Astronomical Data Analysis Software and Systems IX
ASP Conference Series, Vol. 216, 2000
N. Manset, C. Veillet, and D. Crabtree, eds.

Compression of Mosaic CCD Images with CompFITS2

S. Gaudet, J.-P. Véran, & D. Delisle

NRC Canada, Herzberg Institute of Astrophysics, 5071 West Saanich Rd., Victoria B.C., Canada V8X 4M6

B. Pirenne

Space Telescope European Coordinating Facility, ESO, Karl-Schwarzschild-Str. 2, D-85748 Garching bei München, Germany

Abstract. The advent of large CCD mosaics is creating a challenge for archive centers. With nightly production rates in excess of 20 GB/night, it is necessary to have efficient methods to reduce the size of the data on archive media while preserving the information content. We report here on recent improvements to the compFITS (Véran & Wright 1994) non-lossy astronomical data compression method that will allow to efficiently address this challenge.

1. Introduction

The advent of large CCD mosaics is creating a challenge for archive centers. With nightly production rates in excess of 20 GB/night, it is necessary to have efficient methods to reduce the size of the data on archive media while preserving the information content.

We report here on recent improvements to the compFITS (Véran and Wright 1994) non-lossy astronomical data compression method that will allow to efficiently address this challenge. The method used in compFITS2 (see Figure 1) consists of splitting artificially the bit-planes of the (integer) pixels in two parts: one containing the bit-planes with large, noise-like pixel-to-pixel variations (the least significant bits - LSBs) and the other containing the bit-planes with reasonable entropy (the most significant bits - MSBs). The actual optimal partition is determined by analysis of a subset of the original image. Since current popular non-lossy compression programs such as *compress* or *gzip* are good at compressing the MSBs, but not the LSBs, it is then more efficient to pass to these programs only the compressible MSBs. To compress the MSBs, compFITS2 can use any compression program as a plug-in. The result is a valid FITS file — compFITS2 stores the compressed and uncompressed data for each extension in a binary table extension, thus preserving the primary and extension headers in a readable form.

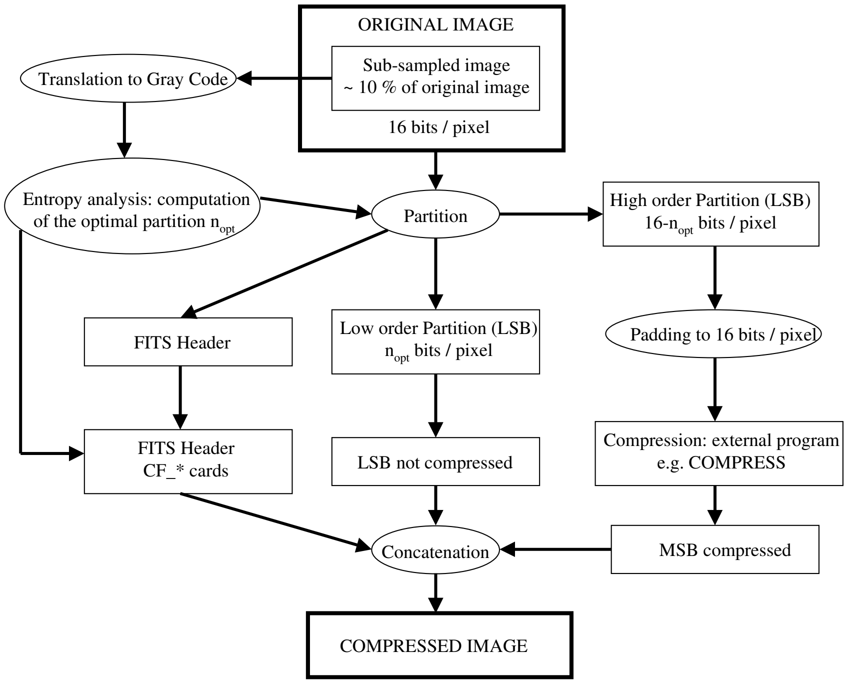

Figure 1. CompFITS2 flowchart.

2. What's New

- Support for multiple image extension FITS files, such as is typical for CCD mosaics;
- *Plug-in* capability for any compression program that takes data on stdin and writes to stdout (e.g. *compact*, *compress* or *gzip*) to compress the MSB part of the image;
- A new sampling algorithm to determine the LSB-MSB threshold;
- The selective decompression of a specific extension;
- Extraction of the MSB part of an image for preview purposes;
- Using FITS binary table extensions where the compressed MSB and the LSB are stored in a one row, two column table. This allows the preservation of the original image extension header while keeping the final result a valid FITS file;
- Using cfitsio (Pence 1992) for reading and writing both the original file and the compressed file;
- Addition of CF_* keywords to the primary and extension headers to fully describe the compression and the decompression required. This was missing in the original implementation so that one had to know which compression program was used in order to be able to decompress the file. These keywords are removed when decompressing the file.

3. Testing of CompFITS2

3.1. Test 1: Non-lossy Compression

The obvious first step in testing was to verify that compFITS2 did indeed preserve content through the compression/decompression process. A simple Unix *cmp* of the original and decompressed files does not always work because of syntactic differences if the original header did not conform entirely to the FITS standard. This is because the cfitsio library produces conforming FITS format files, correcting any non-conforming header cards from the original file. Therefore the original and decompressed files were compared using a set of IRAF tasks:

- Each primary and extension header was compared using *hdiff*;
- Each image was compared by subtracting the original from the decompressed image (*imarith*) and checking that the result was zero (*imstat*).

3.2. Test 2: Relative Performance

Tests were run to characterize whether running compfits2 has introduced a time penalty or provides a significant gain over other methods. Because most compression programs do not support multi-extension FITS format, we ran the tests on a large number of CFHT 8K Camera images (2kx4k detector, 1 image per file, 16.38 MB per file). We tested the original compFITS, a variety of non-lossy compression programs (*compact*, *compress*, *gzip*, *bzip2*) and compFITS2 using the same programs as plug-ins. The results are summarized in Table 1. Results for *gzip* and *bzip2* are not included because they were consistently 10 times slower.

Table 1. Comparison tests on single image files.

Compression Program	Number of Samples	Original Size (MB)	Compr. Size (MB)	Percent of Original	Compr. Time (Sec.)	Decomp. Time (Sec.)
CompFITS2 with compress	1784	16.38	7.67	46.8%	7.60	5.05
Compress only	1784	16.38	9.95	60.7%	6.97	4.27
CompFITS2 with compact	877	16.38	7.13	43.5%	7.77	5.11
Compact only	877	16.38	8.91	54.4%	6.60	3.59
CompFITS (original)	769	16.38	6.97	42.5%	7.84	5.86

3.3. Test 3: Performance on Large CCD Mosaics

The final tests were to determine performance on large multi-extension FITS image files. We chose to use the CFHT 12K Mosaic Camera files (2kx4k detector, 12 images per file, 196.64 MB per file). The tests compared CompFITS2 with the *compress* "plug-in" against the stand-alone version of the *compress* program (*compress* being more of a standard than *compact*). On average, compFITS2 reduced the file size 29% more than *compress* (88 MB vs. 114 MB). On average compFITS2 was 6% slower than *compress* on file compression and 13% slower on file decompression. The results are summarized in Table 2.

4. Conclusions

- Compfits2 on average compresses files to less than half their size;

Table 2. Performance results on large multi-extension FITS files.

Compression Program	File Type	Number of Samples	Original Size (MB)	Compr. Size (MB)	Percent of Original	Compr. Time (Seconds)	Decomp. Time (Seconds)
Compfits2 with compress	Bias	18	196.64	63.96	32.5%	79.04	54.40
	Dark	81	196.64	64.39	32.7%	82.52	55.57
	Flat	43	196.64	108.94	55.4%	84.69	56.59
	Object	171	196.64	96.79	49.2%	84.47	56.42
	Combined	**313**	**196.64**	**88.18**	**44.8%**	**83.69**	**56.11**
Compress only	Bias	18	196.64	73.44	37.3%	67.76	41.93
	Dark	81	196.64	74.81	38.0%	66.68	42.33
	Flat	43	196.64	148.13	75.3%	90.10	54.64
	Object	171	196.64	128.15	65.2%	82.98	53.09
	Combined	**313**	**196.64**	**113.94**	**57.9%**	**78.86**	**49.88**

- Compfits2 attains a 10% to 16% better compression than using a compression program on its own;
- Using compFITS2 adds at most 10% to the compression time compared to using the stand alone compression program;
- Using a 16-bit compression program such as *compact* achieves a higher compression with a faster time;
- Both the CADC and the ST-ECF/ESO archive group are evaluating compfits2 for storing mosaic camera data on DVD-R.

5. Future Work

- The compression achieved by 8 bit compression programs could be optimized by splitting the 16 bit MSB values into two blocks of low and high order bytes;
- Functionality to read from stdin and write to stdout should be added to increase the versatility of the application;
- A better sampling algorithm for the entropy calculation needs to be developed;
- Optimization of file I/O to further reduce the overhead of using cfitsio in compfits2;
- The use of MSBs directly for preview applications should be explored;
- Compfits2 needs to be compared to the approach proposed in (Pence et al. 2000).

References

Pence, W. D. 1992, in ASP Conf. Ser., Vol. 25, Astronomical Data Analysis Software and Systems I, ed. D. M. Worrall, C. Biemesderfer, & J. Barnes (San Francisco: ASP), 22

Pence, W. D., White, R., Greenfield, P. 2000, this volume, 551

Véran, J.-P. & Wright, J. R. 1994, in ASP Conf. Ser., Vol. 61, Astronomical Data Analysis Software and Systems III, ed. D. R. Crabtree, R. J. Hanisch, & J. Barnes (San Francisco: ASP), 519