EcoTransLearn R-package

Version 1.0-0

USER MANUAL

G. WACQUET

IFREMER
LABORATOIRE ENVIRONNEMENT RESSOURCES
CENTRE MANCHE MER DU NORD

TABLE OF CONTENTS

INTRODUCTION

In recent years, Deep Learning (DL) has been increasingly used in many fields, in particular in image recognition, due to its ability to solve problems where traditional machine learning algorithms fail. However, building an appropriate DL model from scratch especially in the context of ecological studies, such as monitoring marine ecosystems, is a difficult task due to the dynamic nature and morphological variability of living organisms, as well as the high cost in terms of time, human resources, and skills required to label a large number of training images. To overcome this problem, Transfer Learning (TL) can be used to improve a classifier by transferring information learnt from many domains thanks to a very large training set composed of various images, to another domain with a smaller amount of training data. To compensate the lack of “easy-to-use” software optimized for ecological studies, we propose the *EcoTransLearn* R-package, which allows greater automation in classification of images acquired with various devices, thanks to different TL methods pre-trained on the generic ImageNet dataset.

INSTALLATION AND EXECUTION

R-package installation

The version 1.0-0 of the *EcoTransLearn* package needs a recent version of R (version 4.0.x or upper). It can be directly downloaded on GitHub at: *https://github.com/IFREMER-LERBL/EcoTransLearn*.

By double-clicking on the R icon on the desktop, or by selecting R in the start menu, a window appears on the screen: this is the R console. This allows to control R directly by command lines. It also allows to display the main results and messages of the different actions performed with *EcoTransLearn*.

The R-packages needed by *EcoTransLearn* (*colorRamps, ggplot2, grid, jpeg, mapplots, maps, randomForest, reticulate, SDMTools, shapefiles, stringr, svDialogs, svMisc, tcltk2, tiff, zooimage*) can be installed directly from the R console, by typing:

install.packages(c("colorRamps","ggplot2","grid","jpeg","mapplots","maps","randomForest", "reticulate","SDMTools","shapefiles","stringr","svDialogs","svMisc","tcltk2","tiff","zooimage"))

Then choose a mirror (default: 0-cloud) to start download and installation.

Python installation

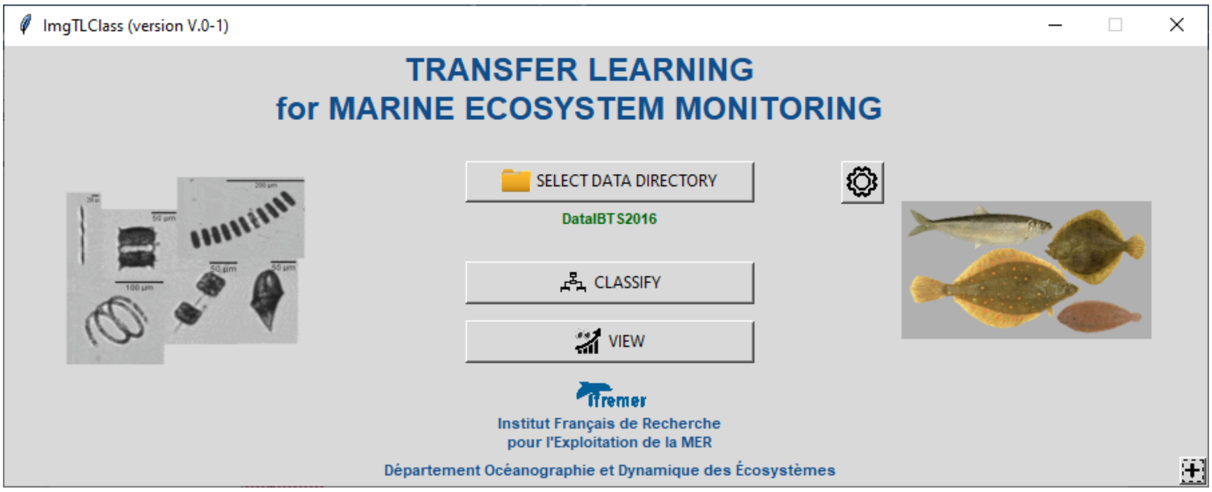
Anaconda is a scientific distribution of Python, which allows to use several applications (such as Spyder, Jupyter Notebook, …) and to manage different libraries. It can be downloaded on the website: https://docs.anaconda.com/anaconda/navigator/install/. The *EcoTransLearn* package needs the version 3.7 (or upper) of Python.

Once the distribution installed, the Python libraries needed by *EcoTransLearn* (*matplotlib, numpy, pandas, sklearn, tensorflow*) can be installed directly by opening an Anaconda Prompt, and by entering the commands:

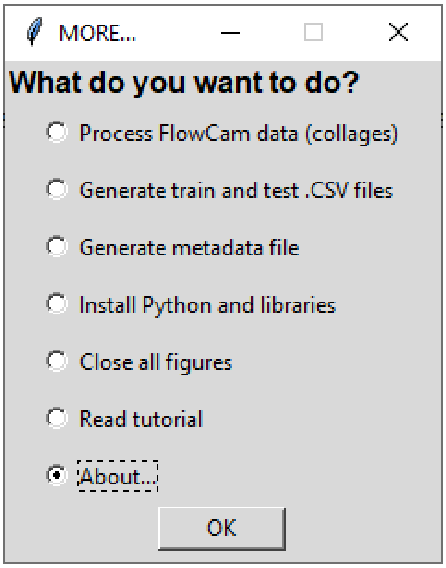
conda install matplotlib
conda install numpy
conda install pandas
conda install sklearn
conda install tensorflow

Launching the Graphical User Interface

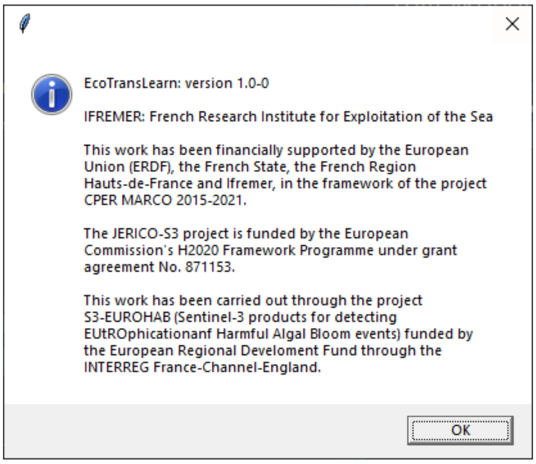
Once the installation of the packages is finished, it is possible to make sure that the previous steps run smoothly by checking that the installed version is 1.0-0. To do this, first type in the R console: **library(EcoTransLearn)**, to load the package, then: **EcoTransLearn()**, to launch the Graphical User Interface (GUI):



Click on the **+** button (bottom right). A new window appears:



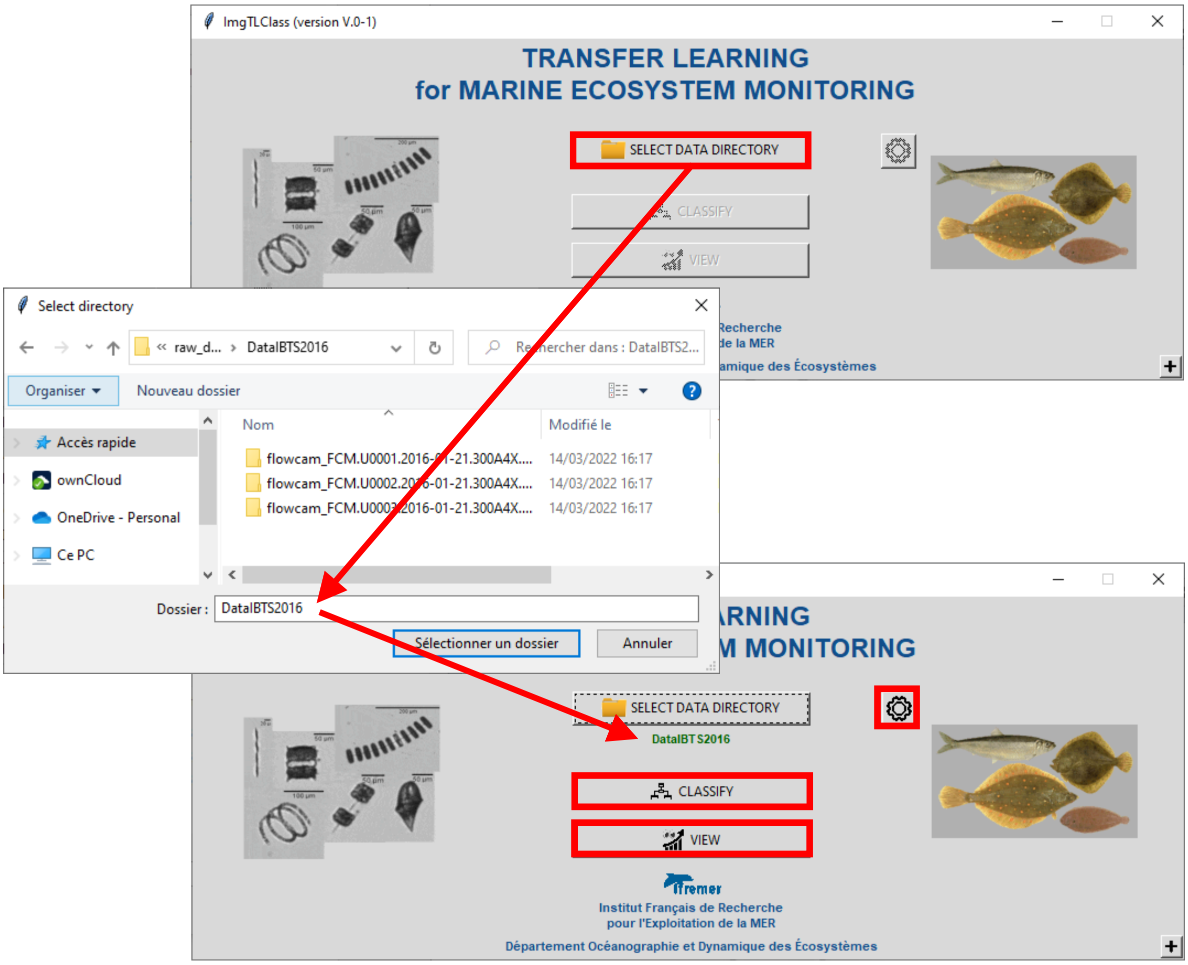
By selecting **About...**, a dialog box appears and informs the user of the *EcoTransLearn* version.



USE OF THE GRAPHICAL USER INTERFACE

For routine use of *EcoTransLearn*, an ergonomic Graphical User Interface is available.

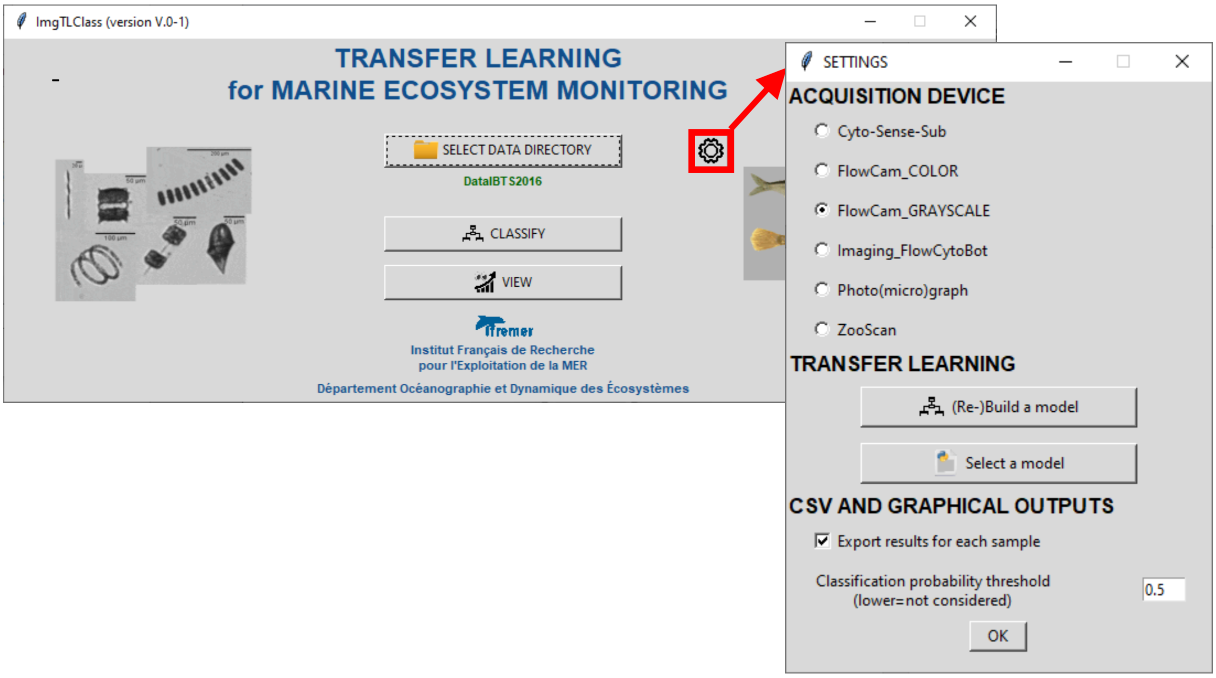
DATA SELECTION button



Click on the **SELECT DATA DIRECTORY** button, select the directory containing the images to classify, then validate by clicking on **OK**: the name of the selected directory is displayed below the selection button, and the **SETTINGS**, **CLASSIFY** and **VIEW** buttons become active.

SETTINGS button

After selection of the input data, the settings window is automatically displayed (without clicking on the **SETTINGS** button). However, it is possible to redisplay this window at any time by clicking on this button.

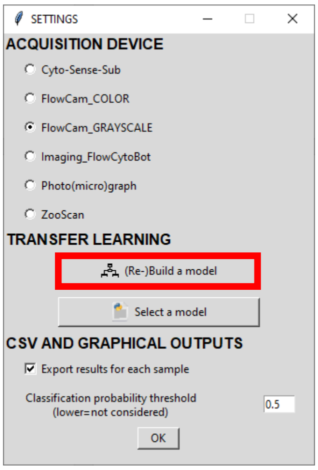


- **ACQUISITION parameters**

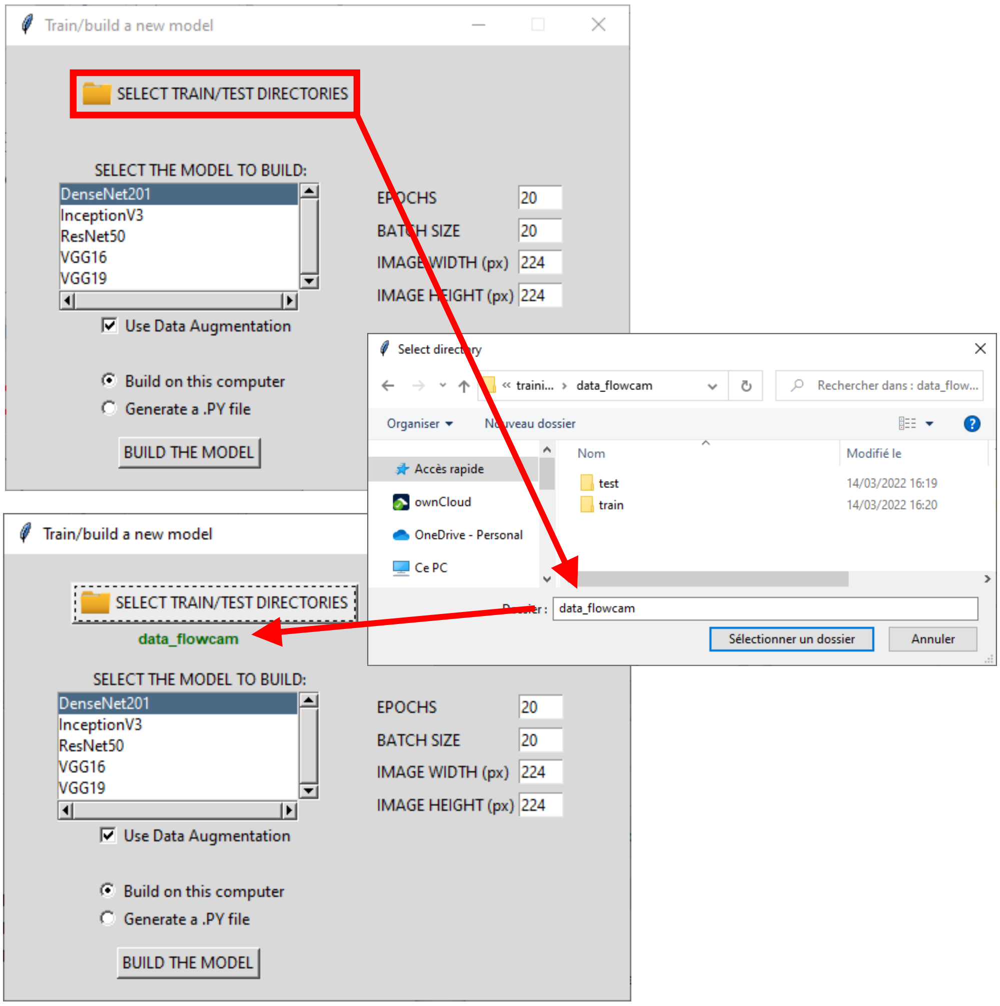
Choose the kind of instrument used for image acquisition.
Note: for the FlowCam_COLOR and the FlowCam_GRAYSCALE, it is possible to directly output the raw data from the instruments (collages). The package allows to directly create the vignettes (1 image per particle) from the collages and the lst file.

- **CLASSIFICATION parameters**
  - **(Re-)Build a model**

    This button allows to build (or rebuild) a classification model by Transfer Learning using a data set including a training set (directory must be named **train**) for learning, and a test set (directory must be named **test**) for validation and evaluation.
    All unsorted images are bulked into these directories. Two CSV files (named 'train.csv' and 'test.csv') allows listing the group of each image (see **Generate train and test .CSV files** section).



By clicking on this button, a new window appears.



Click on the **SELECT TRAIN/TEST DIRECTORIES** button, select the directory containing the two sub-folders **train** and **test**, then confirm by clicking **OK**: the name of the selected directory is then displayed below the selection button

It is then possible to select different Convolutional Neural Network (CNN) architectures in the **SELECT THE MODEL TO BUILD** list, among:

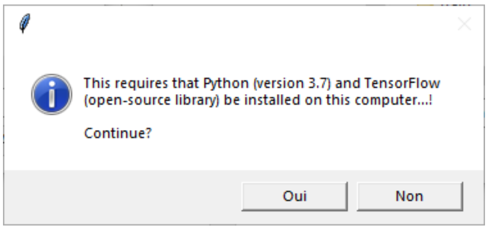
- DenseNet201
- InceptionV3
- ResNet50
- VGG16
- VGG19

and adjust parameters related to images and training step, by setting values for **EPOCH** (default=20), **BATCH SIZE** (default=20), **IMAGE WIDTH** (default=224) and **IMAGE WEIGHT** (default=224), but also choose the possibility of using the technique of data augmentation (**Use Data Augmentation**). In the case of a training set with few images, this option is used to automatically generate additional images from the original images in the training set, by applying geometric transformations such as rotations (by default, rotation_range=45) and horizontal and vertical flips (by default, horizontal_flip=True and vertical_flip=True).

The last step is to choose the material on which to build and adapt the classification model. Depending on the selected option, and after clicking on the **BUILD THE MODEL** button:

- **Build on this computer**

  A dialog box appears:

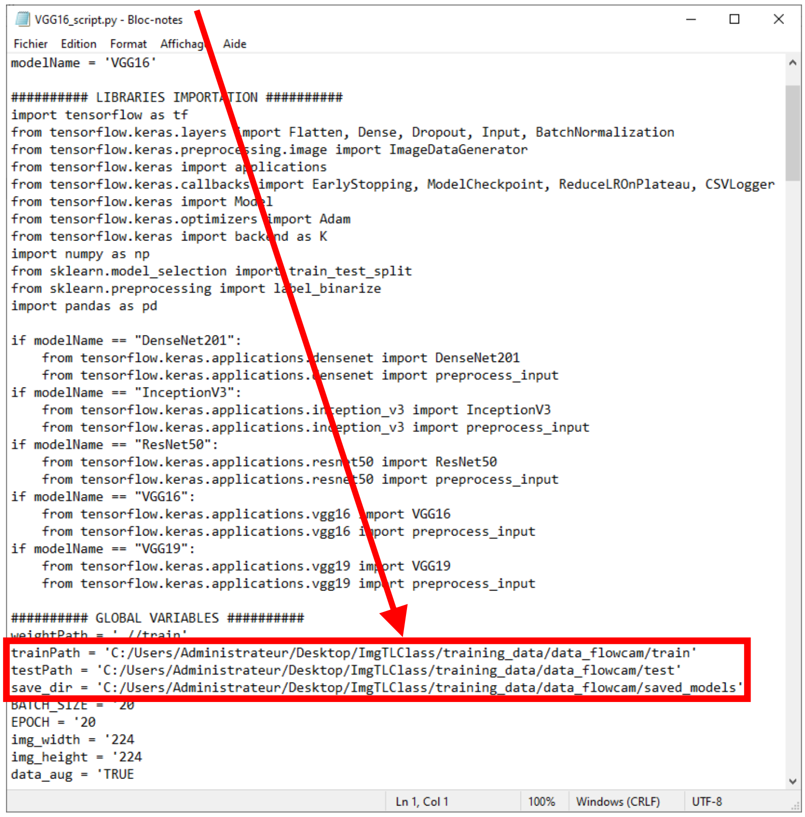


  <u>Warning</u>: the training duration can be long (several hours) depending on the number of images in the training set and according to the parameters defined in the previous step.

- **Generate a .PY file**

  A script is automatically created, and can be run on other hardware (dedicated computer, calculation server, etc.).

  *<u>Warning</u>: three variables must be modified according to the data location (‘**trainPath**’, ‘**testPath**’ and ‘**save_dir**’).*



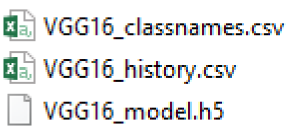
```
VGG16_script.py - Bloc-notes
Fichier Edition Format Affichage Aide
modelName = 'VGG16'

########## LIBRARIES IMPORTATION ##########
import tensorflow as tf
from tensorflow.keras.layers import Flatten, Dense, Dropout, Input, BatchNormalization
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras import applications
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint, ReduceLROnPlateau, CSVLogger
from tensorflow.keras import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras import backend as K
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import label_binarize
import pandas as pd

if modelName == "DenseNet201":
    from tensorflow.keras.applications.densenet import DenseNet201
    from tensorflow.keras.applications.densenet import preprocess_input
if modelName == "InceptionV3":
    from tensorflow.keras.applications.inception_v3 import InceptionV3
    from tensorflow.keras.applications.inception_v3 import preprocess_input
if modelName == "ResNet50":
    from tensorflow.keras.applications.resnet50 import ResNet50
    from tensorflow.keras.applications.resnet50 import preprocess_input
if modelName == "VGG16":
    from tensorflow.keras.applications.vgg16 import VGG16
    from tensorflow.keras.applications.vgg16 import preprocess_input
if modelName == "VGG19":
    from tensorflow.keras.applications.vgg19 import VGG19
    from tensorflow.keras.applications.vgg19 import preprocess_input

########## GLOBAL VARIABLES ##########
weightPath = './train'
trainPath = 'C:/Users/Administrateur/Desktop/ImgTLClass/training_data/data_flowcam/train'
testPath = 'C:/Users/Administrateur/Desktop/ImgTLClass/training_data/data_flowcam/test'
save_dir = 'C:/Users/Administrateur/Desktop/ImgTLClass/training_data/data_flowcam/saved_models'
BATCH_SIZE = '20
EPOCH = '20
img_width = '224
img_height = '224
data_aug = 'TRUE
Ln 1, Col 1    100%    Windows (CRLF)    UTF-8
```

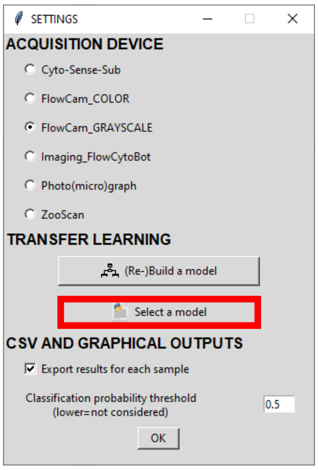
At the end of the execution of the script, a new folder is created at the root of the training/test set. In this directory (named ‘***saved_model***’), a ZIP file is saved and contains all the information necessary for the classification of a new set of images.

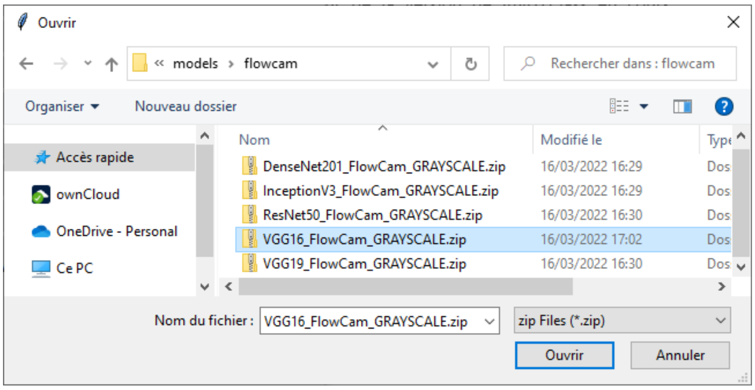


modelName_classNames.csv contains the class names in the training set.
modelName_history.csv contains the evaluation scores of the built model.
modelName_model.h5 is the model built with Python.

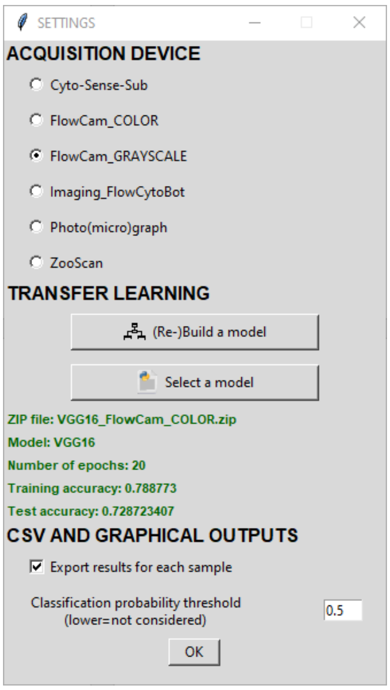
- **Select a model**

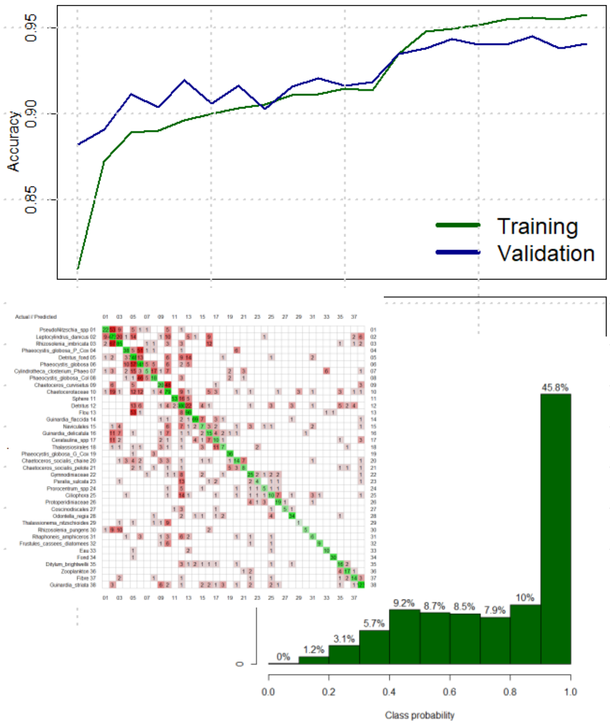
This button allows to choose a model for the automated classification of the images contained in the directory selected during the first step (see section “Selection of input data”).





Select the ZIP file generated during the previous step, then validate by clicking on **Open**: information on the model performance (calculated during the training and validation steps) is then displayed below the selection button, as well as **Accuracy** (≈ percentage of correctly classified data) and **Loss** (≈ distance between actual data and predicted data) curves, confusion matrix, etc.





- **OUTPUTS parameters**
  - **Export results for each sample**

This option allows to export the results for each sub-folder of the directory selected during the first step (see section "Selection of input data").

For each processed sample, three CSV files are created:

- sampleName.csv

Sample	Group	Relative	Count	Prob_threshold	Percent_used
flowcam_FCM.U0001.2016-01-21.300A4X.01	Asterionellopsis	2.941176471	1	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	C_curvisetus	5.882352941	2	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	C_danicus	8.823529412	3	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	C_socialis	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Ciliophora	5.882352941	2	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Dactyliosolen	8.823529412	3	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Dytilum	8.823529412	3	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	G_flaccida	8.823529412	3	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	G_striata	5.882352941	2	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Gymnodinium	2.941176471	1	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Lauderia	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Leptocylindrus	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Odontella	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	P_globosa	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Pleuro_Gyrosigma	5.882352941	2	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Prorocentrum	14.70588235	5	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	PseudoNitzschia	2.941176471	1	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Rhizosolenia	11.76470588	4	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Thalassionema	2.941176471	1	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Thalassiosira	2.941176471	1	0.75	89.88476312

with Sample: the name of the sample
Group: list of all groups in the training set
Relative: relative abundance (in %) of each group
Count: absolute abundance of each group
Prob_threshold: threshold for class probability (see next section)
Percent_used: percentage of images with class probability upper than threshold

- sampleName_CLASSIF.csv

Filename	Class
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_10.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_100.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1000.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1001.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1002.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1003.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1004.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1005.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1006.jpg	Ciliophora
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1007.jpg	Odontella
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1008.jpg	dark
unknown\flowcam_FCM.U0003.2016-01-21.300A4X.01_1009.jpg	fiber

with Filename: the name of the image file
Class: the predicted class of the image

- sampleName_PRED.csv

Asterionellopsis	C_curvisetus	C_danicus	C_socialis	Ciliophora	Dactyliosolen	Dytilum	G_flaccida
3.52E-09	1.24E-08	4.22E-10	3.65E-11	0.000285448	1.65E-06	9.95E-09	7.16E-10
1.34E-11	2.78E-11	7.02E-14	1.16E-16	4.84E-10	5.95E-14	3.10E-13	3.12E-16
1.86E-12	2.97E-09	1.78E-13	8.65E-14	8.47E-05	1.23E-07	4.80E-09	1.05E-11
9.69E-17	1.54E-11	3.29E-18	7.69E-20	3.97E-08	1.30E-10	1.50E-12	2.34E-17
6.27E-08	5.92E-07	3.55E-09	2.57E-09	7.62E-06	1.90E-06	1.73E-07	1.10E-07
1.42E-08	1.77E-08	5.07E-08	7.74E-11	1.00E-05	1.52E-07	1.58E-08	1.10E-11
1.01E-24	1.02E-15	9.25E-27	4.89E-27	3.53E-14	3.10E-17	2.96E-18	2.73E-22
3.96E-15	1.95E-12	1.79E-16	2.53E-18	6.86E-06	1.27E-08	2.07E-13	3.37E-14
6.43E-15	2.69E-09	8.91E-14	5.48E-17	0.001769048	2.66E-08	1.20E-12	4.52E-14

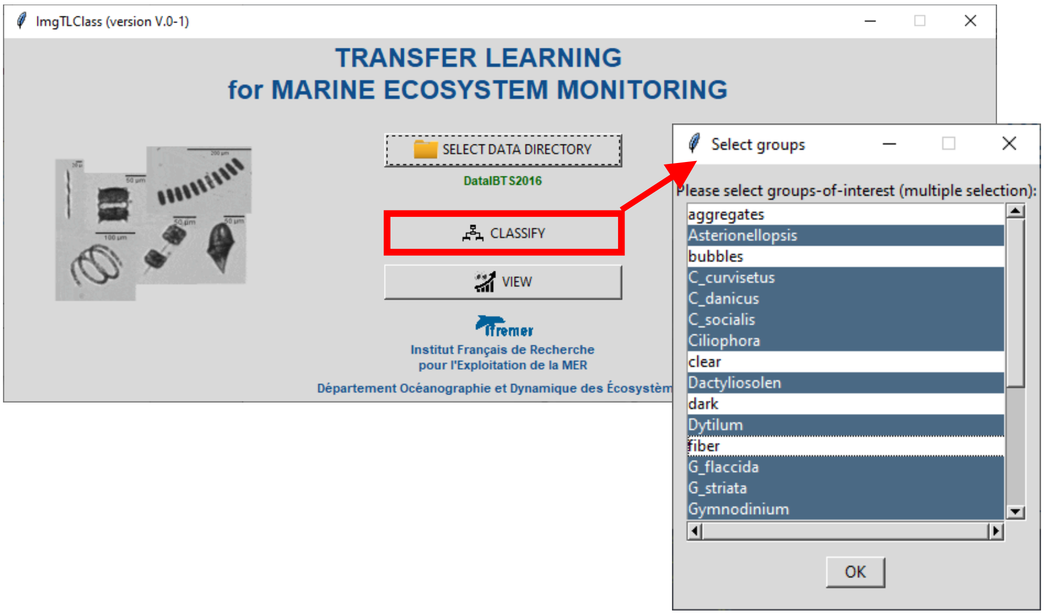
This file contains the class probabilities for each image file and group.

- **Classification probability threshold**

  The threshold value defined at this step allows to take into account only the images having a probability of "good" classification greater than this threshold. To take into account all the images, this value must be set to 0.

CLASSIFY button

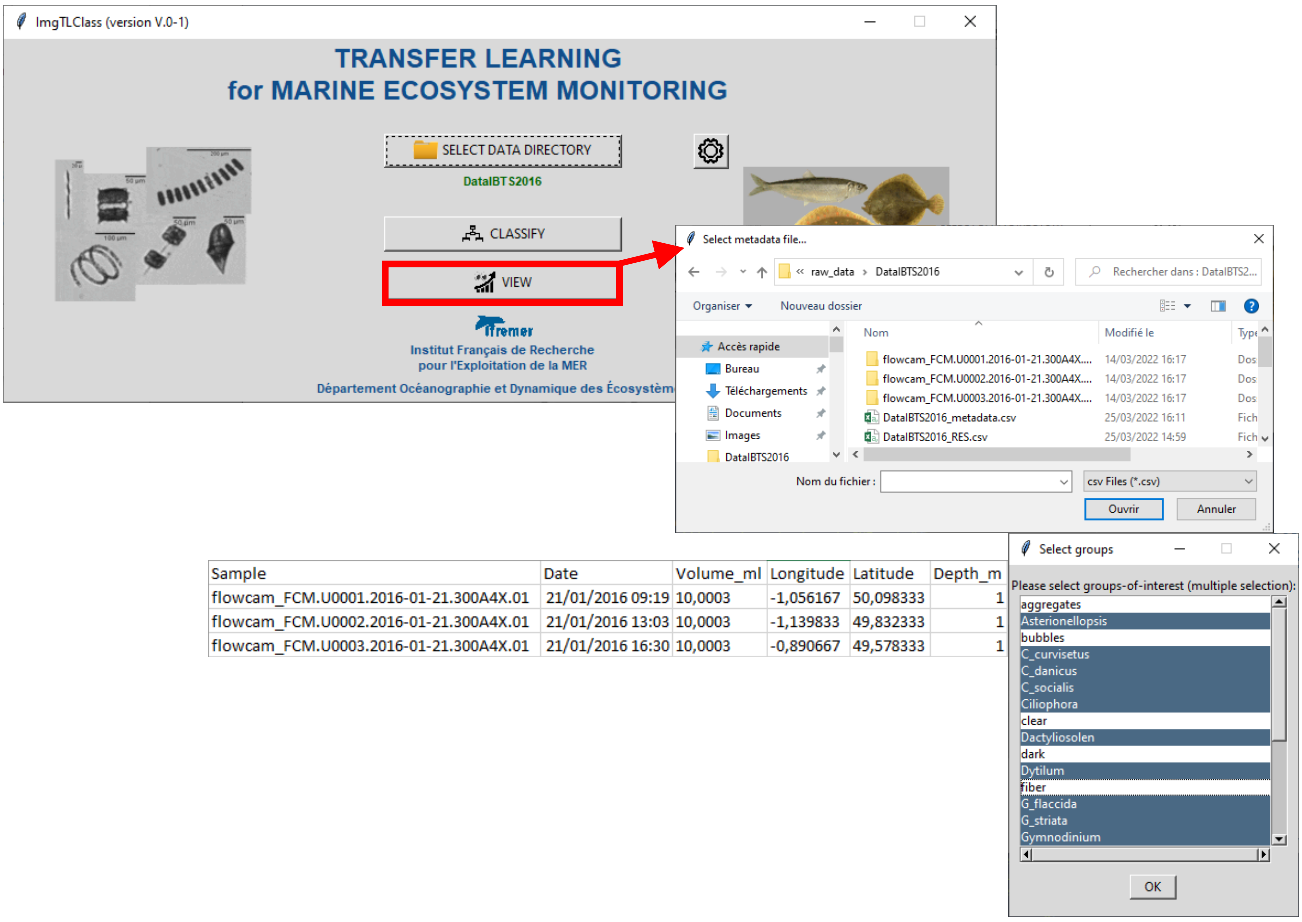
To classify new images, click on the **CLASSIFY** button. A new window appears for group selection.



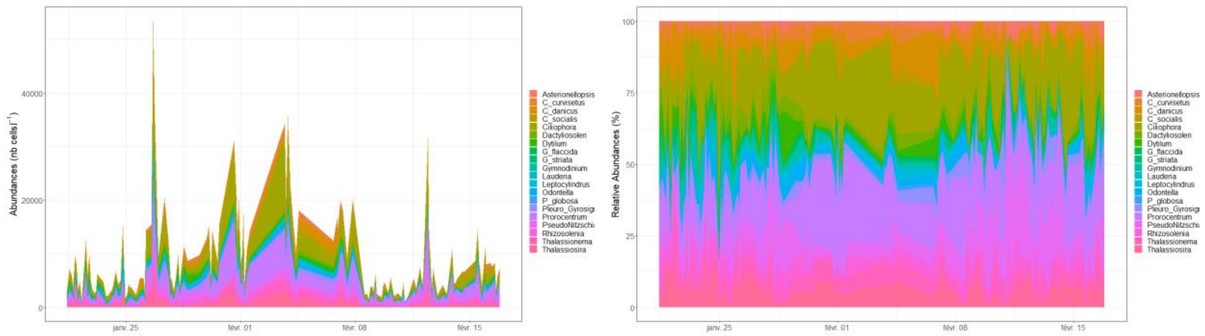
Sample	Group	Relative	Count	Prob_threshold	Percent_used
flowcam_FCM.U0001.2016-01-21.300A4X.01	Asterionellopsis	2.941176471	1	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	C_curvisetus	5.882352941	2	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	C_danicus	8.823529412	3	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	C_socialis	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Ciliophora	5.882352941	2	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Dactyliosolen	8.823529412	3	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Dytilum	8.823529412	3	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	G_flaccida	8.823529412	3	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	G_striata	5.882352941	2	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Gymnodinium	2.941176471	1	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Lauderia	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Leptocylindrus	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Odontella	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	P_globosa	0	0	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Pleuro_Gyrosigma	5.882352941	2	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Prorocentrum	14.70588235	5	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	PseudoNitzschia	2.941176471	1	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Rhizosolenia	11.76470588	4	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Thalassionema	2.941176471	1	0.75	89.88476312
flowcam_FCM.U0001.2016-01-21.300A4X.01	Thalassiosira	2.941176471	1	0.75	89.88476312
flowcam_FCM.U0002.2016-01-21.300A4X.01	Asterionellopsis	1.388888889	1	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	C_curvisetus	1.388888889	1	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	C_danicus	12.5	9	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	C_socialis	0	0	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Ciliophora	6.944444444	5	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Dactyliosolen	5.555555556	4	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Dytilum	6.944444444	5	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	G_flaccida	4.166666667	3	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	G_striata	5.555555556	4	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Gymnodinium	5.555555556	4	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Lauderia	0	0	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Leptocylindrus	0	0	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Odontella	4.166666667	3	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	P_globosa	0	0	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Pleuro_Gyrosigma	4.166666667	3	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Prorocentrum	15.27777778	11	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	PseudoNitzschia	8.333333333	6	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Rhizosolenia	6.944444444	5	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Thalassionema	5.555555556	4	0.75	90.75812274
flowcam_FCM.U0002.2016-01-21.300A4X.01	Thalassiosira	5.555555556	4	0.75	90.75812274

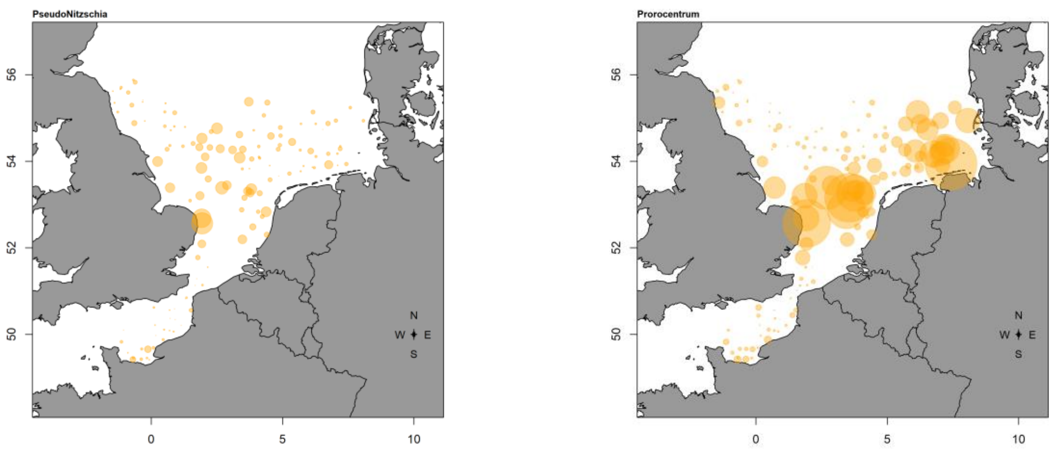
VIEW button

By clicking on this button, a new window appears for group selection.



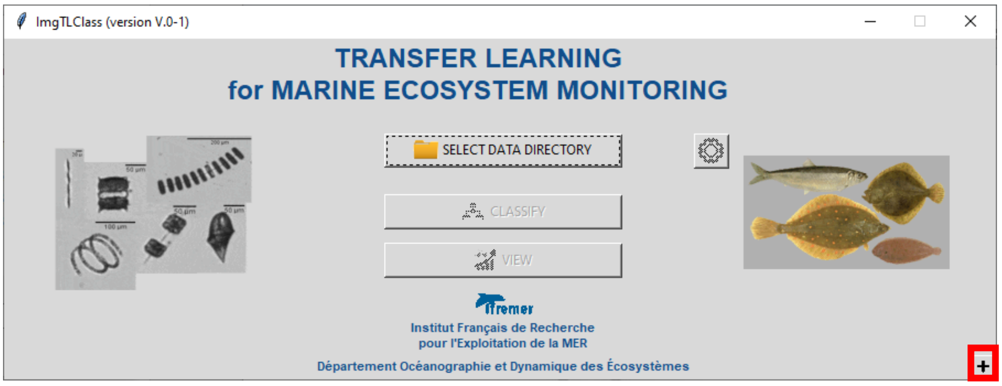
Sample	Date	Volume_ml	Longitude	Latitude	Depth_m
flowcam_FCM.U0001.2016-01-21.300A4X.01	21/01/2016 09:19	10,0003	-1,056167	50,098333	1
flowcam_FCM.U0002.2016-01-21.300A4X.01	21/01/2016 13:03	10,0003	-1,139833	49,832333	1
flowcam_FCM.U0003.2016-01-21.300A4X.01	21/01/2016 16:30	10,0003	-0,890667	49,578333	1



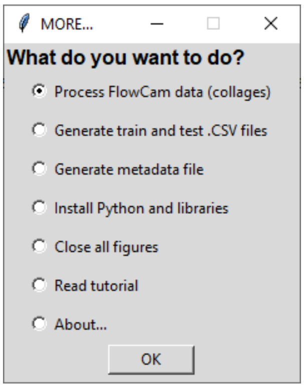


MORE… (+) button

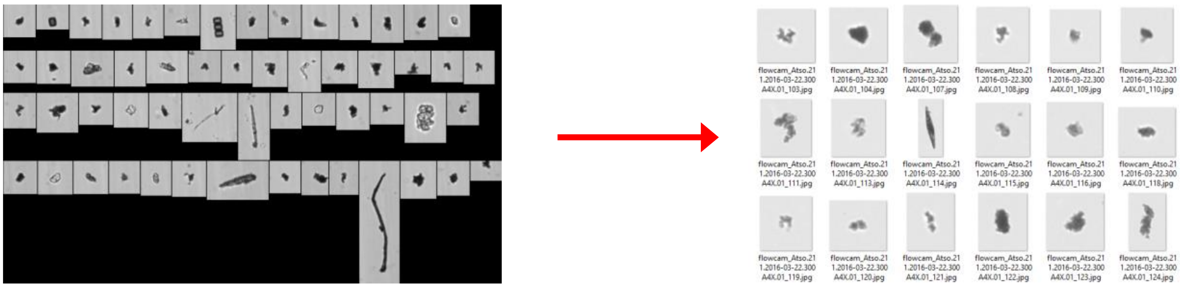
To help the user in the different data formatting steps for image analysis, several options are available. To view the list of these additional tools, click on the **+** button (**MORE…**, located at the bottom right of the main window).



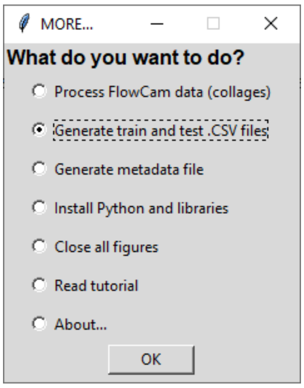
❖ **Process FlowCam data (collages)**



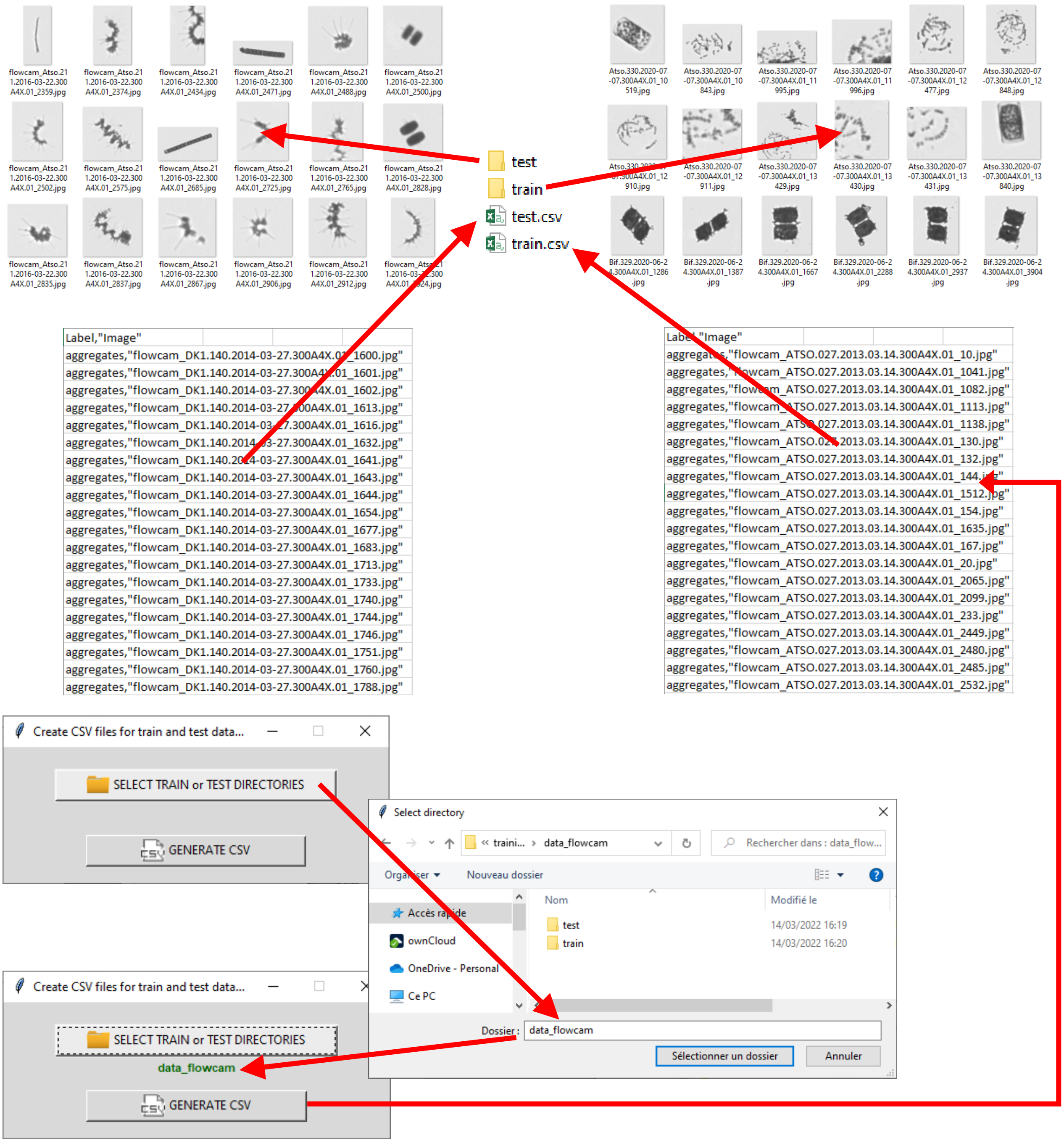
Raw data from the FlowCam is often presented as collages (one file with multiple images). It is then possible to cut and save vignettes (one image file per particle) from these collages. To do this, choose **Process FlowCam data (collages)**.



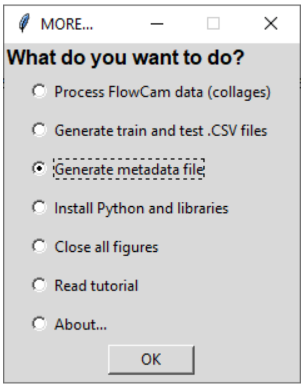
❖ Generate train and test .CSV files



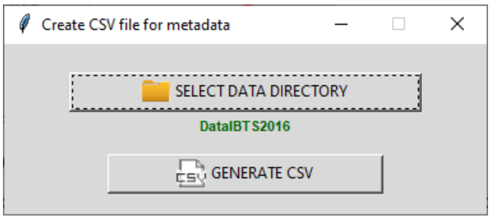
The images for training and test sets can be sorted into different subfolders (corresponding to groups). However, *Ecotranslearn* works with unique directories for **train** and **test**, in which images are put in bulk. To find out which group each image belongs to, a CSV file must be created. This is possible by selecting this option.



❖ Generate metadata file



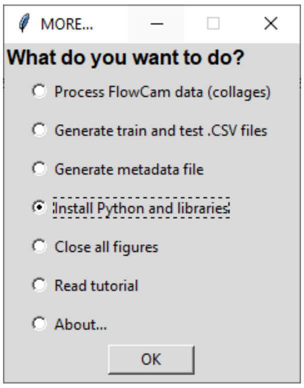
For the visualisation of classification results, a metadata file is required. This CSV file contains information on each sample, such as Date, Volume, GPS coordinates, and other metadata. A template can be generated thanks to this option.



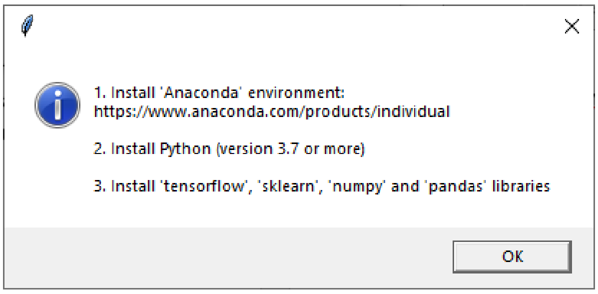
Sample	Date	Volume_ml	Longitude	Latitude	Depth_m
R1.2021-03-04.300A4X.01	2021-03-04 16:58:14	2,9994	NA	NA	1
R1.2021-03-17.300A4X.01	2021-03-17 14:55:44	5,0002	NA	NA	1
R1.2021-03-25.300A4X.02	2021-03-25 14:31:37	2,7657	NA	NA	1

with Sample: the sample name
Date: the date of sampling (format: YYYY-mm-dd HH:MM:SS)
Volume_ml: imaged volume (ml)
Longitude: sample longitude (ex: -1,056167)
Latitude: sample latitude (ex: 50,098333)
Depth: sampling depth (m)

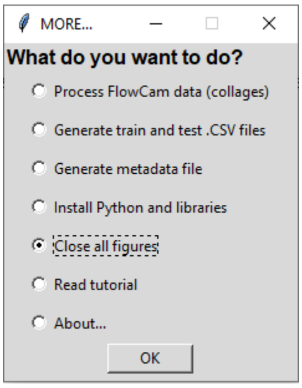
❖ Python and libraries requirements



This option allows having information on Python installation and the required libraries.



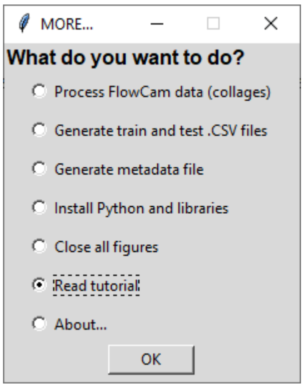
❖ Close all figures

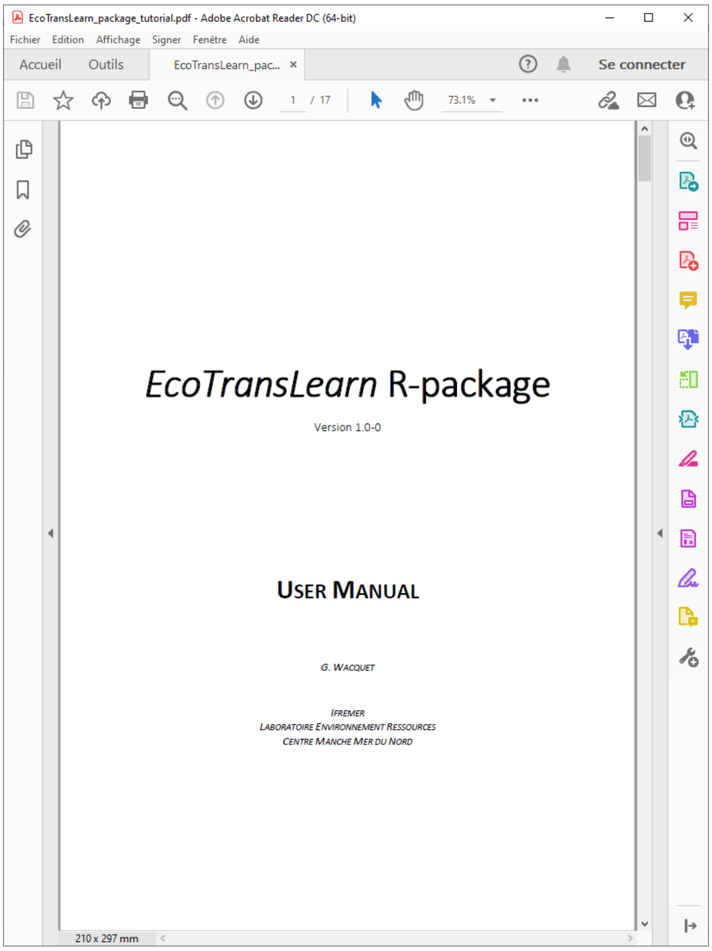


This option allows closing all figures.

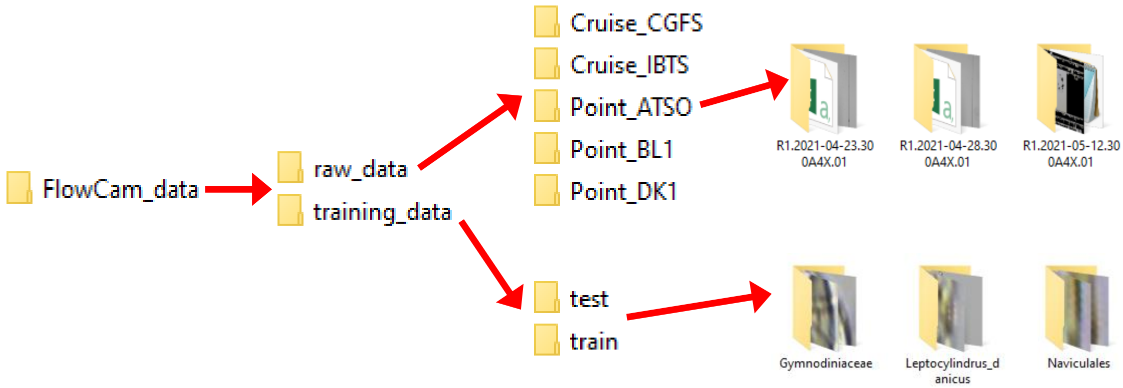
❖ Read tutorial

A user manual is available (PDF file).





Recommended folder tree



For each acquisition device, it is recommended to create a dedicated folder, containing two sub-folders: one for the raw data to be processed (‘***raw_data***’), and one for the training data (‘***training_data***’).
‘***raw_data***’ should contain all samples with images to classify, or samples with collages files (for the FlowCam device).
‘***training_data***’ should contain a 'train' folder and a 'test' folder, where the images will be sorted into different subdirectories (which will be bulked up using the **Generate train and test .CSV files** option).