

New Results in Perceptually Lossless Compression of Hyperspectral Images

Chiman Kwan*, Jude Larkin

Applied Research LLC, Rockville, Maryland, USA Email: *chiman.kwan@signalpro.net

How to cite this paper: Kwan, C. and Larkin, J. (2019) New Results in Perceptually Lossless Compression of Hyperspectral Images. *Journal of Signal and Information Processing*, **10**, 96-124. https://doi.org/10.4236/jsip.2019.103007

Received: July 31, 2019 **Accepted:** August 23, 2019 **Published:** August 26, 2019

Copyright © 2019 by author(s) and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0). http://creativecommons.org/licenses/by/4.0/

Abstract

Hyperspectral images (HSI) have hundreds of bands, which impose heavy burden on data storage and transmission bandwidth. Quite a few compression techniques have been explored for HSI in the past decades. One high performing technique is the combination of principal component analysis (PCA) and JPEG-2000 (J2K). However, since there are several new compression codecs developed after J2K in the past 15 years, it is worthwhile to revisit this research area and investigate if there are better techniques for HSI compression. In this paper, we present some new results in HSI compression. We aim at perceptually lossless compression of HSI. Perceptually lossless means that the decompressed HSI data cube has a performance metric near 40 dBs in terms of peak-signal-to-noise ratio (PSNR) or human visual system (HVS) based metrics. The key idea is to compare several combinations of PCA and video/image codecs. Three representative HSI data cubes were used in our studies. Four video/image codecs, including J2K, X264, X265, and Daala, have been investigated and four performance metrics were used in our comparative studies. Moreover, some alternative techniques such as video, split band, and PCA only approaches were also compared. It was observed that the combination of PCA and X264 yielded the best performance in terms of compression performance and computational complexity. In some cases, the PCA + X264 combination achieved more than 3 dBs than the PCA + I2K combination.

Keywords

Hyperspectral Images (HSI), Compression, Perceptually Lossless, Principal Component Analysis (PCA), Human Visual System (HVS), PSNR, SSIM, JPEG-2000, X264, X265, Daala

1. Introduction

Hyperspectral images (HSI) have found a wide range of applications, including

remote chemical monitoring [1], target detection [2], anomaly and change detection [3] [4] [5], etc. Due to the presence of hundreds of bands in HSI, however, heavy burden in data storage and transmission bandwidth has been introduced.

For many practical applications, it is unnecessary to compress data losslessly because lossless compression can achieve only two to three times of compression. Instead, it will be more practical to apply perceptually lossless compression [6] [7] [8] [9]. A simple rule of thumb is that if the peak-signal-to-noise ratio (PSNR) or human visual system (HVS) inspired metric is above 40 dBs, then the decompressed image is considered as "near perceptually lossless" [10]. In several recent papers, we have applied perceptually lossless compression to maritime images [10], sonar images [10], and Mastcam images [11] [12] [13].

In the past few decades, there are some alternative techniques for compressing HSI. In [14], a tensor approach was proposed to compress the HSI. In [15], a missing data approach was presented to compress HSI. Another simple and straightforward approach is to apply PCA directly to HSI. For instance, in [3], the authors have used 10 PCA compressed bands for anomaly detection. There are also some conventional, simple, and somewhat naïve approaches, to compressing HSI. One idea known as split band (SB) is to split the hundreds of HSI bands into groups of 3-band images and then compress each 3-band image separately. Another idea known as the video approach (Video) is to treat the 3-band images as video frames and compress the frames as a video. The SB and Video approaches have been used for multispectral images [13] and were observed to achieve reasonable performance.

One powerful approach to HSI compression is the combination of PCA and J2K [16]. The idea was to first apply PCA to decorrelate the hundreds of bands and then a J2K codec is then applied to compress the few PCA bands.

In the compression literature, there are a lot of new developments after J2K [17] in the past 15 years. X264 [18], a fast implementation of H264 standard, has been widely used in Youtube and many other social media platforms. X265 [19], a fast implementation of H265, is a new codec that will succeed X264. Moreover, a free video codec known as Daala, emerged recently [20]. In light of these new codecs, it is about time and worthwhile to revisit the HSI compression problem.

In this paper, we summarize our study in this area. Our aim is to achieve perceptually lossless compression of HSI at 100 to 1 compression. The key idea is to compare several combinations of PCA and video/image codecs. Three representative HSI data cubes such as the Pavia and AVIRIS datasets were used in our studies. Four video/image codecs, including J2K, X264, X265, and Daala, have been investigated and four performance metrics were used in our comparative studies. Moreover, some alternative techniques such as video, split band, and PCA only approaches were also compared. It was observed that the combination of PCA and X264 yielded the best performance in terms of compression performance (rate-distortion curves) and computational complexity. In the Pavia data case, the PCA + X264 combination achieved more than 3 dBs than the PCA + J2K combination. Most importantly, our investigations showed that the PCA + X264 combination can achieve more than 40 dBs of PSNR at 100 to 1 compression. This means that perceptually lossless compression of HSI is achievable even at 100 to 1 compression.

The key contributions are as follows. First, we revisited the hyperspectral image compression problem and extensively compared several approaches: PCA only, Video approach, Split Band approach, and a two-step approach. Second, for the two-step approach, we compared four variants: PCA + J2K, PCA + X264, PCA + X265, and PCA + Daala. We observed that the two-step approach is better than PCA only, Video, and Split Band approaches, as perceptually lossless compression can be achieved at 100 to 1 ratio. Third, within the two-step approach, our experiments showed that the PCA + X264 combination is better than other variants in terms of performance and computational complexity. To the best of our knowledge, we have not seen such a study in the literature.

Our paper is organized as follows. Section 2 summarizes the HSI data, the technical approach, the various algorithms, and performance metrics. In Section 3, we focus on the experimental results, including the PCA only results, video approach, split band approach, and two-step approach (PCA + video codecs). Four performance metrics were used to compare different algorithms. Finally, some concluding remarks are included in Section 4.

2. Data and Approach

2.1. Data

We have used several representative HSI data in this paper. The Pavia and AVIRIS image cubes were collected using airborne sensors and the Air Force image was collected on the ground. The numbers of bands in the three data sets vary from one hundred to more than two hundred.

Image 1: Pavia [21]

The first image we had tested was the Pavia data with a $610 \times 340 \times 103$ image cube. The image was taken with a Reflective Optics System Imaging Spectrometer (ROSIS) sensor during a flight over northern Italy. Figure 1 shows the RGB bands of the Pavia image cube.

Image 2: AF image

The second image was the image cube used in [3] and it consists of 124 bands and has a height of 267 pixels and a width of 342 pixels. The RGB image of this data set is shown in **Figure 2**.

Image 3: A VIRIS

The third image was taken from NASA's Airborne Visible Infrared Imaging Spectrometer (AVIRIS). There are 213 bands with wavelengths from 380 nm to 2500 nm. The image size is $300 \times 300 \times 213$. Figure 3 shows the RGB image of the data cube.



Figure 1. RGB image of the Pavia image cube.



Figure 2. RGB image of the AF image cube.



Figure 3. RGB image of the AVIRIS image cube.

2.2. Compression Approaches

Here, we first present the various work flows of several representative compression approaches for HSI. We then include some background materials for several video/image codecs in the literature. We will also mention two conventional performance metrics and two other metrics motivated by human visual systems (HVS).

PCA only

PCA is also known as Karhunen-Loève transform (KLT). Comparing with discrete cosine transform (DCT) and wavelet transform, PCA is optimal because it is data-dependent whereas the DCT and WT are independent of input data. The work flow is shown in **Figure 4**. After some preprocessing steps, PCA compresses the raw HSI data cube (N bands) into a pre-defined number of bands (r bands) and those r bands will be saved or transmitted. At the receiving end, an inverse PCA will be performed to reconstruct the HSI image cube.

Split Band (SB) Approach

This idea is very simple. The HSI bands are divided into groups of 3-band images. Each 3-band image is then compressed as a still image with an image codec. This approach has been observed to work well for multispectral (MS) image cubes [13] where there are only nine bands. The work flow is shown in **Figure 5**.

Video Approach

This approach is similar to the SB approach. Here, the 3-band images are treated as video frames and then a video codec is then applied. Details can be found in **Figure 5**.

We include some details for some of the blocks.

• Pre-processing

The preprocessing has a few components. First, it is important to ensure the input image dimensions to have even numbers because some codecs may crash if the image size has odd dimensions. Second, the input image is normalized to double precision with values between 0 and 1. Third, the different bands are saved into tiff format. Fourth, all the bands are written into YUV444 and Y4M formats.

• Codecs

Different codecs have different requirements. For J2K, we used Matlab's video writer to create a J2K format with certain quality parameters. We then used Matlab's data reader to decode the compressed data and the individual frames will be retrieved. For X264 and X265, the videos are encoded using the respective encoders with certain quality parameters. The video decoding was done within FFMPEG. For Daala, we directly used the Daala's functions for encoding and decoding.

• Performance evaluation

In the evaluation part, each frame is reconstructed and compared to the original input band. Four performance metrics have been used.



Figure 4. PCA only compression work flow.



Figure 5. Work flow for SB and Video approaches.

Two-step Approach: PCA + Video

The two-step approach has been used in [13] [16] before. In [13], X265 was observed to perform better in the second step. In [16], the second step was a J2K codec. However, the study in [13] was for MS images rather than an HSI.

The work flow for the two-step approach is summarized in **Figure 6**. In the second step, we propose to treat the PCA bands as a video.

Brief Review of Relevant Compression Algorithms

Instead of reinventing the wheels, we will use image codecs in the market and objectively evaluate different codecs and eventually recommend the best codec to our customer.

With the above in mind, we include a brief overview of some representative codecs.

DCT based algorithms

• JPEG [22]:

JPEG is the very first image compression standard. The video counterparts are the MPEG-1 and MPEG-2 standards.

• JPEG-XR [23]:

It was developed by Microsoft. The performance is comparable to JPEG-2000. It is mainly used for still image compression.

• VP8 and VP9 [24] [25]:

These video compression algorithms are owned by Google. The performance is somewhat close to X-264. We did include VP8 and VP9 in our study because they are not as popular as X264 and X265.

• X-264 [18]:

X264 is the current state-of-the-art in video compression. Youtube uses X264. It has good still image compression.

• X-265 [19]:

This is the next-generation video codec and has excellent still image compression and video compression. However, the computational complexity is much more than that of X264. In general, X265 has the same basic structure as previous standards.



Figure 6. Two-step approach to HSI compression.

Several studies concluded that X265 yields the same quality as X264, but with only half of the bitrate. It should be noted that X264 and X265 are optimized versions of H264 and H265, respectively.

• Daala [20]

Recently, there is a parallel activity at xiph.org foundation, which implements a compression codec called Daala [20]. It is based on DCT. There are pre- and post-filters to increase energy compaction and remove block artifacts. Daalaborrows ideas from [26].

The block-coding framework in Daala can be illustrated in Figure 7.

In this study, we compared Daala with X264, X265, and J2K in our experiments.

Wavelet-based algorithms

J2K is a wavelet [17] [27] [28] [29] based compression standard. It has better performance than JPEG. However, J2K requires the use of the whole image for coding and hence is not suitable for real-time applications. In addition, motion-J2K for video compression is not popular in the market.

2.3. Performance Metrics

In almost all compression systems, researchers used peak signal-to-noise ratio (PSNR) or structural similarity (SSIM) to evaluate the compression algorithms. Given a fixed compression ratio, algorithms that yield higher PSNR or SSIM will be regarded as better algorithms. However, PSNR or SSIM do not correlate well with human perception. Recently, a group of researchers investigated a number of different performance metrics [30]. Extensive experiments were performed to investigate the correlation between human perceptions with various performance metrics. According to the results found in [30], it was determined that two performance metrics correlate well with human perception. One image example shown in **Figure 8** demonstrates that HVS and HVSm have high correlation with human subjective evaluation results. In the past, we have used HVS and HVSm in several applications [11] [12] [13].

3. Experimental Results

Here, we briefly describe the experimental settings. In PCA only approach, a program was written for PCA. The input is one hyperspectral image and the number of principal components to be used in the compression. The outputs are the PCA bands. The performance metrics are generated by comparing the original hyperspectral image with the inverse-PCA outputs.

In the Video only approach, we used ffmpeg to call X264 and X265. For Daala, we used the latest open-source code in Daala's website. For J2K, we used the



Figure 7. Daala codec for block-based image coding systems.



SSIM = 0.80, 1 SIM II v S-M = 25.50 dB SSIM says: "This is better!"

PSNR-HVS-M says: "This is better!"



built-in J2K function in Matlab. In each codec, there is a quantization or quality parameter (qp) that controls the compression ratio. We chose around 50 qp parameters in our experiments in order to generate smooth performance curves.

In the two-step approach, the PCA is applied first, followed by the video codecs.

3.1. Experiment 1: Pavia Data

PCA Only

Here, we applied PCA directly to compress the 103 bands to 3, 6, and 9 bands, which we denote as PCA3, PCA6, and PCA9, respectively. From **Figure 9**, one can see that PCA3 achieved 33 times of compression with 44.75 dB of PSNR. The other metrics are also high. Similarly, PCA6 and PCA9 also attained high values in performance metrics. This means that PCA alone can achieve reasonable compression performance. However, if our goal is to achieve 100 to 1 compression with higher than 40 dBs of PSNR, then the PCA only approach may be insufficient.

Video Approach

As mentioned earlier, the video approach treats the HSI data cube as a video

where each frame takes 3 bands out of the data cube. There are 35 frames in total in the video for the Pavia data. We then applied four video codecs (J2K, X264, X265, and Daala) to the video. Four performance metrics were generated as shown in **Figure 10**. If one compares the metrics in **Figure 9** and **Figure 10**, one can see that Video approach is slightly better than the PCA only approach. For instance, at 0.03 compression ratio, PCA3 yielded 38.2 dBs and the Video approach yielded more than 40 dBs in terms of PSNR. X265 performed better than others at compression ratios less than 0.1.

Split Band (SB) Approach

Here, SB approach means that every 3 bands in the hyperspectral image cube are treated as a separate image. We then applied four image codecs to each 3-band image. The averaged metrics from all 3-band images were computed. Figure 11 summarizes the performance metrics. J2K has better scores in three out of four metrics. Comparing the Video and SB approaches as shown in Figure 10 and Figure 11, one can see that the Video approach is slightly better. For example, at a compression ratio of 0.05, J2K has 43 dBs (HVSm) using the SB approach and X265 has 52 dBs (HVSm) using the Video approach.

Two-step approach

Two-step approach first compresses the HSI cube by using PCA to a number of bands (3, 6, 9, etc.) The second step applies a video codec to compress the PCA bands. We have five case studies below.



Figure 9. Performance of PCA only: (a) PSNR in dB for Pavia; (b) SSIM for Pavia; (c) HVS in dB for Pavia; (d) HVSm in dB for Pavia.



Figure 10. Performance of video approach: (a) PSNR in dB for Pavia; (b) SSIM for Pavia; (c) HVS in dB for Pavia; (d) HVSm in dB for Pavia.



Figure 11. Performance of SB approach: (a) PSNR in dB for Pavia; (b) SSIM for Pavia; (c) HVS in dB for Pavia; (d) HVSm in dB for Pavia.

PCA3 + Video Codec

Figure 12 summarizes the two-step approach (PCA3 + Video). It can be seen that, at 0.01 compression ratio, the two-step approach can get above 40 dBs of PSNR. The other metrics are also high. Daala has better visual performance (HVS and HVsm) than others. We can also notice that the PCA3 + Video approach can attain much higher compression than PCA only, SB, and Video approaches. That is, the compression ratio can be more than 100 times compression with close to 40 dBs of HVSm in the two-step approach whereas the SB and Video approach cannot achieve 100 to 1 compression with the same performance metrics (40 dBs).

PCA6 + Video

Figure 13 summarizes the PCA6 + Video results. At 0.01 compression ratio, the PCA6 + Video approach appears to be slightly better than that of PCA3 + Video. X264 is better than others in three out of four metrics. In particular, at 0.01 compression, X264 has 45 dBs in terms of HVSm. This value is very high and can be considered as perceptually lossless.

PCA9 + Video Approach

From Figure 14, it is clear that PCA9 + Video is slightly worse as compared to the PCA6 + Video case. For example, at 0.01 compression ratio, PCA9 + Video has 44 dBs in terms of PSNR whereas PCA6 + Video has 45 dBs of PSNR. X264 is better than other codecs.



Figure 12. Performance of two-step (PCA3 + Video) approach: (a) PSNR in dB for Pavia; (b) SSIM for Pavia; (c) HVS in dB for Pavia; (d) HVSm in dB for Pavia.



Figure 13. Performance of two-step (PCA6 + Video) approach: (a) PSNR in dB for Pavia; (b) SSIM for Pavia; (c) HVS in dB for Pavia; (d) HVSm in dB for Pavia.



Figure 14. Performance of two-step (PCA9 + Video) approach: (a) PSNR in dB for Pavia; (b) SSIM for Pavia; (c) HVS in dB for Pavia; (d) HVSm in dB for Pavia.

PCA12 + Video

As shown in **Figure 15**, the performance of PCA12 + Video is somewhat similar to PCA9 + Video.

PCA15 + Video

As shown in **Figure 16**, the performance of PCA15 + Video is somewhat similar to PCA12 + Video.

Comparison of different combination of the two-step approaches

The performance comparison of different combinations of the two-step approaches is summarized in **Table 1**. First, we observe that PCA3 and PCA6 have better performance than PCA9 to PCA15. Second, PCA6 has better performance in X264 and Daala. Third, for PCA6, we observed that X264 is 3.16 dBs in terms of PSNR better than J2K. In terms of HVSm, X264 is 4.2 dBs better than J2K. This is quite significant. For PCA6, Daala has slightly better performance than X264 and X265. However, we noticed that Daala took more computational times than X264. Hence, for practical applications, X264 may be a better choice for HSI compression.

3.2. Experiment 2: AF Image Cube

PCA only approach

Here, we applied PCA directly to compress the 124 bands to 3, 6, and 9 bands, which we denote as PCA3, PCA6, and PCA9, respectively. From Figure 17, one



Figure 15. Performance of two-step (PCA12 + Video) approach: (a) PSNR in dB for Pavia; (b) SSIM for Pavia; (c) HVS in dB for Pavia; (d) HVSm in dB for Pavia.



Figure 16. Performance of two-step (PCA15 + video) approach: (a) PSNR in dB for Pavia; (b) SSIM for Pavia; (c) HVS in dB for Pavia; (d) HVSm in dB for Pavia.



Figure 17. Performance of PCA only: (a) PSNR in dB for AF image cube; (b) SSIM for AF image cube; (c) HVS in dB for AF image cube; (d) HVSm in dB for AF image cube.

		(a) J2K					
Pavia	PSNR	SSIM	HVS	HVSm			
PCA3	43.30	0.89	37.20	40.60			
PCA6	41.40	0.82	36.63	41.00			
PCA9	39.20	0.78	34.50	38.50			
PCA12	37.75	0.73	32.25	36.00			
PCA15	37.00	0.70	31.00	34.00			
(b) X264							
Pavia	PSNR	SSIM	HVS	HVSm			
PCA3	43.79	0.92	37.51	41.10			
PCA6	44.56	0.89	39.99	45.20			
PCA9	43.75	0.88	39.40	45.00			
PCA12	43.50	0.88	39.25	45.10			
PCA15	43.50	0.87	39.00	45.00			
(c) X265							
Pavia	PSNR	SSIM	HVS	HVSm			
PCA3	43.67	0.95	37.5	41			
PCA6	43.4	0.88	38.73	44			
PCA9	41.65	0.86	37	42			
PCA12	40.5	0.83	35.5	40.5			
PCA15	40.15	0.82	35	39.8			
(d) Daala							
Pavia	PSNR	SSIM	HVS	HVSm			
PCA3	43.54	0.92	37.70	41.20			
PCA6	43.40	0.88	39.88	45.50			
PCA9	42.05	0.87	39.00	45.25			
PCA12	41.30	0.86	38.25	44.80			
PCA15	41.00	0.85	38.00	44.30			

Table 1. Performance comparison of different combinations of two-step approach. Bold numbers indicate the best performing method for each column.

can see that PCA3 achieved 42 times of compression with 40.7 dBs of PSNR. The other metrics are also high. Similarly, PCA6 and PCA9 also attained high performance, but lower compression ratios. This means PCA alone can achieve reasonable compression. However, PCA only is not enough to achieve 100 to 1 compression.

Video approach

Comparing the performance of video approach (**Figure 18**) with the PCA only approach (**Figure 17**), one can immediately notice that the Video approach allows higher compression ratios to be achieved. For instance, at 0.01 compression ratio, X265 achieved about 38 dBs in PSNR. X265 performs well for small ratios (high compression).

SB approach

Comparing the SB approach in **Figure 19** with the Video approach in **Figure 18**, the Video approach is better. For instance, if one looks at the PSNR values at 0.05 compression ratio, one can see that the X265 codec in the Video approach has a value of 44 dBs whereas the best codec (J2K) has a value of 41.5 dBs.

Two-step approach

Here, the PCA is combined with the Video approach. That is, the PCA is first applied to the 124 bands to obtain 3, 6, 9, 12, and 15 bands. After that, a video codec is applied to further compress the PCA bands.

PCA3 + Video

From Figure 20, we can see that the PCA3 + Video can achieve 0.01 compression ratio with more than 40 dBs of PSNR. Hence, the performance is better than the earlier approaches (PCA only, Video, and SB). Daala has better performance in terms of HVS and HVsm.

PCA6 + Video

From Figure 21 and Figure 20, we can see that PCA6 + Video is better than PCA3 + Video. For example, at 0.01 compression ratio, Daala has 44 dBs (HVSm) for PCA6 + Video whereas Daala only has 34.75 dB for PCA3 + Video. X264 has better metrics in PSNR and SSIM, but Daala has better performance in terms of HVS and HVSm.

PCA9 + Video

Comparing Figure 21 and Figure 22, PCA9 + Video is worse than that of PCA6 + Video. For instance, at 0.01 compression ratio, PCA9 + Video has 42 dBs (PSNR) and PCA6 + Video has slightly over 44 dBs of PSNR. Daala has better scores in HVS and HVSm, but X264 has higher values in PSNR and SSIM.

PCA12 + Video

As shown in **Figure 23**, the performance of PCA12 + Video is worse than some of the earlier combinations. For example, Daala's HVSm value is 40 dBs at 0.01 compression ratio and this is lower than PCA6 + Video (**Figure 21**) and PCA9 + Video (**Figure 22**). PCA12 + Video is better than PCA3 + Video (**Figure 20**).

PCA15 + Video

From Figure 24, we can see that PCA15 + Video is similar to PCA3 + Video (Figure 20), but worse than the other PCA + Video combinations (Figure 21, Figure 22, Figure 23).

Comparison of different combinations of the two-step approach

From **Table 2**, we have the following observations. First, PCA6 + Video combination has the best performance for each codec. Second, X264 has the best performance in PSNR whereas Daala has the best performance in HVS and HVSm. Third, X264 is 1.45 dBs better than J2K for PCA6 case. As mentioned earlier, X264 is faster to run than Daala. Hence, X264 may be more suitable in practical applications.

3.3. Experiment 3: AVIRIS Image Cube

PCA only approach

Here, we applied PCA directly to compress the 213 bands to 3, 6, and 9 bands, which we denote as PCA3, PCA6, and PCA9, respectively. From **Figure 25**, one can see that PCA3 achieved 72 times of compression with 39.7 dBs of PSNR. The other metrics are also high. Similarly, PCA6 and PCA9 also attained high performance, but lower compression ratios. This means PCA alone can achieve reasonable compression.

Table 2. Performance comparison of different combinations of two-step approach. Bold numbers indicate the best performing method for each column.

(a) J2K							
Air Force	PSNR	SSIM	HVS	HVSm			
PCA3	40.31	0.92	33.39	34.76			
PCA6	42.25	0.86	38.00	41.90			
PCA9	40.25	0.76	36.10	40.00			
PCA12	39.00	0.71	34.65	38.25			
PCA15	38.20	0.67	33.75	37.10			
(b) X264							
Air Force	PSNR	SSIM	HVS	HVSm			
PCA3	40.45	0.92	33.5	34.86			
PCA6	43.7	0.90	39.55	43.75			
PCA9	41.6	0.84	37.05	42			
PCA12	38.85	0.71	34.22	37.61			
PCA15	36.7	0.57	31.2	33.7			
(c) X265							
Air Force	PSNR	SSIM	HVS	HVSm			
PCA3	40.41	0.92	33.48	34.85			
PCA6	42.75	0.89	38.5	42.6			
PCA9	40.75	0.82	36.4	40.5			
PCA12	38.98	0.76	34.19	37.7			
PCA15	37.75	0.7	32.5	35.4			
(d) Daala							
Air Force	PSNR	SSIM	HVS	HVSm			
PCA3	40.38	0.92	33.57	34.88			
PCA6	42.80	0.89	39.70	44.25			
PCA9	41.20	0.84	38.60	44.25			
PCA12	38.75	0.75	35.53	40.15			
PCA15	37.53	0.69	33.65	37.50			

DOI: 10.4236/jsip.2019.103007



Figure 18. Performance of Video approach: (a) PSNR in dB for AF image cube; (b) SSIM for AF image cube; (c) HVS in dB for AF image cube; (d) HVSm in dB for AF image cube.



Figure 19. Performance of SB approach: (a) PSNR in dB for AF image cube; (b) SSIM for AF image cube; (c) HVS in dB for AF image cube; (d) HVSm in dB for AF image cube.



Figure 20. Performance of PCA3 + Video approach: (a) PSNR in dB for AF image cube; (b) SSIM for AF image cube; (c) HVS in dB for AF image cube; (d) HVSm in dB for AF image cube.



Figure 21. Performance of PCA6 + Video approach: (a) PSNR in dB for AF image cube; (b) SSIM for AF image cube; (c) HVS in dB for AF image cube; (d) HVSm in dB for AF image cube.



Figure 22. Performance of PCA9 + Video approach: (a) PSNR in dB for AF image cube; (b) SSIM for AF image cube; (c) HVS in dB for AF image cube; (d) HVSm in dB for AF image cube.







Figure 24. Performance of PCA15 + Video approach: (a) PSNR in dB for AF image cube; (b) SSIM for AF image cube; (c) HVS in dB for AF image cube; (d) HVSm in dB for AF image cube.



Figure 25. Performance of PCA only: (a) PSNR in dB for AVIRIS image cube; (b) SSIM for AVIRIS image cube; (c) HVS in dB for AVIRIS image cube; (d) HVSm in dB for AVIRIS image cube.

Video approach

Here, the 213 bands are divided into groups of 3 bands. As a result, there are 71 groups, which are then treated as 73 frames in a video. After that, different video codecs are applied. The performance metrics are shown in **Figure 26**. Comparing with PCA only approach, the video approach is slightly inferior. For instance, PCA6 has PSNR of 44 dBs at a compression ratio of 0.028 whereas the Video only approach has about 42.5 dBs at 0.028 ratio.

SB Approach

Here the 73 groups of 3-band images are compressed separately. The results shown in **Figure 27** are worse than the video approach. This is understandable as the correlations between frames were not taken into account in the SB approach.

Two-step approach

We have the following five case studies based on the number of PCA bands coming out of the first step.

PCA3 + Video

From Figure 28, the performance metrics appear to flatten out after a compression ratio of 0.005. The maximum PSNR value is below 40 dBs. Other metrics are also not very high. Comparing with the PCA only, Video, and SB approaches, PCA3 + Video does not show any advantages.

PCA6 + Video

From Figure 29, we can see that PCA6 + Video has much better performance than PCA3 + Video as well as PCA only, Video, and SB approaches. At 0.01 compression ratio, the PSNR values reached more than 42 dBs. Other metrics also performed well. Daala has higher scores in HVS and HVSm. X265 is slightly better in PSNR and SSIM.

PCA9 + Video

Comparing Figure 29 and Figure 30, we can see that PCA9 + Video has better metrics than that of PCA6 + Video. For instance, at 0.01 compression ratio, PCA9 + Video has achieved 40 dBs of HVSm (Daala), but PCA6 + Video has 38.5 dBs.

PCA12 + Video

Comparing **Figure 30** and **Figure 31**, we can see that PCA12 + Video is slightly worse than that of PCA9 + Video.

PCA15 + Video

Comparing **Figure 31** and **Figure 32**, it can be seen that PCA15 + Video is slightly worse than PCA12 + Video.

Comparison of different combinations in the Two-step approach

Table 3 summarizes the performance metrics of different combinations of the two-step approach. First, PCA6 performed better than other PCA + Video combinations. Second, X265 performed better than other codecs. However, since X265 requires a lot of computational power and X264 is only slightly inferior to X265, it is better to use the PCA6 + X264 combination. Third, the HVSm values in Daala and X265 are somewhat strange, but we could not find an explanation

_

_

_

for that even though we repeated the experiments several times. This behavior only happened in the AVIRIS data. Fourth, we noticed that X264 is 0.25 dBs in PSNR better than that of J2K.

Table 3. Performance comparison of different combinations of two-step approach at 0.01 compression ratio for the AVIRIS data. Bold numbers indicate the best performing method for each column.

(a) J2K							
AVIRIS	PSNR	SSIM	HVS	HVSm			
PCA3	32.35	0.68	29.8	31.6			
PCA6	42.90	0.902	36.2	38.48			
PCA9	42.57	0.88	36.9	39.9			
PCA12	41.9	0.84	37	41.08			
PCA15	41.95	0.8	36.25	40.5			
(b) X264							
AVIRIS	PSNR	SSIM	HVS	HVSm			
PCA3	37.26	0.76	30.2	31.84			
PCA6	43.15	0.91	36.43	38.68			
PCA9	42	0.87	36.01	39.2			
PCA12	40.85	0.82	35.4	39.1			
PCA15	40.4	0.796	35.05	38.9			
(c) X265							
AVIRIS	PSNR	SSIM	HVS	HVSm			
PCA3	37.24	0.79	30.10	31.70			
PCA6	43.30	0.92	36.51	38.73			
PCA9	41.93	0.88	35.90	38.90			
PCA12	41.11	0.86	35.45	39.20			
PCA15	41.00	0.85	35.50	39.50			
(d) Daala							
AVIRIS	PSNR	SSIM	HVS	HVSm			
PCA3	37.27	0.78	30.75	32.49			
PCA6	42.93	0.91	36.50	38.77			
PCA9	42.20	0.89	37.10	40.20			
PCA12	41.02	0.85	36.90	41.55			
PCA15	40.75	0.84	37.05	42.30			



Figure 26. Performance of Video approach: (a) PSNR in dB for AVIRIS image cube; (b) SSIM for AVIRIS image cube; (c) HVS in dB for AVIRIS image cube; (d) HVSm in dB for AVIRIS image cube.



Figure 27. Performance of SB approach: (a) PSNR in dB for AVIRIS image cube; (b) SSIM for AVIRIS image cube; (c) HVS in dB for AVIRIS image cube; (d) HVSm in dB for AVIRIS image cube.



Figure 28. Performance of PCA3 + Video approach: (a) PSNR in dB for AVIRIS image cube; (b) SSIM for AVIRIS image cube; (c) HVS in dB for AVIRIS image cube; (d) HVSm in dB for AVIRIS image cube.



Figure 29. Performance of PCA6 + Video approach: (a) PSNR in dB for AVIRIS image cube; (b) SSIM for AVIRIS image cube; (c) HVS in dB for AVIRIS image cube; (d) HVSm in dB for AVIRIS image cube.



Figure 30. Performance of PCA9 + Video approach: (a) PSNR in dB for AVIRIS image cube; (b) SSIM for AVIRIS image cube; (c) HVS in dB for AVIRIS image cube; (d) HVSm in dB for AVIRIS image cube.







Figure 32. Performance of PCA15 + Video approach: (a) PSNR in dB for AVIRIS image cube; (b) SSIM for AVIRIS image cube; (c) HVS in dB for AVIRIS image cube; (d) HVSm in dB for AVIRIS image cube.

4. Conclusion

In this paper, we summarize some new results for HSI compression. The key idea is to revisit a two-step approach to HSI data compression. The first step adopts PCA to compress the HSI data spectrally. That is, the number of bands is greatly reduced to a few bands via PCA. The second step applies the latest video/image codecs to further compress the few PCA bands. Four well-known codecs (J2K, X264, X265, and Daala) were used in the second step. Three HSI data sets with diversely varying numbers of bands were used in our studies. Four performance metrics were utilized in our experiments. We have several key observations. First, we observed that compressing of the HIS to six bands has the best overall performance in all of the three HSI data sets. This is different from the observation in [16] where more PCA bands were included in the J2K step. Second, the X264 codec gave the best performance in terms of compression performance and computational complexity. Third, the PCA6 + X264 combination can be 3 dBs better than the PCA6 + J2K combination in the Pavia data at 100 to 1 compression and this is quite significant. Fourth, even at 100 to 1 compression, the PCA6 + X264 combination can attain better than 40 dBs in PSNR for all of the three data sets. This means the compression performance is perceptually lossless at 100 to compression.

Acknowledgements

This research was supported by NASA Jet Propulsion Laboratory under contract

80NSSC17C0035. The views, opinions and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of NASA or the U.S. Government.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- Ayhan, B., Kwan, C. and Jensen, J.O. (2019) Remote Vapor Detection and Classification Using Hyperspectral Images. *Proceedings SPIE, Chemical, Biological, Radiological, Nuclear, and Explosives (CBRNE) Sensing XX*, Vol. 11010, 110100U. https://doi.org/10.1117/12.2518500
- [2] Zhou, J., Kwan, C. and Ayhan, B. (2017) Improved Target Detection for Hyperspectral Images Using Hybrid In-Scene Calibration. *Journal of Applied Remote Sensing*, 11, Article ID: 035010. <u>https://doi.org/10.1117/1.JRS.11.035010</u>
- [3] Zhou, J., Kwan, C., Ayhan, B. and Eismann, M. (2016) A Novel Cluster Kernel RX Algorithm for Anomaly and Change Detection Using Hyperspectral Images. *IEEE Transactions on Geoscience and Remote Sensing*, 54, 6497-6504. https://doi.org/10.1109/TGRS.2016.2585495
- [4] Zhou, J., Kwan, C. and Budavari, B. (2016) Hyperspectral Image Super-Resolution: A Hybrid Color Mapping Approach. *Journal of Applied Remote Sensing*, 10, Article ID: 035024. https://doi.org/10.1117/1.JRS.10.035024
- [5] Qu, Y., Wang, W., Guo, R., Ayhan, B., Kwan, C., Vance, S. and Qi, H. (2018) Hyperspectral Anomaly Detection through Spectral Unmixing and Dictionary Based Low Rank Decomposition. *IEEE Transactions on Geoscience and Remote Sensing*, 56, 4391-4405. <u>https://doi.org/10.1109/TGRS.2018.2818159</u>
- [6] Wu, H.R., Reibman, A., Lin, W., Pereira, F. and Hemami, S. (2013) Perceptual Visual Signal Compression and Transmission. *Proceedings of the IEEE*, 101, 2025-2043. <u>https://doi.org/10.1109/JPROC.2013.2262911</u>
- [7] Wu, D., Tan, D.M., Baird, M., DeCampo, J., White, C. and Wu, H.R. (2006) Perceptually Lossless Coding of Medical Images. *IEEE Transactions on Medical Imaging*, 25, 335-344. <u>https://doi.org/10.1109/TMI.2006.870483</u>
- [8] Oh, H., Bilgin, A. and Marcellin, M.W. (2013) Visually Lossless Encoding for JPEG 2000. *IEEE Transactions on Image Processing*, 22, 189-201. https://doi.org/10.1109/TIP.2012.2215616
- [9] Tan, D.M. and Wu, D. (2016) Perceptually Lossless and Perceptually Enhanced Image Compression System & Method. U.S. Patent 9,516,315,6.
- [10] Kwan, C., Larkin, J., Budavari, B., Chou, B., Shang, E. and Tran, T.D. (2019) A Comparison of Compression Codecs for Maritime and Sonar Images in Bandwidth Constrained Applications. *Computers*, 8, 32. https://doi.org/10.3390/computers8020032
- [11] Kwan, C., Larkin, J., Budavari, B. and Chou, B. (2019) Compression Algorithm Selection for Multispectral Mastcam Images. *Signal & Image Processing: An International Journal*, **10**, 1-14. <u>https://doi.org/10.5121/sipij.2019.10101</u>
- [12] Kwan, C. and Larkin, J. (2018) Perceptually Lossless Compression for Mastcam Images. *IEEE Ubiquitous Computing, Electronics & Mobile Communication Con*-

ference, New York, 8-10 November 2018, 559-565. https://doi.org/10.1109/UEMCON.2018.8796824

- [13] Kwan, C., Larkin, J. and Chou, B. (2019) Perceptually Lossless Compression of Mastcam Images with Error Recovery. *Proceedings SPIE, Signal Processing, Sensor/Information Fusion, and Target Recognition* XXVIII, Vol. 11018. https://doi.org/10.1117/12.2518482
- [14] Li, N. and Li, B. (2010) Tensor Completion for On-Board Compression of Hyperspectral Images. *IEEE International Conference on Image Processing*, Hong Kong, 517-520. <u>https://doi.org/10.1109/ICIP.2010.5651225</u>
- [15] Zhou, J., Kwan, C. and Ayhan, B. (2012) A High Performance Missing Pixel Reconstruction Algorithm for Hyperspectral Images. 2nd International Conference on Applied and Theoretical Information Systems Research, Taipei, 27-29 December 2012.
- [16] Du, Q. and Fowler, J.E. (2007) Hyperspectral Image Compression Using JPEG2000 and Principal Component Analysis. *IEEE Geoscience and Remote Sensing Letters*, 4, 201-205. <u>https://doi.org/10.1109/LGRS.2006.888109</u>
- [17] JPEG-2000. https://en.wikipedia.org/wiki/JPEG_2000
- [18] X264. http://www.videolan.org/developers/x264.html
- [19] X265. https://www.videolan.org/developers/x265.html
- [20] Daala. https://xiph.org/daala/
- [21] http://lesun.weebly.com/hyperspectral-data-set.html
- [22] JPEG. https://en.wikipedia.org/wiki/JPEG
- [23] JPEG-XR. https://en.wikipedia.org/wiki/JPEG_XR
- [24] VP8. https://en.wikipedia.org/wiki/VP8
- [25] VP9. https://en.wikipedia.org/wiki/VP9
- [26] Tran, T.D., Liang, J. and Tu, C. (2003) Lapped Transform via Time-Domain Preand Post-Filtering. *IEEE Transactions on Signal Processing*, **51**, 1557-1571. <u>https://doi.org/10.1109/TSP.2003.811222</u>
- [27] Kwan, C., Li, B., Xu, R., Tran, T. and Nguyen, T. (2001) Very Low-Bit-Rate Video Compression Using Wavelets. *Wavelet Applications* VIII, *Proceedings SPIE*, Vol. 4391, 176-180. <u>https://doi.org/10.1117/12.421197</u>
- [28] Kwan, C., Li, B., Xu, R., Tran, T. and Nguyen, T. (2001) SAR Image Compression Using Wavelets. *Wavelet Applications* VIII, *Proceedings SPIE*, Vol. 4391, 349-357. https://doi.org/10.1117/12.421215
- [29] Kwan, C., Li, B., Xu, R., Li, X., Tran, T. and Nguyen, T. (2006) A Complete Image Compression Codec Based on Overlapped Block Transform with Post-Processing. *EUROSIP Journal of Applied Signal Processing*, 2006, Article ID: 010968. https://doi.org/10.1155/ASP/2006/10968
- [30] Ponomarenko, N., Silvestri, F., Egiazarian, K., Carli, M., Astola, J. and Lukin, V. (2007) On Between-Coefficient Contrast Masking of DCT Basis Functions. *Proceedings of the 3rd International Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale*, Arizona.