Determining the accuracy of SPC Pictures taken using Equi-Width Histograms versus Equi-Depth Histograms, Parallel versus Asynchronous Data Collection and 1.5 FWHM versus 5.0 FWHM under variable Attenuation using SPCSimLib [SMI25]

Elliott H. VanOrman, Intern at the Institute for Computing in Research, working for Professor Atul Ingle and Kaustubh Sadekar

July 30, 2025

INTRODUCTION

These slides tell the story of how I compared the error in results of a Single Photon Camera (SPC) when using Equi-Width Histograms (EWH) versus Equi-Depth Histograms (EDH), parallel versus asynchronous data collection, and 1.5 FWHM versus 5.0 FWHM laser pulses. Data was collected using the python library SPCSimLib, specifically by simulating a single SPC pixel, and measuring the error each method produced. My analysis showed using Equi-Width Histograms, asynchronous data collection, or 1.5 FWHM pulses creates pictures matching reality more closely than those created using Equi-Depth Histograms, parallel data collection, or 5.0 FWHM pulses, to a statistically significant level. From this, I concluded that using Equi-Width Histograms, Asynchronous data collection and 1.5 FWHM pulses produces more accurate results than using Equi-Depth Histograms, Parallel data collection, and 5.0 FWHM pulses.

BACKGROUND INFORMATION

Single Photon Cameras, or 'SPC's for short, are cameras that function similarly to LIDAR, in that they produce 3D photographs by measuring the time it takes for light to travel to a point and bounce back. Unlike LIDAR however, SPCs use individual photons instead of lasers. [Ing25] [Sad25]

When a picture is taken, the SPC begins sending out pulses of photons towards the thing it is taking a picture of. When these photons are reflected back onto the detector, it records the timestamp. We can then calculate the distance away from our camera using the formula

$$\frac{T * \frac{1}{2}}{C}$$

where T is the time it took photons to reflect back to the detector and C is the speed of light. [Sad25]
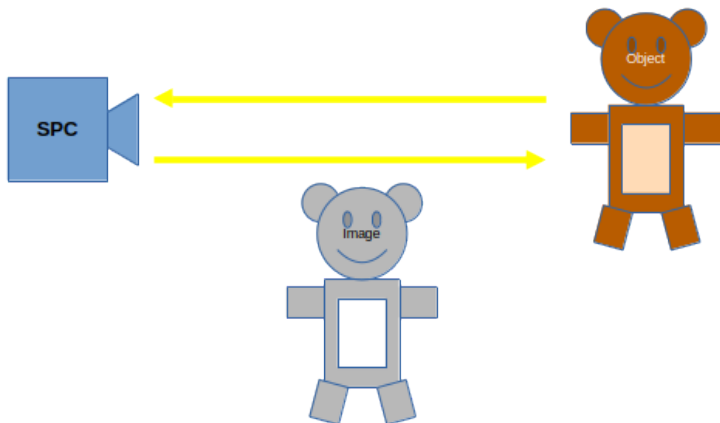
Figure: How SPCs Work.

There are almost always ambient photon levels, however, meaning that while timestamps do cluster around the 'true' time it took photons to reflect back, there are always a few 'noise' timestamps due to background illumination. They can be minimized via turning off lights and waiting for nightfall, but this is often not enough. There are multiple methods to determine when the true center of the pulse (called the 'peak') is located. After the 'peak' is located, it is used as T in the previously shown formula [Sad25].

Taking an accurate picture, collecting data accurately, and sorting the noise from the real data is quite complicated however, and there are several factors to consider. Namely:

► The kind of histogram used to locate the peak.

► Whether collectors collect data asynchronously or in parallel.

► How long the laser fires.

There are two types of histograms to use when finding the peak: [Sad25]

► Equi-Width Histograms, histograms with buckets of equal width but variable depth. For EWHs, The 'peak' bucket is the bucket with most detections.

► Equi-Depth Histograms, histograms with buckets of variable width but equal depth. For EDHs, the 'peak' bucket is the bucket of the shortest width.

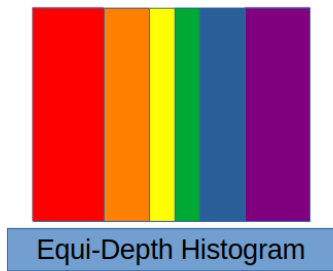► In both cases, The center of the 'peak' bucket is recorded as the 'peak' of the pulse.

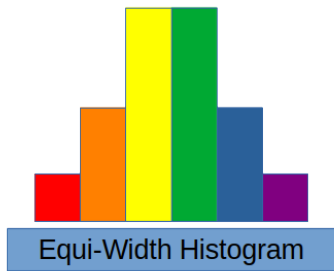Equi-Width Histogram

Equi-Depth Histogram

Figure: The two kinds of histogram.

Before I explain the difference between asynchronous mode and parallel mode, I first need to explain deadtime.

- ▶ Due to the physical limitations of SPC cameras photon readings over time do not always match reality.
- ▶ When a detector is hit by a photon, it takes a certain amount of time for said detector to log a timestamp.
- ▶ While this time is in progress, the detector cannot record new hits. In other words, a detector cannot start recording a new timestamp until it has finished recording the first.
- ▶ photons that hit the detector during that recording time go undetected.
- ▶ This time is usually around 75ns in real life, and is refered to as 'deadtime' [Sad25]

There are two modes in which one collects data: [Sad25]

► Parallel (or, Not Freerun) Mode. In Parallel mode, detectors are reset at the beginning of each pulse, resetting their deadtime, but skewing timestamps to the right as early photons will block later photons

► Asynchronous (or, Freerun) Mode. In Asynchronous mode, detectors are simply allowed to run, their deadtime elapsing in full but leaving their timestamps less skewed.

Then there is FWHM:

- ► FWHM, unlike histograms and modes, is a continous measurement of how long each laser pulse fires.
- ► FWHM, or Full Width at Half Maximum, refers to the length of the curve of intensity of the laser when it fires. It can also affect the error of the laser, as the increaed length of the laser would also increase the width of the 'peak' of the pulse.

- Unfortunately, due to the skew in data, it often becomes necesary to filter out a certain percent of incoming photons.
- This filtering is called 'attenuation' [Ing25], and it does reduce the overal number of photons hitting an SPC.
- Because of the fact that any one datapoint blocks other datapoints from being recorded for a period of time causes it to actually increase the amount of photons not blocked by early recordings.
- this reduces the skew of the timestamp data and increasing accuracy. [Sad25]

this matters because there are many methods that can be used to receive and interpret data from SPCs, but it has not yet been determined which produces the most accurate results. I hypothesized that...

- ▶ Equi-depth histograms would produce more accurate results than Equi-Width histograms.
- ▶ asynchronous (non-freerunning) mode would produce more accurate results than parallel (freerunning) mode.
- ▶ 1.5 FWHM would produce the same amount of error as 5.0 FWHMs.

METHODS USED TO COLLECT DATA

In order to determine what produced the most accurate results, I used the Python library SPCSimLib [SMI25] to simulate a pixel of an SPC. To find error, I...

1. Simulated taking a picture with the pixel.

2. Subtracted the 'true distance' (the real distance from the camera in the simulation) from the 'estimated distance' (the distance the camera estimated).

3. Normalized said distance from -1 to 1 by dividing it by the scale of the distance (tbins). (This normalized difference is what I refer to when I say 'error'.)

4. Repeated the previous three steps ten times and took the average for each set of values. (to reduce fluctuations)

5. Repeated the previous four steps one hundred and one times, each time incrementing the attenuation multiplier by 0.01.

6. Put the resultant data into a file labeled with the settings the SPC used while simulations were being taken.

7. then I changed the settings and repeated all previous steps until I had run out of combinations of settings to test.
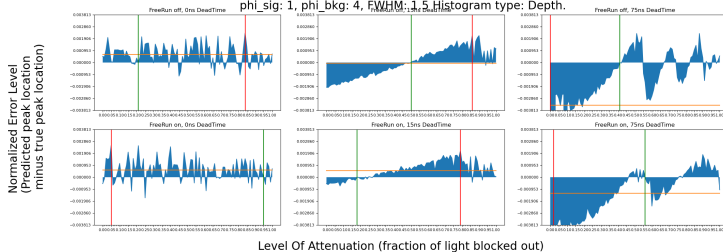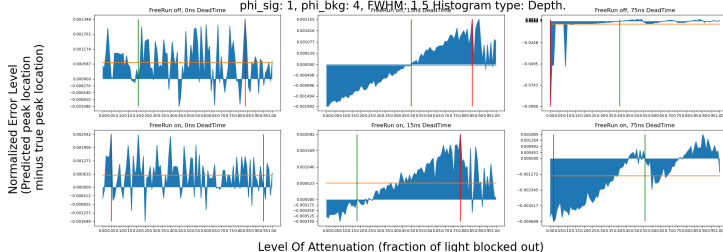
Figure: The two formats in which siad data was graphed.

The top graph and the bottom graph present the same data in two different formats.

▶ The top graph represents error on the same scale across all graphs.

▶ The bottom represents each dataset with it's own scale.

▶ Blue represents normalized error.

▶ Orange is the mean error for all attenuation values.

▶ Red is the attenuation value which produced the least accurate result.

▶ Green is the attenuation value which produced the most accurate result.

in addition, I ran statistical analysis on the data to check my hypotheses. To do this, I put all my data in two large datasets for every hypothesis, one for each differing setting. For every hypothesis, I marked the first dataset d_1 and the second dataset d_2. I then performed a paired t-test on the data, where each observation's partner was an observation taken under circumstances identical to the first's except for the variable being tested.
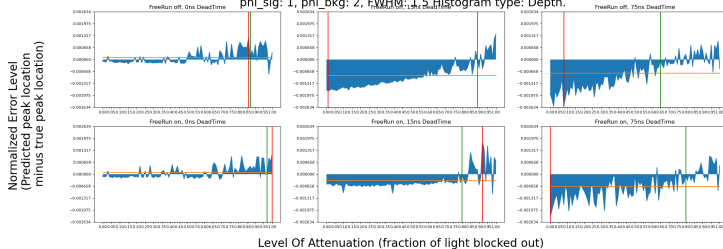
RESULTS OF DATA COLLECTION

Using Paired t-tests, I found the following Ts and P-Values for the following alternate hypotheses:

- $\mu_{EDH} = \mu_{EWH}$: $T = 14.59122$
  - $\mu_{EDH} > \mu_{EWH}$: $p = 1.79689 \times 10^{-48}$
  - $\mu_{EDH} \neq \mu_{EWH}$: $p = 3.59378 \times 10^{-48}$
  - $\mu_{EDH} < \mu_{EWH}$: $p = 1.0$
- $\mu_F = \mu_{nF}$: $T = -31.63417$
  - $\mu_F > \mu_{nF}$: $p = 1.0$
  - $\mu_F \neq \mu_{nF}$: $p = 1.63522 \times 10^{-218}$
  - $\mu_F < \mu_{nF}$: $p = 8.17611 \times 10^{-219}$
- $\mu_{1.5} = \mu_{5.0}$: $T = -55.67372$
  - $\mu_{1.5} > \mu_{5.0}$: $p = 1.0$
  - $\mu_{1.5} \neq \mu_{5.0}$: $p = 0$
  - $\mu_{1.5} < \mu_{5.0}$: $p = 0$

I rejected the null for all alternatve hypotheses with p-values less than 0.01. Graphs showing examples of the accepted hypotheses are as follows:
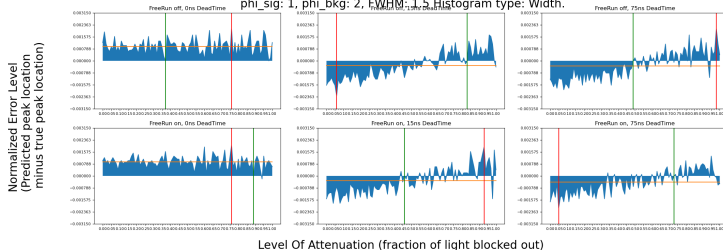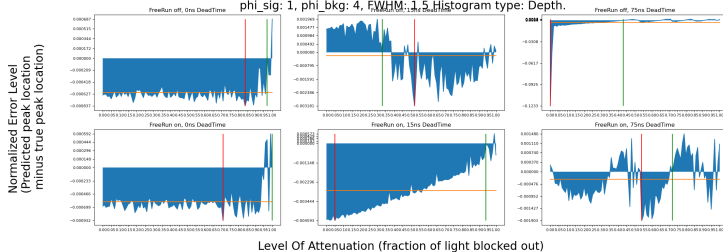
Figure: EDH vs EWH
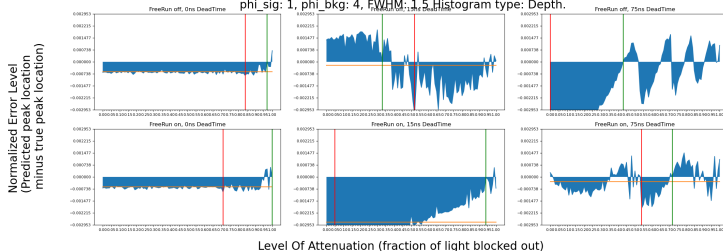
Figure: Freerun and non-Freerun on different y axes.
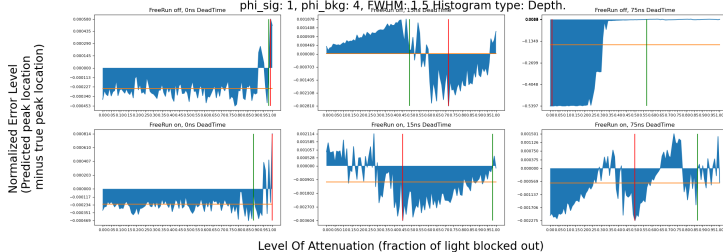
Figure: 1.5 FWHM vs 5.0 FWHM.

Discussion and Analysis of data

In Short, the results show that:

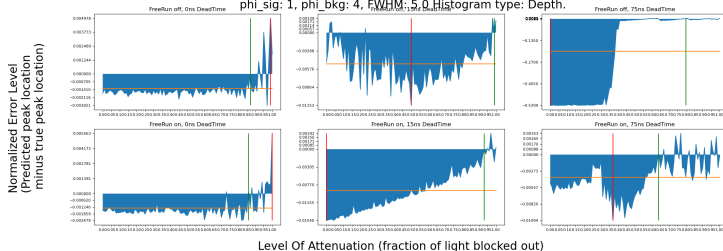- ▶ EDH Error Calculation produces more error than EWH Error Calculation, which does not support my hypothesis that EDH Error Calculation produces less error than EWH Error Calculation.

- ▶ FreeRun simulations produce less error than Non-Freerun simulations, which does support my hypothesis that FreeRun (asynchronous) simulations produce less error than Non-Freerun (parallel) simulations.

- ▶ 1.5 FWHM simulations produce less error than 5.0 FWHM simulations, which does not support my hypothesis that FreeRun simulations produce less error than Non-Freerun simulations.

While some of these results were different from my expectations, they are still explainable observations. Some possible (but not necesarily correct) explanations are as follows:

▶ EDH error calculation may be prone to overvaluing shorter but wider peaks like the ones produced by skewed data and undervaluing tall but thin peaks like the ones produced at the actual peak, while EWH error calculation simply measures the highest bucket.

▶ FreeRun simulations are less prone to producing skewed data, as the point where individual detectors start detecting photons within a cycle varies, thus allowing photons from later into each cycle hit the detector more often.

▶ shorter FWHM simulations may produce taller, thinner peaks as the photons are clustered closer together, thus making the peak stand out from the rest of the data better.

My findings have a few implications for practial use of SPCs. The specific implications is that when trying to minimize error while taking SPC photographs...

- ▶ One should use EWH error caclulation instead of EDH error calculation.
- ▶ One should use Asynchronous data collection (Freerunningmode) instead of Parallel data collection.
- ▶ One should set their laser's FWHM to 1.5 instead of 5.0.

FLAWS AND LIMTIS

All data was collected using the SPCSimLib library, so any flaws in the methods by which it collects data would cause flaws in the data I analyzed and graphed. Additionally, each simulation was prone to it's own fluctuations, and while I did take an average of 10 repetitions to reduce the error in my error measurements (meaning the difference between my average error measurements and base reality) there were still fluctuations in the data, despite the relatively clear trends.

CONCLUSIONS

As Single Pixel Cameras develop and their use increases, it is important to know what settings will allow said SPCs to produce the most accurate pictures.

- ▶ I found that calculating error with EWH error calculation, using Asynchronous Data Collection and 1.5 FWHM lasers produced the most accurate results.

- ▶ This implies that to produce highly-accurate pictures for SPC cameras, one should use said settings.

- ▶ From this, I propose that these become standard settings for SPC cameras.

- ▶ I also suggest that more research be done into FWHM to figure out the optimal FWHM.

BIBLIOGRAPHY

[Ing25]   Atul Ingle. *Personal communication*. 2025.

[Sad25]   Kaustubh Sadekar. *Personal communication*. 2025.

[SMI25]   Kaustubh Sadekar, David Maier, and Atul Ingle.
          "Single-Photon 3D Imaging with Equi-Depth Photon
          Histograms". In: *European Conference on Computer
          Vision* (2025), pp. 381–398.