

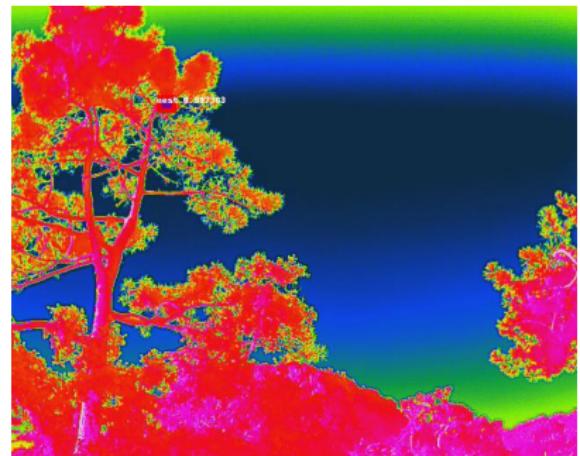
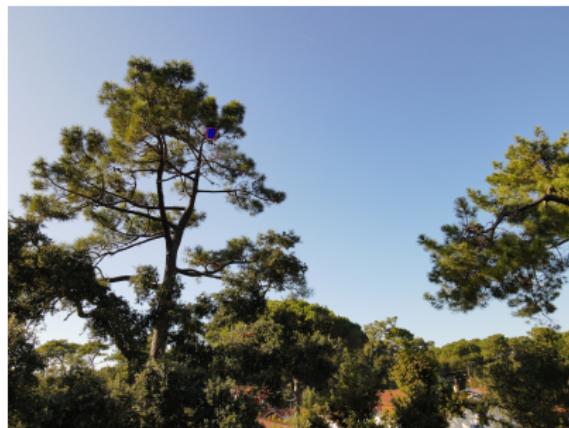
Présentation soutenance du Stage de Master 2 : Bees for Life

Alexis Hoffmann

Tuteur : Pascal Desbarats, Enseignant référent : Serge Chaumette

5 Juillet 2022

Introduction



Sommaire

► Contexte

1. Association Bees For Life
2. Reprise du projet
3. Objectifs

Sommaire

- ▶ Contexte
 - 1. Association Bees For Life
 - 2. Reprise du projet
 - 3. Objectifs
- ▶ Développement de l'application
 - 1. Environnement de développement
 - 2. Qt
 - 3. Architecture
 - 4. Réalisation
 - 5. Tests

Sommaire

- ▶ Contexte
 - 1. Association Bees For Life
 - 2. Reprise du projet
 - 3. Objectifs
- ▶ Développement de l'application
 - 1. Environnement de développement
 - 2. Qt
 - 3. Architecture
 - 4. Réalisation
 - 5. Tests
- ▶ Ouverture

└ Contexte

└ Association Bees For Life

1. Association Bees for life



LES 3 PROJETS DE BEESFORLIFE

1

APPLICATION signalement et gestion destruction des nids de frelons asiatiques intégrable dans le site @ des collectivités

EN COURS AJOUT FONCTIONNALITES NOVALDI

TRANSPARENCE INTERVENTION DESINSECTISEUR POUR LE PARTICULIER

2

La mise au point de techniques de REPERAGE des nids de frelons asiatiques: Drone+ Caméras visible et thermique+ post traitement en Intelligence Artificielle(I.A) avec LaBRI (Uni.Bordeaux)

2022: APPLI V02 DE BEESLIFE APPLICATION INTELLIGENCE ARTIFICIELLE "DEEP LEARNING"

3

Solution de destruction et élimination des nids de frelons asiatiques sans produit chimique dans le cadre de la réglementation et des bonnes pratiques (Brevet INPI déposé en 2020)

AU 28 AVRIL : MISSION DE 2 ETUDIANTS INGENIEURS 4 MOIS : REALISER LE GRAPPIN

2. Reprise du projet : l'existant

- ▶ Modèle de détection Mask R-CNN (Kaiming He *et al.*[1]).
- ▶ Fonctionnalités de visualisation : simultanées, séparées, coordonnées, position, ...
- ▶ Bugs de fuite de mémoire, de mise à jour, de stabilité.
- ▶ Installation et lancement rudes pour une personne non-habituée du terminal.
- ▶ Chargement des images dans des dossiers spécifiques

3. Objectifs

- ▶ Installer l'environnement de développement anaconda sur le Mac du client.
- ▶ Faire tourner la détection sur son ordinateur.
- ▶ Mettre en place une architecture modulaire et claire pour l'intégration de nouveaux modèles.
- ▶ Optimisation dans le chargement des images et dans le défilement.
- ▶ Améliorer l'expérience utilisateur.

1. Environnement de développement

- ▶ Projet sous python 3.6 avec tensorflow 1.8

1. Environnement de développement

- ▶ Projet sous python 3.6 avec tensorflow 1.8
- ▶ Anaconda : un outil pour gérer différentes versions de Python dans des environnements séparés, mélange entre venv et pip

1. Environnement de développement

- ▶ Projet sous python 3.6 avec tensorflow 1.8
- ▶ Anaconda : un outil pour gérer différentes versions de Python dans des environnements séparés, mélange entre venv et pip
- ▶ Création à partir d'un fichier yaml

```
name: bfl-pyside2
channels:
  - defaults
  - conda-forge
dependencies:
  - conda-forge::python=3.6.15
  - pip
  - conda-forge::pyside2=5.13.2
  - conda-forge::eixfread3
  - conda-forge::qrcode=7.3.1
  - conda-forge::scikit-image=0.15.0
  - conda-forge::tensorflow=1.8.0
  - conda-forge::keras=2.1.6
  - conda-forge::folium
  - conda-forge::pytest-qt=4
  - conda-forge::pytest-cov=3
  - conda-forge::pytest-xvfb=2
  - conda-forge::pytest-dependency
  - conda-forge::black
  - pip:
      - pyperclip-pyside==1.8.2
```

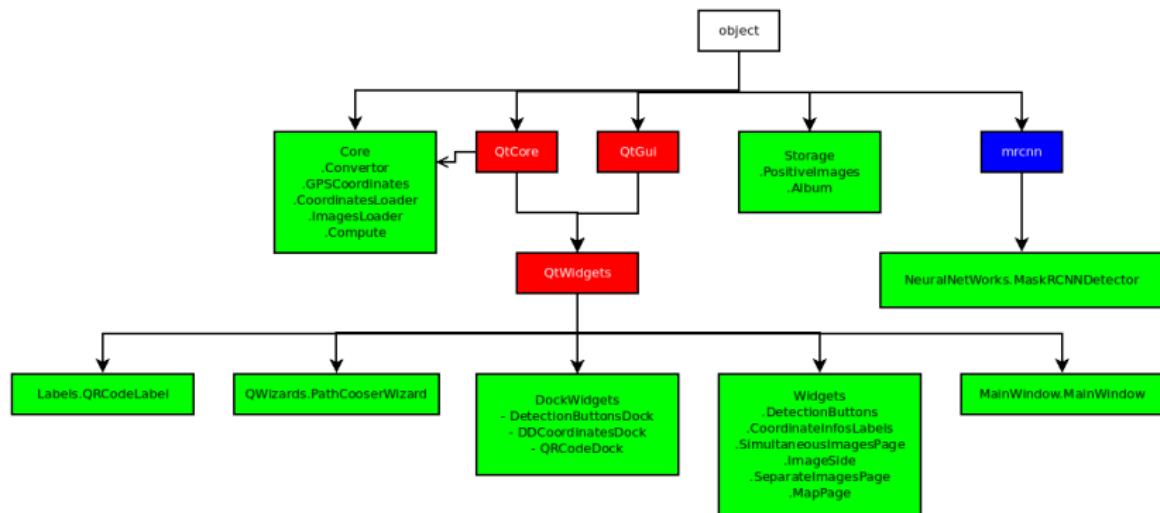
Figure – Fichier *environment.yml* du projet

2.

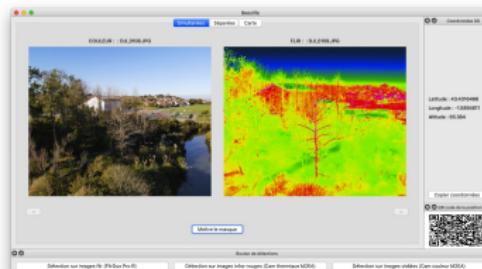
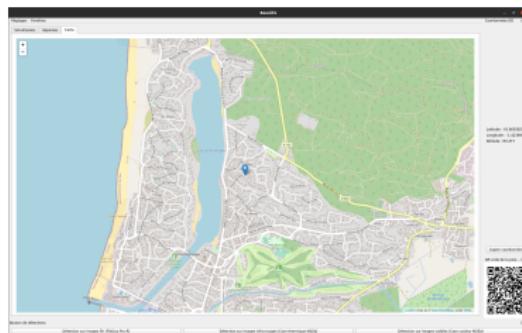
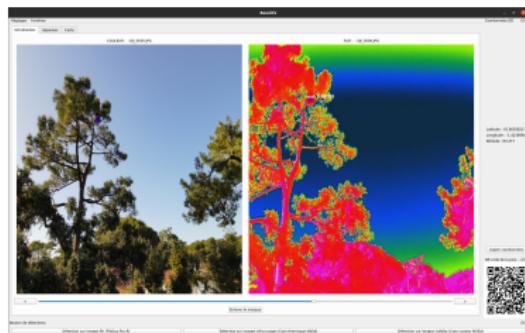


- ▶ Framework d'interface utilisateur multiplateforme
- ▶ Gestion et manipulation de fichiers, de simulation 3D, de données de capteurs
- ▶ Architecture singulière en module
 - ▶ QtCore : fonctionnalités non graphiques comme les slots (actions associées aux clics des boutons), ou encore la gestion des fichiers
 - ▶ QtGui : manipulation d'images et autres
 - ▶ QtWidgets : éléments graphiques (boutons, menus, ...)
 - ▶ QtWebEngine : éléments pour gérer du contenu web

3. Architecture



4. Réalisation



5. Tests avec pytest

- ▶ pytest-xvfb : permet de faire tourner les tests sans fenêtres avec Xvfb, un serveur X qui tourne sans affichage du matériel
- ▶ autres plugins : pytest-qt, pytest-cov et pytest-dependency
- ▶ 67% de couverture du code en tests

```
tests/test_core.py ..... [ 77%]
tests/test_mainwindow.py . [ 88%]
tests/test_neuralnetworks.py s [ 83%]
tests/test_storage.py s [ 86%]
tests/test_widgets.py s... [100%]

----- coverage: platform linux, python 3.6.15-final-0 -----
Name           Stmts   Miss  Cover
-----
src/Bf1Qt/Core.py      97      0  100%
src/Bf1Qt/DockWidget.py    28      0  100%
src/Bf1Qt/Label.py       29      1  97%
src/Bf1Qt/MainWindow.py    89     32  64%
src/Bf1Qt/NeuralNetworks.py  56     35  38%
src/Bf1Qt/Storage.py      84     36  57%
src/Bf1Qt/Widget.py      327     96  71%
src/Bf1Qt/Wizard.py      116     75  35%
src/Bf1Qt/__init__.py        2      0  100%

TOTAL              828    275  67%
```



```
===== 33 passed, 3 skipped in 2.86s =====
```

Ouverture

- ▶ Élimination des faux positifs pour labéliser les images
- ▶ Entraînement de nouveaux modèles (ex : ViT ou DETR)
- ▶ Intégration dans l'application
- ▶ Rédaction du rapport dans son entièreté

 Kaiming He, Georgia Gkioxari, Piotr Dollar, Ross Girshick,
Mask R-CNN, Facebook AI Research (FAIR), 2017