STIS CCD Read Noise During Side-2 Operations

Thomas M. Brown
December 6, 2001

ABSTRACT

Due to a failure of the primary (Side-1) STIS electronics, the STIS has been operating on its redundant (Side-2) electronics since July 2001. CCD data taken on Side 2 show an increase in effective read noise of approximately 1 e^- pix^{-1} for gain=1 and 0.2 e^- pix^{-1} for gain=4. The increased read noise manifests as a herring-bone pattern that can be seen easily in short-exposure CCD images. When the 2-D CCD images are converted to 1-D time series, Fourier transforms demonstrate that the read noise patterns are temporally correlated. The frequency of the read noise pattern ranges from 15.5 to 18 kHz, which corresponds to a spatial frequency on the detector of approximately 3 pixels (horizontally, in the direction of serial clocking). The gross variations in this frequency are correlated with the temperature of the CCD housing, but the correlation is too loose to use this temperature to predict the frequency. Simple filtering of the data can sometimes mitigate the pattern noise, but the filter must be carefully tuned to a narrow frequency range (tens of Hz) centered at the precise frequency of the pattern, to avoid introducing artifacts in the data. I provide a code that can be used to analyze and filter the pattern noise in STIS CCD data.

Introduction

On 16 May 2001, the primary (Side-1) set of STIS electronics failed, and operations were subsequently resumed on the redundant (Side-2) set. Although most operations on Side 2 are identical to those on Side 1, some of the thermal control of the STIS CCD has been lost (see *Instrument Science Report STIS 2001-003* for details), and there is an obvious pattern in STIS CCD data, due to an increase in the read noise. This increase in read noise is most obvious in short exposures (e.g., bias frames), because they are dominated

by read noise (as opposed to Poisson noise from the dark, sky, and source counts), and because the increase in read noise is spatially correlated on the scale of about 3 pixels. The pattern is not fixed on the detector: it changes from exposure to exposure, and drifts within each image, such that it appears as a "herring bone" pattern. Data taken through the nominal amplifier (D) show an increase in read noise of approximately 1 e$^-$ pix^{-1} for gain=1 and 0.2 e$^-$ pix^{-1} for gain=4; the other noisier amplifiers (A, B, & C) show slightly larger increases. For reference, data taken through the D amplifier during its last few months on Side 1 had a read noise of approximately 4.4 e$^-$ pix^{-1} for gain=1 and 7.5 e$^-$ pix^{-1} for gain=4.

In this report, I will characterize the pattern noise seen during Side-2 operations, and demonstrate an algorithm that can be used, in some cases, to mitigate it. The code used for filtering the pattern noise is fairly simple, and feedback regarding more sophisticated methods of signal processing that may improve the quality of CCD data would be appreciated.

Analysis

To characterize the pattern noise in STIS CCD data, I looked primarily at calibration data, such as the daily bias measurements. I also looked at imaging and spectroscopy for a few General Observer programs. All of my analyses were done with **stisnoise.pro**, an IDL routine that transforms the 2-D CCD images into a 1-D time series and returns the Fast Fourier Transform (FFT). The 1-D time series is simply the concatenation of all pixels in a CCD image, in the order that they are read out, with appropriate time spacing to account for the serial and parallel clocking on the CCD. **stisnoise.pro** is available from STScI at ftp://ftp.stsci.edu/pub/instruments/stis/stisnoise.pro.

Bias Data

Figure 1 shows a section of a bias frame from the archive (o68se1010_raw.fits[sci,1]). Like all calibration data, these data are public. In panel *a*, the image is shown without filtering. The bright objects are cosmic-ray hits. Figure 2 shows a portion of the power spectrum for this bias image, in the vicinity of the peak from the pattern noise. The peak in the power spectrum is at 16.65 kHz, which corresponds to 2.73 pixels on the detector. The pattern appears to drift across the image (Figure 1*a*), creating a "herring bone", because the frequency of the pattern changes slowly with time, and because it is a pattern associated with the digital signal of the detector as it is read out (i.e., it is not a pattern fixed to specific pixels on the detector).

Figure 3 shows how the frequency of the pattern noise varies during 1000 bias exposures taken during Side-2 operations, as part of program 8903. The frequency of the pattern noise is correlated with the temperature of the CCD housing, but there is a lot of variation at any one temperature. Thus the frequency is probably dependent upon the tem-

perature of a component within STIS, and the temperature of that component tracks loosely with the temperature of the CCD housing. However, given the wide range in frequency at any CCD housing temperature, this temperature is not very useful for predicting the precise frequency of the pattern noise.
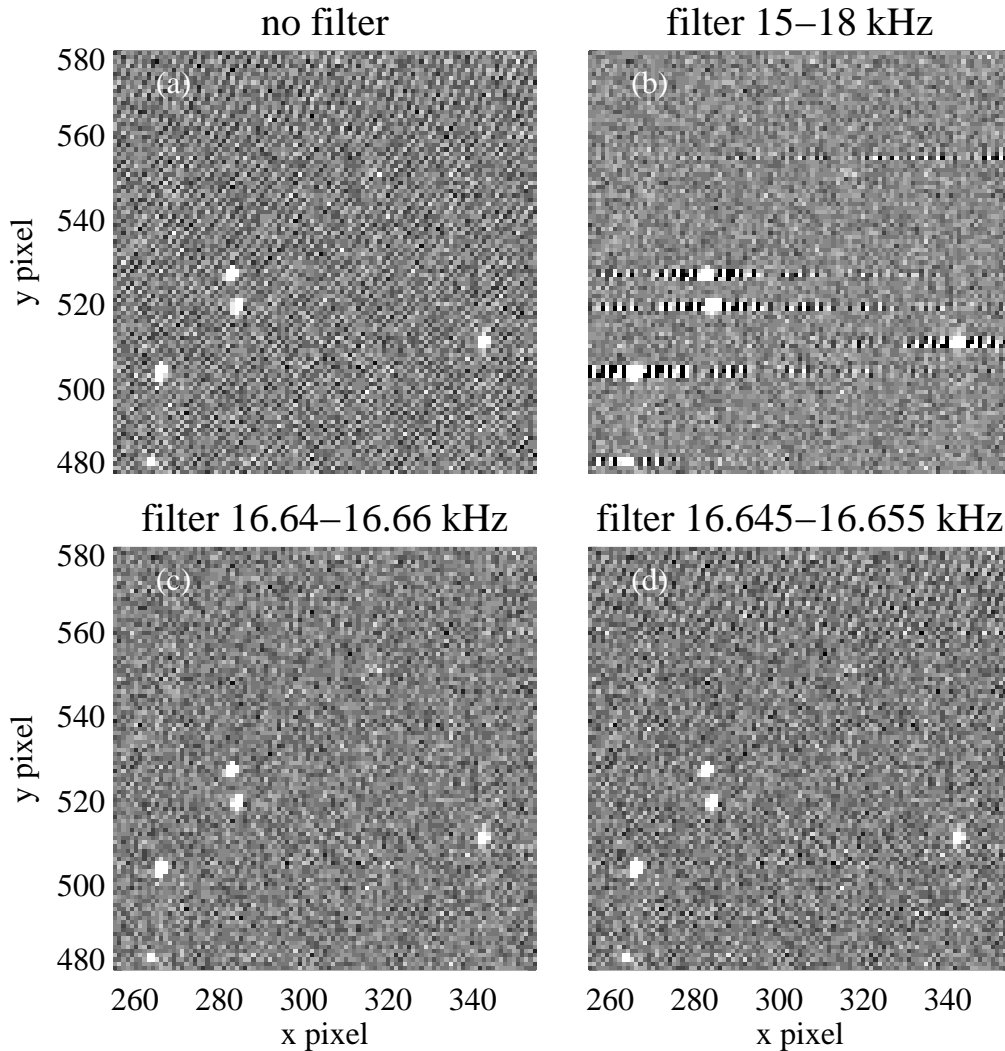

Figure 1: *a)* A section of a CCD bias frame from the archive. Note the obvious herringbone pattern in the read noise. *b)* The same image as in panel *a*, but filtered from 15 - 18 kHz. Because the filter is far too wide, it causes ringing around the objects (cosmic ray hits) in the image. *c)* The same image as in panel *a*, but with a narrow filter placed at precisely the frequency of the pattern noise. *d)* The same image as in panel *a*, but with a filter that is too narrow. Note that the pattern noise is still present in part of the panel.

    Properly filtering out this pattern requires a filter that is both narrow in frequency and well-centered on the frequency of the pattern. This can be seen in panels *b* through *d* of Figure 1. In panel *b*, frequencies from 15 to 18 kHz have been removed from the image, but such a wide filter causes "ringing" around the objects (cosmic rays) in the image. Thus, one cannot simply filter the whole range of frequencies possible for the pattern noise without introducing significant artifacts into the data. In panel *c*, a far more appropriate filter has been used to remove frequencies from 16.64 to 16.66 kHz in the image, with much better results. In panel *d*, a filter was used to remove frequencies from 16.645 to 16.655 kHz. Although the peak in the power spectrum is at 16.65 kHz, the frequency is not constant (hence the breadth to the peak in Figure 2), and some of the pattern noise escapes the filter.
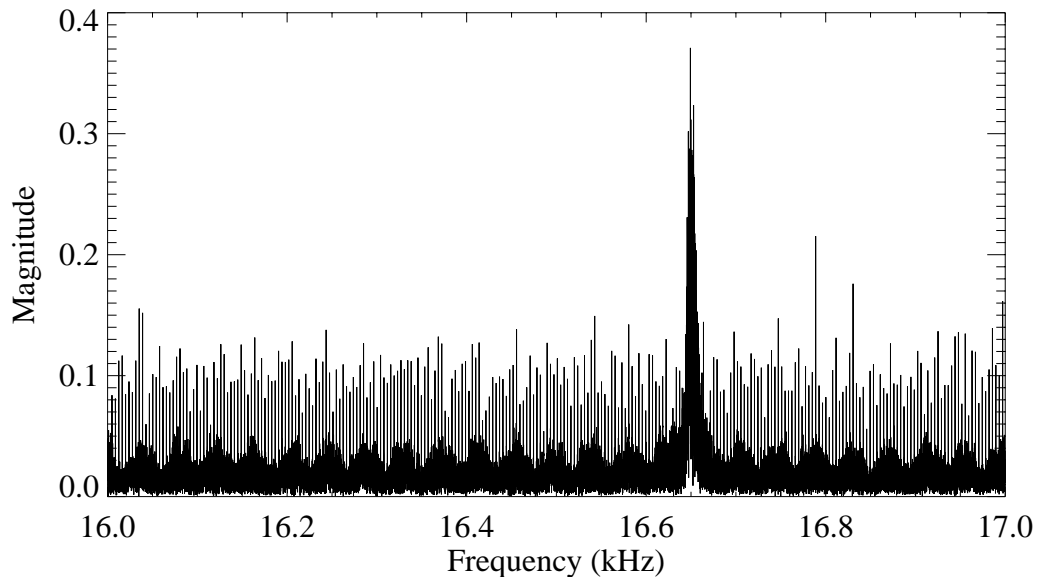

Figure 2: The power spectrum of the bias frame shown in Figure 1. The image was converted to a 1-D time series, and then Fourier transformed. The peak in the power spectrum, at 16.65 kHz, corresponds to a horizontal pattern with 2.73 pixel spacing on the detector.
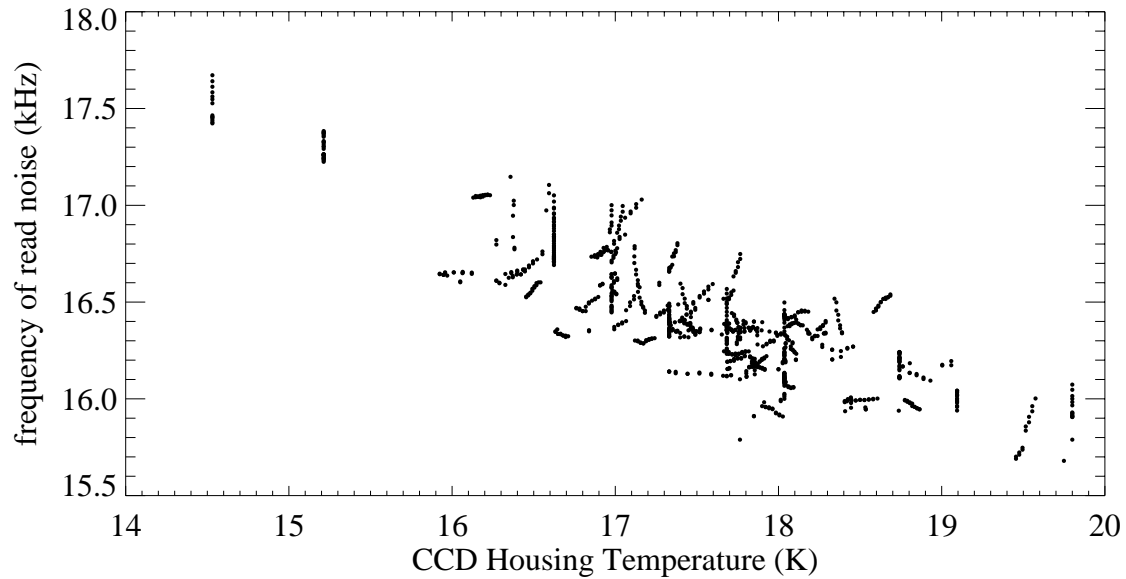
Figure 3: The frequency of the pattern noise versus housing temperature.

The filtering shown in Figure 1 does not only produce a qualitatively improved image (by eye) - it also *quantitatively* improves the noise characteristics of the image. The read noise in Figure 1*a* is 5.8 e$^-$ pix^{-1}, but the noise remaining in the image after appropriate filtering (Figure 1*c*) is only 4.3 e$^-$ pix^{-1}. The filtering has removed the additional pattern component to the read noise, leaving the white noise component that was prevalent in Side-1 operations.

Cosmic-Ray Rejected Images

The FFT analyses and filtering shown above were performed on raw bias frames; that is, these CCD images did not go through any of the pipeline processing, such as overscan subtraction, bias subtraction, or cosmic-ray rejection. It can be easier to apply these techniques to real data if they have gone through such pipeline processing, because the pattern may be seen more easily once cosmic-ray rejection has been performed. This is true even though cosmic-ray rejection produces the sum of several images, each with a unique frequency and phase for the pattern noise. An example of this can be seen in Figure 4, which shows the power spectra for data from program 8912. Panel *a* shows the power spectrum in the vicinity of the pattern noise frequency, for a raw (unprocessed) exposure of a star field in omega Cen; the pattern from the read noise is impossible to see in the white noise. The image analyzed in panel *a* is actually one part of a 5-exposure series. Panel *b* shows the power spectrum of the resulting image when this series of exposures is processed through the pipeline and combined with cosmic-ray rejection; the peak in the FFT is obvious. Figure 5 shows the cosmic-ray rejected image, with and without filtering of the

pattern noise. Generally, it is best to find and filter the pattern noise using the pipeline-processed data.
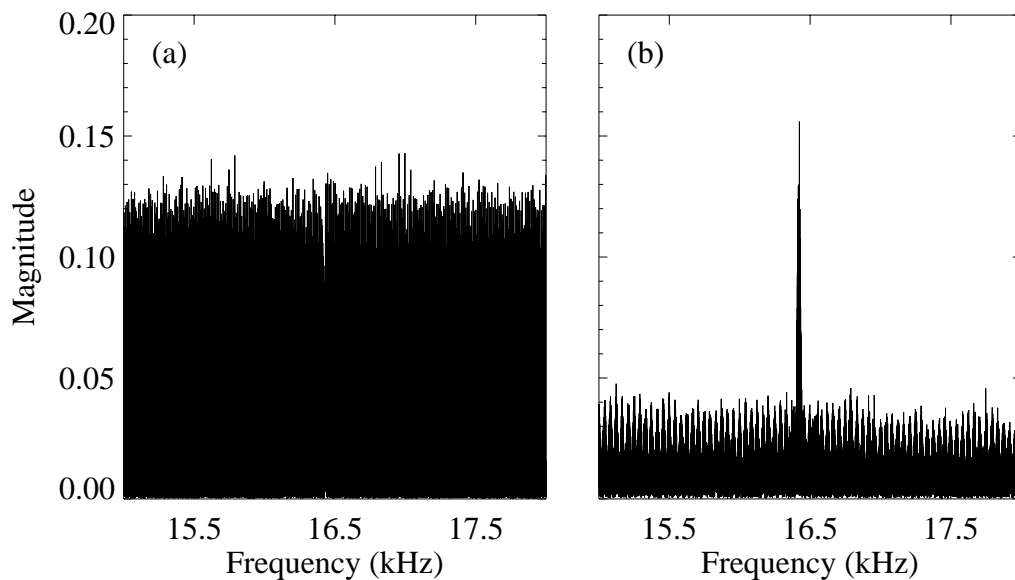

Figure 4: The power spectra of images from program 8912 (a star field in omega Cen). *a*) The power spectrum from the first raw exposure in a series of 5 exposures shows no obvious peak due to the pattern noise. *b*) The power spectrum of the pipeline-processed and cosmic-ray combined data shows an obvious peak from the pattern noise. This peak can be used to filter the cosmic-ray rejected image (Figure 5).
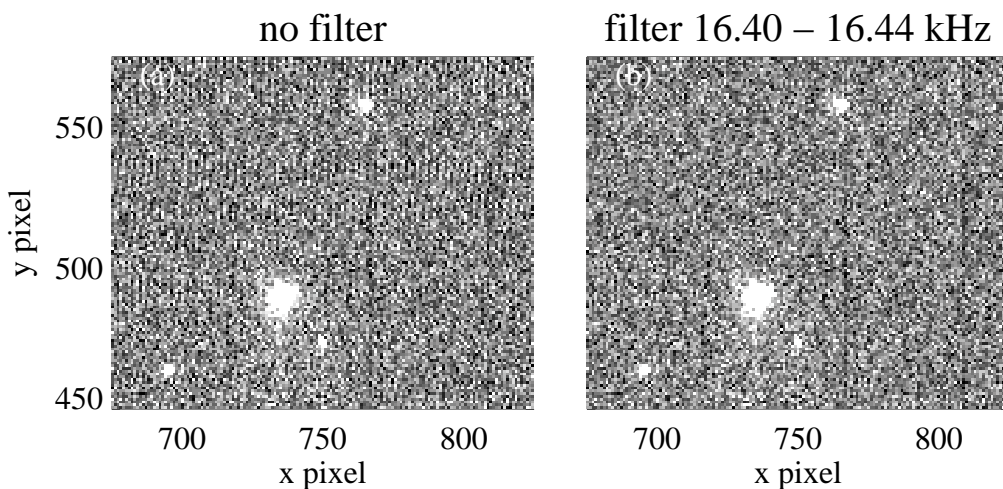

Figure 5: A section of the cosmic-ray rejected image of a star field in omega Cen. *a*) The data after pipeline processing but without filtering of the pattern noise. Note the herring-bone pattern. *b*) The data after applying a filter to block the frequencies shown in Figure 4.
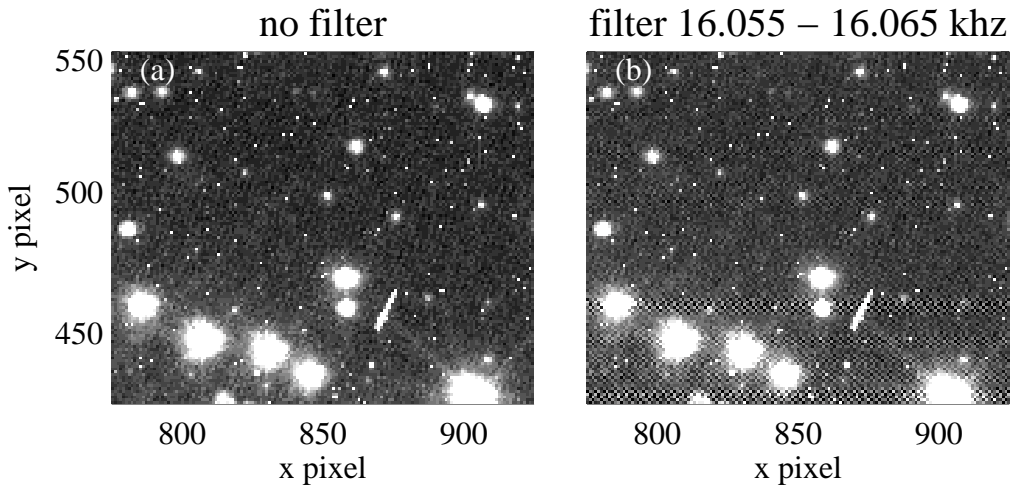
Figure 6: a) A section of a crowded image of 47 Tuc. b) The same section with a narrow filter applied. Notice the degradation of the image near the bright stars.

Although filtering can sometimes mitigate the pattern noise, it does not work well in instances of bright crowded fields; it is difficult to find the frequency of the pattern noise in the FFT in such cases, and even narrow frequency filters can introduce severe artifacts. This can be seen in Figure 6, which shows a crowded, bright star field in 47 Tuc (program 9266). Although the eye can pick out the pattern noise in the less exposed regions, FFTs of both the raw and the cosmic-ray rejected images do not reveal the pattern frequency. Hunting for the pattern frequency by trial and error, using narrow filters, produces severely degraded images. Of course, if filtering degrades the data only in certain sections of the filtered image, one could in principle replace these sections with the corresponding sections from the unfiltered image, or from an image filtered in a different manner.

Diffuse Targets

The most dramatic gains of filtering are made in short-exposure images of diffuse, faint, or sparsely populated fields. Figure 7 shows a 3.6 sec exposure of the galaxy UGC2847 (program 8591), with pipeline processing and cosmic-ray rejection (the PI of this program, D. Richstone, kindly allowed these data to be shown for demonstration in this report). The pattern noise is extremely obvious in this image. Fortunately, it is also very obvious in an FFT of the image, and thus a narrow filter can be tuned precisely, to remove the pattern from the data completely.
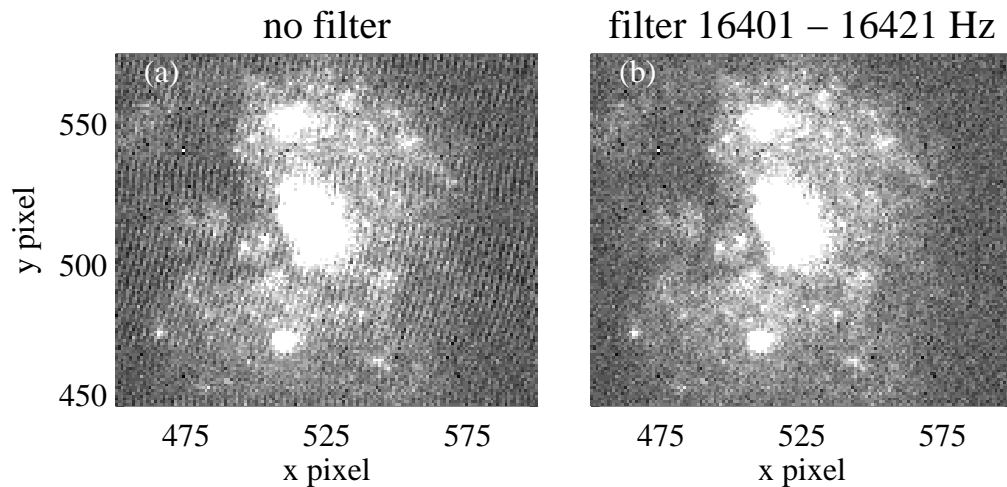
Figure 7: A section of a CCD image of UGC2847 (permission kindly given by D. Richstone). *a*) The pattern noise is very striking in this short exposure, and makes it difficult to see the galactic structure. *b*) Application of a narrow filter completely removes the pattern noise without introducing artifacts. Filtering works best in sparse, faint, diffuse images, and the gains are most obvious in short exposures where the dominant noise component is the read noise.

Spectral Images

Spectral images can also be filtered to reduce the pattern noise, prior to spectral extraction, with some success. Because the filtering affects horizontal patterns in STIS images, care must be taken to ensure that the filter does not compromise periodic features of the data (such as a series of absorption lines that repeats on small detector scales, which might appear in high-resolution spectroscopy). However, the large horizontal extent of "objects" in spectral images means that these data are less prone to filter artifacts - as far as the filter is concerned, the dispersed light is an extended object. This can be seen in Figure 8, which shows a G430L spectral image of NGC428 (from snapshot program 9070). The images were taken without a CR-SPLIT; instead, two exposures were taken with a small vertical dither, to allow removal of both hot pixels and cosmic rays. A section of the shifted and combined data is shown in Figure 8*a*. The power spectrum of this image is shown in Figure 9, suggesting a two-stage filter from 16.225-16.240 kHz and 16.255-16.270 kHz. The result of such filtering is shown in Figure 8*b*. To show the effects on the spectra, Figure 8*c* and Figure 8*d* show the counts in rows 888-891 for the spectral images in Figure 8*a* and Figure 8*b*, respectively. The spectra in both the filtered and non-filtered case are nearly identical, apart from lower noise in the filtered image.
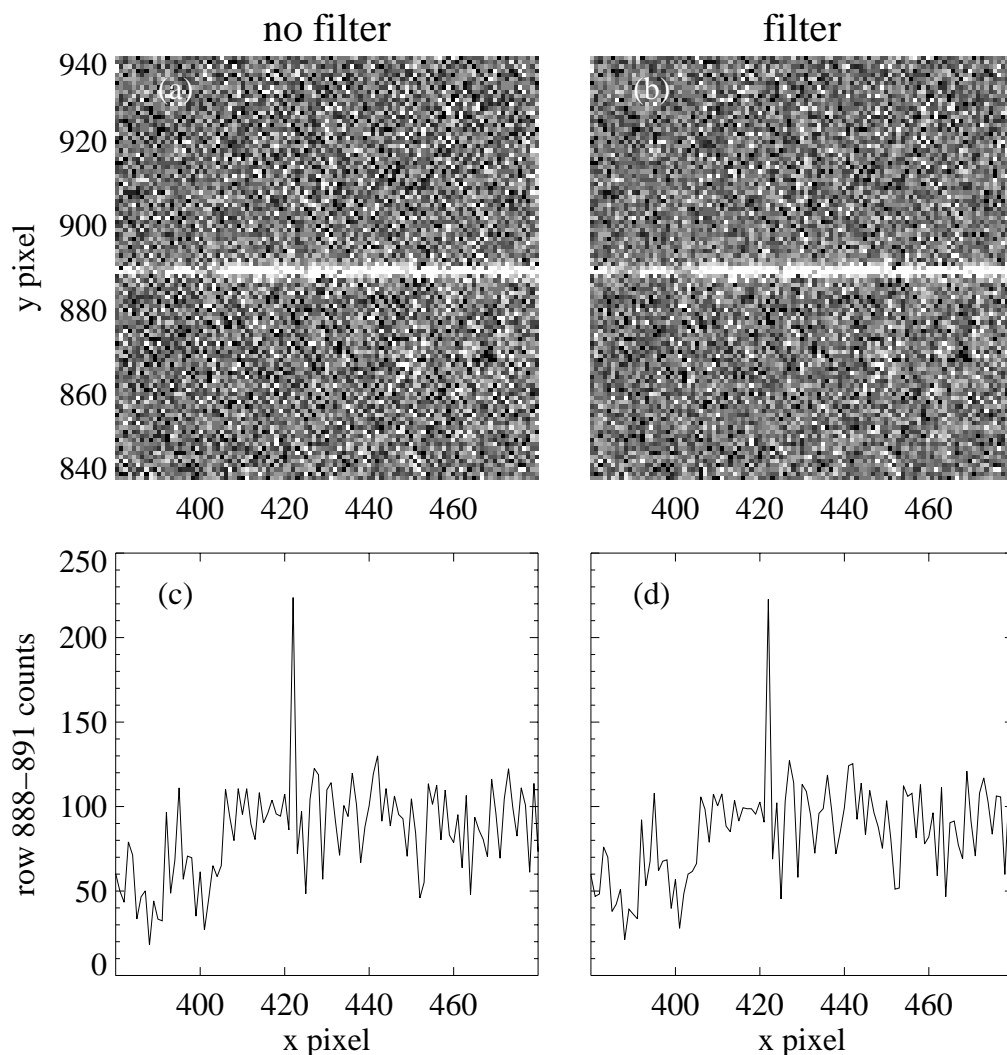
Figure 8: Sections of spectral images of NGC428 (program 9070), each with a row sum showing the effect on the one-dimensional spectrum. *a*) A section of the spectral image, without filtering. *b*) The same section as in *a*, with filtering of the noise shown in Figure 9. *c*) A row sum for the spectral image shown in *a*. *d*) A row sum for the spectral image shown in *b*.

Conclusion

STIS CCD images taken during Side-2 operations show an increase in read noise of about 1 e$^-$ pix^{-1}. This additional read noise appears in the readout at a frequency that varies from 15.5 to 18 kHz, producing a herring-bone pattern in CCD images. This pattern has a horizontal scale of approximately 3 pixels. The frequency of the pattern noise drifts from image to image and even within images, and it is apparently tied to the temperature of a component in the STIS electronics.

Depending upon their needs, users may wish to filter their CCD images to mitigate this increase in read noise. Filtering can produce both cosmetic gains (which might make images more appropriate for display) and a true decrease in the noise present in the data. However, this filtering must be done with a carefully tuned filter: The filter should be narrow in frequency (tens of Hz) and well-centered on the frequencies of the pattern noise. Even carefully tuned filters produce at least some artifacts in many STIS CCD images, so the use of a filter is usually a trade-off, and the appropriateness of filtering must be decided on a case-by-case basis (thus, such filtering will not likely be implemented in the pipeline).

Although simple frequency blocking in Fourier space can be implemented with a package written for this purpose (**stisnoise.pro**; see Appendix), STIS data could very well benefit from far more sophisticated filters available in digital signal processing software. Feedback on more advanced filtering techniques would be appreciated.
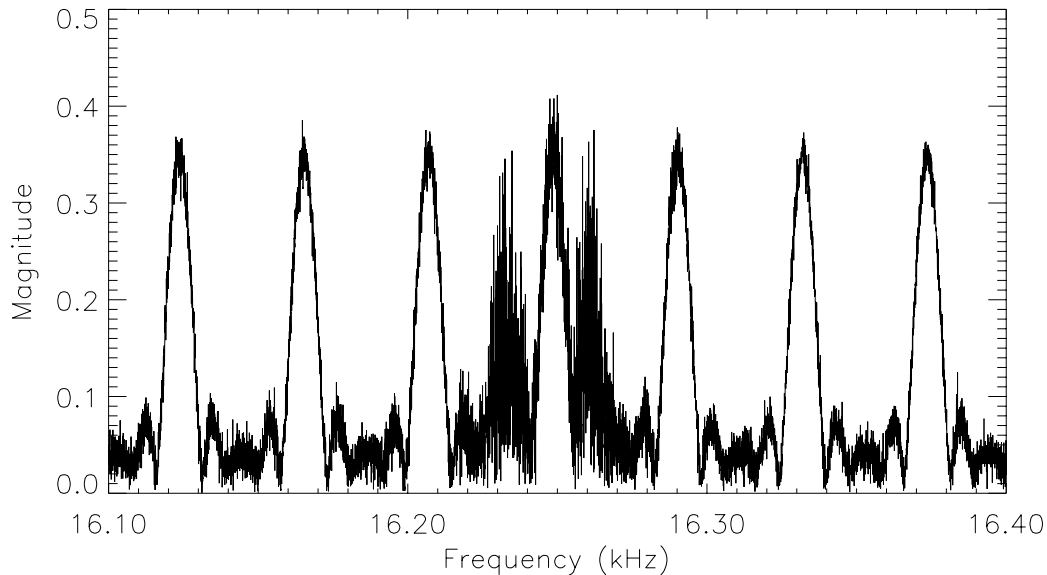

Figure 9: The power spectrum of the spectral image shown in Figure 8*a*. Because an FFT was performed on the sum of two images that were shifted before coaddition, the power spectrum shows a series of peaks (harmonics).The pattern noise produces two additional peaks in the power spectrum at 16.225-16.240 kHz and 16.255-16.270 kHz. Thus the pattern can be filtered by applying two filters in succession, blocking each of these frequency ranges.

Acknowledgements

APPENDIX: STISNOISE.PRO

The code I wrote for this analysis, **stisnoise.pro**, is written in Interactive Data Language (IDL), available from Research Systems, Incorporated (RSI). **stisnoise.pro** is available from STScI at ftp://ftp.stsci.edu/pub/instruments/stis/stisnoise.pro. The code calls upon routines built into IDL, and routines from the IDL Astronomy User's Library (e.g., **sxpar**, **readfits**, **writefits**). The IDL Astronomy User's Library is maintained by W. Landsman, and is available at http://idlastro.gsfc.nasa.gov/. The basic idea of the code comes from a program by W.J. McCann & G. Hartig, called **dino.pro**, used for analyzing CCD data from the Advanced Camera for Surveys.

The pattern noise seen in STIS 2-D CCD images is really a pattern in time, as the CCD is clocked out; the timing pattern then manifests itself as a spatial pattern in a reconstructed 2-D CCD image. Thus, to analyze the pattern noise, it is appropriate to construct the time series of the CCD signal from the CCD image. Each row is clocked along the y-axis of the detector (parallel shifts), and then read out sequentially along the x-axis (serial shifts); for the STIS CCD, the serial and parallel shift periods are 22 µs and 640 µs, respectively, and the directions of these shifts depend upon which amplifier is used for reading out the CCD. The time series is simply the concatenation of all rows on the CCD, with gaps representing the parallel shifts and any missing data (e.g., from overscan subtraction). If the full CCD is read (as opposed to a subarray), raw CCD images are 1062x1044 pixels (including 19 leading and 19 trailing pixels of physical overscan in each row, and 20 rows of virtual overscan), while images having undergone overscan subtraction and trimming (i.e., showing only the physical pixels on the detector that are illuminated) are 1024x1024 pixels.

After converting the CCD image (either raw or overscan-trimmed) to a time series, **stisnoise** performs a Fast Fourier Transform (FFT) on the time series, returning the frequency and magnitude of the power spectrum. If the frequency of the pattern noise can be discerned from the FFT, there are several options for filtering the signal to try and mitigate the pattern noise:

- median - this is just a simple median filter applied to the time series, with user-specified width (in pixels). E.g., to filter the data with a 3-pixel median (horizontally), type
  ```
  stisnoise,"infile.fits",freq,mag,median=3,outfile=outfile.fits".
  ```

- boxcar - a simple boxcar smoothing, with user-specified width (in pixels). E.g., to filter the data with a 3-pixel boxcar (horizontally), type
  ```
  stisnoise, "infile.fits",freq,mag,boxcar=3,outfile="outfile.fits".
  ```

- lee - a Lee filter. See the IDL help for **leefilt** for more details. The user specifies the width (in pixels) and the standard deviation. Note that the actual width used by the filter is 2N+1, where the user specifies N. E.g., to apply a 5-pixel wide filter with a standard deviation of 5, type
  ```
  stisnoise,"infile.fits",freq,mag,lee=[2,5],outfile="outfile.fits".
  ```

- digfilter - the digital_filter routine from IDL (see the IDL help for more details). The user specifies the filter boundaries (in Hz), the power relative to the Gibbs phenomenon (50 is a good choice), and the number of terms in the kernel. E.g., to apply a band stop from 16.3 kHz to 16.4 kHz using 101 terms, type
  ```
  stisnoise,"infile.fits",freq,mag,digfilter=[16400,16300,50,101],$
  outfile="outfile.fits".
  ```
  To apply a band pass in the same range, type
  ```
   stisnoise,"infile.fits",freq,mag,digfilter=[16300,16400,50,101],$
  outfile="outfile.fits".
  ```

- wipe - a square filter applied in frequency space. The user specifies the boundaries of the filter (in Hz) and the scale factor (from 0 to 1, and normally 0). The real and imaginary components of the power spectrum are rescaled by the scale factor, and the result is inverse-transformed. E.g., to remove all power from 16.3 kHz to 16.4 kHz in the FFT, type
  ```
  stisnoise,"infile.fits",freq,mag,wipe=[16300,16400,0],$
  outfile="outfile.fits".
  ```

- window - a tapered version of the wipe filter. The user specifies the center of the filter window (in Hz), the width of the filter window (in Hz), and the tapering of the filter edges (in Hz). E.g., to more gradually remove power from 16.3 kHz to 16.4 kHz, type
  ```
  stisnoise,"infile.fits",freq,mag,window=[16300,16400,10],$
  outfile="outfile.fits".
  ```

In the examples above, no FITS extension was specified for the input files, so the extension is assumed to be 1 by default (the first science exposure). If there are multiple science exposures, the extension keyword can be used to specify each exposure. Regardless of which extension is chosen in the input file, the output file is a FITS image with the primary header of the input image in extension 0, and the filtered image in extension 1 (with its extension header). Some subsequent processing may require that the FITS images be in the nominal STIS format (i.e., first SCI exposure in extension 1, the ERR and DQ for that exposure in extensions 2 and 3, the second SCI exposure in extension 4, etc.). To do this, the user can create a copy of the input file, `copy.fits`, then use **imreplace** in **IRAF** to set SCI extensions to 0, and then use **imcalc** to add the filtered images to the appropriate SCI extensions. E.g., assuming that `input.fits` is a STIS image, and `output.fits` is the output from **stisnoise.pro**, one could do the following:
```
cl>mscopy input.fits copy.fits
cl>imreplace copy.fits[SCI,1] 0
cl>imcalc copy.fits[SCI,1],output.fits[1] copy.fits[SCI,1] im1+im2
```
Note that this would assume keep the ERR and DQ extensions unchanged, even though the SCI extension has been filtered.

The frequency and magnitude returned by the routine will show the power spectrum with the filter applied, if a filter is chosen. Typically, **stisnoise** should be run once without filtering and output keywords, to see the power spectrum, and then run again, with a filter and an output file specified. Most users will find the *wipe* or *window* keywords the most

useful. All of the filtering applied in this report was done with the *wipe* subroutine, using the frequencies listed and a scale of 0.