

X-ray Spectroscopy Laboratory

Prepared by: Suresh Sivanandam

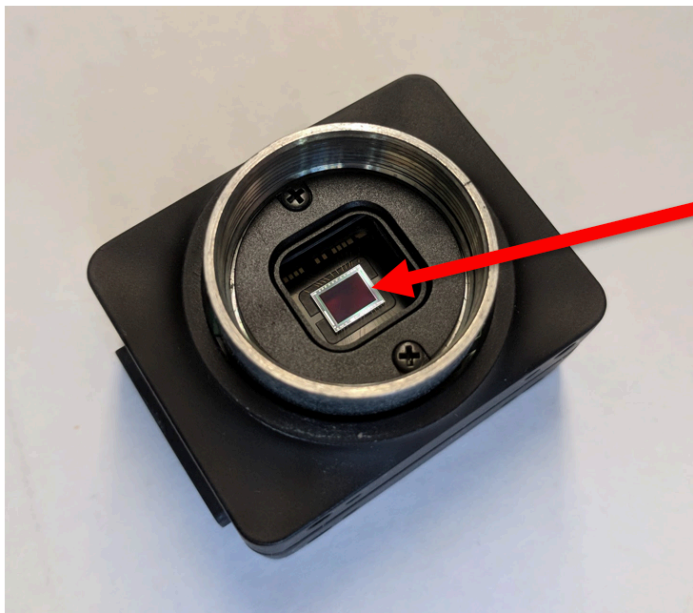
Revised by: Michael Reid

Dunlap Institute for Astronomy & Astrophysics, University of Toronto

1 Safety and Equipment Care

In this lab, you will be using an iron-55 (Fe55) radiative source that produces soft X-rays. Please handle it responsibly. The source is safe to handle under normal laboratory conditions and does not normally pose an external exposure hazard. However, don't remove it from the lab and definitely do not ingest it.

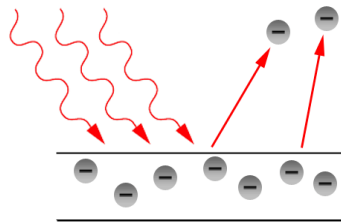
The camera you will be using has had an internal protective window removed so that the CCD chip inside is exposed directly to the air. Please DO NOT touch the exposed CCD chip with any object (fingers, pens, tissues, etc.) for any reason. Doing so will permanently damage the chip. When the camera is not in use, keep the lens cap screwed on to protect the CCD from dust.



Do not let any
object touch the
exposed CCD chip.

2 Introduction & Objectives

At X-ray, ultraviolet, optical and infrared wavelengths most astronomical instruments employ the photoelectric effect to convert photons into electrons, which can then be counted and recorded. Since the 1930's astronomers have used photomultiplier tubes, which employ the photoelectric effect in vacuum. Modern detectors are based on solid-state devices constructed from semiconductor materials. In this lab we will investigate the properties of a CCD-style integrated circuit that uses the photoelectric effect in silicon to record visible and X-ray light.



Above: The photoelectric effect, in which electrons are ejected from a material following the absorption of a photon of sufficient energy.

The process of photon detection and counting is not perfect—detectors have flaws that introduce noise and various systematic errors into the measurement process. The purpose of this lab is to construct a soft X-ray spectrometer using a commercial optical CCD camera, which has had its window removed (so that the window will not block incoming X-rays). In the process, you will explore the operation of a CCD sensor and learn how to characterize the properties of a CCD detector, which in turn will be used to calibrate the X-ray spectrometer.

In this lab you will:

1. Run the Fly Capture software application to control the CCD sensor;
2. Explore the effects of the device controls, including exposure time, gain, and frame rate on the recorded images;
3. Examine images and note the presence of noise in the images;
4. Collect observations of a scene in visible light and corresponding dark frames and read the data into Python and examine the statistical properties of the data to measure the gain and read noise of the detector.
5. Collect observations of the Fe55 X-ray source and process the data in Python in order to detect and determine the spectrum of X-ray photons generated by the source.

Please see Appendix §8 for lab components needed.

3 The CCD image sensor

We will use an image sensor based on a CCD (charge-coupled device) integrated circuit. A CCD sensor is a two-dimensional array of pixels where charges are shifted down each column one row at a time to a horizontal shift register. Charge is generated in each pixel by the photoelectric effect in proportion to the number of visible-light photons or the energy of an X-ray photon incident at that location. The charge within pixels in each row is then shifted onto a capacitor and read one pixel at a time. This voltage is then amplified by a preamplifier within the sensor and by another adjustable gain amplifier within the camera. Finally, this analog signal is digitized by an analog-to-digital convertor, and the data is conveyed to a computer.

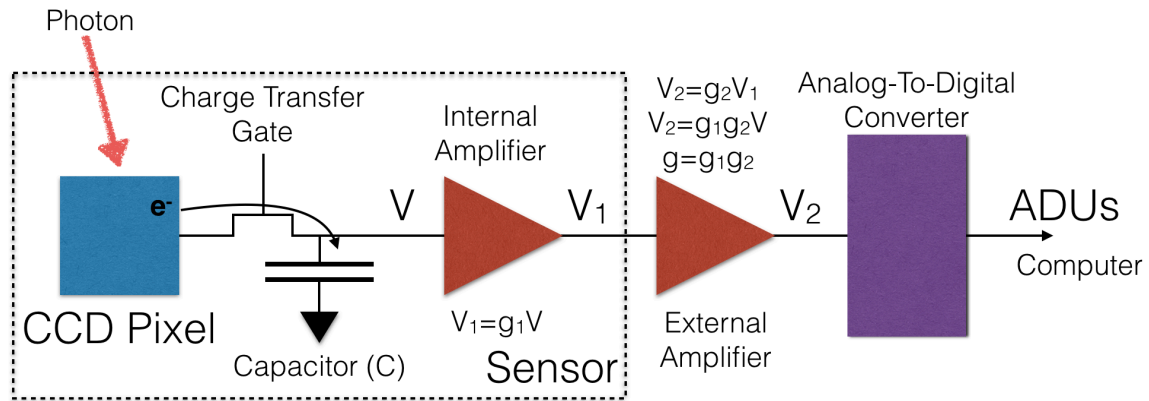


Figure 1: Analog signal chain for a CCD sensor.

The digital signal is recorded in units known as analog to digital units (ADU) after being amplified by both the sensor and the camera electronics as shown in Figure 1. Assuming that the measuring circuits are linear there is a linear relationship between ADU and the number of photoelectrons collected; this constant of proportionality is known as the **gain**. The entire CCD array is automatically reset before the start of an exposure by clearing any remaining charge built up on the sensor. This establishes a well-defined charge state at the start of the subsequent exposure.

The photo-charge, Q_{PE} , is determined by measuring the voltage induced on a capacitor of capacitance, C ,

$$V = Q_{PE}/C . \quad (1)$$

This voltage is therefore directly proportional to the number of incident visible-light photons or the energy of a single X-ray photon detection.

3.1 Types of noise

The above considerations imply various noise sources—both statistical and systematic—associated with the detection of light.

- **Poisson noise**—this is the fundamental noise associated with counting photons (or photoelectrons). The defining characteristic of a Poisson distribution is that the variance and the mean are equal. To a good approximation, photons follow Poisson statistics, so the standard deviation in an experiment that counts on average N_{PE} photoelectrons is $\sqrt{N_{PE}}$.
- **Dark current**—free electrons are created in the CCD by the photoelectric effect, but also by thermal fluctuations (phonons in the crystal lattice) and quantum tunneling. This "dark current" (current in the absence of external illumination) produces a systematic offset or error, which must be subtracted to find the true level of illumination. Dark current also contributes Poisson noise because the total number of charge carriers counted is increased to $N_{PE} + N_D$, where N_D is the mean number of e⁻ from dark current, and the corresponding standard deviation is $\sqrt{N_{PE} + N_D}$.
- **Read noise**—measurement error associated with "reading" the voltage of a given CCD pixel, as represented in Eq. 1. For example, in an *RC* (Resistor-Capacitor) circuit, the rms voltage due to thermal fluctuations is $\sqrt{kT/C}$. Therefore, by Eq 1, the rms charge is \sqrt{kTC} . If $C = 1$ pF and $T = 273$ K then the rms charge introduces a measurement noise of 380 e⁻ rms. While there are ways to circumvent read noise, e.g., by double correlated sampling, there is always some statistical error associated with measuring the voltage represented in Eq. 1.

3.2 Point Grey Chameleon3 detector

We use a two-dimensional sensor manufactured by Point Grey comprising a silicon CCD array (Figure 2). This device uses the solid-state photoelectric effect, where a UV/optical photon of sufficient energy ($h\nu > 1.13$ eV for silicon at room temperature) creates an electron-hole pair in the vicinity of the depletion region of a reverse-biased diode junction. The version of the camera we are using is a monochrome version (see

Table 1).

In the first part of this lab, we will be using a 25 mm lens attached to the CCD detector for the purposes of taking in-focus optical images and to adjust the aperture size of the camera (see Figure 3).

Table 1: Point Grey sensor properties

Specification	Value
Sensor type	1/3-inch
Model	Sony EXview HAD Monochrome CCD ICX445
Pixels count	1288 (H) x 964 (V) – 1.3 Megapixels
Pixel size (square)	3.75 μm
Digitization	12-bit (8-bits are used)
Gain Range	0 – 24 dB
Shutter Range	0.046 ms – 32 seconds
Saturation Capacity	$\sim 9000 e^-$

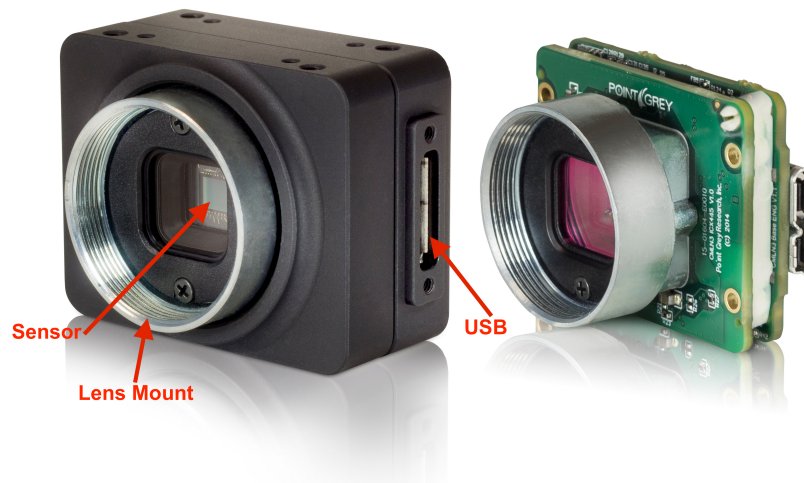


Figure 2: The Point Grey Chameleon3 image sensor. The light grey area at the center is the CCD array (4.86mm x 3.62mm).



Figure 3: The C-mount 25-mm camera lens. The lens screws into the camera body. The designation C on the aperture ring means closed.

4 Controls

The first step of this lab is to log into a Linux laptop and start the FlyCap2 software application (Applications > All > FlyCap2). Be sure that the USB cable is plugged into the laptop and the yellow indicator light is illuminated on the back of the camera. When you start FlyCap2, you'll be asked to select the camera—only one option should be offered. Click "OK" and the control window should open with a live display from the camera (be sure that the lens cap is removed).

The FlyCap2 tool bar presents some useful options including the camera control window ("Toggle Camera Control Dialog"--the icon with the gear) and histogram tool (the icon with the coloured boxes). Open the control window and examine the Camera Settings. This is where you control the exposure time, gain (conversion from output voltage to ADUs), and frame rate. **To adjust the parameters manually "uncheck" AUTO besides each setting. "On/Off" must be checked for the parameter to have its intended effect.**

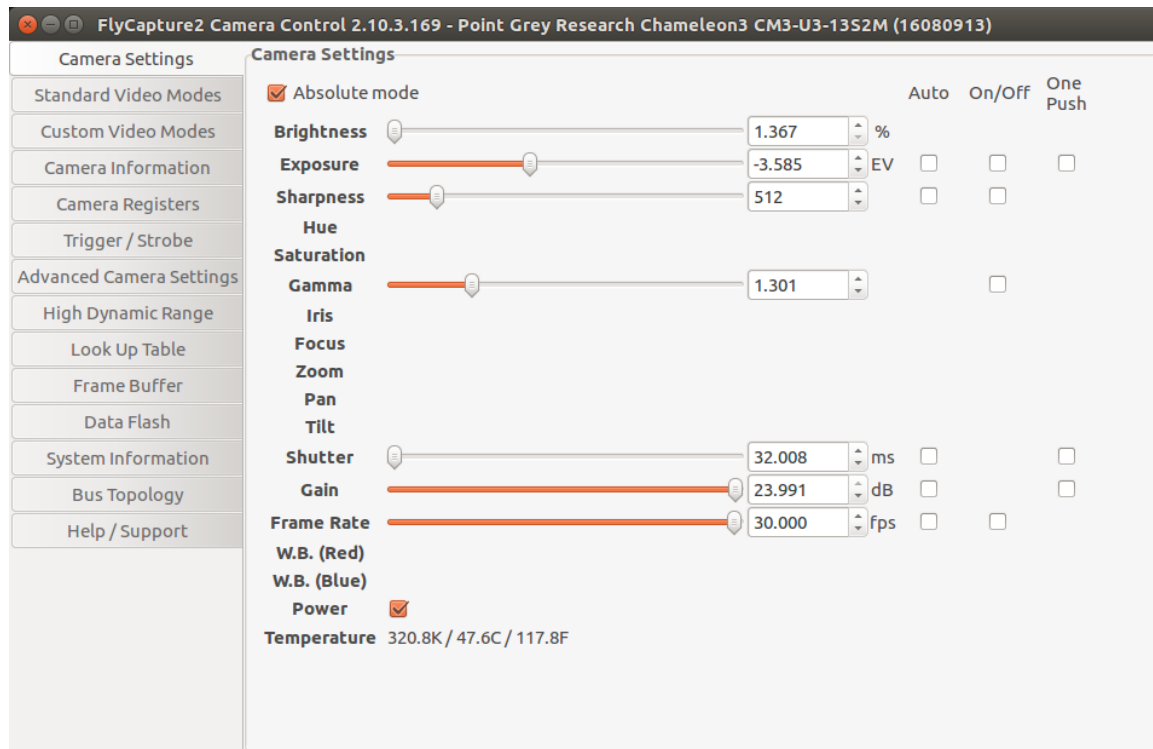


Figure 4: FlyCap2 camera control window. The automatic Brightness and Exposure controls have been disabled and the effect of those parameters have been turned off, the Shutter (exposure time) is set manually to 32 ms and the gain is set at maximum (24dB).

1) Remove the lens cap off the camera (**remember never touch the actual lens!**). Now focus the lens on the camera to form an in-focus image on a stationary target in the room. If you can't focus your image, check to ensure that the black spacer ring is present between the camera and the lens.

The Camera Settings panel (Figure 4) lists a number of options including Brightness, Exposure, Gamma, Shutter, Gain and Frame Rate.

- **Brightness** modifies the display by adding a constant to the data value. If the Shutter AUTO button is checked then adjusting brightness will also change the shutter (exposure) time.
- **Exposure** means auto exposure and controls the Shutter and/or Gain if the AUTO option of either of these is enabled.
- **Sharpness** either smooths or sharpens the image. This should be turned off for our measurements.
- **Gamma** effects how images are displayed by introducing a non-linear relation between the measured signal from the ADC and the value presented in the image.
- **Shutter** determines the exposure time. The exposure time can be adjusted between 0.046 ms – 32 s.

- **Gain** the gain of the ADC can be commanded and is variable from 0 to 24dB. This adjusts the external amplifier in the signal chain (Figure 1), and therefore the sensitivity and also the dynamic range of the camera.
- **Frame rate** determines the maximum rate at which frames are read out. For example, if the frame rate is set to 30 Hz (default) and exposure time is 100ms, then a 10 Hz frame rate will be used.

For quantitative work make sure that all the AUTO buttons are unchecked and all On/Off buttons are also unchecked (i.e. off). Otherwise the software may vary the exposure time, gain, and frame rate in response to the illumination level, causing unexpected results. Set the gain to maximum to start.

2) As a group, using the camera lens and detector adjust each of the parameters to determine their use and impact. For some parameters like Exposure and Sharpness, you will need to turn them on using the On/Off buttons, though they are not essential parameters in this lab.

The cursor read out (bottom of the live display window) and the histogram tool (Figures 5 and 6) can be used to examine the data values.

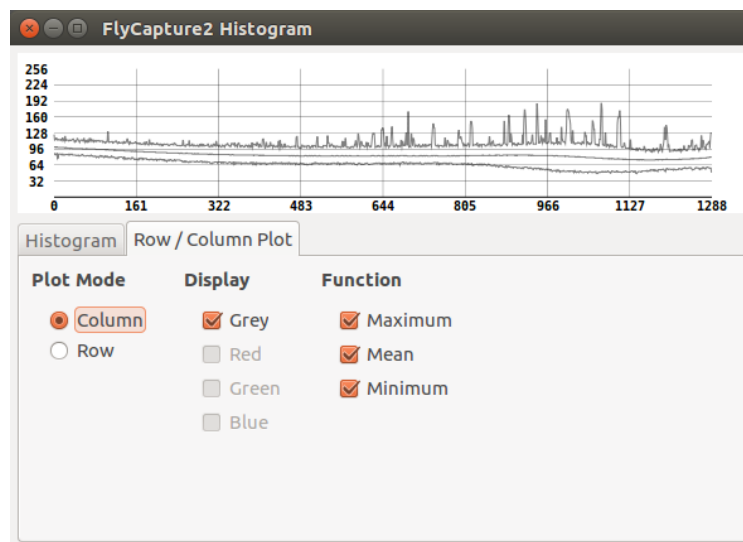


Figure 5: The histogram tool can be used to plot rows or columns.

5 Activities

5.1 Saturation

The CCD sensor has a limited dynamic range. Turn off the automatic options in the Camera Settings windows and examine the live display and the histogram tool as you change the exposure time aka “Shutter” between 0.06 and 134 ms. When you are

examining the histogram, be sure that the “Grey” channel is checked, otherwise no data will be plotted.

3) What is the maximum data value returned by the camera? _____

4) Now what would have been the maximum value for an 8-bit camera?

5.2 In the dark!

Place the lens cap on the camera. Set the Frame Rate slider to its minimum value 1 Hz. You should be able to vary the exposure time from 0.046 ms to 1 s.

5) Describe what you see in the live view window as you increase the exposure time. Is the image dark? uniform? static? Use the zoom tool by scrolling in the live view window so that you can clearly see individual pixels. *Note: you can use the Brightness slider to add a constant offset to the output values so that the pixel values are not all zero.*

6) The histogram tool (Figure 6) reports a mean value. How does the mean value vary with exposure time?

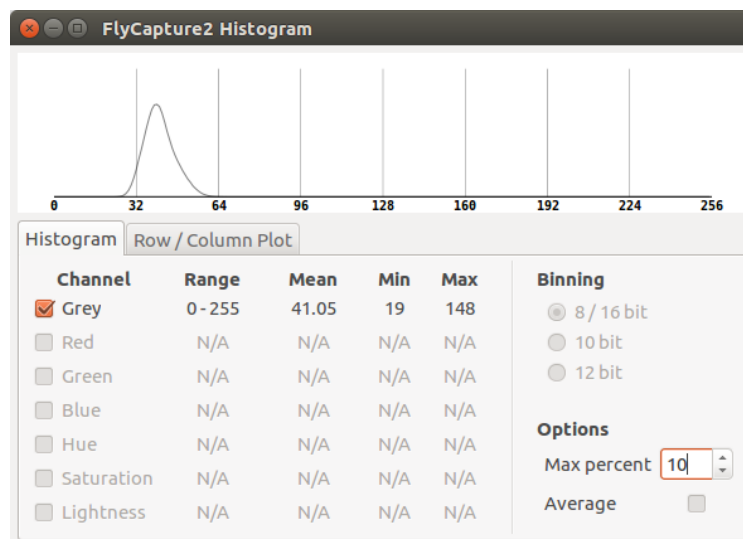


Figure 6: The histogram tool showing a histogram of values centered near the lower-range of possible values. Note that the “Grey” channel is checked and the y-axis range has been adjusted (Max Percent) to 10% so that the peak is clearly visible.

5.3 Histograms

Point the camera at a uniform target. A piece of plain white Xerox paper or a clean patch of lab wall should work well—especially if the lens is out of focus. Examine the live image to make sure that there is no obvious structure in the image. Set the lens

aperture to $F/1.4$ (wide open—see Figure 3), set the gain to maximum (24 dB), and turn ON the automatic brightness, exposure, and shutter. Examine the histogram plot—the mean value should have stabilized to about 180-220 ADU. This procedure allows the camera to optimize the settings to get an optimal count rate.

7) Now turn **OFF** the automatic exposure control. Adjust the F -stop and watch the histogram as you change the aperture of the camera. Describe how the shape (e.g, height, width) of the histogram changes with different aperture sizes? What happens to the mean and standard deviation of the histogram when the aperture is largest ($F/1.4$) and smallest ($F/8$)? (Make sure that none of the exposure parameters are changing. If they are, turn off the corresponding AUTO mode and try again.)

8) Now point the camera to a non-uniform source. Describe what happens to the histogram?

5.4 Measuring the read noise & gain

Now for some quantitative measurements! For this activity we need to save a sequence of images so that we can compute the mean and standard deviation of the signal from each pixel in the image sensor. This also allows us to explore the relationship between signal (the pixel's mean) and the noise (the pixel's standard deviation).

Point the camera at a scene with bright and dim regions. Adjust the settings so that dimmer regions have close to zero signal (0 ADU) and the brighter regions are close to saturation (255 ADU.) Focus is not critical, but out-of-focus images will blur the distinction between light and dark to a more uniform grey. You will find the histogram tool is useful in this exercise.

As we want to explore the statistical properties of a sequence of images, choose a stationary object. **A constant illumination to the detector is imperative for this experiment!** If someone turns the lights on or off, you will need to start again; if someone walks in front of the camera then the changes you record will not be due to intrinsic fluctuations in a pixel.

9) Take some test exposures to adjust the exposure time to confirm that you have a good range of pixel values (make sure the automatic exposures options are turned off.) Use the Brightness slider to make sure the minimum value is consistently above zero ADU so that you see the full range of the data values from 0–255. (The Brightness control simply adds a constant value and will not effect the fluctuations.)

Remember to pause the acquisition in FlyCap2 while you are taking data with the script. To do otherwise will cause the script to crash. The images will be saved using the same settings currently in use by FlyCap2.

10) Collect a sequence of images, say 20-30 and save to disk using the Python script (**acquireImages.py**). See Appendix §7 for more detail on the scripts. This script will save images in an uncompressed TIFF format. More data is better, but if you wait too long then the illumination may change.

11) Repeat the experiment, but this time with the lens cap in place. We want the mean value in our final data to reflect how much light illuminated each pixel, so we need these dark images so that we can subtract them from our original pictures.

12) We are now ready to analysis your data. First create two text files that list all the dark (dark.txt) and light (light.txt) frames in a single column within the respective directories that the data are saved in and use the Python script (**calcGain.py**) to analyze the data. Please see Appendix §7 and the script for more details.

13) After running the script (**calcGain.py**) on your data, answer the following questions:

- Describe the trend revealed by the scatter plot of mean pixel values and their variance, i.e., standard deviation squared.
- Why is the variance not zero when the signal is zero?
- What happens to the relation between mean and variance at high signal levels?
- What is the meaning of the slope of the graph?

14) Using the information in Appendix §6, what is the read noise and gain of your camera?

Read noise = _____ ; Conversion Gain = _____

Hint: the "gain" you're trying to calculate is the conversion factor from ADU to e^- , so it's a number in e^-/ADU .

5.5 Detecting X-ray Photons

We are going to use the method of direct detection to observe the X-ray photons in this component of the lab. When the X-ray photon hits the CCD, it generates not one, but several electron-hole pairs. The number of electron-hole pairs the photon generates is proportional to its energy, which is why we can turn our CCD camera into an X-ray spectrometer. The electrons are captured and measured just like they are when we use optical light, but in this case, we will be able to see **individual** X-ray photons emitted by the Fe55 source!

The Fe-55 source you will use has a very low activity of 0.025 milliCurie and is sealed. It is relatively safe to handle. **However, only the lab coordinators who are properly trained will physically handle the X-ray source. Do not try to handle it yourself. Do not take the source back with you! I will get into a lot of trouble!**

For this section of the lab, you will have to remove the lens as the sensor needs to view the X-ray source directly. Be careful as the sensor is exposed and is very fragile! Do not put your finger into the camera barrel or drop anything on the sensor. If the camera is not in use, please put the supplied camera cover on.

1) Collect a sequence of images after placing the X-ray source facing the camera sensor. Make sure the camera is in a dark environment. Compare with a sequence of images taken without the source and see if you observe any differences. One way to look for differences is to take a difference image. Use the Python code (**subImages.py**) to plot the difference image. Compute multiple difference images and compare them to one another. Answer the following questions.

- What do you observe?
- If there are changes between the multiple difference images, can you explain why? What is the reason for using difference images?
- Do you observe any negative values, and if so what is the reason?

Note: You will need relatively long exposures (> 1 s) to see a significant number of X-ray photons.

2) It is important that we set the dynamic range correctly so that the X-ray photon measurements don't saturate the detector. Look at the histogram plot to see if a significant portion of the X-ray photon hits are saturated. If they are, you will need to lower the detector gain and redo the measurement. If you do need to lower the detector gain, you will need to redo the gain measurement you carried out in Section 4.4 to determine what the e^- to ADU conversion gain is for the new setting.

This part is crucial for the next section. You should set the camera gain such that you use the maximum dynamic range of the sensor, *i.e. the most energetic X-ray photon should be close to saturating the detector*. This ensures you have the dynamic range to see a few spectral lines from the Fe55 source. *Hint: The maximum number of photoelectrons you would expect for each X-ray hit is around 1800 e^- (6.5 keV).*

For a quick estimate what gain to change to, you can use the following equation:

$$g_1/g_2 = 10^{(g_{DB1}-g_{DB2})/20} \quad (2)$$

where g_1 and g_2 are in linear units while gDB_1 and gDB_2 are in decibel units. After changing the camera gain be sure to carry out the mean-variance analysis to compute the new gain before acquiring another X-ray dataset.

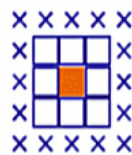
5.6 Measuring the Energy of X-ray Photons

The number of electrons produced by a single X-ray photon hit with energy E is given by the Equation 3:

$$E_\gamma = N_e \times W_{Si} \quad (3)$$

where E_γ is the energy of the photon in eV, N_e is the number of electrons detected per X-ray photon hit, and W_{Si} is the electron-hole pair production energy within silicon of 3.66 eV at room temperature (Scholze et al. 1998). This means a 6.5 keV X-ray photon will generate 1776 e^- of charge in a silicon detector. The electron-hole pair production energy is a fixed property of materials, though W has a slight dependence on temperature. This is different from the energy (1.13 eV) required for an optical photon to create a single electron-hole pair in silicon because the high energy photons also inject additional energy into modes within the crystal lattice and heat it up.

Now to carry out a spectral measurement of the source, we need to perform some image analysis. When an X-ray photon hits a sensor, it does not distinguish between pixel boundaries, and sometimes it does not deposit all of its energy within a single pixel. These events are not useful for spectroscopy since some of the photon energy may have been lost and not properly accounted for. Therefore, we need to first select for “single-pixel” events as shown the diagram below before constructing an X-ray spectrum.



1) Using the Python code (**xraySpectrum.py**) provided for filtering for single-pixel events, select single pixel events and plot their values. Do you find a clustering of certain values? One of the parameters you may need to adjust is the filtering threshold (**hit_threshold**) to ignore pixels that are simply noisy and are not real X-ray detections.

2) Plot an energy histogram of X-ray photon detections after scaling for the ADU value to the corresponding photon energy by inputting your measured value of the conversion gain in to the code. If you are successful, you should see at least one prominent X-ray line. Investigate different values of **hit_threshold**, **bin_threshold**, and **bins** to determine if you can improve the quality of the spectrum and pull out

more emission lines (I have measured three lines in this lab). You can also try using different pairs of X-ray images or take new data.

3) Bonus: Can you determine the theoretical spectral resolution of this spectrometer? (*Hint: Think Poisson statistics.*)

6 Appendix: Signals & Noise—Measuring gain

Our CCD sensor uses the photoelectric effect to measure the intensity of light. The number of photoelectrons in a given pixel, N_{pe} , per exposure is proportional to the visible-light intensity. Each pixel also generates a “dark current”, even when there is no illumination, so that the total charge is $N = N_{pe} + N_d$. In the first step of our analysis we subtract off a dark image so that we are left with N_{pe} .

During readout of the image the accumulated electric charge is deposited on a capacitor of capacitance C , thereby generating a voltage

$$V_{pe} = Q/C = Ne/C, \quad (4)$$

where e is the charge of the electron. Typical values of C are a few pF; one electron (1.60×10^{-19} C) on 1 pF generates a signal of 160 nV.

The signal generated is a number returned by a digital circuit that converts voltage to an 8-bit number (0-255): this number is directly proportional to the voltage. By convention, the signal from the analog to digital converter (ADC) is measured in *ADUs* (analog to digital units). The voltage measuring circuit has a constant of proportionality, g , with units of *ADU* per volt. Refer to Figure 1, the analog signal chain for the CCD to understand what determines g . Thus, the number in your data file is

$$ADU = g Ne/C + ADU_0. \quad (5)$$

The quantity ADU_0 is an offset or bias—a non-zero count that is returned even when the number of photoelectrons is zero (this offset is a practical detail introduced to the circuit because the ADC can only generate positive numbers.) Note that the combined quantity ge/C has units of *ADU* per electron. For convenience, this quantity is often known simply as the gain.

We can use our knowledge of Poisson statistics applied to counting photoelectrons to deduce the gain factor in Eq. 5. First, notice that Eq. 5 implies that the signal in *ADU* depends on the number of photoelectrons and the bias value, i.e.,

$$ADU = ADU(N, ADU_0). \quad (6)$$

To find the error in the measured signal we can apply the fundamental formula for error propagation, which states that the variance in some quantity f , which is a function of u, v, w, \dots is

$$\sigma_f^2 = \left(\frac{\partial f}{\partial u} \right)^2 \sigma_u^2 + \left(\frac{\partial f}{\partial v} \right)^2 \sigma_v^2 + \left(\frac{\partial f}{\partial w} \right)^2 \sigma_w^2 + \dots, \quad (7)$$

where we have assumed that u, v, w, \dots are independent quantities with zero covariance.

Two types of noise contribute to the standard deviation of ADU measured, or σ_{ADU} . By applying the law of error propagation, Eq. 7, to Eq. 6 we find

$$\sigma_{ADU}^2 = \left(\frac{\partial ADU}{\partial N} \right)^2 \sigma_N^2 + \left(\frac{\partial ADU}{\partial ADU_0} \right)^2 \sigma_0^2. \quad (8)$$

The first partial derivative is just the gain, ge/C , while the second is unity. The variance σ_N^2 in the first term is associated with Poisson noise, for which we know that the variance equals the mean, i.e., $\sigma_N^2 = N$. In general there is some noise associated with each measurement, σ_0 , which is known as the read noise. Thus, Eq. 8 simplifies to

$$\sigma_{ADU}^2 = \left(\frac{ge}{C} \right)^2 N + \sigma_0^2. \quad (9)$$

By substituting $geN/C = ADU - ADU_0$ from Eq. 5 we have

$$\sigma_{ADU}^2 = \frac{ge}{C} (ADU - ADU_0) + \sigma_0^2. \quad (10)$$

Thus, a plot of the variance, σ_{ADU}^2 , which is in units of ADU^2 , versus the mean bias subtracted signal, $ADU - ADU_0$, should be a straight line with slope m equal to ge/C (units of ADU/e^-) and intercept of read noise squared (units of ADU^2). An example of applying Eq. 10 the camera used in our lab is shown in Figure 7.

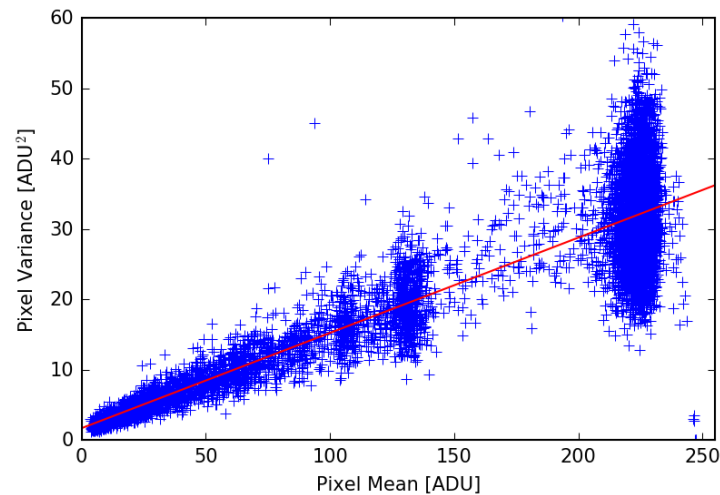


Figure 7: Variance/mean plot output by `calcGain.py` for the camera used in the lab for an unknown camera gain setting. The red line shows the linear fit to the mean-variance data. The slope of this curve is related to the gain of the camera system while the y-intercept is related to its read noise.

7 Appendix: Python data reduction scripts

There are several python scripts provided to help you carry out the required data analysis in this lab. You will need to modify some of the parameters to carry out your analysis. You can use your favourite text editor and **ipython** or Spyder Python IDE to edit and run your code.

List of scripts:

captureImages.py: Captures N number of images from camera and saves them as TIFF files under a specified directory. The script will create the directory if it does not exist.

calcGain.py: Given a set of light.txt and dark.txt files that you create, which are located in two specified directories where the data files exist, this script will carry out dark subtraction of the illuminated frames, and compute the mean and variance of each pixel. It will then plot the mean-variance data and carry out a linear regression to determine the gain (i.e. slope) and read noise (i.e. y-intercept) of the camera.

subImages.py: Given two images, this script will subtract two specified images and plot the resultant image for inspection.

xraySpectrum.py: Given two images, camera gain and threshold values, this script selects for single-pixel events and generates a plot of the X-ray hit values in ADUs and a histogram plot, which displays the measured X-ray spectrum. A key parameter is the **hit_threshold**, which is the value where the code considers a pixel to contain an X-ray hit.

8 Appendix: Lab Components

This lab is currently designed for 21 students split between 7 identical stations and requires the following equipment:

- 7 x CCD detectors with windows removed and sensor covers
- 7 x 25 mm Camera Lenses
- 7 x white LED sources (if needed)
- 7 x Computers with
 - FlyCapture2 software
 - Python
- 7 x White sheets of paper
- 7 x Boxes
- 2 x Fe-55 sealed radioactive sources

9 Acknowledgements

I would like to thank James Graham and Shelley Wright for developing the detector characterization component of this lab.