# Guest Observer Handbook for EXES Data Products

***Release : EXES Data Handbook Rev. D***

**M. Clarke, W. Vacca**

**Mar 22, 2022**

## Contents

**I Introduction** **3**

**II SI Observing Modes Supported** **3**

**1 EXES observing techniques** **3**

**2 EXES instrument configurations** **3**

**III Algorithm Description** **4**

**3 Overview of data reduction steps** **4**

**4 Reduction algorithms** **4**
4.1 Correct nonlinearity and coadd readout frames . . . 5
4.2 Process flat field . . . 7
4.3 Despike . . . 8
4.4 Debounce . . . 9
4.5 Subtract sky . . . 9
4.6 Flat correct . . . 11
4.7 Clean bad pixels . . . 11
4.8 Undistort . . . 11
4.9 Coadd frames . . . 13
4.10 Identify apertures . . . 14
4.11 Extract spectra . . . 15
4.12 Combine multiple spectra . . . 17

**5 Other Resources** **17**

**IV Flux calibration** **17**

**V Data products** **17**

6 Filenames 18

7 Pipeline Products 18

8 Data Format 20

VI Appendix: Change notes for the EXES pipeline 20

9 Significant changes 20
9.1 EXES Redux v2.2.0 . . . . . 21
9.2 EXES Redux v2.1.0 (2021-11-22) . . . . . 21
9.3 EXES Redux v2.0.0 (2021-10-21) . . . . . 21
9.4 EXES Redux v1.6.0 (2019-04-02) . . . . . 21
9.5 EXES Redux v1.0.2 (2015-07-17) . . . . . 22
9.6 EXES Redux v1.0.1 (2015-05-14) . . . . . 22
9.7 EXES Redux v1.0.0 (2015-03-17) . . . . . 22

# Part I

# Introduction

This guide describes the data produced by the SOFIA/EXES data reduction pipeline (Redux) for guest investigators. The EXES observing modes for both imaging and spectroscopic observations are described in the SOFIA Observer's Handbook, available from the Proposing and Observing page on the SOFIA website.

This handbook applies to EXES Redux version 2.2.0.

# Part II

# SI Observing Modes Supported

## 1 EXES observing techniques

EXES is a mid-infrared instrument, which implies that its observations must be taken with care in order to remove the effects of the spatially and temporally variable sky background. Unlike some other mid-infrared instruments, EXES does not make use of the chopping secondary mirror, but it does use telescope nods to alternate the object observation with a sky observation.

EXES has four basic nod patterns: stare, nod-on-slit, nod-off-slit, and map. The stare mode is simply a continuous observation of a source, with no telescope offsets to sky. This mode is primarily used for testing, rather than for science observations. In either of the nodding modes, the telescope moves in order to take frames at two different positions (the A position and the B position). For nod-along-slit mode, the frames alternate between placing the source at the bottom and top of the slit. This mode can be used for relatively compact sources (FWHM less than a quarter of the slit). For nod-off-slit mode, the frames alternate between the source and a patch of blank sky. This mode is useful for extended sources or crowded fields. Sky background for either nodding mode is removed by subtracting the two adjacent nod beams. Typically, EXES does a BA nod pattern, so that the nod pair is every second frame minus the previous frame. In map mode, the telescope starts in blank sky, then steps the source across the slit, and then returns to blank sky. The sky frames are averaged together to reduce their noise, and then subtracted from each source frame. This mode is used to map extended sources.

Before each observation sequence, EXES takes a series of calibration frames: typically a spectrum of a blackbody in the same configuration as the science observation and a dark frame. The dark frame is subtracted from the black frame, and the difference is used as a flat field for the 2D object spectrum. The black frame is also used to determine the edges of the orders in cross-dispersed modes.

## 2 EXES instrument configurations

EXES is a spectroscopic instrument that uses a slit to provide a spatial dimension and gratings to provide a spectral dimension. EXES has two gratings used for science: the echelle grating and the echelon. The echelle grating, used alone, produces single-order long-slit spectra. The Medium resolution configuration for EXES uses the echelle grating as the primary disperser at angles 30-65 degrees; the Low resolution configuration uses the echelle grating at angles less than 30 degrees. When the echelon grating is used as the primary disperser and the echelle grating is used as a secondary disperser, EXES produces multi-order, cross-dispersed spectra. In this configuration, the wavelength coverage and the number of orders is determined by the echelle angle. The High-Medium resolution configuration uses an echelle angle of 30-65 degrees. The High-Low resolution configuration uses an angle of less than 30 degrees.

The slit length and width for all modes can be varied depending on the observational needs, although the length is limited in the cross-dispersed modes by the requirement that echelon orders must not overlap.

EXES also has an imaging configuration, but it is used primarily for testing and for acquiring targets for spectroscopic observations. It is not used for science, so it will not be addressed in this document.

# Part III

# Algorithm Description

## 3 Overview of data reduction steps

This section will describe, in general terms, the major algorithms that the EXES Redux pipeline uses to reduce an EXES observation.

See Fig. 1 for a flowchart of the processing steps used by the pipeline.

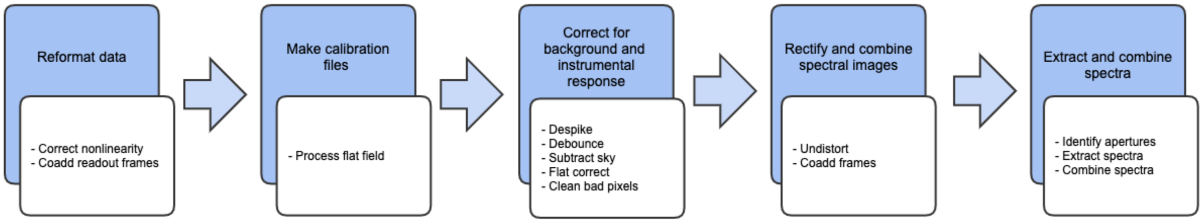

Fig. 1: Processing steps for EXES data.

## 4 Reduction algorithms

The following subsections detail each of the data reduction pipeline steps:

- Correct nonlinearity and coadd readout frames
- Process flat field
- Correct outliers (despike)
- Correct optical shifts (debounce)
- Subtract sky
- Flat correct
- Clean bad pixels
- Rectify spectral orders (undistort)
- Coadd frames
- Identify apertures
- Extract spectra

- Combine multiple spectra

## 4.1 Correct nonlinearity and coadd readout frames

Raw EXES data is stored in FITS image files, as a 3D data cube in which each plane corresponds to a digitized readout of the array. Readouts can be destructive, or non-destructive. The readout method can vary for EXES, so the pattern of readout actions used for a particular observation is recorded in the OTPAT header keyword. The value for OTPAT is a combination of the letters S, T, N, D, and C, each with a digit afterwards. The letters have the following meaning:

- S = spin (no reset, no digitization). This action effectively increases the integration time.
- T = trash (reset, no digitization). This action resets the array without storing the previous value.
- N = non-destructive read (no reset, but digitized). This action stores the current value of the array.
- D = destructive read (reset and digitization). This action stores the current value, and resets the array.
- C = coadd (reset and digitization). This action coadds previous reads in hardware and resets the array.

The digit after each letter gives the number of times the action was taken, minus one. For example: S0 is one spin, N3 is four non-destructive reads.

The pattern listed in the OTPAT can then be repeated any number of times ($n_{pat}$), so that the final number of frames in the raw file is $n_{pat}(n_{nd} + n_d)$, where $n_{nd}$ is the number of nondestructive reads and $n_d$ is the number of destructive reads per pattern.

Since near-infrared detectors do not have completely linear responses to incoming flux, each frame in the raw file may optionally be corrected for nonlinearity before proceeding. The nonlinearity coefficients for EXES were determined by taking a series of nondestructive flat exposures, followed by a single destructive read. Each readout was subtracted from a bias frame, then the resulting counts in each pixel were fit with a polynomial. The linearity correction algorithm reads in these polynomial coefficients from a 3D array and uses them to calculate a correction factor for each pixel in each raw readout. Pixels with counts below the bias level, or above the maximum value used to calculate the nonlinearity coefficients at that pixel, have no correction applied.

After linearity correction, in order to create a usable estimate of the input flux, the non-destructive and destructive reads must be combined in a way that maximizes the signal-to-noise. The EXES pipeline currently has support for combining frames taken with the patterns listed in the following subsections. Equations for the net signal and the associated variance come from Vacca et. al., 2004 (see *Other Resources*). After this initial calculation of the variance for each image pixel, the variance plane is propagated along with the image through all pipeline steps.

After the frames for each pattern are combined according to the algorithms below, the pipeline checks whether multiple patterns were taken at the same nod position. This is recorded in the header keyword NINT. If NINT is greater than 1, then NINT coadded patterns are averaged together before proceeding. Optionally, one or more of the initial patterns may be discarded before averaging.

### All destructive mode

In this mode, each frame is a destructive read. There may be spins in between reads to increase integration time, but there are no non-destructive reads. In this case, the flux is simply the read signal ($s$), minus a standard bias frame ($b$) stored on disk, subtracted from a zero value ($z$), corresponding to a dark current, and divided by the integration time ($\Delta t$) and the pre-amp gain factor ($g_p$):

$$I_{net} = z - \frac{(s - b)}{\Delta t \; g_p}.$$

The dark current is typically zero for EXES.

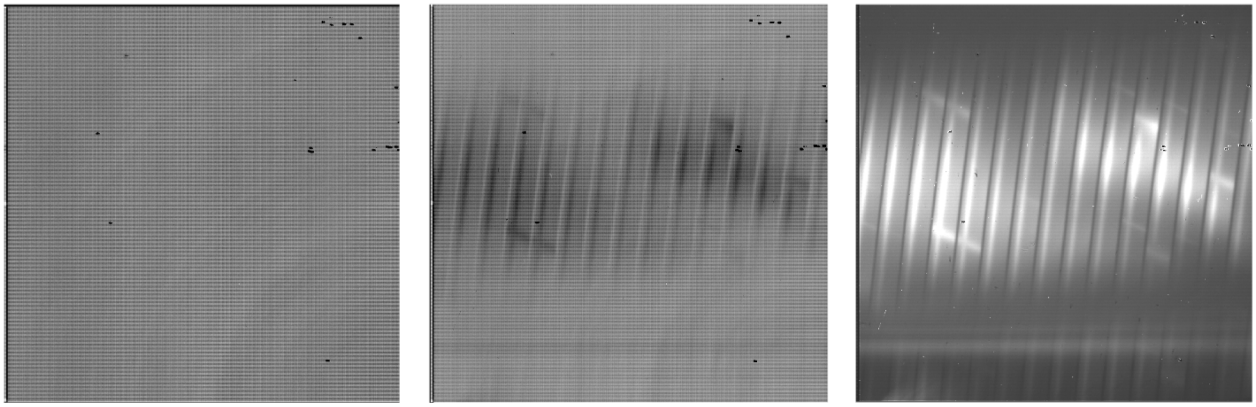

Fig. 2: Sample raw frames in nod-off-slit, High-Medium configuration: a pedestal image (left), a final destructive read (middle), and the Fowler coadd (right). The bright spectrum is visible as a dark line in the final raw readout and a bright line in the coadd.

The variance associated with this readout mode is the Poisson noise (first term) plus the readout noise (second term):

$$V = \frac{I_{net}}{g\,\Delta t} + \frac{{\sigma_{read}}^2}{g^2\,\Delta t^2},$$

where $\sigma_{read}$ is the read noise, and $g$ is the gain factor (to convert from ADU to electrons).

This readout mode may optionally be used for any readout pattern, ignoring the non-destructive reads and using the final destructive read only.

### Fowler mode

In this mode, a number of non-destructive reads are taken at the beginning of the integration, the array integrates for a while, then a number of non-destructive reads and a single destructive read are taken at the end of the integration. The initial reads are averaged together to estimate the “pedestal” level; the final reads are averaged together to estimate the total signal. The pedestal is then subtracted from the total signal to get the net signal. We then subtract the negative signal from a zero value ($z$) and divide by the pre-amp gain factor ($g_p$) and the integration time ($\Delta t$, the time between the first pedestal read and the first signal read). The net flux is then:

$$I_{net} = z - \frac{1}{n_r\,\Delta t\,g_p}\left(\sum_{i=1}^{n_r} s_i - \sum_{i=1}^{n_r} p_i\right).$$

We assume that the number of reads ($n_r$) is the same for the pedestal and the signal.

The variance associated with this readout mode is:

$$V = \frac{I_{net}}{g\,\Delta t}\left(1 - \frac{\delta t\left({n_r}^2 - 1\right)}{3\,\Delta t\,n_r}\right) + \frac{2\,{\sigma_{read}}^2}{g^2\,\Delta t^2\,n_r}.$$

Here, $\delta t$ is the frame time, $\sigma_{read}$ is the read noise, and $g$ is the gain factor to convert from ADU to electrons.

Sample OTPAT values for Fowler mode look like this:

- **N0 D0**: a single non-destructive/destructive pair. This is the minimal Fowler case: the first read is the pedestal, the second read is the signal. There are 2 frames per pattern.
- **N3 S15 N2 D0**: a Fowler 4 set. This set comprises 4 non-destructive reads, 16 spins to increase the integration time, 3 non-destructive reads, and a destructive read. There are 8 frames per pattern.

### Sample-up-the-ramp mode

In this mode, non-destructive reads are taken at even intervals throughout the integration. The count values for all reads are fit with a straight line; the slope of this line gives the net flux. As with the other modes, we then subtract from the zero value and divide by the pre-amp gain and the integration time. The net flux for the evenly-sampled case is calculated as:

$$I_{net} = z - \frac{12}{n_r\ (n_r+1)\,\Delta t\ g_p}\left(\sum_{i=1}^{n_r} s_i\left[i - \frac{n_r+1}{2}\right]\right),$$

where $n_r$ is the total number of reads (non-destructive and destructive).

The variance associated with this readout mode is:

$$V = \frac{6\ I_{net}\left({n_r}^2+1\right)}{5\ g\ \Delta t\ n_r\ (n_r+1)} + \frac{12\ {\sigma_{read}}^2\,(n_r-1)}{g^2\ \Delta t^2\ n_r\ (n_r+1)}.$$

Sample OTPAT values for sample-up-the-ramp mode look like this:

- **N0 S3 N0 S3 N0 S3 D0**: Sample-up-the-ramp with 4 spins in between each non-destructive read, ending with a destructive read. There are 4 frames per pattern.

### Coadding

If the OT pattern contains a coadd (C), it is treated as a destructive read for which the appropriate combination has already been done. That is, if the pattern indicates Fowler mode, for example, the intensity is simply calculated as for a destructive read:

$$I_{net} = z - \frac{s}{\Delta t\ g_p},$$

but the variance is calculated from the net intensity as for the Fowler mode, above.

## 4.2 Process flat field

Each EXES reduction requires a flat observation taken in the same mode as the science data. The instrument configuration (eg. HIGH_MED or MEDIUM), central wavenumber, slit length, slit width, and grating angle must all match. This flat observation is used to correct for spectral and spatial instrumental gain variations, and to calibrate the source intensity of the science observation.

The flat observation consists of an observation of a blackbody in EXES's calibration box (*black*) and a dark frame (*dark*). If the blackbody's temperature is known, then dividing the sky-subtracted science frame by *black-dark*, normalized by the blackbody function, gives a calibrated intensity as follows (Lacy et. al., 2002, see *Other Resources*).

$$\begin{aligned} S_\nu(black - dark) &= B_\nu(T_{black})\ R_\nu \\ S_\nu(obj - sky) &\approx I_\nu(obj)\ R_\nu \\ I_\nu(obj) &\approx S_\nu(obj - sky)\ \frac{B_\nu(T_{black})}{S_\nu(black - dark)} \end{aligned}$$

where $S_\nu$ is the measured signal, $B_\nu(T)$ is the blackbody function at temperature *T*, and $R_\nu$ is the instrumental responsivity. The master flat frame produced by the EXES pipeline, therefore, is the blackbody function calculated at the temperature recorded in the flat frame's header (BB_TEMP), divided by *black-dark* (Fig. 3).

In all instrument configurations, the *black* frame is also used to determine an illumination mask that defines the good pixels for extraction. In the cross-dispersed modes, the *black* frame is further used to determine the edges of the orders for extraction and to check the optical distortion parameters. The pipeline cleans and undistorts the *black* frame, takes its derivative, then performs a 2D FFT of the resulting image to determine the spacing and orientation of the orders on the array. In particular, the value of $k_{rot}$, the angle of the order rotation, is adjusted automatically at this step: it is calculated from the characteristics of the FFT, then used to recompute the undistortion of the *black* frame, and redo the FFT, until the value of $k_{rot}$ converges, or 5 iterations are reached. This process must be closely monitored: if the *black* frame has insufficient signal, or the optical parameters used to calculate the distortion correction are insufficiently accurate, the spacing and rotation angle may be wrongly calculated at this step. These values can be manually overridden in parameters for the pipeline if necessary.

The distortion parameters and order definitions, as determined from the *black* frame, are written to the header of the master flat, to be used in later reduction steps.

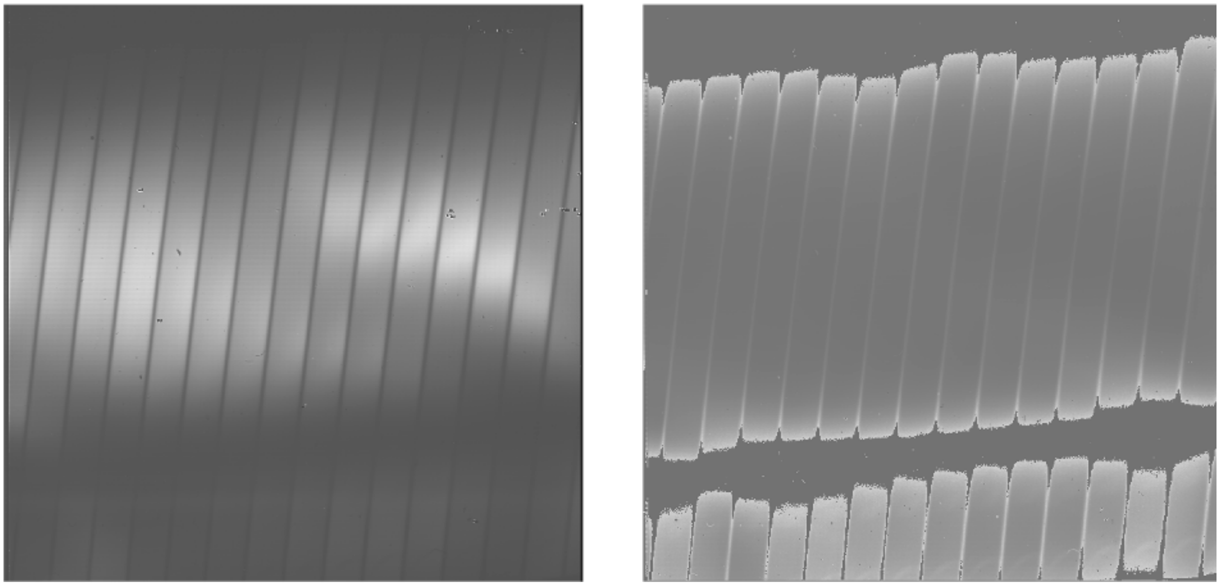

Fig. 3: Sample flat in High-Medium configuration. The left image is the black frame; the right image is the final processed flat frame. Unilluminated regions are set to zero in the final frame.

## 4.3 Despike

After coadding the raw frames to make A and B nod frames, the pipeline attempts to identify and correct spikes (outlier data points) in the individual images. All A frames are compared, and any pixels with values greater than a threshold number of standard deviations from the mean value across the other frames are replaced with that mean value. B frames are similarly compared with each other. The threshold for identifying spikes is a tunable parameter; the default value is 20 sigma. Also in the despike step, frames with significantly different overall background levels (“trashed” frames) may be identified automatically and removed from subsequent reduction. It is common, for example, for the first frame to look significantly different from the rest of the frames. In this case, leaving out this frame may improve the signal-to-noise of the final result.

## 4.4 Debounce

After despiking, there is an optional step called debouncing, which may help alleviate the effects of small optics shifts ("bounces") between the nod beams. If there is a slight mismatch in the order placement on the array, it can lead to poor background subtraction when the nods are subtracted. In the debouncing algorithm, each nod pair is undistorted, then the B nod is shifted slightly in the spatial direction and differenced with the A nod. The shift direction that results in a minimum squared difference (summed over the spectral direction) is used as the bounce direction. The amplitude of the shift is controlled by the bounce parameter, which should be set to a number whose absolute value is between 0 and 1 (typically 0.1). If the bounce parameter is set to a positive number, only the above shift (the first-derivative bounce) will be corrected. If the bounce parameter is set to a negative value (e.g. -0.1), the debouncing algorithm will also attempt to fix the second-derivative bounce by smoothing the A or B nod slightly; the amount of smoothing is also controlled by the absolute value of the bounce parameter. Note that if the observed source is too near the edge of the order, it may confuse the debouncing algorithm; in this case, it is usually best to turn debouncing off (i.e. set the bounce parameter to 0). The default is not to use the debounce algorithm.

## 4.5 Subtract sky

In either nod-on-slit or nod-off-slit mode, each B nod is subtracted from each adjacent A nod (Fig. 4). This step usually removes most of the background emission from the image, but if there were changes in the sky level between the two nod frames, there may still be some residual sky signal. For the nod-off-slit mode, this residual signal can be estimated and corrected for before subtracting the nods, by subtracting a weighted fraction of the B data from the A data. The weighting is chosen to minimize the squared difference between the A and B nods. For the nod-on-slit mode, the mean background at each wavelength may be subtracted after nod subtraction and distortion correction, so that the wavelengths align with the rows or columns. The pipeline performs this step immediately before coadding, if desired.

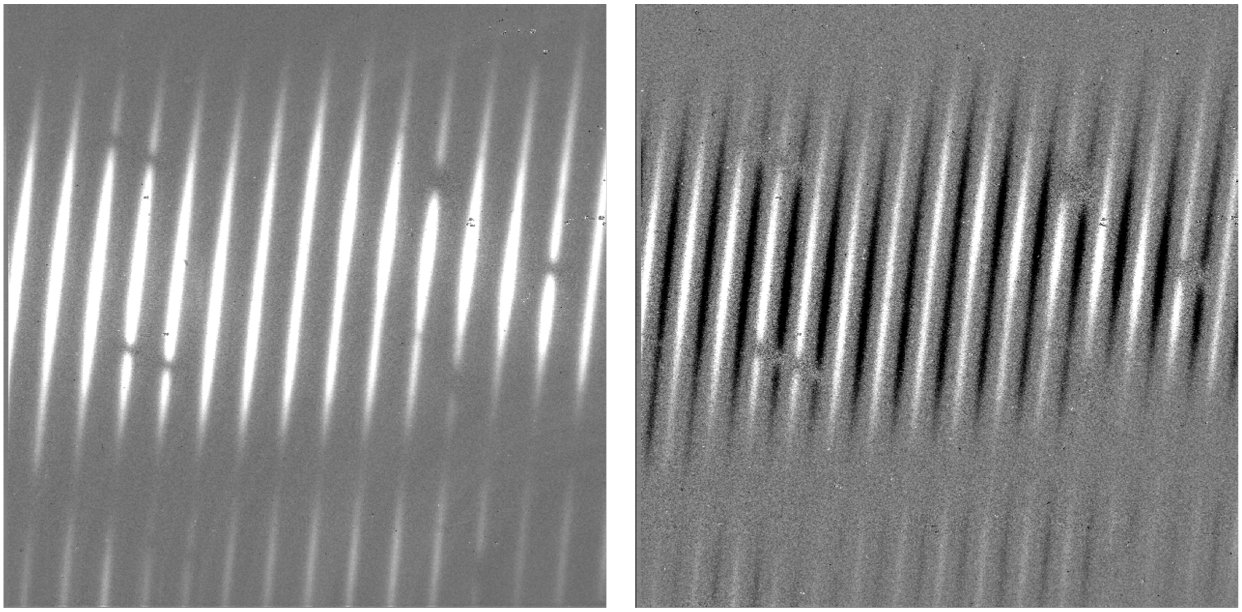

Fig. 4: Background subtracted frames in nod-off-slit (left) and nod-on-slit (right), in High-Medium configuration.

For the mapping mode, each of the steps across the source is treated as an A frame. The three sky frames taken at the end of the map are averaged together and this average is subtracted from each A frame (Fig. 5). This mode is usually used for extended sources that fill the entire slit, in which case there is no way to estimate or correct for any residual background remaining after sky subtraction. The three sky frames at the end of the map can be part of the map (without science target signal) or dedicated sky positions away from the source.

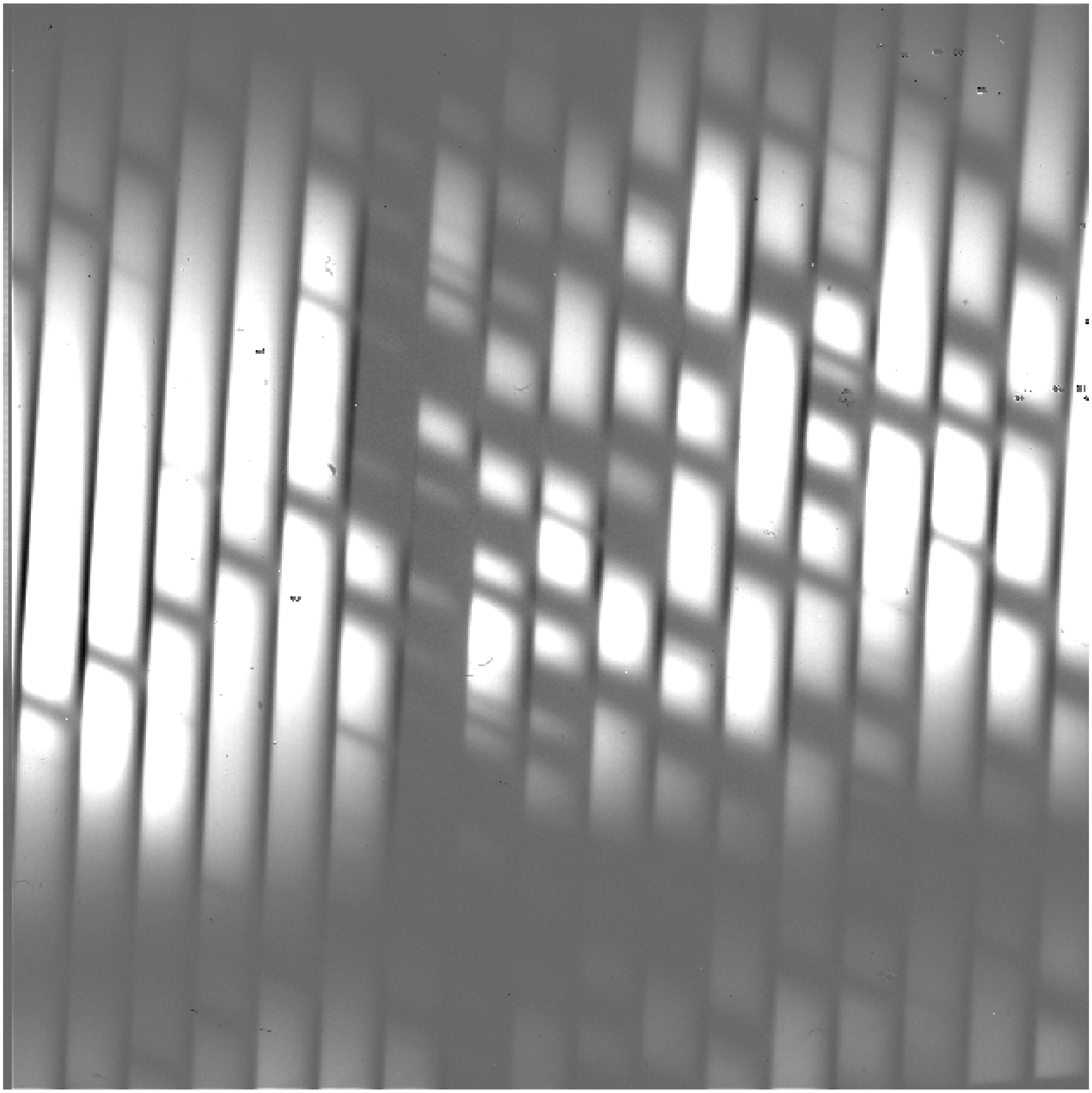

Fig. 5: Background subtracted image in High-Medium configuration with map mode. The extended source fills the entire slit.

## 4.6 Flat correct

After background subtraction, each science frame is multiplied by the processed flat calibration frame, described above. This has the effect of both correcting the science frame for instrumental response and calibrating it to intensity units.

## 4.7 Clean bad pixels

In this step, bad pixels are identified and corrected. Bad pixels may first be identified in a bad pixel mask, provided by the instrument team. In this FITS image, pixels that are known to be bad are marked with a value of 0; good pixels are marked with a value of 1. Alternately, bad pixels may be identified from their noise characteristics: if the error associated with a pixel is greater than a threshold value (by default: 20) times the mean error for the frame, then it is marked as a bad pixel. Unlike the despike algorithm, which identifies outliers by comparing separate frames, outliers in this algorithm are identified by comparing the values within a single frame.

Bad pixels are corrected by using neighboring good values to linearly interpolate over the bad ones. The search for good pixels checks first in the y-direction, then in the x-direction. If good pixels cannot be identified within a 10-pixel radius, then the bad pixel will not be corrected. Note that bad pixels are also fixed or ignored in the extraction process, so it is not critical to correct all bad pixels at this stage.

## 4.8 Undistort

Next, the pipeline corrects the image for optical distortion, resampling the image onto a regular grid of spatial and spectral elements. The EXES distortion correction uses knowledge of the optical design of the instrument, as well as data recorded in the header of the observation, to calculate and correct for the optical distortion in each observation.

The optical parameters used to calculate the distortion correction are listed in the table below. Most of these parameters should not change, or should change rarely, so they have default values listed in configuration tables available to the pipeline (the *.dat* files listed below). Values in the *tortparm.dat* configuration table tend to change over time, as the instrument is adjusted, so their defaults vary by date. Some of the distortion parameters must come from the headers and do not have default values. One parameter is tuned at run-time (*krot*), and one must be manually optimized by the user (*waveno0*).

Table 1: Optical distortion parameters.

| **Parameter** | **Default value** | **Source** | **Comment** |
|---|---|---|---|
| *hrfl* | (varies by date) | *tortparm.dat* | High resolution chamber focal length |
| *xdfl* | (varies by date) | *tortparm.dat* | Assumed cross- dispersed focal length |
| *slitrot* | (varies by date) | *tortparm.dat* | Slit rotation |
| *detrot* | (varies by date) | *tortparm.dat* | Detector rotation |
| *hrr* | (varies by date) | *tortparm.dat* | R number for echelon grating |
| *brl* | 0.00 | *tortparm.dat* | Barrel distortion parameter |
| *x0brl* | 0.00 | *tortparm.dat* | X-coordinate for the center of the barrel distortion |
| *y0brl* | 0.00 | *tortparm.dat* | Y-coordinate for the center of the barrel distortion |
| *hrg* | -0.015 | *headerdef.dat* | Gamma (out of plane) angle for the echelon grating (radians) |
| *xdg* | 0.033 | *headerdef.dat* | Gamma angle for cross- dispersion grating (radians) |
| *xdlrdgr* | 0.001328 | *headerdef.dat* | Low resolution groove spacing (cm) |
| *xdmrdgr* | 0.003151 | *headerdef.dat* | Medium resolution groove spacing (cm) |
| *pixelwd* | 0.0025 | *headerdef.dat* | Pixel width |
| *waveno0* | (none) | Header keyword WAVENO0 | Planned central wavenumber |
| *echelle* | (none) | Header keyword ECHELLE | Echelle grating angle (deg) |
| *krot* | (varies by date) | *tortparm.dat* /<br>Optimized at run-time | Rotation between chambers |

Several of these parameters are then combined or recalculated before being used in the distortion correction. The cross-dispersed R number (*xdr*) is calculated from the groove spacing (*xdlrdgr* or *xdmrdgr*), the grating angle (*echelle*), and the central wavenumber (*waveno0*). The expected order spacing for cross-dispersed modes is calculated from the cross-dispersed R number (*xdr*), the cross-dispersed focal length (*xdfl*), the pixel width (*pixelwd*), and the echelon gamma angle (*hrg*). The order spacing and the rotation angle (*krot*) are optimized for cross-dispersed modes when the calibration frame is processed, as described above.

The distortion correction equations calculate undistorted coordinates by correcting for the following effects for the cross-dispersed modes:

- slit skewing within orders due to the echelon "smile" (*hrg, hrr, slitrot*)
- nonlinearity of the echelon spectrum (*hrr, pixelwd, hrfl*)
- skewing by spectrum rotation on the detector due to the angle between the chambers, and the cross-dispersion smile (*xdg, xdr, krot, pixelwd, xdfl, hrfl, hrr*)
- nonlinearity of the cross-dispersion spectrum (*xdr, pixelwd, xdfl*)
- barrel distortion (*brl, x0brl, y0brl*)

For the long-slit modes, the distortion correction accounts for:

- skewing by the cross-dispersion smile (*xdg, xdr, slitrot, pixelwd, xdfl*)
- nonlinearity of the cross-dispersion spectrum (*xdr, pixelwd, xdfl*)
- barrel distortion (*brl, x0brl, y0brl*)

When the undistorted coordinates have been calculated, the 2D science image is interpolated from raw *(x,y)* pixel coordinates onto these coordinates. By default, a cubic interpolation is used that closely approximates a sinc interpolation function; bilinear interpolation is also available.

After distortion correction, the spatial and spectral elements should be aligned with the columns and rows of the image (Fig. 6). For the long-slit modes, the x-axis is the dispersion axis; for cross-dispersed modes, the y-axis is the dispersion axis.

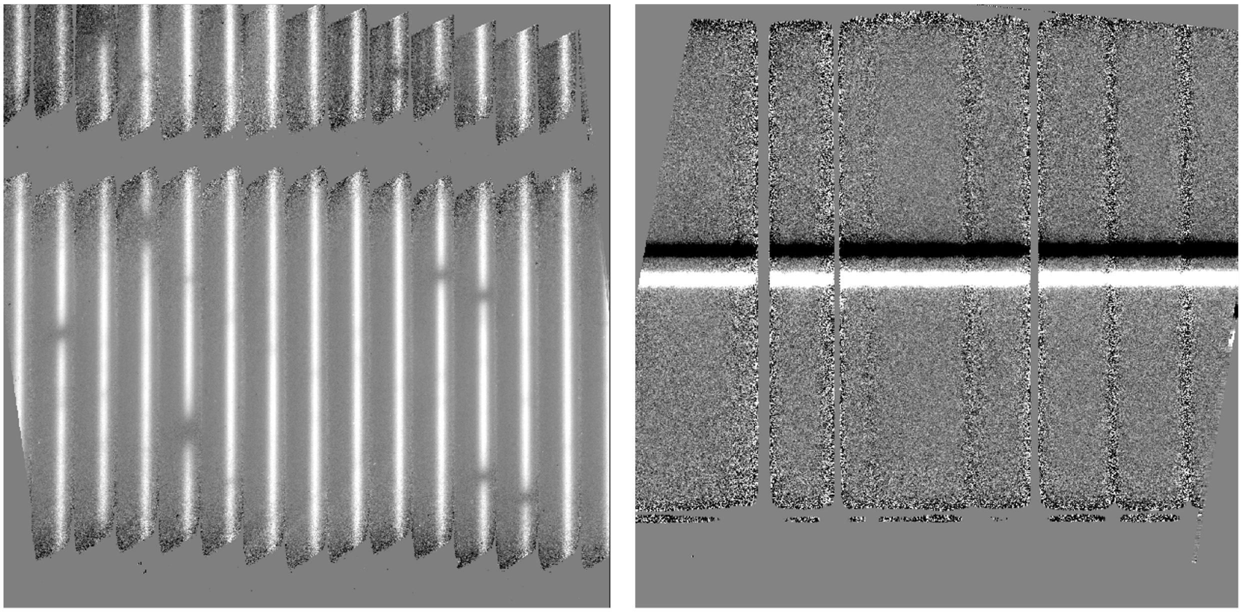

Fig. 6: Undistorted images in High-Medium configuration (left) and Medium configuration (right).

## 4.9 Coadd frames

The final step in the image processing prior to spectral extraction is coadding of the background-subtracted, undistorted frames. If desired, coaddition can be skipped, and individual spectra can be extracted from each frame, to be combined later. For faint sources, however, coaddition of the 2D image is recommended, for more accurate extraction.

Prior to coadding, the spectra can be shifted in the spatial direction to account for slight shifts in the position of the object in the slit between nod pairs. The shift is calculated by creating a template of the spectrum from all frames, averaged in the spectral direction, and cross-correlating each frame with the template.

By default, the pipeline weights all input frames by their correlation with this spectral template, so that if a frame is unusually noisy, or missed a nod, or had some other error, it will not contribute significantly to the coadded frame. However, it is possible to force the pipeline to use an unweighted coadd, if desired, or to specify the weights to use directly, or to explicitly identify frames for exclusion from the coadd.

The pipeline typically only coadds frames within a single file. Optionally, it may instead coadd frames from all files. This action is performed as an robust mean across all input frames, with or without weights. In cases where the target has not moved across the array between input filess, this option may help increase signal-to-noise prior to extraction.

The final action in the coadd step is to rotate images taken in the cross-dispersed mode in order to align the spectra with the image rows (Fig. 7). This enables the pipeline to handle the spectral extraction in a consistent way for both the long-slit and the cross-dispersed data.

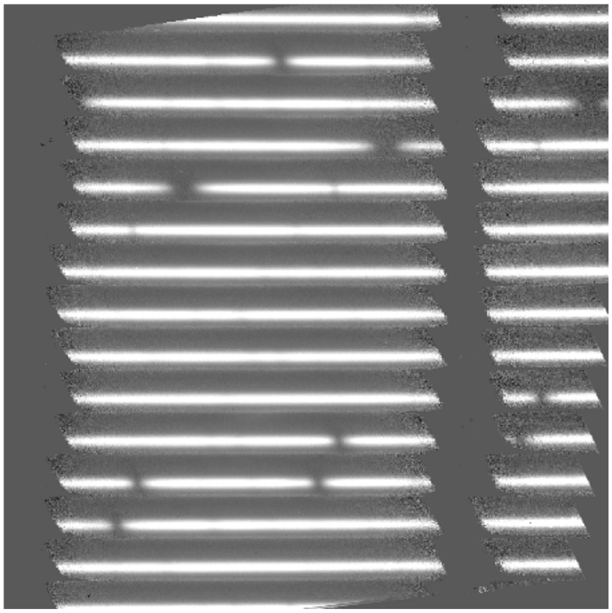

Fig. 7: Coadded image in High-Medium configuration, rotated to align the dispersion axis with the image rows.

## 4.10 Identify apertures

In order to aid in spectral extraction, the pipeline constructs a smoothed model of the relative intensity of the target spectrum at each spatial position, for each wavelength. This spatial profile is used to compute the weights in optimal extraction or to fix bad pixels in standard extraction (see next section). Also, the pipeline uses the median profile, collapsed along the wavelength axis, to define the extraction parameters.

To construct the spatial profile, the pipeline first subtracts the median signal from each column in the rectified spectral image to remove the residual background. The intensity in this image in column *i* and row *j* is given by

$$O_{ij} = f_i P_{ij}$$

where $f_i$ is the total intensity of the spectrum at wavelength *i*, and $P_{ij}$ is the spatial profile at column *i* and row *j*. To get the spatial profile $P_{ij}$, we must approximate the intensity $f_i$. To do so, the pipeline computes a median over the wavelength dimension (columns) of the order image to get a first-order approximation of the median spatial profile at each row $P_j$. Assuming that

$$O_{ij} \approx c_i P_j,$$

the pipeline uses a linear least-squares algorithm to fit $P_j$ to $O_{ij}$ and thereby determine the coefficients $c_i$. These coefficients are then used as the first-order approximation to $f_i$: the resampled order image $O_{ij}$ is divided by $f_i$ to derive $P_{ij}$. The pipeline then fits a low-order polynomial along the columns at each spatial point *s* in order to smooth the profile and thereby increase its signal-to-noise. The coefficients of these fits can then be used to determine the value of $P_{ij}$ at any column *i* and spatial point *j* (see Fig. 8, left). The median of $P_{ij}$ along the wavelength axis generates the median spatial profile, $P_j$ (see Fig. 8, right).

The pipeline then uses the median spatial profile to identify extraction apertures for the source. The aperture centers can be identified automatically by iteratively finding local maxima in the absolute value of the spatial profile, or can be specified directly by the user. By default, a single aperture is expected and defined for nod-off-slit mode; two apertures are expected for nod-along-slit mode.

Besides the aperture centers, the pipeline also specifies a PSF radius, corresponding to the distance from the center at which the flux from the source falls to zero. By default, this value is automatically determined from the width of a

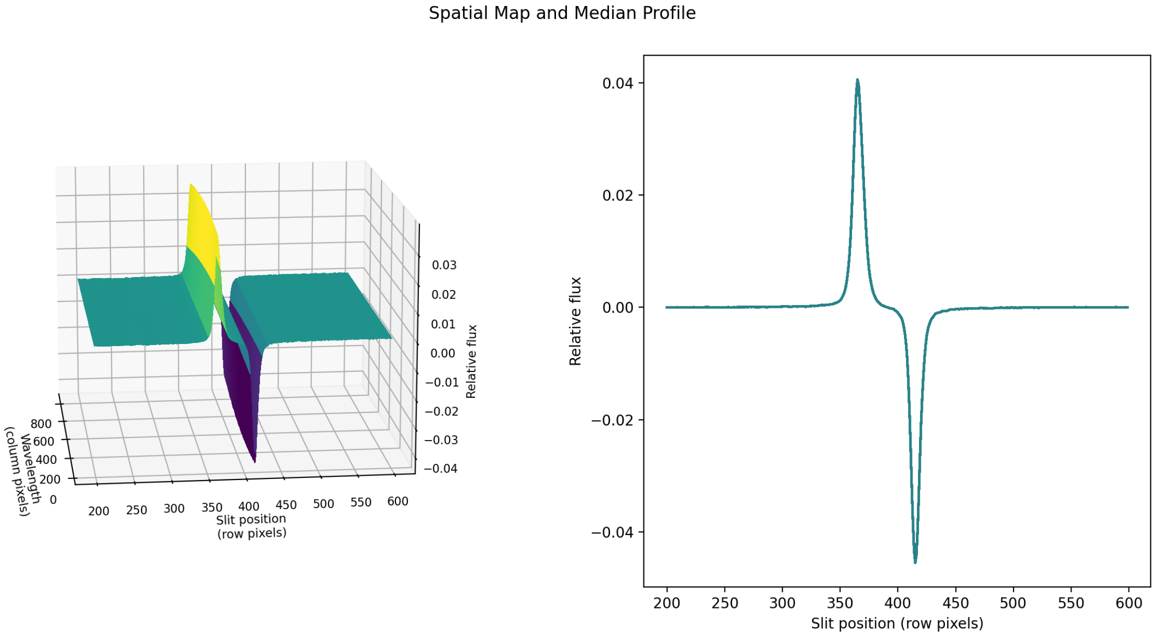

Fig. 8: Spatial model and median spatial profile, for the Medium configuration image in Fig. 6. The spatial model image here is rotated for comparison with the profile plot: the y-axis is along the bottom of the surface plot; the x-axis is along the left.

Gaussian fit to the peak in the median spatial profile, as

$$R_{psf} = 2.15 \cdot \text{FWHM}.$$

For optimal extraction, the pipeline also identifies a smaller aperture radius, to be used as the integration region:

$$R_{ap} = 0.7 \cdot \text{FWHM}.$$

This value should give close to optimal signal-to-noise for a Moffat or Gaussian profile. The pipeline also attempts to specify background regions outside of any extraction apertures, for fitting and removing the residual sky signal. All aperture parameters may be optionally overridden by the pipeline user.

## 4.11 Extract spectra

The spectral extraction algorithms used by the pipeline offer two different extraction methods, depending on the nature of the target source. For point sources, the pipeline uses an optimal extraction algorithm, described at length in the Spextool paper (see the *Other Resources* section, below, for a reference). For extended sources, the pipeline uses a standard summing extraction.

In either method, before extracting a spectrum, the pipeline first uses any identified background regions to find the residual sky background level. For each column in the 2D image, it fits a low-order polynomial to the values in the specified regions, as a function of slit position. This polynomial determines the wavelength-dependent sky level ($B_{ij}$) to be subtracted from the spectrum ($D_{ij}$).

### Standard extraction

The standard extraction method uses values from the spatial profile image ($P_{ij}$) to replace bad pixels and outliers, then sums the flux from all pixels contained within the PSF radius. The flux at column *i* is then:

$$f_{i,\mathrm{sum}} = \sum_{j=j_1}^{j_2} (D_{ij} - B_{ij})$$

where $j_1$ and $j_2$ are the upper and lower limits of the extraction aperture (in pixels):

$$j_1 = t - R_{PSF}$$
$$j_2 = t + R_{PSF}$$

given the aperture center (*t*). This extraction method is the only algorithm available for extended sources.

### Optimal extraction

Point sources may occasionally benefit from using standard extraction, but optimal extraction generally produces higher signal-to-noise ratios for these targets. This method works by weighting each pixel in the extraction aperture by how much of the target's flux it contains. The pipeline first normalizes the spatial profile by the sum of the spatial profile within the PSF radius defined by the user:

$$P'_{ij} = P_{ij} \Big/ \sum_{j=j_1}^{j_2} P_{ij}.$$

$P'_{ij}$ now represents the fraction of the total flux from the target that is contained within pixel *(i,j)*, so that $(D_{ij} - B_{ij})/P'_{ij}$ is a set of *j* independent estimates of the total flux at column *i*. The pipeline does a weighted average of these estimates, where the weight depends on the pixel's variance and the normalized profile value. Then, the flux at column *i* is:

$$f_{i,\mathrm{opt}} = \frac{\sum_{j=j_3}^{j_4} M_{ij} P'_{ij} (D_{ij} - B_{ij}) / (V_{D_{ij}} + V_{B_{ij}})}{\sum_{j=j_3}^{j_4} M_{ij} {P'_{ij}}^2 / (V_{D_{ij}} + V_{B_{ij}})}$$

where $M_{ij}$ is a bad pixel mask and $j_3$ and $j_4$ are the upper and lower limits given by the aperture radius:

$$j_3 = t - R_{ap}$$
$$j_4 = t + R_{ap}$$

Note that bad pixels are simply ignored, and outliers will have little effect on the average because of the weighting scheme.

The variance for the standard spectroscopic extraction is a simple sum of the variances in each pixel within the aperture. For the optimal extraction algorithm, the variance on the $i^{th}$ pixel in the extracted spectrum is calculated as:

$$V_i = \sum_{j=j_3}^{j_4} \frac{M_{ij}}{{P'_{ij}}^2 V_{ij}}$$

where $P'_{ij}$ is the scaled spatial profile, $M_{ij}$ is a bad pixel mask, $V_{ij}$ is the variance at each background-subtracted pixel, and the sum is over all spatial pixels $j$ within the aperture radius. The error spectrum for 1D spectra is the square root of the variance.

### Wavelength calibration

Wavelength calibration for EXES is calculated from the grating equation, using the optical parameters and central wavelength for the observation. It is stored as a wavelength calibration map identifying the wavelength for each column in each spectral order.

The wavelength calibration is expected to be good to within one pixel of dispersion, if the central wavelength is accurate. To refine the wavelength solution, the pipeline allows the user to identify a single spectral feature in the extracted spectrum. Using this feature, the central wavelength is recalculated and the wavelength calibration is adjusted to match. However, the distortion correction also depends on the central wavelength, so if the recalculated value is significantly different from the assumed value, the reduction should be repeated with the new value to ensure an accurate distortion correction and dispersion solution.

## 4.12 Combine multiple spectra

The final pipeline step is the combination of multiple spectra of the same source, from separate observations, apertures, and orders. For all configurations, the individual extracted 1D spectra for each order are combined with a robust weighted mean, by default.

For cross-dispersed configurations, the pipeline performs an automatic merge of the spectra from all orders. This algorithm uses the signal-to-noise ratio in overlapping regions to determine which pixels to exclude, and performs a weighted mean of any remaining overlapping regions. Artifacts near the edges of orders may optionally be manually trimmed before the merge.

# 5 Other Resources

For more information on the reduction algorithms used in the EXES package, adapted from the TEXES pipeline, see the TEXES paper:

> TEXES: A Sensitive High-Resolution Grating Spectrograph for the Mid-Infrared, J.H. Lacy, M.J. Richter, T.K. Greathouse, D.T. Jaffe and Q. Zhu (2002, PASP 114, 153)

For more information on the reduction algorithms used in FSpextool, see the Spextool papers:

> Spextool: A Spectral Extraction Package for SpeX, a 0.8-5.5 Micron Cross-Dispersed Spectrograph, Michael C. Cushing, William D. Vacca and John T. Rayner (2004, PASP 116, 362).
>
> A Method of Correcting Near-Infrared Spectra for Telluric Absorption, William D. Vacca, Michael C. Cushing and John T. Rayner(2003, PASP 115, 389).
>
> Nonlinearity Corrections and Statistical Uncertainties Associated with Near-Infrared Arrays, William D. Vacca, Michael C. Cushing and John T. Rayner (2004, PASP 116, 352).

# Part IV

# Flux calibration

EXES spectra are calibrated to physical intensity units ($erg/s/cm^2/sr/cm^{-1}$) when they are multiplied by the blackbody-normalized calibration frame. This process does not account for residual atmospheric effects. These may be corrected by dividing by a telluric standard spectrum, or a model of the atmosphere.

# Part V

# Data products

## 6 Filenames

EXES output files are named according to the convention:

FILENAME = *F[flight]_EX_SPE_AOR-ID_SPECTEL1SPECTEL2_type_FN1[-FN2][_SN]*.fits

where *flight* is the SOFIA flight number, *EX* is the instrument identifier, *SPE* specifies that it is a spectral file, *AOR-ID* is the AOR identifier for the observation, *SPECTEL1SPECTEL2* are the keywords specifying the spectral elements used, *type* is the three-letter identifier for the product type (listed in the table below), and *FN1* is the file number corresponding to the input file. *FN1-FN2* is used if there are multiple input files for a single output file, where *FN1* is the file number of the first input file and *FN2* is the file number of the last input file. If the output file corresponds to a single frame of the input file, there may also be a trailing serial number (SN) that identifies the input frame associated with the output file.

## 7 Pipeline Products

The following table lists all intermediate products generated by the pipeline for EXES spectral modes, in the order in which they are produced. The product type is stored in the FITS headers under the keyword PRODTYPE. By default, the *raw*, *undistorted*, *coadded*, *spec*, *combspec*, *mrgordspec*, *flat*, *mask,* and *wavemap* products are saved. If a subarray readout was used, the second dimension for the image products may be smaller than 1024. Currently the pipeline does not handle or create any spectral mapping products (e.g. data cubes).

For calibration purposes, it is often useful to set pipeline parameters to skip nod subtraction and extract a sky spectrum rather than a science spectrum. Data products reduced in this mode have separate product types, with ‘sky_’ prepended to the standard product name, and file codes that begin with the letter ‘S’.

Table 2: Intermediate and final data products for EXES reductions

| Step | Description | Product Type | File code | Saved by default | Size |
|---|---|---|---|---|---|
| Load Data | Raw frames coadded according to OTPAT; variance planes appended at the end of all raw frames | *raw* | RAW | Y | 1024 x 1024 x 2 $N_{pat}$ where $N_{pat}$ is the number of readout patterns |
| Make Flat | Calibration frame created from *black-dark* | *inverseflat* | IFT | N | 1024 x 1024 x 2 |
| Make Flat | Wavelength calibration map | *wavemap* | WVM | Y | 1024 x 1024 x 2 |

continues on next page

Table 2 – continued from previous page

| Step | Description | Product Type | File code | Saved by default | Size |
|---|---|---|---|---|---|
| Despike | Outlier pixels identified and corrected. Comparison is across similar frames. | *despiked* | DSP | N | 1024 x 1024 x 2 $N_{pat}$ |
| Debounce | Corrected for optics shifts between nods | *debounced* | DBC | N | 1024 x 1024 x 2 $N_{pat}$ |
| Subtract Nods | B nods or sky frames subtracted from A nods or map steps | *nodsubtracted* | NSB | N | 1024 x 1024 x 2 $N_A$<br>where $N_A$ is the number of A nods or map steps |
| Flat Correct | Multiplied by flat | *flatted* or *sky_flatted* | FTD or SFT | N | 1024 x 1024 x 2 $N_A$ |
| Clean Bad Pixels | Bad pixels identified from variance or bad pixel mask | *cleaned* or *sky_cleaned* | CLN or SCN | N | 1024 x 1024 x 2 $N_A$ |
| Undistort | Corrected for optical distortion | *undistorted* or *sky_undistorted* | UND or SUN | Y | 1024 x 1024 x 2 $N_A$ |
| Undistort | Undistorted flat frame (1/inverseflat) with order definition keywords | *flat* | FLT | Y | 1024 x 1024 x 2 |
| Coadd Pairs | All frames averaged together, if desired | *coadded* or *sky_coadded* | COA or SCO | Y | 1024 x 1024 x $N_C$<br>where $N_C$ is the number of coadded frames (typically $N_C$ = 1) |
| Coadd Pairs | All remaining frames, separated into different files | *image* or *sky_image* | IMG or SIM | N | 1024 x 1024 x 2 |
| Extract Spectra | 1D extracted spectra from all orders and apertures | *spec* or *sky_spec* | SPC or SSP | Y | $N_w$ x 4 x $N_{ord}N_{ap}$,<br>where $N_w$ is the number of wavelength bins, $N_{ord}$ is the number of orders, and $N_{ap}$ is the number of apertures |

continues on next page

Table 2 – continued from previous page

| Step | Description | Product Type | File code | Saved by default | Size |
|---|---|---|---|---|---|
| Extract Spectra | Bad pixel mask indicating pixels ignored during extraction | *mask* or *sky_mask* | MSK or SMK | Y | 1024 x 1024 |
| Combine Spectra | 1D spectra from separate files or apertures combined together | *combspec* or *sky_combspec* | CMB or SCM | Y | $N_w$ x 4 x $N_{ord}$ |
| Merge Orders | 1D spectrum from all cross-dispersed orders merged together | *mrgordspec* or *sky_mrgordspec* | MRD or SMD | Y | $N_{mw}$ x 4 where $N_{mw}$ is the number of merged wavelength bins |

## 8 Data Format

All files produced by the pipeline are FITS single-extension image files. All image products are 3-D arrays of data, where the first half of the planes are the image frames and the second half of the planes are the variances associated with each pixel in each frame. The extracted spectral products are one-dimensional spectra, stored in four rows of data. The first row is the wavelength, the second is the flux, the third is the error (standard deviation), and the fourth is a reference model atmospheric transmission spectrum. If there were multiple orders in the spectrum (e.g. the High-Medium configuration) or multiple apertures selected (e.g. for the nod-on-slit mode), then the spectrum for each aperture and order is stacked into a different plane. The length of the row varies depending on the data, but is typically around 1000 pixels before merging.

## Part VI

# Appendix: Change notes for the EXES pipeline

## 9 Significant changes

Below are listed the most significant changes for the EXES pipeline over its history, highlighting impacts to science data products. See the data handbooks or user manuals associated with each release for more information.

For previously processed data, check the PIPEVERS keyword in the FITS header to determine the pipeline version used.

## 9.1 EXES Redux v2.2.0

*User manual: Rev. D*

- Added HRR and DETROT to distortion correction parameters in configuration files, so that the defaults can be overridden by date.
- Fixed a bug in integration tossing in the initial read and coadd, in the case where a single valid integration remains after tossing bad ones.

## 9.2 EXES Redux v2.1.0 (2021-11-22)

*User manual: Rev. C*

- Designated separate product names and types for sky spectra to avoid overwriting science products.
- Added additional support for intermediate reductions, including sky spectrum products and undistorted files.
- Added capability to compose flat fields from separate subarrays (rasters).
- Modified assumed plate scale by mode to more accurately reflect anamorphic magnification effects for cross-dispersed modes.

## 9.3 EXES Redux v2.0.0 (2021-10-21)

*User manual: Rev. B*

- Adopted EXES PI team version for SOFIA DPS support as a facility class instrument.
- Modified NAIF ID handling to not write default -9999 value for sidereal targets.
- Added option to coadd frames across all files with outlier rejection.
- Added ASSC_MSN keyword to track all input mission IDs and ASSC_OBS to track all input OBS-IDs.
- Removed a scaling factor in the coadd step, originally intended for unit conversion and flux conservation, that was historically inconsistently applied to flux and variance planes.

## 9.4 EXES Redux v1.6.0 (2019-04-02)

*User manual: Rev. A*

- Added handling for darks to allow for black-dark flat generation method
- Added option to reject 1 or 2 initial readout patterns before readout coadd
- Added option to use only destructive frames for readout coadds
- Modified options to allow manual selection of bad pixel masks
- Allowed manual overrides for the edges of order masks
- Added option to debounce in the spectral direction, as well as the spatial
- Added manual options to set asymmetric apertures with start and end values
- Added options to subtract a dark frame instead of nods, for sky emission spectra

## 9.5 EXES Redux v1.0.2 (2015-07-17)

*User manual: Rev. A*

- Modified nonlinearity correction to allow for a lower-limit plane in the coefficient reference file.

## 9.6 EXES Redux v1.0.1 (2015-05-14)

*User manual: Rev. -*

- Added ASSC_AOR keyword to track all input AOR-IDs.
- Attached reference atmospheric transmission data to spectra.
- Improved wavelength calibration.

## 9.7 EXES Redux v1.0.0 (2015-03-17)

*User manual: Rev. -*

- Initial release.