A METHOD TO REMOVE FRINGES FROM IMAGES USING WAVELETS

P. ROJO[1,2] AND J. HARRINGTON[1,3],

ABSTRACT

We have developed a new method that uses wavelet analysis to remove interference fringe patterns from images. This method is particularly useful for flat fields in the common case where fringes vary between the calibration and object data. We analyze the efficacy of this method by creating fake flats with fictitious fringes and removing the fringes. We find that the method removes 90% of the fringe pattern if its amplitude is equal to the random noise level and 60% if the fringe amplitude is $\approx 1/10$ of the noise level. We also present examples using real flat-field frames. A routine written in IDL that implements this algorithm is available from the authors and as an attachment to this paper.

Subject headings: methods: data analysis — technique: image processing

1. INTRODUCTION

The current class of telescopes with primary mirrors larger than ~ 8 m in diameter allows researchers to attain an unprecedentedly high signal-to-noise ratios. In addition, ever-increasing computer capabilities have permitted quantitative analyses able to distinguish trends weaker than the noise level. This has not only allowed observations of fainter objects, but also observations of weak sources spatially indistinguishable from a bright source, such as the spectrum of an extrasolar planet orbiting a main-sequence star. In such cases, systematic errors that would not have been of importance when analyzing the bright source are of concern when considering the fainter source. Previously ignored systematic errors thus require algorithms able to correct them.

One such systematic effect is the appearance of fringes in data arrays. The strength of these fringes varies from instrument to instrument; we have seen it range from less than 1/10 of the noise amplitude to roughly 5 times the noise amplitude.

Fringe correction methods found in the literature are either specific to the instrument or assume a global fringe period (e.g., Malumuth et al. 2003a,b; Mellau & Winnewisser 1995). The latter correction type makes uses of Fourier filtering, a technique that is less than satisfactory in the common case where the pattern's period or amplitude varies over the image. Another common approach has simply been to ignore the fringes in the hope that division of the debiased data array by the flat field frame will eliminate the pattern, which is supposed to remain the same. However, flat fields and object frames do not always share the same fringe pattern because flexure and variations in the illumination geometry can change the pattern's amplitude or period even on short timescales (Figs. 1 and 2). Therefore, flat-field division could add a second set of fringes rather than correct the first set.

Here we present an algorithm to clean two-dimensional (2D) arrays that uses the wavelet transform, a local spectral technique (e.g., Starck & Murtagh 2002; Torrence & Compo 1998[4]). Making use of the wavelet transform's linearity prop-
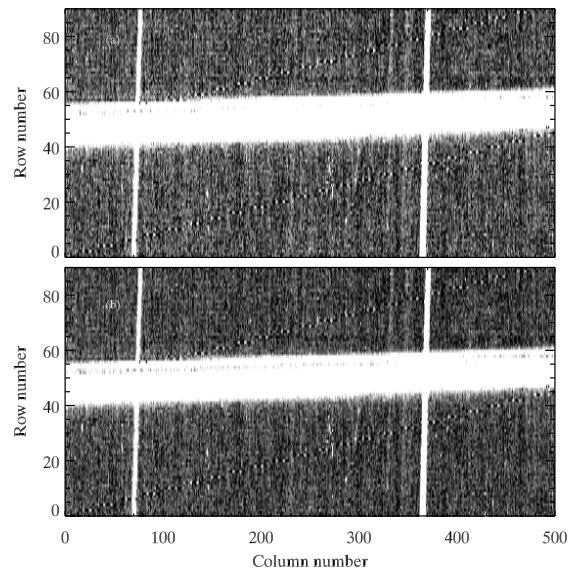

FIG. 1.— Failure of flat field to correct fringes. *Top*: Debiased spectrum frame. *Bottom*: Spectrum frame after flat field and bias correction. The fringe pattern is still visible. These images are subsections of a spectrum frame obtained with NIRSPEC at the Keck telescope. The horizontal white pattern is the spectrum's trace. The bright, white, vertical lines are the sky emission lines.

erty, the algorithm isolates the fringe pattern in wavelet space, does an inverse transform, and then obtains a clean image by subtracting the reconstructed fringe pattern. The challenge is to do this correctly in the presence of noise. The algorithm presented here is not tuned to any specific data set and has been tested on flat-field frames from ISAAC (Moorwood et al. 1998) at the Very Large Telescope (VLT) and NIRSPEC (McLean et al. 1998) at the Keck telescope.

The algorithm also provides the framework for an extension to remove fringes from object frames by interpolating the fringe pattern over spectra or point sources. Such an extension requires the design of another algorithm to interpolate fringe parameters, which is beyond the scope of the present work. Thus, our method improves the quality of extracted data when the fringes in the flat fields differ from the fringes in the data, but will not solve the problem completely if the fringes in the data array are significant. We show that there are many cases where the fringes change between arrays obtained at different times.

[1] Center for Radiophysics and Space Research, Cornell University, Ithaca, NY 14853-6801 USA; pato@das.uchile.cl, jh@physics.ucf.edu
[2] Departamento de Astronomía, Universidad de Chile, Casilla 36-D, Santiago, Chile
[3] *current address:* Department of Physics, University of Central Florida, Orlando, FL 32816-2385
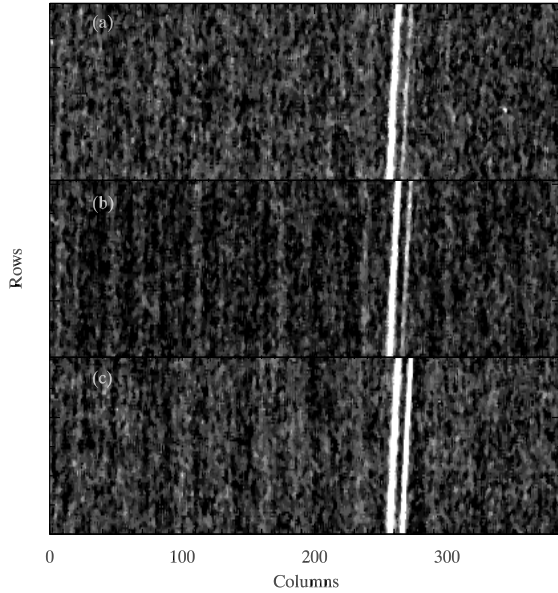[4] See http://atoc.colorado.edu/research/wavelets/

FIG. 2.— Fringe variation in consecutive frames. Shown are subsections of sky frames taken with NIRSPEC on the Keck II telescope (45 s integration). The fringe pattern can be seen varying between these frames. Panels a, b, and c were taken in consecutive order.

We implemented our algorithm in IDL (a product of Research Systems, Inc., Boulder, Colorado). The package (defringeflat) includes tutorial documentation. It is available under the GNU General Public License from our Web sites[5] and as a tar file in the electronic edition of this paper.

Section 2 gives a mathematical model of fringe formation. Section 3 describes the algorithm. Section 4 discusses performance in the presence of noise. Finally, §5 discusses an applied example, summarizes the benefits and limitations, and presents our conclusions.

2. FRINGES

Fringes are produced by the interference of light reflecting between parallel surfaces in an instrument. They appear in many detectors of visible and infrared light. If we ignore multiple reflections, a mathematical formulation (Rieke 2003) for the total intensity of light (I_a) received on the position x, y of the detector array is given by

$$I_a(x,y) = I_n(x,y) + I_r(x,y) + 2\sqrt{I_n(x,y)I_r(x,y)}\cos\psi(x,y), \quad (1)$$

where I_n is the non-reflected intensity, I_r is the reflected intensity, and ψ is the phase difference between the two beams. We choose the x coordinate such that

$$\psi(x,y) = 2\pi x/P(x,y) + \xi(x,y), \quad (2)$$

where P and ξ are the period and phase of the fringe's pattern, respectively. Let I_i be the incoming intensity before interaction with the instrument. Then, $I_r(x,y)$ is proportional to the intensity incident at a nearby position:

$$I_r(x,y) = a(x+\delta x, y+\delta y)I_i(x+\delta x, y+\delta y), \quad (3)$$

where δx and δy are small displacements and the factor a includes reflectivity. Note also that $a \ll 1$. If we can assume that the incoming intensity field and the reflection geometry are homogeneous on very short spatial scales, then $a(x+\delta x, y+\delta y)I_i(x+\delta x, y+\delta y) \approx a(x,y)I_i(x,y)$. On the other

hand, the non-reflected intensity (I_n) is proportional to the incoming intensity (I_i) in the same coordinate, thus $I_n(x,y) = b(x,y)I_i(x,y)$. Due to energy conservation, the proportionality constant $b(x,y)$ is restricted by the previous assumption to comply with

$$a(x,y) + b(x,y) = s(x,y), \quad (4)$$

where $s(x,y) \lesssim 1$ accounts for losses due to scattering and absorption. Omitting the dependence on x and y for clarity, we find for each position

$$I_a = bI_i + aI_i + 2I_i\sqrt{ab}\cos\psi \quad (5)$$

$$= I_i\left[s + 2\sqrt{ab}\cos\left(2\pi x/P + \xi\right)\right] \quad (6)$$

$$= I_i[s + F], \quad (7)$$

where

$$F = 2\sqrt{ab}\cos\left(2\pi x/P + \xi\right) \quad (8)$$

is the oscillating fringe term. When interacting with the detector array, Eq. 7 is modulated to obtain the detected intensity I_d. Including detection noise, the modulation is given by

$$I_d = I_i\eta[s + F] + \epsilon, \quad (9)$$

where ϵ, which varies rapidly between pixels, includes all noise sources and η includes quantum efficiency, and pixel collecting area, among other factors. On the other hand, η can be decomposed as

$$\eta = \eta_0 + \eta_r, \quad (10)$$

where η_0 is the smoothly varying component and η_r is the rapidly varying component, which includes uncorrelated differences between the sensitivities of neighboring pixels. Typically, $\eta_r \ll \eta_0$. Bringing it all together, we obtain

$$I_d = I_i\eta s + \epsilon + I_i\eta_r F + I_i\eta_0 F. \quad (11)$$

Our algorithm makes use of the linearity property of wavelets to find and subtract the term $I_i\eta_0 F$, which is the predominant contributor at the period of the fringe pattern. The other terms will only contribute in that frequency to a background level in the amplitude of the wavelet transform. This background is considered in our algorithm (see step 2 in §3).

Then, ηs can be corrected through flat-fielding to get the sought intensity I_i with a modified noise $\bar{\epsilon}$ given by

$$\bar{\epsilon} = \epsilon + I_i\eta_r F. \quad (12)$$

With typical values, $I_i\eta_r F \ll \epsilon$.

3. ALGORITHM

The main steps in our procedure are listed in Table 1. Figures 3 − 9 illustrate the steps of the algorithm using an example flat field. Their captions contain details regarding the example array, while the main text only refers to the algorithm in general. The example flat field is included in the defringeflat package.

All array borders whose values are not consistent with the image must be cropped. The fringes are allowed to have several different patterns, which do not need to look like straight lines. There are only two requirements. First, the period (P) of the fringe term (eq. [8]) should change smoothly across the array; and second, only on a per-row basis, the period must be at least several pixels, but it must also have at least a few oscillations per row. To attain the second condition it is acceptable to rotate the image by $90°$. There are no constraints on how the phase ξ can change across rows or the range over which P can vary. Hence, the algorithm can handle many patterns that do not look like plane waves, such as patterns resembling wood grain.

TABLE 1
STEPS OF THE DEFRINGING ALGORITHM

Step	Description	Figure
0...	Original image with fringe	Fig. 3
1...	*For each row*	
	Compute enhanced row	Fig. 4
	Compute wavelet transform	Fig. 5
2...	*For each pixel in row*	
	Fit fringe transform's profile	Fig 6
3...	*For the whole array*	
	Smooth fit parameters (optional)	Fig. 7
4...	*For each row*	
	Reconstruct wavelet array	Fig. 5
	Inverse transform	Figs. 5, 8
	For the whole array	
	Subtract fringe pattern to obtain clean image	Fig. 9
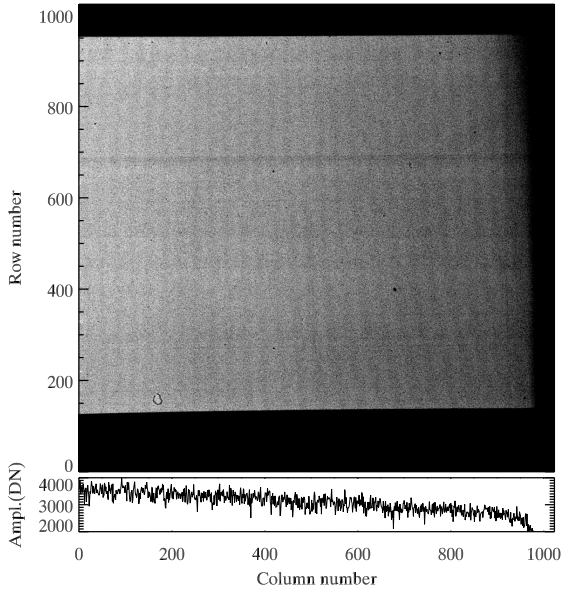
FIG. 3.— *Top*: Sample image with fringes. This flat field was obtained with the ISAAC instrument at the VLT. Each of the numeric parameters indicated in the captions from Figs. 3 to 9 were found to be the most appropriate for this particular example, but will need to change for different images. Columns 901–1024 and rows 0–149 and 951–1024 were vignetted and thus were cropped before analysis. Periodicity can be estimated by eye at ~ 40 pixels in the center of the image. *Bottom*: Middle (512th) row.

Step 1: Enhanced Row and Wavelet Transform

For each image row we combine several surrounding rows to suppress random noise and remove bad pixels. To do this, we replace each pixel in the row with the median of a $1 \times n$ sub-image centered on the pixel and traversing n rows (the bin width). We then subtract a polynomial fit from the median-averaged row to obtain an "enhanced row" (Fig. 4). This subtraction significantly diminishes the large-period (low-frequency) oscillations of each row (and their corresponding wavelet amplitudes), allowing the next step to proceed more efficiently.

We then compute the wavelet transform of each enhanced row. The result for each row is a two-dimensional, complex array, whose two dimensions are column number and period. There are several real and complex wavelet bases to choose from, but step 2 of this algorithm requires a complex basis because real bases are not able to separate phase from amplitude
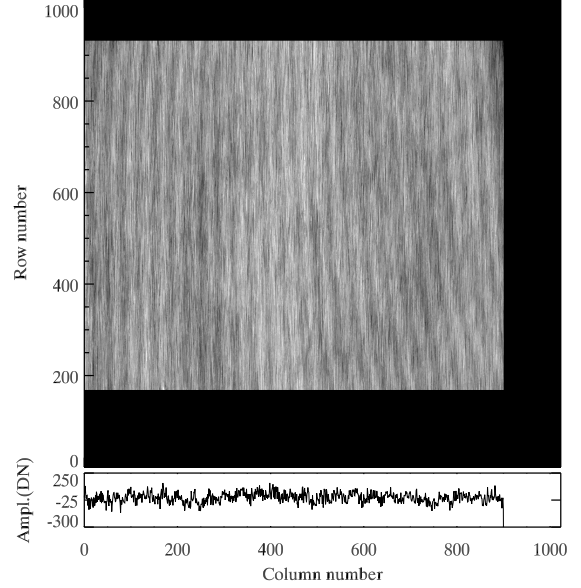

FIG. 4.— *Top*: Enhanced rows. Each pixel of the array in Fig. 3 is first replaced by the median average of the 41 closest pixels in the vertical direction. A polynomial fit to each row is then subtracted. The fringe pattern is enhanced and some bad pixels are removed. Note that the usable data area is reduced by 20 rows on the top and bottom because of the averaging. *Bottom*: Middle (512th) row.

information. For this particular example, we used the Morlet wavelet because its functional form is the familiar quantum-mechanical wave packet

$$\Psi(t) = \pi^{-1/4} e^{-t^2/2 - i\omega_0 t}, \tag{13}$$

which makes it well suited for smoothly varying periods. Here, t and ω_0 are non-dimensional. For the Morlet basis, ω_0 is the only parameter; it dictates the minimum number of oscillations per row. The Morlet basis also has the advantage of being compact in the frequency domain.

In addition, the accompanying code allows the user to choose from several other popular wavelets as they could be better suited for particular data. Steps 2 and 3 are computed over the complex array amplitudes (the wavelet array). The phases of the complex array must be stored for use in step 4.

Step 2: Parametric Fit of Fringe Transform

At the period of the fringe pattern, the wavelet array will contain a prominent "fringe transform" pattern traversing the columns. Its amplitude depends on the amplitude of the fringe pattern (Fig. 5). This algorithm's success rests on our ability to distinguish this feature from the background noise level of the wavelet array. The fringe transform may vanish for particular columns, but it should be clearly distinguishable in most of each wavelet's array. Improved sampling in period can be obtained by interpolation or by decreasing the spacing between discrete scales in the wavelet transform. A compromise should be chosen; the latter approach is more accurate but demands more computer resources.

We next either extract or fit the fringe transform's amplitude versus period for a given column (Fig. 6). Starting from a reference column, the fringe profile is isolated by finding the first local minima on both sides of the reference period. Then one method is chosen to represent the profile within the minima: either we use the actual data within the minima (the "trueshape" method) or a parametric function can be fitted to
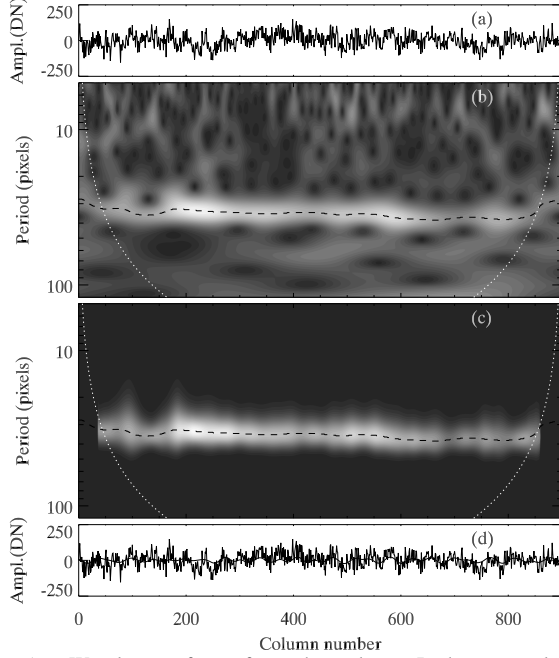
FIG. 5.— Wavelet transform of an enhanced row. In the center plots, the dotted line marks the COI; wavelet values below this boundary should not be trusted. The dashed line shows the fitted trace. (*a*) Middle enhanced row of Fig. 4. (*b*) Amplitude of the Morlet wavelet transform of (*a*). The wavelet is interpolated in period by a spline from the period sampling of the transform, and the fringe transform, a coherent pattern corresponding to a fringe with a period of 35 pixels, is clearly visible. (*c*) Reconstructed fringe transform using a Gaussian fit (cf. Fig. 6). (*d*) Fringe pattern after applying an inverse wavelet transform to (*c*), plotted over the input data.
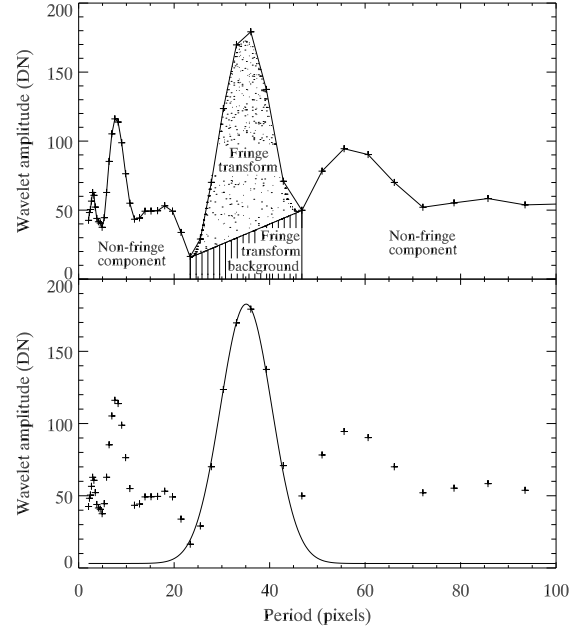

FIG. 6.— Cross–section along a column of the wavelet array. *Top*: Components diagram. Crosses are the amplitude of the wavelet points. The data points surrounding the region labeled as "fringe transform" are exactly what the trueshape method would reconstruct or what the parametric functions will fit. The region labeled "fringe transform background" is attributable to non-fringe components; in some of the tests it was used for the wavelet reconstruction (keep) and in some it was not (nokeep). *Bottom*: Gaussian fit (with background) to the region between the minima. Crosses are as above, and the solid line is the fitted profile.

the profile. Only the latter approach will allow execution of the optional step 3. The value of the profile must be zero outside the fringe transform. Inside, on the other hand, it is recommended that the fringe transform profile exclude a background level (attributable to non-fringe image components, see discussion in §2). The highest point in the profile is used as the new reference period for the next column. The procedure is repeated for the whole fringe transform, extending in both directions from the reference column to the cone of influence (COI) boundary, beyond which the wavelet values are significantly contaminated by edge effects.

To fit the profile we have experimented with plain Gaussian fits with variable center (Gaussian with Variable Center, "GVC") and Gaussian functions in which the center is fixed at the maximum height (Gaussian with Fixed Center, "GFC"). Both Gaussian alternatives were considered without a constant background parameter (*noback*), and with this parameter. In the latter case, the background value can be kept or not when reconstructing ("keep" and "nokeep" respectively). In total, we have implemented six parametric fitting methods (that can be smoothed or not in step 3) and two trueshape fits (nokeep and keep), for a total of 14 fitting methods. The Gaussian shape was chosen not only because it is a natural choice to fit a peak, but also because it is the frequency-domain representation of the Morlet wavelet. The relative fringe-removal efficiency of these fits and of trueshape is discussed in §4.

Step 3: Optional Parameter Smoothing

If a functional parametric fit was used in the previous step, one can reduce the effects of noise by forcing the reconstructed fringe's parameters to vary smoothly. After repeating steps 1 and 2 for every row, a 2D array is obtained for each of

the fitted parameters. First, we "patch" each of the parameter arrays by finding outliers beyond a given number of standard deviations from the neighborhood median and replacing them by that median value. Then, we smooth the array with a boxcar filter. Figure 7 shows an example.

Step 4: Reconstruction of the Fringe Pattern

We next evaluate the parameters to obtain the fringe's wavelet amplitudes (Fig. 5). Far from the reconstructed fringe transform the amplitude must be zero because any non-zero value there will cause unwanted noise in the reconstructed fringe. In particular, if a keep method is chosen, the reconstructed amplitude is set to zero outside the local minima. Finally, we apply an inverse wavelet transform to the reconstructed wavelet amplitude and the corresponding complex phases (see step 1).

We repeat these steps for every row to obtain the image's isolated-fringe pattern (Fig. 8). Due to the optional smoothing, the method to obtain the enhanced rows, and the COI boundary, the recovered-fringe pattern will have smaller borders than the original image. The fringe pattern can now be subtracted from the original image (Fig. 9). Figure 10 shows another example of this algorithm for a flat field from NIRSPEC at Keck.

4. PERFORMANCE TESTS

The ratio of fringe-pattern amplitude to the pixel-to-pixel variation (or noise) level varies among different instruments. We tested the algorithm's performance at different noise levels by using a synthetic image consisting of a fringe pattern, a background intensity, and random noise with a Gaussian distribution that mimics pixel-to-pixel flat-field variations and photon noise.

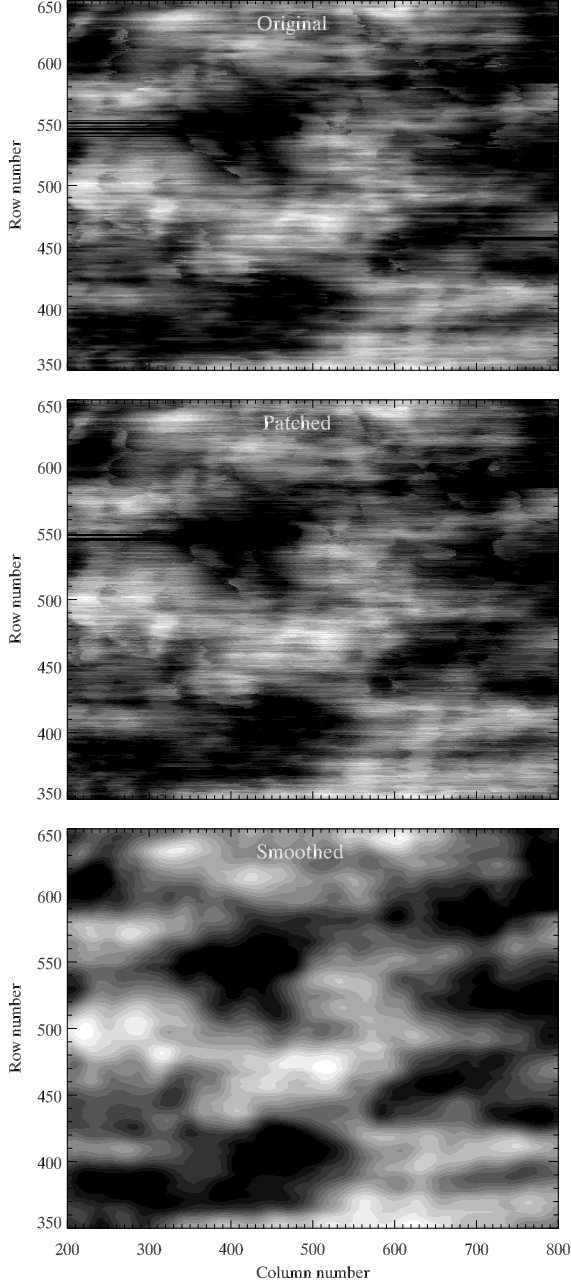FIG. 8.— *Top*: Reconstructed fringe pattern (cf. Fig. 5, *bottom*). *Bottom*: Middle (512th) row.

FIG. 7.— Gaussian height parameter smoothing. *Top*: Gaussian fit height parameter for central portion of example image. *Middle*: Parameter after replacing all values more than $\pm 1.5\sigma$ from the local median level with that level (patched array). *Bottom*: Patched array after smoothing with a 19 pixel boxcar filter. This procedure is repeated for each of the other Gaussian fit parameters.

FIG. 9.— *Top*: Cleaned image: flat field of Fig. 3 minus the fringe pattern of Fig. 8. Note that some of the edges remain uncorrected (see text). *Bottom*: Middle (512th) row.

The fringe pattern was created using an analytic function that mimics the oscillating pattern in our example image. Its functional form is

$$F(x,y) = A\sin(\nu(x,y)x + \phi(y)), \qquad (14)$$

where x and y are the position indices in the array, $\phi(\ldots)$ and $\nu(\ldots)$ are linear functions fit to the phase and frequency, respectively, of our example's fringe pattern, and A is the amplitude. For these tests we keep the amplitude constant, but there is no reason for A to be constant in a real image, nor is there any reason for a non-constant amplitude to adversely affect our algorithm. The background level is a double-linear function in both the x- and y-directions and has an edge taper.
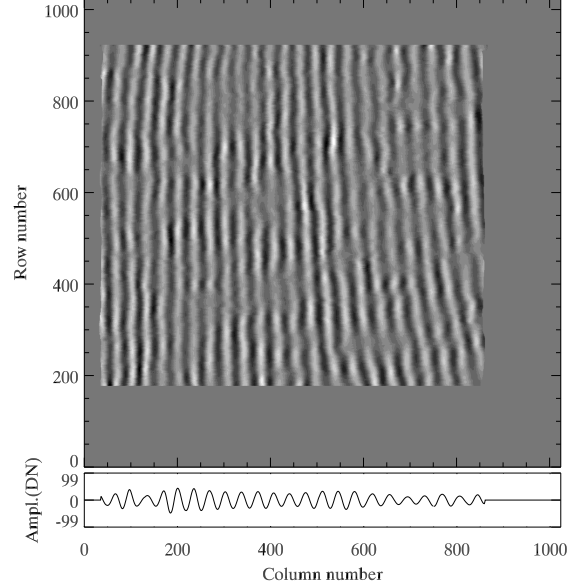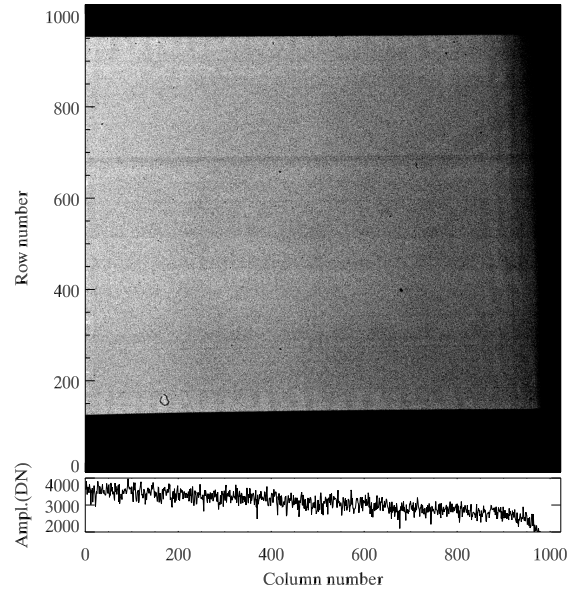
We define noise strength as the standard deviation of the Gaussian noise divided by the standard deviation of the noiseless fringe pattern ($2^{-1/2}A$, due to its sinusoidal nature). Figure 11 shows the fraction of remnant fringe after running the algorithm on simulated data with different fitting functions and varying noise strength. The remnant fringe level is not strongly dependent on noise strength and all methods show very similar behavior with slight numerical differences when the noise strength is below ≈ 8. However, GFC consistently gives the best results in all cases, even improving when smoothing at high noise levels. Most of the methods remove over 95% of the fringe at noise strength of ≈ 0.5 and over 55% at a noise strength of ≈ 9 (equivalent to Fig. 3's noise strength). The plot in Figure 11 (bottom) confirms the intuitive result that the method yields better absolute results for
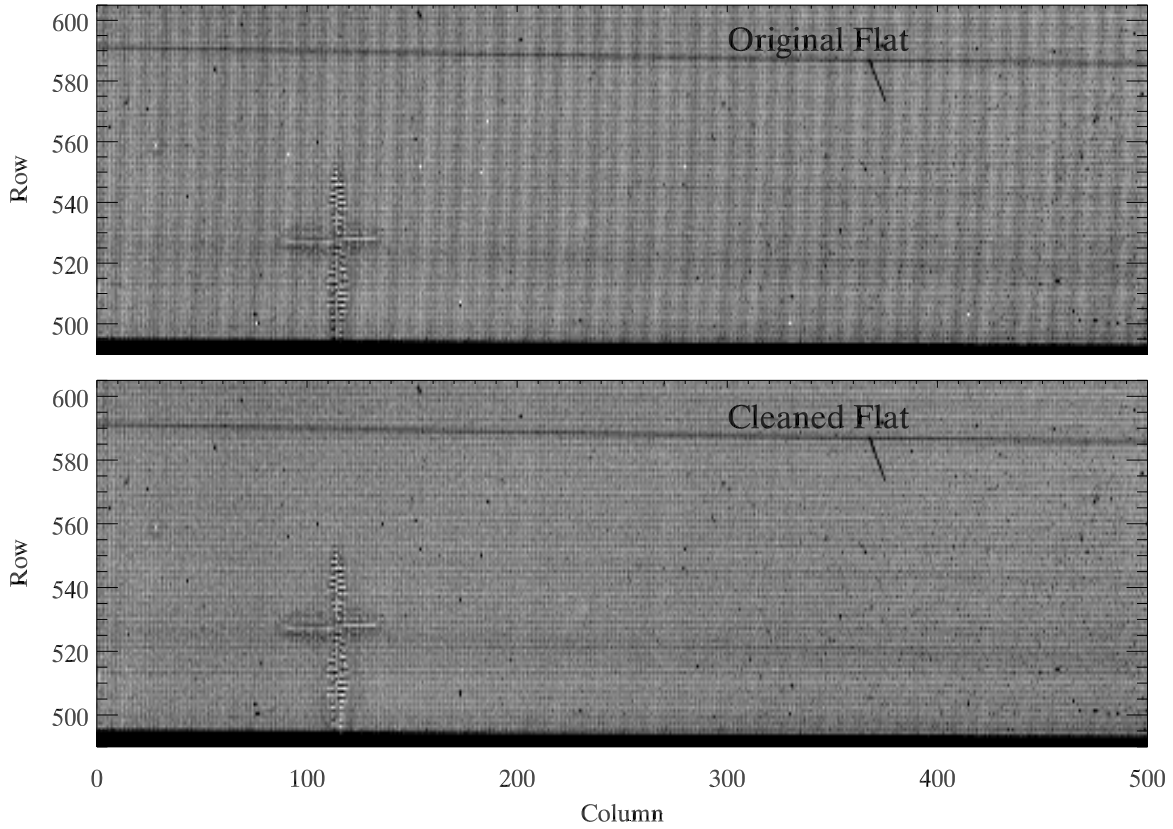
FIG. 10.— Example of fringe removal from a second instrument. The frames present a portion of a debiased flat-field frame from one order of the high-resolution NIRSPEC spectrograph at the Keck II telescope. *Top*: Original flat field. *Bottom*: Same flat field defringed by our method. The algorithm had to be applied twice; once for a fringe of period ~ 20 pixels and then for a fringe of period ~ 40 pixels.
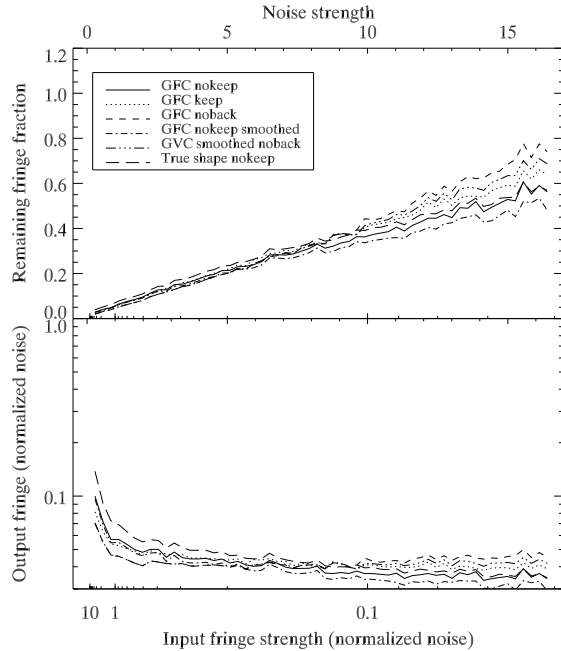

FIG. 11.— Remaining fringe for varying noise strengths. Only 6 of the 14 methods are shown above, for clarity. Omitted methods are similar to plotted methods and fall within the range of traces shown. The two nokeep methods give the best results. In general, all methods give similar results for low noise strength, but smoothed nokeep methods are better for high noise strength. *Top*: Fraction of fringe remaining. *Bottom*: Absolute fringe remaining when noise level is scaled to 1.
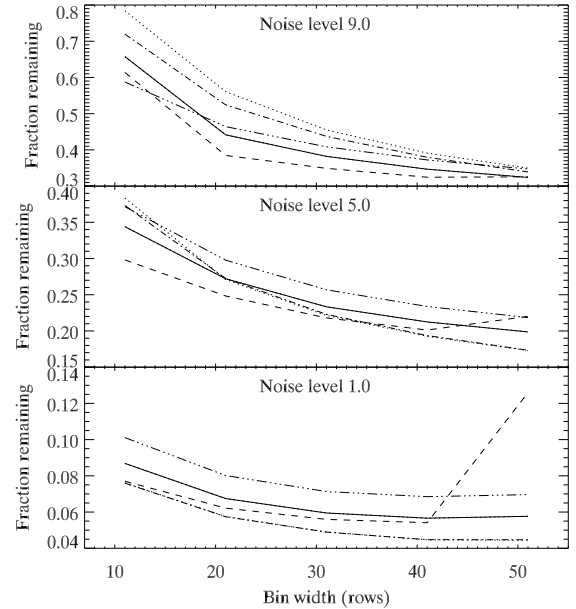

FIG. 12.— Remaining fringe for different enhanced-row bin widths. The panels show the same synthetic fringe pattern as Fig. 11 at three selected noise levels. Line styles are the same as in Fig. 11.

smaller initial fringe amplitudes.

Figure 12 shows the effect of varying the bin width. If the width is too small when computing the enhanced row, the noise is insufficiently suppressed. For low noise, a bin width that is too large will begin to average out the fringe.

The algorithm is limited by the degree to which the analytic profile fitting function mimics the data. Figure 13 shows an example of a difficult profile, which gives very different fits
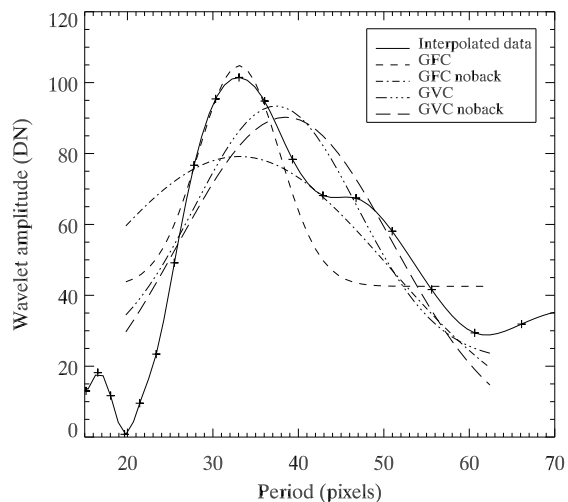
FIG. 13.— Different fitting methods applied to a profile with a complicated shape. Such shapes are due to noise and are the main limiting factor for this algorithm. This profile comes from the fringe transform for row 798, column 627 of our example array. Crosses show the data points, while the solid line is the interpolated profile. Other lines are explained in the key. The profiles are only fitted within the local minima at both sides of the reference period.
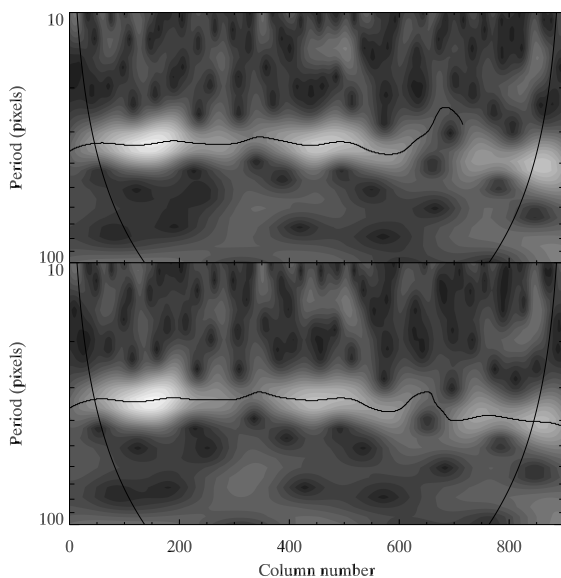

FIG. 14.— Example of missed trace. *Top*: Wavelet array from row 798 of Fig. 4. Around column 650 the trace goes in the wrong direction, towards a short period, and disappears around column 720. *Bottom*: Wavelet array from row 799 of Fig. 4. The array is similar to the top plot, but now the trace is correct through the last column. Patching in step 3 is likely to correct cases like this.

unimportant when analyzing only the bright source are of concern when considering the fainter source. Hence, those errors must be reduced either in the instrument design or in the data analysis. To that effect, we have developed a general algorithm to remove fringe patterns from imaging data such as flat-field frames while preserving other patterns. Cleaning flat fields is especially useful when the fringe pattern varies between them and the object data.

Consider the particular example of trying to detect the spectral modulation of an extrasolar planet as it transits its star using an instrument like ISAAC at the VLT. The equivalent noise strength for a flat field of this instrument is ≈ 9. On the other hand, no fringe was detected on the object frames up to a level equivalent to a noise strength of ≈ 1.5. Hence, according to Figure 11, removing the fringe in the flat fields through our method would reduce the systematic noise in the data frame by at least 40%. Considering the flat-field intensity, this translates into residual noise in the data frame $\approx 0.25\%$ of the intensity of the star. A typical molecular spectral variation is still below that level, of order 10^{-4} times the stellar intensity. However, it will now be easier to use the constancy of the planetary signal over many frames to attain the required sensitivity.

There are three main limitations of this algorithm when applied to a flat field. First, the shape of a fringe in wavelet space may be much more complicated than any reasonable fitting function, resulting in a partially corrected fringe. Second, to be able to follow the trace, the change in the fringe's period must be smooth. Finally, there is a region along the borders where the fringe pattern cannot be recovered.

The algorithm could be improved by finding a parameter-space interpolation mechanism that would allow defringing of object frames. Also, a method could be found to fit the entire fringe transform pattern simultaneously in the 3D wavelet space of row, column, and period. The 2D wavelet transform may be more appropriate for this approach.

Our IDL implementation of this algorithm and its documentation appear as an electronic supplement to this article. Updated versions are available on our Web sites or by e-mail request.

when using the different fitting functions. Another source of error is the potential for the algorithm to miss the correct trace in the presence of high noise in the wavelet array (Fig. 14). Also, the reconstructed fringe pattern is smaller than the input data due to the factors listed in step 4. For the example of Figure 3, this area is $\approx 85\%$ of the cropped input image, or over 90% if only considering the pixels lost for each surviving row, on average.

5. DISCUSSION AND CONCLUSIONS

Seeking a signal from a faint source that is spatially indistinguishable from a bright source is a long-standing observational challenge. Systematic errors that would have been

REFERENCES

Malumuth, E. M., Hill, R. J., Cheng, E. S., Cottingham, D. A., Wen, Y., Johnson, S. D., & Hill, R. S. 2003a, in Proceedings of the SPIE., Vol. 4854, 567

Malumuth, E. M., Hill, R. S., Gull, T., Woodgate, B. E., Bowers, C. W., Kimble, R. A., Lindler, D., Plait, P., & Blouke, M. 2003b, PASP, 115, 218

McLean, I. S., Becklin, E. E., Bendiksen, O., Brims, G., Canfield, J., Figer, D. F., Graham, J. R., Hare, J., Lacayanga, F., Larkin, J. E., Larson, S. B., Levenson, N., Magnone, N., Teplitz, H., & Wong, W. 1998, in Proc. SPIE Vol. 3354, p. 566-578, Infrared Astronomical Instrumentation, Albert M. Fowler; Ed., ed. A. M. Fowler, 566

Mellau, G. C. & Winnewisser, B. P. 1995, in ASP Conf. Ser. 81: Laboratory and Astronomical High Resolution Spectra, 138

Moorwood, A., Cuby, J.-G., Biereichel, P., Brynnel, J., Delabre, B., Devillard, N., van Dijsseldonk, A., Finger, G., Gemperlein, H., Gilmozzi, R., Herlin, T., Huster, G., Knudstrup, J., Lidman, C., Lizon, J.-L., Mehrgan, H., Meyer, M., Nicolini, G., Petr, M., Spyromilio, J., & Stegmeier, J. 1998, The Messenger, 94, 7

Rieke, G. H. 2003, Detection of light: from the ultraviolet to the submillimeter (Detection of light: from the ultraviolet to the submillimeter, by G.H. Rieke. 2nd ed. Cambridge, UK: Cambridge University Press, 2003)

Starck, J.-L. & Murtagh, F. 2002, Astronomical image and data analysis (Berlin: Springer)

Torrence, C. & Compo, G. P. 1998, Bull. of the Am. Meteor. Soc., 79, 61