

# User Guide

*Version:* 1.2  
*Date:* 17.12.2006  
  
*Status:* Valid  
*Author:* A. Hueni  
*File:* \UserGuide\_V1.2.doc  
*Pages:* 53

*Classification:*  
*Distribution:*

## History

Version	Date	Author	Remark
0.1	03.04.2006	A. Hueni	First draft
0.11	06.04.2006	A. Hueni	Modified MySQL installation
0.2	16.06.2006	A. Hueni	Modified Tutorial for new example set
1.1	20.10.2006	A. Hueni	Added description of new SpectraProc Features. Added description of new SpectraProc Query interface. Added new section in the Tutorial about disabling of outliers. Added change history of added features, fixed bugs and known bugs.
	16.11.2006	A. Hueni	Added another table change statement for the spectrum table (see Upgrading to SpectralDB V1.1)
1.2	17.12.2006	A. Hueni	Added description how to change the database connection string for SpectraProc Query

## Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>5</b>
<b>2</b>	<b>Installation and Configuration .....</b>	<b>6</b>
2.1	Database .....	6
2.1.1	Installing MySQL .....	6
2.1.2	Installing MySQL-Administrator and Query Browser .....	6
2.1.3	Setting up the Spectral Database Schema, Tables and Users .....	6
2.1.4	Network Access .....	6
2.2	SpectraProc Application .....	7
2.3	SpectraProc Query Interface .....	8
2.3.1	Installing TCL/TK .....	8
2.3.2	Installing Mysqld Package .....	8
2.3.3	Installing SpectraProc Query Interface .....	8
2.3.4	Running SpectraProc Query Interface .....	8
<b>3</b>	<b>Upgrading the Database .....</b>	<b>9</b>
3.1	Upgrading from MySQL version 4 to 5 .....	9
3.2	Upgrading SpectraProc DB V1.0 to V1.1 .....	9
<b>4</b>	<b>Design of Sampling Experiments .....</b>	<b>10</b>
4.1	Overview .....	10
4.2	Hierarchical Structure .....	10
4.3	Directory Structure and Spectral Files .....	10
<b>5</b>	<b>Operation .....</b>	<b>12</b>
5.1	Overview .....	12
5.1.1	Dataflow .....	12
5.1.2	File System Interfaces .....	12
5.1.3	SpectraProc Operation .....	13
5.1.4	Spectral Processing Concept .....	13
5.2	SpectraProc .....	14
5.2.1	User Interface .....	14
5.2.2	Creating a new Study .....	15
5.2.3	Deleting an existing Study .....	16
5.2.4	Study Report .....	16
5.2.5	Selecting Studies .....	16
5.2.6	Loading Spectra .....	17
5.2.7	Processing Chain Settings .....	17
5.2.7.1	Waveband Filtering .....	17
5.2.7.2	Smoothing .....	18
5.2.7.3	Sensor Synthesizing / Downsampling .....	20
5.2.7.3.1	4.5.3.1 Ratio Sensors .....	20
5.2.7.3.2	Gaussian Sensors .....	20
5.2.7.3.3	Supplied Sensors .....	22
5.2.7.4	Derivative Calculation .....	22
5.2.7.5	Feature Space Transformation .....	23
5.2.7.5.1	DI (DGVI) .....	23
5.2.7.5.2	NTBI .....	23

5.2.7.5.3	PCT .....	24
5.2.7.5.4	Creation, Modification and Deletion of Feature Space Definitions.....	24
5.2.7.5.5	Feature Space Selection .....	25
5.2.8	File Export.....	26
5.2.9	Saving the Contents of the Report Window.....	28
5.2.10	Spectral Mixtures .....	28
5.2.10.1	Endmember Designation.....	28
5.2.10.2	Abundance Settings .....	29
5.2.11	Library Building .....	29
5.2.12	Analysis.....	29
5.2.12.1	Separability.....	29
5.2.12.2	Discriminant Analysis.....	30
5.2.12.3	Principal Component Analysis .....	31
5.2.12.4	Unmixing.....	32
5.2.13	Adding new Sensors.....	33
5.2.14	Linking to a GIS System .....	35
5.2.14.1	Setting up the Spectral DB as a Data Source.....	35
5.2.14.2	Establishing a Database Connection in ArcCatalog.....	35
5.2.14.3	Adding Table Data to a Map .....	37
5.3	SpectraProc Query Interface .....	39
5.3.1	Introduction .....	39
5.3.2	Database Connection .....	39
5.3.3	User Interface .....	39
5.3.4	Report Windows .....	40
5.3.4.1	Campaign Report .....	40
5.3.4.2	Species Report .....	41
5.3.4.3	Site Report .....	41
5.3.4.4	Spectrum Report .....	42
<b>6</b>	<b>Tutorial.....</b>	<b>43</b>
6.1	Overview.....	43
6.2	Examine the Folder and File Structure .....	43
6.3	Creating a new Study and Loading the Spectra .....	43
6.4	Get to Know Your Data.....	43
6.5	Discriminant Analysis in PCT Feature Space.....	47
6.6	Disabling Outliers.....	48
<b>7</b>	<b>Change History.....</b>	<b>51</b>
7.1	SpectraProc .....	51
7.1.1	Added Features .....	51
7.1.2	Fixed Bugs.....	51
7.1.3	Known Bugs.....	52
7.2	SpectraProc Database.....	52
7.2.1	Added Features .....	52
<b>8</b>	<b>References.....</b>	<b>53</b>

## 1 Introduction

Field spectroradiometry has become increasingly popular in the last few years. The technology has advantages over conventional techniques, allowing the non destructive sampling of objects and possibly enabling the user to gain critical information more quickly and cheaply. As a result, many scientists are now actively researching applications of hyperspectral sensing. The operation of the instruments tends to be relatively easy and data are collected quickly. However, the interpretation of these data is not so simple. The main issue when dealing with hyperspectral data is their dimensionality which is the result of sampling a wide spectral range in very narrow bands. This is in itself a problem because the influence of noise on narrow channels is much higher than on traditional broadband channels. Hyperspectral data are more complex than previous multispectral data and different approaches for data handling and information extraction are needed (Vane and Goetz, 1988; Landgrebe, 1997).

Hyperspectral data are essentially multivariate data, consisting of hundreds or even thousands of variables. It has been shown that more bands do not automatically imply better results. Although the separability of classes does increase with growing dimensionality, the classification accuracy does not follow this trend endlessly but will decrease at a certain point. This is called the Hughes Phenomenon and is caused by the ever increasing number of samples needed to build sound statistics if the dimensionality grows (Landgrebe, 1997). In practice this means that more samples must be collected to ensure successful statistical analyses. It is necessary, therefore, to collect a large number of spectral data files, each containing a hyperspectral spectrum. The sheer number of files and variables can become overwhelming. Interestingly, very few studies concerned with hyperspectral data have ever mentioned how the data had been organised and stored.

A further issue that is rarely addressed is the reusability of the data. Reference data is usually compiled in so called spectral libraries. The majority of the publicly available spectral libraries are distributed as physical files. This has drawbacks such as low flexibility and low query performance (Bojinski et al., 2003). Another drawback of most libraries is their restriction in the number of spectra per class. In many cases, only one reference spectrum is supplied. This reduces any statistical analysis to first order statistics. The use of average values may be useful in some circumstances, however, Landgrebe (1997) noted that the reduction of data to mean values results in a loss of information. Second order statistics contain vital information about the distribution of data in spectral or feature space and should therefore be included in spectral data collections.

The time and effort that is spent in collecting spectral data, combined with the characteristically large number of files, makes it clear that spectral data should be well organised. Otherwise valuable data can be lost or loses its value because of missing metadata. Considering the above, it seems logical to employ a database to store spectral data in a suitable form. A further characteristic of hyperspectral data is the data redundancy. It has been shown that neighbouring wavebands have a high degree of correlation (Thenkabail et al., 2004). This redundancy is created by oversampling, i.e. the spectral signal is sampled at small enough steps to describe very narrow features that could be discriminating (Shaw and Manolakis, 2002). The redundancy and general noisiness of the data usually mean that certain pre-processing must be carried out before any useful analysis can be performed.

SpectraProc and 'SpectraProc Query' bundled with the associated spectral database is a solution for the efficient storage and pre-processing of field spectroradiometer data. The system has been successfully used in studies concerned with New Zealand native vegetation, soil properties and pastures.

## 2 Installation and Configuration

### 2.1 Database

#### 2.1.1 Installing MySQL

SpectraProc runs on versions 4 and 5 of MySQL database. SpectraProc Query only runs on version 5.

- ☐ Download the installation wizard ZIP file for Windows from [www.mysql.com](http://www.mysql.com): mysql-5.0.26-win32.zip
- ☐ Install the above ZIP file by running it
- ☐ Use standard configuration
- ☐ In the configuration section select:
  - Install as windows service
  - Root password: 'root' (or any other password, but make sure you can remember it)

#### 2.1.2 Installing MySQL-Administrator and Query Browser

- ☐ Download the windows install MSI file from [www.mysql.com](http://www.mysql.com) mysql-gui-tools-5.0-r4-win32.msi
- ☐ Install both applications by running the above MSI file

#### 2.1.3 Setting up the Spectral Database Schema, Tables and Users

- ☐ Start the Query Browser, enter 'localhost' as Server Host, 'root' for User Name, 'root' for Password (or your own password if you have chosen so), 'spectral\_db' for Default Schema. Click 'Yes' when asked if you want to create this new schema.
- ☐ Run MySQL Administrator
- ☐ Go to the User Administration tab and create a new user. User name and password are both 'SpectraProc'. Edit the Schema privileges by adding SELECT, INSERT, UPDATE AND DELETE to the assigned privileges. Assign Resources: max\_questions: 0, max\_updates: 0, max\_connections: 0 (This allows unlimited access. If you want to limit the access select positive values)
- ☐ For the new user, add 'localhost' to the list of hosts from which the user SpectraProc can connect (click on the icon under User Accounts to do this).
- ☐ Go to Restore and load the supplied Spectral\_DB\_XXXXXXX.SQL file. Select 'spectral\_db' as the target schema. This will create all tables and populate them

#### 2.1.4 Network Access

The SpectraProc application can access the database over the network if the following configurations are performed:

- ☐ The IP addresses of all machines from which the user 'SpectraProc' can connect are added to the 'SpectraProc' privileges. Alternatively a range of IP addresses can be configured using a netmask (see MySQL online documentation for details) or a value of '%' will allow access from any possible machine.
- ☐ The port of the MySQL server (3306 by default) is not blocked by Microsoft Windows

## 2.2 SpectraProc Application

Copy the SpectraProc directory supplied on the installation CD to C:\Program Files\. Open the directory after copying it and open spectra\_config.txt. Edit the configuration file as described below.

- ☐ Server: if the database is running on a machine different to the machine this copy of SpectraProc is installed on change 'localhost' to the IP address of the database server. If the database is access over the network also see 2.1.4.
- ☐ Database: if the database has been given a different name during the restore operation change the name of the database.
- ☐ Default filter settings: these are the waveband filter ranges that are used as default values in the waveband filter setup (see 5.2.7.1). More filter ranges can be added to the configuration file by adding a new line starting with 'filter\_br' for each range.
- ☐ Output drive and directory: specify the output drive and the output directory. All file output is written to this directory. This must be a valid (i.e. existing) pathname.

```
# Configuration settings for spectraProc

# database settings
server localhost
user SpectraProc
password SpectraProc
database spectral_db

# default filter settings
filter_br 1350-1440
filter_br 1790-1980
filter_br 2360-2500

# specification of file output drive and directory
output_drive C:
output_dir \Data\MPhil Remote Sensing\SpectraProc_output
```

Hint: access to SpectraProc is made easier by placing a shortcut to the executable on the desktop.

## 2.3 SpectraProc Query Interface

The SpectraProc Query Interface is written in TCL/TK. Access to the database is provided by a package called mysqltcl. TCL/TK and the package must be installed on the same machine as the SpectraProc Query Interface.

SpectraProc Query only runs on MySQL version 5 and SpectraProc DB V1.1.

### 2.3.1 Installing TCL/TK

TCL/TK is open source software and can be downloaded for free on the web on

<http://www.activestate.com/Products/ActiveTcl/>.

Choose C:\TCL as installation directory.

### 2.3.2 Installing Mysqltcl Package

Mysqltcl is a package that handles the communication between a TCL client and a MySQL database. The package can be downloaded on <http://www.xdobry.de/mysqltcl/#downloads>. Select the Windows binary version (ZIP). The ZIP file contains a folder called mysqltcl-3.02. This folder needs to be copied into the TCL library. If TCL was installed on the C: drive (C:\TCL) then the library folder is: C:\Tcl\lib.

### 2.3.3 Installing SpectraProc Query Interface

The source files of SpectraProc Query are supplied in a folder. The folder can be copied wherever seems convenient, e.g. in C:\Program Files.

### 2.3.4 Changing the Database Connection

By default the SpectraProc Query connects to the local host. If the SpectraProc database is installed on a different machine the connection string must be edited manually.

Open the file spectra\_proc\_query.tcl in some editor. Then replace the string localhost with the name or IP address of the database server.

```
#-----  
# Database Connection  
#-----  
set db_handle [mysqlconnect -host localhost -db spectral_db -user SpectraProc -password SpectraProc]
```

### 2.3.5 Running SpectraProc Query Interface

The main source file is called spectra\_proc\_query.tcl. Double clicking this file will invoke the TCL interpreter and start the application.

Tip: create a shortcut to spectra\_proc\_query.tcl on your desktop for easy access.



## 3 Upgrading the Database

### 3.1 Upgrading from MySQL version 4 to 5

For detailed information please see: <http://dev.mysql.com/doc/refman/5.0/en/windows-upgrading.html>

It is essential that you perform a backup of ALL your data and reload your data into your new database after the upgrade!!!

Please note that when upgrading the MySQL database you must also upgrade the SpectraProc database schema.

### 3.2 Upgrading SpectraProc DB V1.0 to V1.1

The SpectraProc database schema needs some minor modification as the version 5 of MySQL performs more consistency checks and therefore needs default values for BLOBs (Binary large objects). V1.1 also adds a new field 'disabled' to the spectrum table.

Both SpectraProc V1.1 and the new SpectraProc Query interface run **only** on SpectraProc DB V1.1.

The following three statements need to be executed in MySQLQueryBrowser. You need to be logged in as database administrator. Note that if your schema (database) is not called 'spectral\_db' you need to alter these statements accordingly.

```
ALTER TABLE `spectral_db`.`spectrum` ADD COLUMN `disabled` BOOLEAN NOT NULL DEFAULT 0 AFTER  
`asd_comment`;
```

```
ALTER TABLE `spectral_db`.`pca_data` MODIFY COLUMN `eigenvectors` MEDIUMBLOB DEFAULT null,  
MODIFY COLUMN `eigenvalues` BLOB DEFAULT null;
```

```
ALTER TABLE `spectral_db`.`statistic` MODIFY COLUMN `mean` BLOB DEFAULT null,  
MODIFY COLUMN `cov` BLOB DEFAULT null;
```

```
ALTER TABLE `spectral_db`.`spectrum` MODIFY COLUMN `reflectances` BLOB;
```

## 4 Design of Sampling Experiments

### 4.1 Overview

The data collected during sampling campaigns must be organised in a structured way in order to allow the automated import into the spectral database. Section 4.2 explains the background of the structure used and section 4.3 gives a practical example of a directory and file structure that should be adopted for data collection.

Preferably sampling experiments should be designed to reflect the structure used by SpectraProc. Alternatively, existing data can be arranged to meet the requirements.

### 4.2 Hierarchical Structure

A hierarchical data structure that reflects the real world and the setup of sampling campaigns for vegetation is used. This structure is derived from the following conditions:

1. Reflectances of several different species are captured
2. In order to describe the in-species variation, several specimens of a species are sampled
3. The variability of the specimens is described by several measurements per specimen

The spatial extent where a specimen is sampled is termed a sample site, thus a species contains a number of sample sites. The sites are numbered in the order of sampling. At each site, several readings are taken to capture the variation exhibited by the specimen in question. A site therefore contains a number of spectra. This leads to a hierarchical directory structure (Figure 1).

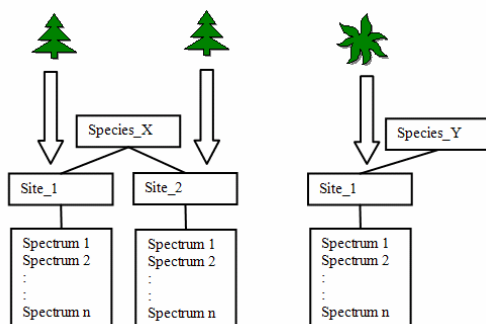


Figure 1: Hierarchical directory structure

Although the term species is used it essentially represents the different classes found in a study. These classes can either be assigned due to already existing classification systems for e.g. plants or minerals. In other cases a hypothesis might exist that a group of objects can be separated into classes. If so the setup of the experiment should mirror this hypothesis. If no such assumption exists all objects can be put into the same class (i.e. species) and the identification of classes could then be carried out by e.g. cluster analysis.

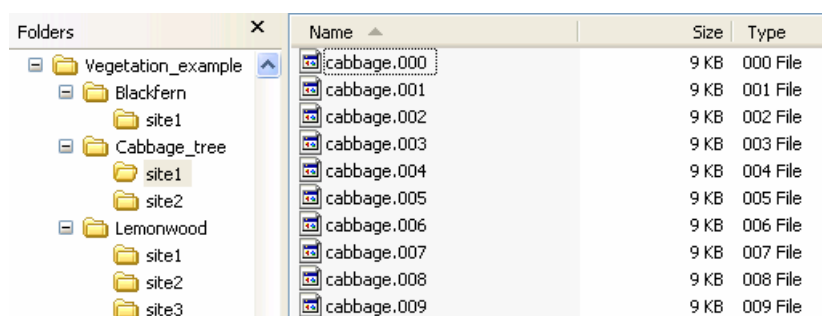
### 4.3 Directory Structure and Spectral Files

The hierarchical structure is used for the directory structure that holds the spectral files. Ideally, this directory structure is setup when designing the experiment.

Figure 2 shows an example of a directory structure. The main directory 'Vegetation\_example' holds all species directories of the study. This main directory is the folder that needs to be specified when creating a study in the database (see 5.2.2). The names given to the species directories are stored in the database and are included in file output if 'Folder name' is specified as desired name type (see 5.2.8 for more information on file export).

The species directories contain the site directories. These can be named freely and the sites will be automatically numbered when read into the database. Thus, naming a site '1' or 's1' is perfectly acceptable.

The site directories contain all spectral files collected at this site. They are auto-numbered by the ASD capturing software.



Folders	Name	Size	Type
Vegetation_example	cabbage.000	9 KB	000 File
Blackfern	cabbage.001	9 KB	001 File
site1	cabbage.002	9 KB	002 File
Cabbage_tree	cabbage.003	9 KB	003 File
site1	cabbage.004	9 KB	004 File
site2	cabbage.005	9 KB	005 File
Lemonwood	cabbage.006	9 KB	006 File
site1	cabbage.007	9 KB	007 File
site2	cabbage.008	9 KB	008 File
site3	cabbage.009	9 KB	009 File

Figure 2: Example of a directory structures holding spectral files

## 5 Operation

### 5.1 Overview

#### 5.1.1 Dataflow

A typical dataflow is illustrated in Figure 3. An ASD FieldspecPro spectroradiometer is used to capture the radiance of field objects. A GPS connected to the field laptop records the spatial position of the field object. Reflectance and metadata are automatically saved on the field laptop in a binary file for every reading taken.

These binary files are transferred to a laboratory computer where they are read by SpectraProc and stored in the relevant tables in the spectral database.

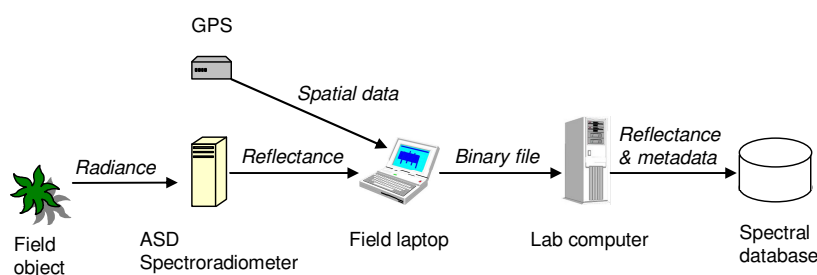


Figure 3: Dataflow and involved hardware

#### 5.1.2 File System Interfaces

SpectraProc provides input and output interfaces to the file system (see Figure 4). Input file formats are: ASD binary file as produced by the ASD FieldSpecPro Spectroradiometer, ENVI Z-Profiles that are signatures extracted from hyperspectral imagery in ENVI and sensor specifications in a proprietary, tabulator separated format. ASD files can be imported into the database as part of a study or loaded into memory for classification against a study dataset. ENVI Z-Profiles can be loaded for classification only. Sensor specification files are a way of defining new sensors in the database. Output can be written in three data formats: (1) CSV (Comma Separated Values) for import into various 3<sup>rd</sup> party applications like spreadsheets or statistic packages, (2) ENVI Spectral Library for import into ENVI and subsequent use for e.g. signature matching and (3) ARFF which is a special format used by WEKA. WEKA is a collection of machine learning algorithms for data mining tasks (University of Waikato, 2005).

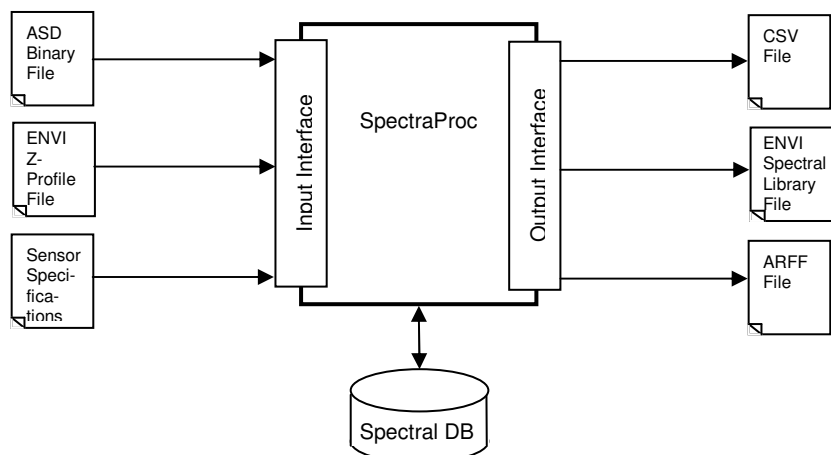


Figure 4: File System Interfaces

### 5.1.3 SpectraProc Operation

SpectraProc is study based, i.e. spectral data is grouped into studies. Advisably, a new study is created to hold data for every new sampling experiment. The spectral data files of studies must be structured according to the hierarchical structure introduced in 3. This structuring allows the automated loading of spectra into the database.

Typically, the operations carried out on a study are:

1. Creation of a new study
2. Loading of spectra
3. Repeated pre-processing and analysis

In ongoing studies, new data can be added to the database by simply selecting the loading operation again. Only new spectra will be added to the database in this case.

### 5.1.4 Spectral Processing Concept

The spectral database only stores the raw spectral data. Further processing of the spectra is performed at runtime and the results are held in memory. Once a spectrum is loaded from the database it is put through a cascade of operations as shown in Figure 5. The result of every stage is saved in a separate data structure in memory. This allows easy file export of spectral data at any processing step.

Consequently, the structure of this processing chain implicates that a change of processing parameters of any stage will influence the results of the subsequent stages. A complete set of pre-processing parameters thus describes the processing from raw to feature space transformed data.

All processing settings are stored in the database when a library is built. Repeated processing of the data with the same settings will always result in the same final data set.

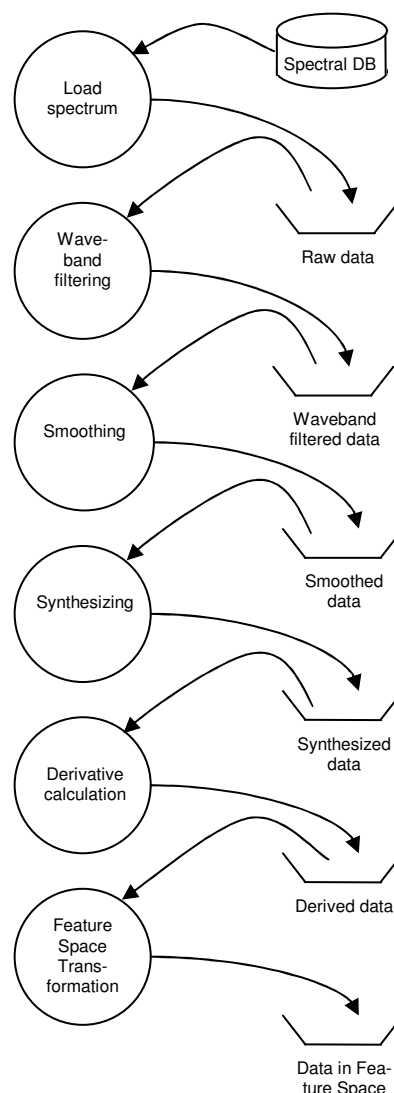


Figure 5: Spectral data processing cascade

## 5.2 SpectraProc

### 5.2.1 User Interface

The graphical user interface (GUI) is based on the structure of the processing cascade (see Figure 6). The left side of the main window consists of controls for the selection of the study and the main settings for smoothing filter, synthesizing, derivative calculation, feature space transformation and classifier discriminant function. Processing details are entered in pop up windows which are explained in the relevant sections. The text output panel (report window) in the middle of the main window is used to display processing and error information.

The listbox on top of the text output panel is used to display spectra files that are loaded directly into memory. The 'Indiv. Classify' button under it classifies the selected, individually loaded spectra against the current library.

The library status box on the top right of the screen indicates whether statistical information has been compiled for the current pre-processing settings.

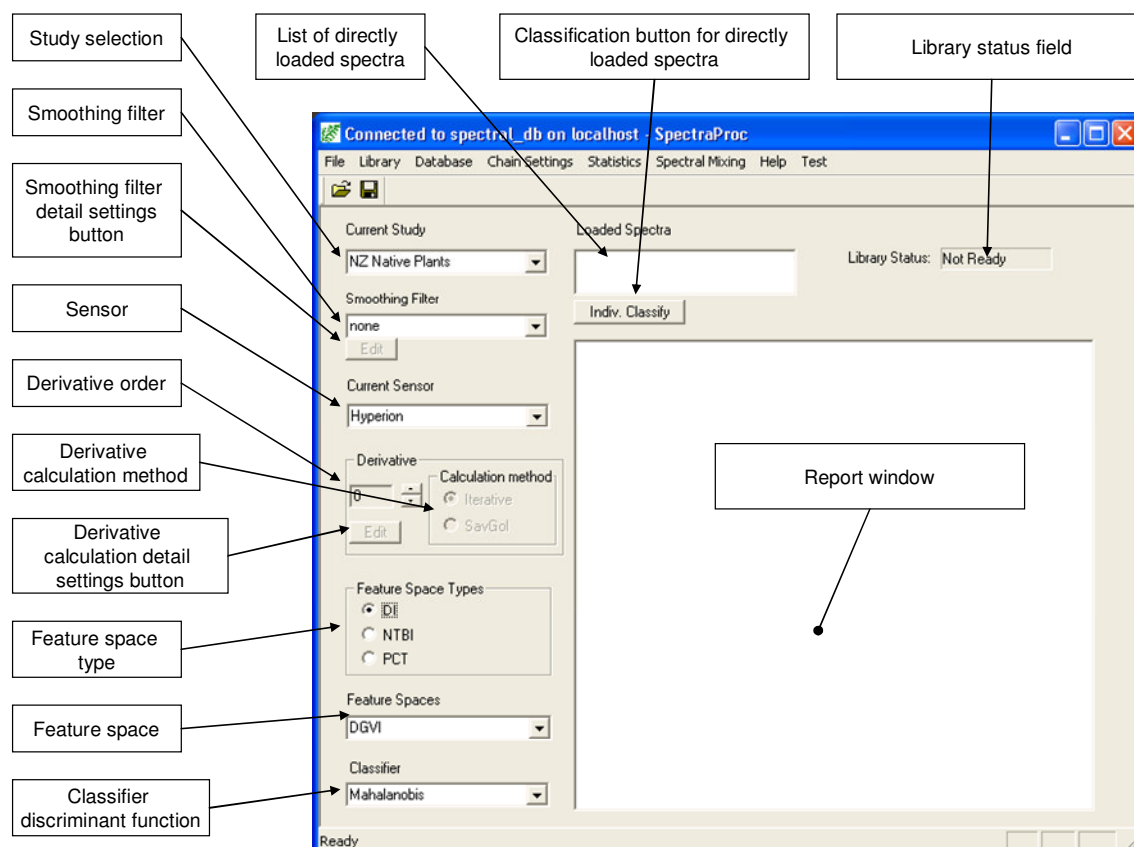


Figure 6: Main window elements

### 5.2.2 Creating a new Study

To create a new study, select 'Database' -> 'Create New Study' (Figure 7). This brings up the new study dialog (Figure 8).

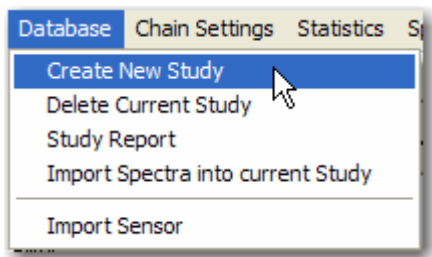


Figure 7: Create New Study menu entry

Enter a study name of 45 character maximum and not containing special characters that can not be part of any Windows file name like: \ / : \* ? < > " |.

The study description is optional and can contain any string up to 200 characters long.

The species directory path is a Windows pathname pointing to the directory that contains all species folders of this study. To set the path either type it in manually or select the 'Browse' button that brings up a directory tree (Figure 9) and select the appropriate directory.

The number of samples per species for library inclusion can be set to any positive number. If a species has less than the specified number of samples it will not be included in the statistics calculations when building libraries.

Once a new study is created it will automatically be selected as the current study.

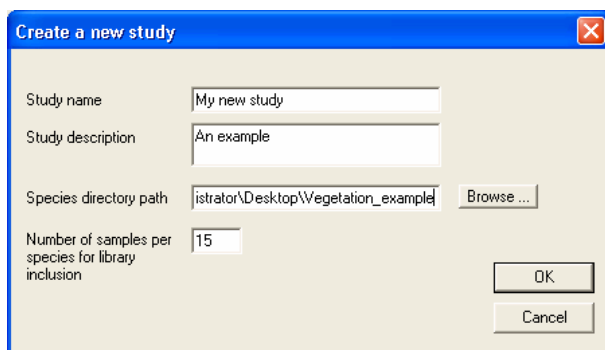
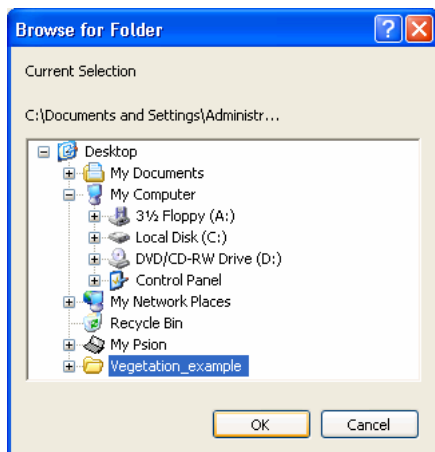


Figure 8: New study dialog

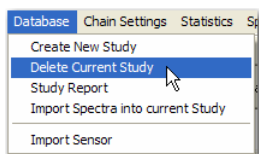
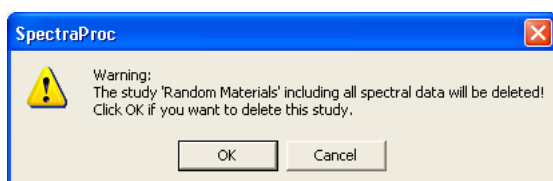


*Figure 9: Directory tree dialog*

### 5.2.3 Deleting an existing Study

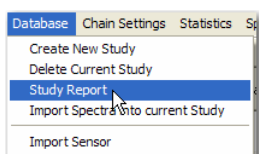
To remove a study and all associated data from the database select this study as the current study, then select 'Database'-'>'Delete Current Study' (Figure 10). Before any data is deleted a warning will be displayed (Figure 11). To delete all data of the current study click OK, to abort the operation click 'Cancel'.

The message window will list the involved operations of the deletion.

*Figure 10: Delete Current Study menu entry**Figure 11: Deletion warning box*

### 5.2.4 Study Report

To display name, description, datapath, number of minimum spectra per endmember, number of species, number of spectra and the capturing timeframe of a study select 'Database'-'>'Study Report' (Figure 13).

*Figure 12: Starting a study report*

### 5.2.5 Selecting Studies

To select a study to work with, select the appropriate study from the drop down list in main window (Figure 13).



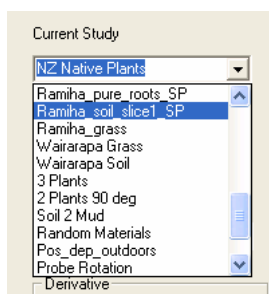


Figure 13: Selection of current study

## 5.2.6 Loading Spectra

To load the spectra of a study into the database make sure the correct study is selected. Please refer to 5.2.3 on the process of selecting the current study. Then select 'Database'-'>' 'Import Spectra into current Study' (Figure 14). A popup window will appear indicating the loading progress. Additionally, the loaded species are listed in the message window.

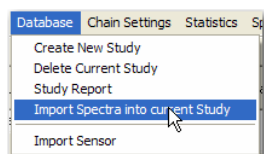


Figure 14: Import Spectra menu entry

## 5.2.7 Processing Chain Settings

### 5.2.7.1 Waveband Filtering

Waveband filtering is the process of removing unwanted regions of the spectrum. All subsequent processing operations will then ignore these regions.

A typical example is the removal of water bands. These water band regions are usually found around 1350-1440 nm, 1790-1980 nm and 2360-2500 nm (Figure 15). If contact probes are used the noise can be minimal because the atmosphere is effectively non existent.

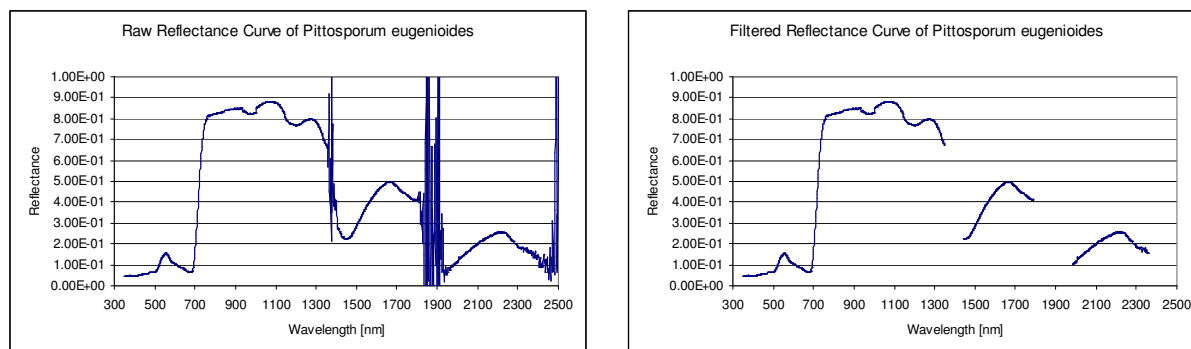


Figure 15: An example of pre and post filtering of water noise bands

The waveband filter regions are freely configurable for each study. To modify the current filter settings select 'Chain Settings'-'>' 'Waveband Filter Setup' (Figure 16).

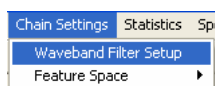


Figure 16: Waveband Filter Setup menu entry

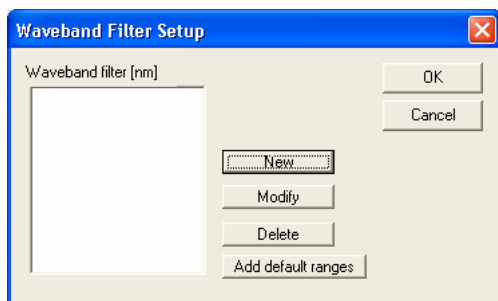


Figure 17: Waveband Filter Setup dialog (no regions defined)

The Waveband Filter Setup dialog lists the waveband filter ranges in the listbox on the left side (Figure 17). To add a new filter region click 'New'. In the pop-up window enter the lower and upper wavelengths in nanometres of the region (Figure 18). After clicking on OK in this dialog, the newly created filter range is shown in the Waveband Filter Setup dialog (Figure 19).

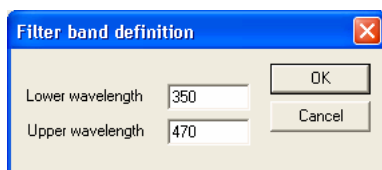


Figure 18: Filter band definition dialog

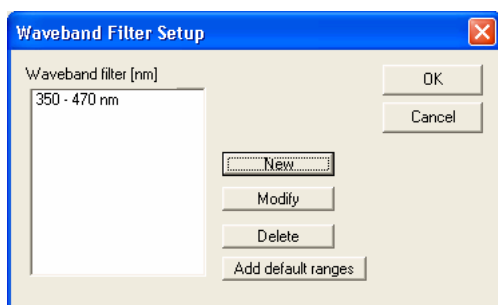


Figure 19: A new waveband filter range added to the list of filter ranges

To modify an existing range, select the range to be modified in the listbox and click 'Modify'. To delete an existing range, select the range and click 'Delete'. To add the default ranges as defined in the configuration file click on 'Add default ranges' (see 2.2 on how to configure the default filter ranges).

#### 5.2.7.2 Smoothing

Smoothing applies a smoothing function to the spectra. The default smoothing filter type is 'None' meaning no smoothing is carried out. The smoothing type is selected using the drop down list box in the main window (Figure 20). Two different Savitzky-Golay implementations are offered in this

list: Moving Window (MW) and Fast-Fourier-Transformation (FFT). The output is exactly the same, however, due to internal overhead the MW method is currently the faster of the two.

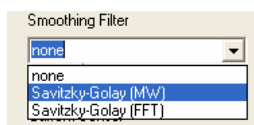


Figure 20: Smoothing filter type drop down list box

The processing details of the smoothing filter are set by clicking on the 'Edit' button below the list box. This brings up a dialog where the polynomial order and the filter window size can be changed (Figure 21). The filter size is automatically incremented resp. decremented in steps of two to keep the filter size and uneven number, i.e. the number of points on the left and right of the point to be smoothed are equal. The polynomial order sets the order of the curve that is used in the least squares approximation. Note that the Savitzky-Golay coefficients for quadratic and cubic functions are the same and the same applies to orders 4 and 5 (Savitzky and Golay, 1964).

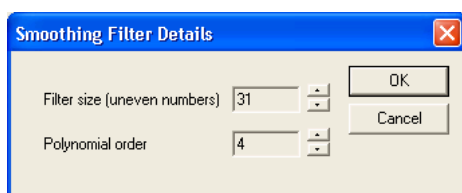


Figure 21: Smoothing Filter Details dialog

The result is automatically filtered to remove artefacts that appear at the start and end of every valid waveband segment (Figure 22). The new valid segment sizes are calculated by:

$$\lambda_{u\_smoothed} = \lambda_u - pos\_filter\_size$$

$$\lambda_{l\_smoothed} = \lambda_l + neg\_filter\_size$$

where

$\lambda_l$ ,  $\lambda_u$  = lower and upper segment wavelengths

Thus every segment loses information of  $filter\_size - 1$ .

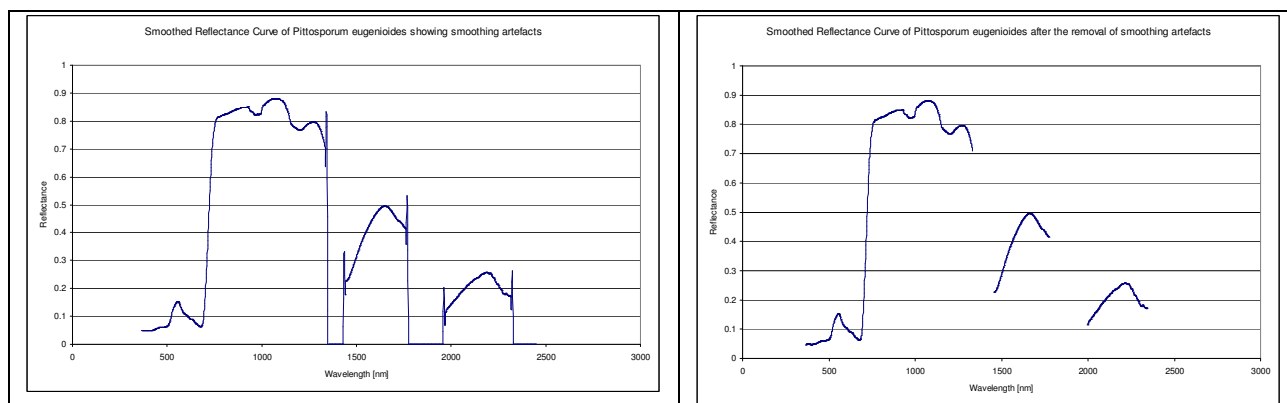


Figure 22: A smoothed signature of *Pittosporum eugenoides* before and after the removal of smoothing artefacts

Generally, bigger filter sizes remove more noise while higher polynomial orders fit the original data values better.

### 5.2.7.3 Sensor Synthesizing / Downsampling

The sensor synthesizing/downsampling is effectively a data reduction operation. The response of the sensor selected as the current sensor is calculated from ASD data.

The synthesizing of other sensor responses using ASD data is useful due to several reasons:

1. Reduction of dimensionality
2. Direct comparison of airborne/spaceborne sensor and ground data
3. Implicit smoothing of the data
4. Prediction and assessment of the usefulness of a certain sensor

Two classes of sensors are supported: Ratio and Gaussian.

#### 5.2.7.3.1 4.5.3.1 Ratio Sensors

The sensor element function of these sensors is modelled by a number of known coefficients, thus the synthesizing operation is simply a convolution of a defined wavelength region using these coefficients.

An example of such ratios is shown for Landsat7 TM band 1 (Figure 23).

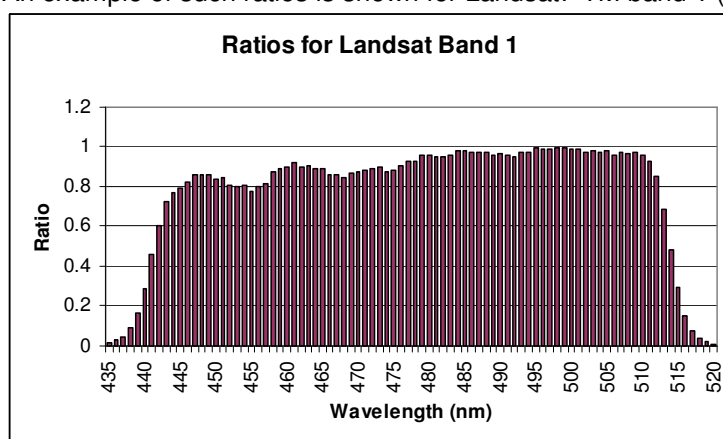


Figure 23: Ratios for Landsat7 TM band 1

The convolution is calculated by:

$$r_j = \frac{\sum_{i=lw\_j}^{uw\_j} c_i \cdot r_i}{\sum_{i=lw\_j}^{uw\_j} c_i}$$

where

$r_j$  = the synthesized reflectance value of the j-th synthesized band

$c_i$  = the coefficient for wavelength i

$r_i$  = reflectance value of i-th ASD band

$lw\_j$  = lower wavelength of the j-th band

$uw\_j$  = upper wavelength of the j-th band

#### 5.2.7.3.2 Gaussian Sensors

The sensor element response function of these sensors is best approximated by a Gaussian function. The sensor elements are technically defined by the middle wavelength and the full width at half the maximum (FWHM) (see Figure 24).

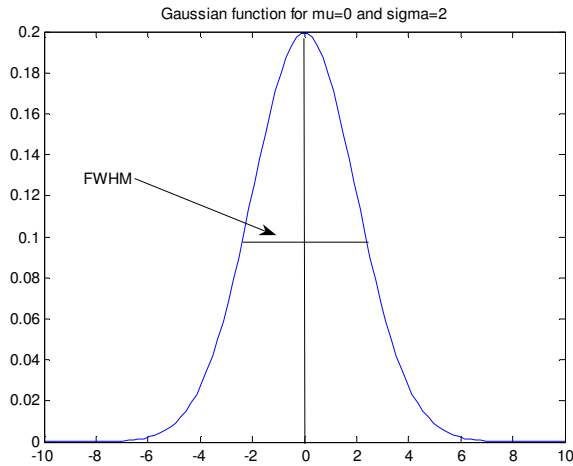


Figure 24: Gaussian curve illustrating the FWHM measure

The coefficients used for the convolution operation are given by the Gaussian function:

$$c_i = f(\text{wavelength\_band\_}i) = \frac{1}{\sqrt{2 \cdot \pi} \cdot \sigma} \cdot e^{-\frac{1}{2} \left( \frac{\text{wavelength\_band\_}i - \text{wavelength\_band\_}j}{\sigma} \right)^2}$$

where

$\text{wavelength\_band\_}i$  = wavelength of the  $i$ -th ASD band

$\text{wavelength\_band\_}j$  = wavelength of the  $j$ -th band of the sensor to be synthesized

$c_i$  = the  $i$ -th coefficient for the convolution operation

The band convolution is calculated by

$$r_j = \frac{\sum_{i=\mu-\text{range}}^{\mu+\text{range}} c_i \cdot r_i}{\sum_{i=\mu-\text{range}}^{\mu+\text{range}} c_i}$$

where

$r_j$  = the synthesized reflectance value of the  $j$ -th synthesized band

$c_i$  = the coefficient determined by the Gaussian function for the wavelength of the  $i$ -th ASD band

$r_i$  = reflectance value of  $i$ -th ASD band

$\text{range}$  = defines the range of values to be used for the band convolution symmetrically to the middle ASD waveband. The middle ASD waveband is the one closest to the average wavelength of the  $j$ -th synthesized band.

The nature of the Gaussian function effectively carries out a smoothing of the data. The range of values used for the convolution is set to 3 times the standard deviation:  $\text{range} = 3 \cdot \sigma$ , i.e. 99.74% of all contributing values are used (Papula, 1994). This range is practically not usable for all bands because some wavelengths may have been filtered previously, segmenting the spectrum. In these situations, the range is symmetrically reduced to avoid filtered regions.

## 5.2.7.3.3 Supplied Sensors

The sensors currently supplied with the software are:

Name	Type	Description
Hyperion	Gaussian	Sensor as flown on EO-1
ASD	Ratio	Effects a 1:1 copy of ASD data, i.e. no data reduction
Opt_Hyperion	Gaussian	The Hyperion sensor according to the blueprint, i.e. all sensor elements are operating
Downsampling_by_10	Ratio	Effects a downsampling by selecting every tenth waveband
Landsat7	Ratio	Sensor as flown on Landsat7 ETM satellite
ACS_210 Red NIR	Gaussian	Crop circle sensor simulation with red – NIR illumination
ACS_210 Yellow NIR	Gaussian	Crop circle sensor simulation with yellow – NIR illumination
Downsampling_by_5	Ratio	Effects a downsampling by selecting every fifth waveband

To change the current sensor select a sensor from the sensor drop down list box (Figure 25).

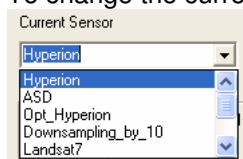


Figure 25: Selection of current sensor

It may be advisable to apply the downsampling only to smoothed data in order to avoid aliasing. The combination of smoothing and downsampling is called decimation (Fliege, 1994).

## 5.2.7.4 Derivative Calculation

The derivative order is set to zero by default, i.e. no derivative is calculated. To change the order use the up and down arrows next to the box showing the derivative order (Figure 26). If the order is changed to a non-zero value, the calculation method can be selected.

Iterative calculation is based on the gradient of the linear curve segment between two wavebands:

$$\rho'(b_i b_{i+1}) = \frac{\rho(b_{i+1}) - \rho(b_i)}{\lambda(b_{i+1}) - \lambda(b_i)}$$

where

$\rho(b_i)$  = reflectance of band i

$\lambda(b_i)$  = wavelength of band i

$\rho'(b_i b_{i+1})$  = first derivative of linear curve segment between reflectances of band i and band i+1

The n-th derivative is thus calculated by applying the above formula n times.

The SavGol method uses Savitzky-Golay coefficients to calculate the derivatives. In this case the derivative is actually the derivative of the approximating function at the wavelength of interest. Thus the SavGol method automatically performs a smoothing of the data. This can be useful when dealing with noisy data where a single smoothing operation is not removing enough noise. Both methods loose N data points per valid segment where N is the derivative order.

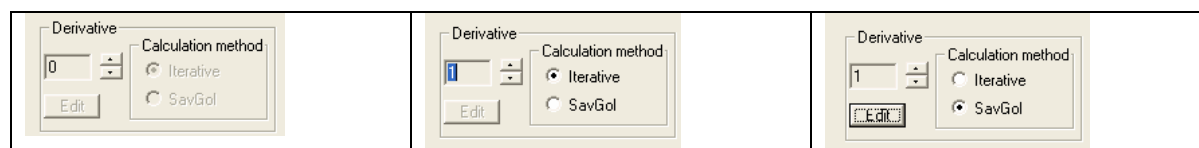


Figure 26: Derivative calculation settings: zero derivative (left), first derivative iterative (middle) and first derivative SavGol method (right)

For the SavGol method the filter size and polynomial order can be modified by clicking on the 'Edit' button. This brings up the Derivative Filter Details dialog (Figure 27). The filter size and order are automatically restricted to meet the following minimal conditions for the calculation of the derivative:

$$\text{polynomial\_order} \geq \text{derivative\_order}$$

$$\text{filter\_size} \geq \max(\text{polynomial\_order} + 1, \text{derivative\_order} + 1)$$

Thus the minimal filter size depends on both the polynomial and the derivative order. A minimal filter size of (derivative\_order+1) will result in the removal of n = derivative\_order number of points.

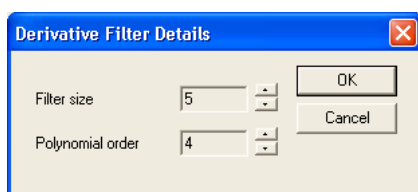


Figure 27: Savitzky-Golay settings for derivative calculation

Note that too big filter sizes remove relatively lot of data because the derivative calculation takes place after the data reduction step (sensor synthesizing/downsampling). E.g. a filter size of 5 applied to data downsampled by factor ten will remove 4 data points, i.e. 40nm, per segment.

#### 5.2.7.5 Feature Space Transformation

Three types of feature spaces are implemented:

- ☐ Derivative Indices (DI)
- ☐ Normalized Two Band Indices (NTBI)
- ☐ Principal Component Transformation (PCT)

##### 5.2.7.5.1 DI (DGVI)

DGVIs (Derivative Greenness Vegetation Indices) are an example of a DI. These indices are trying to describe the shape of the reflectance curve. The DGVI calculation is based on the equations used by Thenkabail et al. (Thenkabail et al., 2004) and Elvidge and Chen (Elvidge and Chen, 1995):

$$DGVI = \sum_{i=m}^n \frac{\rho'(b_{i-1}b_i) - \rho'(b_i b_{i+1})}{\Delta b_i}$$

where

$\rho'(b_{i-1}b_i)$  = first derivative of reflectance curve between  $b_{i-1}$  and  $b_i$

m..n = start and end band number of DGVI area

$b_i$  = centre wavelength of band i

i = band number

$\Delta b_i$  = step width:  $b_{i+1} - b_{i-1}$

##### 5.2.7.5.2 NTBI

Normalized two band indices are calculated by:

$$NTBI = \frac{\rho(b_x) - \rho(b_y)}{\rho(b_x) + \rho(b_y)}$$

#### 5.2.7.5.3 PCT

Principal components transformation requires as input the eigenvectors of a given dataset. A principal component analysis (PCA) must be carried out before PCT can be applied to the data. PCA results in eigenvalues and eigenvectors. The eigenvectors are used in the transformation matrix G. If only the first n components are to be used, the transformation matrix is a sub-matrix of G, consisting of the first n column vectors. The dimension of the resulting feature space is therefore n:

$$\underset{n \times 1}{y} = \underset{n \times m}{G} \cdot \underset{m \times 1}{x}$$

where

m = original size of data space

n = new size of data space, equal to the number of selected components

#### 5.2.7.5.4 Creation, Modification and Deletion of Feature Space Definitions

Based on the three basic feature space types new feature spaces can be created. Existing feature spaces can be edited or deleted. For any of these functions navigate to the 'Chain Settings'-'>Feature Space' submenu (Figure 28).



Figure 28: Feature Space Submenu

To create a new feature space select 'Copy and Edit' or 'New' in the 'Feature Space' submenu. 'New' opens up an empty dialog for the definition of a new feature space. 'Copy and Edit' copies the settings of the currently selected feature space into the dialog. The settings can subsequently be changed before creating the new feature space in the database.

To modify an existing feature space select 'Edit'.

All the above three commands invoke the same dialog, shown here on the example of a NTBI feature space consisting of three individual NTBIs (Figure 29). The name of the feature space can be any alphanumeric string of maximal 45 characters, the description can be any alphanumeric string of maximal 250 characters. The bands listbox lists the band combinations used to calculate the indices. E.g. index 1 uses bands 550nm and 468nm. New band combinations are entered by clicking on 'New', existing band combinations are modified by clicking modify. Both commands bring up a dialog box to define the two bands (Figure 30). To delete an existing band combination select the combination in the list and click 'Delete'.

The dimension field in the Feature Space dialog is non editable for DIs and NTBIs where it reflects the number of band combinations. For PCT feature spaces, the dimension field specifies how many components are to be used for the transformation. In the given example, the first 10 components will be used (Figure 31).



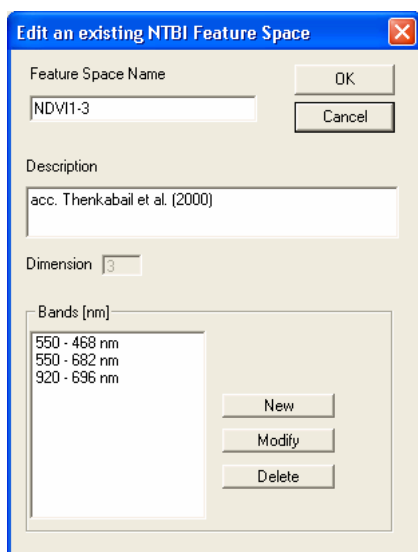


Figure 29: Feature Space edit dialog on the example of a NTBI feature space

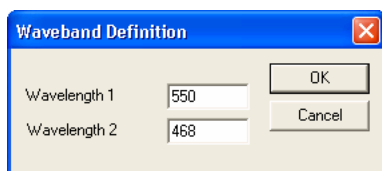


Figure 30: Waveband Definition Dialog

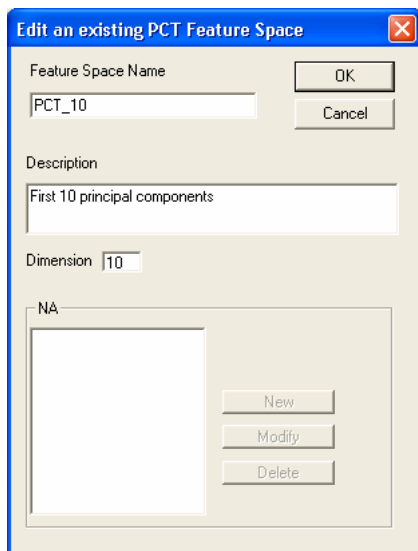


Figure 31: PCT Feature Space dialog

To delete an existing feature space make sure the currently selected feature space is the one to be deleted. Then select 'Delete' from the 'Feature Space' submenu. The current feature space is removed from the database.

#### 5.2.7.5.5 Feature Space Selection

The feature space selection is comprised of two panels: (a) a selection of the feature space type offering DI, NTBI and PCT and (b) a drop down list box that contains all feature spaces of the current type (see Figure 32).

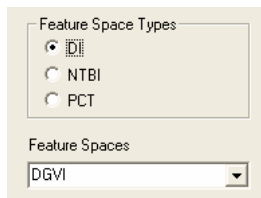


Figure 32: Feature space settings

### 5.2.8 File Export

SpectraProc features a powerful file export that facilitates the analysis and visualisation of the data in third party products. All output files are written to the directory

The output is controlled by a single dialog (see Figure 33). The file export dialog is displayed by selecting 'File'-'>'Export'.

Data can be exported at any processing stage:

- ☐ Raw
- ☐ Filtered (removal of noisy band regions)
- ☐ Smoothed
- ☐ Synthesized
- ☐ Derived
- ☐ Feature Space transformed

The processing for these stages depends upon the parameters that are defined at that point in time for the processing cascade. If e.g. 'Synthesized' is selected, every spectrum is put through the cascade of filtering, smoothing and synthesizing and is then written to the file. I.e. if a certain stage is selected all the preceding stages are automatically performed.

The data can be averaged either on site level or on species level. If the site mean is chosen a column containing the site number will be added to the output file. Selecting the 'All spectra' option performs no averaging but writes every spectrum to the file. The averaging is the last step of the processing, e.g. if the 'Synthesized' and 'Mean per species' is selected, then all spectra of each species are processed till the synthesizing stage and then averaged and written to the file.

Three file formats are currently supported:

- ☐ CSV (Comma Separated Values)
- ☐ ENVI spectral library file
- ☐ ARFF (WEKA file format)

The species that are written to the file can be controlled by selecting on of four options:

- ☐ Include all species: all species are written
- ☐ Include only library relevant species: only species that have equal or more than the minimum number of spectra are written. The minimum number of spectra is set when defining a new study (see 5.2.2)
- ☐ Only endmembers: only species that have been designated as endmembers are written
- ☐ Only mixtures: only species that are not designated endmembers are written

The name that is used as species name can be one out of five choices:

- ☐ Common: common name is defined in database
- ☐ Maori: Maori name as defined in database
- ☐ Latin: Latin name as defined in database
- ☐ Folder: name of the folder that contains the site directories of the species

- ☐ Auto number: incrementing numbers are assigned to the species according to their order in the database

The current version of the software supports only one file level setting: study. I.e. all species of the current study are written to one single file.

Four checkboxes allow for the following operations:

- ☐ Transpose: the observations are written as columns instead of rows. This option can make the import of data into other programs easier.
- ☐ Include filtered bands: this is for visualisation purposes only. The regions that were removed in the waveband filtering stage are written as empty cells, i.e. when plotted the filtered regions will appear as gaps.
- ☐ Offset: each species will be set off by a freely choose able percentage of 100% reflectance. This is to be used for visualisation purposes only.
- ☐ Interpolate: this option is only available when the 'Synthesized' stage is selected as output step. The synthesized data is interpolated to ASD data using straight line segments. This is useful for the assessment of the smoothing effect of the synthesizing operation e.g. by plotting a noise spectrum (smoothed minus interpolated synthesized data).
- ☐ Sort species numerically: Sorts the species by interpreting the species names as numbers

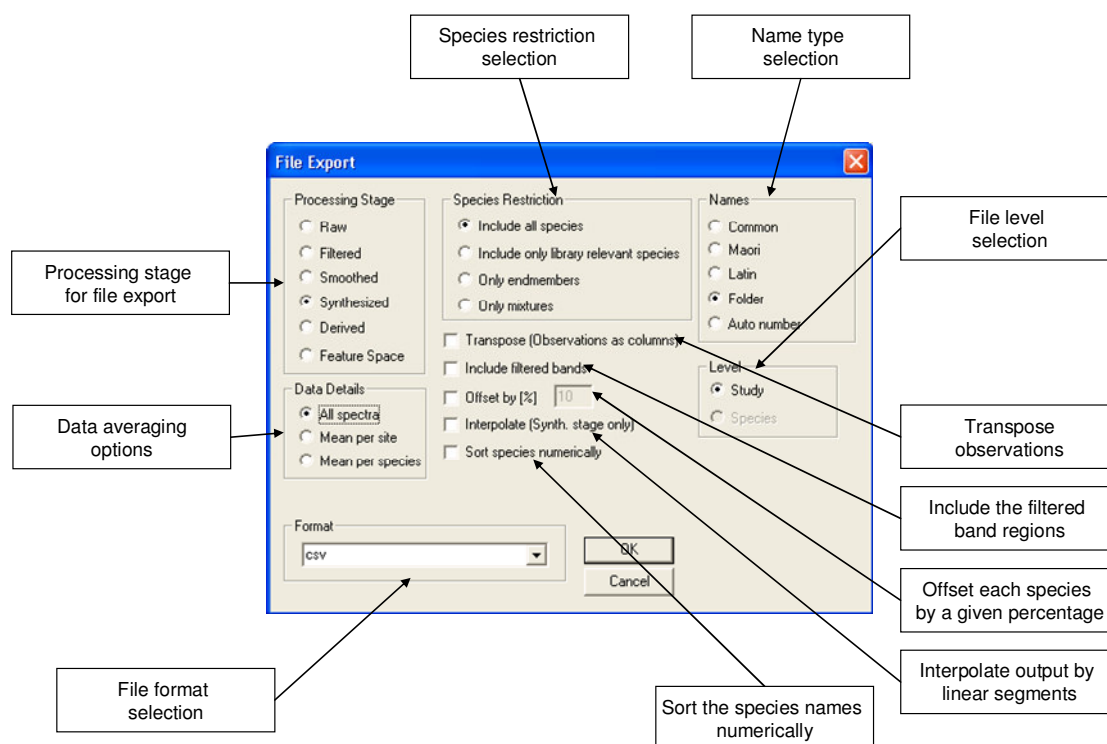


Figure 33: File Export dialog

## 5.2.9 Saving the Contents of the Report Window

The contents of the report window can be saved to a file by selecting: File -> Save Report Window Content. The file is written to the default output directory and named report\_window\_output.txt (see Figure 34).

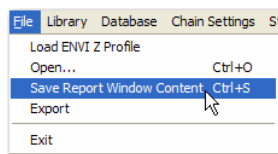


Figure 34: Saving the report window content

## 5.2.10 Spectral Mixtures

Spectral mixtures occur when more than one endmember is present in the field of view. If the endmembers are known the mixtures can be unmixed to give estimates of the abundance of each endmember in the mixture. For the process of unmixing refer to 5.2.12.4.

The unmixing process needs to have a set of endmembers and a set of mixtures to be unmixed using the endmember information. For this purpose species can be designated as endmembers. If the spectral mixtures are the result of a controlled mixing experiment the abundances of the endmembers are usually known. These known endmember fractions can be entered into the database for each mixture, thus allowing the automatic calculation of the error involved in the estimation by the unmixing procedure.

### 5.2.10.1 Endmember Designation

To display the endmember designation dialog choose 'Spectral Mixing' -> 'Endmember Selection' (Figure 35).

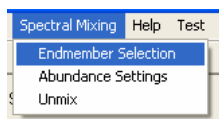


Figure 35: Endmember Selection menu entry

Initially, all species will be listed in the 'Classes' list while the 'Endmembers' list is empty. To designate a species as an endmember select the class by clicking on it, then click on the arrow pointing towards the right to shift this class into the Endmember list. Similarly, to move an endmember back to the Classes list, select the endmember and click the arrow pointing towards the left.

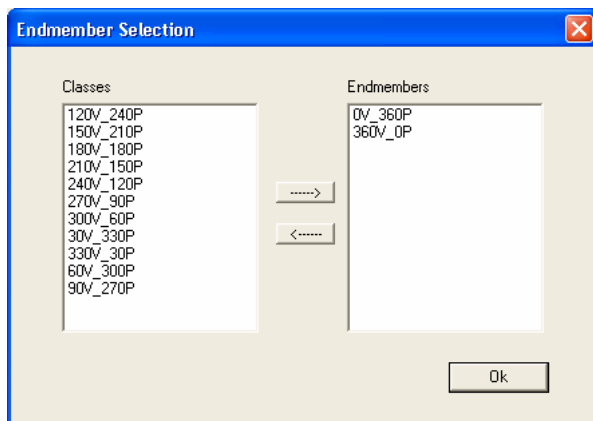


Figure 36: Endmember Selection dialog

### 5.2.10.2 Abundance Settings

Before invoking the abundance settings the endmembers should be selected as described in 5.2.10.1.

To display the abundance settings dialog select Spectral Mixing'->'Abundance Settings'.

All mixtures are listed in the 'Classes (excl endmembers)' list box. The abundances of the endmembers are set for each of these mixtures by:

1. Selecting the mixture
2. Select the endmembers that are contained in the mixture one after another from the 'Available endmembers' list box and shift them down into the 'Selected endmembers' list box.
3. For every selected endmember do: select the endmember by clicking on it, then enter the abundance in the provided box. Click 'Save'. The total abundance is updated. When defining the abundance of the last endmember, click on 'Complement'. This automatically assigns the abundance to the last endmember by summing the total abundance to 1.

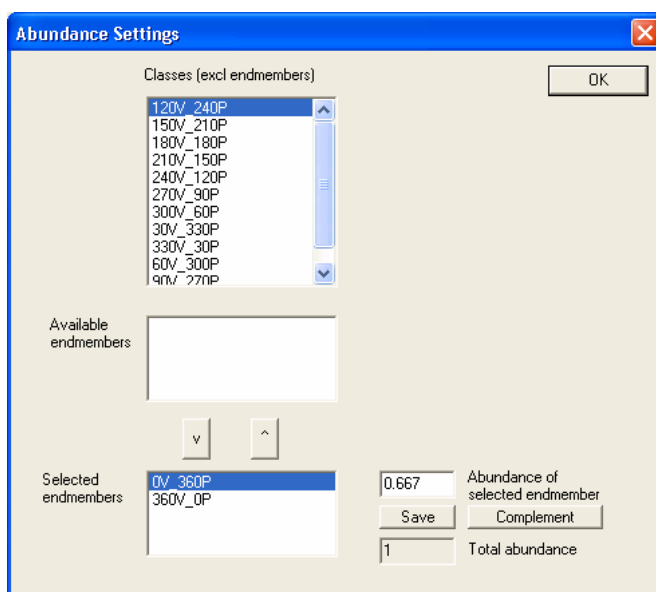


Figure 37: Abundance Settings dialog

## 5.2.11 Library Building

A library is a collection of statistics data built for a certain configuration of the processing chain. The 'Library status' field in the main window indicates if a library has been built for the current processing settings. To build a library select 'Library'->'Build'.

## 5.2.12 Analysis

### 5.2.12.1 Separability

Before running the separability analysis the library for the current pre-processing settings must be built. See 5.2.11 on how to build libraries. The JM and B distance analyses are carried out in the currently selected feature space. I.e. the discriminating power of feature spaces can be assessed by the separability analysis.

To run the separability analysis select 'Statistics'->'Separability Report'. In the displayed JM/B Distance dialog select the name and distance type and press OK (Figure 38).

A short message will be displayed in the message window, looking similar to:

```
Separability Analysis  
Using JM and B distance
```

```
JM and B output written to file:  
C:\Data\MPhil Remote Sensing\SpectraProc_output\JM and B output.csv
```

The output filename is shown in this message. Note that only library relevant species are included in this report.

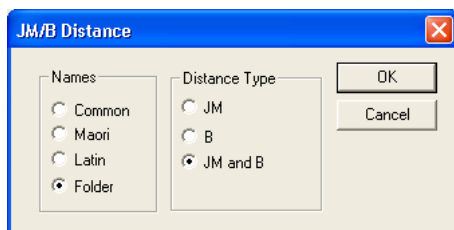


Figure 38: JM/B Distance dialog

The JM distance is listed in the upper triangular matrix, the B distance in the lower triangular matrix (see Table 1). The JM distance is asymptotic to 2.0. A JM distance of 2.0 indicates a complete separability and a distance of 1.9 a good separability (Richards, 1993). The B distance is not asymptotic and can get infinite (Inf.) due to limits in numerical precision.

Table 1: Example of a JM and B output using a 10 dimensional DI feature space

	Blackfern	Cabbage_tree	Lemonwood
Blackfern	-	1.93	1.90
Cabbage_tree	3.41	-	1.76
Lemonwood	2.99	2.11	-

### 5.2.12.2 Discriminant Analysis

Discriminant analysis (DA) uses a discriminant function to assign a class to samples. In the case of the calibration set being subjected to DA the result is an assessment of the separability of the classes. The DA is carried out in the currently selected feature space. The DA result is thus also an assessment of the discriminating power of the feature space.

Note that the discriminant function used is the one currently selected in the main window.

Before running DA the library for the current pre-processing settings must exist. See 5.2.11 on how to build libraries.

To run the DA select 'Statistics'-'>'Discriminant Analysis'. A dialog will be displayed to select the names to be used in the output (Figure 39).

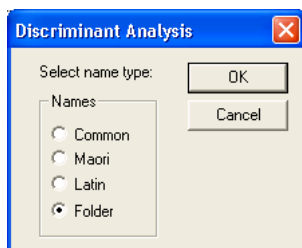


Figure 39: Discriminant Analysis dialog

A message stating the file name of the error matrix and the overall accuracy is reported in the message window, similar to the example shown below.

Discriminant Analysis

-----  
Error matrix written to file:

C:\Data\MPhil Remote Sensing\SpectraProc\_output\Error Matrix.csv

-----  
Overall Accuracy: 78.46154  
-----

The error matrix written to a file contains the errors of omission and commission, the total accuracy, the producer and user accuracy and the minimum, maximum and mean for both producer and user accuracy.

Table 2: Error matrix of DA using a 3 dimensional NTBI feature space

	Blackfern	Cabbage_tree	Lemonwood	Row Total	User Accuracy
Blackfern	11.00	0.00	0.00	11.00	100.00
Cabbage_tree	7.00	19.00	6.00	32.00	59.38
Lemonwood	0.00	1.00	21.00	22.00	95.45
Column Total	18.00	20.00	27.00	65.00	
Producer Accuracy	61.11	95.00	77.78		78.46
Prod. Acc. Min	61.11				
Prod. Acc. Max	95.00				
Prod. Acc. Mean	77.96				
User Acc. Min	59.38				
User Acc. Max	100.00				
User Acc. Mean	84.94				

To run the DA with an independent data set choose 'Statistics'-'>'Discr. Analysis using indep. set'. A dialog is displayed to select the study that contains the independent set (Figure 40). Note that in order to work correctly the folder names of corresponding species in the training and in the independent study must be identical.

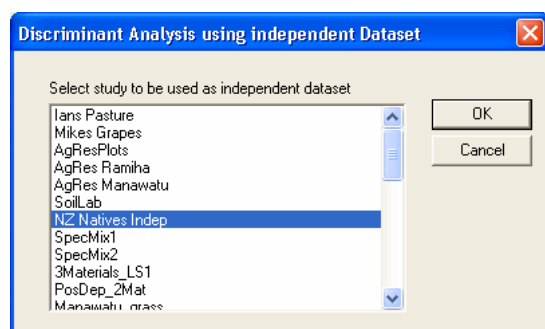


Figure 40: Selection dialog for independent data sets

### 5.2.12.3 Principal Component Analysis

Principal components analysis (PCA) is applied after the derivative calculation step. I.e. the PCA is carried out for the current settings of waveband filtering, sensor synthesizing and derivative calcula-

tion. To run PCA select 'Statistics'-'>'PCA'. A message stating the output file similar to the one below is displayed in the message window:

```
Principal Component Analysis
```

```
-----
```

```
PCA output written to file:
```

```
C:\Data\MPhil Remote Sensing\SpectraProc_output\pca_output.csv
```

```
-----
```

The two first components will be plotted as a scatterplot in a graphics window, this plot is also known as scoreplot.

The pca\_output file contains the eigenvalues, proportions, cumulative proportions (see Table 3 for an example) and the eigenvectors (also known as factor loadings). The eigenvectors can possibly be used to assess the significance of the wavelengths in respect to the each component. The eigenvalues and eigenvectors are stored in the database.

*Table 3: First ten eigenvalues and proportions of a PCA*

PC#	Eigenvalue	Proportion	Cumulative
1	2.211	0.944	0.944
2	0.097	0.042	0.986
3	0.019	0.008	0.994
4	0.007	0.003	0.997
5	0.005	0.002	0.999
6	0.001	0.001	0.999
7	0.000	0.000	0.999
8	0.000	0.000	1.000
9	0.000	0.000	1.000
10	0.000	0.000	1.000

#### 5.2.12.4 Unmixing

The unmixing procedure implemented is experimental at this stage and works on Landsat7 data only. Before starting the unmixing, make sure that the current sensor is Landsat7, the study has the mixtures and endmembers defined and the abundances are entered for each mixture.

The model used is unconstrained (Keshava and Mustard, 2002):

$$\hat{a}_U = (S^T S)^{-1} S^T x$$

where:

$x$  = spectrum vector to be unmixed ( $L \times 1$ )

$S$  = endmember matrix ( $L \times M$ ) consisting of  $M$  endmembers with the columns being the endmember spectra vectors

$\hat{a}_U$  = the unconstrained least squares solution for the abundances of the endmembers in the spectrum vector  $x$

To start the unmixing select 'Spectral Mixing'-'>'Unmix'. For each mixture the unmixing results are displayed in the message window; first stating the calculated abundances for each endmember, then the abundances stored in the database and finally the absolute errors for each endmember abundance (see example of such an output below).

```
Spectral Unmixing
```

```
-----
```

```
120V_240P
```

```
-----
```

```
0.69  0V_360P
```

```
0.35  360V_0P
```



```
simulated mixture abundances from database:  
0.67 0V_360P  
0.33 360V_0P
```

```
absolute errors:  
0.03 0V_360P  
0.02 360V_0P
```

### 5.2.13 Adding new Sensors

New sensors can be added to the database if their properties are known. The sensors details must be entered in a tabulator formatted file as described below. Two classes of sensors are supported: Gaussian and Ratio (see also 5.2.7.3).

The header structure is identical for both sensor classes:

```
<Sensor name>  
<Sensor description>  
<Column headers>
```

The sensor name should not include any of the following characters: \ / : \* ? < > " | .

The column header line is not interpreted by the program and exists only for the convenience of the user preparing the sensor description file.

For Gaussian sensors the file body contains one line per sensor band:

```
<Band> TAB <Average Wavelength (nm)> TAB <FWHM (nm)> TAB <Calibrated (Y/X)>
```

For Ratio sensors the file body contains 1 to many lines per band to model the response (see also Figure 23):

```
<Band> TAB <Wavelength(nm)> TAB <Ratio>
```

Examples of Gaussian and Ratio sensor description files are given in tables 4 and 5:

Table 4: Example of a Gaussian sensor (Hyperion)

Hyperion			
Hyperion Sensor	as flown on EO1		
Hyperion Band	Average Wavelength (nm)	FWHM (nm)	Calibrated
1	355.59	11.3871	X
2	365.76	11.3871	X
3	375.94	11.3871	X
4	386.11	11.3871	X
5	396.29	11.3871	X
6	406.46	11.3871	X
7	416.64	11.3871	X
8	426.82	11.3871	Y
9	436.99	11.3871	Y
10	447.17	11.3871	Y
11	457.34	11.3871	Y
12	467.52	11.3871	Y
13	477.69	11.3871	Y
14	487.87	11.3784	Y

Table 5: Example of a Ratio sensor (Landsat7)

Landsat7

Landsat 7 Thematic Mapper Sensor

Band	Wavelength (nm)	Ratio
1	435	0.016
1	436	0.027
1	437	0.048
1	438	0.094
1	439	0.167
1	440	0.287
1	441	0.459
1	442	0.605
1	443	0.728
1	444	0.769

To import a new sensor select 'Database'-'>'Import Sensor'. In the Import Sensor dialog select the sensor response type and the input file by clicking on 'Browse'. Then click OK and the sensor data will be imported. An according message will be displayed in the message window when the sensor is loaded.

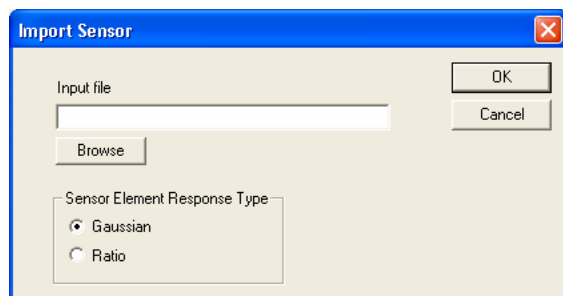


Figure 41: Import Sensor dialog

## 5.2.14 Linking to a GIS System

If a GPS unit is used to record the spatial position of the sampling sites during the sampling process every site in the database contains the latitude, longitude and altitude in WGS84 format. The database tables can be accessed by a GIS system if ODBC connections are supported. The connection is shown here on the example of ArcGIS.

### 5.2.14.1 Setting up the Spectral DB as a Data Source

Note that before the Spectral DB can be configured as a data source the MySQL ODBC driver (available from the MySQL website) must be installed on the system. Start 'Data Sources (ODBC)' from the Windows administrative tools. In the New Data Source dialog (Figure 42) select the MySQL ODBC Driver and click on 'Finish'.

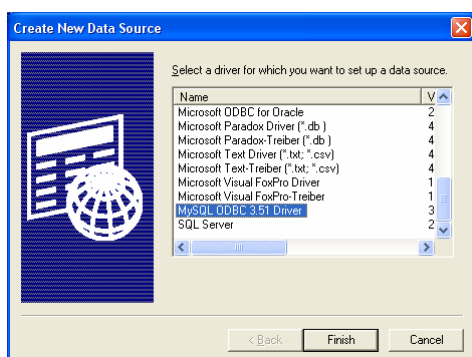


Figure 42: New Data Source dialog

This brings up the ODBC configuration dialog (Figure 43). Type in a name for the Data Source Name. Set the Server to the address of the database server. Enter the User name and Password (SpectraProc). Select the database schema containing the spectral data from the Database drop-down list.

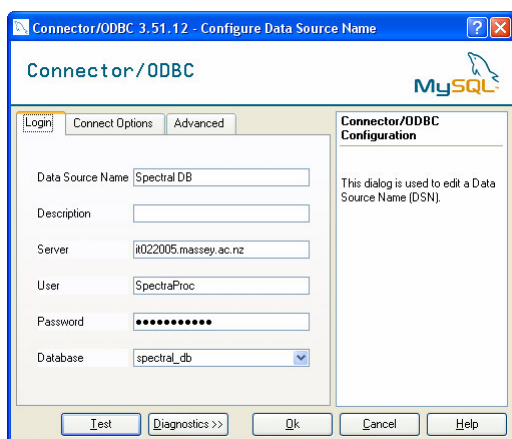


Figure 43: MySQL ODBC Configuration dialog

### 5.2.14.2 Establishing a Database Connection in ArcCatalog

Start ArcCatalog and select 'Add OLE DB Connection' (Figure 44).

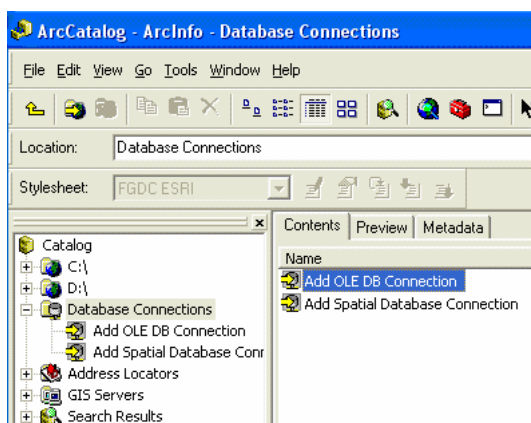


Figure 44: Adding an OLE DB Connection in ArcCatalog

In the Data Link Properties select 'Microsoft OLE DB Provider for ODBC Drivers' (Figure 45) and click 'Next>>'.

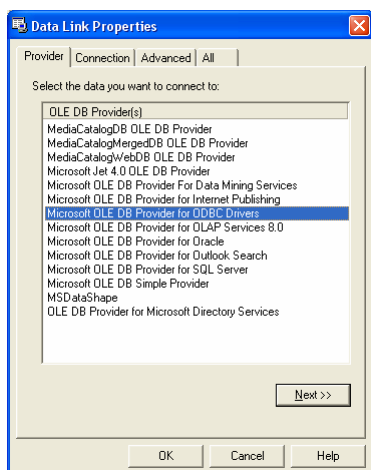


Figure 45: Data Link Properties; Provider tab

In the Connection tab select the data source from the dropdown list and enter again the username and password (SpectraProc) (Figure 46). Click 'OK'.

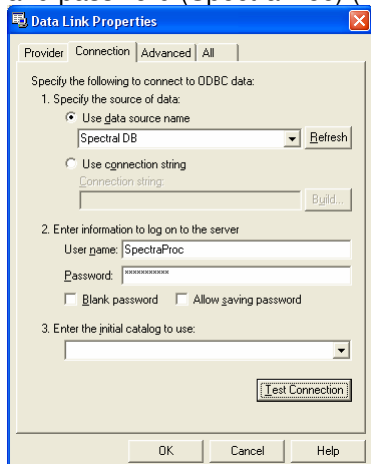


Figure 46: Data Link Properties; Connection tab

The database is now listed in ArcCatalog and the tables are displayed when selecting the database (Figure 47).

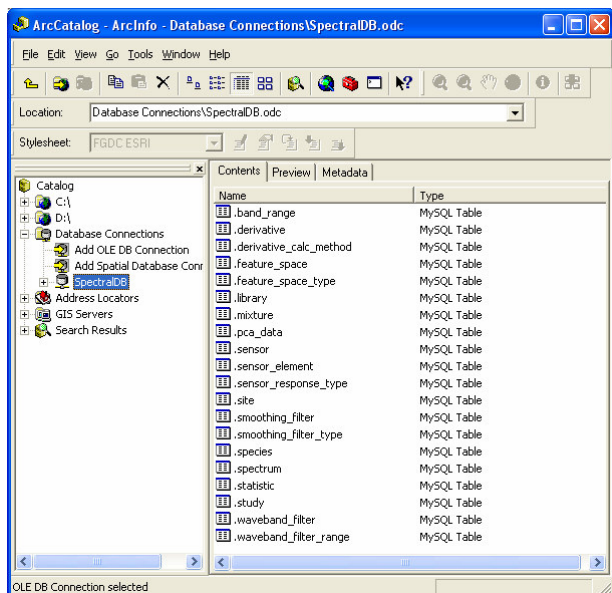


Figure 47: SpectralDB and tables in ArcCatalog

#### 5.2.14.3 Adding Table Data to a Map

To add table data to a map select 'Add Data' in ArcMap and choose the SpectralDB under Database Connections (Figure 48). Then select the site table as the data source (Figure 49). The sampling sites can then be displayed on the map by selecting 'Display XY Data' and using the longitude and latitude fields of the site table as X and Y coordinates.

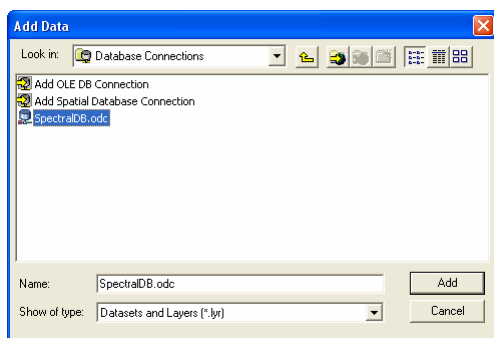


Figure 48: Selecting the SpectralDB as data source

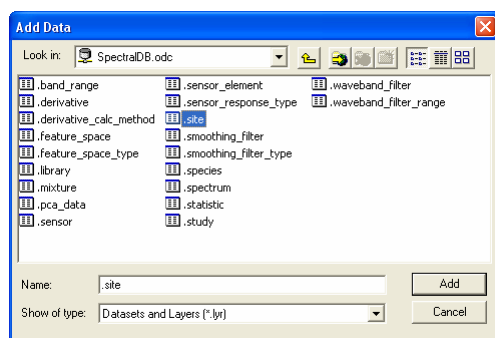


Figure 49: Selecting the site table as data source

## 5.3 SpectraProc Query Interface

### 5.3.1 Introduction

SpectraProc Query was written as an example of an intuitive, interactive query building interface. It was written in TCL/TK with the goal to get to know the characteristics of that language. It has proven to be extremely powerful for rapid development of graphical user interfaces. A further advantage are the string handling capabilities that are useful for the building of SQL queries. One of the shortcomings of SpectraProc is the limitation in graphic output. Certainly most users would like to see their spectral data visualized. Furthermore, an overview over the different studies (campaigns), species and sites was not available via the SpectraProc application. SpectraProc Query can build reports on four different levels: campaign (study), species, site and spectrum. The report output can be restricted by specifying conditions.

### 5.3.2 Database Connection

The database connection is currently hard coded and SpectraProc Query will consequently only run on the localhost, i.e. the same machine as the database is installed on. This can be changed by editing the file `spectra_proc_query.tcl`. The **localhost** string must be replaced with the network name of the database server (this is similar to running SpectraProc over a network).

```
#-----  
# Database Connection  
#-----  
set db_handle [mysqlconnect -host localhost -db spectra_db -user SpectraProc -password SpectraProc]
```

### 5.3.3 User Interface

This section describes the user interface of SpectraProc Query. Please refer to Figure 50 for an example of the GUI.

As indicated above, the resulting rows in reports can be restricted by setting conditions. These conditions are situated at the left side of the GUI.

The campaign conditions refer to conditions that restrict the campaigns (studies). The campaigns can be restricted by either choosing a specific campaign name in the Campaign Name Selection box or by entering a string in the Campaign Name Wildcard String box. The latter also accepts wildcards: % stands for any string and \_ stands for a single character.

The species conditions restrict the species by their folder names. A name can be selected in the Species Name Selection box or be entered as a string with wildcards in the Species Name Wildcard String box. A restriction of the campaigns has a direct impact on the names available in the list of species names: only species contained in the resulting campaigns are available in the list.

The result set can be temporally restricted by entering dates in the Collection Date From and To Fields. The dates have the format DD MM YYYY. They greyed out fields to the right of the date fields automatically show the resulting dates when one of the date fields is filled. The years are automatically restricted to the maximal or minimal year of all records in the database.

Spatial restrictions can be added by specifying a threshold for the altitude in the Minimal/Maximal Altitude Fields.

The report level is chosen by clicking on the option buttons in the Report Level Selection.

The resulting SQL query is displayed in the Auto-Built SQL Query Text Field.

The number of records (rows) that will be selected by the current query is displayed in the Result Set Counter field.

Any change in conditions or report level will automatically update the auto-built query and the number of resulting rows.

Finally, there are two buttons at the bottom right side of the GUI. 'Exec Query' will run the current query and display a report in a new window. 'Reset' will return all conditions to their starting values and reset the level to 'Campaign'.

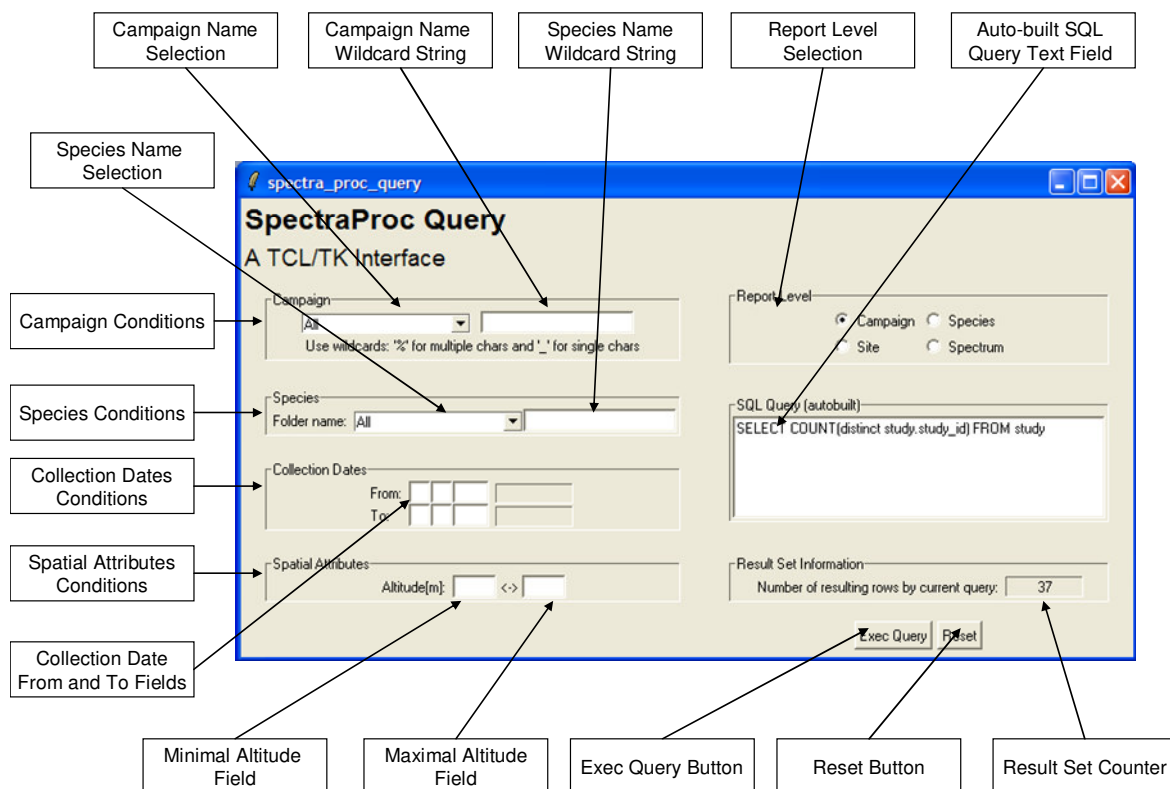


Figure 50: SpectraProc Query GUI

### 5.3.4 Report Windows

The number of report windows is not restricted by the software, although a limit might exist due to memory or operating system restrictions.

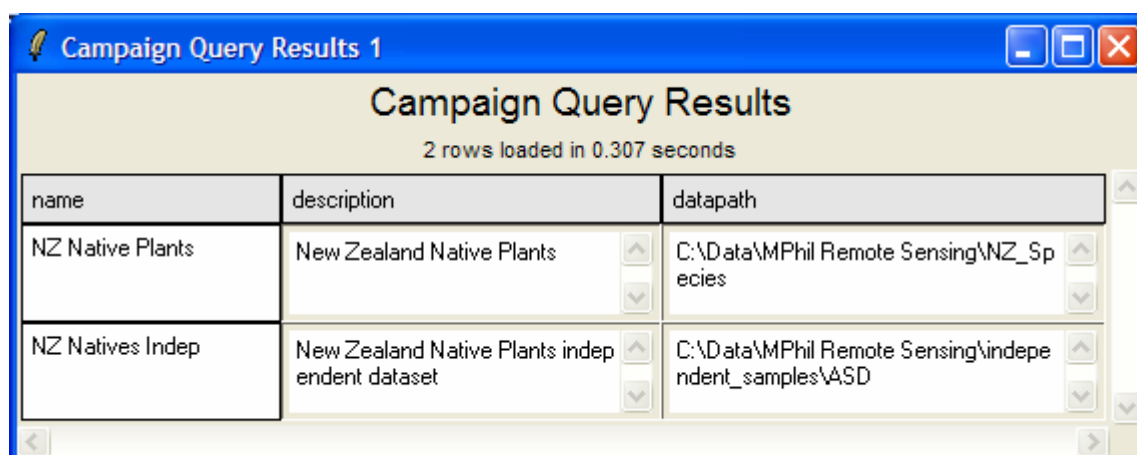
All reports start with a title that contains the level of the report plus a subtitle indicating the time taken for the report building. This is followed by the main table containing the data.

#### 5.3.4.1 Campaign Report

An example of a campaign report is shown in Figure 51.

The reported attributes are: campaign name, description and file system path.





**Campaign Query Results 1**

**Campaign Query Results**  
2 rows loaded in 0.307 seconds

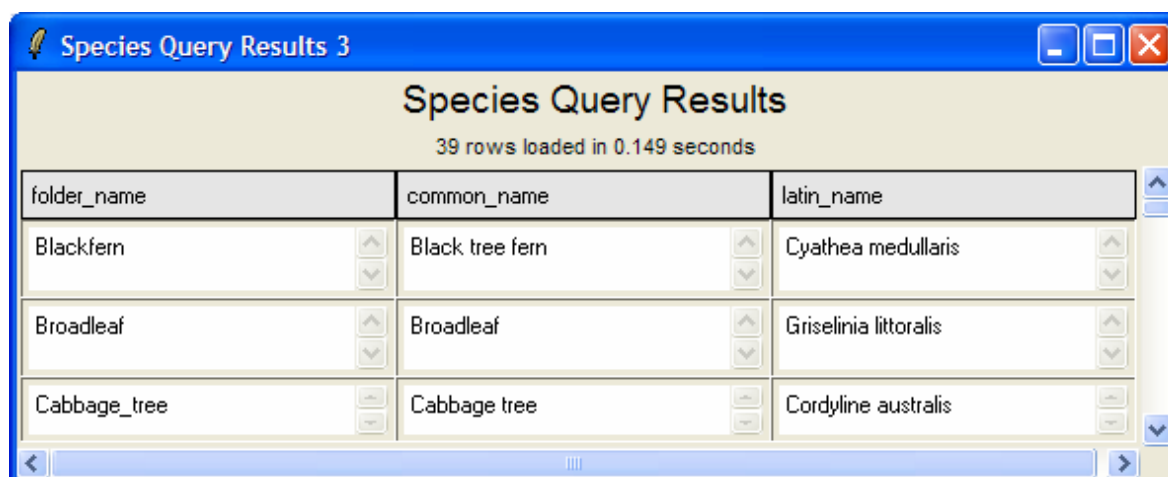
name	description	datapath
NZ Native Plants	New Zealand Native Plants	C:\Data\MPhil Remote Sensing\NZ_Species
NZ Natives Indep	New Zealand Native Plants independent dataset	C:\Data\MPhil Remote Sensing\independent_samples\ASD

Figure 51: Campaign Report

#### 5.3.4.2 Species Report

An example of a species report is shown in Figure 52.

The reported attributes are: folder\_name, common\_name, latin\_name.



**Species Query Results 3**

**Species Query Results**  
39 rows loaded in 0.149 seconds

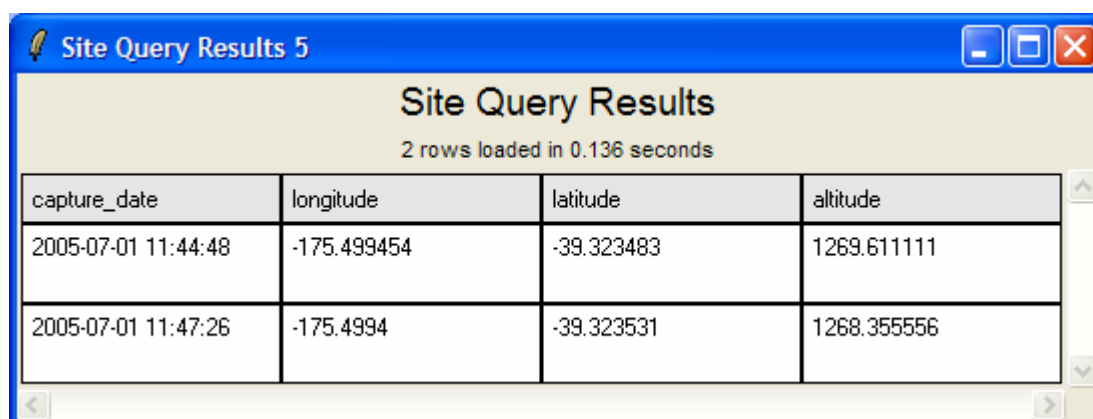
folder_name	common_name	latin_name
Blackfern	Black tree fern	Cyathea medullaris
Broadleaf	Broadleaf	Griselinia littoralis
Cabbage_tree	Cabbage tree	Cordyline australis

Figure 52: Species Report

#### 5.3.4.3 Site Report

An example of a site report is shown in Figure 53.

The reported attributes are: capture\_date, longitude, latitude and altitude.



Site Query Results

2 rows loaded in 0.136 seconds

capture_date	longitude	latitude	altitude
2005-07-01 11:44:48	-175.499454	-39.323483	1269.611111
2005-07-01 11:47:26	-175.4994	-39.323531	1268.355556

Figure 53: Site Report

#### 5.3.4.4 Spectrum Report

An example of a spectrum report is shown in Figure 54.

The reported attributes are: reflectances as a spectral plot, disabled flag as checkbox, spectrum number (auto-number by ASD), pathname, ASD comments (as entered in the capturing software on the field laptop).

The graphics output is relatively computing intensive. One should be careful not to start a report with too many (several hundred) resulting rows as the report can run for a while (e.g. 45 seconds for 1100 spectra). Note that the spectral resolution of the graphic is reduced (decimated by factor 10) in order to speed up plotting time. I.e. the data points are spaced at 10 nm intervals. This may result in some noise being less accentuated than in the raw data.

The 'disabled' checkbox offers some interaction with the database. If checked it will set the field 'disabled' in the spectrum table to TRUE. Likewise, unchecking sets the field to FALSE. Setting this field has a direct impact on the processing by SpectraProc. Disabled (i.e. disable = TRUE) spectra will not be included in any processing. It is therefore a chance to remove outliers from the data while still retaining all data in the database. An example illustrating this functionality is included in the tutorial (see 6.6).

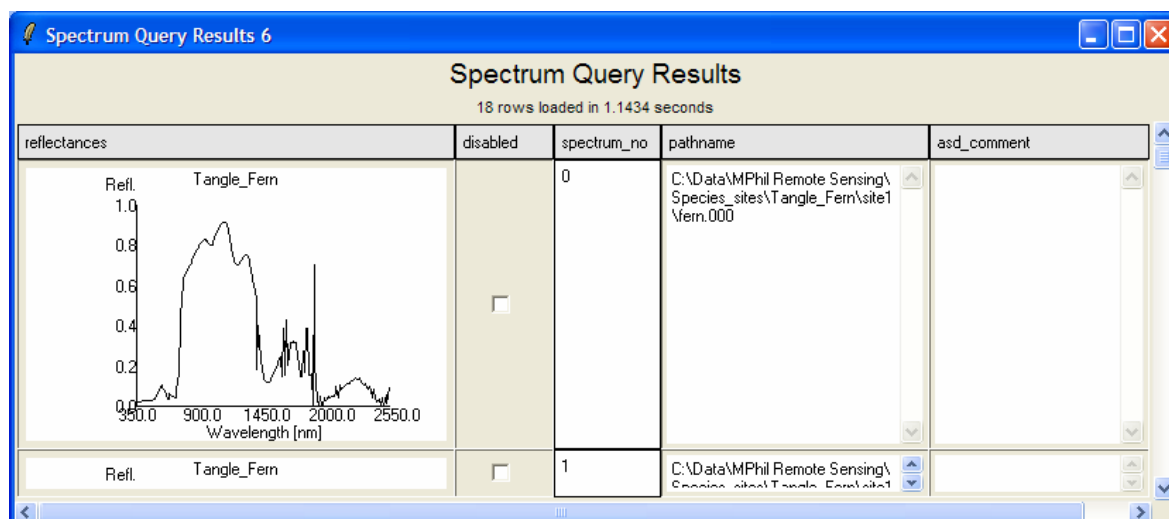


Figure 54: Spectrum Report

## 6 Tutorial

### 6.1 Overview

The tutorial is based on an example dataset comprised of three plant species. The folder containing the species folders is called 'Vegetation\_example'.

### 6.2 Examine the Folder and File Structure

Open the folder 'Vegetation\_example'. It contains three species folders: Blackfern, Cabbage tree and Lemonwood. Open each of these species folders and examine the contents of the site directories contained in them. Blackfern has only one sample site while Lemonwood and Cabbage tree have two. Also browse inside the site directories to find the ASD binary files.

### 6.3 Creating a new Study and Loading the Spectra

Create a new study as detailed in 5.2.2. As study name type 'your-name\_example', e.g. 'Mikes\_example'. The 'Species directory path' is in this case the path pointing to the folder named 'Vegetation\_example'. Set the number of samples per species for library inclusion to 10. Once the new study is created it will be automatically selected as the current study (otherwise select it (see 5.2.5)) and import the spectra (see 5.2.6).

### 6.4 Get to Know Your Data

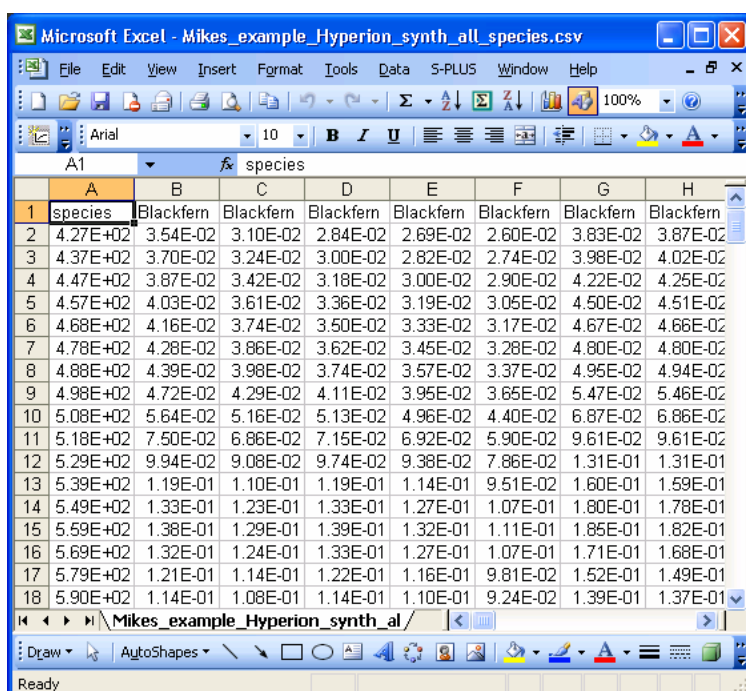
Check the settings of the processing chain:

- ☐ Smoothing filter: none
- ☐ Current Sensor: Hyperion
- ☐ Derivative: 0

Use the 'Export' function (see 5.2.8) to export the Hyperion synthesized spectra to a CSV file:

- ☐ Processing Stage: Synthesized
- ☐ Data Details: All Spectra
- ☐ Species Restriction: Include all species
- ☐ Names: Folder
- ☐ Transpose: Ticked
- ☐ Format: csv

This will create a file in the SpectraProc output directory. The output file name is shown in the message window. Open this file by double-clicking which should load it directly into Excel. The columns displayed should be similar to the example shown in Figure 55. Column A contains the average wavelengths of the Hyperion bands. The following columns contain the spectral data for each spectrum. The column headers are the species folder names.



	A	B	C	D	E	F	G	H
1	species	Blackfern	Blackfern	Blackfern	Blackfern	Blackfern	Blackfern	Blackfern
2		4.27E+02	3.54E-02	3.10E-02	2.84E-02	2.69E-02	2.60E-02	3.83E-02
3		4.37E+02	3.70E-02	3.24E-02	3.00E-02	2.82E-02	2.74E-02	3.98E-02
4		4.47E+02	3.87E-02	3.42E-02	3.18E-02	3.00E-02	2.90E-02	4.22E-02
5		4.57E+02	4.03E-02	3.61E-02	3.36E-02	3.19E-02	3.05E-02	4.50E-02
6		4.68E+02	4.16E-02	3.74E-02	3.50E-02	3.33E-02	3.17E-02	4.67E-02
7		4.78E+02	4.28E-02	3.86E-02	3.62E-02	3.45E-02	3.28E-02	4.80E-02
8		4.88E+02	4.39E-02	3.98E-02	3.74E-02	3.57E-02	3.37E-02	4.95E-02
9		4.98E+02	4.72E-02	4.29E-02	4.11E-02	3.95E-02	3.65E-02	5.47E-02
10		5.08E+02	5.64E-02	5.16E-02	5.13E-02	4.96E-02	4.40E-02	6.87E-02
11		5.18E+02	7.50E-02	6.86E-02	7.15E-02	6.92E-02	5.90E-02	9.61E-02
12		5.29E+02	9.94E-02	9.08E-02	9.74E-02	9.38E-02	7.86E-02	1.31E-01
13		5.39E+02	1.19E-01	1.10E-01	1.19E-01	1.14E-01	9.51E-02	1.60E-01
14		5.49E+02	1.33E-01	1.23E-01	1.33E-01	1.27E-01	1.07E-01	1.80E-01
15		5.59E+02	1.38E-01	1.29E-01	1.39E-01	1.32E-01	1.11E-01	1.85E-01
16		5.69E+02	1.32E-01	1.24E-01	1.33E-01	1.27E-01	1.07E-01	1.71E-01
17		5.79E+02	1.21E-01	1.14E-01	1.22E-01	1.16E-01	9.81E-02	1.52E-01
18		5.90E+02	1.14E-01	1.08E-01	1.14E-01	1.10E-01	9.24E-02	1.39E-01

Figure 55: Example of a SpectraProc CSV output

Create an X-Y plot. The result should be similar to the one shown in Figure 56. Maybe surprisingly the plot shows not the expected spectra but some noise that reaches values of up to 140. To see the vegetation spectra rescale the Y axis to a maximum of 1.0 (see Figure 57).

The noise is caused by atmospheric absorption (Thenkabail et al., 2004). As first step this waterband noise should be removed from the data before any further processing or analysis is carried out.

To remove the noise affected regions from the spectra open the waveband filter dialog (see 5.2.7.1) and click on 'Add default ranges'. This adds three wavelength regions that are known to contain waterband noise.

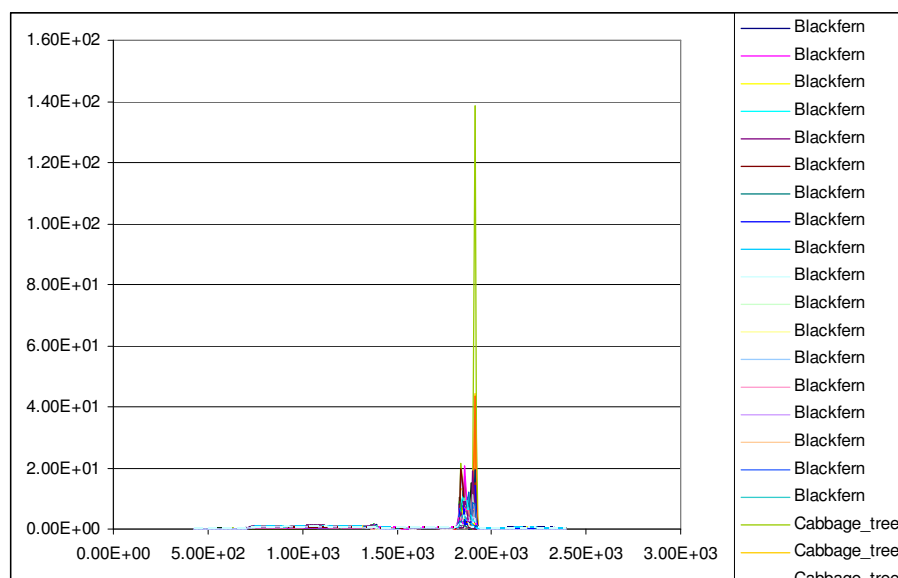


Figure 56: A plot showing mainly noise

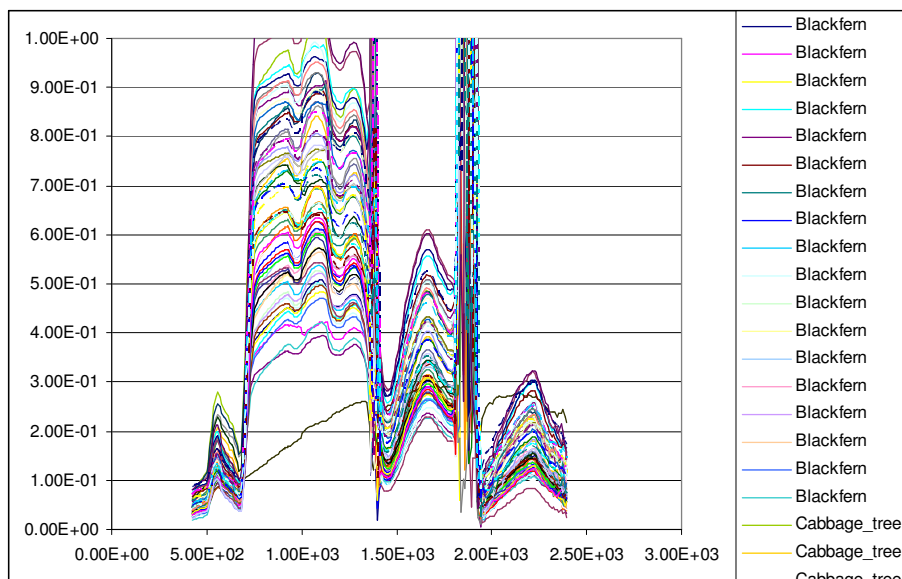


Figure 57: The same plot as above but rescaled to a reflectance of 1.0

Once the waveband filter is setup, repeat the export function (close the previously written file first in Excel) and load it again into Excel and plot it. The plot will look similar to the one shown in Figure 58. Note that now the noise regions are plotted as straight line segments. For visualization purposes this should be avoided as it suggests that data exists in the filtered regions.

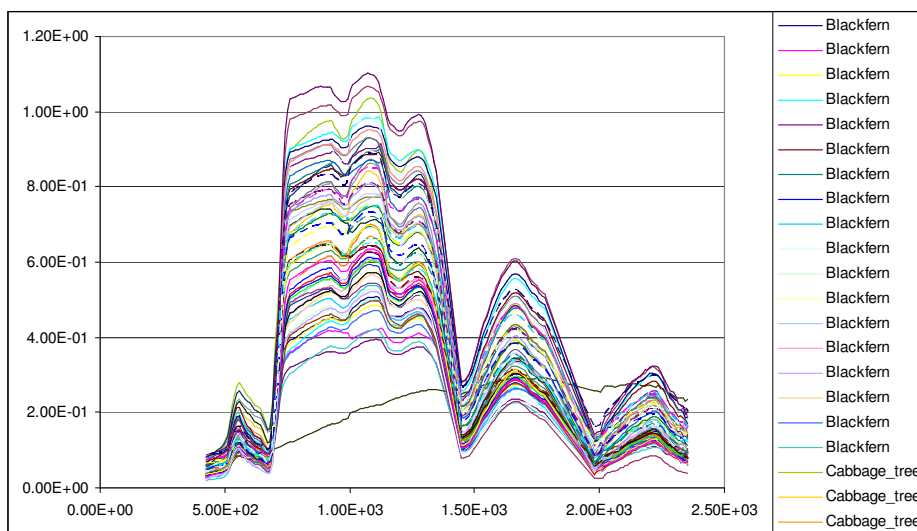


Figure 58: Spectra after setting up the waveband filter regions

To plot the filtered regions as gaps, select the option 'Include filtered bands' in the file export dialog. If the data is plotted again the result will be similar to the one shown in Figure 59.

Two other points are noteworthy: (a) some reflectance values are higher than one and (b) one Cabbage tree spectrum seems to be an outlier.

The reflectances higher than one could be the effect of changing illumination conditions during the sampling or a result of the BRDF.

The outlier problem can currently not be solved automatically. Two options exist: (a) remove the concerned spectrum from the file system and then delete, recreate and reload the study or (b) re-

move the concerned spectrum from the database using SQL commands. Future versions of the software may feature automatic outlier detection and elimination.

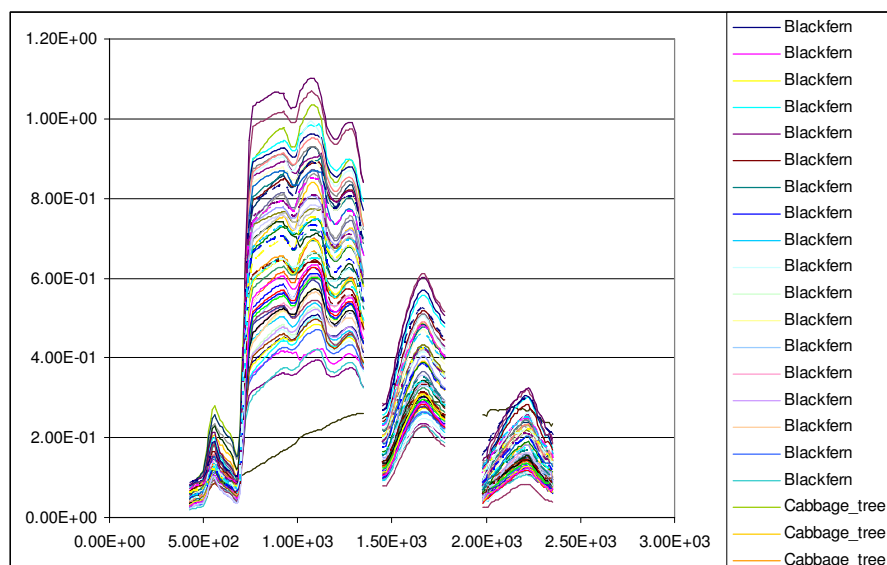


Figure 59: Waveband filtered data export with the 'include filtered bands' option

To generate stacked plots select the option 'Offset by [%]' in the file export dialog and set the percentage to 100. Then plot the data again. The spectra of each species are offset by the same amount (see Figure 60). If only the species mean is selected, the resulting plot is similar to the ones shown in diverse journal articles (see Figure 61). Note that when creating graphs in these cases the Y axis scale should not displayed as it makes physically not much sense (reflectances bigger than 1).

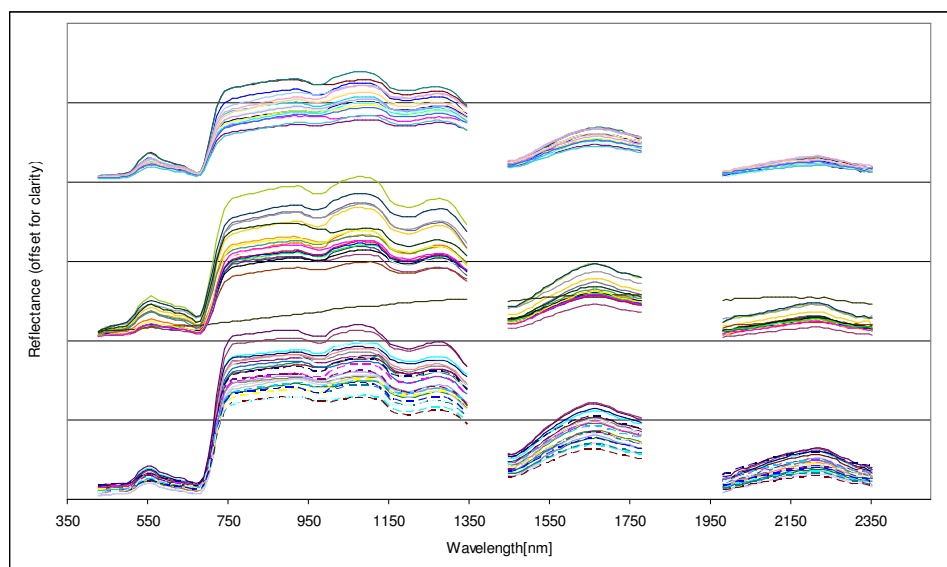


Figure 60: Stacked plots offset by 100% showing all spectra per species

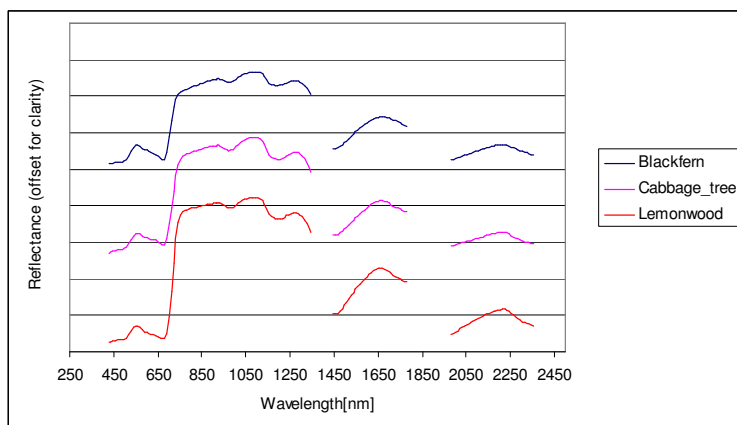


Figure 61: Stacked plot showing the mean per species

## 6.5 Discriminant Analysis in PCT Feature Space

Discriminant Analysis (DA) is carried out in feature spaces. As an example the data will be transformed by a principal component transformation (PCT). The number of components used in the transformation has a direct impact on the classification accuracy achieved by DA as will be demonstrated hereafter.

Select 'PCT' as the current feature space type. Note that the 'Feature Spaces' dropdown listbox states that there exist no feature spaces of this type (see Figure 62). A PCT is only possible if a Principal Components Analysis has been carried out. Perform PCA (see 5.2.12.3 for details).

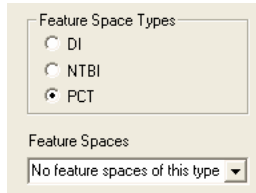


Figure 62: PCT feature spaces

Open the PCA output file in Excel and plot the first 10 eigenvalues against the component number. The result should be similar than the one showed in Figure 63. This graph is called 'screeplot'. The sharper the drop off in eigenvalues the better are the results that can be expected from the following PCT.

Note that once the PCA has been performed the 'Feature Spaces' dropdown listbox contains a selection of feature spaces. The numbers at the end of the names are equal to the number of components used in the feature space. E.g. PCT\_10 will effect a PCT using the first 10 components of the transformation matrix, i.e. the data will be reduced to 10 variables (dimensions).

Select 'PCT\_5' as feature space.

The discriminating power of feature spaces can be assessed by carrying out a discriminant analysis. Before the DA can be executed the library for the current processing chain settings must be built (see 5.2.11 how to build libraries). Then subject the data to DA (see 5.2.12.2 for details). The resulting overall accuracy should be around 90.7%. The DA results are detailed in the DA output file 'Error\_Matrix.csv' (see Table 6). The columns are the species that were classified. The number of correctly classified spectra is contained in the diagonal elements. The errors of omission are written in the off-diagonal elements. E.g. 6 Blackfern spectra were wrongly classified as Cabbage tree. Close the DA output file, select PCT\_10 as feature space and repeat the process of library building and DA and inspect the DA output file once more. The classification accuracy should have risen to 100%.

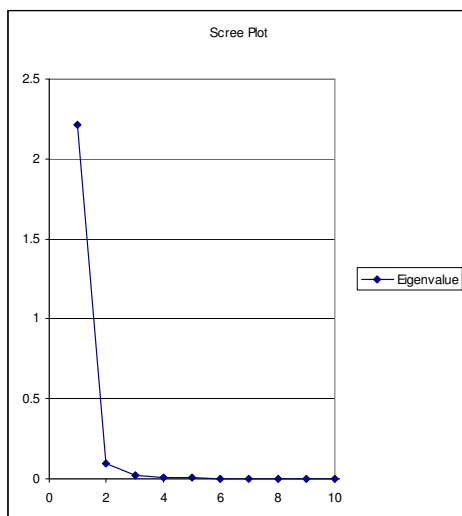


Figure 63: Scree plot of the eigenvalues of the first 10 components

Table 6: Error Matrix for DA on PCT\_5 feature space

	Blackfern	Cabbage_tree	Lemonwood	Row Total	User Accuracy
Blackfern	12	0	0	12	1.00E+02
Cabbage_tree	6	20	0	26	7.69E+01
Lemonwood	0	0	27	27	1.00E+02
Column Total	18	20	27	65	
Producer Accuracy	6.67E+01	1.00E+02	1.00E+02		9.08E+01

## 6.6 Disabling Outliers

By combining SpectraProc and SpectraProc Query, outliers can be identified and disabled. In the example dataset you might have spotted one cabbage tree spectrum that looks different (low reflection in the NIR) than all others. If PCA is run on this data set, the outlier can also be spotted (see Figure 64).

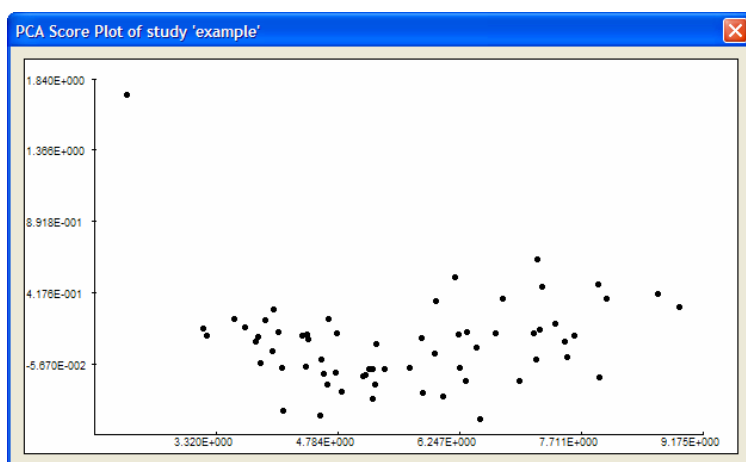


Figure 64: Score Plot showing an outlier



Now start SpectraProc Query, select the 'example' campaign in the Campaign Conditions, select 'Cabbage Tree' as Species Condition, select 'Spectrum' as report level and click 'Exec Query' (see Figure 65).

The screenshot shows the 'spectra\_proc\_query' window with the following settings:

- Campaign:** 'example' (selected in the dropdown)
- Species:** 'Cabbage\_tree' (selected in the dropdown)
- Collection Dates:** From: [ ] To: [ ]
- Spatial Attributes:** Altitude(m): [ ] <-> [ ]
- Report Level:** ☐ Campaign ☐ Species ☐ Site ☒ Spectrum
- SQL Query (autobuilt):**

```
SELECT COUNT(distinct spectrum.spectrum_id) FROM study, species, spectrum WHERE name = 'example' AND folder_name = 'Cabbage_tree' AND study.study_id=species.study_id AND study.study_id=spectrum.study_id AND species.species_id=spectrum.species_id
```
- Result Set Information:** Number of resulting rows by current query: 20
- Buttons:** 'Exec Query' and 'Reset'

Figure 65: Conditions set to select Cabbage Tree spectra from the 'example' campaign

In the spectrum report (see Figure 66) scroll down till you see the wrong looking cabbage tree spectrum (spectrum number 9 of site 1, you can deduce the site number from the pathname). Tick the checkbox in order to disable the spectrum. Now rerun the PCA calculation in SpectraProc and the outlier will vanish from the Score plot (see Figure 67).

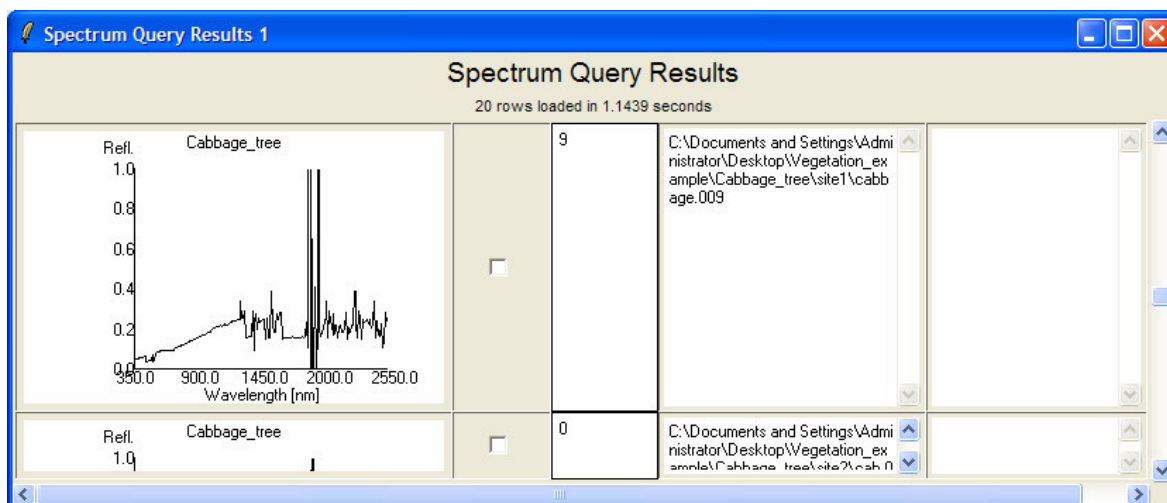


Figure 66: Spectrum Report showing a wrong looking Cabbage Tree spectrum

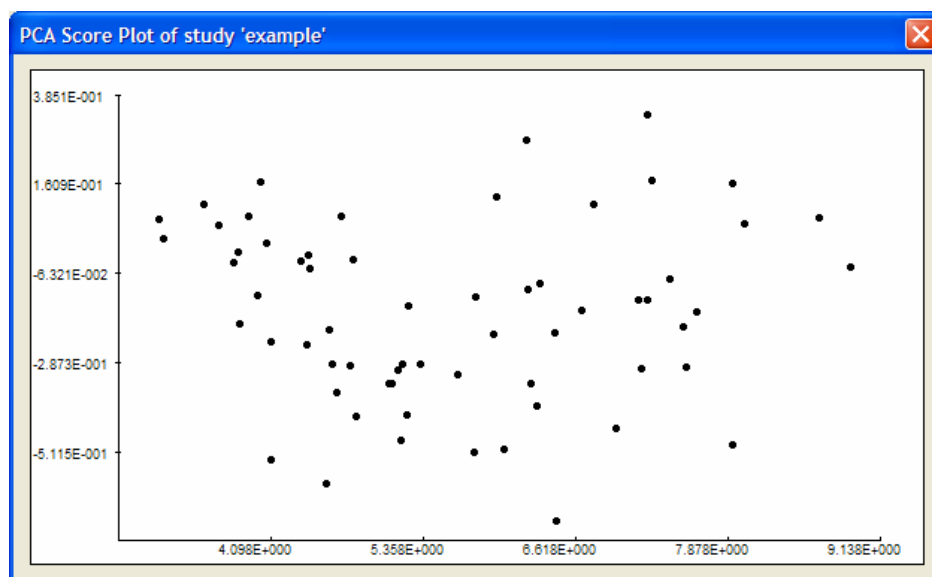


Figure 67: Score Plot after disabling an outlier

To assess the impact of an outlier on the building of covariances and thus on the classification result build a library using a PCT feature space of low dimensionality, e.g. 2, and run 'Discriminant Analysis'. Repeat library building and discriminant analysis after re-enabling the outlier in SpectraProc Query. You should see a significant change in classification accuracy:

-----  
Principal Component Analysis

PCA output written to file:

C:\Data\MPhil Remote Sensing\SpectraProc\_output\pca\_output.csv

-----  
New library built and reloaded

-----  
Discriminant Analysis

Error matrix written to file:

C:\Data\MPhil Remote Sensing\SpectraProc\_output\Error Matrix.csv

-----  
**Overall Accuracy: 70.31250**

-----  
Principal Component Analysis

PCA output written to file:

C:\Data\MPhil Remote Sensing\SpectraProc\_output\pca\_output.csv

-----  
New library built and reloaded

-----  
Discriminant Analysis

Error matrix written to file:

C:\Data\MPhil Remote Sensing\SpectraProc\_output\Error Matrix.csv

-----  
**Overall Accuracy: 47.69231**

## 7 Change History

### 7.1 SpectraProc

#### 7.1.1 Added Features

SpectraProc Version	Feature	Date
V1.1	Eigenvectors are written into the PCA output file	13.4.2006
	When creating a new study the new study is automatically selected as the active study	24.4.2006
	Study information report	26.4.2006
	PCA Score Plot shown when performing PCA	10.5.2006
	The contents of the report window can be saved to a file by selecting: File -> Save Report Window Content. The file is written to the default output directory and named report_window_output.txt	20.10.2006
	Disabled spectra are not utilized in the calculations (disabling spectra with the SpectraProc Query tool)	20.10.2006
	Depending on the sensor dimensionality, only feature spaces with less or equal number of dimensions are selectable in the GUI. E.g. DGVIs (dimension 10) cannot be built if Landsat data is synthesized.	20.10.2006
	PCA can now be carried out on multispectral sensors like Landsat and Cropcircle (ACS210)	20.10.2006

#### 7.1.2 Fixed Bugs

SpectraProc Version	Problem	Fix	Date
V1.1	Mixtures not deleted when deleting studies.	Include mixtures in the delete study function	12.4.06
	Crash when trying to write PCA output and file is open	Check for open files when writing PCA output file	13.4.2006
	Feature space list not updated when derivative settings change	Update feature space list when changing derivative calculation	18.4.2006
	ARFF file output produces corrupt files	Attributes list fixed	18.4.2006
	PCA crashes when no data is available	Check if data exists before starting the eigenanalysis	24.4.2006
	PCA on Landsat7: crashes because 1 <sup>st</sup> feature space is PCT_10 (cant build matrix)	Limit the dimensionality of the feature spaces to smaller or equal the number of bands of the sensor.	20.10.2006

### 7.1.3 Known Bugs

<b>SpectraProc Version</b>	<b>Problem</b>	<b>Work around</b>
V1.1	The application will crash when building a library using a feature space that requires certain wave-lengths on sensors with not enough band information. Can occur when e.g. redefining the DGVI's to have only 5 dimensions and then try to calculate DGVI's from Landsat data.	Just avoid doing this. There is a logic explanation why this cannot be done. Implementing a check to disable such user actions is too time consuming at this point.
	The application can sometimes just refuse to load. This is some effect of the Symantec Internet security (SpectraProc not being in the list of programs that are allowed to access the database port), probably coupled with hibernating the machine.	A reboot of the system helps here.

## 7.2 SpectraProc Database

### 7.2.1 Added Features

<b>SpectraProc Database Version</b>	<b>Feature</b>	<b>Date</b>
V1.1	Added 'disabled' field to spectrum table	20.10.2006
	Removed the 'not null' constraint and added a default value of null for the BLOBS in: pca_data, spectrum and statistic	

## 8 References

- Bojinski, S., Schaepman, M., Schlaepfer, D. & Itten, K. (2003). "SPECCHIO: a spectrum database for remote sensing applications." Computers & Geosciences **29**: 27-38.
- Elvidge, C. D. & Chen, Z. (1995). "Comparison of broadband and narrowband red and near-infrared vegetation indices." Remote Sensing of Environment **54**: 38-48.
- Fliege, N. J. (1994). Multirate Digital Signal Processing. Chichester, John Wiley & Sons.
- Keshava, N. & Mustard, J. F. (2002). "Spectral Unmixing." IEEE Signal Processing Magazine **19**(1): 44-57.
- Landgrebe, D. (1997). On Information Extraction Principles for Hyperspectral Data, Purdue University.
- Papula, L. (1994). Mathematik fuer Ingenieure und Naturwissenschaftler, Viewegs.
- Richards, J. A. (1993). Remote Sensing Digital Image Analysis. Berlin, Springer Verlag.
- Savitzky, A. & Golay, M. J. E. (1964). "Smoothing and Differentiation of Data by Simplified Least Squares Procedures." Analytical Chemistry **36**(8): 1627-1639.
- Shaw, G. & Manolakis, D. (2002). "Signal Processing for Hyperspectral Image Exploitation." IEEE Signal Processing Magazine **19**(1): 12-16.
- Thenkabail, P. S., Enclona, E. A. & Ashton, M. S. (2004). "Accuracy assessment of hyperspectral waveband performance for vegetation analysis applications." Remote Sensing of Environment **91**: 354-376.
- University of Waikato (2005). WEKA. <http://www.cs.waikato.ac.nz/~ml/weka/>.
- Vane, G. & Goetz, A. F. H. (1988). "Terrestrial Imaging Spectroscopy." Remote Sensing of Environment **24**: 1-29.