Wed. Mar. 28, 2018

- Ch. 8 Digital Image Processing
- Remaining topics including
  - Image Correction/Calibration
  - Digital Filtering
  - Principal Component Analysis

- Lab Projects for final two weeks of lab
3 Divisions of Image Processing

• **Image Restoration**
  – Corrects for instrumental “problems”, producing “true” image

• **Image enhancement**
  – Simple operations to highlight info of interest

• **Information extraction**
  – Convert image into information about surface
Image Restoration

• “Photometric” corrections
  – Correct periodic line striping (16 detectors / scan in TM)
  – Correct for atmospheric scattering
  – Filter random noise (could consider this “enhancement”)
  – Calibrate data (book doesn’t discuss this)
    • Reflectance for “in-sunlight” images
    • Radiance (W m$^{-2}$ $\mu$m$^{-1}$ str$^{-1}$) for thermal emission

• “Geometric” corrections
  – Restore line offsets, replace bad lines by interpolation, correct scan or camera distortions
Image Enhancement

- Contrast enhancement
- Density slicing
- Edge enhancement
- Form digital mosaics
- Transform (Intensity, Hue, Saturation)
- Merge data sets (upcoming lab)
- Create synthetic stereo images from topography information or multiple looks
Information Extraction

• Image Ratios (to extract composition info.)
• Band Depth Ratios (to estimate mineral abundance)
• Temperature, thermal inertia modeling
• Change detection (to look for variability)
• Principle-components analysis
• Multispectral classification

• More complex math operations such as:
  – NDVI (Normalized Digital Vegetation Index)
3 Divisions of Image Processing

• Image Restoration
  – Corrects for instrumental “problems”, producing “true” image

• Image enhancement
  – Simple operations to highlight info of interest

• Information extraction
  – Convert image into information about surface
TM Striping

- TM really has 16 detectors / band, recording 16 pixels per cross-scan
- Different (and varying) sensitivities of 16 detectors result in repeated “stripe” pattern
- More prominent in older Landsat data – better corrected in newer data
Transmission (or mirror scan errors) can shift lines, or groups of them
Examining location of “linear” features like coastline or roads can provide offsets
Again – newer Landsat data has fewer problems
Atmospheric Scattering

- Worst at short wavelengths (TM Band 1)
- Some fraction of light lost from all regions of image
- “Uniformly” added into all pixels, producing offset.
- Percent effect is greatest for what should be dark (shadowed) regions

A. Plot of TM band 7 versus band 1 for an area with shadows. Offset of the line of least-squares fit along band 1 axis is caused by atmospheric scattering in that band.

From our text by Sabins
Atmospheric Absorption

- Divide actual spectrum by transmission curve for atmosphere
- In regions where transmission is ~0, get 0/0 so very noisy

Corrected spectra from Lab 8 (Goldfield NV – AVIRIS)
Geometric Distortions

- **Systematic** distortions usually corrected in Landsat data before images released
- **Nonsystematic** distortions can remain

![Diagram of geometric distortions](image)

**Figure 8-10** Geometric distortions of Landsat images. From Bernstein and Fermyough (1975, Figure 3).
Distortion Correction

- Can correct distortion using known shape of regular (linear?) features
- Fiducial (reseau) marks sometimes superposed on original images

From our text by Sabins
Typical Intensity Calibration with Modern Cameras

- Subtract “dark frame” exposure taken with shutter closed, to eliminate electronic offset and also electrons that “leak” into the detector pixels over time.

- Divide image by “flat field” exposure of uniform brightness source, to correct for sensitivity variations among the pixels.

- “Non-linear” detectors can require more complicated corrections

- To produce “reflectance”, divide by expected solar reflected flux from an albedo=1 surface.

\[
I_{\text{Calibrated}} = \frac{I_{\text{Raw}} - I_{\text{Dark}}}{I_{\text{Flat}} - I_{\text{Dark}}}
\]

\[
R = I_{\text{Calibrated}}/I_{\text{Solar}}
\]

where \( I_{\text{Solar}} \) is expected signal from an albedo=1 surface.
3 Divisions of Image Processing

• **Image Restoration**
  – Corrects for instrumental “problems”, producing “true” image

• **Image Enhancement**
  – Simple operations to highlight info of interest

• **Information Extraction**
  – Convert image into information about surface
Image Enhancement

- Contrast enhancement
- Density slicing
- Edge enhancement
- Form digital mosaics
- Transform (Intensity, Hue, Saturation)
- Merge data sets
- Create synthetic stereo images from topography information or multiple looks
Filtering

Replace the 0 and the 90 by the average of the 9 around that location: 43 and 53

Can implement “threshold” replacement.
Sharpening

From our text by Sabins

\[
\begin{pmatrix}
0 & -1 & 0 \\
-1 & 5 & -1 \\
0 & -1 & 0
\end{pmatrix}
\times
\begin{pmatrix}
40 & 40 & 40 & 40 & 35 & 35 & 35 & 40 & 40 & 40 & 40 & 40 & 40 & 45 & 45 & 45 & 45 & 45 \\
40 & 40 & 45 & 35 & 35 & 35 & 35 & 40 & 40 & 40 & 40 & 40 & 40 & 40 & 45 & 45 & 45 & 45 \\
40 & 40 & 40 & 40 & 35 & 35 & 35 & 35 & 40 & 40 & 40 & 40 & 40 & 40 & 40 & 45 & 45 & 45 \\
40 & 40 & 40 & 40 & 40 & 35 & 35 & 35 & 35 & 40 & 40 & 40 & 40 & 40 & 40 & 45 & 45 & 45
\end{pmatrix}
\]

\[4 \times 40 - 1 \times 40 - 1 \times 40 - 1 \times 40 - 1 \times 35 = 5\] so add 5 to center pixel.

To avoid necessity of adding result to center value, many people would actually write kernel as following (i.e. increase center by 1):

\[
\begin{pmatrix}
0 & -1 & 0 \\
-1 & 5 & -1 \\
0 & -1 & 0
\end{pmatrix}
\]

You then just copy result to center pixel.
Figure 8-20  Edge enhancement using a directional filter.
Edge Detection

**Figure 8-18** Edge enhancements of TM images of the Jabal an Nabil area, northwest Saudi Arabia.
Color Transformations

• Because of properties of eye (three separate sensors for Red, Green, Blue), it takes 3 numbers to specify perceived properties of light

  – Simplest is RGB (tied closely to how “eye” works)
    • R: Intensity in red channel
    • G: Intensity in green channel
    • B: Intensity in blue channel

  – Sabins describes the alternate system IHS (tied closely to how “brain” works)
    • I: Intensity (i.e. bright vs. dim)
    • H: Hue (i.e. red vs. yellow vs. green vs. cyan vs. blue vs. magenta)
    • S: Saturation (i.e. deep red vs. pink or deep blue vs. light blue) – a kind of “Color contrast”
    • This system is often called HSL (Hue, Saturation, Lightness)

  – ENVI supports related systems
    • HSV (Hue, Saturation, Value)
    • HLS (Hue, Lightness, Saturation)
HSV System

- Hue measured as angle around cylinder
- Saturation measured as distance from central axis (=white)
- Value measured from bottom (black) to top (bright)
Merging color data

• Often one has:
  – High spatial resolution data in only one band
  – Lower spatial resolution data in multiple bands

• Often
  – Intensity varies rapidly and in complex way because of topography, shadowing, etc.
  – Color varies “slowly” because only a limited number of different compositions are present.
    • Note: The boundaries between different compositions could still be sharp but there are not a lot of those boundaries.

• Your “eye”, or at least your “brain” is use to observing the above difference.
  – It “demands” higher resolution in intensity than it does in color information.

• To merge data
  – Convert the low resolution color information into HSV or HLS
  – Discard the low resolution V or L “intensity” with the high resolution intensity from the single band
  – Convert the resulting HSV or HLS data back into RGB to display

• To your “eye” it will look like much like a high resolution color image
• You may see “artifacts” at places where there are sharp intensity AND color boundaries
Raster vs. Vector Data

• Raster data:
  – An matrix of cells, each of which contains some value
    • Good for:
      – Intensity in an image
      – Radar reflectivity vs. position
      – Height vs. position
  – There is some implicit relationship between row and column index \((i, j)\) and position in space \((x, y)\)
    • If there is a constant scale the relationship is simple and linear

• Vector data:
  – A list of \((x, y)\) coordinates, and often some value specified at that coordinate
    • Good for:
      – Plotting lines in space (roads, rivers, borders, contours, etc)
      – Heights of discrete locations such as mountain peaks
      – Locations of discrete features

• ENVI deals primarily with raster data, but can incorporate vector data from “maps”
3 Divisions of Image Processing

• Image Restoration
  – Corrects for instrumental “problems”, producing “true” image

• Image Enhancement
  – Simple operations to highlight info of interest

• Information Extraction
  – Convert image into information about surface
Multispectral Classification: Cluster Analysis

- Salton Sea, CA.
- By eye – four general types of terrain:
  - Water
  - Agriculture
  - Desert
  - Mountains

From our text by Sabins
Multispectral Classification: Cluster Analysis

- Use bands #2, #3, #4
  - Plot #2, #3, #4 for each pixel
  - Look for clusters in data
    - 4 clusters with gaps between
  - Define 4 ellipsoids:
    - If pixel falls inside given cluster it is classified as type 1,2,3,4
    - If pixel falls outside ellipsoid it will not be classified (unknown or mix)

- Surface of ellipsoid: Decision boundary
- Volume in ellipsoid: Decision space

- Note – we can only plot 3 – but computer could look for 6-D clusters using all 6 bands

From our text by Sabins
Figure 8-31 Multispectral classification of Landsat data for the Salton Sea and the Imperial Valley, California.

From our text by Sabins
K-Means: Way to determine clustering

- Unsupervised Classification
- Iterative cluster analysis technique
- Works best if clusters are “round”, not long and skinny.

Can use other techniques first (like PCA) to get around that.

1) Divide data into “K” classes
   (in first iteration this assignment is “random”)
2) Find “center” of each class in n-dimensional data space as defined by the “Mean”
3) Find distance of each data point (each pixel) from center of all the classes and reassign it to the closest class

Repeat steps 2 and 3 until either:
   “Few” pixels jump from one class to another
   or
   Reach specified max. number of iterations
Image Classification:

- **Supervised Classification**
  - Specify “training sites” with ground-truth or specify “type spectra”

- **Unsupervised Classification**
  - Search for statistical regularities in data without additional information

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised</td>
<td>Need ground-truth or lab spectra.</td>
</tr>
<tr>
<td>Identifies types of terrain.</td>
<td>Gets confused if additional unknown classes are present</td>
</tr>
<tr>
<td>Using extra information, can recognizing subtle differences.</td>
<td></td>
</tr>
<tr>
<td>Unsupervised</td>
<td>Doesn’t identify surface type.</td>
</tr>
<tr>
<td>Needs no additional information.</td>
<td>Confused by subtle differences.</td>
</tr>
</tbody>
</table>
Thermopolis Example

- **Supervised classification**
  - You pick “training sites” of known character.

- **Unsupervised classification**
  - Program decides on it’s own different types of terrain.
  - You usually specify # of types, etc.

From our text by Sabins
Thermopolis: Supervised Classification

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>CLASS</th>
<th>PERCENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Redbeds</td>
<td>8.4</td>
<td></td>
</tr>
<tr>
<td>Sandstone</td>
<td>48.3</td>
<td></td>
</tr>
<tr>
<td>Shale</td>
<td>18.9</td>
<td></td>
</tr>
<tr>
<td>Agriculture</td>
<td>16.2</td>
<td></td>
</tr>
<tr>
<td>Native vegetation</td>
<td>5.2</td>
<td></td>
</tr>
<tr>
<td>Water and shadows</td>
<td>1.9</td>
<td></td>
</tr>
<tr>
<td>Unclassified</td>
<td>1.1</td>
<td></td>
</tr>
</tbody>
</table>

A. Supervised-classification map and explanation.

From our text by Sabins
Thermopolis: Unsupervised Classification

<table>
<thead>
<tr>
<th>SYMBOL</th>
<th>CLASS</th>
<th>PERCENT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Redbeds, lower</td>
<td>2.2</td>
</tr>
<tr>
<td></td>
<td>Redbeds, upper</td>
<td>11.7</td>
</tr>
<tr>
<td></td>
<td>Sandstone</td>
<td>38.2</td>
</tr>
<tr>
<td></td>
<td>Shale</td>
<td>22.3</td>
</tr>
<tr>
<td></td>
<td>Agriculture, A</td>
<td>6.5</td>
</tr>
<tr>
<td></td>
<td>Native vegetation</td>
<td>13.7</td>
</tr>
<tr>
<td></td>
<td>Agriculture, B</td>
<td>2.7</td>
</tr>
<tr>
<td></td>
<td>Agriculture, C</td>
<td>2.7</td>
</tr>
</tbody>
</table>

Plate 15  Digital classifications of TM image, Thermopolis, Wyoming.
Principal-components

- Statistical method for determining “different ways” in which data vary
- Suppose you measure weight, height for each person in room, and plot that data.
  - Most of the variation falls along a single line which could be described as “big people” vs. “small people”
    - Define 1\textsuperscript{st} eigenvector: \( \text{bigness} = W + H \)
  - Smaller variation falls along a line at right angles to this:
    - Define 2\textsuperscript{nd} eigenvector: \( \text{heaviness} = W - H \)
  - If you had a third measurement – kg of body-fat, you could define a 3\textsuperscript{rd} eigenvector which distinguished between being heavy because you were fat, or heavy because you had lots of muscle.
    - In general:
      \[
      y_1 = a_{11} x_1 + a_{12} x_2 \\
      y_2 = a_{21} x_1 + a_{22} x_2
      \]
    - For remote sensing
      - Different measurements \( x_1, x_2, \ldots \) different bands
      - First eigenvector \( y_1 \) might be light rocks vs. dark rocks
      - Second eigenvector might be tan vs. red rocks
      - Third eigenvector might be “clay rich” vs. “clay poor”
  - Total # of “eigenvectors” equal original # of bands
  - First eigenvectors explains most of variation
  - Last eigenvectors may just be “noise”
Thermopolis TM Data

- 6 TM bands (excluding TIR), so expect 6 PC Eigenvectors

From our text by Sabins
Thermopolis TM Data

- PC Components from Fig. 08.27
- First component is often mostly topography related shading
- Second component is often the dominant “color” differences

From our text by Sabins
Principal-components

- Statistical method for determining “different ways” in which data vary
- Suppose you measure weight, height for each person in room, and plot that data.
  - Most of the variation falls along a single line which could be described as “big people” vs. “small people”
    - Define 1st eigenvector: bigness = W + H
  - Smaller variation falls along a line at right angles to this:
    - Define 2nd eigenvector: heaviness = W – H
  - If you had a third measurement – kg of body-fat, you could define a 3rd eigenvector which distinguished between being heavy because you were fat, or heavy because you had lots of muscle.
  - In general:
    - \[ y_1 = a_{11} x_1 + a_{12} x_2 \]
    - \[ y_2 = a_{21} x_1 + a_{22} x_2 \]
- For remote sensing
  - Different measurements \( x_1, x_2, \ldots \) different bands
  - First eigenvector \( y_1 \) might be light rocks vs. dark rocks
  - Second eigenvector might be tan vs. red rocks
  - Third eigenvector might be “clay rich” vs. “clay poor”
- Total # of “eigenvectors” equal original # of bands
- First eigenvectors explains most of variation
- Last eigenvectors may just be “noise”
Thermopolis TM Data

- 3rd, 4th, etc. components may be more subtle “color” variations
- Last components are mostly noise, or minor “color” differences