SES'2005
Scientific Conference "SPACE, ECOLOGY, SAFETY" with International Participation
10–13 June 2005, Varna, Bulgaria

COMPRESSION OF MULTISPECTRAL IMAGES ON-BOARD OBSERVATION SATELLITES

Tanya Vladimirova, Andreas Steffens

Surrey Space Centre
University of Surrey, Guildford, GU2 7XH, UK

Keywords: Multispectral images, observation satellites, on-board

Abstract: *Compression reduces the amount of data to be sent to ground whilst preserving its content, allowing a lower bit-rate link to be used or more data to be sent over the same link. This paper presents the results of an investigation into image compression methods for multispectral images for use on-board small satellites. Suitable compression methods are described. An error resilience scheme is proposed and implemented. An improvement to a Neural-Network based data compression method is discussed. The investigated compression methods are compared in terms of error resilience, compression ratio and execution time.*

1. Introduction

Earth Observation Satellites require transmission to ground of an extensive amount of imaging data. The data transmission capacity of onboard equipment these days is reaching up to several times as large as the down link circuit capacity provided by present satellites. High-speed data compression is required in order to transmit the image data to ground stations in real-time. Data compression is seen as a solution to the "Bandwidth Versus Data Volume" dilemma of modern spacecraft. Compression reduces the amount of data to be sent whilst preserving its content, allowing a lower bit-rate link to be used or more data to be sent over the same link.

The Surrey Satellite Technology Limited (SSTL) has developed the Disaster Monitoring Constellation (DMC) program. The DMC program is a novel international partnership, comprising a network of five low cost small satellites and ground stations. The satellites are designed and manufactured by SSTL as a Know-How transfer to the participating countries: the United Kingdom, Nigeria, Algeria, Turkey and China. From a low Earth orbit (LEO), each satellite provides 32 metre, multispectral imaging, over a 600 km swath width. The DMC program offers the possibility for daily revisiting of any point on the globe [1]. The DMC satellites have contact with ground stations 4 times a day, which only lasts between 10 and 15 minutes. The maximum data rate of the down-link is 40 Mbits per second, as long as there are minimal atmospheric irritations like clouds, rain or dust. A download of 1GByte imaging data would take about 4 minutes at optimal speed. To transmit the data, the High Level Data Link Control (HDLC) protocol is used in the best effort mode. That means it neither provides a reliable connection nor any error correction during the transmission.

This paper presents the results of a research project, the aim of which was to investigate existing image compression methods and to develop fault-tolerant software suitable for use on-board a DMC satellite. Five different image compression programs capable of lossless, near lossless and loss afflicted compression were used in this study.

Another objective of the project was to test the fault tolerance and error resilience of the software and to develop methods to prevent faults. Furthermore pre-processing stages were investigated and developed to improve the compression ratio.

The paper is structured as follows. Section 2 describes the investigated image compression methods. Section 3 outlines the proposed fault-tolerant compression scheme and section 4 presents testing results.

2. Image Compression Methods

The selected image compression algorithms, which are listed in Table 1, are capable of performing three types of compression: lossless, near-lossless and loss-afflicted or lossy. Lossless compression means that every data sample is bit identical when recovered. Therefore, the decompressed image data have exactly the same content and there is no difference between the original and the recovered image file. Near lossless compression means that during the image compression all visually irrelevant parts were discarded. Therefore, a viewer who is not aware of the original content will never be aware of the missing part. Loss afflicted compression means that during the compression both all visually irrelevant and visually relevant parts were discarded. A viewer is therefore aware of obviously missing parts in the image content.

Table 1. Investigated Image Compression Algorithms

Algorithm	**Lossless compression**	**Near lossless compression**	**Loss afflicted compression**	**General compression method**	**Image compression method**
p5	**X**			**X**	
LZMTF	**X**			**X**	
JPEG lossless	**X**				**X**
JPEG baseline		**X**	**X**		**X**
JPEG 2000	**X**	**X**	**X**		**X**

The p5 program [2] is a text compression program that interprets the entire incoming data as a bit stream. It employs a Neural Network (NN) to predict pixel probability. This program was selected because it achieves high compression ratios. Additionally, the use of an NN instead of a context model results in less memory usage and therefore it is better suited to on-board operation. The move-to-front Lempel-Ziv (LZMTF) algorithm [3] is based on the Lempel-Ziv algorithm invented in 1977 (LZ77). JPEG baseline is the well-known standard of the Joint Photographic Experts Group, which is supported by almost all imaging programs. JPEG 2000 is the present state-of-the-art standard for compressing and storing images. JPEG lossless is an implementation of the low complexity lossless compression (LOCO I) algorithm for image processing developed by the University of British Columbia, which complies with the JPEG standard.

3. Fault Tolerant Image Compression Scheme

Compression and decompression programs for the selected algorithms were developed in C and C++ using Microsoft Visual Studio. The compression programs and their reverse counterparts are targeted at the SSTL DMC satellite platform and therefore they were named using the prefixes SSTLDMC and unSSTLDMC followed by an abbreviated name of the corresponding algorithm: p5, LZMTF, jls, jpg or j2k. All programs, apart of the SSTLDMCj2k, are equipped with an error resilience mechanism, which is based on a fine-grain tiling process with image processing being performed on a tile-by-tile basis.

The even and the odd pixels have different processing stages and different outputs and amplifiers in the DMC imagers. Therefore the pixels of the even CCD cells are a bit brighter and the remaining odd pixels are a bit darker or vice versa, which results in a distortion of the image, which can be detected visually. This effect decreases the compression ratio, because the image contains more information than necessary. The problem can be corrected in the following way, referred to as Integer Gamma Difference (IDG) compensation. The average of the pixel values in every even column and odd column, representing the average brightness of all even and odd columns, are calculated. The difference between these values is then used to correct every odd column pixel by subtracting the calculated difference.

The programs process images represented in the SSTL Imaging Mode data Image file (IMI) format and convert them to a non-standard compressed image file format. A non-standard format was selected because the formats of almost all standard image or compression methods, which are available, do not provide any kind of error resilience. However, they produce useful compression results and therefore they are integrated into the resultant file format. The compression programs, apart of SSTLDMCj2k, follow several stages to compress an image, as shown in Figure 1.

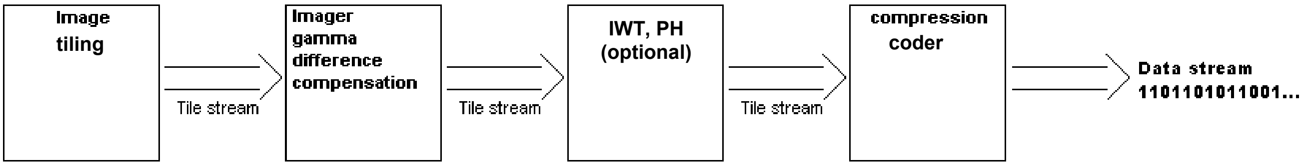

Figure 1. Fault Tolerant Image Compression Scheme

At the first stage the image is divided into tiles with a size of 625 x 625 pixels. The smallest DMC image covers an area of 80 km x 80 km and has 2500 x 2500 pixels, so it is broken down to 16 tiles (4 x 4 image tile array). The rest of the stages in Figure 1 process the tiles in a serial fashion, one tile at a time. The tile stream is passed on to the second stage, where the IGD correction is applied. The next stage calculates the integer wavelet transform (IWT) to minimise the entropy of the image tile. Here either the Haar IWT or the 5-3 IWT, employed in JPEG 2000, are used. At this stage other decorrelation techniques as the Peano-Hilbert (PH) scanning can be performed too [4]. This stage is optional depending on the compression configuration that is utilized, which is indicated in the tile header.

The actual compression algorithm is performed at the compression coder stage, which is algorithm specific. The p5 and LZMTF algorithms employ parameters, the initial optimal values of which are obtained empirically and then have to be adjusted for each tile at run time. For example, the compression ratio achieved by the NN-based p5 algorithm depends on the values of two parameters which may be changed at run time: the long-term learning rate (RL) and the short-term learning rate (RS). The initial optimal values for these parameters are derived from experimental data using a modified “hill climber” optimisation algorithm. We have proposed and implemented a method for automatic adjustment of the parameters RL and RS during run time. Initially two or more optimal pairs of empirical parameter values are passed to the compression stage. The first pair is then used for the first tile in the image tile array, the second pair for the second tile, etc. After the compression of these tiles it is determined which of the pairs produced the best result by comparing the size of the compressed tiles. The better parameter pair is employed for all of the remaining tiles in that row of the image tile array.

The compression programs are able to switch off processing stages in case of failure. The IWT stage can be switched off in the event of insufficient memory or not enough mass memory space. The compression coder stage can be switched off in the event of insufficient memory or if the coder encounters an irregular internal state.

4. Testing Results

The test image set is composed of 17 unprocessed SSTL DMC multispectral satellite images with total size of 3.5 GByte. These image files were divided into tiles, which are then decomposed into 3 components, corresponding to the individual optical wavebands (green, red, near infrared). The components of each tile are stored in three grey-scale files with binary Portable Grey Map (PGM) format. For p5 this separation results in 10,356 PGM files having unique optimal values for the RL and RS parameters, which must be obtained individually for each tile component. The binary PGM file format was selected because it looks very similar to the data stored in memory, when SSTLDMCp5 generates the tiles of the large image. The only difference between the raw data in the memory and the file data is a 15 to 17 bytes ASCII header in the PGM file, which has practically no impact on the result.

The diagram in Figure 2 displays the results of the error resilience testing, where the fault-tolerant programs are shown on the left hand side and the original programs on the right hand side. The recovery results have been split up into four categories, as follows:

- **Decoder crashes:** The decoder did not return an image file. Either it was not able to process the compressed file or it recognized the fault and denied further processing.
- **Partial recovery:** The decoder was able to process the compressed file, no matter it recognizes a fault or not. It writes an image file, but the image file was corrupted and just a part of it can be used.
- **Useful recovery:** The image can be visualized and the content is identifiable, but there is a slight error in the image.
- **Full recovery:** The decoder was able to recover the entire image file without any loss.

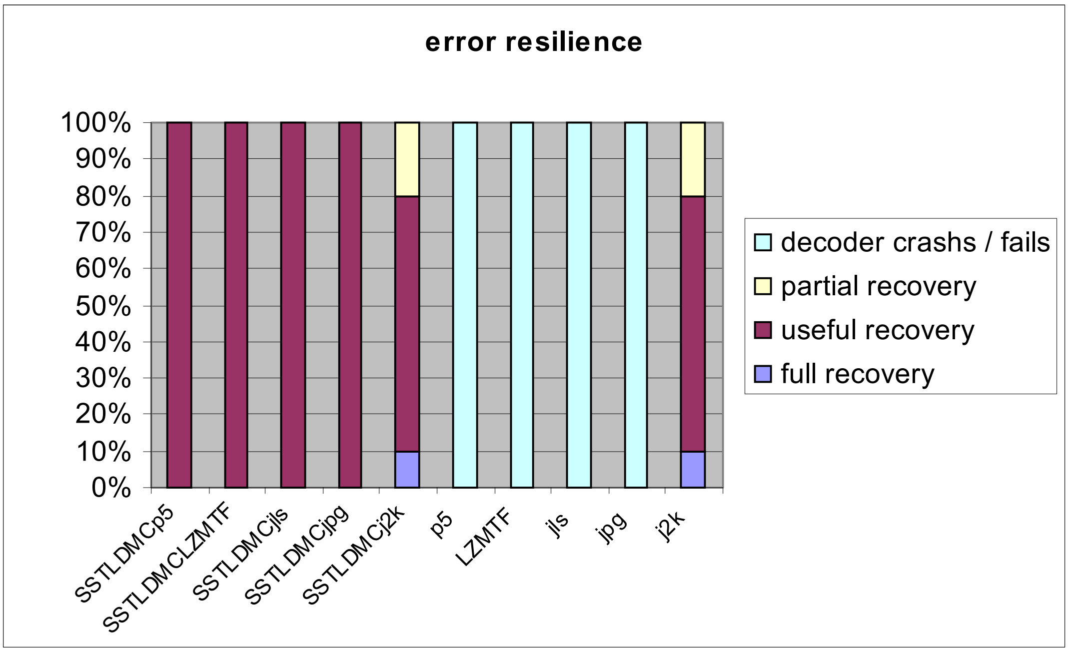

Figure 2. Error Resilience Test

It can be seen that all original programs apart of JPEG 2000 are very vulnerable to data errors prohibiting reconstruction of the data after a fault or crash. However, all

SSTLDMC implementations are resistant to data errors, because of the image tiling. If one tile of the data stream is lost it does not influence the other tiles. For JPEG 2000 the results are the same, since it already provides error resilience and was only expanded to read in IMI files.

Table 2 shows the performance results for lossless compression in the form of the ratio between the size of the original (uncompressed) and of the compressed images, which is averaged across the image test set. Table 3 presents the performance results for near lossless compression. The bottom half of the tables represents the compression ratios for the cases when IGD compensation is applied.

Table 2. Compression Ratios for Lossless Compression

Name	no pre-stage	Peanno-Hilbert scanning	Haar IWT	Haar IWT cascaded	5-3 IWT	5-3 IWT cascaded
SSTLDMCp5	2.31	2.00	**2.36**	2.30	1.63	1.84
SSTLDMCLZMTF	1.63	1.52	**1.84**	1.78	1.28	1.38
SSTLDMCjls	**2.58**	1.86	2.31	2.32	1.30	1.68
SSTLDMCj2k	**2.44**	1.71	1.01	1.12	1.37	1.71
IGDC SSTLDMCp5	2.37	2.36	**2.39**	2.39	1.65	1.65
IGDC SSTLDMCLZMTF	1.73	1.73	**1.88**	1.84	1.10	1.26
IGDC SSTLDMCjls	**2.62**	2.28	2.30	2.31	1.03	1.42
IGDC SSTLDMCj2k	**2.45**	1.99	0.89	0.89	1.03	1.43

Table 3. Compression Ratios for Near Lossless Compression

Name	no pre-stage	Peanno-Hilbert scanning	Haar IWT	Haar IWT cascaded	5-3 IWT	5-3 IWT cascaded
SSTLDMCjpg	**5.65**	3.84	1.47	1.52	2.40	3.34
SSTLDMCj2k	**2.76**	1.76	1.01	1.11	1.42	1.86
IGDC SSTLDMCjpg	**6.07**	4.47	1.32	1.32	1.70	2.61
IGDC SSTLDMCj2k	**2.89**	2.14	1.00	1.00	1.02	1.49

Based on the results in Table 2 and 3 and on the experimental results for loss afflicted compresson, the recommended compression methods and pre-processing stages respective to the compression categories are:

lossless: → **JPEG lossless** with IGD correction
near lossless: → **JPEG baseline** with IGD correction
loss afflicted: → **JPEG 2000** with IGD correction

The compression programs would be executed on the Solid-State Data Recorders (SSDR) in the imaging payload of a DMC satellite. Each SSDR provides 1 MByte of main memory and operates with CPU speed of 100 MHz at a peak performance of 190 MIPS [5]. Figure 3 shows the estimated execution time that one SSDR needs in order to compress 1 GByte of image data respective to the compression method. The tests wee run on a Pentium 4 machine at 2.8 GHz using the test image set of size 3.5 GByte, however the timings were scaled down to correspond to 1 GByte of image load, because this is the storage capacity that a DMC satellite can provide. The SSDR on-board run time in Figure 3 was estimated using the MIPS peak performance.

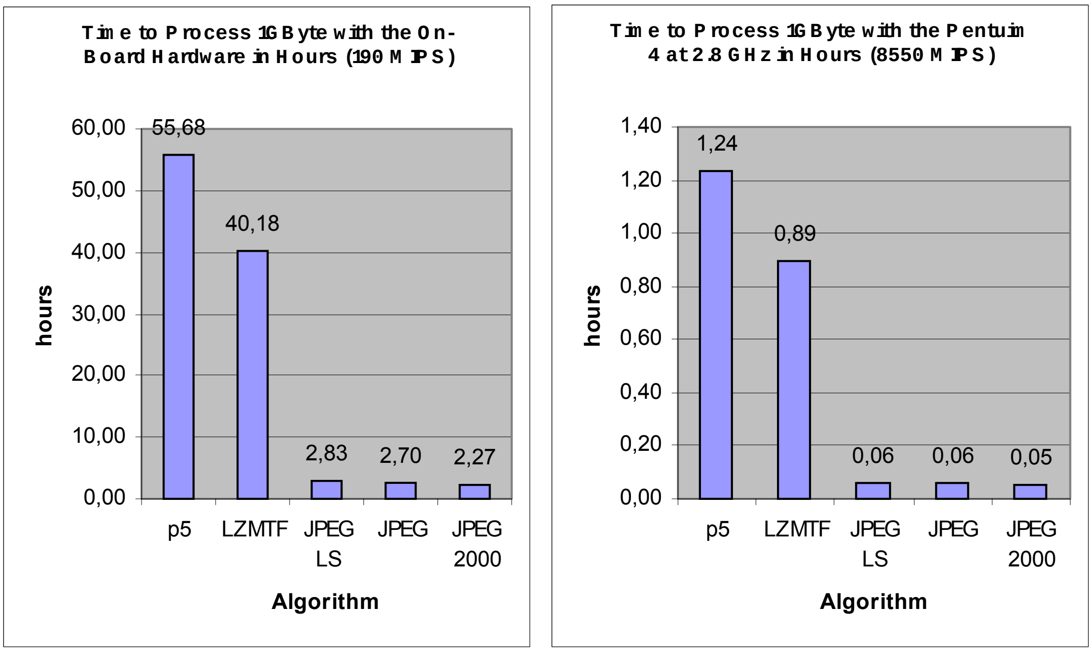

Figure 3. Compression Execution Time

As can be seen from Figure 3, the JPEG compression algorithms require significantly smaller execution times to compress 1 GByte of data in comparison to the p5 and LZMTF algorithms. Taking into account that the JPEG compression algorithms have also shown a better compression performance, as outlined above, they should be preferred for use on-board small satellites. However the estimated on-board timings in Figure 3, even with respect to the JPEG algorithms, are quite high. This illustrates the fact that the present computing resources on-board small satellites are not suitable for running complex DSP algorithms sufficiently fast.

5. Conclusions

This paper has presented the results of an investigation into image compression algorithms for use on-board small satellites in the SSTL DMC program. A fault-tolerant compression scheme was proposed based on a fine-grain tiling mechanism, which can be applied to any image compression algorithm. A good illustration of the effectiveness of the scheme is the NN-based p5 algorithm, which achieved 100% useful recovery although originally the entire file content became corrupted in the event of a single bit-flip in the compressed file. A new automatic adjustment method for parameter values at run time was designed. Comparison of compression algorithms in terms of resilience to errors, compression performance and execution time was presented, on the basis of which practical recommendations were given.

References

---

1. Surrey Missions: Disaster Monitoring Constellation, Datasheet, 2002, SSTL
2. M. Mahoney. Fast Text Compression with Neural Networks – Proc. of the 13th International Florida Artificial Intelligence Research Society Conference, pp 230-234, AIAA Press, 2000.
3. Mark Nelson. The Data Compression Book - IDG Books Worldwide, Inc., 1995
4. S.Atek, T.Vladimirova, A. da Silva-Curiel, M. Sweeting. Fast Neural Network Based On-Board Image Compressor – Proceedings of the 54th International Astronautical Congress, ref. number IAC-03-B.6.08, Sept. 29 – Oct. 3, 2003, Bremen, Germany.
5. MPC8260 Solid State Data Recorder, ISSUE-9034-1, #39030, 14 August 2002, SSTL