Zoo/PhytoImage version 4

Computer-Assisted Plankton Images Analysis

User Manual

The ZooImage development team
February 2014

Ph. Grosjean & K. Denis: Numerical Ecology of Aquatic Systems, UMONS, Belgium
X. Irigoien, G. Boyra & I. Arregi: AZTI Tecnalia, Spain
A. Lopez-Urrutia: Centro Oceanográfico de Gijón, IEO, Spain
M. Sieracki & B. Tupper (FlowCAM plugin)

1. Introduction

Zooplankton or phytoplankton samples analysis is traditionally associated with long and boring sessions spent counting fixed plankton items under the binocular with formaldehyde vapors floating around. Although this picture of a planktonologist will probably remain for a while, there seems to be another way to gather data about zooplankton: computer-assisted analysis of plankton digital images. A whole suite of hardware to take pictures of our animals, both in situ and/or from fixed samples, is now available: Flowcam, laser OPC, VPR, Zooscan,... (more to come with Holocam, Sipper, Zoovis, HAB Buoy, ...), not forgetting the use of a digital camera on top of a binocular or with a macro lens. But digital images of zooplankton are barely usable as such: they must be analyzed in a way that biologically and ecologically meaningful features are extracted from the pixels. A software doing such an analysis is thus indispensable.

Zoo/PhytoImage aims to provide a powerful and feature-rich software solution to use zooplankton or phytoplankton pictures of various origins and turn them into a table of usable measurements (i.e., abundances, total and partial size spectra, total and partial biomasses, ...). Zoo/PhytoImage is not tight with any of the previously cited devices, and it is not going to be a commercial product. It is distributed for free (GPL license, distributed through its web site, http://www.sciviews.org/zooimage) and it is open, meaning it provides a general framework to import images, analyze them, and export results from and to a large number of systems. So, everybody can use Zoo/PhytoImage... but better yet, every developer can also contribute to it! The Open Source approach of wiring many willing developers around the world in a common project has already shown its efficiency: Linux, Apache, but also R or ImageJ in the field of statistics and image analysis, respectively, are good examples of it. Zoo/PhytoImage is based on ImageJ and R, and it runs on Linux... but it can also be run on Windows, Mac OS, or various Unixes[1]. Zoo/PhytoImage's best qualifying is "reusability". It is born by reusing various features of great existing software like ImageJ, or R, and it provides itself reusable components, for the benefit of both users and developers.

Zoo/PhytoImage can be used on images acquired in different situations: *in situ* (like VPR or HAB Buoy) or in the lab (fixed samples scanned with the Zooscan, for instance). The general framework of Zoo/PhytoImage is designed in a way that the software is capable of dealing effectively with images of various origins and characteristics. Consequently, it is not a streamlined and rigid system. It is rather made of a collection of different and customizable applications collected together in a single system. This user's manual will guide you in your first use of Zoo/PhytoImage.

> *This manual describes current version of ZooImage (4.0-0), which is* ***not*** *a public version ! It is geared towards early adoption among our partners : UMONS, IFREMER, BelSpo, ULCO and LISIC. The functions presented here will eventually land in the next public version 5. However, 4/5 of the code is commonwith version 3, whichi is public and downloadable from CRAN (http://cran.r-project.org).*

[1] The current version is developed mainly on Mac OS X, but is also tested on Windows and Linux Ubuntu.

2. Changes from version 1 and 2

Zoo/PhytoImage version 1.2 was the latest public version distributed on http://www.sciviews.org/zooimage until now. Version 2 of the software was not public and contained several developments made for us (UMONS university) and our main partners : IFREMER in France and Belspo (Belgian Science Policy) in Belgium.

Version 3 of ZooImage collects most of these developments into a relifted system, and it is distributed on CRAN (http://cran.r-project.org). Finally, recent additions made in version 4 do complete the set of features. Main changes are :

- Updated code for running on latest R version 3,
- Complete internal refactoring to make it compatible with Linux and Mac OS X, in addition to Windows. Version 3 also supports Windows Vista, 7 and 8, in addition to Windows XP.
- A new storage format, called ZIDB, that is much faster to retrieve vignettes.
- Routines to build, sort and use test sets, similarly to training sets.
- Functions to display vignettes directly insode R graphs (using R scripts).
- Improved handling of confusion matrices, with the possibility to change prior probabilities of the classes and inspect how this changes the shape of the confusion matrix.
- A battery of summary statistics for the confusion matrix (recall, precision, F-score, specificity, ...)
- New and improved graphs for the confusion matrix, including F-score plots and dendrograms depicting hierarchical classification of the classes according to their confusion.

2.1. New data storage format

Among all those change, the most important one for end users is probably the new storage format, named ZIDB for Zoo/PhytoImage DataBase.

Data storage format is a key aspect for data analysis software. In statistics, there is a consensus towards a 'case-by-variable' format that is suitable for most (but simplest?) datasets. It presents the data in a two-dimensional table with variables in columns and cases (or individuals) in rows. Additional names for columns and/or for rows are allowed. Such data can be stored in plain text, being ASCII, UTF-8, ... encoded, using a predefined field separator and one row per line. The most commonly used format is CSV for « comma-separated values ». It uses either the comma (, English version) or a semi-colon (; French version). Another frequent variant is the TSV format, which uses tabulations as field separators.

CSV or TSV are readable by all software, making them the most universal storage format for case-by-variable data. Excel (or other spreadsheet) formats can be used as well, but they are a little bit less widely recognized.

CSV or TSV are not the most efficient formats when it comes to memory usage or speed. Since numbers as stored as character strings, they consume much more memory than their binary counterpart. It is also impossible to retrieve some data in the middle of a table without reading all previous data since the offset in memory where those data are stored in not predictable.

Another shortcoming of the CSV or TSV format is the impossibility to associate metadata in addition the main two-dimensional table. Yet, this format remains one of the best to store small to mid-sized raw datasets and make sure they will be most readable in the future.

> *In Zoo/PhytoImage, we use a variant of the TSV format where the two-dimensional table of features measured on each particle is prepended by a section defining associated metadata in a key=value pairs set. It is the _dat1.zim file.*
>
> *The same data is also duplicated in own binary R format (.RData) which is much faster to load than the original TSV file.*

For the images, there is a large number of formats availables. The most widespread used ones are TIFF, JPEG, GIF and PNG. TIFF is the most versatile one, but the number of subformats that exist makes it difficult to read on some software for the most exotic configurations. It is the preferred format for RAW plankton images to be processed by Zoo/PhytoImage.

JPEG is a lossy-compression format that is restricted to RGB 24-bit images only. It is the most efficient (lowest size of the file) for images that should only be viewed. However, the compression algorithm introduces artifacts in the picture that cannot be reliable analyzed when compression factor is too large. This format is reserved for vignettes (small images containing only one particle) when they are only used for visual classification of the particles (no further image analysis on them).

GIF and PNG are image format that use lossless-compression algorithms. PNG was proposed as an alternative to the older GIF format because of patent and licence problems : the non-free licence of the GIF format was a problem in the past, but now, the patent has expired and this format can be freely used. However, PNG being defined later, it offers more flexibility, like for instance, the possibility to define an alpha channel defining the transparency of the pixels in addition to their color. GIF can only tag one color as being fully transparent, and the other ones are fully opaque. In Zoo/PhytoImage, the PNG format can be used in addition to JPEG for vignettes if a lossless compression is required and the image has a chance to be further analyzed at a later stage after the vignette is created. This alternate format for vignettes was introduced in version 3 of Zoo/PhytoImage.

In Zoo/PhytoImage, data for one sample contain three components :

1. A case-by-variable table containing fetures extracted by image analysis on each blob (particles or individual items in the images). This table is stored in TSV format and in binary R own format .Rdata containing a data frame for quick loading in R,

2. A list of metadata information about the sample contained in a plain test file with an 'ini file' organization with one 'key=value' per line. The same information is also stored as attributes of the R's data frame.

3. A series of 'vignettes' that are cropped subsections of the initial images containing the picture of a single particle, and enhanced for visual identification. These vignettes were stored as JPEG images in version 1 and 2, but PNG format is now also accepted from version 3.

Since there can be easily thousands of particles, and thus vignettes, in one single sample, it is not convenient to keep all these items in separate files on disk. In version 1 and 2, Zoo/PhytoImage did compressed (zipped) these component in a single archive file with the ZID extension (for Zoo/PhytoImage Data). This approach is simple and ensures readability of the data since the unzip program required to extract the components is widely available. However, unzipping the archive to access the vignettes is a slow operation. This format prevents, thus a fluid and fast display of the vignettes for best user interaction and experience.

Starting from version 3, Zoo/PhytoImage now uses a custom binary format called ZIDB (for Zoo/PhytoImage DataBase). This format is indeed a hash table followed by binary versions of the different components. Fast C functions are used to access the different components for very fast retrieval of any vignettes, the features or the metadata. This format is a little bit less portable, but is easily accessible from R, and R itself is now widely available. In term of disk storage, the new ZIDB format is marginally (usually, around 5%) less compressed. So, you need an little extra storage space too.

Of course, a series of function have also be added to import data from the old ZID format, and to convert back and forth between the two formats.

3. Installation

3.1. Hardware requirement

Image analysis and automatic classification of images are computer-intensive processes, and you will likely analyze lots of objects (typically, hundreds of thousands, or millions of them). Thus, you need a recent and powerful computer to run Zoo/PhytoImage decently. Consider especially :

- A fast and recent multicore, and multithreaded processor.
- **4Gb of RAM memory** or more. Depending on the size of the images you want to analyze, you may need even more. Very large images issued from a flatbed scanner require at least 1Gb of RAM. Zooscan images may require even more! Nowadays, it is very easy to use 16Gb, or 32Gb of RAM on 64bit systems, so, consider this option seriously.
- After the processor speed and the RAM, the next most important part of your computer to work with images is **the graphic card and the screen**. Chose a rapid, optimized graphic card capable of displaying 1280×1024, or 1600×1200 pixels or more with 24/32bit color depth (millions of colors), associated with a high quality screen of no less than 19'. Dual-screen configuration can help too, since it gives more space for displaying side-by-side images and plots.
- Although Zoo/PhytoImage optimizes disk space by compressing all files, dealing with lots of high-resolution pictures is consumming a lot of space on disk. You will need a **fast hard disk of at least 2-4Tb of capacity**. One small SSD disk greatly improves the speed of the analysis when used to store the few samples currently manipulated.
- Finally, a good **backup system** is also required, unless you use a RAID system.

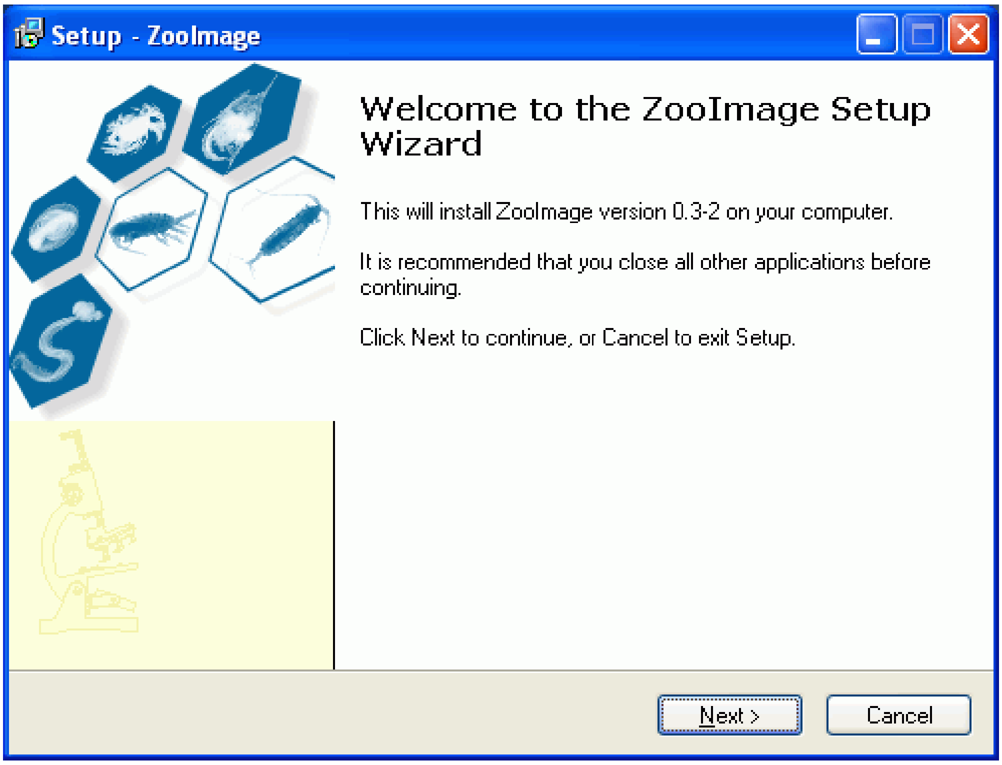
3.2. Download of the software

The software is available for download on the ZooImage website (http://www.sciviews.org/zooimage/). It can also be installed from within R, through CRAN : run `install.packages("zooimage")` from the R console.

Linux or Mac OS X users will have no problems installing R, and then Zoo/PhytoImage that way. The following section details installation from the Windows installer, as it exists since early versions (screenshots not updated).

3.3. Installation of Zoo/PhytoImage under Windows

Zoo/PhytoImage will uses about 400Mo of space on your hard disk, when installed. You just have to execute the "ZooImage_[x.y-z]Setup.exe" file that you downloaded and to follow the installer's instructions step-by-step. Default values for the options should be fine, if you don't understand them.



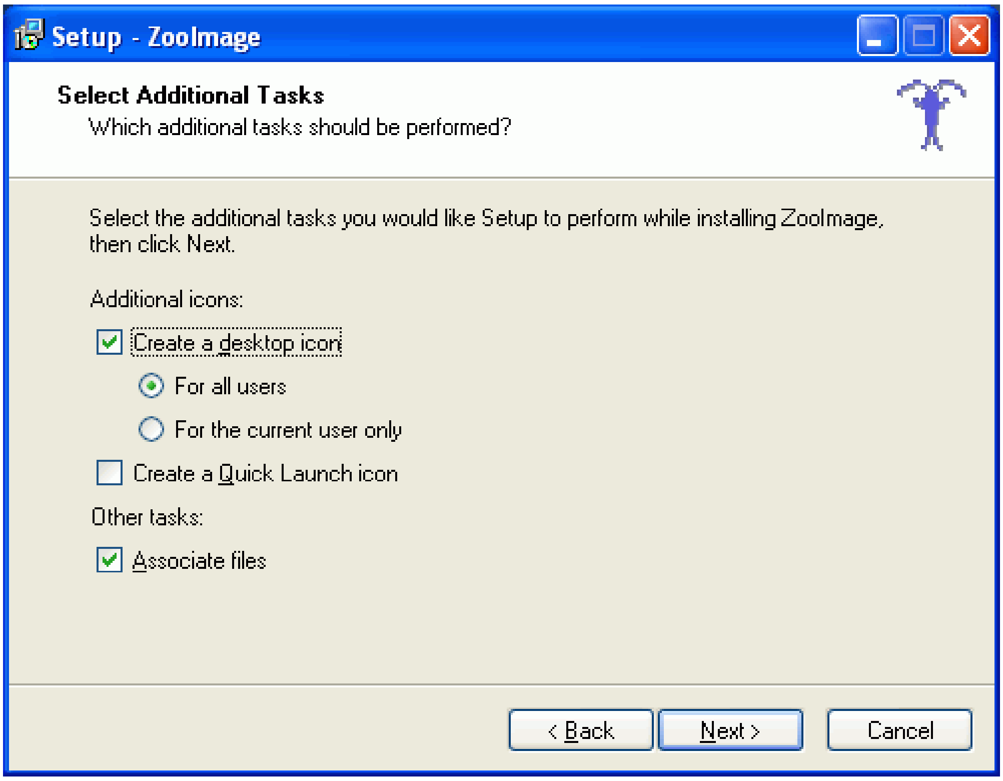
Welcome to the ZooImage Setup Wizard

This will install ZooImage version 0.3-2 on your computer.

It is recommended that you close all other applications before continuing.

Click Next to continue, or Cancel to exit Setup.

The first screen of the ZooImage installation assistant.



Select Additional Tasks

Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing ZooImage, then click Next.

Additional icons:

- ☑ Create a desktop icon
  - ● For all users
  - ○ For the current user only
- □ Create a Quick Launch icon

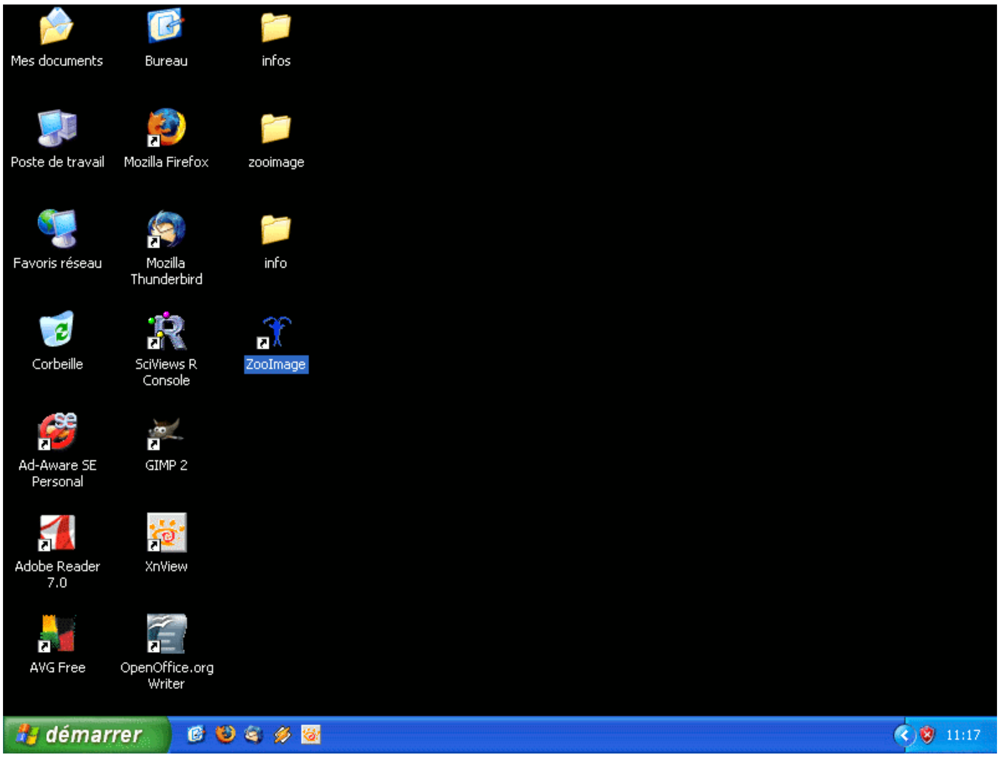
Other tasks:

- ☑ Associate files

Another screen of the Zoo/PhytoImage assistant. You can create desktop and quick launch icon (in the quick launch bar).

It is very important to ***associate files*** *with Zoo/PhytoImage: those files have special extensions and it will not be possible to open them by a double-click in the Windows explorer if you don't select this option. So,* leave this option checked unless you have good reason to change it!

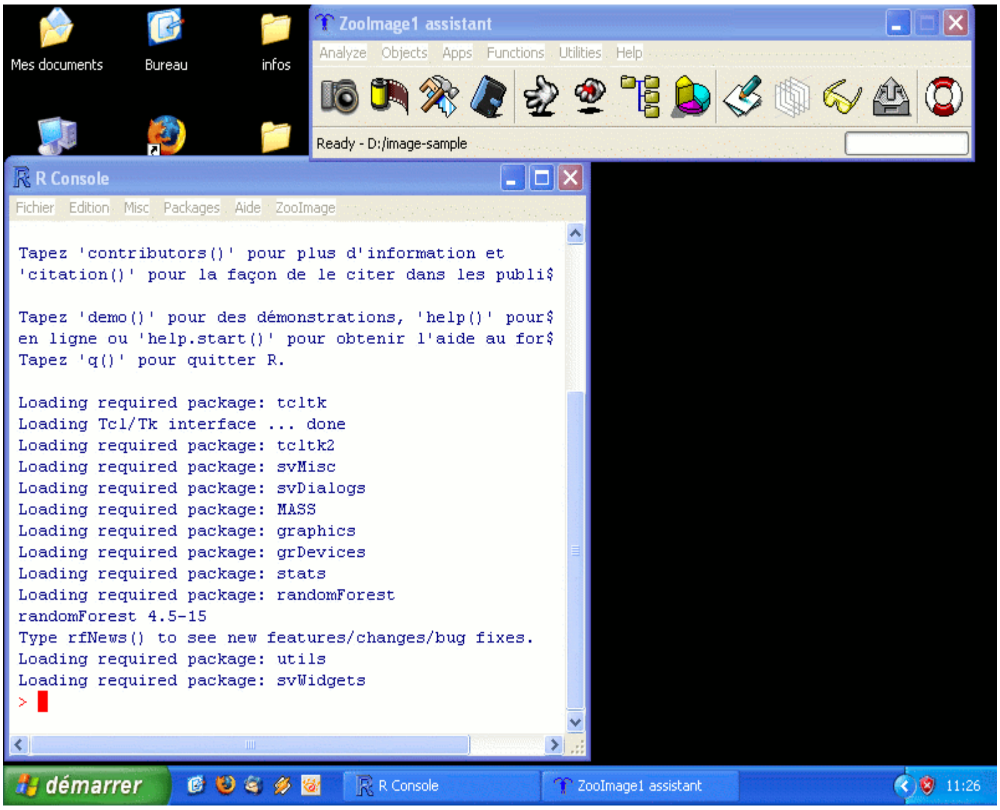
At the end of the installation, you should have a `ZooImage` entry in the start menu, and possibly a `ZooImage` icon on your desktop (if you left that option checked).



An example desktop with the `ZooImage` *icon (a little blue copepod) currently selected.*

4. First use of Zoo/PhytoImage

This quick tutorial will show you how to analyze the “Spain_Bioman” example images installed with the software. When you double-click on the `ZooImage` icon on the desktop, or select the `ZooImage (R)` entry in the start menu, two windows appear on screen: the **R console** and the **ZooImage assistant**.



```
Tapez 'contributors()' pour plus d'information et
'citation()' pour la façon de le citer dans les publi$

Tapez 'demo()' pour des démonstrations, 'help()' pour$
en ligne ou 'help.start()' pour obtenir l'aide au for$
Tapez 'q()' pour quitter R.

Loading required package: tcltk
Loading Tcl/Tk interface ... done
Loading required package: tcltk2
Loading required package: svMisc
Loading required package: svDialogs
Loading required package: MASS
Loading required package: graphics
Loading required package: grDevices
Loading required package: stats
Loading required package: randomForest
randomForest 4.5-15
Type rfNews() to see new features/changes/bug fixes.
Loading required package: utils
Loading required package: svWidgets
>
```

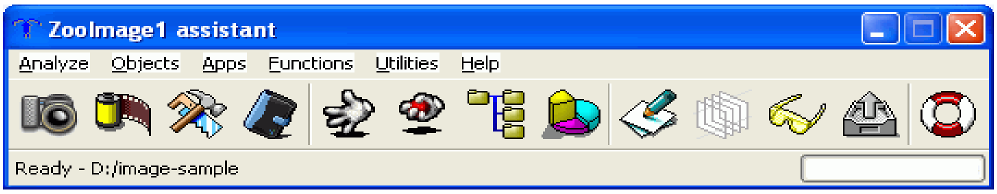
The two first windows appearing when you start Zoo/PhytoImage. At bottom-left, the R console (you can interact with R there) and at top-right the ZooImage assistant window.

The **R console** allows you to control R[2] directly through command lines. You should not worry about this window, unless you are familiar with the R language. However, it logs important results and messages from your actions in Zoo/PhytoImage. So, you are better not to minimize it.

The **ZooImage assistant** window is a toolbox with a menu on top and a status bar on bottom. It will guide you during the whole process. Basically, you just have to click on the buttons from left to right to run the various steps of your analysis.

A Zoo/PhytoImage analysis is subdivided in three parts, as it the toolbox. For each part, you have four buttons:

[2] R is the statistical software & environment on which ZooImage is based.



Ready - D:/image-sample

The three parts of the ZooImage process, materialized by three times four buttons. The last button shows the ZooImage user's manual.

- The first part deals with image importation and process.

  1. **Acquire images**. Start an external acquisition software (Vuescan, or any other program).

  2. **Import existing images**. Possibly convert the format of the images and/or rename them. If images are already in correct format, this function just make sure they have suitable *metadata* associated.

  3. **Process images**. Basically, ImageJ is started. You are supposed to used one of the ZooImage-specific plugins in ImageJ to process your pictures.

  4. **Make .zid files**. 'Zid' files stands for 'ZooImage Data' files. They contain all you need for the rest of the treatment, i.e., images of each individual[3], their measurements and the metadata. Yet, they store this information in a compressed way.[4]

- The second part help you to make an automatic classifier optimized for your zooplankton series.

  1. **Make a training set**. This function prepares a directory with a hierarchy of subdirectories representing your manual classification (you can freely modify this structure at will) and extract vignettes from the samples you want to use for making your manual training set. You then have to manually classify them on screen by moving them to their respective directories with the mouse.

  2. **Read training set**. Once you manually sorted the vignettes, this function collect this information into ZooImage. Statistics about you calssification (number of vignettes in each group) is the displayed.

  3. **Make classifier**. Use a manual training to train an automatic classifier. You have the choice of various algorithms. You got some statistics at the end of the process to evaluate performances of your classifier (cross-validation).

[3] These particular images are called 'vignettes' in ZooImage terminology.

[4] If you started with uncompressed high-resolution 16bit grayscale pictures in TIFF format, you usually end up with .zid files that weight about 100 times less than the original pictures.

4. **Analyze classifier**. Further analyses of your classifier's performances. Currently, only the confusion matrix showing differences between manual and automatic classification[5], is calculated. Other diagnostic tools will be added in future versions.

- The third part uses this classifier and the measurements done on all objects identified in your pictures (first part) to calculate automatically abundances, biomasses and size spectra in all your samples. You can then visualize results, or export them.

1. **Edit samples description**. Series of samples are identified by a list written in a specific Zoo/PhytoImage format. This list contains also further metadata about the series, and you have the opportunity to append various other measurements to the samples data (temperature, salinity, fluorescence, etc.).

2. **Process samples**. This is the workhorse function that process each sample of a given series one after the other, (1) identifying all individuals using your automatic classifier, (2) computing abundances per taxa, (3) calculating size classes in total and in each taxa for size spectra representations and studies, and (4) computing biomasses in total and per taxa, using a table of conversion from ECD[6] to cabon content, dry weight, etc. Data are converted per m^3, if suitable 'dilution' information is available in the metadata.

3. **View results**. Graphically present results. You can draw composite graphs (up to 12 different graphs on the same page), either time series of abundances or biomasses changes[7], or size spectra of given samples.

4. **Export results**. Results are written on the hard disk in ASCII format. This format is readable by any other software (Excel, Matlab, etc.).

> *Although you can export your results to analyze them in a different software, you don't* ***have*** *to do so. Zoo/PhytoImage operates in a R session, and the thousands of R functions are available for producing even the most sophisticated statistical analyses and graphs without leaving Zoo/PhytoImage/R.*

- **Manual**. Display the PDF version of the user's manual.

[5] The confusion matrix is shown both in tabular and in graphical presentations.

[6] ECD = Equivalent Circular Diameter.

[7] Spatial representations are not handled yet in this version, but they are planned in future versions.

5. Acquire digital images of zooplankton or phytoplankton

Zoo/PhytoImage is **not** a digitizing software. It is only designed to analyse existing digital images. However, for convenience, it binds to your favorite external acquisition software (it should be hardware-specific). As an example, if you use a digital camera with a dedicated capture software[8], you can specify that software in Zoo/PhytoImage and start it from the ZooImage assistant in one click.

Zoo/PhytoImage can be used with **Vuescan**, an excellent and very capable software to acquire pictures from more than 400 commercial flatbed scanners and from more than 100 different RAW formats of digital cameras. Here we explain how to use Vuescan with a flatbed scanner to get digital zooplankton images... **but it should be clear that it is just an example: you are free to use any hardware/software combination you like to acquire your images!**

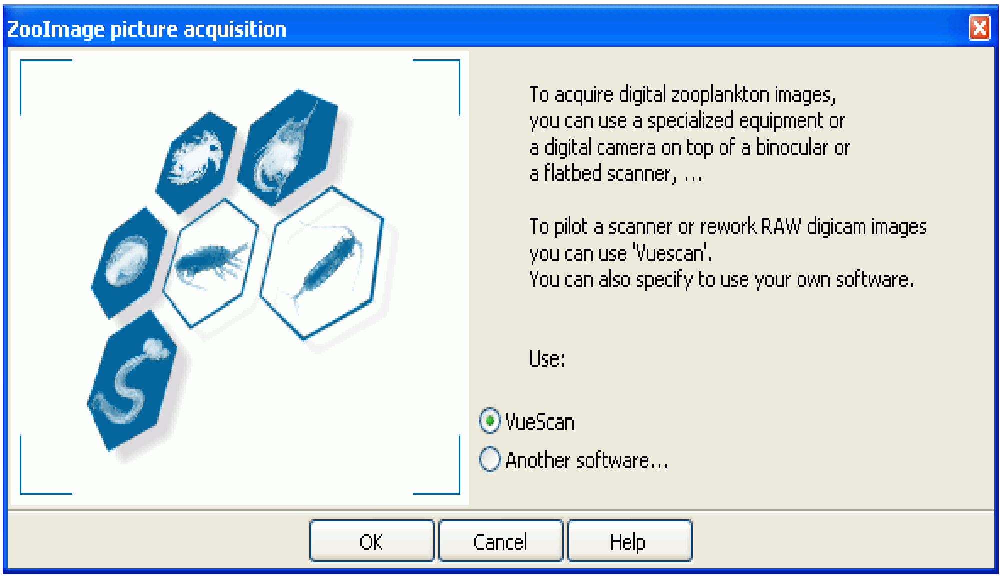
> *Vuescan is not a free software. It is a shareware distributed in two versions: personal and professional. You need the professional version. Its license is about $89, and you have to register your license with the author of Vuescan (see instruction in the Vuescan online help). We got the right to redistribute the trial version with ZooImage, but you have to unleach full features by entring your license code before you can use it in production.*

5.1. The 'acquire images' tool

> *In this manual, we use examples images installed with the software. So, you do not need to acquire your own image to practice with Zoo/PhytoImage. As an example, we show you how you can get your own images using Vuescan.*

To start your image acquisition software from the ZooImage assistant window, use the menu entry `Analyze → Acquire images...`, the shortcut `Ctrl+A`, or click on the first button in the toolbar.

[8] For instance, Canon or Nikon digital reflex camera are bundled with specific capture software that you can use to save directly your picture on your hard disk.



ZooImage picture acquisition

To acquire digital zooplankton images,
you can use a specialized equipment or
a digital camera on top of a binocular or
a flatbed scanner, ...

To pilot a scanner or rework RAW digicam images
you can use 'Vuescan'.
You can also specify to use your own software.

Use:

● VueScan

○ Another software...

You have a dialog box that let you choose the program to start (either Vuescan, or another one). Select Vuescan and click `OK`. Vuescan is opened. Once the software is registered, you can switch in `Advanced` mode by clicking on the corresponding button at the bottom (if Vuescan is started in `Guide me` mode). You have to parameterize Vuescan for your acquisition device (digital camera or flatbed scanner) and the type of images you want. Vueimage allows you to record both uncompressed TIFF files with 16bit gray levels and JPEG 24bit color files. These two types of files correspond respectively to the `Gray16bits 2400dpi` and `Color24bits 600dpi` plugins in ImageJ (see hereunder).

> *Vuescan offers a wide range of options for digitizing your pictures. A couple of options are very sensitive in the context of your image analysis. Additional documents are in preparation to list best Vuescan options for several digitizing devices (Zooscan, ...). You are also welcome to contribute your own recipe.*

6. Import images

Once your images are stored on your hard disk, you must prepare them for use in Zoo/PhytoImage. Use the menu entry `Analyze → Import images...`, the shortcut `Ctrl+I`, or click on the second button in the toolbar.

Zoo/Phytomage image importation is indeed performing several tasks to make sure your pictures are in correct formats and all required metadata are associated. In the current version, the function just checks the presence of metadata files, but more exhaustive control and processes are planned in future versions. **It means you have to do the rest of housekeeping manually!** Here is what you should do:

- Make sure that all the images you want to process are in one directory on your hard disk. Do not mix pictures you want to process with other ones on the same directory. **Keep them separate**. For instance, have one `d:\ImageProcess` directory where you store your fresh images and place them in one `d:\ImageDone` directory as soon as they are processed.[9]

> *Since Zoo/PhytoImage always starts from the current active directory when you have to browse for files and subdirectories, it saves time to switch it to the one where you store your raw images. The active directory is displayed in the status bar of the ZooImage assistant window. To change it, use the Utilities → Change active dir... menu entry.*

- Make sure your images are in a correct format: uncompressed TIFF with 16bit gray scale (preferably with a resolution of 2400dpi) for the *Gray16bits 2400dpi* plugin and 24bit color JPEG (preferably with a resolution of 600dpi and with the lowest compression level) for the *Color24bits 600dpi* plugin. Other file formats will be accepted in the future. Use general graphic utilities like Imagemagick or XnView to convert image that are not in one of these formats.

- Make sure you respect the naming convention imposed by Zoo/PhytoImage, which is:

  `SCS.YYYY-MM-DD.SS+PP.EXT`

  With this convention, the images are easily identifiable in a long series, both by the software and by the human. In particular, sorting files alphabetically results in a chronologial sorting of the images, according to sampling dates.

  1. `SCS` is the identifying code of the “Series - Cruise - Station”. Use three to four letters to identify the point within all you series/cruises/stations data.

[9] Once the images are completelly processed, you just need the resuting .zidb files somewhere on your hard disk. So, you can delete original pictures after making a backup on DVDs or external hard disks and save a lot of disk space!

2. YYYY-MM-DD is the date of sampling in year-month-day format. If for some reasons the day or the month is unknown, use 00.

3. SS is a code to uniquely identify each sample (useful when several samples are taken the same date at the same station).

4. PP is the image identifier. Zoo/PhytoImage manages different images per sample, and even, images of different fractions at different dilutions of the same sample[10]. Zoo/PhytoImage will carry all required calculations, including collecting together results from the six images in a single .zid file, calculating abundances and biomasses per m^3, taking into account the two fractions at different dilutions, etc.

5. EXT is the file extension according to the file format. It must be tif (lowercase) for TIFF images and jpg (lowercase) for JPEG pictures.

> *You do no have to conform to the Zoo/PhytoImage naming convention of the images. However, the minimum is to use NAME+PP.EXT with whatever string you want that uniquely identify one sample, being at least A if you have only one image per sample, and EXT as above. Thus, as a minimum, TIFF images should end with +A.tif and JPEG images with +A.jpg.*

That say, we will now practice on the example pictures.

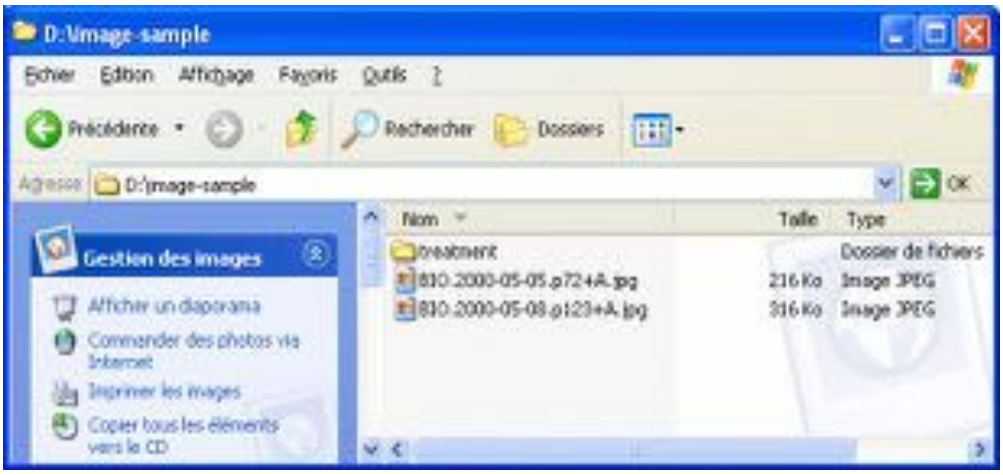
1. Prepare an empty directory on your hard disk (let’s say, `D:\image-sample`, but you can freely choose another partition or directory name).

2. Swith the active directory there, using the `Utilities → Change active dir...` menu and select that directory.

3. Copy the two example pictures `BIO.2000-05-05.p72+A.jpg` & `BIO.2000-05-08.p123+A.jpg` that are located in the `\examples\_raw` subdirectories of your Zoo/PhytoImage installation directory (by default, it is `C:\Program Files\ZooImage` on English versions of Windows) in that directory. **Do not copy corresponding .zim files.**

4. You should have something like this (without the treatment subdirectory):

[10] For instance, you filter you sample on a 1000µm sieve and apply different dilutions for the ‘large’ fraction and the ‘small’ one. Just decide to call your large fraction ‘A’ and your small fraction ‘B’. Now, if you make three pictures for each fraction, PP will be A1, A2, A3, B1, B2, B3, respectively for the six pictures related to the sample.

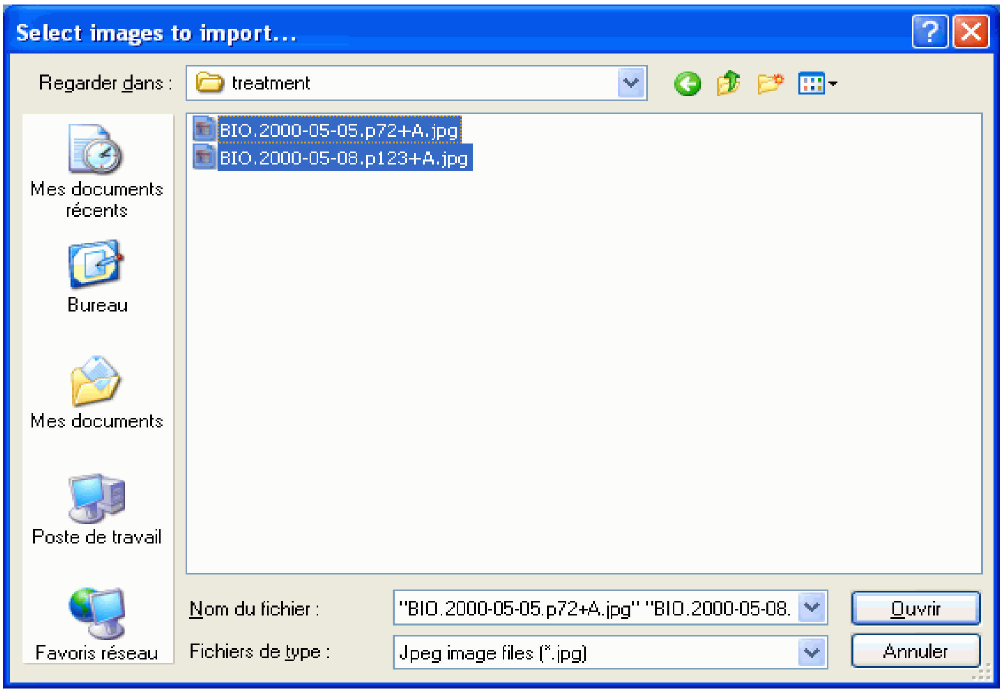


Adresse: D:\image-sample

Nom	Taille	Type
treatment		Dossier de fichiers
BIO.2000-05-05.p72+A.jpg	216 Ko	Image JPEG
BIO.2000-05-08.p123+A.jpg	316 Ko	Image JPEG

Now, click on the second button on the toolbar, the one with the following icon:

Zoo/PhytoImage asks you for the images that should be imported. Select both images.



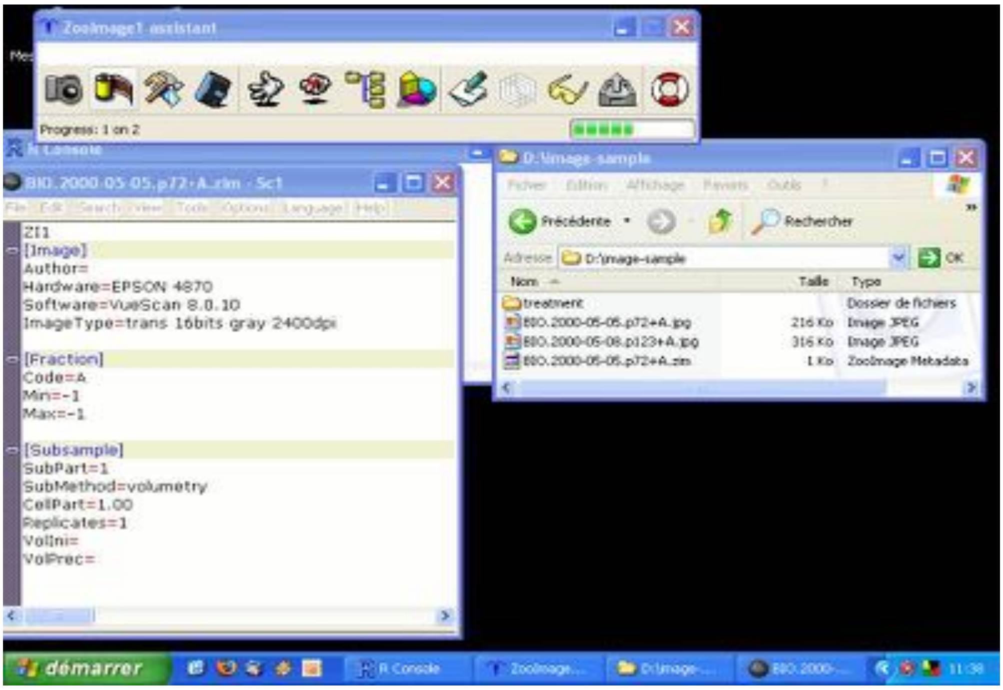
Regarder dans : treatment

BIO.2000-05-05.p72+A.jpg
BIO.2000-05-08.p123+A.jpg

Nom du fichier : "BIO.2000-05-05.p72+A.jpg" "BIO.2000-05-08.

Fichiers de type : Jpeg image files (*.jpg)

It is then supposed to check that image formats and names are correct, and possibly propose to change or convert them, but that feature is not implemented yet. It then checks if metadata files (files with .zim extensions) are associated. Since you did not copy these files with your images, they are not found and Zoo/PhytoImage creates them. It also displays their content in the built-in metadata editor (Sc1), each file in turn.



```
ZI1
[Image]
Author=
Hardware=EPSON 4870
Software=VueScan 8.0.10
ImageType=trans 16bits gray 2400dpi

[Fraction]
Code=A
Min=-1
Max=-1

[Subsample]
SubPart=1
SubMethod=volumetry
CellPart=1.00
Replicates=1
VolIni=
VolPrec=
```

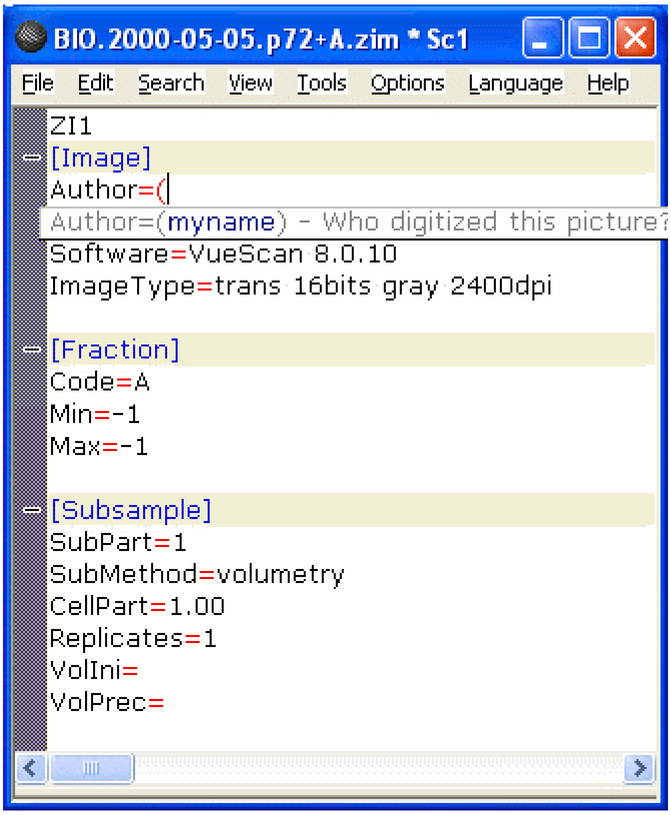
Nom	Taille	Type
treatment		Dossier de fichiers
BIO.2000-05-05.p72+A.jpg	216 Ko	Image JPEG
BIO.2000-05-08.p123+A.jpg	316 Ko	Image JPEG
BIO.2000-05-05.p72+A.zim	1 Ko	ZooImage Metadata

You are supposed to fill these data correctly. Here is how you can use the metadata editor:

- It is a plain text editor. Type your text as usual.

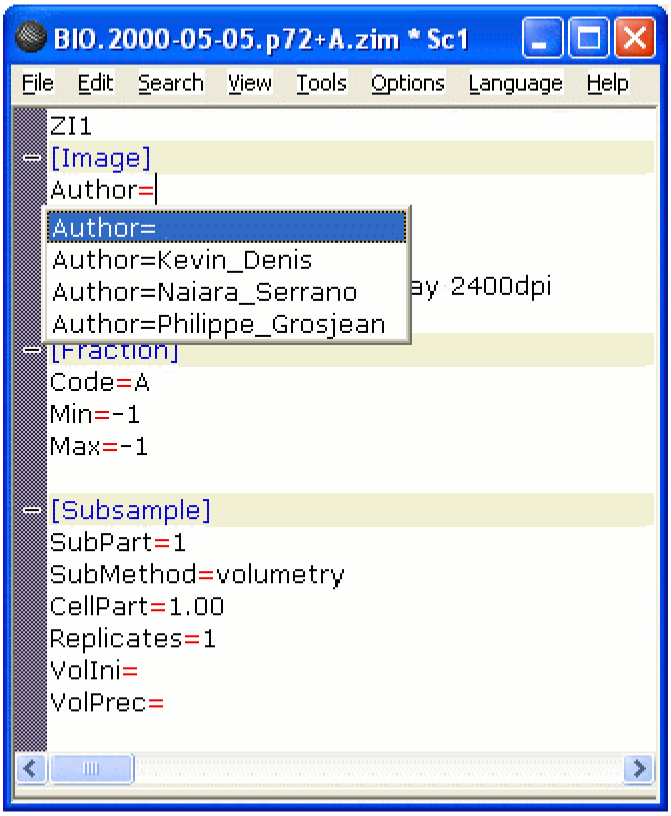
- You don't have to save your changes. When you close the window, changes are automatically saved and Zoo/PhytoImage switches to the next file.

- If you want help about a given entry, type an opening parenthesis just after the equal sign. You got a tip with information about that entry.



```
ZI1
[Image]
Author=(
Author=(myname) - Who digitized this picture?
Software=VueScan 8.0.10
ImageType=trans 16bits gray 2400dpi

[Fraction]
Code=A
Min=-1
Max=-1

[Subsample]
SubPart=1
SubMethod=volumetry
CellPart=1.00
Replicates=1
VolIni=
VolPrec=
```

- You can also have a list of proposition for that entry. Place the caret just after the equal sign and hit `Ctrl+I`. A list displays default entries. **This way of entering metadata should be preferred, because it avoids typing errors!**



```
ZI1
[Image]
Author=
Author=
Author=Kevin_Denis
Author=Naiara_Serrano          ay 2400dpi
Author=Philippe_Grosjean
[Fraction]
Code=A
Min=-1
Max=-1

[Subsample]
SubPart=1
SubMethod=volumetry
CellPart=1.00
Replicates=1
VolIni=
VolPrec=
```

- If needed, you can enter additional metadata. Just use the `key=value` syntax. If you want to create another topic, enter it in a separate line in square brackets like `[topic]`.

Zoo/PhytoImage does not create separate .zim files fo **each** picture. It only create separate .zim files for each fraction of each sample. So, if you have a lot of pictures related to the same sample and fraction (this is likely to be the case if you work with FlowCAM or VPR images), you just have to fill one .zim file for all of them!

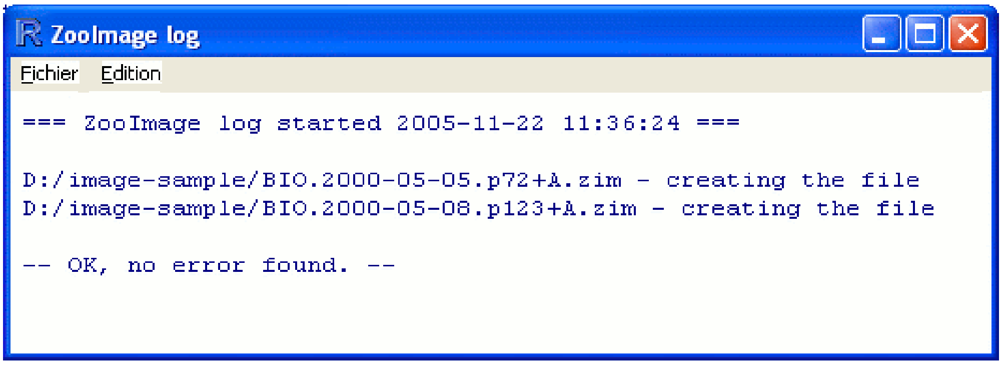
> *You can customize both the default entries in the metadata and the list of proposed values are customizable. Just edit those files: \bin\MetaEditor\templates\default.zim and ...\zim.api from the base Zoo/PhytoImage directory. Note that you cannot use spaces in the list of suggestions in the zim.api file. Use the underscore instead. ZooImage will convert it in a space in due time. So,* `Author=Alfred_Hitchcock` *should be entered in the list of possible completions, instead of* `Author=Alfred Hitchcock`.

Meaning of the metadata entries

Entry	Topic	Explanation
ZI1	-	This is not an entry. It just tells it is a ZooImage1 file.
Author	Image	Who digitized the picture?
Hardware	Image	Device used to digitize the picture.
Software	Image	Acquisition software and version.
ImageType	Image	Type of image. For instance `trans 16bits gray 2400dpi` means image acquired in transparency of 16bit gray scales and a resolution of 2400dpi.
Code	Fraction	The same fraction identifier as in the file name `A`, `B`, etc.
Min	Fraction	Minimum mesh size used to retrieve this fraction in µm. Use `-1` if none.
Max	Fraction	Maximum mesh size used to retrieve this fraction in µm. Use `-1` for none.
SubPart	Subsample	Part of the sample that was digitized. If the picture contains only 10% of the organisms in your sample, `SubPart=0.1`, for instance.
SubMethod	Subsample	Method used to get the part (`volumetry`, `Motoda`, `Falsom`, etc.)
CellPart	Subsample	Part of the cell containing the plankton that was actually digitized .

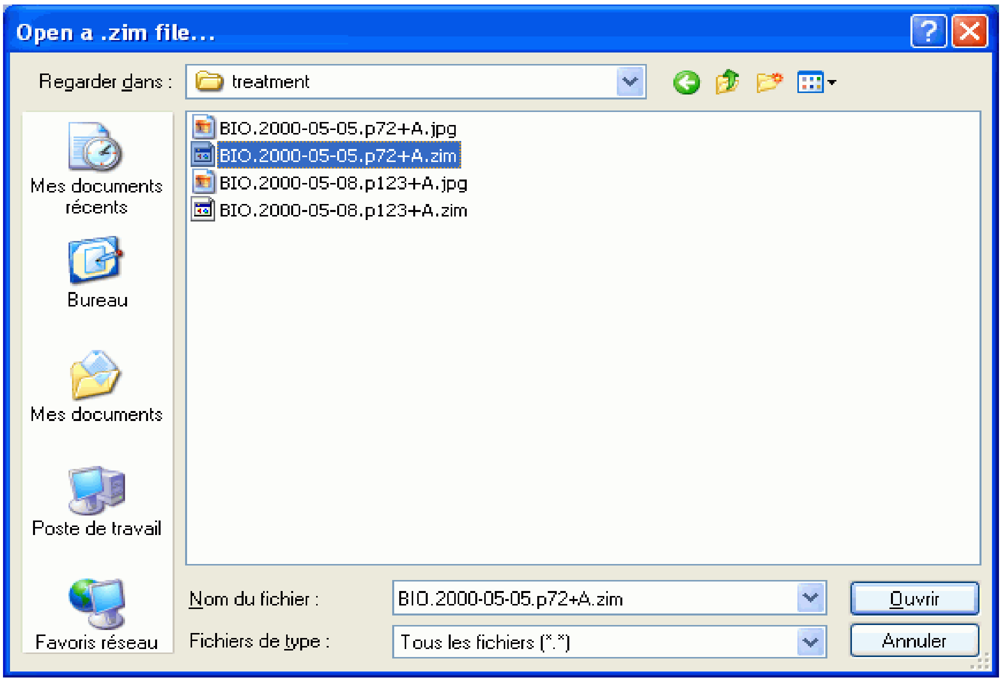
Replicates	Subsample	If you did replicated images with the same protocol for that fraction, how many replicates do you have? Note: ZooImage with average results among raplicates instead of summing them.
VolIni	Subsample	The volume of seawater that was sampled in m³. This is required to calculate abundances and biomasses per m³.
VolPrec	Subsample	The precision on the sampled volume estimate in m³. This will be used for error evaluation (not implemented yet).

At the end of the 'import' process, you should get a report in a `ZooImage log` window that pops up. It should look like this:



```
=== ZooImage log started 2005-11-22 11:36:24 ===

D:/image-sample/BIO.2000-05-05.p72+A.zim - creating the file
D:/image-sample/BIO.2000-05-08.p123+A.zim - creating the file

-- OK, no error found. --
```

Take care that you should have the `– OK, no error found. –` message at the end of the log. For only two pictures, this log is not very useful, but imagine the advantage of logging individual error if you import thousands of pictures and when all the checkings (file names, formats, etc.) will be activated! Now, you `D:\image-sample` directory should look like this:



Regarder dans : treatment

BIO.2000-05-05.p72+A.jpg
BIO.2000-05-05.p72+A.zim
BIO.2000-05-08.p123+A.jpg
BIO.2000-05-08.p123+A.zim

Nom du fichier : BIO.2000-05-05.p72+A.zim

Fichiers de type : Tous les fichiers (*.*)

7. Process images

To process your images, use the menu entry `Analyze → Process images...`, the shortcut `Ctrl+J`, or click on the third button in the toolbar.

Zoo/PhytoImage will now switch to ImageJ, a free image processing software. Before doing so, a dialog box proposes to close Zoo/PhytoImage. Whether you can leave Zoo/PhytoImage open at the same time as ImageJ or not depends on the amount of RAM memory required by the image process, compared to the one you got on your computer. The small example pictures we are dealing with do not require much RAM. So, if you have something like 512Mb on your machine, you should be safe to keep both Zoo/PhytoImage and ImageJ opened simultaneously. If you analyze very large pictures, you should close Zoo/PhytoImage and all other running programs **before** starting your image processing in ImageJ. As an example, 16bit gray pictures of 60 million pixels (for instance, 10000×6000 pixels) require 900Mb of RAM allocated to ImageJ[11]. You need at least 1Gb of actual RAM in your computer for dealing with such images.

> *The maximum amount of RAM you can allocate to ImageJ is system dependent. On 32bit system, do not try to allocate more than 1.6Gb to ImageJ[12], or the program will crash! Of course, you need at least 2Gb of actual RAM in your machine to use that maximum value. Although we did not tested the* `Gray16bits 2400dpi` *plugin with images larger than 10000×6000 pixels, the maximum allocatable RAM value should work with images of about 100 million pixels. Thus, currently the largest 16bit gray images you can deal with in ImageJ is something like 10000×10000 pixels[13]. At 2400dpi, it is a little bit less than 10x10cm of cell size. If you have larger cell area, just take several separate pictures and both ImageJ and Zoo/PhytoImage will take them into account (you just loose measurement on objects that are cut at the edges of the composite images). On 64bit systems, you don't have these limitations and should be able to analyze much larger pictures.*

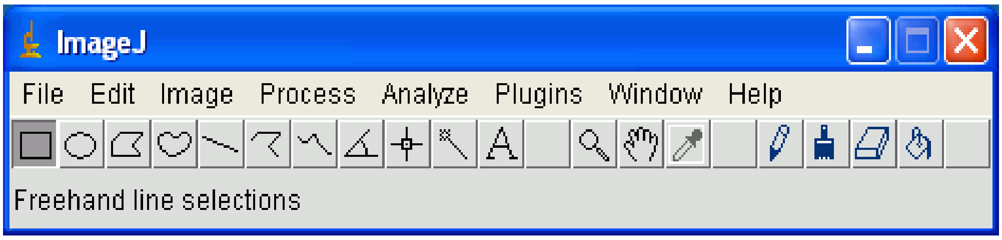
Start now ImageJ by click on the third button

Click `OK` in the dialog box and the ZooImage assistant window is minimized and replaced by the equivalent ImageJ main window as:

[11] The current configuration of ImageJ installed with Zoo/PhytoImage is to allocate a maximum of 900Mb to the program.

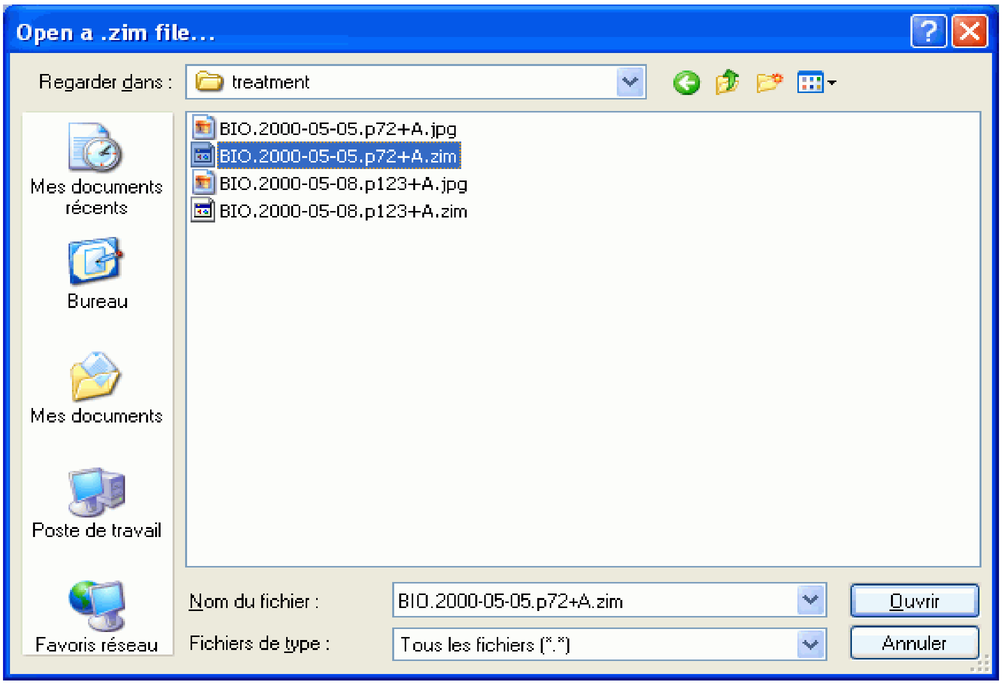
[12] You can change this value in ImageJ with the menu entry `Edit → Options... → Memory`. You have to restart ImageJ for the changes to take effect.

[13] With a different treatment, one could process larger images, but silhouette detection would be less accurate and there will be no background elimination.



Freehand line selections

Zoo/PhytoImage plugins are collected together in the menu `Plugins → ZooImage`. For our images, we have to select the `Color24bits 600dpi` plugin. The plugin first asks you to select a .zim file. **Do not select on image file here**.



Regarder dans : treatment

BIO.2000-05-05.p72+A.jpg
BIO.2000-05-05.p72+A.zim
BIO.2000-05-08.p123+A.jpg
BIO.2000-05-08.p123+A.zim

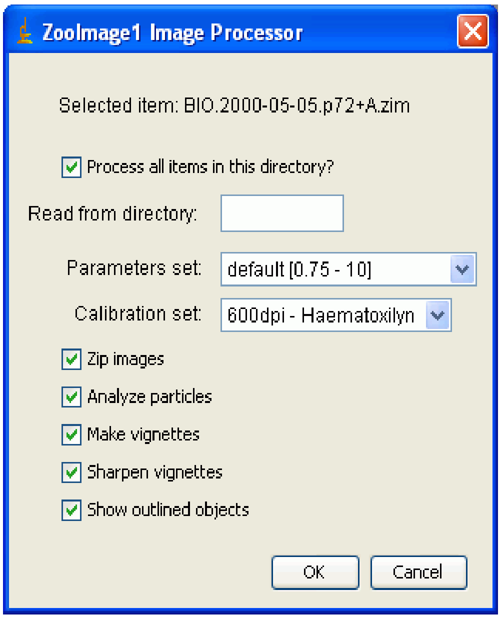
Nom du fichier : BIO.2000-05-05.p72+A.zim

Fichiers de type : Tous les fichiers (*.*)

The reasons why you have to select the .zim file instead of the corresponding image are:

- We are sure you have metadata associated with the image(s),
- As explained here above, you could have several images for the same sample/fraction. The plugin will process **all** images associated with the selected .zim file, not only one. In the example, we have only one image for each .zim file, but that feature is designed with FlowCAM or VPR images in mind.

  You then have a dialog box with parameterization of your process:



ZooImage1 Image Processor

Selected item: BIO.2000-05-05.p72+A.zim

☑ Process all items in this directory?

Read from directory:

Parameters set: default [0.75 - 10]

Calibration set: 600dpi - Haematoxilyn

☑ Zip images

☑ Analyze particles

☑ Make vignettes

☑ Sharpen vignettes

☑ Show outlined objects

- The name of the selected .zim file is displayed.

- You can **process all items in this directory** (all images that have associated .zim files), or only that one [*keep this checked now*].

- You can optionally **read images from a different directory**. This function is useful if you saved your large images on DVDs or external disks. You just have to copy the small associated .zim files in your process directory and you point to the directory that contains the images on your DVD [*leave this blank now*].

- The **parameters set** drop-down list allows you to select alternate configurations. Currently, alternate configurations are hard coded in the plugin, but users will be able to edit them freely in future versions. Parameters set defines minimum and maximum particle size to consider, which measurement is done, which threshold is used for separating particles from background, etc. [*leave the default value now*].

- The **calibration set** drop down list is similar to parameters set, but define calibration data, i.e., pixel size and calibration curves for grayscales and/or color channels, possibly depending on the lighting, staining of the sample, etc. [*leave the default value now*].

- **Zip images** rewrites the pictures in a zip-compressed TIFF format. This is not useful for JPEG images because they are already compressed. [*So, uncheck this option now*].

- **Analyze particles** do the measurements on the particles after processing the images [*leave this option checked now*].

- **Make vignettes** estract small images for each identified object, called 'vignettes' in Zoo/PhytoImage's terminology [*leave this option checked now*].

- **Sharpen vignettes** optionally applies a "sharpen" filter on the pictures in the vignettes. This often enhances the quality of the vignettes, but is not necessary for some kinds of pictures [*leave this option checked now*].

- **Show outlined objects** displays a composite image with the detected object outlines superposed to the grayscale image. This is a very useful diagnostic to determine if segmentation and detection of the objects was correct [*So, leave this option checked now*].

> *The **show outlined objects** option works only for the last picture processed. so, either uncheck **process all items in this directory**, or be prepared to wait for the last picture to get this diagnostic image! You should zoom in the image (`Image → Zoom → 100%` entry menu) and pan it by selecting the hand button and dragging the image content in the window to best see the result.*

When you start the process by clicking `OK` on the dialog box, ImageJ do the following work:

- It opens a `Log` window and reports its activity in it.

- It opens each image in turn, process it, and possibly measure particles and extract vignettes. You can follow the process on the screen. Note that a scale bar is added in the top-right corner of each vignette for convenience.

- It possibly displays the outlined objects of last picture if it was requested. also, the last table of measurements is left open for inspection.

> *If the process failed somewhere look if your images are of the right type, if they are not too big for the RAM memory allocated and if the correct plugin, parameters set and calibration set where selected. Look at the log file and the images produced in the `_work` directory to help you track the problem.*

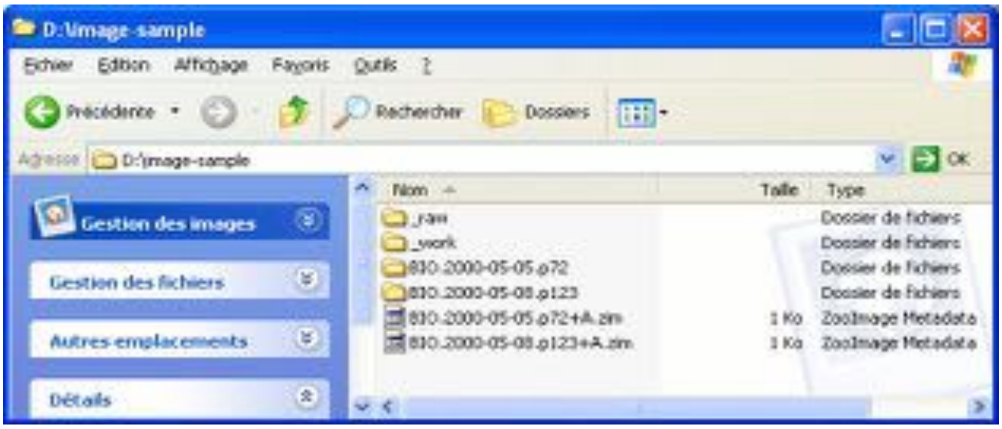
Always check the log file, seeking for errors, and take the habit to inspect outlines objects and table of measurements, at least, for the last image in your series. The plugins created several subdirectories in your process directory:

1. A **_raw** subdirectory contains raw images that were successfully processed.

2. A **_work** subdirectory contains temporary intermediary images left there for further inspection and diagnostic. Once you are satisfied with the treatment, you can delete the whole `_work` subdirectory to save

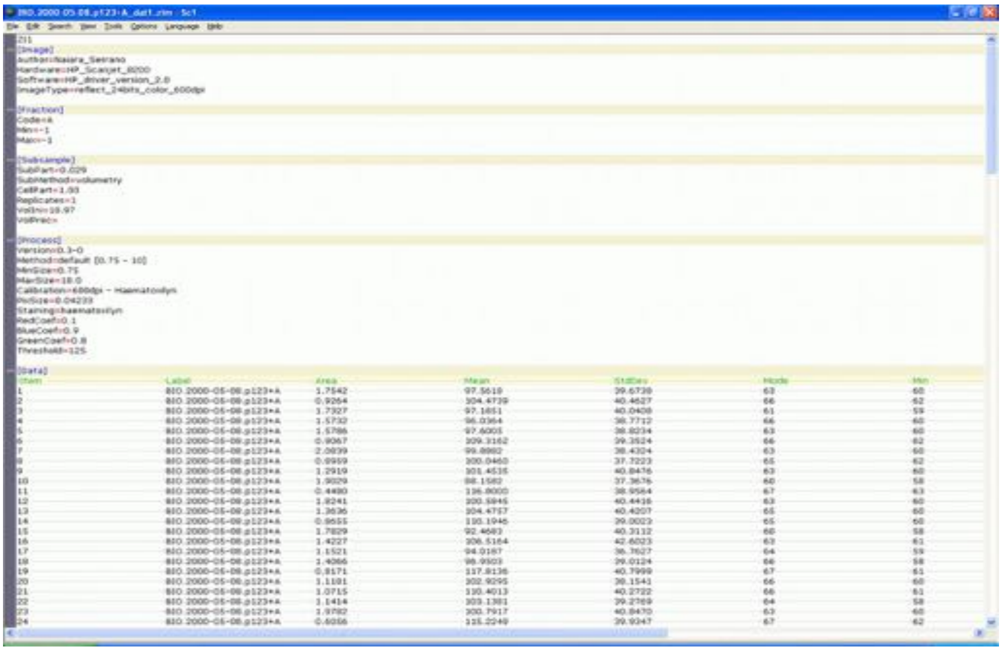
space on your hard disk.

3. One separate subdirectory for each sample, bearing the sample name (everything before the + sign in the images/.zim file names. This subdirectory contains all the vignettes for the sample (possibly combining various images and/or fractions) and `_dat1.zim` file(s) with metadata plus measurements for each image.



Nom	Taille	Type
_raw		Dossier de fichiers
_work		Dossier de fichiers
BIO.2000-05-05.p72		Dossier de fichiers
BIO.2000-05-08.p123		Dossier de fichiers
BIO.2000-05-05.p72+A.zim	1 Ko	ZooImage Metadata
BIO.2000-05-08.p123+A.zim	1 Ko	ZooImage Metadata

Here is how a `_dat1.zim` file looks like. Notice that you have two new sections appended at the end of your metadata: `[Process]` that gives information on the processing parameters used and `[Data]` with a table of measurements don on each particle.



```
ZI1
[Image]
Author=Naiara_Serrano
Hardware=HP_Scanjet_8200
Software=HP_driver_version_2.0
ImageType=reflect_24bits_color_600dpi

[Fraction]
Code=A
Min=-1
Max=-1

[Subsample]
SubPart=0.029
SubMethod=volumetry
CellPart=1.00
Replicates=1
VolIni=18.97
VolPrec=

[Process]
Version=0.3-0
Method=default [0.75 - 10]
MinSize=0.75
MaxSize=10.0
Calibration=600dpi - Haematoxilyn
PixSize=0.04233
Staining=haematoxilyn
RedCoef=0.1
BlueCoef=0.9
GreenCoef=0.8
Threshold=125

[Data]
!Item           Label                        Area        Mean        StdDev      Mode        Min
1               BIO.2000-05-08.p123+A        1.7542      97.5619     39.6739     63          60
2               BIO.2000-05-08.p123+A        0.9264      104.4739    40.4627     66          62
3               BIO.2000-05-08.p123+A        1.7327      97.1651     40.0408     61          59
4               BIO.2000-05-08.p123+A        1.5732      96.0364     38.7712     66          60
5               BIO.2000-05-08.p123+A        1.5786      97.6003     38.8234     63          60
6               BIO.2000-05-08.p123+A        0.9067      109.3162    39.3524     66          62
7               BIO.2000-05-08.p123+A        2.0839      99.8982     38.4324     63          60
8               BIO.2000-05-08.p123+A        0.6959      100.0460    37.7223     65          62
9               BIO.2000-05-08.p123+A        1.2919      101.4535    40.8476     63          60
10              BIO.2000-05-08.p123+A        1.9029      88.1582     37.3676     60          58
11              BIO.2000-05-08.p123+A        0.4480      136.8000    38.9564     67          63
12              BIO.2000-05-08.p123+A        1.8241      100.5845    40.4416     63          60
13              BIO.2000-05-08.p123+A        1.3636      104.4757    40.4207     65          60
14              BIO.2000-05-08.p123+A        0.9655      110.1946    39.0023     65          60
15              BIO.2000-05-08.p123+A        1.7829      92.4683     40.3112     60          58
16              BIO.2000-05-08.p123+A        1.4227      106.5164    42.6023     63          61
17              BIO.2000-05-08.p123+A        1.1521      94.0187     36.7627     64          59
18              BIO.2000-05-08.p123+A        1.4066      98.9503     39.0124     66          58
19              BIO.2000-05-08.p123+A        0.8171      117.8136    40.7999     67          61
20              BIO.2000-05-08.p123+A        1.1101      102.9295    38.1541     66          60
21              BIO.2000-05-08.p123+A        1.0715      110.4013    40.2722     66          61
22              BIO.2000-05-08.p123+A        1.1414      103.1381    39.2769     64          58
23              BIO.2000-05-08.p123+A        1.9782      100.7917    40.8470     63          60
24              BIO.2000-05-08.p123+A        0.6056      115.2249    39.9347     67          62
```

Once you have done with your image processing, you can close ImageJ and return to Zoo/PhytoImage (either restore the ZooImage assistant window, or restart the program, depending if you minimized or close it when you started ImageJ).

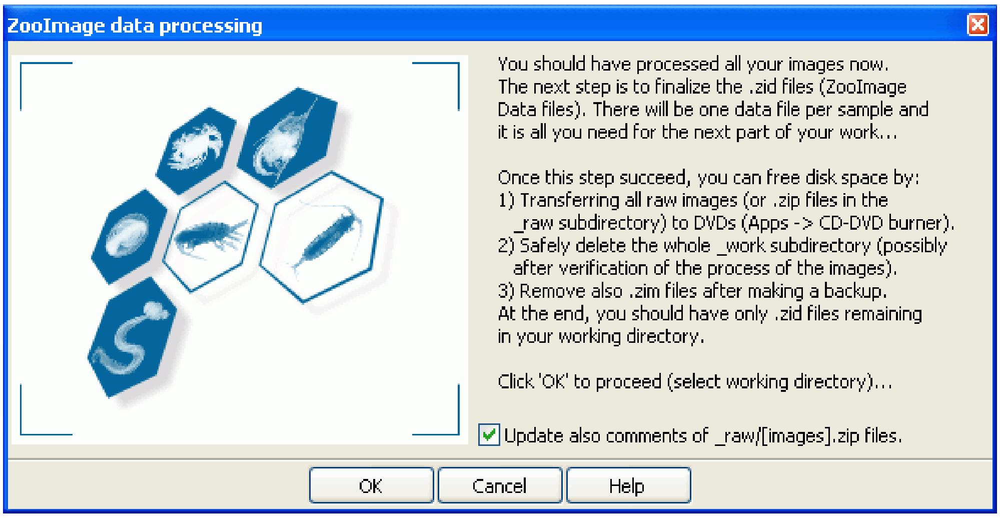
8. Create .zid files

To finalize your images import/process, you must now build .zid files. In the ZooImage assistant, use the menu entry `Analyze → Make .zid files...`, the shortcut `Ctrl+Z`, or click on the fourth button in the toolbar.

The first part of your analysis (import and process of your images) is almost done. You have now to create the **.zidb files**. These are special *ZooImage DataBase* files that contain all you need for the rest of the analysis, but saves as much disk space as possible[14]. Those .zidb files represent a convenient solution to keep all required data of even long series (thousands of samples) on a standard hard disk of 100-300Gb. In such a case, high-resolution raw images consume litterally **terabytes** of space and cannot be all kept on the hard disk at the same time! Just process your series bit by bit, and backup raw images from time to time to solve the problem.

Now, click on the fourth button in the ZooImage assistant:

This shows the following dialog box:



You should have processed all your images now.
The next step is to finalize the .zid files (ZooImage Data files). There will be one data file per sample and it is all you need for the next part of your work...

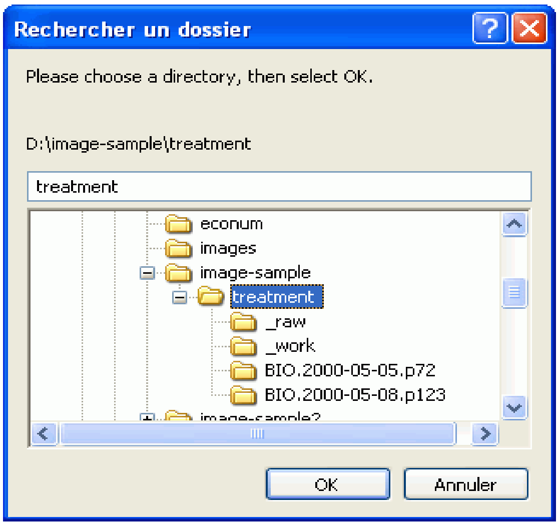
Once this step succeed, you can free disk space by:
1) Transferring all raw images (or .zip files in the _raw subdirectory) to DVDs (Apps -> CD-DVD burner).
2) Safely delete the whole _work subdirectory (possibly after verification of the process of the images).
3) Remove also .zim files after making a backup.
At the end, you should have only .zid files remaining in your working directory.

Click 'OK' to proceed (select working directory)...

☑ Update also comments of _raw/[images].zip files.

Instructions should be clear. By clicking `OK`, you compute .zidb files for your processed samples. The option **update also comments of _raw/[images].zip files** add .zim data as comments to zipped image files (if you selected that option in the process). [*Since we did not zipped images, we should uncheck that option now and click* `OK`]. You are prompted for a directory where treated data are located; give you working directory (`D:/image-sample/treatment`).

[14] You reach easily a compression factor close to 100 or more, starting with uncompressed 16bit TIFF images: 6 times 120Mb of raw images, that is, 720Mb compresses to 4-10Mb in the corresponding .zidb files!

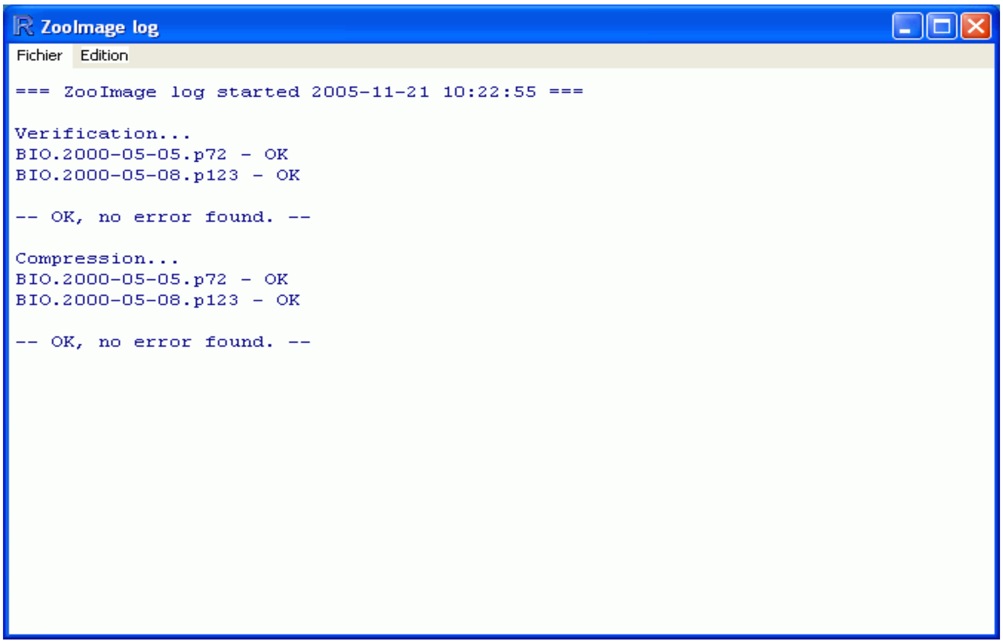


Please choose a directory, then select OK.

D:\image-sample\treatment

treatment

Zoo/PhytoImage computes .zidb files and issues a report at the end of the process. For convenience, it first quicky checks if all files are corrects. Stay in front of the computer during checking. Once it succeed, you can take a coffee break during the process that can be long if you processed a lot of samples. **Make sure there is no error reported once the compression is done.**



```
=== ZooImage log started 2005-11-21 10:22:55 ===

Verification...
BIO.2000-05-05.p72 - OK
BIO.2000-05-08.p123 - OK

-- OK, no error found. --

Compression...
BIO.2000-05-05.p72 - OK
BIO.2000-05-08.p123 - OK

-- OK, no error found. --
```

Cleaning the hard disk at the end of the process

Once all your .zidb files are created, it is time to save space on your hard disk. You should do the following from time to time:

1. Delete the `_work` subdirectory, once you are confident with the image processing of all your samples.

2. Back up your original images (in the `_raw` subdirectory) + the corresponding .zim files on DVDs, external hard disk, tapes, etc. **Always back up your raw image files: you would**

perhaps have to redo your analysis with a better algorithm in the future... and .zidb files do not contain required data for reprocessing the images! Once it is done, delete the `_raw` subdirectory and all remaining .zim files in the treatment directory to free disk space.

3. Check this: in your processing directory, you should only have .zidb files remaining (one per sample, no mather how many pictures you had for each sample) and no additional subdirectories or files (except, perhaps, .zis files and manual training sets if you already build them, see later in the manual).

9. Manually classifying vignettes

In order to train the computer to (semi)-automatically recognize zooplankton taxa on the basis of images measurements done in Zoo/PhytoImage, you have to make a manual training set. In Zoo/PhytoImage, you can have a relatively complex organization of the different groups (taxa, ecological groups, or any other grouping of the plankton that suits your needs) in a **hierarchical tree**. Hence, you have relationship between the groups (for instance, *Sapphirina intestinata* and *Sapphirina ovatolanceolata* are collected together in the *Sapphirina sp* group. *Copilia sp* and *Sapphirina sp* form your *Sapphirinidae* group. *Sapphirinidae* together with *Oncaeidae* and *Corycaeidae* (which contain also corresponding subgroups) are collected together in the *Poecilostomatoida*, etc. Up to the top group called *Copepoda*.

> *You can also decide to make other groupings, like ecological groups, or even mix the styles. You are here 100% free of the groups you create, but there are a couple of constraints: (1) make logical hierarchy of your groups and subgroups; (2) keep in mind the parameters (abundances, biomasses and partial size spectra) that you want to calculate on these groups; (3) make only groups where you can actually classify vignettes with a reasonable accuracy solely on the visual inspection of these vignettes; (4) it is useless to make groups for very rare items –you need at least ten to fifteen example vignettes in each group in your training set, 30 to 50 is even better–; (5) ultimately, the most pertinent grouping is the one that the computer can actually discriminate with a reasonable accuracy!*

You have to classify all kinds of items. Even those you are not interested in (may be, bubbles, marine snow, phytoplankton if you are only interested by zooplankton, etc.). Indeed, you have to recognize those items to elimine them from the countings... and you need a group in the training set for that!

You don't need to classify **all** vignettes. When you have about 50 items in a group and you think it is well representative of the overall variability in shapes of that group, you don't need to add more vignettes. Also, fuzzy objects, unrecognizable ones, multiple or part (except for VPR images), rare taxa, etc. do not need to be classified. Abberrant individuals which are not likely to occur often in your samples should be eliminated too. You have a special top group named '_' in the hierarchy for all these items. **All vignettes in the '_' top group or any of its subgroups will not be considered in the training set.**

For biomasses calculations, it could be useful to further split groups depending on the orientation of the animals: conversions formulas could be different for 'lateral' or 'dorso-ventral' views of the same animals. Make subgroups for them, if you want to take advantage of these different conversion formulas. Ex: *Oithona sp lateral* versus *Oithona sp dorsal*.

Make sure you use **unique names** for **all levels** of all groups. Do not use a classification like *Nauplius* subgroup in *Copepoda* and *Nauplius* subgroup in *Malacostraca*. Indeed, the program will manipulate groups independently for some treatments and how to differente *Nauplius* from *Nauplius* then, when you don't use the grouping hierarchy? Correct presentation should be: *Copepoda nauplius* in *Copepoda* versus *Malacostraca nauplius* in *Malacostraca*.

> *Zoo/PhytoImage does not check uniqueness of group names for the moment : you have to care about this by yourself!*

9.1. Preparing a manual training set from .zidb files

To install files and directories required for making a manual training set, use the menu entry `Analyze → Make training set...`, the shortcut `Ctrl+M`, or click on the fifth button in the toolbar.

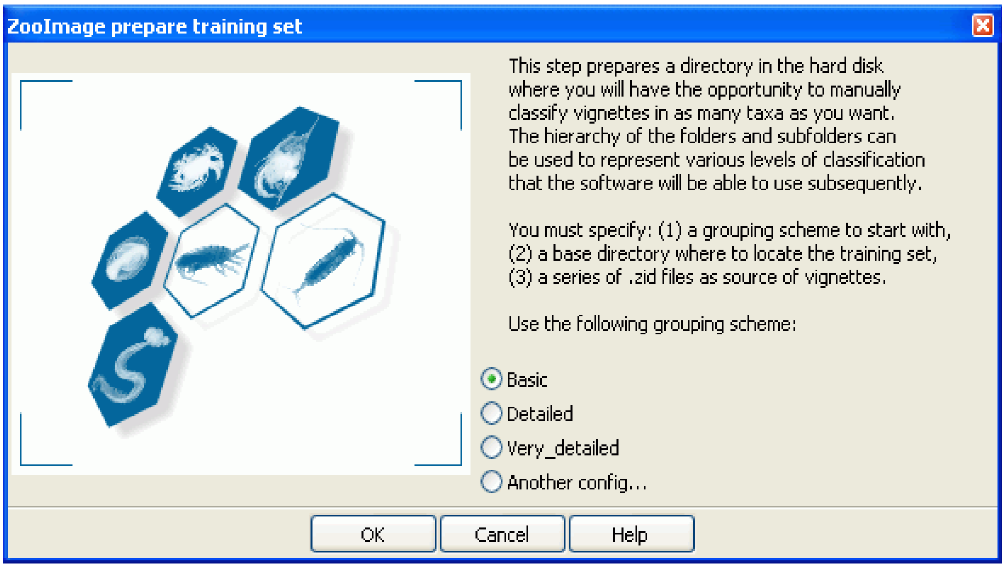
You must first decide which samples you will use in the training set. Select a couple of samples (i.e., a couple of .zidb files) that are representative of the whole variability in your series. Choose samples that span on the whole time scale (possibly several years) and the whole considered geographic area. Choose also samples collected at different seasons, if this applies. Depending on the number of groups you want to make you will need a couple of hundred vignettes to a couple a thousands of them (maximum 10 to 20.000 items for very detailed training sets). Knowing the average number of vignettes you have in a sample, you can determine how many samples you need (usually a couple a tens).

If you want to make your training set rapidly, starting with a long historical series already available in your laboratory, it could be interesting to first choose the representative samples that will be used in the training set and digitize them in priority. That way, you do not have to wait that all the samples in the series are digitized and processed to make your training set! Also, if different people are digitizing the sample (technicians) and making the training set (specialized taxonomists and biostatisticians), you could have work done in parallel once the few samples required for the training set are digitized.

To experiment with our example images, **create first an empty directory dedicated to this training set**. You can create it anywhere on your hard disk, but if you create a subdirectory in your process directory (`D:/image-sample`), make sure you **prepend its name with an underscore** (like `_train`, for instance). That way, ZooImage will ignore it in further processing of your images. Of course, do not use `_raw` or `_work` for the name of this subdirectory, since these names are reserved for the image processing treatment (see importing images). *[Create now an empty `_train` subdirectory in you processing dir].*

Now, click on the fifth button on the ZooImage assistant toolbar:

A dialog box with instructions appears on screen.



ZooImage prepare training set

This step prepares a directory in the hard disk where you will have the opportunity to manually classify vignettes in as many taxa as you want. The hierarchy of the folders and subfolders can be used to represent various levels of classification that the software will be able to use subsequently.

You must specify: (1) a grouping scheme to start with, (2) a base directory where to locate the training set, (3) a series of .zid files as source of vignettes.

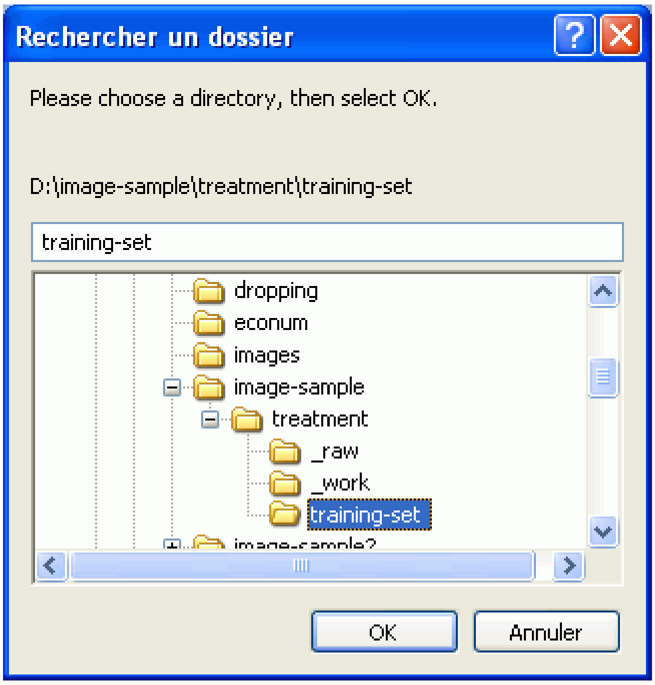
Use the following grouping scheme:

● Basic
○ Detailed
○ Very_detailed
○ Another config...

You have to select a config file. That file will create the initial hierarchy of groups as a series of subdirectories in your training set folder. You can choose “Basic”, “Detailed” and “Very detailed”, or select a different config file with a .zic extension. *[Choose now the “Basic” configuration and click* OK*].*

Initial groups config files are customizable, and you can save other ones everywhere on your hard disk. Just respect their (simple) syntax and save them with a .zic extension. `Basic.zic`, `Detailed.zic` *and* `Very_Detailed.zic` *files are located in the subdirectory* `\bin\R\R-2.2.0\library\zooimage\etc` *of the ZooImage root dir (usually* `C:_Program files\ZooImage`*).*

You now have to select the base **empty** directory where you want to install files and folders for your new manual training set:



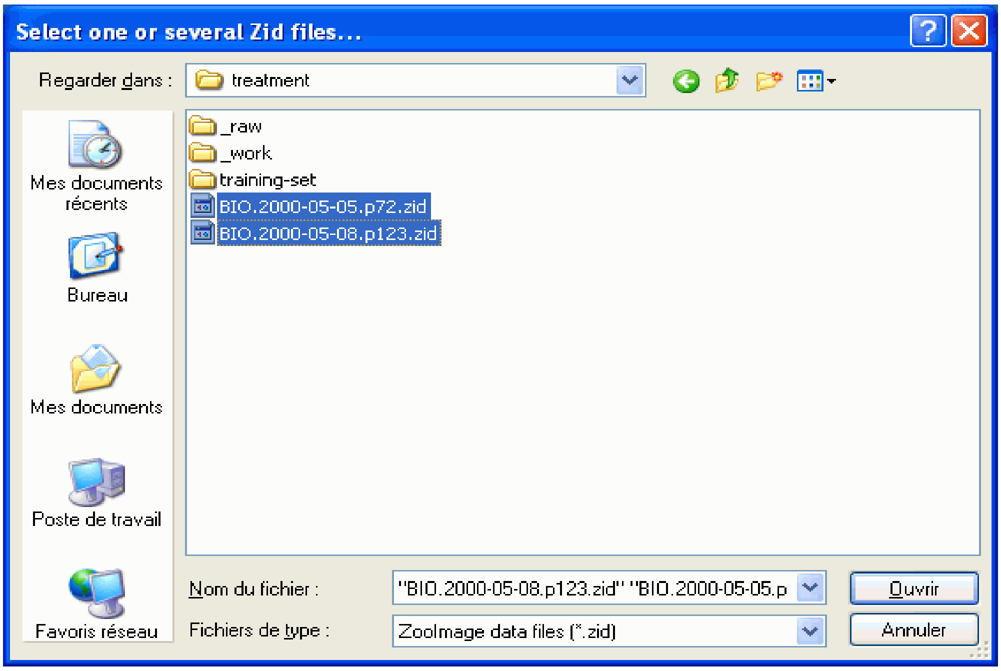
Rechercher un dossier

Please choose a directory, then select OK.

D:\image-sample\treatment\training-set

training-set

Select your `D:\image-sample\treatment\_train` directory. finally, the program asks you to select the .zidb or .zid files corresponding to the samples you want to use to build your manual training set (they must be all located in the same directory). *[Select now our two example samples* `BIO.2000-05-05.p72.zid` *and* `BIO.2000-05-08.p123.zid`*)].*



Select one or several Zid files...

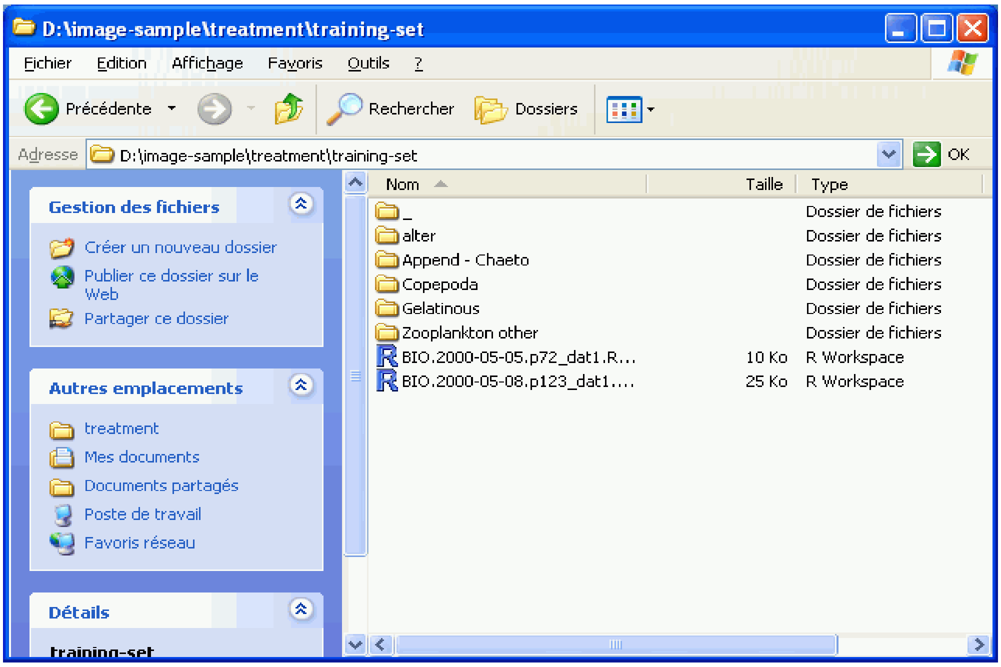
Regarder dans : treatment

_raw
_work
training-set
BIO.2000-05-05.p72.zid
BIO.2000-05-08.p123.zid

Nom du fichier : "BIO.2000-05-08.p123.zid" "BIO.2000-05-05.p

Fichiers de type : ZooImage data files (*.zid)

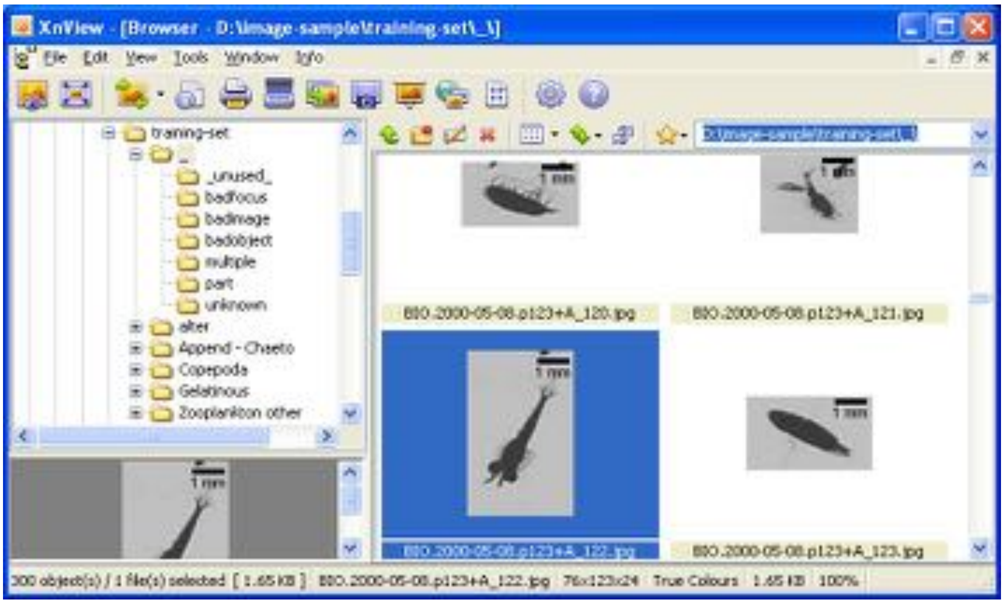
Zoo/PhytoImage creates required folders, extract data about these samples (`_dat1.Rdata`) files in the training set's root directory, and places all corresponding vignettes in the _ subdirectory. A log file indicates if there were errors creating these files and folders. At the end of the process, Zoo/PhytoImage starts **XnView** in the _ subdirectory. If you inspect the files on your computer, you should see something like this:



Adresse: D:\image-sample\treatment\training-set

Nom	Taille	Type
_		Dossier de fichiers
alter		Dossier de fichiers
Append - Chaeto		Dossier de fichiers
Copepoda		Dossier de fichiers
Gelatinous		Dossier de fichiers
Zooplankton other		Dossier de fichiers
BIO.2000-05-05.p72_dat1.R...	10 Ko	R Workspace
BIO.2000-05-08.p123_dat1....	25 Ko	R Workspace

Now, switch back to XnView.


XnView - [Browser - D:\image-sample\training-set_\]
File
Edit
View
Tools
Window
Info
training-set
_
unused
badfocus
badimage
badobject
multiple
part
unknown
alter
Append - Chaeto
Copepoda
Gelatinous
Zooplankton other
D:\image-sample\training-set_\
1 mm
BIO.2000-05-08.p123+A_120.jpg
BIO.2000-05-08.p123+A_121.jpg
BIO.2000-05-08.p123+A_122.jpg
BIO.2000-05-08.p123+A_123.jpg

300 object(s) / 1 file(s) selected [1.65 KB] BIO.2000-05-08.p123+A_122.jpg 76x123x24 True Colours 1.65 KB 100%

XnView is a free software for non commercial use[15]. It is both an image viewer/manager and an image converter. Here, we only use its ability to work with thumbnails of images in directories and manage them. We don't use all its features!

Depending how you organize XnView windows, the browser has a tree of directories, a thumbnail of images and a preview panel for the currently selected picture. You can change XnView configuration in `Tools → Options...`. If the directories tree is not visible, select `View → Folder`

[15] If you are in a private company, you will have to get a license for XnView before you can use it!

`Tree`. If you do not have a thumbnail view in the XnView browser main window (you can have an icon list, or tabular view of the files as well), select `View → View As → Thumbnails`. **Both the folder three and the main window in thumbnail mode are required for the rest of the work**.

Now, begin to classify the vignettes manually by moving them in the corresponding directory in the tree by drag&drop with the mouse. It is easier to move vignettes first in top directories (all copepods in *Copepoda*, all appendicularians and chaetognathes in *Append - Chaeto*, etc.). Then, you open the *Copepoda* subdirectory and classify vignettes from there to deeper levels (*Gymnoplea* or *Podoplea*), etc. Of course, this work should be done by, or with help of trained taxonomists.

> *It makes sense to ask different taxonomists to classify the **same** vignettes independently, so that you can check unmatching results and build a consensus that is supposed to bear less errors than a single manual training set. **We may add tools for analyzing and building consensus training sets in the future in ZooImage, but it is not the case yet in the current version.***

You are not restricted to the groups and subgroups already made. You can freely modify the structure of the tree; change directories, add or delete other ones. In the tree panel of XnView browser, you right-click in a directory and select `New Folder`, `Delete` or `Rename` entries to rework the tree. Make sure all people that build the training set (or similar training sets) have the same perception of each group. Define clearly which kind of object should go in which group, print these directives and keep them on your desk for reference when you classify your vignettes.

Also, if you plan to build a consensus training set, collecting together independently trained data, or if you want to build similar training sets for different series, you must work in two stages:

- First define the **structure of the tree** with all concerned people and define clearly which vignette should go in each group. At the end of the process, it should be useful to have a definition file (with a .zic extension) off this reworked tree. Distribute this .zic file to all collaborators and ask them to make their training sets with the same tree **without modifications**.

- Second, build your manual training set with the tree and groups you just defined.

When you classify your vignettes, you should try as much as possible to classify them down to the most detailed subgroups. If there are many vignettes you cannot classify deeper than a certain level, although your tree has more detailed groups, it means that you were too ambitious in the level of details you want to reach in the tree. Rework your tree and eliminate problematic subgroups where you cannot classify those vignettes.

A final pass is required before you can use your training set: you must rework or eliminate rare subgroups were you have too few items in them (let's say, less that 8-10 vignettes). Two alternatives:

1. Merge them with other subgroups, making less detailed groups, but with enough vignettes.

2. Decide not to include these rare groups in the training set. Keep them, but move the directories to the _ top folder (remember that this _ top folder contains all subgroups and vignettes that will be ignored in the classification).

> *Never forget that including rare groups in your training set will only have the consequence to reduce the total identification accuracy and the accuracy of other, major, groups –due to missclassification of other items in these rare groups–. The only (exceptional) situation where you would like to keep a rare group is when you are specifically interested by tracking target rare organisms in your whole set of images.*

When you rework your groups, make sure you do not have also **too many vignettes** in the most abundant ones. It is useless to have hundreds or thousands of items in one group. If it is the case, randomly eliminate vignettes (you can create the same group under the _ top folder and move the vignettes there, so that you keep them correctly classified but do not take them into account in the learning stage). Consider that if you have more than 50 vignettes in a group, you can begin to eliminate randomnly items down to 50 images per group.

Making a manual training set is a difficult and time-consumming task !

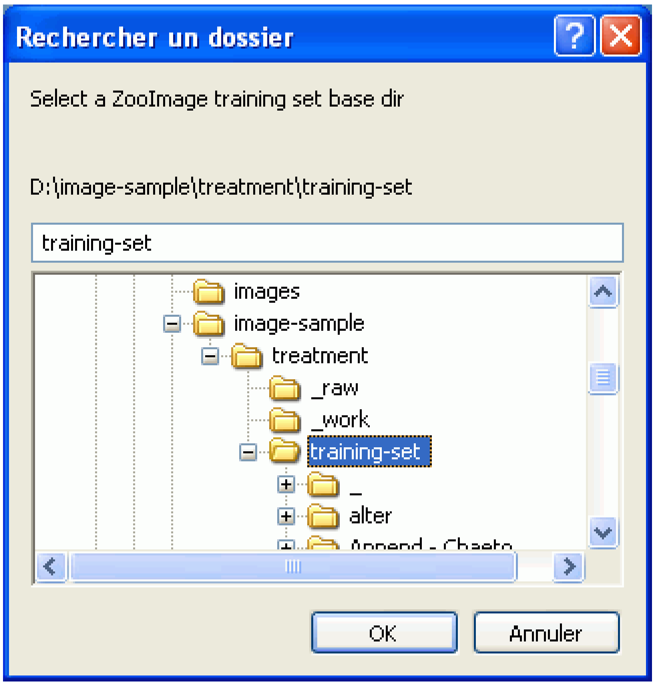
> *You have an example training set installed with Zoo/PhytoImage. You can inspect it in XnView, or even read it in Zoo/PhytoImage, if you like. This example training set is located in the* `\examples\_train` *subdirectory of your Zoo/PhytoImage folder (*`C:\Program Files\ZooImage` *by default on Windows). This training set was build using 29 samples… thus more than the two available in you* `\examples` *subdirectory. Look at it to have an idea on how you should balance items in the different groups.*

9.2. Reading a manual training from disk

To read a training set from directories where vignettes were manually classified, use the menu entry `Analyze → Read training set...`, the shortcut `Ctrl+T`, or click on the sixth button in the toolbar.

Once you are satisfied with your manual training set (or after reworking it, guided by the inspection of the confusion matrix, see hereunder), you have to read it in Zoo/PhytoImage. Click on the sixth button on the ZooImage assistant toolbar:

The program asks you for the top folder where your manual training set is located. *[Select now your directory, that is `D:\image-sample\treatment\_train`].*



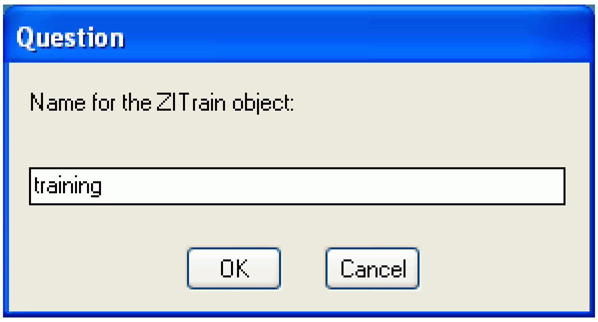
Rechercher un dossier

Select a ZooImage training set base dir

D:\image-sample\treatment\training-set

training-set

You are then prompted for a name to give to the 'ZITrain' object that will be created:

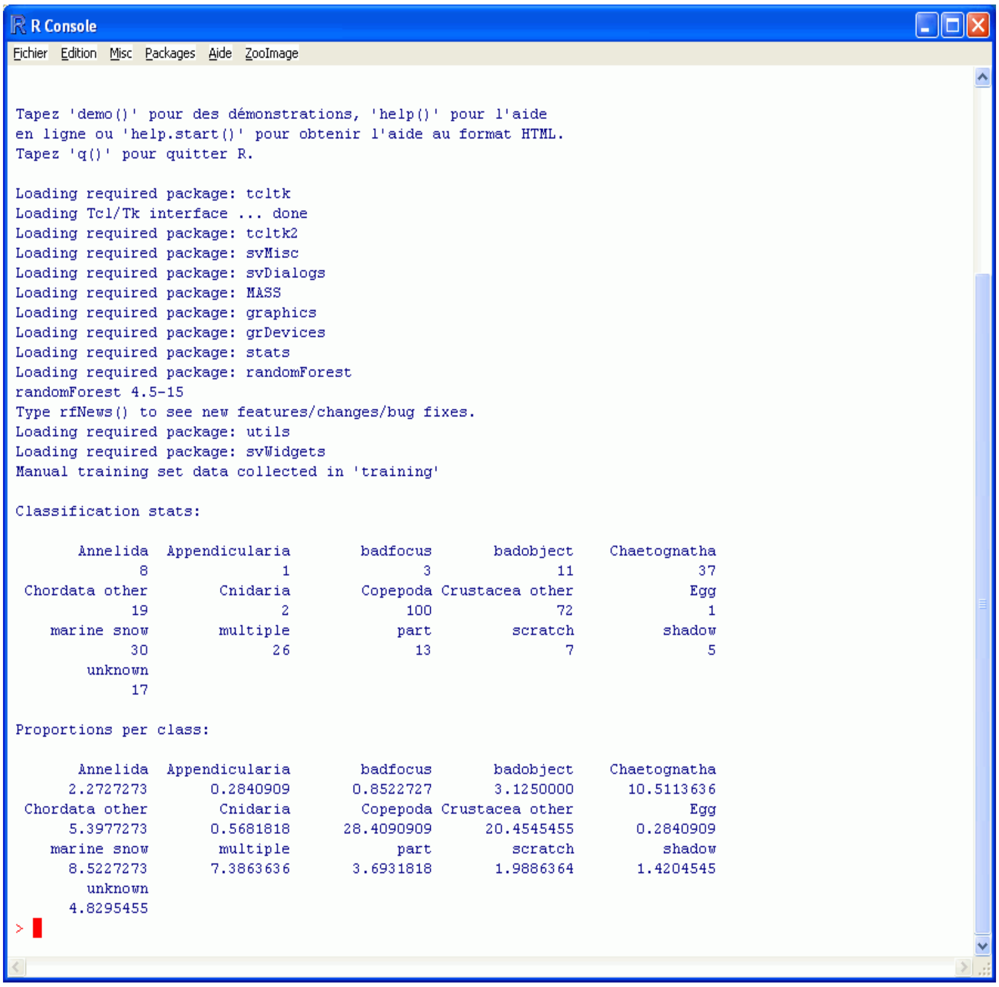


Question

Name for the ZITrain object:

training

[Call your object simply 'training' and click `OK`*].* Zoo/PhytoImage processes the tree (it takes a while for large training sets) and then displays basic statistics about your training set, that is, the number of vignettes in each group in the R Console window:



```
Tapez 'demo()' pour des démonstrations, 'help()' pour l'aide
en ligne ou 'help.start()' pour obtenir l'aide au format HTML.
Tapez 'q()' pour quitter R.

Loading required package: tcltk
Loading Tcl/Tk interface ... done
Loading required package: tcltk2
Loading required package: svMisc
Loading required package: svDialogs
Loading required package: MASS
Loading required package: graphics
Loading required package: grDevices
Loading required package: stats
Loading required package: randomForest
randomForest 4.5-15
Type rfNews() to see new features/changes/bug fixes.
Loading required package: utils
Loading required package: svWidgets
Manual training set data collected in 'training'

Classification stats:

       Annelida  Appendicularia       badfocus      badobject    Chaetognatha
              8               1              3             11              37
 Chordata other        Cnidaria       Copepoda Crustacea other             Egg
             19               2            100             72               1
    marine snow        multiple           part        scratch          shadow
             30              26             13              7               5
        unknown
             17

Proportions per class:

       Annelida  Appendicularia       badfocus      badobject    Chaetognatha
      2.2727273       0.2840909      0.8522727      3.1250000      10.5113636
 Chordata other        Cnidaria       Copepoda Crustacea other             Egg
      5.3977273       0.5681818     28.4090909     20.4545455       0.2840909
    marine snow        multiple           part        scratch          shadow
      8.5227273       7.3863636      3.6931818      1.9886364       1.4204545
        unknown
      4.8295455
> 
```

If you see that you have too much or too few items in some groups (like here, only one *Appendicularia* and a hundred *Copepoda*), go back to XnView and rework them before rereading your training set. Note that you have too few samples available in the examples for filling each group with enough items. For the rest of the demonstration, you can read the example training set installed with Zoo/PhytoImage as well.

10.Making and analyzing an automatic classifier

In Zoo/PhytoImage, classifier algorithms used range in a category called “machine learning”. Basically, you ‘feed’ the algorithm with example identifications together with measurements done on the same objects, and the algorithm learns how to recognize the groups according to the measurements. It is a very simple scheme, but it has proven efficient in many situations.

Many algorithms exist, and many are implemented in R over which Zoo/PhytoImage is running. The Zoo/PhytoImage dialog box gives access only to a couple of them. Moreover, in order to simplify the process, only default values are given for parameters. The solution you will obtain is, thus, often suboptimal.

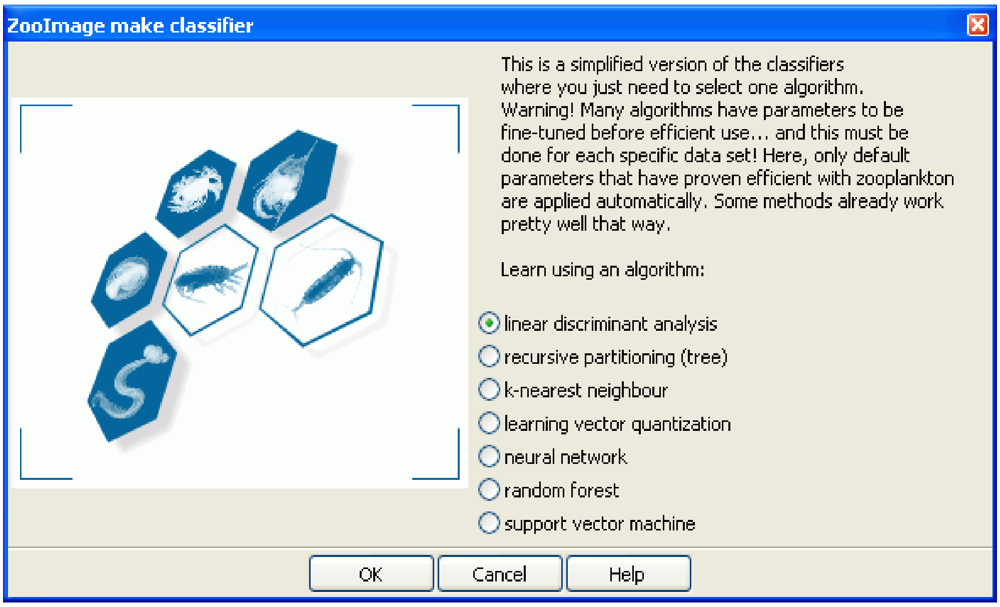
> *Many “machine learning” algorithms should be put in the “do not try this at home!” category. It means that you need a trained biostatistician to get the best from them and to analyze results to make sure they produce* ***consistent****,* ***reliable*** *and* ***accurate*** *identification of your plankton items. Everything was voluntary simplified in the Zoo/PhytoImage dialog box, just to give a flavor of these algorithm to everybody, and to allow a round-trip process of your data in an easy way.* ***Don’t be fooled by the apparent simplicity of the process using Zoo/PhytoImage dialog boxes!*** *For serious analyses, consider to fine-tune your classifier with a biostatistician that will use all the functions provided by R (he will program code in R’s native language, instead of just clicking with the mouse on a few options in the dialog box). There is no warranty on the results, and we would not endorse responsability of the consequences for false results published after using “uncertified” ‘toy’ classifiers!*

10.1.Training a classifier

To train an automatic classifier with you manual training set, use the menu entry `Analyze → Make classifier...`, the shortcut `Ctrl+C`, or click on the seventh button in the toolbar.

Having a ‘ZITrain’ object in memory, you can now create a ‘ZIClass’ object, that is, an automatic classifier that learns how to recognize your zooplankton based on the examples you give in your manual training set. Click on the seventh button on the ZooImage assistant toolbar:

The next dialog box appears. It displays a warning message about the simplified learning phase and proposes a variety of “machine learning” algorithms to use.

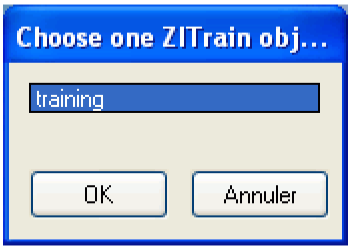

ZooImage make classifier

This is a simplified version of the classifiers where you just need to select one algorithm. Warning! Many algorithms have parameters to be fine-tuned before efficient use... and this must be done for each specific data set! Here, only default parameters that have proven efficient with zooplankton are applied automatically. Some methods already work pretty well that way.

Learn using an algorithm:

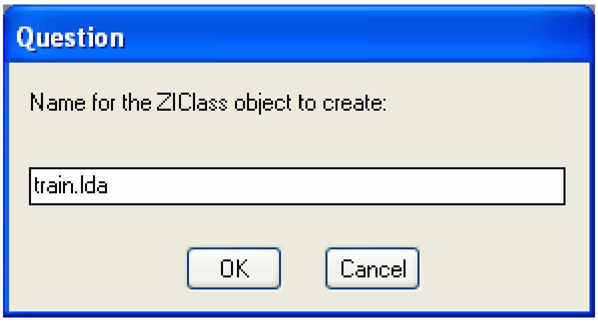
● linear discriminant analysis
○ recursive partitioning (tree)
○ k-nearest neighbour
○ learning vector quantization
○ neural network
○ random forest
○ support vector machine

Choose the one you want to use. *[Now we will use the simplest algorithm: linear discriminant analysis. select it and click* `OK`*]*. The program asks then which ‘ZITrain’ object he should use. You have probably only one training set in memory: the `training` object you just created.


Choose one ZITrain obj...

training

[Select it and click `OK`*]*. The program then asks for a name for the ‘ZIClass’ object that is about to be created.


Question

Name for the ZIClass object to create:

train.lda

[Enter `train.lda` *and click* `OK`*]*. The algorithm learns how to recognized your zooplankton, based on your manual training set. When it is done, its performances are assessed using a method called “10-fold cross-validation”. Then, a summary of the results (total accuracy and error by group) is reported to the R Console.

If you want, you can now test and compare other algorithms with the same training set. Also, if you notice that one or several groups have consistently high errors, it means they are not well separated. Could you consider reworking them in the context of your analysis? Look also at the confusion matrix (hereunder) for further diagnostic tools.

10.2. Analyzing classifier performances

Further diagnostic tools are provided to study the performances of your classifier, use the menu entry `Analyze → Analyze classifier...`, the shortcut `Ctrl+N`, or click on the eigth button in the toolbar.

Having a 'ZIClass' object in memory, you should calculate a **10-fold cross-validated confusion matrix** between your manual and the automatic classification. The confusion matrix is a square matrix that compares all groups af the manual classification with all groups of the automatic classification. The number of items in each cell corresponds to the counting of objects. The diagonal (from top-left to bottom-right) corresponds to cells where both identifications are the same. This is thus the counting of **correctly** predicted items. All cells outside of the diagonal depict disagreement in both classifications. They are usually attributed to errors done by the automatic classifier, starting form the hypothesis that there is **no error** in the manual training set.

To calculate and display the confusion matrix for your classifier, click on the eigth button on the ZooImage assistant toolbar:

... and select your 'ZIClass' object in the dialog box *[You probably have only one, so, select it and click `OK`]].*

According to those analyses, you could decide to rework the groups that are difficult to separate in your manual training set, to reread it and train a new classifier with these optimized groups.

> *Other diagnostic tools are also accessible from the same dialog box in version >= 4. Experiment by yourself with it and discover the different diagnostic plots available here...*

11. Manipulating Zoo/PhytoImage objects

You don't have, of course, to read manual training sets and train classifiers again and again each time you launch Zoo/PhytoImage. You can **save** and **restore** existing objects. The `Objects` menu provides functions to do so:

- `Objects → Load` reloads one or several objects form a .RData file. The .RData file is a binary format that is used by R to save its variables. You can save several objects in the same file, and thus, you reload them all at once in this case. The .RData files can be exchanged between computers, even on different platforms (for instance, .RData files generated on Windows are totally compatible with those made on Linux/Unix or MacOS X).

- `Objects → Save` gives you the opportunity to select one or more 'ZIxxx' objects (Zoo/PhytoImage specific objects) present in memory, and to save them in a file.

- `Objects → List` prints the list of all Zoo/PhytoImage objects currently in memory.

- `Objects → Remove` permanently deletes one or several objects from memory. Consider using this command to free memory if you created a lot of objects that you don't need any more.

> *The .RData files are very convenient to exchange training sets and toroughly-tested classifiers with your colleagues. Everything is included in the .RData files to reuse those manual training sets and/or these classifiers on a different computer.*

R has a mechanism to save and restore automatically all objects in memory when you quit the program and restart it from the same active directory. When you quit R (`File → Exit` on the R Console, or click the close button of the R Console), you have a question: "Save workspace image?" that appears. If you click `No`, R exists without saving anything. If you click `Yes`, it saves the data in the file .RData in the current active directory (the one reported in the status bar of the ZooImage assitant window). It also saves the history of commands in a .Rhistory file in the same directory. The next time you start R, you can restore this .RData file if you like. **It is far better to use the `Objects` menu and selectively save/restore given objects than to systematically rely on this mechanism!** This way, you can also choose a meaningful name and directory where you store your data! So, if you save your objects using the `Objects` menu of Zoo/PhytoImage, you can systematically answer `No` to "Save workspace image?" when you quit R/ZooImage.

12. CALCULATING, VISUALIZING AND EXPORTING SERIES

This section supposes that you have already made .zidb files from your raw images (part I) and that you have a valid 'ZIClass' object in memory (part II) either that you just created, or that your reloaded from a .RData file.

Up to now, all treatments were made at the sample level. You never had more than one sample loaded in memory. A sequence of samples (or images) was always treated one-by-one by Zoo/PhytoImage, possibly reporting long processes in a log file, so that you can leave the software unattended doing the calculation and come back later to see the results (it seems that the koffee room will be more crowded that usual. **This is a feature!** Zoo/PhytoImage is not designed as a toy program that would be just able to calculate a couple of demo examples, but that will crash with an "out-of-memory" message with any serious dataset!

When we speak about serious datasets in the field of zooplankton image analysis, it really means:

- **Terabytes of raw images** to process[16]. Since you can backup your raw images and ZooImage cares about **storing highly compressed data in .zidb files**, you can really process very large series containing thousands, or even tens of thousands of samples with a simple PC. You can store, indeed, all these tens of thousands .zid files in a single hard disk of 200-300Gb[17].

- **Almost unlimited number of images per sample**, and also possibly, **complex samples processes** with replicates and with various separate fractions (different dilutions, or even, different processes for each fraction[18]). Zoo/PhytoImage will perform all the calculations: averaging replicates, adding data from the fractions after applying corrections for different dilutions, and rescaling results to express them per square meter of seawater automatically.

- **Almost unlimited number of objects in each samples** (the current limit is probably around a few hundreds of thousands items per sample, that is, the size of a matrix R can store in memory at once with a 2-4Gb RAM computer). This is not really a limitation because a few thousands to a few tens of thousands of objects are enough to evaluate the composition of a single sample, even for relatively rare taxa (with 10.000 objects measured in a sample, even rare taxa representing 1% of the sample composition will be represented by about 100 individuals).

Of course, processing time is in proportion with the size of the series, but Zoo/PhytoImage proposes various mechanisms to recover after a fealure

[16] The only limitation is currently the maximum allocatable memory of 1.6Gb in ImageJ under 16bit systems that limits the size of **one** image to 100 millions of pixels. But 64bit systems, currently available today, overcome that limitation. Otherwise, Zoo/PhytoImage allows an almost unlimited number of images per sample.

[17] A typical .zid file with 2000-3000 objects weights only about 5Mb.

[18] For instance, using a Zooscan for the large fractions and a FlowCAM for the smaller ones.

to process a sample, and the error is reported in the log file. So, it is possible to spot the error and to reprocess only the guilty sample(s) later on[19].

So, OK, it seems relatively easy to accumulated huge amount of data using Zoo/PhytoImage. But then, how do we digest this huge quantity of information? The third part of the analysis deals with the calculation of biologically meaningful statistics that summarize each sample: abundances, biomasses and size spectra (total or per taxa). Hence, from the measurement of a couple of thousands objects in your images, you summarize the information into a few tens of numbers for each sample. All these numbers are then collected in a single table, with one line per sample. These tables are stored in 'ZIRes' objects (Zoo/PhytoImage Results). They are most suitable for the space-time analysis at the series level, which can be done in R/ZooImage directly, or you can export the tables to analyze them in another software like Matlab, for instance.

12.1.Creating and documenting a series

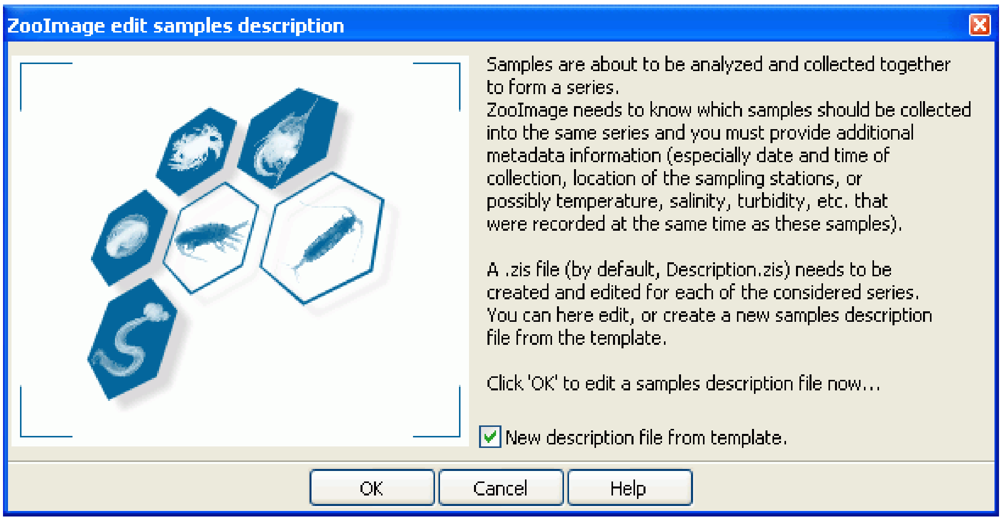
A series is a collection of samples plus a few additional metadata. To edit a series description file (.zis file), use the menu entry `Analyze → Edit samples description...`, the shortcut `Ctrl+D`, or click on the nineth button in the toolbar.

Until now, your .zid files had independent lives, totally ignoring each other. It is now time to tell to Zoo/PhytoImage which .zid files you want to collect together in a space-time series. This is done by editing a samples description file with a .zis extension. You can create as many .zis files as you like, making thus different series (for instance, a variation in time at a single station for one series; a spatial coverage of the area at a given time for another series, etc.).

[As an illustration of this principle, you will create now a mini-series, collecting together the two example samples we are analyzing]. Click on the nineth button on the ZooImage assistant toolbar:

The following dialog box appears with an explanatory message and a single option:

[19] Note that Zoo/PhytoImage does not have yet a mechanism to incrementally add data to a 'ZIRes' object, but that mechanism is planned for future versions.



Samples are about to be analyzed and collected together to form a series.
ZooImage needs to know which samples should be collected into the same series and you must provide additional metadata information (especially date and time of collection, location of the sampling stations, or possibly temperature, salinity, turbidity, etc. that were recorded at the same time as these samples).

A .zis file (by default, Description.zis) needs to be created and edited for each of the considered series. You can here edit, or create a new samples description file from the template.

Click 'OK' to edit a samples description file now...

☑ New description file from template.

You can either create a new description file from the template (check the option), or edit an existing one (uncheck it). *[Create a new file, and thus, leave the option checked and click* `OK` *now]*. After telling where you want to store the description file, the MetaEditor opens a template. You have to fill it in order to tell to Zoo/PhytoImage which samples are included in the series.

> *The .zidb files corresponding to all samples included in the series are supposed to be in the same directory as the .zis files themself.*

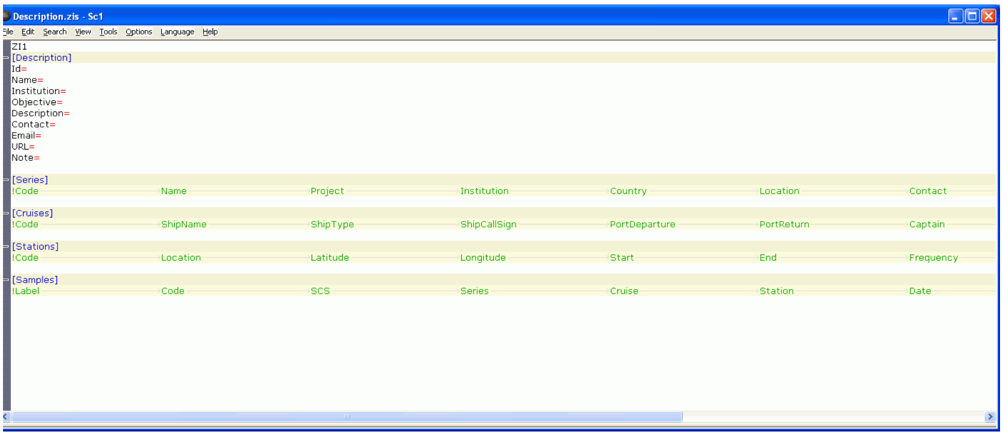
A complete description of data and metadata in .zis files is found in the annexes. You do not have to fill all field. Also, you can add additional keys, if you want. Major fields that you **have to fill correctly** are:

Key	**Section**	**Comment**
Id	Description	The short identifiant of the series.
Name	Description	A longer name for this series.
Description	Description	A short description of the series[20].
Contact	Description	The name of a responsible person of this series.
Email	Description	The email address of the contact.
Label	Samples	The complete label of the sample, as in the file names.
Code	Samples	A code for this sample.
Date	Samples	The data of sampling (in `yyyy-mm-dd` format).
Latitude	Samples	The latitude of sampling (in `+/-x.xx`).
Longitude	Samples	The longitude of sampling (in `+/-x.xx`).

[20] Don't overlook these metadata : they will allow you to calculate abundances and biomasses per water volume in the field, to locate your samples in space or time for further analysis, etc.

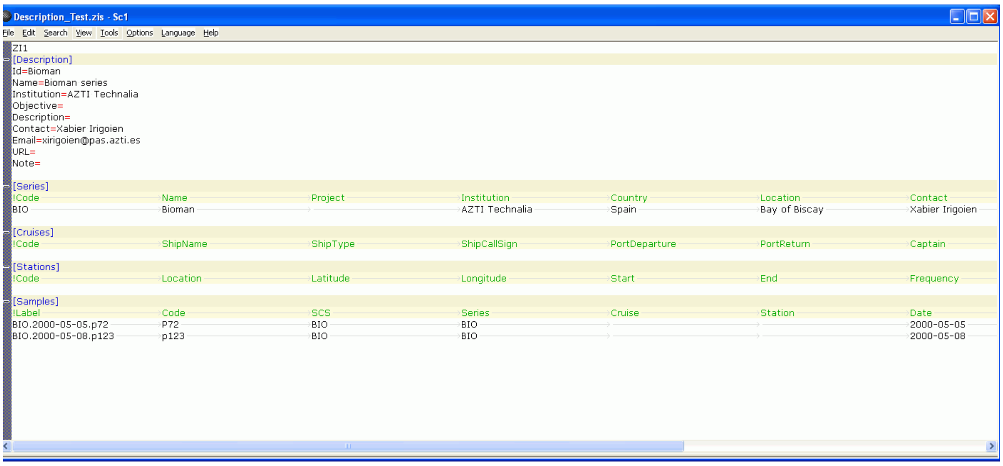
> *Considering the large amount of fields in this file, it would be convenient to reimplement it in a database. Any volunteer to reprogram this part of the software in an Open Source database like MySQL out there?*

The MetaEditor displays the `Description.zis` template.



```
ZI1
[Description]
Id=
Name=
Institution=
Objective=
Description=
Contact=
Email=
URL=
Note=

[Series]
!Code        Name         Project      Institution   Country        Location       Contact

[Cruises]
!Code        ShipName     ShipType     ShipCallSign  PortDeparture  PortReturn     Captain

[Stations]
!Code        Location     Latitude     Longitude     Start          End            Frequency

[Samples]
!Label       Code         SCS          Series        Cruise         Station        Date
```

You have to fill it to obtain something like this:


Description_Test.zis - Sc1
File Edit Search View Tools Options Language Help

```
ZI1
[Description]
Id=Bioman
Name=Bioman series
Institution=AZTI Technalia
Objective=
Description=
Contact=Xabier Irigoien
Email=xirigoien@pas.azti.es
URL=
Note=

[Series]
!Code                Name         Project      Institution     Country        Location       Contact
BIO                  Bioman                    AZTI Technalia  Spain          Bay of Biscay  Xabier Irigoien

[Cruises]
!Code                ShipName     ShipType     ShipCallSign    PortDeparture  PortReturn     Captain

[Stations]
!Code                Location     Latitude     Longitude       Start          End            Frequency

[Samples]
!Label               Code         SCS          Series          Cruise         Station        Date
BIO.2000-05-05.p72   P72          BIO          BIO                                           2000-05-05
BIO.2000-05-08.p123  p123         BIO          BIO                                           2000-05-08
```

You can just close the window, and your changes are saved automatically.

12.2. Calculating samples

To process all samples in one series, use the menu entry `Analyze → Process samples...`, the shortcut `Ctrl+S`, or click on the tenth button in the toolbar.

To process all samples in a given series, click on the tenth button on the ZooImage assistant toolbar:

… and select the corresponding .zis file *[Select your* `Description.zis` *file]*. The program then asks to select a classifier. *[Select your* `train.lada` *object]*. You have also to specify the limits for the different size classes to consider for the size spectra. The default value creates a regular sequence from 0.25mm to 2mm with a class width of 0.1mm (`seq(0.25, 2, by = 0.1)`). If you clear this entry, the program understands that you do not want to calculate size spectra for these samples. *[Keep default values and click* `OK` *now]*.
The last question is a name for the `ZIRes` object to create. *[Give* `results` *and click* `OK` *now]*.

> *We still have to implement the table of parameters for the biomass conversion in the program!*

Zoo/PhytoImage calculate each sample in turn and generate a log file. Once the process is done, you should get a log file indocating that there is no error.
Your `ZIRes` object is now created (if no error occur; look at the log). If there are arrors, the most probable cause is a problem in the `Description.zis` file, or corresponding .zid files that are not located in the same directory as the .zis file. Make the corrections and start the analysis again.

12.3. Visualizing results

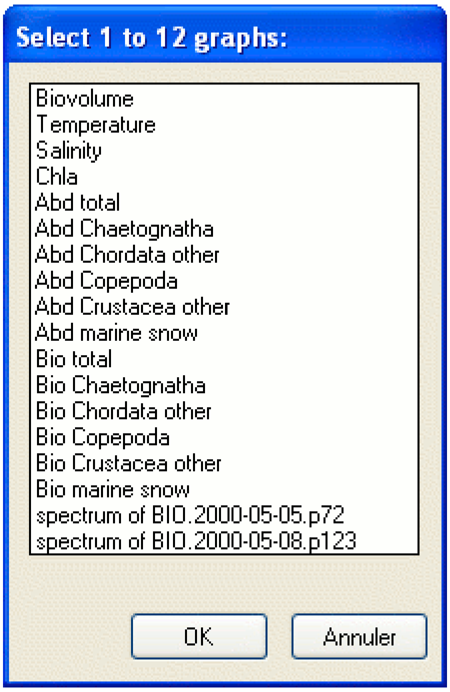
To visualize you series, use the menu entry `Analyze → View results...`, the shortcut `Ctrl+V`, or click on the eleventh button in the toolbar.

Having now calculated a `ZIRes` object that contains abundances, biomasses and size spectra, one can visualize graphs (or composite graphs with up to 12 graphs on the same page) of that series.

> *Currently, the program proposes only a limited number of graphs and you cannot customize colors, titles, etc.). These graphs are sufficient for a rapid inspection of time series, but spatial components are not handled yet. Graphs in R are very flexible, and you can visualize your data in many other ways...*

To make graphs of your results, click on the eleventh button on the ZooImage assistant toolbar:

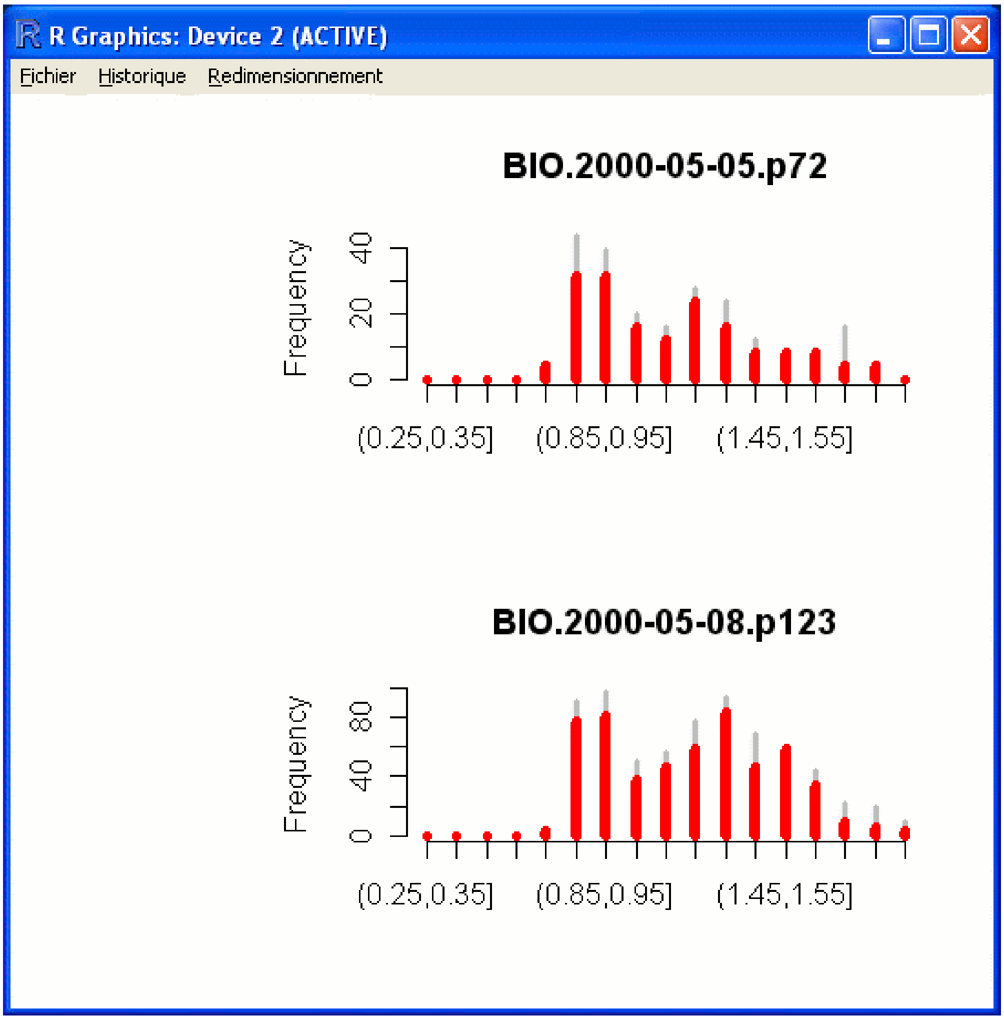
… and select the 'ZIRes' object you just created (`results`). You have then a list of possible graphs:



Select 1 to 12 graphs:

Biovolume
Temperature
Salinity
Chla
Abd total
Abd Chaetognatha
Abd Chordata other
Abd Copepoda
Abd Crustacea other
Abd marine snow
Bio total
Bio Chaetognatha
Bio Chordata other
Bio Copepoda
Bio Crustacea other
Bio marine snow
spectrum of BIO.2000-05-05.p72
spectrum of BIO.2000-05-08.p123

As the title of the list says, you can select between 1 and 12 graphs to draw. If you select the two spectra at the bottom of the list, the program asks also if you want to plot the spectra of a given taxa (in red, superimposed on top of the total size spectra). *[Select Copepoda in the list and click* OK*]*. You should obtain a composite graph similar to this one:



You should experiment with the different possible options here.

> *You can copy these graphs in Word. Just use* `File → Copy to clipboard → As Metafile` *in the graph window menu (or use* `Copy as metafile` *in the context menu after right-clicking on the window). Then, paste this graph in Word. If required, you can resize the graph window first, to adjust the size of the graph relative to the size of the text. If you have lots of graphs on the same page, you are better to maximize the graph window first.*

You can open several graph windows simultaneously, for comparison. In the `Utilities` menu of the ZooImage1 assistant, you have three entries in the `R Graphs` submenu: `New`, `Activate next` and `Close all`. They are self-explicit. The `Utilities → R Graphs → Activate next` switches the "active" flag to the next graph window. Indeed, there is only one active graph window at a time. It is the window that will receive the next graph(s). Its name ends with `(ACTIVE)`. The name of all other graph windows, if any, end with `(inactive)`. To send the next graph in a different window as the active one, use the `Activate next` menu entry until the target window becomes active.

12.4. Analyzing results in R

All Zoo/PhytoImage objects inherit from data frames, which are the basic case-by-variable type in R. Consequently, all the analysis and graphing functions of R can also be used without change on Zoo/PhytoImage objects. Look at the abundant litterature and the more than 5000 additional R packages available on CRAN (http://cran.r-project.org) to perform your analyses. Look, in particular at the task views about **environmetrics**, **graphics**, **machine learning**, **spatial**, **spatio-temporal** and **time series** for further tools that can be useful to analyse you plankton samples or series.

12.5. Exporting results

To write the result tables as ASCII files, use the menu entry `Analyze → Export results...`, the shortcut `Ctrl+E`, or click on the twelve button in the toolbar.

If, despite all the potentials of R to analyze your series right in the current environment, you want to export data, you can do it easily. Click on the forelast button on the ZooImage assistant toolbar:

Select your `ZIRes` object in the dialog box and indicate a directory (preferrably empty) where to place the tables. Zoo/PhytoImage exports one table for abundances and biomasses, and then it exports a separate table with size spectra for each sample. These are tabulation-delimited ASCII files. They should be easy to read from any other software (Microsoft Excel, Matlab, Python with Numpy/Scipy/Pandas, Julia, ...).

12.6. Further work with training/test sets

Version 3 of Zoo/PhytoImage introduces additional tools that add more flexibility in building training sets, visualizing how vignettes are automatically classified, and managing test sets.

These tools are accessible through the **Analyze** menu :

- **Add vignettes to training set** allows to complete existing tra ining sets by adding more vignettes to them without breaking the training set structure,

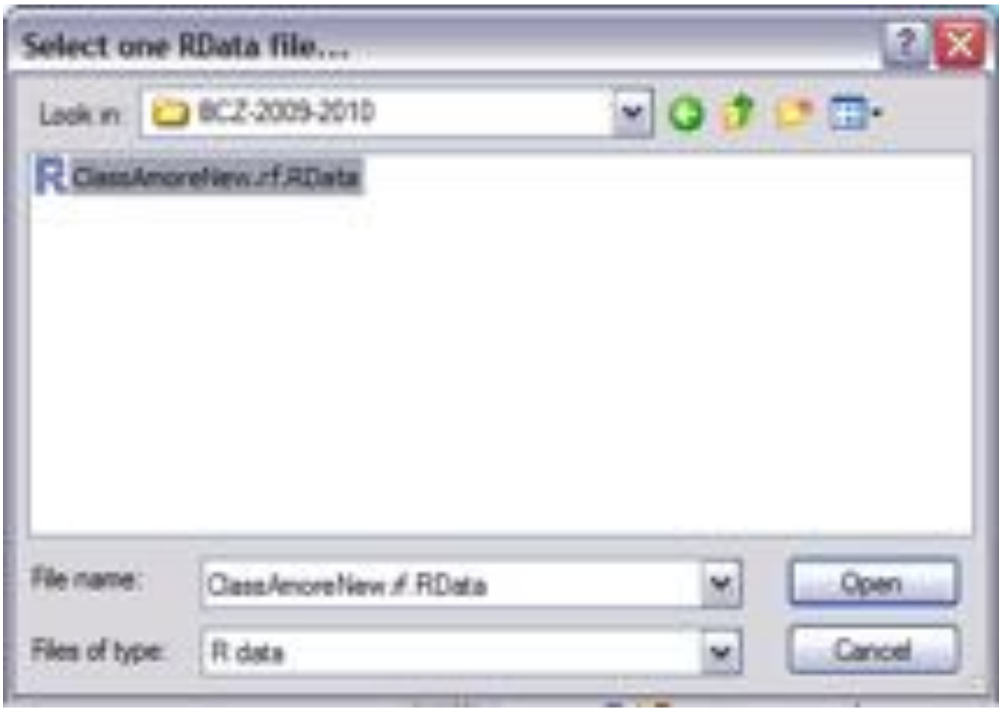
- **Automatic classification of vignettes** allows to select one sample and to represent the same folder hierachy as the one used in the original training set, with their vignettes pre-sorted according to the automatic prediction done by the chosen classifier. This serves as two purposes : (1) to visually check the quality of the classifier through the vignettes identifications, and (2) to allow for further manual correct (validation) of that classification. In this case, you can read the test set back as you do with a training set and you obtain a fully validated classification of your sample.

- **Validate classification** is a new tool that combines advanced statistical tools and a new user interface to easy (partial)-validation of classification. The tools detects so-called **suspect** items and present them first step-by-step so that optimisation procedure is more efficient. Typically, validation of only one third of all vignettes yields same level of error correction than a 90-95 % random validation procedure ! It is also combined with tools to m*odel* the error specifically for that sample, and to perform statistical correction according to that model. The combination of suspect detection and error correction provides even faster improvement of the validation : by manually validating 15-20 % only of the vignettes, one gets abundance by groups calculations with typically **less than 10 % of error for all groups**.

12.7.Smart validation of classification

Here is how to use the **validate classification** tool.

First, make sure you have created or loaded a suitable classifier (ZIClass object). Typically, you save your classifiers on disk in .Rdata files. So, to retrieve one, go to the menu **ZooImage** → **Load objects**, navigate to the folder where you store your classifier(s) and select the one you need :

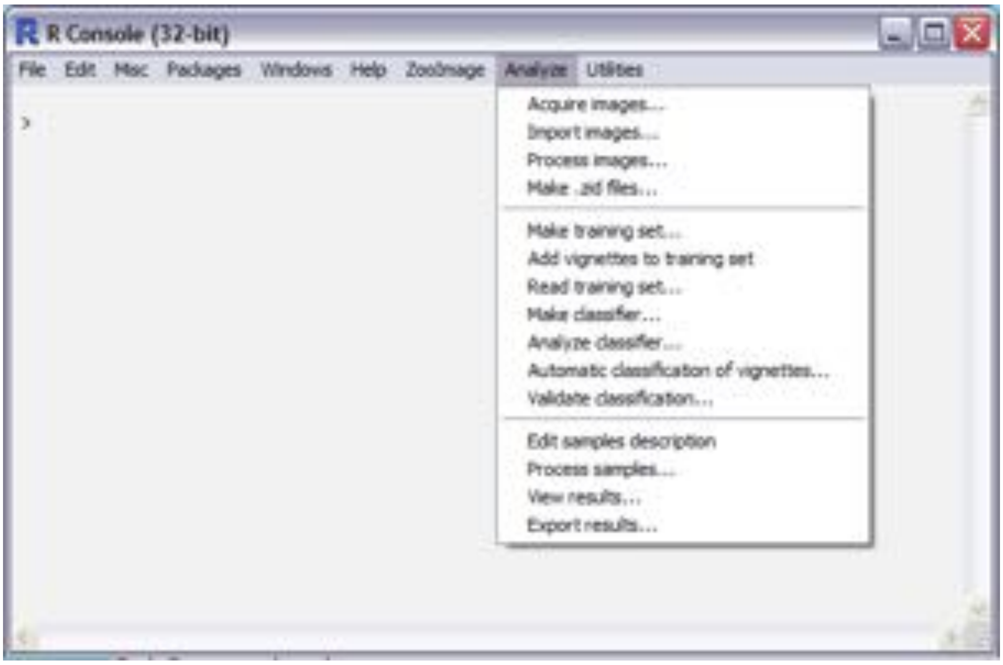


Look in: BCZ-2009-2010

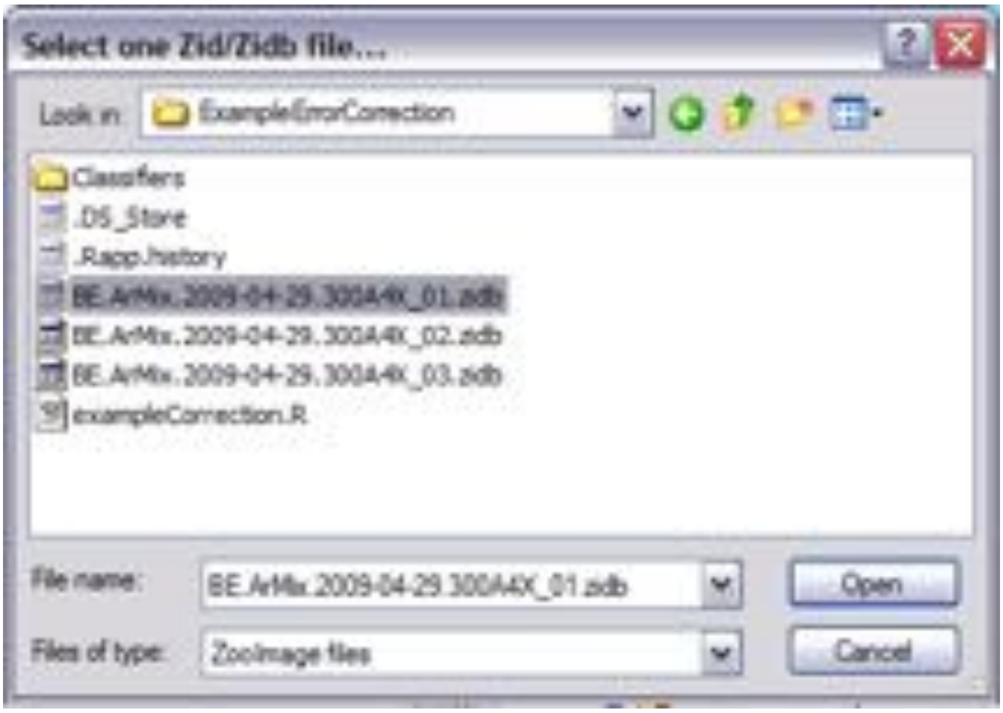
File name: ClassAmoreNew.rf.RData

Files of type: R data

Now that your classifier object is in memory, select **Validate classification** in the **Analyze** menu :



Select the ZID or ZIDB file of the sample you want to validate :

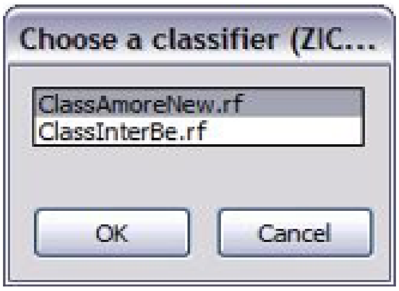

Select one Zid/Zidb file...
Look in:
Classifiers
.DS_Store
.Rapp.history
BE.ArMix.2009-04-29.300A4X_01.zidb
BE.ArMix.2009-04-29.300A4X_02.zidb
BE.ArMix.2009-04-29.300A4X_03.zidb
exampleCorrection.R
Open
Cancel

Look in: ExampleErrorCorrection

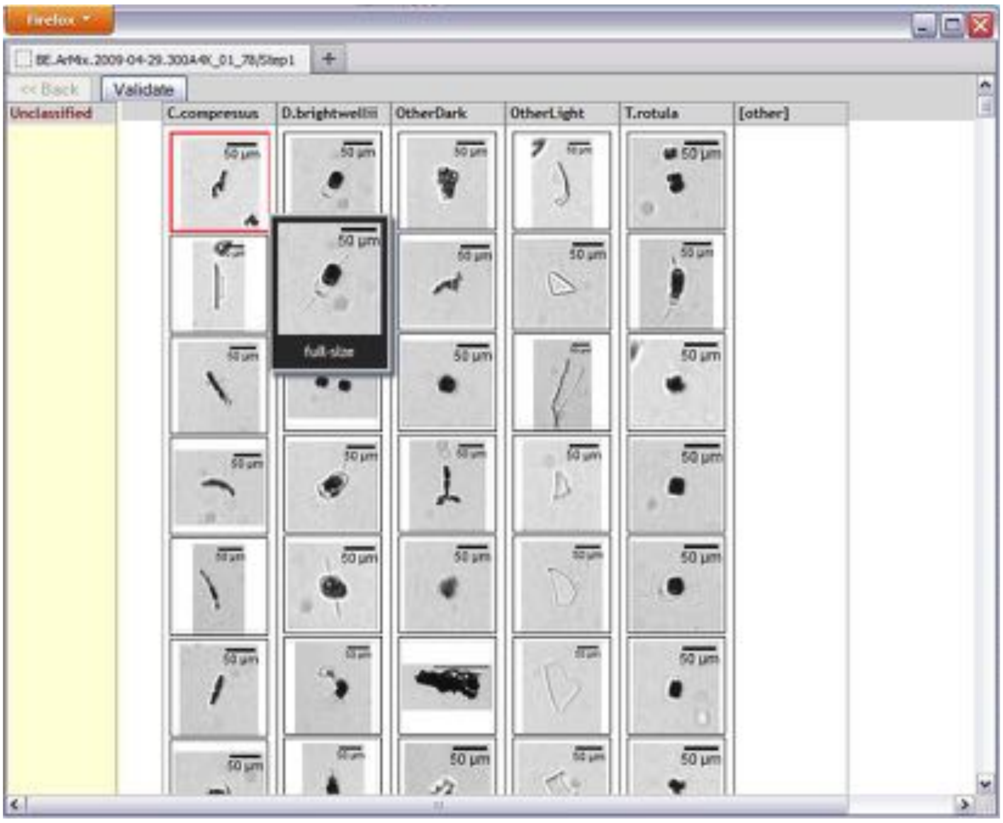
File name: BE.ArMix.2009-04-29.300A4X_01.zidb

Files of type: ZooImage files

If there is no classifier found in memory, an explicit message invites you to create or load one first. Otherwise, Zoo/PhytoImage asks you now which one of all classifiers found in the current R session you want to use :



Once it is done, Zoo/PhytoImage creates a web page that presents you a first set of (by default) 1/20th of the vignettes in the sample :



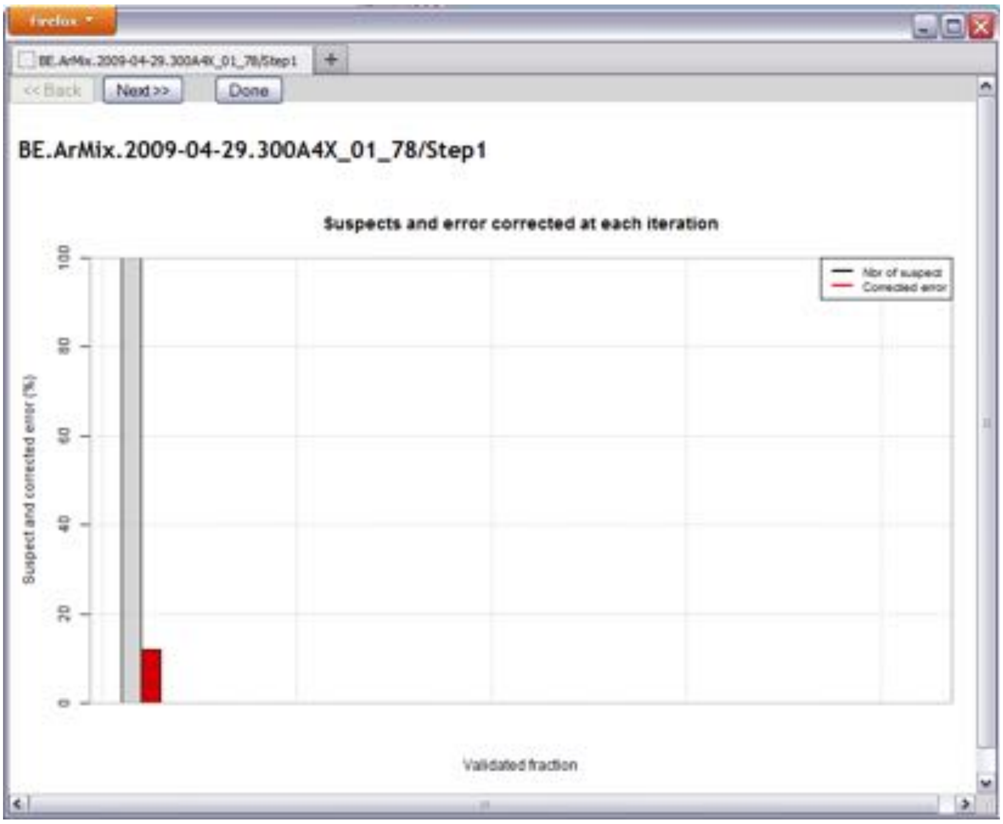
This page presents a first series of particles, randomly selected in the sample, as they are sorted automatically by the chosen classifier. Each class is represented by one column in the page (e.g., **C.compressus**, **D.brightwellii**, etc. in the example). All vignettes classified in one group are presented in the correponding column.

Moving the cursor on top of one vignette automatically triggers a floating window that dispalys the corresponding particle in full-size view for inspection.

All the vignettes can be freely drag and dropped everywhere. Thus, you can rearrange the vignettes in order to performs required corrections. For very long grids with tens or even hundreds of columns, you can use a special yellow area on the left named '**Unclassified**' to temporary store items that you want to relocate in a distant position in the grid. However, you cannot leave items in that special area when you validate your work.

For all particles that you cannot recognize, or that do not belong to the pre-specified classes, you have a special class **[other]** at the extreme right of the grid.

Once you have done with the validation of these vignettes, click on the **Validate** button. A report of the validation process done during that first step is displayed :



BE.ArMix.2009-04-29.300A4X_01_78/Step1

Firefox
BE.ArMix.2009-04-29.300A4X_01_78/Step1
<< Back
Next >>
Done
Suspects and error corrected at each iteration
Nbr of suspect
Corrected error
Suspect and corrected error (%)
0
20
40
60
80
100
Validated fraction

It present a barplot with gray bars representing the proportion of suspect items in the fraction just validated. During the first step, no model is calculated yet... so, all items are considered as suspect. A red bar at its right indicates the fraction of items that were incorrectly classified and that you just corrected. In the present case, it amounts at around 15 %. *This is a very good indication of the overall error in that classification, since this first sample is purely randomly selected !* Thus, you know that you have a totla of about 15 % error and that you already corrected 1/20th of that error.

If you continue to validate random subsamples, you still have to look at the remaining 19/20th of the sample. If you decide to accept a remaining error of less than 5 % of the total, you will still need to validate 2/3, that is roughly 12/20th of the whole sample. But wait... **doing so do not guarantee that you have less than 5 % error in *all* groups.** Typically, you will leave far more error in the rarest groups. Thus, you are better to *validate everything*, or...

... The smart validator provides a much more efficient way of validating your sample with this goal in mind of less than 5 % error in *all* groups. To reach this goal, a statistical model and a bayesian probability is calculated for each particle telling if it has a chance to be suspect (understand, probably wrongly classified) or not.

The model also considers several additional aspects :

- The probability returned by the classifier for the second class predicted for the particle is compared with the probability for the first, selected class. The idea is that, if the difference between those two probabilities is small, one should consider the particle is close to the border between the two classes and should be checked,

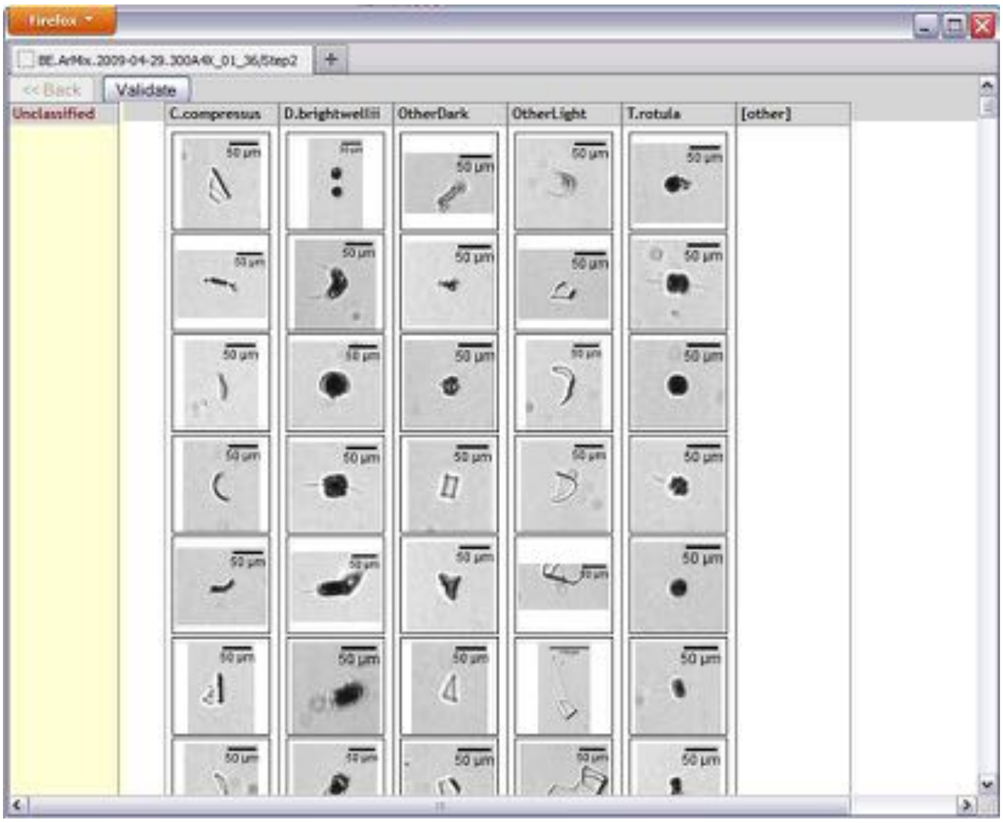
- The number of particles classified in the same class for the whole sample. If there are few of them, it is a rare group. It implies two consequences : (1) the probability of false positive increases, and (2) the class has more probabilities to contains no particles for that sample (because that taxonomic group is absent there, at that time). So, the probability to be suspect increases with the scarcity of particles classified in the same class,

- The information from the confusion matrix is used to determine which classes tend to be less good discriminated. Again, that information increases the probability of the corresponding particles to be suspect,

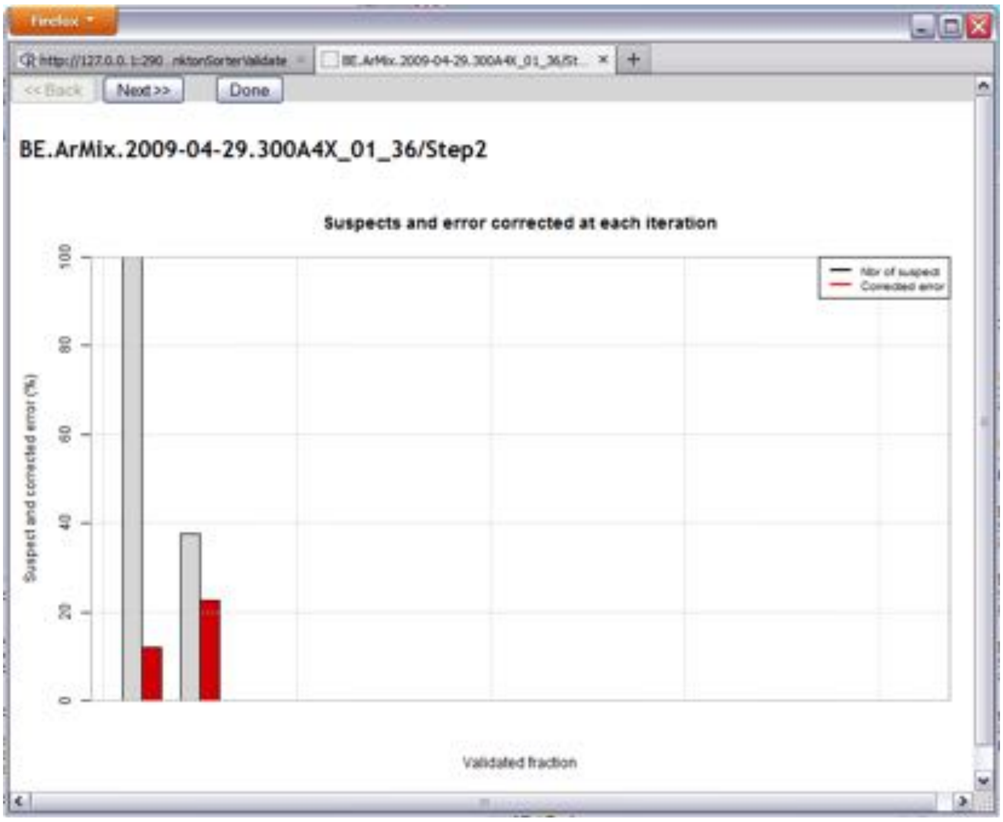
- Possibly, 'biological information' can be supplied too (not from the menu/dialog box, but by calling **correctError()** directly in the R console, see its help page at **?correctError**). That biological information should indicate if a given class has chances or not to be found in that sample. Say you know from the geographic location, from the time of the year, from the water temperature, or simply from a quick inspection of the sample under the microscope that class A is very unlikely to be present, and class B is certainly there. Just indicate a low value (say 0.01) to class A and a high value (say 0.99) to class B. Note that the numbers you provide are not necessarily restricted between 0 and 1, but the concept is easier to consider if you look at these weight like pseudo-probabilities of occurrence of the class in your sample.

Zoo/PhytoImage use the first set of particles as a training set to detect suspect items, using all features peasured on these particle, plus the additional variables described here above. Several algorithms can be use, but random forest is used by default.

So, when you click **Next**, Zoo/PhytoImage presents you another subset of the particles in the sample. But this time, the subset is not randomly chosen, but rather mainly selected in the suspect items. As a consequence, the proportion of error happens to be higher. Thus your validation work is more efficient because you start to focus on the really problematic particles now !



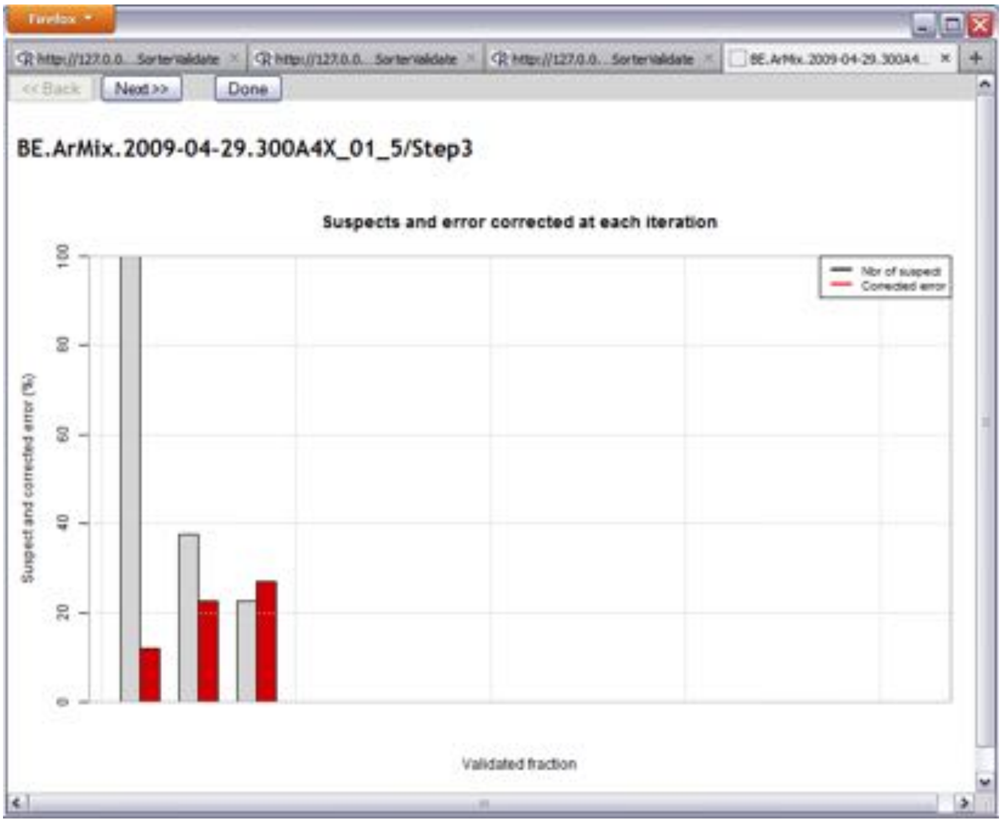
It is usually quite clear that this second set presents much more errors that the previous one... and you will also notice that, indeed, you got also much more « problematic » particles (hard to recognize particles, cropped items, blobs with strange forms, etc.). Do not hesitate to use the **[other]** group to collect what you cannot place elsewhere (but be consistent on what you do here). Click **Validate** when you have done with this second step.



BE.ArMix.2009-04-29.300A4X_01_36/Step2

Firefox
http://127.0.0.1:290...nktonSorterValidate
BE.ArMix.2009-04-29.300A4X_01_36/St...
<< Back
Next >>
Done
Suspects and error corrected at each iteration
Nbr of suspect
Corrected error
Suspect and corrected error (%)
0
20
40
60
80
100
Validated fraction

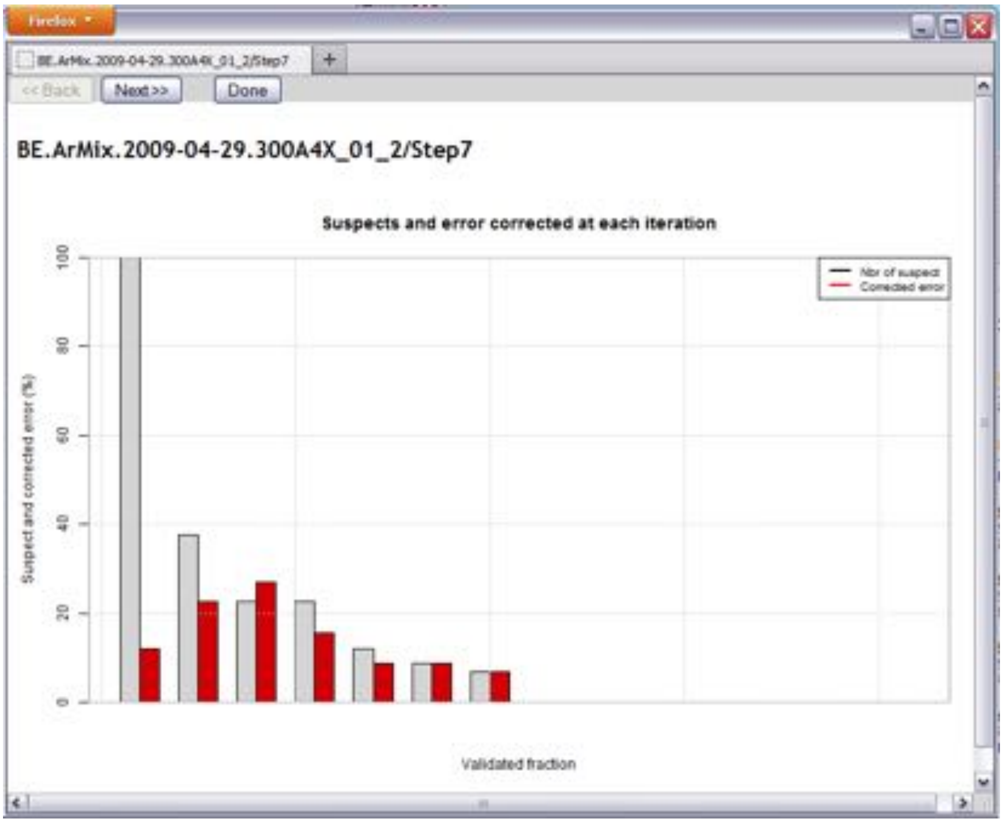
In the report, the barplot has now a second series of gray/red bars. As you can see here, the identification of suspect items is mildly efficient (recall the training set contains very few particles... 1/20th of the whole sample). Yet, you almost doubled the fraction of erroneous particles at that step. Run it a third time :



BE.ArMix.2009-04-29.300A4X_01_5/Step3

Firefox
http://127.0.0...SorterValidate
http://127.0.0...SorterValidate
http://127.0.0...SorterValidate
BE.ArMix.2009-04-29.300A4...
<< Back
Next >>
Done
Suspects and error corrected at each iteration
Nbr of suspect
Corrected error
Suspect and corrected error (%)
0
20
40
60
80
100
Validated fraction

On this sample, the algorithm predicts a relatively low amount of suspect items (on other samples, with a higher proportion of initial error, this fraction can easily reach 80 to 90 %). Nevertheless, the fraction of erroneous particles has increased a little bit more. You are now concentrating the error more efficiently. Continue with a few sets :


Firefox
BE.ArMix.2009-04-29.300A4X_01_2/Step7
<< Back
Next >>
Done
Suspects and error corrected at each iteration
Suspect and corrected error (%)
0
20
40
60
80
100
Nbr of suspect
Corrected error
Validated fraction

BE.ArMix.2009-04-29.300A4X_01_2/Step7

Here, after step 7, you notice two important things. First, the detection of suspects now closely matches actual error. Detection improves with the fraction of sample already validated that can be used for training the detection algorithm. Second, residual error drop to less than 10 %.

From this moment on, you know that you have manually validated all erroneous particles down to about 5 %. But, since the model is also used to calculate a *correction factor* for the remaining items, the calculation of abundances per classes will become quite good. Also remember that particles from rare groups were preferrably selected in the few first sets. This ensures you a good prediction for those rare groups, otherwise often problematic.

So, with this in mind, you can reasonably consider that the validation could end now and that you can trust the correction introduced by this partial validation, further helped with the statistical correction by the suspect detection model.

Click the **Done** button. Look now at the R Console. You got the corrected abundance of particles in the different classes, at it stand after the last step. Moreover, the results are saved in the `<sample>_valid` object. You can further explore it, and of course, you can use it as definitive classification of this sample.

13. Use of Zoo/PhytoImage at the R command line

A complete and detailed description of the use of zooimage functions inside the R Console is described in Chapter 12 of the following book :

Yanchang Zhao and Yonghua Cen (Eds.). Data Mining Applications with R. ISBN 978-0124115118, December 2013. Academic Press, Elsevier.

We encourage the interested readers to download the accompanying files from http://www.sciviews.org/zooimage/Data mining with R/ . There is a fully commented R script and an example dataset that browses the features available at the command line.

Here is an outline of most important tools, in additions to what you can already do using the graphical user interface and to menu in Zoo/PhytoImage 4 :

- Vignettes are accessible directement within R and can be included anywhere in R plots, or displayed as a gallery. The code to do so looks like this :

```
## Lazy loading data from one ZIDB file in R

db1 <- zidbLink(path_to_zidb)

## Contains data in *_dat1 and vignettes in *_nn

items1 <- ls(db1)

vigs1 <- items1[-grep("_dat1", items1)]

## Display a 5*5 thumbnail of the first 25 vignettes (Fig. 13.3)

zidbPlotNew("The 25 first vignettes in MTPS.2004-10-20.H1")

for (i in 1:25) zidbDrawVignette(db1[[vigs1[i]]], item = i, nx = 5, ny = 5)
```

- The `summary` method of a ZIClass object (a classifier) displays a lot of summary statistics, like recall, precision, specificity, F-score, balanced accuracy, etc. These statistics are calculated group-by-group. See the help page of the ZIClass object (`?ZIClass`).

- The ZIClass object has a `confusion` method that creates a confusion matrix with four specific plots : image, barplot, stars and dendrogram. The barplot is a new view of F-score called, « F-score » by group plot. See `?confusion` and the example in the R script. The star plot can also be used to compare two classifiers applied to the same test set.

- There are also complements about the way Zoo/PhytoImage calculates abundances and biomasses/biovolumes. You can calculate these quantities at different detail levels and indicate which groups are out of interest (e.g., marine snow and zooplankton if your study focuses on phytoplankton).

- The confusion object can be adjusted for various prior probabilities (abundances per groups) using the `prior()` function. This allows you to visualize the impact of different sample composition in the false positive and false negative rates per groups.

- Do not forget also all the R tools available to manipulate machine learning objects. See the machine learning task views at http://cran.r-project.org/web/views/MachineLearning.html.

Finally, chapter 12 in the Data mining applications with R book presents a collection of bibliographical references (64), most of them pointing on publications whose analyses were done using Zoo/PhytoImage. This is also an excellent source of inspiration showing in practice how Zoo/PhytoImage can be used.

14. Annexes

14.1. Data and metadata in .zis files

Here is the explanation of the **data** and **metadata** in this `description.zis` file:

Key	Section	Comment
ZI1	-	This is not a key, but just an identifiant telling it is a ZooImage1 file.
Id	Description	The short identifiant of the series.
Name	Description	A longer name for this series.
Institution	Description	The institution that owns the series, i.e., where original biological material is stored, if any.
Objective	Description	The goal(s) of this study.
Description	Description	A short description of the series[21].
Contact	Description	The name of a responsible person of this series.
Email	Description	The email address of the contact.
URL	Description	An optional URL pointing to a Web page that further describes the series, if any.
Note	Description	A short general comment about this series.
Code	Series	The code of a sub-series.
Name	Series	The name of a sub-series.
Project	Series	The project in which this sub-series is included.
Institution	Series	The owner of the sub-series, as above for the series.
Country	Series	Country(ies) concerned by this sub-series.
Location	Series	Place(s) concerned by this sub-series.
Contact	Series	As above for the sub-series.
Email	Series	Idem.
URL	Series	Idem.
Note	Series	Idem.

[21] Fill these metadata : many of these are used by Zoo/PhytoImage for its calculations !

Code	Cruises	A code for a cruise.
ShipName	Cruises	The name of the ship.
ShipType	Cruises	The type of the ship.
ShipCallSign	Cruises	Immatriculation of the ship.
PortDeparture	Cruises	Self-explicit…
PortReturn	Cruises	Idem.
Captain	Cruises	Name of the captain.
Coordinator	Cruises	Name(s) of the scientific coordinator(s) on board.
Investigators	Cruises	Name(s) of additional scientific staff on board.
Start	Cruises	Date of departure in `yyyy-mm-dd`.
End	Cruises	Date of arrival at the final destination in `yyyy-mm-dd`.
SouthmostLat	Cruises	Southmost latitude reached in `+/-x.xx` (degree.decimal).
WestmostLong	Cruises	Westmost longitude reached in `+/-x.xx`.
NothmostLat	Cruises	Northmost latitude reached in `+/-x.xx`.
EastmostLong	Cruises	Eastmost longitude reached in `+/-x.xx`.
Project	Cruises	The project to which this cruise belongs.
URL	Cruises	An optional URL pointing to a web page that further describes this cruise.
Note	Cruises	A short comment about this cruise.
Code	Stations	A code for this station.
Location	Stations	The name of location of this station.
Latitude	Stations	The latitude of the station (in `+/-x.xx`).
Longitude	Stations	The longitude of the station (in `+/-x.xx`).
Start	Stations	The date at which sampling started at the station (in `yyy-mm-dd`).
End	Stations	The date at which sampling was stopped (if any, in `yyyy-mm-dd`).
Frequency	Stations	The frequency of sampling (in no of samples per day).
Depth	Stations	The maximum depth at the station location (in m).

Description	Stations	A short description for this station.
Note	Stations	A short note concerning this station.
Label	Samples	The complete label of the sample, as in the file names.
Code	Samples	A code for this sample.
SCS	Samples	The `SCS` for that sample.
Series	Samples	The series code to which that sample belongs.
Cruise	Samples	The cruise code corresponding to the sample (if any).
Station	Samples	The station code.
Date	Samples	The data of sampling (in `yyyy-mm-dd` format).
Time	Samples	The time of sampling (in `hh::mm::ss`).
TimeZone	Samples	The time zone (lag from GMT in `+/-x` hours).
Latitude	Samples	The latitude of sampling (in `+/-x.xx`).
Longitude	Samples	The longitude of sampling (in `+/-x.xx`).
CoordsPrec	Samples	Precision of lat./long. (radius in m).
Operator	Samples	Who collected this sample?
GearType	Samples	The type of gear used to collect the sample.
OpeningArea	Samples	The opening area (if collected with a net, in m^3).
MeshSize	Samples	For a net only, size of the mesh (in µm).
DepthMin	Samples	Minimum depth of sampling (in m).
DepthMax	Samples	Maximum depth of sampling (in m).
SampVol	Samples	Volume of seawater sampled (in m^3).
SampVolPrec	Samples	Precision of sampled volume (in m^3).
TowType	Samples	Type of tow (vertical, horizontal, oblique, etc.).
Speed	Samples	Speed during tow (in m/s).
Weather	Samples	Weather conditions during sampling.
Preservative	Samples	Preservative used (for instance, buffered formaldehyde 4%).
Staining	Samples	Staining used (if any).

Biovolume	Samples	Rough estimation of the biovolume after sedimentation (in mm^3).
Temperature	Samples	Temperature of the water at sampling (in degree Celcius).
Salinity	Samples	Salinity of sampled water (in per thousands).
Chla	Samples	Chlorophyll alpha in the sampled water.
Note	Samples	A short note about this sample.
…	Samples	You can add any additional measurement done on the sample here…