

## MID514 : Vision par Ordinateur

### TP1 : Détection et correspondance des Points d'intérêts Assemblage d'images par l'utilisation des features

Objectif du TP :

- Vérifier la robustesse des détecteurs de points d'intérêt Harris et SIFT.
- Créer des panorama d'images en combinant deux ou plusieurs images.

La construction de panorama se fait par les étapes suivantes :

1. Détection des features dans les images à assembler.
2. Trouver les listes de correspondances des features entre les images.
3. Estimer les transformation entre les images
4. Utiliser les transformations estimées pour mettre les images sur la même références.
5. fusionner les images
6. aveugler l'image résultat (photometric registration)

**Remarque :** Organiser vos programmes en utilisant des fonctions.

#### Exercice 1 :

1- Utiliser la fonction matlab *detectHarrisFeatures* (la fonction opencv *cv2.cornerHarris()*) pour détecter les features dans les images ship1 et ship2. Vérifier visuellement que l'algorithme Harris est robuste par rapport à la rotation.

2 - Est ce que vous Obtiendrez le même résultats sur les images castle1 et castle2.

3 - Refaire le même travail (1 et 2) en utilisant la fonction matlab *detectSURFFeatures* pour détecter les features SURF (version accélérée du détecteur SIFT).

#### Exercice 2 :

Le programme *panorama\_template.m* (.py pour opencv) permet de créer une mosaïque de deux images.

En premier lieu essayer de modifier ce programme en sélectionnant manuellement des correspondances entre les deux images (vous pouvez utiliser pour cela les fonctions matlab *ginput* ou *cpselect* ou la fonction opencv *cv2.selectPoints()*). 4 couples de correspondances sont théoriquement suffisants pour estimer les transformations projectives mais le résultat est instable. De préférence utiliser plus de point. Vous pouvez aussi utiliser la fonction *cpcorr* pour raffiner vos sélections.

Pour estimer les paramètre de transformation, vous pouvez utiliser la fonction matlab *svd* (*cv2.SVD()* en opencv) .

$$[U, S, V] = \text{svd}(Z); \quad h = V(:, \text{end}); \text{reshape}(h)$$

Tester Votre programme avec des images des séquence case1, case2.

Vérifier que vous pouviez améliorer vos résultats en utilisant la méthode du RANSAC pour estimer les paramètres de transformation.

**Exercice 3 :** Modifier le programme `panorama_template.m` pour construire une image panoramique à partir de plusieurs images. Tester votre code sur les séquences `case1`, `case2`, et `case3`.

**N.B :** bien choisir l'image de référence.

**Exercice 4 :** Modifier votre code pour créer plusieurs images panoramiques à partir d'une séquences contenant un mélange d'images correspondantes à des scènes différentes. Tester votre code sur la séquence `case4`.