

DES NUAGES 3D ET DES IMAGES VERS UNE MAQUETTE NUMÉRIQUE

ANNEXE 1 – SCRIPT D’OPTIMISATION MICMAC



Diplômant : Della Casa Bruno

Professeur responsable : Barras Vincent

Expert : Brahier Julien

Mandant : Bureau d'études ROSSIER SA, Maurer Nicolas

Date : 22 mai 2018



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch



Environnement construit
& Géoinformation

TABLE DES MATIERES

1. Introduction.....	1
2. Préparation pour l'exécution du script	1
3. Marche à suivre des scripts	1
4. Remarques	3

TABLE DES ILLUSTRATION

Figure 1 Choix des images pour le calcul	2
Figure 2 Choix des images pour la calibration.....	2
Figure 3 Exemple de fichier contenant les points fixes.....	3

1. INTRODUCTION

Cette annexe contient une explication du traitement de photogrammétrie avec l'aide du logiciel MicMac. MicMac est un logiciel qui fonctionne en ligne de commande. Il faut donc avoir une certaine connaissance du programme pour l'utiliser pleinement.

Il est difficile de calculer un traitement complet de l'aérottriangulation de beaucoup de photo. Il est alors nécessaire d'avoir plusieurs étapes de calculs du positionnement de photo. La première étape est le calcul de la calibration de l'appareil. Ensuite, un nombre de photo est calculées ensemble pour se positionner. Nous rajoutons des images au premier calcul d'aérottriangulation et ensuite au deuxième et ainsi de suite. La création des commandes prend ainsi du temps et demande à l'utilisateur d'être derrière sont ordinateurs pour lancer la prochaine commande à la fin du calcul.

Un fichier batch permet d'y indiquer plusieurs calculs qui se lanceront les uns après les autres. Par contre, ces commandes doivent toujours être écrites à la main. Pour qu'elles se créent automatiques, un script python a été créé.

2. PRÉPARATION POUR L'ÉXÉCUTION DU SCRIPT

La première étape est l'installation de python sur son ordinateur. Pour la version, le script fonctionne avec python 3.

Il faut ensuite installer les divers modules qui permettent de faire fonctionner les scripts, il suffit de lancer ses lignes de code dans l'invite de commande.

- `py -3 -m pip install tkinter` //module permettant d'ouvrir un boîte de dialogue
- `py -3 -m pip install xlwt` //module permettant d'écrire un fichier excel
- `py -3 -m pip install lxml` //module permettant de lire des fichiers xml

Après avoir installé ces divers éléments, les scripts fonctionnent. Il y a par contre une contrainte à l'utilisation de ce script, il faut que les photos soient prises dans l'ordre de position. Cela veut dire qu'il ne faut pas par exemple faire de photos d'une pièce à l'intérieur, puis prendre des photos à l'extérieur et finir par faire le lien entre les 2 blocs de photos. Il faut faire le lien entre les 2 blocs entre les 2 prises de photos.

3. MARCHE À SUIVRE DES SCRIPTS

L'optimisation est séparée en 2 scripts. L'un permet de créer les diverses lignes de commande tandis que l'autre permet d'analyser les résultats de l'aérottriangulation.

La première étape est de copier les scripts dans le dossier contenant notre projet avec les photos.

Il faut ensuite lancer le script « 01calcul_commande.py ». Une boîte de dialogue s'ouvre, il nous suffit de sélectionner les photos pour la photogrammétrie.

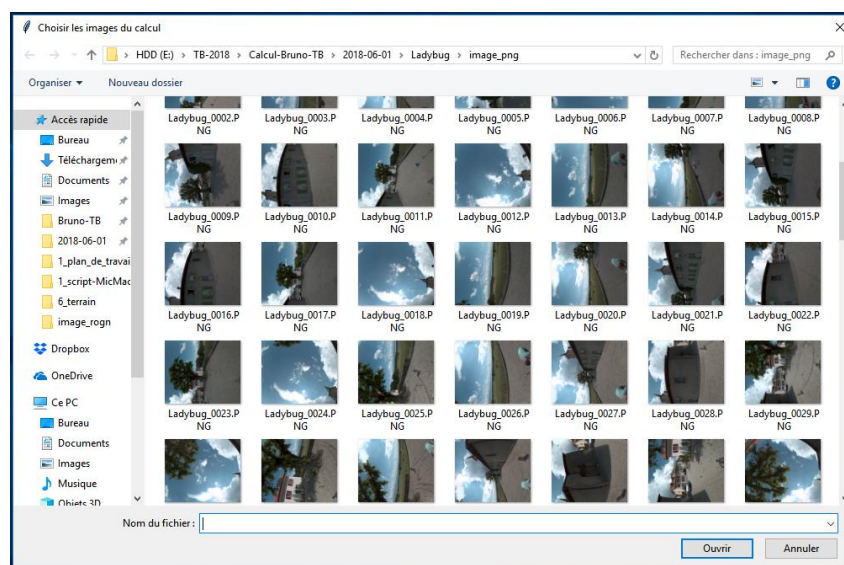


Figure 1 Choix des images pour le calcul

Après avoir choisi les photos, le script nous demande si les photos ont des coordonnées GPS. Si elles ont des coordonnées GPS, cela permet de réduire le temps de calcul des points homologues car il choisit les images qui sont proche pour en rechercher les points.

La prochaine étape est de choisir la taille d'image pour le calcul des points homologues. Normalement, une taille d'image de 1500 suffit largement. Il est possible que suivant les photos vous deviez prendre une plus grande taille. Une grande taille implique un plus grand temps de calcul pour un résultat quasiment équivalent.

Il faut ensuite choisir le modèle de calibration. Pour la plupart des appareils, le modèle RadialStd ou RadialExtended fonctionne très bien. Il faut par contre utiliser le modèle FishEyeBasic pour les photos de type FishEye. Sur ce site se trouve une description des différents modèles (<https://micmac.ensg.eu/index.php/Tapas>).

La prochaine étape est de choisir les images pour la calibration. Il est conseillé de prendre entre 5 et 15 images pour le calcul avec des objets situés sur différents plans. Ces images ne doivent pas forcément être des images utilisées pour les autres calculs. Par contre, ses images doivent se situer aux mêmes endroits que les autres.

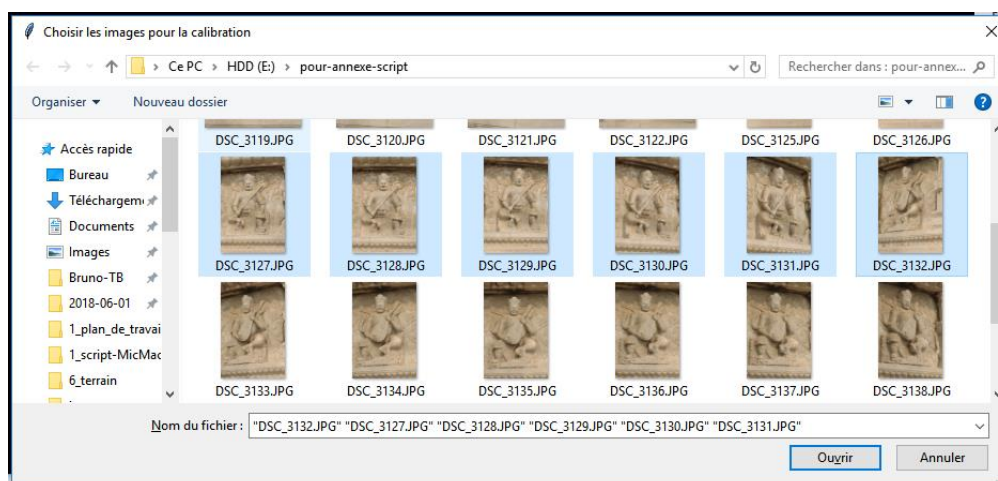


Figure 2 Choix des images pour la calibration

Il faut ensuite définir le pas entre les différents calculs d'aérottriangulation. Cela veut dire le nombre de photos qu'il rajoute petit à petit au calcul d'aéro.

Les prochaines étapes seront décrites dans le schéma joint dans cette annexe.

Le deuxième script « 02Analyse_resultat.py » se lance automatiquement après les aérottriangulations. Il sert à récupérer les valeurs de précision des calculs et de les enregistrer dans un fichier Excel. Cela permet de voir plus facilement s'il y a eu un souci lors du calcul.

4. REMARQUES

En cas de référencement du projet via des points fixes, les coordonnées doivent être petite sinon le calcul ne fonctionne pas bien. Le fichier doit être mis au même endroit que les images. Le fichier des points doit être décrit de la même manière que ci-dessous :

```
#F= N X Y Z
8001 651.100 431.500 500.250
8002 622.446 419.254 499.589
8003 612.114 433.045 500.290
8004 635.499 437.353 500.748
8005 622.070 432.446 501.820
```

Figure 3 Exemple de fichier contenant les points fixes

Ce script marche dans la plupart des jeux de photos. Il est par contre possible qu'il nécessite un petit ajustement des fichiers batch en cours de calcul (rajout ou enlèvement d'image pour certaines aérottriangulations). Si les photos ont été bien faites le déroulement se fait de manière automatique sans changement.