THE UNIVERSITY OF ARIZONA

BE/BAT 485/585

Remote Sensing Data and Methods

Lab - 7

Instructor: Kamel Didan[1,2]

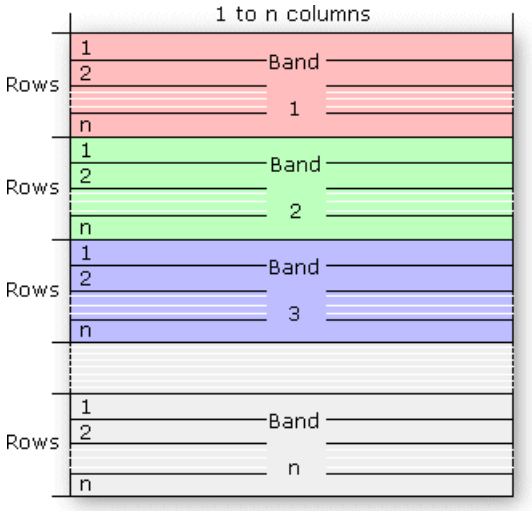
Helpers: Dr. Armando Barreto[1,2]

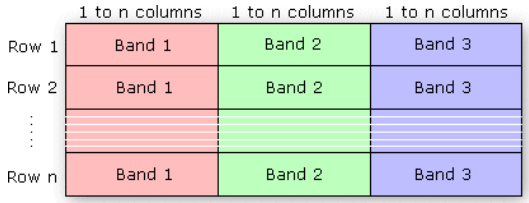
Mr. Truman Combs[1,2]

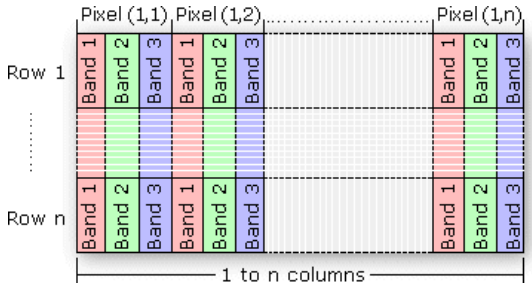
[1]BE Dept., University of Arizona,

[2]VIP Lab.

BSQ, BIL, BIP







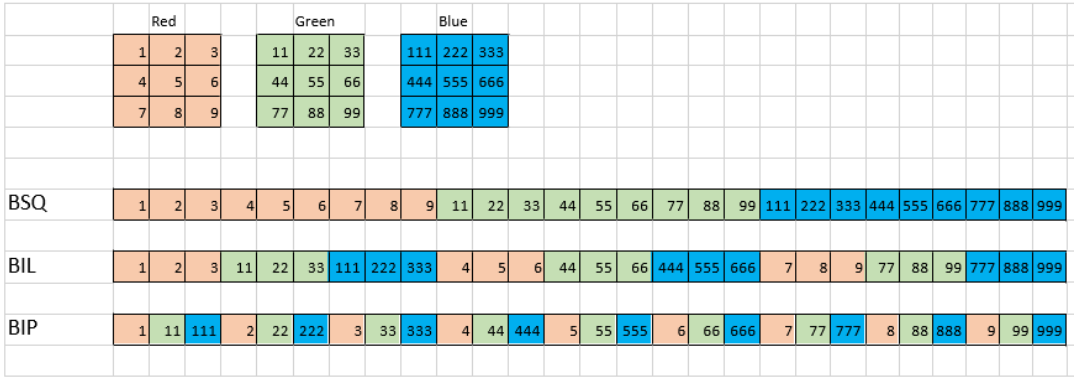
- Band sequential (BSQ) format stores information for the image one band at a time. In other words, data for all pixels for band 1 is stored first, then data for all pixels for band 2, and so on.
  - Value=image(c, r, b)
- Band interleaved by line (BIL) data stores pixel information band by band for each line, or row, of the image. For example, given a three-band image, all three bands of data are written for row 1, all three bands of data are written for row 2, and so on, until the total number of rows in the image is reached.
  - Value=image(c, b, r)
- Band interleaved by pixel (BIP) data is similar to BIL data, except that the data for each pixel is written band by band. For example, with the same three-band image, the data for bands 1, 2 and 3 are written for the first pixel in column 1; the data for bands 1, 2 and 3 are written for the first pixel in column 2; and so on.
  - Value=image(b, c, r)

More info @
https://desktop.arcgis.com/en/arcmap/latest/manage-data/raster-and-images/bil-bip-and-bsq-raster-files.htm

Exercise #1: File Data Structure

- **Data Input**
  - BSQ NEON Data
- **Read and write**
  - BSQ
  - BIP
  - BIL

interleave	array shape
'bip'	(R, C, B)
'bil'	(R, B, C)
'bsq'	(B, R, C)

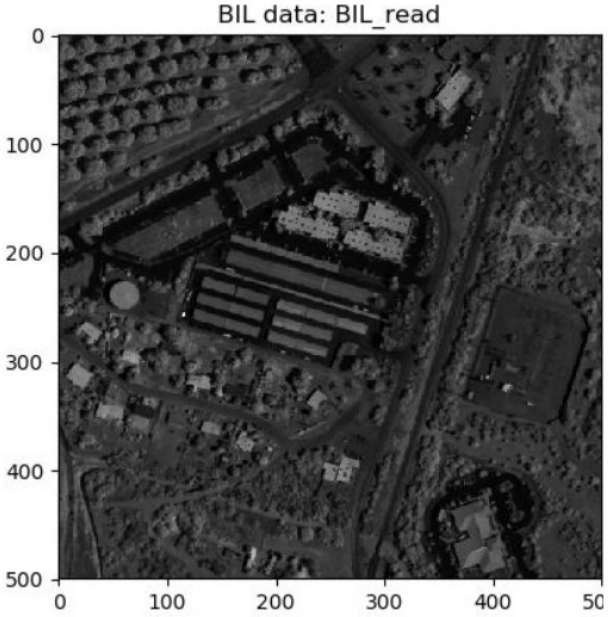


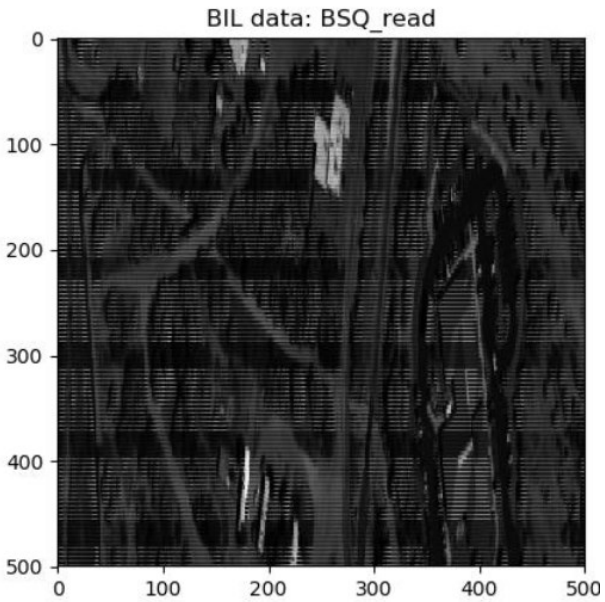
- **Homework**
  - Write a function to save a 2D array as BIP file format.
  - Test it by reading back the file using the BIP_read function.
  - It should display correctly

Instructions:

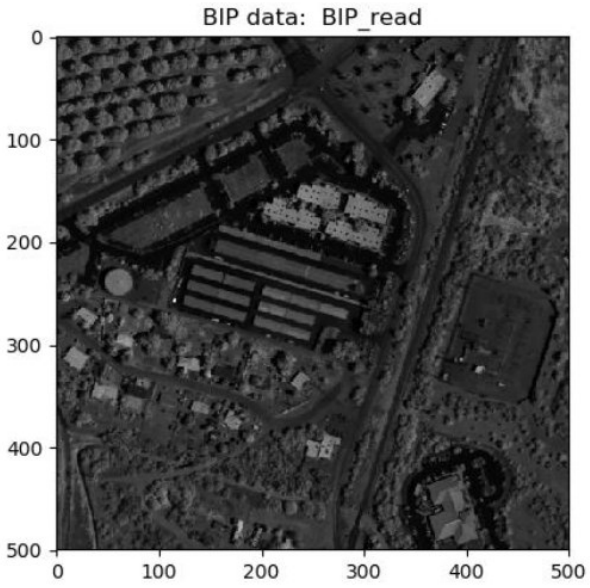
- Download from D2L files:
  - NEON_GreenValley.bsq (you should have this file already)
  - NEON_GreenValley.bip (same file but in BIP Structure)
  - BE485_Lab7_Ex1.ipynb

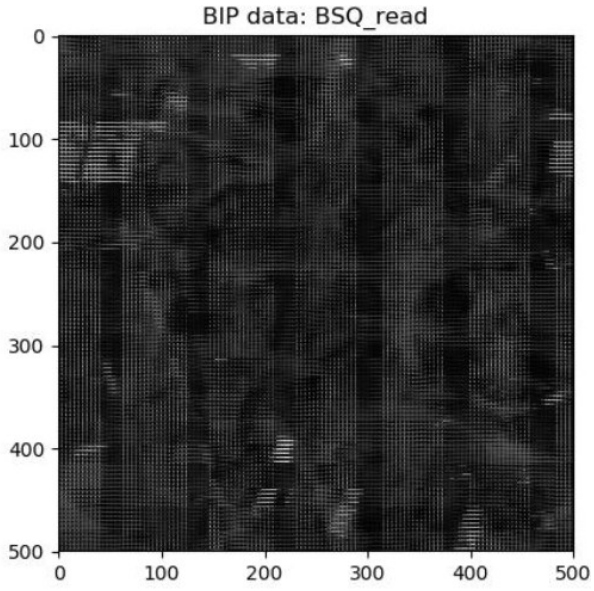
NIR band

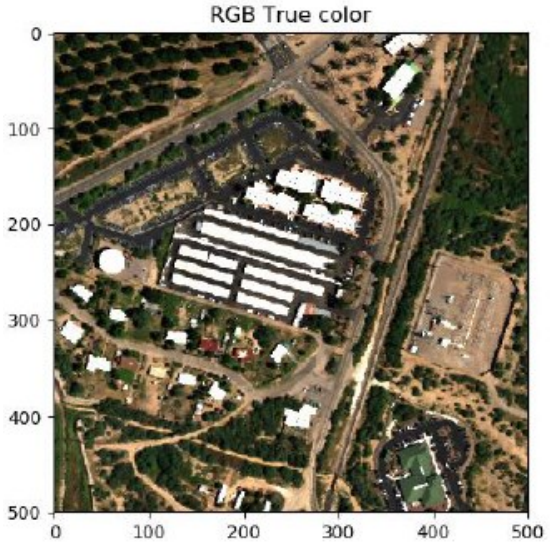




Read using the BSQ function





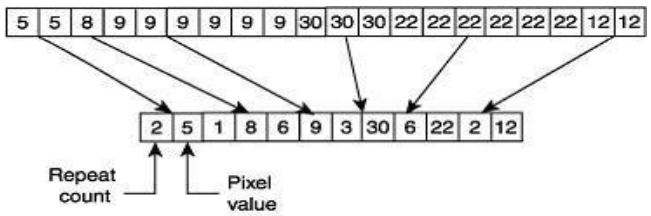


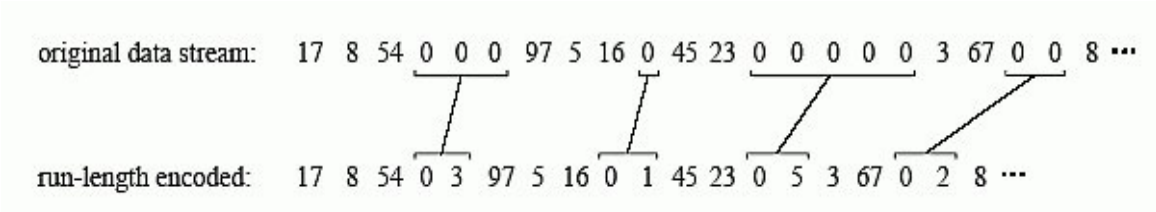
Data Compression

Data compression enables the encoding of information using fewer bits than the original representation. This is accomplished by identifying repetition and redundancy and rearranging the data. Compression is useful because it reduces file size, minimizes resources use, and storage and transmission cost (as we move more to a cloud-based computing environments).

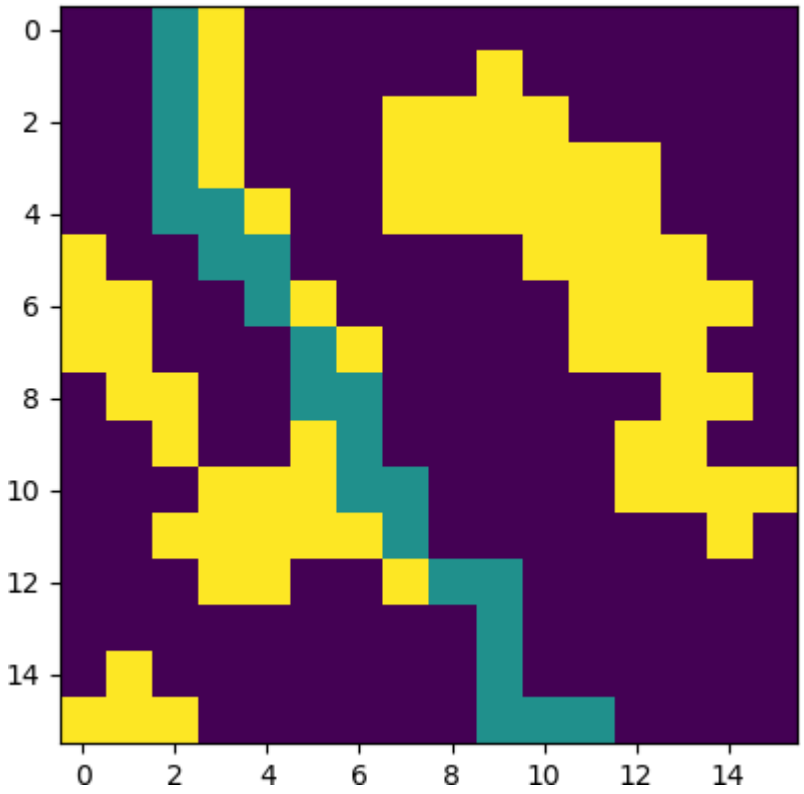
Run Length encoding (RLE) is a very simple form of lossless data compression in which runs of data (that is sequences in which the same data value occurs in many consecutive data elements) are stores as a single data value and count.

Run Length Encoding - RLE





Example Image

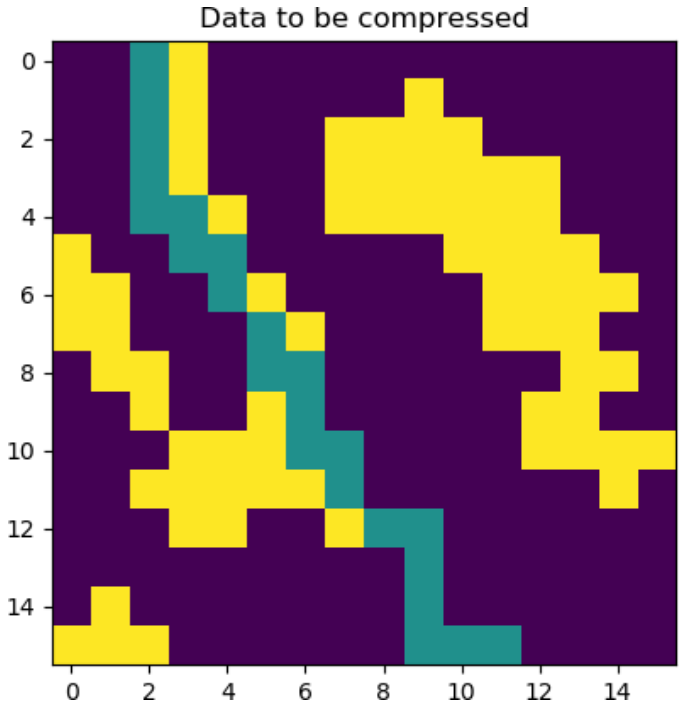


Values at each pixel

1	1	2	3	1	1	1	1	1	1	1	1	1	1	1	1
1	1	2	3	1	1	1	1	1	3	1	1	1	1	1	1
1	1	2	3	1	1	1	3	3	3	3	1	1	1	1	1
1	1	2	3	1	1	1	3	3	3	3	3	3	1	1	1
1	1	2	2	3	1	1	3	3	3	3	3	3	1	1	1
3	1	1	2	2	1	1	1	1	1	3	3	3	3	1	1
3	3	1	1	2	3	1	1	1	1	1	3	3	3	3	1
3	3	1	1	1	2	3	1	1	1	1	3	3	3	1	1
1	3	3	1	1	2	2	1	1	1	1	1	1	3	3	1
1	1	3	1	1	3	2	1	1	1	1	1	3	3	1	1
1	1	1	3	3	3	2	2	1	1	1	1	3	3	3	3
1	1	3	3	3	3	3	2	1	1	1	1	1	1	3	1
1	1	1	3	3	1	1	3	2	2	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	1
1	3	1	1	1	1	1	1	1	2	1	1	1	1	1	1
3	3	3	1	1	1	1	1	1	2	2	2	1	1	1	1

Exercise #2: Data Compression

- **Data Input**
  - BSQ Data layer
    - NRows = 16
    - NCols = 16



- **Homework**
  - Write code to compress the input data layer using the Run Length Encoding (RLE) Algorithm

Instructions:

- Download from D2L files:
  - Lab7_Ex2Data.bsq
  - BE485_Lab7_Ex2.ipynb