Inel 6007
Introduction to Remote Sensing
Chapter 5
Spectral Transforms –PCT, contrast enhancement
Prof. Vidya Manian
Dept. of Electrical and Computer Engineering
Affine Transformations

\[ \mathbf{z} = \mathbf{A}^T \mathbf{x} + \mathbf{b} \]

K-dimensional \hspace{2cm} p-dimensional
Affine Transformations deal with Rotation and Scaling of the Spectral Signatures
Linear Projection

\[ |V^T P| \frac{V}{|V|} \]

Band j

Band i
Projection to a Lower Dimension

PCA

SVDSS

INEL6007 (Spring 2010)  ECE, UPRM
Projection to a Lower Dimension

OIDPP

IDSS

INEL6007 (Spring 2010)  ECE, UPRM
Principal Component Transformation

PC Transformation is multivariate statistical analysis technique used in remote sensing for:

- Dimensionality reduction.
- Denoising
- To visualize data with more than 3 bands.
  - PCA stretch (more later)
- Also known as the Karhunen-Loeve Transformation (KLT) and the Hotelling transformation
Some basic multivariable statistics

• The first principal component is selected by finding a linear projection that maximizes variance

\[ z_1 = a_1^T x + b_1 \]

• The variance of \( z_1 \) is given by

\[ \text{var}(z_1) = a_1^T C_x a_1 \]

• \( a_1 \) is determined by solving the optimization problem

\[ a_1 = \arg \max_q q^T C_x q, \quad \text{subject to} \quad q^T q = 1 \]
Digression: Eigenvalues & Eigenvectors

\[ \mathbf{A} \mathbf{v}_i = \lambda_i \mathbf{v}_i \]

\[ [\mathbf{A} - \lambda_i \mathbf{I}] \mathbf{v}_i = 0 \]

where \( \lambda_i \) is the i-th eigenvalue and \( \mathbf{v}_i \) is the corresponding i-th eigenvector

Finding the Eigenvalues

\[ \det [\mathbf{A} - \lambda_i \mathbf{I}] = 0 \]

Finding the Eigenvectors by solving

\[ [\mathbf{A} - \lambda_i \mathbf{I}] \mathbf{v}_i = 0 \]
Basic multivariable statistics (cont.)

• It can be shown that the optimal solution for the first PC is

\[ \mathbf{a}_1 = \mathbf{v}_1 \]

the first eigenvector of the covariance matrix \( \mathbf{C}_X \) and that

\[ \text{var}(z_1) = \lambda_1 = \lambda_{\text{max}} \]

its maximum eigenvalue.

• The remaining principal components \( z_i \) for \( i=2,3,\ldots,K \) are given by

\[ z_i = \mathbf{v}_i^T \mathbf{x} + b_i \]

with \( \text{var}(z_i) = \lambda_i \) where \( \lambda_i \) and \( \mathbf{v}_i \) are the i-th eigenvalue and eigenvector respectively.

• Notice that eigenvalues are ordered in descending order:

\[ \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_K \]

and the eigenvectors are orthonormal, i.e.

\[ \mathbf{v}_i^T \mathbf{v}_i = 1 \text{ and } \mathbf{v}_i^T \mathbf{v}_j = 0 \]
Principal Components Transform

\[ z = V^T x + b \]

where \( V = [v_1, v_2, \ldots, v_n] \) is the matrix of covariance eigenvectors as defined previously.

- The bias term can be selected as

\[ b = 0 \quad \text{or} \quad b = -V^T \mu_x \]
Pictorial Representation

Band 2

μ₂

μ₁

Band 1

v₁

v₂
PCA Transform Summary

• Data are transformed along orthogonal axes which are dependent on the variance within the image.
• The first Principal Component is along the axis of maximum variance.
• Each succeeding axis has less variance.
Principal Components Properties

\[ z = V^T x + b \]
\[ z_i = v_i^T x \quad i = 1, 2, \ldots, n \]
\[ C_Z = V^T C_x V = \text{diag}\{\lambda_1, \ldots, \lambda_n\} \]

- PCs are uncorrelated
  \[ \text{cov}(z_i, z_j) = [C_Z]_{ij} = 0, \text{ for } i \neq j \]
Simple MATLAB Algorithm

- \( X = \) image matrix with one column per band
- \( C = \text{cov}(X); \)
- \([U, S, V] = \text{svd}(C);\)
  - For covariance matrices SVD = Eigenvalue Eigenvector decomposition
  - SVD produces singular values in descending order
  - \( V \) is the transformation matrix
- \( \text{PCA} = X*U = X*V^T \)
PC Transformation

• For most remote sensing data sets, the first few PC components contain > 98% of the variance.

\[
\text{% variability}(p) = 100 \times \frac{\sum_{i=1}^{p} \lambda_i}{\sum_{i=1}^{n} \lambda_i}
\]

• Most of the time the higher PCs (in number) contain only noise.
PCT Example

- Original data in DN-space

<table>
<thead>
<tr>
<th>Pixel</th>
<th>DN1</th>
<th>DN2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

\[
\mu = \begin{bmatrix} 3.50 \\ 3.50 \end{bmatrix}
\]

\[
C = \begin{bmatrix} 1.9 & 1.1 \\ 1.1 & 1.1 \end{bmatrix}
\]

\[
R = \begin{bmatrix} 1 & 0.761 \\ 0.761 & 1 \end{bmatrix}
\]
PCT Example (cont.)

- Step 1. Find eigenvalues
  - Solve the characteristic equation (two solutions)
    \[
    \begin{vmatrix}
    C - \lambda I
    \end{vmatrix} = 0
    \]
    \[
    \begin{vmatrix}
    1.9 - \lambda & 1.1 \\
    1.1 & 1.1 - \lambda
    \end{vmatrix} = 0
    \]
    \[
    \lambda^2 - 3\lambda + 0.88 = 0
    \]
    \[
    \lambda_1 = 2.67
    \]
    \[
    \lambda_2 = 0.33
    \]
  - Therefore
    \[
    C_{PC} = \begin{bmatrix}
    2.67 & 0 \\
    0 & 0.33
    \end{bmatrix}
    \]
  - Note that PC1 contains 89% of the total data variance, and PC2 contains 11%, i.e.
    \[
    \frac{\lambda_1}{\lambda_1 + \lambda_2} = \frac{2.67}{2.67 + 0.33} = 0.89
    \]
PCT Example (cont.)

• Step 2. Find eigenvectors
  – Substitute eigenvalues into \([\lambda I - C_X]v=0\)

• First eigenvalue yields dependent equations

\[
\begin{bmatrix}
1.9 - 2.67 & 1.10 \\
1.10 & 1.1 - 2.67
\end{bmatrix}
\begin{bmatrix}
e_{11} \\
e_{21}
\end{bmatrix} = 0 
-0.77e_{11} + 1.10e_{21} = 0 
1.10e_{11} - 1.57e_{21} = 0
\]

• Solving either equation \(e_{11} = 1.43v_{21}\)
• PCT requires orthonormality of the eigenvectors \(e_i^T e_i = 1\)

\[
e_1 = \begin{bmatrix}
0.82 \\
0.57
\end{bmatrix} 
\quad e_2 = \begin{bmatrix}
-0.57 \\
0.82
\end{bmatrix} 
\quad W_{PC} = \begin{bmatrix}
0.82 & 0.57 \\
-0.57 & 0.82
\end{bmatrix}
\]
Principal Component 2-band Example

$$\mu_X = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \Sigma_X = \begin{bmatrix} 5 & 3 \\ 3 & 4 \end{bmatrix}$$
Principal Component Example

Eigenvector of the largest eigenvalue

\[ \mu_x = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad C_x = \begin{bmatrix} 5 & 3 \\ 3 & 4 \end{bmatrix} \]

Eigenvalues =

\[ \begin{bmatrix} 5.8498 & 0 \\ 0 & 1.2879 \end{bmatrix} \]

Eigenvalues =

\[ \begin{bmatrix} -0.7612 & 0.6486 \\ -0.6486 & -0.7612 \end{bmatrix} \]

The first PC contains

% var = 5.8498/(5.8498+1.2879)*100

= 82% of the total variability
Linear Projection

\[ v_i v_i^T x \]
Histogram of the data projected in the principal component

\[ z_1 = v_1^T x \]
2nd Example: First Principal Component
Histogram of the data projected in the principal component

\[ z_1 = v_1^T x \]
2nd Example: Second Component
Histogram of the data projected in the second component

$z_2 = v_2^T x$
LANDSAT TM Simulator
PCA Example
Añasco, PR
Covariance Matrix

MATLAB 7.4.0 (R2007a)

>> C = cov(X)

C =

3.4605e+02 2.2988e+02 3.1223e+02 2.2719e+02 -8.5901e+01 -1.8544e+02 3.7154e+00
2.2988e+02 2.1671e+02 3.3439e+02 3.2095e+02 2.4839e+02 2.3274e+02 9.3089e+01
3.1223e+02 3.3439e+02 5.7595e+02 5.8751e+02 4.9501e+02 4.8577e+02 2.1495e+02
2.2719e+02 3.2095e+02 5.8751e+02 6.8552e+02 8.6826e+02 9.5655e+02 3.1919e+02
-8.5901e+01 2.4839e+02 4.9501e+02 8.6626e+02 2.2270e+03 2.7072e+03 5.7257e+02
-1.8544e+02 2.3274e+02 4.8577e+02 9.5655e+02 3.3229e+03 3.1919e+02 6.7323e+02
3.7154e+00 9.3089e+01 2.1495e+02 3.1919e+02 5.7257e+02 6.7323e+02 2.1928e+02

>>
PCA Variances and Transformation Matrix

```matlab
>> [U,S,V]=svd(C); diag(S)

ans =

     6.1731e+003    1.2510e+003    1.3250e+002    3.0028e+011    4.3484e+000    1.4807e+000    1.0605e+000

>> V

V =

   1.5223e-001 -5.8170e-001  -1.9601e-001   3.5607e-001   8.6099e-002 -6.6465e-001  1.5522e-001
   2.5057e-001 -4.6960e-001  -3.9198e-001  8.0647e-002 -4.9321e-001   5.0900e-001 -2.3280e-001
   7.2308e-001  2.9012e-001  1.9343e-001 -1.3341e-002 -4.3098e-002 -2.6015e-001 -5.4893e-001

INEL6007(Spring 2010)  ECE, UPRM
### Cumulative Variances

#### PC Variances for Landsat Image

<table>
<thead>
<tr>
<th>Singular Value</th>
<th>Magnitude</th>
<th>% Var.</th>
<th>Cumulative Variability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\lambda_1$</td>
<td>$6.1731 \times 10^3$</td>
<td>81.2949</td>
<td>81.2949</td>
</tr>
<tr>
<td>$\lambda_2$</td>
<td>$1.2510 \times 10^3$</td>
<td>16.4740</td>
<td>97.7689</td>
</tr>
<tr>
<td>$\lambda_3$</td>
<td>$0.1325 \times 10^3$</td>
<td>1.7449</td>
<td>99.5138</td>
</tr>
<tr>
<td>$\lambda_4$</td>
<td>$0.0300 \times 10^3$</td>
<td>0.3954</td>
<td>99.9093</td>
</tr>
<tr>
<td>$\lambda_5$</td>
<td>$0.0043 \times 10^3$</td>
<td>0.0573</td>
<td>99.9665</td>
</tr>
<tr>
<td>$\lambda_6$</td>
<td>$0.0015 \times 10^3$</td>
<td>0.0195</td>
<td>99.9860</td>
</tr>
<tr>
<td>$\lambda_7$</td>
<td>$0.0011 \times 10^3$</td>
<td>0.0140</td>
<td>100.0000</td>
</tr>
</tbody>
</table>
7 principal components for the LANDSAT TM Simulator Image over Añasco

INEL6007(Spring 2010)
PCT Benefits

• Why use the PCT?
  – Decorrelates spectral data

• Multispectral bands are often highly-correlated because of
  – material spectral correlation
  – topography
  – sensor band overlap

• Decorrelation separates “independent” components into separate “bands”
  – Compresses the variance
    \[ \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_K \]
  – Compression can potentially reduce data computation burden
PCT Decorrelation

- Non-vegetated scene
  - PCT removes spectral redundancy
PCT Contrast Extraction

- Vegetated scene
  - PCT extracts contrast between bands 3 and 4 (red and NIR) due to the vegetation “red edge”
PCT Noise Detection

- PCT can isolate spectrally-uncorrelated noise
PC Dimensionality Reduction

• Choose $p$ by selecting the number of PCs that explain the variability above a threshold $T$. 

$$\% \text{ variability}(p) = 100 \times \frac{\sum_{i=1}^{p} \lambda_i}{\sum_{i=1}^{n} \lambda_i} \geq T$$

• $T$ is usually selected above 95%

• Most of the time the higher PCs
  – $p+1, p+2, p+3, \ldots, n$
  contain mostly “noise.”
PC Dimensionality Reduction (cont.)

• Set $V_p=[v_1, v_2, \ldots, v_p]=V( :, 1:p)$ the matrix

\[
\begin{pmatrix}
Z_1 \\
Z_2 \\
\vdots \\
Z_p
\end{pmatrix}
= \begin{pmatrix}
V_1^T X \\
V_2^T X \\
\vdots \\
V_p^T X
\end{pmatrix}
= V_p^T X
\]

• In MATLAB notation $Z_p=X*V( :, 1:p)^T$
• Need to use reshape to get back the images.
PC Dimensionality Reduction

• Many applications use the extracted PCA, $z \in \mathbb{R}^p$, features instead of the original spectral signature, $x \in \mathbb{R}^n$ with $n \gg p$
PCA for signal denoising

• PCA Summary

  – Original PCT
    \[ z = V^T x, \]
  – Inverse PCT
    \[ x = Vz \]
  – “Signal Components”
  – “Noise Components”
    • \( Z_{p+1}, Z_{p+2}, \ldots, Z_n \)
PCA for signal denoising (cont.)

- Typical denoising involves computation of the PC and neglection of the higher order PCs (associated with noise) and reconstruction of the image.
  - Dimensionality reduction
    \[ z_p = V_p^T x \]
  - Filtered reconstruction
    \[ x_p = V_p z_p \]
    where \( x_p \) is the filtered version of \( x \), \( V_p = [v_1, v_2, \ldots, v_p] \) is a matrix formed by the first \( p \) eigenvectors of the covariance matrix, and \( z_p = [z_1, z_2, \ldots, z_p]^T \) are the first \( p \) PCs.

- Optimal from a reconstruction point of view
  - Mean squared error optimality
Denoising example (T=98%)
Standardized PC (SPC)

- The standardized principal components \( r_i \) for \( i=1,2,3,...,n \) are given by
  \[
  \bar{u}_i = \frac{x_i - \mu_i}{\sigma_i} \quad r_i = q_i^T \bar{u}
  \]

  where \( q_i \) is the \( i \)-th eigenvector of the correlation matrix \( R_X \) and \( u_i \) are the normalized random variables with unit variance and zero mean.

- Notice that correlation eigenvalues are ordered in descending order:
  \[
  \rho_1 \geq \rho_2 \geq \ldots \geq \rho_n
  \]
  and eigenvectors are also orthonormal
  \[
  q_i^T q_i = 1 \quad \text{and} \quad q_i^T q_j = 0
  \]

- Notice that \( \text{var}(r_k) = \rho_k \)

- Particularly useful with data of differing scales.
Why not use PCT?

- It is data-dependent
- \( V \) coefficients change from scene-to-scene. \( V \) is computed from the covariance of the data \( C_X \).
- Makes consistent interpretation of PC images difficult
  - Spectral details, particularly in small areas, may be lost if higher-order PCs are ignored
- **Computationally expensive** for large images or for many spectral bands
  - Calculation of covariance matrix is the culprit
    - Alternate method using SVD but still VERY computationally expensive
- Not necessarily optimal from a classification point of view
  - Classes are not necessarily separated
Review questions

• What is a principal component transformation. What does it achieve with respect to the RS data.
• What are the advantages and disadvantages of the method.
• How is the PCT calculated.
• Ex 5-1, 5-2, 5-3, 5-4.
Tasseled-Cap Components
Different Projections
Tasseled-Cap Components

- Linear spectral transform like the PCT
  \[ w = A^T x \]
- The TCT matrix is fixed for a given sensor

<table>
<thead>
<tr>
<th>sensor</th>
<th>axis name</th>
<th>( W_{TC} )</th>
<th>bias</th>
</tr>
</thead>
</table>
|        |                      | \[ \begin{array}{c}
| L-1    | soil brightness     | +0.433 +0.632 +0.586 +0.264  \\
| MSS    | greenness           | -0.260 -0.562 +0.600 +0.401  \\
|        | yellow stuff        | -0.829 +0.322 -0.039 +0.194  \\
|        | non-such            | +0.223 +0.120 -0.543 +0.810  \\
| L-2    | soil brightness     | +0.337 +0.603 +0.676 +0.783  \\
| MSS    | greenness           | -0.283 -0.660 +0.577 +0.388  \\
|        | yellow stuff        | -0.400 +0.428 +0.303 +0.094  \\
|        | non-such            | +0.016 +0.428 -0.452 +0.882  \\
| L-4    | soil brightness     | +0.303 +0.2793 +0.4743 +0.5585 +0.5082 +0.1863  \\
| TM     | greenness           | -0.7848 -0.7415 -0.5416 +0.7343 +0.0840 +0.1803  \\
|        | wetness             | +0.1599 +0.1973 +0.2279 +0.3406 +0.7112 +0.4572  \\
|        | haze                | -0.8642 -0.0849 -0.4392 -0.0580 -0.2012 -0.2798  \\
|        | TC5                 | -0.3280 +0.0549 +0.1075 +0.1855 -0.4357 +0.8085  \\
|        | TC6                 | +0.1084 -0.0922 -0.4120 +0.0573 -0.0251 +0.0238  \\
| L-5    | soil brightness     | +0.2909 +0.2493 +0.4806 +0.5568 +0.4438 +0.1706  \\
| TM     | greenness           | -0.2728 -0.2174 -0.5508 +0.7221 +0.0733 -0.1648  \\
|        | wetness             | +0.1446 +0.1761 +0.3322 +0.3396 -0.6210 -0.4186  \\
|        | haze                | -0.8461 -0.0731 -0.4840 -0.0032 -0.0492 +0.0119  \\
|        | TC5                 | +0.0549 -0.0232 +0.0339 -0.1957 +0.4162 -0.7823  \\
|        | TC6                 | +0.1186 -0.0869 +0.4094 +0.0571 -0.0228 +0.0220  \\
| L-7    | soil brightness     | +0.3561 +0.3972 +0.3904 +0.6966 +0.2286 +0.1596  \\
| ETM+   | greenness           | -0.3344 -0.3544 -0.4556 +0.6966 -0.0242 -0.2630  \\
|        | wetness             | +0.2626 +0.2141 +0.0926 +0.0656 -0.7629 -0.5388  \\
|        | haze                | +0.0805 -0.0498 -0.1950 -0.1327 +0.5752 -0.7775  \\
|        | TC5                 | -0.7252 -0.0202 +0.6683 +0.0631 -0.1404 -0.0274  \\
|        | TC6                 | +0.4000 -0.1712 +0.4332 +0.0602 -0.1095 +0.0985  \\

TC transformation Matrix for Different Sensors
TCT Benefits

• Why use the TCT?
  – It is a fixed reference

• Same reference for every image from a given sensor permits consistent interpretation
  – Components are related to geophysical properties of the scene

• First component is “soil brightness”

• Second component is “greeness”
### Example

<table>
<thead>
<tr>
<th>sensor</th>
<th>axis name</th>
<th>( W_{TC} )</th>
<th>bias</th>
</tr>
</thead>
</table>
|        |                          | \begin{bmatrix}
| L-1    | soil brightness         | +0.433 +0.632 +0.586 +0.264  |
| MSS    | greenness               | -0.290 -0.562 +0.600 +0.491  |
|        | yellow stuff            | -0.829 +0.522 -0.039 +0.194  |
|        | non-such                | +0.223 +0.120 -0.543 +0.810  |
| L-2    | soil brightness         | +0.332 +0.603 +0.676 +0.263  |
| MSS    | greenness               | +0.283 -0.660 +0.577 +0.388  |
|        | yellow stuff            | +0.900 +0.428 +0.0759 -0.041 |
|        | non-such                | +0.016 +0.428 -0.452 +0.882  |
|        |                          | \end{bmatrix}                |      |
| L-4    | soil brightness         | \begin{bmatrix}
| TM     | greeness                | +0.3037 +0.2793 +0.4743 +0.5585 +0.5082 +0.1863 |
|        | wetness                 | -0.2848 -0.2435 -0.5436 +0.7243 +0.0840 -0.1800 |
|        | haze                    | +0.1509 +0.1973 +0.3279 +0.3406 -0.7112 -0.4572 |
|        | TC5                     | +0.8242 -0.0849 -0.1392 -0.0580 +0.2012 -0.2768 |
|        | TC6                     | -0.3280 +0.0549 +0.1075 +0.1855 -0.4357 +0.8085 |
|        |                          | +0.1084 -0.0022 -0.4320 +0.0573 -0.0251 +0.0238 |
|        |                          | \end{bmatrix}                |      |
TCT Drawbacks

• Why not use the TCT?
  – Nonoptimal compression of data
  – Derivation of WTC requires multitemporal data for each sensor
Contrast Enhancement
Problems in Visual Analysis of Multispectral Imagery

- Most images do not fill the dynamic range of the sensor
- Most images also do not fill the dynamic range of the display system
Contrast Enhancement

• Contrast enhancement means “stretching” the data range to fill the display system range
• Contrast enhancement is a mapping from the original DN data space to a gray level (GL) display space

\[ GL = T(DN) \]

– Examples:
  • Linear Stretch
  • Histogram Equalization Stretch
• Parameters of transformation T based on global or local image statistics (histogram)
Contrast Stretches

Example of a Linear Stretch Function
Histogram for Linear Stretch

First band (Green) for sample SPOT image
Raw Unstretched Data

First band (Green) for sample SPOT image
Linearly Stretched Data

First band (Green) for sample SPOT image
Different Mappings can be Defined for Contrast Enhancement
Histogram Equalization

• One attempts to change the histogram through the use of a function $GL = T(DN)$ into a histogram that is constant for all brightness values.

• This would correspond to a brightness distribution where all values are equally likely.
Histogram Equalization Stretch

- Histogram equalization is achieved by using the Cumulative Density function of the image as a transformation function, after appropriate scaling of the ordinate axis.
Histogram Equalization Stretch

The most numerous pixels have the greatest contrast
Histogram Equalization Stretch

First band (Green) for sample SPOT image
Comparison Between Histogram Equalization Methods

Linear Stretch

Histogram Equalization Stretch

INEL6007(Spring 2010) ECE, UPRM
# Contrast Enhancement Maps

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Equation</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>min-max</td>
<td>[ GL = \frac{255}{DN_{\text{max}} - DN_{\text{min}}} (DN - DN_{\text{min}}) ]</td>
<td>sensitive to outliers</td>
</tr>
<tr>
<td>histogram equalization</td>
<td>[ GL = 255 \cdot CDF(DN) ]</td>
<td>produces uniform histogram</td>
</tr>
</tbody>
</table>
| normalization      | 1. \[ GL = \frac{\sigma_{\text{ref}}}{\sigma} (DN - \mu) + \mu_{\text{ref}} \]  
2. \[ GL = 255, \, GL > 255 \]  
\[ GL = 0, \, GL < 0 \] | matches means and variances |
| threshold          | \[ GL = 255, \, DN \geq DN_T \]  
\[ GL = 0, \, DN < DN_T \] | binary output                |
| reference          | \[ GL = CDF_{\text{ref}}^{-1}[CDF(DN)] \] | matches histograms           |
Contrast Stretch Examples

Application to GOES Data
Normalization Stretch

- Linear scaling of DN mean and sigma to specified values, followed by saturation (clipped at the ends)

\[ GL = aDN + b, \]
\[ \mu_{GL} = a\mu_{DN} + b, \]
\[ \sigma^2_{GL} = a^2\sigma^2_{DN} \]

- Consistent behavior (robust) over wide range of images
Reference Stretch

- Generalization of the Normalization Stretch
  - Not only the second order statistics but the whole pdf (or CDF)
- Match the CDF of the image being processed to a reference CDF, for example from another image, and then backward mapping
  - useful for
    - multitemporal or multisensor radiance matching
    - matching image to reference contrast
- Gaussian Normalization is a common case
Multitemporal Normalization

(INEL 6007 (Spring 2010) ECE, UPRM)
Thresholding

- Binary "clipping" of DN values to low and high values
  - Useful for
    - segmentation of certain images, e.g. clouds/water, land/water
Color Images

• Techniques used for single-band imagery can be extended to color, but . . .
  – Sensitivity of the human vision system to shifts in color and saturation require special attention

• Min-max stretch
  – Stretch the DNs in each band over their respective minmax range
  – Good news:
    • Easy to calculate and implement
    • No data lost by saturation
  – Bad news:
    • Sensitive to outlier DNs
    • Color balance can change unpredictably
Linearly Stretched Data

Three band CIR combination:
Band 1 (spectral green) displayed as blue
Band 2 (spectral red) displayed as green
Band 3 (spectral NIR) displayed as red

Original

Linear Stretch
Histogram Equalization Stretch

Three band combination:
Band 1 (spectral green) displayed as blue
Band 2 (spectral red) displayed as green
Band 3 (spectral NIR) displayed as red
Comparizon Between Histogram Equalization Methods

Linear Stretch

Histogram Equalization Stretch

INE6007(Spring 2010) ECE, UPRM 79
Color Normalization

• Normalization stretch
  – “Standardized” stretch
  – Good news:
    • Average color is grey
    • Contrast controlled by single parameter, the desired output standard deviation
  – Bad news:
    • Some data are lost in saturation
Color Decorrelation

- Decorrelation stretch
  - Enhance small spectral deviations in highly correlated spectral bands
  - Commonly used in geology
  - Good news:
    - Decorrelates bands
      - Emphasizes differences among bands
      - Can be applied to any number of bands
  - Bad news:
    - Produces highly saturated colors
Color Enhanced
TM Images of
Cuprite Mining
District in NV
RGB Color Coordinate System

RGB Color Coordinate System

Red (255, 0, 0)

Yellow (255, 255, 0)

White (255, 255, 255)

Gray Line

Magenta (255, 0, 255)

Black (0, 0, 0)

Green (0, 255, 0)

Cyan (0, 255, 255)

Blue (0, 0, 255)
Intensity Hue Saturation Color Coordinate System

Intensity, Hue, Saturation (IHS) Color Coordinate System

INEL6007(Spring 2010) 87ECE, UPRM
IHS Color Coordinate System

- The vertical axis represents **intensity (I)** which varies from black (0) to white (255).
- The circumference of the sphere represents **hue (H)**, which is the dominant wavelength of color.
  - 0 at the midpoint of red tones and increase counterclockwise around the circumference of the sphere to conclude with 255 adjacent to 0.
- **Saturation (S)** represents the **purity** of the color and ranges from 0 at the center of the color sphere to 255 at the circumference.
  - 0 represents a completely impure color in which all wavelengths are equally represented and which the eye will perceive as a shade of gray that ranges from white to black depending on intensity.
Transformation Equations

\[ I = R + G + B \]

\[ H = \frac{G - B}{I - 3B} \]

\[ S = \frac{I - 3B}{I} \]
Color Spaces

- HSI color coordinate system
- Hexcone model
  - similar to a cylindrical coordinate system, but based on RGB color cube
  - value = max(R,G,B) used instead of intensity
  - efficient CST from RGB to Hue-Saturation-Value (HSV)

*project subcube faces onto orthogonal plane intersection at the vertex of subcube*
Example Ramp Spectrum CSTs
Color-Space Transforms

- Color-space transforms
  - Human vision system perceives hue (H), saturation (S) and intensity (I), not RGB
  - Therefore, control over color appearance is best done in HSI space
CST for Contrast Enhancement

• Intensity stretch
  – Good news:
    • Improves contrast without changing hue or saturation
    • Based on human vision system model
  – Bad news:
    • Can be applied only to color (3-band) images
    • Based on human vision system model
Color Contrast
Enhancement
3-D Scatterplots
Watch Out
Watch Out

New vectors appear
(loss of inter-band correlation)

New colors appear
Note the Change in Colors due to Processing
Spatial Domain Blending

\[ \text{output} = (1 - \alpha) \times \text{image0} + \alpha \times \text{image1} \]

**TABLE 5-4.** Base image to be used to manipulate different color image properties with the interpolation/extrapolation blending algorithm. The magnitude of the effect for all properties is increased for \( \alpha \geq 1 \) and decreased for \( 0 \leq \alpha \leq 1 \).

<table>
<thead>
<tr>
<th>property to be modified</th>
<th>base image0</th>
</tr>
</thead>
<tbody>
<tr>
<td>intensity</td>
<td>black</td>
</tr>
<tr>
<td>contrast</td>
<td>grey</td>
</tr>
<tr>
<td>saturation</td>
<td>greyscale version of image1</td>
</tr>
</tbody>
</table>
Final Remarks

• Several linear and nonlinear spectral transforms were studied
  – Qualitative and quantitative analysis
  – Band ratios → Vegetation studies
  – PCA → Data compression, data dependent, no-uniform application
  – Tasseled-Cap → Sensor specific
  – Color-composites can be enhanced for visual analysis
    → not for quantitative analysis

• Local transforms are also possible
Review questions

• What is the purpose of contrast stretching
• How does contrast stretch applied to PC work?
• What is the difference between global and local transforms
• Explain linear stretch, nonlinear stretch, normalization and reference stretch
• Explain histogram equalization
• Explain thresholding and for what purpose it is used.
• What is the difference between stretching single band and color images
• What are color perceptual spaces.
• What is the best way to assign HSI for visual interpretation.
• Ex 5-6 and 5-7