*Mahaut GALICE – Hugo ROCHE*

*Polytech Dijon*

*Compte-rendu tp images*

# INTRODUCTION

Le but de ces TP est de réaliser du traitement d’images sous Octave ou Python à l’aide de fichiers fournis. RAJOUTER INFOS

Table des matières

[INTRODUCTION 1](#_Toc167785800)

[PRISE EN MAIN RAPIDE 3](#_Toc167785801)

[TRANSFORMÉE DE FOURIER 8](#_Toc167785802)

[a. Harmoniques pures 8](#_Toc167785803)

# PRISE EN MAIN RAPIDE

Cette première partie du TP nous permet de prendre en main les logiciels utilisés. Nous avons fait le choix de travailler en Python.

Le premier programme, demo.py, nous permet de lire une image avant d’en afficher les composantes. On obtient également l’image donnée en noirs et blancs. On réalise un test à partir de l’image « CerisierP.jpg ».

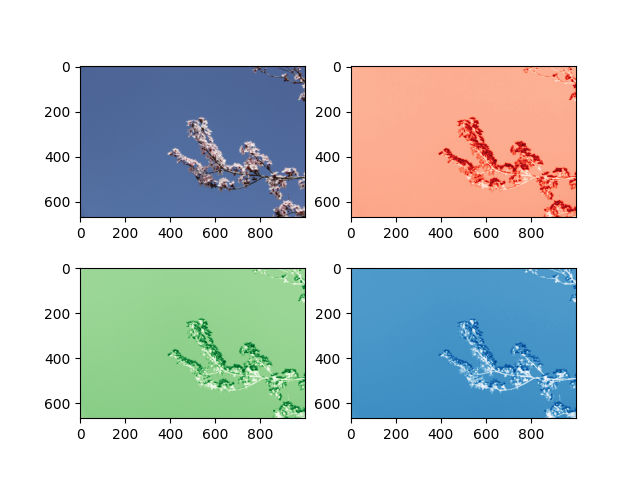


Figure : composantes RGB de l'image cerisier

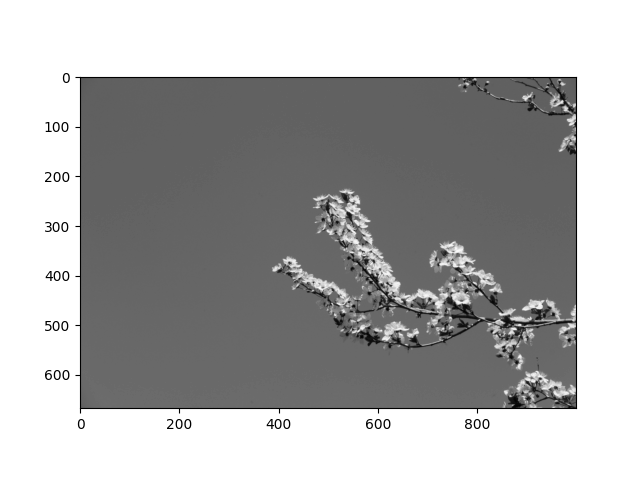


Figure : Image cerisier transformée en noir et blanc

On teste ensuite la fonction histogramme, incluse dans le module pyplot. Pour ce faire on va utiliser la ligne de commande :

**plt.hist(imgG)**

Si l’on souhaite l’afficher on va passer par la suite de lignes de code :

**plt.figure()** *#créer une nouvelle figure***plt.hist(imgG)** *#retourne l’histogramme de notre image***plt.show()** *#affiche la figure de l’histogramme*

Après avoir testé la fonction sur l’image du cerisier on obtient l’histogramme suivant :

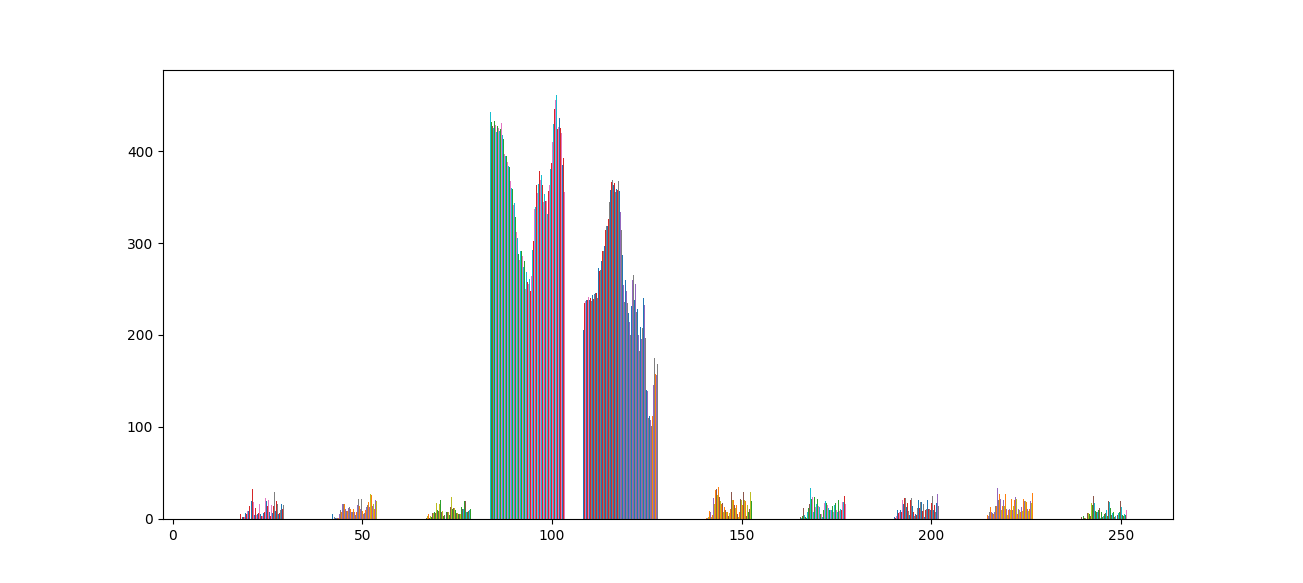


Figure : histogramme de l'image du cerisier en niveau de gris

Il s’agit ensuite de binariser l’image à partir d’un seuil définit grâce à l’histogramme. Après analyse rapide, on peut placer le seuil à environ 110.   
Le programme suivant permet de binariser l’image donnée puis de l’afficher avec un seuil définit et entré en paramètre.

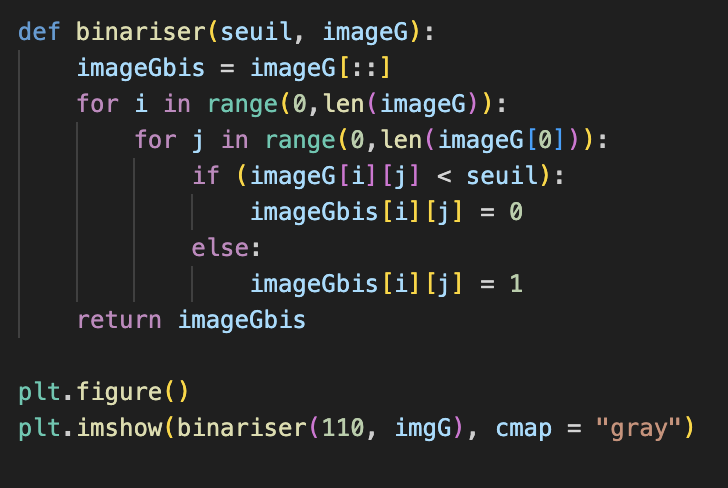


Figure : Fonction binariser en Python

Elle retourne l’image suivante :

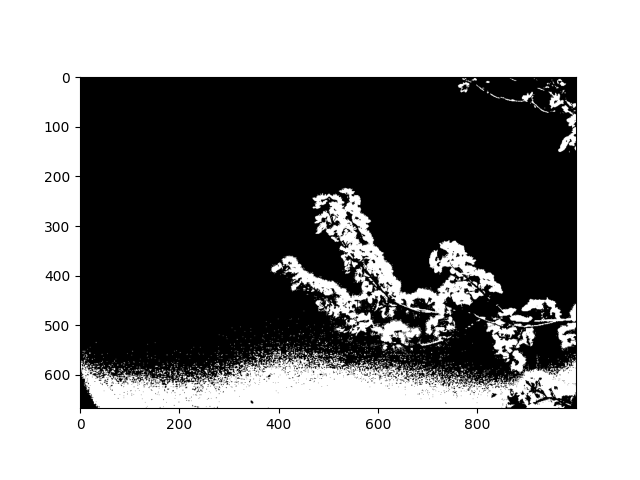


Figure : image de cerisier binariser

Pour calculer le seuil à l’aide de la méthode des moments on utilise la fonction suivante :

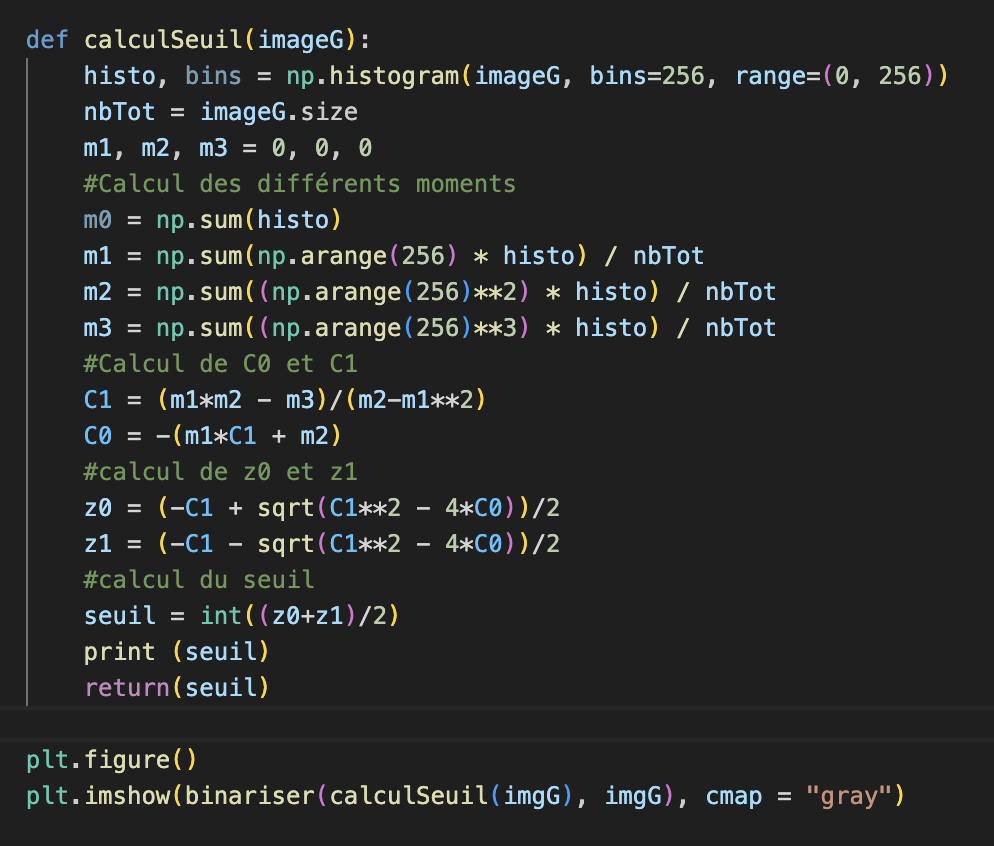


Figure : Script permettant de calculer le seuil

On rappelle ensuite la fonction binariser précédente mais avec le nouveau seuil calculé. L’image retournée cette fois ci est la suivante :

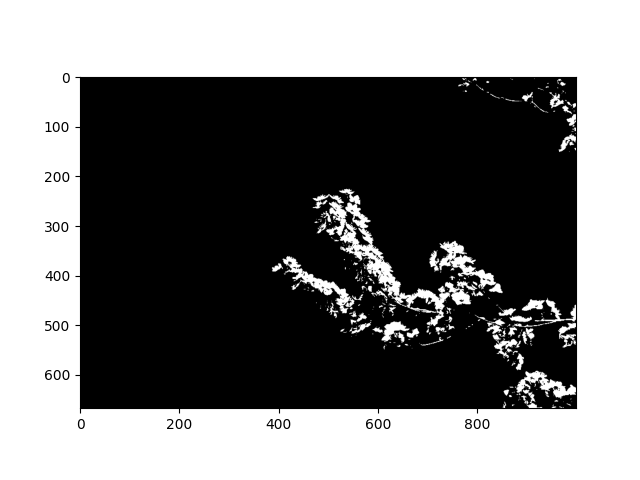


Figure : Image binarisée avec seuil calculé

FINIR PRISE EN MAIN

# TRANSFORMÉE DE FOURIER

Le but de cette partie du TP est d’appliquer la transformée de Fourier (TFD) à différentes images.

Dans un premier temps, d’après la formule donnée, on comprend que pour appliquer la TFD 2D il faut appliquer la TFD 1D à chaque ligne de notre image, puis à chaque colonne de la matrice qui en résulte.

## Harmoniques pures

Pour créer l’image demandée et la visualiser on réalise le script suivant :

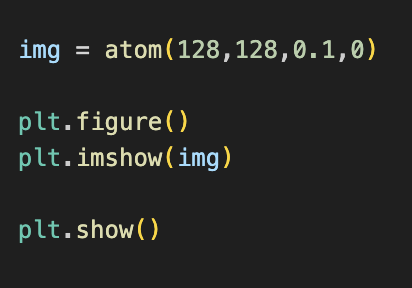


Figure : création et affichage d'une image de 128\*128 de fréquence 0.1 sur N et 0 sur M

L’image retournée est la suivante :

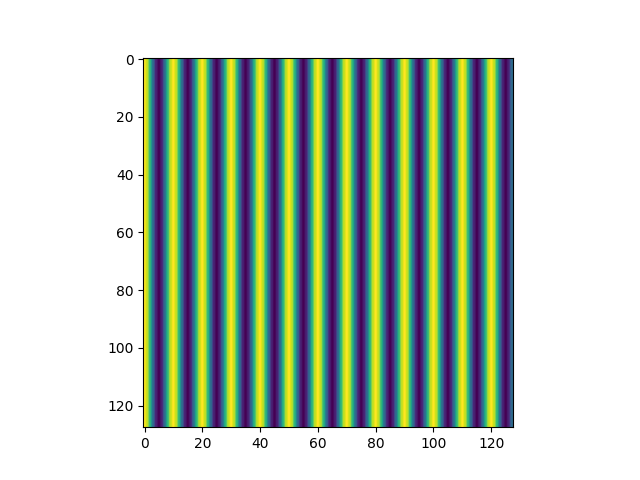


Figure : Affichage de l'image crée avec le code ci-dessus

On voit donc s’afficher une successions de bandes verticales claires et foncées, avec une périodicité spatiale de 10, ce qui est cohérent avec le résultat attendu puisque la fréquence rentrée est de 0.1.