

# **PCID and ASPIRE 2.0 – the next generation AMOS image processing environment**

**Charles L. Matson, Charles C. Beckner, Jr.**

*Directed Energy Directorate, Air Force Research Laboratory, Kirtland AFB, NM*

**Kathy Borelli**

*KJS Consulting, Haiku, HI*

**Tom Soo Hoo, Shiho You, Brandoch Calef**

*Boeing LTS, Kihei, HI*

**Maria Murphy, Ron Vioria**

*Maui High-Performance Computing Center, Kihei, HI*

## **ABSTRACT**

We describe the next generation AMOS image processing environment that is in the process of being deployed for site-wide use. It consists of the bispectrum and PCID algorithms used to remove atmospheric blurring from images of space objects collected using AMOS telescopes and a GUI front end, called ASPIRE 2.0, to these algorithms and to the site satellite database called AMOSphere. We give an overview of the PCID algorithm and its properties. We also provide screen shots of several ASPIRE 2.0 web pages and describe how they are used to access raw data, process the data, and manage the restored images.

## **1. INTRODUCTION**

One of the missions of the Air Force Maui Optical and Supercomputing (AMOS) site [1] is to generate high-resolution images of space objects using the Air Force telescopes located on Haleakala and the computers at the Maui High-Performance Computing Center (MHPCC) [2]. Because atmospheric turbulence greatly reduces the resolution of space object images collected with ground-based telescopes, methods for overcoming atmospheric blurring are necessary. One such method is the use of adaptive optics systems to measure and compensate for atmospheric blurring in real time. A second method is to use image restoration (reconstruction) algorithms on one or more short-exposure images of the space object under consideration. At AMOS, both methods are used routinely. In the case of adaptive optics, rarely can all atmospheric turbulence effects be removed from the imagery, so image restoration algorithms are useful even for adaptive-optics-corrected images. Historically, the bispectrum algorithm has been the primary image restoration algorithm used at AMOS [3]. It has the advantages of being extremely fast (processing times of less than one second) and insensitive to atmospheric phase distortions. In addition to the bispectrum algorithm, multi-frame blind deconvolution (MFBD) algorithms have also been used for image restoration. It has been observed empirically and with the use of computer simulation studies that MFBD algorithms produce higher-resolution image restorations than does the bispectrum algorithm. MFBD algorithms also do not need separate measurements of a star in order to work. However, in the past, MFBD algorithms have taken minutes to hours to generate image restorations, limiting their use to non-time-critical image restorations.

Recently, an MFBD algorithm called Physically-Constrained Iterative Deconvolution (PCID) [4] has been efficiently parallelized and is able to produce image restorations in as little as a few seconds. In addition, it has been shown that PCID achieves or closely approaches the theoretical limits to image restoration quality for a variety of scenarios. For these reasons, PCID is now being transitioned to be the default site-wide image restoration algorithm. Because the algorithm can be complicated to use, a GUI is being developed to be the front end to the PCID algorithm. This interface, called the Advanced SPeckle Image Reconstruction Environment (ASPIRE) version 2.0, is the next generation of the current ASPIRE GUI used as a front end to the bispectrum algorithm. ASPIRE 2.0 will be the front-end GUI to PCID, the bispectrum algorithm, and the AMOS site data repository called AMOSphere. In this paper we describe ASPIRE 2.0 and PCID and how to use them to obtain high-resolution images.

## 2. PCID

The PCID algorithm is a multi-frame blind deconvolution algorithm that jointly estimates the pristine image and the point-spread functions (PSFs) that generated the sequence of blurred images used as input to the algorithm. The estimates of the pristine image and the PSFs are obtained by minimizing the weighted sum (over image pixels and measurement frames) of the measurement/model residuals. The measurement/model residuals are defined as the differences between the actual measurements and the current models of the measurements, where the current models of the measurements are generated from the current estimates of the pristine image and PSFs by convolving each PSF with the pristine image. The residuals are weighted by the noise covariance matrix, which is a diagonal matrix since the noises corrupting the measurements are spatially and temporally statistically independent. We minimize the cost function using a parallelized version of a conjugate gradient routine.

The pristine image is always estimated pixel-by-pixel in the image domain. The PSFs can also be estimated pixel-by-pixel in the image domain; however, in most cases, higher-quality restored images result from estimating the pupil phase and using a Fourier-optics model to generate the PSFs from the pupil phase [5]. Even when the raw data is collected using spectral filters whose widths are hundreds of nanometers, the pupil-phase estimation approach produces excellent results. We parameter the pupil phase in terms of a Zernike polynomial expansion and estimate the coefficients of the polynomials.

To minimize noise amplification, the PCID algorithm must be regularized. Two approaches are available for regularization. The first approach is to include a regularizing term in the cost function that, when minimized, minimizes noise amplification. One such term can be generated using a Tikhonov regularization approach, but many other options also exist. A scaling parameter multiplies the regularization term and is chosen to optimally trade off the resolution and the noise levels in the deblurred image. A second approach to regularization is to use Fourier-filter-based regularization in the data-matching term. This approach causes the higher spatial frequencies in the restored image to be reconstructed more slowly than the lower spatial frequencies. By terminating the minimization process prior to convergence, excessive amplification of the higher spatial frequencies is avoided.

The PCID algorithm has been parallelized to run efficiently on commodity clusters (i.e., clusters consisting of computers that do not have specialized hardware in them, connected by standard networks such as Ethernet, Infiniband, or Myrinet). We have demonstrated excellent scalability for tens to hundreds of nodes, depending on the speed of the interconnect and the sizes of the raw image arrays. Image reconstruction times range from a few seconds to hours, depending on the number of frames of data used, the number of conjugate gradient iterations, and the sizes of the data arrays.

We have carried out an algorithm-independent analysis to evaluate how close to the optimal solutions the PCID algorithm comes for a variety of imaging scenarios. For this analysis, we used Cramér-Rao lower bound (CRB) theory [6] because fundamental limits to both restoration accuracy in image pixel intensities and achievable spatial resolution can be generated. For the comparison, we calculated the CRBs for twenty separate imaging scenarios. We next used PCID to generate fifty restorations for each imaging scenario. We then calculated sample variances from the PCID restorations. Finally, we divided the sample variances by the CRBs for each of these twenty scenarios. When PCID achieves the CRBs, the ratio is one; otherwise, the ratio will be greater than one. For these twenty imaging scenarios, the average value of the ratio was 1.02, demonstrating that the quality of the PCID image restorations are within a few percent of the theoretical limits.

PCID and ASPIRE 2.0 are hosted on two AMD-Opteron-based Cray XD1 [7] clusters at the MHPCC. These two clusters started out as a single Cray XD1 cluster, but were obtained by splitting the single cluster into two parts. One of these two clusters, called Polaris, is used for routine AMOS data reduction while the second, called Hoku, is used for research and development supporting the software on Polaris as well as for other R&D tasks.

## 3. ASPIRE 2.0

Effective image processing at AMOS involves more than just employing image restoration algorithms that produce high-quality restorations. It is also necessary to be able to access the raw data to be processed and the parameters that describe how the raw data was collected, to calibrate the raw data, to submit multiple image processing jobs to the host computer, to visualize and store the restored images, and to accomplish all of these tasks in an intuitive and

easy-to-use manner. The ASPIRE 2.0 environment is being created to achieve all of these goals. Prior to ASPIRE 2.0, a image processing environment called Maui Image Manager and On-line System Archive (MIMOSA) was used for these functions, but was primarily intended to be an interface to an older version of the bispectrum algorithm and to manage the job queue on the computer used to process the images. MIMOSA was then upgraded to include beta-level support for the PCID algorithm and was renamed ASPIRE 1.0. The ASPIRE 2.0 environment greatly extends the ASPIRE 1.0 capabilities by including operational support to PCID, increased automatic parameter selection, improved queue management capabilities, and additional image visualization and analysis tools, among other improvements.

The ASPIRE 2.0 environment is a web-based environment that consists of a number of technologies and languages including ColdFusion, CFML, XHTML, CSS, Perl, Java, JavaScript, MySQL, and Apache. The user accesses ASPIRE 2.0 with a web browser on her client machine. Innovative techniques have been implemented to enable the ASPIRE 2.0 user to focus on tasks at a higher level without the burden of dealing with the infrastructure at a lower level. The AMOSphere and ASPIRE web and database servers reside on a node of the Polaris cluster.

A key feature of ASPIRE 2.0 is that it will be seamlessly integrated with the AMOSphere, the AMOS site data repository and the location from which the vast majority of data that is generated and processed by the site is disseminated to customers. One example of this integration is that the users who log into ASPIRE 2.0 will automatically be given login privileges to AMOSphere, and vice versa. Another example is that, when a user logged into ASPIRE 2.0 needs data from AMOSphere, there will be links on the ASPIRE 2.0 pages to AMOSphere. Also, when a user logged into AMOSphere desires to process the data she is viewing, there will be a link to the ASPIRE 2.0 image processing page.

The ASPIRE home page is shown in Fig. 1. There are two sets of links on this page, one set that runs vertically down the left hand side of the page, and the other set that runs horizontally across the top of the page. The vertical set of links is intended to be task-based; that is, these links lead to pages based upon a task that the user desires to accomplish. The horizontal set of links, on the other hand, is intended to be tool-based. When a user desires to access a specific tool (such as the image viewer or AMOSphere), she would use these links.

In this section, we describe both tasks listed in the left-hand column, and the Image Viewer tool. The first task is to generate new image recoveries. To start this task, the user clicks on the "Generate New Recoveries" link in the left-hand column. The page that appears is shown in Fig. 2. As can be seen, the user is guided through the seven steps that must be completed to successfully generate some image recoveries. As many of the inputs as possible are filled in automatically for the user, based upon the data itself. In Step 1, the user must select the dataset to process to begin (if the user arrives at this page from AMOSphere, the data will have already been selected in AMOSphere and the Raw File Location field will already be filled in). The user can type the name and path for the data directly into the field, or use the icons to the right of the field to search for the data in either AMOSphere or in the computer file system. Once the data is selected, the user chooses what frames inside the data file to include in the recovery process. Tools that allow the user to browse through the raw images in the data file and view the observation parameters are accessed via the Select Frames icons.

In Step 2, the user selects the desired algorithm, either PCID or bispectrum. In Step 3, the calibration data files for the raw image data are chosen. These fields are already filled in, based on the data file selected in Step 1, but the user can override these choices if she desires. In Step 4, a prefix is chosen by the user that will be prepended to all of the output files to identify them and the submitted jobs in the queue.

All the algorithm parameters are entered in Step 5. Only those parameters relevant to the algorithm chosen are presented to the user. The user must choose the ensemble size and the number of recoveries or the stride between recoveries. The regularization filter defaults to a Gaussian and the width of the Gaussian is determined from the data if not specified by the user. The user then selects a desired image quality based upon the amount of time required to carry out the processing. Medium quality is the default value. Each quality choice corresponds to specific settings for the number of conjugate gradient iterations, the number of data frames used for each restoration, and whether or not to incorporate de-aliasing (when appropriate). The last step (Step 7) the user takes is to submit the job to the batch queue on the processing computer.

The second task to be discussed is the 'Manage Completed Recoveries' task. A screen shot is shown in Fig. 3. At the top of the main display window is a list of jobs that are either running or have finished running but have not yet had their results analyzed and either stored or discarded. Any running job can be canceled by the owner of the job. Any job still in the list, running or completed, can be resubmitted if desired. The purpose of this option is to be able to rerun a job with a different set of parameters to try to get a better recovery. By clicking on the 'Resubmit Job' link for a given job, all the parameters of that job are used to populate the 'Generate New Recoveries' page to expedite the resubmission process. Finally, for all the jobs that have been completed, the recoveries can be viewed with a movie viewer, and/or can be stored into AMOSphere, and/or can be deleted without storing into AMOSphere.

The middle region of this screen shot contains the details of the particular job selected in the list above. These details will be of interest only to advanced users. Finally, the bottom region of this window shows the job queue and all the jobs in the queue. This information is particularly useful to help users, who want to submit one or more jobs, to get an idea of who is ahead of them in the queue and what kind of turn around time might be expected. If the user has a high-priority job but has a stacked queue in front of her, she can be assigned a priority that will jump her to the head of the queue.

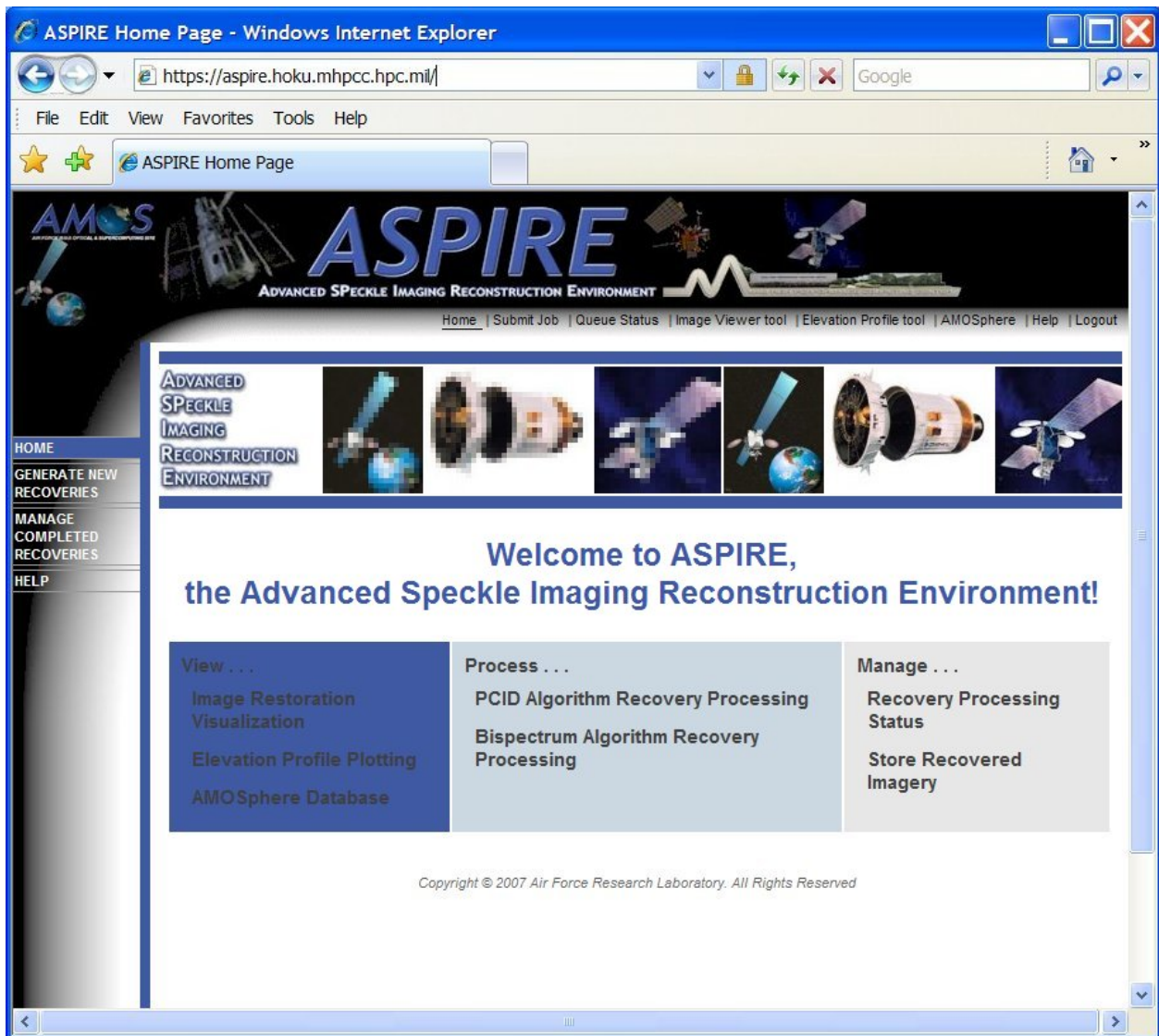


Fig. 1. The ASPIRE home page

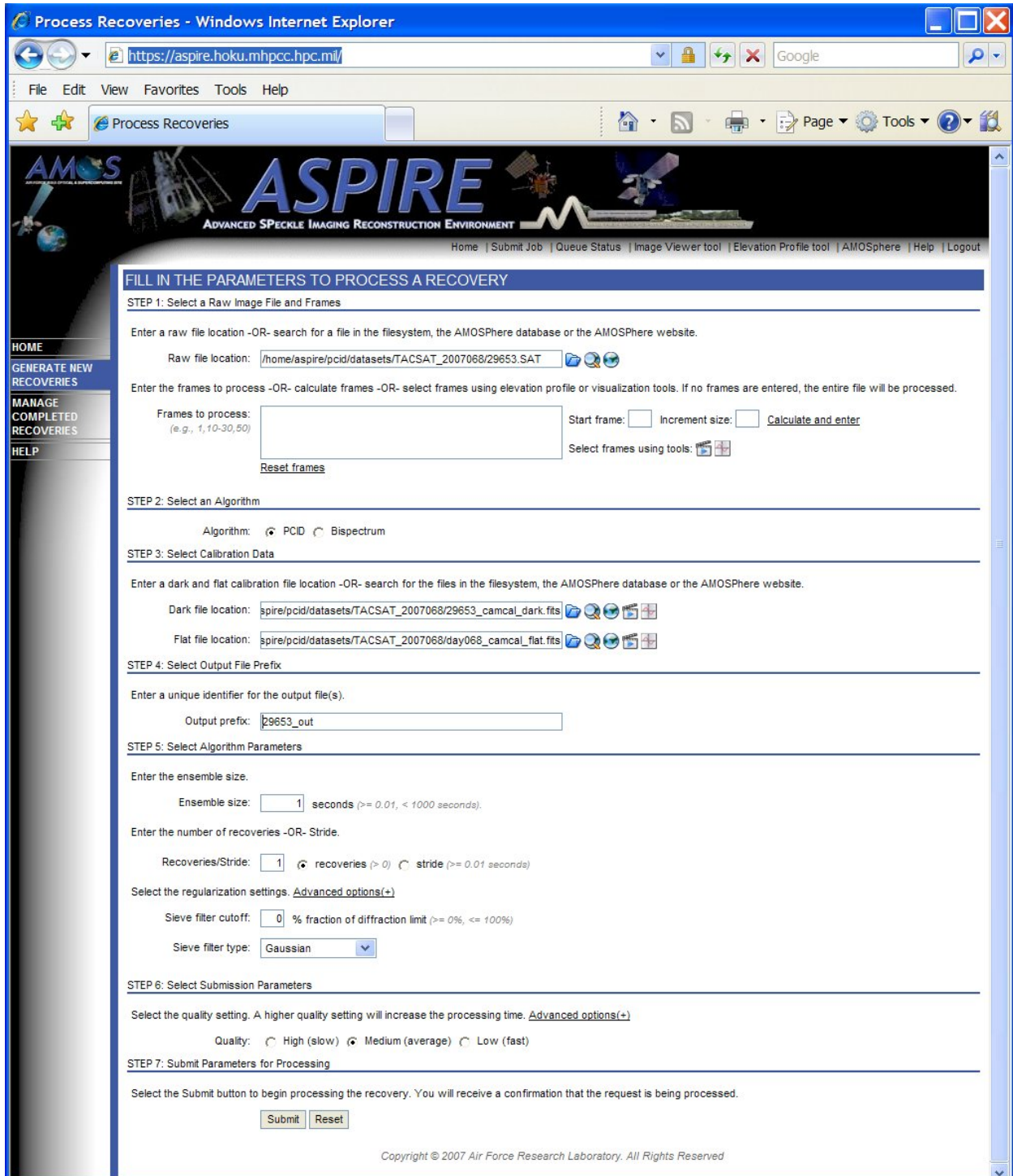


Fig. 2. The ASPIRE “Generate New Recoveries” page

Login to Website - Windows Internet Explorer  
 https://aspire.hoku.mhpc.hpc.mil/

File Edit View Favorites Tools Help

Manage Processed Recoveries

# ASPIRE

ADVANCED SPECKLE IMAGING RECONSTRUCTION ENVIRONMENT

Home | Submit Job | Queue Status | Image Viewer tool | Elevation Profile tool | AMOSphere | Help | Logout

## MANAGE RECOVERIES AND VIEW STATUS

[Hide job details](#) | [Hide queue](#)

USER	DATE	JOB ID	JOB STATUS	CANCEL JOB	VIEW RESULTS	STORE IN AMOSPHERE	RESUBMIT JOB	DELETE FROM DATABASE
bcaief	Mon Jul 2 11:13:13	135453	completed					
bcaief	Mon Jul 2 11:30:53	135456	completed					
bcaief	Mon Jul 2 13:45:12	135461	completed					
matsoncl	Tue Jul 3 07:18:33	135837	running					

### JOB DETAILS - JOB ID 135456

```

=== BEGIN EPILOGUE ===
Mon Jul 2 11:13:13 HST 2007
JID: 135456.ho618n6
EUID: bcaief
EGID: imaging
Job Name: PCID
SID: 7632
Requested limits: ncpus=18,walltime=06:30:00
Resources used: cpupercent=101,cput=00:02:33,mem=83792kb,ncpus=18,vmem=7461788kb,walltime=00:02:28
Queue: compute
Accounting string: DODOT-0000-D00
---- end of usage ----

Clear any jobs on nodes

Doing node ho614n2
Doing node ho616n1
Doing node ho616n4
Doing node ho618n1
=== END EPILOGUE ===
#!/bin/sh
#PBS -A DODOT-0000-D00
#PBS -l walltime=06:30:00
#PBS -l nodes=26:ppn=2#excl
#PBS -j oe
#PBS -q compute
#PBS -N PCID
cd /mnt/lustre/home/aspire/pcid/datasets/TACSAT_2007068
mpirun -kill -order-consecutive $XDLAUNCHER '/mnt/lustre/home/bcaief/src/pcid_v5/pcid mfb/mfb_nameлист > mfb/mfb.out'

```

### JOB QUEUE

active jobs-----

JOBID	USERNAME	STATE	PROC	REMAINING	STARTTIME
135784	rnelson	Running	50	22:07:03	Tue Jul 3 08:53:07
135887	vecknerq	Running	12	1:57:16	Tue Jul 3 11:33:20
135837	matsoncl	Running	60	-00:37:31	Tue Jul 3 07:18:33

3 active jobs      122 of 128 processors in use by local jobs (95.31%)  
 61 of 64 nodes active (95.31%)

eligible jobs-----

JOBID	USERNAME	STATE	PROC	WCLIMIT	QUEUETIME
135879	vecknerq	Idle	12	2:20:00	Tue Jul 3 06:06:58
135880	vecknerq	Idle	12	2:20:00	Tue Jul 3 06:06:58
135998	matsoncl	Idle	60	6:00:00	Tue Jul 3 11:14:43

3 eligible jobs

blocked jobs-----

JOBID	USERNAME	STATE	PROC	WCLIMIT	QUEUETIME
135843	vecknerc	Deferred	12	2:20:00	Tue Jul 3 06:06:45
135997	alskew	Deferred	1	2:00:00	Tue Jul 3 10:03:01

2 blocked jobs

Total jobs: 8

Copyright © 2007 Air Force Research Laboratory. All Rights Reserved

Fig. 3. The ASPIRE “Manage Completed Recoveries page

Next, the 'Image Viewer' tool will be discussed. A screen shot of the 'Image Viewer' page is shown in Fig. 4. As currently laid out, there are three regions of the page corresponding to three different sets of information. The right-hand side of the page is the 'File Data' column that contains most of the information available about the observation parameters for the dataset under consideration. For example, the object number is given so that the object can be identified. All the elevation, azimuth, and range information is given as well. The lower left-hand region of the page contains the 'Image Visualization' tool that permits the user to scroll through the images in the dataset. The viewer helps the user determine what portions of the pass have the highest quality. Finally, the upper left-hand region contains the 'Elevation Profile' tool that shows the elevation corresponding to each image, normalized to the maximum elevation angle, along with other information to be discussed below.

The 'Image Viewer' tool can be used to assist in the data reduction process as will now be described. First, the user can review the data in the 'File Data' column to get an initial idea of the quality of the raw data in order to determine whether or not it is worth processing the data to generate recoveries. If the dataset looks promising from these numbers, the user can then use the 'Image Visualization' tool to play a movie of all the measurement frames to visually assess the raw data. Using the movie control buttons below the viewer, the user can step through the raw images and identify specific data frames or sets of data frames that should produce the best recoveries. This information can be entered into the 'Generate Recoveries' page. Finally, the 'Elevation Profile' display can be used to determine what frames of data can be used for a single recovery. It is necessary that all the frames of data used for a single recovery to have the same exposure time and the same field of view (FOV). When the raw data was collected using adaptive optics data, an additional constraint is that the closed-loop bandwidth of the adaptive optics be the same for all the frames of data used for a single recovery.

For the specific dataset used for the screen shot in Fig. 4, no adaptive optics compensation was used. The vertical lines in the elevation profile correspond to the currently-selected ensembles to be used for image recovery. All the data between two vertical lines will be used for one recovery. The dotted line shows the exposure times for each raw image. It can be seen that three of the ensembles contains frames of data with varying exposure times. As a result, the user will need to return to the 'Generate Recoveries' screen and modify the ensemble selections to ensure a constant exposure for each ensemble. Generally speaking, the 'Generate Recoveries' automatic parameter selection properties will provide default ensembles that will automatically include only compatible data frames in each ensemble. The 'Elevation Profile' tool is especially helpful when the user overrides the default ensembles to generate her own ensembles so that errors like shown in Fig. 4 can be identified and corrected. Finally, the dashed line running across the top of the display shows the FOV for each data frame. For this dataset, the FOV was kept constant.

The 'Image Viewer' tool can also be used to analyze the quality of the image recoveries generated by PCID or the bispectrum algorithm. The user can click on the 'View Results' icon for the desired set of recoveries and view them with this tool. Based on the quality of the images seen using the 'Image Visualization' movie viewer, the analyst can decide whether or not to store the recoveries in AMOSphere.

#### 4. CONCLUSIONS AND FUTURE WORK

A new generation of image processing software, consisting of the PCID algorithm, the bispectrum algorithm, and the ASPIRE 2.0 GUI, is being developed and transitioned for routine processing of AMOS data. The PCID image processing algorithm is a multi-frame blind deconvolution algorithm that closely approaches the theoretical limits to image quality and has been parallelized to generate image recoveries in seconds to minutes. Although it outperforms the current operational bispectrum algorithm, the bispectrum algorithm will be retained because of its fast execution times (less than a second) and because the quality of its image recoveries is still quite good. The ASPIRE 2.0 GUI will provide an easy-to-use interface to the PCID algorithm as well as the current operational bispectrum algorithm. ASPIRE 2.0 will also be seamlessly integrated with AMOSphere to provide a site-wide image processing and data management capability. It will provide suggested values for parameters needed to run PCID or bispectrum that can be overridden, if desired. In addition, it will provide easy access to AMOSphere or the processing computer's file system to select raw data files and to store image recoveries.

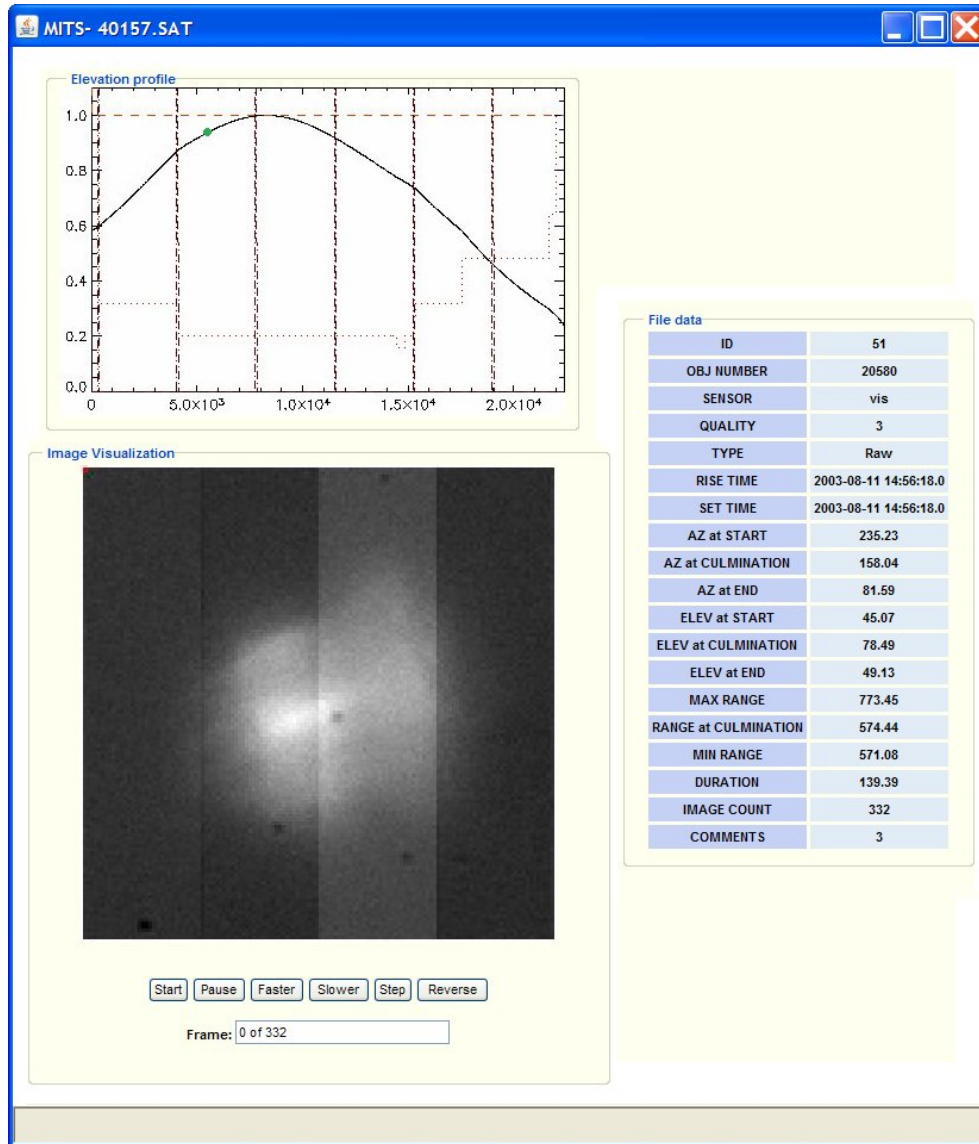


Fig. 4. The ASPIRE 'Image Viewer' page

## 5. ACKNOWLEDGEMENTS

The authors wish to thank the Air Force Research Laboratory's Directed Energy Directorate, the Air Force Office of Scientific Research, and the High-Performance Computing Modernization Office for their financial support that made work possible.

## 6. REFERENCES

<sup>1</sup> <http://www.maui.afmc.af.mil/>

<sup>2</sup> <http://www.mhpcc.hpc.mil/>

<sup>3</sup> K. Schulze and C. L. Matson, "A New Parallel Bispectrum Image Recovery Algorithm for Large Format Sensors," Proceedings of the 2001 AMOS Technical Conference.

<sup>4</sup> C. L. Matson, C. C. Beckner, Jr., K. Borelli, S. M. Jefferies, E. K. Hege, and M. Lloyd-Hart, "A fast and optimal multi-frame blind deconvolution algorithm for high-resolution ground-based imaging of space objects," in preparation.



<sup>5</sup> C. L. Matson and A. Haji, "An algorithm-independent analysis of the quality of images produced using multi-frame blind deconvolution algorithms," Proceedings of the 2007 AMOS Technical Conference.

<sup>6</sup> B. Porat, *Digital Processing of Random Signals -Theory and Methods*, (Prentice-Hall, Englewood Cliffs, 1994), Ch. 3.

<sup>7</sup> <http://www.cray.com/products/xd1/index.html>