

Quantitative Performance of the Mopex Multi-Frame Outlier-Detection Algorithm

RUSS LAHER, ANDREW GRANT, AND FAN FANG

Spitzer Science Center, California Institute of Technology, M/S 314-6, Pasadena, CA 91125; laher@ipac.caltech.edu

Received 2008 August 13; accepted 2008 November 10; published 2008 December 3

ABSTRACT. The Mopex software is used at the Spitzer Science Center (SSC) to produce co-added and mosaicked images from sets of individually processed *Spitzer* images. Until now, quantitative studies of the performance of Mopex’s outlier-detection methods had never been performed. This particular study focuses only on Mopex’s multiframe outlier-detection algorithm, and future studies are still needed to characterize its so-called box and dual methods. The performance of the multiframe method varies with two adjustable parameters, Γ_{TOP} and Γ_{RM} . For a given Γ_{TOP} value, we computed the completeness (C) and reliability (R) of the outlier detection for 101 discrete values of Γ_{RM} uniformly distributed in the full range of possible settings for this parameter, which lie continuously between 0 and 1, inclusive. We characterized the C and R performance as a function of Γ_{RM} in this manner for Γ_{TOP} values of 2, 2.5, 3, 5, and 10 for image data in all four IRAC channels (infrared passbands). Not surprisingly, the performance for IRAC channel 3 is relatively poor because the image data for this channel are markedly noisier. The best performance was obtained for a Γ_{TOP} value of 3, and this applies to all four channels. Generally, setting Γ_{RM} low will maximize completeness at the expense of reliability, and vice versa for setting Γ_{RM} high. For example, for IRAC channel 1 and $\Gamma_{\text{TOP}} = 3$, setting $\Gamma_{\text{RM}} = 0.3$ gives $C = 83\%$ and $R = 41\%$, and setting $\Gamma_{\text{RM}} = 0.8$ gives $C = 52\%$ and $R = 86\%$.

Online material: color figures

1. INTRODUCTION

This brief document presents the results of our measurements of the completeness and reliability of the Mopex software’s multiframe outlier-detection algorithm, which is used in routine image-data reduction at the Spitzer Science Center (SSC).¹ These results are the first of their kind. Our study involved the development of Monte Carlo simulations to compute a realistic data set of cosmic-ray flux densities and pixel positions. The simulated cosmic-ray data consist of isolated pixels, clumps of pixels, and streaks, which we deliberately constructed to mimic the natural cosmic rays observed in real IRAC images. We added the simulated cosmic-ray data into a large number of *Spitzer Space Telescope* Infrared Array Camera (IRAC) images to create test image-data sets for all four IRAC channels (infrared passbands). These test data, along with the known ground truth for the positions of the simulated cosmic rays in the test images, permitted us to apply the well-known formulas for completeness and reliability and compute the performance for all four IRAC channels as a function of the two primary adjustable parameters of Mopex’s multiframe or temporal outlier-detection algorithm,²

which are Γ_{TOP} and Γ_{RM} (the corresponding exact names of these parameters in the Mopex software’s input-configuration file, *mopex.nl*, are *TOP_THRESHOLD* and *RM_THRESH*, respectively).

In the sections that follow, the Mopex multiframe outlier-detection algorithm, the input images employed in this study, and our Monte Carlo simulations of cosmic-ray data are described. The scoring method, its software implementation, and the detection-performance results are then presented. A discussion of the results then ensues and, finally, concluding remarks are made.

2. MOPEX MULTIFRAME OUTLIER-DETECTION

The Mopex multiframe or temporal outlier-detection algorithm exploits the transient nature of most types of outliers, in particular the cosmic rays naturally occurring in the space environment that impinge on digital-imaging detectors. The input images to Mopex, which have been acquired at different times and are generally slightly rotated and offset from one another spatially, are first interpolated by Mopex onto a common pixel reference frame. The resulting interpolated images, which usually have smaller pixels than the input pixels, will also have coregistered pixels among the interpolated images, as well as a ragged edge of pixels with no coverage in each interpolated image. If the interpolated images are sorted in time and stacked

¹ The Spitzer Science Center Web site is found at <http://www.spitzer.caltech.edu>.

² The Spitzer Mosaicker software is available online at <http://ssc.spitzer.caltech.edu/postbcd/doc/mosaicker.pdf>, pp. 35–36 and 40–42.

like plates in a cafeteria, then a vertical line through the stack at a given interpolated-pixel location represents a time axis, and the interpolated-pixel values along the line are independent image observations of the same astronomical object or background. The time-ordering is only important for conceptualizing the outlier-detection process. It is in the tails of the distribution of the values at a given interpolated-pixel location where the outliers can now be easily detected, and positive outliers are usually identified as cosmic rays.

To detect outliers, Mopex calculates the distribution width, σ , of the temporal samples at each interpolated-pixel location using one of two optional methods. If `THRESHOPTION = 1`, σ is computed by the median absolute deviation (MAD), given by

$$\text{MAD}(I_k) = \text{median}(|I_k - M|)/0.6745, \quad (1)$$

which robustly estimates the scatter around the median, M , where I_k are the sample values at a given interpolated-pixel location, and subscript k is an index that refers to the input image from which the interpolated value was computed. The factor of 0.6745 accounts for the relationship between MAD and σ of a normal distribution. If `THRESHOPTION = 2`, the value used in this work, σ is computed more simply by the minimum difference between consecutive temporally ordered samples.

Value I_k is detected as an outlier if

$$I_k < M - \Gamma_{\text{BOTTOM}} \times \sigma \quad (2)$$

or

$$I_k > M + \Gamma_{\text{TOP}} \times \sigma, \quad (3)$$

where Γ_{BOTTOM} and Γ_{TOP} correspond to Mopex input parameters `BOTTOM_THRESHOLD` and `TOP_THRESHOLD`, respectively. If input uncertainty images are provided for the MOPEX processing, as was done in this work, and the smallest uncertainty associated with an interpolated pixel in the stack is greater than σ , then MOPEX replaces σ with the smallest uncertainty. In addition, if the number of samples is less than `MIN_PIX_NUM`, then σ will be set to smallest uncertainty (if input uncertainty images are given). We used `MINPIXNUM = 3` in this work.

Once an outlier has been identified in an interpolated image, the corresponding input image is known via subscript k , and it is of interest to locate the pixel in the input image that contains the outlier. Because interpolated pixels are, in general, smaller than the input pixel, and also spatially shifted and rotated relative to the input pixel, there can be a group of adjacent interpolated pixels with detected outliers than overlap onto an input pixel containing the underlying outlier. The Mopex multiframe outlier-detection algorithm makes use of this feature by comput-

ing the total area of interpolated pixels with outliers that overlap onto an input pixel. The ratio of this to the area of the input pixel is then computed. If this ratio is greater than the Mopex input parameter `RM_THRESH` (hereafter Γ_{RM}), then the input pixel is declared to contain an outlier. This threshold thus specifies the minimum fraction of an input pixel that must be covered by the projection of outlier interpolated pixels in order to be marked as an outlier.

In this work, the number of negative outliers is insignificant compared to the number of positive outliers, and so we set `BOTTOM_THRESHOLD` to a large number and left it fixed (the exact value is unimportant).

3. INPUT IMAGES

Our basic set of input images is from *Spitzer* IRAC observations of the Pleiades in Astronomical Observing Request 6526464. The entire mapping observation contains thousands of stars and galaxies, and a lot of dust, which is readily apparent as vaporous clouds in the background of the longer wavelength images. For each IRAC channel or infrared passband, there are 324 processed images, which is by design a large enough data set to give negligibly low uncertainties in our computed performance results. We downloaded the processed images plus their masks and uncertainty images from the *Spitzer* archive via the SSC's Leopard software.

We performed a quick study of IRAC-image pixels affected by natural cosmic rays, in order to learn how to simulate them. The images for channels 1 and 2 were found to contain single-pixel radiation hits and also transient streaks of up to a dozen or so pixels running along vertical, horizontal, and diagonal lines. The channel-3 images contain mostly isolated hits, and a handful of long streaks affecting up to 20 pixels with no apparent preferred directionality. The channel-4 images contain roughly equal numbers of isolated hits and few-pixel clumps, as well as a handful of long streaks. A very rough inventory revealed that the images for all channels contain approximately equal numbers of isolated hits and multipixel short streaks, which usually affect about 30 pixels per image total, except for channel 3 where the total number averages slightly higher. See Patten et al. (2004) for a more general discussion on the variation of radiation hits in different IRAC channels.

We processed the images for each channel with the Mopex-GUI software and used $\Gamma_{\text{TOP}} = 3$ and $\Gamma_{\text{RM}} = 0$ in computing masks for the natural cosmic-ray glitches and other positive outliers. These settings maximize the chances of finding all positive outliers, but have the side effect of also masking off a lot of pixels with good data (in fact, using these settings, our results show that for every pixel with a positive outlier rejected, there will be two or three pixels with no positive outliers also rejected).

4. MONTE CARLO SIMULATIONS

We simulated cosmic-ray or radiation-hit flux densities by sampling an exponential distribution in the range of 2–20 MJy sr⁻¹ (MegaJanskys per steradian per pixel). Positive outliers with flux densities in this range are easily detected by eye when comparing successive images of the same part of the celestial sky. Figure 1 shows a histogram of the pixel flux densities of the simulated cosmic rays used in this study. Flux-density correlations in cosmic-ray clumps and streaks in the images were simulated by applying up to $\pm 30\%$ random fluctuations. We chose the proportions of isolated hits, clumps, and streaks to match roughly those found in the input IRAC images, and as appropriate for a given IRAC channel.

The pixel positions of the simulated cosmic-ray data were sampled from a uniform distribution. In our simulator, we specified 30 cosmic rays per image, which resulted in ≈ 90 positive-outlier pixels per image, on average, or 29,177 positive-outlier pixels for all 324 images per given IRAC channel (0.137% of the available pixels). Care was taken to sample unique pixel positions for a given image. We added the simulated cosmic-ray pixels to our sets of input images by simply adding the cosmic-ray pixel values to the input pixel values at the prescribed pixel locations. This resulted in enriched sets of IRAC images with simulated cosmic-ray pixels (for all four IRAC channels), suitable for performance testing. The effect of arbitrarily increasing by 3 times the number of simulated cosmic-ray pixels relative to the number of natural cosmic-ray pixels already present in the input images is to improve the scoring statistics while still keeping the cosmic-ray pixel density very low.

5. SCORING METHOD

It is assumed that the detection algorithm or filter outputs continuous values, where higher values are more strongly related to the presence of the class of targets or sources being

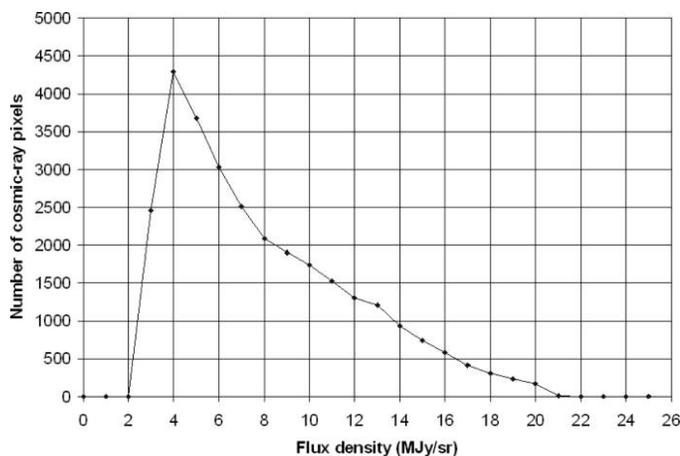


FIG. 1.—Histogram of pixel flux densities of the simulated cosmic rays.

detected than lower values. The filter output is generally thresholded at some level, and filter outputs above threshold are called detections.

In this paper, we are detecting positive outliers in image data as a function of pixel position. The strength of the outlier has a substantial effect on the filter output. Weak positive outliers result in low filter outputs that may fall below the detection threshold and, thus, will remain undetected. This is the crux of the detection problem.

Defining N_T as the number of true detections (i.e., at positions in an image where positive outliers are known to exist), and N_F as the number of false detections (i.e., at positions in an image where positive outliers are known not to exist), the total number of detections can be written simply as

$$N_D = N_T + N_F. \quad (4)$$

We also define N_S as the total number of sources (positive outliers) in a given set of test images, which may or may not escape detection, depending on the detection algorithm and the threshold setting for detection.

Because we simulated cosmic-ray flux densities and positions, the ground truth for the pixel positions of the positive outliers is well established and so we can easily classify whether a detection is true or false.

We define completeness as

$$C \equiv \frac{N_T}{N_S} \quad (5)$$

and reliability as one minus the unreliability, or

$$R \equiv 1 - \frac{N_F}{N_D} = 1 - \frac{N_F}{N_T + N_F} = \frac{N_T}{N_T + N_F} = \frac{N_T}{N_D}, \quad (6)$$

which is simply the ratio of the number of true detections to the total number of detections.

6. SOFTWARE IMPLEMENTATION

We modified Mopex's *mosaic_rmask* module to do the following: (1) read in and store the ground-truth positions of our simulated cosmic-ray pixels; and (2) accumulate statistics for computing the completeness and reliability via equations (4) through (6).

We wanted our performance scores to be free of any influences from natural radiation hits, other positive outliers, and questionable data in pixels that have been masked off from prior processing. We therefore additively combined the masks we obtained for the natural cosmic rays and other positive outliers as described above with the *imask* images that are downloaded with the image data from the *Spitzer* archive, which gave us a new set of combined masks for each IRAC channel. We specified the combined masks as input masks when executing the

mosaic_rmask module, to be used in conjunction with the special logic that we also added to the *mosaic_rmask* module to omit from the scoring any pixel with an associated nonzero combined mask value.

Rather than brute-force execution of the *mosaic_rmask* module with different input settings of Γ_{RM} to compute completeness and reliability as a function of this threshold, we instead implemented the following method in the source code of the *mosaic_rmask* module. We binned the zero-to-one range of possible Γ_{RM} values in steps of 0.01 for a total of 101 bins (indexed from 0 to 100). For each pixel with detection filter output f , which is in the same data units as Γ_{RM} , we computed the bin number j associated with f :

$$j = \text{int}(100 \times f). \tag{7}$$

We then accumulated the following statistics for performance scoring:

$$N_S = N_S + 1, \quad N_{Ti} = N_{Ti} + 1 \forall i < j, \tag{8}$$

or

$$N_{Fi} = N_{Fi} + 1 \forall i < j, \tag{9}$$

depending on whether f is a true or false detection (which is known from ground truth). The completeness and reliability associated with bin i is

$$C_i \equiv \frac{N_{Ti}}{N_S} \tag{10}$$

and

$$R_i \equiv \frac{N_{Ti}}{N_{Ti} + N_{Fi}}. \tag{11}$$

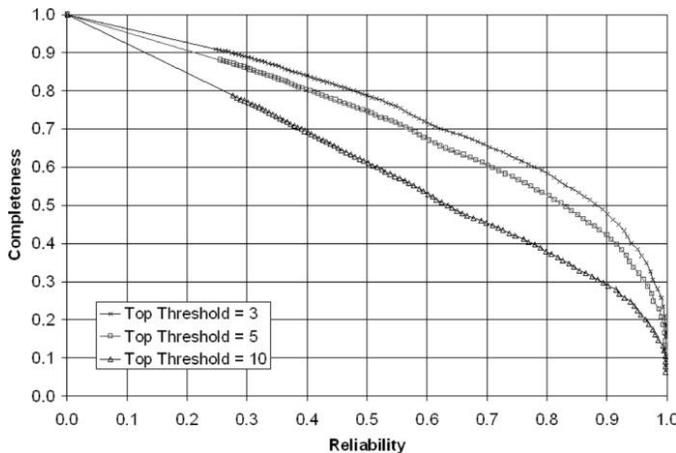


FIG. 2.—Completeness vs. reliability of Mopex’s multiframe outlier-detection algorithm for IRAC channel 1. See the electronic edition of the *PASP* for a color version of this figure.

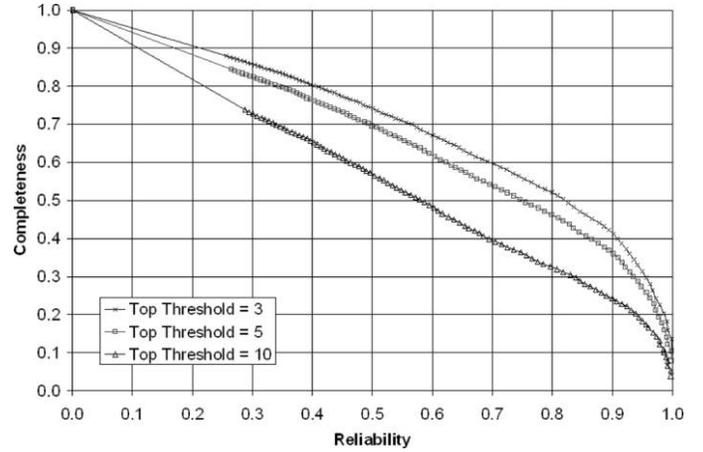


FIG. 3.—Completeness vs. reliability of Mopex’s multiframe outlier-detection algorithm for IRAC channel 2. See the electronic edition of the *PASP* for a color version of this figure.

This simple but powerful method allowed us to generate entire curves for completeness and reliability as a function of Γ_{RM} for a given Γ_{TOP} threshold with a single execution of our specially modified *mosaic_rmask* module.

7. RESULTS

Figures 2 through 5 present completeness versus reliability of Mopex’s multiframe outlier-detection algorithm for all four IRAC channels. The curves in these figures show the completeness that corresponds to a given reliability for different settings of Γ_{TOP} . The figures include curves for Γ_{TOP} values of 3, 5, and 10.

Figures 6 through 8 show the same curves as in Figures 2 through 5, but grouped differently. Each of these figures in-

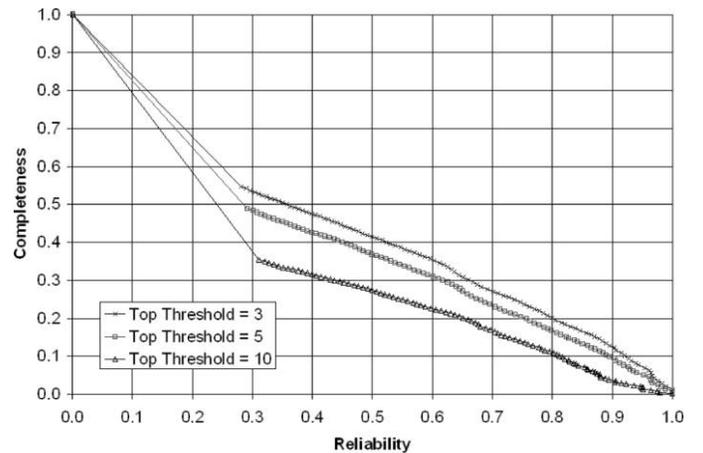


FIG. 4.—Completeness vs. reliability of Mopex’s multiframe outlier-detection algorithm for IRAC channel 3. See the electronic edition of the *PASP* for a color version of this figure.

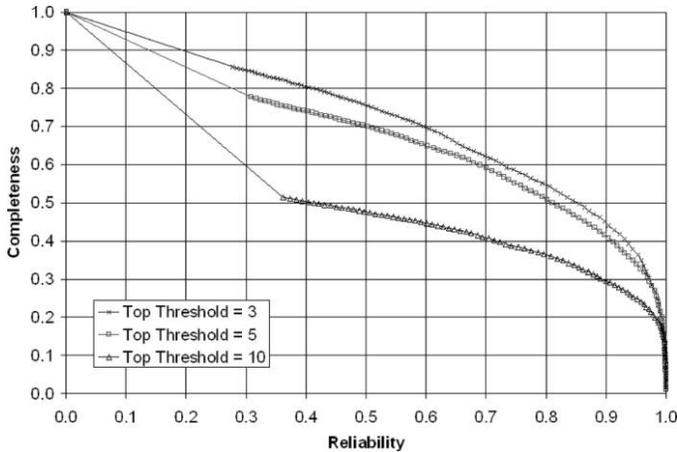


FIG. 5.—Completeness vs. reliability of Mopex's multiframe outlier-detection algorithm for IRAC channel 4. See the electronic edition of the *PASP* for a color version of this figure.

cludes curves for all IRAC channels at the same Γ_{TOP} value. The figures demonstrate how the performance varies from channel to channel for a given value of Γ_{TOP} .

The above figures define the one-to-one relationship between completeness and reliability of Mopex's multiframe outlier-detection algorithm, but do not reveal what setting of Γ_{RM} is needed to achieve an available pair of C and R performance figures for a given Γ_{TOP} and IRAC channel. In choosing a desirable value for Γ_{RM} , Figures 9 through 12 should be consulted. Each of these figures show, for given Γ_{RM} and Γ_{TOP} , the level of performance that can be expected in terms of both completeness and reliability.

This paper would be incomplete without a determination of the uncertainties associated with our completeness and reliabil-

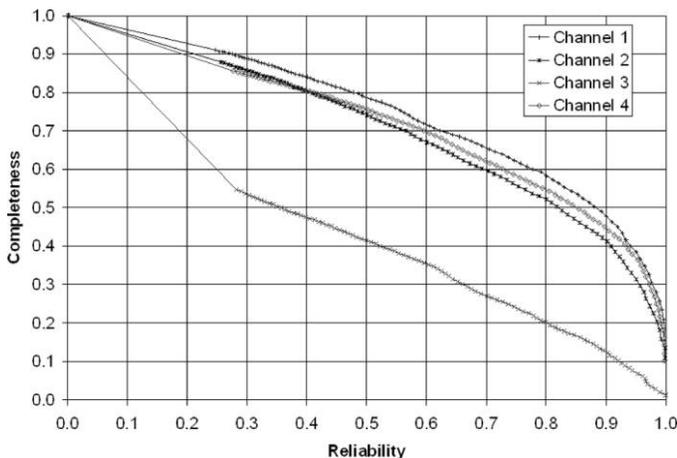


FIG. 6.—Completeness vs. reliability of Mopex's multiframe outlier-detection algorithm for $\Gamma_{\text{TOP}} = 3$. See the electronic edition of the *PASP* for a color version of this figure.

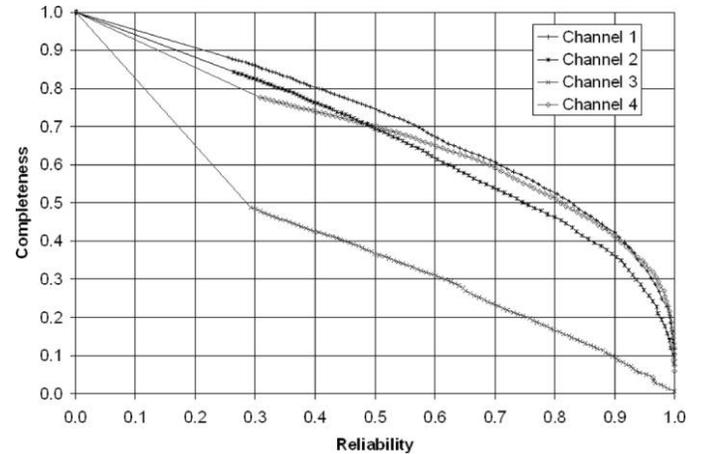


FIG. 7.—Completeness vs. reliability of Mopex's multiframe outlier-detection algorithm for $\Gamma_{\text{TOP}} = 5$. See the electronic edition of the *PASP* for a color version of this figure.

ity estimates. We therefore generated five independent sets of channel-1 IRAC images with simulated cosmic-ray pixels from different random-number-generator seeds, and then computed completeness and reliability versus Γ_{RM} for $\Gamma_{\text{TOP}} = 3$. We found the variation of completeness and reliability to be much less than 1% over the entire valid range of Γ_{RM} , which is from 0 to 1, inclusive. The low uncertainties associated with our performance results are expected because of our large set of input images for each case studied, which we deliberately selected to greatly narrow our margin of error.

Finally, we computed completeness and reliability curves for Γ_{TOP} values of 2 and 2.5 (figures not shown), for all IRAC channels, in order to confirm our present understanding of Mopex's multiframe outlier-detection algorithm. For these extreme cases, we expected the completeness to remain the same as for

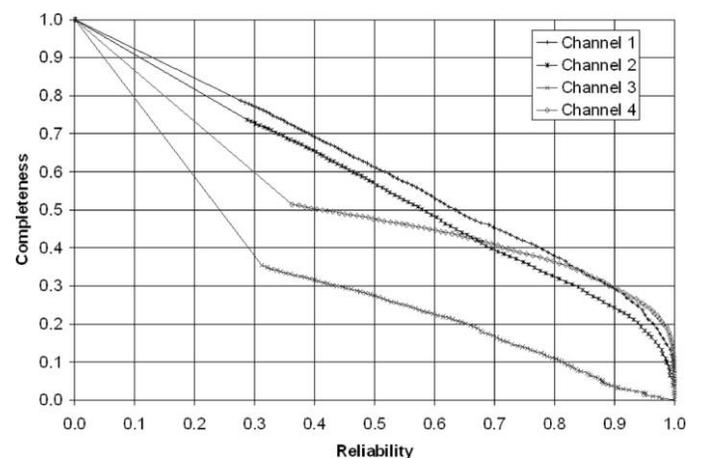


FIG. 8.—Completeness vs. reliability of Mopex's multiframe outlier-detection algorithm for $\Gamma_{\text{TOP}} = 10$. See the electronic edition of the *PASP* for a color version of this figure.

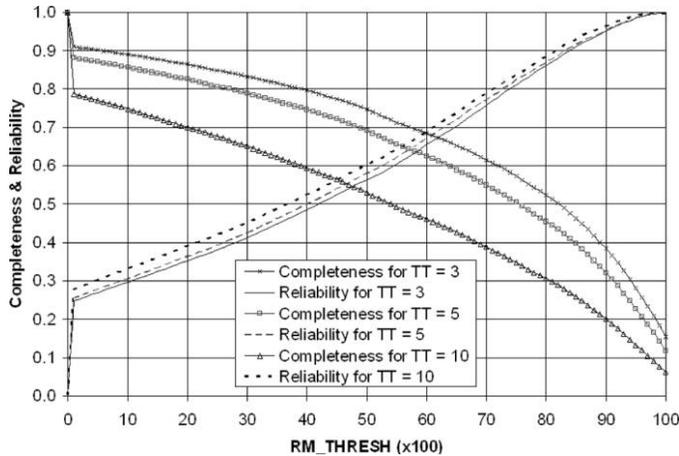


FIG. 9.—Completeness and reliability vs. Γ_{RM} of Mopex’s multiframe outlier-detection algorithm for IRAC channel 1. In the figure legend, “TT” stands for Γ_{TOP} . See the electronic edition of the *PASP* for a color version of this figure.

$\Gamma_{TOP} = 3$ and the reliability to drop precipitously. Indeed, this was the behavior observed.

8. DISCUSSION AND CONCLUSIONS

For most applications, both high completeness and reliability are desired. Our results show that the combinations of highest completeness and reliability values from Mopex’s multiframe outlier-detection algorithm are obtained by setting $\Gamma_{TOP} = 3$ for all IRAC channels. The other primary adjustable parameter of this algorithm, Γ_{RM} , can be set to achieve either the desired completeness or reliability, but not both, because completeness and reliability are interdependent. Generally, setting Γ_{RM} low will maximize completeness at the expense of reliability,

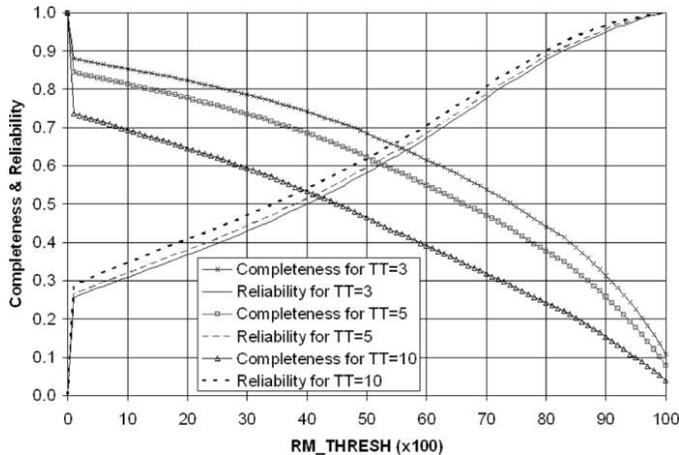


FIG. 10.—Completeness and reliability vs. Γ_{RM} of Mopex’s multiframe outlier-detection algorithm for IRAC channel 2. In the figure legend, “TT” stands for Γ_{TOP} . See the electronic edition of the *PASP* for a color version of this figure.

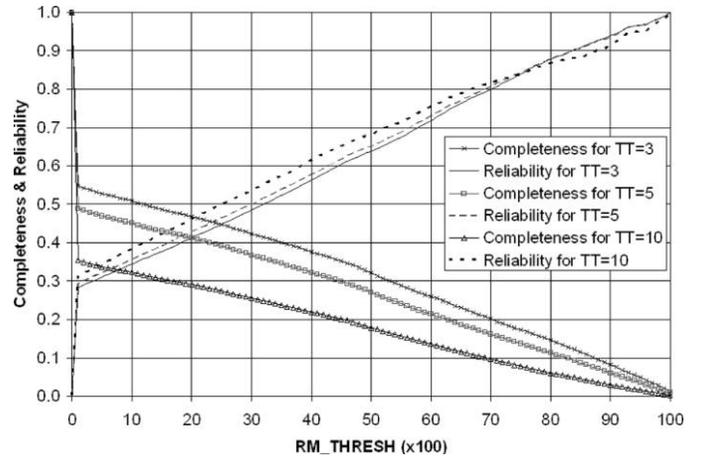


FIG. 11.—Completeness and reliability vs. Γ_{RM} of Mopex’s multiframe outlier-detection algorithm for IRAC channel 3. In the figure legend, “TT” stands for Γ_{TOP} . See the electronic edition of the *PASP* for a color version of this figure.

and vice versa for setting Γ_{RM} high. Figures 9 through 12 in this paper give the expected performance for similar IRAC data sets.

Values of Γ_{TOP} less than 3 should be avoided if high reliability is the goal. A plausible reason for very low-reliability yields for $\Gamma_{TOP} < 3$ is that thresholds decreasing below this level lead to rapidly increasing the area under the curve in the chopped-off positive tail of the data distribution. In other words, low-reliability, high-completeness performance is tantamount to throwing away good data in order to remove as many outliers as possible. But even in this regime, which is achievable by setting Γ_{RM} to a low value (such as 0.05, for example), the maximum achievable completeness we found was only around 90% for $\Gamma_{TOP} = 3$, and this is achievable for IRAC channels 1, 2, and 4, but not 3.

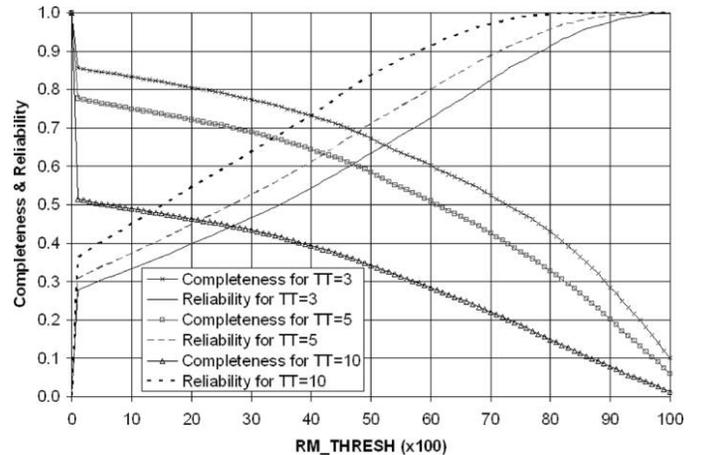


FIG. 12.—Completeness and reliability vs. Γ_{RM} of Mopex’s multiframe outlier-detection algorithm for IRAC channel 4. In the figure legend, “TT” stands for Γ_{TOP} . See the electronic edition of the *PASP* for a color version of this figure.

The poorest C and R performance was for IRAC channel 3, which is not surprising because the image data for this channel are well known to contain significantly higher noise than the other IRAC channels. The maximum achievable completeness for IRAC channel 3 is about 55%, and this decreases approximately linearly with higher Γ_{RM} settings.

Our performance results are strongly dependent on the flux densities we employed in computing the simulated cosmic-ray data. If we had used higher cosmic-ray flux densities, then higher values for both completeness and reliability would have been obtained, and vice versa.

We also note that our performance results apply to the coverage or depth of our particular set of input images, which happens to be low. Examination of the coverage image associated with the mosaic for IRAC channel 1, which was downloaded from the *Spitzer* archive, shows the coverage is generally only 2 samples per mosaic pixel. We also wanted to perform similar completeness and reliability studies for test image-data sets with moderate and high coverages but, unfortunately, we ran out of

time. More work could be done to characterize Mopex's multi-frame outlier-detection algorithm for image-data sets with a wider variety of coverage, and this is being planned.

Other future work at the SSC may involve the implementation of improvements to Mopex's multiframe outlier-detection algorithm, in which case scoring the performance of the new method to prove algorithmic superiority can be done using the techniques explained in this paper, with direct comparisons possible for the same test image-data sets.

In conclusion, aside from the useful, practical results presented in this paper (Figs. 9 through 12 in particular), the scoring methodology described herein can more generally be considered as a prototype for similar work in quantitatively evaluating a wider variety of test image-data sets, the performance of Mopex's other currently implemented outlier-detection methods, and contemplated improvements to those methods.

We are grateful to John Fowler for providing valuable advice on how to simulate cosmic-ray data.

REFERENCES

Patten, B. M., et al. 2004, Proc. SPIE, 5487, 223