

Software Implementation of CCSDS Recommended Hyperspectral Lossless Image Compression

Dharam Shah

Department of E&C Engineering, VGEC, Ahmedabad, Gujarat, India
E-mail: shahdharam7@yahoo.in

Kuhelika Bera

Space Application Center (SAC), Indian Space Research Organization (ISRO), Ahmedabad, Gujarat, India
E-mail: kuhelika@sac.isro.gov.in

Sanjay Joshi

Department of E&C Engineering, VGEC, Ahmedabad, Gujarat, India
E-mail: sdjoshi_74@yahoo.co.in

Abstract—HyperSpectral Imagers (HySI) are used in the spacecraft or aircrafts to get minute characteristics of target element through capturing image in a large number of narrow and contiguous bands. HySI data represented as data cube with two dimensions representing spatial distribution and third dimension providing band information is huge in volume and challenging task to handle. Hence onboard compression becomes a necessary for optimal usage of onboard storage and downlink bandwidth. CCSDS recommended 123.0-B-1 standard [2] has been released with onboard compression scheme of hyperspectral data. The scheme is based on Fast Lossless algorithm and consists of two main functional blocks namely Predictor and Encoder. Predictor algorithm can be implemented in two modes ‘Full Neighborhood Oriented’ and ‘Reduced Column Oriented’. Encoder algorithm also defines two options ‘sample-adaptive’ and ‘block-adaptive’. We have developed a MATLAB based model implementing the compression scheme with all options defined by the standard. Decompression model is also developed for getting back actual data and end to end verification. Four sets of HySI data (AVIRIS, Hyperion, Chandrayan-1 and FTIS) have been applied as input to the developed model for evaluation of the model. Compression ratio achieved is between 2 to 3 and lossless compression is ensured for each set of data as Mean Square Error (MSE) is zero for all hyperspectral images. Also visual reconstruction of decompressed data matches with original ones. In this paper we have discussed algorithm implementation methodology and results.

Index Terms—Fast Lossless, MATLAB, Hyperspectral, Image Compression.

I. INTRODUCTION

Hyperspectral images provide information in large

number of narrow, contiguous spectral bands. It is used for mineralogy, pollution monitoring, atmospheric study, astronomy and many more. Hyperspectral images have three dimensions. Two of which are of Spatial and third one is spectral. Each scene is captured on all the spectral bands. Therefore each image has more information than traditional images. The very high amounts of data produced by such imagery and the bandwidth constraints of satellites, requires the use of on-board image compression techniques. There are various methods of compression as described in [6, 15]. There are methods for Hyperspectral image compression like Low Complexity (LOCO-I) [8], Two Dimensional Context Adaptive Lossless Inter-band Compression (2D-CALIC) [9], Three Dimensional Context Adaptive Lossless Inter-band Compression (3D-CALIC) [10], Modified Context Adaptive Lossless Inter-band Compression M-CALIC [11] Look-Up Tables (LUT) [12] (using a single LUT), Locally Averaged Inter-band Scaling (LAIS)-LUT [13] and LAIS-Quantized LUT (LAIS-QLUT) [14] (LAIS-LUT and LAIS-QLUT using two LUTs). Various Hardware implementations are shown in references [16, 17]. Main technique of all image compression algorithms is to remove redundancy. Hyperspectral images have spectral redundancy other than spatial redundancy. Hyperspectral images differ from traditional images in that they require specific coding techniques that exploit its spectral redundancy to achieve competitive compression performance.

The Consultative Committee for Space Data Systems (CCSDS) composed of the world’s major space agencies, is a multi-national forum for the development of communications and data systems standards for spaceflight [1]. The Multi-spectral and Hyperspectral Data Compression (MHDC) [7] working group of the CCSDS has developed the CCSDS-123.0-B-1 standard [2], which is based on the fast lossless (FL) compression algorithm and intended for onboard lossless coding of

multi- and hyperspectral imagery. Multispectral images have narrow bands over a discrete spectral range, and having less number of bands than hyperspectral images. This recommended standard achieves state-of-the-art compression performance on images captured by a wide variety of hyperspectral/multispectral sensors. It's advantage of low computational complexity facilitates implementation in onboard resource constrained scenarios.

Compression System based on this recommended standard is composed of prediction stage and entropy-coding stage. Prediction Stage has Full/Reduced Prediction mode, Neighbour/Column oriented local sums to adjust best compression for different types of multispectral and hyperspectral imagers. Use of reduced mode in combination with column- oriented local sums gives good compression performance for raw or uncalibrated input images from push-broom imagers. The use of full mode in combination with neighbourhood-oriented local sums gives good compression performance for whiskbroom imagers, frame imagers and calibrated imagery. There are two methods of Encoding: Sample-adaptive & Block-adaptive. Sample adaptive encoding is done on sample by sample whereas block adaptive encoding is done on group of samples using Golomb-rice algorithm [18].

In this Paper, Section II introduces MATLAB implementation of algorithm in detail with various stages. Section III presents test results. The conclusion is given in Section IV.

II. MATLAB IMPLEMENTATION OF ALGORITHM

CCSDS 123.0-B-1 recommended standard defines a payload lossless data compressor that can be applied to hyperspectral & multispectral imagers and sounders. Input image which is three dimensional cube is used as input to compressor block. Compressor gives compressed image which is variable bit-stream. Fig. 1 shows Basic block diagram of the scheme.

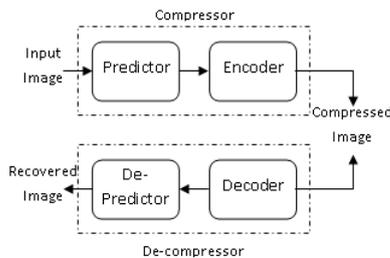


Fig. 1. Basic block diagram of the scheme

Compressor has two functional units: a predictor and an encoder. De-compressor does reverse process of compression. De-compression is required to get original image. De-compressor has two functional units: a decoder and a de-predictor.

A. Predictor

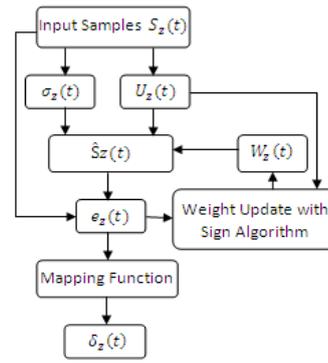


Fig. 2. Flow chart for Predictor

Predictor uses an adaptive linear prediction method to predict the value of each image sample. $S_{z,y,x}$ or $S_z(t)$ is current sample value having D number of bits. Here, x & y represents spatial coordinates, z represents spectral dimension and index t is defined as, $t = y.N_x + x$. Where, N_x represents image width. Prediction of sample $S_{z,y,x}$ is $\hat{S}_{z,y,x}$ is derived by computing 'local sum' $\sigma_z(t)$ and 'local difference' $U_z(t)$ with weight vector $W_z(t)$. Weight vector can be initialised by default and custom method[2]. In default initialization method, Weight vector is initialized by particular default value. A weight initialization vector in custom based initialization method might be selected based on instrument characteristics or training data, or might be selected based on a weight vector from a previous compressed images. Therefore we have selected default initialization method of weight vector. These parameters are dependent on the values of nearby samples in the current spectral band and P previous (i.e., lower-indexed) spectral bands. The user-specified parameter P (range: $0 \leq P \leq 15$) determines the number of previous spectral bands used for prediction. The prediction residual $e_{z,y,x}$ is the difference between the predicted $\hat{S}_{z,y,x}$ and actual $S_{z,y,x}$ sample values. The prediction residual is then mapped to an unsigned integer $\delta_{z,y,x}$ that can be represented using the same number of D bits as the input data sample. These mapped prediction residuals is the predictor output. Flow chart of Predictor is shown graphically in fig. 2.

B. Encoder

A compressed image consists of a header followed by a body as shown in fig. 3. The variable-length header encodes image and compression parameters. Header consists of image, predictor & encoder metadata. The body consists of losslessly encoded mapped prediction residuals from the predictor. Encoder encodes the mapped prediction residual by Sample adaptive entropy coding or Block adaptive entropy coding.

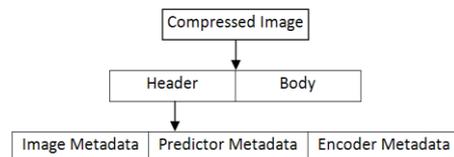


Fig. 3. Compressed image structure

➤ Sample adaptive encoder

The original FL (Fast Lossless) algorithm uses an adaptive coding approach using Golomb-Power-of-2(GPO2) codes. Under the sample-adaptive entropy coding option, each mapped prediction residual $\delta z(t)$ shall be encoded using a variable-length binary codeword. The selection of the code used to encode $\delta z(t)$ is based on the values of the adaptive code selection statistics which consist of an accumulator and a counter that are adaptively updated during the encoding process. Initial counter (γ_0) and accumulator (K) value is in the range of $1 \leq \gamma_0 \leq 8$ and $0 \leq K \leq D-2$ respectively. Here, we have taken $\gamma_0 = 1$ and $K=5$. The interval at which the counter and the accumulator are rescaled is controlled by the user-defined rescaling counter size parameter γ^* , which shall be an integer in the range: $\max\{4, \gamma_0 + 1\} \leq \gamma^* \leq 9$.

➤ Block adaptive encoder

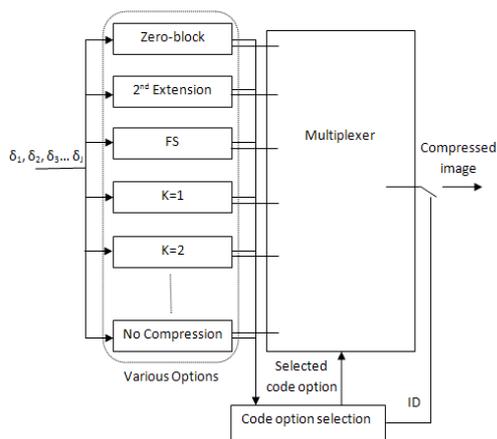


Fig. 4. Block adaptive encoder

The block-adaptive coder, which also makes use of GPO2 codes, is the Rice coding algorithm as formalized in the CCSDS 121.0-B-2 standard [3]. Block adaptive coder encodes samples on group of samples called Block. The block size parameter J shall be equal to 8, 16, 32, or 64. Shorter values of the block length parameter J allow faster adaptation to changing source statistics. Longer block lengths often offer improved overall compression effectiveness because of reduced overhead. So, we have taken $J=16$. The reference sample interval parameter, r shall be a positive integer not larger than 4096. We have taken $r=1024$. Packetized telemetry is used to limit error propagation as described in reference [3]. The Rice algorithm uses a set of variable-length codes options like Zero-block, 2nd extension, Fundamental Sequence, No-compression as shown in fig. 4 to achieve higher compression. Each code is nearly optimal for a particular geometrically distributed source. Variable-length codes compress data by assigning shorter codeword to symbols that are expected to occur with higher frequency. By using several different codes and transmitting the code identifier, algorithm can adapt to sources from low entropy (more compressible) to high entropy (less compressible).

C. Decoder

Parameters used in the Predictor & Encoder are in the header of the compressed image. First, all the parameters are decoded for decoding and de-prediction.

➤ Sample adaptive decoder

Decoding for sample adaptive encoder is simply reversing the procedure of encoding. By calculating number of zeros and ones, we can calculate accumulator and counter value. From that we can calculate the mapped prediction residual $\delta_z(t)$.

➤ Block adaptive decoder

Block adaptive encoder encodes the block with some ID bits. From that ID bits, we get to know about which option is selected for encoding at the decoder side. Decoding for block adaptive encoding options like zero-block, second-extension and no-compression are briefly given in CCSDS 120.0-G-3 informational report [4].

D. De-predictor

De-predictor consists of postprocessor and de-predictor stage. De-predictor stage reconstructs the original samples from previous samples similar to the approach used in DPCM decoding. First Sample $S_z(1)$ of each band is not coded. Second sample $S_z(2)$ is decoded from first sample $S_z(1)$ value and second sample mapped prediction residual $\delta_z(t)$. Similarly all sample values are calculated. The postprocessor reverses the mapping function & gives the predicted value $S_z(t)$.

III. VERIFICATION

CCSDS 123.0-B-1 recommended standard is developed in MATLAB 2012a. Four hyperspectral images taken from different sensors are applied to algorithm and verified.

Dataset-1: It is from AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) calibrated radiance images which is most widely used images for benchmarking hyperspectral image compression algorithms[5]. It has solar reflected spectrum from 400 nm to 2500 nm at 10 nm intervals. We have taken 16-bit Yellowstone calibrated scene 11 raw image of size 512 lines x 677 samples x 224 bands. Three adjacent bands of Yellowstone calibrated scene 11 cropped images of size 256*256 are shown in fig. 10.

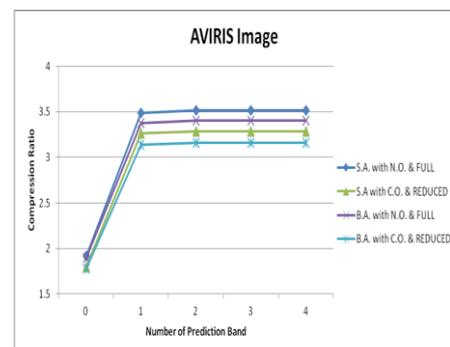


Fig. 5. Compression Ratio vs Number of Prediction band for Dataset1

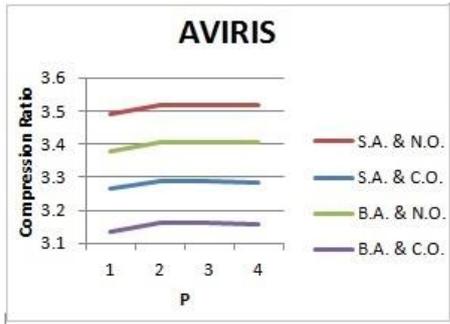


Fig. 5(A). Compression Ratio vs Number of Prediction band for Dataset1

Dataset-2: It is from Hyperion Level 0 images which are provided by the EO-1 Mission, NASA [5]. Hyperion imager has 400 - 2400 nm spectral range and 10 nm spectral resolution. We have taken 12-bit Lake Monona un-calibrated image having width 256 and height 3176 with 242 spectral channels. In the files provided by the hyperion imager, each sample is stored as a 2-byte unsigned integer in little-endian byte order, samples arranged in Band Interleaved by Pixel order. Three adjacent bands of Lake Monona scene cropped images of size 256*256 are shown in fig. 11.

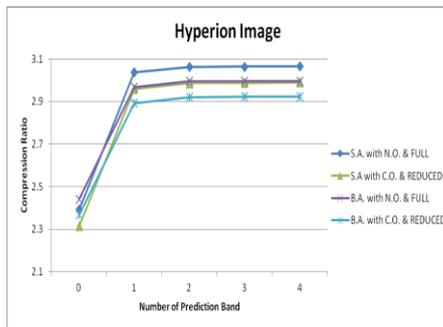


Fig. 6. Compression Ratio vs Number of Prediction band for Dataset2

Dataset-3: It is AHYSI (Airborne Hyper Spectral Imager) chandryan-1(CH-1) experimental data having 64 band data with 512 lines. Each line has 256 samples. AHYSI data are in RAW format. It is in band sequential Order. Its bands are in 400-950 nm range with 8.6 nm spectral resolution. Three adjacent bands of dataset-3 are shown in fig. 12.

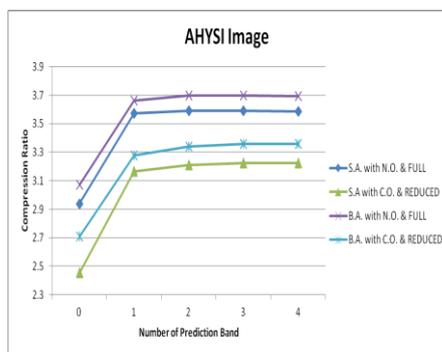


Fig. 7. Compression Ratio vs Number of Prediction band for Dataset3

Dataset-4: It is FTIS (Fourier Transform

ImagingSpectrometer) data which is an experimental data. FTIS data is 32 bands data with 288 lines. Each line has 336 samples. FTIS data are in TEXT format. It is in band sequential Order. Its bands are in visible range. Three adjacent bands of dataset-4 are shown in fig. 13.

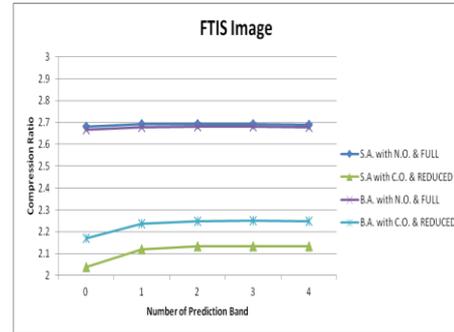


Fig. 8. Compression Ratio vs Number of Prediction band for Dataset4

Table 1. Compression ratio for four images

| En. Type | Local Sum Type | Pre. Mode | Pre. band | AVIRIS (Dataset-1) | Hyperion (Dataset-2) | CH-1 (Dataset-3) | FTIS (Dataset-4) |
|----------|----------------|-----------|-----------|--------------------|----------------------|------------------|------------------|
| S.A. | N.O. | Full | 0 | 1.9200 | 2.3930 | 2.9400 | 2.6802 |
| | | | 1 | 3.4910 | 3.0375 | 3.5736 | 2.6910 |
| | | | 2 | 3.5181 | 3.0629 | 3.5898 | 2.6932 |
| | | | 3 | 3.5178 | 3.0641 | 3.5887 | 2.6919 |
| | C.O. | Reduced | 0 | 1.7958 | 2.3154 | 2.4562 | 2.0386 |
| | | | 1 | 3.2652 | 2.9595 | 3.1665 | 2.1209 |
| | | | 2 | 3.2875 | 2.9859 | 3.2110 | 2.1331 |
| | | | 3 | 3.2868 | 2.9882 | 3.2242 | 2.1345 |
| B.A. | N.O. | Full | 0 | 1.8989 | 2.4415 | 3.0704 | 2.6657 |
| | | | 1 | 3.3768 | 2.9688 | 3.6610 | 2.6783 |
| | | | 2 | 3.4045 | 2.9944 | 3.6959 | 2.6813 |
| | | | 3 | 3.4046 | 2.9956 | 3.6955 | 2.6802 |
| | C.O. | Reduced | 0 | 1.7768 | 2.3673 | 2.7092 | 2.1694 |
| | | | 1 | 3.1365 | 2.8921 | 3.2772 | 2.2376 |
| | | | 2 | 3.1605 | 2.9203 | 3.3395 | 2.2492 |
| | | | 3 | 3.1603 | 2.9227 | 3.3558 | 2.2503 |
| 4 | 3.1593 | 2.9236 | 3.3572 | 2.2492 | | | |

En. Type =Encoder Type, Pre. =Prediction, S.A.=Sample Adaptive, B.A.=Block Adaptive, N.O.=Neighborhood Oriented, C.O. =Column Oriented

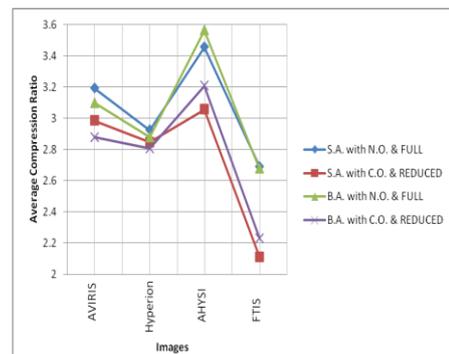


Fig. 9. Average Compression Ratio for different images

Above four figures (10,11,12,13) shows the Input & Output images for three bands only. As it is lossless

algorithm, all input and output images are same for three bands. Small window in the figure show the x, y coordinate, index & RGB value of image. It is also shown in figure (10, 11) that for same coordinates(x&y) of different bands we have different index values. It is due to hyperspectral images. It is shown that both the values are same for input and output images. Mean Square Error (MSE) is Zero in all the cases.

From figures (5,6,7,8), it can be seen that Previous Prediction band P=0 gives least compression ratio result than other values of P because P=0 does not remove any spectral redundancy. Abbreviations used in graphs are shown in the footnotes of Table I. Compression ratio is good for P>=2 than for P<2. Sample values of nearby bands with same spatial coordinates are having nearer value to the current sample value. So, if we take value of P>=2 than it gives good compression performance. For P=2 onwards compression ratio increases slowly. Figure 5(A) is actually zoom version of figure 5. We can see the minute difference in compression ratio between P=1 & p>=2 from figure 5(A) which is for dataset-1. It is similar

for other datasets also. As the number of previous prediction bands P increases, we require more complex hardware. So, P=2 is better for onboard hardware implementation point of view.

Compression Ratio of four images with Neighbour Oriented and Column Oriented prediction mode and with Full and Reduced Prediction mode for both encoding method is shown in Table I for four datasets. From fig. 9, it can be seen that Compression ratio for Full prediction mode in combination with Neighbourhood Oriented is better than for reduced prediction mode in combination with Column Oriented for all images. Average compression ratio is average of compression ratios for p=0 to 4. From figure 9, it can be seen that AHYSI image has more compression ratio. Spectral resolution is difference between two consecutive wavelengths of sensors. The finer the spectral resolution, more the spectral redundancy. AHYSI image has more spectral redundancy because of fine spectral resolution of only 8.6 nm which is less than other three.

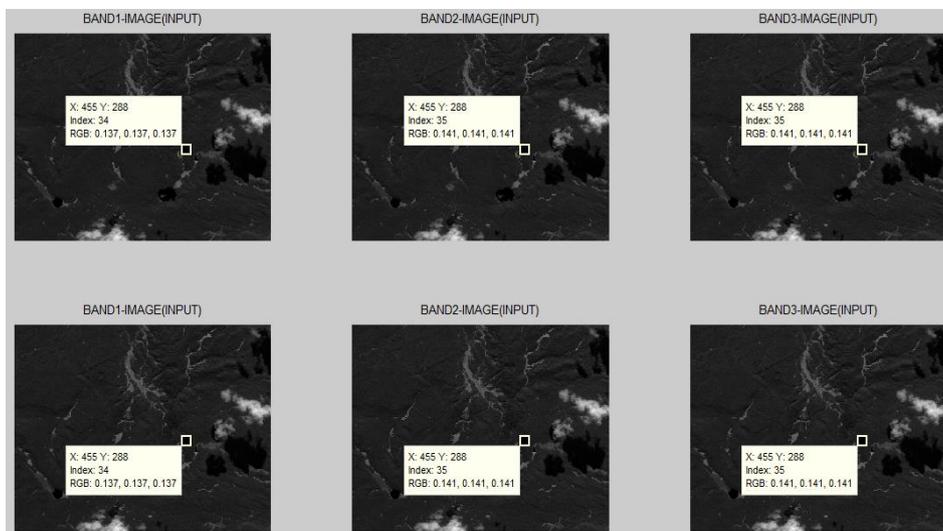


Fig. 10. Inputs and Output of dataset1

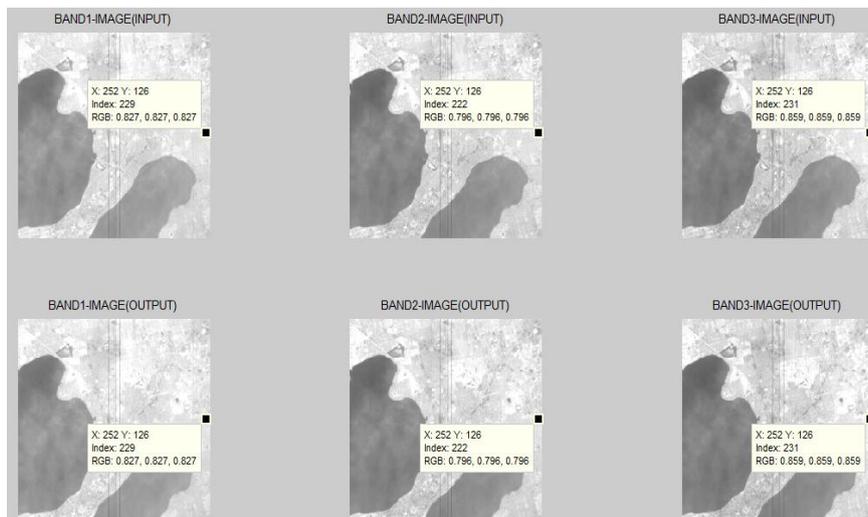


Fig. 11. Inputs and Output of dataset2

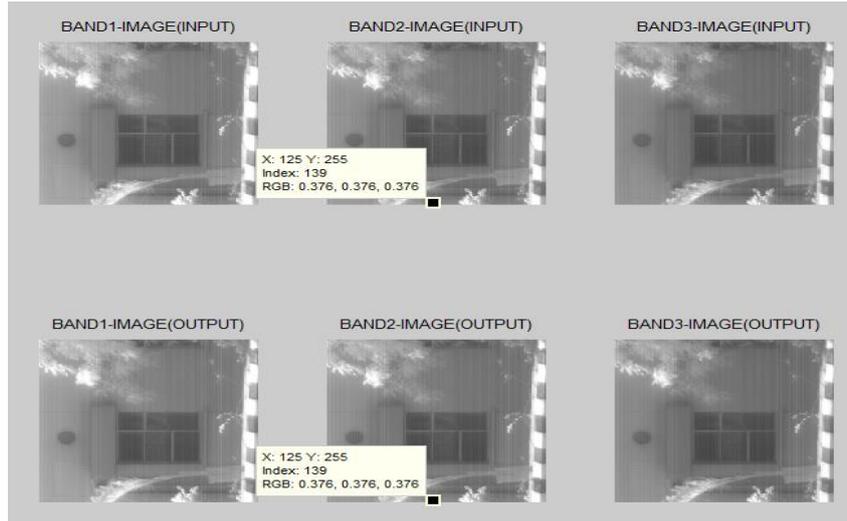


Fig. 12. Inputs and Output of dataset3

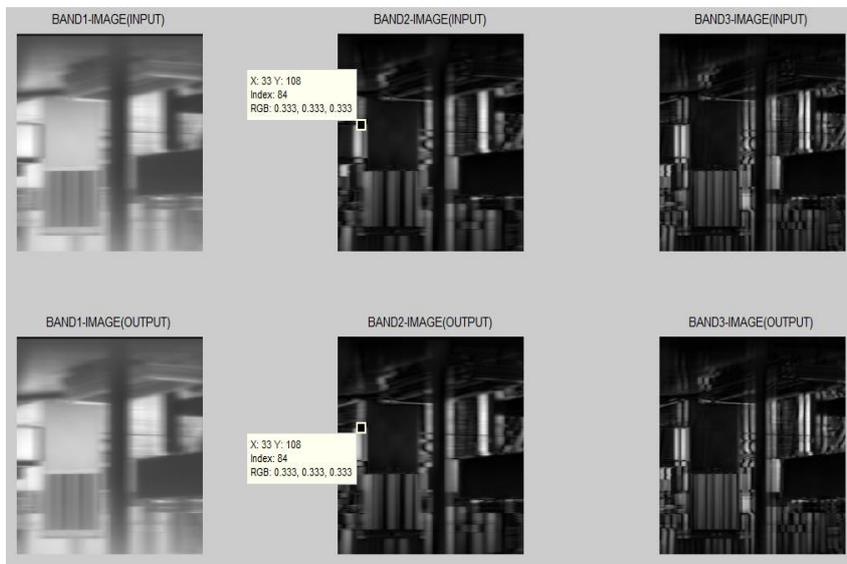


Fig. 13. Inputs and Output of dataset4

IV. CONCLUSION

We have successfully implemented CCSDS recommended algorithm for hyperspectral and multispectral compression namely CCSDS 123.0-B-1 in MATLAB. The developed algorithm is verified by applying 4 sets of real HySI data from different payloads namely AVIRIS, Hyperion, AHySI and FTIS as input to the system and reconstructing the images pixel by pixel. Results show that Compression ratio is about 2 to 3 for previous prediction bands of $P \geq 2$ and MSE is Zero. Compression system can be useful to reduce higher memory requirement constraint or higher bandwidth requirement constraint or both constraints of Hyperspectral imager payloads of space satellites. MSE is equal to zero shows that algorithm is fully lossless. Hyperspectral image compression system hardware can be implemented in FPGA with low complexity.

REFERENCES

- [1] Consultative Committee for Space Data Systems (CCSDS) [Online]. Available: <http://www.ccsds.org>.
- [2] Lossless Multispectral & Hyperspectral Image Compression. Recommendation for Space Data System Standards, CCSDS 123.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, May 2012.
- [3] Lossless Data Compression. Recommendation for Space Data System Standards, CCSDS 121.0-B-2. Blue Book. Issue 2. Washington, D.C.: CCSDS, May 2012.
- [4] Lossless Data Compression. Report Concerning Space Data System Standards, CCSDS 120.0-G-3. Informational Report, Green Book. Washington, D.C.: CCSDS, April 2013.
- [5] AVIRIS & Hyperion Hyperspectral Images [Online]. Available: <http://compression.jpl.nasa.gov/hyperspectral>.
- [6] Khalid Sayood, Introduction to Data Compression, MK Publisher, San Francisco, USA, ISBN 13: 978-0-12-620862-7.

- [7] Multispectral Hyperspectral Data Compression Working Group.[Online].Available:<http://cwe.ccsds.org/sls/default.aspx>.
- [8] G. S. M. Weinberger and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," IEEE Trans. Image Process, vol. 9, no. 8, pp.1309–1324, 2000.
- [9] N. M. X. Wu, "Context-based adaptive, lossless image coding," IEEE Trans. Commun., vol. 45, no. 4, pp. 437–444,1997.
- [10] X. Wu and N. Memon, "Context-based lossless interband compression-extending calic," IEEE Trans. Image Process., vol. 9, no. 6, pp. 994–1001, 2000.
- [11] "Context-based adaptive, lossless image coding," IEEE Trans. Commun., vol. 45, no. 4, pp. 437–444, 1997.
- [12] J. Mielikainen, "Lossless compression of hyperspectral images using lookup tables," IEEE Signal Process. Lett., vol. 13, no. 3, pp. 157–160, Mar. 2006.
- [13] Y. S. B. Huang, "Lossless compression of hyperspectral imagery via lookup tables with predictor selection," Proc. SPIE, vol. 6365, no. 63650L, 2006.
- [14] J. Mielikainen and P. Toivanen, "Lossless compression of hyperspectral images using a quantized index to lookup tables," IEEE Geosci. Remote Sens. Lett., vol. 5, no. 3, pp. 474–478, July 2008.
- [15] Jose Enrique Sánchez, Estanislau Auge, Josep Santaló, Ian Blanes, Joan Serra-Sagristà, Aaron Kiely, "Review and implementation of the emerging CCSDS Recommended Standard for multispectral and hyperspectral lossless image coding", First International Conference on Data Compression, Communications and Processing, IEEE Computer Society, 2011.
- [16] D. Keymeulen, N. Aranki, B. Hopson, A. Kiely, M. Klimesh, and K.Benkrid, "GPU Lossless Hyperspectral Data Compression System for Space Applications," 2012 IEEE Aerospace Conference, March 2012.
- [17] Bormin Huang, Satellite Data Compression, Springer New York Dordrecht Heidelberg London, ISBN 978-1-4614-1182-6.
- [18] Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing, Pearson Education, ISBN 978-81-317-2695-2.

Author's Profile



Dharam J Shah (Surat, 7th July, 1990), Male, is a Assistant Professor at Chhotubhai Gopalbhai Patel Institute of Technology (CGPIT), Uka Tarsadia University (UTU), Maliba campus, Bardoli, Surat, Gujarat, India. He has bachelor of engineering degree in Electronics & Communication from Sarvajani college of engineering & Technology, Surat. He has Master of Engineering in Signal Processing & VLSI Technology from Vishwakarma Government Engineering college. He has 10 months industrial experience at Space Application Center (SAC), Indian Space Research Organization (ISRO). His area of interest is VLSI & Image-processing.



Kuhelika Bera obtained her B.Tech in Radiophysics and Electronics from University of Calcutta. She joined Space Applications Center (ISRO) in 1997 and now working in Sensor Control and Digital Processing Electronics Division. She is involved in the design and development of camera electronics of remote sensing electro optical payloads. Her field of interest is in digital design and signal processing.



Sanjay D Joshi (Ahmedabad, 19th December, 1974), Male, is a Associate Professor at Vishwakarma Government Engineering College Chandkheda, Ahmedabad, Gujarat, India. He has bachelor of engineering degree in Electronics from Birla Vishwakarma Mahavidyalaya, Vallabh Vidhyanagar, Gujarat. He completed Master of Engineering in Electronics and Communication Engg. from Dharmasinh Desai University, Nadiad, Gujarat. He has 17 years of teaching experience.

How to cite this paper: Dharam Shah, Kuhelika Bera, Sanjay Joshi, "Software Implementation of CCSDS Recommended Hyperspectral Lossless Image Compression", IJIGSP, vol.7, no.4, pp.35-41, 2015. DOI: 10.5815/ijigsp.2015.04.04