**ImgTLClass R-package**

Version 1.0-0

**Manuel Utilisateur**

*G. Wacquet*

*Ifremer*

*Laboratoire Environnement Ressources*

*Centre Manche Mer du Nord*

TABLE DES MATIERES

[INTRODUCTION 5](#_Toc100070989)

[INSTALLATION ET EXECUTION 5](#_Toc100070990)

[UTILISATION DE L'INTERFACE GRAPHIQUE 6](#_Toc100070991)

[Bouton DATA SELECTION 6](#_Toc100070992)

[Bouton SETTINGS 7](#_Toc100070993)

[Bouton CLASSIFY 12](#_Toc100070994)

[Bouton VIEW 13](#_Toc100070995)

[Bouton MORE… (+) 14](#_Toc100070996)

# INTRODUCTION

TO DO !

# INSTALLATION ET EXECUTION

La version 1.0-0 de ImgTLClass nécessite une version récente de R (version 4.0.x ou plus). Elle peut être téléchargée directement sur le site du CRAN (<http://cran.r-project.org>).

En double-cliquant sur l'icône R sur le bureau, ou en sélectionnant R dans le menu de démarrage, une fenêtre apparaît à l'écran : il s’agit de la console R. Cette dernière permet de contrôler R directement par lignes de commande. Elle permet également d’afficher les principaux résultats et messages des actions effectuées avec ImgTLClass.

Les packages nécessaires à ImgTLClass :

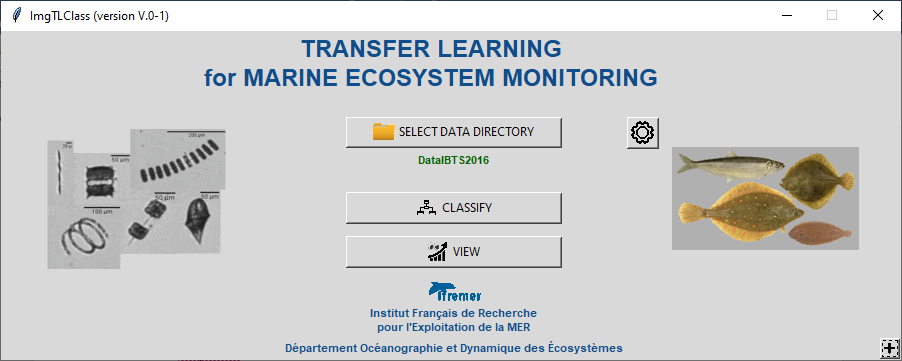
* colorRamps
* ggplot2
* grid
* jpeg
* mapplots
* maps
* randomForest
* reticulate
* SDMTools
* shapefiles
* stringr
* svDialogs
* svMisc
* tcltk2
* tiff
* zooimage

peuvent être installés directement à partir de la console R, en tapant :

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

Choisir ensuite un miroir (par défaut : 0-cloud) pour démarrer les téléchargements et les installations.

Une fois l'installation des packages terminée, il est possible de s'assurer du bon déroulement des étapes précédentes en vérifiant que la version installée est bien 1.0-0. Pour cela, dans un premier temps, taper dans la console R : **require(ImgTLClass)**, pour charger le package, puis : **ImgTLClass()**, pour lancer l’Interface Graphique Utilisateur (GUI) :



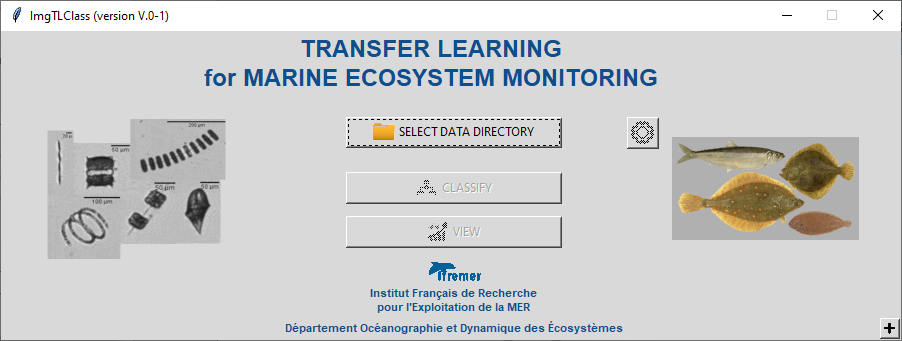
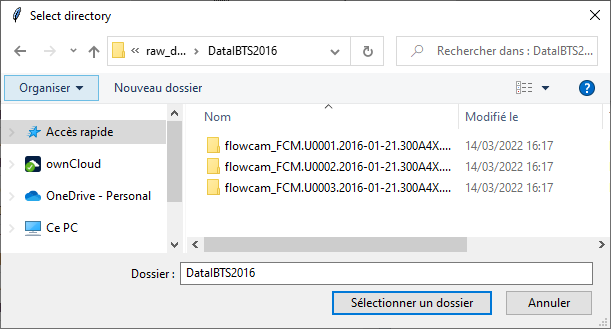
Cliquer sur le bouton **+** (en bas, à droite). Une nouvelle fenêtre apparaît :

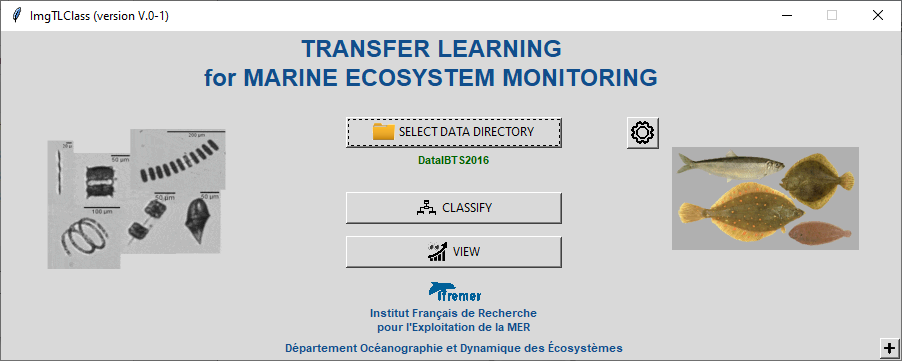
|  |  |
| --- | --- |
|  | En sélectionnant "About...", une boîte de dialogue apparaît et informe l'utilisateur de la version de ImgTLClass en cours d'exécution. |

# UTILISATION DE L'INTERFACE GRAPHIQUE

Pour l'utilisation de ImgTLClass en routine, une interface graphique utilisateur ergonomique est disponible.

## Bouton DATA SELECTION

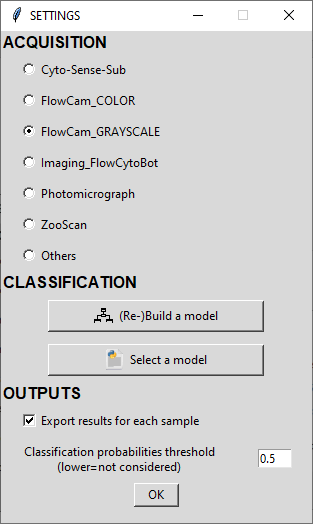
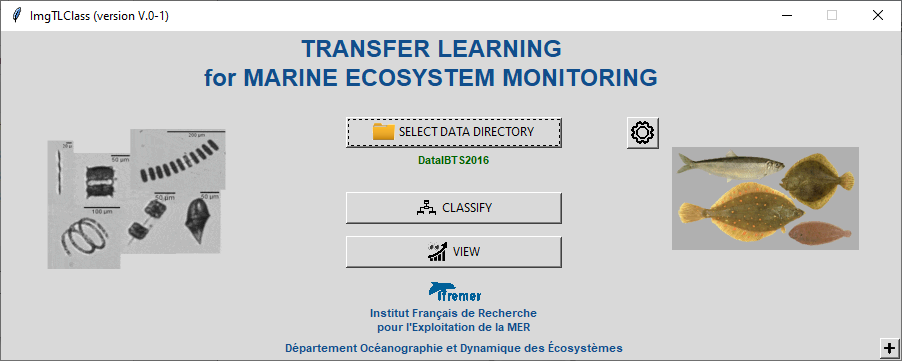




Cliquer sur le bouton **SELECT DATA DIRECTORY**, sélectionner le répertoire contenant les images à classer automatiquement, puis valider en cliquant sur **OK** : le nom du répertoire sélectionné est alors affiché en dessous du bouton de sélection, et les boutons **SETTINGS**, **CLASSIFY** et **VIEW** deviennent alors actifs.

## Bouton SETTINGS

Après sélection des données d’entrée, la fenêtre de paramétrage est automatiquement affichée (sans cliquer sur le bouton **SETTINGS**). Il est toutefois possible de réafficher cette fenêtre à tout moment en cliquant sur ce bouton.

****

* **Paramètres d’ACQUISITION**

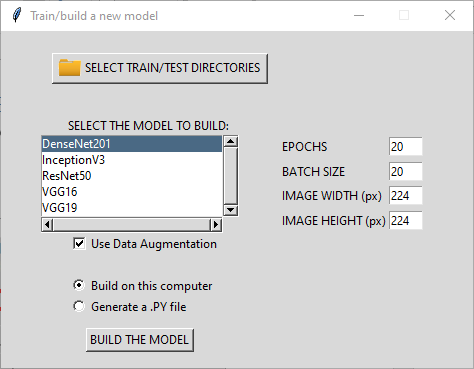
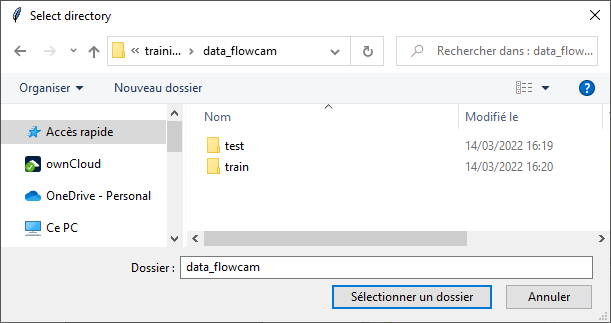
Choisir le type d’instrument utilisé pour l’acquisition des images.

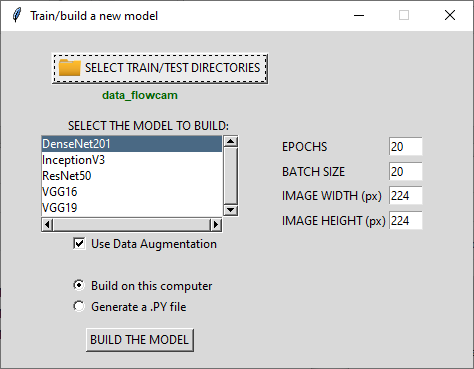
*Remarque : pour le FlowCam\_COLOR et le FlowCam\_GRAYSCALE, il est possible de mettre directement les données brutes en sortie des instruments (collages). Le package permet de créer directement les vignettes (1 image par particule) à partir des collages et du fichier lst.*

* **Paramètres de CLASSIFICATION**
  + **(Re-)Build a model**

|  |  |
| --- | --- |
| Ce bouton permet de construire (ou reconstruire) un modèle de classification par Transfer Learning grâce à l’utilisation d’un jeu de données comprenant un training set (répertoire nommé **train**) pour l’apprentissage, et un test set (répertoire nommé **test**) pour la validation et l’évaluation. |  |

En cliquant sur ce bouton, une nouvelle fenêtre apparaît.





Cliquer sur le bouton **SELECT TRAIN/TEST DIRECTORIES**, sélectionner le répertoire contenant les deux sous-dossiers **train** et **test**, puis valider en cliquant sur **OK** : le nom du répertoire sélectionné est alors affiché en dessous du bouton de sélection.

Il est ensuite possible de sélectionner différentes architectures de Réseaux de Neurones à Convolution (CNN pour Convolutional Neural Network) dans la liste **SELECT THE MODEL TO BUILD**, parmi :

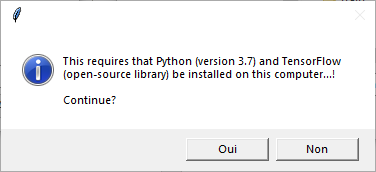
* + - DenseNet201
    - InceptionV3
    - ResNet50
    - VGG16
    - VGG19

et régler les paramètres liés aux images et à la phase d’apprentissage, en fixant des valeurs à **EPOCH** (par défaut=20), **BATCH** **SIZE** (par défaut=20), **IMAGE** **WIDTH** (par défaut=224) et **IMAGE** **WEIGHT** (par défaut=224), mais également choisir la possibilité d’utiliser la technique d’augmentation de données (**Use Data Augmentation**). Dans le cas d’un training set peu fourni en images, cette option permet de générer automatiquement des images complémentaires à partir des images du training set, en leur appliquant des transformations géométriques comme des rotations (par défaut, rotation\_range=45) et des retournements horizontaux et verticaux (par défaut, horizontal\_flip=True et vertical\_flip=True).

La dernière étape consiste à choisir le matériel sur lequel construire et adapter le modèle de classification. Selon l’option sélectionnée, et après avoir cliqué sur le bouton **BUILD THE MODEL** :

* + - **Build on this computer**

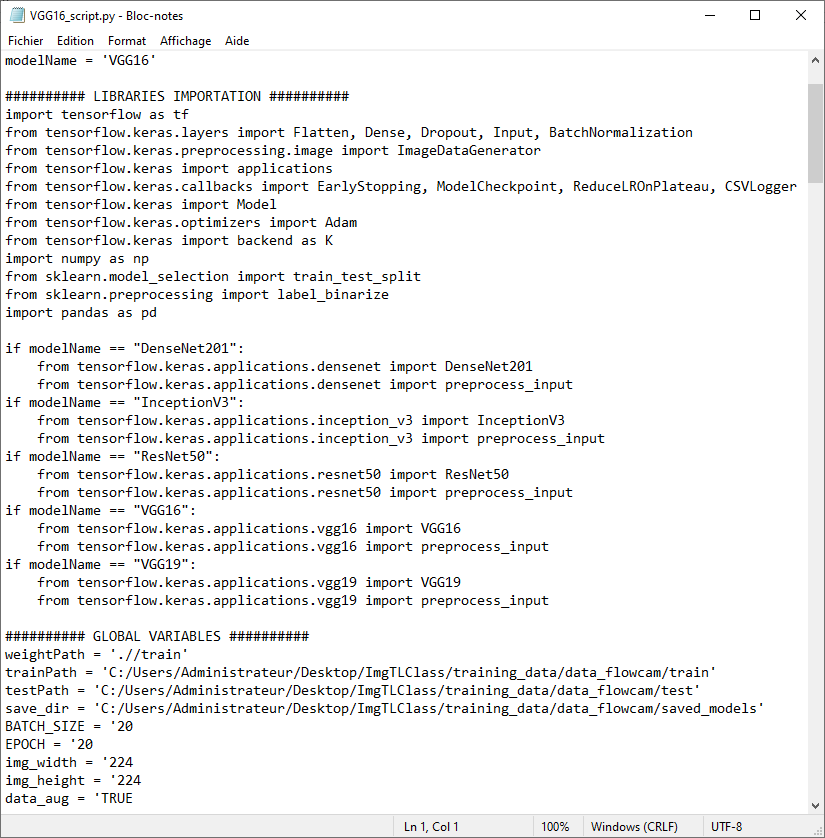
Une boîte de dialogue s’ouvre alors :



*Attention : le temps d’apprentissage peut être long (plusieurs heures) selon le nombre d’images du training set et selon les paramètres définis lors de l’étape précédente.*

* + - **Generate a .PY file**

Un script est créé automatiquement, et peut être exécuté sur un autre matériel (ordinateur dédié, serveur de calcul, …).

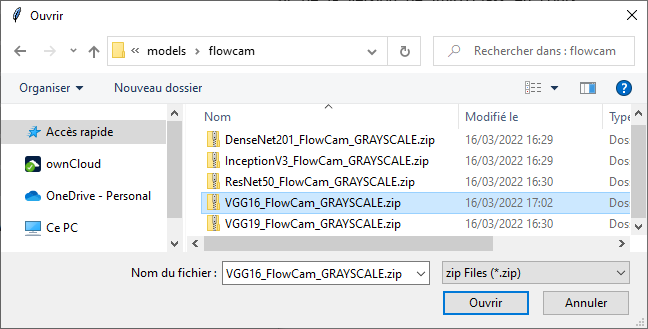


A la fin de l’exécution du script, un fichier .ZIP est créé. Ce dernier contient toutes les informations nécessaires pour la classification d’un nouveau jeu d’images.

|  |  |
| --- | --- |
|  | **modelName\_classNames.csv** contient les noms des classes du training set.  **modelName\_history.csv** contient les scores d’évaluation du modèle créé.  **modelName\_model.h5** est le modèle créé sous Python. |

* + **Select a model**

|  |  |
| --- | --- |
| Ce bouton permet de choisir un modèle pour la classification automatisée des images contenues dans le répertoire sélectionné lors de la première étape (cf. section « Sélection des données d’entrée »). |  |



Sélectionner le fichier ZIP généré lors de l’étape précédente, puis valider en cliquant sur **Ouvrir** : des informations sur les performances du modèle (calculées lors des phases d’apprentissage et de validation) sont alors affichées en dessous du bouton de sélection, ainsi que les courbes d’**Accuracy** (≈ pourcentage de données correctement classifiées) et de **Loss** (≈ distance entre les données réelles et les données prédites).

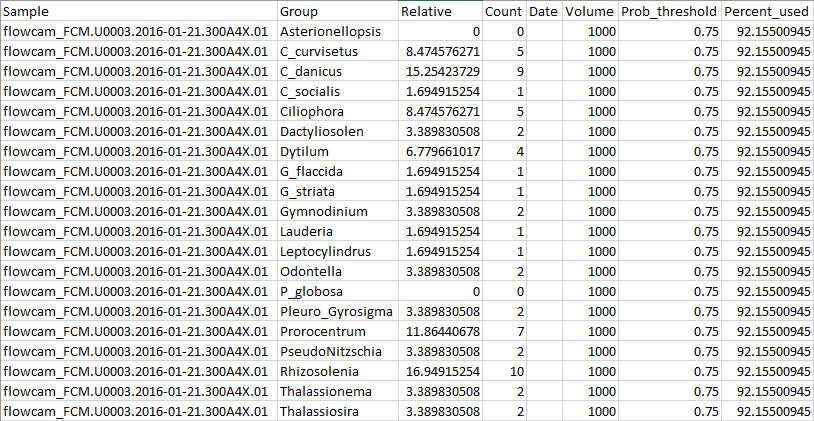
|  |  |
| --- | --- |
|  |  |

* **Paramètres d’OUTPUTS**
  + **Export results for each sample**

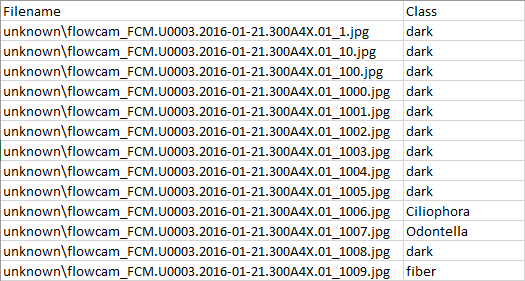
Cette option permet d’exporter les résultats pour chaque sous-dossiers du répertoire sélectionné lors de la première étape (cf. section « Sélection des données d’entrée »).

Pour chaque échantillon traité, trois fichiers CSV sont alors créés :

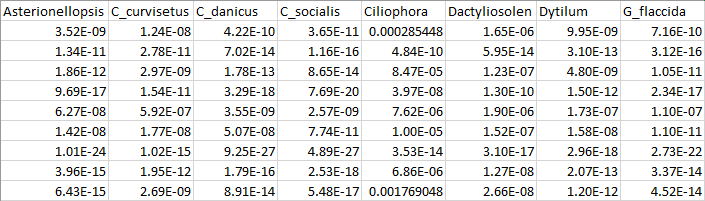
* + - sampleName.csv



* + - sampleName\_CLASSIF.csv



* + - sampleName\_PRED.csv

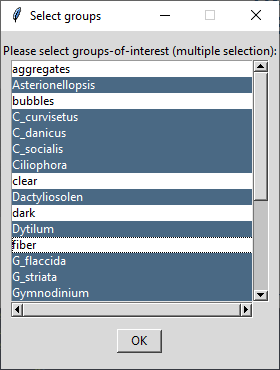
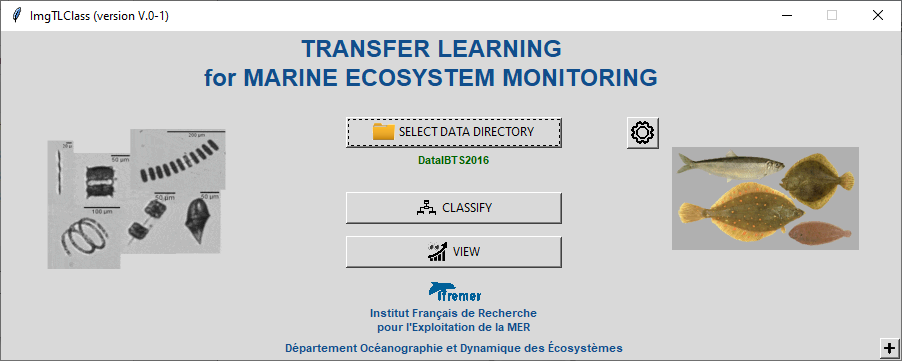


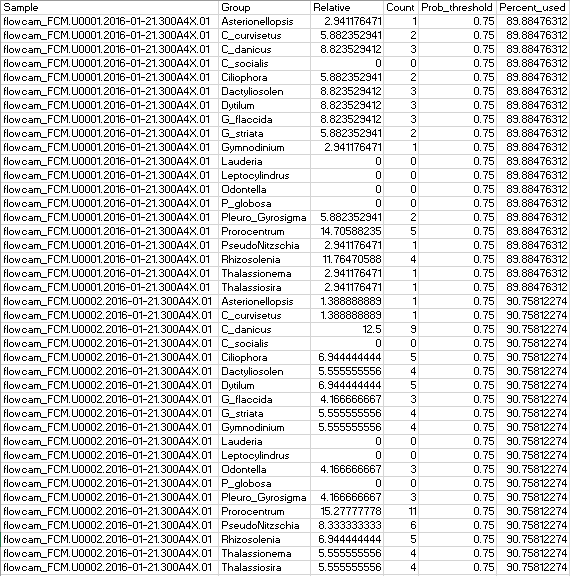
* + **Classification probability threshold**

La valeur de seuil définie à cette étape permet de ne prendre en compte que les particules ayant une probabilité de « bonne » classification supérieure à ce seuil. Pour prendre en considération la totalité des particules, cette valeur doit être fixée à 0.

## Bouton CLASSIFY

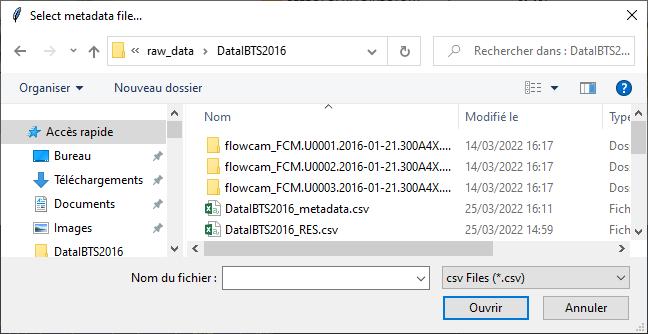
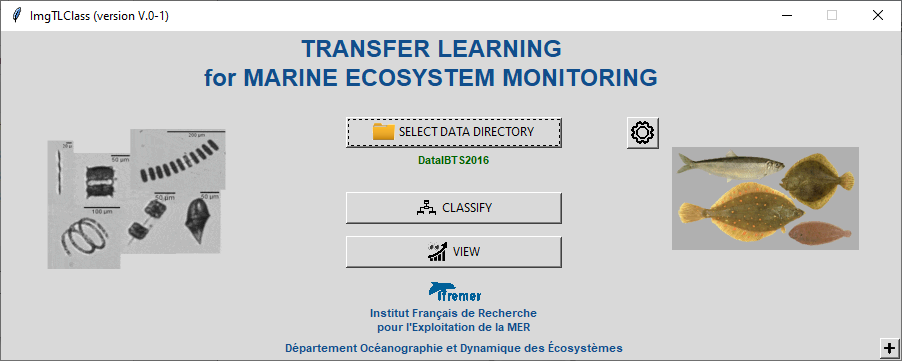
Pour classer de nouvelles images, cliquer sur le bouton **CLASSIFY**. Une nouvelle fenêtre de sélection de groupes apparaît.

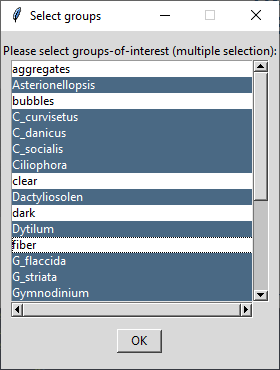
****

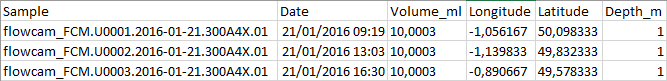


## Bouton VIEW

En cliquant sur ce bouton, une nouvelle fenêtre de sélection de groupes apparaît.

****

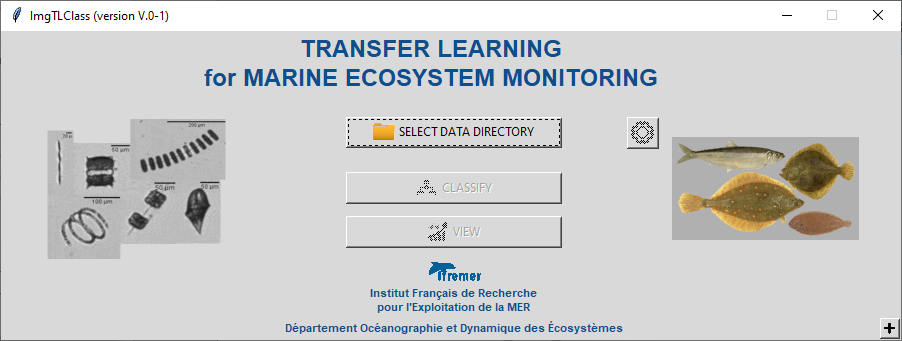
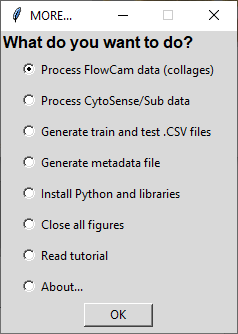




|  |  |
| --- | --- |
| C:\Users\Administrateur\Desktop\FlowCam\Proto_ComputerVision_DeepLearning\DataIBTS2016\DataIBTS2016_ABD.png | C:\Users\Administrateur\Desktop\FlowCam\Proto_ComputerVision_DeepLearning\DataIBTS2016\DataIBTS2016_REL.png |
| C:\Users\Administrateur\Desktop\FlowCam\Proto_ComputerVision_DeepLearning\DataIBTS2016\DataIBTS2016_PseudoNitzschia.png | C:\Users\Administrateur\Desktop\FlowCam\Proto_ComputerVision_DeepLearning\DataIBTS2016\DataIBTS2016_Prorocentrum.png |

## Bouton MORE… (+)

Pour aider l’utilisateur dans les différentes étapes de mise en forme des données pour l’analyse des images, plusieurs options sont disponibles. Pour visualiser la liste de ces outils additionnels, cliquer sur le bouton **+** (**MORE…**, en bas à droite de la fenêtre principale).



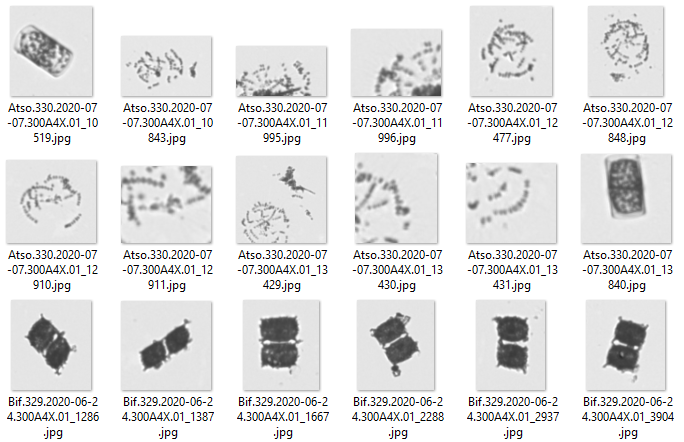
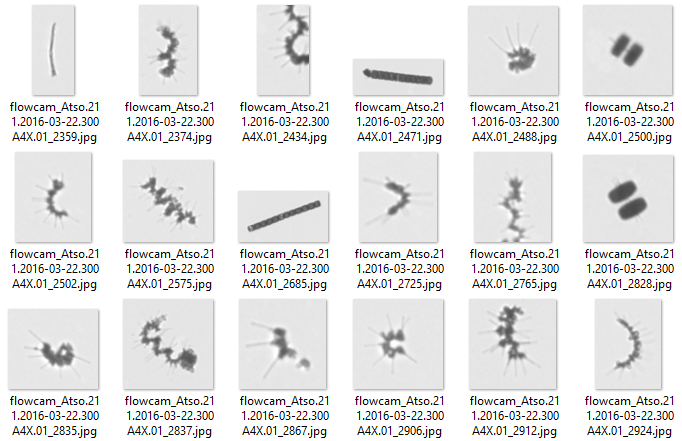
* **Process FlowCam data (collages)**

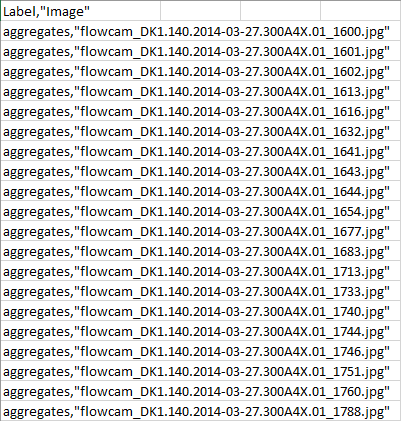
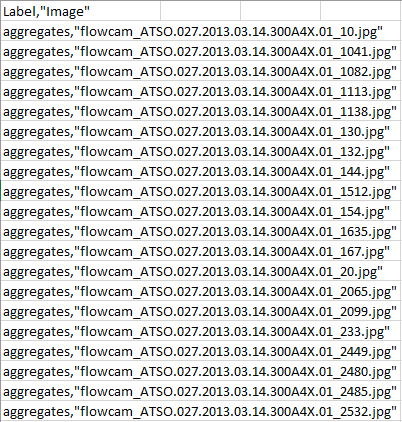
Les données brutes issues du FlowCam sont souvent présentées sous forme de collages (un fichier avec plusieurs images de particules). Il est alors possible de découper et sauvegarder les vignettes (un fichier image par particule) à partir de ces collages. Pour cela, choisir **Process FlowCam data (collages)**.

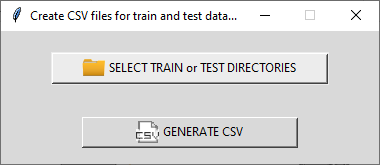
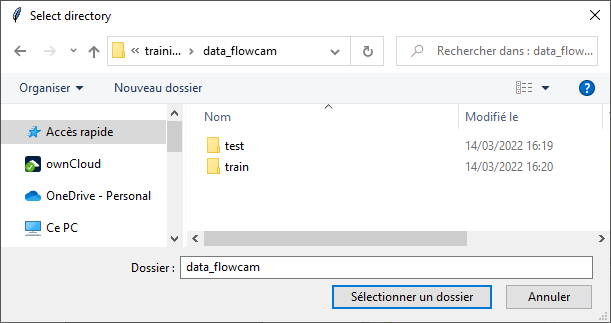
* **Process CytoSense/Sub data**

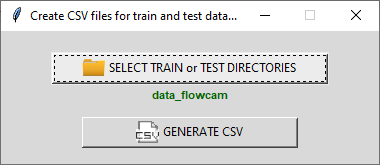
TO DO !

* **Generate train and test .CSV files**



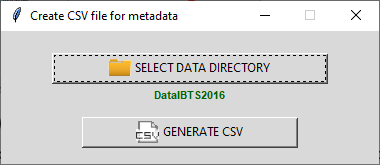


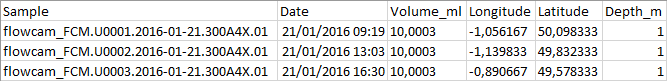




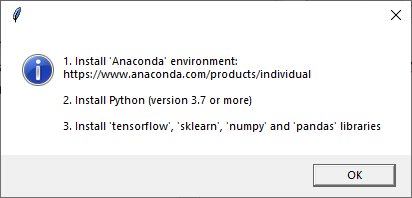
* **Generate metadata file**

Template pour le fichier CSV.





* **Python and libraries requirements**



* **Close all figures**

TO DO !

* **Read tutorial**

PDF file.

