as shown in Figure 6.27 (b). In contrast, as the reference voltage increases, the gray levels of more pixels become over the reference voltage, producing ‘1’ as their outputs in the binary image outputs. The actual shape of the face becomes more recognizable and the boundaries of the object become more reasonable. When the reference voltage gets too high, most of the gray levels are over the reference voltage, generating an almost white image (see Figure 6.27 (h)) for its binary output images. Also, the boundaries of the object become meaningless once again.

Another interesting test is based on the structuring element that defines the effect of the neighboring pixels on the final output value of the pixel. With the 3x3 structuring element in this chip, the coefficients (‘0’ or ‘1’) of the structuring element are controllable externally, which means that the connectivity of the neighboring pixels can be selected. Here, several different connectivities (structuring elements) are explored. Figure 6.28 shows a demonstration of the different structuring element with different connectivities on the binary operation. A structuring element of 3x3 local mask (see Figure 6.28 (b)) is applied to an original image of a triangle, Figure 6.28 (a). With different structuring element of different connectivity, each operation of binary image processing (perimeter detection, binary, erosion and opening) is applied to the original triangle image. Their output images of the binary operation are shown in Figure 6.28.



(b) $V_{ref} = 1.20 \text{ V}$



(c) $V_{ref} = 1.30 \text{ V}$



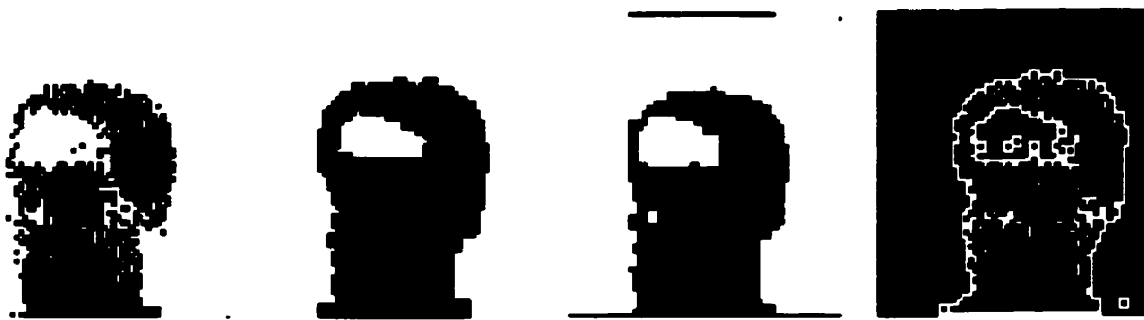
(d) $V_{ref} = 1.40 \text{ V}$



(e) $V_{ref} = 1.50 \text{ V}$



(f) $V_{ref} = 1.56 \text{ V}$



(g) $V_{ref} = 1.60 \text{ V}$



(h) $V_{ref} = 1.70 \text{ V}$



*Figure 6.27. The effects of reference voltage
(As changing V_{ref} , at $V_c = 0.55 \text{ V}$ optimal voltage for best image quality).*



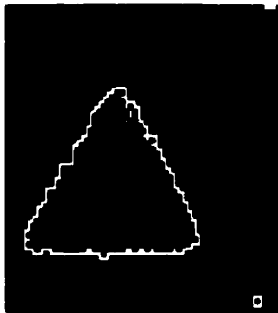
(a) Original Input Image

a	b	c
d	e	f
g	h	i

(b) Structuring Element

(c) 8-connected neighboring pixels

1	1	1
1	1	1
1	1	1



Perimeters



Binary image



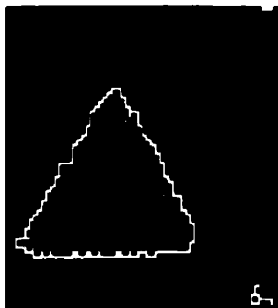
Erosion



Opening

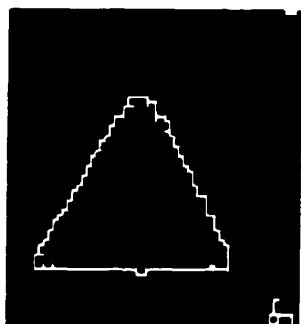
(d) Cross 4-connected neighboring pixels

0	1	0
1	1	1
0	1	0



(e) Diagonal 4-connected neighboring pixels

1	0	1
0	1	0
1	0	1



(f) Cross 4-connected for erosion and Diagonal 4-connected neighboring pixels for dilation

0	1	0
1	1	1
0	1	0

For Erosion

1	0	1
0	1	0
1	0	1

For Dilation

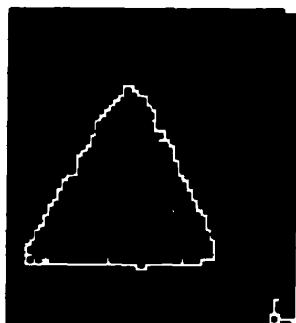


Figure 6.28. Connectivity: effects of different neighboring pixels of structuring element on binary image processing operation.

Interestingly, even with 8-connected and 4-connected neighboring pixels in the structuring elements, the images of the binary operation are not greatly affected. This is because the size of the structuring element is too small to generate significant impact on the output images. Therefore, the size of the structuring element will affect the output binary images considerably. Since the 3x3 structuring element in this chip is relatively small, the changes in its coefficients are negligible. When the size of the structuring element becomes larger to 5x5, 7x7 or larger ones, the selection of the coefficients will influence the appearance of the output image.

6.3.5. Summary and Conclusions

In this section, we have described a design for CMOS active pixel sensor with on-chip binary image processing and its analysis, along with operational performance and experimental results. We have explored the feasibility of local operation integrated with CMOS image sensors, concluding that column processing architecture is the best fit provided the interconnections to neighboring pixels are not excessive. As a demonstration, the operations of binary image processing (global thresholding, erosion, dilation and perimeter detection) are integrated on a single chip with CMOS image sensor array. The on-chip real-time operation allows image capturing and image processing in parallel, thus permitting low frequency processing circuits and reducing power consumption. The binary operation, with each PE implemented per column (also called column processing structure), is designed with digital storage and logic gates, and demonstrated successfully for its real-time operation with low power consumption.

In this particular design of on-chip binary image processing, each processing element is fitted into a 30 μm column width which is larger than that of the average image sensor (< 10 μm). However, custom layout of digital latches and logic will reduce area and optimize the processing power consumption. In addition, as the technology scales down, the size of processing element can shrink and more metal layers can help reduce the area of the pixel interconnections.

Due to a design mistake in the image sensor array, some defects are observed in the image in the normal mode operation. This is attributed to cross-talk between the image sensor array and the readout circuits, which can be eliminated by putting guard ring around the sensors. In

addition, the layouts are not particularly optimized for the spatial or temporal noise. Layout optimization will help in the noise reduction of the images. It will also enhance the degradation seen in the image when both the normal mode and the binary processing mode are on.

The design of the prototype chip is a good demonstration of one possible implementation structure for on-chip image processing. However, it does have limitations in terms of programmability. Many operations of programmable binary image processing require repeated or iterative processing on the images at various stages of the processing. For best results, the images have to be fed back to the same operation over and over again, or to different operations. In contrast, our on-chip binary processing takes the input image straight from the image sensor array, and thus it is not able to do repeated operations on the input image. For repeated computations, a number of the processing components need to be designed on the data path, each independently operating its function each time. However, the design of the repeated operation is a trade-off between the complexity of the design, power and area.

The design of on-chip binary image processing, therefore, is for low level processing applications where low power consumption and design cost are emphasized. This demonstration of the chip is intended to prove functionality and feasibility, and to as a guide to the future research direction. Primary obstacles for on-chip local processing implementations are due to the design complexity of processing elements and the large number of interconnections between neighboring pixels. With the 0.35 μm technology, it is not impossible, but very difficult to implement local masks larger than 3x3. As the technology scales down and more metal layers are available, this restriction on the mask size will be loosened and 5x5 local masks will be easily feasible in the future. In order to get effective results from some of the local operations, at least 5x5 local masks (or larger) should be applied even with the tradeoff of area, power and design complexity. However, instead of voltage operation in these elements, current mode operations are expected to reduce the design complexity. Also, current mode operation can reduce the required processing time by eliminating the phenomenon of charging and discharging on the capacitive nodes [93]. Low power operation is achievable with this current mode because the dynamic range of the output is due to current, not voltage, which is less affected by the low voltage supplies,

expected in more advanced CMOS technologies. A more detailed example of current mode processing is illustrated in Appendix A, where a modified pixel structure of an inverted logarithmic pixel sensor is introduced.

Chapter VII

7. Global Operation

7.1. Introduction

Approaches to image processing fall into two broad categories in terms of operational domain: the spatial and frequency domains. The spatial domain refers to the image plane itself. Approaches in this category are based on direct manipulation of pixels in an image. Frequency domain processing techniques are based on modifying the spatial Fourier transform of an image. An image in spatial domain is converted into frequency domain by the Fourier transform. Since computation of convolution in spatial domain is equivalent to multiplication in frequency domain, manipulation of a linear system on the image becomes multiplication of the image by a filter transfer function in the frequency domain. The resultant image in the frequency domain is converted to the spatial domain by taking the inverse transform. Many basic ideas of smoothing, sharpening or edge detection filters arise from concepts directly related to the Fourier transform because these filters attenuate or intensify only portions of frequency components [74][75].

Frequency domain operations, by the very nature of the frequency transforms, are global operations, where all the pixel values in the image are taken into consideration at once. Of course, frequency domain operation can become a local (or mask) operation, based on a local neighborhood, by performing the transform on small image blocks instead of the entire image. In contrast, spatial domain processing methods include all the three types of point, local and global operation. Global operation in the spatial domain leads to very difficult

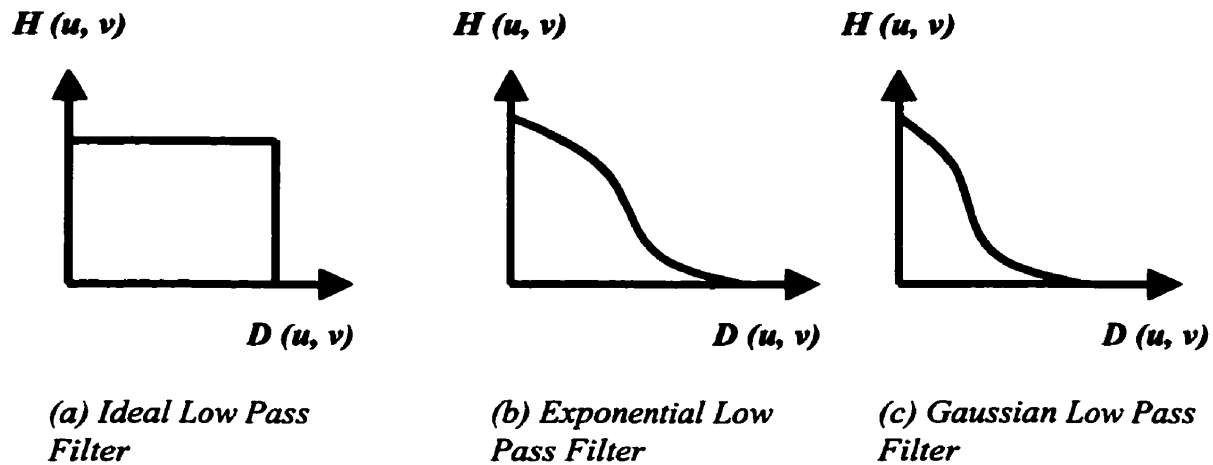


Figure 7.1. Transfer function of different types of low pass filters. $H(u, v)$ is the transfer function and $D(u, v)$ is the distance from the origin.

design issues and implementation methods, due to the connections required to all the pixels in the image. Therefore, global implementations are implemented more like local operation methods by restricting their neighborhood to the localized area. For global operation, frequency domain methods are preferred to spatial domain methods because manipulation in the frequency domain is relatively easier and more powerful, at least in software implementations. There are plenty of examples for the frequency domain processing, which are well established and well documented [74] [75].

Frequency Domain

In the frequency domain, there are generally three types of image enhancement filters: smoothing filters, sharpening filters and homomorphic filters. The smoothing filters in frequency domain are similar to the smoothing filters in spatial domain. In fact, the basic idea of their operation is identical: Edges and sharp transitions of an image, which contribute significantly to the high-frequency content of the transform, are smoothed/blurred out by attenuating a specified range of high-frequency components in the transform of a given image.

An obvious smoothing filter is the ideal low pass filter (see Figure 7.1 (a)), which attenuates or eliminates the high-frequency content of an image. The ideal low pass filter is

theoretically desirable, but in practice there are no filters that match with the ideal operation in hardware. The practical implementations of the low pass filters include exponential low pass filter (see Figure 7.1 (b)) and Gaussian low pass filter (see Figure 7.1 (c)).

In contrast, sharpening operates in the opposite way to smoothing. Because edges and abrupt transitions in an image are associated with high-frequency components of the transform, sharpening is achieved by attenuating or eliminating the low-frequency components without disturbing high-frequency components of the transform. Sharpening filters are subdivided into high pass filters and high frequency emphasis filters. The high pass filter is exactly opposite to the low pass filter, which is also expressed as $(1 - \text{low pass filter})$. Its transfer function is shown in Figure 7.2. The high frequency emphasis filters, also known as Homomorphic filters, use this characteristic that the illumination component of an image is typically associated with slow spatial changes, while the reflectance is with abrupt transitions, relating the high-frequency components of the Fourier transform of the logarithm of an image [74]. This control requires specification of a filter function $H(u, v)$ that affects the low- and high-frequency components of the Fourier transform in different ways, as shown in Figure 7.3. Examples of the high frequency emphasis filters include generalized unsharp masking, inverse blur model, difference of Gaussian (DOG), Laplacian of Gaussian (LOG) and modulated Gaussian (Gabor) filters [74][75].

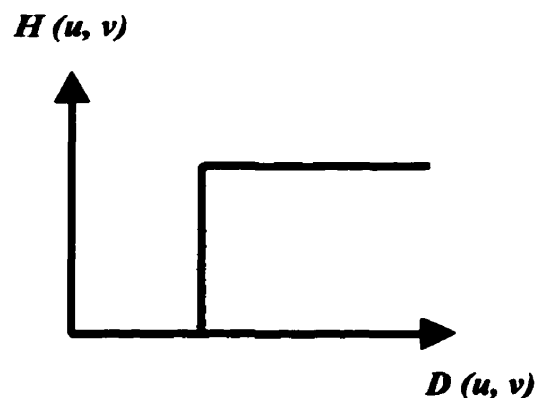


Figure 7.2. Transfer function of ideal high pass filter, which is used as a sharpening filter.

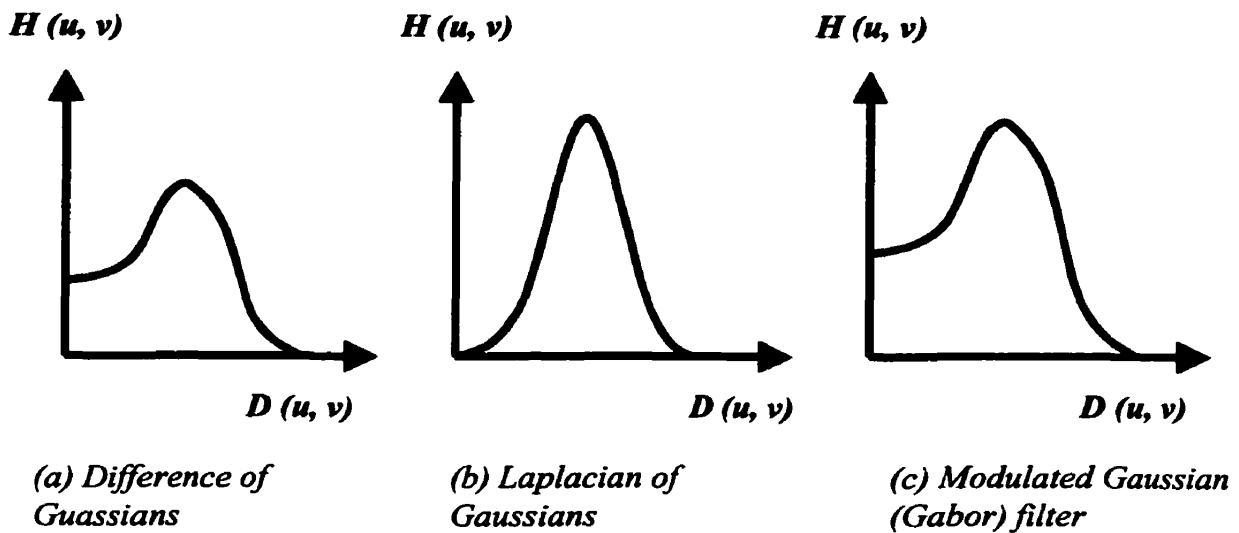


Figure 7.3. Transfer functions of high frequency emphasis filters.

Spatial Domain

In practice, small spatial masks are used considerably more frequently than the Fourier transform because of their simplicity of implementation and speed of operation. From the aspects of on-chip implementations, the spatial masks are often more of interest than the frequency operations. In the spatial domain, global operations can be divided into two categories: One is a global operation of which the output is an image defined from the input image, the other is where the output of the operation is the information extracted from the input image. A resistive Gaussian filter [78] is an example of a global operation with an image output in the spatial domain. Although the strong influence of the neighborhood is limited to 5 or 6 pixels, depending on the resistive values, the resistive network connects all the pixels into a system, and a pixel value is affected by all other pixels. An example of information-extracted global operation includes histogram operation. In order to generate the histogram of the input image, the operation needs all the pixel values of the array, but does not modify them and does not generate a new output image; the output is information about the input image. This output can be used to modify the output image as discussed in Ch.5.

Practically, the on-chip implementation of the spatial global operation with image output is not possible due to the heavy interconnections, unless each pixel is interactive like the network of the Gaussian filter, where one pixel stores other pixel values and this propagates

through the entire array. However, even the resistive Gaussian filter does not give a practical realization. In contrast, the global operation with information output is relatively easier to implement because information of all the pixels can be collected through a common data channel.

7.2. Structure of Global Processing

When it comes to the implementation of on-chip image processing, particularly for global operation, the system level architecture becomes an important step of the design process. Since we have considered global operation in terms of the frequency and spatial domains, we start by examining structural implementation by looking at the possible methods for these domains. In the frequency domain, one of the most essential components would be a Fourier transformer that converts from the spatial domain to the frequency domain. As before, it can be integrated at pixel, column or chip levels. The design of the transformer, however, takes a significant number of transistors, which is not appealing for a pixel level implementation. It is more reasonable to implement the transformer at every column of the array or at the output channel(s) of the chip. After the Fourier transformation, the image in the frequency domain would be manipulated by image processing algorithms. The manipulation is based, by the nature of the global operation, on the contents of all the pixel values in the array. One possible (the most adequate) method is to use an analog memory to hold and process the contents of all the pixel values. The analog memory implementation often requires high complexity of design and precision. Also, the design area, which is roughly proportional to the fabrication cost and power, becomes large. Therefore, it is difficult to integrate image sensors, Fourier transformer and image processing elements on a single chip, even with low-level image processing.

In the spatial domain, implementation of global operation is relatively easier than in the frequency domain, simply because of the absence of need for a Fourier transformer. Spatial domain processing for image output can be implemented with pixel level integration and analog memory processing. Because it is practically impossible to have one pixel connected to all other pixels with one designated channel between the pixels, the connection should have a characteristic of propagation similar to a resistor network. So, pixels which are not directly connected together may affect each other indirectly, because of the propagation

effects of the grid connections. In pixel processing integration, therefore, each pixel should have a characteristic of this holding and propagating, where, after the pixel captures an image signal with its photodetector, global processing occurs over all pixels of the array in parallel. The pixel still holds its own image signal, and processes and propagates all other pixel values. The main advantage of this implementation is parallel processing where low power consumption can easily be achieved. However, this implementation will suffer from severe reduction of fill factor in the pixel. Because of the nature of hold/propagate, the complexity of the circuit design is high, typically involving a large number of transistors, and sometimes, the use of passive elements such as capacitors.

In order to avoid the severe reduction of fill factor, analog memory implementation may be applied for global operation. By placing processing elements out of the photodetector pixels, the entire area of the pixel can be occupied with a photodetector and a readout buffer. As trade-offs for this gain in fill factor, the chip area, power and speed will be sacrificed. Also, there is always a time latency of one image frame because, after the image capture, the image frame is stored and processed in the analog memory, instead of being output directly. Similar to pixel processing integration, the global connections between pixels should be done using the method of propagation. Otherwise, it is practically difficult to implement, especially for large format arrays.

Global operation with information output can be implemented with chip processing, where the processing element is located at the end of common output channel, collecting information from the pixels and generating final output. In order to collect information from all the pixels, the pixel data should go through a common processing element. Therefore, the processing element requires high-speed operation, at least equivalent to the pixel rate, causing high design complexity, high power and a potentially high digital noise level.

The general description and comparisons are summarized in Table 9, in terms of their operation domains. However, there are no easy ways to integrate global operation with image sensors on the same focal plane. The general implementation methods for global operation are neither practical nor feasible due to the heavy interconnections between pixels, and due to the circuit complexity. Instead, the implementations for global operations should be rather application specific and algorithm dependent.

	Spatial Domain		Frequency Domain
	Image output	Information output	
Pixel Processing	Global interconnections reduce feasibility of implementation except for special designs such as Gaussian resistive networks (hold/propagate)	Impractical design due to presence of a common data channel	Fourier transformer at column or chip level, and global interconnections are done at pixel level and analog frame memory structure. Typically, the implementation of on-chip Fourier transformer is complex
Column Processing	Concurrent readout of all pixels is not feasible	Impractical design due to presence of a common data channel	
Chip Processing	Concurrent readout of all pixels is not feasible	Most suitable structure, but modified structures combined with pixel and column processing are more practical	
Frame Memory Processing	Global interconnections reduce feasibility of practical designs. High fill factor but degradations on chip area, power and speed	Impractical design due to presence of a common data channel	

Table 9. General descriptions and comparisons of global operation, for different operation domain

Here, as a demonstration of global operation, we report a 2-D object positioning system with partially global connections, along with its implementation and performance. This implementation is a semi-global operation with information output. This is used to examine the feasibility of on-chip global implementation with information output and the design issues involved in the implementation, and its suitability for different applications such as motion analysis and object extraction.

7.3. 2-D Object Positioning System (OPS)

The 2-D OPS encodes 2-D information into two sets of 1-D information. Figure 7.4 illustrates the basic operation of the OPS; whenever objects are detected, the pixels containing signals above a threshold send flags to the column and row simultaneously. In the OPS array, each pixel has a photo-detector and an in-pixel voltage comparator. Whenever the input light level is higher than the threshold, the in-pixel comparator flags up to its corresponding row and column. Each row and column has an NAND gate function, generating '1' when all pixels in its row/column are over the threshold, as shown in Figure 7.5. Hence, dark objects are detected from a lighter background by presence of a '0' in row or column. For example, as shown in Figure 7.4, pixels corresponding to a circle send flags to their corresponding columns and rows, and thus the final image captured becomes a square. Although some information is lost, the system enables straightforward determination of the presence or absence of an object, as well as its size and/or orientation. Multiple objects can also be characterized. Moreover, simple combinational logic on the latches can be used to apply an object size threshold, which is otherwise difficult to achieve.

The OPS does not require scanning readout but provides a true simultaneous readout, making the frame rate independent of the scanning time of each pixel. Conventionally the frame rate of the array, especially large arrays, depends on scanning time of individual pixels because image signal from each pixel has to be transmitted one by one. By the nature of fast frame rate in the OPS, it can be used in motion detection as well as many other dynamic image acquisition applications.

In addition to the fast readout time, the OPS reduces the image data from N^2 to $2N$, where N represents the number of rows and columns. A dual channel is used for the output, vertical and horizontal outputs, increasing the total output rate. With the fast readout time, data

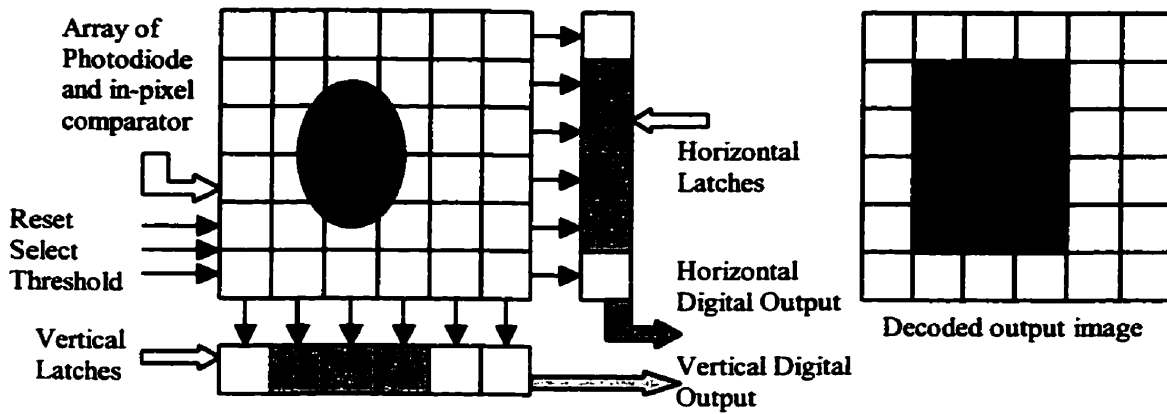


Figure 7.4. Structure of 2-D Object Positioning System and its basic operation. With only two input controls signals (Reset and Select), simultaneous outputs converted in two sets of linear data from the 2-D array plane. Two 1-D data can be further processed and displayed into 2-D plane. A circle in the original plane is interpreted as rectangle in the display.

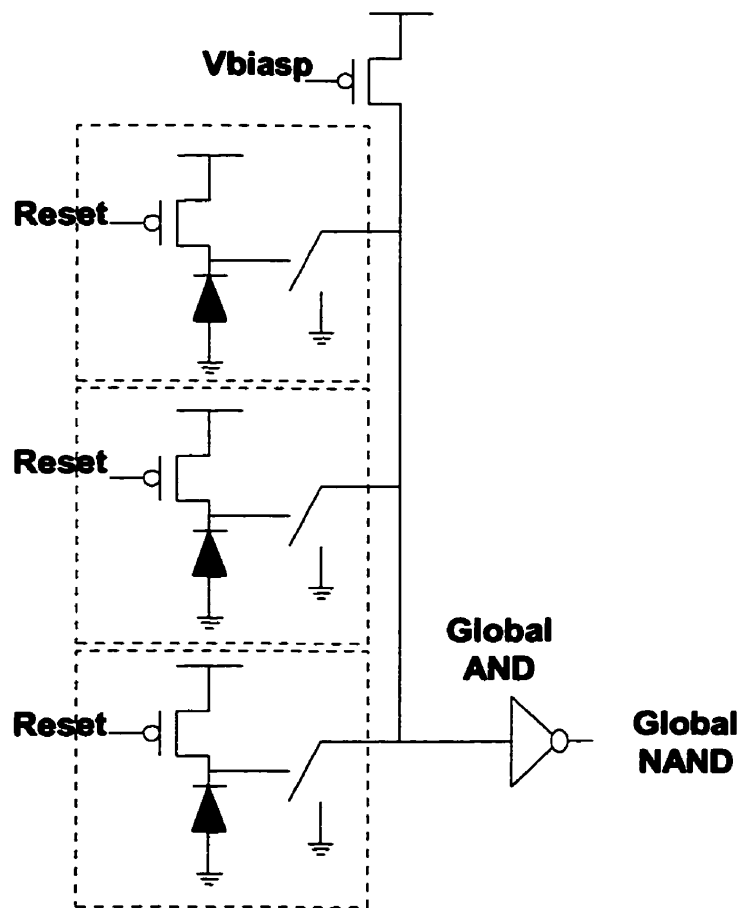


Figure 7.5. Structure of global connections.

reduction and dual output channels, the high frame update rate is a focus of the OPS. In addition, the OPS uses digital signal transmission, and thus it is relatively immune to noise during transmission. Applications of such a threshold-based system include industrial web inspection, earth observation from space, robotic vision, and other applications where object detection with high speed is the primary goal.

The main objectives of this prototype chip are (i) to explore the feasibility of global operation integrated with CMOS APS, (ii) to demonstrate high-speed operation of the OPS and (iii) to address limitations and future research directions of the global operation.

7.3.1. Chip Design

The structure of the OPS is quite similar to that of a standard CMOS image sensor array. The chip whose die photo is shown in Figure 7.7, consists of photodiode pixels, in-pixel comparators, vertical and horizontal latches and shift registers. The overall structure of the

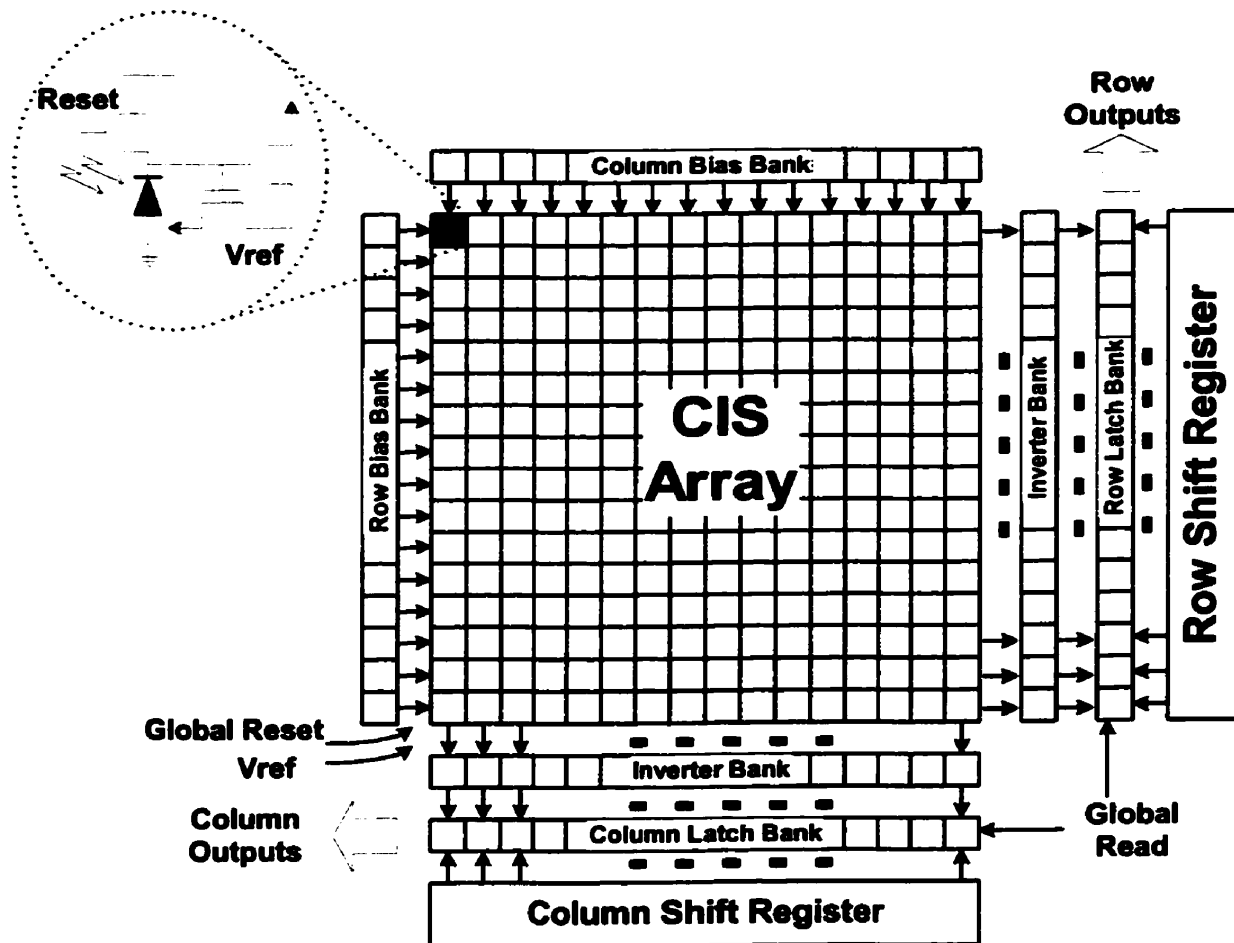


Figure 7.6. Overall structure of CIS array with object positioning system.

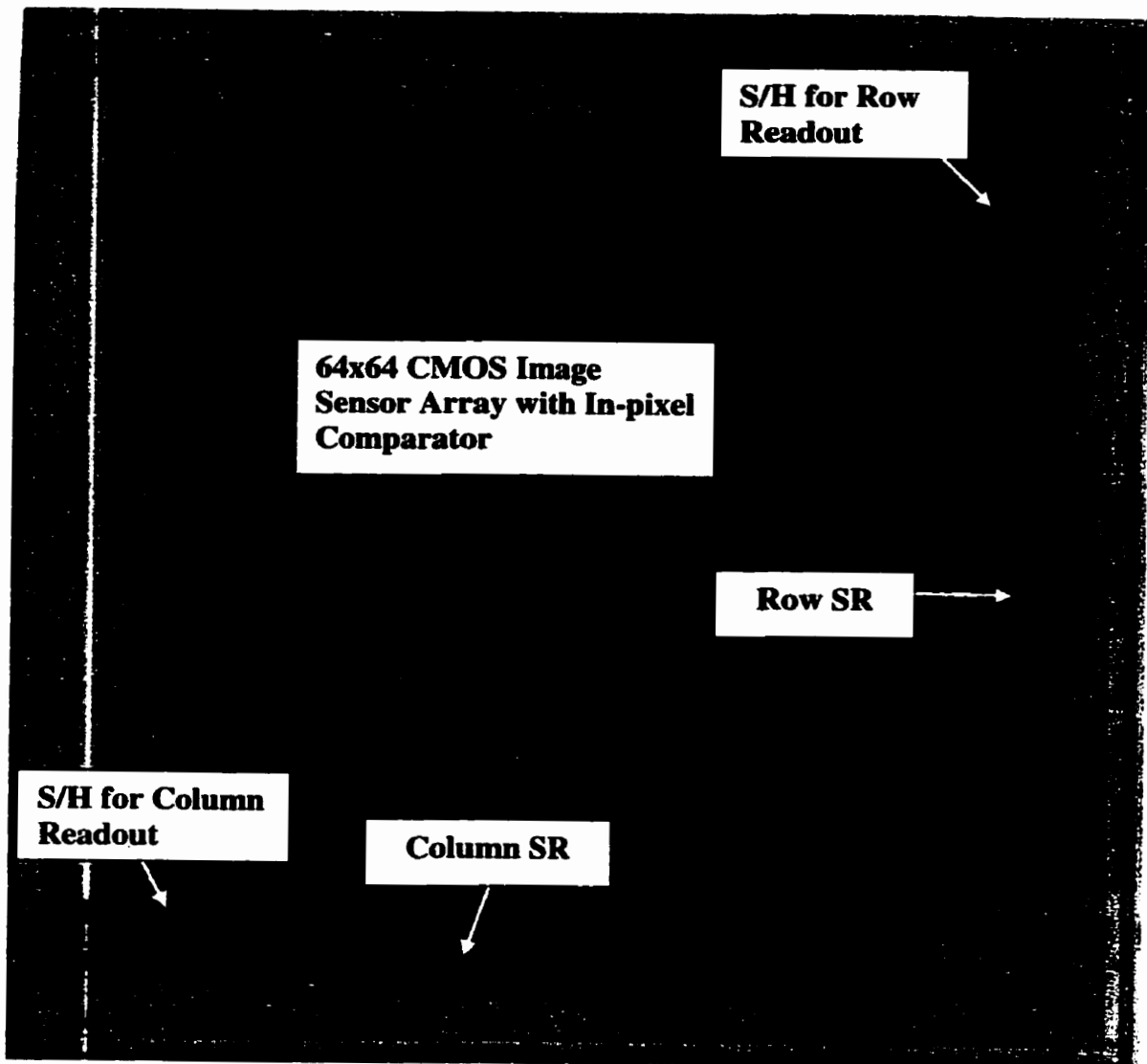


Figure 7.7. Die photo of Object Positioning Chip.

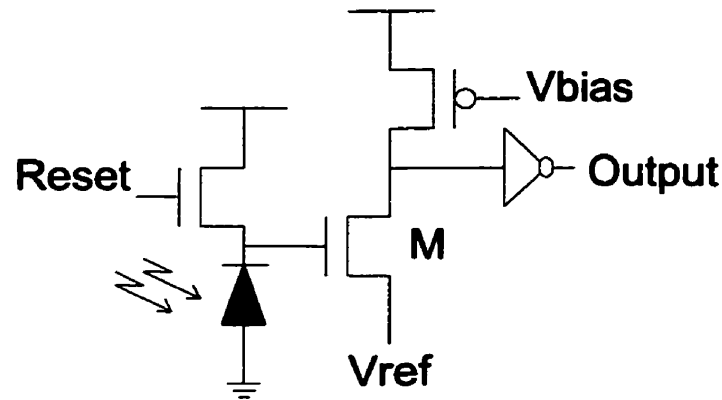


Figure 7.8. Schematic of a pixel for 2-D Object Positioning System. It consists of photodiode and in-pixel comparator. The in-pixel comparator is composed of a common source amplifier and an inverter. The bias transistor and inverter are located outside the pixel.

OPS is shown in Figure 7.6. It has dual output channel where each channel transmits the data from each of the vertical and horizontal lines. The pixel has the same p-n junction photodiode, but it uses an in-pixel comparator instead of a source follower buffer for the pixel readout. An in-pixel comparator should be a simple structure using the fewest transistors possible in order to maintain a high fill factor. The in-pixel comparator uses a common source (CS) amplifier with an inverter at the end of data line to enhance switching activity, as shown in Figure 7.8. Because the inverter and bias transistor can be located outside of the pixel, only two transistors are needed in a pixel. V_{ref} and V_{bias} affect the speed and threshold voltage of the switching. Because the output of the pixel is read out vertically as well as horizontally to the line latches, each pixel has in-pixel comparators for each line. When the pixel detects an over-threshold signal, it sends a flag to both lines simultaneously. Since every pixel in the same line (column or row) is connected together, the values are read out to the lines simultaneously from every pixel in the same output line. If, during the time when the output value of the in-pixel comparator is sent to the outside, the light intensity is higher than the threshold, an output of '1' is transmitted, otherwise, '0'. Hence, whenever the pixel detects that the light intensity is over the threshold, the in-pixel comparator triggers the flag to the output line. Initially, the output of the data line is set to ground. When the light intensity is high enough, making the photodiode voltage lower than V_t of M transistor, the M transistor

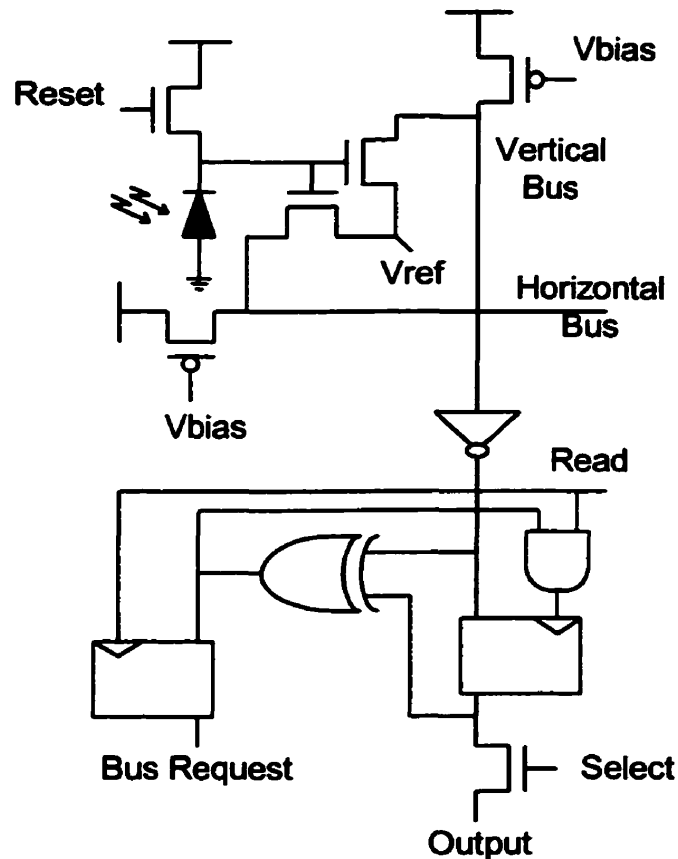


Figure 7.9. Schematics of a pixel and event detection latch in 2-D Object Positioning System.

is switched off and the PMOS bias transistor lets the output node charge up to V_{DD} . Since all the pixels in the same data line are linked together, if any comparator along the line is switched on, the line remains switched on. It is an AND logic function (refer to Figure 7.5).

At the bottom of each data line, there is a skewed inverter before the latch. The inverter enhances the switching sharpness and speed. In the common source amplifier, V_{ref} determines the lowest voltage level of the output voltage on the data line. V_{ref} should be recognized as '0' for the inverter even when it is over $V_{dd}/2$. A skewed inverter was carefully simulated and the optimal size ratio of the transistors, by increasing the size of the PMOS in the inverter was decided. By adding an inverter, the overall logic function becomes NAND gate; the output is '0' only when all the pixels along the line have high light intensity. Otherwise, the output is '1'. Hence, this system detects dark objects on a white background.

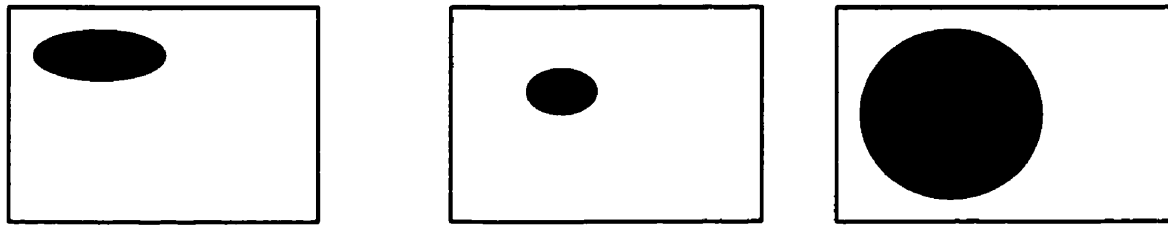
In order to read out data from the array to the serial output, the data is multiplexed out after being stored in the latches in vertical and horizontal lines. Shift registers send enable signals to the latch multiplexer and transmit the data one by one to the serial output channel. The latch uses a simple digital component, either a flip-flop or an inverter based design. In our design, a flip-flop design is used for simplicity.

7.3.2. Demonstration and Tests

The OPS chip was fabricated in 0.35 μm CMOS technology with 3.3 V power supply, and has been demonstrated successfully. Imager characteristics of the chip are shown in Table 10. When a circle is shown to the sensor array, the in-pixel comparators first digitize the shape. The outputs of all the in-pixel comparators in the line are then NAND gated into one output per line. Therefore, the shape of the object becomes a square as shown in Figure 7.10. All the shapes of the objects are encoded into squares or rectangles by the array NAND gates. This mechanism hides some of information that originally exists in the objects. However, some of the critical information, such as position and size of objects, are preserved and encoded into a smaller amount of data at relatively high speed. By the nature of the operation, when more than two objects exist in the field of view, false objects are created in the overlapping area of the objects. When two different circles exist in the white background, the output image contains two original squares and two extra squares which are falsely created in the

Characteristics of Chip	
Chip (Die) Size	2880.5 x 2880.5 μm^2
Array Format	64 H x 64 V (4096) pixels
Pixel Size	23.8 μm x 23.8 μm
Fill Factor	72%
Technology	0.35 μm CMOS
Frame Rate	24 frames/second at 290 lux
V_{DD} Power Supply	3.3 V
Nominal Current	20 mA – 8 mA (conversion chip) = 12 mW at 24 frames/sec
Power Consumption	66 mW – 26.4 mW (conversion chip) = 39.6 mW at 24 frames/sec
Output	1 bit Digital output
Package	68 PGA

Table 10. Characteristics of chip tests.

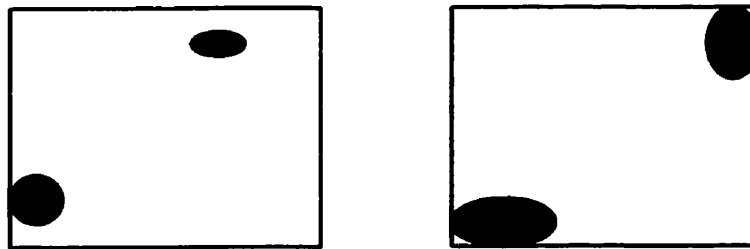


(a) Original Images presented to sensor

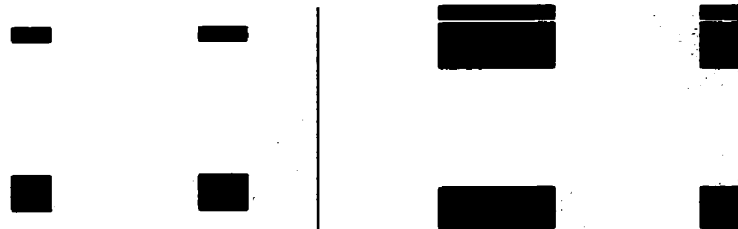


(b) Encoded Images of the objects reconstructed from sensor output

Figure 7.10. Sample images from the 2D object positioning chip. Input shapes are encoded into squares or rectangles in the final output images.



(a) Original Images



(b) Reconstructed Image of the Shapes

Figure 7.11. When multiple objects exist in the input image, there are defects (counterpart objects) in the output image.

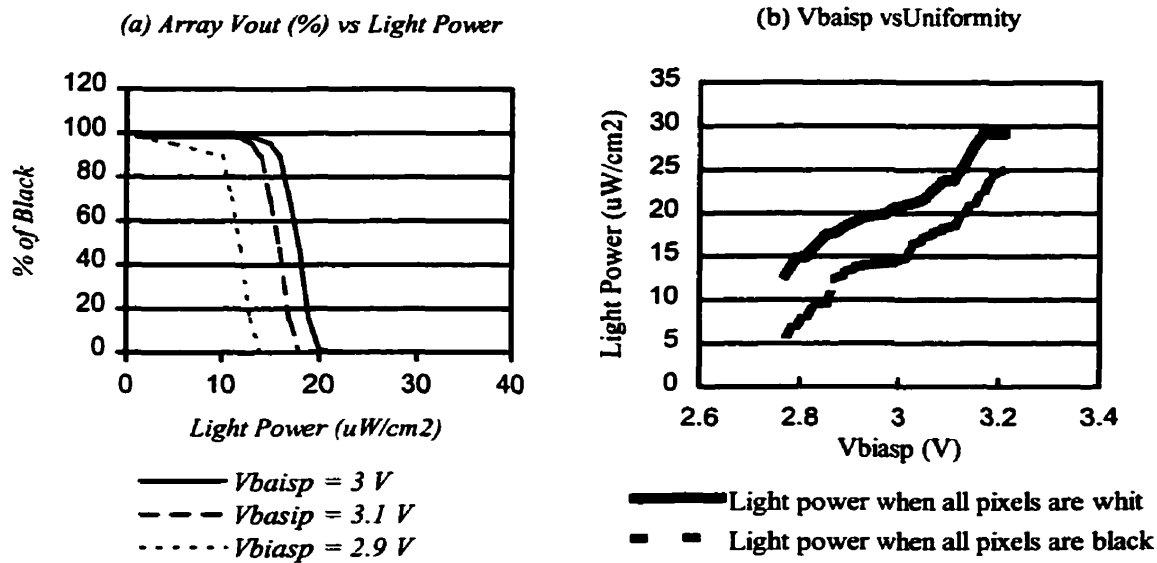


Figure 7.12. Test results of 2-D OPS imager. (a) With different V_{biasp} , the responses of outputs are drawn. (b) Non-uniformity of OPS imager can be measured.

overlapping area of the original ones, shown in Figure 7.11.

Figure 7.12 illustrates the relationship between V_{biasp} and array uniformity. Here, no pattern noise reduction was implemented. Figure 7.12 (a) indicates that not all pixels switch at the same scene illumination intensity, due to a combination of pattern noise in the sensor and non-uniformity in the comparators. In Figure 7.12 (b), the upper line represents the light power at which all the pixels are high and the lower line is vice versa. In the light power gap between the two lines, the white and black spots co-exist due to the non-uniformity response of the pixels as well as the in-pixel comparators. The gap between the two lines represents how much light power difference should exist for the objects to be recognized correctly. The minimum difference in light power is consistent at different biasing voltages. Therefore, it is necessary to remove this non-uniformity before image processing for segmentation, object recognition, model fitting, etc.

7.3.3. Summary and Conclusions

Here, we have seen an example of on-chip implementations for spatial global operation integrated with a CMOS image sensor. The 2-D Object Positioning System extracts the coordinates of objects of interest by detecting a property of the image (in this particular case,

the property is the light intensity). The 2-D OPS encodes 2-D information into two sets of 1-D information. The basic operation of the OPS is as follows; whenever objects are detected, the pixels containing signals above a threshold send flags to the column and row simultaneously. For example, pixels corresponding to a circle (or any other objects) send flags to their corresponding columns and rows, and thus the final image captured becomes a square (or a rectangle). By encoding 2-D information, the OPS enhances the speed of the I/O interface, which is often a bottleneck of the processing speed, especially for vision applications. In a case of $N \times N$ pixels array, the reduction ratio will be $N \times N / 2N = N / 2$, which is significant when the array is large. In addition, the OPS chip operates in real time mode that is advantageous in its operation and applications.

However, the operational speed of these prototype sensors was not as high as was originally expected. This is mainly due to poor photosensitivity of the photodetectors. Although the rest of this chip, other than the photodetectors, can run at high speed, the photodetectors need a long integration time, which becomes the bottleneck of system speed. With optimization in optical performance and in noise level, the positioning system can achieve a high speed operation.

In the particular case of motion detection, the concept of the Object Positioning can be useful due to its high speed and data reduction, even though some of information for the object of interest are lost during the processing and some artifacts occur when multiple objects are located in the same field of view. The main concern in the on-chip implementation of the concept is that the in-pixel comparators suffer from operational mismatch due to V_t and lithographic variations. A better design of the comparator can be achieved with the sacrifice of circuit complexity and design area, where photosensitivity of the pixel would be reduced. As the technology scales down, the scalability of the in-pixel comparator is high.

In conclusion, the implementation of true global operation in the spatial domain is not an easy task, mainly because of its requirements for extensive interconnections, large computational power and high design complexity. Rather than a general implementation of the operation, the approach should be application specific and operational algorithm dependent. Unless the application requires the global interconnections (or partially global), the on-chip implementation of the global operation in spatial domain is not recommended.

Chapter VIII

8. Summary and Conclusions

Raw output images from CMOS sensors are not likely to be optimal for display or further processing mainly because of noise, blurriness and poor contrast. In order to minimize these degradations, image enhancement and processing mechanisms (meaning circuit level designs apart from device level modifications on photoreceptors) are necessary because the device level modifications often meet baseline limitations of the standard process technology.

When real time image acquisition and processing are desired, the integration of image processing (vision) algorithms with image sensors has many advantages. In this thesis, integration of image processing and image sensors is presented with a concept of smart sensors. The integration of vision algorithms and image sensors is an attractive research field, which can provide low fabrication cost, low power consumption and fast processing for various applications. In addition, analog/mixed signal image processing achieves additional advantages of compact size and fast continuous mode to the integration benefits of smart sensors.

This thesis discussed two main concepts: MOSAIC imager and smart sensors. MOSAIC concept was proposed to achieve a large field of view and a high scene update rate. The MOSAIC array is described for a distributed sensor consisting of $10^2 - 10^3$ identical detection modules linked by a serial bus to a central controller. Main challenges of the MOSAIC imagers are large data flow and slow frame rate. The design of the MOSAIC system focuses on enhancement of frame rate, by a single chip solution (i.e. integrating CMOS image sensors and bus interface modules on a same focal plane). Custom bus

interface modules increased performance of the bus connections by an efficient design of zero-wait state, at effective cost. Therefore a MOSAIC imager comprising many single-chip modules is capable of covering a larger field of view (10^2 to 10^3 or more) than the conventional single chip camera system, with the enhanced data update rate. Also, a smart sensor with critical information extraction was proposed here as an alternative solution of the MOSAIC imagers. The on-chip processing of the smart sensors extracts the information at the front end of the imagers, reducing data flow and thus, increasing the field of view and/or update rate.

In the second part of this thesis, integration architectures and design methodologies were investigated for application to analog VLSI implementations for smart sensors. The basic concept of the integration architecture comes from an idea that, for the integration of image processing with CMOS image sensors, vision algorithms and application specifications should be considered, in addition to the selection of appropriate processing circuit. Conventionally, the integration methodology focuses on reducing circuit density of processing element integrated with image sensors. This thesis argues that not only the circuit density is important, but also the algorithms of processing are sometimes more crucial.

Hence, various vision (image processing) algorithms were investigated systematically according to interconnectivity with neighboring pixels (the region of operation). The vision algorithms were partitioned into three major groups: point, local and global operation. These algorithms were once again sub-divided by functionality, size of local masks and operational domain. For each sub-partitioned algorithm, different implementation architectures were proposed and compared in terms of design area, speed, processing time, power and pixel fill factor.

The proposed general guideline is summarized in Table 11, where system level architectural designers and circuit engineers can start their milestone implementations for smart sensors according to algorithms they try to implement. However, designers should consider their applications and design specifications cautiously, and should make proper modifications on individual design components, in order to make less error prone implementations.

		Point Operation			Local Operation		Global Operation		
		Concurrent Operations	Histogram Operations	Interframe Operations	3 x 3 Masks	Masks larger than 3x3	Info. Output (Spatial)	Image Output (Spatial)	Frequency Domain
Pixel Level		(X)		(X)	X			(X)	
Column Level	Local Memory	X			(X)				
	Pipelined				X	X			
Chip Level		X	X		X		(X)		
Frame Memory		X		X	X			X	
Hybrid	Chip + Pixel		(X)				X		
	Chip + Column		X			(X)	X		
	Chip + F.memory						X		(X)

X *Recommended*
 (X) *Highly recommended*

Table 11. Summary of on-chip implementation methodology for image processing algorithms.

Prototype chips for each major group in the vision algorithms were designed and fabricated with 0.35 μm CMOS technology, for the demonstration of on-chip implementation of algorithms. Three prototype chips were implemented: in-pixel intensity transformer for point operation, on-chip binary image processing for local operation, and object positioning system for global operation. These prototype chips were tested and demonstrated successfully.

It is concluded in this thesis that on-chip image processing with image sensors will offer benefits of low fabrication cost, low power consumption, fast processing frequency and parallel processing. Since each vision algorithm has its own applications and design specifications, it is dangerous to predetermine optimal design architecture for every vision algorithm. However, in general, the pixel and column structures appear to be the best choice for typical image processing algorithms such as point operation and local operation.

The implementation of global operation is not recommended in spatial domain because of the heavy interconnections and computational power requirements. Typically, the implementations of the global operation in the spatial domain should be modified and adapted for application-specific environments.

Since CMOS image sensors use a standard process technology, modifications of the image sensing process cannot be achieved easily and optimization of the image sensing properties will not be as good as CCD. Although many microelectronic process companies such as TSMC, UMC and Tower Semiconductor offer the specialized processes for CMOS image sensors, CCD is still superior to CIS in terms of image quality. Typically in order for CIS to obtain equivalent image quality as CCD, special processes are needed, which require modifications on the standard process. The specialized process means expensive fabrication, which is contrary to the low cost concept of CIS. Therefore, even for the image quality enhancement of CIS, circuit level improvements such as image processing circuits are preferred to the process level improvements in the CIS technology. The circuit level improvements are not beneficial only for the image quality, but also for low cost VLSI integration. Sometimes, the VLSI integration of CIS is more emphasized than the image quality enhancement for such applications as portable image devices, machine vision, surveillance and industrial inspections. Therefore, in the future CMOS image sensors will

find their own applications where low cost and high functional image sensing is the driving force even with relatively low image quality.

Appendix A: Inverted Logarithmic Pixel Sensors with Current Readout

A.1. Introduction

Current readout active pixel sensors are inherently advantageous in terms of readout speed because the fixed output line voltage at input of transresistance amplifier prevents charge-discharge phenomena [79]. Another benefit of current readout is current mode processing which is relatively compact in size and simple in its operation [80]. One drawback of the active pixel sensors with current mode is lack of design resources. Because the current mode processing circuitry has not been well studied relatively, most of implementations will have to be custom designs.

This appendix reports a CMOS active pixel sensor structure for a logarithmic pixel with continuous current readout. Because the design is distinct from the main theme of the thesis, it is located in this appendix. Here the arrangement of the photodiode and load found in a conventional logarithmic pixel is reversed. The inverted logarithmic pixel sensor reduces fixed pattern noise and eliminates the dependence of the output voltage swing on the column load, simplifying both of its operation and structural design. We include a detailed design of the inverted logarithmic pixel sensor and its analysis, along with operational performance and experimental results.

A.2. Inverted Logarithmic Pixel Sensors

We report a continuous current readout logarithmic active pixel in which the conventional arrangement of the photodiode and load are reversed. As shown in Fig.A.1(a), a conventional logarithmic pixel employs a photodiode to generate a photocurrent and one or more MOSFETs operating in subthreshold to act as a load. The voltage dropped across the load is dependent on $\ln(i_{photo})$ due to this subthreshold operation. Such a configuration has advantages of continuous operation, thereby enabling temporal as well as spatial random access, and wide dynamic range (~6 orders of magnitude of illumination).

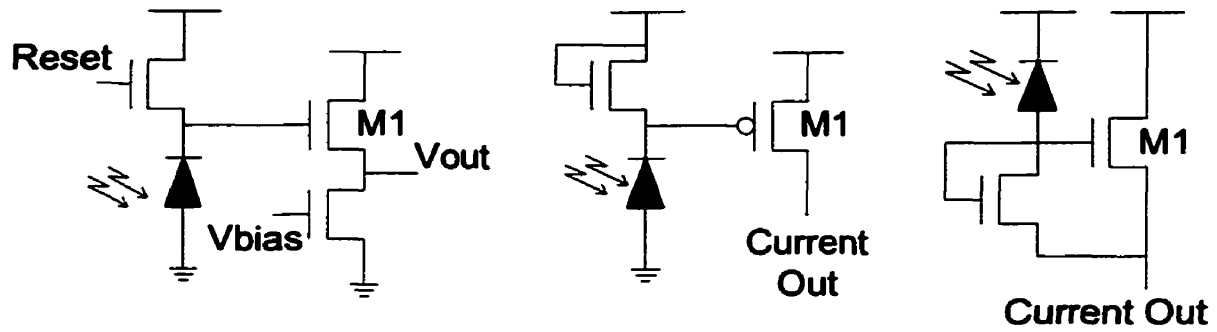


Figure A.1. Structures of logarithmic pixel sensors: (a) conventional log pixel, (b) current readout with PMOS buffer, (c) inverted log pixel.

Disadvantages include high fixed pattern noise, low contrast due to a small voltage swing (typically 200mV for the entire 6 order range of illumination), and relatively poor response at low illumination [81] [82]. The complement of current readout technique of this pixel structure is also possible, where the load and photodiode positions are the same as the conventional logarithmic pixel, but a PMOS buffer transistor is used (see Fig.A.1(b)). The voltage generated across the load by the photocurrent appears as V_{gs} of PMOS buffer transistor (M1 in Fig.A.1(b)). As the light intensity increases, the V_{gs} of the PMOS transistor increases, generating output current which is equal to $K(V_{gs} - V_T)^2$. However, PMOS transistors are known to have higher lithographical mismatch than NMOS [89][90], so this structure is expected to display higher fixed pattern noise. However, as technology is developed and more attentions are brought to every level of process, mismatch of PMOS transistors is not necessarily worse than that of NMOS transistors any more. Therefore, it is process dependent.

Another contribution to the low voltage swing for the conventional logarithmic pixel is a trade off in the choice of column bias; to maximise V_{out} , a low V_{bias} is required, but a high V_{out} reduces V_{gs} for M1. In our design, (see Fig.A.1(c)) the positions of the photodiode and load are reversed, so the voltage generated across the load appears directly as V_{gs} . Now the PMOS transistors consume more area than NMOS because of the implementation of wells, the use of PMOS transistors is often avoided.

Moreover, the response of the pixel is now dependent on local, as opposed to global, matching of MOSFET characteristics. A larger than average local W/L (caused by

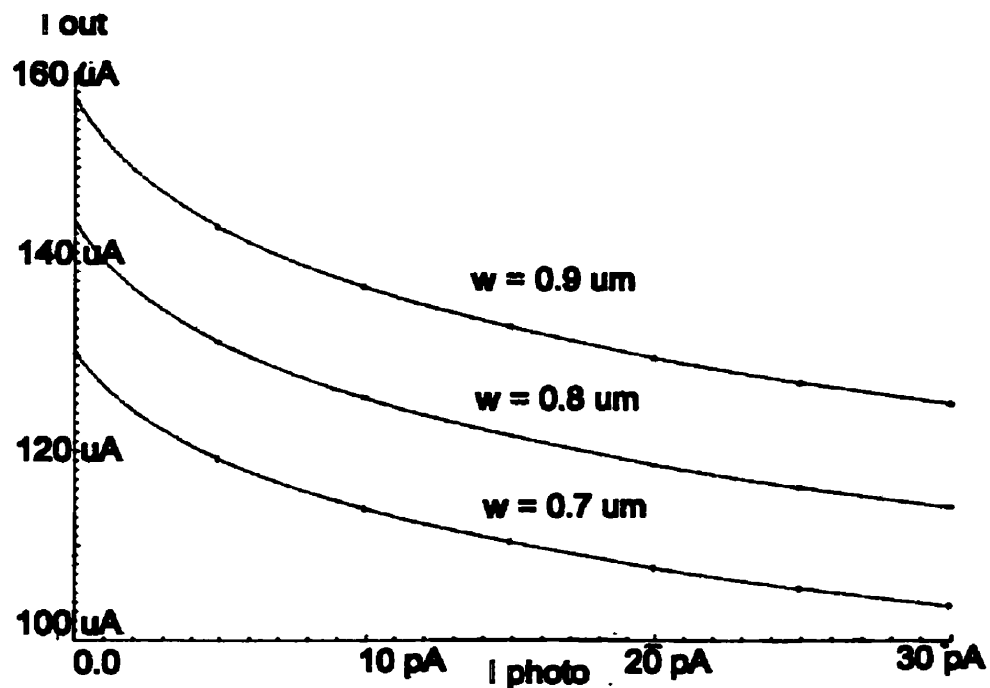


Figure A.2. Simulated effect of lithographic deviation on a regular logarithmic pixel sensor. As varying w with a fixed $l = 0.35 \mu\text{m}$, the output current of the driving transistor, $M1$, changes significantly. At $I_{ph} = 10 \text{ pA}$, variation of w leads to approximately $30 \mu\text{A}$ ($\sim 130\%$ of output swing) of output current, while the output swing between $I_{ph} = 0$ to 30 pA , is only $23 \mu\text{A}$.

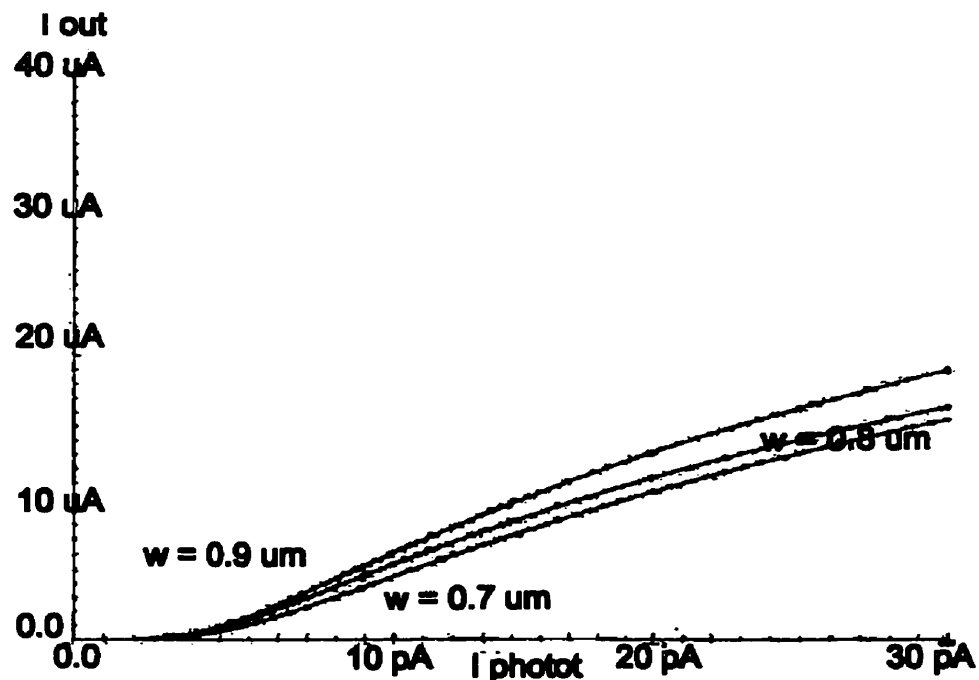


Figure A.3. Simulated effect of lithographic deviation on an inverted logarithmic pixel sensor. A partially little variation of output current is caused by the lithographical deviation of W/L : about 20% of output swing.

lithographic deviation) means that, while the photocurrent generates less voltage across the load, I_{ds} for M1 will be increased in partial compensation. In contrast, a higher than average W/L in the conventional pixel logarithmic (see Fig.A.1(a)) leads to an increase of the M1 V_g , which is compounded by the increased W/L of M1 itself. Simulations of the effects of the lithographic deviation, shown in Fig.A.2 and A.3, illustrate fixed pattern noise suppression of the inverted logarithmic pixel sensors. Variations of W/L of transistors in the conventional logarithmic pixel sensor produce a large variation of output current, about 142% variation of output swing (from $I_{ph} = 0$ and 30 pA), while generating only 18% in the inverted logarithmic pixel sensor. Hence, this inverted logarithmic pixel is expected to display reduced pattern noise and larger output swing than conventional logarithmic pixels, while maintaining continuous readout and wide optical dynamic range.

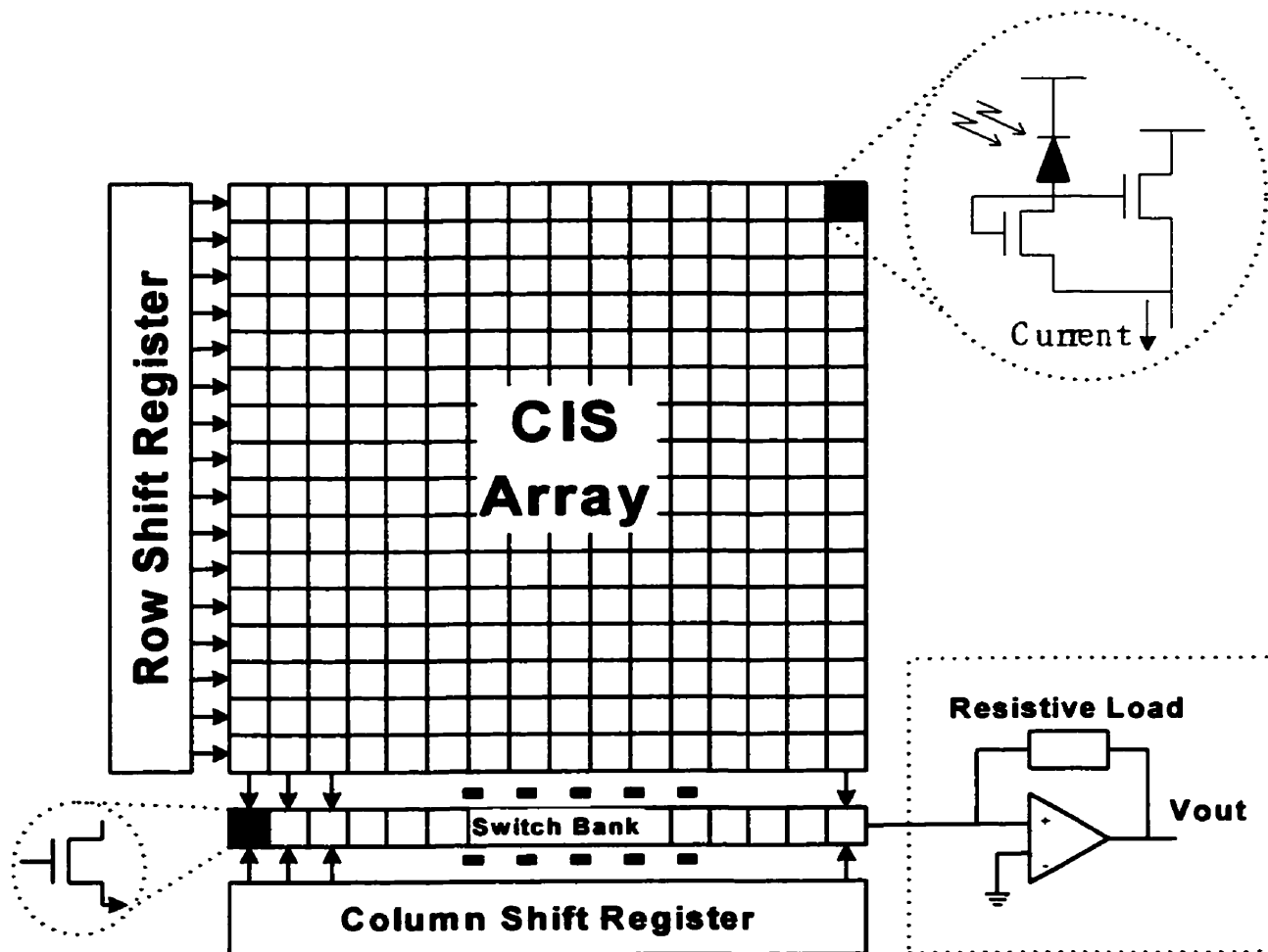
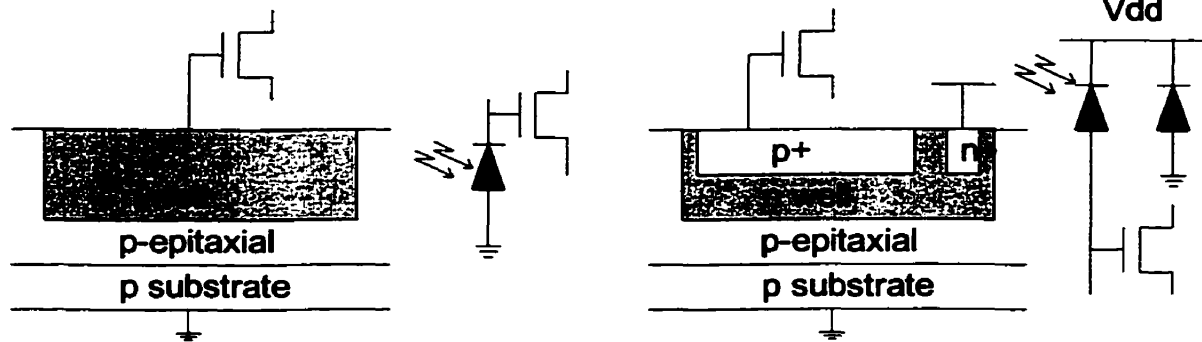


Figure A.4. Schematic view of the sensor structure. The output current is converted to a voltage by an external transresistance amplifier. The use of a single conversion circuit improves uniformity, and can be integrated on-chip.



(a) Single junction photodiode with floating diffusion at n-type side

(b) Double junction photodiode with floating diffusion at p-type side

Figure A.5. Structures of photodiode used for the inverted logarithmic pixel sensors.

A.3. Testing and Measurements

Here, this pixel structure has been implemented with a current-mode readout (Fig.A.4), where the column load is replaced by an off-chip transresistance amplifier. Since location of photodiode and loads is reversed, the layout of photodiode (shown in Fig.A.5(b)) is different from the normal one (Fig.A.5(a)). Current readout has well-known advantages of reduced column charging/discharging, low noise, and ease of analog signal processing. However, it is not normally implemented in integrating pixels owing to the difficulty of on-chip pattern noise correction; this is of lesser concern here because continuous pixels typically require off-chip pattern noise correction. In our case, reading out i_{out} also serves to decompress the logarithmic dependence of V_{out} on i_{photo} , since now $i_{out} \propto (V_{gs})^2 \propto [\ln(i_{photo})]^2$. In normal operation $V_{ref} = 0V$.

Photoresponse characteristics for single pixels with various numbers of load transistors are shown in Fig.A.6. A consequence of the structure shown in Fig.A.1(c) is that M1 operates in sub-threshold at low light intensities. Now $i_{out} \propto e^{V_{gs}}$ and $V_{gs} \propto \ln(i_{photo})$, giving a region where $i_{out} \propto i_{photo}$. At higher illumination, the $[\ln(i_{photo})]^2$ variation is observed. Fig. A.7(a)

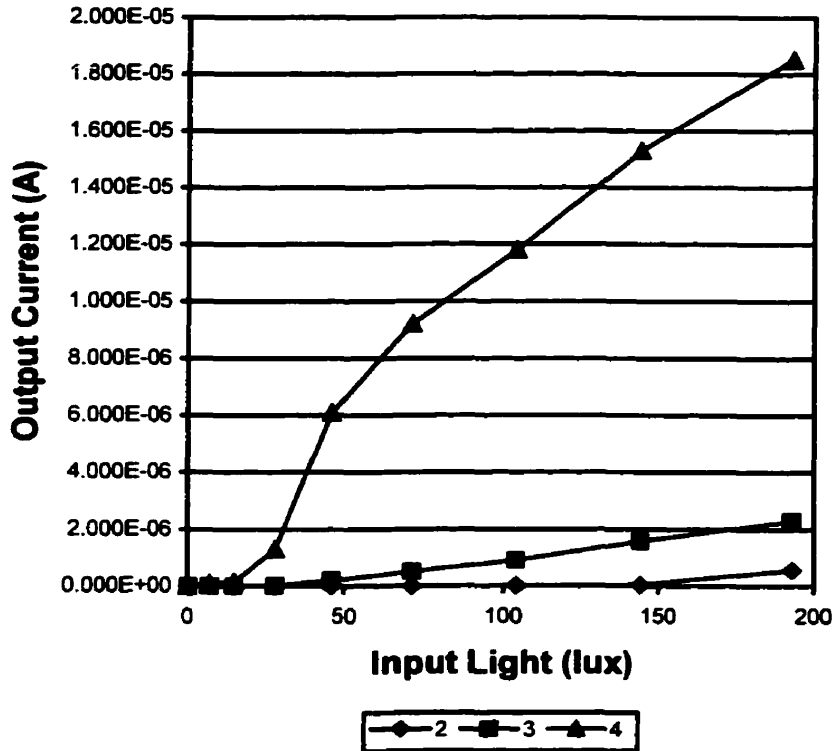
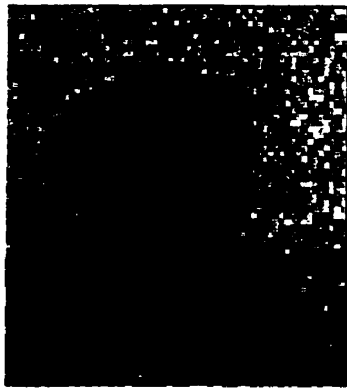


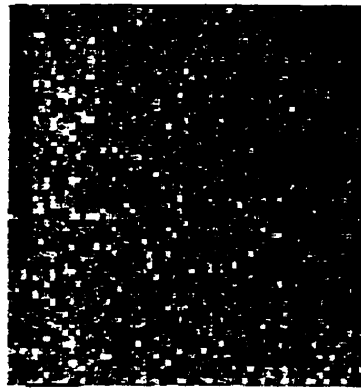
Figure A.6. Variation of the photoresponse of the inverted logarithmic pixel with number of load transistors.

shows a sample unprocessed image captured on a 64 x 64 array of 30 μ m pixels with 3 load transistors, implemented in a standard 0.35 μ m CMOS process (see also Fig.A.8), whose chip testing is summarized in Table A.1. While the image can clearly be seen (in contrast to many conventional logarithmic pixel sensors), pattern noise is still significant. Some of this is due to the fabrication process, which gives a high fixed pattern noise even for integrating mode sensors ($\sim 1.3\%$ of saturation). An image corrected by subtraction of a background reference is shown in Fig.A.7(b). To obtain best results, this reference image is a white image captured at the same average illumination as the original, indicating the presence of photo-response non-uniformity (PRNU). The PRNU for the sensor is plotted in Fig.A.9; this is to be compared with conventional logarithmic pixels where PRNU is typically $\sim 50\%$ of the mean.

To illustrate the advantages of current-mode readout for low voltage operation, the supply voltage has been reduced from its standard value of 3.3V (Fig.A.10); the sensor works well down to $V_{DD} = 2.5V$, but degrades rapidly thereafter. Variation of images with V_{ref} is shown



(a) Unprocessed Image



(b) White Background
Pattern Noise



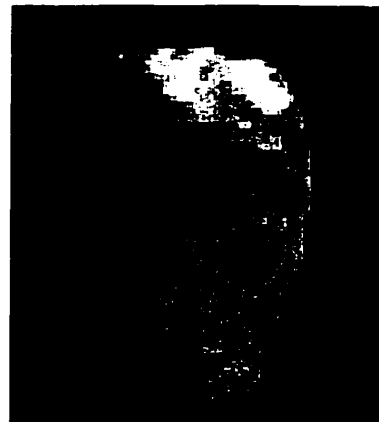
(c) Pattern Noise
Subtraction



(a) Unprocessed
Image



(b) Illumination Pattern
Noise



(c) Pattern noise
subtracted Image

Figure A.7. (a) Raw image captured under room light of approximately 200 lux, (b) White background image, (c) Image corrected by subtraction of a white image.

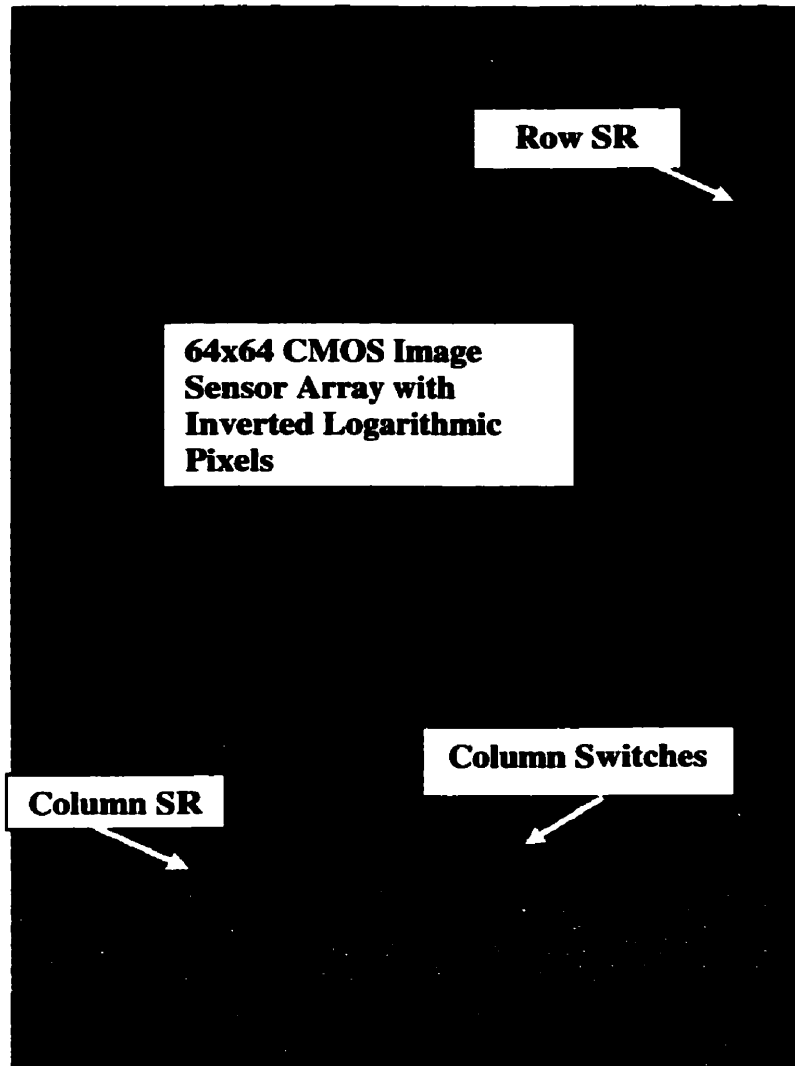


Figure A.8. Photograph of the image sensor die. Total die area is 16 mm².

Chip size	3204.5x3204.5 μm^2
Pixel size	30.8x30.8 μm^2
Format of array	64x64
Fill factor	81.63%
Maximum frame rate	200 KHz (Sampling Rate)
Power consumption	3.30 mA x 3.3 V = 10.89 mW at 50 KHz sampling rate
Background illumination	180 lux
Technology	0.35 μm CMOS
V _{DD} Power Supply	3.3 V
Output	analog output
Package	68 PGA

Table A.1. Electrical and Optical characteristics of the Inverted Logarithmic Sensor chip.

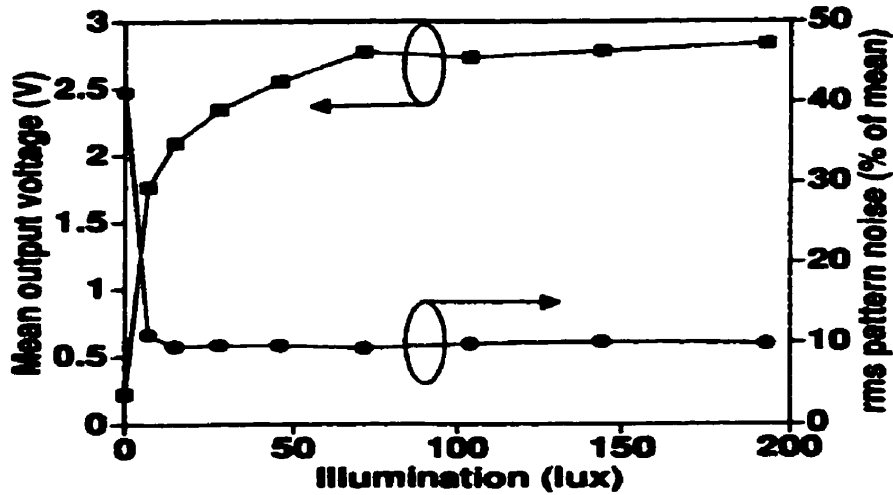


Figure A.9. Variation of rms pattern noise with illumination. In the absence of a well-defined saturation signal, pattern noise is expressed as a percentage of the mean output voltage at each point.

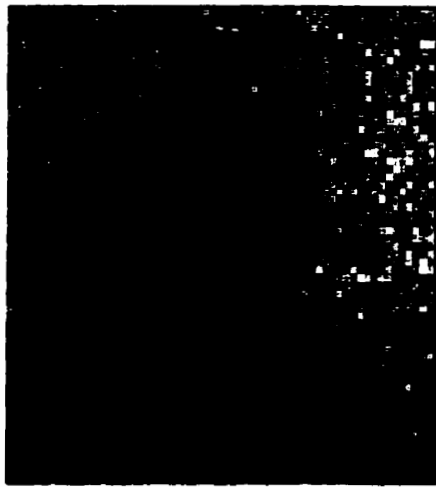
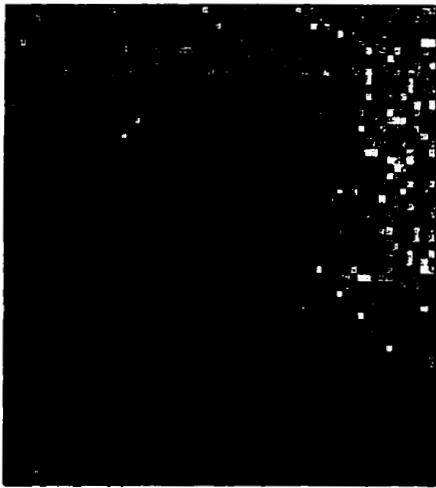
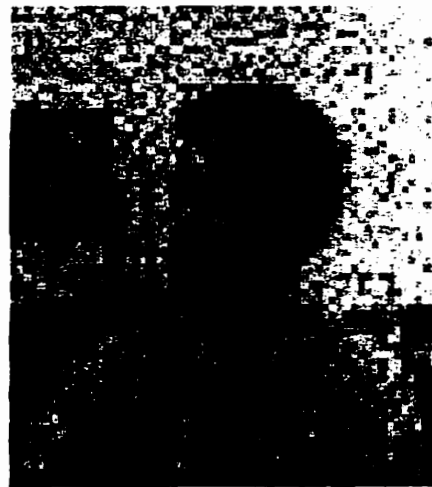
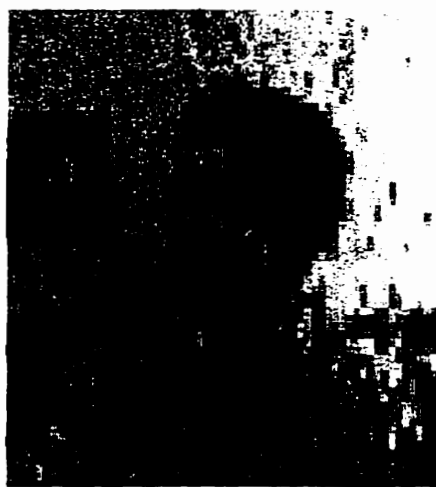
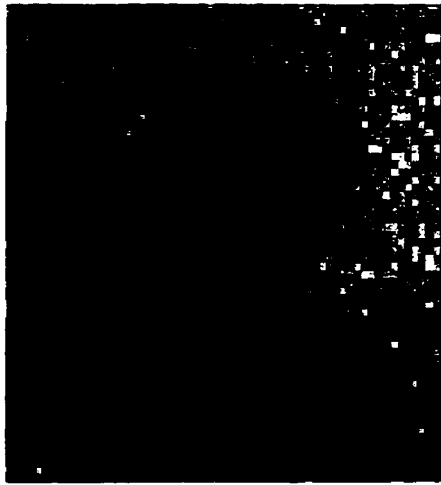
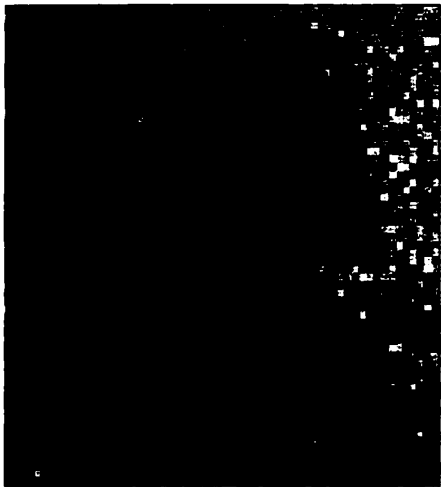
(a) $V_{dd} = 3.3 \text{ V}$ (b) $V_{dd} = 2.5 \text{ V}$ (c) $V_{dd} = 2.3 \text{ V}$

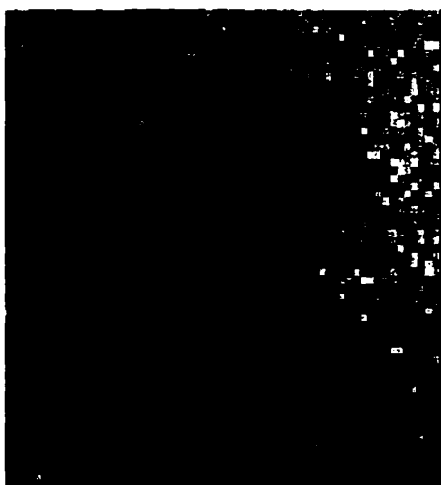
Figure A.10. Effect of image sensor V_{DD} on image quality. V_{DD} is nominally 3.3 V for this technology.



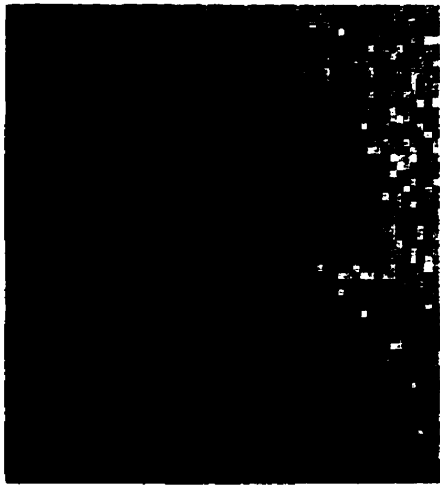
(a) $V_{in} = 0.1 \text{ V}$



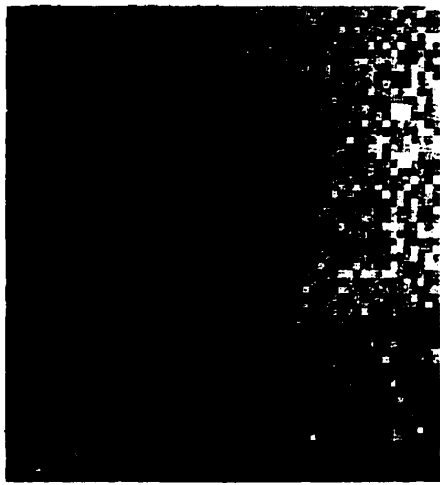
(b) $V_{in} = 0.2 \text{ V}$



(c) $V_{in} = 0.3 \text{ V}$



(d) $V_{in} = 0.4 \text{ V}$



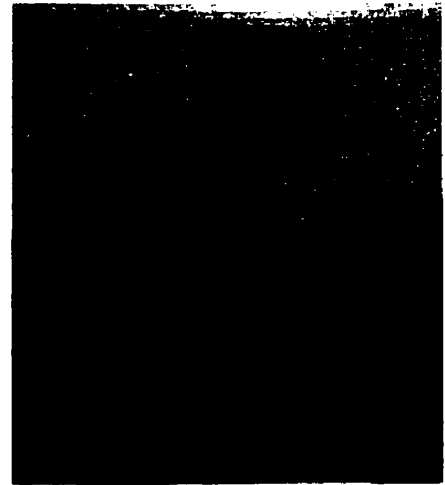
(e) $V_{in} = 0.5 \text{ V}$



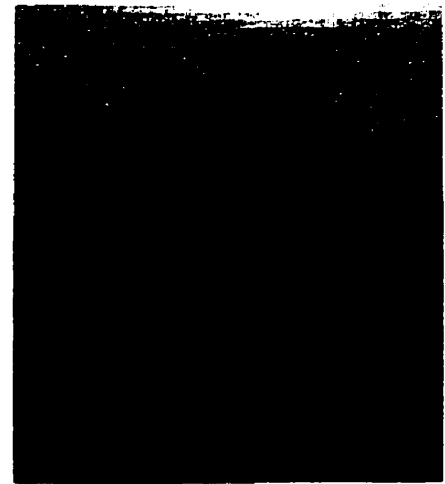
(f) $V_{in} = 0.7 \text{ V}$

Figure A.11. Effect of transresistance amplifier reference voltage on image quality.

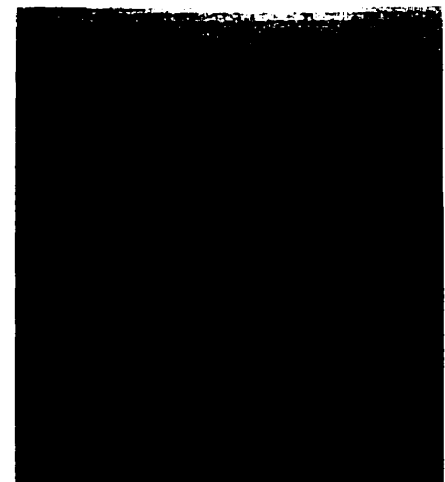
(a) Sampling
Rate = 10 KHz



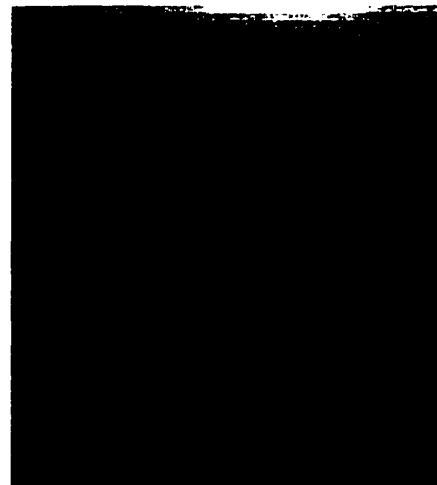
(b) Sampling
Rate = 20 KHz



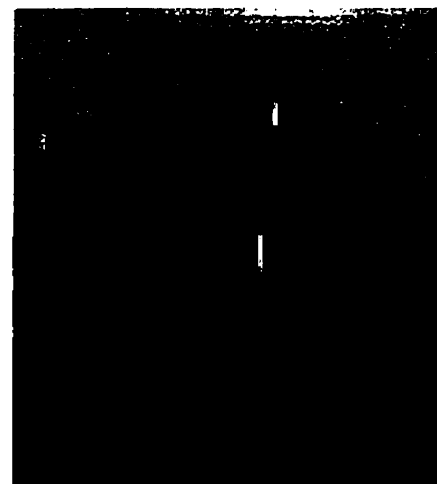
(c) Sampling
Rate = 50 KHz



(d) Sampling
Rate = 100 KHz



(e) Sampling
Rate = 200 KHz



(f) Sampling
Rate = 500 KHz

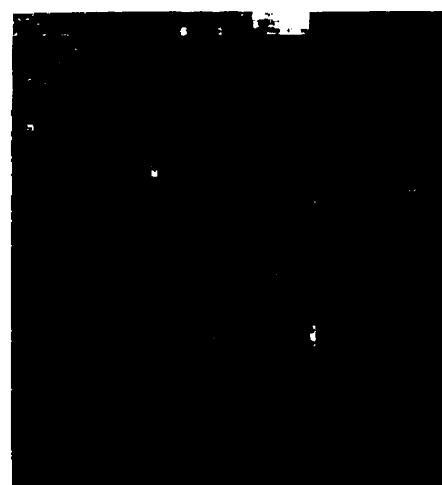
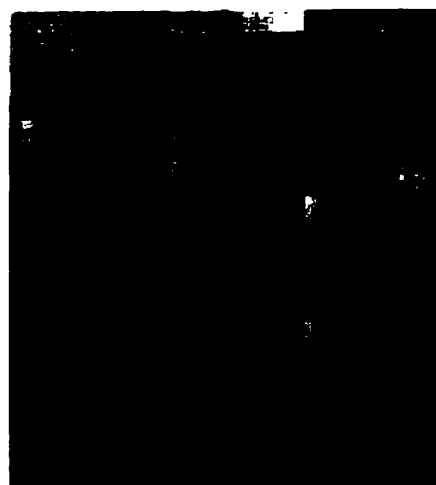


Figure A.12. Effect of data sampling rate on image quality. Because current readout does not have charging/discharging phenomena, it can achieve high frame rate.

in Fig.A.11, illustrating the relative independence of the pixel operation to column voltage, and hence insensitivity to the input resistance of any subsequent image processing stages. There are no charging/discharging phenomena in current readout mode, eliminating RC time constant. Thus, the inverted log pixel sensor with current readout can have high frame rate. Maximum data rates (directly related to frame rates) of active pixel sensors in integration mode typically depend on RC time constant in S/H and its output drivers. In contrast, the inverted log pixel sensor does not have any S/H's and output drivers, experiencing no time delay in data readouts. However, due to slow time response of logarithmic transistors in subthreshold region, there are degradations on images at high data rate. Fig.A.12 shows different images captured at different data sampling rates. With a particular processing technology, typical maximum data rate of APSs in integration mode is around 50 KHz, while the inverted log pixel does not experience any degradations on images at 100 KHz, generating higher frame rate.

A.4. Conclusions

The reversed arrangement of the photodiode and load causes the voltage generated across the load by the photocurrent to appear directly as the gate-source voltage of the in-pixel buffer transistor. This configuration eliminates the dependence of the voltage swing on the column load. Pattern noise is also reduced over conventional logarithmic pixels because global variations of threshold voltage are less significant. In addition, a readout technique of the pixel sensor demonstrates reduced signal compression, improved output swing and increased frame rate. However, the independence of the load is not as effective as expected: images are degraded when V_{ref} is higher than 0.7 V. Although pattern noise is expectedly reduced, there are still noticeable degradations by the pattern noise and thus further processing on the images is required.

Appendix B: Basic Procedures for Image

Capture Test

The very first step in the CMOS image sensor test is to capture an image of the best quality with the chip, verifying test board connections, control input patterns, image display software setup and, most importantly, the design of the chip. Here is the procedure of the image-capturing test.

1. Wiggling Test

Without a lens, as a light source is turned on and off (or a light source is swirled in front of the image sensor chip) in a dark room, the output voltage of the chip should go up and down and the images in the display should become bright and dark, indicating the chip is capable of sensing the incident light. Generally, the wiggling test verifies the input control patterns, the image display system and the test board connections. It does not give a full verification of the setup, but gives a basic setup to start the test with.

2. Flash Light Circle Test

With an approximate setup of a lens (focal length adjustments and alignments to the chip), as a light source (e.g. a flash light) is turned on in front of the image chip in a dark room, the output image should contain a white circle. If any circles cannot be captured, the bias voltages can be changed until an appropriate shape of the circle is captured. If any circles cannot be obtained with many different bias voltages, check the input patterns, the image display system and the test board connections. This step helps to find the appropriate bias voltages for the chip.

3. Final Image Capture

With the same setups (input patterns, display system, bias voltages and test board), a stationary object is placed in front of the chip with an appropriate illumination. As the focal distance and the alignment of the lens to the chip are changed, the chip should capture the stationary object. The first image might be blurred, but an appropriate adjustment of the lens will sharpen the image and give an image of the best quality with the chip.

Appendix C: Image Sensor Characteristics

C.1. Basic Measurements

- Measure the frame rate (sampling rate) at which best image quality for a same image is obtained.
- Measure the power consumption (nominal current flowing from Vdd) for various images at the frame rate.
- Measure and save image files at a fixed wavelength and at a fixed frame rate while changing the illumination (light power or intensity, but lux is preferred) from 0 to until the output voltage is saturated. In addition, the wavelength and frame rate can be varied for another test. This measurement can be directly used for photosensitivity, PRNU, saturation level. Also, it can be used for calculation of conversion efficiency.
- The above measurement is under an assumption that there is enough settling time between the different illuminations (due to light temperature). After changing the light intensity of the light source, the light intensity fluctuates until it settles down at a constant value. Sometimes it is difficult to avoid the light temperature effects. Instead of changing the light intensity, the integration time is changed here. Measure and save image files (> 100 files at each Tint) at a fixed light illumination and at a fixed wavelength (typically at 540 nm of green light) as changing the integration time. For each image file, the mean and variance is calculated.
- Measure and save image files at fixed illumination (light power, w/m^2 , is preferred) and a fixed frame rate as changing wavelength of the incoming ray using a monochrometer. The illumination and frame rate can be varied for another measurement. This measurement can be used for spectral response.
- Measure and save image files in dark room as changing integration time (sampling rate). This measurement can be used directly for FPN and dark current. If you want to measure temporal noise save enough image files (>100 files) under a careful environment setup because the temporal noise is very sensitive to any environment change

C.2. Imager Characteristics Extraction and Calculation

Fill factor [%]

Photo-sensitive area / total pixel area

Frame rate [frames/second]

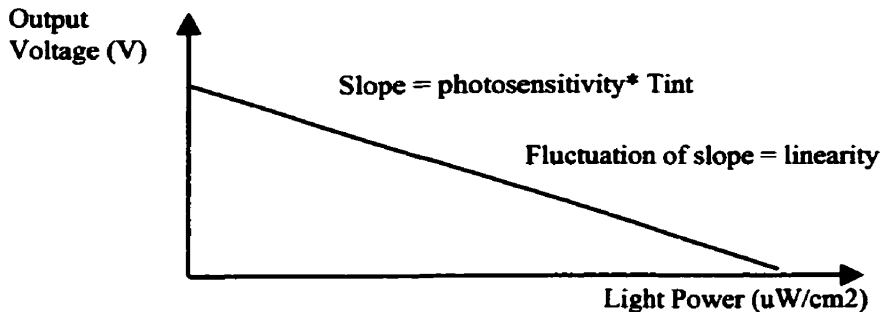
The maximum frame rate (sampling rate)

Power [mW]

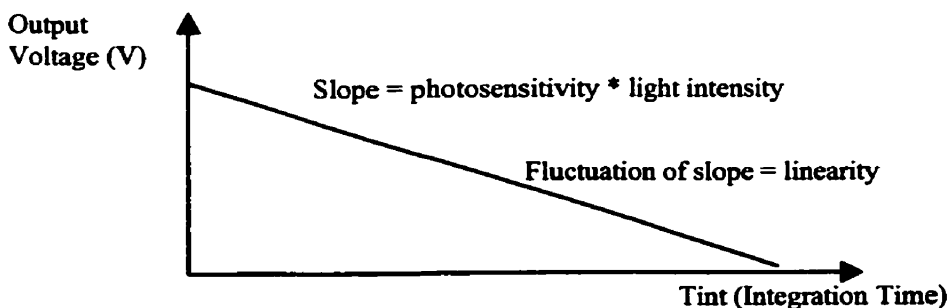
Measure power (current from Vdd node of the power supply) when the imager captures normal image (background of the lab) and in the dark room

Photosensitivity [V/(lux*sec)] & Linearity

(1) As different light intensity (lux is preferred to light power, w/m^2) shines on the sensor array, measure the output voltage at a given frame rate (sampling rate)



(2) As the integration time changes, measure the mean value of the output voltage at a fixed illumination



Quantum Efficiency

With the previous measurements of saturation level, conversion gain, the QE can be calculated as follows

$$QE = \text{Saturation Level} / (\text{light power} * \text{Effective photodiode area} * \text{reflection loss} * T_{int} * g_c)$$

Conversion Efficiency

(1) After measuring the output voltage (or ΔV) at a given light intensity (light power) when the output voltage is saturated, calculate the total number of electrons generated by the light intensity with assumed QE, optical reflection, fill factor and photodiode capacitance. Then the conversion efficiency is the saturation level divided by the total number of the photon-generated electrons.

$$\text{Total \# of photon-generated electrons } (n_e) = (P \times A_{\text{eff}} \times QE \times R \times T_{\text{int}}) / E_{\text{photon}}$$

Where

P = light power per unit area

A_{eff} = photo-sensitive area (approximately pixel area \times fill factor)

QE = Quantum efficiency (about 40%~60%)

R = Optical reflection (about 60 ~ 70%)

T_{int} = integration time

E_{photon} = Energy per photon ($E_{\text{ph}} = hf = 1.24 / \text{wavelength in nm}$)

Conversion Efficiency = Saturation level / n_e

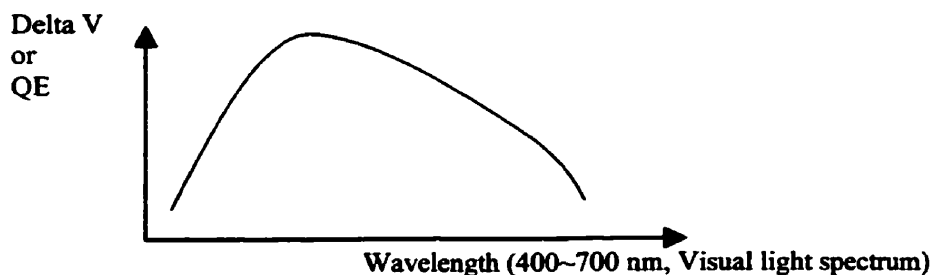
(2) Calculate the mean and the variance of the images captured at a fixed T_{int} . The conversion efficiency is simply

$$g_c = \text{variance} / \text{mean}$$

This calculation is repeated for different T_{int} . The same resultant value should be obtained for the different T_{int} .

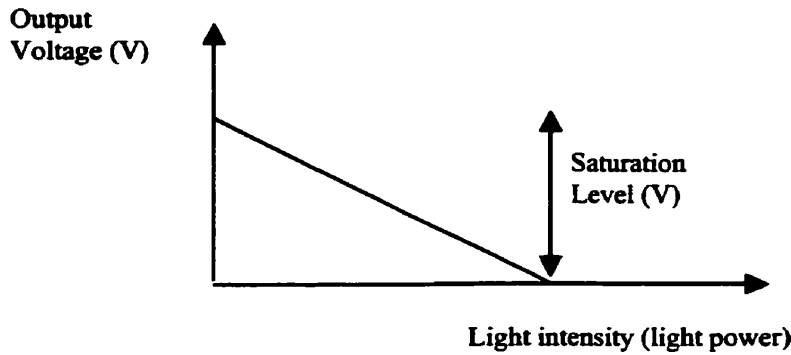
Spectral Response

Measure output voltage (or $\Delta V = V_{\text{out in dark}} - V_{\text{out at light}}$) at different wavelengths (different wavelength filters in the monochromator) at a fixed light illumination

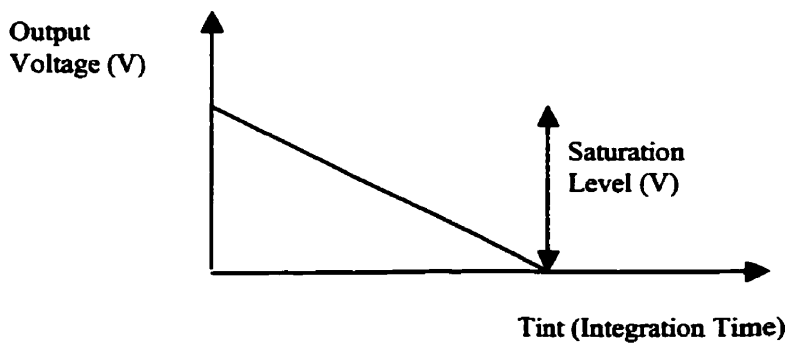


Saturation Level

(1) It is the maximum output voltage swing. As the light power increases at a given frame rate (sampling rate), the maximum ΔV will saturate to a minimum voltage value. The saturation level = Highest output voltage – Lowest output voltage.

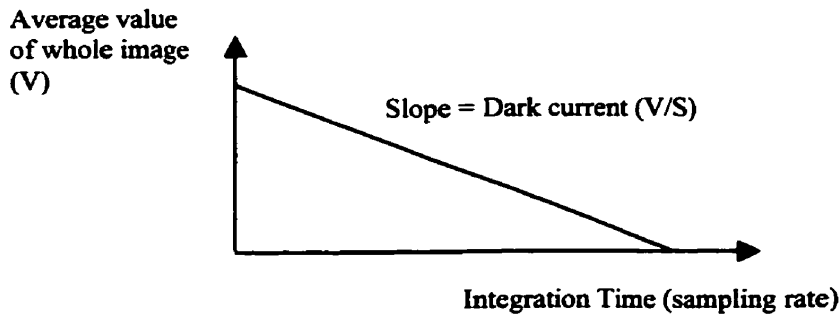


(2) As Tint changes, plot the mean output values of the images along with the integration time.



Dark Current

Measure and save the whole frame (whole image) in the dark room as changing integration time (sampling rate).



Fixed Pattern Noise (FPN)

Measure and save the whole frame (whole image) in the dark room and calculate the variance/standard deviation of the image file in V_{pp} or V_{rms} or % (V_{rms} / saturation level)

PRNU (Photo-Response Non-Uniformity)

(1) Measure and save the whole frame (whole image) at different light power (intensity) and at a fixed frame rate and at a fixed wavelength. Then calculate the variance/standard deviation of the image file in V_{rms} or $V_{p/p}$ or % (V_{rms} / mean)

(2) Measure and save the image files at different Tint and at a fixed light illumination and at a fixed wavelength. Then calculate the variance/standard deviation of the image files in Vpp, Vrms and %.

Temporal Noise

Measure the consecutive samples of the output voltage for one pixel in the array with different Tint (typically the value measured in the dark room is used for SNR and DR). The number of samples should be large (>100) and careful environment is required because the temporal noise measurement is very sensitive to the environment.

Signal to Noise Ratio (SN or SNR)

Calculated as Saturation level / output temporal noise in the dark

Dynamic Range

Calculated as Saturation intensity / Temporal noise equivalent intensity. It should be same or nearly same as SN if the photoresponsivity is linear (Output voltage difference is proportional to input light intensity)

Photon Shot Noise, KTC noise

They either cannot or difficult to be measured separately. They are included in part of temporal noise or readout noise.

Effective Capacitance

Measured from test structure of photodiode.

Measured from calculation with light power, QE, photosensitive area (fill factor), optical reflection and output voltage. The total # of photons generated by the light intensity is calculated and then degradation by QE, photosensitive area and optical reflection is applied to the total # of photons. Then the effective capacitance = charge of photon-generated electrons / output voltage (ΔV)

C.3. Image Sensor Characteristics

Quantity	Unit	How to obtain	Range for commercial devices
Technology	μm		
Vdd	V		< 3.3 V
Output format	Analog or digital		
Chip Size	mm x mm	Physical measure	> 4 x 4 mm ²
Pixel Size	$\mu\text{m} \times \mu\text{m}$	Physical measure	< 10 x 10 μm^2
Format of Array		Physical measure	> 64 x 64
Fill Factor	%	Physical measure	> 40 %
Max. Pixel rate	Pixels/sec Or MHz	Single serial port	40 Mhz (megapixels/sec) (analog) 100 Mhz (digital)
Frame Rate	Frames/sec or ms	Fastest frame rate with reasonable image quality	> 30 frames/sec
Integration Time	Second		From the frame time (e.g. 33 milliseconds) to less than one row read time (e.g. a few microseconds)
Power	W	Power at the nominal frame rate	30 ~ 100 mW for VGA format (640 x 480)
Illumination Lux or Flux	lux or W/m ²	Illumination in environment 1 W/m ² = 70lx (visible white light) to 180lx (visible + NIR)	150~250 lux for room light
Photo-sensitivity	V / (lx*Sec) Or (V/s)/(W/m ²)	Output voltage [V/s] vs. Input Light power [W/m ²] Output voltage [V/s] vs. Input Light lux [lx]	0.7 ~ 3.5 V/lux*sec

Spectral Response	A/W	Average output current (or QE) in unit area, at wavelengths under a light power [W/m^2], OR Ratio between photo current and light power for a given wavelength.	
Conversion Efficiency	$\mu\text{V}/e^-$	Output voltage per unit of input signal charge Variance / Mean of output	3 ~ 35 $\mu\text{V}/e^-$ (output referred)
Saturation Range (Level)	V	Max output voltage in dark – Min output voltage in bright	500 mV ($V_{\text{dd}} = 3.3\text{V}$) ~ 2 V ($V_{\text{dd}} = 5\text{V}$)
PRNU (Photo Response Non Uniformity)	V_{rms} or V_{pp} or %	V_{rms} at a given wavelength and illumination % = $V_{\text{rms}} / \text{mean level}$ or $V_{\text{pp}} / \text{mean level}$	1~10 % V_{pp} or 0.8~2 % V_{rms} of mean level 1 mV ~ 50 mV V_{rms}
Fixed Pattern Noise (FPN)	V_{rms} or V_{pp} or %	Static spread of (dark) voltages of all pixels of array $V_{\text{pp}} / \text{saturation level}$	1 mV ~ 30 mV rms < 1 % V_{pp}
Temporal Noise (N)	V_{pp} or V_{rms} or %	RMS of consecutive samples of the output voltage for one pixel	< 200 μV rms
Signal to Noise Ratio (S/N, SNR)	dB	Output signal voltage range / output signal noise in the dark	>40 dB
Dynamic Range (DR)	dB	Saturation intensity / Noise equivalent intensity If input light is linear with output signal, SNR = DR	>50 dB
Dark Signal	V/Sec	Signal voltage drop in the dark, due to dark current (Output voltage in dark – output voltage) / Integration time	0.1 ~ 1.6 V/sec

Dark Current	A or A/cm²	(Apparent) photodiode current in the dark per pixel, or normalized per unit area	300 pA/cm²
Quantum Efficiency	%	# of generated electrons / # of "impinging photons" Or QE = SR x hv / q	20% (photogate) ~ 40% (photodiode)
Effective Capacitance	fF	Photo charge / Output voltage	10 fF ~ 100 fF
Pixel Current	A		100 fA ~ 10 pA
Input referred Read Noise	e⁻	All the noise measured and input referred	15 e⁻ (photogate) ~ 50 e⁻ (photodiode)

References

- [1] Eric R. Fossum, "CMOS Image Sensors: Electronic Camera-On-A-Chip", IEEE Transactions on Electron Devices. Vol.44, No.10, pp.1689-98, Oct.97
- [2] S. Morrison. "A new type of photosensitive junction device", Solid-State Electron, Vol.5, pp.485-94, 1963
- [3] J. Horton, R. Mazza, and H. Dym, "The scanistor-A solid-state image scanner", in Proc. IEEE, Vol.52, pp.1513-28, 1964
- [4] M.A. Schuster and G. Stull, "A monolithic mosaic of photon sensors for solid state imaging applications", IEEE trans. Electron Devices, Vol.ED-13, pp.907-12, 1966
- [5] G.P. Weckler, "Operation of p-n junction photodetectors in a photon flux integration mode", IEEE J. Solid-State Circuits, Vol.SC-2, pp.65-73, 1967
- [6] R. Dyck and G. Weckler, "Integrated arrays of silicon photodetectors for image sensing", IEEE Trans. Electron Devices, Vol.ED-15, pp.196-201, 1968
- [7] P. Noble, "Self-scanned silicon image detector arrays", IEEE Trans. Electron Devices, Vol.ED-15, pp.202-209, 1968
- [8] W.S. Boyle and G.E. Smith, "Charge-coupled semiconductor devices", Bell Syst. Tech. J. Vol.49, pp.587-93, 1970
- [9] R. Melen, "The tradoff in monolithic image sensors: MOS versus CCD", Electron., Vol.46, pp.106-111, May 1973
- [10] S. Ohba et al., "MOS area sensor: Part II-Low noise MOS area sensor with anti-blooming photodiodes", IEEE Trans. Electron Devices, Vol.ED-27, pp.1682-7, Aug. 1980
- [11] K. Senda, S. Terakawa, Y. Hiroshima, and T. Kunii, "Analysis of charge-priming transfer efficiency in CPD image sensors", IEEE Trans. Electron Devices, Vol.ED-31, pp. pp.1324-8, Sept. 1984
- [12] H. Ando et al., "Design consideration and performance of a new MOS imaging device", IEEE Trans. Consumer Electronics, Vol.ED-32, pp.1484-9, May 1985
- [13] D. Renshaw, P. Denyer, G. Wang and M. Lu, "ASIC image sensors", IEEE Int. Symposium of Circuits and Systems, pp.3038-41, 1990
- [14] S. Mendis, S. Kemeny, and E. Fossum, "CMOS active pixel image sensor", IEEE Trans. Electron Devices, Vol.ED-41, pp.452-3, 1994

- [15] Alireza Moini, "Vision chips or seeing silicon", Technical Report, Centre for High Performance Integrated Technologies and Systems, The University of Adelaide, March 1997, (www.eleceng.adelaide.edu.au/Groups/GAAS/Bugeye/visionchips/index.html).
- [16] J.J. Zarnowski, M. Pace, M. Joyner "1.5 FET-per-pixel standard CMOS active column sensor", Proceedings of SPIE, Vol.3649-27
- [17] M.N. Al-Awa, et al. "A Real Time Vision Architecture using a Dynamically Reconfigurable Fast Bus", Image Processing and Its Applications, pp.470-4, July 1995
- [18] Doug Tody, "The Data Handling System for the NOAO Mosaic", Astronomical Data Analysis Software and Systems VI, ASP Conf. Series 125, pp.451-4
- [19] Qinglian Guo, et al, "Generation of High-Quality Images for Tele-medicine and Tele-pathology Efforts", Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, vol.20, No.3, pp.1288-91, 1998
- [20] Sunetra Mendis, "CMOS Active Pixel Image Sensors with On-Chip Analog-to-Digital Conversion", Ph.D. thesis at Columbia University, 1995
- [21] P. Lee, et al. "An active pixel sensor fabricated using CMOS/CCD process technology", 1995 IEEE Workshop on CCDs and Advanced Image Sensors, Dana Point, CA
- [22] C.Y. Wu and C.Chiu, "A new structure of the 2-D silicon retina", IEEE Journal of Solid-State Circuits, Vol.30, No.8, August 1995
- [23] E. Vittoz, "Analog VLSI signal processing: Why, where and how?" Analog Integrated Circuits and Signal Processing, Vol. 6, pp. 27-44, 1994
- [24] G.R. Nudd, et al. "A charge-coupled device image processor for smart sensor applications", SPIE Proc. Vol 155, pp.15-22, 1978
- [25] T. Knight, "Design of an integrated optical sensor with on-chip processing", PhD thesis, Dept. of Electrical Engineering and Computer Science, MIT, Cambridge, Mass., 1983
- [26] C.L. Keast and C.G. Sodini, "A CCD/CMOS-based imager with integrated focal plane signal processing", IEEE Journal of Solid State Circuits, Vol. 28, No. 4, pp.431-7, 1993
- [27] A.M. Chiang, M.L. Chuang, "A CCD programmable image processor and its neural network applications", IEEE Journal of Solid State Circuits, Vol. 26, No. 12, pp. 1894-1901, 1991
- [28] W. Yang, "Analog CCD processors for image filtering", SPIE Proc., Vol. 1473, pp.114-27

- [29] E.R. Fossum, "Architectures for focal plane processing", *Optical Eng.*, Vol.28, No.8, pp. 866-871, 1989
- [30] Z. Zhou, B. Pain, E. Fossum, "Frame-transfer CMOS Active Pixel Sensor with pixel binning," *IEEE Trans. On Electron Devices*, Vol. ED-44, pp.1764-8, 1997
- [31] Abbas El Gammal, "Pixel Level Processing – Why, What and Why?" *SPIE*, Vol. 3650, pp.2 - 13, 1999
- [32] Bedabrata Pain, "Approaches and analysis for on-focal-plane analog-to-digital conversion", *SPIE*, Vol. 2226, 1994
- [33] T. Chen, P. Catrysse, A. El Gamal, and B. Wandell, "How Small Should Pixel Size Be?" *SPIE*, Vol. 3965, San Jose, CA, January 2000
- [34] David X. D. Yang, "A 640 x 512 CMOS Image Sensor with Ultrawide Dynamic Range Floating-Point Pixel-Level ADC", *IEEE Journal of Solid-state Circuits*, Vol. 34, No. 12, pp. 1821 – 34, 1999
- [35] Kiyoharu Aizawa, "Computational Image Sensor for On Sensor Compression", *IEEE Transactions on Electron Devices*, Vol. 44, No. 10, pp. 1724 – 30, 1997
- [36] Miguel Arias-Estrada, "Computational Motion Sensors for Autoguided Vehicles", 30th ISATA Conference on Robotics, Motion and Machine Vision in Automotive Industry, pp. 101 – 8, 1997
- [37] Orly Yadid-Pecht, "CMOS Active Pixel Sensor Star Tracker with Regional Electronic Shutter", *IEEE Journal of Solid-state Circuits*, Vol. 32, No. 2, pp.285 – 8, 1997
- [38] Makoto Nagata, "A Minimum Distance Search Circuit using Dual-Line PWM Signal Processing and Charge Packet Counting Techniques", *ISSCC97 (97 International Solid State Circuits Conference)*, pp. 42 –3, 1997
- [39] M.J.M. Pelgrom, A.C.J. Duinmaiger, A.P.G. Welbers, "Matching properties of MOS transistors", *IEEE Journal of Solid-state Circuits*, Vol. 24, pp.1433-40, 1989
- [40] R. Forchheimer, A. Astrom, "Near-sensor image processing: a new paradigm", *IEEE Transactions on Image Processing*, Volume: 3 Issue: 6, pp.736 -746, Nov. 1994
- [41] S. Jung, R. Thewes, T. Scheiter; Goser, K.F.; Weber, W."A low-power and high-performance CMOS fingerprint sensing and encoding architecture", *IEEE Journal of Solid-State Circuits*, Volume: 34 Issue: 7, pp. 978 –984, July 1999
- [42] G. Matheron, "Elements pour une Theorie des Milieux Poreuz," *Masson*, Paris, 1967

- [43] P. Maragos, R.W. Schafer. "Morphological filters. Part I: Their set-theoretic analysis and relations to linear shift-invariant filters. Part II: Their relations to median order-statistic, and stack filters", IEEE Trans. Acoust. Speech Signal Processing Vol. 35, pp.1153-84, 1987
- [44] J.S.J. Lee, R.M. Haralick, L.G. Shapiro, "Morphologic edge detection", IEEE Trans. Robotics Automation, RA-3, pp.142-56, 1987
- [45] P. Maragos, R.W. Schafer, "Morphological systems for multidimensional signal processing", Proc. IEEE Vol. 78, pp. 690-710, 1990
- [46] L.F.C. Pessoa, P. Maragos, "MRL-filters: A general class of nonlinear systems and their optimal design for image processing", IEEE trans. Image Processing, Vol. 7, pp.966-78, 1998
- [47] J.G.M. Schavemaker, M.H. Reinders, R. Van den Boomgaard, "Image sharpening by morphological filtering", IEEE Workshop on Nonlinear Signal & Image Processing MacKinac Island, Michigan, Sept. 1997
- [48] D. Schonfeld, J.Goutsaias, "Optimal morphological pattern restoration from noisy binary images", IEEE Trans. Pattern Analysis Machine Intelligence Vol. 13, pp. 14-29, 1991
- [49] N.D. Sidiropoulos, J.S.Baras, C.A. Berenstein, "Optimal filtering of digital binary images corrupted by union/intersection noise", IEEE Trans. Image Processing, Vol. 3, pp. 382-403, 1994
- [50] H. J.A.M. Heijmans, C. Ronse. "Annular Filters for Binary Images", IEEE Transactions on image processing, Vol. 8. No 10. pp.1330-40, 1999
- [51] E. Oron, A. Kumar, Y. Bar-Shalom, "Precision tracking with segmentation for imaging sensors", IEEE Transactions on Aerospace and Electronic Systems, Volume: 29 Issue: 3, pp.977 -987, July 1993
- [52] T.G. Morris, S.P. DeWeerth, "Analogue VLSI morphological image processing circuit", Electronic letters, Vol. 31, No. 23, pp1998-9, 1995
- [53] R.K. Krishnamurthy, R. Sridhar, "A CMOS wave-pipelined image processor for real-time morphology", Computer Design: 1995 IEEE International Conference on VLSI in Computers and Processors, 1995. ICCD '95. Proceedings, pp.638 -643, 1995
- [54] E. O'Rourke, J.B. Foley, "Specification, design and implementation of a digital binary image processing ASIC", IEE Colloquium on Applications Specific Integrated Circuits for Digital Signal Processing, pp.8/1 -8/5, 1993
- [55] M.T. Rigby, G.J. Awcock, "VLSI design methodologies for application specific binary sensors", Sixth International Conference on Image Processing and Its Applications, 1997, Volume: 1, pp. 166 -170, 1997

- [56] J.D. Legat, P. De Muelenaere, "A high performance SIMD processor for binary image processing", Proceedings of the IEEE Custom Integrated Circuits Conference, pp. 17.4/1 - 17.4/4, 1990
- [57] M. Schwarzenberg, M. Traber, M. Scholles, R. Schuffny, "A VLSI chip for wavelet image compression" IEEE International Symposium on Circuits and Systems, 1999. ISCAS '99. Proceedings of the 1999, Volume: 4, pp.271 -274, 1999
- [58] R. Dominguez-Castro, S.Espejo, A. Rodrigues-Vazques, R. A. Carmona, P.Foldesy, A. Sarandy, P. Szolgay, T. Sziranyi, T. Roska, "A 0.8 μm CMOS Two-Dimensional Programmable Mixed-Signal Focal-Plane Array Processor with On-Chip Binary Imaging and Instructions Storage", IEEE Journal of Solid-State Circuits, Vol. 32, No. 7, pp.1013-26, 1997
- [59] T. Nezuka, T. Fujita, M. Ikeda, K. Asada, "A binary image sensor with flexible motion vector detection using block matching method", Asia and South Pacific Design Automation Conference, 2000. Proceedings of the ASP-DAC 2000, pp.21 -22, 2000
- [60] S. Jung, R. Thewes, T. Scheiter, K.F. Goser, W. Weber, "A low-power and high-performance CMOS fingerprint sensing and encoding architecture", IEEE Journal of Solid-State Circuits, Volume: 34 Issue: 7, pp. 978 -984, July 1999
- [61] L. Zheng, K. Aizawa, M. Hatori, "Implementation of a 2D motion vector detection on image sensor focal plane," IEEE International Symposium on Circuits and Systems, 1999. ISCAS '99. Proceedings of the 1999, Volume: 5, pp.156 -159, 1999
- [62] N. Bourbakis, N. Steffensen, B. Saha, "Design of an array processor for parallel skeletonization of images", IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, Volume: 44 Issue: 4, pp.284 -298, April 1997
- [63] W.C. Fang, T. Shaw, J. Yu, J. B. Lau, Y.C. Lin, "Parallel morphological image processing with an opto-electronic VLSI array processor", 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1993. ICASSP-93, Volume: 1, pp. 409 -412, 1993
- [64] W. Yang, "A charge coupled device architecture for on focal plane image signal processing", 1989 International Symposium on VLSI Technology, Systems and Applications, 1989, Proceedings of Technical Papers, pp.266 -270, 1989
- [65] S. Kawahito, D. Handoko, Y. Tadokoro, "A CMOS image sensor with motion vector estimator for low-power image compression", Instrumentation and Measurement Technology Conference, 1999. IMTC/99. Proceedings of the 16th IEEE , Volume: 1, pp.65 -70, 1999
- [66] T. Morris, E. Fletcher, C. Afghahi, S. Issa, K. Connolly, J.C. Korta, "A column-based processing array for high-speed digital image processing", 20th Anniversary Conference on Advanced Research in VLSI, 1999. Proceedings, pp.42 -56, 1999

- [67] K. Chen, A. Astrom, P.E. Danielsson, "PASIC: a smart sensor for computer vision", *Pattern Recognition*, 1990. Proceedings, 10th International Conference on, Volume: ii, pp.286-291, 1990
- [68] J.C. Gealow, C.G. Sodini, "A pixel-parallel image processor using logic pitch-matched to dynamic memory", *IEEE Journal of Solid-State Circuits*, Volume: 34 Issue: 6, pp.831-839, June 1999
- [69] Y. Ni, J. Guan, "A 256x256 Pixel Smart CMOS Image Sensor for Line-Based Stereo Vision Applications", *IEEE Journal of Solid state circuits*, Vol. 35, No. 7, pp.1055-61, 2000
- [70] Z. Zhou, B. Pain, R. Panicacci, B. Mansoorian, J. Nakamura, E.R. Fossum, "On-focal-plane ADC: Recent progress at JPL", *SPIE*. Vol. 2745, pp.111-122
- [71] A. Simoni, G. Torelli, F. Maloberti, A. Sartori, M. Gottardi, L. Gonzo, "256x256 Pixel CMOS digital camera for computer vision with 32 algorithmic ADCs on board", *IEE. Processing of Circuits Devices Systems*, Vol. 146, No. 4, pp. 184-190, 1999
- [72] S. Kawahito, M. Yoshida, M. Sasaki, K. Umehara, D. Miyazaki, Y. Tadokoro, K. Murata, S. Doushou, A. Matsuzawa, "A CMOS Image Sensor with Analog Two-dimensional DCT-Based Compression Circuits for One-Chip Cameras", *IEEE Journal of Solid-state Circuits*, Vol.32, No.12, pp.2030-41, 1997
- [73] C.K. Chow, T. Kaneko, "Automatic Boundary Detection of the Left Ventricle from Cineangiograms", *Computer and Biomed. Res.* Vol. 5, pp. 388-401
- [74] Gonzalez and Woods, *Digital Image Processing*, Addison Wesley, 1993
- [75] S.E. Umbaugh, *Computer Vision and Image Processing*, Prentice Hall PTR, 1999
- [76] A. Bovik, *Handbook of Image & Video Processing*, Academic Press, 2000
- [77] C. Koch, H. Li, "Vision Chips Implementing Vision Algorithms with Analog VLSI circuits", *IEEE Computer Society Press*, 1995
- [78] H. Kobayashi, L. White, A.A. Abidi, "An active resistor network for gaussian filtering of images," *IEEE Journal of Solid-State Circuits*, Vol. 26, No. 5, pp.737-748, May 1991
- [79] J.Nakamura, B.Pain, E.R. Fossum. "On-Focal-Plane Signal Processing for Current-Mode Active Pixel Sensors", *IEEE Transactions on Electron Devices*, Vol. 44, No. 10, pp.1747-57, 1997
- [80] L.G. McIlrath, et.al. "Design and Analysis of a 512x768 Current-Mediated Active Pixel Array Image Sensors", *IEEE Transactions on Electron Devices*, Vol. 44, No. 10, pp.1706-1, 1997

- [81] D. Scheffer, B. Dierickx, G. Meynants, "Random Addressable 2048 x 2048 Active Pixel Sensor", *Transactions on Electron Devices*, Vol. 44, No 10, pp.1716-20, 1997
- [82] S.Kavadias, et.al. "A Logarithmic Response CMOS Image Sensor with On-Chip Calibration", *IEEE Journal of Solid-state Circuits*, Vol. 35, No.8, pp.1146-52, 2000
- [83] F. Pardo, et.al "Space-Variant Nonorthogonal Structure CMOS Image Sensor Design", *IEEE Journal of Solid-state Circuits*, Vol.33, No.6, pp.842-9, 1998
- [84] J. Couiombe, M. Sawan, C. Wang, "Variable resolution CMOS current mode active pixel sensor", *The 2000 IEEE International Symposium on Circuits and Systems*, 2000. *Proceedings. ISCAS 2000 Geneva*, Volume: 2, pp. 293 -296, 2000
- [85] H. Simmermann, *Integrated Silicon Opto-electronics*, Springer Series in Photonics, 2000
- [86] B. Pain, *CMOS Digital Image Sensors*, Short Course (SC193) at Photonics West, SPIE conference, 2001
- [87] R. Hornsey, *Design and Fabrication of Integrated Image Sensors*, ICR Short Course Notes, 1998
- [88] R.J. Baker, H.W. Li, D.E. Boyce, *CMOS Circuit Design, Layout, and Simulation*, IEEE Press, 1998
- [89] M.J.M. Pelgrom, A.J. Duinmaijer, A.P.G. Welbers, "Matching Properties of MOS Transistors", *IEEE Journal of Solid-State Circuits*, Vol. 24, No. 5, pp.1433-40, 1989
- [90] J. Bastos, M. Steyaert, R. Roovers, P.Kinget, W. Sansen, B. Graindourse, A. Pergoot, Er. Janssens, "Mismatch characterization of small size MOS transistors", *Prof. IEEE 1995 Int. Conference on Microelectronic Test Structures*, Vol. 8, pp.271-6, 1995
- [91] C.L. Keast, C.G. Sodini, "A CCD/CMOS process for integrated image acquisition and early vision signal processing", *Proc. SPIE Charge Coupled Devices and Solid State Spatial Sensors*, Vol. 1242, pp. 152-61, 1990
- [92] C.L. Keast, C.G. Sodini, "A CCD/CMOS based imager with integrated focal plane processing", *IEEE Journal of Solid State Circuits*, Vol. 28, No. 4, pp. 431-7, 1993
- [93] P. Dudeck, P.J. Hicks, "A CMOS General-Purpose Sampled-Data Analog Processing Element", *IEEE Transactions on Circuits and System-II: Analog and Digital Signal Processing*, Vol. 47, No. 5, pp467-73, May 2000
- [94] S. Anderson, W.H. Bruce, P.B. Denyer, D. Renshaw, G. Wang, "A Single Chip Sensor & Image Processor for Fingerprint Verification", *IEEE 1991 Custom Integrated Circuits Conference*, pp.12.1.1-4, 1991