Draft version October 4, 2016
Preprint typeset using LaTeX style AASTeX6 v. 1.0

# TESTS ON IMCOM: GENERATING NIRCAM MOSAICS

Sara Crandall and Brant Robertson
Department of Astronomy & Astrophysics, University of California Santa Cruz
211 Interdisciplinary Sciences Building
Santa Cruz, CA 95064, USA

This document logs the steps taken to generate mosaic mock images for NIRCam use.

## 1. AN INTRODUCTION TO IMCOM

IMage COMbination (IMCOM) may be a useful tool for generating mosaic mock images for NIRCam. This software was designed by Rowe et al. (2011) for use on WFIRST, and it allows undersampled images to be combined into an oversampled image. This would be useful for NIRCam, as some bands provide undersampled images. DRIZZLE is commonly used for image combination, but may have resolution loss resulting from reduced control of the PSF in the combined image. The Fourier/linear algebra technique is also useful, but can be less reliable when positions are not well known. A least-squares tool is beneficial in that output noise can be minimized. IMCOM uses a combination of these three methods to reconstruct an image with a controlled PSF, knowing the positions of exposures, that also minimizes noise.

To ensure that IMCOM is the right tool for NIRCam use, we will run a few tests. As described in the software's documentation, a baseline set of arguments are required. The command line looks like

$$./\mathrm{imcom}\ \ < \mathrm{config\_file} > < \mathrm{U/S} > < \mathrm{U/S\_max} > < \mathrm{U/S\_tol} >. \tag{1}$$

Here, the input $< \mathrm{U/S} >$ will either minimize the leakage objective, $U_\alpha$, (the local leakage or difference between the desired PSF and the reconstruction) or the noise covariance for each output pixel, $\Sigma_{\alpha\alpha}$. $< \mathrm{U/S\_max} >$ defines the maximum leakage objective or noise covariance, $U_\alpha^{\max}$ and $\Sigma_{\alpha\alpha}^{\max}$ respectively, depending on the previous option. $< \mathrm{U/S\_tol} >$ defines $\Delta U_\alpha^{\mathrm{tol}}$ or $\Delta\Sigma_{\alpha\alpha}^{\mathrm{tol}}$ that specify when the iteration should stop. The program will finish when $U_\alpha^{\max} - \Delta U_\alpha^{\max} < U_\alpha \leq U_\alpha^{\max}$ or $\Sigma_{\alpha\alpha}^{\max} - \Delta\Sigma_{\alpha\alpha}^{\max} < \Sigma_{\alpha\alpha} \leq \Sigma_{\alpha\alpha}^{\max}$.

## 2. TEST_0

Test_0 will determine which IMCOM inputs are useful for our purposes by varying the above options.[1] IMCOM has two examples included with the package (userxy0 and userxy1) which we will use those for our initial tests. Userxy1 gives the user more freedom to designate the input pixel locations.

To begin, example userxy0 will be used and the U/S option will be varied. The following inputs were used:

$$\mathrm{config\_example\ U\ 1.d} - 3\ 1.\mathrm{d} - 6\text{[2]} \tag{2}$$

and

$$\mathrm{config\_example\ S\ 1.d} - 3\ 1.\mathrm{d} - 6. \tag{3}$$

The results are seen in Figure 1 along with an ideal output provided with the IMCOM package. Residuals between the ideal and "U" & "S" options are also shown. All three residuals are near zero.

Next, the userxy0/userxy1 option will be varied. The input

$$\mathrm{config\_example\ U\ 1.d} - 3\ 1.\mathrm{d} - 6 \tag{4}$$

is used. Figure 2 shows the results, including the residuals from the ideal output minus the IMCOM output. The

sacranda@ucsc.edu, brant@ucsc.edu

[1] There are other inputs that can be specified such as the convolution and objective kernels, designated as "soft inputs" in Rowe et al. (2011), but they won't be varied at the moment.

[2] Double precision floating point numbers are used

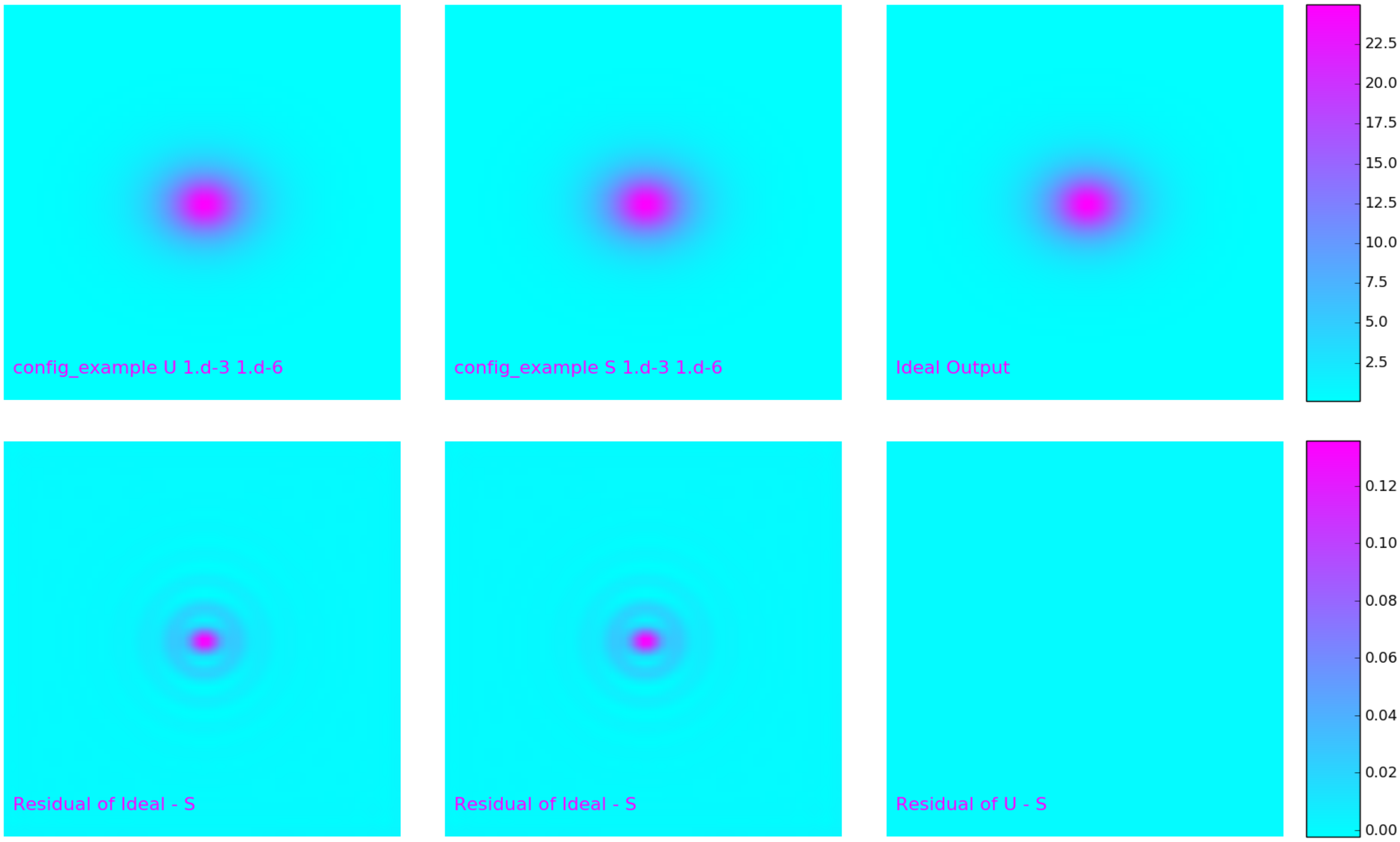


**Figure 1**. The top-left and top-center images use the "$U$" and "$S$" options, respectively. The top-right image is the ideal output included with IMCOM. The bottom row shows residuals that are near zero.

userxy1 option is less defined and has a larger residual. This may not be surprising as the userxy1 option allows a user input for pixel centering, and may result in a more realistic output image.

## 3. TEST_1

We will move forward with using the userxy1 example, as the pixel locations need to be provided. We will also focus on minimizing the leakage objective, as we are concerned with how well the output image matches the inputs.[3] With these options in mind we will vary U_max and U_tol, and call this exercise "Test_1." U_max and U_tol were varied from 1.d-1 to 1.d-10, as IMCOM gives warning messages for smaller values. Figure 3 shows several examples of this test along with the residuals of the ideal minus the output images. We see that smaller values as well as small differences in U_max and U_tol result in smaller residuals between the output and ideal images.

## 4. TEST_2

We now move on to "Test_2," where we see how well IMCOM can handle noise in the input images. First, we will use small U_max and U_tol inputs to decrease the differences in output and ideal output images (as demonstrated in Test_1). We use the IMCOM input

$$\text{config\_example U } 1.\text{d} - 9 \ 1.\text{d} - 10 \tag{5}$$

for the userxy1 example. The overall goal here is to investigate how the differences in the IMCOM output and ideal images are affected by the signal to noise ratio ($S/N$).

To add noise, we draw random values from a Poisson distribution given by

$$P(k) = \frac{\lambda^k e^{-\lambda}}{k!}, \tag{6}$$

[3] Minimizing U_tol can result in a reconstruction that is not trustworthy, as the algorithm may not reach an ideal reconstruction.

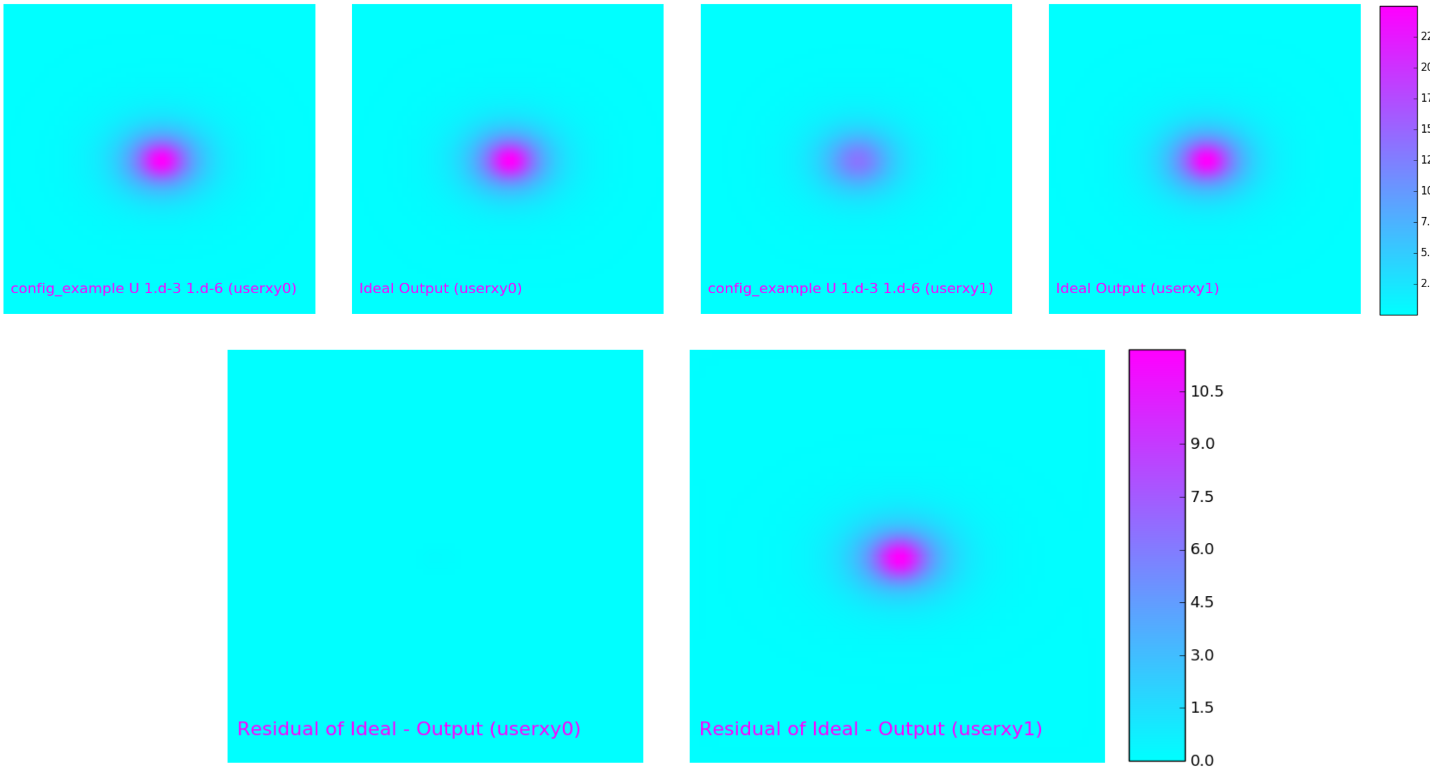


**Figure 2**. The top images use the userxy0 example with the left, center, and right panels being the IMCOM output, ideal output, and residual respectively. The bottom images use the userxy1 example with the left, center, and right panels being the IMCOM output, ideal output, and residual respectively.

and add to existing input images. Here $\lambda$ depends on the $S/N$, which we will vary. That is

$$\lambda = \left(\frac{< S_p >}{S/N}\right)^2, \tag{7}$$

where $< S_p >$ is the average signal from the input images given by

$$< S_p >= \frac{\sum_i p_i \Delta A_i}{\sum_i \Delta A_i}. \tag{8}$$

Here, $p_i$ is the pixel and $\Delta A_i$ is the area of the pixel.

We generate noisy images for $0 < S/N \leq 15$ in increments of 0.25. Figure 4 shows the input exposures, IMCOM output, ideal, residual images for $S/N = 5$ as an example.

We further explore how IMCOM handles noise by combining images one exposure at a time (see Figure 5). However, the output of combing only two exposures is quite poor, and so by putting each combination output on the same scale, the images are washed out????

To quantify how the noise is being handled by IMCOM, we can plot $\chi^2$ vs. $S/N$ (see Figure 6). We define $\chi^2$ as

$$\chi^2 = \frac{1}{N-1}\sum_i^N \frac{(H_i - I_i)^2}{|H_i|}, \tag{9}$$

where $H$ & $I$ are the output and ideal image for pixel $i = 1, 2, ...., N$. This plot is promising, as $\chi^2$ is reasonable for values $S/N > 2$. There are deviations from the general trend around 7$< S/N < 8$, but it is within normal expected error.

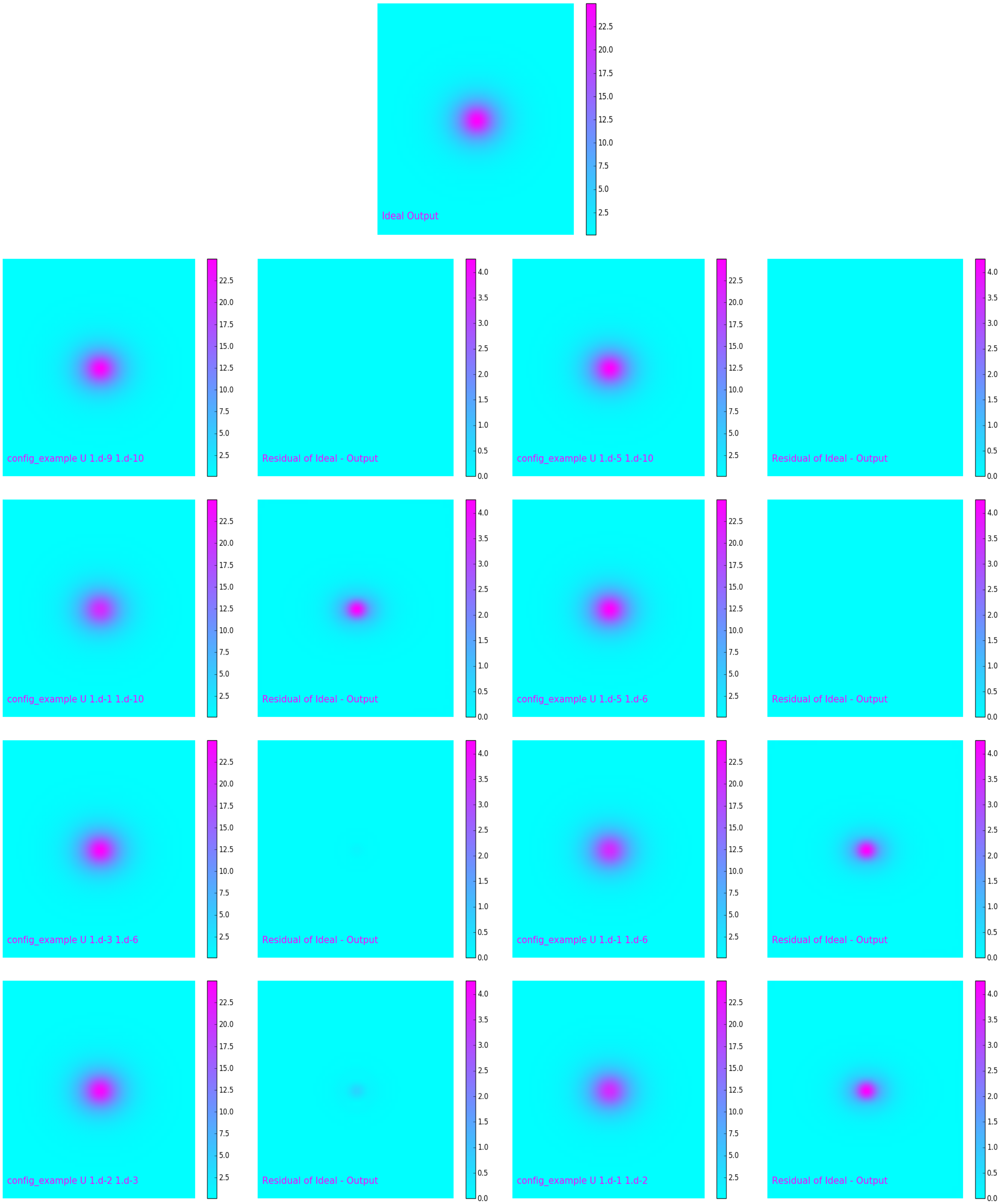


**Figure 3**. Output images using userxy1 example, U option, and varying U_max & U_tol.

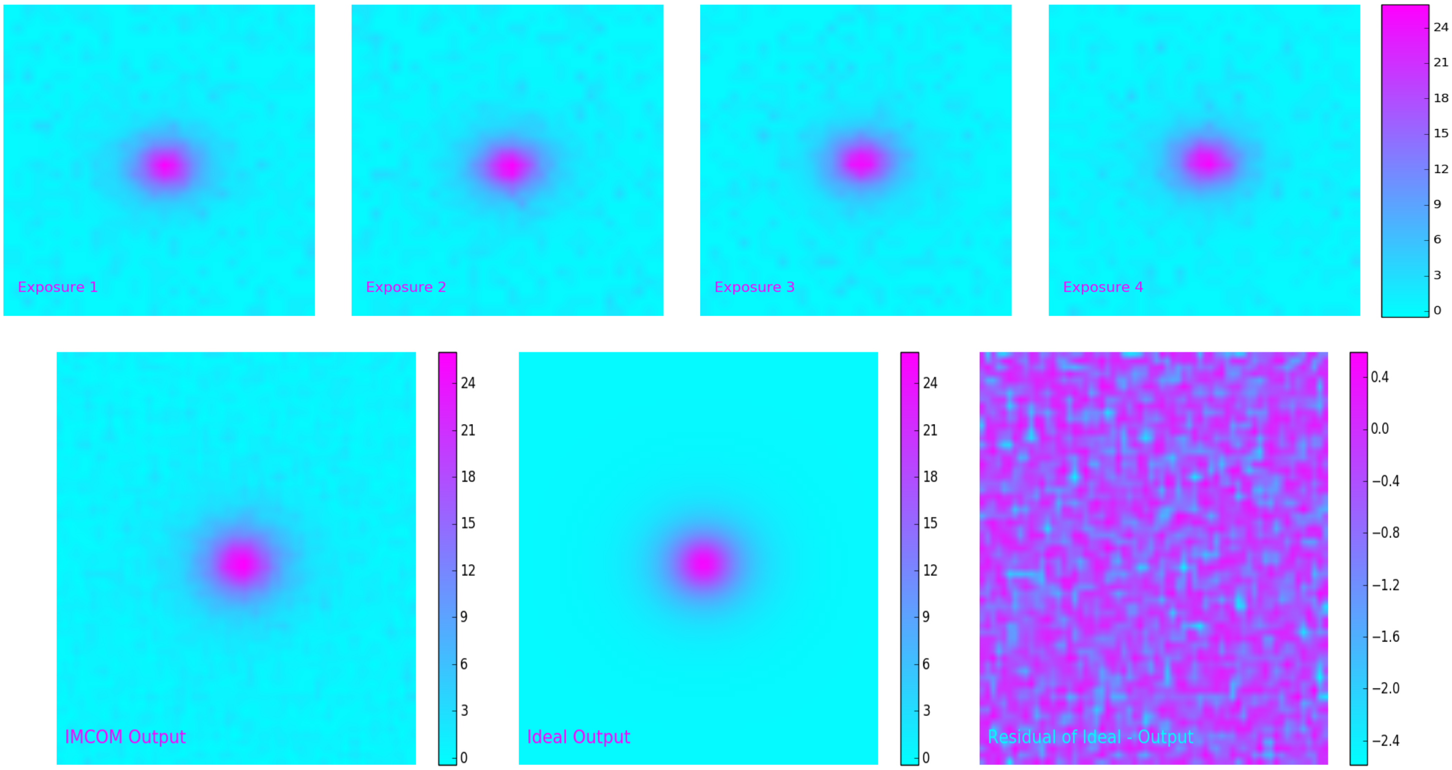


**Figure 4**. Noise added to input images for the $S/N = 5$ case.

## 5. TEST CONCLUSIONS

Because IMCOM (Rowe et al. 2011) makes use of the benefits from common mosaicing tools, it is a ideal software package to generate JWST NIRCam mock mosaic images. It will be useful in that it gives control of the PSF in the combined image, minimizes noise, and is used by knowing the positions of the pixels.

After running several tests on provided example images, it appears that IMCOM allows enough input freedom for the user. As an initial test, IMCOM can handle added noise to input images for at least $S/N > 2$.

In conclusion, IMCOM is a promising tool for NIRCam use, and we will continue to use it by testing mock images generated by Christopher Willmer.

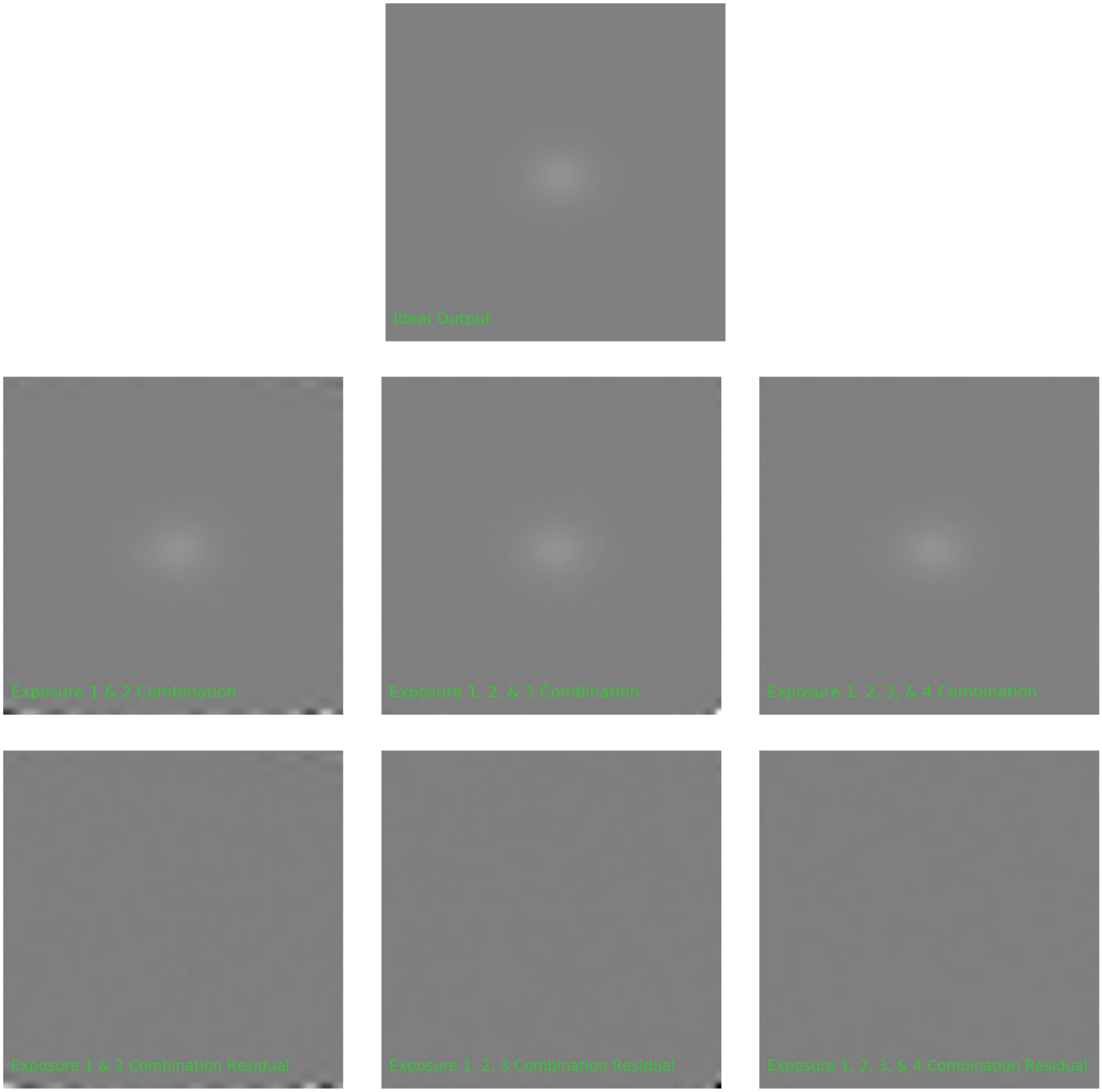


**Figure 5**. Noise added to input images for the $S/N = 5$ case. Each middle panel shows the image combination with increasing exposure number. The bottom panels show the residuals between the corresponding IMCOM outputs and the ideal image.

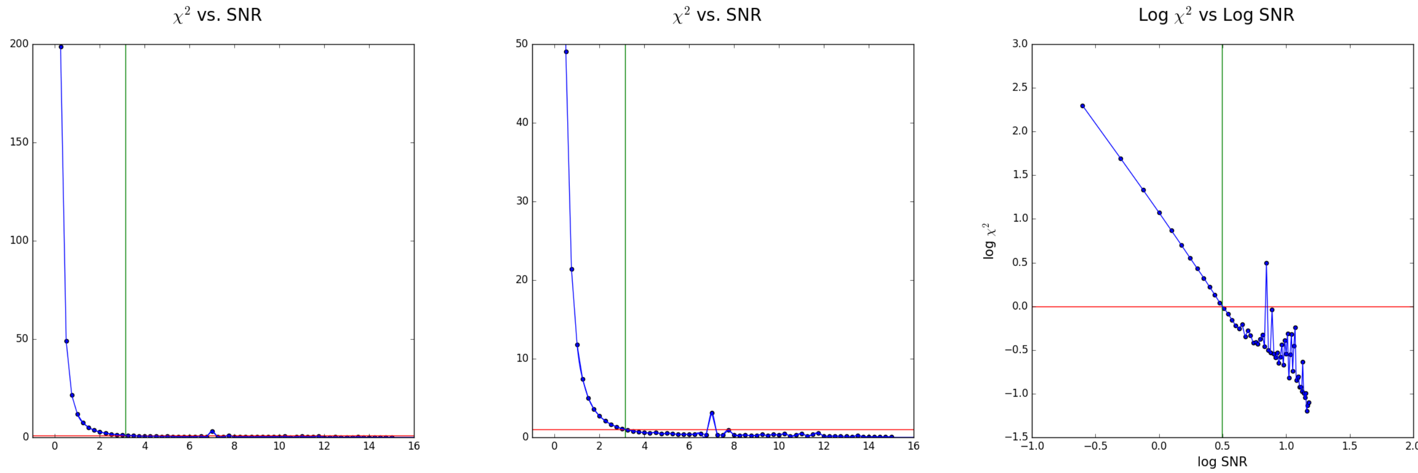


**Figure 6**. $\chi^2$ vs. $S/N$. for $S/N \leq 15$. The center plot is a zoomed in version of the left, and the right plot shows a log of $\chi^2$ vs. $S/N$. The red line shows $\chi^2 = 1$ and the green line shows where $\chi^2$ intersects with 1.

## REFERENCES

Rowe, B., Hirata, C., & Rhodes, J. 2011, ApJ, 741, 46