Detection and Analysis of Single Event Upsets in Noisy Digital Imagers with Small to Medium Pixels

by

Rohan Thomas

B.A.Sc., Simon Fraser University, 2017

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Applied Science

in the
School of Engineering Science
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY
Fall 2020

Declaration of Committee

Name:	**Rohan Thomas**
Degree:	**Master of Applied Science**
Thesis title:	**Detection and Analysis of Single Event Upsets in Noisy Digital Imagers with Small to Medium Pixels**
Committee:	**Chair:** **Ash M. Parameswaran,** Professor, Engineering Science
	Glenn Chapman Supervisor Professor, Engineering Science
	Israel Koren Committee Member Professor Emeritus, Electrical and Computer Engineering University of Massachusetts Amherst
	Marinko Sarunic Examiner Professor, Engineering Science

Abstract

Camera sensors are shrinking, resulting in more defects seen through image analysis. Due to cosmic radiation, camera experience both permanent defects known as hot pixels and temporal defective spikes which are Single Event Upsets (SEUs). SEUs manifest themselves as temporal random bright areas in sequential dark-frame images that are taken with long exposure times. In the past, it was difficult to separate SEUs from noise in dark-frame images taken with DSLRs at high sensitivity levels (ISO) and cell phone cameras at modest sensitivity levels. However, recent software improvements in this research have enabled the analysis of defect rates in noisy digital imagers – by leveraging local area and pixel address distribution techniques. In addition, multiple experiments were performed to understand the relationship of SEUs and elevation. This study reports data from imagers with pixels ranging from 7 μm (DSLR cameras) down to 1.2 μm (cell phone cameras).

Keywords: SEUs; CMOS; cosmic radiation; noise reduction; elevation

Dedication

I dedicate this to my late grandfather, Thomas Philip.

Acknowledgements

I am thankful to my entire thesis committee – Dr. Glenn Chapman, Dr. Ash Parameswaran, Dr. Marinko Sarunic, and Dr. Israel Koren – for attending and being active participants during my thesis defence. I owe special thanks to my senior supervisor Dr. Glenn Chapman for his flexibility and allowing me to pursue this research while working full time at NetApp. Your years of industry and academic knowledge have played a key role in the inspiration of this research. On a similar note, I would like to thank Dr. Israel and Dr. Zahava Koren for their assistance in this research.

I would like to thank my research colleagues – Joel Meneses and David Yin – as well as my team at NetApp for supporting me during the entire process. I am grateful to my management team at NetApp for being flexible while I completed my degree.

In closing, I am grateful to my parents, brothers, sisters, and friends for your help and support while I was working on this degree. Most importantly, I would like to thank God – I would not be able to do this without you!

Table of Contents

List of Tables

List of Figures

List of Acronyms

3T	Three Transistors
4T	Four Transistors
APS	Active Pixel Sensor
BC	British Columbia
CFA	Colour Filter Array
CMOS	Complementary Metal Oxide Semiconductor
DSLR	Digital Single-lens Reflex
GPU	Graphics Processing Unit
IC	Integrated Circuit
ISO	International Organization for Standardization
OLED	Organic Light Emitting Diode
OS	Operating System
SEU	Single Event Upset

Chapter 1.

Introduction

1.1. The Emergence of Digital Camera Technology

The camera has been around for almost 200 years [1], with records of the 'pinhole' camera being used for solar observatory in the early 1800s. Many technological advancements were made over the years including the invention of the first film camera. Some of the earliest innovators were Nicéphore Niépce in 1816 who created the first real photograph and Louis-Jacques-Mandé Daguerre in 1837, inventor of the Daguerreotype which is the earliest publicly available photographic process. However, these first cameras were rare and very expensive to manufacture. Further technological advances made the film cameras easier and cheaper to manufacture – making them more common among regular civilians. Photographers of the current generation may not recall the early days when film was the default technology for amateur and professional photography. Users quickly adopted film cameras during the early days due to the relatively low cost and easy operation. Purchasing one was not limited to only the rich or the highly skilled. They were commonly purchased by average citizens as they became readily available and fairly affordable.

American companies such as Kodak were the leaders of the helm in the film era, having a larger market share than the likes of the Japanese manufacturers: Canon, Nikon, and Sony to name a few. In the height of its dominance, Kodak's slogan was "You press the button, we do the rest" [2]. Kodak was known to make the overall experience of ownership for a film camera straightforward and easy. During the film days, photographers were able buy the camera and film from Kodak, capture photos on the film, and finally return the film back to Kodak to be processed. Yet the process was expensive and relatively slow. The development of these sensors, combined with powerful in camera image processors and LCD displays lead to the development in the mid-1990s of the digital camera, which has now seen over 25 years of constant development. From being an expensive commodity in its initial days, digital cameras only became mainstream about

10 years after by about 2000. The photography market that once had Kodak as the leader of the pack, underwent a huge change which now saw electronic oriented companies such as Canon and Nikon emerge as the new leaders of the digital camera market and film companies almost disappearing. Now digital cameras totally dominate the field. A popular beginner DSLR – Canon EOS Rebel T7i – costs around $700 CAD at the time of writing this thesis [3]. The prominent Japanese camera manufacturers such as Nikon, Canon and Sony sell a range of digital cameras models for photographers of all levels. In fact, they are not the only ones, there are other manufacturers from around the world – for example Europe – that offer similar cameras. However, many of the camera manufacturers are indeed from Japan.

The digital camera hardware (sensor array and lens) alone is not what made digital camera technology valuable. The integration of the software in the camera with the imager hardware made the digital camera a powerful alternative to film cameras. At its inception, the digital imager had a small number of megapixels (i.e. 1-2) which resulted in much lower resolution compared to lower cost consumer film camera. Many of the current cameras contain software algorithms that have the ability to reduce unwanted errors in images. For example, this could be reduction in noise, pixel defects, and over exposure to name a few. This use of inbuilt software algorithms required skill on the part of the photographer or the print processors in film cameras. Another difference is that certain problems such as white balance of the image were originally dependent on the particular type of film that was used at the time of image capture. Now with digital cameras, the internal software can automatically correct the white balance. The software automatically adjusts the white balance based an analysis of the scene or at the photographer's preference. Even if the white balance was not adjusted on the camera, photo editing software on computers can be used to adjust the white balance of digital images after the image has been captured. Film cameras recorded the images on film negatives, which made it a complex process requiring skilled technicians to configure the white balance after capturing the image. The image written on the film negative will remain as-is. In contrast, digital cameras save digital images to file in digital RAW format – a digital version of the film negative.

Figure 1.1 illustrates the distribution (film vs. digital) of camera sales from 1990 to 2015. The diagram clearly depicts the immediate impact of the digital camera. The introduction of commercial digital cameras (1998) into society saw a steady decline in film

camera sales until it fully took over in the early 2000s. Even though film sales continued after, they almost seemed negligible in the overall camera sale totals.

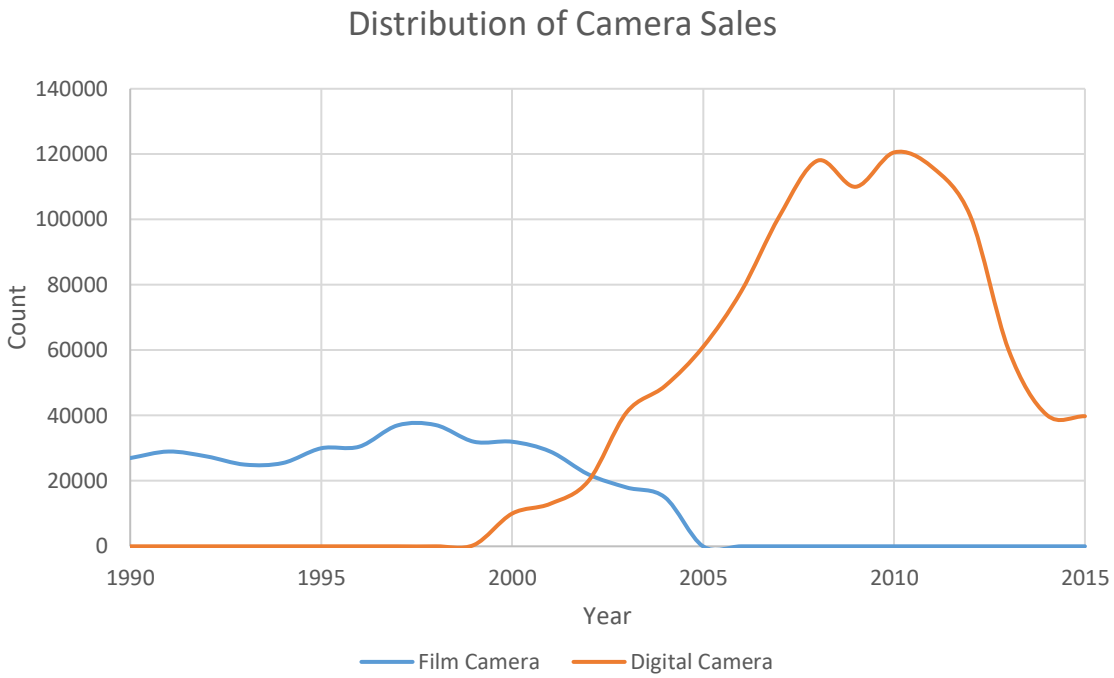


Figure 1.1 History of Camera Sales from 1990 to 2015 (taken from Richter [4])

One observation to make is that DSLRs (Digital Single-Lens Reflex cameras) initially did not make up a significant portion of the total digital camera sales – they were not as mainstream and popular as they are today. Their current popularity is most likely due to the way digital cameras, with their instantaneous image feedback, made it possible to significantly improve the skills of a photographer and thus increase the interest in both amateur and professional photography. However, the decline starting after 2010 is another time period to observe. This coincides with the period of time when iPhones and Android smartphones began their dominance in the cell phone market. The first generation of the iPhone was introduced to consumers in 2007 and the first Android phone the year after. Both of them were well adopted by both consumers and businesses. An interesting fact to note is that all these smartphones came with integrated digital cameras. By replacing regular cell phones, these smartphone cameras would later remove the necessity for a point-and-shoot or DSLR camera. Figure 1.2 illustrates the total number of cell phone shipments (by model) during the first quarter of 2018.

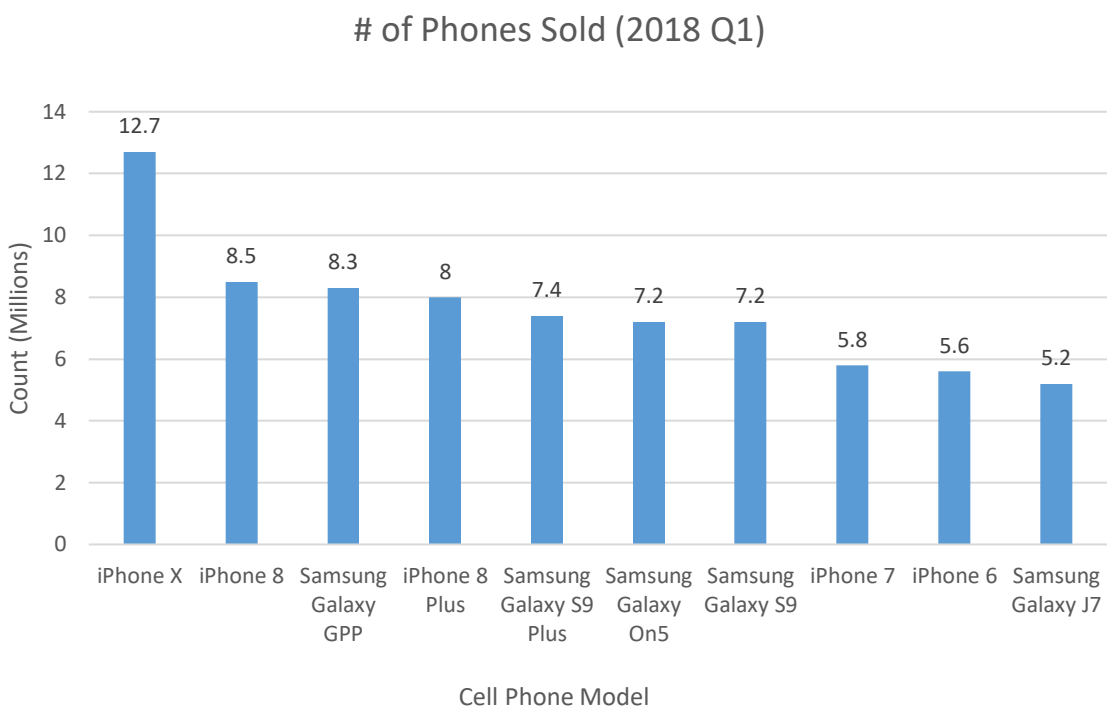


Figure 1.2 Total number of cell phone (models) shipped during Q1 2018 (taken from Richter [5])

For the general user, the cell phone camera – oftentimes fitting in their pockets – meets most of their photography needs. In 2019, Apple revealed the iPhone 11 Pro Max as the first triple-camera system to further improve the quality and features of cell phone camera photography [6]. One may ask how the digital camera now dominates the photography industry. The reason is because of the many advantages it brings to the table. One example is that a digital photograph is a lot simpler and cheaper to the user than processing film for a film camera. As a rough comparison, film images costs over 50 cents per photo while digital camera images costs less than a fraction of a cent per photo. Now digital cameras have become the most popular devices for all types of photography due to their high availability, low cost, and easy compatibility. This is further enhanced by the rise of photo editing software like Adobe photo editing software such as Photoshop and Lightroom. Digital cameras have been embedded into various electronics recently: cell phones, tablets, camera sensors (think of autonomous cars) to name a few.

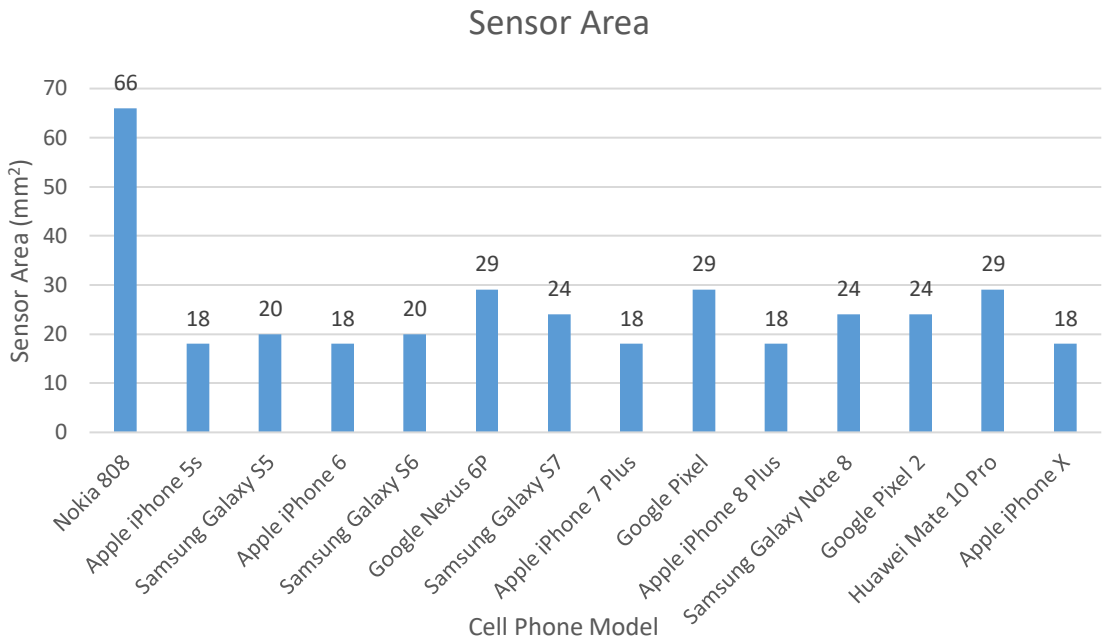


Figure 1.3 Sensor area of different cell phone models (taken from Bhardwaj [7])

The size of the digital camera sensor has been getting smaller over time, especially inside cell phones. This has been a result of a reduction in size of the pixel themselves. Figure 1.3 illustrates the sensor area of different cell phone models. The majority of the cell phone models shown above have sensors areas in the range of 18-29 mm^2.

Camera manufacturers – both DSLR and cell phone – have been working on techniques to make pixels smaller. There has been a trend in the market of increasing the number of megapixels for a given sensor size. This can be achieved by shrinking the overall size of a single pixel and results in fitting more pixels within the entire sensor. Some phone manufacturers such as Nokia have been known of having enormously high number of small pixels in their cell phone cameras. However, Nokia has not been able to conclusively show any performance or quality benefits of having very high megapixel counts. Figures 1.4 and 1.5 illustrate the pixel sizes (in microns) of different cell phone and DSLR models respectively. From comparison we can see the pixel sizes of the cell phone cameras are a lot smaller – in the general range of 1.1 to 1.5 µm compared to larger than 3µm (DSLRs).

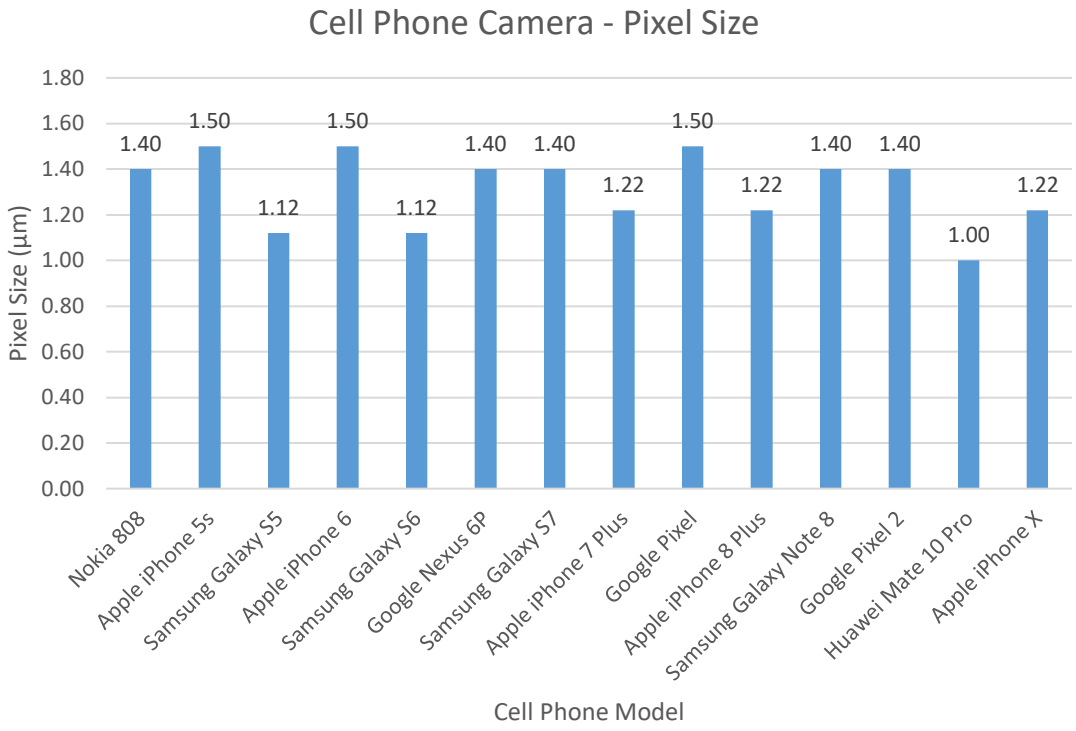


Figure 1.4 Pixel Size of different cell phone camera models (taken from Rehm [8])

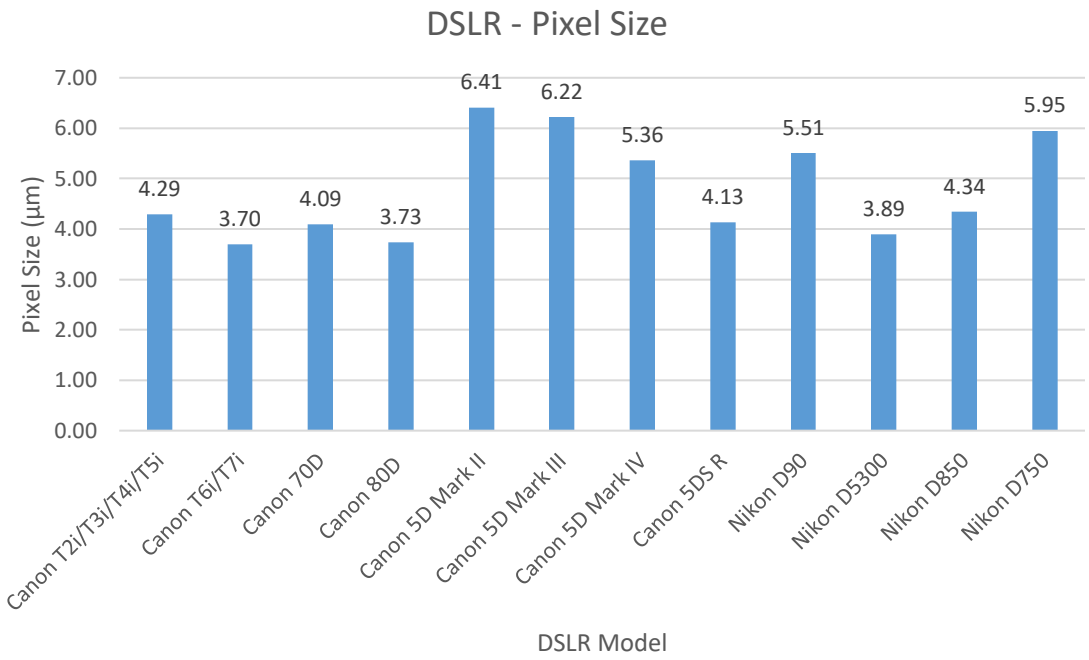


Figure 1.5 Pixel Size of different DSLR models (taken from Rehm [8])

In this thesis we will explore DSLRs with sensors that have pixels in the range of 3-10 µm and cell phones cameras with pixels in the range of 1-2 µm.

1.2. The Defects in Digital Camera Sensors

In every discipline of engineering, reliability and fault tolerance are areas of interest. We need to keep a close eye when discussing reliability in the context of digital imagers. Camera sensors began to experience defects because they are susceptible to cosmic damage. These defects can either be manufacturing time defects that get hidden before being sold or defects that develop over time. Film-based cameras did not have the potential for repeatable defects, as each image was taken on either a different part of the film or a different film roll. This is equivalent to changing the digital camera sensor for every picture. In digital cameras, all images are taken on the same sensor. Hence, defects both accumulate over time and appear instantaneously. The reliability of a camera will directly influence the images it captures. If we were to exclude any software bugs or component defect – for the most part, the life of a digital camera is proportional to the life or reliability of its camera sensor. Most of the time digital cameras are used by photographers for a long time and may be used to capture precise photographs. Visible defects in the digital camera's sensor may affect the overall photo quality.

The topic of defects is one that can be discussed to death as many people have very contrasting opinions. But for the most part – like any other fabricated or manufactured product – digital camera sensors do not come absent of defects. If this was not the case, they would have perfect yield. A single defect may not be very important, however multiple defects can degrade the image. In artificial intelligence applications such as iris recognition, multiple defects can cause a real problem. In our field of study there are two categories of defects: one being manufacturing defects and the other being post manufacturing or induced defects. The defects that are seen in digital cameras from manufacturing are referred to as manufacturing defects. The semiconductor wafers have evident defects that will affect the camera sensors. One thing to note is that these defects themselves appear to cluster together. On the other hand, post manufactured defects

occur during operation or 'infield.' They are caused by purely random external influences that damage or defect the camera sensor. Prior studies in [9, 10] have depicted that the most common source for defects are the random external sources as opposed to manufactured degradation. To give us better intuition of image defects, Figure 1.6 depicts an image of a pen containing defects.

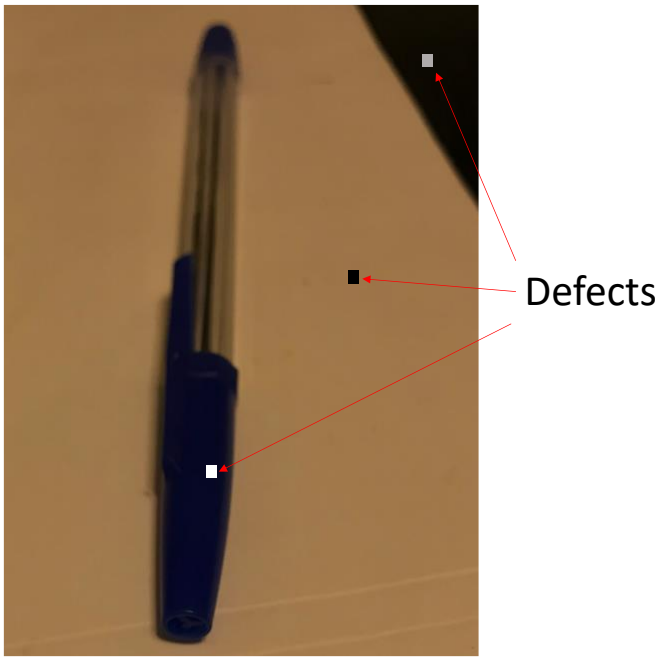


Figure 1.6 Cell Phone image containing defects

1.2.1. Cosmic Rays

In the past, researchers had debated the cause of defects in Integrated Circuits (ICs). However, the general consensus has been the damage caused by cosmic ray particles is the main source [9, 10, 11, 12]. The research surveyed in Ziegler [11] characterizes cosmic rays and described them as unknown source energetic particles from outer space that cause interference and damage in electronics. More specifically they subdivide into four main categories of particles– namely: primary, solar, secondary, and terrestrial. To give some detail, primary cosmic rays are composed of nuclei, protons and neutrons. They come from within our galaxy and enter our solar system and earth. A

subset of those are solar cosmic rays which are generated by nuclear reactions in the sun. Secondary cosmic particles (or cascade particles) are produced in the atmosphere when a primary cosmic ray collides with the atmosphere creating a cluster of lower energy particles such as muons – sometimes protons and neutrons. Lastly, terrestrial cosmic rays are the ones that eventually make it to earth. While these are less than 1% of the primary particles, however, are able to cause defects in ICs. The sun's cycle affects the solar magnetic field which in turn affects the amount of galactic cosmic rays reaching the earth. Ziegler [11] also mentions that at sea level, the cosmic rays are primarily composed of neutrons, protons, pions, muons, electrons, and photons.

Since solar cosmic rays originate from the sun, they cause terrestrial cosmic rays to vary in intensity over time due to variations in solar activity. Figure 1.7 (taken from Ziegler [11]) illustrates the terrestrial cosmic ray intensity in Colorado over 40 years (1950-1990). The cosmic ray intensity is expressed in terms of the relative intensity (percent change) from the 1954 flux.

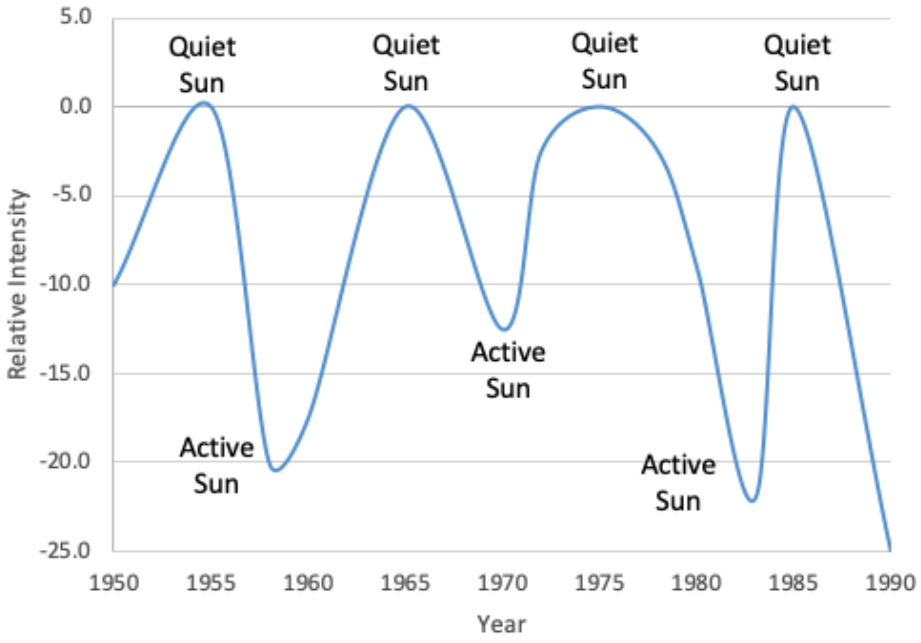


Figure 1.7 Terrestrial cosmic ray intensity (Number of particles) – Colorado [1950-1990] (taken from Ziegler [11])

It can be seen that the intensity rises to a relative maximum (Quiet Sun) or falls to a relative minimum (Active Sun). The intensities during the active sun periods are lower than during the quiet sun periods because the sun's magnetic fields are stronger. There are a lot of particles that make it to sea level during Active Sun periods compared to Quiet Sun periods when fewer particles make it. Hence, the particular year within the sun's 22-year cycle is very important. A full cycle consists of an 11-year low intensity sub cycle and an 11-year high intensity sub cycle. The writing of this thesis took place during a Quiet Sun period. The intensity changes in the graph are from the sun's spots and magnetic field. As shown in Ziegler [11] and Niedzwiecki [13], the topic of neutrons and cosmic rays have been heavily studied. These concepts were mentioned in this thesis as an important background before further exploring defect concepts. Temporal defects – which will be discussed shortly – are affected differently in different parts of the sun's cycle. This will be a thought to keep in mind when discussing defects throughout the thesis.

An important factor that will be discussed later in this thesis is the effect elevation has on cosmic ray particles. Increasing air density reduces the particle flux, which means there is less air to reduce the neutron flux at higher elevations. As a result, the number of cosmic ray particles increase with elevation. Equations (1.1) and (1.2) (taken from Ziegler [11]) shows the relationship between the neutron flux and altitude (or atmospheric pressure).

$$I_2 = I_1 \exp\left(\frac{A_1 - A_2}{L}\right) \tag{1.1}$$

Where I_1 is the flux at altitude A_1, I_2 is the flux at altitude A_2, and L is the absorption length.

$$A = 1033 - (0.03648H) + (4.26 \times 10^{-7}H^2) \tag{1.2}$$

Where H is the height in feet.

Understanding the relationship in (1.1) and (1.2) will be extremely important when studying and analyzing temporal defects at various elevations in digital cameras later in the thesis. As mentioned earlier, ICs are hit by cosmic rays and these generate in-field defects in the devices. The defects can be categorized as permanent faults that accumulate over time or short-lived transient upsets that are random in location and time.

Often times these defects are difficult to record or observe in regular ICs because they develop deep within the circuit. They are only measurable by their output signals and sometimes create faults that can render the entire circuit useless. What will be shown later in the thesis is that, digital camera sensors have the ability to record these defects caused by cosmic rays. Dark-frame images (images where no light enters the camera sensor) are recordings that represent the camera's sensor. These images allow us to identify pixel defects (both permanent and temporal) in the camera sensor. As a result, digital imagers can be used as cosmic ray detectors Niedzwiecki [13]. With this in mind, there are two types of digital camera defects that we can categorize.

When it comes to defects in digital cameras, the user may not notice the defects at first, but they become apparent over time. The first type of defect that we can identify are hard permanent defects that are induced from cosmic shortly after the fabrication process [9, 14, 15]. The second type of defect that we can also identify are soft temporal defects that appear randomly in a given image and disappear in the following image. Even though there are two different types of defects, both happen very granular at a pixel level. We detect the defects by capturing an image to represent the input light intensity. A defective pixel will cause the output image to show that pixel with noticeable errors (i.e. bright spots). Energy deposited via cosmic radiation in an integrated circuit is going to be a function of the energy of the particle. The impact of cosmic rays is different in soft and hard defects. In soft defects, the low energy cosmic rays create temporal charges while high energy cosmic rays damage the circuit crystal and create a permanent defect.

1.2.2. Permanent Defects – Hot Pixels

The first category of defects are permanent defects which are repeatable. If the cosmic ray charge has a high enough density, it will damage the substrate of the IC and create a permanent defect. These persist at the same pixel location throughout multiple dark-frame images – even with multiple tests at different points of time or with different settings. Figure 1.8 identifies some permanent defects – a hot pixel and saturated pixels in an iris image taken from Bergmüller [16].

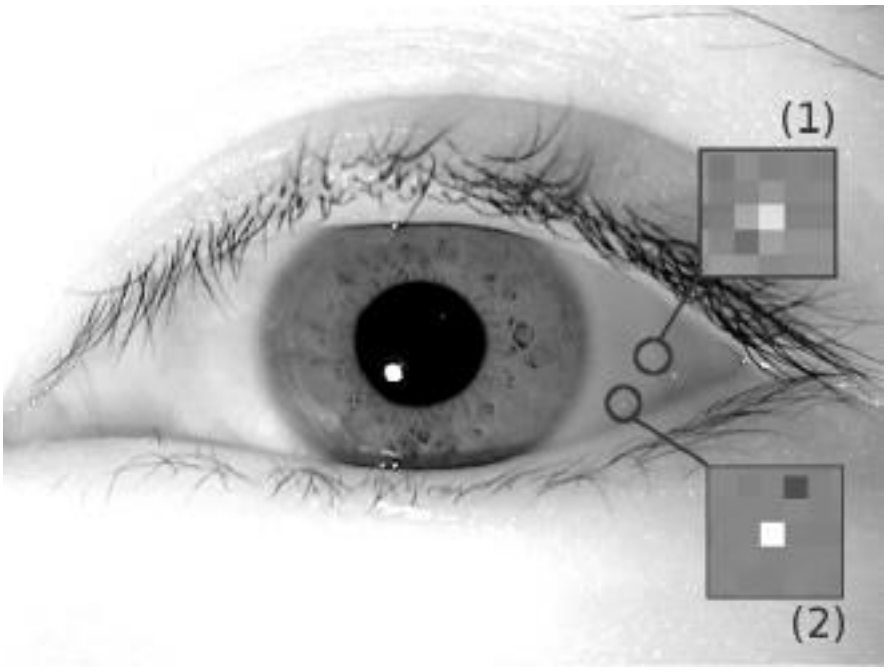


Figure 1.8 Hot pixel (1) and stuck pixels (2) in an iris image (taken from Bergmüller [16])

Stuck pixels (as seen in Figure 1.8) are mainly due to manufacturing and can be categorized as permanent stuck high (white) or stuck low (dark) defects. Oftentimes stuck high defects are just saturated pixels that will become unsaturated with shorter exposure times. However, in our research we have found that stuck pixels do not really appear in images. This is because camera manufacturers almost always mask them in the camera sensor before making the camera available to consumers.

The more important in-field permanent defect is the Hot Pixel (as seen in Figure 1.8). Research in Leung [17] has shown that these manifest in camera sensors (as with other ICs) shortly after fabrication and build up over time due to cosmic rays. Unlike regular ICs which become unusable when permanent defects appear, digital imagers have the advantage of only experiencing circuit and image degradation when hot pixels appear. Hot pixels have been known to accumulate as cameras age and at a rate that is highly dependent on pixel size. You can identify hot pixels in images as bright dots – with their brightness increasing with longer exposure times and/or higher sensitivity (ISO). Many studies have pointed out that hot pixels are created at a higher rate when cameras are primarily at higher altitudes – due to the presence of more cosmic radiation. The values of

Hot Pixels can vary as the intensity of the incident ray changes. Figure 1.9 (taken from Chapman [18]) plots the Hot pixel output vs. exposure time for three types of hot pixels – offset hot pixel, a standard hot pixel, and a regular pixel. The stuck pixels that were mentioned earlier have a very high slope and will eventually reach saturation.

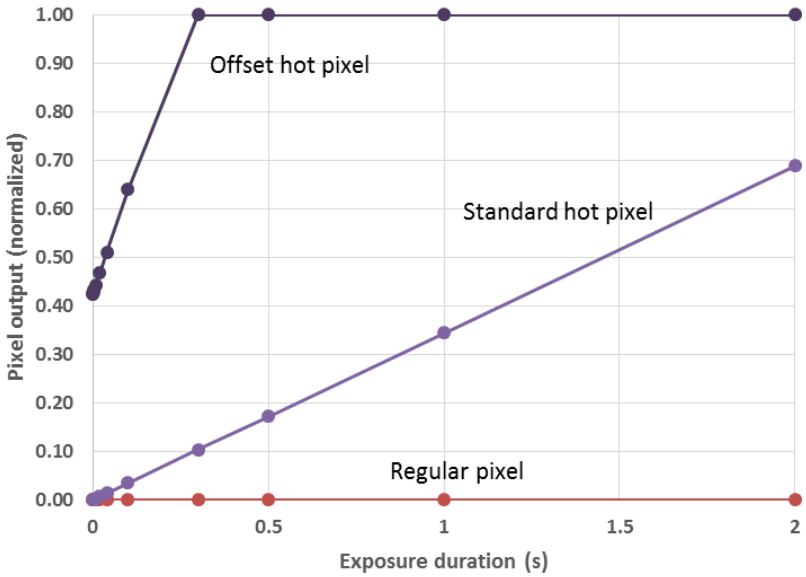


Figure 1.9 Hot pixel output vs. Exposure time – a good regular pixel, a standard hot pixel, and an offset hot pixel (taken from Chapman [18])

Figure 1.9 does a linear fit for the pixel dark-frame response versus the exposure time. For standard hot pixels, there is no offset value and they will have an impact on images with long exposure times – say larger than 1 second. In contrast, partially stuck hot pixels with a large offset will persist as a bright spot in all images regardless of the exposure time. The amplification of the pixel signal by the gain (ISO) setting also amplifies the values of both the hot pixel dark current and the offset.

Identifying and correcting hot pixels presented itself as a challenging task but past research has shown methods to correctly tackle this problem. We will not go into too much detail regarding Hot Pixels since research over 14 years has adequately discussed this topic. For the most part, the high-level concept and experimental conditions will be discussed to acknowledge their behaviour in digital cameras.

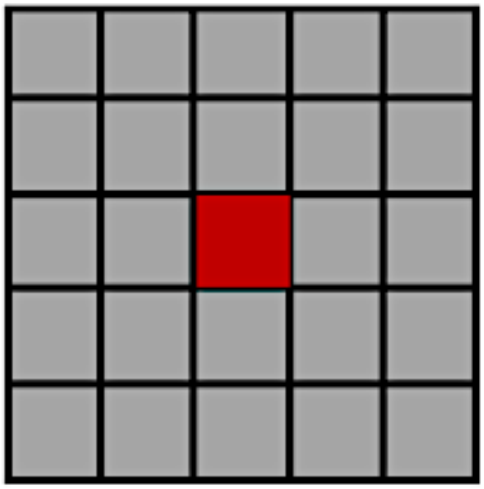


Figure 1.10 5x5 pixel example containing a Hot pixel

Figure 1.10 portrays an example of a 5x5 pixel square containing a hot pixel in the center. In a sensor containing no defects, the 5x5 pixel square should not have any bright pixels. The presence of this bright pixel clearly illustrates the hot pixel defect. The concepts of exposure time and ISO (light sensitivity) will be explained in the next chapter. Increasing both of these settings will cause the hot pixel intensity to increase as well. Figures 1.11 and 1.12 diagrammatically show the influence that exposure time and ISO have on hot pixel intensity respectively. It can be observed that hot pixels become brighter by increasing either the exposure time or ISO level. When increasing the ISO level, the background dark pixels get brighter as well.

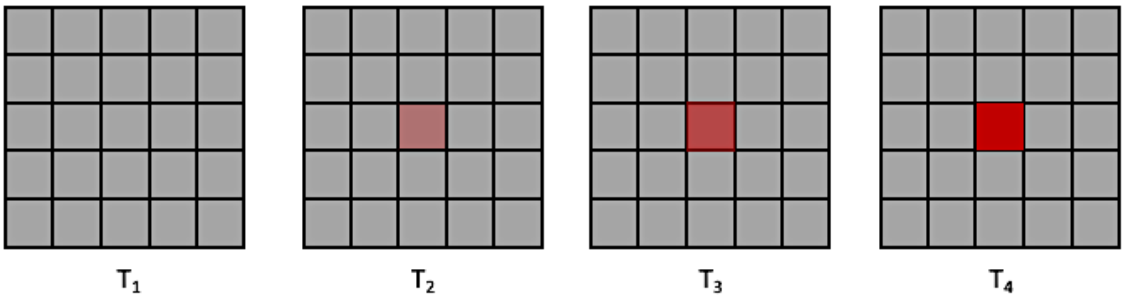


Figure 1.11 The intensity of a Hot pixel increasing with Exposure time ($T_1 < T_2 < T_3 < T_4$)

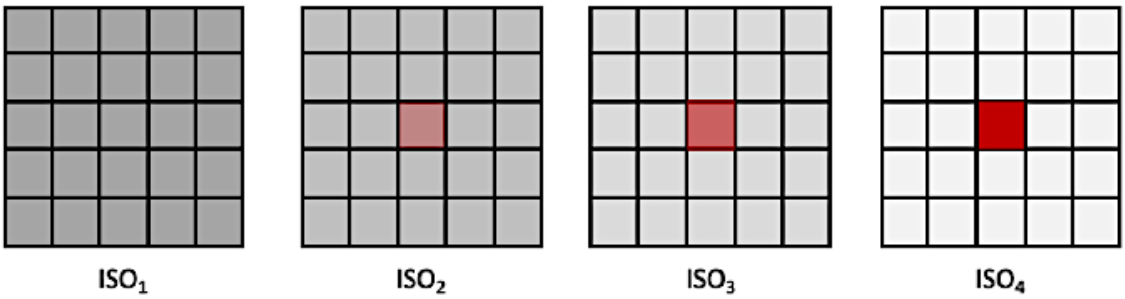


Figure 1.12 The intensity of a Hot pixel increasing with ISO Level ($ISO_1 < ISO_2 < ISO_3 < ISO_4$)

In hot pixel research the experimental process involves capturing dark-frame images over a broad range of ISO levels with increasing exposure times for each ISO setting. The data allows us to measure (for a given ISO setting) the increase in hot pixel intensity in relation to increasing the exposure time. This data is used by a software detection algorithm written in MATLAB that analyzes hot pixels. Noisy pixels become an issue when increasing exposure time and ISO as they can be falsely identified as hot pixels. Hence, the hot pixel detection algorithms are optimized to deal with high levels of noise.

As mentioned earlier, the research in this thesis will not go into the detection algorithms for hot pixels. Our research done in [19, 20] explored hot pixels in detail but the focus of this thesis will be on temporal defects. The intent of this section was to give an overview regarding the creation of permanent hot pixels due to cosmic radiation, the influence that exposure time and ISO have on them and lastly help us differentiate when discussing temporal defects.

1.2.3. Temporal Defects – Single Event Upsets

Until now we have only discussed hard permanent defects – specifically hot pixels. Much like regular ICs, digital imagers are susceptible to transient in-field defects that randomly appear and quickly disappear. The literature [9, 15, 21] suggests that all these events appear to be caused by cosmic ray particles striking the sensor at random times

and locations. The collision of cosmic ray particles with the sensor happens at a low angle and energy – hence, it just deposits charge instead of damaging the crystal. These soft defects in regular ICs are often referred to as Single Event Upsets (SEUs) and are fairly different than permanent defects. SEUs are the transient, short-lived, cosmic ray induced events in integrated circuits Mukherjee [22]. These can produce errors in the computation of the IC by flipping bits in memory or changing results in sub sections (inverters or flop-flops) with processing units. As transistors get smaller, the amount of energy needed to flip a bit is smaller and the sensitivity to particles rise. SEUs in ICs are of considerable interest as induced errors may potentially impact the chip functionality. However, a lot of the prior research has artificially simulated SEUs through direct nuclear radiation or lasers [22, 23, 24].

Other literature has focused on various other types of integrated circuits such as Field Programmable Gate Arrays (FPGAs) and Standard Random Access Memory (SRAM) [25, 26]. While 'black matter' or SEUs appear in regular ICs, studying them is very difficult as the event occurs deep within the IC but its effect is only detected as changes in the output of the chip. There are three main bodies that are interested in the behaviour of SEUs – the space community, the aircraft industry and the military. Research in Brogna [27] shows protecting the operation of space vehicles from CPU SEUs. As primary cosmic radiation is greater in space, so will the impact of SEUs be greater on satellites and other electronics in space. Similarly, the aircraft industry is concerned about SEUs as [28, 29, 30] show cosmic radiation is also greater at higher elevations on earth. Lastly, the military is researching to prevent SEUs for obvious reasons Banu [31]. Despite the research performed to understand the count of SEUs in traditional ICs, very few have been able to see inside SEUs. They have focused on muons and streaks of SEUs (explained more in Chapter 3) which occur less frequent and are not relevant to the SEU community.

Related to this research, [21, 32] have shown that digital camera sensors have the ability to retain the charge deposited via dark frame images, allowing for the detection of cosmic ray damage. Since the values are read out in a series of dark-frame images, they can be used as radiation event detectors and directly measure information about the natural background radiation (or cosmic radiation) impact on ICs. By this, digital imagers give us an understanding on how cosmic radiation is impacting sensors as well as the ability to look inside SEUs. In general, digital cameras have a range of pixel sizes – high end DSLRs have large pixels (4-7 µm) and cell phone cameras having micron size pixels.

As they integrate charge, digital imagers allow us to record the rate of SEUs, their location, charge deposited, and charge spread. When it comes to digital camera sensors, an SEU happens at a random pixel location in a single image. They appear as transient bright pixels that appear at a given pixel location in a single image of a sequence of images. The SEU will not appear at that same pixel location in other images – causing no lasting damage [10,17]. Figure 1.12 gives an illustration of an SEU randomly appearing and disappearing in time.

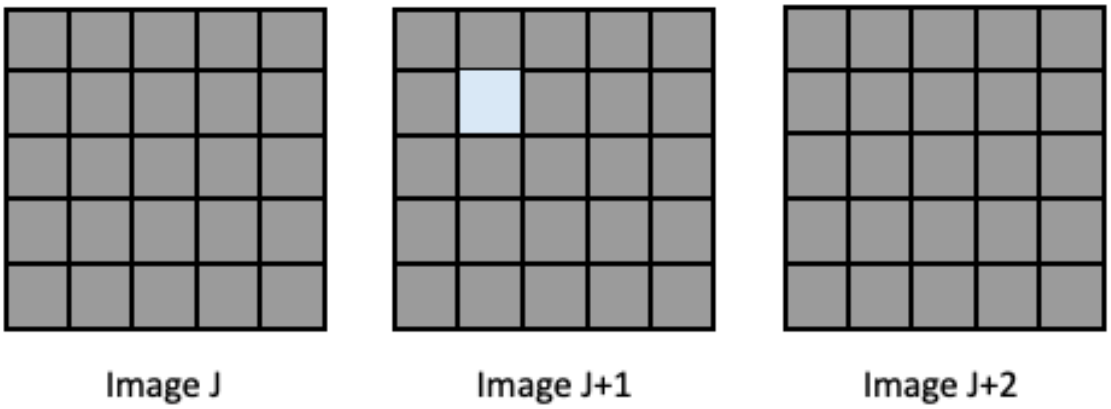


Figure 1.13 An SEU randomly appearing and disappearing in a series of 3 images

Research in Thomas [9] has shown that an SEU in a digital camera is over 100 times more likely to be observed than a permanent hot pixel as we have the ability to see weaker events. The same research introduced the basic experimental setup used to capture SEUs in high end DSLRs at modest ISO levels. However, this setup failed when detecting SEUs in both high end DSLRs at high ISOs and cell phone cameras at modest ISO levels. It did not have the ability to analyze high powered cosmic ray occurrence rates. This was due to the presence of high noise which made it difficult to identify SEUs. Therefore, analyzing the nature of both DSLR and cell phone cameras with high noise levels is very important and will be discussed in this thesis. Our objective is to turn studies of SEUs in digital imagers into a tool to characterize the injected charge distribution and charge area resulting from typical cosmic ray influxes. This will apply to imagers with less noisy large pixels and ones with more noisy micron sized pixels. This thesis will dive deep into SEUs in noisy digital cameras and extend the research that has already been done on this topic. Lastly, earlier when we looked at cosmic rays, we mentioned their

relationship with elevation. However very little research has been done to show the noticeable relationship of SEU rate starting from sea level to higher elevations. This topic will be covered in Chapter 5.

1.3. Summary

The rise in popularity of digital photography has been quite apparent in the recent years. This has been due to the increased sales in DSLRs and cell phones. Along with this, digital camera manufacturers have been shrinking the size of their camera sensor pixels – this has led to certain complications such as higher amounts of noise. As more cameras and cell phones are being sold, it is paramount to look inside the transient defects that appear during operation such as single event upsets. Past research has explored hot pixels and single event upsets in certain regular conditions – DSLRs at low ISO levels. However, very seldom has work been done on cell phones at low ISO levels and DSLRs at high ISO levels – both of which generate high amounts of noise. Hence, this thesis will focus on identifying weaker SEUs in the midst of noise for both DSLRs and cell phone cameras.

After this introductory chapter, the rest of this thesis is structured in the following manner: Chapter 2 will look at the technology used in digital cameras – their sensors in particular. The underlying technology has changed substantially over the years; hence, this chapter will give you better intuition. Continuing, the chapter will go into the fundamentals of digital cameras and digital photography pertaining to this research. In Chapter 3, single event upsets will be more thoroughly discussed as well as the experimental method used to take SEU dark-frame images in both DSLRs and cell phones. Chapter 4 extents this by walking through the different SEU detection algorithms and noise reduction techniques. At the end of the chapter we will see experimental results from three detection algorithms. This leads into Chapter 5, where we will look at analyzing camera sensor noise, SEU charge distribution, elevation, and the SEU charge ball size. Chapter 6 will wrap up the thesis with concluding thoughts and suggestions for future SEU research.

Chapter 2.

Digital Camera Sensors and Photography

2.1 Overview

The digital camera has many components that work together to transform light into the final high-quality digital image for the user. However, the most important role in the entire process is that of the digital camera's image sensor. Many features of the digital camera are dependent on the image sensor. For example, the ISO (sensitivity), the resolution of the images, dynamic range capabilities, and even the compatibility a camera has with lenses are all influenced by the image sensor. That is why it is paramount to do a review of camera image sensors when exploring the behaviour of temporal defects (single event upsets) in both cell phone cameras and DSLRs. Early digital cameras used Charge-Coupled Device (CCD) image sensors and present-day digital cameras use Complementary Metal-Oxide Semiconductor (CMOS) image sensors. The intent of this chapter is to summarize the present-day technologies used in image sensors and the theory behind photo detection. When it comes to modern-day photography, many options like image gain (or ISO) and light exposure time play an important role – they will be examined towards the end of the chapter. This imager understanding is necessary to understand how the imaging sensor allows us to detect cosmic rays impacts on devices.

2.2 Photodetector Theory

Many may wonder what the underlying process that a silicon sensor undergoes to actually take a photograph. Incident light is just one type of electromagnetic radiation that is received by a photodetector. At its plainest state, a photodetector (or photo sensor) takes energy in the form of incident light and turns it into an electronic signal. This signal will then undergo signal processing via a digital microprocessor and transition to another component in the camera. If it were not for the image processor, present day imager arrays

would not be able to produce high quality images. Since a photodetector is a semiconductor, it inherently has all of its fundamental attributes – that is its electrons can reside in different energy bands. At the lowest possible temperature, 0°K – also known as absolute zero – electrons will reside at the lowest possible band which is known as the valence band. In the conduction band, electrons typically flow to generate current. However, at the temperature of absolute zero, there are no electrons seen in the conduction band. Since both of the bands – the valence band on the bottom and the conduction band at the top – are separated, they form an energy level called the energy band gap (E_g). More specifically, the energy band gap (measured in electron volts) extends from the top of the valence band to the bottom of the conduction band. The energy band gap is a forbidden energy level where no electrons can exist. This is also equal to the electron-hole pair energy.

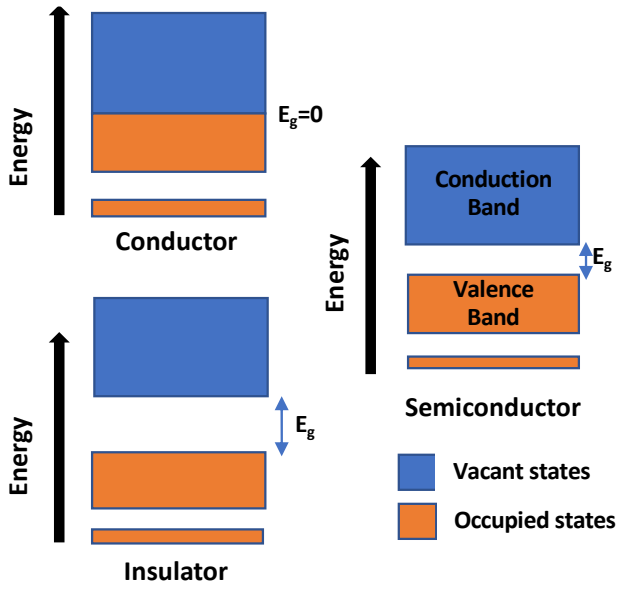


Figure 2.1 Energy bands and gaps of conductors (E_g=0), insulators (large E_g), and semiconductors (small E_g) (taken from Lecher [33])

Figure 2.1 above – taken from Lecher [33] – depicts the energy bands (the conduction and valence bands) as well as the energy gap between them. Pay attention to the difference in E_g for a conductor versus an insulator. A conductor will have very little or even no energy band gap ($E_g = 0$). When it comes to an insulator, there is a certain energy band gap level which was defined earlier as E_g.

The movement of photons through different energy states will cause a hole to be left in the bottom valence band as the photon excites an electron away from the valence band to the higher conduction band. This can be calculated with Equation 2.1 below:

$$E = \frac{hc}{\lambda} \tag{2.1}$$

Analyzing Equation 2.1 makes use of Planck's constant h and the speed of light c. Equation (2.1) uses λ which is the wavelength of the photon. Alternatively, Equation (2.2) uses v which is the frequency of the photon. Hence, Equation (2.1) can be rewritten as Equation (2.2):

$$E = hv \tag{2.2}$$

In order for a semiconductor to detect photon energy, the electron-hole pair energy in the semiconductor has to be less than the photon energy *E*. In the case of silicon this is 1.14 eV. As a result, the electron now has the ability to jump from the lower valence band to the higher conduction band. In contrast, when the energy of a photon is less than the electron-hole pair energy, the semiconductor itself will just absorb its energy. The photon will not have sufficient energy to jump from the valence band to the conduction band. It can be noted that there is a threshold between insufficient bandgap energy and sufficient bandgap energy. For the most part, in the study of semiconductors this level is most commonly referred to as the cut-off frequency. This is the frequency that photons need to meet in order to be excited from the valence band to the conductive band, causing electron-hole pairs. Photons with energies that are greater than the cut-off frequency will be excited from the valence band to the conduction band, while photons with lower energies will not.

Every type of semiconductor material has its own unique cut-off frequency. For example, diamond has an energy bandgap of 5.5 eV – electron volts – while Silicon and Germanium have energy bandgaps of 1.14 eV and 0.67 eV respectively. As you can see, the energy band gap – and in turn the cut-off frequency – has a very paramount role when it comes to a semiconductor's overall ability to conduct energy. Referring back to the insulators and conductors in Figure 2.1, when a semiconductor has a smaller or no energy bandgap, it is a conductor. When a semiconductor has a larger energy bandgap, it is an

insulator. Additionally, temperature can also have a major impact. When the heat of a semiconductor rises, the exerted heat energy will cause more photons to be excited from the valence band to the conduction band. This in turn generates a charge from the moving photons. Figure 2.2 illustrates this excitation of an electron from the valence band to the conduction band.

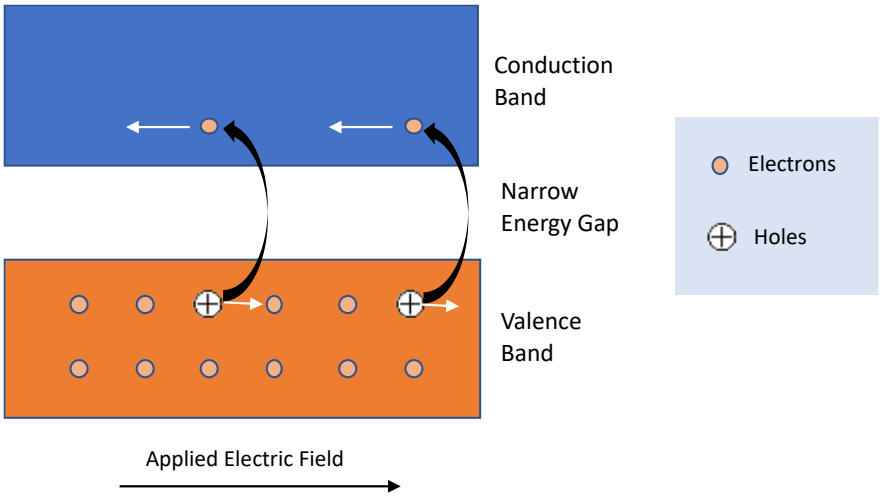


Figure 2.2 Electron excitation from valence to conduction bands (taken from Dux [34])

Some of the light passing through the semiconductor will be lost as it will interact with both the photons and the electrons in the semiconductor – meaning, not all of the light will be preserved. The strength of the photons will decay with an exponential relationship. Formulaically, we define the initial light intensity as I_O which gets absorbed by the Beer-Lambert exponential decay of Equation 2.3.

$$I(x) = I_O * e^{-\alpha z} \tag{2.3}$$

This equation is referred to as the Beer-Lambert Law. This law essentially shows the overall absorption of a material. In Equation 2.3, α is the absorption coefficient of the material (in cm^{-1}) and z is the depth into the surface of the semiconductor. The absorption of photons depends on the type of semiconductor material being used and the initial light energy (I_0) from Equation 2.3. Different semiconductor materials will have different

absorption coefficients and I_0 values. In this research we are dealing with electronic devices, therefore silicon is a suitable semiconductor material to analyze. Recall that Silicon has an energy band gap of 1.14 eV. To properly understand the absorption of light in silicon crystal, we should get a better understanding of the absorption coefficient of silicon at different wavelengths. Figure 2.3 depicts the absorption coefficient of silicon for different wavelengths. Since visible light falls in the wavelength range of 400 to 700 nm, the red, green, and blue components of light in silicon will fall under that category.

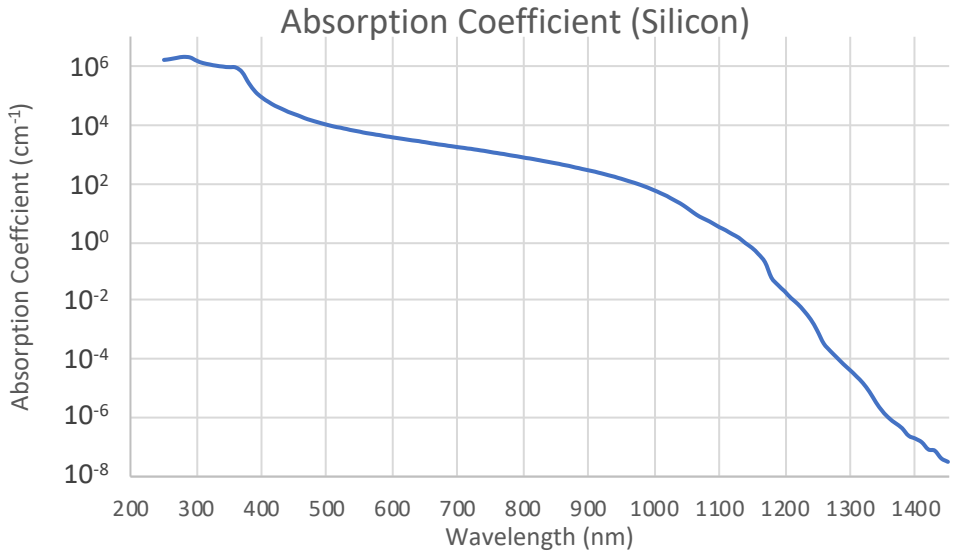


Figure 2.3 Absorption coefficient versus wavelength for Silicon (taken from Green [35])

When discussing photodetectors, attention should be placed on the photo carriers that are created when an electron is excited from the valence band to the conduction band. Even though they can be used to depict light intensity as well as electron-hole pairs in the process, the photo carrier itself is very short lived and does not give much time to be measured or analyzed. To measure these better, other electronic devices are often leveraged such as photodiodes. These will be looked at in further detail.

2.3 Photodiode Theory

In modern imaging sensors the base pixel element detecting the photons is the classic photodiode. Consider the PN junction diode (Figure 2.4).

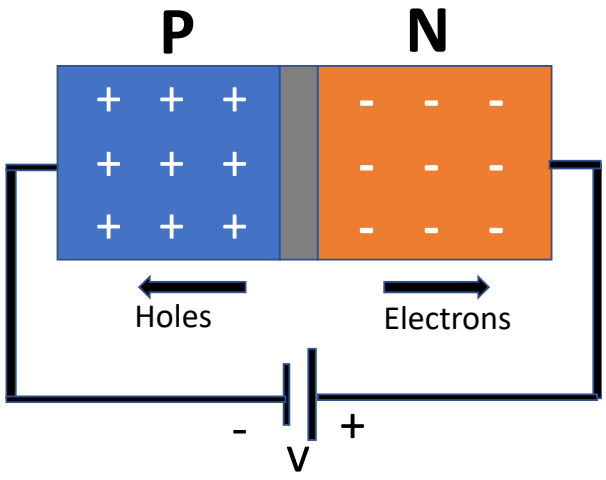


Figure 2.4 PN Junction Diode in reverse bias (taken from Nasir [36])

There are different types of biasing in which a diode can be configured – zero biasing, forward biasing, and reverse biasing. The PN junction diode in Figure 2.4 is in reverse bias. In general, a PN junction diode is another type of electronic device and is most commonly made with some sort of semiconductor material such as silicon, germanium, and gallium arsenide. However, often times this semiconductor is silicon as it can withstand higher temperatures than other materials. Figure 2.5 shows the general I/V characteristic curve of a P-N junction diode.

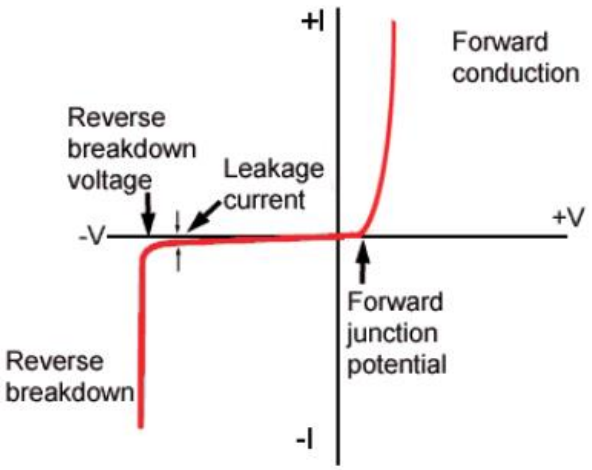


Figure 2.5 I/V characteristic curve of a P-N junction diode (taken from [37])

In a PN junction there is a region of negative charge and a region of positive charge. The region of positive charge contains holes and the region of negative charge contains electrons. To create the regions, impurity material is added to the base of the semiconductor material. The negatively charged region is referred to as the N-type and the positively charged region is referred to as the P-type. Another way to think of the PN junction diode is if a block of P doped semiconductor were to be attached to a block of N doped semiconductor. Since the negatively charged region is touching the positively charged region, short flows happen between the regions. Electrons flows from the negatively charged N-type region to the positively charged P-type region. Similarly, holes flow from the positively charged P-type region to the negatively charged N-type region. The existence of both flows leads to diffusion and causes something known as the junction effect. This essentially is the creation of another type of region between the P and N regions called the 'depletion region.' Unlike the P-type and N-type regions that have positive charge and negative charge respectively, the depletion region is an area where no charge carriers exist. However, the diffusion in the depletion region causes an electric field. This results in a net zero charge by forcing the negative and positive charges to diverge from joining each other. In the event where the potential in the N-type region is lower than the potential in the P-type region, the electrons will flow toward the P-type region from the N-type region. Electrons cannot flow from the P-type region back to the N-type region as the depletion region prohibits that flow.

Returning to the topic of biasing, there are 3 types of biasing shown in Figure 2.6: forward biasing, zero biasing, and reverse biasing.

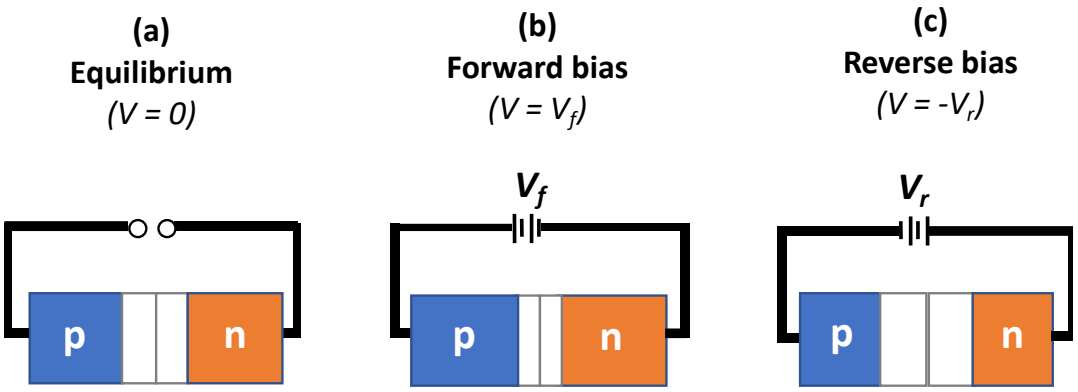
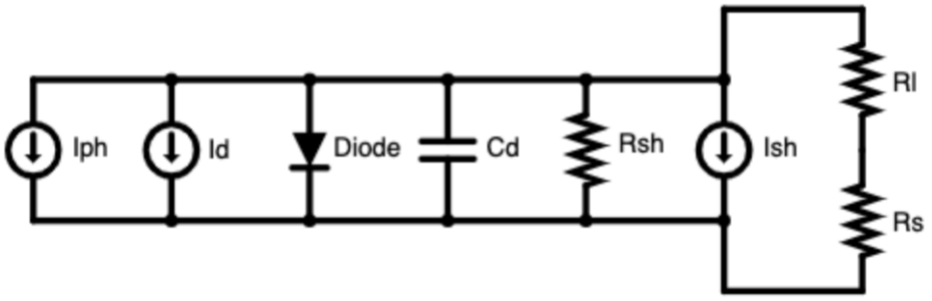


Figure 2.6 PN junction diode in (a) Zero Bias, (b) Forward Bias, (c) Reverse Bias (taken from Anderson [38])

When a PN junction diode is forward biased, the potential within the depletion region lowers, hence shrinking the depletion region area. As a result, the carriers begin to move through the junction in a 'forward' direction of current. When a PN junction diode is reverse biased, the holes and the electrons are drawn away from the PN junction. This causes the width of the depletion region to get bigger. Recall the reverse breakdown voltage from Figure 2.5. This essentially is the largest reverse bias voltage that the diode can handle.

Since the PN junction diode has been discussed, now it's time to understand the photodiode as it is a type of PN junction diode. In terms of the biasing mode, the photodiode is in reverse bias. It takes an input of light and changes it into electronic energy. The input of light will be in the form of a photon and creates the excitation to an electron which was discussed earlier. The electron will move from the valence band to the conduction band, leaving a hole in the valence band. This electron and hole combination are the photo carrier.



Legend

Iph: Photo Current
Id: Dark Current
Cd: Capacitance of Diode
Rsh: Shunt Resistance
Ish: Shunt Current
Rl: Load Resistance
Rs: Series Resistance

Figure 2.7 Circuit model of a Photodiode (taken from [39])

Figure 2.7 gives a circuit representation of the photodiode. This is not the exact making of the inside of a photodiode but gives a representative illustration of the function inside the photodiode – namely capacitance, resistance, and current characteristics. When there is no light coming into the photodiode, there is a small current value present due to the reverse bias voltage. This voltage then eventually results in a leakage current called the dark current as there is no light coming into the photodiode. There are factors that can influence the dark current – one being the width of the depletion region. Hence, a dark current can be evident when the depletion region is wide. When there is thermal energy present, then there is a photoelectric current.

The photodiode has both benefits and some shortcomings. Looking first at the advantages, photodiodes can be grouped together to form a bigger photo sensor. They also have the ability to process images fairly quickly compared to some CMOS sensors. But the shortcoming of the photodiode will ultimately come down to the dark current that was just mentioned. It's been found that a substantial quantity of dark current is created in photodiode sensors. To complement the current analysis of photodetectors, further analysis will be done on present-day technology. The next section will discuss the present CMOS implementation in modern digital cameras and relate to the current SEU defect research.

2.4 CMOS Sensors

As mentioned earlier, the most popular technology used in modern digital imagers is CMOS sensors. Since their introduction into the market, many of the camera manufacturers started to quickly adopt the technology. This was a change for an industry that was primarily accustomed to the dominance of the CCD camera technology for many decades. Since we use modern digital cameras in this research, this implies that their underlying sensors are based on CMOS technology. Hence, in the remaining sections of this thesis, only CMOS sensors will be discussed.

The CMOS sensor that is used in modern day digital cameras is referred to as the Active Pixel Sensor (APS). Hence, you may hear both of the terms used interchangeably in this thesis. In the early days of the CMOS sensor, it was referred to as the passive pixel sensor because the output current at each pixel row got amplified. When the APS was invented, it got its name as the output pixel current got amplified rather than the pixel row getting amplified. The combination of many APS circuits in the camera form to create a CMOS camera sensor. Since the APS uses complimentary MOS transistors (CMOS) it can easily integrate with other peripheral circuits in the camera's sensor.

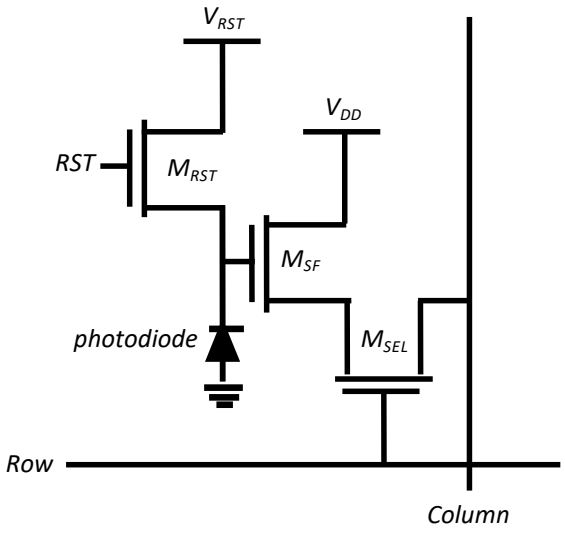
The major understanding to obtain before moving forward is the difference between CCDs and CMOS sensors. With CCD technology, it is important to note that sensitivity of the sensor was dependent on the overall size or area of the CCD itself. During the capturing of an image, the collected electrons are stored in the area of the CCD pixel. When the CCD pixel area increases, the sensitivity of the pixel increases and when the CCD pixel area decreases, the sensitivity of the pixel decreases. However, the CMOS sensor is inherently different. The overall sensitivity of a CMOS sensor is independent of its area. Regardless of whether the manufacturer enlarges or shrinks the area of the CMOS pixel sensor, the sensitivity of the APS will remain the same.

CCDs are known for their high-power consumption. The opposite behavior is seen with CMOS sensors. Its name suggests that there is a complementary pair of transistors in the sensor. However, though there is a pair, both transistors are not always active as only one is turned on. By this we can see that the overall power consumed is effectively reduced in half. Additionally, it reduces the sensor's total exposure to noise compared to

the CCD. The only time power is consumed is the moment there is a logical change in the circuit's output. From all of these advantages it can be reiterated why CMOS sensors are the default technology used in modern digital cameras.

There is a general layout when it comes to designing a pixel sensor in the CMOS sensor. Some designs involve more transistors than others. Typically, after designing a single CMOS pixel, you can reuse and combine multiple of those CMOS pixels to form a CMOS cell. These pixels in the CMOS cell or array are called 'active pixels' – hence, the active pixel sensor name. Looking at the composition of the active pixel sensor, it uses a photodiode which takes up around twenty-five percent of the device. For the purposes of our analysis and understanding we will discuss two types of CMOS pixel designs – the first design being the 3T (Three Transistor) CMOS pixel design and the second design being the 4T (Four Transistor) CMOS pixel design. The main responsibility of these transistors in both the 3T and 4T pixel designs is to take in light and transform it into electrical energy. The capacitance is larger than the rest of the circuit. In fact, the charge is collected at the gate of the output transistor. As the technology shrinks, the capacitance scales with the size of the photodiode. Hence, the photocurrent and capacitance scale but the sensitivity stays constant when we shrink the pixel. This stays true for DSLRs, but not for cell phone pixels which have micron size pixels. In this case, stringent edge effects start appearing on the pixel. This is the reason, unlike DSLRs, cell phone pixels experience high levels of noise even at modest ISO (sensitivity) levels. To tackle this, we leverage different detection methodologies for cell phones.

The most basic active pixel sensor is the photodiode APS. In this circuit design, there are three transistors that dictate the function of the pixel. An important trait of the photodiode to point out is that it takes up the majority of the CMOS sensor's space. Dissimilarly, the three transistors take up much less space. Since this is a photodiode APS, the photodiode that is part of the circuit takes in incident light (in the form of photon light) and convert it into both current and voltage. One thing to keep in mind is that the photodiode in the circuit is brought to high in order to cause the light entering inside it to discharge itself. We measure the output of the pixel cell as this pixel discharge. By doing this, the circuit removes the existence of any outliers when it comes to extracting outputs. Figure 2.8 gives a good illustration of the more common design for a basic 3T CMOS pixel.



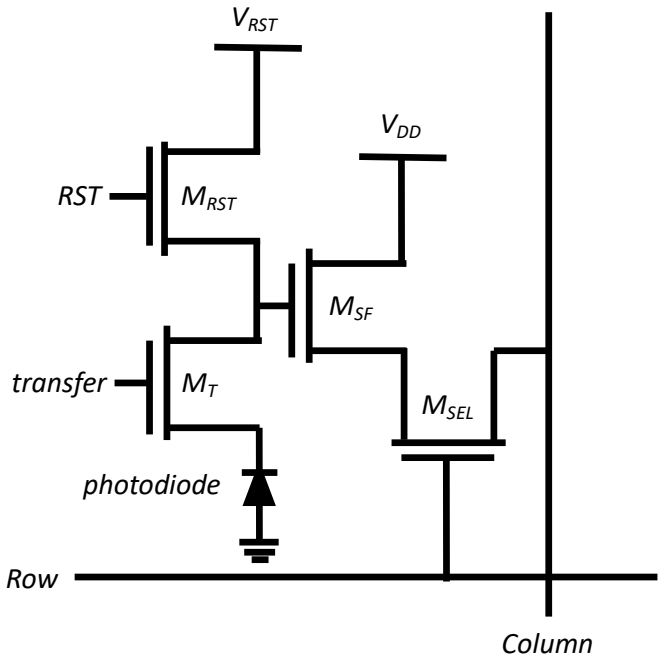
Legend

V_{RST} : Voltage Reset
V_{DD} : Voltage Drain Drain
M_{RST} : Reset Transistor
M_{SF} : Source Follower Transistor
M_{SEL} : Row Selector Transistor

Figure 2.8 Basic 3T CMOS pixel

The three transistors in this circuit that control the operation of the pixel are most commonly known as the row selector transistor, the source follower transistor and the reset transistor. It is beneficial to discuss the role of each of these three transistors in more detail. The first transistor to discuss is the source follower transistor or amplifier. The purpose of this transistor is to temporarily act as a buffer in the design. Its role is to gather the photodiode output while not removing the diode's charge. The second transistor is the reset transistor. The reset transistor acts as a module to either take the circuit into a reset mode or take it out of that reset mode. When a digital camera is turned on by the user, we see the need for the reset transistor. At 'ON', the pixel needs to be reset for a short period of time when the other software and firmware components in the camera are being initialized. After this initial setup, the reset can be turned off and the pixel can be brought out of the reset stage to its regular function by the reset transistor. The last transistor of discussion is the selection transistor. This transistor enables control circuits from higher levels to select the specific pixel values it wants to read. This is achieved via column and row selectors.

Another common design involves four transistors instead of three. This is a more multifaceted design and is referred to as the 4T CMOS pixel design. Figure 2.9 illustrates the design for a 4T CMOS pixel.



Legend

V_{RST} : Voltage Reset
V_{DD} : Voltage Drain Drain
M_{RST} : Reset Transistor
M_{SF} : Source Follower Transistor
M_{SEL} : Row Selector Transistor
M_T : Transfer Transistor

Figure 2.9 Basic 4T CMOS pixel

To simplify things a little, we can view a 4T CMOS design similarly to a 3T CMOS design. Taking the 3T CMOS design, adding a transistor between the photodiode, source follower and reset transistors changes it into a 4T CMOS design. The added transistor is called the transfer transistor. This transfer transistor makes the overall circuit faster and increases the precision when getting the total charge from the photodiode. By no means are these the only CMOS pixel designs used in digital photography. In fact, recently there has been

an increased emphasis on increasing the dynamic range of pixels and sensors as a whole – the name for this is high dynamic range (HDR). These CMOS pixel designs will have special transistors to help achieve this HDR functionality.

In linear algebra, the concept of a matrix represents element values that are structured in an array by rows and columns. The same approach can be taken to visualize a CMOS sensor. In a CMOS camera sensor, array elements are CMOS pixels and are structured in an array to form the rectangular CMOS sensor array. Additionally, each given pixel circuit in the CMOS sensor array leverages a row and a column selection circuity. Fundamentally this uses a big multiplexer to read out the pixel's value based on the selected row and column. Something to point out is that the pixels have a reset signal that is shared between themselves. This allows them to use a signal for multiple pixels as opposed to having one per pixel. Looking at the more present-day CMOS designs, instead of reading out each pixel individually, the circuits will read the values of the pixels in batches of rows or columns. This helps speed up the read-out time when reading out the individual pixel values and results in better overall performance.

Another area to make mention of is the overall area composition of the pixel circuitry. It was mentioned that the photodiode takes up a majority of the space in the CMOS pixel's circuitry. This helps the circuit to maximize its light exposure. Since there are other transistors in the pixel, the entire circuit cannot be used for light absorption. Hence, designs that add additional transistors reduce the overall fill factor. The term used to describe the ratio of light sensitive area to the total area in a pixel cell is referred to as the pixel fill factor. A higher fill factor corresponds to less transistors in a circuit and lower fill factor corresponds to more transistors in a circuit. When more transistors are added to the circuit, the total exposure improves at the expense of a more complex design. To illustrate the fill factor concept more clearly, we will take a look at Figure 2.10.

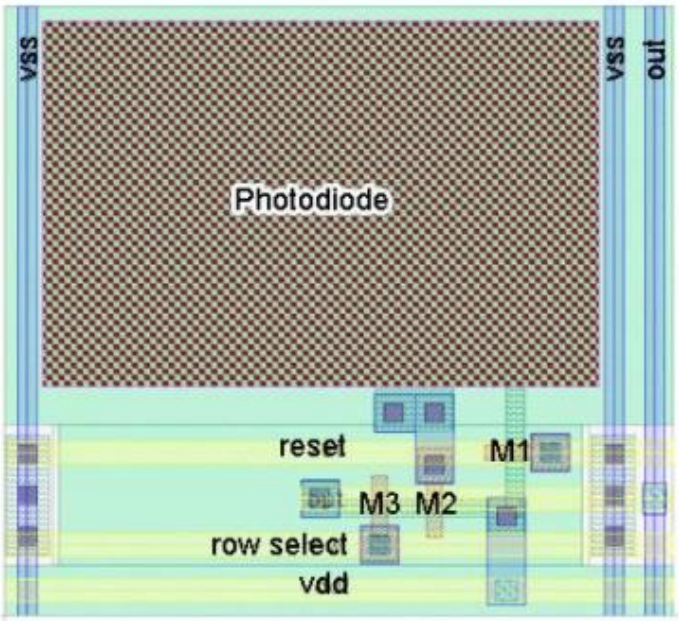


Figure 2.10 Pixel Layout (taken from Rehm[8])

Figure 2.10 depicts the pixel layout of a 3T CMOS pixel cell of 8 μm in size. Recall that the photodiode takes up a majority of the area compared to the other transistors. In this example the fill factor is ~50% and the light sensitivity area is 4.44 μm x 6.75 μm. In all newer cameras, micro-lenses will be used to boost the light collection of the transistor. This way, the fill factor becomes greater than 90%.

2.5 Digital Camera Photography

To better understand camera sensor defects, it is important to examine the fundamentals when dealing with photography and digital cameras. Figure 2.11 shows an image taken of a Canon T5i DSLR's touchscreen display.

Figure 2.11 Image taken of a Canon T5i DSLR's configuration screen

This display provides the photographer with many digital configuration options that they may choose from. Most modern digital cameras have the capability to configure digital photography settings such as ISO and shutter speed for example. These will not be explained immediately; however, they will be explained towards the end of the chapter.

It was discussed previously how an active pixel sensor is the most granular CMOS sensor for an individual pixel. Multiple CMOS pixel sensors are combined to form an array of CMOS pixel sensors – this array is referred to as the CMOS camera sensor. The entire camera sensor as a whole has the responsibility to gather and record the incident light entering during a given period of time. The light accepting process generates the resulting captured image. The photo capture process is a fairly complex concept that has many parts. Some of the aspects that are involved in the process of image capture are exposure time to light, sensitivity to light (generally known as ISO), interpreting the captured colours, and image formatting. Until now we have primarily discussed the hardware components of a digital camera and its sensor. However, the remaining sections of this chapter will give a quick overview of the various concepts that are used in digital photography. We will dive more granularly into the individual sensor pixels by looking at things at a sensor's point of view. Additionally, by doing this we will look at how everything works together in the digital camera to finally produce the captured image.

2.6 The Colour Filter Array and Bayer Pattern

One inherent problem in Digital cameras is the need to convert the full wavelength sensitivity of the photodiode pixel, and into pixel arrays that can sense, and thus reproduce, colour images. This is most commonly done using the Bayer mosaic pattern, which can be seen of Figure 2.12, which is applied over the sensor array. It is the most common colour filter array and is an RGB combination of red filters, green filter and blue filters band pass type filters applied over the pixel. In this section we will explain the Colour Filter Array (CFA).

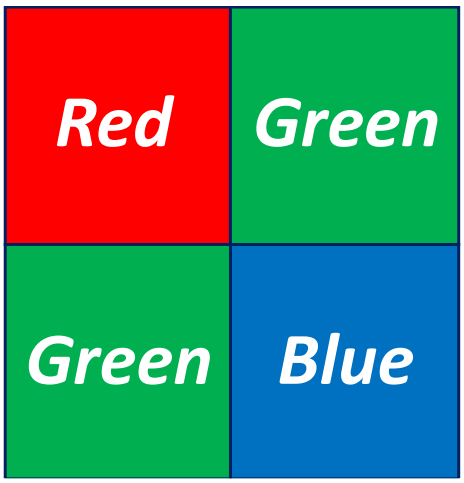


Figure 2.12 The Bayer Mosaic Pattern

Pixels are monochromatic, meaning they only contain information regarding one colour and oftentimes that colour is unknown. The APS pixel must undergo a process known as colour separation where it converts the image in an array of RGB spectral bins. A widely used mechanism to perform colour separation is the CFA. When a sensor uses the CFA mechanism it is referred to as a CFA sensor. In general, an APS is required to process a wide range of color wavelengths. To achieve this, ideally each pixel in the image sensor is covered with a colour filter. One thing to note is that the area of the colour filter is equivalent to the area of the pixel itself. Theoretically, the respective colour filter only lets the respective colour wavelength to pass through and rejects the other colour wavelengths. An example of red, green, and blue filters are as follows: A red filter will allow wavelengths from 625 nm to 740 nm, the green filter will allow wavelengths from 500 nm

to 565 nm, and the blue filter will allow wavelengths from 440 nm to 485 nm [40]. If a pixel location has a corresponding blue filter, it will only allow blue light (440 nm to 485 nm) to pass through to the photodiode. If any of the other colors – red light or green light – were to be sent to the blue filter, the photodiode would not receive any charge. Hence, there would be no output for the pixel.

The similar behaviour is portrayed by red and green filters as well. Only light that meets the wavelength requirements of the colour filter at that given pixel location will be allowed to pass through to the photodiode and gather charge. Figure 2.13 below illustrates the behaviour of light passing through red, green, and blue colour filters.

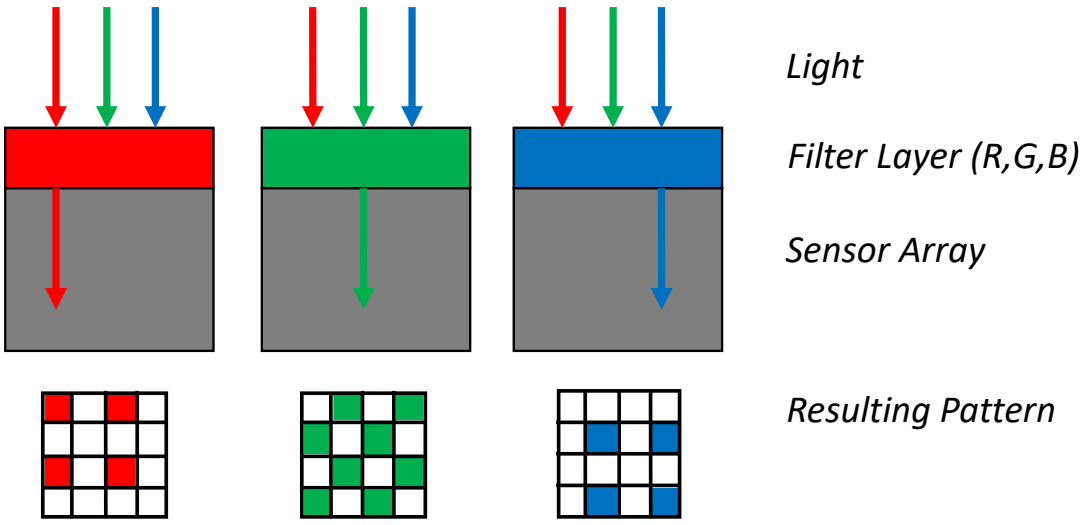


Figure 2.13 Color Filters for Red, Green, and Blue (taken from [41])

Out of all the light wavelengths, green light is able to penetrate to the eye the most in comparison to the red-light wavelength and blue light wavelength. Hence, the Bayer mosaic pattern takes that into consideration as it will typically have twice the amount of green in comparison to the other colours. The Bayer mosaic pattern involves a red layer filter, a green layer filter and a blue layer filter – with each Bayer filter compassing a 2x2 pixel square. The most common 2x2 pixel composition of the Bayer mosaic pattern has one red component, two green components, and one blue component – this is generally referred to as the RGGB Bayer pattern. Although GRGB and RGBG combinations are also feasible, they are not as popular as the RGGB layout. This concurs with our previous

statement that the human eye is more delicate to green light and the Bayer pattern accounts for this by containing two green components for every red and blue pair. Although this is the most common approach, some manufacturers rely on different implementations for their colour organization. Lastly, multiple 2x2 Bayer patterns will be cascaded together to form the colour filter array. Figure 2.14 below portrays an example of a colour filter array.

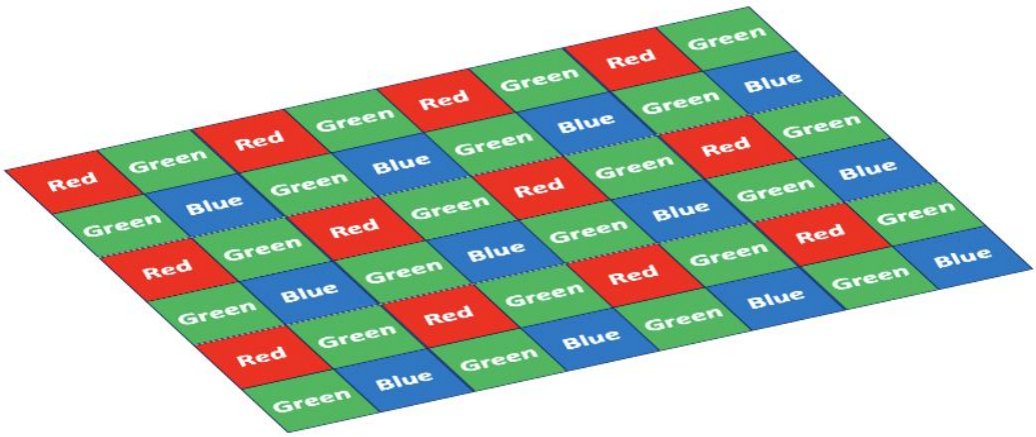


Figure 2.14 Color Filter Array for an image sensor

2.7 Colour Demosaicing

Ideally every pixel in the image array would have a true colour measurement of red, green, and blue levels. While this happened in classic colour film, and in certain specialized pixel designs (the Foveon pixel) regular cameras cannot do this. In most imaging sensors a given pixel location will only have one of the three colours as it will only have one of the respective colour filters – as per the CFA layout. To compensate for this the images must be converted using the concept of ‘demosaicing, which uses software interpolation algorithms to calculate the estimated colour breakdown (red level, green level, blue level) of each pixel in the image sensor. By doing this, a digital camera can

leverage both the Bayer mosaic pattern and demosaicing to calculate a close approximation of the image colour of every pixel in a captured image.

Some related issues with the CFA method or Bayer mosaic pattern is that potential occurrences of Moiré patterns and resolution loss in parts of an image – sometimes in multiple parts of a captured image. A Moiré pattern is a severe interference pattern that can be seen in the captured image but not in the original scene. Sometimes certain camera manufacturers associate these patterns with anti-aliasing filters in digital cameras. Hence, these manufacturers remove the anti-aliasing filters in their digital cameras altogether. Another known CFA related issue is that they generally preserve twice the amount of green light information in comparison to red light and blue light (recall the RGGB Bayer mosaic pattern). As a result, a lot of information involving red light and blue light is not recorded in the image capture process. This fortifies the argument that digital cameras using the CFA approach are not fully sensitive to red and blue. As we will see, Hot pixels and SEUs can both be affected by colour demosaicing as their pixel values and colours get affected. Before moving on, it is paramount to discuss a few of the common demosaicing algorithms that are used in modern digital photography. From Leung [42] we see they are:

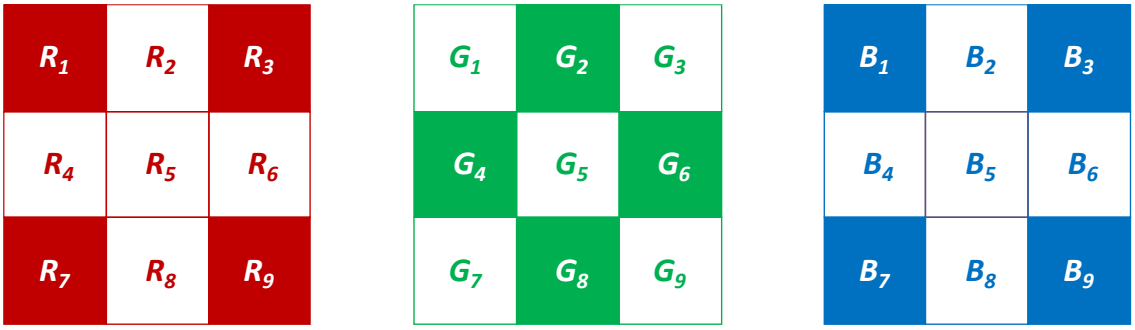
- The Bilinear Demosaicing Algorithm
- The Median Demosaicing Algorithm
- The Kimmel Demosaicing Algorithm

We will discuss all three demosaicing algorithms before proceeding further. Please keep in mind that these are not perfect filters that often result in false colour at edge boundaries, Moiré patterns and resolution loss.

2.7.1 The Bilinear Demosaicing Algorithm

The Bilinear demosaicing algorithm is the simplest and almost always used as the initial step for other more complex methods. The idea of this algorithm is to calculate the value of a missing colour in a pixel location by interpolating the values of adjacent pixels

in the missing colour plane. Each pixel location will have the value of one colour and will require the calculation of the two other colours. However, the two estimated colour values for each pixel will be calculated separately – not relying on each other. Figure 2.15 depicts the colour kernels that are used to interpolate the respective missing colours in a pixel. This method is known to perform rather quickly, however, it suffers from many artifacts such as Moiré patterns.



(a) Red Pixel Kernel **(b) Green Pixel Kernel** **(c) Blue Pixel Kernel**

Figure 2.15 Color Filter Array for an image sensor

The CFA from Figure 2.14 and the kernels for red, green, and blue in Figures 2.15 (a)–(c) respectively can help us formulate the equations that are used in Bilinear demosaicing. Equations 2.4, 2.5, and 2.6 depict the method used to calculate the missing red, green, and blue values respectively.

$$R_2 = \frac{R_1 + R_3}{2};\ R_4 = \frac{R_1 + R_7}{2};\ R_6 = \frac{R_3 + R_9}{2};\ R_8 = \frac{R_7 + R_9}{2};\ R_5 = \frac{R_1 + R_3 + R_7 + R_9}{4} \quad (2.4)$$

$$G_5 = \frac{G_2 + G_4 + G_6 + G_8}{4} \quad (2.5)$$

$$B_2 = \frac{B_1 + B_3}{2};\ B_4 = \frac{B_1 + B_7}{2};\ B_6 = \frac{B_3 + B_9}{2};\ B_8 = \frac{B_7 + B_9}{2};\ B_5 = \frac{B_1 + B_3 + B_7 + B_9}{4} \quad (2.6)$$

The equations above are used to find the missing colours in each pixel throughout the camera sensor. As mentioned previously, the Bayer pattern uses a RGGB layout –

hence, there are 2 green pixels for every red and blue pixel. As a result, the bilinear interpolation equation (2.5) always evaluates using 4 green pixels while the red bilinear interpolation equation (2.4) uses only 2 red pixels; unless in the case of R_5 where the missing red pixel is surrounded by 4 red pixels – but this is less frequent. The same applies to blue pixels as there is a higher frequency of interpolating the missing blue pixel using 2 red pixels instead of using 4 red pixels. Therefore, the interpolations for missing green pixels are more accurate than the other colours in bilinear interpolation. Overall, the bilinear demosaicing algorithm is suited for slowly changing areas. However, the location and direction of edges create many problems such as colour artifacts and fringing. Alternatively, the Median and Kimmel demosaicing algorithms are optimized to reduce these effects.

2.7.2 The Median Demosaicing Algorithm

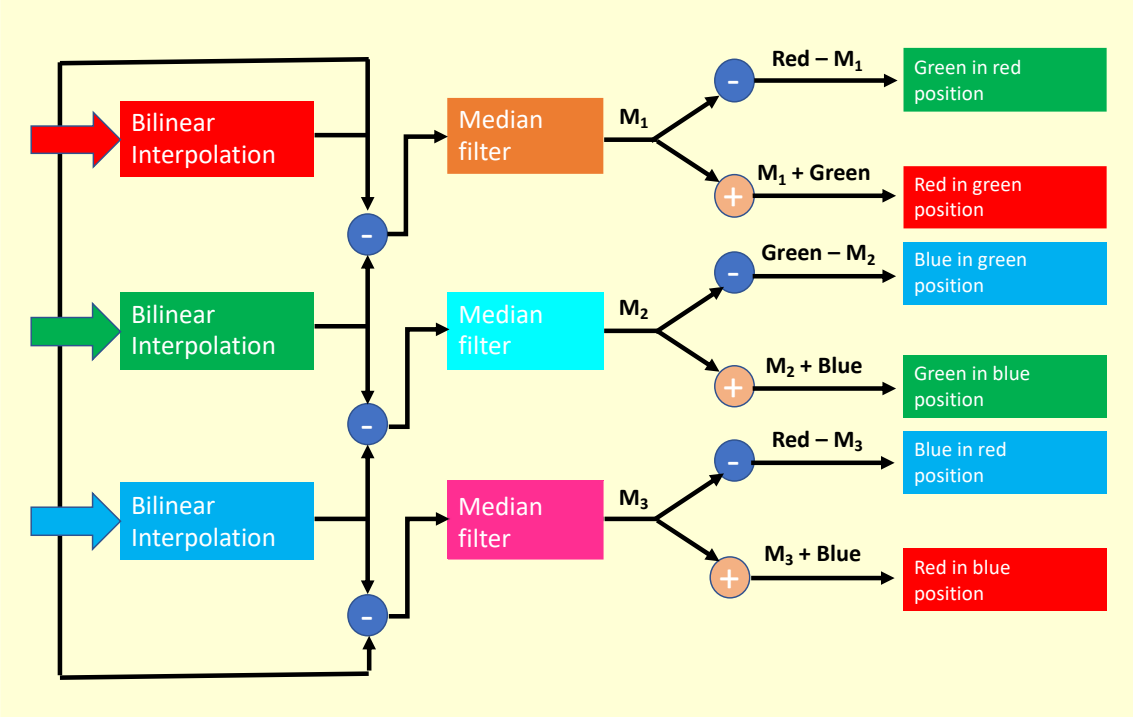


Figure 2.16 The Freeman Median Demosaicing Algorithm (taken from Zapryanov [43])

To improve the demosaicing algorithm the basic bilinear interpolation is corrected by processes such as the Median demosaicing algorithm by created Freeman [43]. The major variation from the earlier method is that this method leverages other colour planes rather than relying on the same colour plane. Figure 2.16 shows the workflow of the median interpolation by Freeman. The filter in this algorithm looks at the surrounding pixels to find the median value.

The median method begins by recovering the missing colours using bilinear interpolation. Next, it uses equations (2.7) through (2.9) to compute the difference between each colour plane. The functions $f_R(x, y)$, $f_G(x, y)$, $f_B(x, y)$ express the value at pixel location (x, y) for the red, blue, and green colour planes respectively. Variations to the algorithm – such as Cok's implementation – use ratios instead of differences [43].

$$D_{RG}(x, y) = f_R(x, y) - f_G(x, y) \tag{2.7}$$

$$D_{GB}(x, y) = f_G(x, y) - f_B(x, y) \tag{2.8}$$

$$D_{RB}(x, y) = f_R(x, y) - f_B(x, y) \tag{2.9}$$

After calculating the differences, a median filter is applied to the values of $D_{RG}(x, y)$, $D_{GB}(x, y)$, and $D_{RB}(x, y)$ to generate values for M_1, M_2, and M_3. The role of the median filter is to remove any large inconsistencies between the colour planes by looking at the adjacent pixels. For each pixel location (x,y), the median filter is often applied to either a 3x3 pixel area or 5x5 pixel area around (x,y). Finally, the algorithm will perform a correction in which M_1, M_2, and M_3 are used to correct the bilinear interpolated red, green, or blue colours. The median method is much better at reducing artifacts at edge regions in the image. This becomes important when dealing with sensor defects as they do fall under the category of edges that might eventually result in artifacts. Unfortunately, the orientation of the edge becomes important and so a more complex iterative method is applied.

2.7.3 The Kimmel Demosaicing Algorithm

Unlike the Bilinear and Median demosaicing algorithms, the final algorithm that will be discussed is an adaptive method – the Kimmel demosaicing algorithm Leung [42]. Adaptive methods are more complex and leverage local area equations to achieve the best results. The Kimmel interpolation is composed of multiple methods – bilinear interpolation, a weighted gradient, and colour interpolation. Before proceeding to the actual steps of the algorithm it is important to take a look at the green pixel kernel in Figure 2.17.

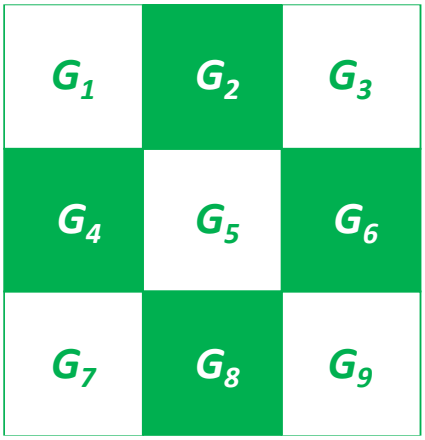


Figure 2.17 Green Pixel Kernel

As with any other demosaicing algorithm, the Kimmel method begins by using bilinear interpolation. Figure 2.17 shows that G_2, G_4, G_6, and G_8 are used in the regular bilinear method to interpolate the missing green value (G_5). However, this time we will use a weighted version to compute the missing green values. The intent of this is to adjust the bilinear algorithm that we saw earlier into the direction of the local edge. Hence, equation (2.10) calculates the missing green value (G_5) using weighted bilinear interpolation.

$$G_5 = \frac{E_2 G_2 + E_4 G_4 + E_6 G_6 + E_8 G_8}{E_2 + E_4 + E_6 + E_8} \tag{2.10}$$

For the next set of equations, it is important to take a look at the Kimmel gradient kernel in Figure 2.18. The reason for using P_i instead of the R_i, G_i, or B_i notation is to show that the calculations remain the same irrespective of the particular colour at P_5.

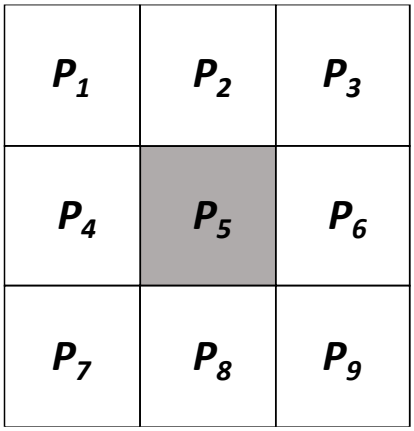


Figure 2.18 Kimmel Gradient Kernel

After understanding the Kimmel gradient kernel in Figure 2.18, we can use (2.11) to calculate the weights.

$$E_i = \frac{1}{\sqrt{1 + D^2(P_5) + D^2(P_i)}} \tag{2.11}$$

Equation (2.11) is used to calculate each weight factor that is used in equation (2.10). The value of D that is shown in (2.11) represents the gradient function. To find the value of D we will use the appropriate equation from (2.12) through (2.15) – depending on the direction of P_i in respect to P_5.

$$D_{\mathrm{y}}(P_5) = \frac{P_2 - P_8}{2} \tag{2.12}$$

$$D_{\mathrm{x}}(P_5) = \frac{P_4 - P_6}{2} \tag{2.13}$$

$$D_{\mathrm{yd}}(P_5) = \max\left\{\frac{P_1 - P_5}{\sqrt{2}}, \frac{P_9 - P_5}{\sqrt{2}}\right\} \tag{2.14}$$

$$D_{\mathrm{xd}}(P_5) = \max\left\{\frac{P_3 - P_5}{\sqrt{2}}, \frac{P_7 - P_5}{\sqrt{2}}\right\} \tag{2.15}$$

Something to point out is that (2.12) and (2.13) will be used to calculate the vertical and horizontal gradients respectively. Equations (2.14) and (2.15) will be used to calculate the +45° and -45° diagonals. This completes the first step in the Kimmel demosaicing which is the weighted bilinear interpolation.

The second step involves interpolating the missing red and blue components by means of ratios. Equations (2.16) and (2.17) show this step. You will notice that both equations are relying on green values. This is because there is more green information than blue and red information in the Bayer filter – leading to the notion that the ratio between the colour planes stays the same.

$$R_5 = \frac{\left(E_1 \cdot \frac{R_1}{G_1}\right) + \left(E_3 \cdot \frac{R_3}{G_3}\right) + \left(E_7 \cdot \frac{R_7}{G_7}\right) + \left(E_9 \cdot \frac{R_9}{G_9}\right)}{(E_1 + E_3 + E_7 + E_9)} \times (G_5) \tag{2.16}$$

$$B_5 = \frac{\left(E_1 \cdot \frac{B_1}{G_1}\right) + \left(E_3 \cdot \frac{B_3}{G_3}\right) + \left(E_7 \cdot \frac{B_7}{G_7}\right) + \left(E_9 \cdot \frac{B_9}{G_9}\right)}{(E_1 + E_3 + E_7 + E_9)} \times (G_5) \tag{2.17}$$

The final step involves a correction process to ensure the proper colour ratio. The computed green values from weighted bilinear interpolation are recalculated using the computed red and blue values from (2.16) and (2.17) respectively. This is shown by equations (2.18) through (2.20).

$$G_{R_5} = \frac{\left(E_2 \cdot \frac{G_2}{R_2}\right) + \left(E_4 \cdot \frac{G_4}{R_4}\right) + \left(E_6 \cdot \frac{G_6}{R_6}\right) + \left(E_8 \cdot \frac{G_8}{R_8}\right)}{(E_2 + E_4 + E_6 + E_8)} \times (R_5) \tag{2.18}$$

$$G_{B_5} = \frac{\left(E_2 \cdot \frac{G_2}{B_2}\right) + \left(E_4 \cdot \frac{G_4}{B_4}\right) + \left(E_6 \cdot \frac{G_6}{B_6}\right) + \left(E_8 \cdot \frac{G_8}{B_8}\right)}{(E_2 + E_4 + E_6 + E_8)} \times (B_5) \tag{2.19}$$

$$G_5 = \frac{G_{R_5} + G_{B_5}}{2} \tag{2.20}$$

After recalculating the green values using (2.18) through (2.20), the red and blue values need to be readjusted by recalculation using equations (2.21) and (2.22) respectively.

$$R_5 = \frac{\sum_{i=1}^{9} \left(E_i \cdot \frac{R_i}{G_i} \right)}{\sum_{i=1}^{9} E_i} \times G_5 \,, \qquad i \neq 5 \tag{2.21}$$

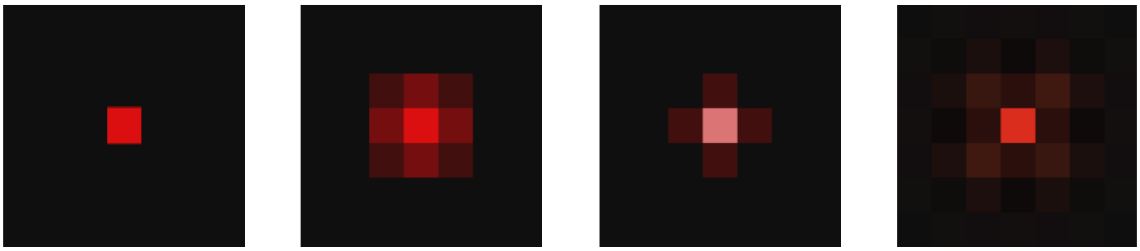
$$B_5 = \frac{\sum_{i=1}^{9} \left(E_i \cdot \frac{B_i}{G_i} \right)}{\sum_{i=1}^{9} E_i} \times G_5 \,, \qquad i \neq 5 \tag{2.22}$$

The final correction process is iterated multiple times – typically 3 or more times. This finally concludes the Kimmel demosaicing algorithm. The benefit of combining multiple methods – bilinear interpolation, weighted gradient, and colour interpolation – is that it reduces the number of artifacts that are produced compared to other demosaicing algorithms. Hence, this is why many camera manufacturers and photography software programs choose the Kimmel demosaicing algorithm. There are many other complex demosaicing methods, but they are not as popular.

2.7.4 Demosaicing Imager Defects

We have gone over three types of demosaicing algorithms – Bilinear interpolation, the Freeman Median method, and the Kimmel algorithm. However, these methods are designed for relatively large objects within the imaging field and do not responds well to single defects like hot pixels. Given the research in camera sensor defects it is paramount to discuss the impact that demosaicing can have. As mentioned earlier, bilinear demosaicing takes into account the adjacent pixels and produces artifacts in the image. If the camera sensor had a defect captured in a dark-frame image, then bilinear demosaicing creates artifacts, spreading the impact of the defect pixels and interfere with our defect

analysis (see Figure 2.19(a)). The two more advanced methods (the Median and Kimmel algorithms) create less artifacts, but the Kimmel algorithm actually spreads the defect even further – see Figure 2.19(d). Figure 2.19 illustrates a single pixel defect and the impact by each of the three demosaicing algorithms.



(a) Original (b) Bilinear (c) Median (d) Kimmel

Figure 2.19 Original single pixel defect and impact of demosaicing algorithms (taken from Chapman [18])

From Figure 2.19(b) we can see that bilinear interpolation creates the most damage compared to the Median and Kimmel results in Figures 2.20(c) and 2.20(d) respectively. While the Median creates little damage, it causes a significant reduction in the colour intensity of the original image because it spreads the defect into other colours (hence the pastel image). This is something that even the bilinear method does not do. As a result, the Kimmel method provides the most desirable demosaicing outcome of the three – maintaining the colour intensity of the original image and minimizing the artifact effect on edges. Towards the end of this chapter we will discuss certain image formats – such as digital RAW – that do not perform demosaicing of any kind. These image formats are ideal for defect analysis there is no chance for demosaicing artifacts.

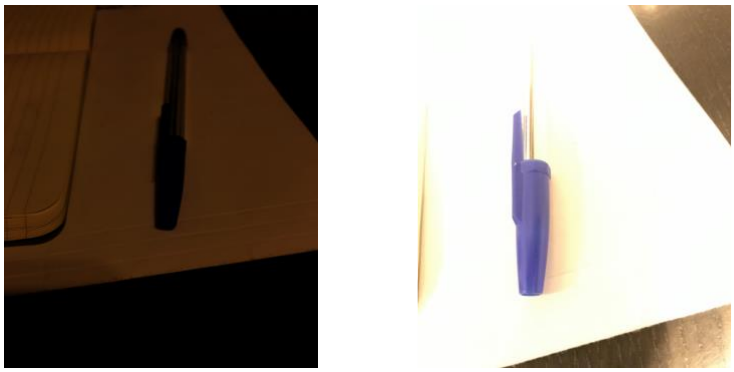
2.8 Exposure Time

Outside of CMOS technology and engineering – in the context of general photography – the most important concepts are exposure time, ISO, and f-number (or aperture). First, let us take a look at and discuss exposure. Before going any further, it is

important to define the term of exposure. Typically, light exposure is defined by the following formula:

$$H = E \cdot t \tag{2.7}$$

In (2.7), H is defined as the luminous exposure of a sensor – measured in lx·s (lux seconds). H is calculated by two inputs which are E and t. E is defined as the illumination of the sensor's surface area – measured in lux (units are lx). In physics, this is also sometimes referred to as the total flux of the surface area. Lastly, t is the exposure time – measured in seconds (s). Together these two terms multiply to output the luminous exposure, H. Hence, the product units are lx·s. From mathematical intuition we can see that this is a linear equation and that the luminous exposure scales linearly in proportion to the exposure time (t). As we increase the exposure time, the total exposure goes up. Similarly, as we decrease the exposure time, the total exposure goes down.



(a) (b)

Figure 2.20 (a) An under exposed image of a pen and (b) an over exposed image of a pen

The two images in Figure 2.20 illustrate a pen at different exposures. Figure 2.20 (a) shows an under exposed image of the pen and Figure 2.20 (b) shows an over exposed image of the pen. Underexposure gives the picture a very dark or low-light environment while overexposure gives a very bright or high-light environment. In most scenarios both overexposure and underexposure image captures are not ideal and cause problems to the final image. For example, an underexposed image will have the underlying trait of being too gloomy in contrast to the actual real scene viewed by a human eye. Similarly,

an overexposed image will not be able to retain certain details in the captured image as the overexposure tends to hide them. This is also not the actual real-life scene viewed by a human eye.

For most people trying to do digital photography, it becomes paramount to know how to change the exposure time of image they are about to capture. This is controlled by a configurable feature known as shutter speed. Exposure time and shutter speed may be used interchangeably. When a picture is taken, its pixels are exposed to the light for a certain duration. Exposure time is essentially the time the camera's pixels are exposed to the incident light. Slower shutter speed results in a longer exposure, which means the camera's pixels are exposed to the incident light for a longer duration. Faster shutter speed results in a shorter exposure, which means the camera's pixels are exposed to the incident light for a shorter duration.

When it comes to the units used to express shutter speed or exposure time, cameras use seconds as well as fractions of seconds. One of the quickest cameras on the market is capable of capturing an image with a shutter speed of 1/8000 seconds. This theoretically means that at this exposure speed the camera can take 8000 images in a single second. However, in reality since there are other behind-the-scene tasks that need to complete after each image capture (probably taking a few seconds), the camera will not be able to take 8000 continuous photos in a span of a second. On the long side of things, most DSLR cameras are capable of achieving exposures as high as 30 seconds. If an image were taken with a 30 second shutter speed, that means that the pixels will be exposed to the incident for that long. As expected, cameras are not limited to a high and a low shutter speed. There are ranging shutter speeds that will be available between the longest and shortest exposure times. Since this thesis will focus on digital camera defects that are categorized as single event upsets, varying exposure times will play an important role in the test data that is presented. The pixel and defect characteristics are not always constant as the exposure time changes – hence, the necessity to further analyze it. Prefacing a correlated topic that will be explored in greater length later in this thesis is how increasing exposure time results in more noise being seen in our test images. This directly influences our analysis and will be discussed in greater detail in the following chapters.

2.9 Aperture

In relation to the previous section of exposure time is the concept of aperture. It was mentioned that the shutter speed option on the camera configures the exposure time value in which the camera allows its pixels sensors to be exposed to light. To help understand this better, a common analogy that is mentioned to relate exposure time and aperture is that of the human iris [44]. Aperture is essentially the size of the light opening or how much light is allowed to come through the lens into the camera sensor. One thing to note is that this concept is not something that was introduced with digital photography. In fact, it existed in the days of film photography as well. When contrasting to shutter speed, you may think of aperture as the amount of light allowed in the camera sensor instead of the duration that light is allowed into the camera sensor. This can be better illustrated via an example. For a fixed shutter speed and a bright environment, the aperture or light opening does not need to that high. For the same shutter speed in a dark environment, a higher aperture setting is required as it requires more incident light entering into the camera [44]. Hence, a simple method to classify the two is to think of shutter speed as the duration of light exposure and aperture as the size of the light opening. Photographers typically look at f-numbers on lenses to determine the maximum aperture, however f-stops and f-numbers will not be discussed in this thesis.

2.10 ISO

In general photography – that is both digital and film – the concept of ISO refers to the sensitivity to light during image capture. Increasing the ISO increases the camera's sensitivity to light and generally makes your image brighter. To the same effect, decreasing the ISO decreases the camera's sensitivity to light and makes your image darker. Dating back to film cameras, ISO was known as film speed or sensitivity. When it came to film photography, the film sensitivity was a constant that was determined by the type of film that was being used. Lower film sensitivity meant that the image capture was less sensitive to light. The development of bigger grains led to the production of higher

ISO film – this came at a trade-off with image resolution. Once the film was in the camera, the ISO could not be changed as it was fixed. In order to brighten the image capture, the exposure time would have to be increased. The inherent relationship between ISO and the exposure time is fairly straightforward in the context of film photography since the aperture is held constant. For example, if ISO 400 and an exposure time of one second are used for a given scene, then for ISO 800 the needed exposure time will be half of a second. In this example doubling the ISO reduces the exposure time into half of its original value. This reiterates that ISO and exposure time are inversely proportional to each other. Unlike in film photography, in digital photography there is no concept of fixed film speed or sensitivity for a respective film type. Since there is no film, the light sensitivity is referred to as ISO.

There are many other factors that come into play when discussing light sensitivity in digital cameras – some of these include pixel noise, sensor characteristics, and the camera's software processing. Due to all of these factors, the sensitivity of the outputted image capture will be different than the light sensitivity seen by the human eye at the same scene. When it comes to ISO settings, they are configurable on a digital camera as opposed to a film camera. This is possible by amplifying the output of the pixel sensor to the correct amplification in order to get the desired sensitivity. In the midst of this, the charge of the photodiode in the APS circuit does not get amplified by the ISO level itself. ISO may be thought of as an amplifier that increases the gain of the signal. When the gain is increased the pixel information from the output at the top gets thrown away and the noise at the bottom gets amplified. When the gain is turned up, more thermal heat is produced in the pixel sensor. This results in more noise being generated as well – recall the grainy look of higher ISO images. Modern digital cameras leverage a lot of noise suppression to combat the high noise levels at high ISO levels. As a result, this is a trade-off with image resolution. Typically, to reduce the thermal noise in the image, the noise suppression algorithms smear or blur the image. This is an undesirable outcome when doing defect analysis. Another thing to take note of is that the same amplification or gain level is applied to all the pixels in the digital camera sensor in order to make it uniform. Hence, when a pixel location of a transient defect gets amplified, the single event upset will appear more apparent or intense at higher ISO levels.

Since ISO is a fundamental part of defect analysis, it will be looked at in more detail. Both SEU and noise amplification from increasing ISO levels will be discussed

further in the following chapters. In terms of configuring the ISO settings, each digital camera manufacturer has their own defined settings and implementations. For the most part, the manufacturers try to make the digital camera sensitivity settings output equivalent images as their film counterparts. However, digital photography is not quite at the resolution quality as film photography for a given ISO level.

2.11 Digital Image Format

The last item that will be covered in this section is the format of digital images. This section will briefly discuss different image formats that are used in digital photography and the preferred format when detecting image defects. The conversion of RAW image data to a format that is consumable by the end user, such as JPEG is typically performed through a workflow of multiple stages. To make it clearer, it is important to look at the image processing method inside the camera more granularly. After the image has been captured by the CMOS sensor – that is, after the lens' shutter has been closed and the light absorption process has ended – the image will be generated as RAW data. This form of the image is most commonly known as the RAW image. RAW data is essentially the read-out representation of the imager's pixels. The camera will run its noise suppression algorithm on the image between the image capture and generating the RAW image. Typically, a RAW image stores its pixel values in either 12-bit or 14-bit values. With a 12-bit value there are approximately 68 billion color values and with a 14-bit value there are approximately 4 trillion color values. Another point is that RAW images have no post-processing, compression, or demosaicing algorithms performed on them. Hence, making them very large and hard to view. Given this nature, RAW images give photographers the flexibility to process images in the way of their choosing before converting to compressed image formats such as JPEG.

During the conversion process to compressed image formats, the RAW image data will undergo noise suppression and demosaicing. The noise suppression process is often a proprietary solution implemented by the camera manufacturer. Depending on the make and model of the camera there may be one or more noise suppression stages. Noise in an image can result from a number of sources such as sensor heat, long exposure times,

and high ISO values to name a few. To help combat this, noise suppression is often used; but, can have both advantages and disadvantages. The benefit of noise suppression is that it has the ability to remove or reduce the intensity of noise throughout the image. The disadvantage is that it can remove clarity or sharpness to areas in an image. In addition, some styles of photography view noise as an important effect in the image scene. The other process that happens during the image conversion process is demosaicing. As we saw earlier, each pixel location in a RAW image only contains information of one colour (Red, Green, or Blue). As a result, demosaicing algorithms are used to create a true colour image with interpolated values for the missing colours at each pixel location. Similar to noise suppression, each camera manufacturer typically implements their own proprietary demosaicing process – with the Kimmel algorithm being quite popular. Demosaicing has the ability to create a true colour image with interpolated values for every pixel location. However, regardless of the algorithm used it can create problems in the resulting image (similar to noise suppression). Some of these problems are viewable artifacts in an image such as zipper effects, bright dots, and blurring. When a manufacturer designs noise suppression and demosaicing algorithms they have to keep these factors in mind.

Earlier, Figure 2.19 showed that demosaicing algorithms can create artifacts that are troublesome to deal with in defect analysis. That is one of the main reasons why lossless formats such as digital RAW (e.g. CR2) are used to avoid demosaicing in other formats. A problem that arises with compressed formats like JPEG is that they further spread the artifacts created by demosaicing.

(a) TIFF | (b) JPEG 12 (low loss) | (c) JPEG 7 (medium loss) | (d) JPEG 3 (high loss)

Figure 2.21 A TIFF image of a single pixel defect and the impact of JPEG compression (taken from Chapman [18])

Figure 2.21(a) shows a lossless single pixel defect in TIFF format that has already undergone bilinear demosaicing (same as Figure 2.19(b)) and contains a noticeable 3x3 pixel artifact. Something to note is that the TIFF format is a lossless format that has undergone zero compression but has undergone demosaicing. Figures 2.21(b)-(c) show various levels of JPEG compression that further cause the artifact effect. At JPEG 12 compression, the artifact shows resemblance to the TIFF image – albeit at a lower resolution. At JPEG 7 compression it spreads into an 8x8 pixel artifact and at JPEG 3 compression it spreads into a 16x16 pixel artifact. JPEG 7 and JPEG 3 also distort the colour of the defect. One thing to point out that even though it has the lowest quality, JPEG 3 compression is the most commonly used JPEG compression. From this we can see that JPEG is an undesirable image format for defect analysis as it creates and spreads artifacts that are not in the original lossless image format.

To avoid the possibility of creating further artifacts during the conversion of RAW image data to JPEG via noise suppression and demosaicing, RAW images are used for defect analysis. JPEG images may have the defects visible at a lower value and not in their correct positions due to these methods as well. Thus, the inherent characteristic of RAW images having little to zero alteration helps identifying the exact locations of defects much easier. With this in mind, the experimental results in this paper will only leverage RAW images.

2.12 Summary

This chapter has examined the various circuit concepts that are used in the image sensor of digital cameras. There are many sensor technologies that a camera manufacturer may use. However, industry standards such as CMOS technology and more specifically the active pixel sensor have been adopted by most. This chapter has also discussed the fundamentals of Digital Cameras and Photography that need to be taken into consideration when dealing with digital images and their defects. Concepts such as the Bayer filter pattern and the Colour filter array in the context of color demosaicing as well as ISO, aperture, and exposure time in the context of photo capture were introduced.

Lastly, the importance of using RAW images in defect analysis was heavily stressed. The next three chapters will get into more thorough detail regarding Single Event Upsets seen in digital cameras as well as the experimental methods used to detect them in both DSLRs and cell phones.

Chapter 3.

Single Event Upsets in Digital Cameras

3.1. Introduction

Up to this point in the thesis we have discussed in detail the fundamentals of digital photography as well as the progression of the digital imager design over the years – particularly leading to the popularity of the current CMOS sensor technology. Some of the more common permanent and non-permanent defects have been touched on as well. The ones that are visible in CMOS imagers are known as hot pixels (permanent) and SEUs (temporal) [22]. Both hot pixels and SEU defects are caused by cosmic ray particles hitting the digital camera sensor at random times and locations. Since the research of hot pixels is a more mature space, SEU research will be the focus area from this chapter onwards as it has a lot of room to grow.

The goal of this chapter is to discuss SEUs in digital imagers. In particular, the necessity to pursue this research as well as the basic concepts that characterize SEUs. Past research provided an introduction to SEU analysis for DSLRs at modest ISO levels but lacked the ability to identify SEUs with long exposure times in cell phones at modest ISO levels and DSLRs at high ISO levels – all due to the presence of high noise. Hence, the SEU research in this thesis will focus on these missing aspects of the previous research. Within the topic of SEUs there are three types – single pixel SEUs, SEU clusters, and SEU streaks – which have different characteristics. Lastly, the experimental procedures to capture SEU dark-frame images will be shown for both DSLRs and cell phone cameras. This will serve as a precursor to the next chapter where we will dive into the algorithms that are used to supress noise and detect SEUs in digital camera sensors.

3.2. Digital Camera SEUs

SEUs are of significant interest in ICs as they occur at a higher rate than permanent defects. For many ICs, an SEU could completely ruin their intended function and sometimes in environments where faults cannot be tolerated. Take the example of ICs for data encryption. It would be undesirable if the system were to encrypt or decrypt the data incorrectly due to an SEU. Another technology that has low tolerance for these faults is control systems. Since these are regular integrated circuits, it is very hard to study and explore what actually happens during the SEUs. There are methods to create models and artificial evidence, but very few ways of exploring real SEUs as there is no way of seeing into them. Hence, SEUs are often difficult to observe for most types of ICs as they are temporary events that get lost in the system. In contrast, digital camera pixels are also susceptible to SEUs and have the ability to retain the deposited charge – making them easier to observe, record, and analyze. When dark-frame images are captured in a series, they inherently contain the readout values of the camera sensor's pixels. This sensor data can be later used for SEU analysis which helps identify the location of SEUs, the rate of SEU occurrence, and various characteristics of the SEU charge ball. The physics community had initially proposed using digital cameras for the purposes of detecting cosmic rays. However, since SEUs are a result of cosmic radiation we decided to use digital cameras to detect SEUs. With this in mind, our research leverages digital cameras to develop an understanding of SEUs which is almost impossible to do with traditional ICs.

When this current research began, past research focused on permanent defects in digital cameras – most commonly known as hot pixels. As mentioned in Chapter 1, these are the defects that present themselves shortly after the camera sensor is manufactured and grow throughout the lifetime of the camera. They appear as high intensity spots in the images and persist in a series of images. Their intensities tend to increase if either the ISO level or exposure time increase. Past research focused on different ways to detect hot pixels, understanding their creation process and their rate of occurrence. When it came to SEUs, much of the existing research scratched the surface on this topic. As a result, our research focuses on areas that were not addressed before. These are identifying SEUs in DSLRs at high ISO levels and cell phones at modest ISO levels. In additional the relationship of elevation to the SEU occurrence rate will be

explored. Before progressing onto these more advanced aspects of the research, we will begin by going over the basics of SEUs in digital cameras.

3.2.1. Basic Concept of Digital Camera SEUs

A single event upset in a digital imager occurs when cosmic particles collide with an imager sensor pixel and gets recorded during image capture. Cosmic charge deposits energy which the camera sensor records in the image. When the cosmic particle comes into contact with the silicon it loses the energy to the silicon and the energy creates an electron hole pair. Keep in mind the electrons are accepted by a pixel the same way photons are accepted by a pixel. This is important because it creates a ball of charge that upsets the IC – in this case the camera sensor. As previously mentioned, camera sensors are the only ICs that tell us information such as the charge distribution, area of charge, and the occurrence rate.

Unlike hot pixels which are permanent, single event upsets are temporal. They do not appear in previous and subsequent images that are captured. To help us understand, Figures 3.1 through Figure 3.4 illustrate examples of hot pixels and SEUs. The example in Figure 3.1 below depicts a diagrammatic view of a zoomed-in area of 5x5 pixels containing a permanent hot pixel that is seen in a sequence of four images. The most important characteristic is the hot pixel persists through all four images – that is images *j, j+1, j+2* and *j+3*.

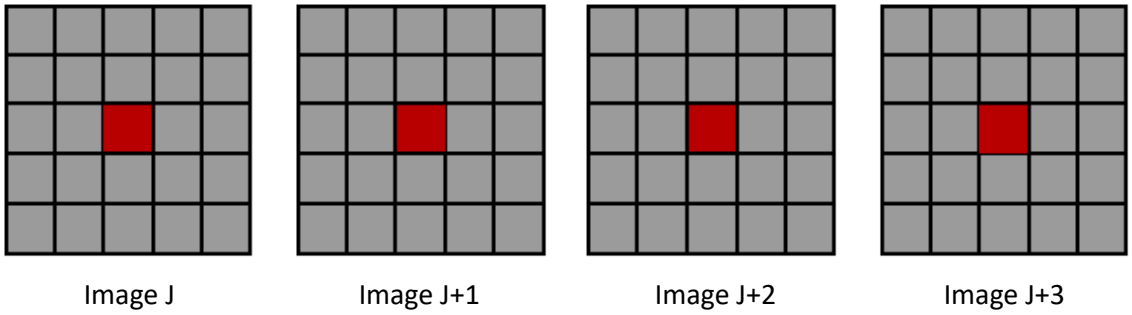


Figure 3.1 Example of a hot pixel

The example in Figure 3.2 below depicts a diagrammatic view of a zoomed-in area of 5x5 pixels containing an SEU that appears in a single image of a sequence of four images. The most important characteristic is the SEU appears only once throughout all four images – that is images *j, j+1, j+2 and j+3.* This particular type of SEU is contained in a single pixel.

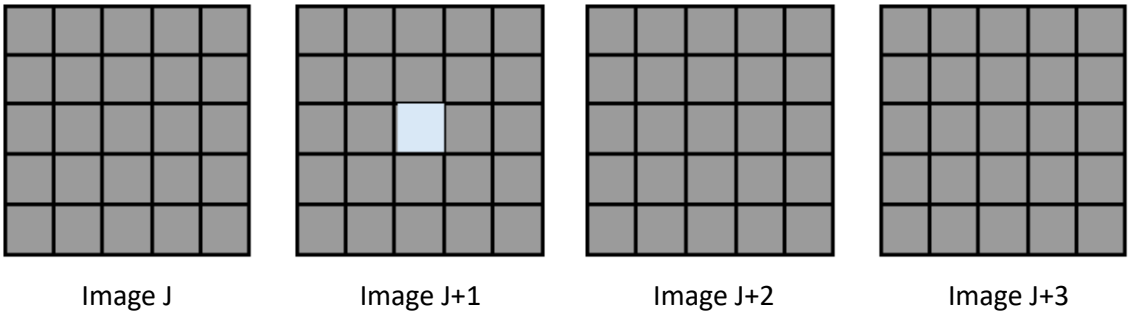


Figure 3.2 Example of an SEU contained in a single pixel

The example in Figure 3.3 below depicts a diagrammatic view of a zoomed-in area of 5x5 pixels containing an SEU that appears in a single image of a sequence of four images. This is a different type of SEU than the one illustrated in Figure 3.2. This particular type of SEU covers multiple pixels instead of a single pixel. Similarly, to a single pixel SEU, the most important characteristic of the SEU cluster is that it appears only once throughout all four images – that is images *j, j+1, j+2 and j+3*.

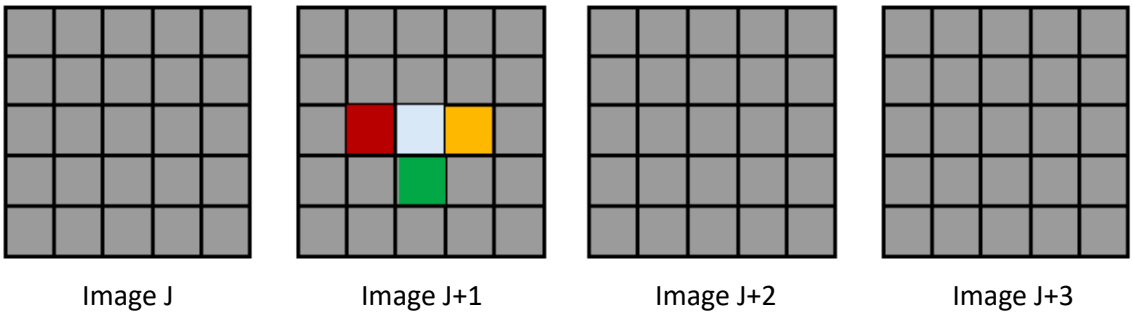


Figure 3.3 Example of an SEU cluster spanning multiple pixels

The example in Figure 3.4 below depicts a diagrammatic view of a zoomed-in area of 5x5 pixels containing an SEU that appears in a single image of a sequence of four images.

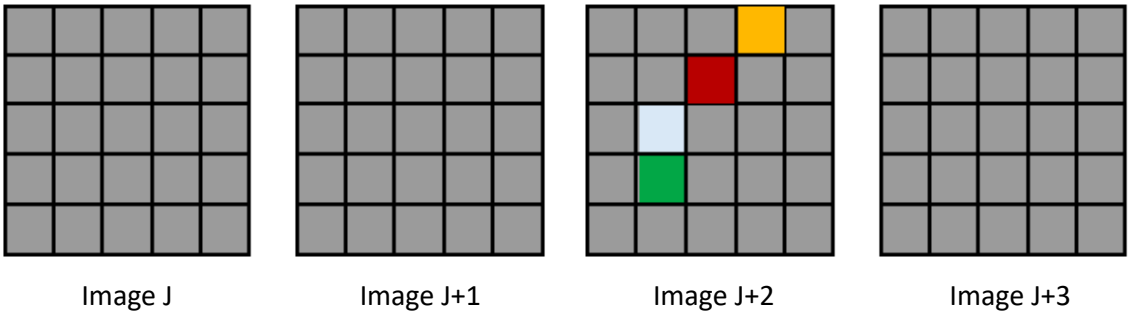


Figure 3.4 Example of an SEU streak spanning multiple pixels

This is a different type of SEU than the ones illustrated in Figures 3.2 and 3.3. This particular type of SEU is not clumped together like a cluster but covers multiple pixels in a streak pattern. Similarly, to the other types of SEUs, the most important characteristic to make note of the SEU streak is that it appears only once throughout all four images – that is images *j, j+1, j+2 and j+3*.

It is important to note that Figures 3.1 through 3.4 are diagrammatic examples of hot pixels and SEUs, not their actual representation in digital camera images. Likewise, the colours used in the figures are completely arbitrary. The main details to pay attention to are the respective high intensity defects (hot pixels or single event upsets) which contrast with their background. In the SEU cases, a fixed ISO level, exposure time, and dark-room environment are all necessary to capture dark-frame images. A unique characteristic of an SEU is that they appear random in both time and location – leaving a very close to zero probability of reappearing in adjacent images. Regardless of its particular colour, an SEU will randomly appear as a bright pixel (or group of bright pixels) at a given pixel location exactly only once in a sequence of adjacent dark-frame images. It will not appear at that respective pixel location in the few images prior and the few images following. If found appearing before and after, that potential SEU could easily be noise or a hot pixel – these false positives will be discussed later. In addition, to get a visualization of an actual SEU in a dark-frame image, actual snapshots of SEUs from real-life digital camera dark-frame images will be shown shortly. A lot of times there is an

argument to be made that SEUs generally do not influence photography for the most part. In non-dark-frame images many consider an SEU as extra image noise as opposed to a defect. Hence, you will find many ignoring it, except for cases where high radiation levels exist such as photography in outer space.

Much of the research in this field of study has been around regular ICs. This is because SEUs in these environments can cause issues with the overall operation of the circuit – creating unpredictable outcomes [22, 47, 21]. But an undesirable pattern in all of these studies is that they have synthetically created SEUs by the means of laser energy or nuclear radiation. On a more related note, past SEU research has identified digital cameras as instruments that can be used for cosmic ray detection [21, 49]. However, only recently have researchers began to identify the cell phone as a useful digital camera to collect cosmic ray information. In our past research, it has been shown that the active pixel sensor (APS) has the capability of being able to store and integrate all of the cosmic charge that enters the pixel diode. As SEUs appear as high intensity pixels in a dark-frame image, cameras are quite accurate at recording SEU values. They also provide us with distributions in relation to noise and charge values. Overall, the plan of this research is to continue on from past research and understand the degradation in images caused by these SEUs and discuss possibilities for camera makers to avoid them appearing in the image capturing process.

3.2.2. Types of Digital Camera SEUs

Figures 3.2 through 3.4 introduced us to the idea of the different types of SEUs – that is, a single pixel SEU (Figure 3.2), an SEU cluster (Figure 3.3) and an SEU streak (Figure 3.4). Figure 3.5 diagrammatically illustrates the three types of SEU again. In this case they all appear close to each other but as mentioned will not always be.

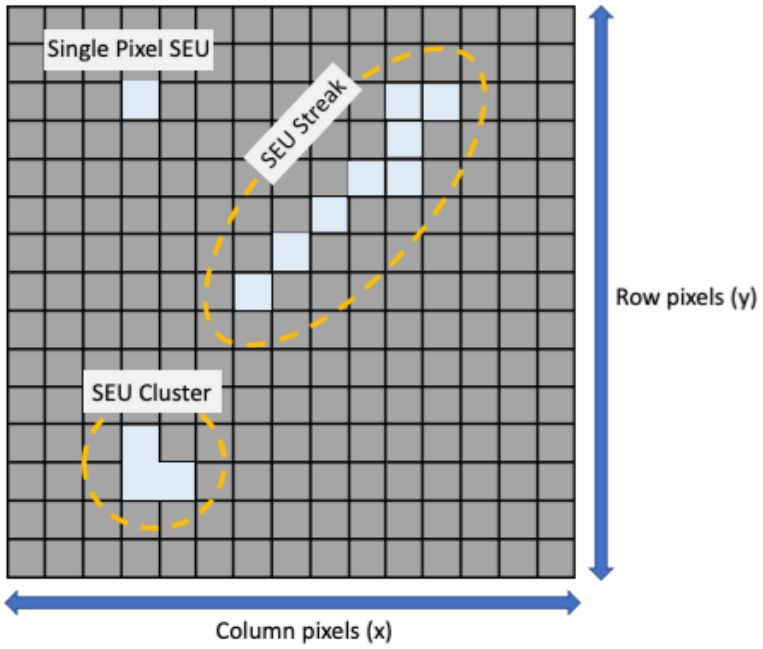


Figure 3.5 Three different types of SEUs

In Figure 3.5 there are three different types of SEUs that are shown – namely a single pixel SEU in the top left of the figure, an SEU streak of pixels in the top right of the figure, and an SEU cluster of pixels in the bottom left of the figure.

As mentioned earlier, cosmic energy collides with a pixel sensor and gets deposited. This event causes a build-up in the circuit's charge and results in an SEU. In this section we will spend some time giving more detail regarding the different types of SEUs. First, we will start by discussing the simplest case of the three, that being a single pixel SEU. As the name suggest, the important characteristic of this type of SEU is that the event upset caused by the cosmic particles is confined to a single pixel. Unlike the other types of SEUs, the charge in a single pixel SEU will not bleed over to neighboring pixels. The next type of SEU that was shown at the bottom left of Figure 3.5 is an SEU cluster. Unlike the single pixel SEU, this type of SEU is made up of multiple adjacent SEU pixels – forming a 'cluster'. The shape of an SEU cluster is not always identical or even the same orientation every time. However, for the most part it will follow the shape of an 'L' or a 'T' – possibly rotated or flipped. Another characteristic to understand is that the cosmic charge deposits into one pixel and bleeds over to the adjacent pixels. Hence, the SEU cluster will also have its charge value spread over all of its pixels instead of being confined to a single pixel. This characteristic is also true for the last type of SEU – an SEU

streak (sometimes referred to as tracks or worms in SEU literature Groom [45]). Figures 3.6 through 3.8 display three clear examples of SEU streaks.

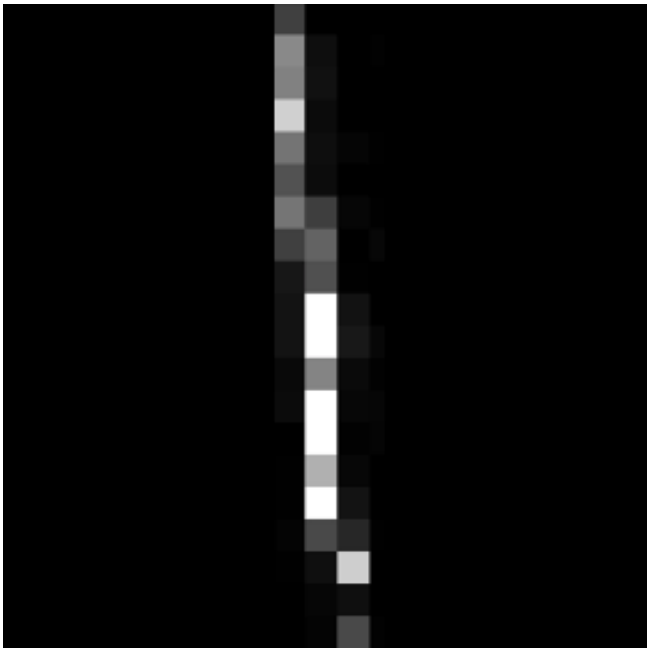


Figure 3.6 SEU Streak identified during SEU Analysis (Example 1 – 20x3 pixels, 6.41 µm pixels)

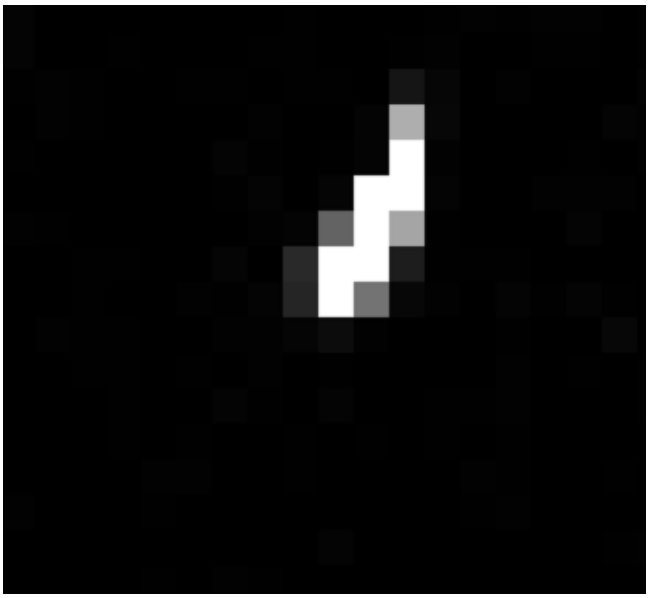


Figure 3.7 SEU Streak identified during SEU Analysis (Example 2 – 7x4 pixels, 6.41 µm pixels)

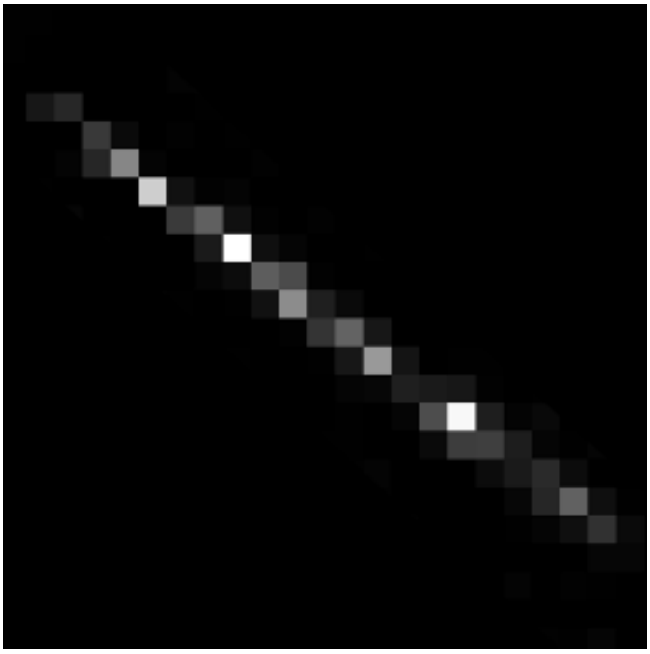


Figure 3.8 SEU Streak identified during SEU Analysis (Example 3 – 16 x 21 pixels, 6.41 µm pixels)

The three examples of SEU streaks in Figures 3.6 through 3.8 show that streaks do not all appear in the same sizes and orientations. In fact, most of the time they look quite different from one another. The SEU streak in Figure 3.6 portrays a generally vertical streak that appears to have stronger intensity towards the bottom. The next streak shown in Figure 3.7 appears very bright and follows the very definition by resembling a lightning streak. The bright intensity depicts the high charge value being fairly equally distributed throughout the streak's pixels. Lastly, the longest and most fascinating streak of the three happens to be the streak shown in Figure 3.8. This particular streak starts at one corner of the zoomed in area of the image and proceeds to the extreme opposite corner along a diagonal trajectory. You will notice that certain areas of the streak have a higher intensity than the others. For example, the streak is fairly dull in nature except for a few notable areas – that is the bright spots in the streak. These areas have a higher distribution of the SEU charge than the rest of the pixels in the SEU. This also means they contribute to a higher percentage of the total SEU streak charge than the other pixels that are also part of the SEU streak. Streaks essentially tell us that charge can be spread over a line of area. This means that multiple ICs can be affected by a single cosmic particle. The effect of streaks can be understood similarly to a traditional cloud chamber. Cosmic rays shoot through the cloud chamber and leave behind charge – this causes condensation to

accumulate as water vapor. The same behaviour can be seen in the case of digital imagers. Cosmic rays leave back packets of energy which result in SEU streaks that affect multiple pixels (ICs). In our research we have found that streaks account for less than 5% of the total SEUs in an image dataset.

It is fortunate that digital camera sensors have this ability to record single event upsets through dark-frame images of medium to high sensitivity levels and long exposure times. When an event occurs, the charge value is recorded in the pixel's location in the image and the information is available after the image capture. As mentioned earlier, traditional integrated circuits face difficulties recording single event upsets as the charge area and intensity get lost in the circuit itself. Our past research and detection algorithms had already been able to detect and analyze the three types of SEUs. Our focus was to be able to detect and analyze them in environments that were previously not possible. These are when dealing with DSLRs at high ISO levels and cell phone cameras at modest ISO levels. At these ISOs, the SEUs are generally weaker and often get lost with the noise. We will discuss this phenomenon in more detail throughout the remainder of this thesis. In the next section we will discuss the process of experimentally detecting SEUs, prior to the next chapter where we will discuss the different SEU detection algorithms.

3.3. Capturing Dark-frame Images for SEU Analysis

Up until this point in this chapter we have discussed some of the fundamental concepts of SEUs. In this section we will look at the experimental procedure involved in capturing dark-frame images that can be used to detect SEUs. Dark-frame images of digital cameras have not only the ability to record the existence of SEUs but also their respective deposited charge value. Hence, with large datasets of time-lapsed dark-frame images, pixel analysis allows us to calculate the intensity and energy distribution of deposited cosmic ray charges, the energy vs occurrence rate, the total area of the charge ball, and the correlation between the number of SEUs and the elevation at which the image has been taken.

3.3.1. General SEU Experimental Setup with DSLRs

Before getting into the various detection algorithms it is paramount to understand the experimental procedure. In the past, the focus had been on capturing dark-frame images with DSLRs for SEU analysis. The SEU analysis tools used in past research lacked the ability to detect SEUs in images with high ISO levels or images with long exposures. This is because increasing the exposure time or the ISO level generates more thermal noise in the camera's sensor. We will understand this topic of noise in more detail in the next chapter. In comparison, our focus has been on capturing dark-frame images with both DSLRs and cell phone cameras for SEU analysis. More specifically taking dark-frame images with DSLRs at higher ISO levels as well as with cell phone cameras at modest ISO levels. Previously in our research it had been a struggle to separate this noise from the actual SEUs, but the current detection algorithms enable us to make such separations within higher ISO images.

The max exposure times used for DSLRs and cell phone cameras were 30 seconds and 4 seconds respectively. At exposure times longer than these, high amounts of noise would make the dark-frame images unusable for the SEU analysis tools. The highest tolerable exposure time is different when comparing camera models and is also dependent on the ISO setting used. Increasing the ISO also amplifies the collected charge. ISOs that are not used are levels greater than ISO 12800 in DSLRs or greater than ISO 800 in cell phones. Our initial research started off with capturing dark-frame images with DSLRs in digital RAW format. As mentioned in the previous chapter, the images taken in digital RAW format are free of demosaicing, compression, and additional noise suppression which is very beneficial when analyzing SEUs. Compressed formats such as JPEG remove important details and alter the recorded pixel data of the image [46]. Hence, dark-frame images are taken in digital RAW format to give the best results as it allows the SEU analysis tools to look for SEUs with a dataset of generally unaltered pixel values. The dark-frame images provide benefit to the analysis tools themselves by presenting the stored charge data collected during the light exposure period. Hence, to leverage the benefit of long exposure time, a dataset was composed of dark-frame images with a fixed ISO level and exposure time.

To understand the differences between datasets, different ISO levels and exposure times were used to capture multiple datasets for a given camera. Looking at the datasets that were collected, ISO levels ranged from ISO 100 to ISO 25600 for DSLRs and from ISO 100 to ISO 800 for cell phone cameras. As mentioned earlier, the max exposure times for DSLRs and cell phone cameras were 30 seconds and 4 seconds respectively. By this, dark-frame images in an image dataset record both the temporal and spatial occurrences of all SEU events that have occurred during that exposure period. Since SEUs appear during one exposure period and disappear in the next exposure period, having set exposure times in a dataset gives the SEU analysis tools the ability to see that behavior. Looking at the process of capturing dark-frame images for SEU analysis is quite different than the process for capturing dark-frame images for hot pixel analysis. The hot pixel image capturing process involves linear fitting after taking dark-frame images at a fixed ISO level and increasing exposure times [46]. Since this research does not focus on hot pixel detection or analysis, the experimental method for hot pixels will not be discussed.

First, we will take a look at the general environmental setup for DSLRs and get into the environmental setup for cell phone cameras in the next section. Cell phone cameras have very specialized environment conditions that do not apply to general digital cameras such as DSLRs. The first prerequisite for capturing these images is for the camera to be in a darkroom used for film development. Alternatively, any ‘pitch-black’ room or box that blocks out all light may be used. The environment that was used in this research ensured that no outside light entered the already ‘black’ room. Hence, this guaranteed that no light would enter the imaging array of the camera. By adhering to this strict, no-light environment the only details that an image will record (other than their black pixel values) are SEUs, Hot Pixels and noise. Since we are only interested in SEUs, the detection algorithms would be able to later separate out false positives such as Hot Pixels and noise from the desired SEUs.

In order for a pixel to be considered a potential SEU, the pixel will have to show no signs of an SEU in the first image, the existence of an SEU in the second image, and return to showing no signs of an SEU in the third image. Figure 3.9 illustrates the process to better understand these stages in SEU detection. This figure illustrates the nature of an SEU event in respect to the pixel readout at its location in the before and after images. At this point it is beneficial to set the numbering convention for the images that are in the

three-image sequence to make the sequence more intuitive. The middle image will be classified as the current image as it is in between two images – the previous and the next images. The previous image will be defined as image *j*, the current image will be defined as image *j+1*, and the next image will be defined as image *j+2*.

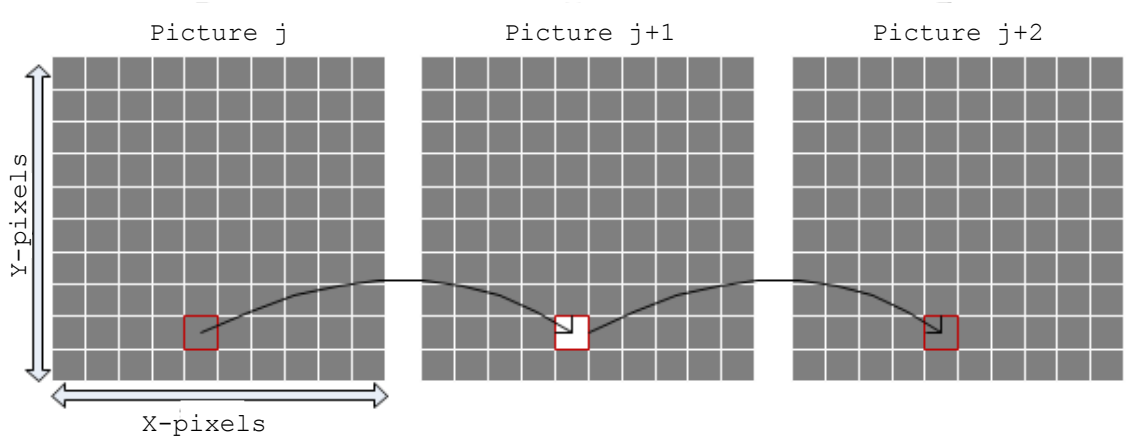


Figure 3.9 The SEU Detection Algorithm demonstrating an SEU that was detected in image 'j+1' and not present in images 'j' and 'j+2' (Taken from Chapman [47])

The stages in Figure 3.9 are meant to show the appearance of an SEU in the series of images *j, j+1, j+2*. In other words, the SEU will be detected at a given pixel location in image *j+1* and not in image *j* and image *j+2*. The same pixel location that was identified as an SEU in image *j+1* will be a dark pixel (denoting a very small signal value) in image *j* and image *j+2*. The pixel location in image *j+1* that is identified as an SEU will appear as a very bright pixel (denoting the deposited cosmic charge value).

A single SEU dataset is composed of dark-frame images that have the exact same gain level (ISO) setting and exposure time duration. In the early stages of our research, the dark-frame image datasets contained roughly 200 images. However, as our SEU detection software improved – 200 images were not enough. To fully benefit from the higher accuracy of the pixel address distribution method (explained in the next chapter), each dataset contains around 1000 images. Between each time lapsed image is a user-defined wait interval for the image sensor to cool down between shots. Neglecting to follow this step will lead to the build-up of high amounts of thermal heat in the CMOS pixel sensor as well as other circuits within the camera. This results in an image having a lot of thermal noise. These amounts of noise make the dark-frame images unusable for SEU analysis

and can be avoided entirely by inserting wait times between the shots. Therefore, the total time being used for a single image in an SEU dataset is equal to the exposure time plus the wait time interval. From this we can understand that an entire SEU dataset will take a very long time to be captured. For example, if the exposure time is set to 30 seconds and the standard wait interval for the camera sensor to rest is 60 seconds, the total time to complete one dark-frame image is 90 seconds. Figure 3.10 clearly illustrates this scenario.

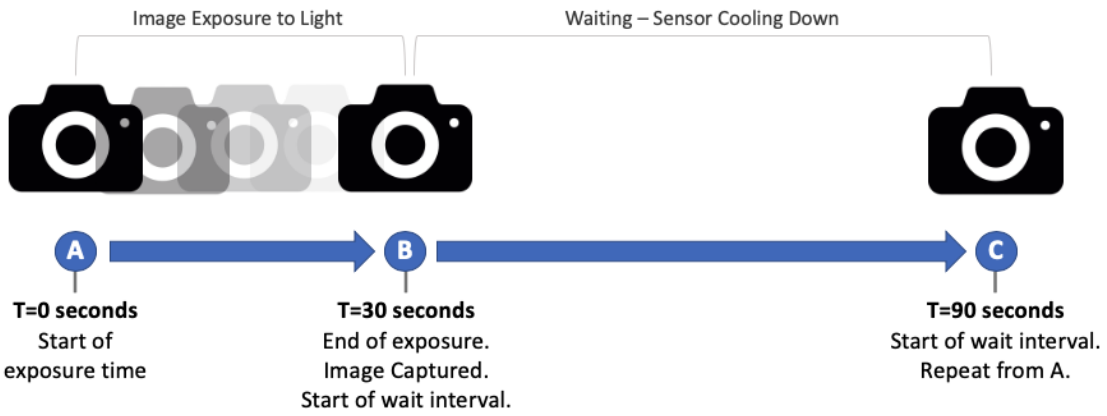


Figure 3.10 Timeline of capturing a single Dark-Frame image for SEU detection with a DSLR. (Exposure time=30 seconds, Waiting time=60 seconds)

As Figure 3.10 shows, the total time spent is 90 seconds per image in the dataset. In total, capturing 1000 images will take over 1500 minutes (or 25 hours) to complete. Earlier in our research, the memory card capacity was an issue and required copying over the images after a few hundred images were stored on the memory card. However, moving to larger memory cards solved this issue, allowing for an entire image dataset to be stored on a single memory card. Since most DSLR models use camera batteries that do not have very long battery life, changing batteries multiple times was required to capture an entire image dataset of 1000 images. Towards the end of the research, camera power packs connecting to power outlets were used. This would allow a camera to capture images for 25 hours uninterrupted.

The main objective of capturing these dark-frame images is to obtain precise temporal readings of the pixels in the camera sensor during a specific time period as well as at various camera settings, i.e., gain level (ISO) and exposure time. In order to capture

1000 dark-frame images in every image dataset, a robust image capturing process had to be put in place. Manually taking thousands of dark-frame images would be prone to errors and almost impossible. Hence, the process of capturing all of the dark-frame images in an image dataset is fully automated – requiring no manual human involvement once started. Image settings are initially configured with fixed time intervals (both exposure time per image and waiting time between images) and are controlled by an external computer software tool. This control tool is most commonly known in photography as an intervalometer. Typically, intervalometers are used to take time-lapsed photos of outside scenes but can also be used for the purposes of taking dark-frame images for SEU analysis. There are many types of intervalometers available today, but two types were used in this research:

1. Software intervalometer program – This particular type of intervalometer required no other physical hardware other than the DSLR and a laptop for the software to run. The intervalometer interfaces from the computer to the DSLR via a wired connection or wireless communication. This is a very commonly used intervalometer method as it is readily available to any DSLR photographer who owns a laptop – and often times is free. The main benefit of this intervalometer type is that it does not require any other physical devices and can save images directly to a computer. Saving images onto a computer instead of a DSLR reduces the battery usage as well as thermal noise produced by the camera as it is doing less work.
2. Hardware intervalometer – This particular type of intervalometer is a hardware device that can remote control a DSLR by connecting via a wired connection or through wireless communication. Depending on the make and model of the hardware intervalometer, the configuration options differ. However, most common configuration options are exposure time, wait time, and lastly the number of iterations to capture. A significant benefit to this type of intervalometer is that it is relatively inexpensive and does not require any other devices such as a laptop or a cell phone.

There are other types of intervalometers such as built in DSLR intervalometers that essentially integrate the functionality of a hardware or software intervalometer into the DSLR itself. Some find this appealing as everything is done on the DSLR. However, since this type of intervalometer was not used in the research it will not be discussed. Of the two intervalometers used in this research, the software intervalometer program is the most common as it provides the mentioned advantages of consuming less camera battery

power and having the ability to store RAW image files on an external computer or hard drive.

Now that we are aware of the general experimental setup and the process involved in capturing dark-frame images with DSLR cameras for SEU analysis we may proceed to cell phone cameras. However, as mentioned earlier, the process involved in taking dark-frame images with cell phone cameras is slightly different. The next section will explain their experimental setup.

3.3.2. SEU Experimental Setup with Cell Phones

The current trend in the IC industry has been shrinking the overall size of the transistor in CMOS circuits. Therefore, it is paramount in our research to study the topic of SEU detection in cell phone camera sensors. More specifically, cell phone manufacturers have specialized in including inexpensive small cameras with the largest number of megapixels possible. As a result, manufacturers have made the quality of everyday photography rather similar to that of DSLR photography. When comparing the price of the cell phone camera to a DSLR you will see that the cell phone camera system will cost around 10 dollars in comparison to a DSLR costing anywhere from 500 dollars to 5000 dollars. Presently, the size of cell phone camera pixels ranges from 1.4 μm down to 1.2 μm and sensor areas are generally around 24 mm^2. In our most current research, the cell phone camera pixel sizes were generally in that range. A disadvantage of cell phone cameras is that they generate a substantial amount of noise in comparison to DSLR sensors. One of the main reasons for this is the absence of high-end noise suppression algorithms that are included in DSLR cameras. Computationally these algorithms are quite expensive, hence, cannot be used in cell phone cameras.

For the longest time cell phones did not support saving images in digital RAW format –some still do not support it. Ever since the release of Android version 7 (Nougat) in 2017, camera applications on Android cell phones have had the ability to support digital RAW images. Unlike Android smartphones, there has not been a viable option to capture RAW images on iPhones. As mentioned earlier, digital RAW format is essential for

detecting and analyzing SEUs in any camera. However, this format in cell phone comes with its drawbacks. Cell phone cameras have been optimized for JPEG format – hence, they do not include some of the optimizations in digital RAW format.

Overall there are two major drawbacks when dealing with images taken with cell phone cameras. Firstly, the highest exposure time that we are capable of setting before the image starts to contain too much noise is 4 seconds. This is substantially lower in comparison to DSLRs which are capable of reaching exposure times of 30 seconds or longer. Longer exposure times give the ability to detect SEUs with lower rates. Secondly, the maximum gain level (ISO) of cell phone cameras (ISO 800) is again quite lower in comparison to DSLRs that are capable of ISO 25600 in some cases. Historically, looking at when the DSLR maximum was at ISO 800, one would have to look back to around 18 years ago. These two drawbacks clearly show that cell phone cameras are somewhat inferior to DSLRs.

Another point that needs to be addressed is that there are many other electronic components and software processes operating in the cell phone that contribute to the overall temperature of the cell phone. Unlike DSLRs, cell phones are used for multiple purposes as well as have many other applications and services running simultaneously. Hence, in order to reduce the amount of noise seen in cell phone dark-frame images, regulating their temperature is very essential. DSLRs do not have this issue as they are only used to capture images and do not have any other non-camera applications and services running in parallel. There are only two times when thermal heat is dissipated in a DSLR sensor. This first being when the image has been captured and read out. The second being when the image is being stored. Both of these operations do not take much time. Therefore, in the case of cell phone cameras, it is all the more important to time space the images being captured in an SEU dataset. Like in the case of DSLRs, this will help reduce the thermal noise that is generated by the cell phone cameras during image capture.

During our research we were able to create a unique method to effectively capture SEU dark-frame images with cell phone cameras. As mentioned earlier, cell phones are prone to substantial thermal heat generated from the camera sensor, communication and other background processes. To alleviate this, we perform a few actions prior to capturing the dark-frame images. Firstly, we enable ‘airplane mode’ on the cell phone to turn off and

block out any external communication on the cell phone. Secondly, we turned the brightness of the cell phone screen to the lowest brightness setting possible with the intent to eliminate some thermal heat generated from its OLED (Organic Light-Emitting Diode) display as well as conserve battery power. Generally speaking, a lot of the cell phone's battery gets drained due to its screen being turned on – hence, the need to turn it low since turning it off is not an option.

On a related note, the cell phone's screen as well as the lens of the camera were covered to block out all light during image capture. Refusing to do this would cause light to enter the camera sensor during image capture and affect the dark-frame image. Lastly, to ensure that the cell phone is cooled down to a lower temperature we place it in a refrigerator (on a metal portion of the frame) that has an inside temperature of approximately 4°C. This step is crucial as it essentially lowers the internal temperature of the cell phone by approximately 16°C – further lowering the thermal noise level during image capture. This step was not a practice done during image capture with DSLRs as some preliminary experiments during the beginning stages of the research provided evidence that it had no impact on the noise levels. Setting the refrigerator's temperature lower than 4°C would bring it closer to that of a freezer which is undesirable.

If you recall, with DSLRs we leveraged either a software intervalometer program or a physical hardware intervalometer to capture time spaced images. Unfortunately, these options for the most part do not exist for cell phones. In seldom cases where the cell phone applications have intervalometers, the support for digital RAW format is not included. Fortunately, we developed an innovative work-around solution that mimics the behaviour of an intervalometer for Android camera applications. As most are familiar, Android operating systems as well as most of the other cell phone operating systems support voice control. This gives them the ability to listen to human voice commands and capture images with their standard camera applications. Knowing this functionality, we developed a short Python script that outputs the word 'cheese' as many times as the user specifies in the script.

The Python script has the following workflow:

1. Imports the necessary Python libraries.
2. Imports the WAV file recording of a human voice saying 'Cheese.'
3. Runs a loop to repeat as many iterations as specified. In the loop:

a. Plays the cheese sound once per iteration.
b. Waits a duration equaling the exposure time plus 30 seconds – enough time for the sensor to cool down.
    i. Example 1: 32 seconds if the exposure time is 2 seconds.
    ii. Example 2: 34 seconds if the exposure time is 4 seconds.

4. Repeat (3) until the completion of the total number of iterations

Figure 3.11 below provides a code snippet of the short Python program that provides this intervalometer functionality.

```
## Intervalometer program written in Python
## Written by Rohan Thomas

# – Start of script – #

import simpleaudio
import time

# 'Cheese' wav file (less than one second long)
filename = '/Users/rmt/cheese/Intervalometer_Python/trigger.wav'
sound = simpleaudio.WaveObject.from_wave_file(filename)

# For loop to repeat process 1000 times
for num in range(0,1000):
            play_var = sound.play() # Play sound
            play_var.wait_done()
            time.sleep(34) # wait 34s before next iteration

# – End of script – #
```

Figure 3.11 Code snippet of the Python intervalometer program

Running the code in the above figure plays the word cheese every 34 seconds for 1000 images. The end result would provide us with a single dataset containing 1000 cell phone dark-frame image. To cause the cell phone camera to properly capture dark-frame images in a refrigerator, we placed an audio speaker in the same refrigerator. It would trigger the cell phone camera every time it outputs the word 'cheese.' The cell phone camera application would be preconfigured with the desired sensitivity level (ISO) and exposure time prior to inserting it into the refrigerator – it would not be changed by the Python program. Figure 3.12 below clearly illustrates this configuration that was just described – 4 seconds of exposure time and 30 seconds of waiting intervals. One major advantage of this intervalometer method is that it can trigger multiple cell phone cameras at a time. All that needs to be satisfied is that triggering speaker that outputs 'cheese'

needs to be in the same refrigerator as all of the cell phones. Any single one of the DSLR intervalometer methods that were mentioned earlier could at most trigger one DSLR at a time.

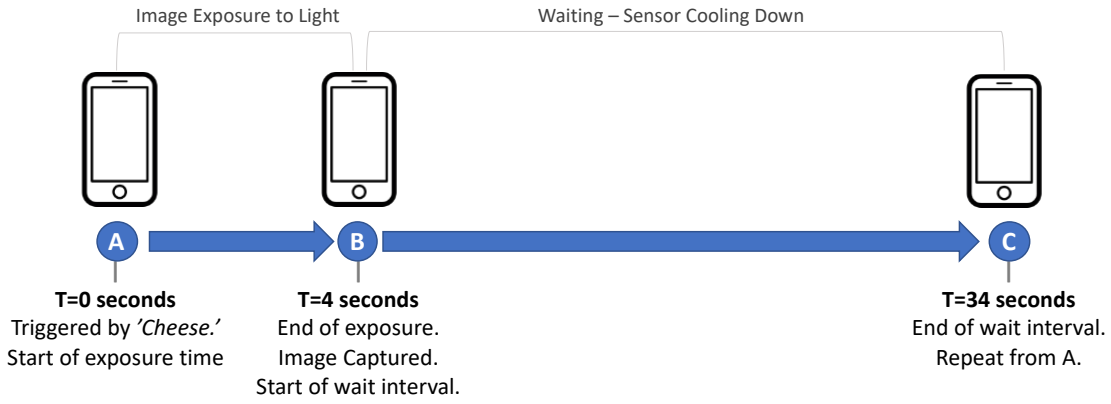


Figure 3.12 Timeline of capturing a single Dark-Frame image for SEU detection with a Cell Phone camera. (Exposure time=4 seconds, Waiting time=30 seconds)

An important point to mention is regarding the storage of images when using a cell phone to capture them. Almost all present-day cell phones ship with gigabytes of dedicated flash storage. However, most of this storage space gets consumed with its respective flavour of UNIX operating system as well as services and applications. This in turn leaves very little to no room for storing digital RAW image files. To overcome this – since Android operating systems support expandable storage – we insert a Micro-SD card to store the captured dark-frame images. The total available storage for the dark-field image datasets is limited by the capacity of the expandable storage card. In our experiments a single 100 gigabyte Micro-SD card had the capacity of supporting over 1000 images. Hence, this gives us the ability to store an entire 1000 image dataset.

Overall, this provides us with a robust method to capture SEU dark-frame images for cell phone cameras. Studying cell phone cameras is important for detecting SEUs and for other future applications. As cell phone sensors become smaller over time, they store less deposited charge but on the other hand become more portable to travel around with humans as well as act as cosmic ray detectors. This is related to a concept that has been discussed by the physics community where cell phone cameras distributed around the world would act as cosmic ray detectors since a significant portion of the world's population owns a cell phone. Time will indicate to us the adoption of this idea throughout the world.

As mentioned earlier, dark-frame images with long exposure times can record the existence of SEUs and their respective deposited charge values. A characteristic that is associated with long exposure times (e.g. 30 seconds) is a high amount of noise. This further increases in cell phones cameras as they have much more noise than DSLRs even at modest ISOs. The same issue is noticed with DSLRs at high ISO settings. As a result, the next chapter will walk through the different SEU detection algorithms that try to tackle the noise problem in different ways. The most recent method, the Pixel address distribution method uses distributed noise reduction to properly discard image noise and identify SEUs in cell phone cameras at modest ISO levels and DSLRs at high ISO levels.

3.4. Summary

This chapter has discussed many theoretical concepts related to Single Event Upsets in digital cameras – their different forms and how they contrast to hot pixels. The experimental process for capturing dark-frame images with DSLRs and cell phone cameras was discussed. As prior research was rather preliminary, the motivation of this research is to analyze SEUs in DSLRs at high ISO levels (greater than ISO 1600) and cell phone cameras at modest ISO levels (lower than ISO 1600). The next chapter will present the various SEU detection algorithms that have been developed over time as well as methods to overcome high levels of noise.

Chapter 4.

SEU Detection Algorithms

4.1. Overview

In the previous chapter the theoretical concepts related to Single Event Upsets in digital cameras were discussed – namely the different types and their respective characteristics. Additionally, the general experimental procedures for capturing dark-frame images were discussed for both DSLRs and cell phone cameras. The goal of this chapter is to discuss the research advancements that have been made in respect to SEU detection in digital camera sensors – including newer methods to identify weaker SEUs. We will look at DSLRs at higher ISO levels and cell phone cameras at modest ISO levels – both resulting in high amounts of noise. The newer methods have been optimized to differentiate SEUs from false positives such as noise and hot pixels. The topics will include a discussion on noise analysis in SEU images and different SEU detection algorithms that have been used. The three algorithms that we have developed in this research are (from oldest to newest):

- **The Threshold Method:** Identifies an SEU at a pixel location *k* using a user defined threshold value.
- **The Local Area Distribution Method:** Identifies an SEU at a pixel location *k* using a noise threshold value calculated by the local area noise in a single dark-frame image.
- **The Pixel Address Distribution Method:** Identifies an SEU at a pixel location *k* using the distributed mean and standard deviation of *k* over 1000 dark-frame images.

The most accurate method that is currently being used is the Pixel Address distribution method. The experimental results of each of the methods will be compared in detail towards the end of the chapter.

4.2. The Threshold Method

Over the period of our research we have explored different SEU analysis techniques – the first of which is the Threshold Method [48]. In the early development stages, the Python intervalometer script (recall from Chapter 3) had not been written. Hence, this method was initially only used for DSLR images. In addition, a single image dataset contained a maximum of 200 dark-frame images. The benefit of capturing 1000 images in a single dark-frame image dataset was not learned until more recently. As mentioned previously, DSLR sensors have larger pixels which are great for recording pixel values with a certain level of noise reduction in digital RAW format. The threshold method is a software algorithm implemented in MATLAB (called seu_analysis_threshold.m) that begins by reading in the dark-frame images of an SEU image dataset one by one. Recall, all images in a single image dataset are of a fixed ISO and exposure time. The tool would accept a pre-defined threshold value and always store at most three images at a time in order to compare pixel values of the previous, current, and next images (recall from Figure 3.9). In order to properly identify the presence of SEUs (whether a single pixel, clusters, or streaks) at a pixel location *k*, the software would perform the following steps:

1. Check for an increase in pixel intensity at pixel location *k* by subtracting the previous image *j* from the current image *j+1*. Identify whether the resulting difference is larger than the predefined threshold.

2. Check for a decrease in pixel intensity at pixel location *k* by subtracting the next image *j+2* from the current image *j+1*. Identify whether the resulting difference is larger than the predefined threshold.

3. Check that both (1) and (2) are satisfied. If so, the pixel location *k* in the current image *j+1* is classified as an SEU. Equation 4.1 shows this logically. If only one of (1) and (2) are satisfied, reject pixel location *k* in current image *j+1*.

$$\left(\left(Image_{j+1}(x_k, y_k) - Image_j(x_k, y_k)\right) > Threshold\right)$$

$$\wedge$$

$$\left(\left(Image_{j+1}(x_k, y_k) - Image_{j+2}(x_k, y_k)\right) > Threshold\right)$$

$$\Longrightarrow \; Image_{j+1}(x_k, y_k) \text{ is an SEU} \tag{4.1}$$

Where x_k, y_krepresents pixel location k

Since this algorithm looks at the value at pixel location *k* and compares its value in the current image to the same location in the previous and next images, it should theoretically not classify a hot pixel as an SEU event. Hot pixels by definition should appear at a given pixel location in every image – that is every pixel location *k* in images *j*, *j+1*, and *j+2*.

The initial implementation of the algorithm performed very slow as it read in all the images (up to 200) at the start and held them in the computer's memory until the script was completed. Additionally, it used nested for-loops which are contrary to MATLAB coding best practices. This resulted in high CPU and memory usage as well as took many hours for the full analysis script to complete. In an effort to reduce the total compute utilization and time we leveraged parallel processing within MATLAB to replace the old nested for-loop approach. An example of this is the find() function in MATLAB which searches for values that meet a certain criterion – in our case exceeding a threshold value. The find() function removes the need to iterate inside each image one pixel at a time. This is a reason the performance of the script greatly improved. Prior to this change, analyzing 200 images took 4 hours. After implementing the find() function, the analysis reduced to 10 minutes.

After the Python intervalometer was developed for cell phones we began to capture and analyze their dark-frame images. When we attempted to analyze cell phone dark-frame images we often discovered the inefficiency of using a threshold value to deal with inherently high noise levels. Many of the pixels that were classified as SEUs were essentially sensor noise even at modest sensitivity levels (ISO) and stood out as false positives. Also, increasing the threshold value too high would exclude more SEU

candidates than necessary. The same was observed for DSLR cameras at high sensitivity levels (ISO). The newer analysis methods have tried to make optimizations to improve this behaviour, with only one being successful. In relation to the threshold method, the end of the chapter will contain SEU results for a range of camera models (both DSLRs and cell phones) using this method.

4.3. Local Image Noise

As mentioned earlier, cell phone images captured in digital RAW format do not undergo the same compute intensive noise suppression algorithms as images captured by DSLR cameras. Hence, cell phone camera images are vulnerable to high levels of thermal pixel noise. To get a better understanding of the distribution of the thermal noise within a cell phone camera sensor we wrote a short MATLAB script (called local_noise.m) that performed noise analysis [49]. This script read in a single dark-frame image and created a histogram of its pixel noise values using noise bin widths of 256 – i.e. 8-bit of a 16-bit pixel output value. Instead of simply plotting the histogram as pixel count versus noise output bins, a change was made in the script. The histogram would be displayed with a modified pixel count metric in the form of a percentage of the bin count to the total pixels. This is essentially dividing the pixel count in a bin by the total number of pixels in the cell phone camera sensor (e.g. 12 megapixels or 1.2×10^7 pixels). In addition, to better view the distribution, the y-axis is outputted in log base-10 scale. Figure 4.1 shows an example of a histogram generated for a dark-frame image (ISO 800, T=4 sec) captured with a Samsung S7 cell phone (1.4 μm pixels).

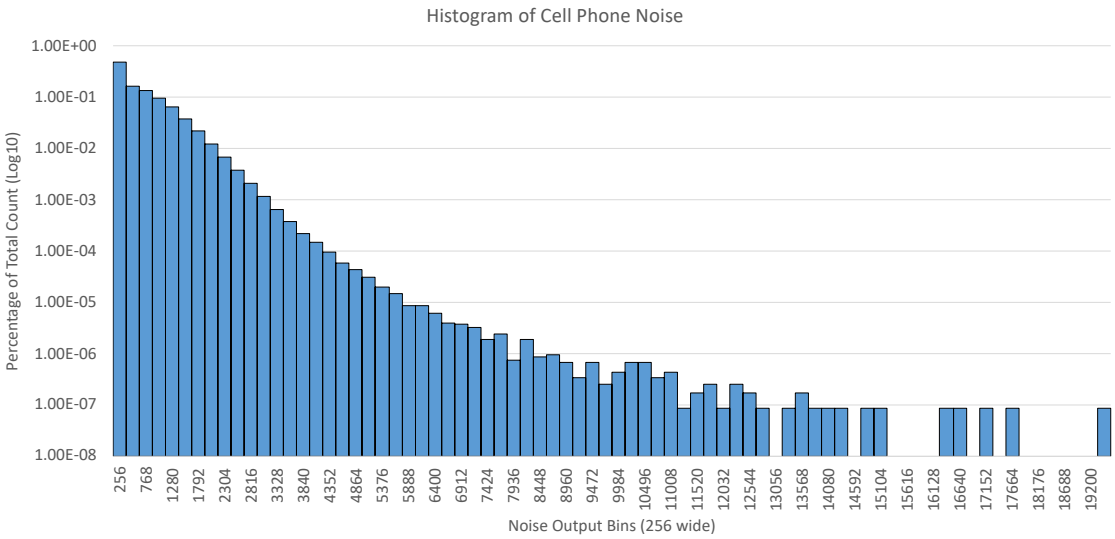


Figure 4.1 Cell phone dark-frame pixel count vs noise output bins of 256 (ISO 800, T=4 sec, Pixel Size = 1.4 μm)

By inspecting the histogram, it is clear that the distribution is much closer to a Poisson distribution (exponential distribution) as opposed to a common Gaussian distribution. Given the principles of an exponential distribution, in order for a value to be above the noise floor, the noise threshold must be the mean plus 7 standard deviations (denoted as $\mu+7\sigma$). If this was a typical Gaussian distribution, the noise threshold would be the mean plus 3 standard deviations. Pictorially, this threshold can be seen as the 7th bin from the left in the histogram which also denotes the point where the probability of it being random noise is less than 0.1%. Every SEU image dataset will have a different noise threshold as the camera model, ISO, and exposure time settings will vary. For comparison, Figures 4.2 and 4.3 show histograms generated from a low ISO dark-frame image (ISO 800, T=30 sec) and a high ISO dark-frame image (ISO 3200, T=30 sec) respectively captured with a Canon MD Mark II DSLR (6.41 μm pixels).

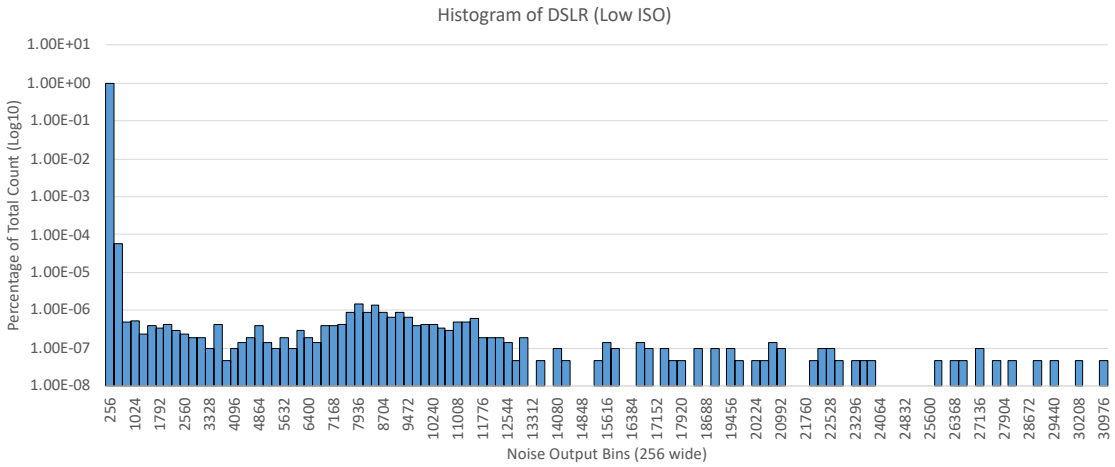


Figure 4.2 DSLR dark-frame pixel count vs noise output bins of 256 (ISO 800, T=30 sec, Pixel Size = 6.41 μm)

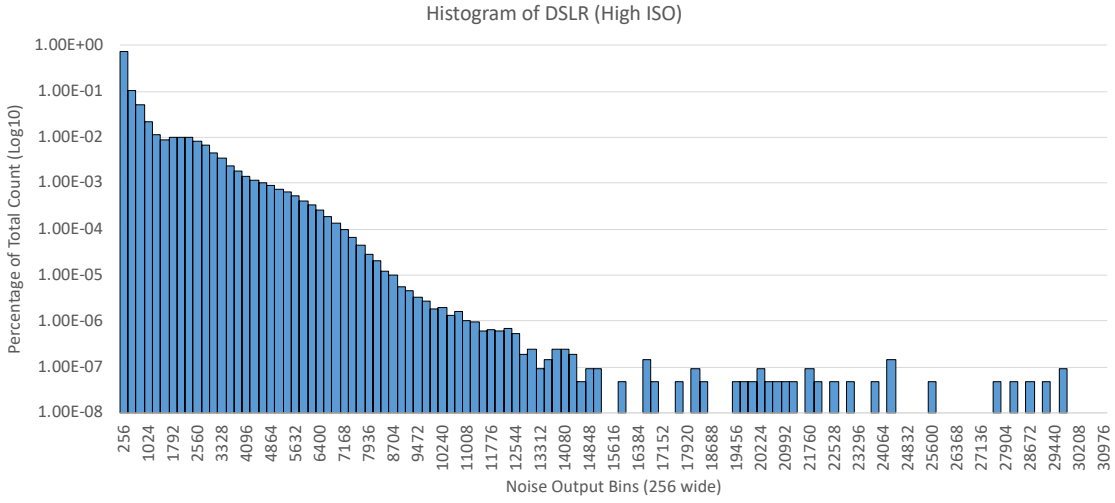


Figure 4.3 DSLR dark-frame pixel count vs noise output bins of 256 (ISO 3200, T=30 sec, Pixel Size = 6.41 μm)

It can be seen that the DSLR at a modest ISO level (ISO 800) in Figure 4.2 has very modest levels of noise – much less than the cell phone at the same ISO level in Figure 4.1. Further analyzing, the histogram of the DSLR at a high ISO level (ISO 3200) in Figure 4.3 has very high levels of noise, similar to the cell phone at a lower ISO level (ISO 800) in Figure 4.1. This proves that cell phones at modest ISO levels experience similar noise levels to DSLRs at high ISO levels. The next method that will be introduced – the Local Area Distribution Method – will leverage noise threshold values in the SEU

analysis. The initial threshold along with the noise threshold will be used in the initial steps of the next method. Using a basic user defined threshold had the inability to differentiate SEUs from false positives such as noise and hot pixels – hence, the need to explore other techniques such as local area noise.

4.4. The Local Area Distribution Method

Following the implementation of the Threshold method was the development of an algorithm that took a deeper look at the local noise distribution around a given pixel [49]. This analysis method is most commonly referred to as the Local Area Distribution method. Like before, we will define the previous image as image *j*, the current image as image *j+1*, and the next image as image *j+2*. Additionally, the image before the previous image will be defined as image *j-1* and the image after the next image will be defined as image *j+3*. There are two major ideas that differentiate the Local Area Distribution method from the previous Threshold method. The objective of making these two requirements is to discard potential SEUs that are either noisy pixels or hot pixels. These may appear in the local area around the potential SEU in the previous or following images – hence, the decision to look at images *j-1* and *j+3* as well. These requirements are:

- A potential SEU at pixel location *k* in the current image *j+1* should never repeat at the same pixel location *k* in images *j-1, j, j+2,* or *j+3.* The threshold method did not consider images *j-1* and *j+3*.

- A potential SEU at pixel location *k* in the current image *j+1* should never repeat within a close proximity to the same pixel location *k* in images *j-1, j, j+2,* or *j+3.* The threshold method did not consider pixels within a close proximity to pixel location *k* in images *j-1, j, j+2,* and *j+3*.

Figures 4.4 and 4.5 illustrate a potential SEU accepted by the method and a potential SEU rejected by the method respectively.

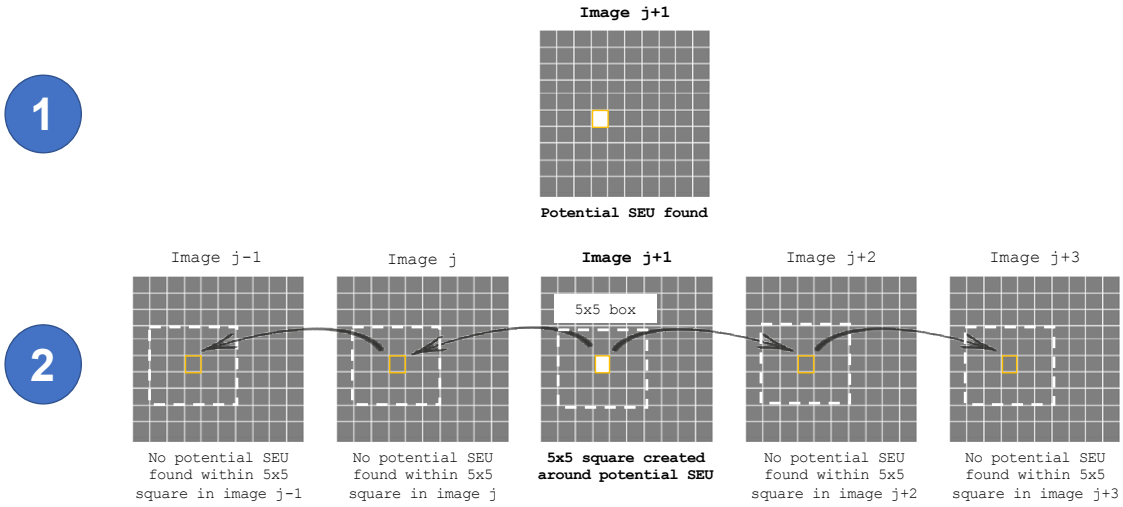


Figure 4.4 A potential SEU accepted by the Local Area Distribution method

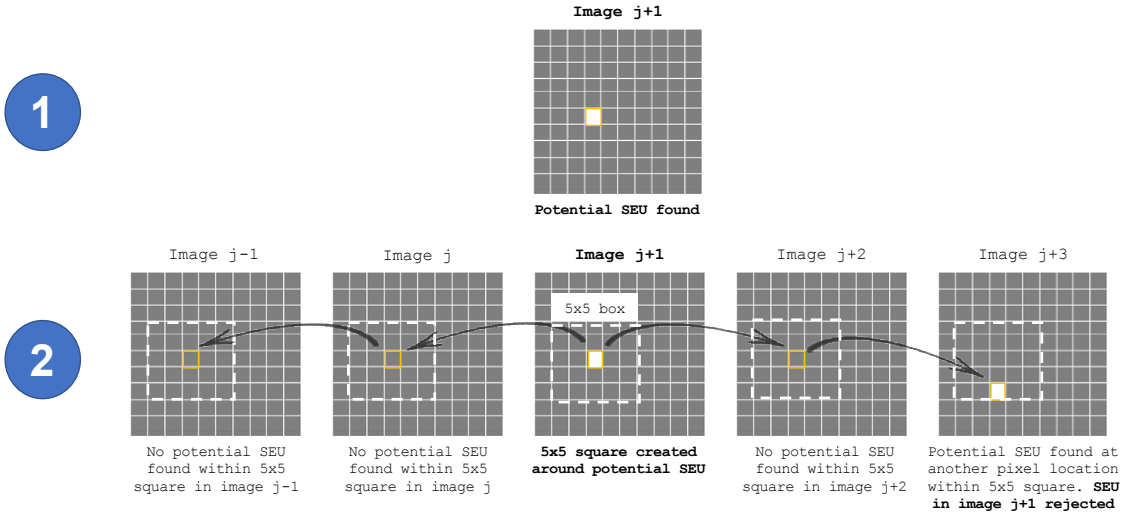


Figure 4.5 A potential SEU rejected by the Local Area Distribution method

Similar to the threshold method, the local area distribution method [49] is a software algorithm implemented in MATLAB (called seu_analysis_lad.m) that begins by reading in the dark-frame images of an SEU image dataset one by one – again, of a fixed ISO and exposure time. The tool would again accept a pre-defined threshold value and always store at most five images at a time (that is, images *j-1, j, j+1, j+2,* and *j+3*). In order to properly identify the presence of SEUs at a pixel location *k* in the current image *j+1*, the algorithm performs the following steps:

1. Check for an increase in pixel intensity at pixel location *k* by subtracting the previous image *j* from the current image *j+1*. Identify whether the resulting difference is larger than 10,000.

Step 2 is targeting cell phone cameras.

2. Check that the pixel intensity at pixel location *k* in the current image *j+1* is greater than the minimum noise threshold. As mentioned in the previous noise section, the noise threshold value is defined as the mean (denoted by μ) plus 7 standard deviations (denoted by 7σ) of the mean.

3. Check that both (1) and (2) are satisfied. If so, the pixel location *k* in the current image *j+1* is classified as a potential SEU candidate but unverified. If so, proceed to (4). If only one of (1) and (2) are satisfied, reject pixel location *k* in current image *j+1*. Equation 4.2 shows this step logically:

$$\left(Image_{j+1}(x_k, y_k) > 10{,}000\right)$$

$$\wedge$$

$$\left(Image_{j+1}(x_k, y_k) > Noise\ Threshold\right)$$

$$\implies Image_{j+1}(x_k, y_k) \text{ is a potential SEU but unverified} \tag{4.2}$$

Where x_k, y_krepresents pixel location k and $Noise\ Threshold = \mu + 7\sigma$

Step 4 will look at two images before and two images after.

4. Check that the pixel intensity at pixel location *k* in the current image *j+1* is not similar to the pixel intensity at the corresponding location in image *j-1* and image *j+3*. Since SEUs are random, the probability of them reappearing at the exact same location it did either two images prior or two images latter is almost zero.

Step 5 will look at the local area noise distribution.

5. Check the intensities of the local areas around pixel location *k* in the previous, current, and next images by creating a 5x5 square around (and

not including) pixel location *k* in images *j*, *j+1*, *j+2*. The three centerless squares can be represented as following:

$$\boldsymbol{A_j} = \begin{matrix} (x_{k_j}-2, y_{k_j}-2) & (x_{k_j}-1, y_{k_j}-2) & (x_{k_j}, y_{k_j}-2) & (x_{k_j}+1, y_{k_j}-2) & (x_{k_j}+2, y_{k_j}-2) \\ (x_{k_j}-2, y_{k_j}-1) & (x_{k_j}-1, y_{k_j}-1) & (x_{k_j}, y_{k_j}-1) & (x_{k_j}+1, y_{k_j}-1) & (x_{k_j}+2, y_{k_j}-1) \\ (x_{k_j}-2, y_{k_j}) & (x_{k_j}-1, y_{k_j}) & Empty & (x_{k_j}+1, y_{k_j}) & (x_{k_j}+2, y_{k_j}) \\ (x_{k_j}-2, y_{k_j}+1) & (x_{k_j}-1, y_{k_j}+1) & (x_{k_j}, y_{k_j}+1) & (x_{k_j}+1, y_{k_j}+1) & (x_{k_j}+2, y_{k_j}+1) \\ (x_{k_j}-2, y_{k_j}+2) & (x_{k_j}-1, y_{k_j}+2) & (x_{k_j}, y_{k_j}+2) & (x_{k_j}+1, y_{k_j}+2) & (x_{k_j}+2, y_{k_j}+2) \end{matrix} \quad (4.3)$$

Where $\boldsymbol{A_j}$ represents the 5x5 pixel square around k in image j and

x_k, y_k represents pixel location k in image j

$$\boldsymbol{A_{j+1}} = \begin{matrix} (x_{k_{j+1}}-2, y_{k_{j+1}}-2) & (x_{k_{j+1}}-1, y_{k_{j+1}}-2) & (x_{k_{j+1}}, y_{k_{j+1}}-2) & (x_{k_{j+1}}+1, y_{k_{j+1}}-2) & (x_{k_{j+1}}+2, y_{k_{j+1}}-2) \\ (x_{k_{j+1}}-2, y_{k_{j+1}}-1) & (x_{k_{j+1}}-1, y_{k_{j+1}}-1) & (x_{k_{j+1}}, y_{k_{j+1}}-1) & (x_{k_{j+1}}+1, y_{k_{j+1}}-1) & (x_{k_{j+1}}+2, y_{k_{j+1}}-1) \\ (x_{k_{j+1}}-2, y_{k_{j+1}}) & (x_{k_{j+1}}-1, y_{k_{j+1}}) & Empty & (x_{k_{j+1}}+1, y_{k_{j+1}}) & (x_{k_{j+1}}+2, y_{k_{j+1}}) \\ (x_{k_{j+1}}-2, y_{k_{j+1}}+1) & (x_{k_{j+1}}-1, y_{k_{j+1}}+1) & (x_{k_{j+1}}, y_{k_{j+1}}+1) & (x_{k_{j+1}}+1, y_{k_{j+1}}+1) & (x_{k_{j+1}}+2, y_{k_{j+1}}+1) \\ (x_{k_{j+1}}-2, y_{k_{j+1}}+2) & (x_{k_{j+1}}-1, y_{k_{j+1}}+2) & (x_{k_{j+1}}, y_{k_{j+1}}+2) & (x_{k_{j+1}}+1, y_{k_{j+1}}+2) & (x_{k_{j+1}}+2, y_{k_{j+1}}+2) \end{matrix} \quad (4.4)$$

Where $\boldsymbol{A_{j+1}}$ represents the 5x5 pixel square around k in image $j+1$ and

$x_{k_{j+1}}, y_{k_{j+1}}$ represents pixel location k in image $j+1$

$$\boldsymbol{A_{j+2}} = \begin{matrix} (x_{k_{j+2}}-2, y_{k_{j+2}}-2) & (x_{k_{j+2}}-1, y_{k_{j+2}}-2) & (x_{k_{j+2}}, y_{k_{j+2}}-2) & (x_{k_{j+2}}+1, y_{k_{j+2}}-2) & (x_{k_{j+2}}+2, y_{k_{j+2}}-2) \\ (x_{k_{j+2}}-2, y_{k_{j+2}}-1) & (x_{k_{j+2}}-1, y_{k_{j+2}}-1) & (x_{k_{j+2}}, y_{k_{j+2}}-1) & (x_{k_{j+2}}+1, y_{k_{j+2}}-1) & (x_{k_{j+2}}+2, y_{k_{j+2}}-1) \\ (x_{k_{j+2}}-2, y_{k_{j+2}}) & (x_{k_{j+2}}-1, y_{k_{j+2}}) & Empty & (x_{k_{j+2}}+1, y_{k_{j+2}}) & (x_{k_{j+2}}+2, y_{k_{j+2}}) \\ (x_{k_{j+2}}-2, y_{k_{j+2}}+1) & (x_{k_{j+2}}-1, y_{k_{j+2}}+1) & (x_{k_{j+2}}, y_{k_{j+2}}+1) & (x_{k_{j+2}}+1, y_{k_{j+2}}+1) & (x_{k_{j+2}}+2, y_{k_{j+2}}+1) \\ (x_{k_{j+2}}-2, y_{k_{j+2}}+2) & (x_{k_{j+2}}-1, y_{k_{j+2}}+2) & (x_{k_{j+2}}, y_{k_{j+2}}+2) & (x_{k_{j+2}}+1, y_{k_{j+2}}+2) & (x_{k_{j+2}}+2, y_{k_{j+2}}+2) \end{matrix} \quad (4.5)$$

Where $\boldsymbol{A_{j+2}}$ represents the 5x5 pixel square around k in image $j+2$ and

$x_{k_{j+2}}, y_{k_{j+2}}$ represents pixel location k in image $j+2$

Pool all three 24-pixel (5x5) arrays together and flatten to form a 72-pixel vector (denoted by $\boldsymbol{P_{72}}$) and check the following:

a. Check that the pixel intensity at pixel location *k* in the current image *j+1* is greater than the mean (of the 72-pixel vector, denoted by $\mu_{P_{72}}$) plus 7 standard deviations (denoted by $7\sigma_{P_{72}}$) of the mean.

b. Calculate the absolute difference between pixel intensity *k* in the current image *j+1* and the previous image *j*. Check that the resulting difference is greater than 2 standard deviations (denoted by $2\sigma_{P_{72}}$) of the mean.

c. Calculate the absolute difference between pixel intensity *k* in the current image *j+1* and the next image *j+2*. Check that the resulting difference is greater than two standard deviations (denoted by $2\sigma_{P_{72}}$) of the mean.

6. Check that (5a), (5b), and (5c) are satisfied. If so, the pixel location *k* in the current image *j+1* is ready to proceed to the last step (7). If any of (5a), (5b), or (5c) are not satisfied, reject pixel location *k* in current image *j+1*. Equation 4.6 shows this step logically:

$$\left(Image_{j+1}(x_k, y_k) > \mu_{P_{72}} + 7 * \sigma_{P_{72}}\right)$$

$$\wedge$$

$$\left(\left|Image_{j+1}(x_k, y_k) - Image_j(x_k, y_k)\right| > 2 * \sigma_{P_{72}}\right)$$

$$\wedge$$

$$\left(\left|Image_{j+1}(x_k, y_k) - Image_{j+2}(x_k, y_k)\right| > 2 * \sigma_{P_{72}}\right)$$

$$\Longrightarrow \; Image_{j+1}(x_k, y_k) \text{ is a potential SEU but unverified} \qquad (4.6)$$

Where x_k, y_krepresents pixel location k, $\mu_{P_{72}}$ is the mean of the 72-pixel array,

and $\sigma_{P_{72}}$ is the standard deviation of the 72-pixel array

Step 7 does a pixel address lookup using previously verified SEUs that have been recorded.

7. Check that the potential SEU at pixel location *k* in the current image *j+1* is not a repetition of a previously verified SEU at the corresponding location or at a location within a 5x5 square around that corresponding location. If

no match is found, the pixel location *k* in the current image *j+1* is verified unique and is classified as an SEU. If there is a match with one of the previously verified SEUs, reject pixel location *k* in current image *j+1*.

This algorithm was used on dark-frame images captured on both DSLR and cell phone cameras. Locations that passed all the criteria listed above are identified as SEUs and more granularly defined as either single pixel, cluster, or streak SEUs. When introducing this method, it was mentioned that it was designed to target improvements in the software by eliminating hot pixels and noise. When it came to hot pixels being falsely detected as SEUs, this method was observed to have improved over the previous Threshold method. Our hypothesis was that hot pixels that were getting misidentified as SEUs would now be caught by expanding our search area to 5x5 squares around pixel address *k* in images *j-1, j, j+1, j+2, and j+3*. Likewise, using both the threshold of 10,000 and the noise threshold (7 standard deviations more than the mean) in this method, showed a reduction in the number of false positives due to noise. However, many false positives that were from inherently noisy pixels were still being counted as SEUs. This is because there was no single noise threshold that did not exclude too many SEU candidates. As a result, this method was still not capable to detect SEUs in cell phones at modest ISO levels and DSLRs at high ISO levels. The next algorithm will get into details on tackling this persisting issue. Lastly, it was mentioned that the threshold method had become fairly performant after making certain software optimizations – i.e. being able to analyze 200 images in 10 minutes. The added checks associated with the Local Area Distribution method caused the script to take twice as long. Hence, analyzing 200 images took around 20 minutes. But this was an acceptable trade-off as the SEU detection process became more accurate.

4.5. Pixel Noise Distribution

As mentioned previously, cell phones have very high levels of thermal pixel noise since their digital RAW images do not undergo any noise suppression algorithms – something different from DSLR cameras. Cell phone cameras even at modest sensitivity levels (less than ISO 1600) show this behaviour. When it comes to DSLRs, images taken

with modest sensitivity levels have tolerable amounts of noise. However, when capturing images with DSLRs at higher sensitivity levels (greater than ISO 1600) they also show high levels of thermal pixel noise – much like the cell phone cameras. Another factor that makes it more problematic is that the noise varies from pixel to pixel and even in specific areas of the sensor.

Earlier we saw that the noise of a single cell phone camera sensor was analyzed by a MATLAB script (called local_noise.m) in order to understand the distribution of the thermal noise within a cell phone camera sensor. This essentially created a histogram of the pixel noise values in a single dark-frame image by using noise bin widths of 256 and a percentage of the total count for each bin. This provided insight on the local area noise and further helped set a noise threshold for the Local Area Distribution Method. However, this method still showed the existence of noisy SEU false positives when it came to cell phone cameras at modest ISO levels and DSLRs at high ISO levels. Hence, it became clear that looking at the local area noise of a single dark-frame image was truly not effective. Therefore, as a new approach we decided to understand the distribution of each pixel address in the camera sensor over 1000 images. Despite the effort and time involved in capturing 1000 images per image dataset for every camera, it remained vital in order to properly understand the distribution of noise. Therefore, we developed another algorithm in MATLAB (called noise_map.m) to generate a colour noise map of the camera sensor using 1000 dark-frame images [50].

Keep in mind that each image dataset consisted of 1000 images and had its own combination of camera model, ISO setting, and exposure time. Figure 4.6(a)–(c) above show the algorithm creating a colour noise map from the dark-frame images. For illustrative purposes, this camera captures images each having dimensions of 4032 x 3024 (x-dimension vs. y-dimension). The script began by averaging the noise values of all the pixels in the sensor over the 1000 images. This process can be seen in Figure 4.6(a)–(b) and resulted in a single averaged image.

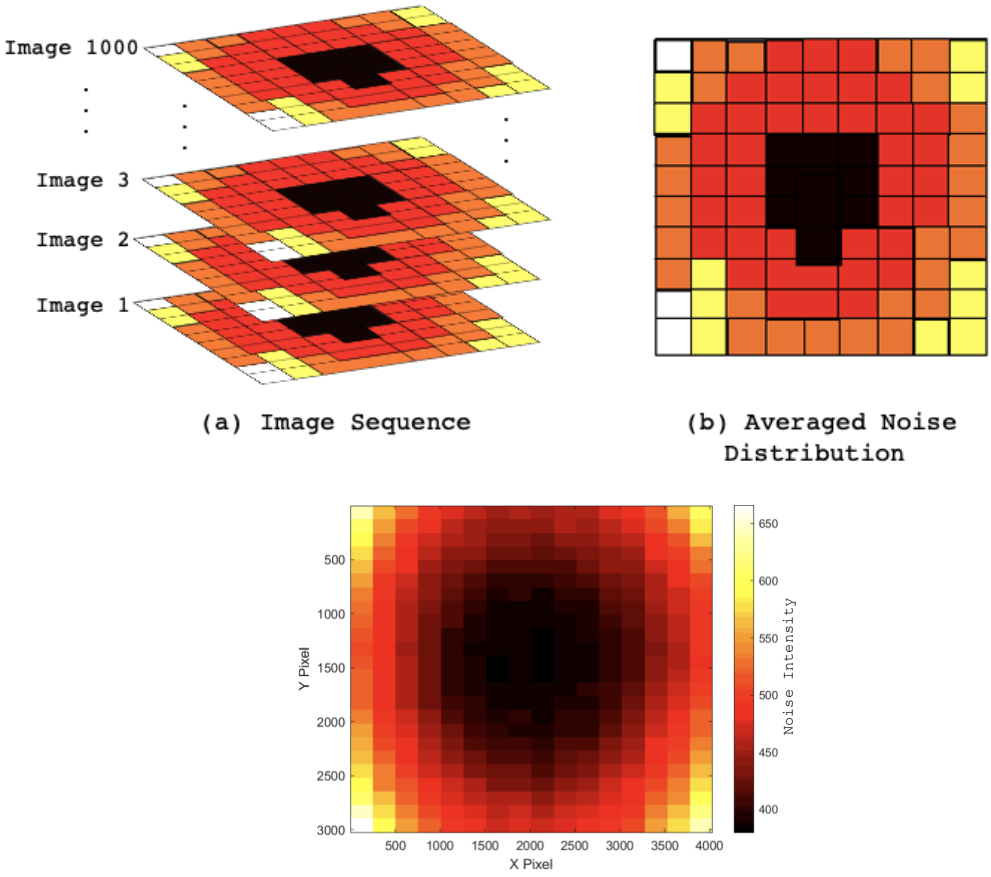


Figure 4.6 Colour noise map of a camera sensor. (a) Averaging 1000 images. (b) The averaged image. (c) Creating area boxes and generating a noise map.

Since each image was 4032 pixels wide and 3024 pixels high, the script created smaller area boxes filling the entire image. The area boxes had the following dimensions:

- x-dimension: 252 pixels ($\frac{4032}{252} = 16$ area boxes spanning the width of the image)
- y-dimension: 126 pixels ($\frac{3024}{126} = 24$ area boxes spanning the height of the image)

In total there were 384 area boxes (16*24) that were able to fit in the 4032 x 3024 image. As a next step, the algorithm would iterate through all of the area boxes and replace

their respective pixel values with the average value of the area box. This occurred for all of the area boxes in the image until the image consisted solely of averaged area boxes. The last step of the algorithm is to output a colour noise map of the 'area-boxed' image. This can be seen in Figure 4.6(c). It can be noted that the colour noise map is represented similarly to a traditional heat map, with darker and lighter colours indicating more and less noisy areas respectively. Also shown with the colour noise map is a colour scale that identifies the noise intensity of a given colour strength used by the color noise map. This particular analysis was used on both cell phone at modest ISO levels and DSLR cameras at high ISO levels. One thing to note is that every camera will have different area box dimensions as all cameras have a particular sensor size – further resulting in different dark-frame image sizes.

A thorough breakdown of colour noise maps for different camera models will be shown at the end of the chapter. However, before proceeding to the final SEU analysis algorithm (the Pixel Address Distribution method), we need to fully understand the behaviour of noise in both types of camera sensors. This will be done by analyzing the colour noise map results from a cell phone and a DSLR – more specifically, the Samsung S7 and the Canon 5D Mark II. The results of both cameras can be seen in Figures 4.7 and 4.8 respectively.

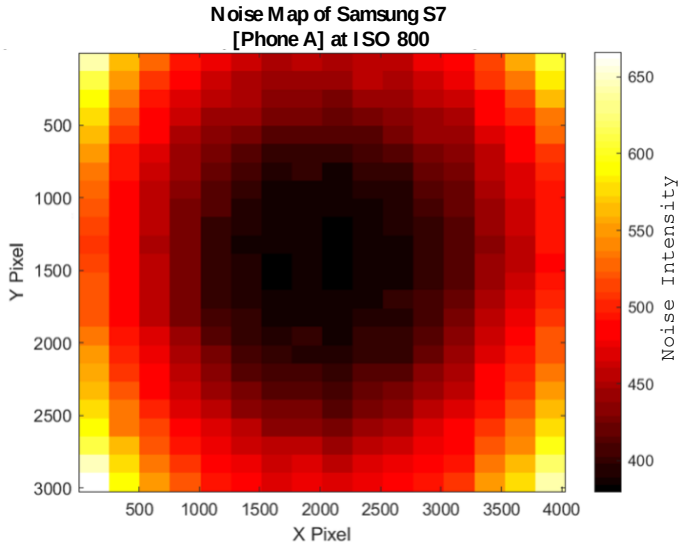


Figure 4.7 Noise Map of Samsung S7 at ISO 800 (Sensor Size: 4.76 mm x 4.29 mm)

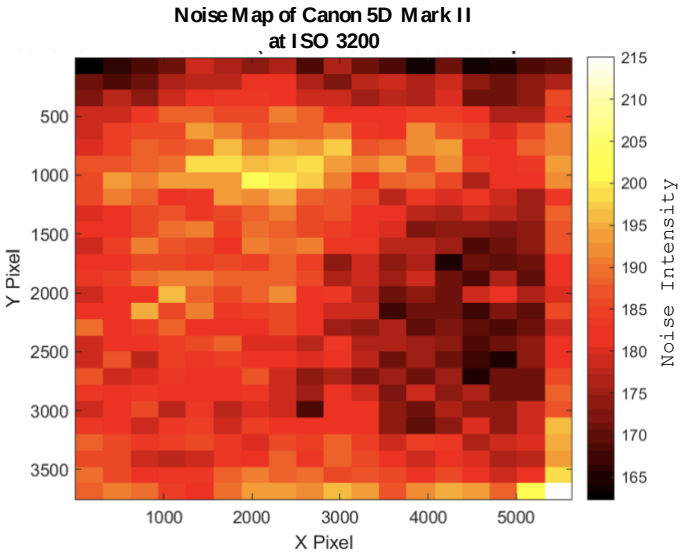


Figure 4.8 Noise Map of Canon 5D Mark II at ISO 3200 (36 mm x 24 mm sensor)

By analyzing the noise maps in the above figures, it is evident that they show different characteristics and patterns. The DSLR in Figure 4.8 shows randomly distributed noise, no distinct pattern, and a generally lower range of noise intensities than the cell phone camera. Contrastingly, the cell phone camera in Figure 4.7 exhibited a distinct pattern of 'three noisy outer rings' and a generally higher range of noise intensities than the DSLR. Since the ring behaviour was fairly astonishing, another noise map was generated with images captured by an identical Samsung S7 camera and its resulting noise maps followed the same noise distribution pattern. One reasoning for this similarity in noise distributions may be due to similar manufacturing conditions. Since both cell phones are the same model and made by the same manufacturer, they most likely would have been fabricated under the same conditions.

Recent experiments that were performed observed that some of the pixels (and some areas of the sensor) to be inherently too noisy – generating false positives in the SEU analysis. In an attempt to solve the noise problem that persisted in the two earlier SEU analysis algorithms (the Threshold method and the Local Area Distribution method) we proposed for the next SEU analysis method to follow our most recent noise analysis and consider the pixel noise distribution in all 1000 images. Follow-up experiments led to the development of another MATLAB script (called pixel_histogram.m) that would read in a

1000 image dataset for this purpose of observing the distribution of noise in individual pixels [50]. For every pixel address in the sensor, the algorithm would calculate the mean and standard deviation over 1000 dark-frame images. The script then would randomly select 100 pixel addresses in the sensor to generate histograms. It would then query the database for every pixel address that is selected and obtain their mean, standard deviation, as well as simply their values in the 1000 images. Lastly, using these values it plots a histogram for each pixel address using bins of dynamically sized widths and optionally display the mean and standard deviation of that respective pixel address. Figures 4.9(a)-(b) and 4.10(a)-(b) are examples of histograms for different pixel addresses. A characteristic that will be important to understand when implementing the Pixel Address Distribution method is the statistical distribution of the pixel address histograms. In general, most of them followed a Poisson (exponential) distribution in contrast to the well-known Gaussian distribution.

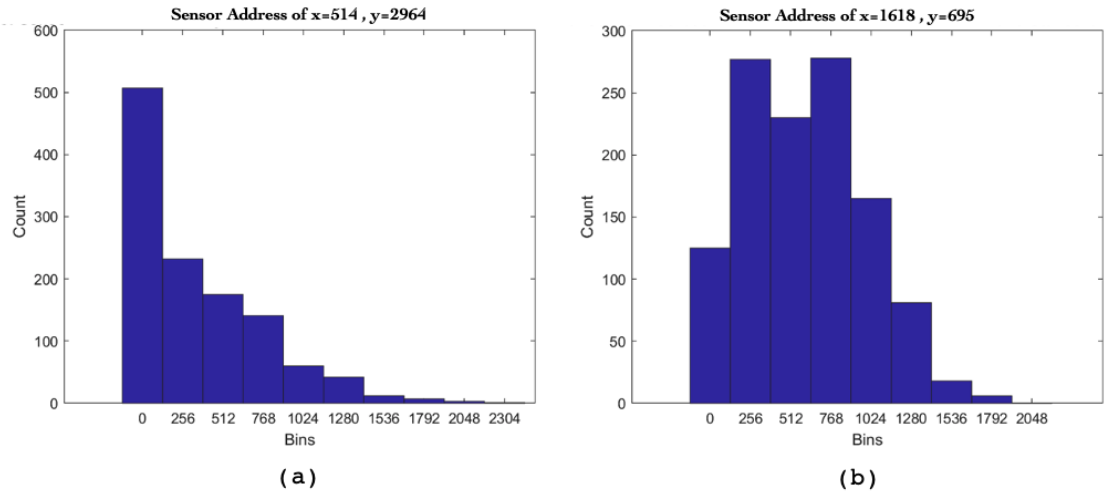


Figure 4.9 Two randomly selected pixel addresses of a Samsung S7 from 1000 dark-frame images (ISO 800) showing: (a) Poisson Distribution (b) Gaussian Distribution

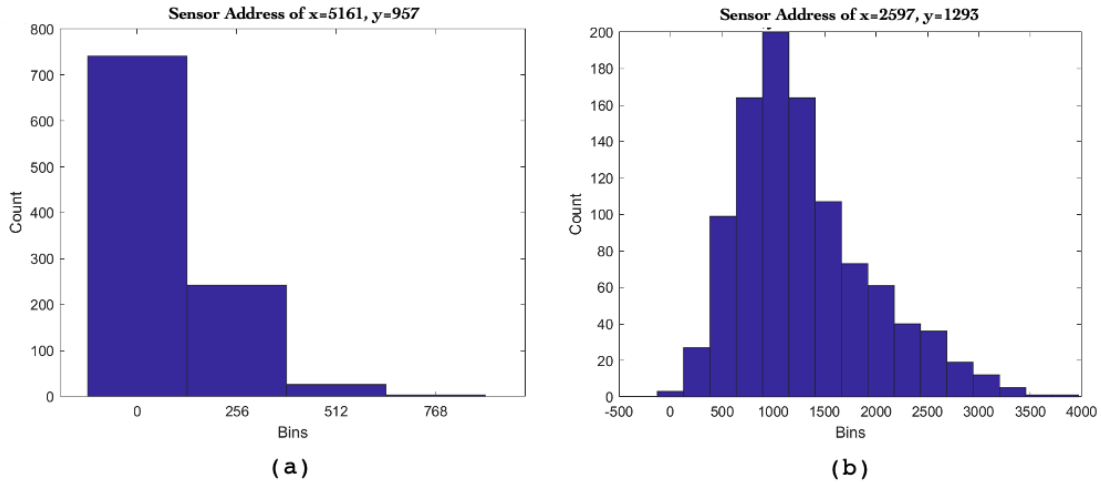


Figure 4.10 Two randomly selected pixel addresses of a Canon 5D Mark II from 1000 dark-frame images (ISO 3200) showing: (a) Poisson Distribution (b) Gaussian Distribution

The Figure pairs above each show pixel addresses with a Poisson (or exponential) distribution (Figures 4.9(a) and 4.10(a)) and pixel addresses with a Gaussian distribution (Figures 4.9(b) and 4.10(b)). After running the script to generate histograms for 100 randomly chosen pixel addresses, it can be observed that a majority of them do indeed follow an exponential distribution instead of a Gaussian distribution. If we go back to the definition of an SEU, they possess the characteristics of being non-permanent and random in nature – something which concurs with exponential distribution. The pixel addresses that showed a Gaussian distribution were a lot less common and most likely represent a very noisy pixel or a permanent hot pixel – both of which should not be falsely identified as an SEU. These observations on the exponential behaviour have influenced the implementation of the most recent SEU detection algorithm – the Pixel Address Distribution method [50].

4.6. The Pixel Address Distribution Method

Following the implementations of the Threshold and Local Area Distribution methods was the development of the most recent SEU analysis algorithm. Its focus is to

look at the distributed pixel address noise for every pixel in a camera's sensor. This analysis method is most commonly referred to as the Pixel Address Distribution method [50]. It follows the same image numbering convention as before with images *j, j+1, and j+2* referring to the previous, current, and next images respectively. However, there is one major difference that differentiates the Pixel Address Distribution method from the Local Area Distribution method. That is, it relies on the distribution of a given pixel's value over 1000 iterations (images) to determine the presence of an SEU rather than relying on the local area noise of a pixel. Since we leveraged many cameras and each had a range of sensitivities to choose from, we constructed a database of images from all of the datasets.

Similar to the previous analysis algorithms, the pixel address distribution method is a software algorithm written in MATLAB (called seu_analysis_pad.m) [50]. A difference is that it does not start analyzing images and pixels at the start. Rather, it leverages the database for a chosen image dataset that has a fixed ISO and exposure time. The algorithm begins by calculating the mean and standard deviation of each pixel address in the camera sensor using their values seen over 1000 images. This is the section of the algorithm where the majority of the computation time and resources are spent. During the initial development, the software would try to read and store 1000 images at once before calculating the mean and standard deviation. This would constantly cause the lab computer to crash as it would fully consume its memory resources. The implemented workaround was to read in the images one-by-one and keeping running totals of the mean and standard deviation while progressing through each image. The following formulas were used to calculate the mean and standard deviation for a pixel address (x_{pa}, y_{pa}) in the camera sensor and will help us understand optimizations using running totals:

$$Mean = \mu = \frac{\sum_{n=1}^{1000} Image_n(x_{pa}, y_{pa})}{m} = \frac{M_{sum}}{m} \tag{4.7}$$

$$Standard\ Deviation = \sigma = \sqrt{\frac{\sum_{n=1}^{1000}\left(Image_n(x_{pa}, y_{pa}) - \mu\right)^2}{m}} = \sqrt{\frac{S_{sum}}{m}} \tag{4.8}$$

where $n = 1,2,3 \ldots 1000$, $m = 1000$ images , $Image_n(x_{pa}, y_{pa})$ represents the value of a pixel at an location x_{pa}, y_{pa} in image n

$$M_{sum} = Image_1(x_{pa}, y_{pa}) + Image_2(x_{pa}, y_{pa}) + \cdots + Image_{1000}(x_{pa}, y_{pa}) \quad (4.9)$$

$$S_{sum} = \left(Image_1(x_{pa}, y_{pa}) - \mu\right)^2 + \left(Image_2(x_{pa}, y_{pa}) - \mu\right)^2 + \cdots + \left(Image_{1000}(x_{pa}, y_{pa}) - \mu\right)^2 \quad (4.10)$$

The final running totals are M_{sum} and S_{sum}. For M_{sum}, the value of $Image_n(x_{pa}, y_{pa})$ for pixel location x_{pa}, y_{pa} in each image (*n*) is calculated and added to the previous running total until the script has iterated through the 1000 images. For S_{sum}, the value of $\left(Image_n(x_{pa}, y_{pa}) - \mu\right)^2$ for pixel location x_{pa}, y_{pa} in each image (*n*) is calculated and added to the previous running total until the script has iterated through the 1000 images. Please note that after iterating through 1000 images twice, every pixel address in the sensor will have values for M_{sum} and S_{sum} and can be used to calculate the values of (4.7) and (4.8) for every pixel address x_{pa}, y_{pa}. Since the script has calculated the mean and standard deviation for every pixel location in the image sensor, we can now proceed to the next phase of the algorithm.

Next, the script will again iterate over every image in the dataset and further iterate through each pixel address in a given image. In order to properly identify the presence of an SEU at a pixel location *k* in the current image *j+1*, the algorithm does the following:

1. Check that the pixel intensity at pixel location *k* in the previous image *j* is less than the mean (denoted by μ) plus 4 standard deviations (denoted by 4σ) of the mean.

2. Check that the pixel intensity at pixel location *k* in the current image *j+1* is greater than the mean (denoted by μ) plus 5 standard deviations (denoted by 5σ) of the mean. The necessity of this step is to ensure that the potential SEU has a very low probability (less than 10^{-6}) of being a simple statistical realization for data of this particular pixel.

3. Check that the pixel intensity at pixel location *k* in the next image *j+2* is less than the mean (denoted by μ) plus 4 standard deviations (denoted by 4σ) of the mean.

4. Check that (1), (2), and (3) are satisfied. If so, pixel location *k* in the current image *j+1* is classified as an SEU. Equation 4.11 shows this logically. If one or more of (1), (2), or (3) are not satisfied, reject pixel location *k* in current image *j+1*.

$$\begin{gathered}\left(Image_j(x_k, y_k) < \mu + 4\sigma\right) \\ \wedge \\ \left(Image_{j+1}(x_k, y_k) > \mu + 5\sigma\right) \\ \wedge \\ \left(Image_{j+2}(x_k, y_k) < \mu + 4\sigma\right) \\ \Longrightarrow \ Image_{j+1}(x_k, y_k) \text{ is an SEU}\end{gathered} \tag{4.11}$$

Where x_k, y_krepresents pixel location k

We looked for outliers that were clearly noisy pixels such as the same location appearing in many images (which is statistically extremely unlikely) and dropped those. As we increase the number of images for a given camera to 1000, we get more SEU data and better statistical metrics on each pixel. When introducing this method, it was mentioned that it was designed to target improvements when detecting SEUs in cell phone cameras at modest ISO levels and DSLRs at high ISO levels. As far as noisy pixels being falsely detected as SEUs, this method was observed to have improved accuracy over the previous methods. The Pixel Address Distribution method reduced the number of noisy false positives being identified as SEUs and improved the detection accuracy in general. Increasing the number of images in a dataset (1000 images) increased the computation required to calculate the mean and standard deviation for every pixel location. Hence, the Pixel Address Distribution method took substantially longer. The algorithm was able to analyze 1000 images in 5 hours. But this was an acceptable trade-off as the SEU detection became very accurate. Analyzing 1000 images with the other two methods took ~55 minutes for the Threshold method and ~2 hours for the Local Area Distribution method. The results of all three methods will be compared in detail in the next section.

4.7. Results

4.7.1. Colour Noise Maps

Section 4.5 introduced the concept of colour noise maps to illustrate the distribution of noise in a digital camera sensor. The MATLAB script (called noise_map.m) generates a single colour noise map using a 1000 dark-frame image dataset. Recall, an image dataset is composed of a single camera model, ISO setting, and exposure time. At a high level, the script performs the following tasks:

1. Averages the entire image dataset into a single image
2. Creates sub-area boxes spanning the entire image
3. Replaces the pixel values in each area box with their average value
4. Generates a colour noise map from the resulting image

As mentioned earlier, each camera model has specific dimensions for their area boxes. Hence, Table 4.1 lists the dimensions of different DSLRs, and cell phone sensors used in the experiments along with the dimensions of their respective noise area boxes.

Table 4.1 Colour Noise Map Dimensions

Camera Model	Pixel Size (μm)	Sensor Width (px)	Sensor Height (px)	Area Box Width (px)	Area Box Height (px)
Canon 5DS R	4.13	8688	5820	543	194
Canon 5D Mark II	6.41	5634	3753	313	139
Canon T2i	4.29	5202	3465	306	105
Samsung S7	1.4	4032	3024	252	126

Colour noise maps are shown below for the image datasets listed in Table 4.2.

Table 4.2 Dark-frame Image Datasets for Colour Noise Map Analysis

#	Camera Model	ISO	Exposure Time (s)
1	Canon 5DS R	3200	30
2	Canon 5DS R	6400	30
3	Canon 5DS R	12800	30
4	Canon 5D Mark II	3200	30

5	Canon 5D Mark II	6400	30
6	Canon 5D Mark II	12800	30
7	Canon T2i	3200	30
8	Canon T2i	6400	30
9	Samsung S7 [Phone A]	400	4
10	Samsung S7 [Phone A]	800	4
11	Samsung S7 [Phone B]	400	4
12	Samsung S7 [Phone B]	800	4

Figures 4.11 though 4.22 illustrate colour noise maps for the camera datasets listed in Table 4.2. For each camera model, noise maps with increasing ISO levels will be presented to better understand the relationship.

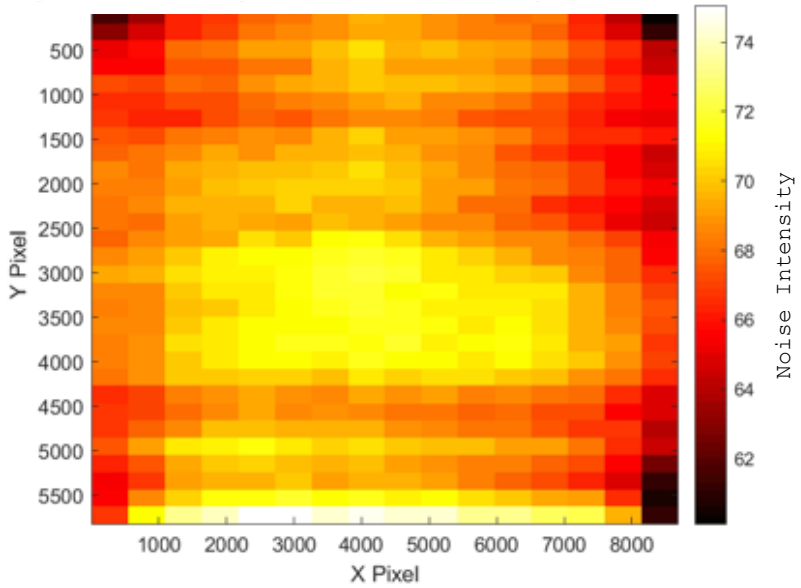


Figure 4.11 Colour Noise Maps for Canon 5DS R (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 4.13 μm)

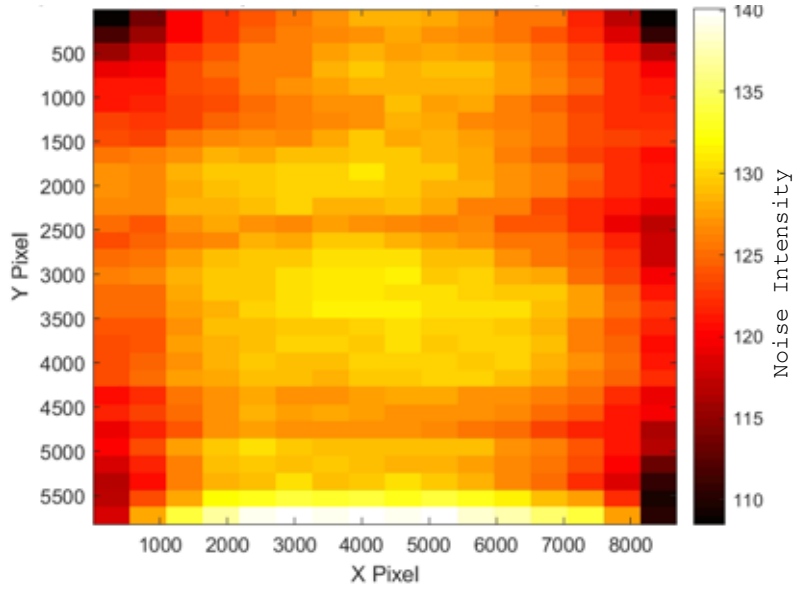


Figure 4.12 Colour Noise Maps for Canon 5DS R (ISO 6400, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 4.13 μm)

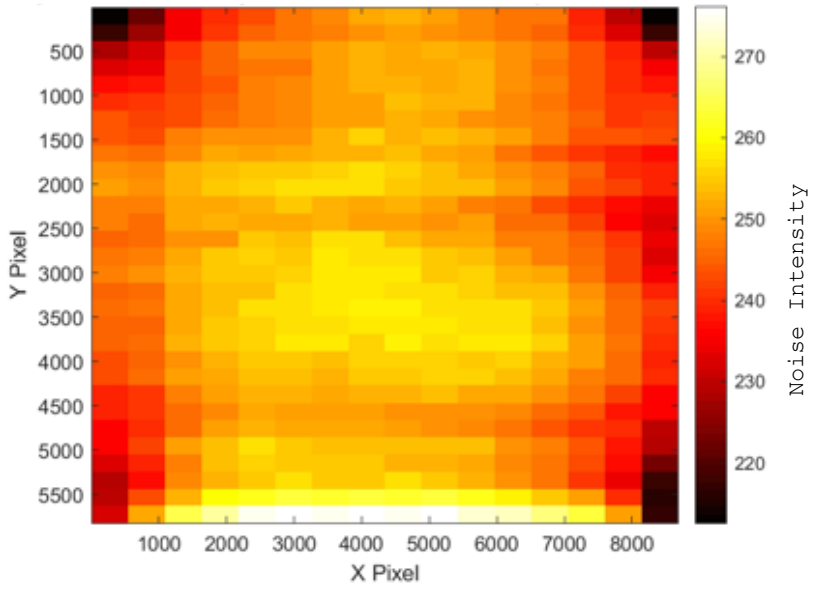


Figure 4.13 Colour Noise Maps for Canon 5DS R (ISO 12800, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 4.13 μm)

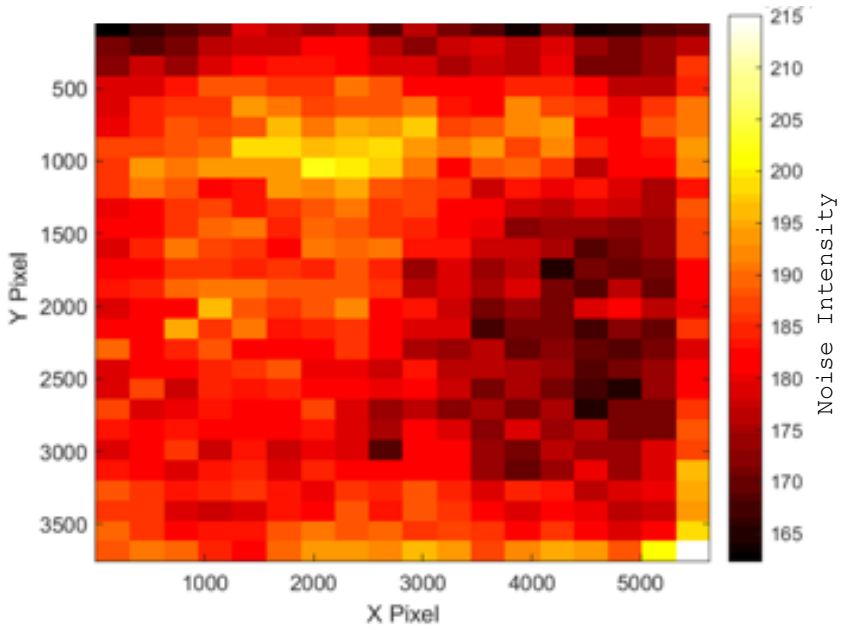


Figure 4.14 Colour Noise Maps for Canon 5D Mark II (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm)

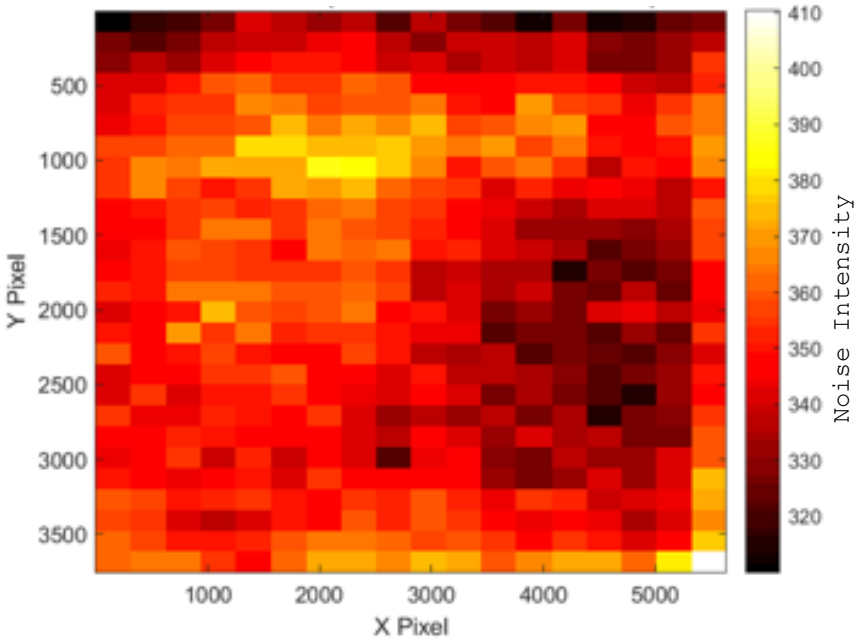


Figure 4.15 Colour Noise Maps for Canon 5D Mark II (ISO 6400, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm)

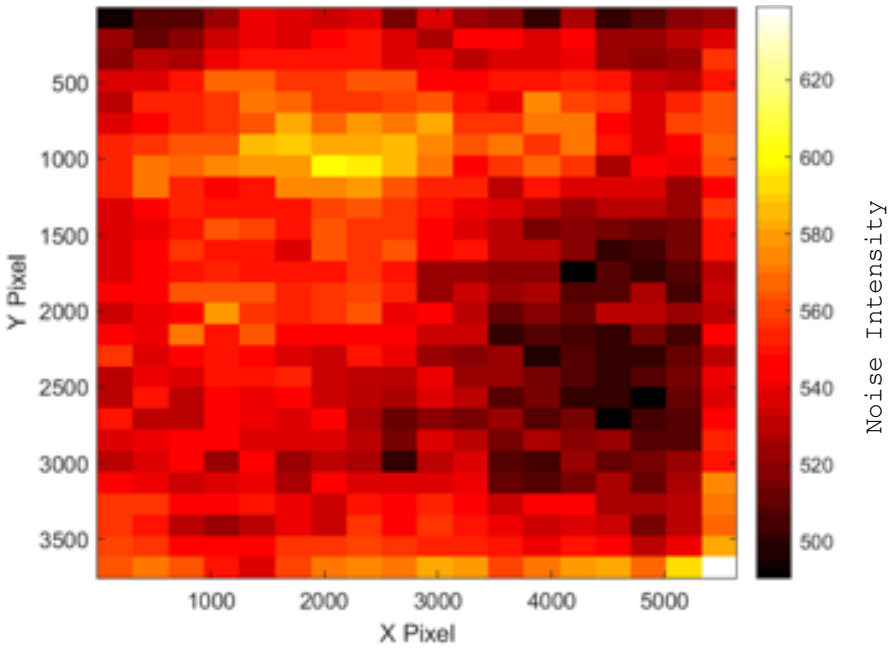


Figure 4.16 Colour Noise Maps for Canon 5D Mark II (ISO 12800, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm)

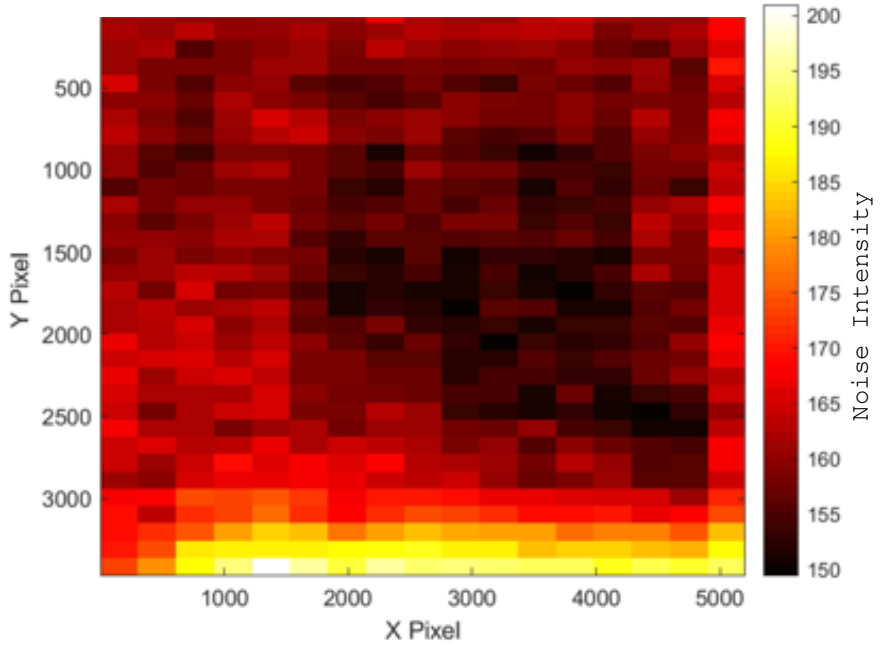


Figure 4.17 Colour Noise Maps for Canon T2i (ISO 3200, T = 30s, Sensor Size: 22.3 mm x 14.9 mm, Pixel Size = 4.29 μm)

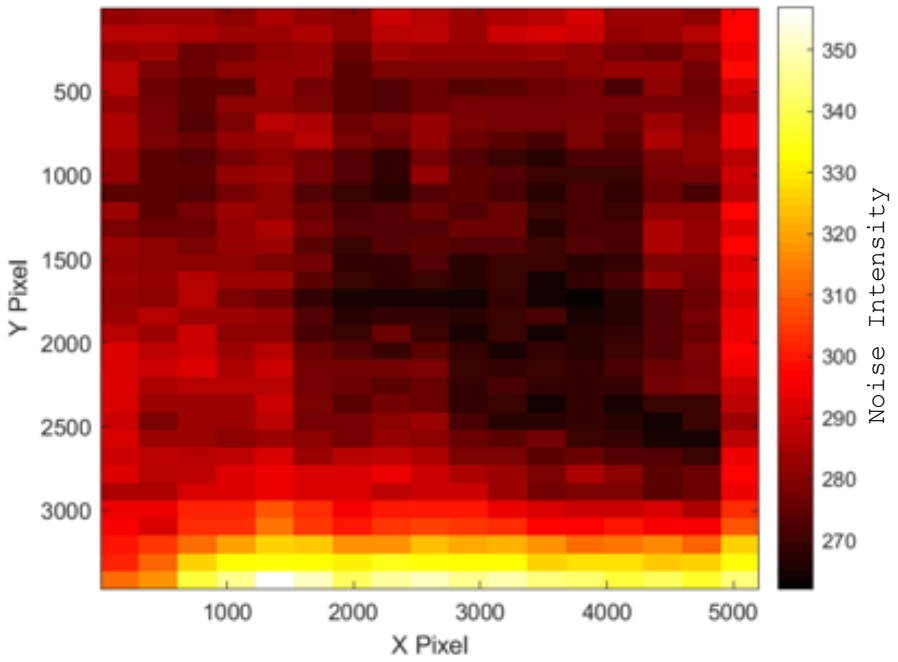


Figure 4.18 Colour Noise Maps for Canon T2i (ISO 6400, T = 30s, Sensor Size: 22.3 mm x 14.9 mm, Pixel Size = 4.29 μm)

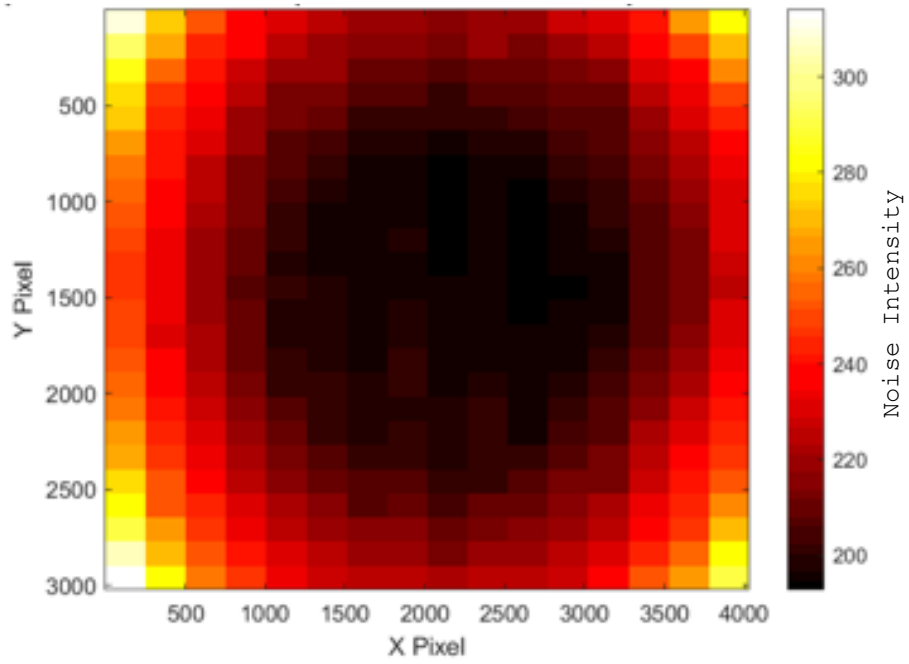


Figure 4.19 Colour Noise Maps for Samsung S7 [Phone A] (ISO 400, T = 4s, Sensor Size: 5.76 mm x 4.29 mm, Pixel Size = 1.4 μm)

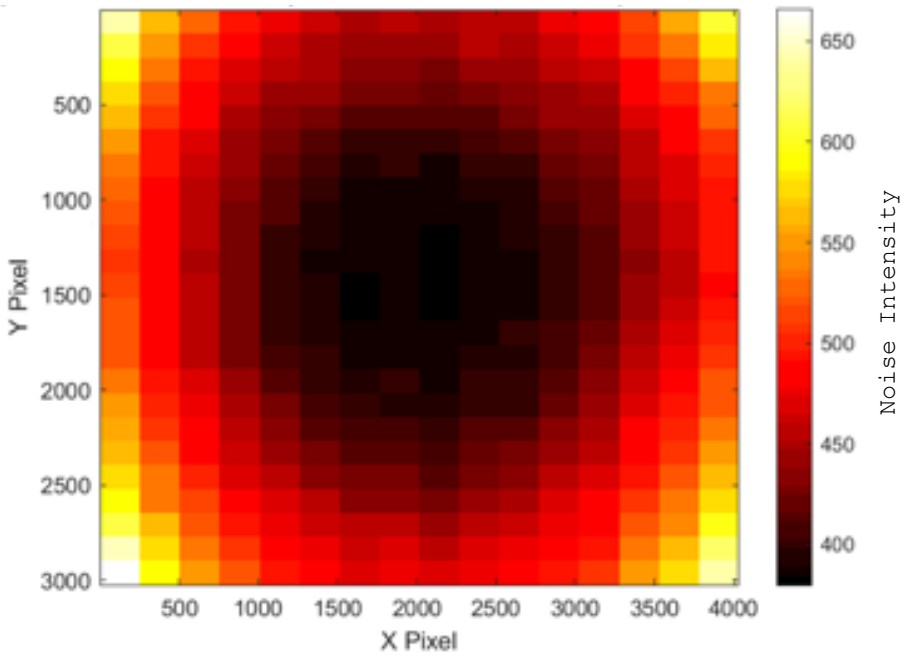


Figure 4.20 Colour Noise Maps for Samsung S7 [Phone A] (ISO 800, T = 4s, Sensor Size: 5.76 mm x 4.29 mm, Pixel Size = 1.4 μm)

The noise maps of the DSLRs (Figures 4.11 through 4.18) and the cell phone camera (Figures 4.19 and 4.20) were considerably different in both their noise distributions and patterns. The three DSLR models followed a more randomly distributed noise pattern with lower intensities – albeit each of them had different patterns and ranges of noise intensity. On the other hand, the cell phone camera seemed to have a pattern of three 'noisy outer rings' with higher intensities. It can be observed for a given camera model, as the sensitivity level (ISO setting) increased the noise pattern remained the same and the noise intensity range depicted by the colour scale increased.

Given the observed ring like pattern in the cell phone camera, we generated noise maps for another cell phone camera (same camera model) in Figures 4.21 and 4.22 below and noticed that it followed the same noise distribution pattern as Figure 4.19 and 4.20. Both cell phone cameras were made by the same manufacturer and were the same model. They would have undergone the same fabrication conditions and have similar heating conditions. This may have been the reason for seeing similar noise distributions.

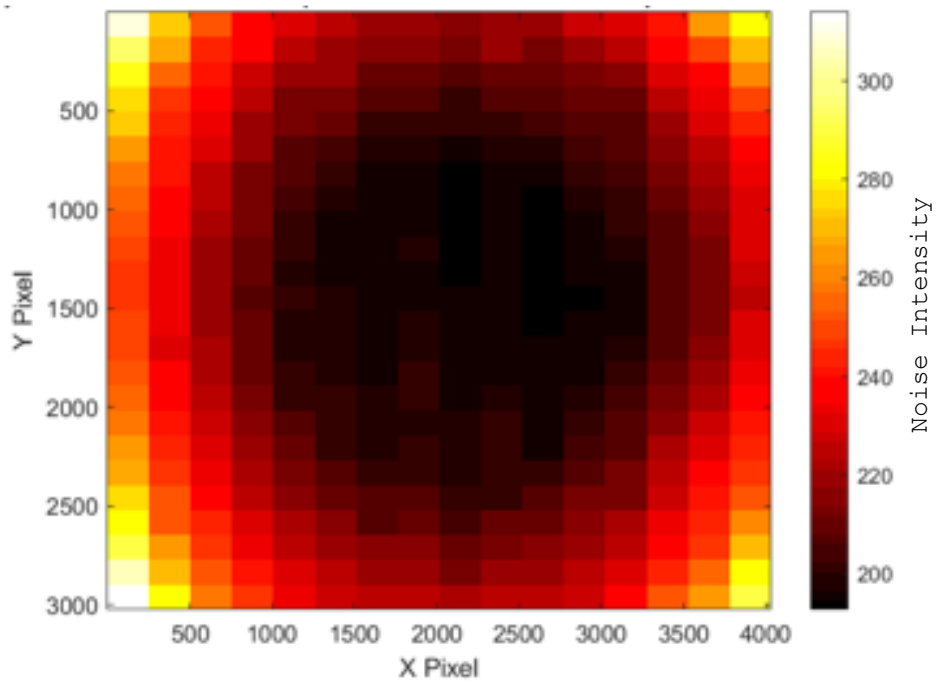


Figure 4.21 Colour Noise Maps for Samsung S7 [Phone B] (ISO 400, T = 4s, Sensor Size: 5.76 mm x 4.29 mm, Pixel Size = 1.4 μm)

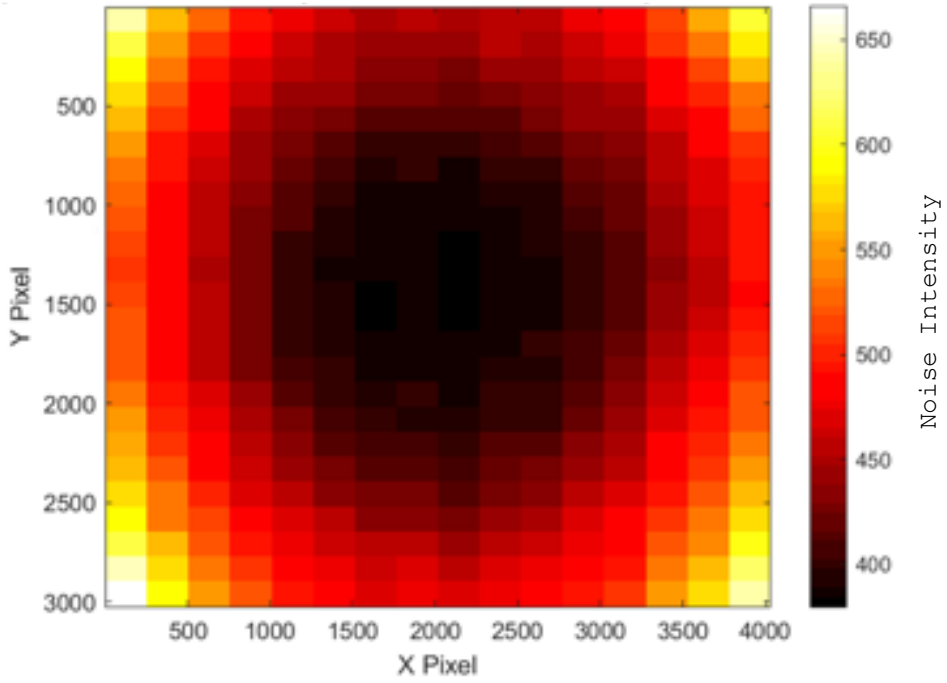


Figure 4.22 Colour Noise Maps for Samsung S7 [Phone B] (ISO 800, T = 4s, Sensor Size: 5.76 mm x 4.29 mm, Pixel Size = 1.4 μm)

4.7.2. Comparison of SEU Detection Algorithms

This section will go over the results of the three SEU detection methods that were discussed earlier in the chapter and demonstrate the high accuracy of the Pixel Address Distribution method. Results of DSLRs at high ISO levels and cell phone cameras at modest ISO levels will be presented here. All the algorithms were run on a standard lab desktop machine – not on GPUs. As a reminder, these are the three algorithms that we have developed in this research (from oldest to newest):

- **The Threshold Method:** Identifies an SEU at a pixel location *k* using a user defined threshold value.
- **The Local Area Distribution Method:** Identifies an SEU at a pixel location *k* using a noise threshold value calculated by the local area noise in a single dark-frame image.
- **The Pixel Address Distribution Method:** Identifies an SEU at a pixel location *k* using the distributed mean and standard deviation of *k* over 1000 dark-frame images.

SEU results using the three detection algorithms are shown below for the image datasets listed in Table 4.3. For each camera model, image datasets with different sensitivity levels (ISO settings) were used to observe the correlation of light sensitivity and the number of SEUs. The exposure time used for DSLRs and cell phones were 30 seconds and 4 seconds respectively.

Table 4.3 Dark-frame Image Datasets for SEU Detection Algorithm Comparison

#	Camera Model	ISO	Exposure Time (s)
1	Canon 5DS R	3200	30
2	Canon 5DS R	6400	30
3	Canon 5DS R	12800	30
4	Canon 5D Mark II	3200	30
5	Canon 5D Mark II	6400	30
6	Canon 5D Mark II	12800	30
7	Canon T2i	3200	30
8	Canon T2i	6400	30
9	Samsung S7 [Phone A]	400	4
10	Samsung S7 [Phone A]	800	4
11	Samsung S7 [Phone B]	400	4
12	Samsung S7 [Phone B]	800	4

The names of the three MATLAB scripts are seu_analysis_threshold.m, seu_analysis_lad.m, and seu_analysis_pad.m. Figures 4.23 through 4.34 represent the SEU results of the three SEU detection algorithms for each camera model.

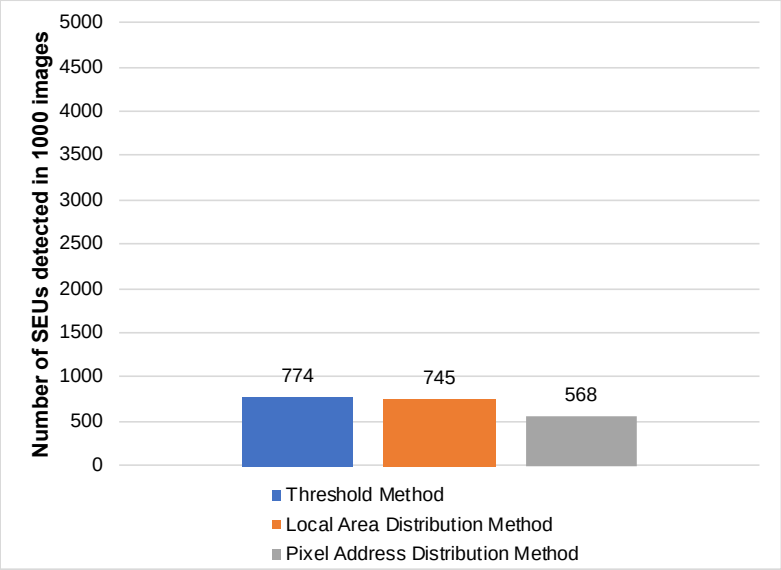


Figure 4.23 SEU algorithm comparison – Canon 5DS R (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm)

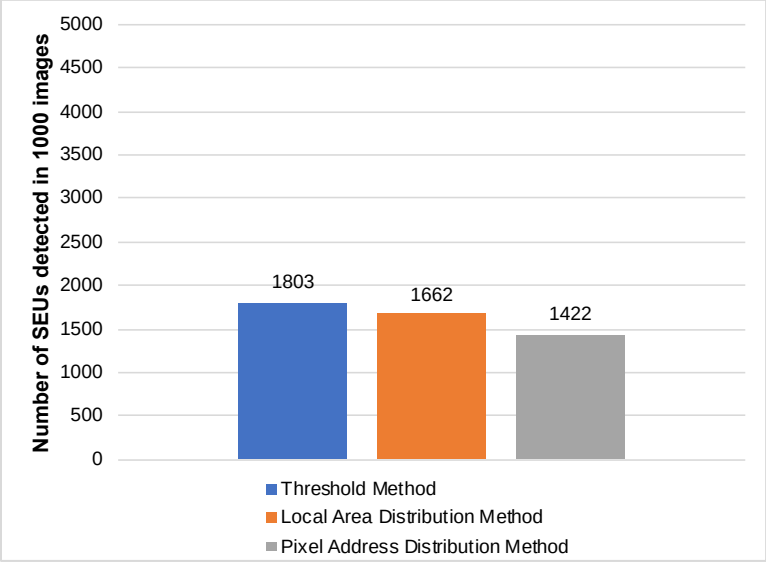


Figure 4.24 SEU algorithm comparison – Canon 5DS R (ISO 6400, T = 30s, Sensor Size: 36 mm x 24 mm)

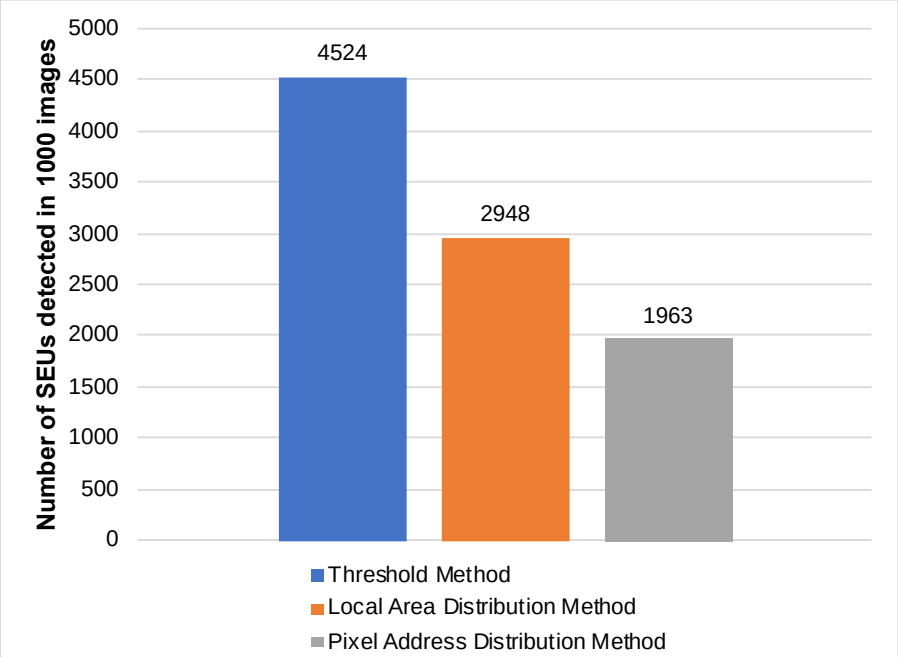


Figure 4.25 SEU algorithm comparison – Canon 5DS R (ISO 12800, T = 30s, Sensor Size: 36 mm x 24 mm)

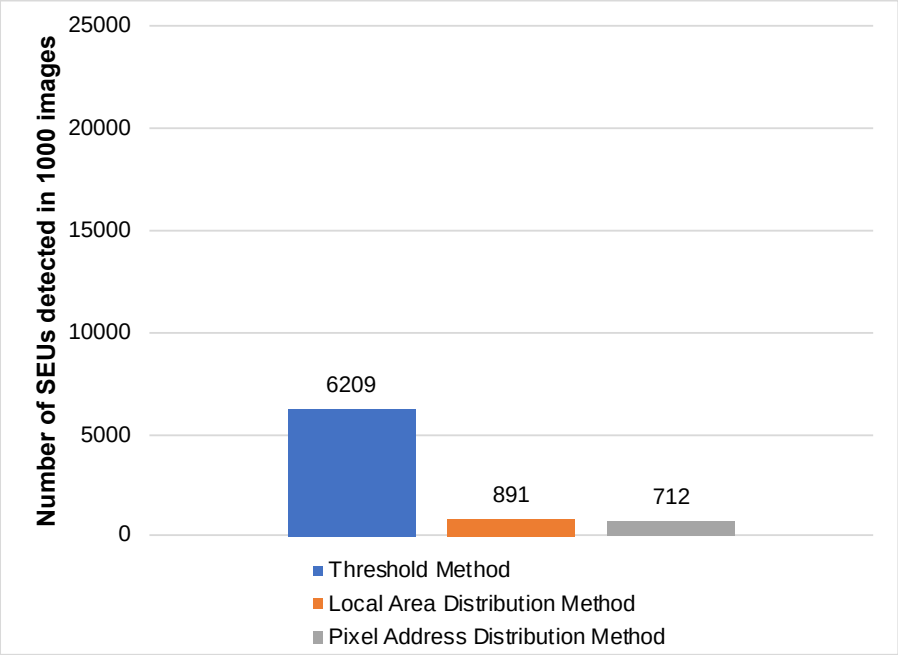


Figure 4.26 SEU algorithm comparison – Canon 5D Mark II (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm)

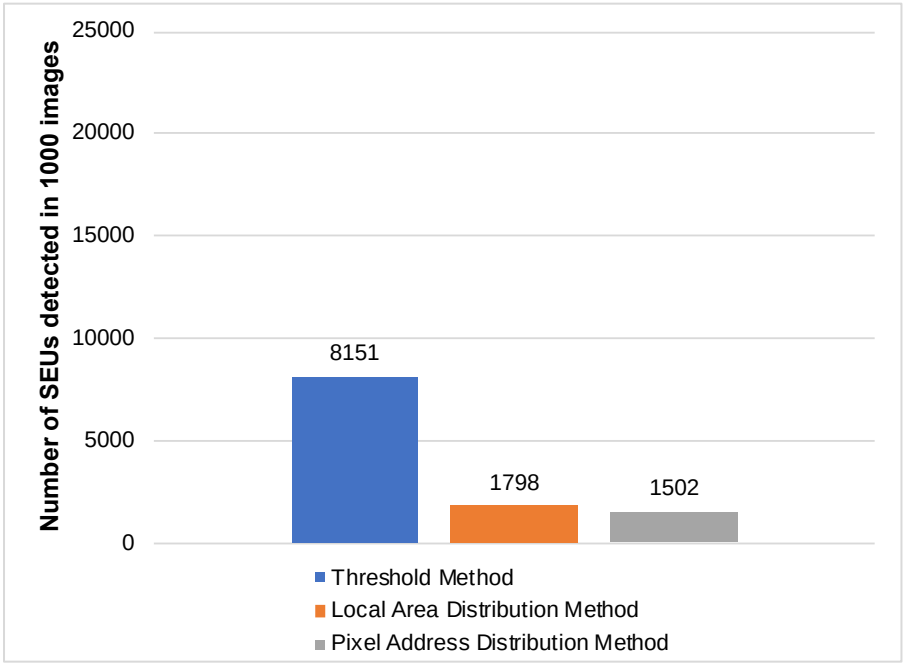


Figure 4.27 SEU algorithm comparison – Canon 5D Mark II (ISO 6400, T = 30s, Sensor Size: 36 mm x 24 mm)

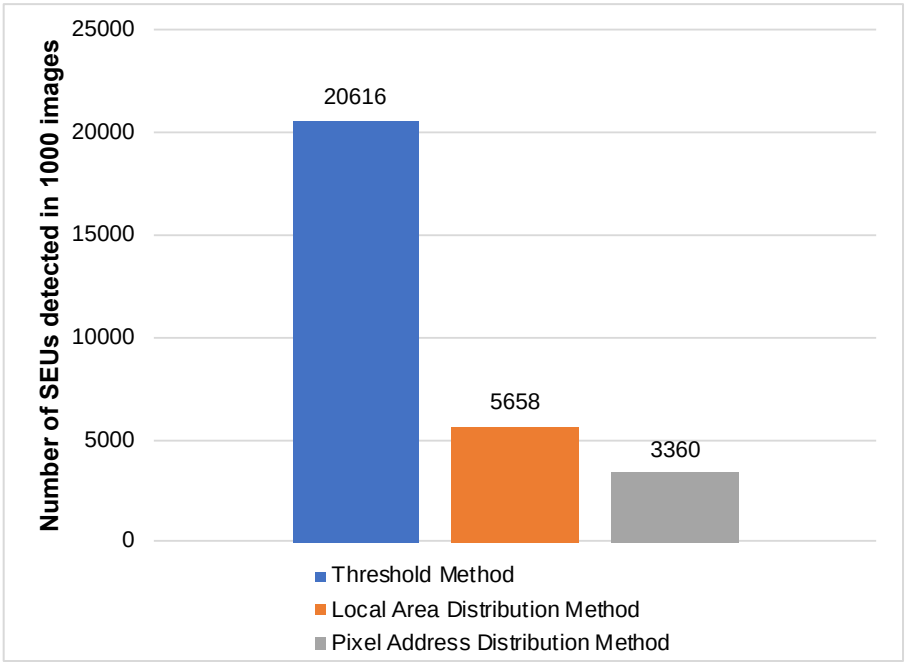


Figure 4.28 SEU algorithm comparison – Canon 5D Mark II (ISO 12800, T = 30s, Sensor Size: 36 mm x 24 mm)

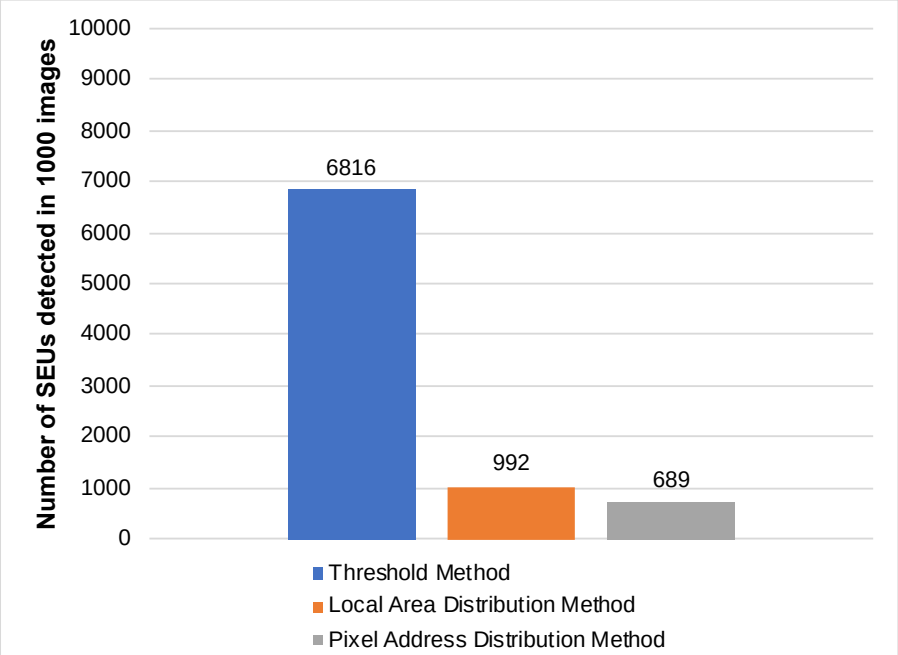


Figure 4.29 SEU algorithm comparison – Canon T2i (ISO 3200, T = 30s, Sensor Size: 22.3 mm x 14.9 mm)

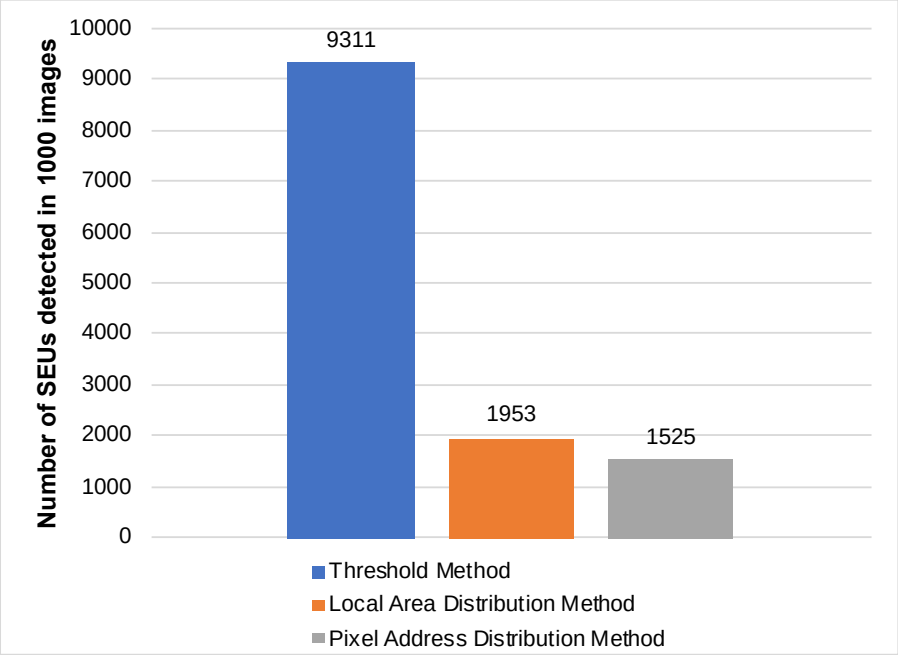


Figure 4.30 SEU algorithm comparison – Canon T2i (ISO 6400, T = 30s, Sensor Size: 22.3 mm x 14.9 mm)

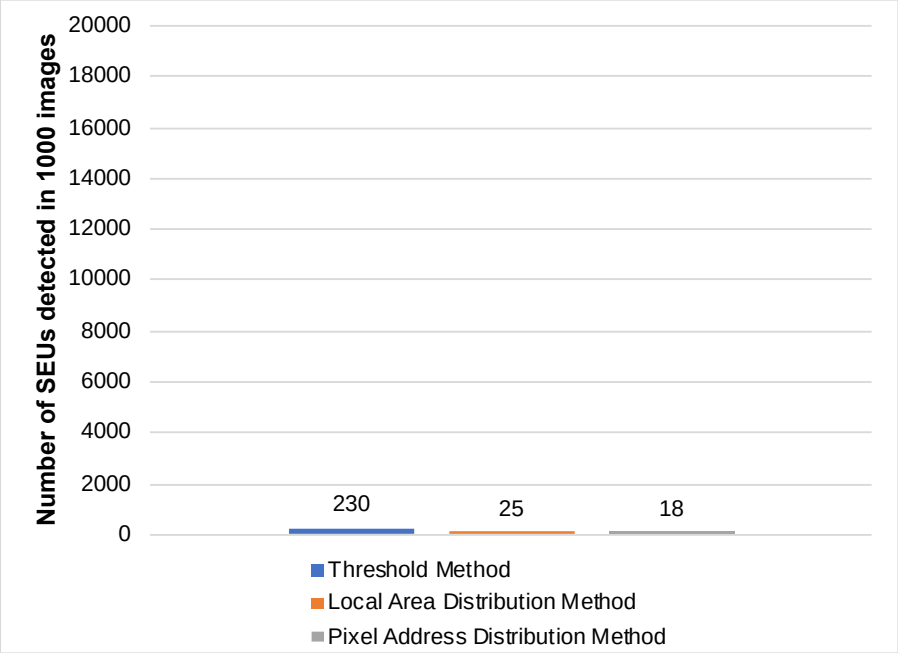


Figure 4.31 SEU algorithm comparison – Samsung S7 [Phone A] (ISO 400, T = 4s, Sensor Size: 5.76 mm x 4.29 mm)

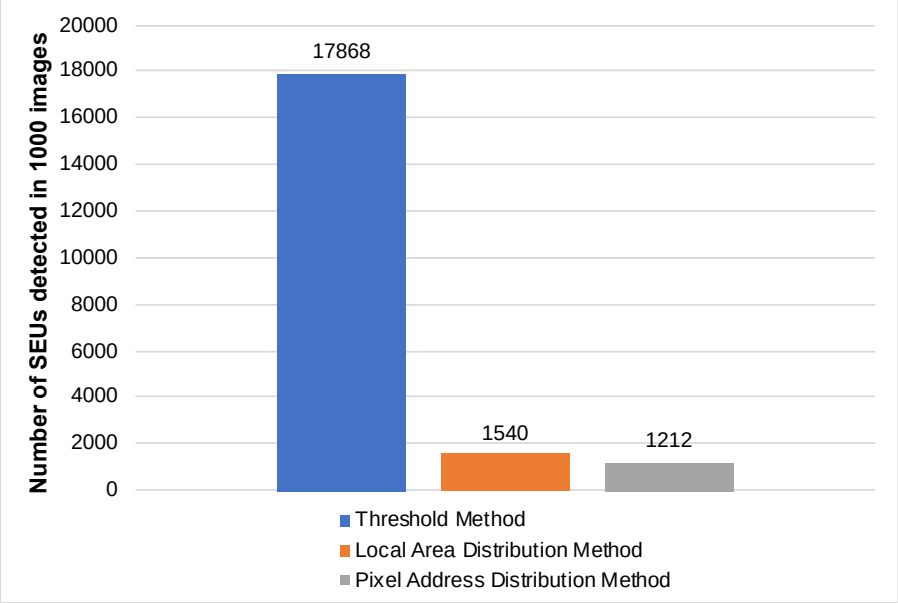


Figure 4.32 SEU algorithm comparison – Samsung S7 [Phone A] (ISO 800, T = 4s, Sensor Size: 5.76 mm x 4.29 mm)

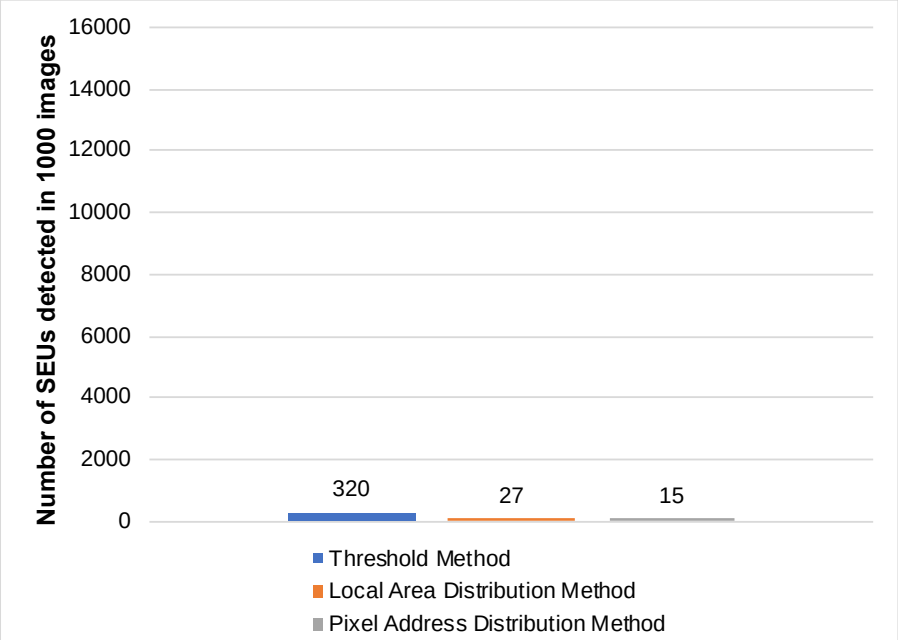


Figure 4.33 SEU algorithm comparison – Samsung S7 [Phone B] (ISO 400, T = 4s, Sensor Size: 5.76 mm x 4.29 mm)

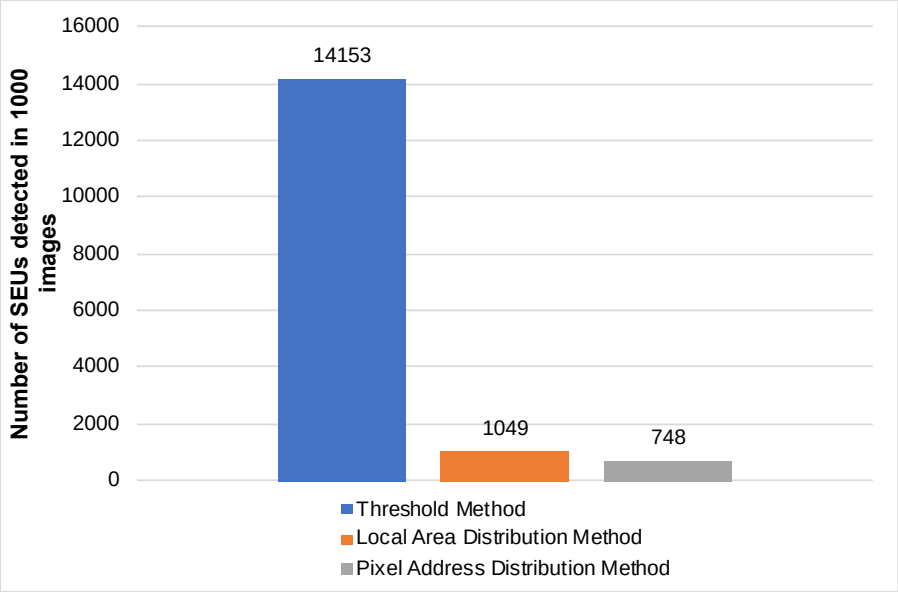


Figure 4.34 SEU algorithm comparison – Samsung S7 [Phone B] (ISO 800, T = 4s, Sensor Size: 5.76 mm x 4.29 mm)

The purpose of these experiments was to leverage the Pixel Address Distribution method to reduce the number of falsely identified SEU events and identify weaker SEUs in noisy image datasets – from DSLRs at high ISOs and cell phones at modest ISO levels.

We wanted to reject any noisy pixels or hot pixels that were being incorrectly considered as SEU events by our previous algorithms. The SEU results depicted in Figures 4.23 through 4.34 clearly show that the Pixel Address Distribution method – the most recent algorithm – is the most superior method. The original threshold method can be visibly seen misidentifying hot pixels and noisy pixels as SEUs. The Local Area Distribution method did a much better job of discarding these false positives than the Threshold Method but was still unable to fully resolve the issue. Therefore, this proves that the Pixel Address Distribution method is able to utilize noise suppression via pixel distribution and properly identify weaker SEUs in DSLRs at high ISOs and cell phones at modest ISO levels.

4.8. Summary

This chapter continued the topic of Single Event Upsets from the previous chapter and discussed noise reduction and SEU detection. The improvements over time have showed that the SEU analysis methods have tried to filter out false positives such as Hot Pixels and noise in order to make the results more accurate. Therefore, our focus has been to separate sensor noise from SEUs in dark-frame images and have the ability to detect weaker SEUs in cell phone cameras at modest ISO levels and DSLR cameras at high ISO levels. The experimental results from a range of digital cameras using all three detection algorithms show that the Pixel Address Distribution method gives us the best results. The next chapter will continue this analysis and further investigate SEUs. Some of the concepts that will be explored are the charge distribution of SEUs, the relationship of SEU occurrence with elevation and the size of the SEU charge ball.

Chapter 5.

SEU Results and Analysis

5.1. Overview

In the previous chapters of this thesis we walked through the various concepts of SEUs in digital imagers as well as defined algorithms to both reduce noise and detect weaker SEUs. Additionally, at the end of the previous chapter we generated colour noise maps and presented experimental results using the three developed algorithms for a range of digital imagers. Specifically, the experimental data showed the higher accuracy of the Pixel Address Distribution method in comparison to the two previous methods. This chapter will continue the SEU analysis from the Pixel Address Distribution method by exploring the following concepts:

- The SEU occurrence rates with respect to time and area
- The SEU charge distribution
- The relationship of SEU events and elevation
- The size analysis of the SEU charge ball

We will walk through these concepts in this chapter before concluding the thesis in the final chapter.

5.2. SEU Occurrence rates

Earlier we saw results of SEUs in terms of count per image dataset – each consisting of 1000 dark-frame images. However, the SEU count per 1000 images does not translate well when comparing image datasets of different sizes. As a result, it is important to calculate the SEU occurrence rate. This will help us develop an understanding of the behaviour and extend it to image datasets of different sizes. Our previous research in Chapman [51] followed the assumption that events caused by cosmic radiation follow a

Poisson process for their occurrence rate. This will also be the assumption taken in this thesis. The equation for the Poisson process is given by equation (5.1).

$$f(k,\lambda,t) = \frac{(\lambda t)^k e^{-\lambda t}}{k!} \quad \text{(5.1) Chapman [51]}$$

Where λ is the event rate (per second), and λt is the expected number of events occurring in t seconds

Given that we are working with variations of pixel sizes, sensor areas, and exposure times, the metrics that will be used to quantify the occurrence rate are:

1. Event rate per second
2. Event rate per second per cm^2

Occurrence rates are shown in for the following image datasets listed in Table 5.1. Occurrence rates for DSLRs are shown in Tables 5.2 through 5.4 and for cell phone cameras in Tables 5.5 and 5.6.

Table 5.1 Dark-frame Image Datasets for SEU Occurrence Rate Analysis

#	Camera Model	ISO	Exposure Time (s)
1	Canon 5DS R	3200	30
2	Canon 5DS R	6400	30
3	Canon 5DS R	12800	30
4	Canon 5D Mark II	3200	30
5	Canon 5D Mark II	6400	30
6	Canon 5D Mark II	12800	30
7	Canon T2i	3200	30
8	Canon T2i	6400	30
9	Samsung S7 [Phone A]	400	4
10	Samsung S7 [Phone A]	800	4
11	Samsung S7 [Phone B]	400	4
12	Samsung S7 [Phone B]	800	4

Table 5.2 DSLR SEU rates (per second and per second per area) at ISO 3200, T=30s

Camera Model	Pixel Size (μm)	Sensor Size (mm x mm)	Rate per second (λt)	Rate per second per cm^2 (λt/cm^2)
Canon 5DS R	4.13	36 x 24	0.0237	0.0027
Canon 5D Mark II	6.41	36 x 24	0.0189	0.0022
Canon T2i	4.29	22.3 x 14.9	0.0230	0.0069

Table 5.3 DSLR SEU rates (per second and per second per area) at ISO 6400, T=30s

Camera Model	Pixel Size (μm)	Sensor Size (mm x mm)	Rate per second (λt)	Rate per second per cm^2 (λt/cm^2)
Canon 5DS R	4.13	36 x 24	0.0501	0.0058
Canon 5D Mark II	6.41	36 x 24	0.0474	0.0055
Canon T2i	4.29	22.3 x 14.9	0.0508	0.0153

Table 5.4 DSLR SEU rates (per second and per second per area) at ISO 12800, T=30s

Camera Model	Pixel Size (μm)	Sensor Size (mm x mm)	Rate per second (λt)	Rate per second per cm^2 (λt/cm^2)
Canon 5DS R	4.13	36 x 24	0.1120	0.0130
Canon 5D Mark II	6.41	36 x 24	0.0974	0.0113

Table 5.5 Cell Phone SEU rates (per second and per second per area) at ISO 400, T=4s

Camera Model	Pixel Size (μm)	Sensor Size (mm x mm)	Rate per second (λt)	Rate per second per cm^2 (λt/cm^2)
Samsung S7 [Phone A]	1.4	5.76 x 4.29	0.0006	0.0024
Samsung S7 [Phone B]	1.4	5.76 x 4.29	0.0005	0.0020

Table 5.6 Cell Phone SEU rates (per second and per second per area) at ISO 800, T=4s

Camera Model	Pixel Size (μm)	Sensor Size (mm x mm)	Rate per second (λt)	Rate per second per cm^2 (λt/cm^2)
Samsung S7 [Phone A]	1.4	5.76 x 4.29	0.0404	0.1635
Samsung S7 [Phone B]	1.4	5.76 x 4.29	0.0249	0.1009

By examining the SEU rates of the various camera models in Tables 5.2 through 5.6, we can see that the SEU occurrence rates increase significantly as the ISO level is doubled. This is consistent for all the camera models. The DSLR SEU occurrence rates experience a 110% to 150% increase when going from ISO 3200 (Table 5.2) to ISO 6400 (Table 5.3) and a 105% to 124% increase when going from ISO 6400 (Table 5.3) to ISO 12800 (Table 5.4). The cell phone cameras see an even more significant percent increase when doubling the ISO level. The cell phone SEU occurrence rates experience a 4,880% to 6,630% increase when going from ISO 400 (Table 5.5) to ISO 800 (Table 5.6). This huge spike in cell phone SEU occurrence rates when going from ISO 400 to ISO 800 might be due to the fact that there are very few events at ISO 400 – less than 20 to be more accurate. Since the smaller pixels in cell phones (1.4 μm) are less sensitive to light than DSLRs, the gain may need to be increased to ISO 800 before really noticing any SEU events. As mentioned earlier, shrinking the pixel size does not change the sensitivity level until you reach the micron pixel size that cell phone camera pixels fall within.

5.3. SEU Charge Distribution

In the previous chapter, results from the three SEU detection algorithms were shown – in particular, the Pixel Address Distribution method results. However, we did not explore the charge distribution of the identified SEUs. To better visualize and understand the charge distributions, we created a MATLAB script (seu_charge_analysis.m) that generated a histogram of the SEU results for each image dataset. The script scanned the charge values of all the SEUs that were output from the Pixel address distribution method and generated a histogram using bins widths based on the SEU charge ranges. The histograms have bin widths of 8192 (or 2^{13}) representing the charge ranges of the SEUs identified in an image dataset. Charge distribution histograms of SEU results are shown for the image datasets in Table 5.7.

Table 5.7 Dark-frame Image Datasets for SEU Charge Distribution Analysis

#	Camera Model	ISO	Exposure Time (s)
1	Canon 5DS R	3200	30
2	Canon 5DS R	6400	30
3	Canon 5DS R	12800	30
4	Canon 5D Mark II	3200	30
5	Canon 5D Mark II	6400	30
6	Canon 5D Mark II	12800	30
7	Canon T2i	3200	30
8	Canon T2i	6400	30
9	Samsung S7 [Phone A]	400	4
10	Samsung S7 [Phone A]	800	4
11	Samsung S7 [Phone B]	400	4
12	Samsung S7 [Phone B]	800	4

Figures 5.1 through 5.8 contain the charge distribution histograms of the DSLR results and Figures 5.9 and 5.12 contain the charge distribution histograms of the cell phone results.

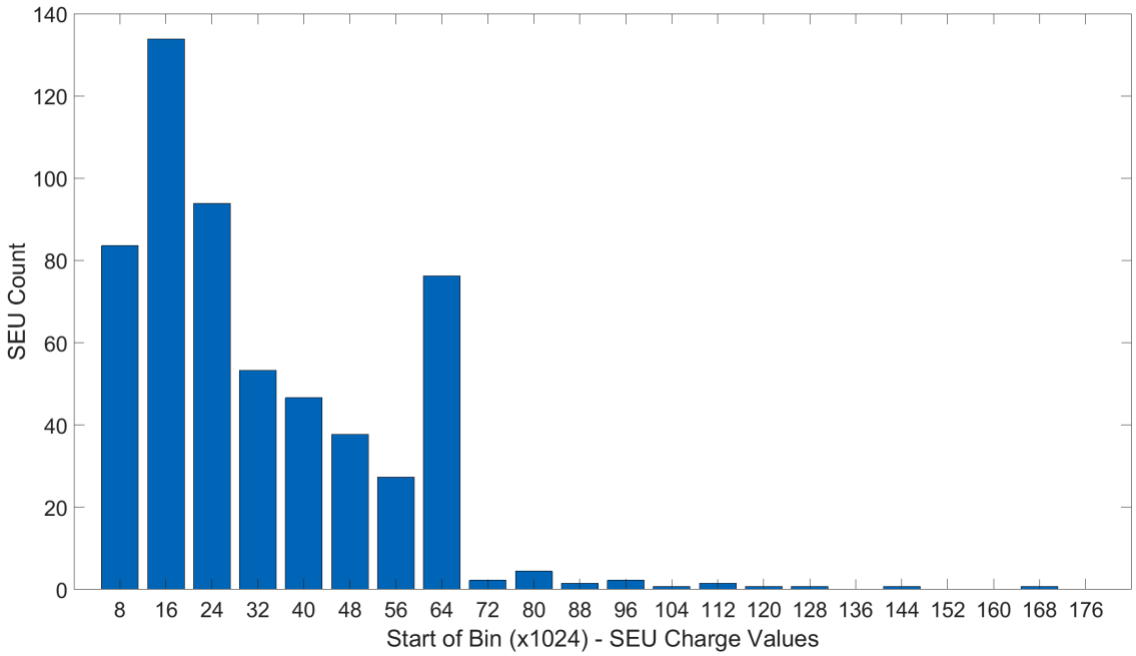


Figure 5.1 Histogram of SEU charge distribution: Canon 5DS R (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 4.13 μm, Bin Width = 8192)

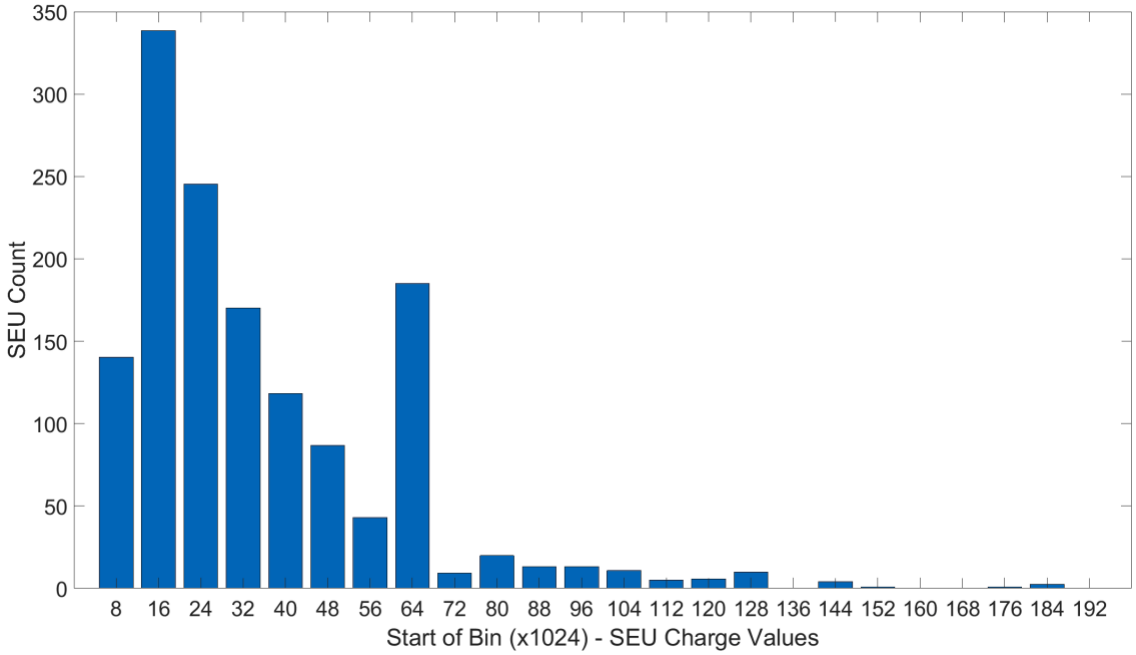


Figure 5.2 Histogram of SEU charge distribution: Canon 5DS R (ISO 6400, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 4.13 μm, Bin Width = 8192)

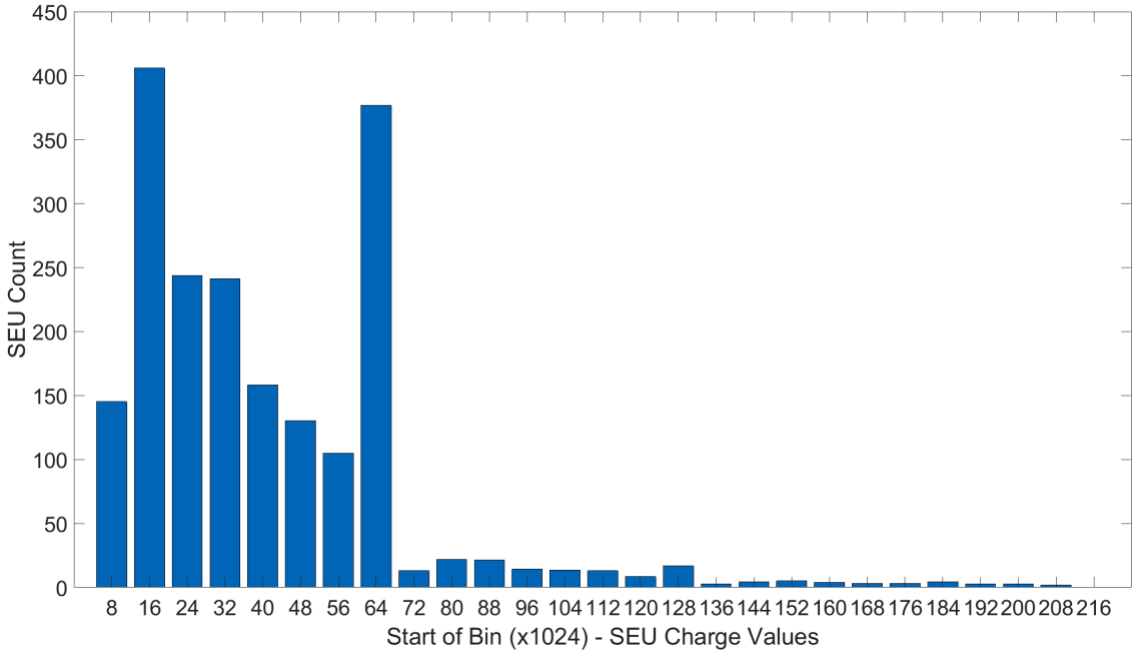


Figure 5.3 Histogram of SEU charge distribution: Canon 5DS R (ISO 12800, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 4.13 μm, Bin Width = 8192)

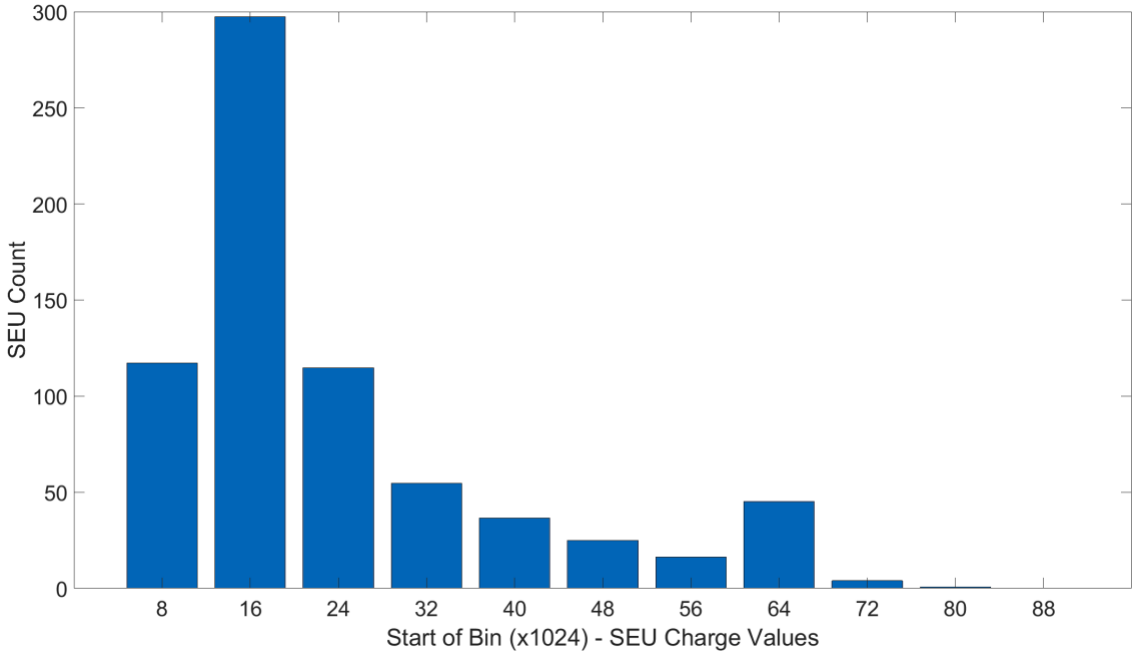


Figure 5.4 Histogram of SEU charge distribution: Canon 5D Mark II (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm, Bin Width = 8192)

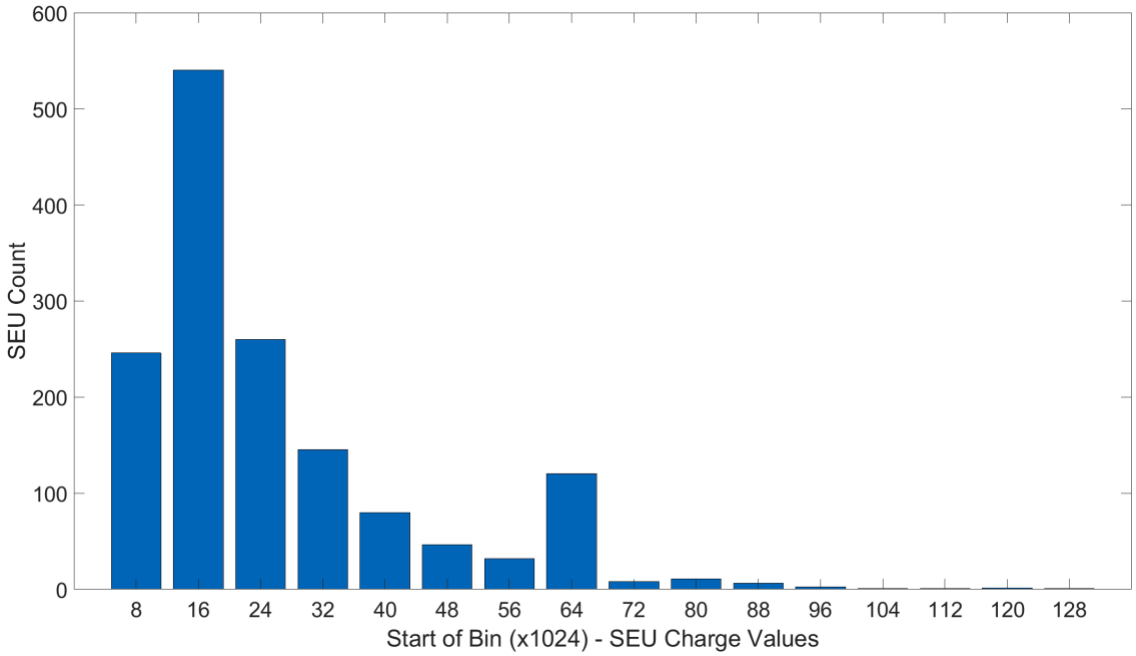


Figure 5.5 Histogram of SEU charge distribution: Canon 5D Mark II (ISO 6400, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm, Bin Width = 8192)

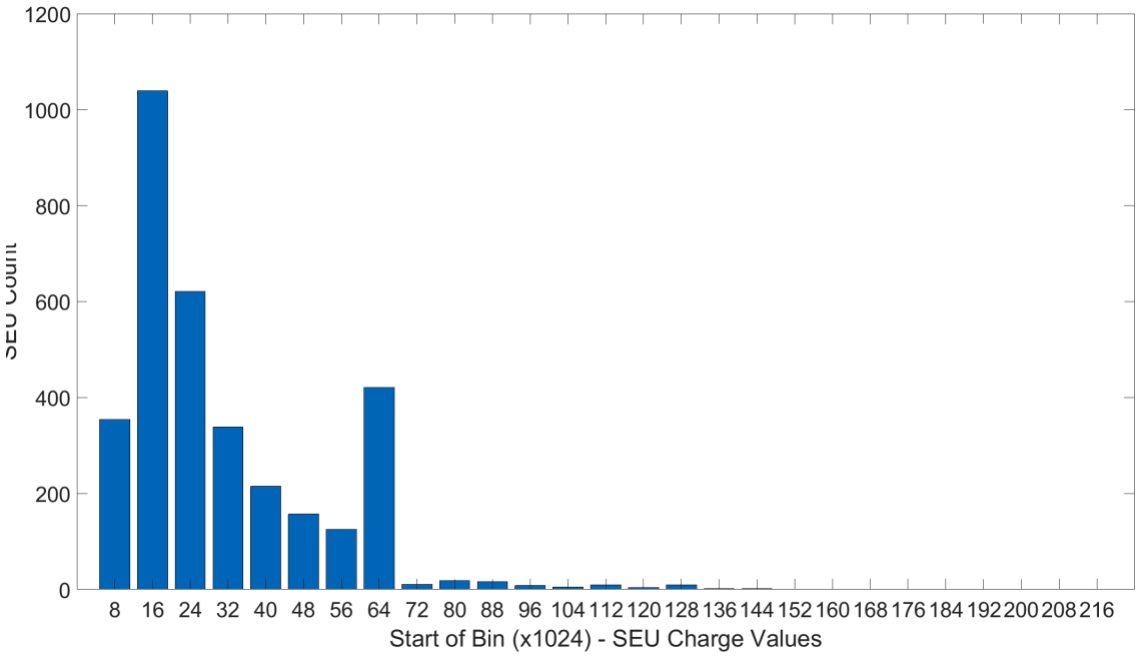


Figure 5.6 Histogram of SEU charge distribution: Canon 5D Mark II (ISO 12800, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm, Bin Width = 8192)

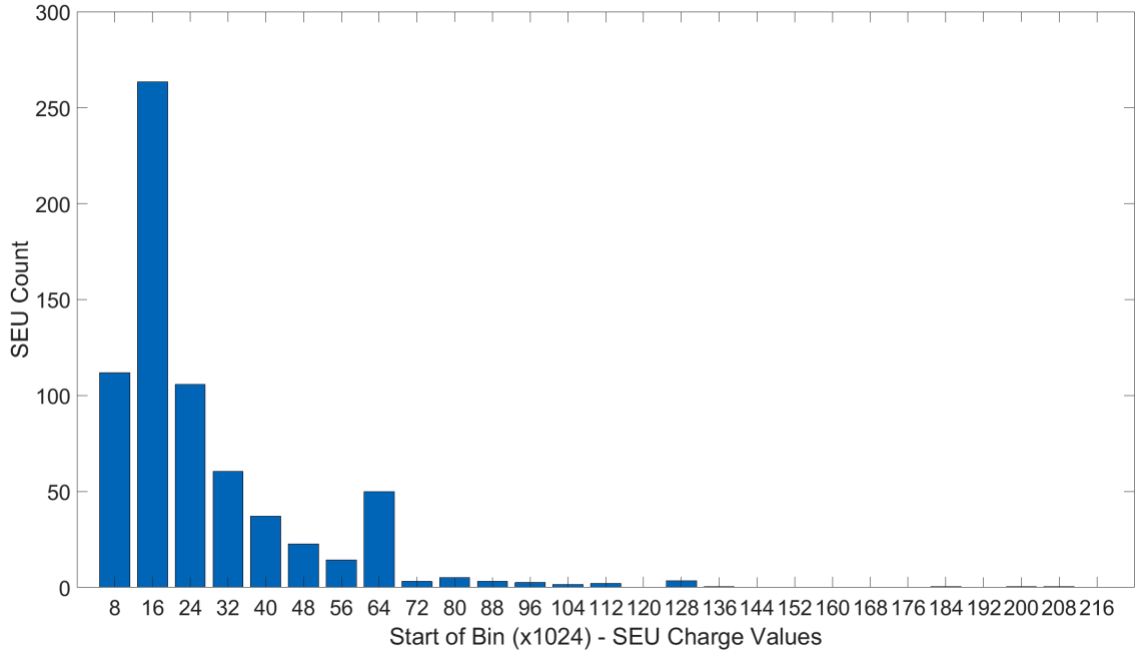


Figure 5.7 Histogram of SEU charge distribution: Canon T2i (ISO 3200, T = 30s, Sensor Size: 22.3 mm x 14.9 mm, Pixel Size = 4.29 μm, Bin Width = 8192)

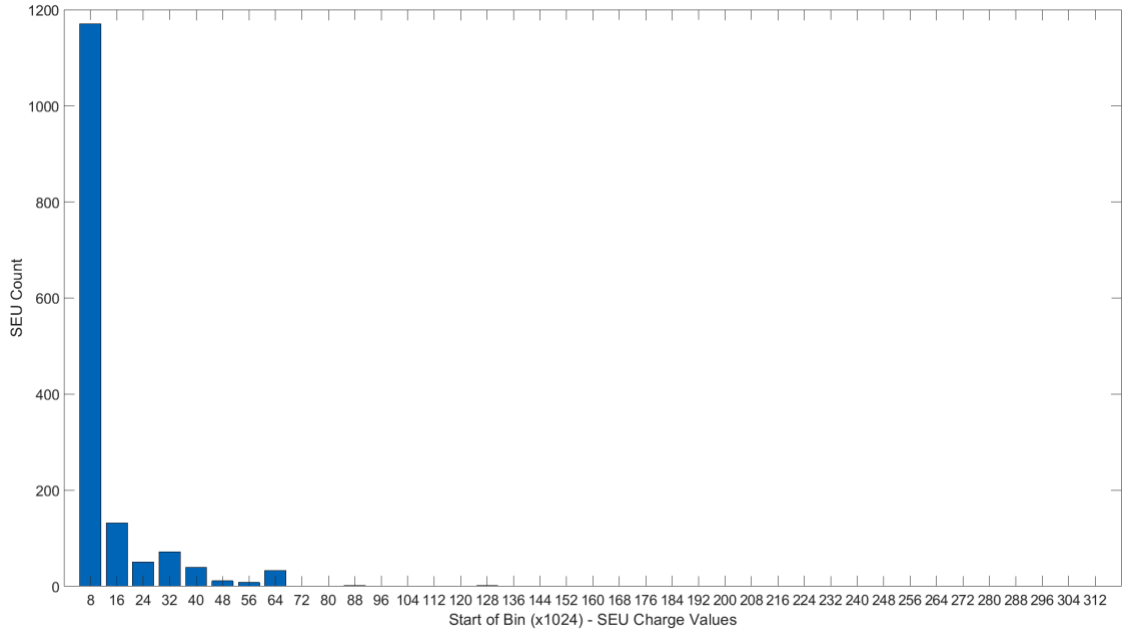


Figure 5.8 Histogram of SEU charge distribution: Canon T2i (ISO 6400, T = 30s, Sensor Size: 22.3 mm x 14.9 mm, Pixel Size = 4.29 μm, Bin Width = 8192)

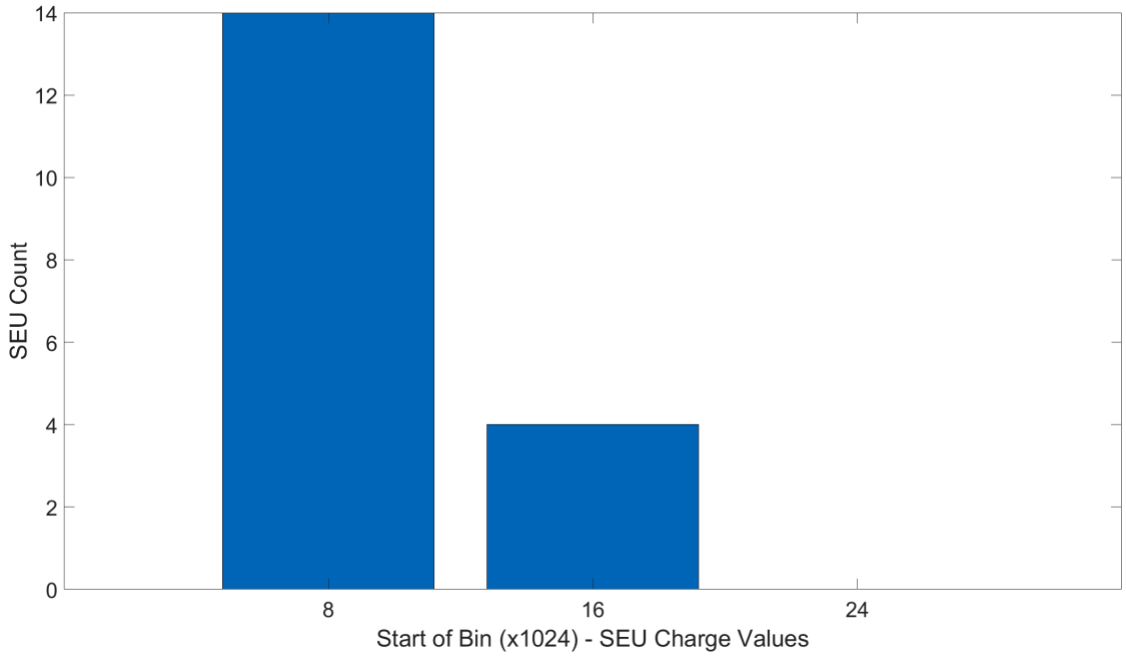


Figure 5.9 Histogram of SEU charge distribution: Samsung S7 [Phone A] (ISO 400, T = 4s, Sensor Size: 5.76 mm x 4.29 mm, Pixel Size = 1.4 μm, Bin Width = 8192)

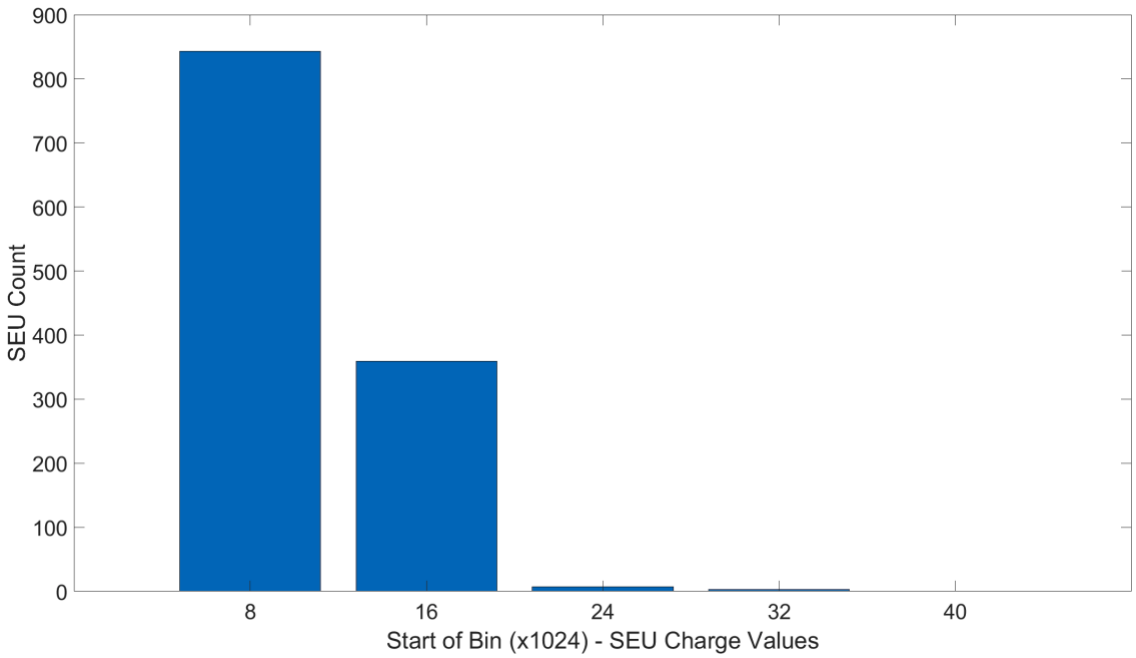


Figure 5.10 Histogram of SEU charge distribution: Samsung S7 [Phone A] (ISO 800, T = 4s, Sensor Size: 5.76 mm x 4.29 mm, Pixel Size = 1.4 μm, Bin Width = 8192)

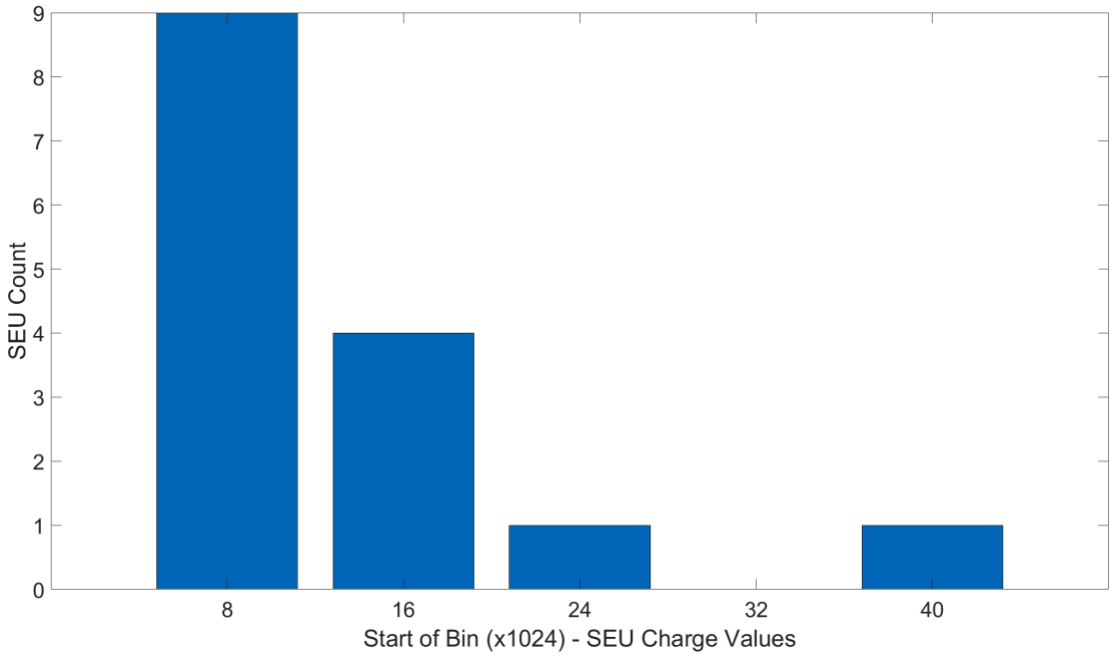


Figure 5.11 Histogram of SEU charge distribution: Samsung S7 [Phone B] (ISO 400, T = 4s, Sensor Size: 5.76 mm x 4.29 mm, Pixel Size = 1.4 μm, Bin Width = 8192)

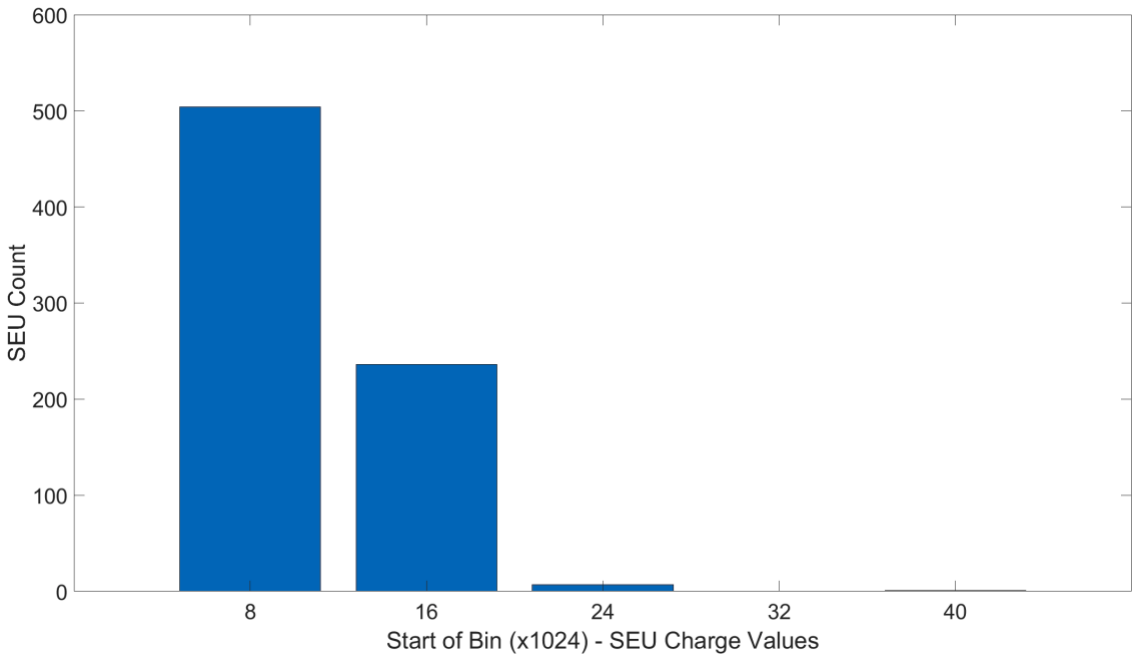


Figure 5.12 Histogram of SEU charge distribution: Samsung S7 [Phone B] (ISO 400, T = 4s, Sensor Size: 5.76 mm x 4.29 mm, Pixel Size = 1.4 μm, Bin Width = 8192)

By analyzing the charge distributions of the various camera models in Figures 5.1 through 5.12, we can see that each camera has its own unique distribution pattern. However, there are two characteristics that are common among all imagers. The first being that their charge distributions get wider as the ISO level increases. In other words, there are higher SEU tail values as the ISO level increases as well as higher bin counts. This is because as the ISO level increases, the gain on the pixel increases. As a result, they become extra sensitive to both cosmic rays and SEUs. The longer tails represent higher bins which are essentially stronger SEUs that we can detect at higher ISOs. We also know that higher ISOs are more susceptible to noise, however, this is less of a concern with the recent noise reduction optimizations of the Pixel Address Distribution method.

The second inherent characteristic found in the DSLR charge distribution histograms (Figures 5.1 through 5.8) is that they have two relative maximums. They reach an absolute maximum at the second lowest charge bin (start of bin: 16x1024), then decline almost exponentially until another relative maximum at the 8th bin from the left (start of bin: 64x1024) and experience a long tail after that. Surprisingly, the relative maximum at the 8th bin does not shift to another bin as the ISO level increases. Instead the count of the 8th

bin increases by at least 100% while the count of other bins also increases. The first peak is centered around the 2nd bin (start of bin: 16x1024) and the second peak is centered around the 8th bin (start of bin: 64x1024). An interesting observation is that the two peaks never shift and are always at the 2nd and 8th bins irrespective of the ISO level. It is very unclear what is causing this shape in the distribution. Hence, future research will have to explore this further. The cell phone charge distributions are rather different than the DSLR charge distributions. The cell phone charge distributions only have a single peak and have fairly narrower distributions. Their highest bin range was the 5th bin from the left (start of bin: 40x1024) compared to the highest bin of 312x1024 for the DSLR in Figure 5.8 – an approximately 87% decrease. This characteristic can be attributed to the fact that cell phones experience weaker SEUs than DSLRs.

5.4. Relationship: SEU rate and Elevation

In Chapter 1 (Section 1.2.1), we introduced the concept of cosmic rays, their impact on ICs, and their role in creating defects. Another concept that we promised to revisit was temporal defects in digital cameras at various elevations – primarily to understand the relationship with elevation. Since the location of Vancouver, BC, Canada is very advantageous with a wide range of elevations, we were able to test four different elevations (24 m, 74 m, 117 m, 366 m) with two DSLR cameras (Canon 5D Mark II and Canon T2i). An additional dataset was captured with the Canon 5D Mark II at an elevation of 1088 m. Something to point out is that the dark-frame images at 24m were captured in the city of Maple Ridge, BC, Canada and the dark-frame images at 1088m were captured at Grouse Mountain in the city of North Vancouver, BC, Canada. The dark-frame images at the other elevations (74 m, 117 m, 366 m) were captured in the city of Burnaby, BC, Canada. Since Maple Ridge and Grouse Mountain are approximately 60 km apart from each other with Burnaby in between, there may be a difference in the amount of cosmic radiation experienced in the three locations. It can be observed that the elevation test in this thesis only leverages DSLRs. The reason cell phones were omitted from this experiment is because it was difficult to share equipment during the global pandemic of COVID-19. For the elevation experiment, Table 5.8 lists the image datasets (consisting of

1000 dark-frame images) that we captured and analyzed using the Pixel Address Distribution method.

Table 5.8 Dark-frame Image Datasets for Elevation Analysis (ISO 3200, T = 30s)

#	Camera Model	Elevation (m)
1	Canon 5D Mark II	24
2	Canon 5D Mark II	74
3	Canon 5D Mark II	117
4	Canon 5D Mark II	366
5	Canon 5D Mark II	1088
6	Canon T2i	24
7	Canon T2i	74
8	Canon T2i	117
9	Canon T2i	366

After capturing the image datasets listed above (~9000 dark-frame images in total), they were analyzed using the Pixel Address Distribution algorithm. To easily understand the relationship of elevation and SEU count, the results were plotted on a graph as 'Total SEU Count per cm^2 in 1000 images' against elevation (m). This plot can be seen in Figure 5.13.

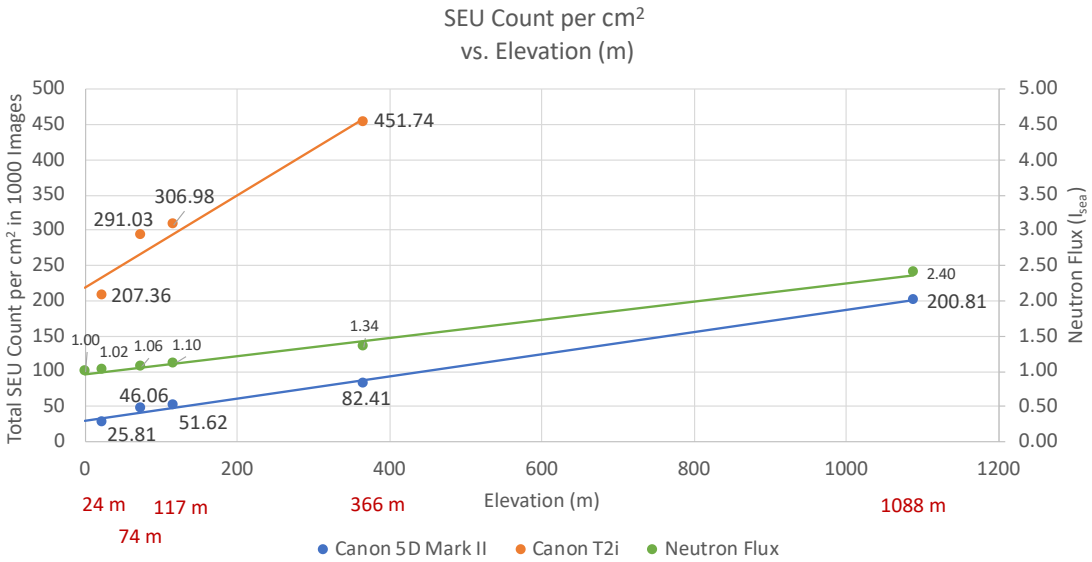


Figure 5.13 SEUs per cm^2 (or Neutron Flux) vs. Elevation (m) (ISO 3200, T = 30s)

Figure 5.13 depicts the SEU counts per cm^2 at four elevations – 24 m, 74 m, 117 m, 366 m – for the Canon 5D Mark II and Canon T2i. An additional elevation (1088 m) is shown for the Canon 5D Mark II. The graph displays that both cameras demonstrate a

linear relationship between SEU count per cm^2 and elevation. Please note that the y-axis for the SEU count per cm^2 is represented by the primary y-axis. Based on the slopes of the two cameras (in a dataset of 1000 images), the Canon 5D Mark II adds ~0.16 SEUs per cm^2 per added metre in elevation and the Canon T2i adds ~0.65 SEUs per cm^2 per added metre in elevation. When analyzing the Canon 5D Mark II further, it can be seen that a 50 m increase in elevation (from 24 m to 74 m) causes a ~78.5% increase in the number of SEUs per cm^2 seen in 1000 dark-frame images. The next increase of 43 m in elevation (from 74 m to 117 m) causes a ~12.1% increase in the number of SEUs per cm^2 seen in 1000 dark-frame images. A 249 m increase in elevation (from 117 m to 366 m) causes a ~59.6% increase in the number of SEUs per cm^2 seen in 1000 dark-frame images. Lastly, a 722 m increase in elevation (from 366 m to 1088 m) causes a ~143.7% increase in the number of SEUs per cm^2 seen in 1000 dark-frame images. Similar percent increases can be seen with the Canon T2i. A ~40.4% increase going from 24 m to 74 m, a ~5.5% increase going from 74 m to 117 m, and a ~47.2% increase going from 117 m to 366 m. It has always been understood that SEU rates in ICs were higher at high elevations, however, this data proves that even a modest elevation increase – say 50 m – has a significant impact. Backed by these results, we can state that high ISO levels and high elevations can lead to 3 to 4 SEUs per cm^2 per dark-frame image.

Additionally, there is one more curve that is plotted on the graph – the theoretical neutron flux at various elevation points. This curve is based on equation (1.1) from Chapter 1 and the relative values of the neutron flux are presented in terms of I_{sea}. Please note that the y-axis for neutron elevation is represented by the secondary y-axis. By analyzing this curve, it can be seen that in the elevation range from sea level to 1200 m there is a linear relationship between the neutron flux and elevation. Even though the neutron flux curve depicts a linear relationship with elevation, its slope is a lot lower than the SEU count increase that was observed. For example, there is a ~3.9% increase in relative neutron flux (I_{sea}) going from 24 m to 74 m, compared to the Canon 5D Mark II increasing by ~78.5% for the same elevation change. There are ~3.7% and ~21.8% increases in relative neutron flux going from 74 m to 117 m and 117 m to 366 m respectively. This is in contrast to the ~12.1% and ~59.6% increases that are seen by the Canon 5D Mark II for the same respective elevation changes. Lastly, there is a ~79.1% increase in relative neutron flux going from 366 m to 1088 m, compared to the Canon 5D Mark II increasing by ~143.7% for the same elevation change. In fact, we are seeing traditional IC level changes in the

case of DSLRs. The recorded data shows us both the event and amount of the charge being deposited. By this, we can conclude that the SEU occurrences in digital cameras scale linearly with increasing elevation – even with modest changes in elevation. As the altitude rises, we see more and more weaker neutrons that are creating SEUs. Specific to imagers is that weaker events are getting noticed by the camera which go unnoticed in other IC systems. As a result, cameras are perfectly suited to view weaker SEU events than traditional ICs.

Since the SEU data is more intuitive when represented as occurrence rates, Table 5.9 and 5.10 list the SEU rates per second (λt) and SEU rates per second per area (λt/cm^2), respectively.

Table 5.9 DSLR SEU rate per second at different elevations (ISO 3200, T=30s)

Camera Model	Sensor Size (mm^2)	Rate per s (λt) at 24 m	Rate per s (λt) at 74 m	Rate per s (λt) at 117 m	Rate per s (λt) at 366 m	Rate per s (λt) at 1088 m
Canon 5D Mark II	36 x 24	0.0074	0.0133	0.0149	0.0237	0.0578
Canon T2i	22.3 x 14.9	0.0230	0.0322	0.0340	0.0500	

Table 5.10 DSLR SEU rate per second per area at different elevations (ISO 3200, T=30s)

Camera Model	Sensor Size (mm^2)	Rate per s per cm^2 (λt/cm^2) at 24 m	Rate per s per cm^2 (λt/cm^2) at 74 m	Rate per s per cm^2 (λt/cm^2) at 117 m	Rate per s per cm^2 (λt/cm^2) at 366 m	Rate per s per cm^2 (λt/cm^2) at 1088 m
Canon 5D Mark II	36 x 24	0.0009	0.0015	0.0017	0.0027	0.0067
Canon T2i	22.3 x 14.9	0.0069	0.0097	0.0102	0.0151	

As we did for the earlier datasets, we ran seu_charge_analysis.m to generate charge histograms for the image datasets at different elevations. Figures 5.14 through 5.18 contain the charge distribution histograms for the Canon 5D Mark II at 24 m, 74 m, 117 m, 366 m, 1088 m respectively and Figures 5.19 through 5.22 contain the charge distribution histograms for the Canon T2i at 24 m, 74 m, 117 m, 366 m respectively.

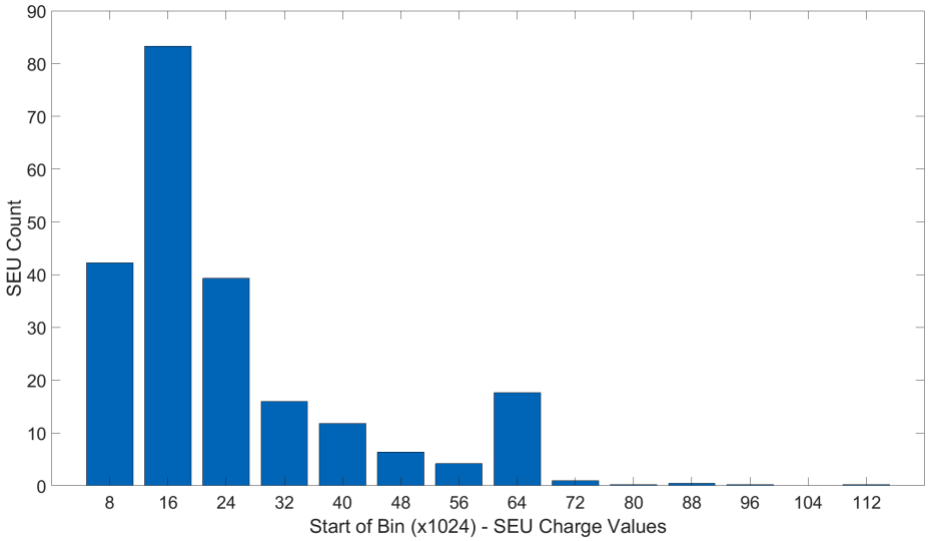


Figure 5.14 Histogram of SEU charge distribution: Canon 5D Mark II, 24 m (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm, Bin Width = 8192)

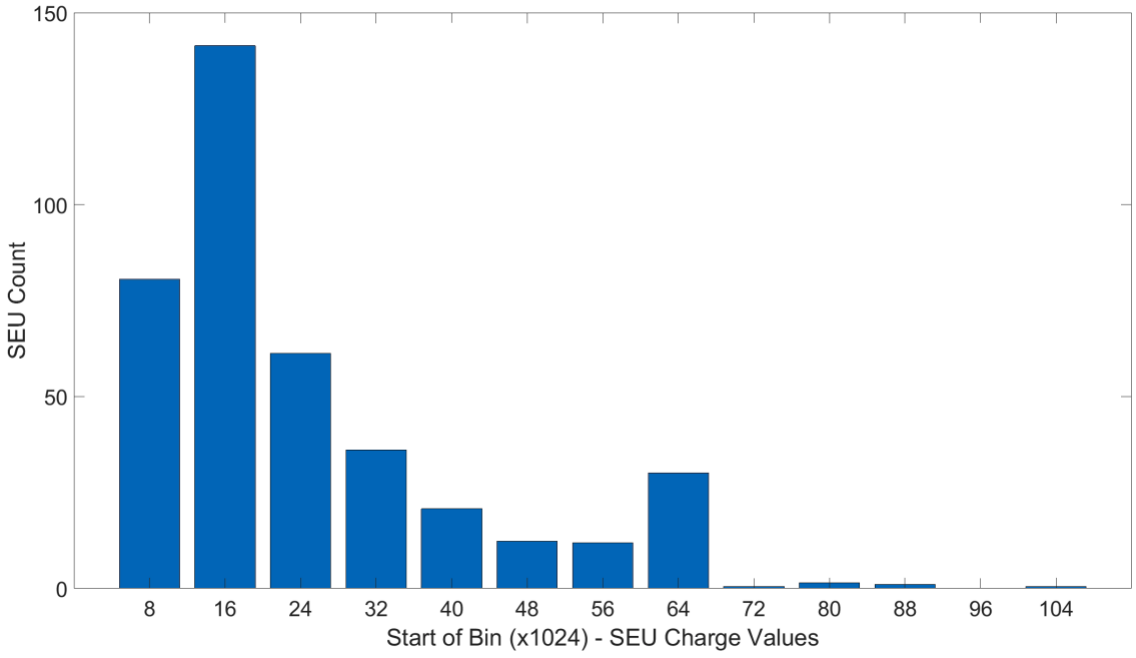


Figure 5.15 Histogram of SEU charge distribution: Canon 5D Mark II, 74 m (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm, Bin Width = 8192)

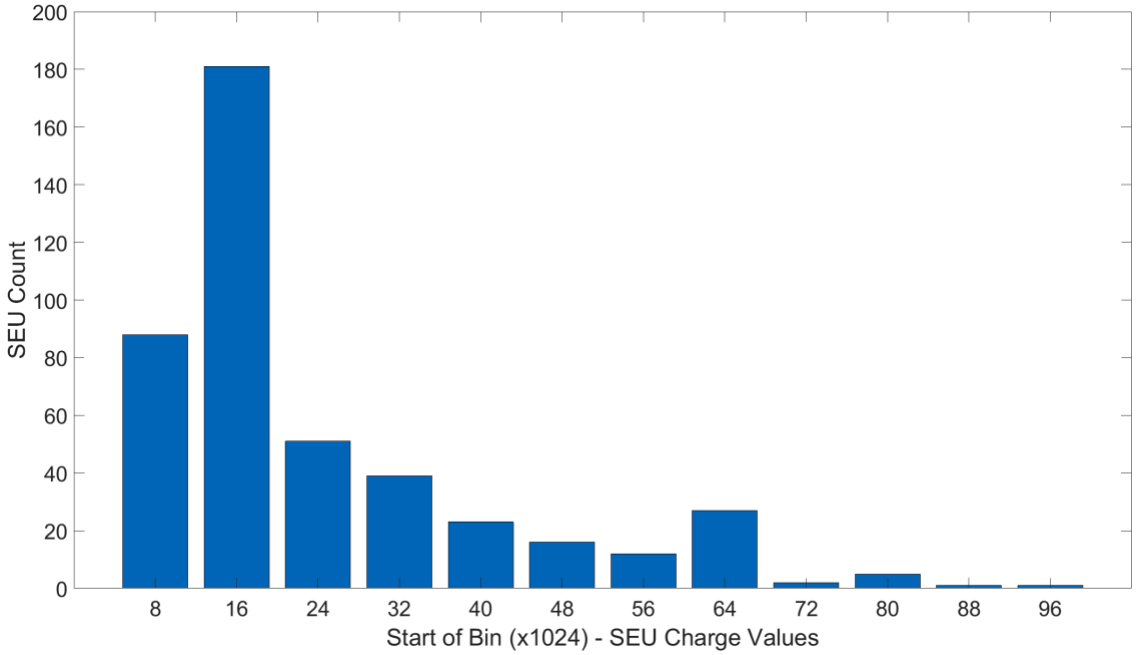


Figure 5.16 Histogram of SEU charge distribution: Canon 5D Mark II, 117 m (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm, Bin Width = 8192)

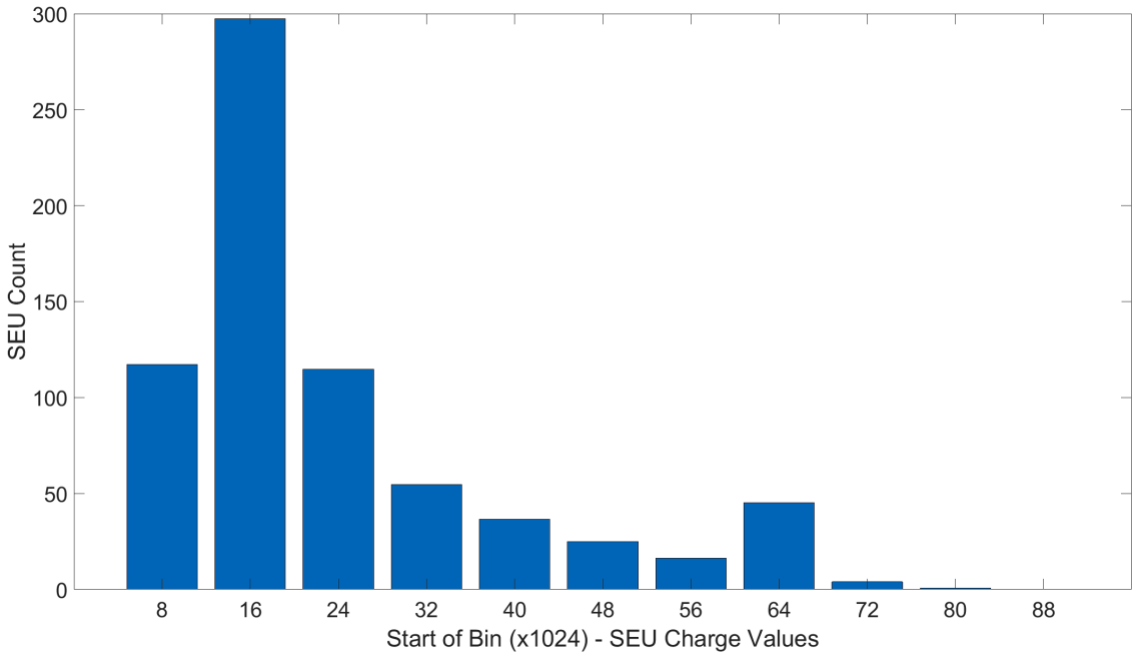


Figure 5.17 Histogram of SEU charge distribution: Canon 5D Mark II, 366 m (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm, Bin Width = 8192)

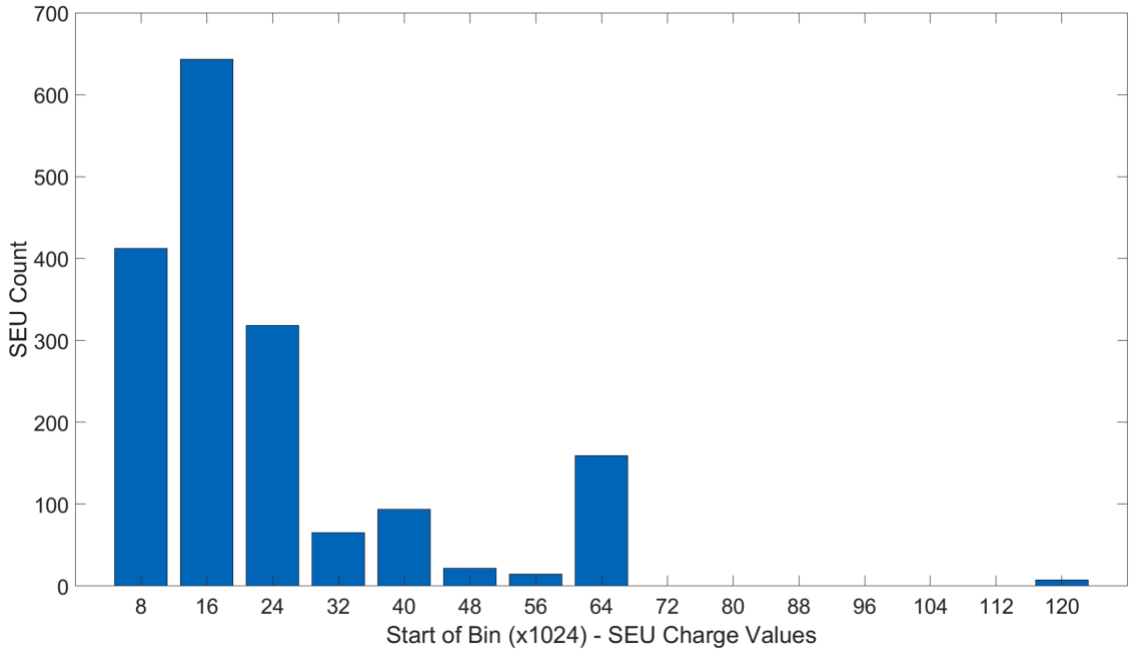


Figure 5.18 Histogram of SEU charge distribution: Canon 5D Mark II, 1088 m (ISO 3200, T = 30s, Sensor Size: 36 mm x 24 mm, Pixel Size = 6.41 μm, Bin Width = 8192)

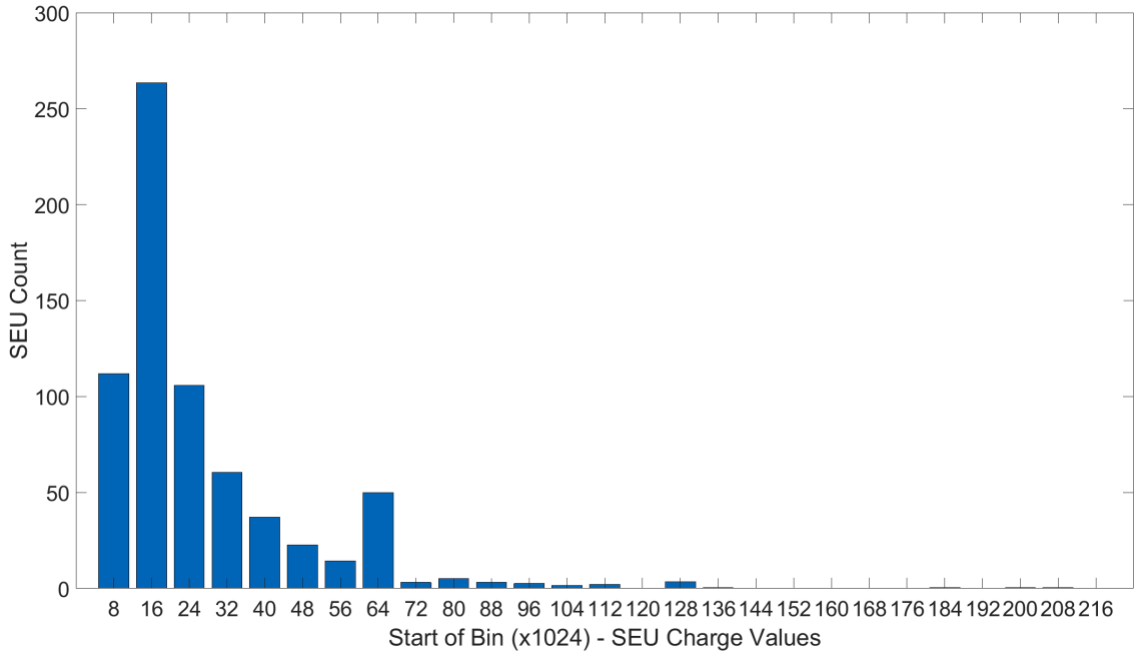


Figure 5.19 Histogram of SEU charge distribution: Canon T2i, 24 m (ISO 3200, T = 30s, Sensor Size: 22.3 mm x 14.9 mm, Pixel Size = 4.13 μm, Bin Width = 8192)

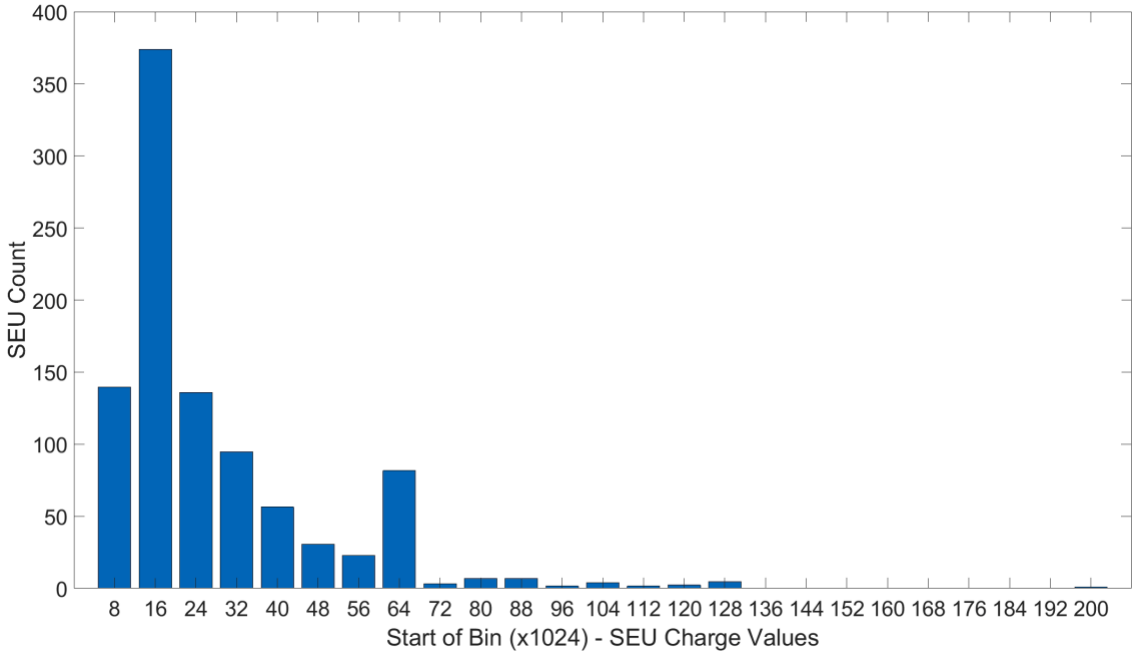


Figure 5.20 Histogram of SEU charge distribution: Canon T2i, 74 m (ISO 3200, T = 30s, Sensor Size: 22.3 mm x 14.9 mm, Pixel Size = 4.13 μm, Bin Width = 8192)

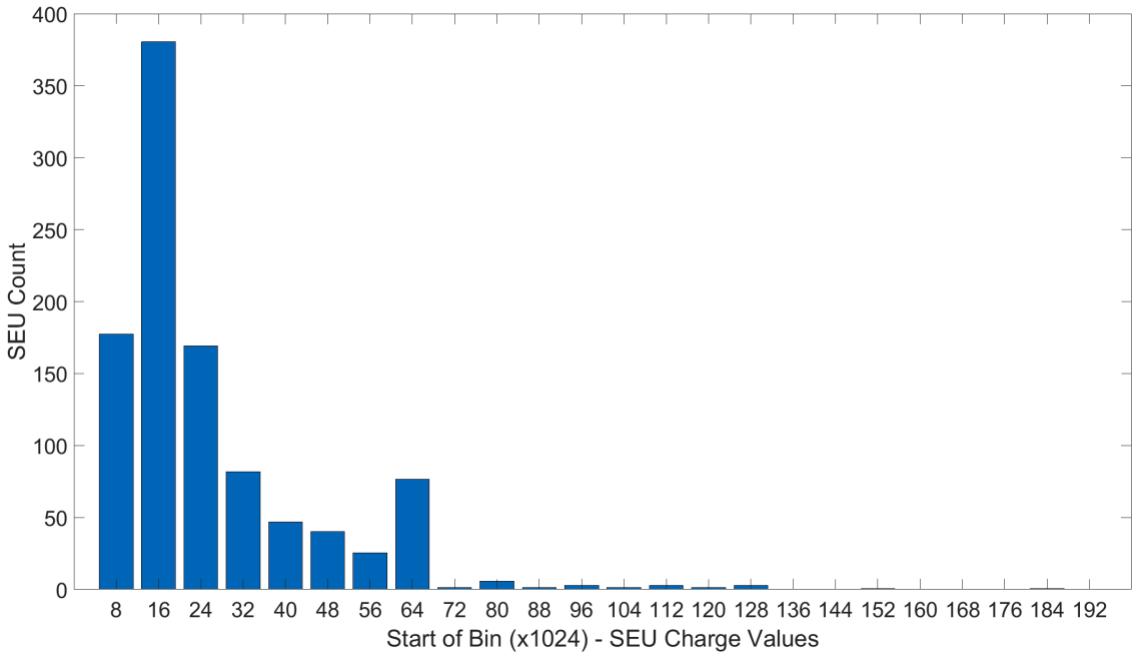


Figure 5.21 Histogram of SEU charge distribution: Canon T2i, 117 m (ISO 3200, T = 30s, Sensor Size: 22.3 mm x 14.9 mm, Pixel Size = 4.13 μm, Bin Width = 8192)

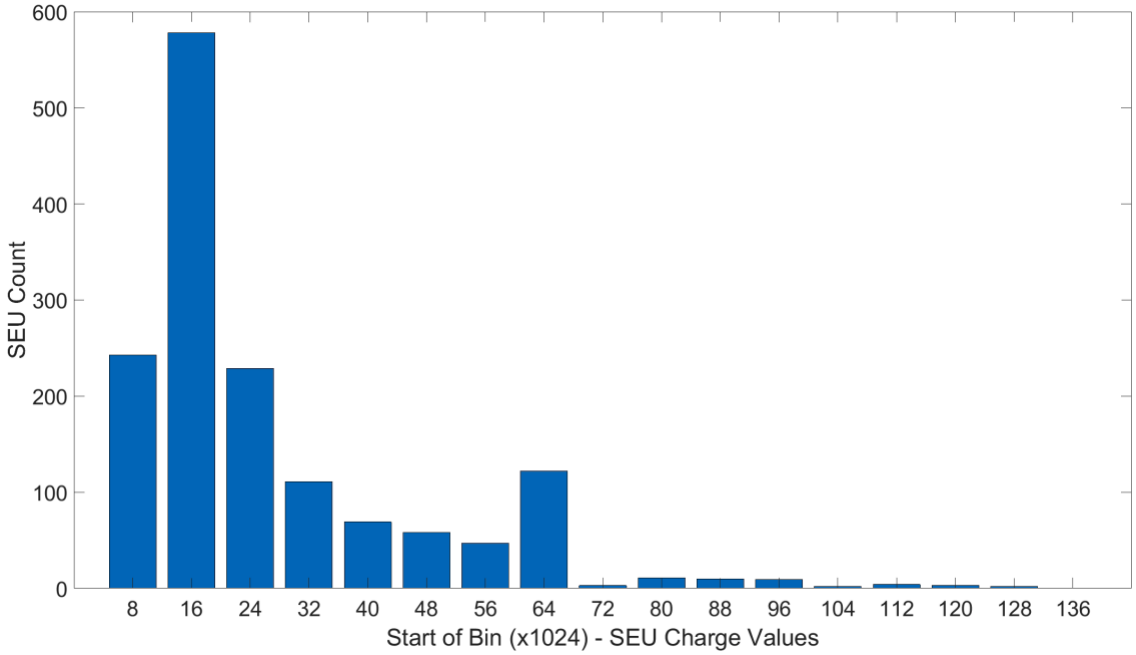


Figure 5.22 Histogram of SEU charge distribution: Canon T2i, 366 m (ISO 3200, T = 30s, Sensor Size: 22.3 mm x 14.9 mm, Pixel Size = 4.13 μm, Bin Width = 8192)

The charge distribution histograms for each camera at different elevations show that the overall pattern of the charge distribution persists at different elevations. The important characteristic to pay attention to is the length of the distribution's tail. At lower elevations, the tails are longer which mean there are stronger SEUs in the overall charge distribution. However, as seen earlier there are less SEUs overall at lower elevations. As elevation increases, the tails become shorter which mean there are weaker SEUs in the overall charge distribution despite there being more SEUs overall. For example, at an elevation of 24 m the highest charge bin for the Canon 5D Mark II is the 14th bin from the left (start of bin: 112x1024) and at an elevation of 366 m the highest charge bin is the 11th bin (start of bin: 88x1024). Essentially, the highest charge bin at an elevation of 366 m is 3 charge bins lower than the highest charge bin at an elevation of 24 m. The similar behaviour can be seen with the Canon T2i. The highest charge bin at an elevation of 24 m is the 27th bin (start of bin: 216x1024) and the highest charge bin at an elevation of 366 m is the 17th bin (start of bin: 136x1024). Essentially, the highest charge bin at an elevation of 366 m is 10 charge bins lower than the highest charge bin at an elevation of 24 m. The only case where this is not true is for the Canon 5D Mark II at 1088 m. At this elevation the charge tail is longer than it is at 366 m. It is very unclear what is causing this behaviour. Hence, future research will have to explore this further. The cause of this bigger gap may

be a correlation to the pixel size and design. A future investigation would be to perform the elevation test with cell phone cameras which have smaller pixels. Overall, we are seeing through these sets of elevation tests that we have developed a very stable detection method that is able to record repeatable increases in elevation. The Pixel Address Distribution algorithm is essentially as system that can accurately measure subtle differences such as marginal elevation changes – e.g. 50 m.

5.5. Analysis of SEU Size

This final section in Chapter 5 will cover the size of single pixel SEU charges in digital imagers based on our prior research from Chapman [52]. In the current research we have seen that SEU streaks (caused by muons) in digital cameras account for less than 5% of the total SEU count in a given image dataset. The percentage slightly varies due to a number of factors such as pixel size, ISO level, and elevation. Other than the case of streaks and clusters, SEU charges in digital camera dark-frame images were never found crossing two adjacent pixels. This observation causes the size of the initial SEU charge ball to be very small in comparison to the size of the single pixel it occurs inside. The ability to experimentally measure the actual size of the SEU fault is not possible. However, we leverage statistical methods on current SEU data to achieve a confident approximation of the SEU charge ball size. As previously assumed, they are fairly small in size in comparison to the pixel itself.

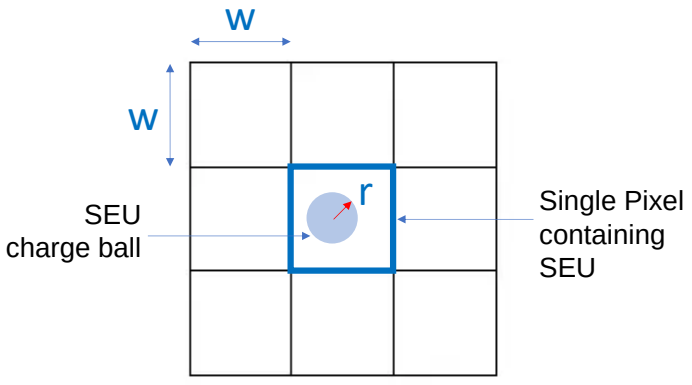


Figure 5.23 Defect model of SEU charge ball in 3x3 pixel area (taken from Chapman [52])

The first step in the statistical method is to create a model of the digital imager with pixels of w x w dimension – where w is measured in microns. This is illustrated in Figure 5.23. The initial SEU charge shape is that of a circle with a random radius, R. Since the entire pixel sensor is assumed to be sensitive to charge, a charge ball landing in the pixel area will cause the formation of a SEU and will spread to the entire pixel as the pixel value is read out at the time of image capture. As mentioned earlier, an SEU spanning a single pixel is known as a single pixel SEU – or in other words, the entire SEU charge remains within the w x w dimensions of the pixel it resides inside.

As mentioned in Chapter 3, an SEU cluster would occur when the SEU charge goes outside the w x w dimensions of the single pixel and spreads to the adjacent pixels. In order for an SEU (with radius $R = r$) to be classified as a single pixel SEU, its center would have to be within the safe area depicted in Figure 5.24.

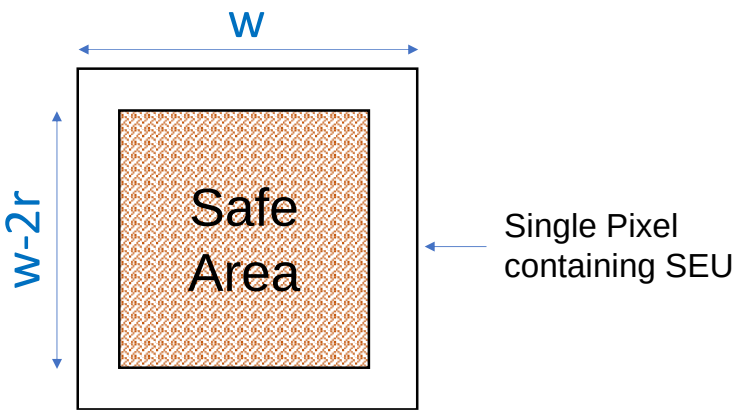


Figure 5.24 Safe area within pixel for SEU charge ball (taken from Chapman [52])

Referring to Figure 5.24, we can see that the safe area for the SEU charge ball is a square with its width and height equal to $w - 2r$ and its center aligned with the center of the pixel. The probability of an isolated SEU (with radius $R = r$) developing uniformly at any location in the pixel would be given by (5.2).

$$Pr(Isolated\ Single\ Pixel\ SEU \mid R = r) = \begin{cases} \left(1 - \frac{2r}{w}\right)^2 & r < \frac{w}{2} \\ 0 & r \geq \frac{w}{2} \end{cases} \qquad (5.2)$$

Building on (5.2), let us now consider N recognized single pixel SEUs and assume that all N single pixel SEUs have been caused by SEU charges with identical radii, $R = r$. This assumes for the worst-case scenario in which all radii of N equal the largest possible radius of the N single pixel SEUs. The probability of all N single pixel SEUs (with radius $R = r$) being isolated would be given by:

$$Pr(N\ Single\ Pixel\ SEUs \mid R = r) = \begin{cases} \left(1 - \frac{2r}{w}\right)^{2N} & r < \frac{w}{2} \\ 0 & r \geq \frac{w}{2} \end{cases} \tag{5.3}$$

In order to calculate the maximum value for the radius (r_{MAX}), Bayes theorem is leveraged in (5.4). This equation evaluates the probability of the charge causing the SEU being less than the upper bound, r_{MAX}.

$$\Pr\left(R \leq r_{MAX} \mid N\ Single\ Pixel\ SEUs\right) = \frac{\int_0^{r_{MAX}} \left(1 - \frac{2r}{w}\right)^{2N} f(r)dr}{\int_0^{\frac{w}{2}} \left(1 - \frac{2r}{w}\right)^{2N} f(r)dr} \tag{5.4}$$

In equation (5.4), $f(r)$ represents an assumed probability density function (PDFs) for the SEU charge ball radius. There are many common PDFs that can be used, however, the most popular is uniform distribution – with equal likelihood of any radius in the range $\left[0, \frac{w}{2}\right]$. The uniform distribution can be seen in (5.5).

$$f_{uniform}(r) = \begin{cases} \frac{1}{w/2} & 0 \leq r \leq \frac{w}{2} \\ 0 & otherwise \end{cases} \tag{5.5}$$

The lower and upper bounds in (5.5) were chosen to consider infinitesimally small 'point' charge balls and the largest possible charge balls respectively. In order to conservatively calculate the maximum SEU charge diameter radius, we equate (5.4) to a confidence level of 99%, substitute (5.5) for $f(r)$ in (5.4) and solve for $\frac{2r_{MAX}}{w}$. If CL=100% were used, it would result in $\frac{2r_{MAX}}{w} = 1$ – not very useful. Since $\frac{2r_{MAX}}{w}$ is left as a normalized value, we can apply it to any camera sensor as it does not depend on a specific pixel size.

Figure 5.25 illustrates the graph of $\frac{2r_{MAX}}{w}$ against increasing N when using uniform distribution. Since the graph is using a 99% confidence level, for N > 200 there is a 99% certainty that $\frac{2r_{MAX}}{w} < 1.15\%$ of the pixel width. Since the distribution saturates for N > 200, $\frac{2r_{MAX}}{w} < 1.15\%$ is valid for the 1000's of SEUs that we observed in a single image dataset. Setting a confidence level of 99% instead of 100% is a very crucial step in solving for $\frac{2r_{MAX}}{w}$.

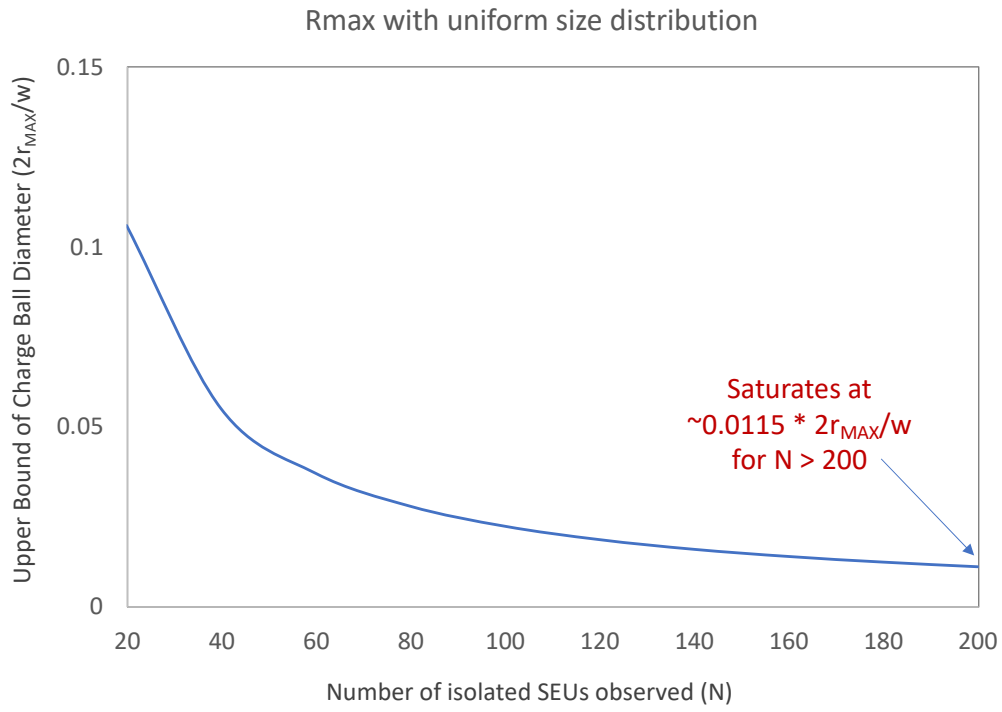


Figure 5.25 Uniform distribution: Upper Bound of charge ball diameter $\left(\frac{2r_{MAX}}{w}\right)$ vs. Number of isolated SEUs (N)

Another more realistic PDF to consider when dealing with SEUs in ICs is an exponential distribution with a radius range of $[0, \infty)$. The exponential distribution can be seen in (5.6) where the rate parameter k is allowed to vary according to the desired spread in radii.

$$f_{exponential}(r) = \begin{cases} k \cdot e^{-k \cdot r} & r \geq 0 \\ 0 & otherwise \end{cases} \quad (5.6)$$

As done for the uniform distribution, to conservatively calculate the maximum SEU charge diameter radius, we equate (5.4) to a confidence level of 99%, substitute (5.6) for $f(r)$ in (5.4) and solve for $\frac{2r_{MAX}}{w}$. Again, setting a confidence level of 99% instead of 100% is a very crucial step in solving for $\frac{2r_{MAX}}{w}$. Also, since $\frac{2r_{MAX}}{w}$ is left as a normalized value, we can apply it to any camera sensor as it does not dependent on a specific pixel size. Figure 5.26 illustrates the graph of $\frac{2r_{MAX}}{w}$ against increasing N using exponential distribution.

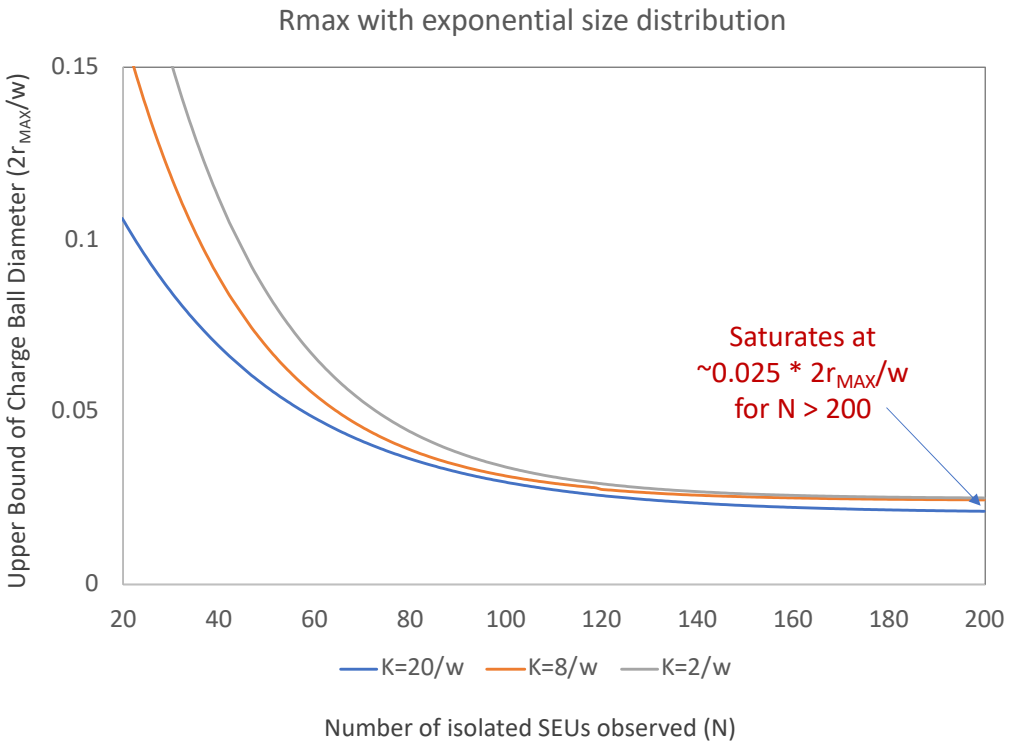


Figure 5.26 Exponential distribution: Upper Bound of charge ball diameter $\left(\frac{2r_{MAX}}{w}\right)$ vs. Number of isolated SEUs (N)

Figure 5.26 shows the upper bound of the charge ball diameter against an increasing number of isolated SEUs (N) for three different values of k. By comparing the three values of k it can be seen that $k = 2/w$ almost resembles a uniform distribution where both smaller and larger temporal defects are probable. Larger values such as $k = 20/w$ depict a very lopsided distribution as only small defects are probable. A similarity between all three k values is that their respective exponential distributions converge after

a large number of isolated SEUs have been identified. Since the graph is using a 99% confidence level, there is a 99% certainty that $\frac{2r_{MAX}}{w} < 2.5\%$ of the pixel width for N >100 and any k. Since the distribution saturates for N > 200, $\frac{2r_{MAX}}{w} < 2.5\%$ is valid for the 1000s of SEUs that we observed in a single image dataset.

From the distributions in Figures 5.25 and 5.26 it can be established that the SEU charge balls are very small 'point' defects in comparison to the pixel size. This is because the values of r_{MAX} are very small. To help us better understand the SEU charge ball size (with CL=99%) let us consider a 4 μm pixel. Using uniform distribution, the SEU charge ball size would be less than 46 nm (for N > 200). Using exponential distribution, the SEU charge ball size would be less than 100 nm (for N > 200). It was indicated previously that most of the digital camera SEUs are single pixel SEUs rather than SEU streaks (< 5%). This leads us to believe that for the most part – excluding streaks – SEUs in ICs are restricted to single transistors or a small number of transistors. In order words they do not extend to large areas or span a large number of ICs.

5.6. Summary

This chapter showed a number of experimental results in relation to SEUs in DSLRs at high ISO levels and cell phone cameras at modest ISO levels. The charge distribution of SEUs showed that increasing the ISO level in both DSLRs and cell phone cameras contributed to larger charge distribution tails – in other words, higher intensity or stronger SEUs. This observation was confirmed by the higher ISO levels also contributing to higher SEU occurrence rates. Although it is true that higher ISO settings lead to higher noise levels, noise reduction in the Pixel Address Distribution method has helped discard false positives and identify SEUs even in these noisy settings.

The earlier literature on cosmic ray particles presented in Chapter 1 concentrated our attention to explore the relationship of SEU count and elevation. The experimental data in the elevation range of sea level to 400 m showed that there is indeed a linear relationship between SEU count and increasing elevation. This aligns with the curve that shows the theoretical neutron flux also having a linear relationship with increasing

elevation in the same elevation range of sea level to 400 m. Currently, we have created a very stable detection method that can report impacts from marginal changes in elevation. Further research will determine the impact of running experiments at elevations greater than 1000 m as well as experimenting with cell phone cameras at various elevations.

Lastly, statistical modelling – using uniform and exponential distributions – shows that the SEU charge balls that eventually cause SEU events at pixel readout are very small ‘point’ defects in comparison to the size of the pixel. Although streaks are observed in dark-frame images, they are seldom – less than 5%. This means that SEUs in ICs are restricted to single transistors or a small number of transistors. As a result, it can be understood that SEUs do not extend to large areas or span a large number of ICs.

Chapter 6.

Conclusion

6.1. Summary

The beginning of this thesis studied the emergence of digital cameras and many aspects of their sensors – namely in DSLRs and cell phone cameras. Despite their advantages in terms of quality, it was presented that film cameras have disadvantages in comparison to digital cameras. As a result, they have been out of popularity in favour of digital cameras in the photography market for over a decade. Chapter 2 took a look at the design and functionality of the CMOS active pixel sensor and its integration within the entire digital camera unit. Leading up to and including the schematics and layouts of the pixel sensors gave a low-level understanding of the circuit design. Today, CMOS camera sensors used in digital cameras are able to hold more pixels as technological advancements have enabled the development of smaller pixels, especially in cell phones. This has caused another market trend in photography where the majority of the consumers are satisfied with the convenience and quality of cell phone cameras. Consequently, there is an industry wide requirement to create even smaller pixels, hence achieving a better sensor density. The shrinkage in pixel size does come with downfalls such as lowered resolution and increased noise.

As noted earlier in the thesis, the phenomenon of cosmic ray particles being deposited in ICs can cause them to experience defects in different forms. Some of the more common types of infield defects that they experience are permanent defects and soft temporal defects. In digital imagers, these defects can be further amplified by a variety of factors such as increased ISO and exposure time, colour demosaicing, noise reduction, and image format compression. The goal of the research in this thesis was to explore the detection of temporal defects in digital camera sensors. The particular areas that were not addressed in previous research or literature were regarding cell phone cameras at modest ISO levels and DSLRs at high ISO levels.

As explained in Chapter 3, the term used for these soft temporal defects in ICs is Single Event Upsets. Digital cameras used in the research, have given us the ability to not only record the existence of SEUs in dark-frame images but also reveal their respective deposited charge values. We have shown that unlike permanent defects such as hot pixels, a given SEU will not persist in multiple dark-frame images in a sequence of images but exist temporarily in one dark-frame image before disappearing. There are three categories of SEUs – single pixel SEUs, SEU clusters, and SEU streaks – but the testing data has shown that streaks (although present) account for less than 5% of the total identified SEUs in an image dataset. SEU clusters are not very common and as a result, the majority of identified SEUs are single pixel SEUs. A major focus area of this research was being able to deal with noise and still have the ability to detect SEUs in digital cameras. As expected, high levels of noise were common in DSLRs at high ISO levels. However, cell phone cameras due to their smaller pixels exemplified similar levels of noise even at modest ISO levels. The ingenious dark-frame image capturing processes for both cell phones and DSLRs have proven to be both effective and reliable when creating large image datasets. Once the automated software intervalometers are configured they have the ability to capture large sets of dark-frame images hour over a day without user interruption. This thesis has walked through three SEU detection algorithms in detail – namely (in order from oldest to newest): The Threshold method, the Local Area Distribution method, and the Pixel Address Distribution method. Distributed pixel analysis over large image datasets (e.g. 1000 dark-frame images) has proven to be beneficial on a number of fronts. Firstly, distributed noise analysis of the sensor via the colour noise maps has shown that digital camera sensor noise is quite randomly distributed throughout the entire sensor. Hence, a simple threshold or image noise threshold are not effective techniques as they incorrectly assume noise is equally distributed throughout the sensor. Secondly, distributed pixel analysis has shown that the older SEU detection results contain a lot of false positives such as hot pixels and noisy pixels. Noisy or hot pixels typically follow a Gaussian distribution while a regular SEU pixel should only follow a Poisson distribution. Finally, the Pixel Address Distribution method has proven to be the most effective algorithm in detecting weaker SEUs in noisier environments as well as discarding these false positives in dark-frame images captured by cell phone cameras at modest ISO levels and DSLRs at high ISO levels. Leveraging statistical modelling and calculating both the distributed mean and standard deviation at each pixel address in the sensor area are key elements that make the algorithm very useful.

The accuracy and success of the Pixel Address Distribution method has given us the confidence to explore more advanced SEU characteristics in digital camera sensors. We have seen there is definitely a relationship between SEU occurrence and the ISO level. Earlier we only had the ability to observe the number or rate of SEUs occurring in an image dataset. Now, the charge analysis has given us the ability to see the shape of the charge distribution pattern of all the SEUs in a given image dataset. We have observed that increasing the ISO level widens the charge distribution and also leads to longer tail values. This indicates the ability of the algorithm to detect both weaker SEUs at lower ISOs and stronger SEUs at higher ISOs. Detecting SEUs at ISO levels greater than ISO 1600 for DSLRs and ISO levels greater than ISO 200 for cell phones cameras was nearly impossible before – given the high noise levels. The Pixel Address Distribution method's capability of leveraging statistical distribution to eliminate noise has made this possible. Another characteristic that was explored was the relationship of SEU occurrence with increasing elevation. This is an area that had been untouched in the past – where it pertained to SEUs in digital camera sensors. Previous literature on cosmic radiation hinted to a possible linear relationship in the context of neutron flux and elevation but it was very much unknown in the case of SEUs in ICs. Dark-frame images from four different elevations – 24 m, 74 m, 117 m, 366 m, 1088 m – were recorded and analyzed using the Pixel Address Distribution method in order to help model the elevation relationship. Interestingly, it was observed from the results that SEU occurrence does indeed have a linear relationship with elevation – similarly to neutron flux, albeit with a higher slope. Backed by these results, we can state that high ISO levels and high elevations can lead to 3 to 4 SEUs per dark-frame image. A very significant observation is the remarkable stability of the detection algorithm to analyze and record repeatable increases in elevation. Even subtle differences such as a marginal 50 m change in elevation can be detected by this reliable system. Lastly, we explored modelling the size of the SEU charge ball by using uniform and exponential distributions. Typically charges land within the safe area of a pixel – a square with its width and height equal to $w - 2r$ and its center aligned with the center of the pixel. We are able to predict – with a confidence level of 99% and for a large number of identified SEUs ($N > 200$) – that the upper bound of the charge ball diameter ($\frac{2r_{MAX}}{w}$) is less than 1.15% of the pixel size using the uniform distribution model and less than 2.5% of the pixel size using the exponential distribution model. From this we can imply that SEU charge balls are generally point-like charges. For the most part (excluding streaks), SEUs

in ICs are restricted to single transistors or a small number of transistors. In order words they do not extend to large areas or span a large number of ICs.

6.2. Suggestions for Future Research

This thesis and research have taken a deep dive in presenting and analyzing SEU characteristics in noisy digital cameras – specifically in cell phone cameras at modest ISO levels and DSLRs at high ISO levels. Some of the various dark-frame image datasets that were collected ranged in ISO level and elevation respectively. However, as mentioned in Chapter 3, only selective Android cell phones support digital RAW format. It would be very insightful to capture dark-frame images in digital RAW format with iPhone cameras if and when they enable support for it. We also observed the lack of noise suppression algorithms in the cell phone cameras due to high computation costs which led to extremely high levels of noise even at modest ISO levels. Most likely, manufacturers will begin to introduce less computationally expensive noise suppression algorithms to combat this noise problem. Ideally as the industry progresses, the next steps of this research should be to capture dark-frame images with as many different digital cameras as possible – DSLRs, mirrorless cameras and cell phones. The Pixel Address Distribution algorithm will be able to analyze dark-frame images of any digital camera as long as they support digital RAW format.

Although we gained understanding of the linear relationship between SEU occurrence and elevation, it has a lot of space to grow. For the elevation tests, only two DSLRs were used – the Canon 5D Mark II and the Canon T2i. Using additional DSLRs for these tests will be helpful in providing additional data points with different pixel sizes. A more important test to perform is to understand the behaviour of SEUs in cell phone cameras at different elevations. As mentioned earlier, given the time of this thesis – circa 2020 – the global pandemic COVID-19 made it nearly impossible to share cell phones for capturing dark-frame images at different elevations. The rather small pixels of cell phone cameras could potentially be further affected by SEUs as the elevation changes. Additionally, there were only five elevations from which we captured dark-frame images –

24 m, 74 m, 117 m, 366 m, and 1088 m. Given Vancouver, B.C is close to Mountains, capturing more dark-frame images at elevations between 1000 m and 1500 m will be very beneficial.

Lastly, all the SEU detection scripts that were used for this research were written in MATLAB and ran on standard lab computers. A lot of researchers in the industry are replacing older algorithms with analytics and deep learning as they provide more intelligent and efficient methods of computing. Open-source technologies such as TensorFlow and PyTorch are common Python engines that should be explored for SEU analysis. To compliment these software platforms, leveraging GPUs instead of desktop machines will provide the software with enough computational power to complete the analysis rather quickly.

6.3. Closing Thoughts

We have experimented and analyzed a large number of dark-field image datasets (each with 1000 images) captured by a range of digital cameras – both DSLRs and cell phones. The colour noise maps, the Pixel Address Distribution method, and the charge analysis have enabled us to understand sensor noise distributions, detect SEUs and discard false positives, and understand charge distribution patterns respectively. Additionally, all of these were performed at different elevations to observe the change in SEU behaviour. Notably, our results indicate a strong rise in SEU rates even with elevation increases of fifty metres. Changes in elevation also indicate to us the effect of weaker SEU charges. Similarly, smaller pixels are also susceptible to weaker SEU charges. Therefore, our current research will be able to predict the impact of the industry trend of shrinking IC sizes. Finally, many physicists have proposed using distributed digital imagers around the world as cosmic radiation detectors. Our reliable SEU detection system paired with this style of crowd sourcing will give them an accurate procedure of detecting cosmic rays.

References

[1] G. H. Chapman, "The technology of Digital Photography," SFU, September 2018. [Online]. Available: http://www.sfu.ca/~gchapman/e895/e895l1a.pdf.

[2] B. A. Weisberger, "You Press The Button, We Do The Rest," American Heritage, 1 November 2019. [Online]. Available: https://www.americanheritage.com/you-press-button-we-do-rest. [Accessed 20 December 2019].

[3] P. Hall, "Best DSLR camera 2019: 10 great cameras to suit all budgets," TechRadar, 1 October 2019. [Online]. Available: https://www.techradar.com/news/best-dslr. [Accessed 2 January 2019].

[4] F. Richter, "Infographic: Digital Camera Sales Dropped 84% Since 2010," Statista Infographics, [Online]. Available: https://www.statista.com/chart/5782/digital-camera-shipments. [Accessed 23 November 2019].

[5] F. Richter, "Infographic: What Smartphones Have Done to the Camera Industry," Statista Infographics, [Online]. Available: https://www.statista.com/chart/15524/worldwide-camera-shipments/. [Accessed 30 November 2019].

[6] "iPhone 11 Pro, Apple (CA)," [Online]. Available: https://www.apple.com/ca/iphone-11-pro. [Accessed 29 December 2019].

[7] P. Bhardwaj, "The iPhone X was the most popular smartphone in the world in the first quarter of 2018, according to estimates," Business Insider, 1 June 2018. [Online]. Available: https://www.businessinsider.com/apple-iphone-x-sales-estimated-to-beat-all-smartphones-in-q1-2018-chart-2018-6. [Accessed 5 January 2020].

[8] L. Rehm, "Disruptive technologies in mobile imaging: Taking smartphone cameras to the next level," DXOMARK, 11 January 2019. [Online]. Available: https://www.dxomark.com/disruptive-technologies-mobile-imaging-taking-smartphone-cameras-next-level. [Accessed 4 January 2020].

[9] R. Thomas, "MASc Thesis: Enhanced Digital Imager Defect Analysis with Smaller Pixel Sizes," Simon Fraser University, Burnaby, BC., Canada, August 2016.

[10] J. Leung, *MASc Thesis: Measurement and Analysis of Defect Development in Digital Imagers*, Simon Fraser University, 2011.

[11] J. F. Ziegler, "Terrestrial cosmic rays," *IBM Journal of Research and Development*, vol. 40, no. 1, pp. 19-39, January 1996.

[12] T. J. O'Gorman, J. M. Ross, A. H. Teber, J. F. Ziegler, H. P. Muhlfeld, C. J. Montrose, H. W. Curtis and J. L. Walsh, "Field testing for cosmic ray soft errors in semiconductor memories," *IBM Journal of Research and Development*, vol. 40, no. 1, pp. 41-50, 1996.

[13] M. Niedzwiecki, K. Rzecki, M. Marek, H. Piotr, K. Smelcerz, D. A. Castillo, S. Karel, B. Hnatyk, J. Zamora-Saa, A. Mozgova, V. Nazari, D. Gora, K. Kopanski, T.

Wibig and D, "Recognition and classification of the cosmic-ray events in images captured by CMOS/CCD cameras," in *36th International Cosmic Ray Conference*, Madison, WI, 2019.

[14] G. Chapman, R. Thomas, R. Thomas, K. Meneses, T. Yang, I. Koren and Z. Koren, "Single Event Upsets and Hot Pixels in Digital Imagers," *Int. Sym on Defect and Fault Tolerance*, pp. 41-46, 2015.

[15] A. Theuwissen, "Influence of terrestrial cosmic rays on the reliability of CCD image sensors. Part 2: experiments at elevated temperature," *IEEE Transactions on Electron Devices,* vol. 55, pp. 2324-2328, 2008.

[16] T. Bergmüller, L. Debiasi, A. Uhl and Z. Sun, "Impact of sensor ageing on iris recognition," Authentic Vision, Austria.

[17] J. Leung, G. Chapman, I. Koren and Z. Koren, "Statistical Identification and Analysis of Defect Development in Digital Imagers,," *SPIE Electronic Imaging, Digital Photography V*, vol. 7250, 2009.

[18] G. Chapman, R. Thomas, K. Meneses, I. Koren and Z. Koren, "Image degradation from hot pixel defects with pixel size shrinkage," *Electronic Imaging,* vol. 2019, 2019.

[19] G. H. Chapman, R. Thomas, K. Meneses, I. Koren and Z. Koren, "Image degradation from hot pixel defects with pixel size shrinkage," in *Proc. of the 2019 Symposium on Electronic Imaging*, 2019.

[20] G. H. Chapman, R. Thomas, K. Meneses, P. Pourbakht, I. Koren and Z. Koren, "Exploring Hot Pixel Characteristics for 7 to 1.3 micron Pixels," in *Proc. of the 2018 Symposium on Electronic Imaging*, 2018.

[21] Z. Török and S. P. Platt, "Application of imaging systems to characterization of single-event effects in high-energy neutron environments," *IEEE Trans. Nucl. Sci,* vol. 53, p. 3718–3725, 2006.

[22] S. Mukherjee, "Architecture Design for Soft Errors," *Elsevier, Inc.*, 2008.

[23] F. Firouzi, M. Salehi, F. Wang and S.-M. Fakhraie, "An accurate model for soft error rate estimation considering dynamic voltage and frequency scaling effects," *Microelectronics Reliability,* 2011.

[24] S. Mitra, T. Karnik, N. Seifert and M. Zhang, "Logic soft errors in sub-65nm technologies design and CAD challenges," *Proc. of the 42nd Design Automation Conference*, pp. 2-4, 2005.

[25] Z. S. et al, "Study on FPGA SEU mitigation for readout electronics of DAMPE BGO calorimeter," *2014 19th IEEE-NPSS Real Time Conference, Nara,* p. 1, 2014.

[26] X. Zhang and X. Shen, "The SEU Cross Section Estimation in SRAM Induced by Protons of Space Environment," *2009 IEEE Circuits and Systems International Conference on Testing and Diagnosis, Chengdu*, pp. 1-4, 2009.

[27] A. S. Brogna et al, "SEU protected CPU for slow control on space vehicles," *Second IEEE International Workshop on Electronic Design, Test and Applications,* pp. 422-424, 2004.

[28] C. S. Dyer, A. J. Sims, J. Farren and J. Stephen, "Measurements of the SEU environment in the upper atmosphere," *IEEE Transactions on Nuclear Science,* vol. 36, no. 6, pp. 2275-2280, 1989.

[29] E. Normand, "Single-event effects in avionics," *IEEE Transactions on Nuclear Science,* vol. 43, no. 2, pp. 461-474, 1996.

[30] D. R. Czajkowski, M. P. Pagey, P. K. Samudrala, M. Goksel and M. J. Viehman, "Low Power, High-Speed Radiation Hardened Computer & Flight Experiment," *2005 IEEE Aerospace Conference,* p. 1, 2005.

[31] R. Banu and T. Vladimirova, "Investigation of Fault Propagation in Encryption of Satellite Images Using the AES Algorithm," *2006 IEEE Military Communications conference, Washington, DC,* 2006.

[32] A. Chugg, R. Jones, M. Noutrie, C. Dyer, K. Ryden, P. Truscott, J. Armstrong and D. King, "Analyses of CCD images of nucleon-silicon interaction events," *IEEE Trans. Nucl. Sci.,* vol. 51, p. 2851–2856, 2004.

[33] S. Lecher, "Solid-State Band Theory," Datenschutzerklärung der TU Braunschweig, [Online]. Available: http://www.pci.tu-bs.de/aggericke/PC4e/Kap_I/bands.htm. [Accessed 13 December 2019].

[34] Dux College, "HSC Physics - Ideas to Implementation - dot point notes," Dux College, [Online]. Available: https://dc.edu.au/dot-point-summary-ideas-to-implementation/. [Accessed 13 December 2019].

[35] M. A. Green, "Optical Properties of Silicon," PVEducation, [Online]. Available: https://www.pveducation.org/pvcdrom/materials/optical-properties-of-silicon. [Accessed 14 December 2019].

[36] B. A. Nasir, S. Saadoon, A. Abdullabas and H. K. Easa, "Power Electronic," *Power Electronic,* p. 5, May 2018.

[37] "Learnabout Electronics," Introduction to Diodes, [Online]. Available: https://learnabout-electronics.org/Semiconductors/diodes_20.php. [Accessed 9 December 2019].

[38] X. H. Anderson, "How the Depletion Region of PN Junction changes under Bias," Electrical Engineering Stack Exchange, 1 January 1966. [Online]. Available: https://electronics.stackexchange.com/questions/220824/how-the-depletion-region-of-pn-junction-changes-under-bias. [Accessed 20 December 2019].

[39] Thorlabs, "Photodiode Tutorial," Thorlabs.com, [Online]. Available: https://www.thorlabs.com/tutorials.cfm?tabID=31760. [Accessed 20 December 2019].

[40] M. University, "Colour," McGill University, [Online]. Available: https://www.cs.mcgill.ca/~rwest/wikispeedia/wpcd/wp/c/Color.htm. [Accessed 10 January 2020].

[41] "Image sensor," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Image_sensor#Color_separation. [Accessed 11 January 2020].

[42] J. Leung, G. H. Chapman, I. Koren and Z. Koren, "Statistical Identification and Analysis of Defect Development in Digital Imagers," *Proc. SPIE Electronic Imaging, Digital Photography V,* vol. 7250, 2009.

[43] G. Zapryanov and I. Nikolova, "Comparative Study of Demosaicing Algorithms for Bayer and Pseudo-Random Bayer Color Filter Arrays," *International Scientific Conference Computer Science,* 2008.

[44] Sony USA, "What is Aperture and Depth of Field in Photography," Sony USA - Electronics, PlayStation, Movies, Music, Product Support, [Online]. Available: https://www.sony.com/electronics/what-is-aperture-depth-of-field. [Accessed 11 January 2020].

[45] D. Groom, "Cosmic Rays and Other Nonsense in Astronomical CCD Imagers," *Lawrence Berkeley National Laboratory,* pp. 301-311.

[46] J. Leung, G. H. Chapman, I. Koren and Z. Koren, "Tradeoffs in imager design with respect to pixel defect rates," *Proc. of the 2010 Intern. Symposium on Defect and Fault Tolerance in VLSI,* pp. 231-239, 2010.

[47] G. H. Chapman, R. Thomas, K. J. Meneses, B. Huang and H. Yang, "Detecting SEUs in Noisy Digital Imagers with small pixels," in *Intern. Symposium on Defect and Fault Tolerance in VLSI*, Delft, 2019.

[48] G. H. Chapman, P. Purbahkt, P. Le, I. Koren and Z. Koren, "Exploring Soft Errors (SEUs) with Digital Imager Pixels ranging from 7 um to 1.2 µm," in *Proc. IEEE Int. Sym on DFT*, 2017.

[49] G. H. Chapman, R. Thomas, K. J. Meneses, I. Koren and Z. Koren, "Analysis of Single Event Upsets Based on Digital Cameras with Very Small Pixels," in *IEEE Int. Symposium on Defect and Fault Tolerance*, 2018.

[50] G. H. Chapman, R. Thomas, K. J. Meneses, B. Huang, H. Yang, I. Koren and Z. Koren, "Detecting SEUs in Noisy Digital Imagers with small pixels," in *IEEE Int. Symposium on Defect and Fault,* 2019.

[51] G. H. Chapman, P. Purbakht, P. Le, I. Koren and Z. Koren, "Exploring Soft Errors (SEUs) with Digital Imager Pixels ranging from 7 um to 1.2 µm," in *Proc. IEEE Int. Sym on DFT*, 2017.

[52] G. H. Chapman, R. Thomas, K. J. Meneses, I. Koren and Z. Koren, "Analysis of Single Event Upsets Based on Digital Cameras with Very Small Pixels," in *IEEE Int. Symposium on Defect and Fault Tolerance*, 2018.

[53] F. Firouzi, M. Salehi, F. Wang and S. M. Fakhraie, "An accurate model for soft error rate estimation considering dynamic voltage and frequency scaling effects," *Microelectronics Reliability,* 2011.

[54] A. Chugg, R. Jones, M. Moutrie, C. Dyer, K. Ryden, P. Truscott, J. Armstrong and D. King, "Analyses of CCD images of nucleon-silicon interaction events," *IEEE Trans. Nucl. Sci,* vol. 51, p. 2851–2856, 2003.