

UTILISATION DE GEFOLKI

Notice explicative

Objet

Notice à l'usage de l'utilisation de GeFolki pour la coregistration d'images

Auteur

Elise Koeniguer

Date de mise à jour du document :

Décembre 2016

Licence et Conditions d'exploitation

Le module **GeFolki** est distribué sous licence GPL.

GeFolki a fait l'objet de deux publications différentes. Si vous utilisez ce code dans une publication ou un travail scientifique, nous vous demandons de bien vouloir citer l'une des deux publications ci-dessous.

- Dans le cadre de la coregistration de **deux images de télédétection de même nature** (même bande du spectre électromagnétique : SAR/SAR de même bande, optique/optique dans le même canal, etc), même si les résolutions sont différentes : on considère que les images sont dites « homogènes. La publication à citer est alors :

Aurélien Plyer, Elise Colin-Koeniguer, Flora Weissgerber, "A New Coregistration Algorithm for Recent Applications on Urban SAR Images", Geoscience and Remote Sensing Letters, IEEE , vol.12, no.11, pp. 2198 – 2202, nov 2015

- Dans le cadre de la coregistration de **deux images de natures différentes** (SAR/LIDAR, SAR-bande X/SAR bande L, SAR/optique, deux bandes optiques différentes, etc.), on considère que les images sont hétérogènes. La publication à citer est alors :

Guillaume Brigot, Elise Colin-Koeniguer, Aurélien Plyer, Fabrice Janez, "Adaptation and Evaluation of an Optical Flow Method Applied to Coregistration of Forest Remote Sensing Images", IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, Volume 9, Issue 7, July 2016

Contenu de la Notice

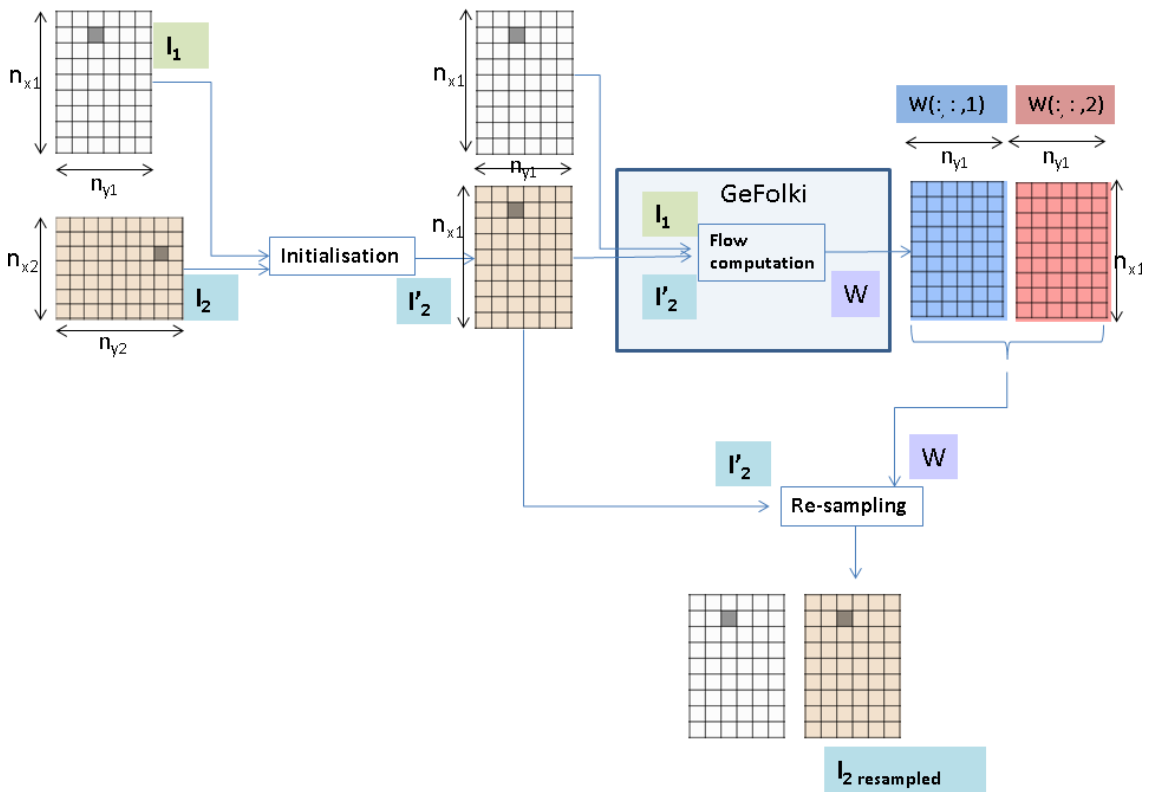
GeFolki est un module permettant de réaliser la coregistration de deux images de télédétection.

Dans cette notice, on trouvera une description :

- Des différents modules permettant d'assurer la coregistration entière de deux images : initialisation, calcul de flot (assuré par GeFolki) et rééchantillonnage.
- Des recommandations sur le choix des différents paramètres intervenant dans GeFolki en fonction du scénario envisagé.

De la manière la plus générale, la mise en correspondance pixellique de deux images peut se décomposer en trois étapes

1. Une étape d'initialisation.
2. Une étape de calcul de déplacements (GeFolki)
3. Une étape de ré-échantillonnage final



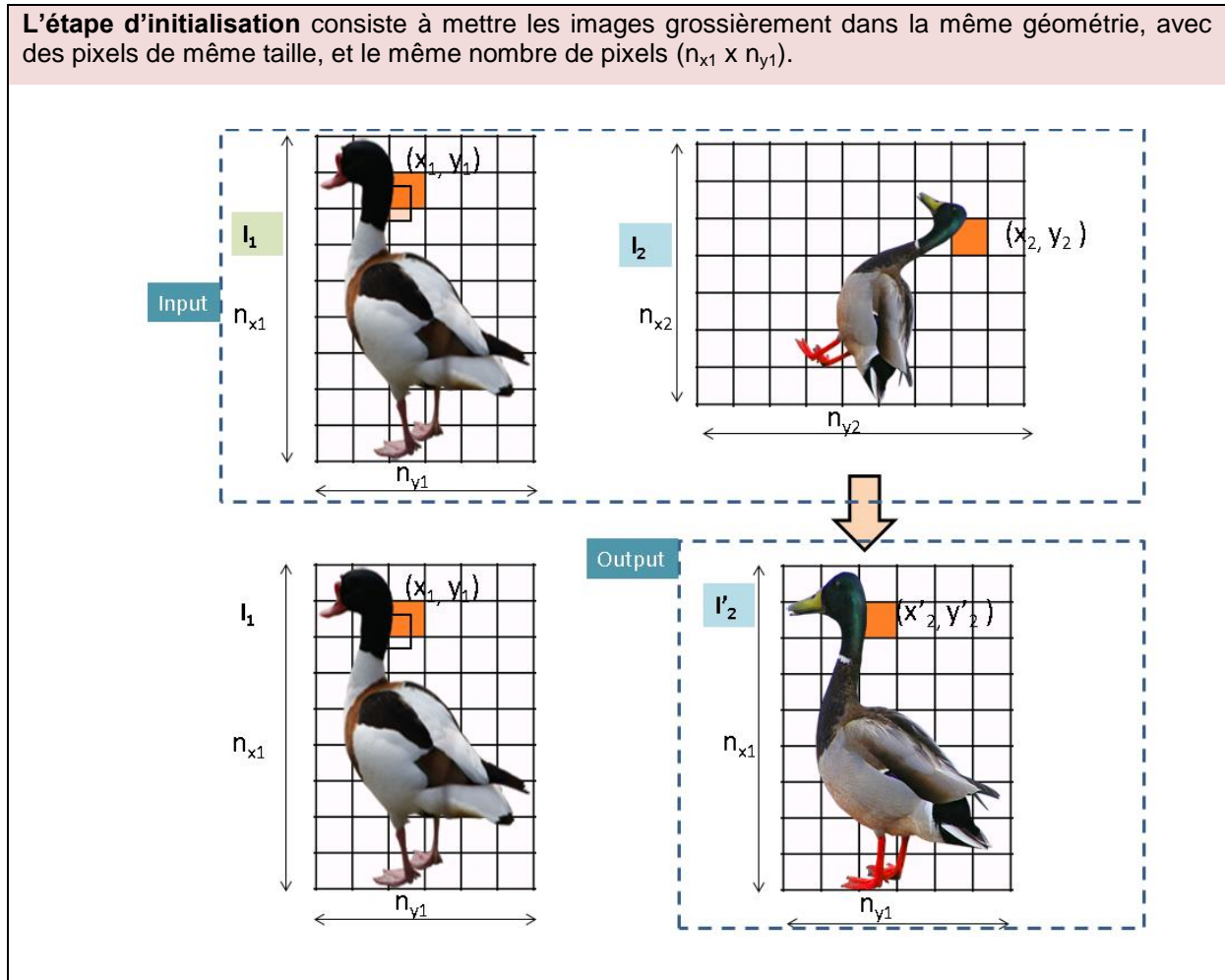
Ces trois étapes et leurs entrées-sorties sont décrites sur le diagramme ci-dessus.

1. L'étape d'initialisation prend en entrée deux images à recaler, de tailles différentes. La sortie de l'initialisation fournit la seconde image esclave transformée, en une image de même taille que l'image maître : $n_{x1} \times n_{y1}$.
2. L'étape de calcul de Flot est le cœur de GeFolki. Elle prend en entrée deux images de mêmes dimensions. Elle donne en sortie une hypermatrice de Flot W à 3 dimensions, de taille $n_{x1} \times n_{y1} \times 2$. $W(:, :, 1)$ correspond à la composante du flot calculé sur l'axe horizontal. $W(:, :, 2)$ correspond à la seconde composante du flot calculé sur l'axe vertical.
3. La troisième étape consiste à ré-échantillonner l'image esclave suivant la matrice de flot obtenue à l'étape précédente. Elle prend donc deux entrées : l'image I_2' et l'hyper matrice de flot W . Elle fournit en sortie I_2 resampled, en correspondance pixellique avec l'image I_1 .

DESCRIPTION DE LA PREMIERE ETAPE : INITIALISATION

Soit I_1 et I_2 les deux images à co-registrer de dimensions respectives (nombre de lignes x nombre de colonnes) $(n_{x1} \times n_{y1})$ et $(n_{x2} \times n_{y2})$

L'étape d'initialisation consiste à mettre les images grossièrement dans la même géométrie, avec des pixels de même taille, et le même nombre de pixels $(n_{x1} \times n_{y1})$.



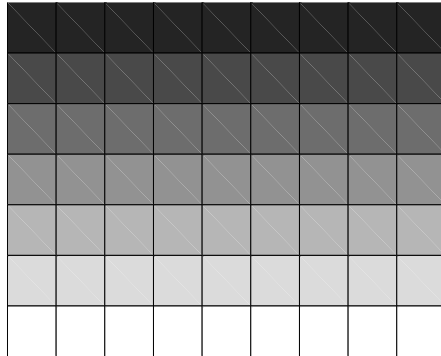
L'image I_1 est considérée comme l'image maîtresse, l'image I_2 est considérée comme l'image esclave.

La commande matlab permettant de décrire les matrices X et Y où X donne le numéro de ligne et Y le numéro de colonne est :

```
[Y,X]=meshgrid((1:ny),(1:nx)) ;
```

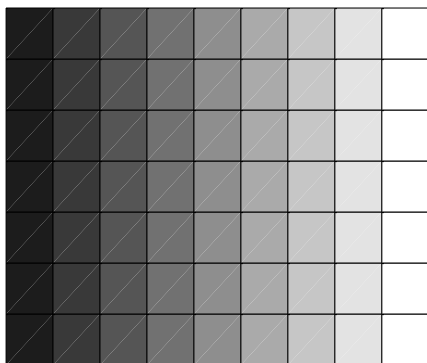
Avec cette convention, et pour l'image esclave présentée ci-dessus de taille 7x9, les matrices X et Y seront des matrices de type :

Matrice X décrivant les lignes



1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7

Matrice Y décrivant les colonnes



1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9
1	2	3	4	5	6	7	8	9

L'initialisation consiste à appliquer à l'image I_2 soit des transformations simples (rotation, homothétie), soit des transformations générales, déterminée généralement par la prise de point d'amer.

Il n'est pas possible de définir cette première étape sous la forme d'un flot, car la donnée du flot considère que la dimension de l'image maîtresse et celle de l'image esclave sont égales.

Cette étape a donc pour but de ramener justement la problématique au calcul d'un flot entre deux images de même taille et de même géométrie. Elle consiste donc à appliquer à l'image esclave une transformation qui permet de passer du système de coordonnées (X_2, Y_2) au système de coordonnées (X'_2, Y'_2) de mêmes dimensions que les coordonnées (x_1, y_1) .

Il y a deux cas de difficultés différents :

- Soit les images ont été acquises dans la même géométrie (deux images SAR en conditions interférométriques, une image VNIR et une image SWIR, etc.). Dans ce cas, la transformation à appliquer se ramène à un simple « rognage » et/ou une translation. L'image esclave ne subit pas de ré-échantillonnage.
- Soit les images ont été acquises par des capteurs ayant des caps /des résolutions différentes. Alors la transformation recherchée va faire appel à un ré-échantillonnage de l'image esclave.

Dans ce second cas, la transformation recherchée étant plus complexe, elle peut s'obtenir par des méthodes différentes :

Premier cas d'initialisation : transformation simple

Dans certains cas, la transformation peut correspondre à des transformations connues simples : une homothétie (seules les pas d'échantillonnage des images diffèrent) ou une rotation (seul le cap de l'image a été modifiée).

Une mise à l'échelle peut se faire par la fonction `imresize`, dans le cas où I_1 et I_2 sont déjà dans la même géométrie :

```
Nx1=size(I1,1);
Nx2=size(I1,2);
I2prime = imresize(I2, [nx1,ny1]);
```

Une rotation s'obtiendra de la même manière par la fonction `imrotate` en matlab.

Second cas :

La transformation recherchée ne s'exprime pas simplement. La phase d'initialisation peut se faire sous la forme de prise de point d'amer. Dans matlab, la prise de ces points d'amer peut se faire avec l'outil « `cpselect` » par la commande :

```
cpselect(I1, I2) ;
```

Cet outil va créer un ensemble de coordonnées pour l'image de départ I_1 , stockées dans `input_points`, et l'ensemble de coordonnées correspondantes pour l'image esclave I_2 , stocké dans `base_points`.

A partir de ces coordonnées, la transformation géométrique peut se calculer par :

```
t = cp2tform(base_points,input_points,'affine');
```

Cette commande permet de créer la transformation affine qui image les points de coordonnées `base_points` sur les points de coordonnées `input_points`. l'image I_2 est transformée en une image I_2' par la commande `imtransform` en matlab :

```
Nx1=size(I1,1);
Nx2=size(I1,2);
I2prime = imtransform(I2, t,'nearest','XYScale',1,...
    'XData', [1,dimy],...
    'YData', [1,dimx],...
    'FillValues', eps);
```

Troisième cas d'initialisation : geocoding

Il est enfin possible d'affecter à chaque image une grille de latitudes et longitudes pour chacun des pixels de nos images, à partir de toutes les données utiles de géoréférencement (trajectoires capteurs et MNT) puis de ré-échantillonner l'image I_2 dans le référentiel de l'image I_1 .

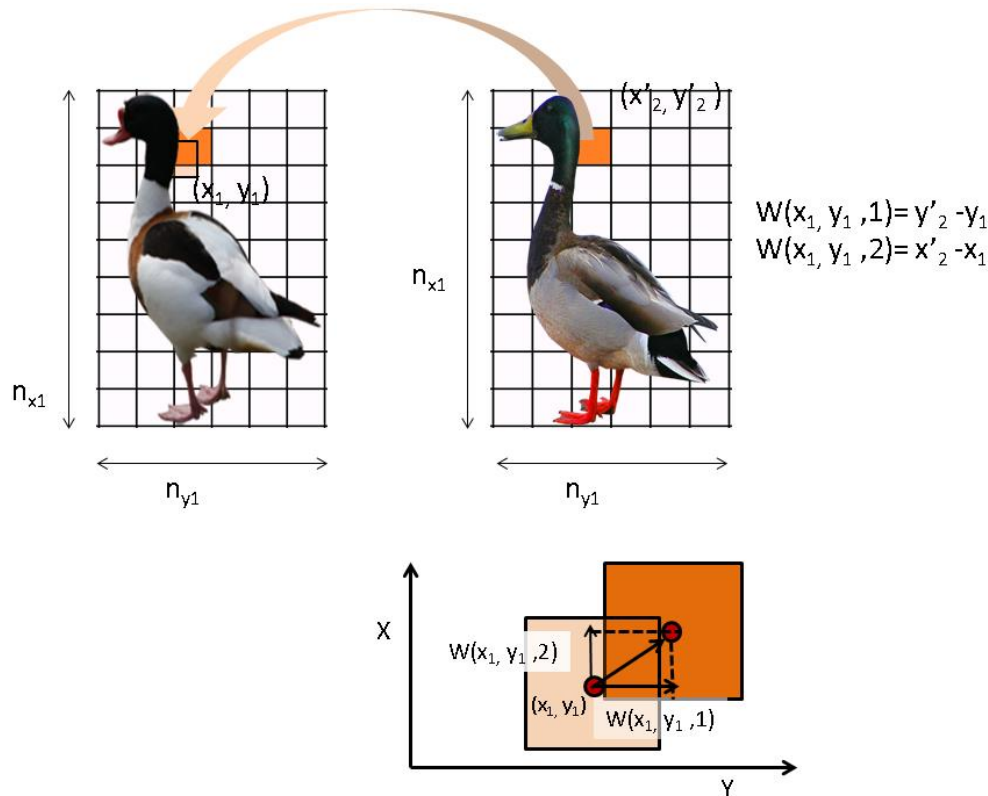
Ces opérations étant spécifiques à chaque capteur, elles ne sont pas décrites dans ce documents.

DESCRIPTION DE L'ETAPE CENTRALE : CALCUL DU FLOT PAR GEFOLKI

Cette étape prend en entrée deux images I_1 et I_2' de même taille.

Le flot décrit le passage des coordonnées de (X_2', Y_2') à (X_1, Y_1) par une matrice \mathbf{W} de dimension $(n_{x1}, n_{y1}, 2)$.

La première composante de \mathbf{W} est le déplacement à effectuer en abscisse de l'image (ou axe Y), et la seconde composante de \mathbf{W} est le déplacement à effectuer en ordonnée (ou axe X).



La fonction GeFolki va être appliquée après l'étape d'initialisation, sur deux images I_1 et I_2' de même taille. Elle donne en sortie une matrice \mathbf{W}

para est une structure qui contient les paramètres de l'algorithme. Cette entrée est optionnelle. Si elle n'est pas précisée, un paramétrage par défaut est proposé.

Pour choisir le paramétrage de l'algorithme, il faut définir une structure de la manière suivante :

```
para = struct('radius', 32:-4:8, ...  
            'levels', 6, ...  
            'iter', 2, ...  
            'contrast_adapt', false, ...  
            'rank', 4);
```

1. Le premier paramètre $r=para.radius$ est le paramètre de rayon. L'algorithme peut être lancé pour plusieurs tailles de rayon de manière itérative. On part généralement du plus gros rayon au plus petit. Pour cette raison, on le définit comme un vecteur décroissant.

Ce rayon r conditionne la taille de la fenêtre $(2r+1)*(2r+1)$ sur laquelle la corrélation entre les deux images est maximisée. Son choix résulte d'un compromis entre robustesse et niveau de détail du flot :

- Un grand rayon rend plus robuste l'algorithme
 - Lorsque le flot présente de variations rapides, le rayon doit être choisi suffisamment petit pour estimer ces variations.
2. Le second paramètre $L=\text{para.levels}$ est le nombre de niveau dans la pyramide d'échelle. Il doit être conditionné par la taille maximale du déplacement recherché : $W_{\max} < 2^L$.
 3. Le troisième paramètre $K=\text{para.iter}$ est le nombre d'itérations utilisées dans la méthode de gradient dédiée à la recherche du minimum. Typiquement, ce paramètre varie entre $K=2$ et $K=10$.
 4. Le paramètre $\text{para.contrast_adapt}$ autorise l'algorithme à rechercher une inversion de contraste. Il doit être activé pour les images hétérogènes, et mis à « false » pour les images homogènes (interférométrie SAR, etc.)
 5. Le paramètre $R=\text{para.rank}$ est le paramètre utilisé dans le filtre de rang. Une valeur typique du paramètre de rang est de $R=4$. (fenêtre 9×9)

Une fois la structure para définie, l'algorithme peut être lancé par la commande :

```
W=GeFolki(I1,I2prime,para) ;
```

DERNIERE ETAPE : REECHANTILLONAGE

Compte tenu de la dernière matrice de flot trouvée, on repasse de la description de l'image I_2' dans le référentiel (X_2', Y_2') au référentiel de l'image maîtresse (X, Y)

Une fois le flot W connu, il faut ré-échantillonner les images sur la nouvelle grille de coordonnées. Plusieurs méthodes d'interpolation sont possibles : bilinéaire, linéaire, plus proches voisins, etc.

```
data1=W(:, :, 1);  
data2=W(:, :, 2);  
I3=(interp2(x', y', I2prime, x'+data2', y'+data1', 'nearest')); % Ici avec une interpolation plus proche voisin
```