

Laboratoire 1 : Introduction à Python pour le traitement d'images

Les travaux pratiques qui vous seront soumis au cours de la session se réaliseront avec le langage de programmation Python. C'est pourquoi nous vous proposons ici une séance d'initiation à ce langage. Ce laboratoire comporte principalement :

- Une installation guidée pour créer un environnement de programmation
- Une introduction des modules python principaux utilisés pour le traitement d'images
- Une série d'exercices suggérés

1. Présentation

- Faites connaissance avec votre démonstrateur (Mathieu Gravel)
- Connectez-vous aux machines de l'UQAM avec vos accès UQAM.
- **Remarque:** c'est une bonne idée d'apporter votre propre machine aux laboratoires pour une plus grande flexibilité d'installation de logiciels.

2. Python & Jupyter

Les TP de ce cours seront réalisés avec le langage Python et en utilisant des notebooks Jupyter. Pour nous assurer que tous ont accès au même environnement de programmation, nous utiliserons `Anaconda`, une distribution de Python.

2.1 Installations

- Installer Anaconda sur votre machine. Choisissez Python 3.
 - <https://www.anaconda.com/distribution/>
- Créer un environnement virtuel en utilisant la définition d'environnement fournie. Pour Linux et Mac, ouvrez un terminal et utilisez la commande suivante:

```
conda env create -f environment.yml
```

Cette commande créera un environnement de programmation nommé `inf600f-h2020`. Pour activer l'environnement, ouvrez un terminal et utilisez la commande

```
conda activate inf600f-h2020
```

- Pour démarrer un serveur jupyter local:

```
jupyter notebook
```

Remarque: Seuls les notebooks contenus dans le dossier où le serveur jupyter a été démarré sont accessibles. Assurez-vous donc d'être dans le dossier du laboratoire.

Remarque pour Linux/Mac: Pour utiliser `conda` avec Linux ou Mac, il faut accepter que l'installation ajoute les liens vers anaconda dans votre profile `.bash_profile` de votre terminal.

Remarque pour Windows: Pour Windows, il faut utiliser l'interface graphique `Anaconda Navigator` et créer un environnement à partir de cette interface. Assurez-vous d'installer toutes les dépendances mentionnées dans le fichier `environment.yml` pour avoir accès aux mêmes modules. Pour Windows, le serveur jupyter peut être démarré à partir de l'interface graphique `Anaconda Navigator`

2.2 Introductions à Python et Jupyter

- <https://realpython.com/jupyter-notebook-introduction/>
- <https://ipython-books.github.io/11-introducing-ipython-and-the-jupyter-notebook/>

2.3 Bibliothèques utilisées dans ce cours

Cette section porte sur la manipulation de bases des images en utilisant les modules `numpy`, `scipy`, `scikit-image`, `matplotlib`, `imageio`. Voici une brève description de chaque module :

- `numpy` : Extension du langage Python destiné à la manipulation de matrices, et en incluant plusieurs fonctions mathématiques opérant sur ces matrices et diverses méthodes d'algèbre linéaire.
- `scipy` : Module Python spécialisé pour l'usage scientifique.
- `scikit-image` : Bibliothèque libre de traitement d'images en Python
- `matplotlib` : Bibliothèque utilisée pour l'affichage des images
- `imageio` : Bibliothèque utilisée pour la lecture et l'écriture d'images.

2.4 Exercices suggérés

- Ouvrir et écrire un fichier image avec `skimage`
- Afficher une image avec `matplotlib`
- Ouvrir et écrire un fichier image avec `imageio`
- Manipuler des images
- Imprimer des informations de bases sur les images (min, max, moyenne, taille, format)
- Calculer des statistiques de bases
- Modifier les blocs du notebook Jupyter
- Éditer une équation dans un bloc markdown
- Écrire une fonction python
- Utiliser une fonction python
- Importer un module local
- Exporter un notebook Jupyter en format PDF
- Afficher l'histogramme des intensités dans une image
- Indexation pour numpy array

Tutoriels en ligne recommandés

- <https://ipython-books.github.io/>
- Scipy-lectures : <https://scipy-lectures.org/packages/scikit-image/>
- https://scipy-lectures.org/advanced/image_processing/index.html#basic-image