Image processing in a Snowball Chamber

Jacob Harris, Department of Physics, Rensselaer Polytechnic Institute, Troy, NY 12180

Cecilia Levy and Matthew Szydagis, Department of Physics, SUNY Albany, Albany, NY 12222

I. Abstract

The authors present a complete detection algorithm for image analysis in a snowball chamber, an apparatus which utilizes supercooled water as a detection medium for radiation, using phase transition from liquid to solid, and has potential as a dark matter detector. In constructing this algorithm, the authors have developed and refined existing code to detect the initial nucleation frame and position of “snowballs”. Although the optimal threshold value and detection area vary between runs, an area that is five pixels in width and in height, coupled with a threshold value of twelve, is usually a good set of conditions to use. The image analysis can now be applied to all other runs, cross-checked with the already completed temperature analysis, and used to identify other phenomena, such as multiple scatters.

II. Introduction

Dark matter (DM) accounts for roughly 85 percent of the matter content of the universe, and is instrumental in its structure and evolution. Direct detection of DM is yet to be achieved, despite strong evidence for its existence.[1]

The Snowball Chamber, located at SUNY Albany, is a newly developed technology based on phase transitions in supercooled water, that could be used in the search for low-mass DM of less than 1 to 10 GeV/c^2. The chamber, operated in 2017 and 2018, contained 20 mL of purified water inside of a fused quartz vessel. The water was cooled far below its normal freezing point, then bombarded with neutrons. The subsequent formation of rapidly growing ice structures (snowballs) points to the chamber's potential as a DM detector.[2] Although the theory of nucleations is uncertain, a potential explanation lies in Seitz theory, in which enough energy is deposited in a small enough radius for a phase transition to occur.

With a completed analysis of a pre-existing library of snowball formation images, the already finished examination of snowball formation temperature can be cross-checked to create a better understanding of nucleation in the chamber. The image analysis involves developing and refining a fast, automated program to pinpoint the location and time of nucleation sites. By finalizing an algorithm to detect snowball formation, the two data analysis channels of temperature and images can be compared with each other, and used to search for more subtle features of the data, particularly the presence of multiple scatters.

III. Scope and Objectives

In the operation of the snowball chamber, dozens of runs were conducted. These runs consist of hundreds of thousands of images that had to be analyzed, in order to get a clearer picture of the conditions necessary for future tests in the chamber. This project aims to create a comprehensive algorithm to detect the time and the location that snowballs form across a run, as well as automate the process of image analysis by improving its efficiency and creating a set of parameters for the analysis of each run.

IV. Recording Observed Nucleations

Before the image analysis could start, the positions of the snowballs' observed nucleation frames were manually recorded to provide a benchmark for the code's detected nucleation frames to compare to. First, all images in the run were converted into grayscale, as the color of the images was irrelevant to the snowball's location; only the brightness of the images was necessary to consider. Images that shared the same UNIX timestamp (Figure 1) were placed into the same data set, which typically consisted of 201 images[3].

For each data set, the frames were visually examined for a continuous growth across consecutive frames. Once one was observed, the numerical position of the frame in the data set where evidence of this growth first presented itself was noted, and the overall number of images in the data set would also be recorded. Following this, the size of each of the previously examined data sets in the run was added to the frame's position in its data set. The result was the overall position of the frame in the data set. This process would repeated for all data sets across the run to form an "answer key."

V. Initial Analysis

After determining the positions of the observed frames, the initial analysis could commence. First, the code converted each image in the run into 8-bit data, assigning each location in the image an integer pixel value from 0 to 255, with 0 corresponding to a completely black pixel and 255 signifying a totally white pixel. Following this, the images were sorted into data sets, using the UNIX timestamps as delimiters.

Background subtraction was then applied to remove unwanted noise from the images; ideally, this process would make the snowball stand out against the backdrop and increase the accuracy of its detection. Small variations in the images were consistently observed, even before the snowball began to form; thus, an average background was necessary to find, as opposed to simply using the first frame in the data set as the background. Snowballs were also rarely observed to form in the first fifty frames of the set. Therefore, the background calculation required finding the average pixel value of each location across the first fifty frames of the data set. These values were then subtracted from their corresponding locations across all frames in the data set, resulting in the removal of the majority of background noise[3] (Figure 2). A small number of data sets were eliminated from analysis at this stage, as their snowballs were observed to form in the initial fifty frames of the set and would interfere with background subtraction as a result.

Once background subtraction was applied, my work began with taking these images and finding the initial nucleation frame for each data set. First, a pixel value, starting at 1 and increasing to include all integers up to 255, was selected; this will be referred to as the threshold. Each of these thresholds was applied to determine the lowest pixel value at which snowballs could consistently be detected; this threshold could then be applied to all other runs.

Every data set was checked in sequence, with each of its background-subtracted images being examined in turn. For these images, a detection area of three pixels in width and three in height was examined; this area's initial position was the top left corner of the image, then moved across the image, then one row of pixels downward. If all nine pixels in an examined area had pixel values greater than or equal to the threshold, and maintained that property for the five images immediately following the image being examined, the criteria for snowball detection was met. The frame where this occurs is called the detected frame. For each data set, this process continued until all images in the data set were examined, or the detection criteria was met.

If a snowball was found, the position of the central pixel of the snowball's detected area, which took the form of [row, column], was recorded in a cell of 255 columns, one for each threshold value, and a number of rows consistent with the number of data sets in the run. The coordinates' position corresponded with the data set analyzed and the threshold used. Similarly, the detected frame's position in the run where the snowball was detected was recorded in a cell of the same dimensions and in the same position as the coordinates.

If no snowball was found, default values of [0,0] and the last frame in the data set was assigned for the coordinates and the detected frame, respectively.

Each data set was examined in this way for each threshold value. Following the examination of the final data set with a threshold of 255, the run's answer key was read into the code. After this, each data set's detected frame was reduced by its corresponding observed frame, such that the difference (Δ frames) was negative for an early detection and positive for a late one. The average and standard deviation for each threshold column were then taken; this yielded a result file with columns representing the threshold value, and the corresponding mean, uncertainty (always zero), and standard deviation of Δ frames (Figure 3).

VI. Further Modifications

The size of the examined images was changed from one size for all images in the run to a different size for the images in each data set. The code was also modified so that the last five frames of a data set were not checked for snowballs, as the detection criteria could not be met with the number of frames.

Variations in the size of the detection area were also implemented, with these areas expanding to five-by-five and two-by-two grids (Figure 3). The range of the areas would eventually increase further to include square grids with a side length between two and fourteen, inclusive. Detection areas with an even side length would have their detected points at the top left corner of the innermost two-by-two region of the detection area, while those with an odd side length would continue to have their detected coordinates at the center of the detection area.

In order to improve the efficiency of the code, a block of code was added before examining each data set. If the code had failed to detect a snowball in the data set at a lower threshold, the set was skipped and the default values described in Section V was inserted at the corresponding locations, as if the code had gone through the data set and could not find a snowball. At least fifteen and up to seventy percent of data set checks were eliminated through this method, significantly reducing the time it took for run analysis.

Finally, in order to aid with data analysis, the full cells showing the detected frames and Δ frames for each combination of data set and threshold were also presented as output files.

VII. Further Analysis

After analyzing runs control 02, control 08, and control a3 with a three-by-three area, error-bar plots of the main result file for each were made (Figure 3). Following these analyses, the same folders were re-analyzed using a five-by-five detection area. This typically reduced the threshold where the average Δ frames was closest to zero (zero point), but also reduced the number of thresholds that hovered near the zero value for average Δ frames. In an attempt to find a balance, two-by-two detection areas were used for these three folders; in contrast to five-by-five areas, the zero point was raised, but the number of thresholds close to zero increased somewhat. All three of these detection areas, in addition to a seven-by-seven area, were applied to all folders. Following this, the detection areas continued to grow by increasing the area's edge length by one pixel; each run was subject to this until the chosen area's result file showed no average values of Δ frames below zero.

VIII. Results

Following the completion of analysis in two-by-two, three-by-three, five-by-five, and seven-by-seven detection areas, the four plots from the result files for each run's analysis were plotted against each other (Figure 4) to confirm the differences detailed in section VII.

The zero point for each data set in the run was found by taking the threshold value in the result file that had the closest average Δ frames to zero; following this, these zero points were plotted in a histogram (Figure 4). The average zero point of these runs was then calculated, and the position of that threshold's detected frames in relation to their data set was then compared with those of the answer key in another histogram (Figure 5).

The Δ frames for each data set across all runs were then sorted by the detection area used to obtain them and plotted. For these plots, a low zero point would indicate a lower threshold necessary to detect a snowball, while low standard deviation of Δ frames would signify widespread agreement within the data sets that the chosen zero point was a good match. Although plotting the Δ frames of a five-by-five detection area resulted in a slightly higher zero point than that of a seven-by-seven detection area, the standard deviation was lower in the former, signifying greater agreement in the data (Figure 6). This showed that a five-by-five detection area appears to be a good estimate for snowball detection.

However, the varying levels of background noise and differing run sizes necessitate a run-by-run analysis. To accomplish this, each run's main result files were examined. The mean Δ frames for each zero point in the run's analysis was noted; of the zero points that were less than ten, the zero point with the closest Δ frames to zero was selected as the optimal threshold, and the detection area that yielded this zero point was selected as the optimal detection area. For each folder, the optimal detection area, the resulting zero point, the average Δ frames, and the standard deviation of the data sets for that zero point were recorded (Figure 7). This signifies the most accurate threshold and detection area for analyzing the run.

IX. Conclusion

The recommended final algorithm consists of a base analysis of a chosen run with a five-by-five detection area. After, the detection area should be raised from five-by-five until no zero point is detected, then lowered from five-by-five until the zero point is greater than ten. Following this, the zero point that has the closest Δ frames to zero of the zero points that are less than ten will be the optimal threshold; this value will be yielded by the optimal detection area for the run.

In the process of developing this algorithm, the image analysis was also automated sufficiently. The code's efficiency was greatly improved through the removal of unnecessary checks of data sets; a host of detection areas to check the data sets were also developed.

The framework for deeper analysis is now laid, especially in the domain of multiple scatters and other phenomena that are critical to understand the workings of the Snowball Chamber and its potential as a DM detector.

X. Acknowledgements

I'd like to thank my mentors, Assistant Professor Cecilia Levy and Associate Professor Matthew Szydagis, both of UAlbany, for providing ample guidance and support in this venture, as well as Yujia Huang of UAlbany for providing the files and initial analysis code.

This project was supported in part by the U.S. Department of Energy, Office of Science, Office of Workforce Development for Teachers and Scientists (WDTS) under the Science Undergraduate Laboratory Internships Program (SULI).

XI. References

[1] Profumo, S., Giani, L., & Piattella, O. F. (2019, October 12). *An introduction to particle dark matter*. arXiv.org. https://arxiv.org/abs/1910.05610.

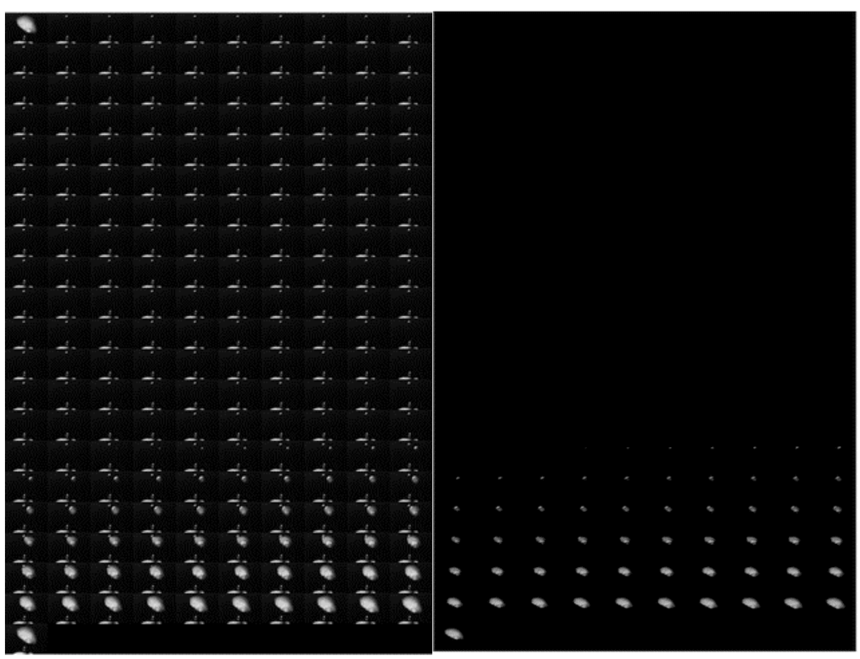
[2] Szydagis, M., Levy, C., Huang, Y., Kamaha, A. C., Knight, C. C., Rischbieter, G. R. C., & Wilson, P. W. (2021, May 22). *Demonstration of neutron radiation-induced nucleation of supercooled water*. arXiv.org. https://arxiv.org/abs/1807.09253.

[3] Huang, Y. (Fall 2021). *Image Data Analysis and design optimization of the Snowball Chamber* [Unpublished Master's Thesis]. SUNY Albany.

XII. Appendices

3.595861665_1212.215335

Figure 1: A typical image and its file name. The first part of the file name is a modified UNIX timestamp; the second part is the number of seconds that have elapsed since collection of the data set started.



(Images courtesy of the Snowball Chamber Collaboration)

Figure 2: Images of a data set before (left) and after (right) background subtraction was applied.

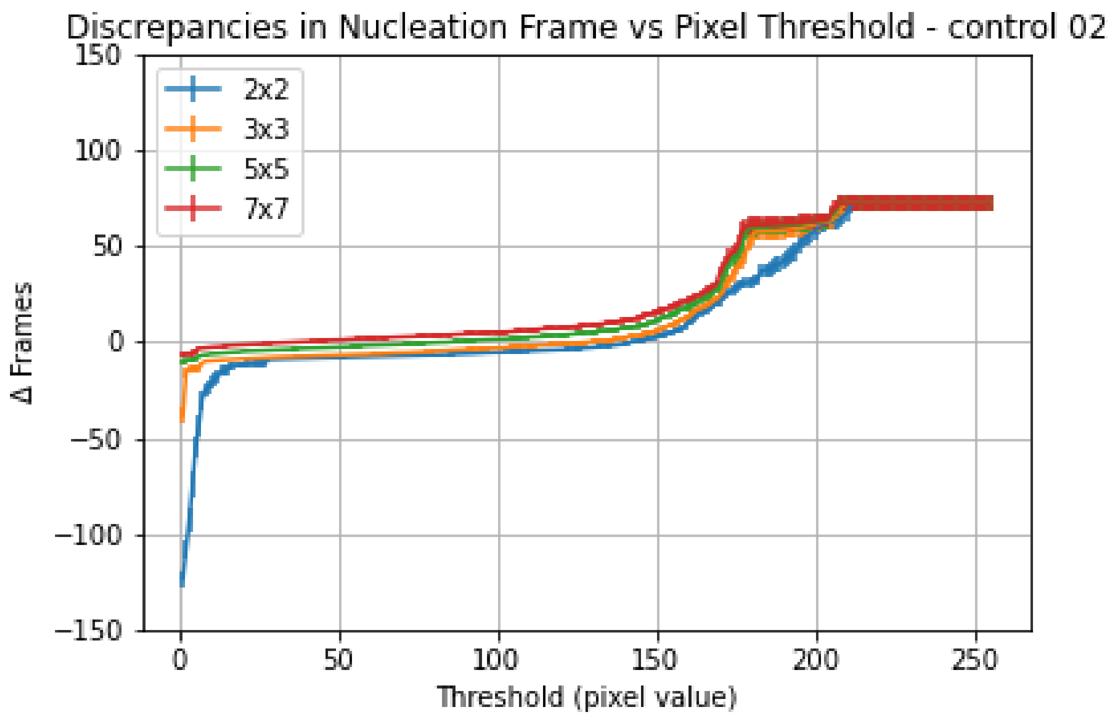


Figure 3: Plots of control 02 result files for various detection areas.

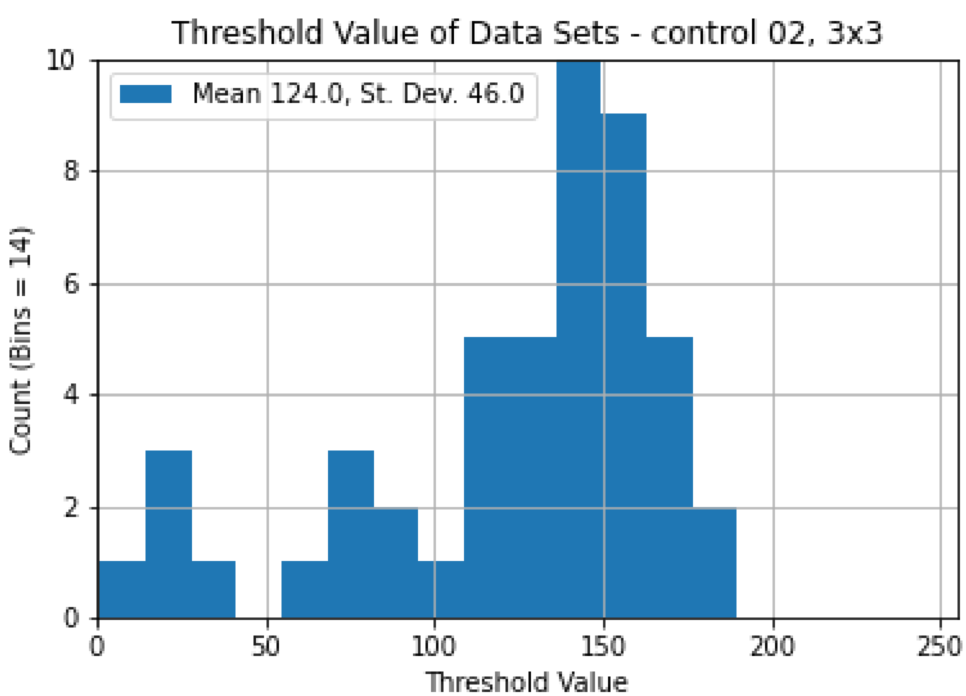


Figure 4: Histogram of zero points for each data set in control 02 (3x3 detection area)

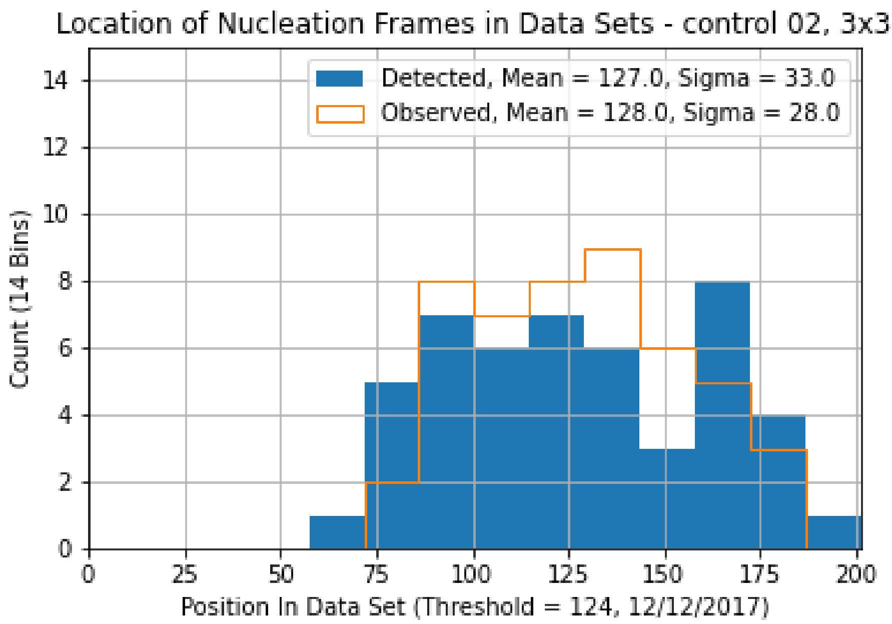
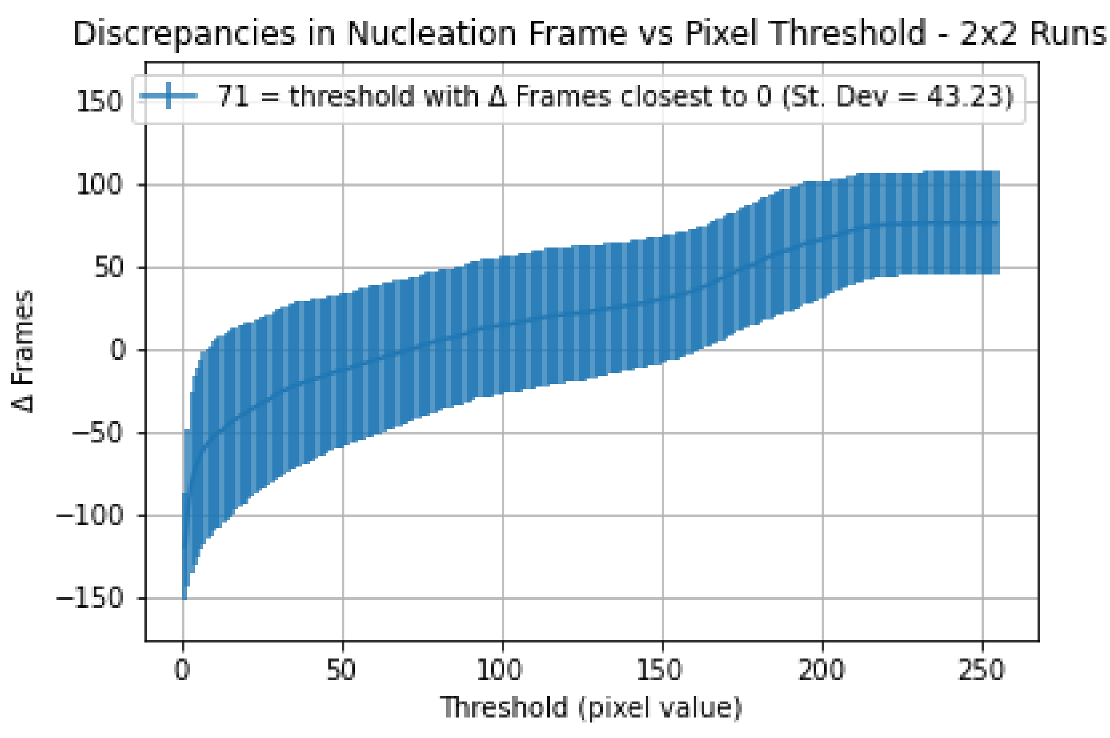
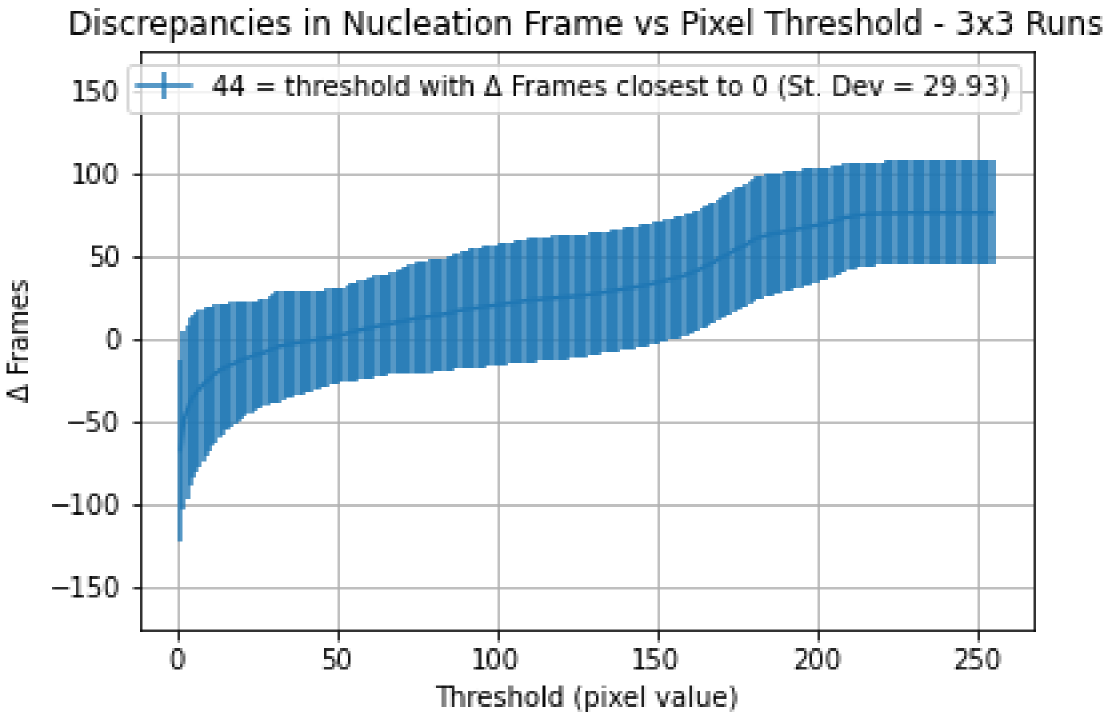
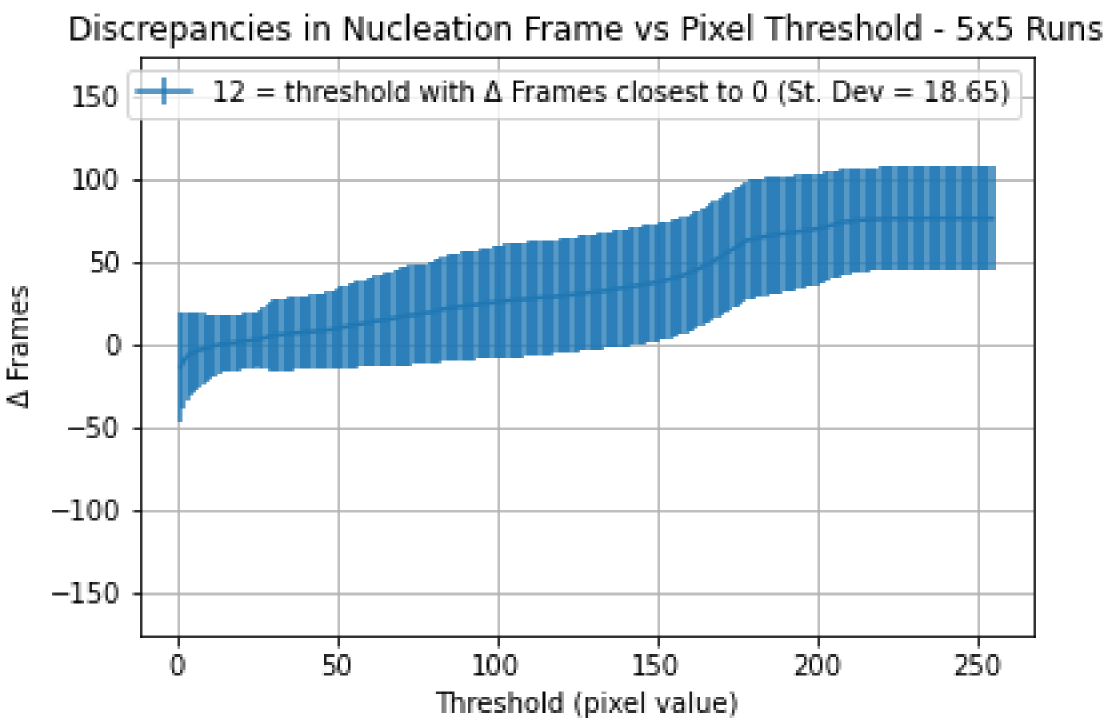


Figure 5: Histogram of position of detected frames in data set, using the above threshold, compared to the position of observed frames in the data set




Discrepancies in Nucleation Frame vs Pixel Threshold - 3x3 Runs
44 = threshold with Δ Frames closest to 0 (St. Dev = 29.93)
150
100
50
0
−50
−100
−150
Δ Frames
0
50
100
150
200
250
Threshold (pixel value)


Discrepancies in Nucleation Frame vs Pixel Threshold - 5x5 Runs
12 = threshold with Δ Frames closest to 0 (St. Dev = 18.65)
150
100
50
0
−50
−100
−150
Δ Frames
0
50
100
150
200
250
Threshold (pixel value)

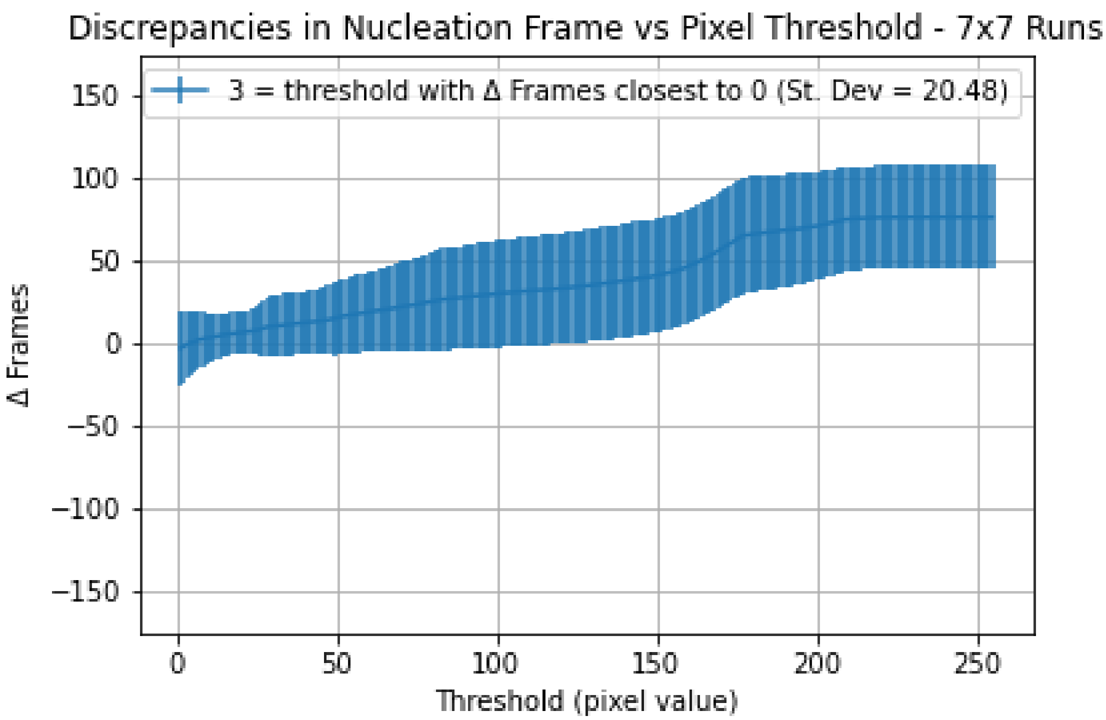


Figure 6: Deviation graphs across all analyzed data sets of all runs, with the zero point of these graphs and the standard deviation of Δ frames at said zero point listed.

Folder	Size	Zero Point	Mean	St. Dev
AmBe 01 - 8 bit	5x5	6	-0.036	2.794
AmBe 02 - 8 bit	3x3	8	0.07	0.353
AmBe 03 - 8 bit	7x7	8	0.021	3.066
AmBe B0 - 8 bit	5x5	4	0.074	0.667
AmBe B1 - 8 bit	11x11	8	1.0	4.665
AmBe blue 09 - 8 bit	5x5	5	-0.05	2.754
AmBe C0 - 8 bit	7x7	5	1.68	2.566
AmBe side A - 8 bit	5x5	3	-0.714	1.737
AmBe top A - 8 bit	5x5	4	-0.536	1.472
control 01 - 8bit for sub	9x9	2	0.275	2.691
control 02 - 8 bit	9x9	8	-0.042	0.741
control 03 - 8 bit	3x3	7	-0.333	0.667
control 05 - 8 bit	3x3	6	0.071	0.473
control 06 - 8 bit	5x5	3	0.39	2.202
control 07 - 8 bit	5x5	2	0.079	2.566
control 08 - 8 bit	3x3	5	0.391	0.774
control 09 - 8 bit	5x5	7	-0.194	0.664
control 10 - 8 bit	7x7	2	0.039	2.24
Control A0 - 8 bit	7x7	5	-0.037	3.555
control a1 - 8 bit	5x5	6	0.018	2.676
Control A2 - 8 bit	5x5	2	0.042	1.449
control a3 - 8 bit	4x4	7	0.808	0.568
control B0 - 8 bit	7x7	1	-0.111	3.112
Control B1 - 8 bit	11x11	1	-0.038	4.803
Control C0 - 8 bit	9x9	2	0.0	2.135
Cs-137 04 - 8 bit	3x3	7	-0.059	0.487
cs-137 05 - 8 bit	3x3	6	-0.383	1.268
cs137 06 - 8 bit	2x2	4	1.214	0.984
fiesta front w Be 10 - 8 bit	7x7	3	-0.188	2.297

Figure 7: Optimal detection area for each folder, with corresponding zero point, average Δ frames and standard deviation of Δ frames.