ROYAL MILITARY ACADEMY
Communication, Information, Systems and Sensors dept.

6th October 2022
173 POL

Remote Sensing Systems

Projects

Contents

1 Introduction

Eleven different projects subjects are proposed. Each of these subjects should be addressed by a single student. Guidelines for the execution of the projects are given below. These guidelines are however not exclusive and other interesting paths may be followed by the students, provided given prior advice.

Each student is supposed to hand in a written report. The reports should demonstrate a *careful* analysis of the problem and provide a description and an explanation of the observed phenomenons. There will be a debriefing of the report. The form of the report will be evaluated according to the standard evaluation grid.

2 Projects

2.1 Hyperspectral unmixing

The aim of this project is to perform the detection of selected objects in an hyperspectral image. The spectral signature of the object to detect is extracted from the hyperspectral image itself.

- Select a feature to detect (by looking at one of the bands) and extract the corresponding spectrum.
- Compute the covariance matrix $\hat{\mathbf{R}}$ of the spectra and the resulting filter $\mathbf{w}$.
- Finally, compute the abundance and check its plausability using the ground truth.

A database of spectra is available at `http://speclib.jpl.nasa.gov/`. The data is available from
`http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes`.

2.2 Registration

The aim of this project is to register an image taken by the Ikonos spacecraft on the corresponding GoogleEarth image. The result is to be provided in the form of a KML (Keyhole Markup Language) file. Given the particularities of the KML, it is necessary to resample the image in order to perform a correct alignment with the underlying Google Earth image. For the same reason, the lines of the resampled image should be orthogonal to the meridians. Hence, the following steps should be addressed

- Identify ground control points (GCP) in the image and in GoogleEarth. Produce a KML file in which each GCP is indicated. Use that KML file to verify that the GCP indeed correspond to the desired feature.
- Compute the required transformation. Discuss the order of the transformation (order 1, order 2) and the associated residuals.
- Resample the image according to the transformation computed in the previous point. Check the result by inserting the image in a KML file (do *not* use the rotation functionality of the KML to "overlay" the image in GoogleEarth).

The image to be used is that of the Esfahan nuclear site in Iran. An example KML file is also provided showing an imperfectly registered image.

2.3 Co-registration

The aim of this project is to co-register two images. taken by the Ikonos spacecraft. One of the image is to be considered as the reference image on which the other image has to be registered. Hence, the following steps should be addressed

- Identify ground control points (GCP) in both images. Overlay the GCP on the images to verify that they indeed match the desired feature.
- Compute the required transformation. Discuss the order of the transformation (order 1, order 2) and the associated residuals.
- Resample the image according to the transformation computed in the previous point. Check the result by overlaying the images on top of each other.

The images to be used are those of the Esfahan nuclear site in Iran.

2.4 Orthorectification

The aim of this project is to orthorectify an image taken by an airplane. A digital elevation model (DEM) is provided. As the spatial resolution of the DEM is very coarse, it will have to be (bilinearly-) interpolated. Similarly, the interpolation in the image is to be performed bilinearly.
The DEM (object) coordinates are provided in UTM-Lambert projection (northing-easting). Hence, the following steps should be addressed:

- Identify the various quantities used (focal distance, altitude), locate the center of the image and the correspoding location of the scene center.
- Consider an object-image that is larger (in surface) than the image. For each pixel of the object-image, compute the location of the corresponding point in the image, taking into account the DEM. Interpolate the image in order to obtain the intensity value of the desired point.

The image to be used is an image of the hilly Virton area. The DEM is provided in RMA format (use the readrma.m routine to read that image). Technical information is provided in the file `virton.txt`.

2.5 Classification and change detection

The goal of this project is to try to classify different zones in an area-of-interest (vegetation, buildings, water) based indicators that you can construct from the different spectral bands in the Sentinel-2 radiometric images. These indicators are:

- Normalized Difference Vegetation Index (NDVI)
- Normalized Difference Built-up Index (NDBI)
- Normalized Difference Water Index (NDWI)

Choose in Terrascope an area of an appropriate size around coordinates (WGS84)[1]:

- latitude = 50.379123,
- longitude = 4.392829

Perform the following actions:

1. Find all the Sentinel-2 images within the period 2017-2019 with no cloud cover[2].
2. Compute the NDVI, NDBI and NDWI for these images.
3. For each image create 3 masks (a mask for each class: vegetation, built-up, water) based on suitable thresholding of NDVI, NDBI and NDWI bands.
4. Compare the classes of each subsequent pair of images to detect changes (difference between masks).
5. Return the dates of any changes and the classes concerned.

[1] You can transform coordinates to other reference systems with the `pyproj` package

[2] For every S2 product in the `terrascope` catalogue, you can find the proportion of cloud cover with `product.properties["productInformation"]["cloudCover"]`

2.6 Despeckling

Speckle is a noise-like phenomenon that arises in coherent imaging systems like SAR. One way to reduce its effect is by using an adaptive filter of the form:

$$\hat{\sigma}\bar{I} + k\,(I - \bar{I})$$

with I the intensity and $\bar{I}$ the averaged intensity. The Lee filter is an adaptive filter that chooses the parameter k as follows:

$$k = \frac{\bar{V}_I - \frac{1}{L}}{\bar{V}_I}$$

with $\bar{V}_I$ the normalized variance. This project studies the impact of the Lee filter:

- Choose an area-of-interest (see below) and select in Terrascope the first SAR image of 2021 that contains the AOI.
- Apply a Lee speckle filter to this image. Comment on the results and interpret the value of k.
- Apply a multi-temporal Lee filter to a sequence of images. This means that you estimate the average intensity for a pixel over multiple images in stead of using a spatial average.
- Compare the equivalent number of looks (ENL) as the number of images increases.

Do this for two area's of of interest with an appropriate size (WGS84 latitude/longitude):

1. one around $(50.442956, 4.509742)$, and
2. another around $(50.610151, 4.70438)$

2.7 GLCM Feature Extraction

Gray Level Co-occurrence Matrix (GLCM) is a second-order statistical analysis method to detect texture in images. Given an image composed of pixels, each with an intensity (a specific gray level), the GLCM is a tabulation of how often different combinations of gray levels co-occur in an image or image section. Texture feature calculations use the contents of the GLCM to give a measure of the variation in intensity (a.k.a. image texture) at the pixel of interest. The method was proposed by Haralick et al. in the article *Textural Features for Image Classification* that can be found on `BelADL`. The goal of the project is to find the combination of GLCM texture features that maximizes the inter-class distance between water, vegetation and built up area. In order to do this, the project asks you to do the following:

1. Consider a region of appropriate size around the "lac de la Gileppe" (between $(5.9634, 50.5714)$ and $(6.0129, 50.5890)$)
2. Find one S1 and one S2 image for that area that were acquired on the same day (or at least very close in time)
3. Select a number of patches (of size 15×15 pixels) that belong to one of the 3 classes (water, vegetation, built up).
4. Compute the GLCM texture features for each patch: mean, variance, correlation, energy, angular second moment, contrast, dissimilarity, homogeneity.

5. Select the combination of features that maximizes the inter-class distance, i.e. that allow you to optimally detect to which class a certain patch belongs.

Try this for both S1 and S2 images. Validate your results on another area-of-interest.

2.8 Generation of a Road Vector Layer

To goal of this project is create a routine to automatically generate a GIS vector layer of roads, based on Sentinel-1 and Sentinel-2 data.

1. Consider an area-of-interest between $(4.4480, 50.7946)$ and $(4.4840, 50.8200)$. Retrieve both S1 and S2 images from this area that are close together in time. The S2 image should contain as few clouds as possible.

2. Extract the pixels that most likely belong to the class ”road” to create a ground truth.

3. Exploit a number of detection algorithms from the `scikit-image` library [3] to automatically detect the roads in the image. Clean up your results, for example with skeletonization. Try this both for the S1 and S2 image and discuss the results.

4. Generate a vector layer containing the detected roads.

5. Validate your algorithm on another area-of-interest.

2.9 Flooding detection

The region around the Walloon village of Pepinster was hit hit by an horrendous flood this summer. This project tries to assess the extent of the flooded areas based on the Sentinel-1 images from before and after the flooding.

1. Find one S1 image just right before 15/07/2021 and one just right after that date (same relative orbit) of the AOI (from $(5.7490, 50.5560)$ to $(5.8120, 50.5770)$).

2. Create a ground truth by visual inspection of the pre-event image (use Google maps or Sentinel-2 images to guide you).

3. Extract the pixels that most likely belong to the class ”water” (ex. thresholding/classification).

4. Estimate the extent of the flooded areas in m^2 and assess the accuracy of the results using the ground truth (e.g. by computing the confusion matrix)

2.10 Ship Counting

The COVID-19 crisis had a large impact on international trading and supply chains. The goal of this project is to estimate how many ships enter the port of Antwerp on a single day and how this figure has evolved in time. Before ships are guided into the harbor (Antwerp or Rotterdam), they typically lie waiting in a holding patterns in front of the Westerschelde.

- Identify a large region in the North Sea where the ships are waiting before entering the harbor.

[3] available in `terrascope`

- Use Sentinel-1 SAR images and image processing methods to count the ships in an image.
- Plot the number of ships as function of time. Identify any seasonal patterns.
- Assess whether the COVID crisis had any impact on the number of ships.

2.11 Movements of military assets

The situation in Ukraine is well known. The goal of this project is to determine wether it is possible/feasible to detect movement of military assets either during the initial phase of the operation (before / after 24/02/2022) or at later stages. While on the principle a large area could be analyzed, it might be more efficient (both in computing time and in terms of false detections) to consider smaller areas.

- Identify a region close to the Russian-Ukraine southern border, possibly with the help of the New York Times maps https://www.nytimes.com/interactive/2022/world/europe/ukraine-maps.html
- Use Sentinel-1 SAR images and image processing methods to determine the changes between images.
- Plot the (supposed) number of military assets as function of time to confirm force build-up and subsequent deployment. You may want to use the area occupied by the assets as a proxy for the number of assets.