Lossless Compression Algorithm for Multispectral Imagers

Irina Gladkova[a] and Michael Grossberg[a] and Srikanth Gottipati[a]

[a]CCNY, NOAA/CREST, 138th Street and Convent Avenue, New York, NY 10031, USA

ABSTRACT

Multispectral imaging is becoming an increasingly important tool for monitoring the earth and its environment from space borne and airborne platforms. Multispectral imaging data consists of visible and IR measurements from a scene across space and spectrum. Growing data rates resulting from faster scanning and finer spatial and spectral resolution makes compression an increasingly critical tool to reduce data volume for transmission and archiving. Research for NOAA NESDIS has been directed to finding for the characteristics of satellite atmospheric Earth science Imager sensor data what level of Lossless compression ratio can be obtained as well as appropriate types of mathematics and approaches that can lead to approaching this data's entropy level. Conventional lossless do not achieve the theoretical limits for lossless compression on imager data as estimated from the Shannon entropy. In a previous paper, the authors introduce a lossless compression algorithm developed for MODIS as a proxy for future NOAA-NESDIS satellite based Earth science multispectral imagers such as GOES-R. The algorithm is based on capturing spectral correlations using spectral prediction, and spatial correlations with a linear transform encoder. In decompression, the algorithm uses a statistically computed look up table to iteratively predict each channel from a channel decompressed in the previous iteration. In this paper we present a new approach which fundamentally differs from our prior work. In this new approach, instead of having a single predictor for each pair of bands we introduce a piecewise spatially varying predictor which significantly improves the compression results. Our new algorithm also now optimizes the sequence of channels we use for prediction. Our results are evaluated by comparison with a state of the art wavelet based image compression scheme, Jpeg2000. We present results on the 14 channel subset of the MODIS imager, which serves as a proxy for the GOES-R imager. We will also show results of the algorithm for on NOAA AVHRR data and data from SEVIRI. The algorithm is designed to be adapted to the wide range of multispectral imagers and should facilitate distribution of data throughout globally. This compression research is managed by Roger Heymann, PE of OSD NOAA NESDIS Engineering, in collaboration with the NOAA NESDIS STAR Research Office through Mitch Goldberg, Tim Schmit, Walter Wolf.

1. INTRODUCTION

In this paper we present a new algorithm for lossless compression of multispectral imager. Multispectral imaging data consists of visible and IR measurements from a scene across space and spectrum. Growing data rates resulting from faster scanning and finer spatial and spectral resolution makes compression an increasingly critical tool to reduce data volume for transmission and archiving. Research for NOAA NESDIS has been directed to finding for the characteristics of satellite atmospheric Earth science Imager sensor data what level of Lossless compression ratio can be obtained as well as appropriate types of mathematics and approaches that can lead to approaching this data's entropy level. Conventional lossless do not achieve the theoretical limits for lossless compression on imager data as estimated from the Shannon entropy. In a previous paper,[1] the authors introduce a lossless compression algorithm developed for MODIS as a proxy for future NOAA-NESDIS satellite based Earth science multispectral imagers such as GOES-R. That algorithm uses a non-linear statistical method based on histogram specification to remap the intensities in neighboring spectral bands in order to predict intensities of the band that needs to be compressed. This paper present a different prediction based compression approach that achieves a higher compression ratios on the tested data sets.

Conceptually, the data stream can be thought of as broken into segments. The first segment is compressed by some standard compressor. If the segments can be ordered, and a predictor found such that each segment can

Further author information: (Send correspondence to I. Gladkova)
E-mail: gladkova@@cs.ccny.cuny.edu, Telephone: 1 212 650 6261

Satellite Data Compression, Communication, and Processing IV
edited by Bormin Huang, Roger W. Heymann, Joan Serra-Sagrista
Proc. of SPIE Vol. 7084, 70840D, (2008) · 0277-786X/08/$18 · doi: 10.1117/12.800819

predict subsequent segment well in some metric, the stream of residual differences between successive segments is small in that metric. Ideally a perfect predictor would give a stream of segments of zero compressed size. As a result, the lossless compression would effectively be infinite. In practice we only require the predictor be accurate enough so that the compressed residuals are small when compared with the size of the information needed to store the predictor. In the new algorithm we present here, our segments are single band images. In previous work we used a single non-linear lookup table predictor which was applied to all footprints (the collection of bands at a spatial pixels) for a given sensor. In contrast, our new algorithm shows that a simple linear predictor can out-perform a more complex non-linear predictor, if we allow the linear predictor to vary spatially across the image. In addition, our previous work naively used spectral order for predictive compression while the new algorithm we present improves the compression performance by reordering the sequence of images that we iteratively predict.

We present the performance of this new algorithm on data from NOAA's AVHRR, EUMETSAT's SEVIRI, and a 14 channel subset of NASA's MODIS imager that provides a proxy for GOES-R.[2] In particular the following table 1 shows the correspondence between the GOES-R specification and the subset of 12 bit MODIS channels used as a proxy:

Table 1. MODIS - GOES ABI

ABI Band No	Center Waveln (μm)	MODIS Band No	Center Waveln (μm)	ABI Res (km)	ABI Bitdepth
01	0.47	3	0.47	1	10
02	0.64	1	0.659	0.5	12
03	0.865	2	0.865	1	10
04	1.378	5	1.240	2	10
05**	1.61	6	1.640	1	10
06	2.25	7	2.130	2	10
07	3.9	22	3.96	2	14
08*	6.19	NA	NA	2	11
09	6.95	27	6.78	2	11
10	7.34	28	7.34	2	12
11	8.5	29	8.55	2	12
12	9.61	30	9.72	2	11
13*	10.35	NA	NA	2	12
14	11.2	31	11.0	2	12
15	12.3	32	12.0	2	12
16	13.3	33	13.4	2	11

2. BACKGROUND

Prediction based compression algorithms are common. A simple example is to take successive differences of samples in a data stream. Runs in the data of constant values result in the difference stream having runs of repeated zeros. Simply recording the length of these runs, called run length encoding can result in significant lossless compression. For example for synthetic images such as logos or rasterized computer generated drawings this simple algorithm can exploit the fact that many adjacent pixels have the same value. Images coming from natural scenes, such as remotely sensed images, are much more complex and require more sophisticated prediction to capture dependencies. Researchers at JPL presented a compression algorithm based on multi-linear prediction of each successive sample of a multi-spectral image.[3] In their algorithm, the multi-spectral image is traversed along a 1-dimensional path of samples (in raster order). At a fixed set of relative locations with respect to the next pixel in the path, the algorithm uses a set of values at pixels already traversed. A multi-linear predictor is then applied to these values to obtain a prediction for the next pixel. The difference between the prediction and the actual value is stored, and the multi-linear predictor is then updated to minimize the error for the actual value at the next pixel. This approach would work well on smooth data, and it has a number of advantages. It only requires the storage of a minimal number of parameters, beyond the residuals themselves. It is relatively simple to implement, fast and adaptive. A disadvantage is that it implicitly assumes that the primary dependencies of the data are its smoothness, and the limited size of the prediction window and its inherent asymmetry limit its performance.

Simple linear prediction in the spatial direction is not competitive with the wavelet based Jpeg2000 algorithm[4,5] which are designed for representing both smooth and piecewise discontinuous elements present in natural images. The dependencies in the spectral dimension are based on different statistics as those found in the spatial dimensions. Spectral dependencies are driven by physical relationships between reflectance of different constituents in the atmosphere and on the earths surface, as well as the properties of solar illumination, thermal emission and radiative transfer. When a group of bands and a group of pixels have brightness effectively generated by a single constitute, such as a cloud, or the ocean surface, it is reasonable to expect that the brightness should be correlated and a prediction based approach effective in that region. This is the motivation for the algorithm we present, and we will show that the intuition is born out by better performance in terms of compression ratios, than other methods. In all of the considered test cases the new algorithm is able to significantly improve the lossless compression of when compared with the current stated-of-the-art lossless compression algorithms.

Table 2. Compression ratios, 14 channels of MODIS. CCSDS,[6] J - Jpeg2000 (Jasper[7]), P07 - our predictor algorithm presented at SPIE07,[1] P - new predictor algorithm presented in this paper.

Name of the Granule MOD01.A2006	PNG	TIFF	Zip	Gzip	Bzip2	7zip	CCSDS	J	P07	P
174.1005.005.2006214124917	1.45	1.79	1.72	1.72	1.72	2.35	2.59	2.99	3.24	3.47
175.0915.005.2006214125006	1.33	1.65	1.67	1.67	1.67	2.19	2.42	2.69	2.80	2.98
176.0955.005.2006214124831	1.38	1.68	1.63	1.63	1.63	2.18	2.47	2.77	3.07	3.24
177.0900.005.2006215131020	1.32	1.63	1.6	1.6	1.6	2.12	2.42	2.7	2.95	3.14
179.0850.005.2006214124349	1.19	1.5	1.5	1.5	1.5	1.95	2.27	2.46	2.64	2.82
180.0930.005.2006214124332	1.37	1.68	1.64	1.64	1.64	2.19	2.47	2.77	3.07	3.25
181.1010.005.2006214124336	1.37	1.73	1.7	1.7	1.7	2.29	2.52	2.88	3.01	3.29

3. COMPRESSION APPROACH

Given a sequence of bands our algorithm relies on Jpeg2000 compression to capture the spatial relationships within the first band. The next, and each successive band is predicted from the previous one using a spatially varying linear predictor. The parameters of the predictor are computed for successive bands in the sequence. Once the parameters are computed, the predictor is used to predict the next band, and the residual difference between the predicted and the actual bands are stored along with the predictor parameters for that pair of bands. The residual images and the coefficients are then compressed using Jpeg2000. The process continues through the sequence until the final band has been predicted. We used the jasper implementation of Jpeg2000.[7]

In previous work we had used a single non-linear predictor for all spatial pixels (footprints). The relationships between the bands, however, is too complex to be captured by a single function for all spatial pixels. As we consider different spatial pixels the constituents of the atmosphere and the earth's surface vary. This results in changes in spectral dependencies. To account for these changes our predictor must vary along spatial dimensions.

In the spectral dimension, dependencies are often discontinuous due to spectral absorption. Hence, we should not assume that bands which are closest spectrally, will predict each other well. Hence, to improve predictive compression, we should optimize over the prediction order. The algorithm we developed for this is described in detail in.[8] The algorithm as applied in this work consists of passing our spatially varying predictor to the sequence optimization algorithm, on a given data set, and using the optimal sequence it returns to do the predictive compression.

3.1 Spatially Varying Linear Prediction

In this section we will describe the spatially varying predictor we have developed to successfully predict related imager bands. We considered three factors when choosing our predictor: compressed size of the residual image (prediction accuracy), compressed size of the predictor parameters, and speed of the algorithm. At one extreme, independently determining a different predictor for each pixel gives an exact prediction. In that case the residual images vanish and they take up essentially no size when compressed. Unfortunately, the predictor parameters are as large as the original data and no compression is achieved. At the other extreme we can create one predictor for all the pixels. This results in a greatly reduced predictor size. This idea is the basis of our previous algorithm.[1] As already noted, that algorithm does not take into account the spatial variations in the relationships between the bands.

What is required then is a compromise of a local predictor which uses a local groups of pixels to determine a predictor which is constrained to have a small number of parameters. One approach to building a local predictor would be to use a radial gaussian weighting kernel at each pixel in order to build the predictor. Because the predictor must be constrained to cope both with the need for a compact form, and limited data in the local region, we can use a locally varying spline. That is we can consider a polynomial predictor which varies spatially.

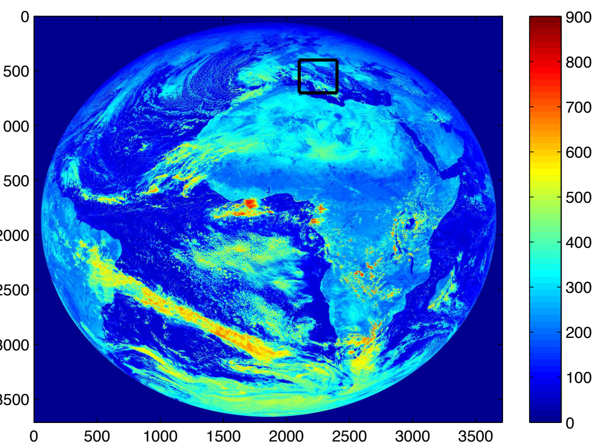

Figure 1. SEVIRI Band 2, Full Disk

One of our considerations is speed. The degree of the local spline predictor will effect the speed. It will also impact the number of compression parameters. For this reason we have picked a very simple linear predictor of the form $y = mx + b$ where m and b are the parameters of our predictor, x represents the a value in the band we are trying to predict, and y represents our prediction in the next band in the prediction sequence. To explain the algorithm, we will use bands 2 and 3 from a granule of EUMETSAT's SEVIRI imager. The full disk band 2 image is shown in Fig. 1. The black rectangle in the image, includes part of Tunisia and Italy, is a 300x300 region of interest, chosen simply to illustrate the algorithm. The region of interest in band 2 shown enlarged in Fig. 2. The values from band 3 for the same region of interest set of are shown in Fig. 3.

Given the band 2 and band 3 data shown in Fig. 2 and Fig. 3 we need to determine a compact predictor which can predict band 3 from band 2. We break the spatial image pixels into N non-overlapping windows $W_1, \ldots W_N$ rectangular spatial blocks, each $r \times c$ pixels. For illustration we will assume $r \times c =$ 10x10. For each of the pixel windows W_i we determine two parameters: m_i and b_i by performing a linear regression minimizing the least square error

$$E_i = \sum_{j \in W_i} ||mx_j + b - y_j||^2, \tag{1}$$

where x_j is a pixel value in the W_i'th block of band 2 and y_j is the same pixel in band 3. The figures Fig. 5, and Fig. 4 the 30x30 image M of scale factors m_i and the 30x30 image B of offsets b_i obtained by independent linear regression of each block using the data in the region of interest from band 2 and 3 shown in Fig. 2 and Fig. 3.

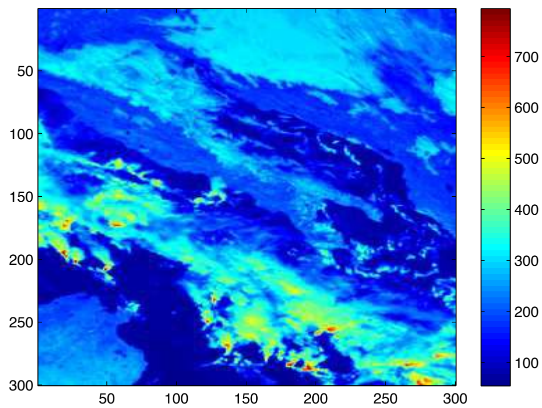

Figure 2. The image shown here is 300×300 pixel cropped section of a SEVIRI Band 2 digital counts. Our algorithm actually operates on the whole image but to illustrate, we will assume that this cropped portion is the part of the data on which we will predict the a corresponding band 3 portion on. This band is compressed using conventional Jpeg2000 compression.

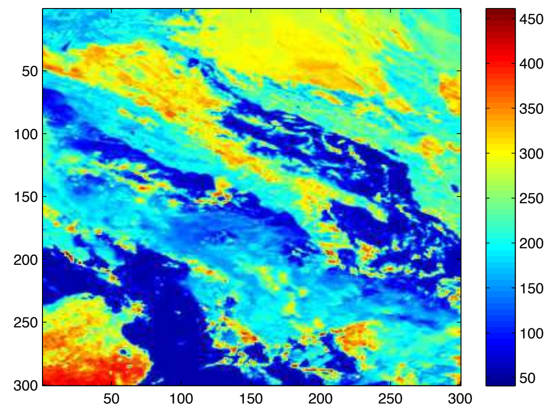

Figure 3. The image shown here is a 300×300 pixel cropped section of SEVIRI Band 3 digital counts corresponding to the band 2 section shown in the previous image. The approach of the compression algorithm is to use a predict this image from the previous band 2 image.

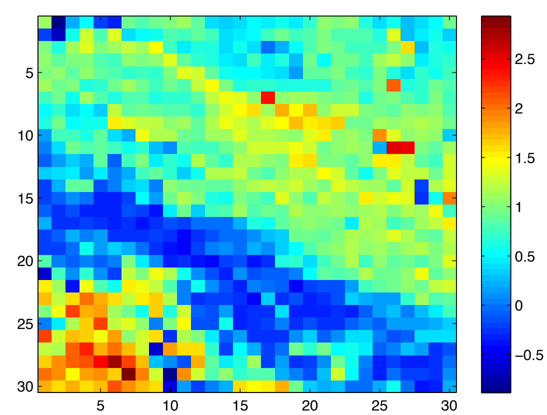

Figure 4. A 30×30 image of prediction offsets as determined by solving a least square linear regression on 10×10 pixel blocks to best predict the values of SEVIRI Band 3 digital counts shown in the previous image from the corresponding Band 2 counts. This offset is computed from the given Band 2 and 3 digital counts during the *encoding* phase. The offset for each block corresponds to the b in a per block linear predictor $y = mx + b$, with x and y being digital counts in band 2 and 3 respectively.

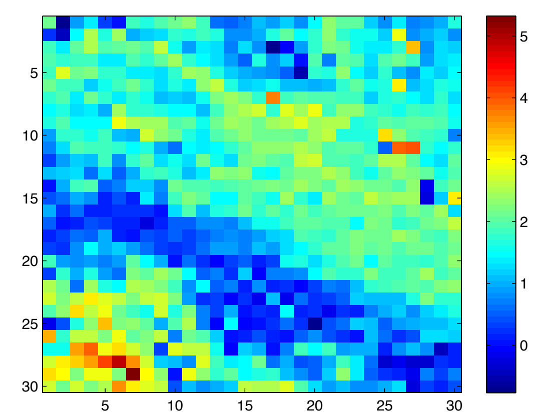

Figure 5. A 30×30 image of prediction scale factors as determined by solving the least square linear regression as in the previous figure. The scale factor, like the offset, is computed during the *encoding* phase. The scale factor corresponds to the m in the per block linear predictor y=mx+b.

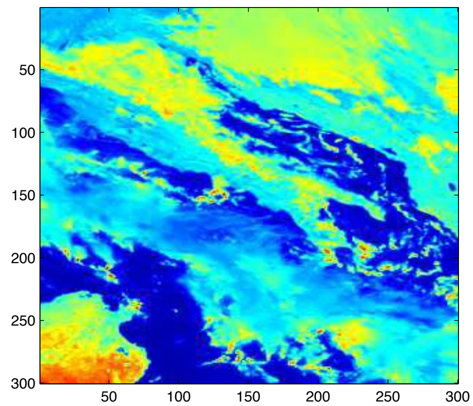

Figure 6. This shows the result of predicting SEVIRI band 3 by multiplying band 2 by the image in Fig. 5 after interpolating to using the a spatially varying linear predictor offset and linear scale factors applied to the values in band 2 using the per block linear predictor

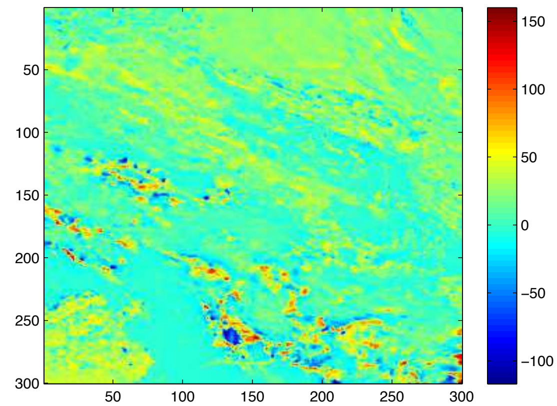

Figure 7. The result of subtracting the predicted image in Fig. 6 from the actual digital counts in Fig. 3.

After having determined the coefficients to predict band 3 from band 2 the next step is to build the predictor. Our block independent regression is an approximation of a smoothly spatially varying linear predictor. To accomplish this we use bilinear interpolation to interpolate prediction coefficients B and M to images $\tilde{B}$ and $\tilde{M}$ which are the same size as C. We then apply $\tilde{B}$ and $\tilde{M}$ (which are the same size as C and P) to the normalized P' to obtain an approximation of the normalized C' given by $\tilde{M} \times P' + \tilde{B}$ where $\times$ is pixel wise multiplication. We rescale $\tilde{M} \times P' + \tilde{B}$ and quantize to match band 3 so that the resulting predicted image $\tilde{C}$ shown in Fig. 6, matches C as closely as possible. Note that $\tilde{C}$ has minimal if any block artifacts. To accomplish lossless compression, we subtract the prediction from the actual band 3 image to obtain an image of residuals $R = C - \tilde{C}$, shown in Fig. 7.

The performance of the compression depends on the residules, R, being significantly easier to compress than band 3, C, itself. As discussed above, even if the predictor is not perfect because information in band 3 that cannot be predicted from band 2, much of the residual image is close to zero and compresses well. The false color image shows small values as green. The predominance of green shows that the predictor performs well. This is one important factor in making the compressed size of the image compact. It is also important to note that the despite using regular tiles there are no blocking artifacts. The is very important because any introduction of sharp artifices will adversely effect the Jpeg2000 post-processor.

The block diagram of Fig. 8 summarizes the one encoding step of our algorithm. We assume that we are given two images in our compression sequence, a parent image P, for instance band 2 of SEVIRI in our previous example, and a child image C, which corresponds to SEVIRI band 3. We assume that P has already been compressed and saved. First values in the images are all rescaled to go from 0 to 1. The rescaled images are broken into blocks and passed to a processor and linear regression is applied to find the prediction parameter images M and B for each block. The values of the predictors are interpolated using bilinear interpolation to the full size of C. After interpolation, the linear prediction based on M and B is applied to P and the result is rescaled and then requantized to the original range to obtain $\tilde{C}$. The predicted child image is subtracted from the actual child image to obtain an image $R = C - \tilde{C}$ of residuals. The images M,B and R are then compressed using Jpeg2000 and they are appended to the compressed data. When the compression sequence is finished, all the data except for the first image, has been covered to Jpeg2000 compressed versions of M,B and R, and the first image is simply compressed using Jpeg2000.

The decoding processes starts by extracting the sequence information (metadata) from the file. Then the first image P in the sequence is extracted using Jpeg2000. Following that M and B are decompressed interpolated and applied to the parent image P, followed by rescaling and quatization to obtain the predicted image $\tilde{C}$. The residuals are added back to obtain the lossless reconstruct $C = R + \tilde{C}$. One feature of our prediction is because we use blocks and a very simple prediction algorithm, this algorithm is easy to parallelize and uses a few simple arithmetic operations. As a result it should be possible to make it extremely fast.

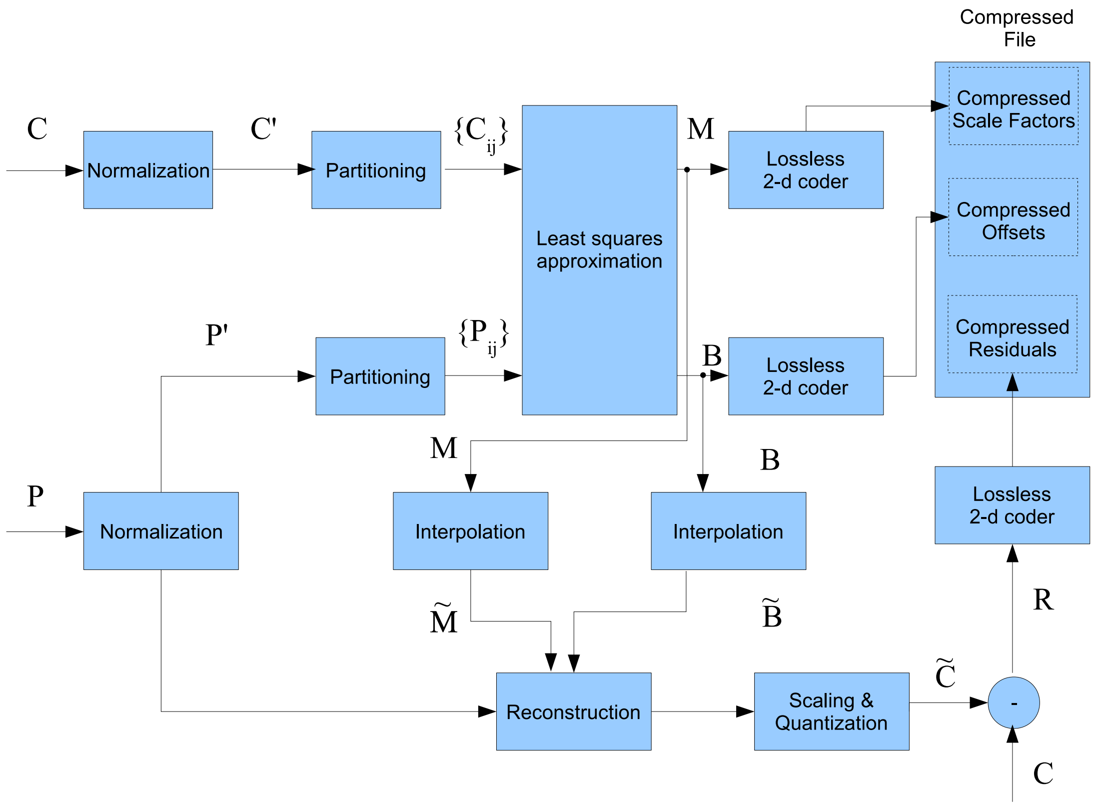

Figure 8. Diagram outlining the compression encoding. The basis of the algorithm is to build from two bands denoted P for parent, and C, for child. After normalizing these images, C' from P' on their stated range, to go from 0 to 1, the parent and child are broken into sets of non-overlapping blocks $\{P_{i,j}\}$, and $\{C_{i,j}\}$ respectively. The predictor parameters are the output of a per-block linear regression fit: a set of scale factors M and offsets B. The per-block regression parameters are then scaled up by bilinear interpolation to $\tilde{M}$ and $\tilde{B}$ giving a per pixel predictor. If we apply this predictor at every pixel to a P' (reconstruction), then rescale and requantize to obtain an approximation of C called $\tilde{C}$. To losslessly reconstruct C we store the image of residuals $R = C - \tilde{C}$ along with the predictor parameters M, and B which are compressed using Jpeg2000. The compression occurs because the compressed size of M, B and R combined, is smaller than that of C because information already contained in P has been removed.

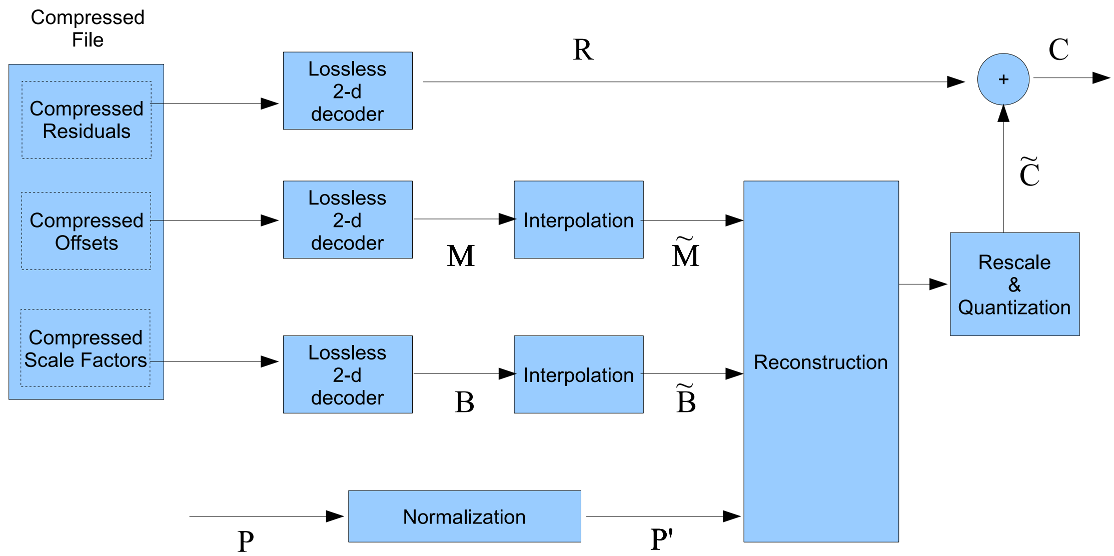

Figure 9. Diagram outlining the decompression decoding. The stored residuals, offsets and scale factors are losslessly decompressed to yield R,M, and B respectively. We assume we have already reconstructed the parent image P from a previous step. The linear prediction specified by M and B is applied to P in followed by quantization and rescaling to create the predicted image $\tilde{C}$. The residules are added back to the prediction $R + \tilde{C} = C$ to recover the child image losslessly.

3.2 Algorithm Parameters

The performance algorithm depends on a number of parameters. We have used the wavelet based jpeg2000 compressor to capture spatial redundancy in the parameter and residual images. It is also important to note that at the start of the prediction sequence there is no initial prediction and the entire image is compressed using jpeg2000. We could replace this step with any spatial image compressor, if we find that the performance is not negatively impacted or if we have access to a better wavelet or non-wavelet based image compressor.

Another important parameter of prediction is the window size. Each row in the figure 10 shows the prediction model offset image, scale factor, and the predicted image computed for the same SEVIRI data shown in previous images. Both of the two images used for the prediction model are scaled to the original image size 300x300 using bilinear interpolation. The predicted band 3 image shown at the far right of the top row clearly shows the need for a spatially varying predictor. The prediction at this course resolution does poorly when compared with the actual band 3 image of Fig. 3 in the lower left of the image (Tunisia) although its performance is reasonable in other parts of the image clearly as we decrease the window size the predicted image becomes closer and closer to the actual band 3 image. As the window size decreases the accuracy of the prediction increases and hence the compressed size of the residual image (e.g. Fig. 7). The improved compression of the residual comes a the cost of having to store more complex offset and model scale factor images.

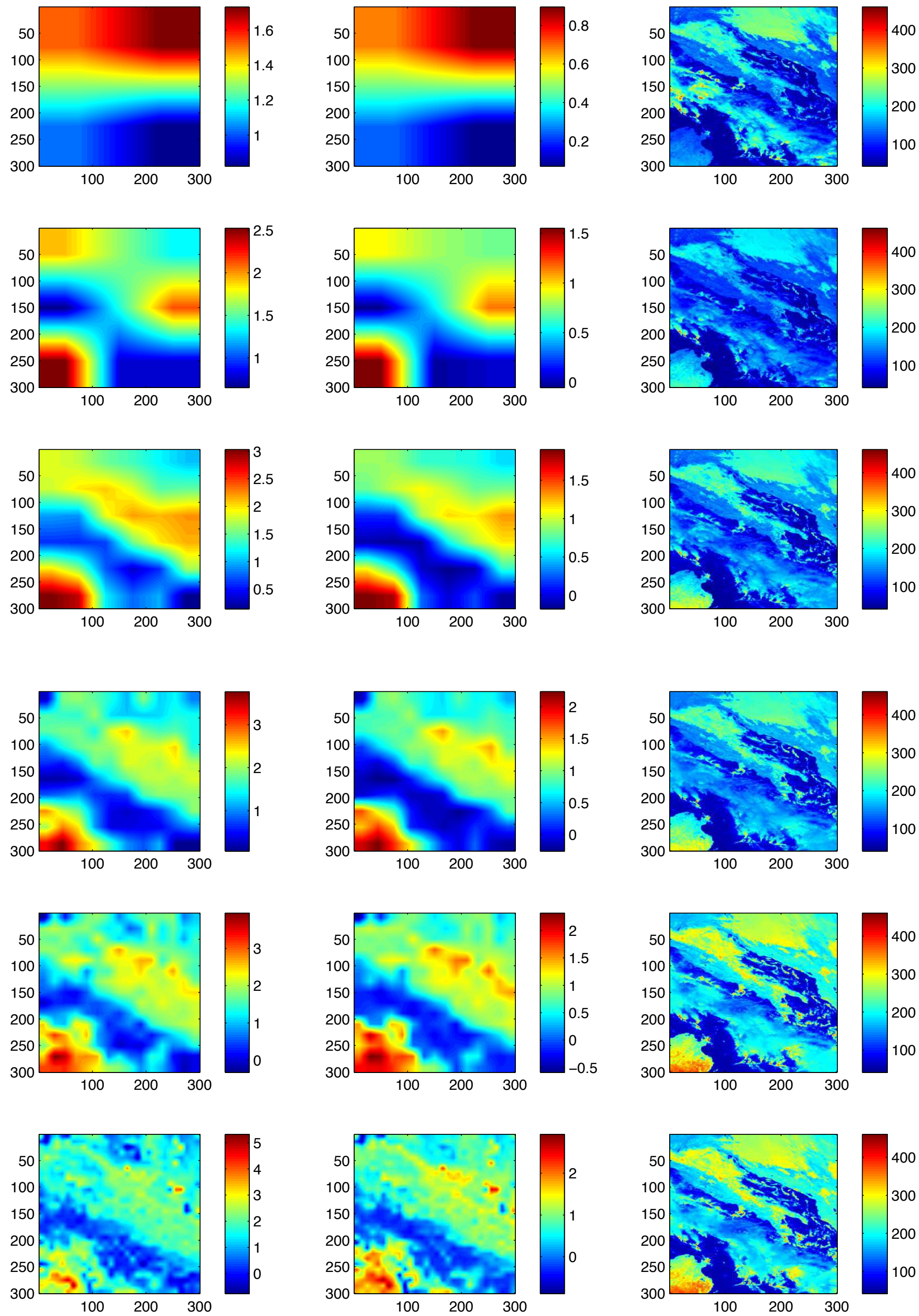

Figure 10. Each row shows the prediction model offset image, scale factor, and the predicted image computed for the same SEVIRI data shown in previous images. The window sizes used in the prediction are 150×150, 100×100, 50×50, 30×30, 20×20, and 10×10, from top to bottom row.

3.3 Sequence Optimization

A key point we noticed in our previous work is that the order of a compression sequence can be very important to the compression performance. One approach to find the optimal sequence of predictions for compression is to try every possible sequence during the encoding stage and use the sequence with the best results. The number of such triles is K factorial where K is the number of band. This number is superexponential and makes this approach impossible for anything but a very small number of bands. The situation is actually somewhat worse since we really should test whether or not we want to use prediction at all or simply compress without prediction. For example for a channel with a large amount of noise, or strong artifacts, prediction may not be beneficial. What we are really searching for is a graph which is the union of directed trees (a forest) in which the edges represent one image predicting another, and the vertices are the bands.

We will show that there is a graph algorithm that solves this problem optimally from a similarity matrix where the edges are weighted by the compression performance. Fig. 11 shows the optimal compression tree computed for one SEVIRI granule. Edges are directed from top to bottom. Vertices are labeled with the band of the imager. The first vertex labeled "0" is a virtual vertex to make the forest a tree. It also indicates that any band directly connected to "0" is just Jpeg2000 compressed.

The technique of computing this tree per granule results in the smallest files size but is very computationally expensive and thus slow. Another approach is to use an entropy based statistical estimation using entropy to determine if there is a single tree which can be used for all granules, or groups of granules. This approach is discussed in another paper in this conference.[8]

We compared results of compression for both day and night with and without the optimal tree sequence. In all cases there was a significant benefit using an optimal tree when compared with simply using the band sequence.

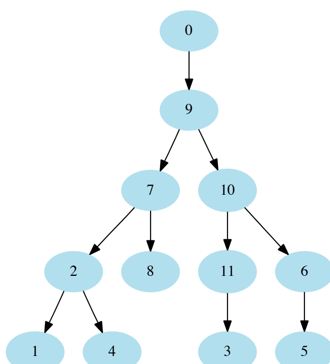

Figure 11. Optimal Tree.

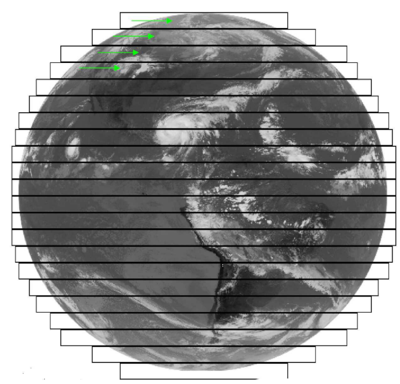

Figure 12. ABI Swathpattern.[9,10]

4. RESULTS

One of the main goals for this work is develop a compression algorithm for future GOES-R type data. Because SEVIRI represents a recent imager on a geosynchronous platform, it is an important reference in estimating future compression results for the future GOES-R mission. One word of caution is that because SEVIRI images represent full disk images some pixels view space. These pixels appear to be precisely zero. Hence if we look at the bits per sample achieved by Jpeg2000 and our algorithm shown in the first two columns of Fig. 3, the lossless bit rate seems remarkable low when computed on the whole image. GOES-R will use a swath pattern as shown in Fig 12. Hence it may be more sensible to consider the bit rates in the first two columns multiplied by $4/pi$. This number represents the ratio of the area of a unit square to a unit disk. The first two columns are multiplied by this number to obtain the last two columns. They represent the bit rate only considering the samples within the disk. In any case, the results show the an improvement over Jpeg2000.

Table 3. Bits per sample, SEVIRI. J – Jpeg2000 (Jasper[7]), P – new predictor algorithm presented in this paper.

Name of the File MSG2-SEVI-MSG15-0100-NA	Day/ Night	J	P	J inside disk	P inside disk
20080717115740	Day	3.31	2.42	4.21	3.08
20080719125741	Day	3.33	2.75	4.24	3.5
20080717131241	Day	3.33	2.61	4.24	3.32
20080715214240	Night	2.46	2.08	3.13	2.65
20080718225740	Night	2.39	2.05	3.04	2.61
20080719031240	Night	2.42	2.07	3.08	2.64

Table 4. Compression ratios, 14 channels of MODIS.

Name of the Granule MOD01.A2006	250m J	250m P	500m J	500m P	1km J	1km P	Total J	Total P
174.1005.005.2006214124917	3.05	3.48	2.66	3.15	3.36	3.96	3.00	3.47
175.0915.005.2006214125006	2.67	2.85	2.45	2.84	3.17	3.88	2.67	2.98
176.0955.005.2006214124831	2.75	3.21	2.48	2.94	3.33	3.93	2.77	3.24
177.0900.005.2006215131020	2.68	3.09	2.44	2.88	3.23	3.86	2.70	3.14
179.0850.005.2006214124349	2.45	2.75	2.23	2.58	2.88	3.57	2.46	2.82
180.0930.005.2006214124332	2.74	3.20	2.49	2.96	3.41	3.99	2.77	3.25
181.1010.005.2006214124336	2.93	3.28	2.55	3.00	3.24	3.87	2.88	3.29

Table 5. Bits per sample, 14 channels of MODIS.

Name of the Granule MOD01.A2006	250m J	250m P	500m J	500m P	1km J	1km P	Total J	Total P
174.1005.005.2006214124917	5.24	4.59	6.03	5.08	4.76	4.04	5.34	4.61
175.0915.005.2006214125006	5.99	5.62	6.53	5.62	5.04	4.12	5.95	5.36
176.0955.005.2006214124831	5.81	4.99	4.80	5.44	6.44	4.07	5.78	4.93
177.0900.005.2006215131020	5.97	5.18	4.96	5.55	6.55	4.15	5.93	5.09
179.0850.005.2006214124349	6.54	5.80	5.55	6.21	7.17	4.48	6.51	5.67
180.0930.005.2006214124332	5.84	5.00	4.69	5.41	6.42	4.01	5.78	4.93
181.1010.005.2006214124336	5.45	4.88	4.94	5.34	6.27	4.13	5.55	4.86

Table 6. AVHRR

Name of the File NSS.GHRR.NN	J CR	P CR	J BpS	P BpS
D07005.S1424.E1610.B0839293.GC.nc	3.48	4.30	4.59	3.72
D07005.S0745.E0940.B0838889.GC.nc	3.76	4.63	4.26	3.45
D07006.S1719.E1906.B0840709.WI.nc	3.47	4.19	4.62	3.81
D07006.S1554.E1725.B0840708.WI.nc	3.58	4.32	4.47	3.70
D07005.S0604.E0750.B0838788.WI.nc	3.58	4.36	4.47	3.67

We also evaluated our algorithm on 14 channels (cf. 1) of MODIS selected as a proxy for the upcoming GOES-R mission. In our actual implementation for MODIS we made a small modification to increase performance of the algorithm. Like many imagers, MODIS exhibits striping due to variations between sensors. To account for this we first separated the data into per-sensor images, and compress these independently using our algorithm.

As data in the tables 3-6 show, the improved performance of our prediction based method over Jpeg2000 grows as the number of channels increases. One important thing to note is that even when there are only two channels in the MODIS 250m bands, where there is least cross spectral dependency, our prediction based compression still beats jpeg2000 by a significant amount. This is important because this data dominates the granule due to its overwhelming size. The performance on this granule alone is usually the performance for the total granule. Another important thing to note is that the algorithm performs very well on the night channels. This is significant because our previous entropy analysis showed this part of the data should allow good compression. Despite that our prior work showed that more naive methods were unable to exploit the dependencies present and achieve good compression, as we are able to now.

Finally, data from AVHRR shows once again, our prediction based method is able to exploit the band dependencies well to achieve good compression.

5. CONCLUSION

We have presented a novel but elegantly simple algorithm for imager compression. To account for different constituents of the atmosphere the earths surface, it uses a spatially varying linear regression across the bands. Despite making allowing for the spatial variation the algorithm remains simple. It can be easily adapted to distributed computation making it potentially very fast. We also have shown that by incorporating our optimal compression tree algorithm we can efficiently search the space of compression sequences to determine the best sequence for compression. Even though this is still expensive on a per granule basis our preliminary investigations seem to confirm that a single or small number of trees could be fixed for an imager and yield near optimal performance without the cost of a per granule search during encoding. Moreover though we have used a Jpeg2000 compressor here to capture the spatial correlations, this algorithm may be replaced with any number of alternatives, for example a more basic discrete cosine transform, or more finely tuned wavelet scheme, depending on the need for performance or speed. The evaluations performed on all data from the 14 channel MODIS proxy, SEVIRI and AVHRR all show benefits to using our prediction based method to exploit redundancy across spectral bands. The relative benefits are probably more significant than absolute numbers (bits per sample/compression ratios) because each data set has peculiarities that make it difficult to compare, especially compression ratios, from one imager to another.

6. ACKNOWLEDGMENTS

This compression research is managed by Roger Heymann, PE of OSD NOAA NESDIS Engineering, in collaboration with the NOAA NESDIS STAR Research Office through Mitch Goldberg, Tim Schmit, Walter Wolf.

REFERENCES

1. I. Gladkova, S. Gottipati, and M. Grossberg, "A new lossless compression algorithm for satellite earth science multi-spectral imagers," in *Proc. of SPIE, Satellite Data Compression, Communications, and Archiving III*, 2007.
2. T. J. Schmit, M. M. Gunshor, W. P. Menzel, J. Gurka, J. Li, and S. Bachmeier, "Introducting the next-generation advanced baseline imager (ABI) on GOES-R," *Bull. Amer. Meteorol. Soc.* **86**, pp. 1079–1096, 2005.
3. M. Klimesh, "Low-complexity adaptive lossless compression of hyperspectral imagery," in *Proc. of SPIE, Satellite Data Compression, Communications, and Archiving II*, **6300**, 2006.
4. M. D.Adams, "The JPEG-2000 Still Image Compression Standard."
5. M. D. Adams and F. Kossentini, "JasPer: A software-Based JPEG-2000 Codec Implementation," *Proc. of IEEE International Conference on Image Processing* **2**, pp. 53–56, Oct 2000.
6. C. 120.1-G-1, "Image data compression," in *Green Book. Issue 1*, (CCSDS Publications: http://public.ccsds.org/publications/GreenBooks.aspx), June 2007.
7. "http://www.ece.uvic.ca/ mdadams/jasper/."
8. M. Grossberg, I. Gladkova, and S. Gottipati, "An analysis of the information dependence between modis emissive bands," in *Proc. of SPIE, Satellite Data Compression, Communications, and Archiving IV*, 2008.
9. T. J. Schmit, J. J. Gurka, M. M. Gunshor, and J. Li, "The ABI on the GOES-R series," in *5th GOES Users' Conference*, (New Orleans, LA), January 2008.
10. "http://cimss.ssec.wisc.edu/goes/abi."