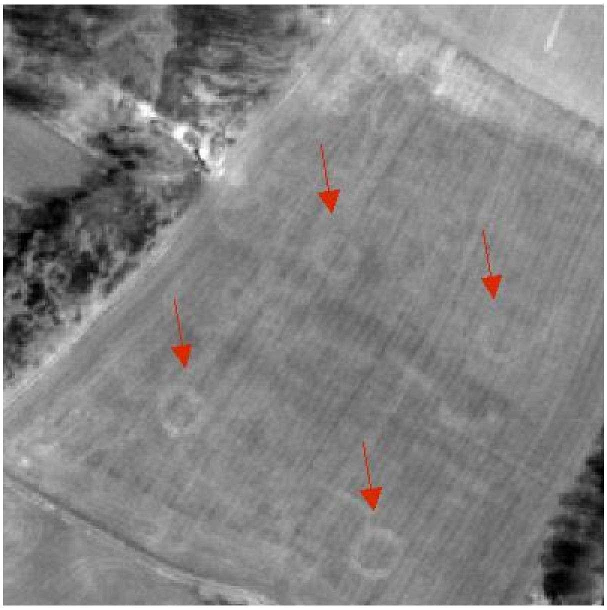
Norsk Regnesentral
NORWEGIAN COMPUTING CENTER

Note

CultSearcher

Computer-assisted detection of potential cultural heritage sites

Software Guide, version 2



Note no SAMBA/15/08

Authors
Jostein Amlien
Marit Holden
Siri Øyen Larsen
Øivind Due Trier
Rune Solberg

Date 16 May 2008

Norsk Regnesentral

Norsk Regnesentral (Norwegian Computing Center, NR) is a private, independent, non-profit foundation established in 1952. NR carries out contract research and development projects in the areas of information and communication technology and applied statistical modelling. The clients are a broad range of industrial, commercial and public service organizations in the national as well as the international market. Our scientific and technical capabilities are further developed in co-operation with The Research Council of Norway and key customers. The results of our projects may take the form of reports, software, prototypes, and short courses. A proof of the confidence and appreciation our clients have for us is given by the fact that most of our new contracts are signed with previous customers.

Norsk Regnesentral
Norwegian Computing Center
Postboks 114, Blindern
NO-0314 Oslo, Norway

Besøksadresse
Office address
Gaustadalléen 23
NO-0373 Oslo, Norway

Telefon · telephone
(+47) 22 85 25 00
Telefaks · telefax
(+47) 22 69 76 60

Internett · internet
www.nr.no
E-post · e-mail
nr@nr.no

Title	**CultSearcher – Software Guide, version 2**
Authors	**Jostein Amlien, Marit Holden, Siri Øyen Larsen, Rune Solberg, Øivind Due Trier**
Date	16 May 2008
Year	2008
Publication number	SAMBA/15/08

Abstract

This report documents the CultSearcher software system for computer-assisted detection of potential cultural heritage sites. Such sites can be visible in agricultural fields as soil or crop marks. The system is a prototype software package for assisting archaeologists in the process of scanning satellite images for potential cultural heritage sites. The prototype system is developed in the IDL/ENVI software. It is operated through a simple graphical user interface and runs under Windows and Linux.

The operation of the system is based on three central processing steps: segmentation, feature extraction and classification. The current version of the system includes detection of amorphous soil marks as well as ring formed crop marks. In both cases the process finalizes by interactively presenting the identified potential cultural heritage sites for the archaeologist, one by one.

Keywords	Quickbird images, cultural heritage sites, soil marks, crop marks, ring graves, segmentation, feature extraction, classification
Target group	Operators and developers, archaeologists
Availability	Open
Project number	236 067
Research field	Earth Observation
Number of pages	42

Contents

1 Introduction

1.1 Scope

The purpose of the software prototype described in this report is to assist archaeologists in searching for potential cultural heritage sites (soil marks and crop marks) in satellite images. The specification of this software, as well as a description of the first and second prototype version, is the subject of separate reports (see [1], [2] and [3]). The third and following versions of the prototype system are described in this report.

1.2 General background

The increasingly intensive use and modification of the landscape resulting from modern demands for efficient infrastructure and land use (agricultural production, mining, energy sources, leisure/tourism facilities etc.) exerts growing pressure on cultural heritage in the landscape.

Given the large costs of surveying the tracts in question by traditional field work, alternatives must be sought. One possible approach is through the use of satellite images. Current experience with the visual analysis of these images indicates that cultural heritage sites can be observed in such images, even those without any apparent above ground manifestations. The costs of acquiring and analyzing such images are not negligible, and such an approach can never entirely replace traditional field work. Nevertheless, the use of satellite images has a large potential as a supplement to the traditional methods that will be very useful in the detection of cultural heritage sites on a regional and national scale.

In recognition of this, a project was started in 2004 by the Norwegian Directorate for Cultural Heritage (in Norwegian, *Riksantikvaren*, RA), in collaboration with the Norwegian Computing Center (*Norsk Regnesentral*, NR), the Norwegian Institute for Cultural Heritage Research (*Norsk Institutt for Kulturminneforskning*, NIKU), the Museum of Cultural History at the University of Oslo (*Kulturhistorisk Museum*), Vestfold County Administration (*Vestfold fylkeskommune*) and the Norwegian Space Centre (*Norsk Romsenter*). The project aims at developing a cost-effective method for surveying and monitoring cultural heritage sites on a regional and national scale. In this project, NR has been responsible for developing the automatic detection methodology and implementing this into a prototype software system.

The aim of the software prototype described in this report is to provide computerized assistance to the operator in the analysis of satellite images. In particular, the software identifies possible sites for further inspection by an archaeologist. This means that the archaeologist may concentrate on analyzing the identified sites rather than the entire image.

1.3 Overview

This document will briefly describe the software prototype through the following sections:

Section 1	Gives an introduction to why a system for detecting potential cultural heritage sites might be useful.
Section 2	Gives a system overview.
Section 3	Contains a user guide.
Section 4	Describes the software in detail for the system developer.
Appendix	Gives examples of text files used by the system.

2 System overview

2.1 Purpose

The software is intended as an aid to archaeologists in their search for cultural heritage sites by identifying potential sites in satellite images, visible as soil and crop marks.

2.2 Software and hardware requirements

The software is based on ENVI/IDL, and requires ENVI 4.4 and IDL 6.4. The software will run on both Windows and Linux platforms.

2.3 Background and methods

In a satellite image, cultural heritage sites may be visible as areas with radiometric properties that differ from their immediate surroundings. There can be many possible reasons for the varying properties, among them different conditions for plant growth. Cultural heritage sites may thus show up as patches that are darker or brighter than their surroundings. This is exploited in the methods used to automatically search for such patches in the satellite images. Such paths may be amorphous, or they may a specific form, like rings.

The methods currently used to search for potential cultural heritage sites are performed in three main steps: *Segmentation, feature extraction* and *classification*. In the first step potentially interesting locations are extracted as image segments, in the second step characteristics of these segments are computed, before the last step undertakes a classification of the various segments by comparing them to inherent class descriptors. Before these main steps are performed, the images have to be imported, and regions/areas of interest must be identified. After the main steps, the results need to be controlled. In addition, the system contains functionality for interactive training the system in recognizing the various classes. All these have been defined as separate processing steps, as described in the next section.

2.4 User interface

The software has a simple graphical user interface (GUI) where the user may choose between doing the processing step by step in an interactive manner or start an automatic process performing all the main steps. In addition, the GUI gives access to the specific training steps. The user will not need to specify a lot of parameters to the analysis, just a few setup parameters are defined in the Interactive GUI. For each step in the interactive mode the user also needs to enter or confirm the file names. Other steps, like *Edit result*, *Mark* and *Create mask*, require more user interaction.

The graphical user interface is realized through the use of ENVI/IDL widgets and menus. The main menu for the CultSearcher software is made available together with the interactive software tools provided by ENVI, enabling easy use of the ENVI utilities. The main menus and the structure of the GUI are illustrated below. Information on how to use ENVI is not included here, but can be found in the user manuals for ENVI.

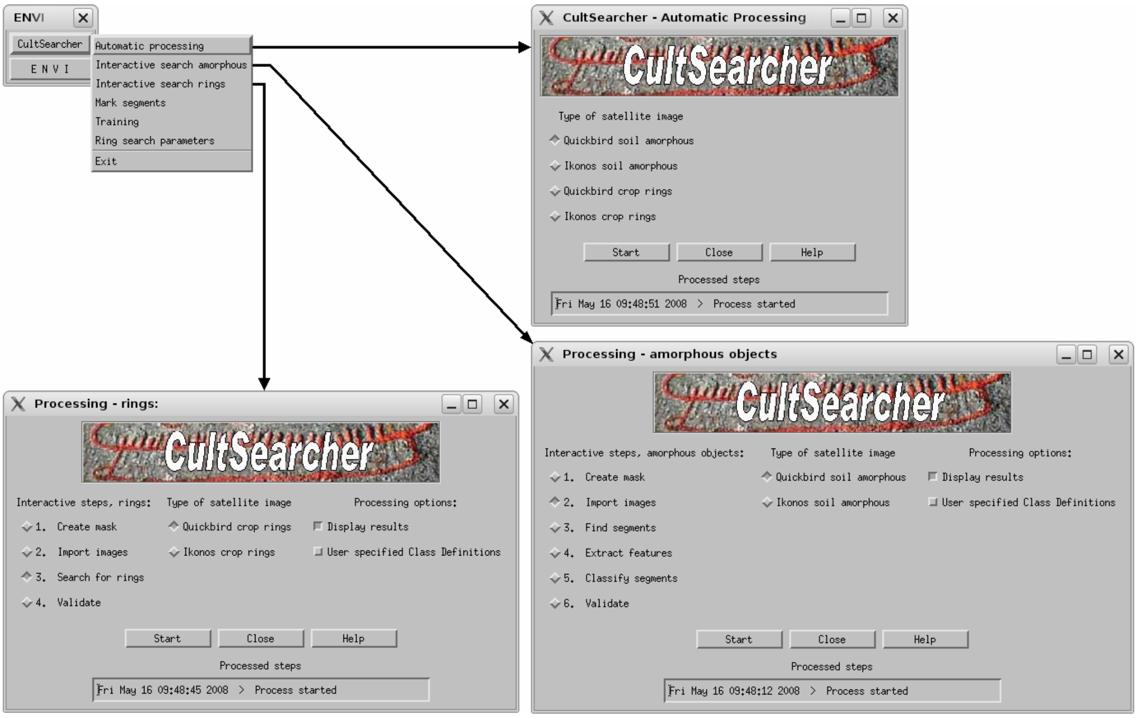


Figure 1. Overview of the main menu and dialogs.

2.5 Overall system design and main modules

The system is operated through a simple graphical user interface (GUI) that provides the user with two modes for running the system; (1) a fully automatic mode and (2) an interactive, stepwise mode, providing the user some more control. The GUIs as well as the processing steps are separate modules in the software system. They are shortly described below

There are four main GUIs (Figure 1). These are:

CultSearcher menu – This menu allows the user to open the main dialog for automatic or interactive processing. In the current version of the system the two training steps mark segments or training are also provided in this menu

Interactive mode, amorphous objects – This is the main dialog for the interactive mode for processing of amorphous objects. In this mode the user controls the processing chain as the user chooses which of the processing steps to run. This mode is useful when trying out various parameter settings, enabling the user to re-run specific steps and inspect the results.

Interactive mode,rings – This is the main dialog for the interactive mode for processing of rings. As above, for amorphous objects, the user controls the processing chain.

Automatic mode – This is the main dialog for the automatic mode. In this mode the system is run automatically, as the steps in the processing chain are run without user interaction. Note that the steps Create mask and Edit result are not included in this mode. These interactive steps should be run from the interactive mode.

There are six processing steps connected to the interactive and/or automatic modes:

Create mask This step defines which areas in the satellite image that will be analyzed. It is available in interactive mode only.

Import image This step extracts a spatial subset from a satellite image together with the corresponding mask making them ready for the analysis.

Segmentation This step identifies potentially interesting locations by performing a segmentation of the satellite image

Extract features This step extracts features or characteristics of the identified segments.

Classification This step analyzes the extracted features of each image segment in order to classify the segments for being or not being a cultural heritage site

Edit result This step is available in interactive mode only. The user will be led through the classified segments, and given the possibility to delete segments that are not believed to represent cultural heritage sites.

In addition there are two processing steps directly available from the *CultSearcher* menu. Together with the first steps in the interactive mode, they are needed when training the system. The training process is currently not implemented as a separate dialogue.

Mark segments This is an interactive process, where the operator is being led through the amorphous segments that were identified in the segmentation step. For each segment the user is asked to assign a class label.

Training This is an automatic process for updating the class descriptions that are used by the classification step. The updating is based on the features of image segments that have been labelled by the operator in the *Mark segments* step.

Please note that no training is required for rings in the current version of the system.

2.6 System modes

2.6.1 The interactive modes

The interactive mode includes all the main steps in the system:

- *Create mask*
- *Import images*
- *Segmentation*
- *Extract features*
- *Classification*
- *Edit results*

The specific training steps, *mark segments* and *training,* are described in a separate section.

The following files are in use:

Panchromatic satellite scene. This image file contains a panchromatic satellite scene. It is assumed that the image has been geo-referenced into some map projection. The image format does not need to be ENVI., but it is recommended. This image file is input to the steps *Create mask* and *Import images.*

Region-of-interest (ROI) file. This file is produced by the step *Create mask.* It is the result of a process where the operator imports a vector file and/or interactively edits the vectors in order to define a mask where the system should perform its analysis. This mask is saved as an ROI file. The ROI mask file will typically contain a number of agricultural fields in a local area. This file is input to the *Import images* step.

Panchromatic image. This file is produced by the step *Import images.* It is a (small) spatial subset of the panchromatic satellite scene, as given by the ROI file. The panchromatic image will be input to the steps *Segmentation* and *Extract features*.

Image mask. This is produced by the step *Import images.* It converts the ROI file into an image mask that corresponds to the imported panchromatic image. The image mask will be input to the step *Segmentation*.

Segment image. This file is produced by the step *Segmentation*. It contains a large number of dark and large segments that were identified as soil or crop marks in the panchromatic image inside the areas defined by the image mask. Depending on the settings the segments may be amorphous or ring-formed. . The segment image is input to the step *Extract features*.

Segment Index Map (object mask index). This image file is produced by the step *Extract features*. Each image segment has a unique label that identifies the image segment in the following files. The segment index map is input to the step *Classification*.

Feature Vector Base (FVB). This text file is produced by the step *Extract features*. For each segment the FVB will contain a reference to the segment index map together with a feature vector retrieved from the segment. The FVB is input to the step *Classification*.

Class Description Base (CDB). This text file is a system file that is used as input in the step *Classification.* The system will automatically select the CDB file according to the system settings. However, the operator may optionally choose to select the CBD manually

Class image. This file is produced by the step *Classification*. Each segment is labelled according to the result of the classification. Typically the segments are classified as noise or as potential cultural heritage sites.

Validated class image. This file is produced by the step *Edit result.* The operator makes the final selection of cultural heritage sites, based on the classification result and visual inspection of the satellite image. All segments that the operator has no confidence in being a cultural heritage sites should be deleted. The remaining segments are considered as potential sites for cultural heritage.

Validated ROI. This file is produced by the step *Edit result.* The outlines of the validated sites are stored in an ROI file.

NOTE that when searching for ring structures, this list will be different.

2.6.2 The automatic mode

The automatic mode includes these processing steps.

- *Import images*
- *Segmentation*
- *Extract features*
- *Classification*

The same files are used as in the interactive mode. They can all be inspected from ENVI after the automatic mode has finished. However, the operator only needs to consider these files (see description above):

- **Panchromatic satellite scene** (input to the automatic mode)
- **Region-of-interest (ROI) file.** (input to the automatic mode)
- **Class image**. (output of the automatic mode.)

2.6.3 Training the system

The aim of the training is to produce and refine a Class Description Base (CDB) such that it later can be applied in the classification. Note that training is not relevant for rings structures.

There is no complete 'training mode' in the current version of the system. However, when undertaking training, the operator is expected to run the interactive mode until the *Extract features* step has completed, and then call the specific training steps.

The specific training steps are *Mark* segments and *Training*. They are available from the *CultSearcher* menu, and are expected to be called after the *Extract features* step has completed. In future versions of the system, a complete training mode may be defined, offering all steps relevant for the training process.

The files involved in the specific training steps are as follow:

Segment Index Map. This image file is produced by the *Extract features* step and is used by the *Mark* steps for displaying the segments to the operator.

Segment Mark List. This text file is produced by the *Mark* step. For each segment it contains a reference to the segment index map and a used assigned class label. It is assumed that the segments identifiers in the segment index map correspond to the identifiers in the FVB. The Segment Mark List file is input to the *Training* step.

Feature Vector Base (FVB). This text file is produced by the *Extract features* step and is used as input to the *Training* step for retrieving statistics of the marked segments.

Class Description Base (CDB). This file will typically be updated by the *Training* step. The CDB file provides class-wise statistics on the relevant features, including mean vectors, standard deviation and co-variances. The statistics in the CDB file are updated by means of the features in the FVB. This means that a CDB file will in general be a result of the processing and marking of several image files.

3 User's Guide

3.1 System installation guide

3.1.1 The main components of a CultSearcher installation

3.1.1.1 Hardware

The user should dispose of a modern, powerful PC with a minimum of 1GB of memory and sufficient disk space for software and data.

3.1.1.2 Software

CultSearcher is a program intended for searching for cultural heritage sites in satellite images of agricultural fields. The program is written in the IDL language as an add-on to the ENVI menu system. In addition to the CultSearcher software, the user must have installed the IDL/ENVI software on the computer it should be run.

3.1.1.3 What is required of hardware and software

We have tested CultSearcher only under the Windows XP Professional operating system and recommend that the same system is employed at the user sites.

The current version of CultSearcher is much more efficient than the previous ones, and should therefore be expected to run satisfactory on any modern, powerful PC with sufficient memory.

3.1.2 Installing CultSearcher

Installing CultSearcher is a three step process, according to the following instructions:

1) The latest version of ENVI/IDL should be installed at each user site. Currently, the latest versions are ENVI 4.4 and IDL 6.4. This software package can be downloaded from the ITT website, following the instructions given there. The users will obtain license key, user name and password from Norsk Regnesentral (Norwegian Computing Center).
2) The installation of CultSearcher is carried out by extracting the CultSearcher.zip file in a suitable directory, e.g. C:\Program Files\ . A number of files and sub directories are then copied to the directory CultSearcher. The sub directory "Src" contains the program source files, as well as a compiled version. The sub directory "CDB" contains a CDB file. An empty sub directory "Data" is also created.
3) Finally the configuration script, i.e. the file configure.vbs (directly under the CultSearcher directory), must be run. This is done by a double click on the file, and the user will be taken through a number of steps in order to adapt CultSearcher to local requirements:
   - specification of the CultSearcher directory (i.e. "C:\ProgramFiles\CultSearcher\, if the suggestion in step 2 were followed).
   - specification of the IDL directory (the directory where IDL64 were put during the ENVI installation, typically C:\Program Files\ITT\ )

After installation, CultSearcher should appear on the menu when starting ENVI as normal.

3.1.3 Directory structure

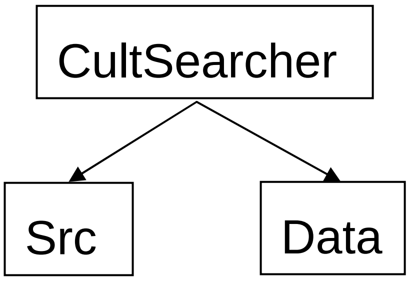


Figure 2. The directory structure.

The contents of the three directories (Figure 2) are as follows:

- **CultSearcher** contains text files, logos, scripts (start-up, make).
- **Src** contains IDL code for each module in the system (one file per module).
- **Data** contains example data.

A detailed explanation of the files in each directory is given in Section 4.

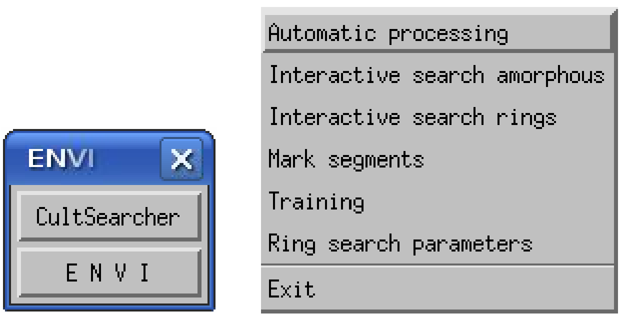
3.2 System operator's guide

The CultSearcher system has a simple graphical user interface (GUI) where the user may choose between (1) doing the processing step by step in an interactive manner, or (2) start an automatic process performing all the main steps. In addition, the GUI gives access to the specific training steps. In the current version, the user interaction is kept fairly simple. The user will not need to specify a lot of parameters to the analysis, but just a few setup parameters are defined in the Interactive GUI. For each step in the interactive mode the user also need to enter or confirm the file names. The interactive modules, like the *Edit result*, *Mark* and *Create mask* steps, will of course require more user interaction.

The graphical user interface is realized through the use of ENVI/IDL widgets and menus. The menu for the CultSearcher software is made available together with the extensive interactive software tools provided by ENVI, enabling easy use of the whole range of ENVI utilities. The main menus and the structure of the GUI are illustrated in the following sections. These illustrations are collected from a *linux* platform, but the graphics will have a similar, though slightly different, appearance under Windows. Information on how to use ENVI itself is not included here, but can be found in the user manuals for ENVI.

3.2.1 Starting the software

The CultSearcher application is started from the ENVI icon on the Windows desktop, or by typing the command `./cult_setup` on a *linux* platform When ENVI starts, CultSearcher will become available from the ENVI start menu (Figure 3a).



(a) (b)

Figure 3. Menus. (a) ENVI startup menu. (b) The CultSearcher pull down menu

The “ENVI” button gives full access to the ENVI functionality. The “CultSearcher” button will trigger the CultSearcher pull-down menu (Figure 3b). The operator may choose to run the CultSearcher system in automatic or two different interactive modes.

In the interactive modes, each step in the process can be run separately, while in the automatic mode all the main steps are run in sequence. In addition, specific training steps are also available.

The details on how the software should be used in the various modes are described in the next sections.

3.2.1.1 Display of images

When an image is displayed, three windows will be opened: A large window showing the image in full resolution, a smaller zoom window, and if necessary, a scroll window showing the entire image (Figure 4).

Image files can also be opened for display using options from the ENVI menu. When an image file is chosen a menu with available bands will be opened. From this the bands to be displayed can be selected. (For more details on the ENVI functionality, see the ENV/IDL manuals).

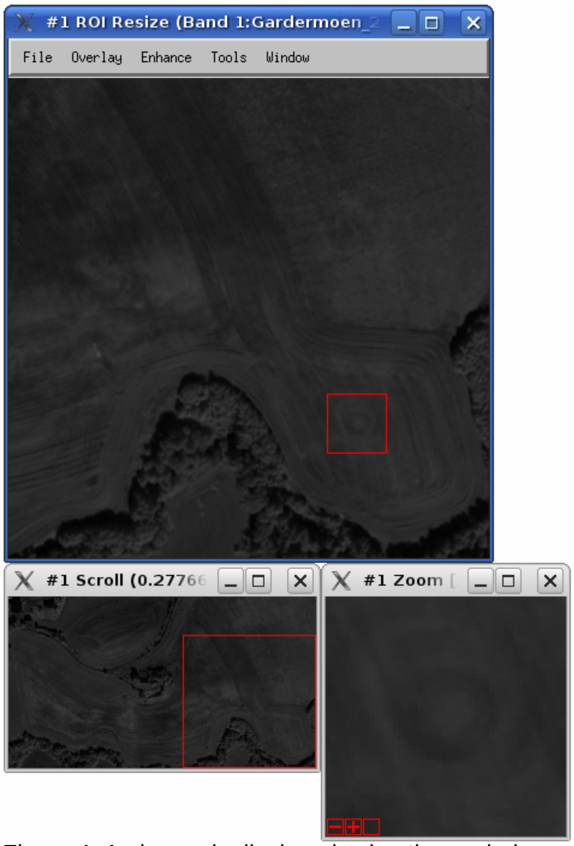


Figure 4. An image is displayed using three windows: a main window at the top, an overview window below left, and a zoom window below right

3.2.2 Processing settings

The search for cultural heritage sites should adapt its approach and choice of methods to the setting of the analysis. The setting includes the type of satellite image, type of marks, and object shape. The setting should be decided by the operator while running the system, in order to select an appropriate approach

The current version of the system recognizes combinations of these settings:

- Type of satellite scene: Ikonos / Quickbird
- Type of marks to look for: crop marks / soil marks. This will depend on the time of the year (soil marks in the spring and crop marks in the summer)

It is not recommended to change the options or settings after the import step.

3.2.3 Processing options

The operator may also set a few options for tailoring the user interaction. Each of these options are either on or off

- *Display results*: Setting this option will complete each step by displaying its result

- *User specified class definitions:* Setting this option will allow the operator to specify their own *class description base*, or CDB, in the *"classify segments"* step. This presumes that an alternative CDB exists, that is, that training has been done.

It is not recommended to change the options or settings after the import step.

3.2.4 Interactive processing modes

In the interactive modes, the execution of individual steps can be controlled by the operator. The steps will typically be run one by one, where input to one step is the output from the previous step. The sequence is fully controlled by the operator, by choosing the desired step in the left pane. The processing settings and options are controlled by the buttons in the middle and right panes. When the start button is pressed, the selected step will be started with the specified settings and options. In the following we will describe the interaction needed for each processing step.

3.2.5 Interactive processing of amorphous objects

Figure 5. The main dialog for interactive processing of amorphous objects

The main steps of the processing of amorphous objects (Figure 5) are:

1. *Create mask*. The areas to be analyzed are identified based on an existing vector file, resulting in a set of regions-of-interest (ROI).
2. *Import image*. The areas to be analysed are retrieved from the satellite scene by means of the ROIs from step 1. The image format is converted to ENVI image file format.
3. *Find segments* (segmentation). Potentially interesting locations within the areas defined by the ROIs are identified by means of image segmentation.
4. *Feature extraction.* The identified locations are described in term of features that are related to colour, shape and size of the segments
5. *Classify segments*. The features that are characterising the segments are used in order to classify the segments into a number of classes that separates potentially interesting locations from less interesting ones.
6. *Edit result.* The final set of potentially interesting cultural heritage locations is finally identified in an interactive step where the each segment is inspected manually.

3.2.5.1 Create mask

The purpose of this step is to convert some vector file to an ROI file, which can be used by the system. An ROI file is a special type of a vector file that is used by ENVI. Note that an ROI file is always associated to an image file, in this case the input satellite scene. The ROI file defines a spatial subset of the satellite scene. Only this subset will be analyzed in the subsequent steps. It is recommended that create mask is performed several times in order to define spatial subsets of appropriate sizes.

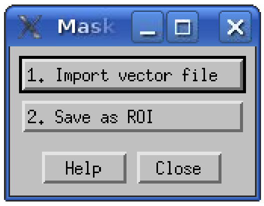


Figure 6. Dialog for create mask.

This processing step is an interactive step. It will be started by displaying the mask menu (
Figure 6). The menu offers two choices, which should be run in sequence.

When clicking the first button, "*1. Import vector fil*e", a file selection menu will be displayed. You will be asked to specify a vector file, and the filter is set to ".evf", but if you wish you may change this to ".shp", depending on the format of your vector file. The specified vector file should contain vectors enclosing the fields (areas) to be analyzed.

If the selected file is a shape file, the *import vector files parameters* dialog (Figure 7) will be displayed. Here, the name of the file you have just specified will appear at the top of the window under "Selected files". You now need to specify the following:

- The "*Native File Projection*" (UTM)
- The "*Datum*" (WGS-84)
- The "*Zone*" (32)

The values in the parentheses correspond to the map projection parameters common for South Eastern Norway.

When these have been specified, press "OK" to continue.

If the selected file is an ".evf"-file, you do not need to set the vector file parameters as for the shape file, and the program will proceed to the next step.

An "*Available Vector List*" will appear with the name of the original vector file. Select the filename in the list, and then click the "*File*" button in the upper left corner. From the pull-down menu that appears, select the option: "*Export layers to ROI*". A new window with the title "*Select Data File to Associate with new ROIs*" will then appear. You will here need to specify the image that you are going to analyze.

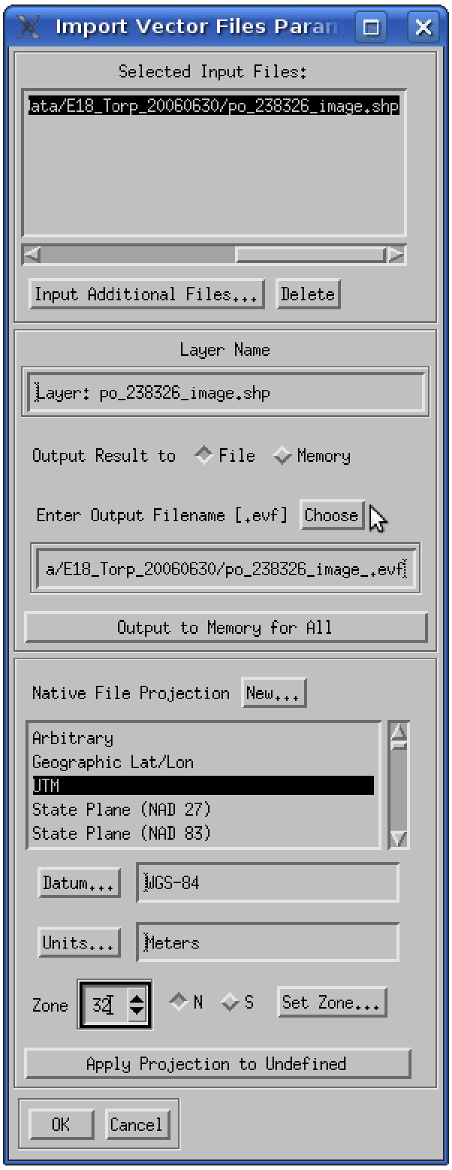


Figure 7. Dialog for specification of input shape file for the mask creation.

Click Open->New File and navigate to choose the right image file if the image name does not appear in the list. This should then be the original image file (e.g. a TIFF image file). Click "*OK*" when the file has been selected.

A menu with the title "*Export EVF Layers to ROI*" will then be displayed. From this, choose the option: "*Convert each record of an EVF layer to a new ROI*" and press "*OK*". When the "*Available Vectors List*" appears this part of the processing is finished.

An alternative to importing a vector file is to draw the ROIs interactively. This can be done in the ENVI part of the system, by selecting "Tools"->"Region of interest"->"Define region of interest". (If needed, see the ENVI documentation for further explanation).

The second choice in the mask menu (
Figure 6), *"2. Save as ROI,"* will complete the create mask step. This sub-step will save the ROIs to a file. From the displayed menu, choose "*Select All Items*" and specify the desired name of the output ROI file and click "*OK*". Be careful when selecting the name of the output file, since this choice will determine the default filenames in the rest of the process. A message box will appear verifying that the creation of the ROI file is finished.

3.2.5.2 Import images

The purpose of this step is to create a working dataset that corresponds to a specified ROI. It is assumed that the ROI defines a smaller subset of the whole satellite scene.

The user interaction for the import step is simply to enter the file names of the input files. These files are:

- Panchromatic satellite scene
- ROI file

The system will get the filenames by showing a file selection menu to the operator. The operator may change the file filter and the directory to look in. If the "*Cancel*" button is pressed, the system will terminate the step and display an error message.

The operator must make sure that the specified files are consistent to each other, i.e. the ROI must be associated with the panchromatic satellite scene. The scene must also be in agreement with the processing settings (image type and mark type) defined in the main menu. The satellite scene must be geo-corrected to some map projection.

The *Import images* step will convert the ROI file to a mask in ENVI image format, and save a spatial subsection of the input scene to new files in ENVI image format. The spatial extent of the output image will correspond to the extent of the ROI.

The imported images will be written to a subdirectory of the directory where the ROI file was found. A default filename is derived from the ROI file. This name is also used for the subdirectory, which will be created, if not already existing. A message box will appear showing the filenames of output files. The names of the created files will consist of the default filename with an additional postfix:

- Panchromatic image: <default_file_name>**_pan**
- Imported mask: <default_file_name>**_msk**

If the *Display results* option has been set in the main menu, the panchromatic image and the imported mask will be displayed.

The system will remember the path and name of the ROI file in order to suggest filenames in the following modules. When working with different images, the operator must therefore always run the import step when switching image scenes, even though the imported files have been produced before. (Alternatively, the interactive menu may be closed and reopened, since the system will remember the path and name of the first file it operates on after opening the menu, even when *"import images"* is not the first operation).

It is not recommended to change the options or settings after the import step.

3.2.5.3 Find segments

The purpose of this step is to identify segments in that have a contrast to the background in order to find potentially interesting cultural heritage sites. These segments are analysed further in the *extract features* and *classify segments* steps.

When this module is started, the system will suggest a panchromatic image and a corresponding imported mask. If the operator is calling the steps in sequence, the suggested files will be the output of the previous step. The operator is expected to verify these suggestions.

When the names of the input files have been determined, the automatic search for segments will start. After some time the process will finish, and a message box will be displayed showing the name of the segment image file that has been produced. This will be the:

- Segmented image mask: <default_file_name>**_seg**

This result is a mask that identifies segments in the panchromatic image that are either darker or brighter than their surroundings. Segments outside the imported mask are ignored.

If the *Display results* option has been set in the main menu, the panchromatic image and the segmented mask will displayed

3.2.5.4 Extract features

The purpose of the feature extraction step is to retrieve some descriptive measurements that can be utilized for the classification of the segments.

When this step is started, the system will suggest a panchromatic image and a corresponding segment image. If the operator is calling the steps in sequence, the suggested files will be the output of the previous steps. The operator is expected to verify these suggestions.

When the names of the input files have been determined, the automatic feature extraction will start. After some time the process will finish, and a message box will be displayed showing the name of the files that has been produced.

The output files will be:

- Segment index map: <default_file_name>**_ftsind**
- Feature vector file: <default_file_name>**_ftsvec**

The segment index map is a refined version of the input segment image, where some segments have been removed, and each of the remaining ones have been assigned to a unique segment index. This index identifies the segment in the feature vector file, where the extracted features are stored.

If the *Display results* option has been set, the panchromatic image and the feature index map will displayed.

3.2.5.5 Classify segments

The purpose of the classify segments step is to assign a class label to each of the segments from the segmentation step.

After the feature extraction, the segments are represented by the feature index map. The classification process examines the extracted features, as found in the feature vector file, and compares them to a Class Description Base (CDB).

When this module is started, the system will suggest a feature index map and the corresponding feature file. If the operator is calling the steps in sequence, the suggested files will be the output of the previous step. The operator is expected to verify these suggestions.

If the *Display results* option has been set, the panchromatic image and the feature index map will displayed. In that case the system will also suggest a panchromatic image when the step starts.

The system will automatically select the Class Description Base (CDB) to apply in the classification. This is determined by the processing settings in the main menu.

However, if the option *User specified CDB* has been set in the main menu, the operator is free to select any CDB file. The operator should be very careful in this case, which is not recommended.

When all input parameters have been determined, the system will analyse the features for each segment and assign class labels. The process will complete in a few seconds, and a message box will be displayed showing the name of the classified segment image file that has been produced. This will be the:

- Class image: <default_file_name>**_cls**

This result is mask where the input segments have been classified into different classes, where each class is given a different colour.

3.2.5.6 Edit result

The purpose of this step is let the user edit the resulting detections by removing segments that are not considered as potential cultural heritage sites.

When this module is started, the system will suggest a panchromatic image and a corresponding class image. If the operator is calling the steps in sequence, the suggested files will be the output of previous steps. The operator is expected to verify these suggestions. As an alternative, the operator may want to select the output of the segmentation step.

The panchromatic image will be displayed with the contours of the segments superimposed. The system will lead the operator through all the segments one-by-one. Each segment will be displayed in the zoom window and the operator asked whether to delete the segment or not. By pressing "*Cancel*", the operator may stop the edit process and save those segments that not have been removed

Two different types of files will be produced in the process:

- Updated result as an image mask: <default_file_name>_cls_edit
- Updated result as a region-of-interest (.roi) file: <default_file_name>_roi_edit

The edited ROI file may be used to produce a shape-file by using interactive ENVI tools.

3.2.6 Interactive processing of rings

The main steps of the processing of rings (Figure 8) are:

1. *Create mask*. The areas to be analyzed are identified based on an existing vector file, resulting in a set of regions-of-interest (ROI).
2. *Import image*. The areas to be analysed are retrieved from the satellite scene by means of the ROIs from step 1. The image format is converted to ENVI image file format.
3. *Search for rings* (segmentation). Potentially interesting locations within the areas defined by the ROIs are identified by means of image segmentation.
4. *Edit result.* The final set of potentially interesting cultural heritage locations is finally identified in an interactive step where the each segment is inspected manually.

Figure 8. The dialog for interactive processing of rings.

3.2.6.1 Create mask

This step is identical with the *create mask* step in processing of amorphous objects, please see Section 3.2.5.1 for the details.

3.2.6.2 Import images

The following description is the same as for processing of amorphous objects, which appeared in Section 3.2.5.2, but with an exception at the end (Section3.2.6.2.1). The common parts are repeated here for easy reference, set in a smaller font height.

The purpose of this step is to create a working dataset that corresponds to a specified ROI. It is assumed that the ROI defines a smaller subset of the whole satellite scene.

The user interaction for the import step is simply to enter the file names of the input files. These files are:

- Panchromatic satellite scene
- ROI file

The system will get the filenames by showing a file selection menu to the operator. The operator may change the file filter and the directory to look in. If the "*Cancel*" button is pressed, the system will terminate the step and display an error message.

The operator must make sure that the specified files are consistent to each other, i.e. the ROI must be associated with the panchromatic satellite scene. The scene must also be in agreement with the processing settings (image type and mark type) defined in the main menu. The satellite scene must be geo-corrected to some map projection.

The *Import images* step will convert the ROI file to a mask in ENVI image format, and save a spatial subsection of the input scene to new files in ENVI image format. The spatial extent of the output image will correspond to the extent of the ROI.

The imported images will be written to a subdirectory of the directory where the ROI file was found. A default filename is derived from the ROI file. This name is also used for the subdirectory, which will be created, if not already existing. A message box will appear showing the filenames of output files. The names of the created files will consist of the default filename with an additional postfix:

- Panchromatic image: <default_file_name>**_pan**
- Imported mask: <default_file_name>**_msk**

If the *Display results* option has been set in the main menu, the panchromatic image and the imported mask will be displayed.

3.2.6.2.1 Import images – specific details for rings

The system will not use the path and name of the ROI file when looking for files in subsequent steps.

3.2.6.3 Search for rings

The purpose of this step is to identify ring-shaped segments that have a contrast to the background in order to find potentially interesting cultural heritage sites.

When this module is started, the system will suggest a panchromatic image and a corresponding imported mask. If the operator is calling the steps in sequence, the suggested files will be the output of the previous step. The operator is expected to verify these suggestions.

The operator is also prompted for a third file, a contrast enhanced image. If this file doesn't exist, or if the operator wishes to use some other parameters when creating this file, they should select "Cancel" in the dialog entitled "pick a contrast enhanced Envi image". In that case, a new contrast enhanced image will be produced.

In any event, the "ring search parameters" dialog (Figure 9) will appear. Unfortunately, it tends to appear completely hidden behind the "available bands list" dialog, so the latter needs to be moved for the former to be seen.

The "ring search parameters" dialog allows the operator to change any of the following six parameters.

Minimum ring radius – this is the radius, in decimetres (0.1 m), of the smallest ring template to be used. Note that for Quickbird, one pixel is 0.6 m, and for Ikonos, 1.0 m.

Maximum ring radius – this is the radius, in decimetres, of the largest ring template.

Radius steps – this is the radius increment, in decimetres, between successive ring templates.

Minimum similarity – for each match between a ring template and a potential ring in the image, a similarity measure is computed. If this similarity measure is greater than the minimum similarity, the ring match is kept, otherwise skipped. By using a higher value for minimum similarity, fewer potential rings will be marked. Conversely, by using a lower value for minimum similarity, more potential rings will be marked.

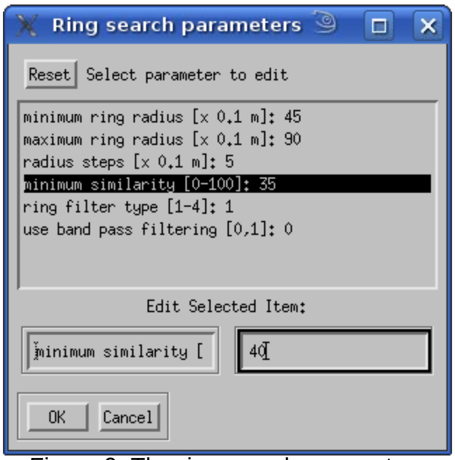


Figure 9. The ring search parameters dialog.

Ring filter type – four varieties are available:

1. ring
2. thin ring
3. thick ring
4. disc

Use band pass filtering – either "0" or "1". If "1" is used, and if no contrast enhanced image has been selected, then band pass filtering in the frequency domain will be applied before making the contrast enhanced image. For details, please see [4].

When the "OK" button is pressed in the "ring search parameters" dialog, the actual processing will start. This may take a little while. When finished, a number of files will be produced.

- Ring centre index map: <default_file_name>**_ringFeatures_ftsind**
- Feature vector file: <default_file_name>**_ringFeatures.txt**
- ROI file containing circles: <default_file_name>**_ringFeatures.roi**

The ring centre index map is an image where each ring is represented by a single pixel, located at the centre of the circle. The value of the pixel identifies the ring in the feature vector file. In the feature vector file, each extracted ring is represented by one line, each line containing one value for each extracted feature. The ROI file contains the coloured circles.

Another set of output files, identical in contents to the above but with different names, will also be produced:

- Ring centre index map: <default_file_name>**_classifiedRings_ftsind**
- Feature vector file: <default_file_name>**_classifiedRings.txt**
- ROI file containing circles: <default_file_name>**_classifiedRings.roi**

These will be the input files to the edit result step.

3.2.6.4 Edit result

The purpose of this step is let the user edit the resulting detections by removing segments that are not considered as potential cultural heritage sites.

The operator is prompted to select a panchromatic Envi image. The system will locate the correct feature vector file.

The panchromatic image will be displayed with the classified rings superimposed. First, a dialog (Figure 10) informs the operator the number of rings to validate, and how to do this. When this dialog is closed, the system will lead the operator through all the rings one-by-one. Each ring will be displayed in the zoom window, and the operator is asked whether to delete the ring or not (Figure 11).

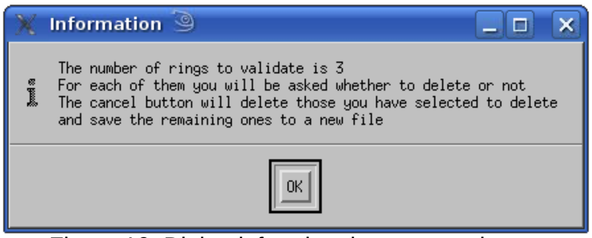


Figure 10. Dialog informing the operator how validation be done.

Figure 11. For each ring, the operator is prompted if the ring should be deleted (yes) or not (no), or to terminate the process (cancel)

By pressing "*Cancel*", the operator may stop the edit process and save those segments that haven't been removed. Alternatively, the process will step through all the rings. In both cases, when the process has finished, a dialog will list the files that have been produced.

The output files will be:

- Ring centre index map: <default_file_name>**_verifiedRings_ftsind**
- Feature vector file: <default_file_name>**_verifiedRings.txt**
- ROI file containing circles: <default_file_name>**_verifiedRings.roi**

As above, the ring centre index map is an image where each ring is represented by a single pixel, located at the centre of the circle. The value of the pixel identifies the ring in the feature vector file. In the feature vector file, each extracted ring is represented by one line, each line containing one value for each extracted feature. The ROI file contains the coloured circles.

3.2.7 Automatic mode

In the automatic mode, the system will automatically process all automatic steps of the process. These are: (1) *Import*, *segmentation,* (2) *feature extraction* and (3) *classification*. The *Create mask* step and the *Edit result* are excluded from the automatic mode because they are interactive steps, which must be called from the Interactive menu. The process is started by pressing the 'Start' button (Figure 12) after having determined the image type, mark type and object type combination, see section 3.2.2. The processing options (see section 3.2.3) are not relevant for the automatic mode.

The user interaction for the automatic mode is simply to enter the file names of the input files. These are somewhat different depending on whether "… soil amorphous" or "… crop rings" has been selected for "type of satellite image" (Figure 12).

Figure 12. The main dialog in automatic mode.

3.2.7.1 Automatic mode – specific details for amorphous segments

The input files are:

- Panchromatic satellite scene
- ROI file

The system will get these filenames by showing a file selection menu to the operator. The operator may change the file filter and the directory to look in. As for the interactive mode the ROI file will determine the default output filename. If the "*Cancel*" button is pressed, the system will display an error message and go back to the main menu. The operator must make sure that the specified files are consistent to each other, i.e. the ROI must be associated with the panchromatic satellite scene. The scene must also be in agreement with the processing settings (image type and mark type) defined in the main menu. The satellite scene must be geo-corrected to some map projection.

The result of the automatic processing will be a class image as described in Section 3.2.5.5. This file will be:

- Class image: <default_file_name>**_cls**

This class image is mask where the input segments have been classified into different classes, where each class is given a different colour.

Before running the automatic mode, an appropriate region-of-interest file (.roi) must be present. Such files can be produced interactively as described in Section 3.2.5.1, or from the ENVI menu. After having run the automatic mode, the resulting class image can be further refined by the interactive edit process described in Section 3.2.5.6.

3.2.7.2 Automatic mode – specific details for rings

The input files are:

- Panchromatic satellite image
- Corresponding mask image
- Contrast-enhanced version of the panchromatic satellite image (optional)

The operator must ensure that these files are consistent with each other. The operator may press the "cancel" button when the system asks for the contrast enhanced image. In this case, a new contrast enhanced image will be created as part of the processing.

In addition to asking for the three input images, the system will display the "ring search parameters" dialog (Figure 9). Please see Section 3.2.6.3 for an explanation of each parameter. By pressing the "OK" button, the processing will start. Progress bars are displayed for the creation of the contrast enhanced image (if missing) and for the ring search. When the processing has finished, the panchromatic image is displayed, with detected rings superimposed.

The following output files are produced.

- Ring centre index map: <default_file_name>**_classifiedRings_ftsind**
- Feature vector file: <default_file_name>**_classifiedRings.txt**
- ROI file containing circles: <default_file_name>**_classifiedRings.roi**

In addition, some intermediate files are produced.

3.2.8 Training functions

Please note that training is not needed for ring detection, only for amorphous segments.

The purpose of the training mode is to teach the system how to classify the segments. The system may be run in a wide range of settings (see Section 3.2.3) including variation in the type of satellite image (Ikonos / Quickbird) and marks to analyze (crop / soil). The system must therefore be able to learn how to classify in many different settings. This knowledge is formalized in terms of a set of CDBs (Class Description Base), each of them being designed for one specific case.

The purpose of the training process is to allow for the production and maintenance of a wide set of CDBs. A CDB refers to the classes and the features to be used for the classification. For each class it provides information on the statistical parameters of the features. These parameters are mean vectors, standard deviation, co-variances, etc. They will differ significantly between the various settings, and the system therefore needs to produce and maintain CDBs for all actual settings. A CDB will typically be made from features extracted from more than one image.

The current version of the system has no complete production chain for the training process. When undertaking training, the operator is expected to run the system in interactive mode and make the calls to the training steps (*Mark* and *Training*) after the *Extract features* step.

3.2.8.1 Mark segments

The "mark segments" step is an interactive process, where the operator is being led through the amorphous segments that were identified in the segmentation step. The purpose is to mark the segments with a class label in order to produce a *marked segment list*.

The operator will be asked for the name of a panchromatic image file, a corresponding FVB file, and a corresponding object mask index file. The operator will also be asked to specify a file that specifies the classes to select from.

When the system starts, it will spend a few minutes to retrieve the contour line around the segments. Afterwards, the system will show one segment a time with the contour line drawn.

The operator will be asked to specify an appropriate class for the segment (Figure 13). The system will only accept class labels that are in accordance with the specified class list

The operator may skip a segment if they are uncertain about what label to assign. These segments will then be repeated later in the marking process. The operator may also delete segments. There may be a large number of segments to mark, so the operator should save the segment mark list from time to time by using the save button.

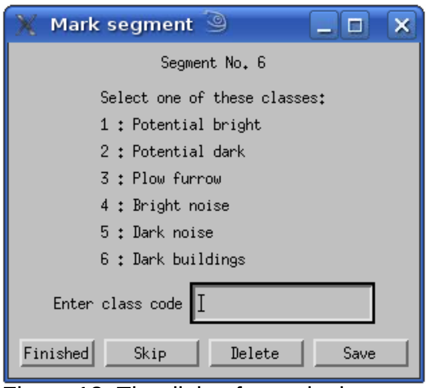


Figure 13. The dialog for assigning class labels to individual segments

3.2.8.2 Training

The purpose of this step is to create class descriptions that can be used in the classification step. Please note that training is not needed for rings, only for amorphous segments.

The training step is an automatic process where a class description base is being updated from features extracted from a segment image together with a corresponding list of segment markings.

When the training module is started, the operator will be asked to specify these files:

- CDB (if not specified, a new CDB will be created)
- FVB
- List of segment markings

If the CDB should be updated with several FVBs, the training module must be called several times.

3.3 Directory structure and filename conventions

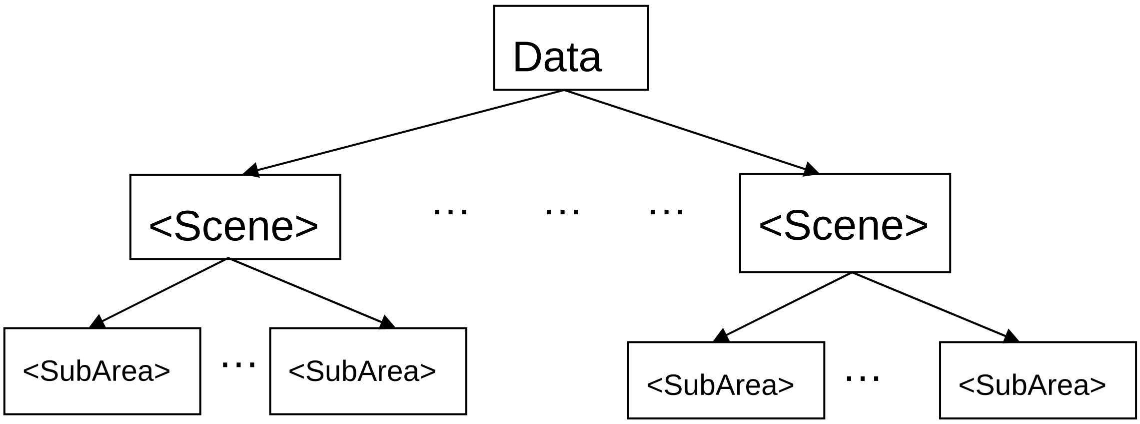


Figure 14. The directory structure of the Data directory

Before starting to use the system the operator should make the directory *Data* with one subdirectory for each satellite scene to be analyzed (Figure 14). When a new satellite scene arrives, the operator should make a new subdirectory for that scene. It is recommended that the scene is converted to Envi format before putting it into this directory.

Besides the directories for each scene the Data directory contains the CDB files generated by the user.

The <Scene> directory should contain the pan image for that scene. For each scene there should also be created at least one region-of-interest (ROI) file, e.g. by means of the interactive "*Create mask*" step. It is recommended that the satellite scene is divided into several sub-areas by making several ROI files. The operator should select the names of these ROI files carefully, as described in the following.

During the processing the system will ask the user to define a local area by specifying the name of an ROI file This filename will also determine the name of a subdirectory in the <Scene> directory, i.e. the <SubArea> directory. All files that are the result of running the CultSearcher with scene <Scene> and ROI <SubArea> will be put in that directory. Also the names of these files will be derived from the ROI filename

The user is recommended to follow some file name conventions when the ROI file is produced. Choices made by the user in the GUI for interactive or automatic mode will be part of most suggested file names. In the description below these will be given as follows

<ImgType> Image type, i.e., either *QB* (for Quickbird) or *IK* (for Ikonos)

<MarkType> Mark type, i.e., either *Crop* or *Soil* marks

<SegmType> Segment type, i.e., either *Amorph* (for Amorphous) or *Rings*

<Spectral> Spectral characteristics of images used, i.e. either *pan* (only the pan image is used) or *mul* (both the pan and the multi-spectral image have been used).

Please note that in the current version of the user interface, multi-spectral images are not supported, but the underlying software is designed with future use of multi-spectral images in mind.

In the current version of the system, there is yet no support for helping the operator to follow these recommendations.

3.3.1 Names of files in the Data directory

The files in this directory are the CDB files. These are given the names <ImgType>_<MarkType>_<SegmType>_<Spectral>.cdb.

3.3.2 Names of files in the Scene directory

The only images in this directory are files that represent the original satellite scene (or rather ENVI versions of them). The panchromatic image should have the extension .pan. If a multi-spectral image is available, it should have the same name with the extension .mul.

3.3.3 Names of files in the SubArea directory

All files that are the result of running the CultSearcher application with scene <Scene> and ROI <SubArea> will be put in the directory <SubArea>. These files will all start with the directory name <SubArea> followed by a string that indicates what kind of result file it is and what choices where made by the user. Below we describe what the different kinds of strings mean:

<Mask>.roi – This is the ROI created by "Create mask".

<Mask>_ <MarkType>_<SegmType>_<Spectral>.fvb …..

The files are described in section 2.5.1

4 System Developer's Guide

4.1 Introduction

The application software is programmed in IDL, with calls to ENVI. The software is organized into software modules as described in Section 2.3. CVS[1] (Concurrent Versions System) is used for version control. The current version of the code is checked out using the command "*cvs co CultSearcher*". The software will run on Windows and Linux. There code has been developed on a Linux platform. Necessary files for installing the system on a Windows platform are included in the directory containing the CultSearcher software. It is assumed that the files in the CultSearcher directory are not modified by the user.

4.2 Directory structure

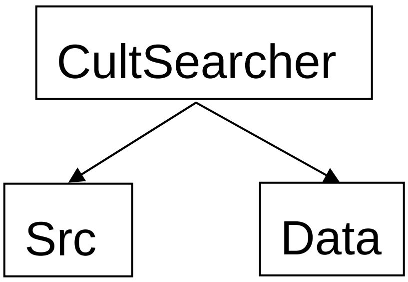


Figure 15. The directory structure of the CultSearcher directory

The directory structure of the CultSearcher directory (Figure 15) is as follows. The directory Src contains all the programs constituting CultSearcher and the directory Data contains test data. The files in the Src directory will be described in the Section 4.3, while the Data directory has the same structure as the Data directory described in Section 3.3.

Besides the two directories Src and Data, the CultSearcher directory contains a few files that are needed in order to make the system, run on both on Windows and Linux platforms. These files will be described in the next three subsections.

4.2.1 Files needed both on a Windows and on a Linux platform

CDB — This directory contains all the files needed for the system. They are expected to contain two files for each setting, as defined in the main menu for interactive processing. There is one file for the panchromatic case (*pan*) and one file for the case when a multi-spectral image is included (*mul*). At the time being, some of these files do not contain a proper class description for the corresponding setting.
The current system has not included any CDB files for ring structures

***.hlp** — All files with the .hlp extension are help files.

cult_logo.tif — This is the logo used by the interface.

display.men — This file contains definitions of menus in display windows.

envi.men — This file contains the definitions of the menus in the ENVI system, including the start menu for CultSearcher

[1] For more about CVS see http://www.nongnu.org/cvs/

e_locate.pro Necessary in order for ENVI to find the files in the subdirectory Src.

envi.mal This file is the basis for automatically generating envi.cfg.

envi.cfg This file is generated automatically based on envi.mal.

4.2.2 Files needed on a Windows platform

configure.vbs The file contains the commands for adapting an installation of CultSearcher to a particular user site. The program is run by either double clicking on the file or by right clicking the file and choosing ”Open with command prompt”.

compile_src.txt Only used when preparing the software for users that have an ENVI licence that does not include IDL. This file contains commands for making precompiled .sav files from the .pro files with ENVI/IDL code. The commands are run from the ENVI/IDL command window.

4.2.3 Files needed on a Linux platform

make_configfile This file updates the ENVI configuration file.

4.3 Detailed description of the system modules and their main functions

In this section we describe each of the GUI and processing modules, their purpose and their interface. The name of the file with IDL code that implements the module is given. All these files are found in the Src directory.

4.3.1 Interactive mode

This mode is controlled by two modules that interact with each other. The cult_interactive module is independent of the application, while the cult_api_interactive contain anything that is related to the application.

cult_interactive.pro: This module contains the general tools for managing the interactive mode. It starts the main GUI for the interactive mode and handles its events. It analyses the events and interacts accordingly with the cult_api_interactive module. It is made as application independent as possible.

cult_api_interactive.pro: This module takes care of all application dependent part of the interactive processing. It makes the initial definition of the processing steps, settings and options, on request form the cult_interactive module. During the processing it will receive the user input concerning processing step, settings and options from the interactive GUI, and from this input it will display menus, determine input/output and other interaction with the operator, before starting the automatic or interactive function for the requested step. These steps are: create_mask, import_images, segmentation, feature extraction classification and edit_result

4.3.2 Automatic mode

cult_automatic.pro: This is the interface for automatic mode. From this the automatic steps import_images (cult_import_images.pro), segmentation (cult_segmentation.pro), feature extraction (cult_feature_extraction.pro) and classification (cult_classification.pro) are called in this order. The module takes care of the necessary interaction with the operator, like asking for file names etc.

4.3.3 Processing modules

This section describes the modules that are needed when processing an image in order to make a classification. The modules will be called under the interactive and/or the automatic mode

4.3.3.1 Create mask

This module is implemented in *cult_create_mask.pro.*

Purpose and functionality

The purpose of this module is to create regions-of-interest (ROIs) defining the areas (e.g. agricultural fields) to be analyzed. These areas are extracted from an existing shape file. The functionality provides an interface to ENVI's tools for creating ROIs from shape files. This functionality is only available in the interactive mode.

Interface

```
FUNCTION  cult_createMask_menu, event
```

This function is an interactive function which utilizes functionality provided in the ENVI system. Input to the function are an image and a corresponding vector file (.shp or .evf) defining the outline of the areas to be analysed. Output is a file representing the vectors in the form of regions of interest (.roi file).

4.3.3.2 Import image

This module is implemented in *cult_import_images.pro*.

Purpose and functionality

The function takes as input a satellite scene and a region-of-interest (.roi) file. The scene consists of one or two image files in a wide range of formats, including ENVI, TIFF/GEOTIFF and PCI. The output is a set of ENVI image(s) and mask that corresponds to the area defined by the ROI.

Interface

```
FUNCTION  cult_importImages,  panFileName, $
                              mulFileName, $
                              roiFileName, $
                              importedPanFileName, $
                              importedMulFileName, $
                              maskFileName, $
                              errMsg
```

panFileName	Filename of input panchromatic ENVI image
mulFileName:	Filename of input multi-spectral ENVI image
roiFileName:	Filename of input ROI-file (Region of Interest)
importedPanFileName:	Filename of imported ENVI panchromatic image
importedMulFileName:	Filename of imported ENVI multi-spectral image
maskFileName:	Filename of imported mask.
errMsg:	Error message (text) if something wrong.

4.3.3.3 Segmentation

This module is implemented in cult_segmentation.pro.

Purpose and functionality

This module performs a segmentation of the panchromatic image into segments that will be further analyzed. The function takes as input an ENVI image and an imported mask, and produces a segment image.

Interface

```
FUNCTION  cult_segmentation,  imgFileName, $
                              maskFileName, $
                              segmFileName, $
                              msg
```

imgFileName:	Filename of input ENVI image
maskFileName:	Filename of input ENVI mask.
SegmFileName:	Filename of mask file containing segmented result.
msg:	Error message (text) if something wrong.

Methods

Segmentation is the process of dividing the areas of the image into different categories based mainly on their spectral characteristics. The current segmentation method works on panchromatic images, and it tries to identify areas that are darker or brighter than the surroundings. This is obtained by using Niblack's method for threshold selection. The method is applied in two passes, first to find dark segments and then to find bright segments. The method works by estimating the mean and the variance in a local area. A local threshold is then derived based on these two parameters. Depending on the combination of parameters, thresholds for detecting local areas that are brighter or darker than their surroundings are found.

Once dark and bright segments have been detected like this, they are post-processed by removing segments that are too small. A final check to verify that segments have a reasonable contrast to their surroundings is also performed. This is done by comparing the means of the different detections with the means of segments surrounding each of the detections. Only if the difference between these two means is large enough will the detections be considered to have a large enough contrast to its local surroundings.

4.3.3.4 Extract features

This module is implemented in cult_extract_features.pro

Purpose and functionality

For each segment identified in the segmentation step, different characteristics (or features) are extracted. These will be used for classifying the segments in the classification step. The retrieved features are stored in a feature vector base (FVB), which is a text file. In order to be able to identify the segments in the FVB, the segments are also being labelled and stored in feature index map, which is an image file.

Interface

```
FUNCTION cult_extractFeatures, PAN=panFileName, $
                               MULTI=multiFileName, $
                               SEGM=segmFileName,   $
                               FTST=ftsFileName, $
                               FTSI=ftsIndexFileName
```

panFileName: Filename of input panchromatic ENVI image
multiFileName: Filename of input multi-spectral ENVI image
segmFileName: Filename of an ENVI mask file resulting from segmentation
ftsFileName: Filename of the produced feature (FVB) file
ftsFileIndexName: Filename of the produced feature index file

Methods

The features that currently are applied to characterize segments are the following:

- Ratio between perimeter and area
- Gradient along boundary of segment
- Relation between height and width (major and minor axes)
- Standard deviation of grey levels within segment
- Difference in local grey levels inside and outside the segment

These features are obtained by tracing the boundary lines around the segments, and utilizing the IDL-ROI functionality in the IDL/ENVI system

The operator may also choose to include also features extracted from the multi-spectral data. In this case one feature is added for each spectral band. These are

- The average pixel value for the segment in each spectral band, i.e. the R, G, B and NIR channels. These values can be considered as the spectral signature of the segment

The features may be redefined later when more experience is gained.

This step will also perform some pre-processing, e.g., this module will dismiss long and narrow structures like plough furrows and road segments.

4.3.3.5 Classification

This module is implemented in cult_classification.pro

Purpose and functionality

The purpose of the classification is to further analyze the characteristics of the areas identified in the segmentation step and classify these areas. The function takes as input the output from the feature extraction and a fixed Class Description Base (CDB), which is a text file containing the features statistics each class. It produces a classified image represented as an ENVI mask.

Interface

```
FUNCTION cult_classifySegments,     FTST=ftsFileName,
                                    FTSI=ftsIndexFileName, $
                                    CDB=classDescFilename,
                                    CLS=classFileName
```

ftsFileName: Filename of the input feature (FVB) file
ftsFileIndexName:: Filename of the input feature index file
classDescFileName: A statistical class description (CDB) file
classFileName: Filename of the resulting output image.

Methods

The classes that have been defined currently consist of two classes representing objects with potential interest as cultural heritage sites, and four classes of different types of noise objects. The classes may be redefined later when more experience is gained. Statistical descriptions of these classes are contained in the CBD file.

Each input segment is first labelled into a class based on the correspondence between its features and the feature statistics in the CDB file. This is undertaken by utilizing classification techniques in the ENVI system. The current version applies the minimum distance classifier.

The next step will label all pixels that belong to a segment into the class that has been determined for that segment. The resulting class image will be saved together with information of how to display the classes, as shown in the following table:

Class number	***Class description***	***Colour***
Class 1	Potential bright cultural heritage	Red
Class 2	Potential dark cultural heritage	Green
Class 3	Plow furrow	Blue
Class 4	Bright noise	Yellow (red and green)
Class 5	Dark noise	Light blue (cyan, blue and green)
Class 6	Dark objects (houses)	Violet (magenta, blue and red)

4.3.3.6 Edit result

This module is implemented in cult_edit_result.pro

Purpose and functionality

This is an interactive process, and the call to this function starts a menu. The user will be led through the identified and classified segments, and may interactively choose to delete segments that are not believed to represent cultural heritage sites. The result from the process is an updated mask and a corresponding region-of-interest (.roi) file.

Interface

```
FUNCTION cult_validateResult, imgFileName, $
                              clsFileName, $
                              roiOutFileName, $
                              clsOutFileName
```

imgFileName: Filename of input panchromatic image
clsFileName: Filename of the input classified image.
clsOutFileName: Filename of the resulting edited classified image.
roiOutFileName: Filename of the ROI representation of the edited result

Methods

The segments are contoured, and the contours shown superimposed on the image in a colour that corresponds to the classification. The user is given the possibility to remove the current segment.

4.3.4 Training modules

4.3.4.1 Mark

This module is implemented in cult_mark_fvb.pro

Purpose and functionality

The purpose of this module is to produce a list of class labels that refers to an FVB file

The function takes as input a panchromatic image, an FVB file, an object mask index file and a class name file. None of these files are specified in the function interface, but will be provided interactively.

Interface

```
FUNCTION  cult_mark_fvb, event
```

event: Default parameter for menu widget functions (provided by the main menu)

Methods

This function starts with the specified segment image, where the contours around all segments are retrieved. This step may take a few minutes. The contour lines are then displayed superimposed on the panchromatic image, one by one, and the operator invited to assign a label from the list of available class labels. The list provided class labels are stored in a text file.

The contouring of the segment image is not perfect, and some contours may contain more than one segment and must be skipped. In order to be able to mark all segments, the function will repeat the process on the remaining segments until all segments have been presented to the operator. The operator will therefore experience that the system appears to stops, but it is just for the recalculating of the contour lines.

4.3.4.2 Training

This module is implemented in cult_training.pro.

Purpose and functionality

The purpose of this step is to create class descriptions that can be used in the classification step

The function takes as input an FVB file and a corresponding SegmentMarkList.

The output will be a CDB (Class Description Base) that contains statistical parameters for the features in the FVB file.

The CDB will typically be an existing file, which will be updated during the Training step

Interface

```
FUNCTION  cult_training, event
```

event: Default parameter for menu widget functions (provided by the main menu)

Methods

The statistics is calculated separately for each class. The segments belonging to a specific class are given by the SegmentMarkList. The feature vectors for this class are selected and the statistical parameters calculated. This process is repeated for all classes, and the statistical parameters written to the CDB, which is a text file.

The statistical parameters for the classes are the mean, standard deviation and variance for each feature, and the co-variances between the features.

These parameters are calculated as using the accumulation approach. During the accumulation, the parameters are represented by their maximum likelihood estimates, but saved to file as unbiased estimates (n-1).

In the typical case the existing parameters are read from the CDB file and converted to ML estimates. Then they are updated by means of the input FVB / SegmentMarkList pair as described above.

Since the FVB and the SegmentMarkList are separated, the system allows for reconfiguring the features be used even if the SegmentMarkList already exists. However, this will require that the SegmentMarkList and FVB still refer to the same segments

4.3.5 Tools

The tools module is implemented in cult_tools.pro.

It contains a wide range of functions that are common to several modules. In the current version of the system all the tools function are still in one large module.

4.4 Details of the installation

The installation process is described in Section 3.1. Here we will in addition give some details about the file configure.vbs.

configure.vbs is a series of commands written in Visual Basic. When this file is run, all adaptation of CultSearcher to the local environment is performed when the program is installed on a Windows platform.

1) The user variable CULT_PROG is set to the directory where CultSearcher is unpacked. This variable is read by CultSearcher during run and tells CultSearcher for example where the files cult_cdb.txt and cult_multi_cdb.txt are located.

2) The user variable CULT_DATA is set to a directory specified by the user. It is the path to where the user's data are located. The program will put the results here.

3) The user variable IDL_PATH is read by IDL when IDL is started. This variable contains a description about where IDL finds its different program files. The variable therefore contains IDL syntax. The variable first contains a directory for CultSearcher, then a directory that is specific for the ENVI/IDL installation. The script configure.vbs, adds the directory for CultSearcher and CultSearchers Src to IDL_PATH . An example of the content of this variable is:

   C:\Documents and Settings\aurdal\My Envi\CultSearcher;+C:\Documents and Settings\aurdal\My Envi\CultSearcher\Src;+C:\Program Files\RSI\IDL63\products\envi43\lib;+C:\Program Files\RSI\IDL63

   Remark the plus signs that are IDL syntax for concatenation of strings.

4) Al last the file envi.cfg is generated as a copy of envi.mal that is modified by substituting _DIRIMG and _DIRMAIN with by the same directory as the one stored in the user variable CULT_DATA and CULT_PROG, respectively.

In addition the program asks for the root of the ENVI installation (typically C:\Program Files\RSI)

The users of the system will run it on a Windows platform. When a new version of the CultSearcher is made available, the code can be unpacked from CultSearcher.zip (WinZip). We suggest it be unpacked into the directory C:\Program_Files\CultSearcher.

For each user, The Norwegian Computing Center obtains information about username, password and license key from ITT support. The host ID (physical address) of the user's PC must be provided by the user. (Choose "Run" from the Start menu in Windows, and type "cmd" in the window that appears. A command window will then pop up. By typing "ipconfig/all" in the command prompt, a list of host details will be written. The host ID is the number called "Physical Address" under "Local Area Connection" in the list.)

5 Appendix

5.1 Example of an FVB

The feature vector base (FVB) is generated in the feature extraction step. In this example, five features have been retrieved for all segments that have passed the pre-classification. Their identifiers are referring to a labelled segment in the feature index image, which is also generated in the feature extraction step.

The FVB is stored as a table in a text file. In this case the table has six columns. The segment identifiers are stored in the first column, and the five features are stored in the other columns. There is one line for each segment.

```
NofFeatures =       5
FeatureVectors = {
          4      9.78034      131.485      3.53438     -30.4101      20.4511
          7      14.4946      82.2104      6.05926     -62.3115      36.5678
          9      27.5096      78.4048      5.19418     -68.1439      42.2198
         12      6.88352      55.2069      2.61045      24.6108      15.4200
         13      16.4669      130.755      6.35609     -45.5351      29.8584
         16      8.11669      95.0224      2.48235     -21.8092      14.1698
         17      15.2376      151.911      4.37637     -41.4120      32.7084
         18      35.1584      40.1516      3.73730     -82.8633      49.4951
         19      14.5407      166.723      4.94587      40.7885      26.3779
         21      7.43959      64.4472      2.52132      18.2456      12.1131
         22      11.8645      137.319      6.73435     -23.5381      15.1603
         26      10.2455      68.9792      3.79955     -21.9700      13.7540
         29      22.4183      105.947      2.80706     -35.1209      24.5621
         31      10.3280      106.130      2.69974      20.5047      14.7063
         32      11.9296      64.1287      4.79186     -29.3538      19.1203
         35      8.28661      51.8145      2.57414     -18.4647      12.5971
         39      11.4574      99.9319      8.67418     -23.4963      15.3783
         43      30.5384      143.706      5.13308     -42.9974      24.9070
         44      23.3506      194.887      5.73359     -38.4297      23.0370
         45      13.3853      63.4983      3.05546      42.2011      27.7248
         46      21.7145      118.052      3.32893      31.0597      22.6955
         47      21.1044      76.9614      4.73991     -36.4259      24.8517
         48      22.3421      83.2975      7.27343     -41.8682      26.3109
         49      26.2178      73.3175      3.81782     -46.0029      32.9895
         50      8.26479      71.4019      3.90007      17.2390      11.7554
         52      22.6218      49.7485      3.54099     -41.0018      25.5537
         53      22.7042      119.259      2.85047     -35.1075      24.0164
         55      9.82924      43.0940      3.19660     -19.8523      13.0709
         56      48.8366      25.8056      2.66829      78.2237      56.9982
         57      12.6749      98.8778      3.99746     -16.4370      12.4816
         58      42.6581      22.2054      2.53132      97.0522      63.6007
         59      66.0871      70.7102      4.27903     -66.4035      50.7016
         61      16.5335      40.6039      2.51565     -35.1719      23.3551
         62      15.3284      90.9643      4.73753      34.3225      19.6281
         63      33.2851      62.7199      4.36508     -39.5340      25.8972
         64      17.7310      198.379      5.88872     -18.0693      17.3065
         66      23.9386      94.2856      5.59476     -24.8044      15.7747
         67      19.3735      62.0635      4.65461     -21.8414      17.4171
         68      21.9346      89.5338      3.37820     -35.9211      24.6595
         69      59.5902      119.466      5.77887     -54.7950      45.7538
         72      7.45995      126.314      4.40975      11.8138      9.27582
         73      25.8777      96.9350      9.61973     -21.1910      36.9256
         74      15.2992      37.0345      3.75892     -15.8250      12.2462
         75      55.1255      33.5963      3.32773      107.091      83.9051
         76      35.1025      62.3915      2.97551     -53.2873      38.5710
         77      59.9781      63.1279      3.48081     -81.3543      48.9282
```

78	18.2791	134.648	2.50059	25.4309	15.9895
79	13.9673	73.5531	6.99180	-27.1786	17.3921
83	11.4424	73.0772	4.84642	-12.2847	11.6590
84	19.3061	112.807	8.63777	-35.2378	26.8195
85	13.8009	162.647	7.38959	20.4892	14.2358
86	40.1942	65.6321	4.40108	-58.8484	39.0175
87	11.0853	38.5822	3.35980	-26.7227	18.1756
89	22.1702	67.9644	6.64577	-24.7227	24.4660
90	83.6294	66.6576	8.21711	-92.3751	68.1548
91	17.6192	81.6884	3.08155	30.7716	19.7768
92	47.6475	26.4721	2.69534	82.2792	57.3935
93	11.5221	52.9219	4.01255	-19.4139	12.6965
94	25.4981	193.153	4.64712	-24.9943	20.9638
96	42.9528	79.1568	5.85436	-50.2501	31.0764
98	20.1709	96.7013	3.60308	-30.9497	23.3185
99	16.1253	109.684	5.02462	33.6625	25.4421
102	46.0684	68.2230	2.67193	71.5057	46.0617
103	36.1592	59.1937	3.88185	-43.7910	30.6090
104	11.0223	160.545	2.47058	17.0899	12.2553
105	9.14525	95.0424	6.32911	-13.9425	9.83651
107	20.0430	47.6452	2.75352	-32.5546	22.3729
108	40.5439	43.1362	2.47212	-57.6512	36.4422
110	53.3548	110.052	6.99047	-53.4930	44.1070
112	13.5230	62.2173	2.73233	-35.0161	25.0501
113	47.4619	89.2945	6.35300	-53.1837	36.3315
114	42.2492	70.9145	5.19188	-65.9546	43.4030
115	47.2610	41.2948	5.08508	-67.9527	47.5010
117	18.6435	122.038	8.09430	25.8805	17.0024
118	30.0492	71.2520	5.93773	-31.4366	21.5440
119	10.2726	144.286	3.69195	18.2143	12.5327
120	27.2841	73.4972	3.06999	-31.2672	25.1317
121	20.5683	37.7393	3.06194	-38.7644	26.0060
122	39.0028	76.4358	3.67671	-50.4755	36.1042
123	7.84931	93.5624	6.62368	15.3336	10.1348
124	22.7358	69.4382	6.53563	-25.5965	25.6070
127	29.3588	91.4917	9.99101	54.2530	30.2458
128	31.5057	103.043	5.93295	-24.8445	22.7752
129	21.6590	56.1157	3.39101	-29.2002	19.5524
130	29.0051	90.5783	9.44967	-35.2506	30.8158
131	36.1838	174.029	9.17367	-25.4936	23.7136
132	65.5649	80.1662	7.32053	-84.5356	53.1523
133	33.9390	22.5983	2.49377	77.8131	50.4864
134	13.7919	83.8687	6.09291	25.9108	19.0563
136	12.7600	37.9673	2.91878	19.4950	12.7909
137	34.6402	89.7111	5.14545	-43.6669	28.2159
139	58.7122	63.1625	4.25699	-72.3835	46.5259
140	13.6747	68.3127	2.41747	-26.6778	17.1810
141	53.1956	65.4630	5.09766	-60.2368	45.0387
142	10.2114	56.5324	3.34503	-14.8091	12.5428
144	9.48935	63.6847	3.82266	-11.0254	8.38138
145	13.3335	70.1378	3.95283	17.5119	13.4357
146	55.8126	51.6163	4.05872	-75.2565	56.0716
147	59.2694	84.1840	9.06076	-56.0686	36.0506
148	31.8419	63.5524	2.66528	-26.5722	27.1611
149	30.6131	20.0116	2.57697	53.8519	31.5690
150	59.4200	48.4155	5.47779	-63.6300	44.6188
151	77.7521	20.2662	2.69128	114.384	66.7768
152	9.42106	48.8785	2.43809	18.2038	13.0006
153	27.2240	42.7105	4.06277	-41.1424	35.4452
154	29.5188	36.1612	3.49794	-26.2614	20.1106
155	59.9449	70.9726	7.86449	-66.5592	47.0983
156	28.3830	121.679	3.98678	43.7568	25.7940
157	16.7510	109.147	7.89927	-24.3141	17.7384
158	14.8232	59.4383	5.54268	25.6349	15.6352
160	34.8616	43.0538	5.01123	-36.8469	35.4394
162	32.9953	41.4847	3.16311	-41.2008	27.4472
165	17.6935	38.8274	2.69172	-21.3470	15.8797

166	34.8667	19.8395	2.54850	72.0260	57.0226
167	23.8222	71.3308	7.28624	31.9510	35.3210
168	38.8560	113.340	2.53369	-16.1262	32.7560
169	27.6695	106.510	8.51449	37.4489	40.9390
170	78.1780	35.9750	4.84412	-58.0893	46.4576
171	36.8630	32.6972	3.61391	-54.6875	35.6580
172	17.6876	73.3755	6.57891	33.0659	21.2063
173	9.31355	36.8935	2.57906	-21.0490	15.4221
174	21.8006	161.545	6.21130	34.5011	21.6591
175	17.8002	45.8656	3.33378	36.3850	24.5955
176	36.1322	59.7318	2.43261	62.2637	44.7486
177	13.1220	73.0277	4.33582	18.2135	12.6832
179	37.6395	40.0366	2.70268	-49.9012	34.3914
181	8.45286	56.4285	3.89488	10.9927	7.62591
183	11.1490	96.1110	8.93315	16.2250	9.95522
184	23.2436	42.2022	2.98198	20.3318	29.7525
185	36.7639	29.3841	3.43971	-37.7234	28.8066
186	12.0382	35.6823	2.75158	19.1112	12.6913
188	11.0778	78.9941	6.05480	21.4325	13.1277
189	8.39067	86.7592	6.59444	20.1771	14.4560
190	15.2096	46.9230	2.75657	26.8970	16.9774
191	88.5106	39.9393	4.98384	-81.8727	52.3431
192	55.2959	84.0936	7.20800	53.9510	56.2390
195	41.6464	35.4242	4.28247	-55.7284	32.6681
196	45.3545	77.8320	8.23404	-39.9059	27.8354
197	34.7011	22.1993	2.95650	-24.9699	24.4040
198	27.4522	37.5346	2.48002	58.3071	45.5185
199	23.5866	76.9704	6.44727	-29.2115	37.1358
201	23.7914	67.3239	7.04928	-38.1258	31.4545
202	13.5635	79.4870	3.23269	21.0747	14.7911
203	14.6269	99.2372	6.67922	32.6141	20.5995
204	69.4700	54.0857	5.79302	-82.4550	48.8112
205	22.5837	44.9108	5.19887	6.32143	36.0220
208	12.0123	109.208	6.14642	20.3114	12.6103
209	12.3109	35.3187	2.58387	18.2356	12.2103
210	20.1297	38.9370	3.95657	22.8138	14.2071
211	68.0102	33.3841	2.45816	-63.6103	42.8696
212	16.6347	48.7521	5.24708	24.1194	16.0903
213	26.8556	58.4157	5.79989	-34.8279	21.2856
214	18.6402	25.1980	3.01954	26.7017	16.6184
215	66.7542	47.9707	5.33982	-47.4925	40.9195
216	13.2218	79.8466	5.58228	17.9379	11.2066
217	21.0908	65.5999	6.41545	-23.7401	22.7421
218	9.93182	68.6041	3.75950	16.9396	12.4405
219	35.2639	53.1646	3.99546	-39.2267	25.2361
220	20.4771	167.624	8.37406	25.7543	16.0338
223	17.3601	38.2699	3.15717	29.8815	38.3585
224	26.2533	17.8670	2.43103	-32.4656	21.1545
225	38.9969	22.2030	2.52023	-43.1823	32.3806
226	20.5973	45.3031	4.19316	-29.5772	24.2790
229	40.9123	21.9423	2.57097	-54.0670	40.7361
230	32.6689	23.8650	3.21613	-34.7616	25.5364
231	14.0253	80.9976	6.05193	17.8012	13.0144
232	14.4571	62.4290	2.43537	17.6442	14.5090
233	33.5232	104.718	8.67792	49.3644	24.8356
234	8.96366	107.654	3.72441	19.1774	13.6820
235	17.8098	69.6641	9.03589	-14.9916	15.6844
236	20.8384	22.7667	3.05572	-19.1371	16.0804
238	46.0284	23.1338	2.82736	64.4894	56.3644
239	15.4659	58.4067	3.45676	19.8813	15.9112
242	12.9193	104.143	4.00858	21.1927	14.8039
243	46.3054	28.8639	2.66220	58.8112	39.3643
244	10.2533	87.2673	2.86140	12.2969	8.03515
246	51.4729	46.5284	5.84872	-57.8811	47.4484
247	30.5726	38.7601	4.97445	-13.0867	39.5877
248	9.40335	79.4786	4.70453	13.0047	8.60164
249	24.0063	24.1359	2.54387	-27.7378	21.5555

```
250      11.6394      49.5394      2.64907      16.0299      10.1123
251      61.6853      52.6675      3.74711     -46.1362      40.1482
254      9.32418      180.268      2.99165      11.6373      8.46036
255      42.7527      33.3264      3.52574      38.4822      58.3965
256      19.4421      35.1186      2.57091      32.9872      20.8786
257      14.4712      38.1487      3.01918      19.1311      13.9472
259      26.8337      31.3579      3.49957      29.8975      18.0176
260      16.9165      22.0960      2.92940      22.2495      14.1781
261      35.4713      58.7914      7.52952     -34.8857      25.7131
263      59.5909      37.6831      5.17631     -99.9433      56.1060
264      15.0634      38.3142      2.47172      17.1500      19.1390
265      14.3482      62.7466      7.77257     -17.7009      10.7196
266      19.7131      65.7046      3.34025     -15.7850      29.0543
267      18.1776      71.1533      2.50150      23.0175      15.2800
268      70.4184      19.0788      2.73518      55.2983      52.6265
269      64.5034      23.0743      2.83513      106.377      70.7715
270      25.6799      32.7350      2.48738     -26.0881      22.4755
272      11.4363      65.2457      2.95915      14.8907      10.1723
273      14.6038      19.8676      2.46167      23.2714      15.4183
}
```

5.2 Example of a CDB

In the classification step feature vectors in the FVB file is classified by comparing each of them to the class descriptions in the Class Description Base (CDB). The CDB contains statistical information for each class. The classes are described in terms of the mean value and standard deviation for each feature. In addition, the co-variances between all features are described. In this example, which corresponds to the FVB above, there are five mean values and standard deviation values for each class, and a 5 × 5 matrix for the co-variances. For each class, class name and number of segments that have been used to calculate the statistics, are given.

```
NofClasses  =       6
NofFeatures =       5
#
#
Class1 =   Potential bright
Thres  =   0.01
Nobs   =            100
Mean   =  { 14.959530,  37.434412,  2.6547835,  -65.1396,   26.867682 }
Stdev  =  { 4.843714071, 18.69342104, 0.381218572, 32.27178024, 5.817169759 }
Cov    =  { 23.461566,  31.241251,  0.26543495,  -82.78413,  18.608384,
            31.241251,  349.44399,  5.4627853,  -132.12868,  54.181476,
            0.26543495,  5.4627853,  0.1453276,  0.0801872,  0.53933603,
           -82.78413,  -132.12868,  0.0801872,  1041.4678,  -119.35516,
            18.608384,  54.181476,  0.53933603,  -119.35516,  33.839464 }
#
Class2 =   Potential dark
Thres  =   0.01
Nobs   =            100
Mean   =  { 10.015025,  57.511079,  2.9904275,  6.3629225,   24.355058 }
Stdev  =  { 4.571565487, 32.78250601, 0.895166889, 16.51907897, 7.103535528 }
Cov    =  { 20.899211,  -22.758794,  0.6503576,  35.502699,  25.404354,
            -22.758794,  1074.6927,  19.702141,  -285.62585,  -32.134013,
             0.6503576,  19.702141,  0.80132376,  -6.9732997,  1.5098271,
             35.502699,  -285.62585,  -6.9732997,  272.87997,  51.441436,
             25.404354,  -32.134013,  1.5098271,  51.441436,  50.460217 }
#
Class3 =   Plow furrow
Thres  =   0.01
Nobs   =            100
Mean   =  { 23.659510,  72.629005,  7.47347,  -116.3993,   26.505082 }
Stdev  =  { 14.61055885, 21.7009827, 1.519798342, 54.82527155, 14.78953718 }
Cov    =  {  213.46843,  -110.9792,  -4.1116241,  154.2055,  204.11808,
            -110.9792,  470.93265,  14.646969,  -111.56342,  -105.47206,
```

```
            -4.1116241,  14.646969,  2.309787,  -53.22522,  -4.9784693,
             154.2055,  -111.56342,  -53.22522,  3005.8104,  265.98909,
            204.11808,  -105.47206,  -4.9784693,  265.98909,  218.73041  }
#
Class4 =   Bright noise
Thres  =   0.01
Nobs   =            100
Mean   =  { 10.072889,  83.306155,  3.2358286,  -45.679039,   16.382210 }
Stdev  =  { 2.438746215, 40.8336332, 0.929914356, 31.70008991, 4.033570503 }
Cov    =  { 5.9474831,  -4.2596254,  0.14805239,  -48.388725,  6.8109441,
           -4.2596254,  1667.3856,  9.4802642,  -83.651176,  -4.3855466,
            0.14805239,  9.4802642,  0.86474071,  1.3033331,  -0.37442532,
           -48.388725,  -83.651176,  1.3033331,  1004.8957,  -78.476359,
           6.8109441,  -4.3855466,  -0.37442532,  -78.476359,  16.269691   }
#
Class5 =   Dark noise
Thres  =   0.01
Nobs   =            100
Mean   =  { 11.130508,  69.223727,   3.6277009,  -21.782307,   17.129218 }
Stdev  =  { 1.791106725, 33.21605335, 1.255729708, 29.72567644, 3.232821987 }
Cov    =  {  3.2080633,  -18.866373,  0.238798,  -4.8693775,   3.900485,
            -18.866373,  1103.3062,  18.865442,  -314.62248,   -24.665009,
              0.238798,  18.865442,  1.5768571,  -17.189833,   0.077004144,
             -4.8693775, -314.62248,  -17.189833,  883.61584, 2.7840948,
              3.900485,  -24.665009,  0.077004144,  2.7840948, 10.451138 }
#
Class6 =   Dark buildings
Thres  =   0.01
Nobs   =            100
Mean   =  { 59.829250,  28.65075,   3.6557975,  69.941825,   89.961500 }
Stdev  =  { 16.79889609, 8.443320733, 2.326412775, 31.51676839, 10.91229811 }
Cov    =  { 282.20291,  39.936583,  22.593248,  170.0643,    70.36978,
            39.936583,  71.289665,  15.272392,  -88.117768,  2.9985994,
            22.593248,  15.272392,  5.4121964,  -14.531451,  -3.4508155,
            170.0643,  -88.117768,  -14.531451,  993.30669,  -41.233169,
            70.36978,  2.9985994,  -3.4508155,  -41.233169,  119.07825 }
#
```

6 Bibliography

[1] L. Aurdal, R. Solberg, and J. Lous, *"Computer assisted detection and classification of possible cultural heritage sites: User requirements and conceptual design"*, Norwegian Computing Center, SAMBA/11/05, 2005.

[2] L. Aurdal, L. Eikvil, and H. Koren, *"Cultural heritage search: Software prototype description"*, Norwegian Computing Center, SAMBA/13/06, 2006.

[3] L. Aurdal, L. Eikvil, and H. Koren, *"Cultural heritage search: Software prototype evolution and status report"*, Norwegian Computing Center, SAMBA/10/07, 2007.

[4] Ø. D. Trier, S. Ø. Larsen and R. Solberg, *“Detection of circular patterns in high-resolution satellite images of agricultural land with CultSearcher”*, Norwegian Computing Center, SAMBA/16/08, 2008.