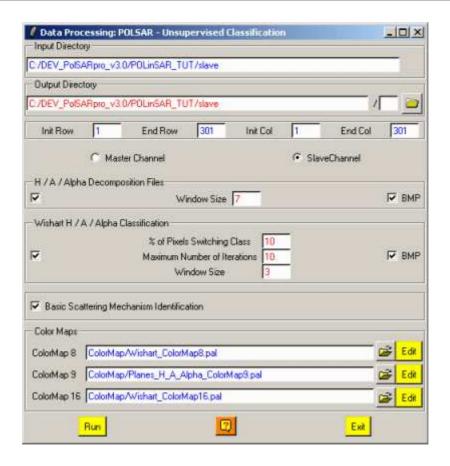# POLSAR Classification

Description:

This function is used to apply sequentially different polarimetric data processes. The different steps of this batch process are:

- H / A / Alpha Decomposition and analysis
- Unsupervised Wishart - H / A / Alpha Classification
- Basic Scattering Mechanisms identification

This basic processing approach provides a first **qualitative analysis** of the fully polarimetric data set processed.

Comments:

Parameters written in Red can be modified directly by the user from the keyboard.

Input/Output Arguments:

Input Directory
Indicates the location of the considered **Main Directory (MD)** containing the polarimetric data sets to be processed.

Output Directory
Indicates the location of the data output directory.

Output Image Number of Rows/Columns:

The output image numbers of rows and columns are initialised to the input data set dimensions.

Users wishing to process a sub-part of the initial image can modify the **Init** and **End** values of the converted images rows and columns.

Note: init and end values have to remain within the range defined by the input image dimensions.

H/A/Alpha Decomposition:

This program creates binary files corresponding to the different polarimetric descriptors obtained from the H/A/Alpha decomposition of a **[T3]** coherency matrix image.

The H/A/Alpha polarimetric decomposition is based on an eigenvector decomposition of the corresponding (3x3) coherency **[T3]** sub-matrix.

An option may be set to simultaneously create the corresponding bitmap image files.

Output Files
The polarimetric decomposition procedure creates the following output binary files.

- entropy.bin, anisotropy.bin, alpha.bin
- alpha1.bin, alpha2.bin, beta1.bin, beta2.bin, p1.bin, p2.bin

The last binary files are necessary for the basic scattering mechanism identification procedure

Processing Parameters
Data to be decomposed may be processed through an additional filtering procedure consisting of a boxcar filter. Users have then to set the size of the (N*N) sliding window used to compute the local estimate of the average matrix.

The default value of N is set to **7**. Users wishing to avoid additional filtering may set N to **1**.

Unsupervised Wishart - H/A/Alpha Classification:

This program creates binary and bitmap image files resulting from the segmentation of polarimetric data using the Wishart H-Alpha and Wishart H-A-Alpha schemes.

Output Files
Each classification procedure creates output binary files and the corresponding optional bitmap image files.

- wishart_H_alpha_class_x.bin (.bmp)
- wishart_H_A_alpha_class_x.bin (.bmp)

The variable x indicates the window size of the eventual additional filtering performed prior to data classification.

Processing Parameters
Data to be decomposed may be processed through an additional filtering procedure consisting of a boxcar filter. Users have then to

set the size of the (N*N) sliding window used to compute the local estimate of the average matrix.

The default value of N is set to **3**. Users wishing to avoid additional filtering may set N to **1**.

The segmentation termination criterion consists of a logical combination of two conditions.

The iterative k-mean clustering procedure is stopped if :

- A sufficiently low percentage of pixels switch class from one iteration to the other.
- The number of iterations reaches a maximum value

Numerical values are automatically set to default values and may be modified.

Basic Scattering Mechanisms Identification:

This program creates binary and bitmap image files resulting from a basic scattering mechanism identification from the segmentation of polarimetric data using the Wishart H-Alpha and Wishart H-A-Alpha schemes..

Output Files   The basic scattering mechanism procedure creates output binary files and the corresponding bitmap image files.

- sgl_class.bin, dbl_class.bin, vol_class.bin, id_class.bin (bmp) where **sgl** stands for single bounce scattering mechanism, **dbl** for double bounce scattering mechanism and **vol** for volume scattering mechanism.

ColorMaps:

The colour coding of the bitmap output files is realized by the way of a 8, 9 or 16 element colormap initialised with arbitrary values. Users have the possibility to modify the elements of the colormap in an interactive way.

There exists also the possibility to create an automatic **Coded Colormap** obtained from an RGB colour coding of each Class Feature Vectors.