Automated Classification of Hyperspectral Images using Spatial-Spectral Features

Authors:

Shivani Dhok
Ankit Bhurane
Ashwin Kothari

Department of Electronics and Communication Engineering
Indian Institute of Information Technology, Nagpur (IIITN)
Contents:

• Hyperspectral Images
• Applications of HSI
• Aim of this work
• Workflow
• Results
• Conclusions
• MATLAB functionalities used
• How MATLAB helped in completion of this work
Hyperspectral Images

- Human eyes see the visible spectrum.
- What if we can think beyond Visible range?
- **Hyperspectral imaging**, collects and processes information from across the electromagnetic spectrum.
- Contiguous spectral bands.
- Obtain Spectrum of each pixel.

---


[c] [https://media.wired.com/photos/5932433f44db296121d69f91/master/w_749,c_limit/hs-3.jpg](https://media.wired.com/photos/5932433f44db296121d69f91/master/w_749,c_limit/hs-3.jpg)

[d] [https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT48z3ezOqMhrn1taijeveY3Z9yCA6T2zGTalxgPGr6cR2pR5ST0](https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT48z3ezOqMhrn1taijeveY3Z9yCA6T2zGTalxgPGr6cR2pR5ST0)
Applications of HSI

- Mining and Geology: Identification of minerals and ores.
- Agriculture: Monitoring the development and health of crops.
- Eye care: Diagnosis of retinopathy and macular edema before damage of eyes.
- More applications for HSI are in the fields of: food processing, mineralogy, surveillance, astronomy, chemical imaging, environment....
Aim of this work:

• To develop an automated algorithm for classification of HSI.
• To increase the classification accuracy of the classification by extracting orthogonal features.

Salinas Data set [e]
Workflow of the method:

- Three publicly available datasets were considered: Salinas-A, Salinas and Botswana[9].

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Resolution</th>
<th>Number of Labelled Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Salinas-A</td>
<td>86 x 83 x 204</td>
<td>6</td>
</tr>
<tr>
<td>Salinas</td>
<td>512 x 217 x 204</td>
<td>16</td>
</tr>
<tr>
<td>Botswana</td>
<td>1476 x 256 x 145</td>
<td>14</td>
</tr>
</tbody>
</table>

## Workflow of the method:

### Feature Extraction stage:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Mathematical Equation</th>
<th>MATLAB Function used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear Predictive Coefficients</td>
<td>Coefficients of following predictor equation: [ I_p(n) = -A(2)I_p'(n-1) - A(3)I_p'(n-2) - \ldots - A(N+1)I_p'(n-N) ]</td>
<td>LPC=lpc(D,9);</td>
</tr>
<tr>
<td>Wavelet Coefficients [1-4]</td>
<td>Approximate and detailed coefficients obtained after successive decomposition of signal using LPF and HPF.</td>
<td>([c, l] = \text{wavedec}(D, 3, 'db2');) (\text{approx} = \text{appcoef}(c, 1, 'db2');) ([\text{cd1}, \text{cd2}, \text{cd3}] = \text{detcoef}(c, l, [1, 2, 3]))</td>
</tr>
</tbody>
</table>

![Diagram](image)
### Workflow of the method:

#### Feature Extraction stage:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Mathematical Equation</th>
<th>MATLAB Function used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average energy</td>
<td>[ E_{\text{avg}} = \frac{\left{ \sum_{n=1}^{N_b} \left( \frac{I_p(n)}{\max(I_p)} \right)^2 \right}}{N_b} ]</td>
<td>[ E = \text{sum}(D.^2)/\text{length}(D); ]</td>
</tr>
<tr>
<td>Fractal Dimension [5]</td>
<td>[ L(k) = \frac{1}{k} \left( \sum_{i=1}^{N_b} \left</td>
<td>I_p(m + ik) - I_p(m + (i - 1)k) \right</td>
</tr>
<tr>
<td></td>
<td>FD is the slope of the plot of ( \log(L(k)) ) against ( \log(k) ).</td>
<td>[ xl = \log10((1:64)); ]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[ yl = \log10(yy(1:64)); ]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[ p1 = \text{polyfit}(xl(index(indexl1:indexl2)), yl(index(indexl1:indexl2)), 1); ]</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[ fd = \text{abs}(p1(1)); ]</td>
</tr>
</tbody>
</table>
## Workflow of the method:

### Feature Extraction stage:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Mathematical Equation</th>
<th>MATLAB Function used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entropy[6]</td>
<td>$E = - \sum_{n=1}^{N_b} \left( \frac{I_p'(n)}{\sum_{n=1}^{N_b} I_p'(n)} \right) \log_2 \left( \frac{I_p'(n)}{\sum_{n=1}^{N_b} I_p'(n)} \right)$</td>
<td>Ent=entropy(D./sum(D));</td>
</tr>
<tr>
<td>Rényi Entropy[7]</td>
<td>$RE(I_p) = -\log \left( \sum_{n=1}^{N_b} \left</td>
<td>I_p(n) \right</td>
</tr>
<tr>
<td>Mean</td>
<td>$\mu = \frac{\sum_{n=1}^{N_b} \left( \frac{I_p'(n)}{\sum_{n=1}^{N_b} I_p'(n)} \right)}{N_b}$</td>
<td>Mean=mean(D);</td>
</tr>
</tbody>
</table>
## Workflow of the method:

**Feature Extraction stage:**

<table>
<thead>
<tr>
<th>Feature</th>
<th>Mathematical Equation</th>
<th>MATLAB Function used</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Deviation</td>
<td>( \sigma = \sqrt{\frac{\sum_{n=1}^{N_b} \left{ \mu - \left( \frac{I_p'(n)}{\sum_{n=1}^{N_b} I_p'(n)} \right) \right}^2}{N_b}} )</td>
<td>Std_dev=std(D);</td>
</tr>
<tr>
<td>Kraskov Entropy[8]</td>
<td>( \hat{H}(I_p) = -\psi(k) + \psi(N_{(p)}) + \log(c_d) + \frac{d}{N_b} \sum_{i=1}^{N_b} \log(\epsilon(i)) )</td>
<td>A user defined MATLAB function used in which following inbuilt functions are used:</td>
</tr>
</tbody>
</table>

where, \( \psi(x_i) = \Gamma(x) / \Gamma(x + 1) \), \( \psi(x) \) is the digamma function, \( c_d \) is the volume of \( d \)-dimensional unit ball, for euclidean norm, \( c_d = \frac{\pi^{d/2}}{\Gamma(1 + \frac{d}{2})} \), \( \epsilon(i) \) is twice the distance of \( x_i \) to its \( k \)th neighbor.
Workflow of the method:

Classification stage:

- Selecting the input and validation method
- Selecting the desired classification machine
- Selecting Classification Learner App
Workflow of the method:

Classification stage:

Classification results in confusion matrix
Workflow of the method:

Performance Evaluation stage:

The confusion matrix obtained is used for calculating performance parameters.
Results for Salinas-A Data set:

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>CLASS (SAMPLES)</th>
<th>ACCURACY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Broccoli_green_weeds_1</td>
<td>99.74</td>
</tr>
<tr>
<td>2</td>
<td>Corn_senesced_green_weeds</td>
<td>99.40</td>
</tr>
<tr>
<td>3</td>
<td>Lettuce_romaine_4wk</td>
<td>99.51</td>
</tr>
<tr>
<td>4</td>
<td>Lettuce_romaine_5wk</td>
<td>99.93</td>
</tr>
<tr>
<td>5</td>
<td>Lettuce_romaine_6wk</td>
<td>99.26</td>
</tr>
<tr>
<td>6</td>
<td>Lettuce_romaine_7wk</td>
<td>99.37</td>
</tr>
<tr>
<td></td>
<td>OVERALL ACCURACY</td>
<td>99.70</td>
</tr>
<tr>
<td></td>
<td>AVERAGE ACCURACY</td>
<td>99.54</td>
</tr>
<tr>
<td></td>
<td>PREDICTION SPEED (obs/sec)</td>
<td>32000.00</td>
</tr>
<tr>
<td></td>
<td>TRAINING TIME (sec)</td>
<td>5.87</td>
</tr>
</tbody>
</table>

Class-wise and overall performance for Salinas-A

<table>
<thead>
<tr>
<th>FEATURES</th>
<th>ACCURACY</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPC</td>
<td>96.90</td>
</tr>
<tr>
<td>φ+LPC</td>
<td>97.80</td>
</tr>
<tr>
<td>φ+LPC+σ</td>
<td>99.00</td>
</tr>
<tr>
<td>φ+LPC+σ+µ</td>
<td>99.00</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg</td>
<td>98.90</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg+FD</td>
<td>99.40</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg+FD+E</td>
<td>99.50</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg+FD+E+RE</td>
<td>99.60</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg+FD+E+RE+H</td>
<td>99.60</td>
</tr>
</tbody>
</table>

Cumulative performance of the Features

Pictorial representation of cumulative performance of the Features
Results for Salinas Data set:

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>CLASS</th>
<th>ACCURACY QUADRATIC SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Brocoli_green_weeds_1</td>
<td>99.61</td>
</tr>
<tr>
<td>2</td>
<td>Brocoli_green_weeds_2</td>
<td>99.84</td>
</tr>
<tr>
<td>3</td>
<td>Fallow</td>
<td>98.79</td>
</tr>
<tr>
<td>4</td>
<td>Fallow_rough_plow</td>
<td>99.35</td>
</tr>
<tr>
<td>5</td>
<td>Fallow_smooth</td>
<td>99.03</td>
</tr>
<tr>
<td>6</td>
<td>Stubble</td>
<td>99.92</td>
</tr>
<tr>
<td>7</td>
<td>Celery</td>
<td>99.86</td>
</tr>
<tr>
<td>8</td>
<td>Grapes_smooth</td>
<td>87.34</td>
</tr>
<tr>
<td>9</td>
<td>Soil_Vinyard_develop</td>
<td>99.82</td>
</tr>
<tr>
<td>10</td>
<td>Corn_senesced_green_weeds</td>
<td>94.99</td>
</tr>
<tr>
<td>11</td>
<td>Lettuce_romaine_4wk</td>
<td>98.31</td>
</tr>
<tr>
<td>12</td>
<td>Lettuce_romaine_5wk</td>
<td>99.95</td>
</tr>
<tr>
<td>13</td>
<td>Lettuce_romaine_6wk</td>
<td>99.24</td>
</tr>
<tr>
<td>14</td>
<td>Lettuce_romaine_7wk</td>
<td>97.94</td>
</tr>
<tr>
<td>15</td>
<td>Vinyard_untrained</td>
<td>67.31</td>
</tr>
<tr>
<td>16</td>
<td>Vinyard_vertical_trellis</td>
<td>99</td>
</tr>
<tr>
<td>OVERALL ACCURACY</td>
<td></td>
<td>92.4</td>
</tr>
<tr>
<td>AVERAGE ACCURACY</td>
<td></td>
<td>98.15769231</td>
</tr>
<tr>
<td>TRAINING TIME (sec)</td>
<td></td>
<td>8034.8</td>
</tr>
<tr>
<td>PREDICTION SPEED (obs/sec)</td>
<td></td>
<td>6200</td>
</tr>
</tbody>
</table>

Class-wise and overall performance for Salinas

Cumulative performance of the Features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPC</td>
<td>85.90</td>
</tr>
<tr>
<td>φ+LPC</td>
<td>88.80</td>
</tr>
<tr>
<td>φ+LPC+σ</td>
<td>90.20</td>
</tr>
<tr>
<td>φ+LPC+σ+μ</td>
<td>90.30</td>
</tr>
<tr>
<td>φ+LPC+σ+μ+Eavg</td>
<td>90.40</td>
</tr>
<tr>
<td>φ+LPC+σ+μ+Eavg+FD</td>
<td>91.00</td>
</tr>
<tr>
<td>φ+LPC+σ+μ+Eavg+FD+E</td>
<td>91.10</td>
</tr>
<tr>
<td>φ+LPC+σ+μ+Eavg+FD+E+RE</td>
<td>92.30</td>
</tr>
<tr>
<td>φ+LPC+σ+μ+Eavg+FD+E+RE+Ĥ</td>
<td>92.40</td>
</tr>
</tbody>
</table>

Pictorial representation of cumulative performance of the Features
### Results for Botswana Data Set:

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>CLASS</th>
<th>ACCURACY QUADRATIC SVM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Water</td>
<td>99.26</td>
</tr>
<tr>
<td>2</td>
<td>Hippo grass</td>
<td>88.12</td>
</tr>
<tr>
<td>3</td>
<td>Floodplain grasses 1</td>
<td>91.63</td>
</tr>
<tr>
<td>4</td>
<td>Floodplain grasses 2</td>
<td>90.70</td>
</tr>
<tr>
<td>5</td>
<td>Reeds 1</td>
<td>81.40</td>
</tr>
<tr>
<td>6</td>
<td>Riparian</td>
<td>75.46</td>
</tr>
<tr>
<td>7</td>
<td>Firescar 2</td>
<td>96.53</td>
</tr>
<tr>
<td>8</td>
<td>Island Interior</td>
<td>94.09</td>
</tr>
<tr>
<td>9</td>
<td>Acacia Woodlands</td>
<td>85.03</td>
</tr>
<tr>
<td>10</td>
<td>Acacia Shrublands</td>
<td>85.48</td>
</tr>
<tr>
<td>11</td>
<td>Acacia Grasslands</td>
<td>92.46</td>
</tr>
<tr>
<td>12</td>
<td>Short Mopane</td>
<td>92.27</td>
</tr>
<tr>
<td>13</td>
<td>Mixed Mopane</td>
<td>90.30</td>
</tr>
<tr>
<td>14</td>
<td>Exposed Soils</td>
<td>95.79</td>
</tr>
</tbody>
</table>

#### Overall Accuracy:

- **OVERALL ACCURACY**: 89.50
- **AVERAGE ACCURACY**: 89.89428571
- **TRAINING TIME (sec)**: 11.212
- **PREDICTION SPEED (obs/sec)**: 5000

### Class-wise and Overall Performance for Botswana

### Cumulative Performance of the Features

<table>
<thead>
<tr>
<th>Feature Description</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPC</td>
<td>48.60</td>
</tr>
<tr>
<td>φ+LPC</td>
<td>72.00</td>
</tr>
<tr>
<td>φ+LPC+σ</td>
<td>81.40</td>
</tr>
<tr>
<td>φ+LPC+σ+µ</td>
<td>81.30</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg</td>
<td>84.90</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg+FD</td>
<td>86.60</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg+FD+E</td>
<td>86.50</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg+FD+E+RE</td>
<td>89.30</td>
</tr>
<tr>
<td>φ+LPC+σ+µ+Eavg+FD+E+RE+Ĥ</td>
<td>89.50</td>
</tr>
</tbody>
</table>

### Pictorial Representation of Cumulative Performance of the Features
Conclusions:

• The proposed method can be used for classification of HSI.

• The method provides an outstanding accuracy in classification.

• The method, for the first time uses LPC as a feature which shows a great predominance in classification of accuracy.

• In this process, data preparation and feature extraction plays a vital role.

• The synergistic effect of the features is observed, which allows a significant increase in accuracy.

The Presented work has been accepted at the 6th International Conference on Signal Processing and Integrated Networks (SPIN 2019) (http://www.amity.edu/spin2019/)
MATLAB Functions used:

• Basic Math Tool Operations
  (Basic mathematical and statistical inbuilt functions like sum(), mean(), entropy(), log(), log10(), max(), std(), length(), abs(), lpc(), polyfit(), etc.)

• Signal Processing Tool Box
  (Wavelet decomposition: \[ c, l = \text{wavedec}(D(k,1:145),3,'db2'); \]
  \[ \text{approx} = \text{appcoef}(c, l,'db2'); \]
  \[ [cd1, cd2, cd3] = \text{detcoef}(c, l, [1,2,3]); \])

• Machine Learning Tool Box
  (Classification Learner)
How MATLAB helped in this work

• Most of the features used had inbuilt functions in MATLAB.

• The features which didn’t have MATLAB functions were easily computed using some other inbuilt functions.

• The classification learner in the Machine Learning toolbox have inbuilt previously modelled classification machines.

• Availability of modelled classification machines, reduced the time required for modelling. Only training time was required.

• MATLAB helped in reducing the computation time.

• It made implementation of the algorithm easy.
References:

Thank You!