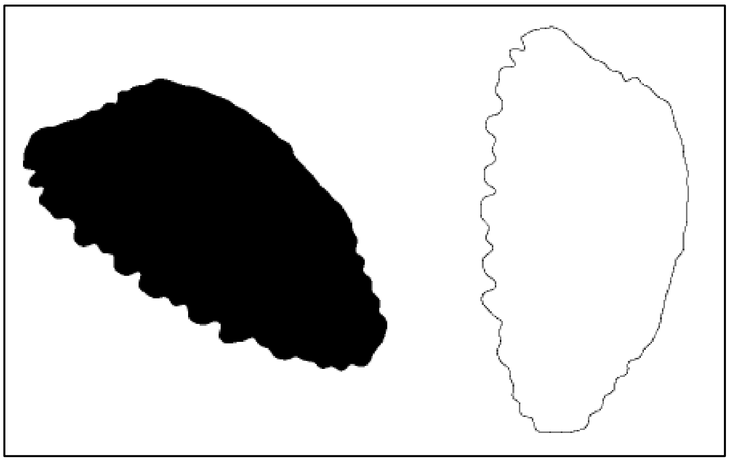
2-D Shape Analysis in IMAGIC-5



This manual is a little "hands-on" first introduction into work with IMAGIC-5 in general and especially on how to use the 2-D shape analysis commands.

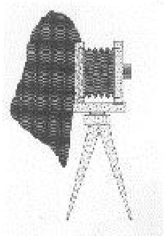
Berlin, Germany
www.ImageScience.de

This manual contains the following chapters:

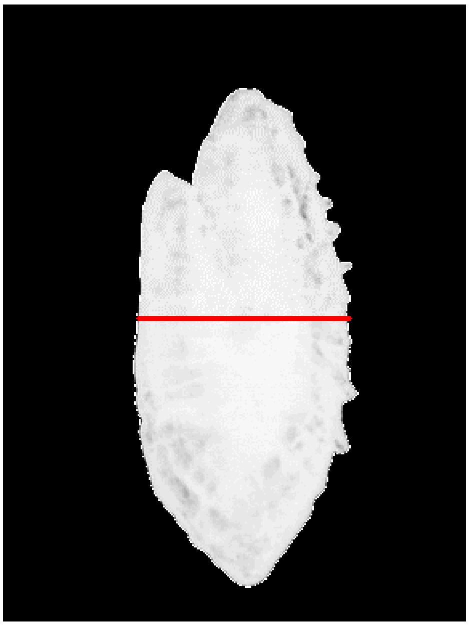
WELCOME TO THE

MAGIC WORLD

OF IMAGIC



What is Fish Otolith Shape Analysis good for?



A fish otolith is the ear stone of a fish. Otolith are widely used for determining the age of fish. Usually, the internal otolith structure is examined. In thin-sections of an otolith, ring-like zones - like the year rings of a tree - may become visible. However, preparations are rather time-consuming and therefore expensive. Also, there is quite some room left for subjective judgements by the so called reader.

This is where otolith shape analysis comes in. While different readers may draw different conclusions on the same sample, it can be expected from a fully computerised system that such uncertainties are minimised or even avoided.

The otolith shape can be obtained unambiguously from digital images. The raw data is then introduced into highly sophisticated image analysis programs which extract relevant shape descriptors and use these against a database to find matches. If proper age-group reference databases, e.g. for different species or different habitats, are available, the identification can be performed in an almost automatic fashion.

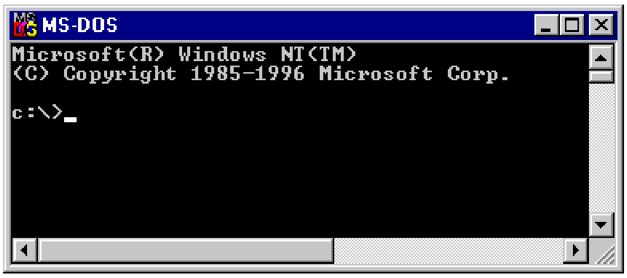
Introduction to IMAGIC-5

Starting IMAGIC-5:

Currently, IMAGIC-5 is only running as a console application in an MS-DOS window. You must therefore first open such a MSDOS window. Use the corresponding icon

MS-DOS or start a Command Prompt using Start / Programs / Command Prompt

You will get an MS-DOS window like this:



```
Microsoft(R) Windows NT(TM)
(C) Copyright 1985-1996 Microsoft Corp.

c:\>_
```

Then change to your working directory:

```
c:\> cd c:\<my directory>
                                          you specify <my directory>
```

To start IMAGIC-5 type:

```
c:\> imagic
```

IMAGIC-5 will respond

```
** IMAGIC-5 welcomes you **

IMAGIC-COMMAND:
```

and is waiting for your command(s).

Some helpful hints when working in an MS-DOS window:

First of all, it is a good idea to activate the so called DOSKEY feature which is available for most systems. When you enter and activate the DOSKEY command, you will later be able to use the arrow keys for scrolling through the list of commands you have entered before.

If you are running Windows NT, another helpful feature is the number of buffered lines of your MS-DOS console window. Use the right mouse button and click into the title bar of the MS-DOS window to activate the properties window. Please check out more details yourself.

Panic and worry: Don't !

- You can always give **?** or HELP to get help at any level of IMAGIC-5.
- You can ask **?** for all questions that appear on the screen to get helpful explanation of what is being asked.
- You can always use the default answers (in brackets) by just giving CR, but, of course, you can also type your own ones.
- IMAGIC-5's general quit command is the * . It is a legitimate answer to any question that appears on the screen and its effect is to leave the current operation.
- At any level of IMAGIC-5 you can give an operating system command by typing **!** or **$** followed by the system command.

Looking for a specific IMAGIC-5 command:

Since IMAGIC-5 responds to over one-hundred different commands, it is clear that we cannot go into details. If you need assistance on valid commands you can retrieve a list of all commands by typing MENU.

Example: try to find a command to import FABOSA into IMAGIC-5:

```
IMAGIC-COMMAND: MENU *import
```

You will get a listing of all command names that have to do with "import":

```
IMAGIC-COMMAND: MENU *import

  &IMPORT-EXPORT-IMAGIC-FORMAT
  &IMPORT-PDB-TO-PLT
```

Now you can ask for specific HELP by typing:

```
IMAGIC-COMMAND: HELP *import
```

How to run an IMAGIC-5 command:

Example: creating test images with command TEST-IMAGE-CREATION

```
IMAGIC-COMMAND: TEST-IM                                 you can always use
                                                        abbreviations of the
                                                        command name

** TESTIM welcomes you **

Output filename, image loc#s      [test] : ?
                                                        you can give a  ?
                                                        to get help:

  The filename consists of any string of characters that is
  compatible with the file system on your computer. The file
  filename may be followed by location numbers as in:
       myimage,1,20
  in which case a total of 20 test images are generated by this
  program which occupy the first 20 locations of the image
  file.
  The file will be stored in the default directory you are
  working in. If you want another directory specify that by
  using the correct prefixes.

Output filename, image loc#s      [test] :
                                                   you can give a system
                                                   command (for example
                                                   to list the directory) by
                                                   using a  !  or  $:

Output filename, image loc#s      [test] : $ dir *.img

                                                   (MSDOS style)
                                                   (Unix style: ! ls *.img)

     my_file1.img
     my_file2.img

Output filename, image loc#s      [test] :
                                                   if you only give RETURN
                                                   then the default value in
                                                   brackets (here "test") is
                                                   used as your answer

Image dimensions X,Y [128,128]           : 64,64

                                                   your own answer
                  etc.
```

The DISPLAY command:

One of the most frequently used IMAGIC-5 commands is DISPLAY-IMAGE. In an XWindows / Windows NT environment, you can (and often will) start multiple display windows:

```
IMAGIC-COMMAND: DISP

** DISPLAY welcomes you **

Input image file, image loc#s        :
```

The first question that will appear on the screen concerns the choice of the file you wish to display; go ahead and try to get one of the test images you just created on the screen. If you have forgotten the names of the images type:

```
Input image file, image loc#s        : $ DIR *.img      VMS/NT
```

or:

```
Input image file, image loc#s        : ! ls *.img       UNIX
```

This is just an operating system call to get a list of the files you own. Look for files with the extension ".img".

Specify the image file you want to display:

```
IMAGIC-COMMAND: DISP

** DISPLAY welcomes you **

Input image file, image loc#s        : test
```

Before continuing IMAGIC-5 first shows you some information about the specified image(s), like:

```
Image name and parameters [and history]:
TEST IMAGE
SIZE: 64,64  LOC: 1  TYPE: REAL  CREA.DATE:21-10-1999
TESTIM:SIEMENS STAR,
```

In a windows environment IMAGIC-5 will ask you to give the display window size, for example:

```
Give size of the display window [512,512]   : 800,600
```

Then IMAGIC-5 will show a list of all parameters (display settings) that are currently set for displaying your images, for example:

```
Current DISPLAY settings:

Input image FILE name                : test
LOCATION numbers                     :   1  to:  3
Output DEVICE                        : XWINDOW
SCALE factor                         :    1.0000
MINX, MAXX                           :    1 to:   64
MINY, MAXY                      :    :    1 to:   64
GREYVALUES                      :    : AUTOMATIC
ERASE screen                     :   : NO
STARTing point  (top left)           :    1 ,    1
Display of NAME and location         : LOCATION
Video lookup table (VLT)             : STANDARD BLACK/WHITE

Change parameter settings
(ERASE,SCALE,LOC,START,SCAN,SAVE,BYE ...)? [NO] :
```

If you do not understand the meaning of a parameter, don't hesitate to answer **?** .

Then just give RETURN, which means "NO MORE CHANGES". IMAGIC-5 will display your image(s) and will return to the parameter setting as shown above.

```
Change parameter settings
(ERASE,SCALE,LOC,START,SCAN,SAVE,BYE ...)? [NO].: NO
```

If you want to change some of the parameters, please look at the words written in uppercase. If you answer the question "Change parameter settings ?" by typing such a word you can change the corresponding parameter(s).

<u>Example:</u>

In the parameter setting above the scale factor is 1.0. If you want a scaling of 2.3, answer:

```
Change parameter settings
(ERASE,SCALE,LOC,START,SCAN,SAVE,BYE ...)? [NO] : SCALE
```

IMAGIC-5 will show you the current image size and ask you the next question:

```
     Image size is :      64 x 64

Give scale factor for display [Def=1]           :
```

You can specify the desired value, like:

```
     Image size is :      64 x 64

Give scale factor for display [Def=1]           : 2.3
```

IMAGIC-5 will change the parameter and show you the new settings:

```
Current DISPLAY settings:

Input image FILE name                   : test
LOCATION numbers                        :   1  to:  3
Output DEVICE                           : XWINDOW
SCALE factor                            :    2.3000
MINX, MAXX                              :    1 to:   64
MINY, MAXY                              :    1 to:   64
GREYVALUES                              : AUTOMATIC
ERASE screen                            : NO
STARTing point  (top left)              :    1 ,    1
Display of NAME and location            : LOCATION ONLY
Video lookup table (VLT)                : STANDARD BLACK/WHITE

Change parameter settings
(ERASE,SCALE,LOC,START,SCAN,SAVE,BYE ...)? [NO] :
```

You don't have to worry if you want to change more than just one parameter, the program will loop back to this question after you changed something. Once the setting will be correct and you want to see the images. Answer "NO CHANGES" by giving RETURN:

```
Change parameter settings
(ERASE,SCALE,LOC,START,SCAN,SAVE,BYE ...)? [NO].: NO
```

IMAGIC will display the images #1, #2 and #3 of file test on the monitor using the specified parameters.

If you want to leave the DISPLAY program give BYE or * . Also try to use command EXIT and see what happens.

```
Change parameter settings
(ERASE,SCALE,LOC,START,SCAN,SAVE,BYE ...)? [NO].: BYE
```

Image sequences in IMAGIC-5

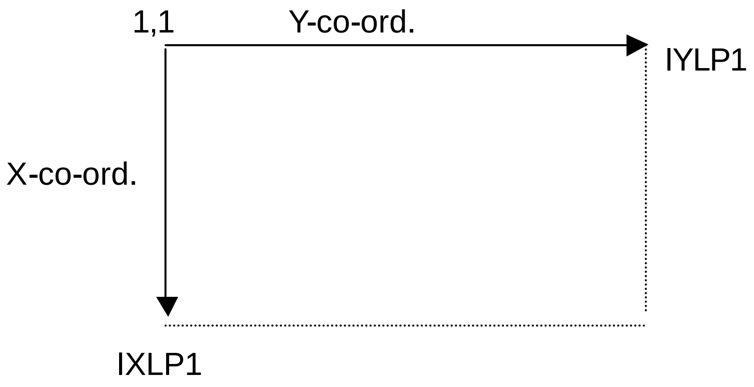
Every IMAGIC-5 image file can contain a sequence of images. All images in the sequence will automatically be treated by the procedure unless the user specifies a set of image location numbers explicitly, like in:

```
IMAGIC-COMMAND: COPY-IM

** COPYIM welcomes you **

Input file, image loc#s                     : test,10,64
                         etc.
```

in which case the images from number 10 to 64 are used-

The IMAGIC-5 co-ordinate system

The IMAGIC-5 co-ordinate system is a right-handed system with its (1,1) origin in the top-left corner of the image. The length of the lines (number of rows/columns) is IYLP1 and the number of lines is IXLP1:



The centre of the image, for almost all operations on the image, is given by:

(IXMID,IYMID) = (IXLP1/2 + 1 , IYLP1/2 + 1).

This point is also the default centre of rotational symmetric masks and the like.

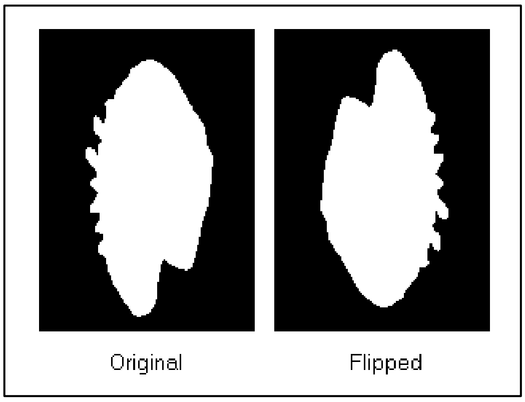
Starting IMAGIC-5 from many windows

You can start IMAGIC-5 in more than only one MS-DOS window.

For example:

It can be very helpful to use one window for command DISPLAY while playing around / testing filter parameters with command BAND-PASS-FILTER in an other window.

2-D Shape Analysis



The purpose of this chapter is to provide the user with guidance in a practical hands-on to perform 2-D shape analysis.

Special Commands used for 2-D shape-analysis

The outer circumference of an object may be called its contour. The shape information we are going to deal with is completely given by the contour line points. However, instead of the contour line points, other descriptors can be used.

IMAGIC-5 provides a number of specialised commands that you can use to start from a set of input images and obtain the respective shape descriptors. Command names and short descriptions are listed in the table below.

IMAGIC command	Description
IMPORT-EXPORT / EM2EM	converts a set of pairs of TIF and TXT files to a special IMAGIC file
CREATE-NFD-CONTOUR	calculates the contour and corresponding Normalised Fourier Descriptors (NFDs)
PRINT-NFD-INFO	creates a text file containing all information contained in a NFD file
HEADERS-INFORMATION	displays image name, parameters and history of IMAGIC and NFD files
GENERATE-IMAGE-FROM-NFD	generates graphical representations from a NFD file
FLIP-NFD-UPSIDE-DOWN	tool to assist flipping, i.e. rotation by 180 degrees, of objects in a NFD file
MSA-CLASSIFY	hierarchical classification of objects in a NFD file
MSA-EXTRACT-SORTED-CLASS	extracts members of classes to separate files and calculates class averages
MSA-SELECT-NFD	selects members of a class from a NFD file
MSA-AVERAGE-NFD	calculates average contours in NFD file format
DISPLAY	multi-frame display window for all kinds of IMAGIC-5 image files

2-D Shape Analysis - Step by Step

Listing of the processing steps:

(1) Importing FABOSA images into IMAGIC-5

(2) Reading header information

(3) If wanted display the IMAGIC-5 images with command DISPLAY

(4) Converting 2-D images to NFD files with command CREATE-NFD-CONTOUR

(5) Checking the calculated contour(s)

(6) If needed print out NFD information with command PRINT-NFD-INFO

(7) Multivariate Statistical analysis using command MSA-CLASSIY

(8) Select certain class contours with command MSA-SELECT-NFD

(9) Average certain classes with command MSA-AVERAGE-NFD to give average contour(s)

(10) If wanted flip contours using command FLIP-NFD-UPSIDE-DOWN

(11) If wanted convert the NFD contours back to "real" images (command GENERATE-IMAGE-FROM-NFD)

(12) Instead of using steps (4) - (7) you can analyse the original images with commands MSA-RUN, MSA-CLASSIFY and MSA-SUM (this is only a test option).

You will find a detailed description of all processing steps in the following chapters.

FABOSA images

FABOSA images are non-compressed TIFF images plus an ASCII text file which contains some additional otolith information. A typical file looks like this:

```
<Working Directory>

</Working Directory>
<Operator>
my_operator
</Operator>
<Date>
November 9,  2002 4:21:45 PM
</Date>
<Image ID>
my_otolith
</Image ID>
<Reference>
38252-A
</Reference>
<Comment>

</Comment>
<Known Age>
unknown
</Known Age>
<Reader>
unknown
</Reader>
<Age By Method>
Method: none Value: unknown
</Age By Method>
<Fish Length>

</Fish Length>
<Fish Weight>

</Fish Weight>
<Catch Site>

</Catch Site>
<Catch Date>

</Catch Date>
<Species>
anchovy
</Species>
<Sex>
unknown
</Sex>
<Otolith Type>
```

```
sagitta
</Otolith Type>
<Body Side>
left
</Body Side>
<Camera>
Leica DC100
</Camera>
<Optical Equipment>
Olympus SZX - 9
</Optical Equipment>
<Magnification>
3 / 592 / 0.5 x 32
</Magnification>
<Illumination>
light from below
</Illumination>
<Appearance>
dark
</Appearance>
<Sulcus Side>
up
</Sulcus Side>
<Environment>
air
</Environment>
```

Most of this information is NOT used in the otolith analysis. It is therefore possible to use TIFF images without this additional text file.

But one important information is the MAGNIFICATION which can be used in the subsequent otolith analysis. If you want to include this information please create a text file <file_name>.txt (where <file_name> is the same as the image file name) which contains three lines like this:

```
<Magnification>
3 / 592 / 0.5 x 32
</Magnification>
```

In the example above we have specified the magnification "3 mm" per "592 pixels". The idea is to image a scale (in the example: 3mm) and count the related number of pixels in the image (here: 592 pixels). The information "0.5 x 32" is the magnification given by the microscope manufacturer and will not be used. Please measure and replace "3" and "592" with the values for your microscope!

PLEASE NOTE:

Although in the following we talk about FABOSA images you can always use "normal" TIFF images.

(1) Importing FABOSA images into IMAGIC-5

Use IMAGIC-5 command IMPORT-EXPORT or EM2EM, respectively, to convert the images into IMAGIC-5 format:

```
IMAGIC-COMMAND: IMPORT-EXP

** EM2EM welcomes you **

Data format of the input to be converted  : FABOSA
                                          FABOSA is the format
                                          of the input images
Export to which data format [?]           : IMAGIC
                                          IMAGIC, of course
Input = set of 2D sections of a 3D volume : NO
                                          always NO for FABOSA
Import a SET of input image files [NO]    : NO
                                          NO:   one input image
                                          YES: a sequence of
                                          images to be imported
```

A little more information to the last mentioned question: If you give YES you have to specify the input FABOSA and output IMAGIC-5 file name and the command starts converting:

```
Import a SET of input image files [NO]    : NO
Input image file [?]                      : my_fabosa_image
                                                      you select
Output image file (NO EXT.) [?]           : my_imagic_image
                                                      you select
```

If you want to import a sequence / a series of images and convert them into one IMAGIC-5 image file you have to specify YES:

```
Import a SET of input image files [NO]    : YES
```

Next, you can select INTERACTIVE or FILE mode:

```
Input file names INTERACTIVE or from FILE :
```

In INTERACTIVE the command expects the file names in the following way: a rootname followed by a positive number plus the usual extension ".tif". The command wants to now the rootname plus the lowest and highest number.

<u>Example</u>:

The input FABOSA files are files are	my_image13.tif	my_image13.txt
	my_image15.tif	my_image15.txt
	my_image16.tif	my_image16.txt
	my_image21.tif	my_image21.txt

then you have to specify:

```
Input file names INTERACTIVE or from FILE : INTERACTIVE
Input rootname (NO EXT.), first#,last#    : my_image,13,21
                                                      you specify

Output image file (NO EXT.) [?]           : my_imagic_image
                                                      you select
```

If no input sequence with common rootname is available you can create a text (ASCII) file in which the input file names are stored line by line. This file can be used to specify the input file names with option FILE:

```
Input file names INTERACTIVE or from FILE : FILE
Text file with file names [import.txt]    : my_images.txt

                                                    you specify

Output image file (NO EXT.) [?]           : my_imagic_image

                                                    you select
```

Example:

Content of text file faroe_cods.txt:

C:\fabosa\faroe_bank\cod1
C:\fabosa\faroe_bank\cod12
C:\fabosa\faroe_bank\fish1
C:\fabosa\faroe_bank\fish10
C:\fabosa\faroe_plateau\test45
C:\fabosa\faroe_plateau\sample3

If option FILE is used with input text file faroe_cods.txt then six input images will be imported to IMAGIC-5 format and all together stored in one IMAGIC-5 output file.

In both options, INTERACTIVE and FILE, IMAGIC-5 will subsequently work through all input files. For each TIF input file IMAGIC-5 checks if there is a corresponding text file containing specific sample information. If there is no such text file available (which should always be available in FABOSA format, but nevertheless ...) sample information entries will be set to unknown or other suitable defaults.

IMAGIC-5 will display some reports during image processing and the history entry for the last processed file. Then the IMAGIC-5 prompt is displayed indicating program completion and IMAGIC-5 is waiting for the next command:

```
HEADER info from "FABOSA" file:
-------------------------------

TIFF input image file         : c\fabosa\my_otolith1.tif
Number of pixels per line     :   582
Number of lines per image     :   768
Number of sections            :     1
TIFF data format              : byte (PACK)

Image name, parameters [and history]:
bc449    sagitta  cod                      26.8.98  Faroe Bank
Size:582,768 Loc:1 Type:PACK Cre.Date: 9/17/1999  Time: 16.38
EM2EM;


HEADER info from "FABOSA" file:
-------------------------------

TIFF input image file         : c\fabosa\my_otolith21.tif
Number of pixels per line     :   582
Number of lines per image     :   768
Number of sections            :     1
TIFF data format              : byte (PACK)

Image name, parameters [and history]:
bc449    sagitta  cod                      26.8.98  Faroe Bank
Size:582,768 Loc:21 Type:PACK Cre.Date: 9/17/1999  Time: 16.38
EM2EM;


IMAGIC-COMMAND :
```

give your next command

Out of the many individual FABOSA image and text files EM2EM / IMPORT-EXPORT has created one IMAGIC-5 image file. What is meant by "one IMAGIC-5 file" are actually two files (see "Introduction to IMAGIC-5") with the same name having extensions .img and .hed for the image and header portions.

It should be noted that whatever format the original images were, IMAGIC-5 converts them to grey scale. With the FABOSA input specifier, the negative image will also be taken so as to achieve the object being bright against a darker background.

(2) Read header information

You can inquire quite a number of different entries stored for each single location in an IMAGIC-5 file. There is a special mode of operation for FABOSA files to print this information on the monitor. Call command HEADER:

```
IMAGIC-COMMAND : HEADERS-INFORMATION


** HEADERS welcomes you **

 Available options:

 ?,         ANGLES,      CLASSNO,     FLOAT
 GREYLEV,   HERM,        HISTGRM,     HOWMANY
 LOOK,      MEANING,     NAMES,       NORM
 PLOT,      PUT_EULER,   REFNO,       SET
 SORT,      STORY,       TITLE,       WIPE
 ZERALL,    ZEROCC,      BYE,         EXIT

 Choose mode of operation [?]                     : look
 Give input file, image loc#s                     : my_otolith,21,22
```

**you select,
note that you can use all images by just typing the file name or using a series of images by specifying the wanted image location numbers**

```
 Options for LOOK:

 ALIGNMENT,  EULER (angular reconst. only)
 FABOSA,     THREED/EULER  (3-D and EULER)
 STATISTICS, INDEX

 Please specify the wanted criteria [?]      : fabosa
```

You will get the following output:

```
Image name, parameters [and history]:
 399      sagitta  cod                      3.9.98   Faroe Bank
 Size:582,768 Loc:21 Type: PACK Cre.Date: 9/17/1999 Time: 16.38
 EM2EM;

 Location     :           21

 File content : image
 Image ID     : bc445
 Magnification: 0.02066 mm/pixel
 Species      : cod
 Otolith type : sagitta
 Sulcus side  : up
 Body side    : left
 Sex          : unknown
 Fish length  :   910.0 mm
 Fish weight  : unknown
 Catch date   : 26.10.98
 Catch site   : Faroe Bank

 Location     :           22

 File content : image
 Image ID     : bc449
 Magnification: 0.02066 mm/pixel
 Species      : cod
 Otolith type : sagitta
 Sulcus side  : up
 Body side    : left
 Sex          : unknown
 Fish length  :   930.0 mm
 Fish weight  : unknown
 Catch date   : 26.8.98
 Catch site   : Faroe Bank

 IMAGIC-COMMAND :                 give your next command
```

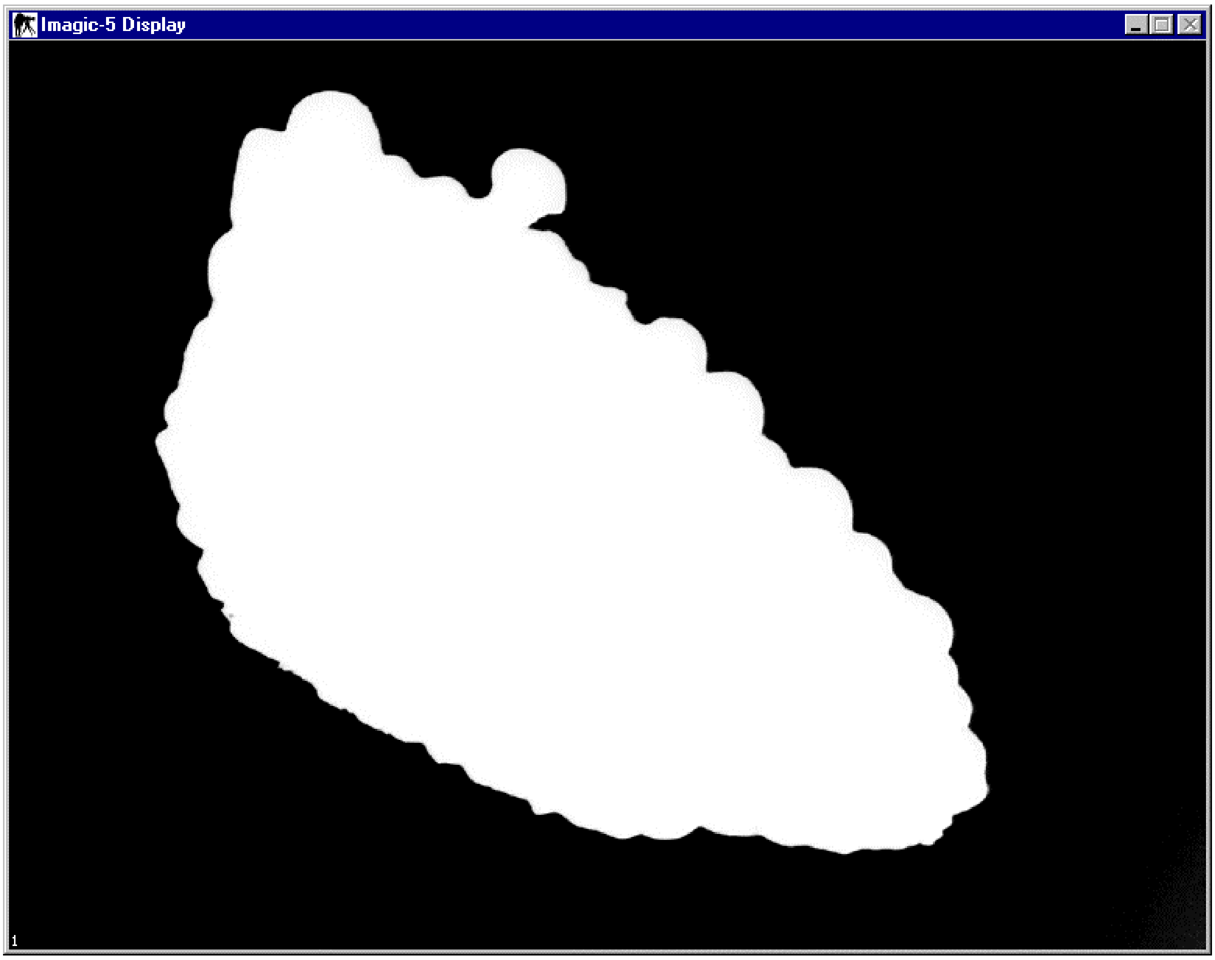
(3) Display the IMAGIC-5 images

The DISPLAY command was already described in "Introduction to IMAGIC-5". As already mentioned the command is DISPLAY:

```
IMAGIC-COMMAND: DISP

** DISPLAY welcomes you **

                    etc.
```

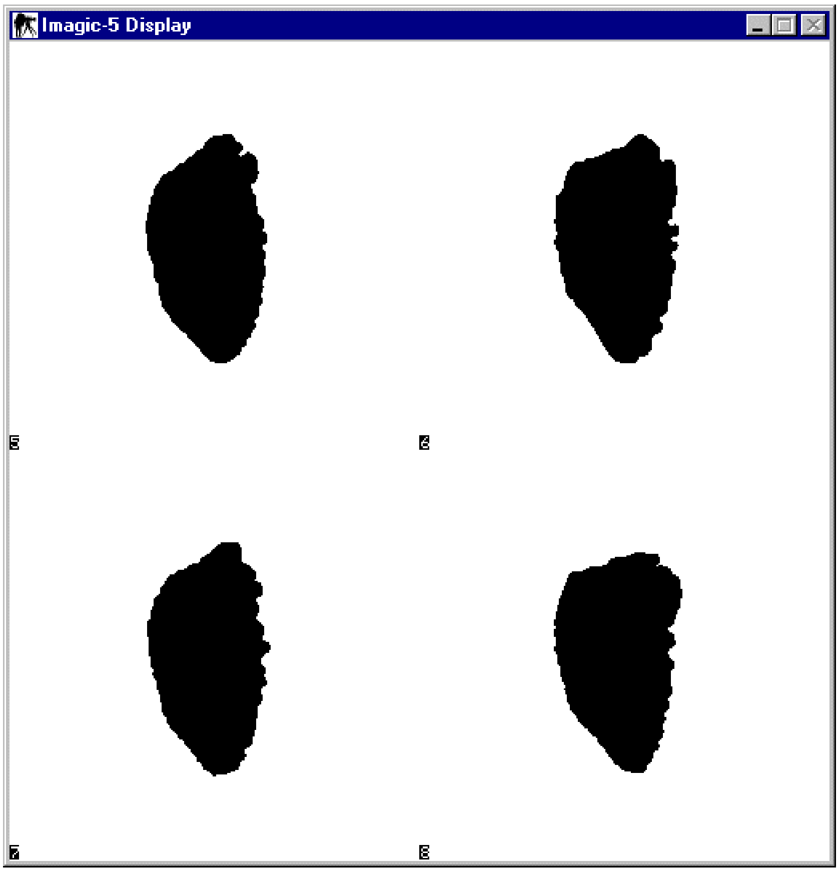
Here some additional DISPLAY features concerning FABOSA images should be noticed, namely window size and the scale factor. If the input file is specified the window size is already entered. The scale factor is usually one (see "Current DISPLAY settings").

In a single-frame displaying window, there is not much use in specifying the display size because the natural choice is the image size:



However, if you have small images, there is a possibility to display a number of images in a sequence.

Let us assume your images are 128 pixels x 128 pixels and the display window is set to 512 by 512 pixels then exactly four frames will be shown in the display at a time:



If the display window is set to 512 by 768 pixels then 24 frames at a time will be displayed.

If the image is larger than the window size it cannot be displayed in the original size and will therefore automatically be truncated. But you can also specify this truncation yourself by specifying a wanted scale factor. This scale factor will be applied before the image is rendered to the screen. This is important because most settings for the scale factors will cause alterations in the image appearance. For example if you have a black/white image with 1-pixel thick lines on it, and you set the scale factor < 1, some lines may fall 'between' the rendered grid points and thus may not be visible in the displayed image.

A final remark to the DISPLAY command: As you will notice, the display window is different from the console (MS-DOS) window. You will enter your commands into the console window, while the graphics is always shown in a separate DISPLAY window. This allows the use of the well known windows shortcut to copy the currently displayed image into the Pastebuffer. The shortcut keys are 'Control C'. Please make sure that, before using 'Control C', the display window is made the active one by clicking into it.

(4) Convert 2-D-images to NFD files

As mentioned before, the 2-D shape of an object is encoded in its contour line points. From these points so-called Normalised Fourier Descriptors (NFDs) are calculated and stored in a file as one-dimensional (1-D) data.

There are a couple of steps involved in the processing of the original 2-D-images.

- Input 2-D image data
- Calculate image pixel histogram
- Determine threshold value according to the procedure by Zack
- Apply threshold for binarisation into black/white image
- Sieve filtering using object analysis to extract a single object
- Edge calculation with subsequent spike and loop filtering
- Generation and sorting of the chain points for the closed contour line
- Interpolation of chain points using equal contour line step width
- Discrete Complex Fast Fourier Transformation of contour points to Fourier Descriptors (FDs)
- Normalisation of FDs into NFDs according to the defined reference system
- Definition of classification parameters
- Output data for 1-D NFD file

The command can be called by typing CREATE-NFD-CONTOUR:

```
IMAGIC-COMMAND: CREATE-NFD

** OTO_INPUT welcomes you **

Give input file, image loc#s                   : cod_imported
                                                        you select
Give output file, image loc#s                  : cod1d
                                                        you select
```

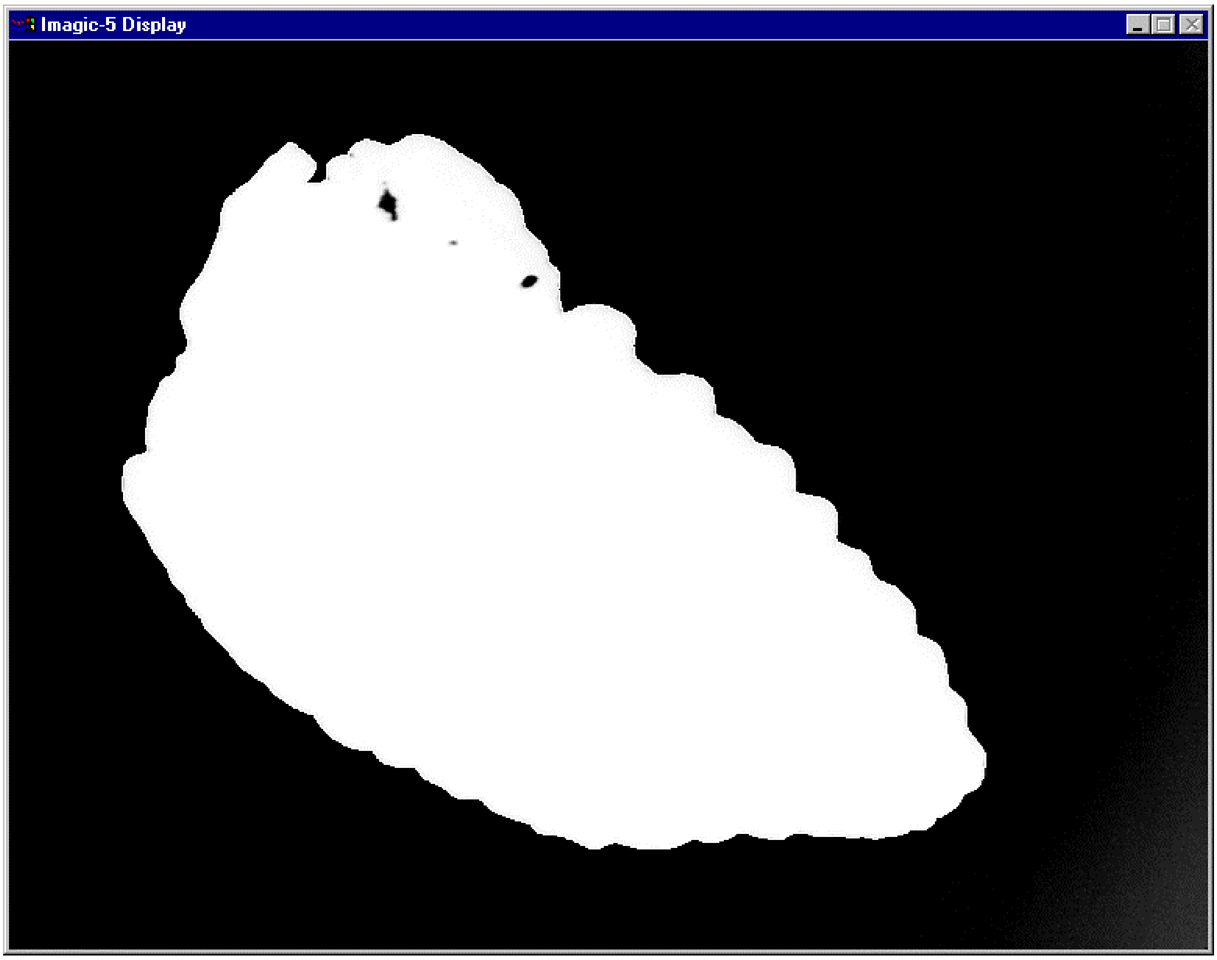
```
Threshold offset value                          : 50
                                                              you select,
                                                              see below

Do you need a log file? [NO]                    :             give a
VERTICAL or HORIZONTAL long axis [VERTICAL]     :             ?  for
Create 2D contour-overlay image output [NO]     :             further
Change the MAGNIFICATION default [NO]           :             help
```

Please note that the input file (here named cod_import) is the name of the IMAGIC-5 file which was previously created with EM2EM in mode FABOSA. To denote the special type of the output file, it is recommended to append '1d' to the rootname. Here cod1d is specified for the output. The "Threshold offset value" is an empirical value that depends on your input image contrast. The meaning of this number and for the other input items will be explained later. First the continuation of the command is described:

The command starts working and prints the following information:

```
Image name, parameters [and history]:
395      sagitta  cod                     15.6.97  Faroe Bank
Size:582,768 Loc: 1 Type:PACK  Cre.Date: 9/03/1999 Time: 14.25
EM2EM;

Detected        2056 pixels in contour #          1
```

Then CREATE-NFD-CONTOUR displays the found contour:



A large white object and a black background is displayed. At a closer look the contour line can be recognised as a white line around the object. There are some black spots inside the object. These have to be eliminated by one of the different sieve filtering procedures applied to the binarised image.

Please check the displayed image (see next chapter: "Analyse the contours in CREATE-NFD-CONTOUR"). The command now asks how to continue:

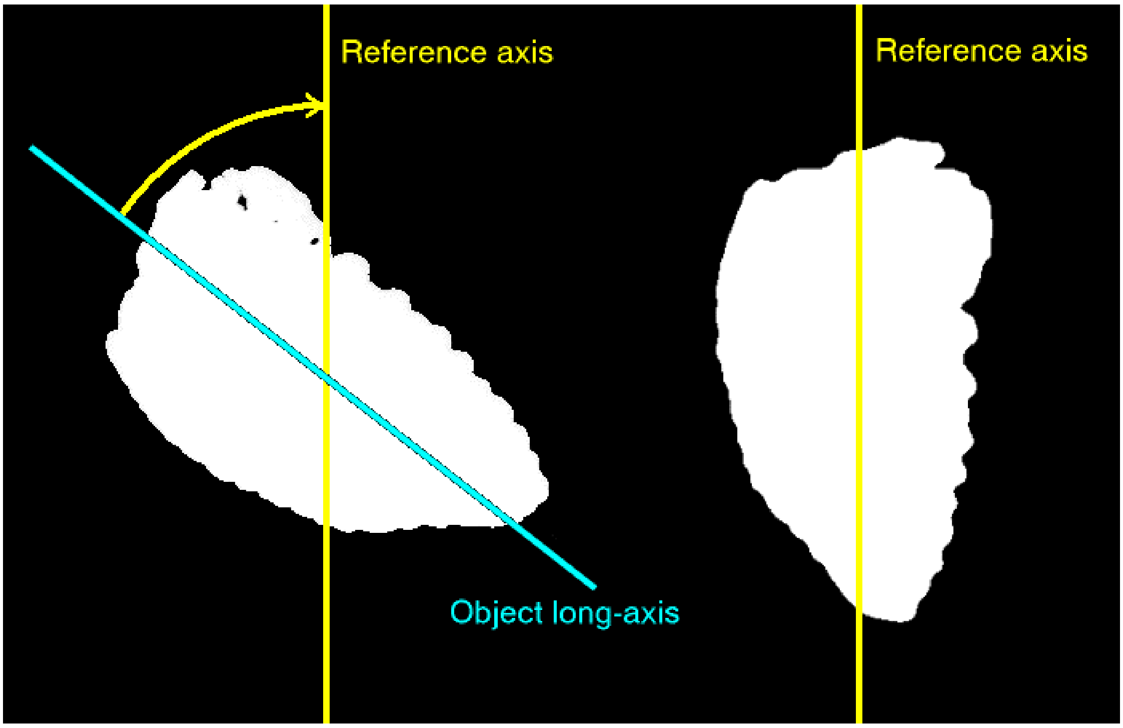
```
NEXT, REPEAT, SKIP, CONTINUOUS or END ? [NEXT]     :
```

The choices are:

NEXT: Continue with the next otolith image.
REPEAT: Repeat the contour finding. New parameters can be used.
SKIP: Skip this image.
CONTINOUS: Do not ask any more. The parameters are ok. Use these values for all following images.
END: Exit the command.

(5) Analyse the contour in CREATE-NFD-CONTOUR

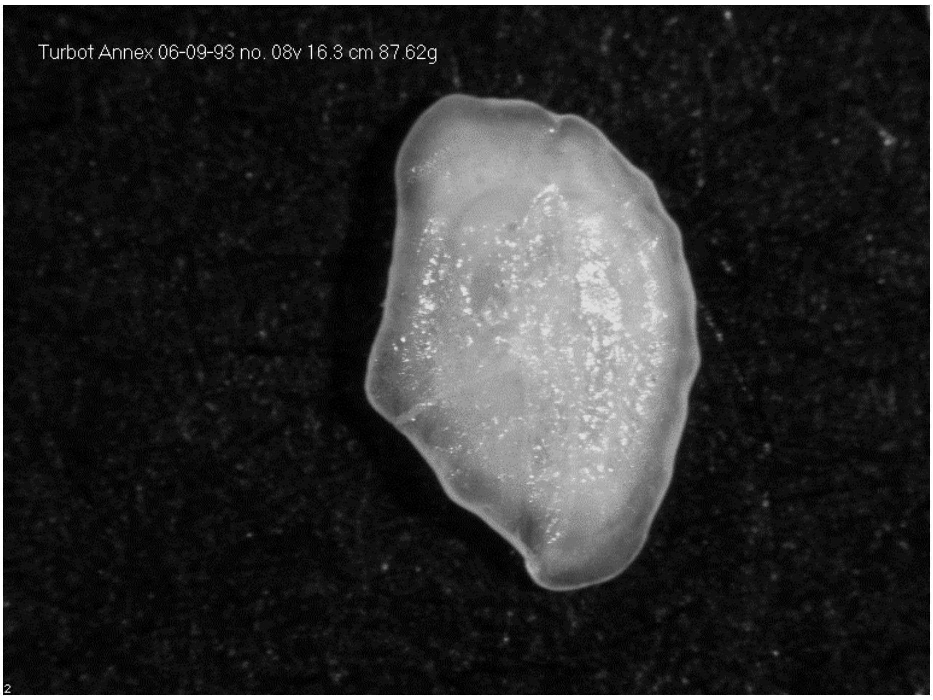
The command CREATE-NFD-CONTOUR has, of course, also generated the Fourier Descriptors of the contour and their Normalisation (NFDs). The normalisation can be envisioned as an alignment of the object long-axis with respect to the vertical reference axis as shown in the next figure:



The long-axis of the object will always be adjusted to the reference axis, i.e. objects will eventually be rotated so that the long-axis runs vertical. The program takes care that the smallest possible rotation will be applied in the normalisation step.

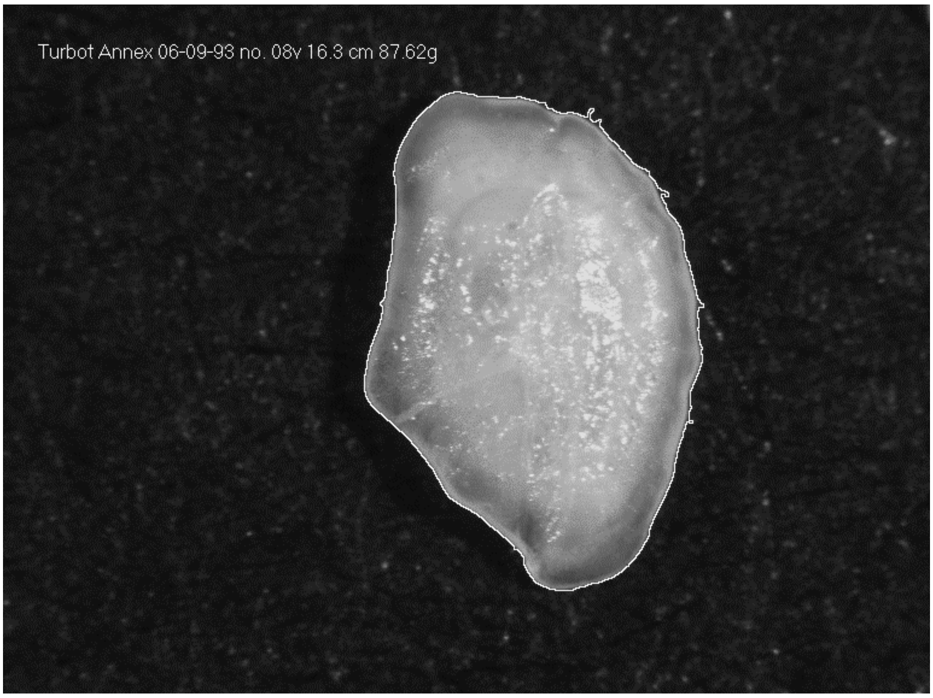
What happens if the object long-axis runs horizontally? The user can specify this. Refer to the example above. What happens is the following: An original image copy is treated as if it was rotated clockwise 90 degrees which causes the reference axis to run vertical. Normalisation is first done with the copy. Then the re-rotation back to the original orientation is considered.

The **threshold offset value** asked by CREATE-NFD-CONTOUR means the following: For each image the grey-scale histogram is calculated and analysed according to a procedure described in the literature (G.W. Zack, W.E. Rogers, and S.A. Latt, 1977). The method gives an estimate of the optimum threshold value for binarisation. Since this threshold is calculated using the individual image histograms the threshold will be characteristic for each image. However, the user may want to change the contour line to lie farther out or closer to the object. This can be arranged by specifying a "threshold offset value" which will additionally be applied to the threshold calculated using Zack's method. This is demonstrated in the next figures.



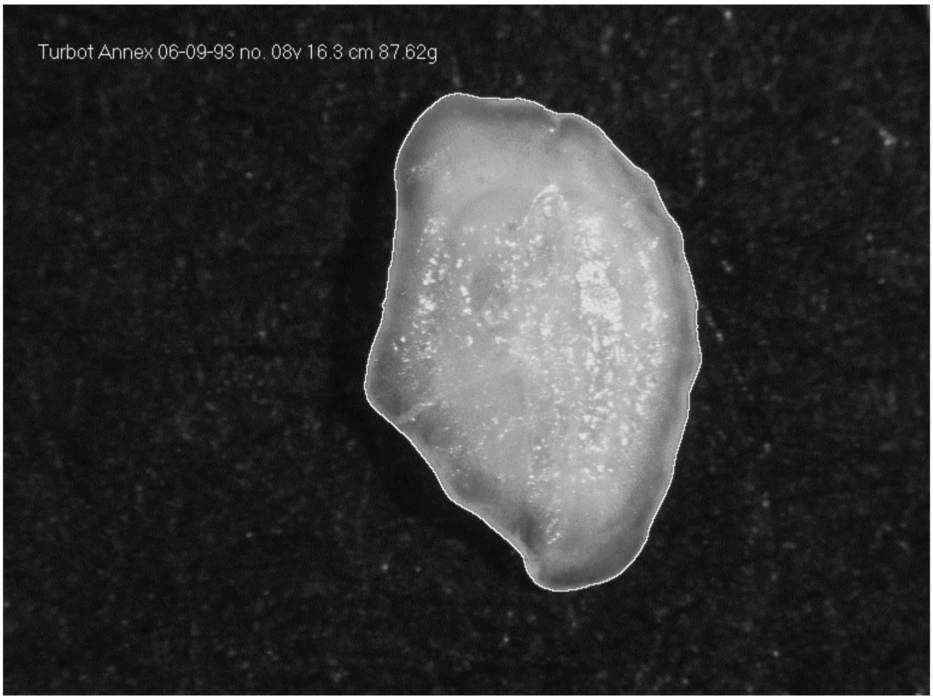
An original image of an otolith from turbot.

This example is a pretty bad image with respect to sharp and highly contrasted edges. Also, the background is not uniform. Don't forget to notice the text annotation which can disturb automatic contour finding.



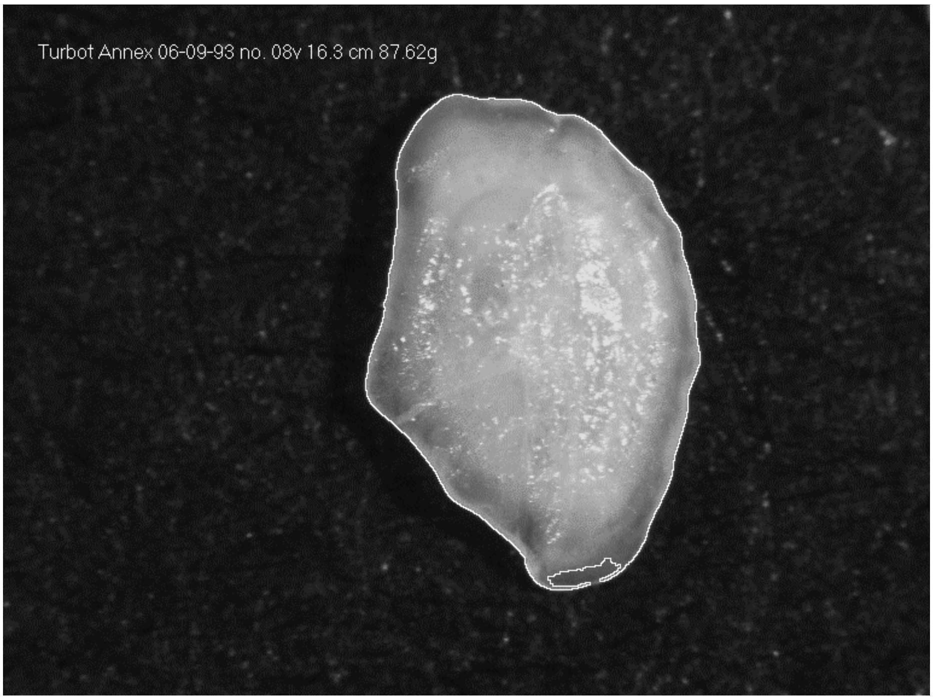
Contour line overlay for threshold offset = 0

If the threshold offset set to zero the original threshold calculated after Zack is used during binarisation, i.e. black and white conversion. The contour lies as far outside the object as possible thus getting as much into the object as possible. This greedy behaviour is not wanted. Please note the blobs and spikes.



Contour line overlay for threshold offset = 20

If the threshold offset is set to 20, the resulting contour line is pretty acceptable. Maybe one can even go up a little more. In practice one has to find a common value for the entire sequence of images.



Contour line overlay for threshold offset = 47

If the threshold offset is too large, the contour line leaks into the object at specific positions. This is happening at the bottom side, for example. Here, the contrast between background and edge is too low.

(6) Print out NFD information (if wanted)

If you are not really interested in methodology development, mathematics etc. you can skip this chapter and continue the 2-D shape analysis with (7).

The IMAGIC-5 command to print out all NFD information is PRINT-NFD-INFO:

```
IMAGIC-COMMAND: PRINT-NFD-INFO

** OTO_PRINTOUT welcomes you **

Give 1D otolith data file, image loc#s        : cod1d
                                                      you select
Text output file name plus extension [out.txt]: cod1d.txt

                                                      you select
```

You will get the following output (NFD information) for every 1-D image stored in the input file:

```
Image #            1


Sample info stored in the HED file

Image type         DAT(105)   =   -1.000000
Gaussnorm          DAT(120)   =    4711.000
Numeigen          IDAT(121)   =          69
Num_Image_Active  IDAT(122)   =          96
N_Metric           DAT(129)   =    1.000000
Active-image flag  DAT(130)   =    1.000000
Image ID           NAME(1:8)  = de1
Otolith Type       NAME(10:17)= sagitta
Species            NAME(19:48)= eel
Catch Date         NAME(50:57)= 18/5/92
Catch Site         NAME(59:80)= Lake Frisksjon
Magnification      DAT(123)   =   5.1282053E-03
Sulcus Side        DAT(124)   =    1.000000
Body Side          DAT(125)   =    1.000000
Sex                DAT(126)   =    2.000000
Fish Lengths [mm]   DAT(127)  =    483.0000
Fish Weight [g]    DAT(128)   =    151.0000

Normalised Fourier Descriptors (NFDs)

 0.0000000E+00
 0.0000000E+00
  10000.00
 6.2059559E-07
 -199.9085
  467.8286
  238.7708
  110.0508
    ...
    ...
    ...
  269.2481
  174.2882
 -254.9601
  121.4125
 -295.5876
  1114.021
 2.1777737E-06
```

```
Parameters stored in the IMG file

MAXDIM_DENS            271
A0_XORIG         1464047.
A0_YORIG         1419246.
A1_COS          -520070.0
A1_SIN          -110150.2
A1_MAGNITUDE     531606.9
A1_NORM          10000.00
K_SELECT                -1
N_COMBI                  1
AK_COS          -49282.55
AK_SIN           32840.42
NUM_DATA              4096
NEW_DIM                128
SIZE_FACTOR     4.9999999E-03
IXLP_ORIG              572
IYLP_ORIG              768

Subset of NFDs stored in DAT(131) to DAT(199)

 -199.9085
  467.8286
  238.7708
  110.0508
  39.40403
  75.77702
    ...
    ...
    ...
  121.4125
 -295.5876
  1114.021
 2.1777737E-06

DAT(199)=SIZE_FACTOR * A1_MAGNITUDE / A1_NORM

 0.2658034



Image #           2
```

etc.

Some comments:

Usually, complex Fourier coefficients obtained from a so-called FFT procedure are sorted in a very peculiar manner. Note the index numbering given below.
General scheme

index = 0, 1, 2, ..., N, -(N/2-1), ... , -1

For N = 128
index = 0, 1, 2 ..., 128, -127, -126, ..., -1

Each coefficient is a complex number and may therefore be denoted as A[index]_COS and A[index]_SIN according to Euler's formula. In the printout, the complex coefficients are listed as follows:

```
A[0]_COS
A[0]_SIN
A[1]_COS
A[1]_SIN
...
...
...
A[-1]_COS
A[-1]_SIN
```

Due to the normalisation procedure, some coefficients are always zero or have a fixed value. These are therefore left out in the parameters that are stored for use in classifications.

Besides the shape information there is one classification parameter reserved for the size (the co-called **size factor**). There is a line in the printout explaining how this size parameter was calculated.

(7) Multivariate Statistical Classification (MSA-CLASSIFY)

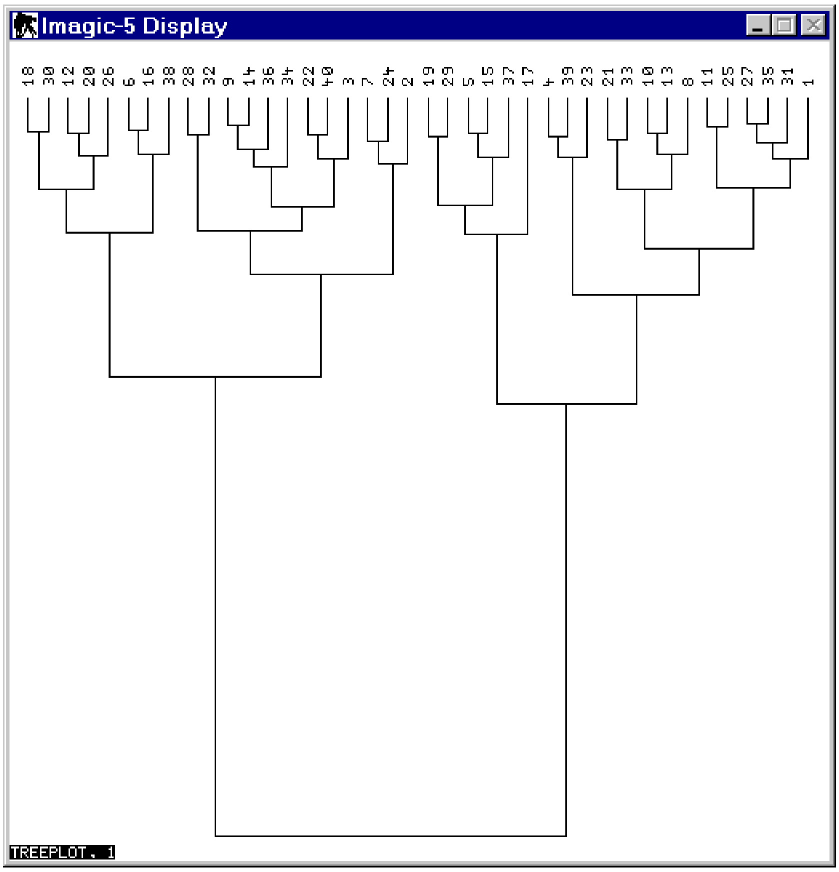
IMAGIC-5 offers sophisticated instruments for hierarchical classification (HAC). The two-step analysis begins with a Ward-type HAC, which is subsequently refined by the Moving-Elements method. The tools have been successfully used in numerous electron microscopical studies.

To classify the 2-D contours (NFDs) call IMAGIC-5 command MSA-CLASSIFY:

```
IMAGIC-COMMAND: MSA-CLASSIFY

** CLASSIFY welcomes you **

Input to be classified:
  IMAGES            PIXEL-VECTORS    SEQUENCES
  FABOSA-CONTOURS
Please specify option [IMAGES]            : fabosa
Input images file (treated by MSA)        : cod1d
Percentage of images to be ignored [0]    : 0

                                   always use 0 for FABOSA contours

Active factors for classification [69]    : 69

                                   use 69 for FABOSA contours

Try a tricky normalisation? [NO TRICKS]   : no
Weight the factorial coordinates [NO]     : no
Plot the classification trees [YES]       : YES

                                   for FABOSA, use yes to get the
                                   tree plots

Maximal amount of other print output      : no
Classification (MSA) maps stored [NO]     : no
What NUMBER of classes do you wish ?      : 40

                                   you select! Use the tree plot to get
                                   an idea (see below)

Name for output result files              : cod1d_classes_1

                                   you select

Mass threshold parameter ? [0]            : 0
```

When dealing with medium size data-sets, a graphical representation of the hierarchical classification tree can be helpful to find the number of classes into which the data-set should be subdivided. Use command DISPLAY to visualise the tree plot.

In the following figure an example of a tree plot is shown. The classification was done for a data-set of 20 left-body-side and 20 right-body-side otoliths. The obvious division into two major classes is therefore expected.



Please also carefully check the results output text file which gives you many statistical measures and information concerning the classification calculated.

If you are not satisfied with the classification because of a wrong class number chosen etc. you can re-do the classification with MSA-CLASSIFY (7) until the classification is correct.

(8) MSA-SELECT-NFD

If the objects in a data-set have been subjected to a classification, usually there are a lot of resulting classes to which the object's membership can be assigned. It would be tedious to select the individual entries belonging to the same class one-by-one from the large set.

This task can easily be performed with command MSA-SELECT-NFD. One of the options of this command is EXTRACT which selects class members and creates a separate NFD file for each class:

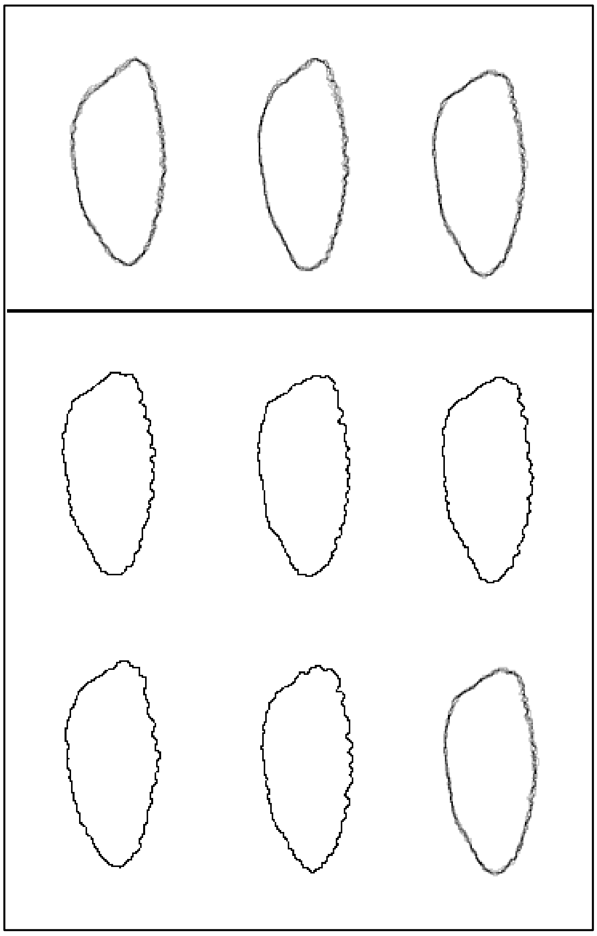
```
IMAGIC-COMMAND: MSA-SELECT-NFD

** OTO_CLASSWORK welcomes you **

Mode of operation:
  EXTRACT  AVERAGE
Please specify option [EXTRACT]          : EXTRACT
1D otolith input file (no loc#s)         : cod1d
Associated classification (.cls) file    : cod1d_classes_1
Output file rootname                     : cod1d_class
```

(9) MSA-AVERAGE-NFD

You may also want to create average contours within every class. You can easily perform this averaging by using option AVERAGE in command MSA-AVERAGE-NFD. It creates averages within every class and stores the average in a NFD (1-D image) file:

```
IMAGIC-COMMAND: MSA-AVERAGE-NFD

** OTO_CLASSWORK welcomes you **

Mode of operation:
  EXTRACT  AVERAGE
Please specify option [EXTRACT]          : AVERAGE
1D otolith input file (no loc#s)         : cod1d
Associated classification (.cls) file    : cod1d_classes_1
Output file rootname                     : cod1d_average

                                                        you select
```

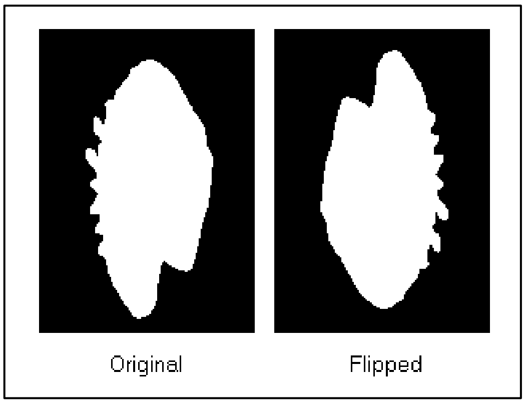
Examples of such averaged NFDs (re-converted and displayed as real images, see (10)) overlaid with the contours of the class members are shown in the next figures.



The upper row shows three average-contours. Five contours and the average thereof are shown in the bottom rows.

(10) FLIP-NFD-UPSIDE-DOWN

You can also flip contours (NFDs) upside-down which might be helpful for some specific analysis (merging right and left otoliths to get larger data-sets, for example):



(NFDs re-converted to images)

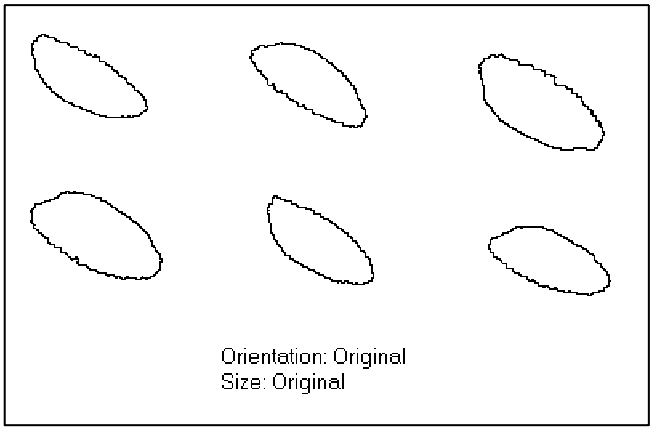
(11) GENERATE-IMAGE-FROM-NFD

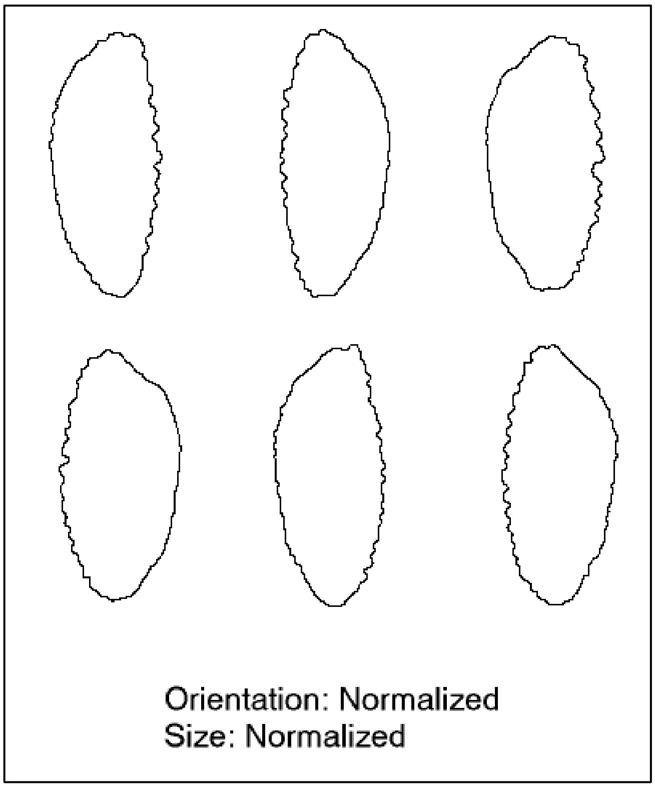
Although derived from 2-D-images the 1-D shape descriptors (NFDs) itself are highly abstract. Therefore, you may want to convert processed NFDs back to an image representations. The corresponding command is GENERATE-IMAGE-FROM-NFD and has the following options:

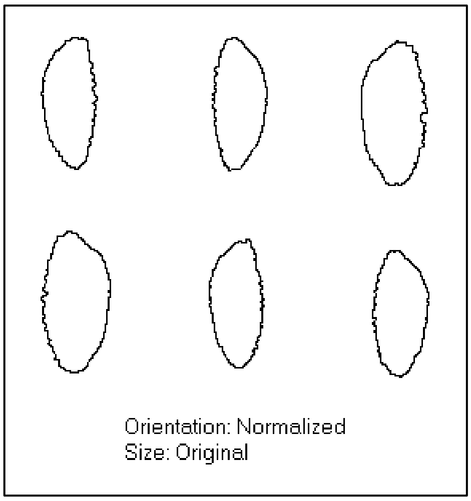
- ➢ Background: black or white
- ➢ Draw Style: draw contour line only or filled objects
- ➢ Size of Objects:
  - small reproduction
  - draw all objects at the same scale (is aware of microscope and camera magnification)
  - reproduce original size
- ➢ Orientation of Objects:
  - reproduce original orientation
  - show objects in normalised orientation

```
IMAGIC-COMMAND: GENERATE-IMAGES

** OTO_GENIM welcomes you **

1D otolith data file, image loc#s   : cod1d_average
                                                          you select
2D image output file, image loc#s   : cod1_average
                                                          you select
Colour of background:
  BLACK  WHITE
Please specify option               : WHITE
                                                          you select
Mode of output image:
  CONTOUR LINE  FILLED OBJECT
Please specify option               : CONTOUR
                                                          you select
Size of reconstructed 2D image:
  SMALL  ORIGINAL  ABSOLUTE
Please specify option               : SMALL
                                                          you select
Mode of orientation:
  ORIGINAL  NORMALISED
Please specify option               : NORM
                                                          you select
```

The following figures will show some examples:







(12) MSA-Classification of the original images

This is not the usual way to analyse contours. But for some methodology reasons etc. you may want to analyse the original images. Please note that in contrast to NFDs these images may not be aligned and may not be normalised so that you cannot directly compare and classify the images. There are a number of alignment commands available in IMAGIC-5, like ALIGN-DIRECT, MULTI-REFERENCE-ALIGN etc. Please use MENU * ALIGN and HELP * ALIGN to get detailed help.

Instead of using steps (4) - (7) you can analyse the original images with commands MSA-RUN, MSA-CLASSIFY and MSA-SUM:

MSA-RUN first creates a so-called MSA hyper space and applies a data reduction algorithm:

```
IMAGIC-COMMAND: MSA-RUN

** MSA welcomes you **

Choose mode: FRESH-MSA, REFINE [F]    : fresh
Distance:
EUCLIDIAN, CHISQUARE,MODULATION       : euclidian

                                          you select,
                                          for FABOSA use EUCLIDIAN

Filename input aligned "images"       : cod_align

                                          you select
```

```
Are there any inactive images? [NO]     : no
                                    some images can be excluded
Give input MSA mask file [none]         : none
                                    give "none" if no mask is wanted
Eigenimages output file                 : cod_eigenim_1
Pixel coordinates output file           : cod_pixcoos_1
Eigenpixel vectors output file          : cod_eigenpix_1
Give number of iterations (<64)         : 24
                                    you select, 24 usually is ok
Give number of eigenimages (<69)        : 69
                                    you select, suggested is: 69
Overcorrection factor (0< ocf< 0.9)     : 0.8
Rootname for registration files, no ext.: cod_msa_1
```

Then start classifying.

```
IMAGIC-COMMAND: MSA-CLASSIFY

** CLASSIFY welcomes you **

Input to be classified:
  IMAGES           PIXEL-VECTORS    SEQUENCES
  FABOSA-CONTOURS
Please specify option [IMAGES]            : IMAGES
Input images file (treated by MSA)        : cod_align
                                    the MSA-RUN treated images !!
Percentage of images to be ignored [0]    : 0
Active factors for classification [69]    : 69
                                    suggested: 69
Try a tricky normalisation? [NO TRICKS]   : no
Weight the factorial coordinates [NO]     : no
Plot the classification trees [YES]       : YES
                                    use yes to get the tree plots
Maximal amount of other print output      : no
Classification (MSA) maps stored [NO]     : no
What NUMBER of classes do you wish ?      : 40
                                    you select! Use the tree plot to
                                    get an idea (see chapter (7))
Name for output result files              : cod1d_classes_1
                                    you select
Mass threshold parameter ? [0]            : 0
```

Please note:

If you want to re-do the classification (to change the number of classes, for example) you don't have to repeat MSA-RUN. You can simply re-do MSA-CLASSIFY.

If you want to create the class averages (class-sum images) you can use command MSA-SUM:

```
IMAGIC-COMMAND: MSA-SUM

** CLASSUM welcomes you **

Input aligned images                          : cod_align
                                    the MSA-CLASSIFY treated images !!
INPUT "CLS" file (NO ext.)                    : cod_classes_1
                                    output file from MSA-CLASSIFY
Output class averages                         : cod_classums_1
Downweight small classes? [YES]               : yes
Do you want to produce summing STATISTICS : no
Fraction of worst class members to ignore : 0
                                                        you select
```

Use DISPLAY to visualise the class averages.

If you want to extract the members of specific classes you can use command MSA-EXTRACT:

```
IMAGIC-COMMAND: MSA-EXTRACT

** CLASSORT welcomes you **

Input images, NO loc#s                  : cod_align
                                    the MSA-CLASSIFY treated images !!
Input classification (.cls) file        : cod_classes_1
                                    output file from MSA-CLASSIFY
All classes wanted [NO]                 : NO              you select
Where to get the wanted classes:
  PLT FILE  INTERACTIVE
Please specify option [INTER]           : INTER           you select
```

```
Class(es) wanted: [1;3]                  : 1-2;4;7-10

  Am using the following locations:
     1 ::     1     2     4     7     8     9    10

One output file only                           :
Root name for output files, NO loc#s     : cod_class_
                                                      you select

Place class-sum image into last location : NO
                                                      you select
```

(13) Publication

Write a nice paper for the Canadian Journal of Fish Biology, etc.

This writing you have to do yourself as, unfortunately the corresponding IMAGIC-5 commands (WRITE-FOR-CAN.J.FISH.BIOL, etc.) are still in preparation.

Some References

WebPages:

www.ImageScience.de
www.ImageScience.de/fabosa
www.ImageScience.de/otolith

IMAGIC-5

M van Heel and W Keegstra (1981) IMAGIC: A fast, flexible and friendly image analysis software system. Ultramicroscopy 7: 113-130.

M van Heel, G Harauz, EV Orlova, R Schmidt and M Schatz (1996) A new generation of the IMAGIC image processing system. J. Struct. Biol. 116: 17-24.

Fourier Descriptors

TP Wallace and PA Wintz (1980) An efficient three-dimensional aircraft recognition algorithm using normalised Fourier descriptors. Comp. graphics and image proc. 13: 99-126.

PE Lestrel (ed.) (1999) Fourier descriptors and their application in biology. Cambridge University Press, USA.

Correlation functions

M van Heel, M Schatz and EV Orlova (1992a) Correlation functions revisited. Ultramicroscopy 46: 304-316.

Multivariate Statistical Analysis and Classification

M van Heel and J Frank (1981) Use of multivariate statistics in analyzing the images of biological macromolecules. Ultramicroscopy 6: 187-194.

M van Heel (1984) Multivariate statistical classification of noisy images (of randomly oriented biological macromolecules). Ultramicroscopy 13: 165-183.

M van Heel (1989) Classification of very large electron microscopical data sets. Optik 82: 114-126.

Image analysis

M Schatz, EV Orlova, P Dube, H Stark, F Zemlin and M van Heel (1997) Angular reconstitution in 3D electron microscopy: Practical and technical aspects. Scanning Micros. 11 (1997) 179-193.

M van Heel, EV Orlova, G Harauz, H Stark, P Dube, F Zemlin and M Schatz (1997) Angular reconstitution in 3D electron microscopy: Historical and theoretical aspects. Scanning Micros. 11 (1997) 195-210.