Compression of AIRS data using Empirical Mode Decomposition *

I. Gladkova†, L. Roytman†, M. Goldberg‡, and J. Weber†

†NOAA-CREST City College of New York, New York, NY;
‡NOAA-NESDIS, Washington, DC

ABSTRACT

In this paper, we consider an application of the Empirical Mode Decomposition (EMD) introduced by Norden E. Huang in 1996 to the compression of 3D hyperspectral sounding data. The EMD is a new data analysis method which is based on expansion of the data in terms of Intrinsic Mode Functions (IMF). These IMFs are based on and derived from the data set. Since EMD adaptively represent the signal as a sum of "well behaved" amplitude/frequency modulated components, we found it very well suited for the whitening part of the compression scheme.

Keywords: Compression, Empirical Mode Decomposition, Intrinsic Mode Functions, Karhunen-Loève

1. INTRODUCTION

In this paper we present a lossless algorithm for compression of the signals from NOAA's environmental satellites. The project's aim is the design, analysis, and implementation of compression techniques that are suitable for the next-generation GOES-R instrument. We are using current spacecraft to simulate data from the upcoming GOES-R instrument and focusing on Aqua Spacecraft's AIRS instrument in our case study.

The AIRS is a high resolution instrument which measures infrared radiances at 2378 frequencies ranging from 3.74-15.4 μm. The AIRS takes 90 measurements as it scans 48.95 degrees perpendicular to the satellite's orbit every 2.667 seconds. We use Level 1A digital counts) data granules, which represent 6 minutes (or 135 scans) of measurements. Therefore, our data set consists of a 90x135x2378 cube of integers ranging from 12-14 bits.

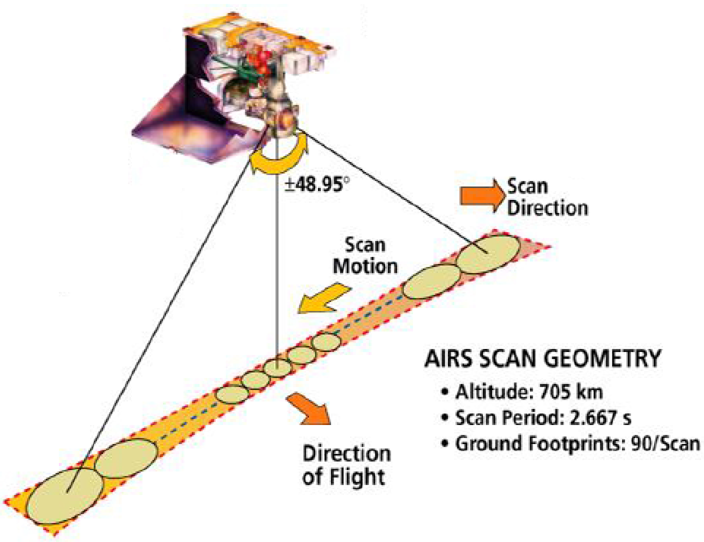

Figure 1. AIRS Scan Geometry (graphic courtesy of NASA JPL).

*Sponsored by NOAA/NESDIS under Tim Schmit (ORA), Roger Heymann (OSD) HES Compression Group

Note that noise in the channels introduce added complexity in compression. Therefore, in practice, we utilize only 1502 out of 2378 channels picked by NOAA for their favorable characteristics. Otherwise, we would need to add an additional step to detect these channels prior to running our compression algorithm.

2. OVERVIEW OF COMPRESSION SCHEME

The algorithm consists of the following steps:

1. Channel Partitioning
2. Whitening
3. Projection
4. Estimated entropy coding of the residuals

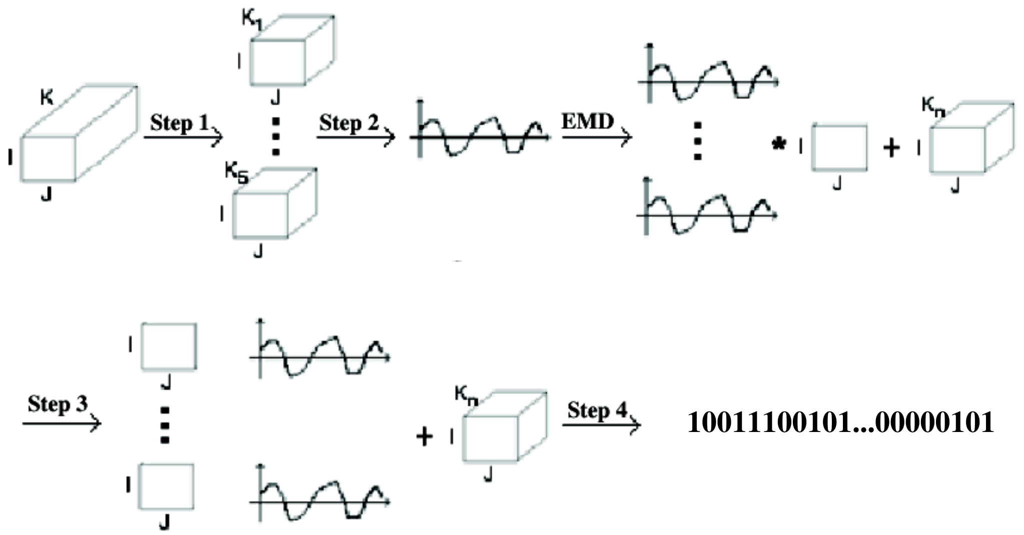

Figure 2. The stages of the compression scheme can be visualized in the above diagram.

In what follows, we give a brief description of each step of the algorithm.

Step 1. During the first stage, the original $90 \times 135 \times 1502$ granule $A = \{a_{ijk}\}$, where $i = 1 : 90$, $j = 1 : 135$, $k = 1 : 1502$, is subdivided into 3 bands of size K_n, $K_1 = 514$, $K_2 = 692$, $K_3 = 296$, and $\sum_{n=1}^{3} K_n = 1502$. The division accounts for the fact that the range of the digital counts a_{ijk} varies (12, 13, and 14 bits) with channel index k. After this subdivision, each of the resulting $90 \times 135 \times K_n$ granules $Y = \{y_{ijk}\}_{k=1}^{K_n}$ will be processed independently.

Step 2. The purpose of the whitening stage is to transform the data so that its distribution is as close to normal as possible. The cost of this transformation is added memory utilization (due primarily to record-keeping data structures needed for the decompression procedure). There are several theorems[2] which state an optimality (in some sense) of the Karhunen-Loève transform provided a normal distribution of the data. Therefore, if the whitening stage is ”cheap” in terms of memory utilization, overall transformation (steps 1-4) is nearly optimal. The ratio of memory needed for step 2 with respect to the total memory occupied by the original granule varied in the 10 considered granules, but on average was approximately 1288:1, i.e. about 0.037% of the memory used by the original granule is needed to force the data to have an almost normal distribution. We felt that this was a relatively low cost for a known optimal approach and so have proceeded along that course. This step is based on Empirical Mode Decomposition. A brief overview of the EMD and its role in the step 2 of our algorithm are given in section 3.

Step 3. As alluded to in the overview of step 2, we apply the Karhunen-Loève transform which is known to have the smallest average distortion when approximating a class of functions by their projection on L orthogonal vectors chosen a priori.[2] If the goal was to develop a lossy compression algorithm, our primary cost would be the error and we would have to define a notion of acceptable average error to determine the number L of projection vectors. Since we design a lossless compression algorithm, our primary cost is memory utilization and we need to define a notion of minimal memory space to save both projection vector coefficients and the residuals of the projection. The number L_n of the projection vectors is chosen to address this concern. Thus, during this step the global part of the information from each of the 3 bins is saved in L_n packets, where each packet contains a 90×135 image of the quantized coefficients $\hat{C}_l$ and a $1 \times K_n$ quantized projection vector $\hat{v}_l$, resulting in $L(90 \cdot 135 + K_n)$ elements in total. The residuals are saved separately through the last (approximated entropy coding) stage of our lossless compression algorithm.

Step 4. After step 3, we have $90 \times 135 \times K_n$ granules of residuals that are approximately normally distributed but have a lower entropy (due to properties of Karhunen-Loève transform). In our computations, the entropy on the average decreased from 8.35 to 3.5 during step 3. Therefore, the lower bound on the number of bits per residual entry r_{ijk} is 3.5 bits. We build our Huffman codebook[3] based on a normal distribution with variance computed from the residuals, rather than an actual codebook. This mitigates issues with errors during transmission, and also makes our program slightly more efficient.

3. EMPIRICAL MODE DECOMPOSITION

In this section we will give a brief overview of the Empirical Mode Decomposition (EMD) introduced by Norden E. Huang.[1] The basic idea is to decompose the given (possibly non-linear and non-stationary) signal $y(t)$ as

$$y(t) = \sum_{k=1}^{K} y_k(t),$$

where the components $y_k(t)$ in some sense are independent of each other and contain both frequency and amplitude variations.

The main idea behind the EMD technique is to represent signal $y(t)$ as a set of intrinsic mode functions. A function is an intrinsic mode function if: (1) the number of extrema and the number of roots are either equal or differ at most by one. (2) at any point, the mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero. This class of functions is well-suited for describing instantaneous frequency even in the non-linear case.

An iterative procedure for decomposing a signal $y(t)$ into IMFs has been presented by Huang,[1] and can be described as the following recursion:

1. Find all local extrema of $y(t)$
2. Find the "envelope" of $y(t)$ by interpolating through all local maxima to obtain $y_{\max}(t)$ and through all local minima to find $y_{\min}(t)$
3. Compute the average $m(t) = (y_{\min}(t) + y_{\max}(t))/2$
4. Find "fast oscillations" or details of the function by subtracting the average $m(t)$: $h(t) = x(t) - m(t)$.
5. Repeat steps 1)-4) on $m(t)$ until $m(t)$ is a monotonic function.

The last $m(t)$ is called the trend of the input signal $y(t)$.

We should note that step 2) is based on interpolation and therefore has infinitely many solutions. Which of these solutions applies is still an open problem. In practice, splines are often used as an interpolator and steps 2)-4) are repeated (so called sifting process) until function $h(t)$ is an IMF.

We will consider the given $90 \times 135 \times K_n$ granule $Y = \{y_{ijk}\}$ as a collection of $90 \cdot 135$ discrete signals $y_{ij}[k]$ and use EMD to decompose an average

$$y[k] = \sum_{i=1}^{90} \sum_{j=1}^{135} y_{ij}[k]/(90 \cdot 135)$$

into a set of intrinsic mode functions $\{h_n\}$ and the trend m (see EMD stage in fig. 2). If the components of this decomposition are linearly independent, they can be used as the basis of subspace H spanned by vectors h_n and m. The discrete signals $y_{ij}[k]$ then can be projected on the subspace H:

$$y_{ij} = \sum_{n=1}^{N} (y_{ij}, h_n)\tilde{h}_n + r\,,$$

where $(y_{ij}, h_n) = \sum_{k=1}^{K} y_{ij}[k]h_n[k]$, $\{\tilde{h}_n\}$ is a dual Riesz basis, i.e. $(h_n, \tilde{h}_k) = \delta[n-k]$, $\delta[k]$ is the discrete Dirac function, and r is the residual vector. We have observed that AIRS data projects exceptionally well on the subspace spanned by EMD components leaving almost normally distributed residuals $r_{ij}[k]$.

The EMD technique has been applied with success in a number of applications, but, although it is very simple in its principle, it lacks theoretical fundamentals. There have been attempts at theoretical expositions of the EMD method for some particular cases (e.g. for uniformly distributed noise[4] and fractal Gaussian noise[5]). In general, however, the EMD method is defined as the output of an iterative algorithm and does not have analytical formulations. Consequently, we are drawing conclusions related to EMD based on numerical observations, which show the strong suitability of EMD as a whitening tool for AIRS data (as is shown in the following figures). In all the figures that follow, the original distribution of each of the three partitions are given in the first row and the EMD-transformed distributions are given in the second row.

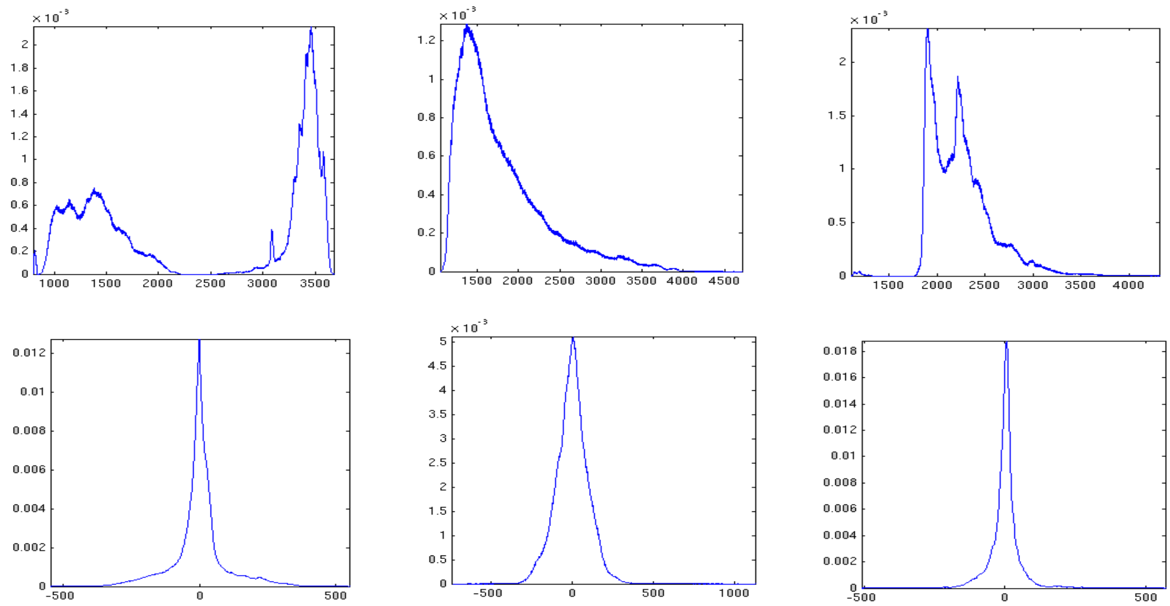

Figure 3. Granule 9

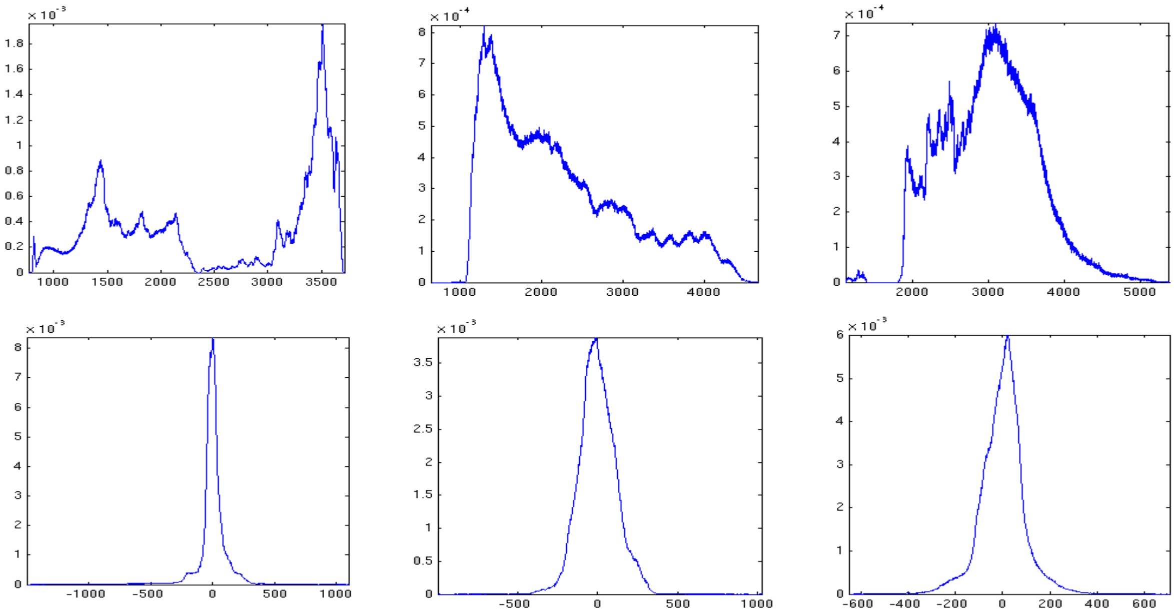

Figure 4. Granule 151

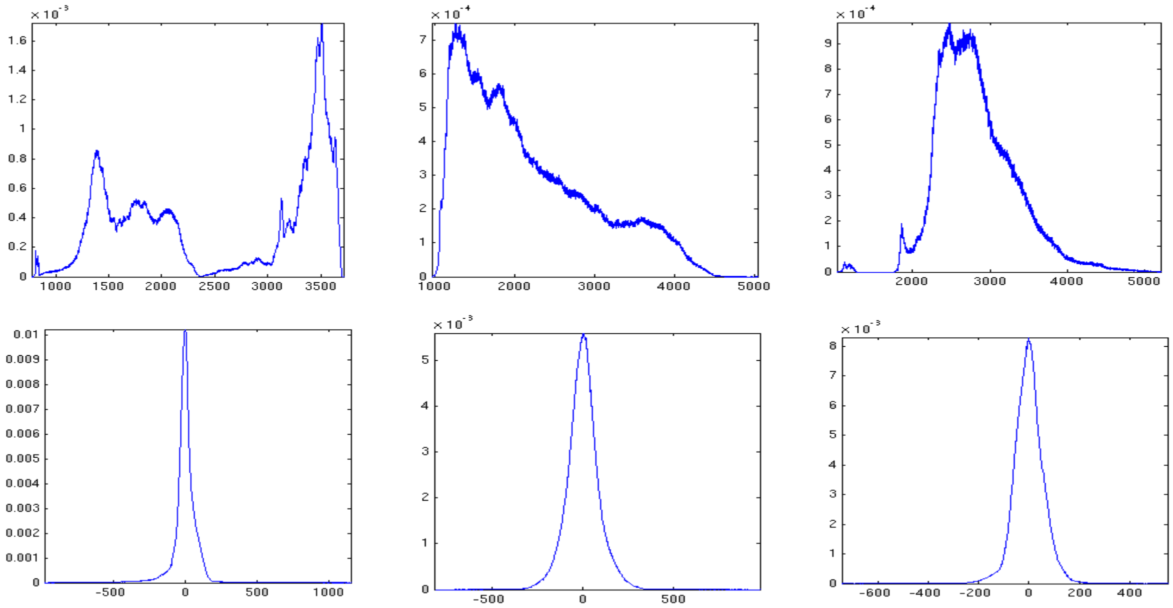

Figure 5. Granule 193

4. APPLICATION OF ALGORITHM AND RESULTS

In this section, we review the details of our compression algorithm using channels indexed 1206-1502 of granule 182 (Asia, Nighttime) as an example.

4.1. Whitening by EMD Method

As discussed in the previous section, we use IMFs to extract oscillatory features from our input signal $y_{ij}[k]$.

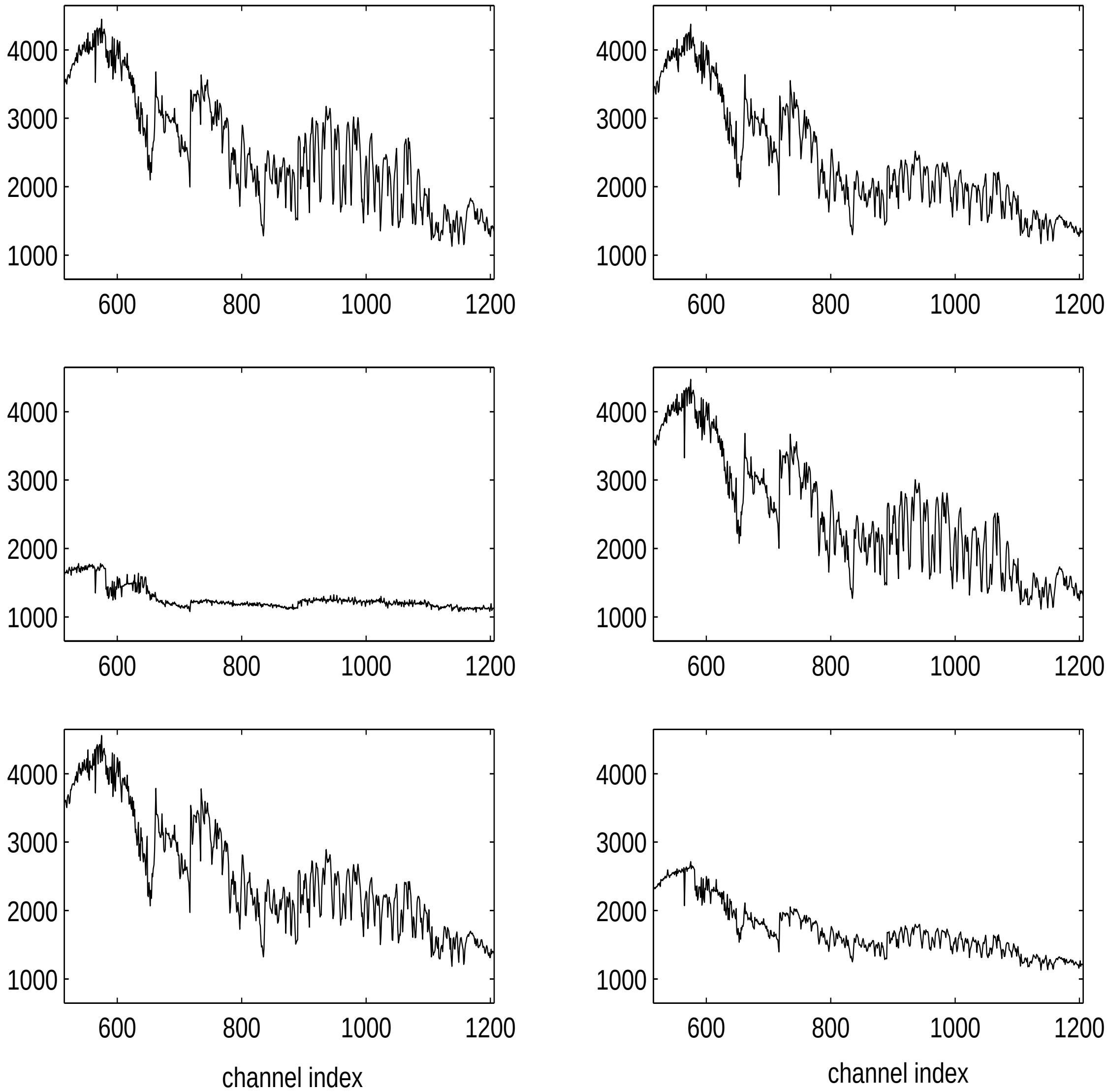

Figure 6. Oscillatory features shared by signals $y_{ij}[k]$.

Because we are not interested in all of the lower oscillations separately, we restrict ourselves to six IMF components combining the last two IMFs $h_{n-1}[k]$ and $h_n[k]$ into a single function defined simply as $h_{n-1}[k] + h_n[k]$ when necessary. Otherwise, the process is exactly the same as the sifting algorithm described in section 3. The result is a series of "well-behaved" frequency and amplitude modulated functions $h_n[k]$:

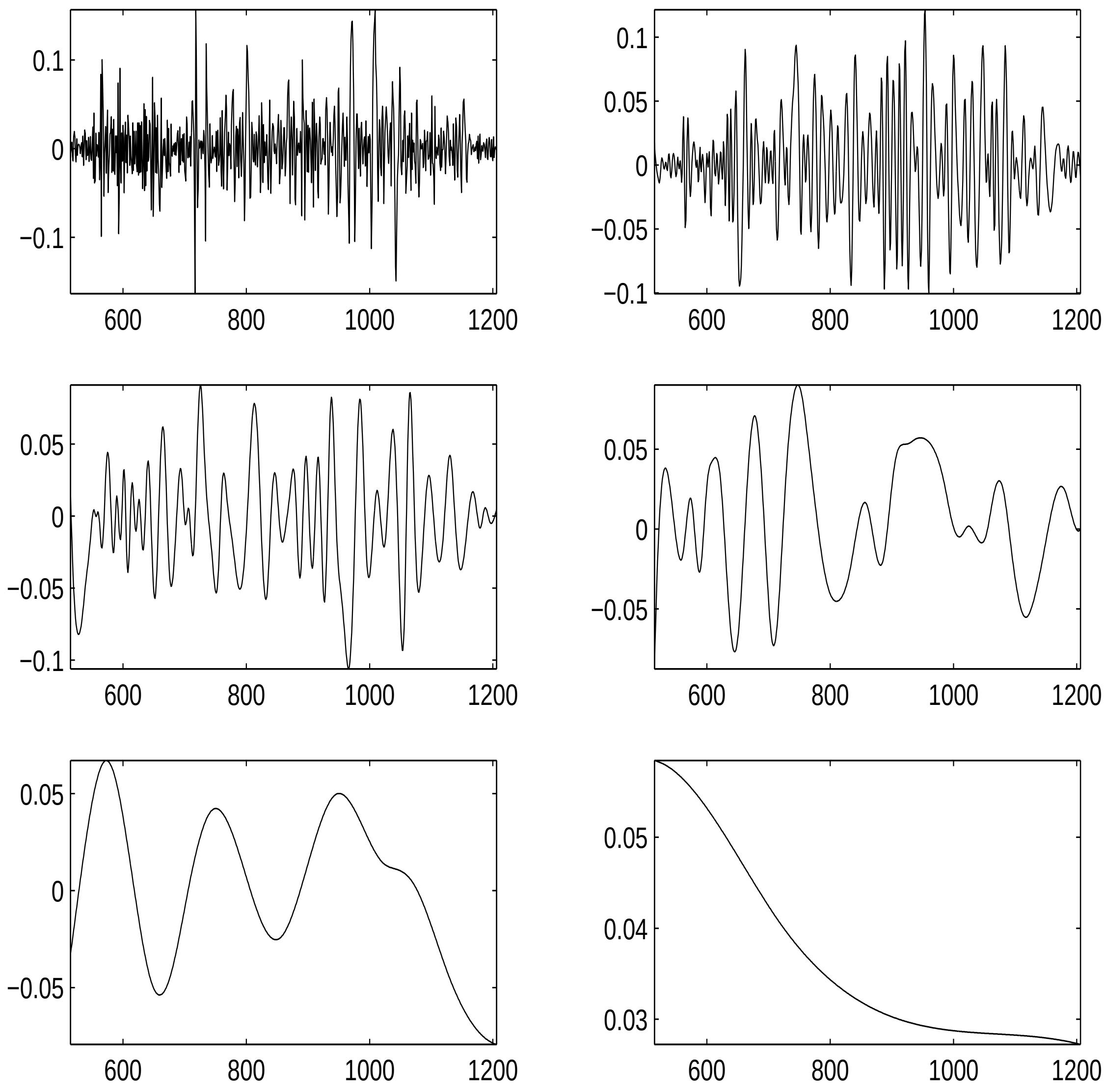

Figure 7. Granule 182 EMD Components.

We then project the given granule (consisting of all discrete signals $y_{ij}[k]$) onto normalized EMD components $\tilde{h}_n[k]$. For each of the n EMD components, the projection coefficients $a_{ij}^{(n)} = \sum_{k=1}^{K} y_{ij}[k]\tilde{h}_n[k]$ can be arranged in a 2D array $A^{(n)} = \{a_{ij}^{(n)}\}$ which are displayed in fig. 9:

The average $y[k]$ will be saved for use in the decompression procedure to compute the normalized EMD components h_n's as well as their duals $\tilde{h}_n$'s. The projection coefficients $a_{ij}^{(n)}$ are generally speaking real-valued numbers and in order to be saved in the finite number of bits, must be quantized. The residual granule consisting of

$$r_{ij}[k] = y_{ij}[k] - \sum_{n=1}^{N} \hat{a}_{ij}^{(n)} \tilde{h}_n[k],$$

where $\hat{a}_{ij}^{(n)}$ are the quantized coefficients, is our "whitened" data set as was shown in figures 3 to 5. Residual granule $R_1 = \{r_{ijk}\}$ is the input to the next step.

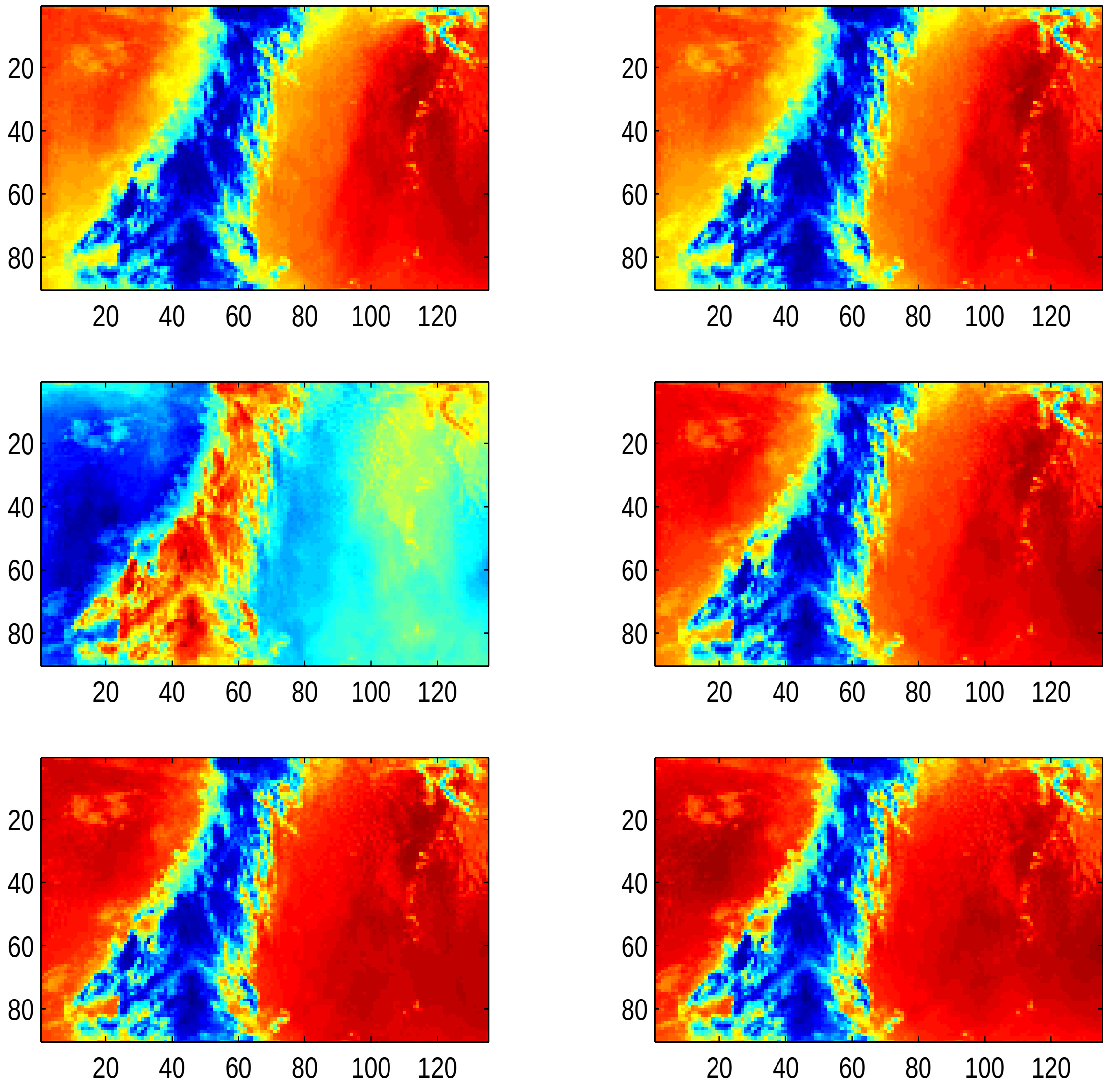

Figure 8. Image of quantized coefficients $\hat{a}_{ij}^{(n)}$ for $n = 1 : 6$.

4.2. KarhunenLoève Transform

Karhunen - Loève basis is an orthogonal basis $\{v_n\}_{n=1}^{N}$ that diagonalizes the symmetric, positive-definite covariance matrix $C = E[r_{ij}r_{kl}]$, where E is the expected value and r_{ij} is the ij'th vector from the residual granule $R1$. Each vector r_{ij} in $R1$ is a zero-mean due to the transformation in the step 2. The vectors v_n are the principal directions (or eigenvectors) of C. Karhunen - Loève (also known as PCA) is a very well-known transform used in compression as well as other applications and a set of theorems about optimal properties of this transform, for example, can be found in.[2] In our experiments, we have used 8, 22, and 13 principal components for channels 1-514, 515-1206, and 1207-1502 pristine channels respectively. Below are eight out of 22 computed projection coefficients quantized and arranged in 2D arrays as in the previous section.

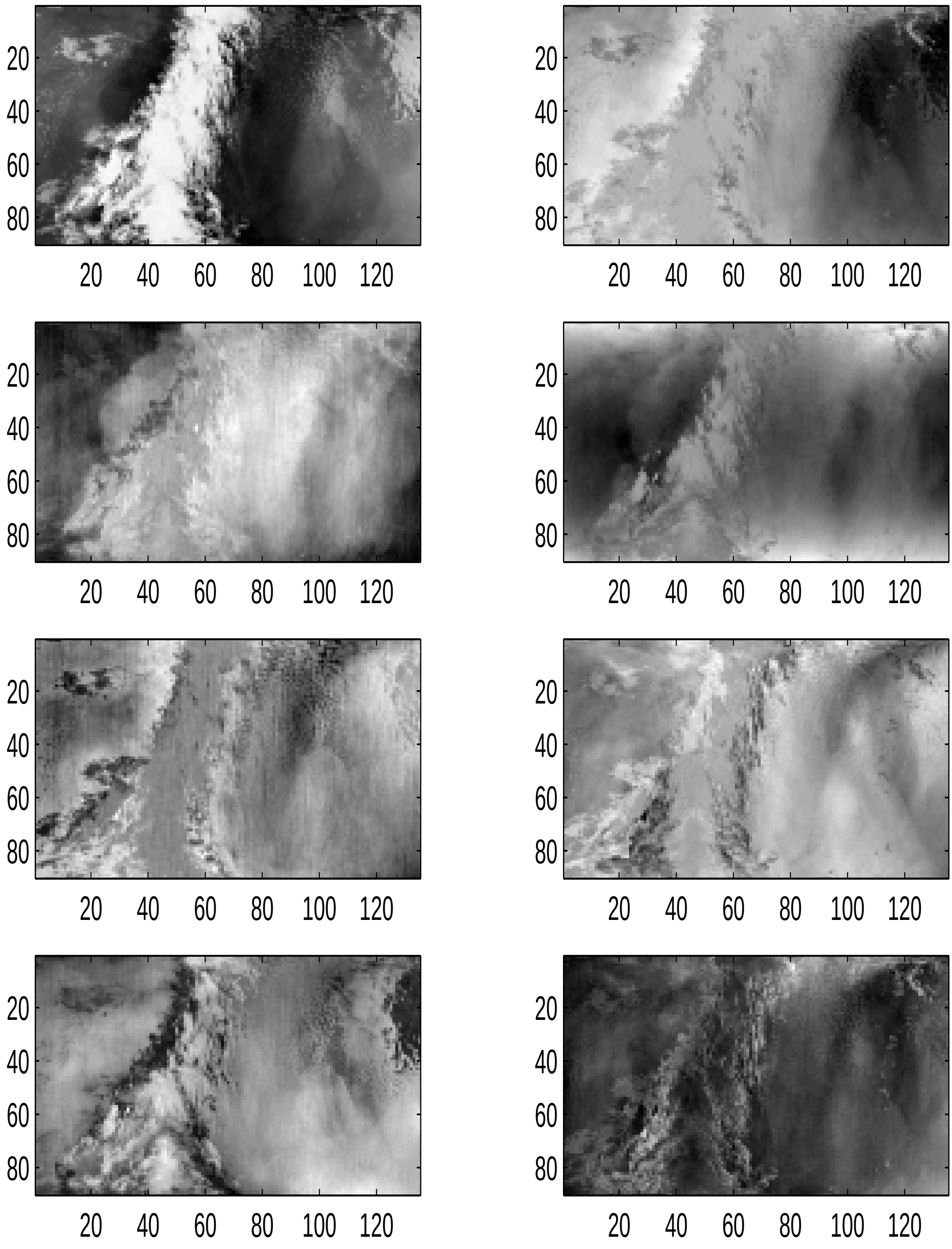

Figure 9. Granule 182 Quantized KarhunenLoève Coefficients.

We should note at this point that arrays of the coefficients, as can be seen in figs. 9 and 8, comprise structured images that can be further compressed. Compression of these images were not carried out in our work to address issues of potential error propagation during transmission. This can be included if desired, and if properties of the communication medium allow, to further increase the compression ratio.

4.3. Huffman Coding

We denote the probability that the (integer) element of the granule R_2 is equal to an integer number n by $p_n = Pr\{r_{ijk} = n\}$ and the range of values in R_2 by $[N_{\min}, N_{\max}]$, where $N_{\min}$ and $N_{\max}$ are the smallest and largest values in R. The Shannon Theorem[6] proves that the entropy

$$\mathcal{H} = -\sum_{n=N_{\min}}^{N_{\max}} p_n \log_2 p_n$$

is a lower bound of the average number of bits per symbol used to encode the values in R_2. The lower entropy bound is nearly reachable with an optimized prefix code.The Huffman algorithm[3] is an example of such a prefix code that minimizes the average bit rate

$$\sum_{n=N_{\min}}^{N_{\max}} p_n l_n,$$

where l_n is the length of the nth code symbol. As was pointed out earlier, the probability distribution p_n is very close to normal, therefore the Huffman codebook l_n can be constructed on the approximation $\mathcal{H} \approx e^{-n^2/(2\sigma^2)}$, where σ^2 is the variance of the residual granule R_2. This way the only information needed to reconstruct the codebook is the variance and the range $[N_{\min}, N_{\max}]$.

4.4. Results

In the table below, we give resulting compression ratios for our 10 test granules.

Granule	Location	Ratio
9	Pacific Ocean, Daytime	3.1653
16	Europe, Nighttime	3.1860
60	Asia, Daytime	3.1118
82	North America, Nighttime	3.2166
120	Antarctica, Nighttime	3.1169
126	Africa, Daytime	3.1065
129	Arctic, Daytime	3.2159
151	Australia, Nighttime	3.0603
182	Asia, Nighttime	3.0021
193	North America, Daytime	3.0827

REFERENCES

1. N.E. Huang, Z.Shen, S.R. Long, M.L. Wu, H.H. Shih, Q. Zheng, N.C. Yen, C.C. Tung and H.H. Lui, The empirical mode decomposition and Hilbert spectrum for nonlinear and non-stationary time series analysis, *Proc. Roy. Soc. London A*, Vol. 454, pp. 903–905, 1998
2. S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 1998
3. D. Huffman, A method for the construction of minimum redundancy codes. *Proc. of IRE*, 40:1098–1101, 1952
4. Z. Wu, N.E. Huang, A study of the characteristics of white noise using the Empirical Mode Decomposition, *COLA Technical Reports*, Jan. 2003, 27pp.
5. P. Flandrin, G. Rilling, P. Conçalvès, Empirical Mode Decomposition as a Filter Bank, *IEEE Signal Proccessing Letters*, 11(2): 112–114, 2004
6. C.E. Shannon, Communications in the presence of noise, In *Proc. Of the IRE*, vol. 37, pp.10-21, Jan.1949