Année 2024/2025
Master Informatique Visuelle

Série de travaux pratiques n°4
Vision par Ordinateur

Exercice 1:

Le programme fourni « sfm.py » permet de calculer la structure 3D d'une scène à partir d'une séquence d'images prises par une caméra calibrée.
Il est demandé :

- dans un premier temps, d'exécuter le programme sur les images à télécharger du site suivant. Visualiser la scène calculée moyennant par exemple l'outil MeshLab. Un dataset de petite taille suffit.
  https://www.maths.lth.se/matematiklth/personal/calle/dataset/dataset.html
- En second lieu, il est demandé de calibrer votre caméra et de prendre des images d'une scène externe. Appliquer le programme fourni et visualiser la scène calculée moyennant par MeshLab.
- Décrire les opérations exécutées pour aboutir à une telle structure (voir figure 1 qui illustre la scène reconstruite pour la dataset door).

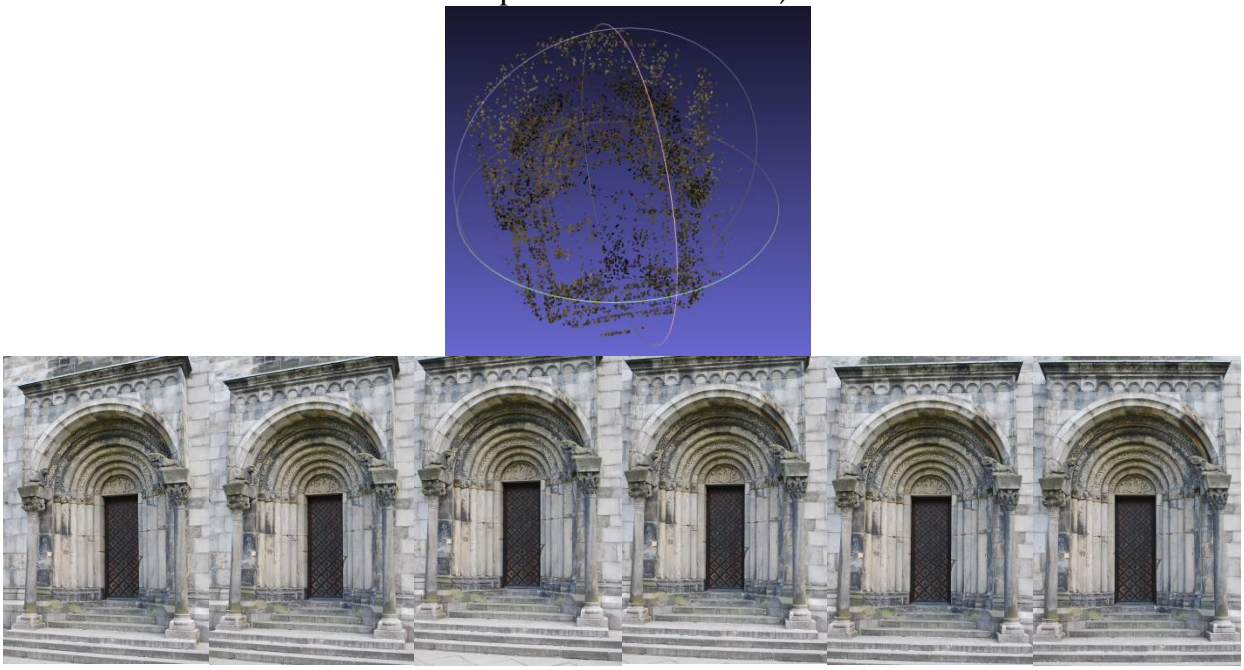

Figure 1. La structure reconstruite (nuage de points) à partir d'une séquence d'images (dataset door)

Exercice 2 :

Il s'agit d'implémenter la factorisation de Tomasi et Kanade. A partir d'un ensemble d'images prise par votre caméra, réaliser les étapes suivantes :

- Détection des points SIFT sur chacune des images
- Mise en correspondance entre les points SIFT
- Choisir les bons appariements,
- Calculer la matrice W après avoir fait un changement de repère (vers le barycentre de chaque nuage de points dans le plan image).
- Calculer la décomposition en valeurs singulières (SVD) de W. Pour cela, utilisez la méthode (linalg.svd) :

  numpy.linalg.svd
  linalg.svd(a, full_matrices=True, compute_uv=True, hermitian=False)

  full_matrices : bool, optional

  If True (default), u and v have the shapes (M, M) and (N, N), respectively. Otherwise, the shapes are (M, K) and (K, N), respectively, where K = min(M, N).
  Returns:
  u : { (..., M, M), (..., M, K) } array

  Unitary matrices. The actual shape depends on the value of full_matrices. Only returned when compute_uv is True.
  s : (..., K) array
  The singular values for every matrix, sorted in descending order.
  v : { (..., N, N), (..., K, N) } array
  Unitary matrices. The actual shape depends on the value of full_matrices. Only returned when compute_uv is True.