The Effects of Cosmic Ray Removal on Extended Source Shape Measurements with the Rubin Observatory

Nathan Woolsey
UC Santa Cruz and SCIPP

June 2024

Abstract

The Rubin Pipeline cosmic ray detection and removal algorithm is a powerful tool that may have unintended consequences. It removes data produced by cosmic rays from the exposure, which may result in information loss when a detected cosmic ray overlaps with a bright source. Understanding how these inconsistencies may arise allows for better use of the algorithm. I tested the algorithm using isolated cosmic rays taken from the commissioning camera dark images in conjunction with simulated data to measure any distortions in the shape measurements of extended sources. I found that the algorithm has a minimal effect on the shape measurement while adjacent to the footprint of an extended source, and distortion increases with proximity to the center of the extended source. This algorithm will be used on images before the creation of difference templates. It will clean initial exposures of artifacts while the templates are in the process of being created.

1 Introduction

A cosmic ray passing through a charge-coupled device (CCD) deposits charge in each pixel it passes through. Stars match the point-spread function (PSF) for how their light is spread onto the CCD, while galaxies are extended sources with unique shapes. A cosmic ray leaves a trail through each pixel, resulting in short streaks of bright pixels that stand out from the background. The difference in these varying morphologies is what the Rubin Observatory's cosmic ray detection algorithm uses to isolate the cosmic rays.

The Rubin cosmic ray detection and removal algorithm is a useful tool that allows us to detect and interpolate cosmic rays from their morphology alone. Normally, cosmic rays are removed through difference imaging, which requires a prior template or multiple exposures. By testing the cosmic ray removal algorithm, we gain more information about image processing that can help commissioning. We would like to use this algorithm to remove cosmic rays from difference images automatically.

The algorithm could also clean individual exposures during commissioning before difference imaging. To make clean sky templates, we have to start with images that may be contaminated with cosmic rays. The performance of this algorithm is important to constructing the most accurate template. Compared to other removal methods, having an accurate algorithm that affects only small regions of the image preserves areas without any cosmic rays at all.

Our tests were done using the Rubin Observatory's simulated data release DP 0.2 and images taken by the Rubin Observatory Commissioning Camera (ComCam). It contains simulated images using the imSim package and covers around 300 degrees of sky centered at right ascension -61.863 and declination -35.790.

2 Algorithm Test

We tested how the cosmic ray removal algorithm distorts the shape measurements of an extended source when a cosmic ray is within the object's footprint. Will the algorithm remove some excess galaxy light when interpolating over the cosmic ray? Will it leave a little bit of cosmic glow in the final product? These are the questions our test seeks to answer.

To test how much information is lost due to a cosmic rays influence, a cosmic ray was repeatedly placed at various positions around an extended source in two pixel interval steps. The overlay was taken from a dark image taken by ComCam, with the noise around the cutout manually removed. The ComCam dark had an exposure time of 300 seconds, and this set of images accurately reflects the cosmic ray distribution at the Rubin Observatory. The cosmic ray is placed 200 pixels away from the center of a simulated galaxy, the cosmic ray remover interpolates over it, and the galaxy's XX, YY, and XY moments are measured using the python package galsim. The cosmic is readded two pixels closer to the simulated galaxy, the algorithm runs again, and the measurements are retaken.

Here are some images saved in the middle of the process, where the cosmic ray intersects the top and bottom of the source's footprint, as well as a plot of the measured shape by the distance between the closest pixel of the cosmic ray and the nearest pixel in the perimeter of the source.

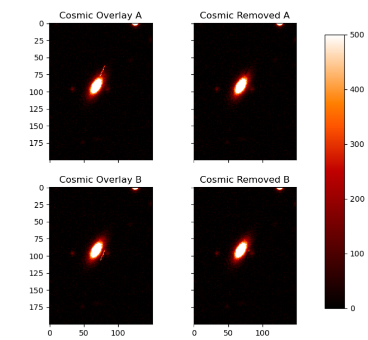

Figure 1: *This figure shows a side-by-side comparison of the source with the cosmic ray before and after removal. Negative values correspond to points where the cosmic ray is entirely within the source footprint.*

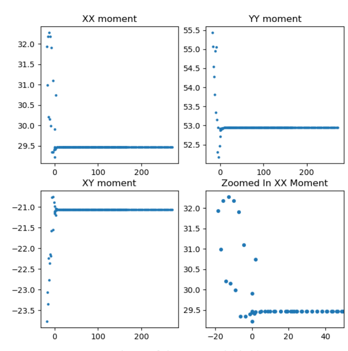

Figure 2: *This figure shows each image moment plotted by the closest approach between the pixels of the cosmic ray and the extended source. The negative values on the x-axis show how the shape distorts when the perimeter of the extended source completely encloses the cosmic ray.*

Figures 1 and 2 provide evidence that deviations greater than 1 unit from the measured image moment only appear 20 pixels from the source's edge. The looping behavior observed in the subplot of figures 5 and 2 is due to how the distance is measured. Since it measures the distance to the perimeter, the greatest distortion occurs when the cosmic ray is closest to the center. The shape of the cosmic ray is much more significant to measuring the closest approach distance here, as it changes which side of the galaxy the cosmic ray is closest to.

To test that the code detects a cosmic ray within the source footprint, the cosmic ray was placed closer to the edge of the source rather than its' centroid. Below is a histogram that displays each shape measurement, with a set of flags that show how the algorithm behaves.

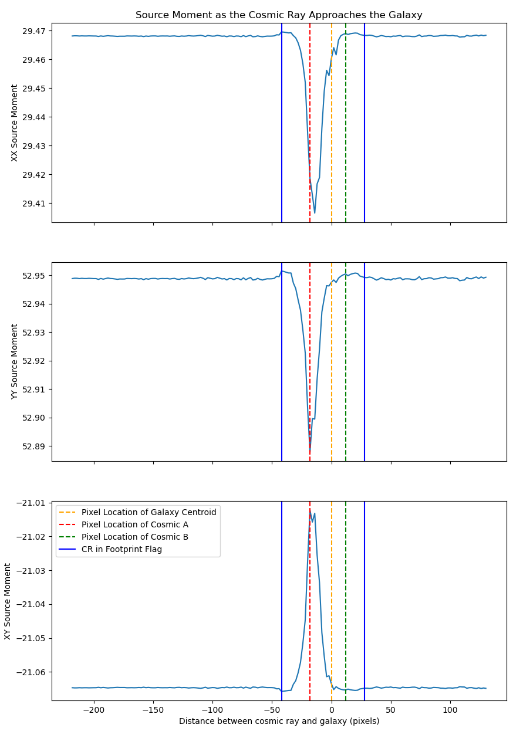

Figure 3: *This figure plots the XX, YY, and XY moments by relative position to the galaxy's centroid. Between the two vertical blue lines, the algorithm sets a flag for the source that notes if a cosmic ray is detected in the source footprint. The vertical red line marks the value of each moment for the first two subfigures (Denoted Cosmic Overlay A and Cosmic Removal B) in figure 4. Similarly, the vertical green line indicates each moment for the second set of subfigures in figure 4. Lastly, the vertical yellow line denotes the position of the galactic centroid.*

The blue lines represent when the cosmic enters the source footprint, the red dashed line corresponds to row A in figure 4, the orange line is the source's centroid, and the green line corresponds to row B in figure 4. The X axis here represents the placement of the Cosmic Ray's cutout

Each subplot of figure 3 shows a noticeable spike in measurement when the cosmic approaches the galactic centroid along the vertical axis. The cosmic ray removal algorithm certainly distorts the shape measurement from the baseline, but it does so in a simple manner. Additionally, the difference in shape in this case is minimal, only affecting each measurement by about 0.05. For extended sources, when the scale of each image moment is in the double digits, there is hardly any difference. Outside of the region where the cosmic is within the source footprint, repeated runs show remarkable stability with measurements. This stability will mean that shape measurements should be accurate for extended sources even without difference imaging.

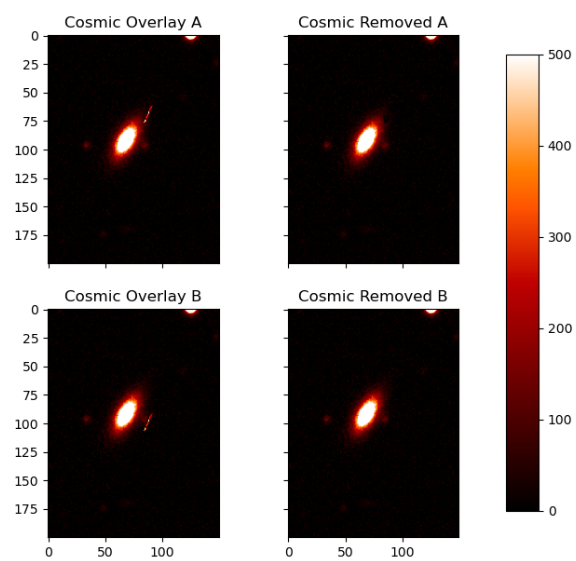

Figure 4: *This figure contains side-by-side images that compare the exposure before and after the cosmic ray was removed. The top set of figures corresponds to the position of the red line in figure 3, while the bottom set corresponds to the position of the green line in figure 3.*

Looking at figure 4, we can see that a small amount of galaxy light has been removed from the edge of our galaxy at about (85, 76). This dark region shows a noticeable decrease in the light from our galaxy. To quantify the shape distortion a bit better, we took the set of points in our cosmic ray, and found the closest distance between the cosmic ray and the edge of the extended source.

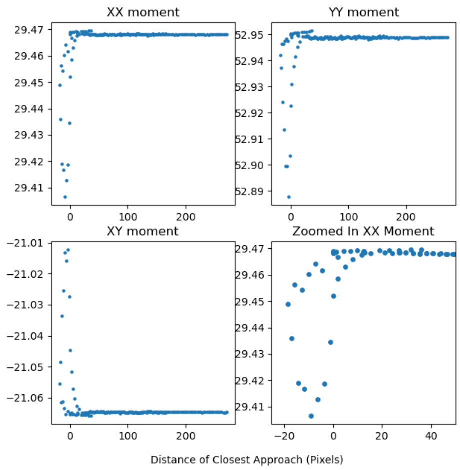

Figure 5: *This image is in the same style as figure 2*

From figure 5, we can see that the shape measurement blows up at around a distance of 20 pixels. We also see the spike described in figure 3. Knowing what range the shape will get distorted with a more accurate measurement will allow us to know which cosmic rays will be most problematic.

3 Next Steps

Our analysis, based on a single galaxy and a single cosmic ray, is just the beginning. For further study, more tests with various combinations of galaxies and cosmic rays are needed. These tests will provide a more comprehensive understanding of the algorithm's behaviors and its impact on the objects it interacts with.

The code used for this project can be found at https://github.com/TheNathan27/RubinAlgorithmTest.

4 Conclusion

Cosmic rays significantly affect the shape measurement when they directly overlap the measured source. When the cosmic ray just barely passes by the galaxy, its removal does not cause significant levels of distortion. This result tells us that the cosmic ray removal and detection algorithm produces the best results on images that lack direct overlap between cosmic rays and other objects in the image. For images that do result in significant overlap, difference imaging will be required to remove problematic cosmic rays.

The cosmic ray algorithm is useful for image processing but cannot do everything. Difference imaging is still a requirement for clean fields. Once the sky templates are made, this should not be an issue. One note is that to run this algorithm on dark images, a PSF must be manually specified. Our testing shows that the algorithm is excellent at what it does outside of source footprints but struggles in cases of overlap. The application of other techniques best addresses this minor issue, and the advantage of its efficiency warrants a place in the image processing pipeline.

5 Acknowledgements

I want to express my deepest appreciation to Steven Ritz, who made this project possible. We worked closely together throughout the entire process, and his advice and encouragement helped me stay focused on the end goal. I learned so much while working on it, and I cannot thank him enough for this opportunity.

I want to give a special thanks to Chris Waters. He helped answer my questions about LSST software, and would help me troubleshoot when I would get stuck on a problem for a while. I would also like to thank Craig Lage, who helped me work with and understand the ComCam dark data, and Jim Chiang, who set up a custom repository for the simulated data for me to use. This project would not have been possible without them.

I would also like to thank my peers Duncan Wood, Renée Nichols, and Adrian Shestakov for their help troubleshooting and providing helpful feedback during the course of this project.