Optimal Compression of Floating-Point FITS Images

W. D. Pence, NASA/GSFC
R. L. White, STScI
R. Seaman, NOAO

Abstract: Lossless compression (e.g., with GZIP) of floating-point format astronomical FITS images is ineffective and typically only reduces the file size by 10% to 30%. We describe a much better compression method that is supported by CFITSIO and the publicly available *fpack* and *funpack* FITS image compression utilities that can compress floating point images by a factor of ~10 without loss of significant astrometric or photometric precision. This technique uses the FITS Tiled Image Compression convention to quantize the floating point image pixel values into scaled integers which are then compressed with the Rice algorithm. The addition of a new "subtractive dithering" technique is described which permits more coarse quantization (and thus higher compression) than was possible with the previous simple linear scaling method.

1. Why don't floating-point images compress well?

Typical floating point format astronomical FITS images only compress by 10% to 30% with lossless compression algorithms like GZIP because many of the mantissa bits in each pixel value are not significant and just contain incompressible random noise.

2. Noise must be removed for better compression.

The noisy mantissa bits can be deleted from the image pixels by quantizing the floating point values into linearly scaled integers:

FloatValue = ScaleFactor * IntegerValue + ZeroPoint

ScaleFactor is numerically equal to the spacing between the discreet intensity levels in the quantized image. The goal is to choose the largest possible value of ScaleFactor to eliminate as much noise as possible while still preserving the required amount of astrometric and photometric information in the image.

3. How is ScaleFactor calculated?

The ScaleFactor is calculated to be a user-specified fraction, q, of the measured Gaussian sigma of the noise in the background regions of the image. For example, if the background sigma = 25 and q = 16, then the ScaleFactor, and the spacing between the quantized levels, will be 25/16 = 1.56 intensity units. Note that this quantization is only an issue for the faint pixels because the CCD A-to-D converter greatly over samples the inherent statistical noise in the brighter pixels.

4. How does q relate to the compression ratio?

The value of q directly determines the resulting image compression ratio, R, via this formula:

$$R = BITPIX / (\log2(q) + 1.8 + K)$$

where K depends on the compression algorithm and is about 1.1 for the Rice algorithm. This relationship is shown in the following table:

q	16	8	4	2	1	0.5
R	4.6	5.4	6.5	8.2	11.0	16.8

5. The need for dithering the pixel intensities.

If all the image pixels are quantized onto the same grid of intensity levels then the derived sky background level will tend to be biased towards the nearest quantized level (e. g., if one takes the median of the pixel values). This can affect the photometry of faint objects in the image when using a coarse quantization grid. This bias can be greatly reduced by randomly dithering the zero point of the quantized levels on a pixel by pixel basis. We use a technique called "subtractive dithering" in which a random value between 0 – 1 is added to the pixel value when scaling it to an integer, and then the same value is subtracted when rescaling back to the quantized floating point value. This randomizes the pixel values without actually adding any noise to the image.

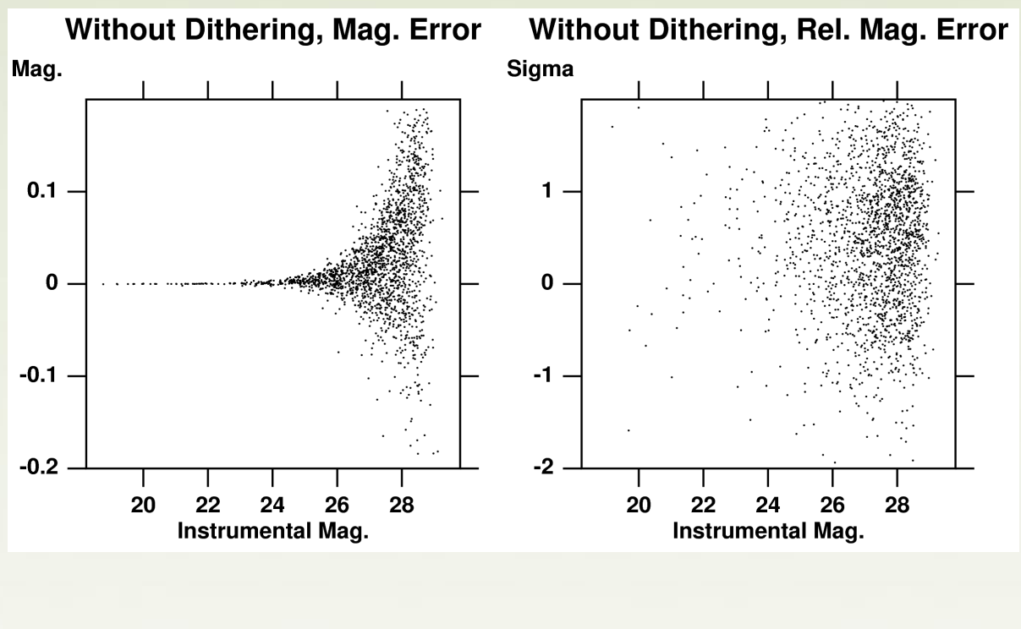

Fig. 1 – The magnitude errors (left) and relative mag. errors (right) in a q = 1 highly quantized image, <u>without</u> dithering. The errors have a significant positive bias and relatively large scatter because of bias in the sky background estimate.

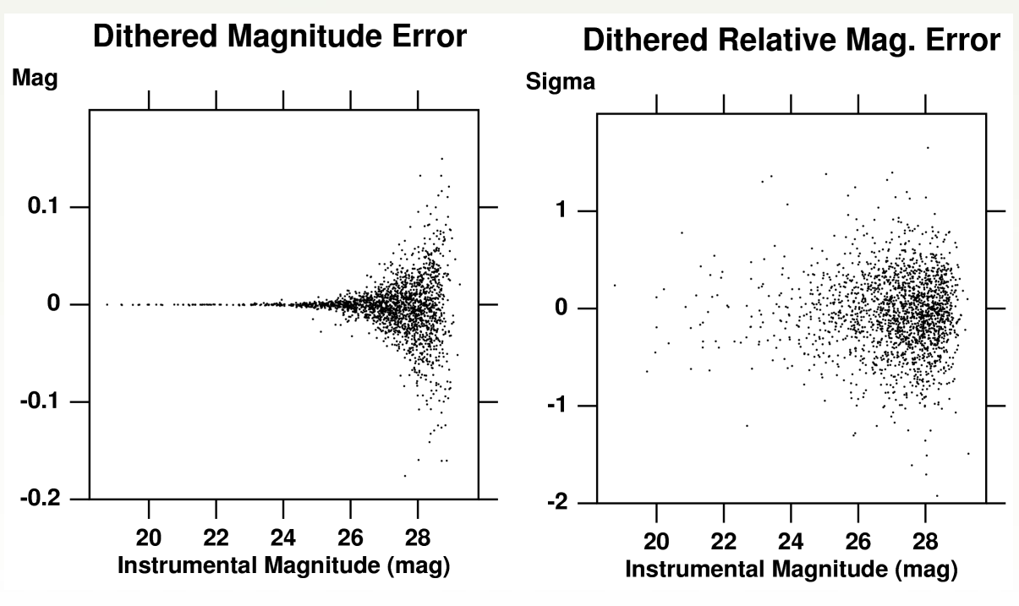

Fig. 2 – Same as Fig. 1, but after applying subtractive dithering during the quantization process. The systematic bias is gone and the scatter is reduced.

Fig. 3 - Central ½ of the Steward CCD image

FITS Tiled Image Compression Convention

- Image is divided into a rectangular grid of tiles
  - default is row-by-row tiling
- Each tile is separately compressed
  - Rice is the default algorithm
- Compressed bytes are stored in a variable length column of a FITS binary table
- In the case of floating point images, the pixels are scaled to integer values before compression
- See the FITS Support Office Web site for details

6. Software Implementation

We used the *fpack* and *funpack* FITS file compression utility programs for this project. These utilities call the CFITSIO library routines which transparently compresses or uncompresses the FITS images using the tiled image compression convention. The latest release of CFITSIO (v3.21) supports the subtractive dithering method described here.

Fpack and *funpack* are publicly available at
http://heasarc.gsfc.nasa.gov/fitsio/fpack/

7. Experimental Procedure

We used 2 representative floating point astronomical CCD images, one from the NOAO Deep Wide-Field Survey and one taken at the prime focus of the Steward Obs. 2.3m telescope (Fig. 3). The *fpack* utility program was used to quantize and compress each image, with q values ranging from 16 to 1, and with corresponding compression ratios ranging from 4.6 to 11. *funpack* was then used to uncompress the quantized image back into a standard FITS format image for further analysis with Sextractor, since it cannot directly read images in the tile-compressed FITS format.

The well known Sextractor (Source Extractor) program was used to generate a catalog of positions and magnitudes of all the sources in the original image and in each of the quantized images. We then compared the residual difference in the positions and magnitudes as a function of the q quantization factor of the compressed image.

8. Results: (Similar results were obtained for both of the sample images but only 1 is shown here).

Figure 4, below, shows the difference in the position of the centroid of each object (in pixel units) as measured by Sextractor in the original image and in each of the quantized images, plotted as a function of the magnitude of the object. The differences are less than a few hundredths of a pixel even in the most highly quantized q = 1 image (which has a compression ratio of 11).

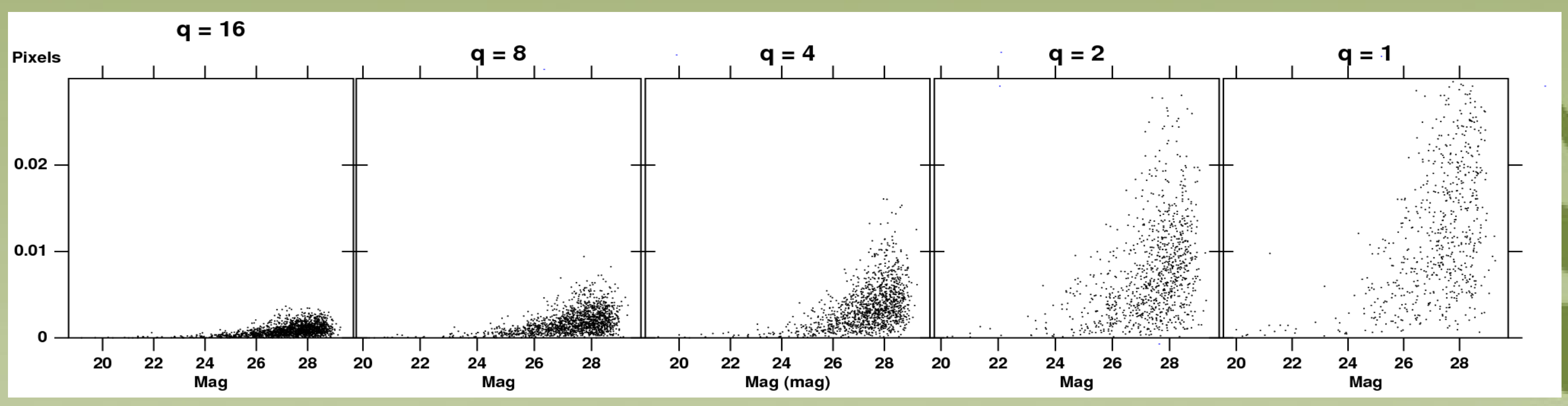

Fig. 4

Figure 5, below, shows the relative magnitude difference (the magnitude difference between the original image and each quantized image, divided by the 1-sigma error on the magnitude measurement) integrated within 12" apertures. The differences increase as the image is more coarsely quantized, but even in the q = 1 image the magnitude differences are still less than 1 sigma.

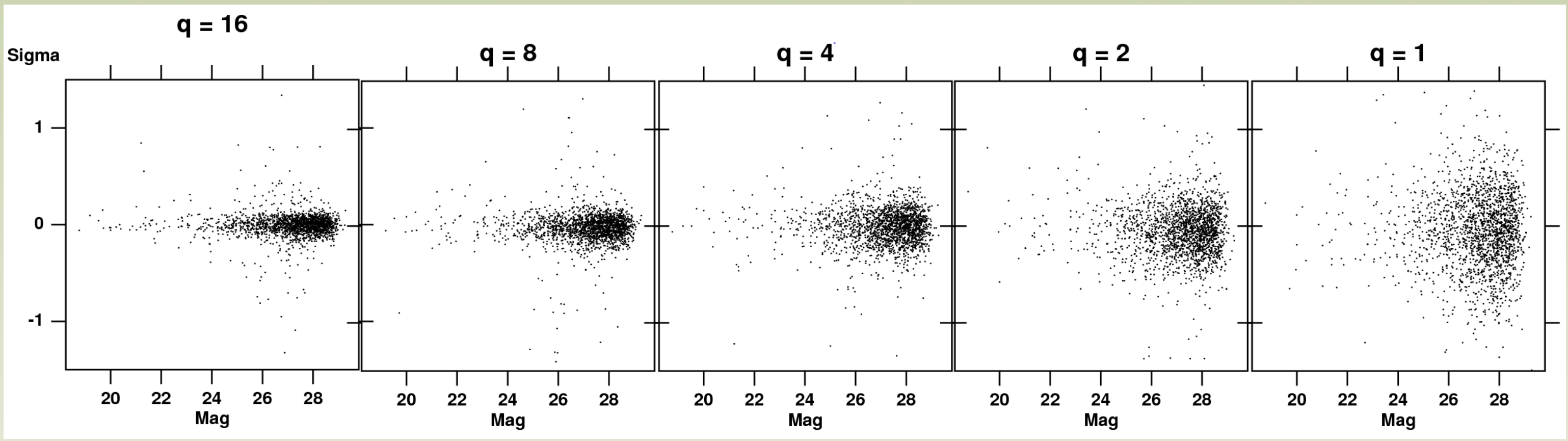

Fig. 5

9. Conclusions

This study demonstrates that typical floating point CCD images similar to those obtained by astronomical survey cameras can be compressed up to a factor of ~10 without significant loss of information. This technique quantizes the pixel values into scaled integers which can then be efficiently compressed using the Rice algorithm. A "subtractive dithering" technique is used to reduce the bias in the sky background level in the quantized image, which would otherwise affect the photometry of faint objects. When using a relatively coarse q = 1 quantization grid, which gives an image compression ratio of 11, the measured positional errors on the objects in the image were less than about 0.03 pixels, and the photometric errors were less than 1 sigma. Adopting a more conservative quantization factor of q = 4 would reduce the positional differences to less than 0.01 pixel and the magnitude differences to less than 0.3 sigma while still producing an image compression ratio of 6.5.

These results do not necessarily apply to other types of images, so users are urged to do experiments using the publicly available *fpack* and *funpack* utility programs to determine the minimum q level that is suitable for their particular data set and scientific application.

天文データの圧縮