Data Preparation

Yifan Zhou

October 18, 2014

1 SUMMARY

1. Cross correlation has the best performance for locating the centroids of star images.

2. Bicubic interpolation generates smaller RMS fluctuation comparing to `FSHIFT.pro` when shift images, but does not strictly obey flux conservation. `FSHIFT.pro` strictly keeps the conservation of flux. Bicubic interpolation causes artifacts around the star images.

2 CENTROID OF PRIMARY STAR

4 methods to locate the centroid of the primary objects were tested for all 78 cosmic ray removed images of ABPIC-B. I compared the 4 sets of results in order to determine the best method to find star image centroid in this situation where the image is slightly saturated.

2.1 METHOD INTRODUCTION

1. `CNTRD.PRO`
   IDL routine `CNTRO.PRO` locates the position where the X and Y derivatives go to zero. The

search area of `CNTRO.PRO` can be adjusted with parameter `EXTENDBOX`.

2. `GCNTRD.PRO`
   IDL routine `GCNTRD.PRO` fits the star image with a 2D-Gaussian function to search for centroid. The search area cannot be defined in this routine. `GCNTRD.pro` is likely to fail when the initial guess of centroid coordinates are far away from the true values. This is inconvenient for pipeline scenario, where images of different dithering positions are assigned with same initial values for fitting. In practice, I ran `CNTRD.PRO` before `GCNTRD.PRO` to provide initial values for Gaussian fitting.

3. Cross Correlation
   Calculate the cross correlation value to align images. The images taken with different rolling angle were treat separately. Images taken in 1st, 3th, and 5th orbits were calculated cross correlation with the first image in 1st orbit and images taken in 2nd, 4th, and 6th orbits with first image in 2nd orbit.

4. World Coordinate System(WCS)
   Use WCS to align every image. The treatment with different rolling angles is the same as cross correlation method. Centroids were calculated with IDL routine `XYXY.pro`.

2.2 Result Comparison

Figure 2.1 and 2.2 demonstrates the distributions of centroid offsets of `GCNTRD.pro`, cross correlation, and WCS relative to the `CNTRD.pro` method in x and y direction.

According to the two plots, most of results calculated with all four methods are constrained within ± 0.2 pixels. Both in x and y directions, the histograms of `GCNTRD.pro` offsets have a fairly strange shape, while the distributions of cross correlation and WCS methods' offsets show a Gaussian like shape. In x-direction, the distributions of cross correlation and WCS have similar behavior that the centers of two histograms both have ~ -0.1 pixels offset. In y direction, histogram of cross correlation demonstrates that the results for cross correlation and `CNTRD.pro` are well agreed. The histogram of WCS is skewed and has a ~ -0.1 pixels offset.

To sum, `GCNTRD.pro` method has the worst performance. In x-direction, the results for cross correlation and WCS agrees well and in y-direction the results for cross correlation and `CNTRD.pro` agrees well. Thus cross correlation method performs best.

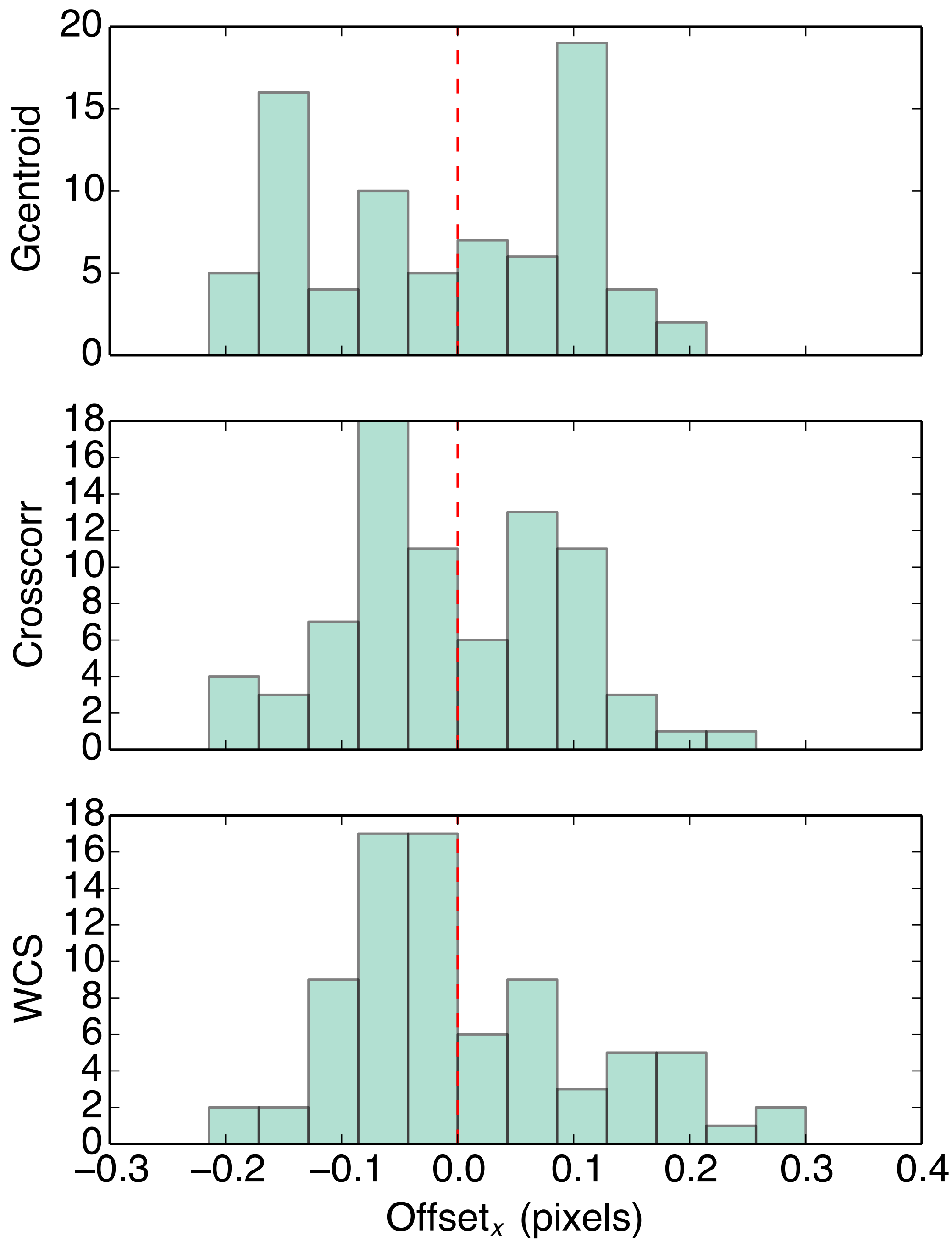

Figure 2.1: x-direction offset histogram

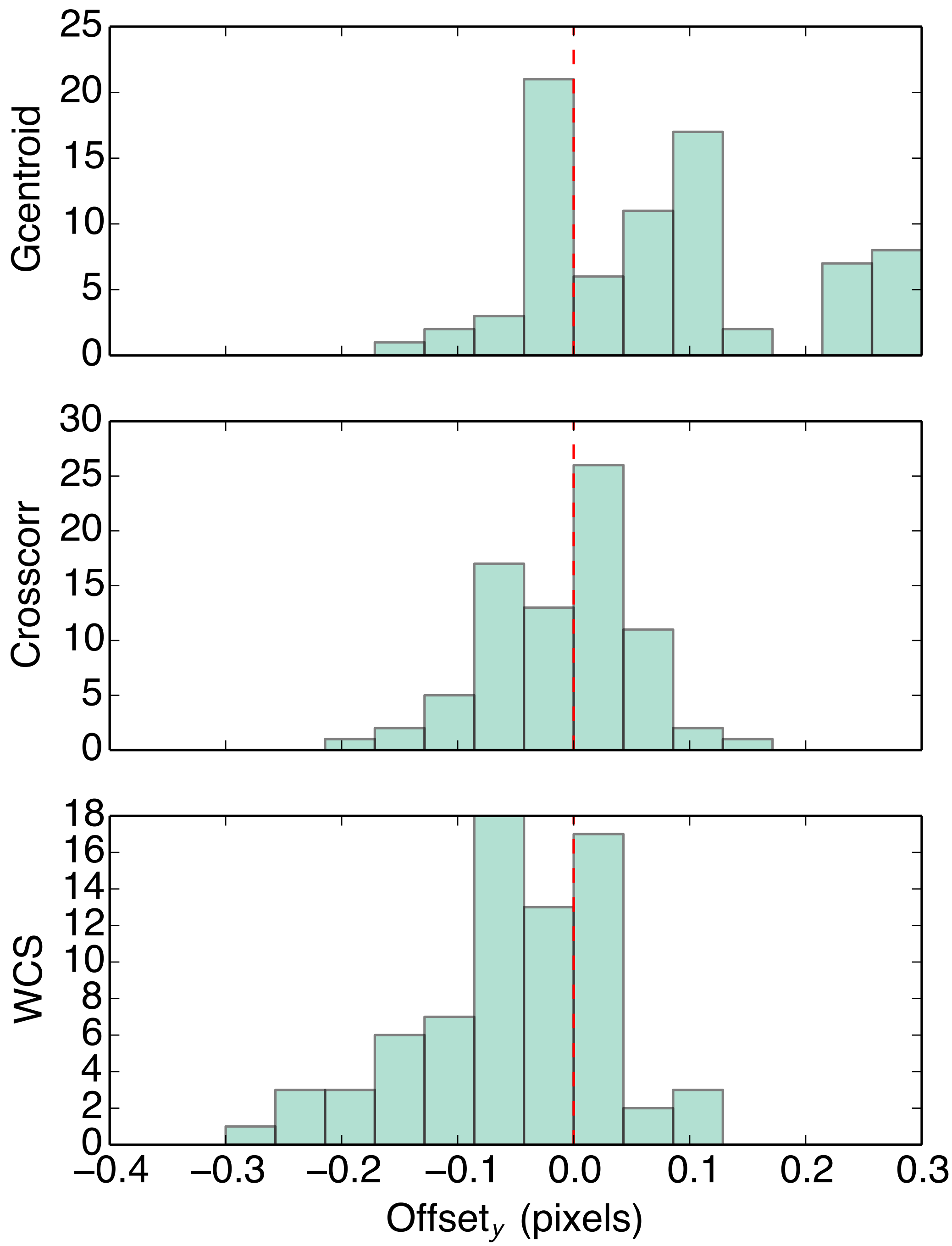

Figure 2.2: y-direction offset histogram

3 Image shift and interpolation algorithm

I tested two image shift methods that uses two different interpolation algorithms. The two algorithms are bicubic interpolation and bilinear interpolation(used by IDL routine `FSHIFT.pro`).

To evalueate the two shift methods, the image is shifted in a random direction with a specific length and then shifted back. Both the root mean square and mean flux change between the original image and the modified image will be calculated as following:

$$\Delta_{\text{rms}} = \frac{\sqrt{(\text{img}_0 - \text{img}_{\text{shifted}})^2}}{N_{\text{pixels}}} \tag{3.1}$$

$$\bar{\Delta} = \frac{\text{img}_0 - \text{img}_{\text{shifted}}}{N_{\text{pixels}}} \tag{3.2}$$

The results are demonstrated with figure 3.1.

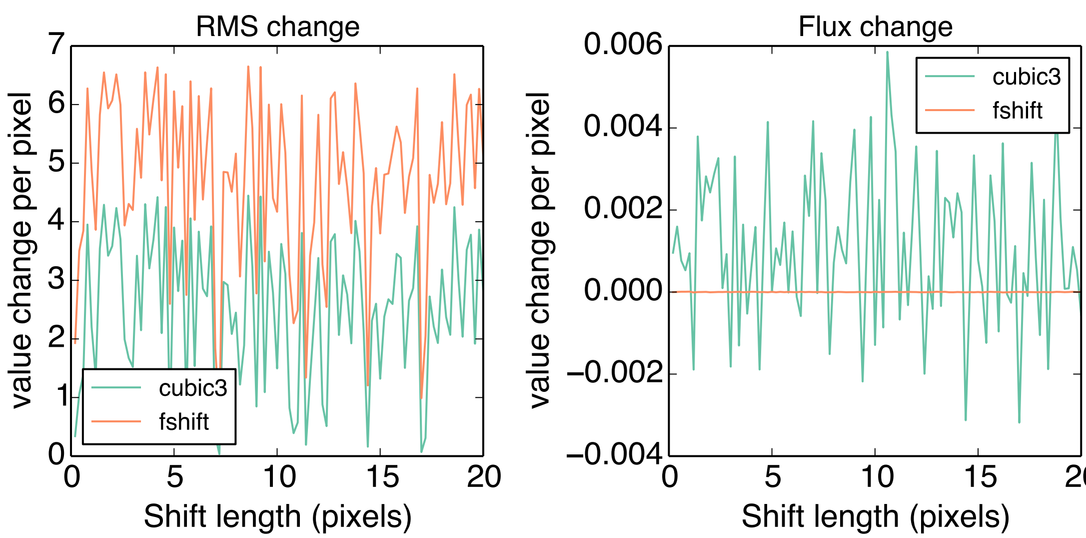

Figure 3.1: Root mean square and mean flux change betwenn the original image and the modified image as a function of shift length.

1. bicubic interpolation comes up with a smaller RMS change, thus the error generated by image shift will be smaller with a bicubic interpolation.

2. `FSHIFT.pro` strictly keeps flux conserved, while bicubic interpolation ends up with a $\sim \pm 0.003$ per pixel change in flux.

3. bicubic interpolation will generate ringing artifacts(figure 3.2)

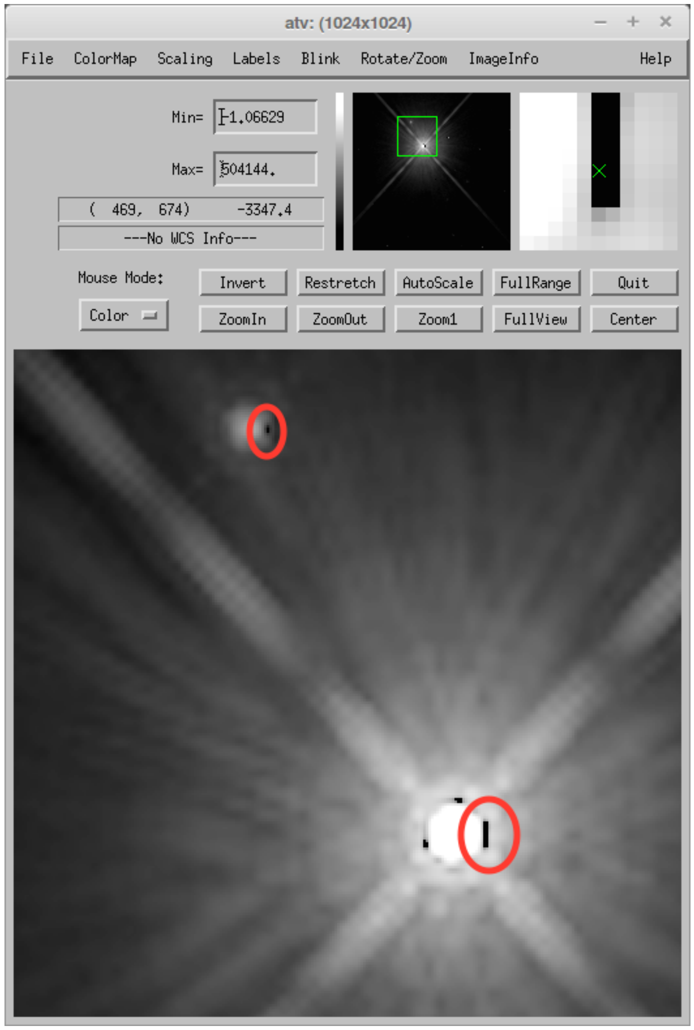

Figure 3.2: Ringing artifacts generated by bicubic interpolation.