Title: Noise analysis using Tucker Decomposition and PCA on spectral images

Author: PADILLA, Efraín, TORRES-ROMÁN, Deni Librado y MÉNDEZ-VÁZQUEZ, Andrés
Content

1. Introduction
2. Problems
3. Noise assumptions
4. Compression
5. Phenomenology observed
6. Experiment
7. Conclusions
8. References
Introduction

Spectral Image (SI) definition:

• The greek word “spectral,” which relates to “colors”, combined with image figuratively mean “Image of colors.”

• Is based on taking a portion of the electromagnetic spectrum and breaking it into pieces for the purpose of analytical computations. [1]

• We represent the SI as a tensor.
Tensor

• A tensor is a multidimensional array. The order of a tensor is the number of its dimensions, also called the ways or modes; therefore an $N$th-order is an array with $N$ dimensions. [2]

Notation

$x$ : a scalar.

$x$ : a 1st-order tensor or a vector.

$X$ : a 2nd-order tensor or a matrix.

$\mathcal{X}$ : a 3rd or higher-order tensor.

$x_{i,j,\ldots,n}$ : the element $(i, j, \ldots, n)$ of a tensor $\mathcal{X}$.

$x_i$ : the $i$th column of a matrix $X$. 

Fig. 2 - Example of a 3rd-order tensor $\in \mathbb{R}^{3 \times 3 \times 3}$
Tensor Examples

$x : a$ scalar. \hspace{1cm} \mathbf{x} : a$ 1st-order tensor. \hspace{1cm} \mathbf{X} : a$ 2nd-order tensor. \hspace{1cm} \mathbf{X}' : a$ 3rd-order tensor.

$x \in \mathbb{R}$ \hspace{1cm} \mathbf{x} \in \mathbb{R}^3 \hspace{1cm} \mathbf{X} \in \mathbb{R}^{3 \times 3} \hspace{1cm} \mathbf{X}' \in \mathbb{R}^{3 \times 3 \times 3}$

Fig. 3 – Tensor Examples
SI Representation

- We represent a SI as a 3rd-order tensor

\[ \mathcal{H} : \text{Hyperspectral Image} \]

\[ \mathcal{H} \in \mathbb{R}^{h \times w \times b} \]

- \( h \) : height
- \( w \) : width
- \( b \) : bands

Where \( h, w \) and \( b \) are the number of pixels (elements) in each Mode-1 (column), Mode-2 (row) and Mode-3 (tube) fibers respectively.
Problems

A spectral image contains abundant spatial and spectral information and is always corrupted by various noises, especially Gaussian noise. [3]
Problems

Noise is a problem in spectral imagery applications.
The performance of spectral analysis tasks (i.e. Classification) depends on the SNR of the spectral image. [4]
Before deal with the Noise...
We need to know about him.
Noise Assumptions [4]

• The presence of different noise sources in a SI makes its modeling and the denoising task very challenging
• Therefore, SI denoising approaches often consider either of the following noise types or a mixture of them.
Noise Assumptions [4]

• Signal Independent Noise
  • Thermal noise and quantization noise in HSI are modeled by signal independent Gaussian additive noise. Usually, noise is assumed to be uncorrelated spectrally. The Gaussian assumption has been broadly used in hyperspectral analysis since it considerably simplifies the analysis and the noise variance estimation.

• Sparse Noise
  • Impulse noises such as salt and pepper noise, missing pixels, missing lines and other outliers often exist in the acquired HSI, and are usually due to a malfunctioning of the sensor.

• Pattern Noise
  • Hyperspectral imaging systems may also induce artifacts in hyperspectral images, usually referred to as pattern noise.
Additive Noise

• Generally, in the state of the art can be found many ways to get:

\[ S = X + N \]

• Where:

\[ S, X, N \in \mathbb{R}^{h \times w \times b} \]

\[ S \rightarrow \text{Noisy SI} \]
\[ X \rightarrow \text{Clean SI} \]
\[ N \rightarrow \text{Noise} \]
To deal with the size of the data...
Compression methods!
PCA

- Is a dimensionality reduction method, that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into smaller one that still contains most of the information in the large dataset.

Fig.5 – PCA example
Tucker Tensor Decomposition

- Is a form of Higher Order of PCA. It decomposes a tensor into a core tensor multiplied (or transformed) by a matrix along each mode. Thus, in the three-way case where $\mathcal{X} \in \mathbb{R}^{I \times J \times K}$ we have. [2]

- To do compression only in the spectral domain we can make $A, B = I$ (For semantic segmentation purposes)

$$
\mathcal{X} \approx \mathcal{G} \times_1 A \times_2 B \times_3 C = \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} g_{pqr} \cdot a_p \circ b_q \circ c_r = [\mathcal{G} ; A, B, C].
$$

Fig.6 – Tucker Decomposition of a three way array
Phenomenology observed in [16]

Accuracy improvement after compression

Normally after a dataset compression stage we expect a loss of information and consequently a worse accuracy in classification tasks, but in this work the **accuracy improve** in some cases! **WHY?**

![Fig.7 – Framework proposed in [16]](image)
Experiment

Analyze the images and its classification accuracy through a Neural Network after PCA and Tucker compression
SI Dataset [16]

Real Dataset with simulated noise

• We used a Sentinel-2 image data set of 115 scenes of (128x128x9) from Central Europe (already with “natural” noise) with simulated additive noise as a case study, can be generated by zero-mean Gaussian noise as seen in [4]

\[
N = [n_{ij}] \text{ where } n_{ij} \sim N(0, \sigma_i^2) \text{ is normally distributed}
\]

• The variance of the noise \( \sigma_i^2 \) variates along the spectral axis according to

\[
\sigma_i^2 = \sigma^2 \frac{\sum_{j=1}^{p} e^{-\frac{(j-p/2)^2}{2\eta^2}}}{\sum_{j=1}^{p} e^{-\frac{(i-p/2)^2}{2\eta^2}}}
\]

Where the power of the noise is controlled by \( \sigma \), and \( \eta \) behaves like the standard deviation of a Gaussian bell curve. \( p \) is the number of bands which is 9.
First Look

With $\sigma = 13$ and $\eta = 72$, scene = 50, band = 1

Fig. 7 – Noise adding to scene 50 – Visualization of band 1
First Look

With $\sigma = 13$ and $\eta = 72$, scene = 50, band = 1

Compressed to 3 components (PCA)/tensorial bands (Tucker)

Band 1 – Original

Band 1 – Noise added

After PCA

After Tucker Decomposition

Fig.8 – Visualization of compressed images of a noisy band
Noise parameters [17]

Real Dataset with simulated noise

• We set an average SNR (Signal to Noise Ratio) of 17dB. We get this with $\sigma = 82$, $SNR_{ave} = 17dB$

\[
SNR = 10 \log_{10} \frac{E(X^T X)}{E(N^T N)}
\]

• The experiment were performed with different values of $\eta = 18, 36, 72$
Experiment

$S \rightarrow$ Input noisy SI
$\mathcal{X} \rightarrow$ Clean SI
$Y \rightarrow$ Pixel labels
$\hat{Y} \rightarrow$ Estimated pixel labels

$S, \mathcal{X} \in \mathbb{R}^{w \times h \times b}$
$Y, \hat{Y} \in \mathbb{R}^{w \times h}$
Multi-Layer Perceptron [18]

Artificial Neural Network as Classifier

- **Parameters:**
  - Hidden Layers: 2, Neurons per layer: 100
  - Activation function: ReLu
  - Solver for weight optimization: ADAM
  - Regularization term: L2 penalty
  - Batch size: 200
  - Learning rate: Adaptative
  - Iterations: 10
  - Train 70%/Test 30% picked random

- Data dimensions (vectorized) = (1,884,160 x PrincipalComponents) [PCA]
  = (1,884,160 x TensorialBands) [Tucker]
Accuracy Analysis – No Noise Added

![Accuracy Analysis Graph]

- Accuracy-PCA
- Accuracy-Tucker

To Components/Tensorial Bands compressed
Accuracy Analysis – Noise Added: $\sigma = 82$ and $\eta = 16$

![Graph showing accuracy analysis](image-url)
Accuracy Analysis – Noise Added: $\sigma = 82$ and $\eta = 36$
Accuracy Analysis – Noise Added: $\sigma = 82$ and $\eta = 72$
Conclusions

• An experiment was proposed to observe the behavior of the classification accuracy after compression methods.
• We can observe that the accuracy generally increment compressing to 6 or 7 bands instead of 8 with Tucker Decomposition.
• We can attribute that the information in this less significative bands is in its majority noise.
• Lower SNR minimize this phenomenon.
• Tucker Decomposition generally perform better compression than PCA.

Future Work

• To improve the classifier to get higher accuracies and maximize this phenomenology.
• Compare with a noise-free spectral image.
References


References


Questions?

Efraín Padilla
+52 1 33 1395 4841
eapadilla@gdl.cinvestav.mx
gdl.cinvestav.mx
• Signal dependent Noise [4]

  • Shot (photon) noise in HSI is modeled by the Poisson distribution for which the noise variance is signal dependent. The noise variance estimation under this assumption is more challenging than in the signal independent case.

  • The generic Hyperspectral pixel $\mathbf{X}$ can be viewed as an $N_B \times 1$ ($N_B$ being the number of sensor channels) modeled as

  $$
  \mathbf{X} = \mathbf{s} + \mathbf{N}(\mathbf{s})
  $$

  Where $\mathbf{s} = [s_1, ..., s_{N_B}]^T$ is the vector denoting the useful signal in the $N_B$ sensor channels and $\mathbf{N}(\mathbf{s}) = [N_1(s_1), ..., N_{N_B}(s_{n_B})]^T$ represents the random noise vector. [6]
Backup - SNR

Evaluate Restoration Results

\[ \text{SNR}_{\text{in}} = 10 \log_{10} \left( \frac{\|X\|_F^2}{\|X - H\|_F^2} \right) \]

\[ \text{SNR}_{\text{out}} = 10 \log_{10} \left( \frac{\|X\|_F^2}{\|X - \hat{X}\|_F^2} \right) \]