Image analysis of hyperspectral data using mathematical morphology

Mauro DALLA MURA[1] and Mathieu FAUVEL[2]

[1]GIPSA-lab, Grenoble Institute of Technology, France mauro.dalla-mura@gipsa-lab.grenoble-inp.fr

[2]DYNAFOR INRA & Institut National Polytechnique de Toulouse, France mathieu.fauvel@ensat.fr

1 Introduction

The purpose of this tutorial is to get familiar with some techniques for the analysis of remote sensing hyperspectral images exploiting the spatial information of the image. When dealing with hyperspectral images with high spatial resolution, the spatial relations of the pixels in the scene are fundamental for the analysis. Classical techniques for hyperspectral images addressing different tasks (e.g., classification, object extraction and change detection) that consider only the spectral characteristics of the pixels are limited since this complementary information source is not properly exploited. Different approaches exist for including the spatial information in the analysis (e.g., post-processing based on spatial regularization, use of spatial features and segmentation). In this tutorial we will focus on the extraction and use of spatial features for hyperspectral image analysis. In particular considering features based on operators defined in the mathematical morphology framework.

At first, an overview of the use of the spatial information in multi- and hyperspectral image analysis is given. The tools needed for extracting spatial features will be presented in details. In particular, we will give a general recall of classical morphological filters and introduce their use for the analysis of remote sensing images (e.g., morphological and attribute filters).

Concerning the applications, particular attention will be given to the task of land cover classification since spatial features play a fundamental role here. The architecture of some spectral-spatial classification techniques proposed in the literature will be reviewed.

The tutorial is composed of three main parts:

- Part I, devoted to the definition of the tools that will be used later for the extraction of the spatial features in hyperspecral images
- Part II, in which we will investigate how these spatial features can be useful for image classification
- Part III, in which we will focus on how to integrate what seen so far in the classification process of a real hyperspectral image (leading to the so called spectral-spatial classification)

2 Set-up

1. Launch Matlab
2. Change the Matlab current directory to the folder with the material for this tutorial. You can do this either in the Matlab 's Current Folder window or in the Matlab 's Command Line by

```
>> cd write/the/path/to/folder
```

3. The material in the folder is organized as follows:
   - `code` directory containing the code you will use
   - `data` directory of the images
   - `lib` directory containing external code
4. Now you are ready to start with Part I!

3 Part I

Most of the operators defined in the mathematical morphology framework are neighborhood operations such as that the result of the processing g for the image f is a function of the image transformation Ψ employed and of the values of f on the neighborhood $\mathcal{N}$ of each pixel:

$$g(x) = \Psi_{\mathcal{N}_x}(f(x))$$

In other words, the result of the processing for the pixel x will be given by the transformation applied on the set of pixels included in its neighborhood $\mathcal{N}_x$. The neighborhood is defined by a mask (i.e., image) of a given shape, size and center. This mask is usually called structuring element (SE).
In the following you will get familiar in how to define and apply these tools in Matlab.

1. Open the file `tutorial_part1.m`, you will find the code for performing the operators that will be presented in the following. You will now test them on the grayscale image of the *cameraman*.

2. **Structuring element**. Type `doc strel` in the command line of Matlab and have a look to the syntax for defining a SE and the different shapes available.
   Define one or two structuring elements and see the mask that is produced. For example,

```
>> se1 = strel('square',11) % 11-by-11 square
```

   .

3. **Basic morphological operators**. Some basic operators are listed below. B is a structuring element and $\breve{B}$ is the reflected SE (i.e., rotation of B of 180° about its origin).

   - **Erosion** $\varepsilon_B(f)$
   - **Dilation** $\delta_B(f)$
   - **Opening** $\gamma_B(f) = \delta_{\breve{B}}[\varepsilon_B(f)]$
   - **Closing** $\phi(f) = \varepsilon_{\breve{B}}[\delta_B(f)]$

4. **Morphological Profile (MP)** Having to process complex scenes a single scale of the SE is typically not adapted to take into account the heterogeneity of the structures in the scene. For this reason, MPs are useful since they process an image in a multiscale fashion. An MP composed of L openings and L closings (i.e., leading to a profile composed of $2 \times L + 1$ images) can be defined as:

$$MP(f) = \{\phi_L(f), \ldots, \phi_\lambda(f), \ldots, \phi_1(f), f, \gamma_1(f), \ldots, \gamma_\lambda(f), \ldots, \gamma_L(f)\}$$

   being the subscript $\lambda = 1, \ldots, L$ denoting the size of the SE (whereas its shape is kept fixed for the whole profile). Thus an MP can be seen as the concatenation of a granulometry (sequence of openings with SEs of increasing size), $G_f^\gamma(\lambda)$, and an anti-granulometry (sequence of closings with SEs of increasing size) $G_f^\phi(\lambda)$, with same values of λs:

$$MP(f) = \{G_f^\phi(\lambda), f, G_f^\gamma(\lambda)\}.$$

   We recall that the absorption property is verified by the profile, such as the values of each pixel in the profile are monotonically decreasing. Thus, the derivative of the MP (denoted as DMP) approximated by the difference between two adjacent levels in the profile:

$$DMP(f) = \{MP_i(f) - MP_{i+1}(f)\}$$

   is composed of $2 \times L$ images of non negative values.

5. **Morphological Directional Profile (MDP)** When dealing with elongated and thin structures, linear structuring elements are the shapes of SEs to consider. Morphological Directional Profile are based on linear SEs of varying size and orientation. A MP is generated for each orientation θ and all the MPs are combined for generating the MDP as follows:

$$MDP(f) = \{\min_{\theta}(\{G_f^{\phi}(\lambda, \theta)\}), f, \max_{\theta}(\{G_f^{\gamma}(\lambda, \theta)\})\}$$

   with $\max_{\theta}(\{G_f^{\gamma}(\lambda, \theta)\}$ denoting the granulometry obtained taking the max of corresponding levels (i.e., equal λ) for the granulometries computed with different orientations θs.

6. **Attribute Profile (AP)** Analogously to the MP, the AP is composed of a sequence of attribute thickenings and attribute thinnings

$$AP(f) = \left\{ \underbrace{\phi^{P_{\lambda_L}}(f), \phi^{P_{\lambda_{L-1}}}(f), \ldots, \phi^{P_{\lambda_1}}(f)}_{\text{thickening profile}}, f, \underbrace{\gamma^{P_{\lambda_1}}(f), \ldots, \gamma^{P_{\lambda_{L-1}}}(f), \gamma^{P_{\lambda_L}}(f)}_{\text{thinning profile}} \right\},$$

   with $P_\lambda : \{P_{\lambda_i}\}$ $(i = 1, \ldots, L)$ a set of L ordered predicates (i.e., $P_{\lambda_i} \subseteq P_{\lambda_k}$, $i \leq k$) checking the value of an attribute $\mathcal{A}$ against the reference value λ_i.

7. **Marginal and reduced ordering of color images** When processing color (or hyperspectral) images, the operators seen so far cannot be directly applied, due to the lack of ordering in vectors. Thus, an ordering relation needs to be arbitrarily considered.

8. **Extended profiles (EMP, EAP)** Based on the reduced ordering, EMP (EAP) are defined as MP (AP) computed on the images obtained by a dimensionality reduction technique applied on an hyperspectral image.

   Here you will test the marginal and reduced ordering on a sample image.

   Q: Test the operators seen so far on a real remote sensing image

4 Part II

In this part some toy examples will be considered having as goal the classification of synthetic images using as features the results of the operators presented in the previous section. We will see how such features can improve the discriminability of classes with respect to the use of the spectral information only.

You will find the material for this part in the file `tutorial_part2.m`.

Since the extracted features will be considered as input to a classifier, we will briefly report the classification procedure as it can be found in the code.

1. Create training and test sets. Take 40 samples randomly chosen from the labeled set for performing the training. The rest will be used for the test. For creating the two sets run:

   ```
   >> [tr_idx, ts_idx] = gen_train_test(test, 40);
   ```

2. Perform the training of the SVM on the training set and run the classification on the whole data
   The SVM will have Radial Basis Function Kernel (Gaussian kernel).

   ```
   >> L = svm_class(data_vec, test, tr_idx);
   ```

   `L` will contain the predicted label of the data. You can see the classification map with the associated colormap by:

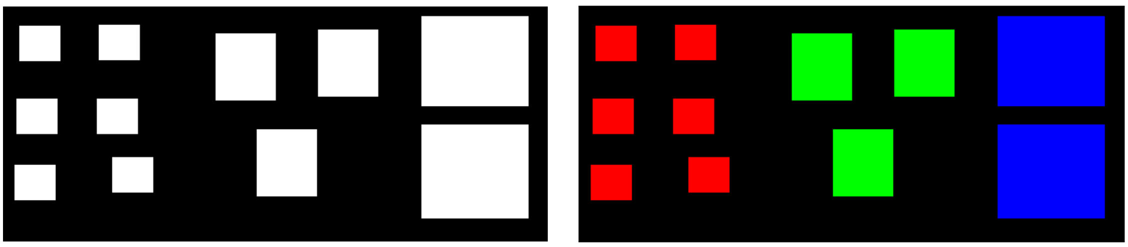
   ```
   >> figure, imshow(reshape(L, nr, nc), class_colormap)
   ```

3. Check the Overall Accuracy (ratio of the number of correctly classified patterns over the total number of labeled samples).

   ```
   >> acc = sum(L(ts_idx) == test(ts_idx)) / length(ts_idx);
   ```

4.1 Toy 1

In this example we suppose we want to perform a discrimination based on the scale of the regions (see Figure 1). Thus, the rectangular objects in the foreground are grouped in different thematic classes according to their scale. Since the synthetic image is binary, the spectral information cannot



(a) (b)

Figure 1: Synthetic dataset used for the toy example 1. (a) Binary image composed of three groups of rectangles having different sizes; (b) Image showing the labeling of the pixels (4 classes).

help in discriminating among the different classes. Here, the size is the only discriminant feature.

In this example you will consider the following spatial features:

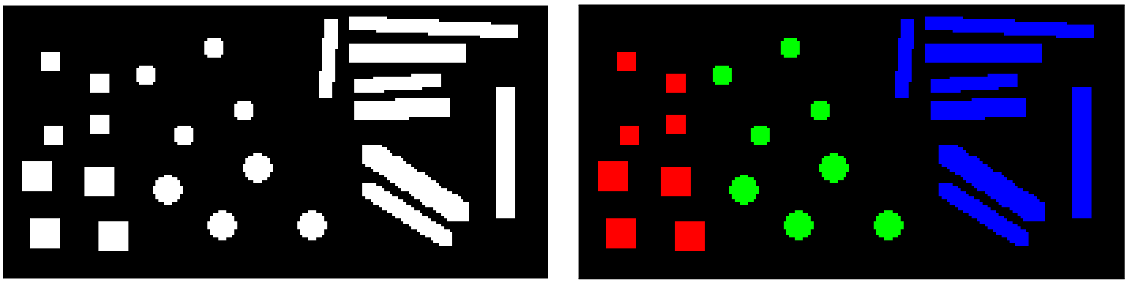
- morphological profiles without reconstruction
- morphological profiles with reconstruction
- attribute profiles with area attribute

Note the importance in using connected filters in order to properly retrieve the shapes of the regions.

Q: **[Optional]** Use as features the results of a multiscale analysis done with a linear filter (e.g., an average filter with increasing size of the window)

4.2 Toy 2

This toy example wants to show the relevance of spatial features related to the shape of the regions. Here the objects in the synthetic image are assigned to different classes according to their shape. You will test the use of MPs and APs using different SEs and attributes.



(a) (b)

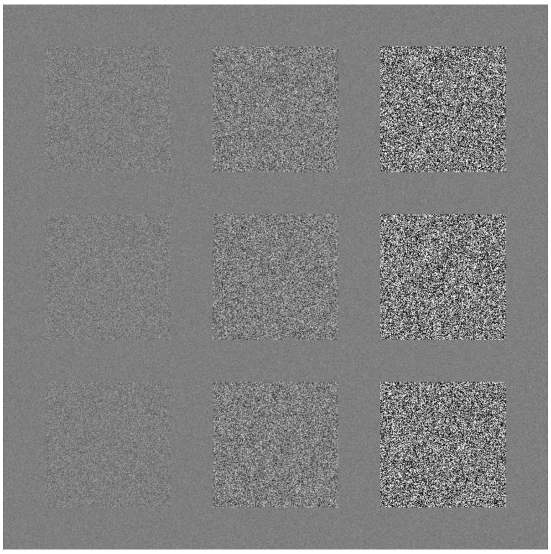
Figure 2: Synthetic dataset used for the toy example 2. (a) Binary image composed of three groups of foreground regions having different shapes (and different scales); (b) Image showing the labeling of the pixels (4 classes).

Q: **[Optional]** Give a try to the Morphological Directional Profile. Typically the features that can be obtained with a MDP can be useful for discriminating between elongated and compact elements. However they require the computation of several filters (in order to consider the multiple possible sizes and orientation of the structuring elements).

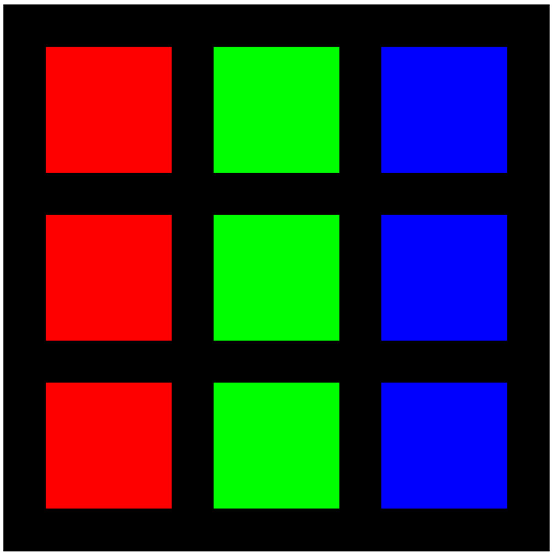
4.3 Toy 3

This task is devoted to the classification of regions having different spectral homogeneity. The graylevels are drawn by four Gaussian distributions of same mean and different standard deviations. Image classification will be performed by considering features extracted from APs with different attributes. Note how by using the standard deviation as attribute leads to the most discriminant features.

Q: Try to use an MP for perform the classification.



(a)



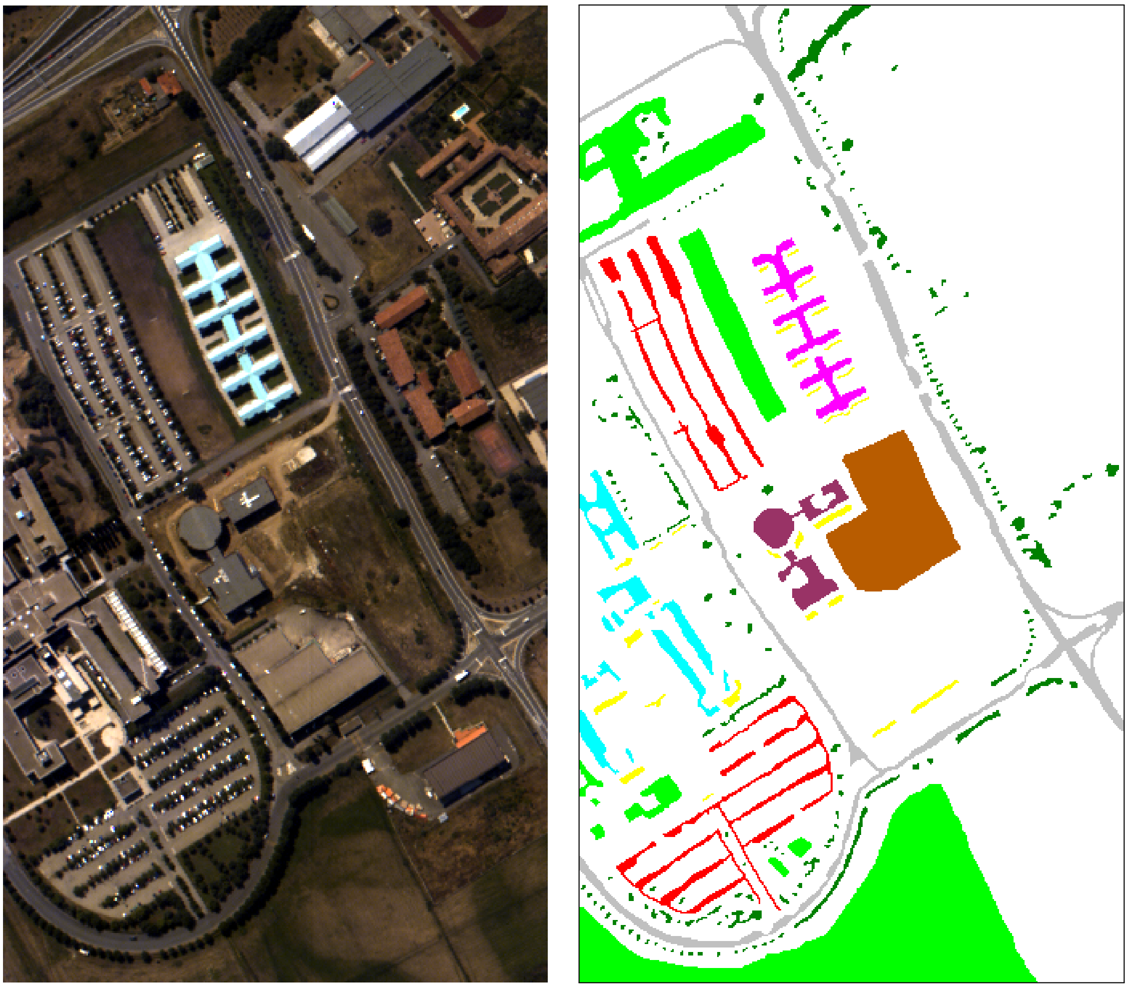
(b)

Figure 3: Synthetic dataset used for the toy example 3. (a) Image composed of nine squared regions with values drawn by a Gaussian distribution $\mathcal{N}(125, 5^i)$ being $i = 1, \ldots, 4$ a number identifying the class; (b) Image showing the labeling of the pixels (4 classes).

5 Part III

5.1 Data

The hyperspectral image of Pavia University will be used in this lab section (Figure 4) and was kindly provided by Prof. Paolo Gamba from the University of Pavia, Italy. This image have been widely used for hyperspectral image classification. The image was acquired by the hyperspectral airborne



(a) (b)

Figure 4: Pavia University data set. (a) True color representation. (b) Test image.

sensor ROSIS-03 (Reflective Optics Systems Imaging Spectrometer). The ROSIS sensor acquires 115 spectral bands, ranging from 0.43 μm to 0.86 μm with a geometrical resolution of 1.3 m. The image was atmospherically corrected but not geometrically corrected (as it is possible to notice the geometrical distortions due to the displacement of the airborne platform during the acquisition). The image acquired over the University area is composed of 103 bands (12 bands were removed due to noise) of 610 $\times$ 340 pixels (see Figure 4(a) for a RGB representation). Nine thematic classes were identified in this scene: trees, asphalt, bitumen, gravel, metal sheets, shadows, self-blocking bricks, meadows and bare soil. The test image can be seen in Figure 4(b).

5.2 Outline

The section is divided into different parts creating a possible processing chain in a real case scenario:

- visualization and exploratory analysis
- dimensionality reduction

- spectral classification
- computation and exploitation of spatial features in classification

5.3 Visualization and exploratory analysis

This part of the session is devoted to the visualization and inspection of the hyperspectral image of Pavia University.

1. Get familiar with the visualization tools in MATLAB

   - Look at one band of the image with the function `imshow` (e.g., if you want to see the 15th band you should type:

     ```
     >> imshow(data(:,:,15),[])
     ```

     You can also use the function `imshow_stretch`, which will automatically stretch the image in order to adjust the mapping of the graylevels to the image dynamic.

   - Show a *true color composition* (R,G,B) of the hyperspectral image by typing:

     ```
     >> imshow_stretch(data(:,:,[bR, bG, bB]))
     ```

     With $b_R = 36$, $b_G = 28$, $b_B = 1$.

   - Show a *false color composition* (NIR,R,G) of the hyperspectral image. With $b_R = 80$, $b_G = 36$, $b_B = 28$

     Q: Compare the true and false color compositions of the hyperspectral image. Are they equally informative?

     Q: From the color representations perform an interpretation the image. Which thematic classes can you determine? Which different land cover classes can be determined thanks to the spectral and which thanks to the spatial resolution of the image?

2. **Look at the bands**

   - Type for example:

     ```
     >>  for i=1:191
     imshow_stretch(data(:,:,i))
     pause(0.1)
     end
     ```

     Q: What can you state? Are the bands significantly different one to the other? Are they "equally informative"?

     Q: Do you think that filters applied to adjacent bands will be significantly different?

3. **Extract some spectral signatures**

   - Show a single band or a RGB composite of the image. Move the cursor on the image, on the bottom right corner of the window you will see the position of the cursor in image coordinates thus the top left corner is $(1, 1)$ and the bottom right corner is $(ncols, nrows)$. Note that the coordinates are listed as (col, row) in the window.

- Note down the image coordinates of some pixels belonging to different land cover types.

  Q: Plot the spectral signature of the selected pixels.

  For example you can plot the values as follows:

```
>> plot(squeeze(data(row1, col1, :)), 'b')
hold on
plot(squeeze(data(row2, col2, :)), 'r')
plot(squeeze(data(row3, col3, :)), 'g')
...
```

  Note that the indexing of the image in MATLAB is done with `(row, col, band)`.

  Q: Which bands/ranges of bands are more important for discrimination?

4. **Check correlation between bands**

   - Are the bands correlated? Have a look to the correlation matrix of the data, which shows the cross-correlation between different bands.

```
>> C = corr(data_vec(1:100:end,:));
imagesc(C)
colorbar
```

     Q: What can you notice?

5.4 Dimensionality reduction

1. Three dimensionality reduction techniques are provided:
   - Principal Component Analysis, PCA (linear unsupervised transformation)
   - Linear Discriminant Analysis, LDA (linear supervised transformation)
   - Kernel PCA, KPCA (non-linear unsupervised transformation)
2. Compute the PCA and the LDA on the original data
3. Look at the eigenvalues in order to find the optimal number of components to retain. Do the components correspondent to the smallest eigenvalues show only noise?
4. Have a look to the retained components

Q: [**Optional**] Do the same analysis for KPCA.

5.5 Spectral classification

This part is devoted to perform a classification of the data using only the spectral information.

Q: Perform a classification considering only i) a single band (arbitrarily chosen), ii) the three bands used for the true color composition, iii) the three bands chosen for the false color composition and iv) the features obtained by a dimensionality reduction technique (e.g., PCA).

5.6 Spectral-Spatial classification

1. Consider the first principal components and compute:
   - Extended Morphological Profile without geodesic reconstruction
   - Extended Morphological Profile with geodesic reconstruction
   - Extended Attribute Profile with area as attribute
   - Extended Attribute Profile with moment of inertia as attribute
   - Extended Attribute Profile with standard deviation as attribute
   - Extended Multi-Attribute Profile (EMAP) obtained by concatenating EAPs built with different attributes.
2. Look at the single images in each profile. Note how the filters produce a progressive simplification of the image and how the spatial characteristics of the data (e.g., shape of objects' boundary) are not distorted since the morphological attribute filters are connected operators (as would happen if applying for instance a low pass filter).
3. Plot the values of the profiles for some selected pixels (as previously done with the spectra).
4. Perform a classification considering each profile separately. Compare the obtained maps and accuracy values against those obtained only with the spectral features.
5. Try to concatenate different features (e.g., different profiles, spatial+spectral features, etc) and classify. The concatenation of two sets of features `D1` and `D2` can be done as

```
>> D = [D1, D2];
```

Q: Do the same with the features extracted by LDA.

5.7 Optional tests

Q:

1. Perform some of the tests already done considering a different number of training samples (change the second input parameter of `gen_train_test`).
2. Apply a dimensionality reduction to the profile prior to classification.
3. Compute the profiles on the features extracted by the KPCA and perform the classification.
4. Modify the value of cumulative variance required dimensionality after the dimensionality reduction techniques (it determines how many components will be retained).