

# Reconstruction 3D photométrique

## Projet Vision (Partie 2)

La reconstruction 3D photométrique consiste à calculer l'objet 3D à partir d'une ou plusieurs images. Elle est considérée comme problème inverse de la formation d'image.

La modélisation des techniques de la reconstruction 3D peuvent être classée en deux grandes catégories selon le modèle de la caméra utilisé : géométrique ou photométrique (voir le cours ch.3).

L'objectif de ce TP est la réalisation d'une application de reconstruction 3D simple avec python suivant trois étapes. La première étape consiste à charger et à préparer les données.

### Etape 1. Préparation de données (4 pts) :

Le dataset utilisé est composé par les fichiers suivants :

- N images du même objet prises par la même position de la caméra mais avec différentes conditions (positions et intensités) de la source lumineuses. Ces images sont stockées en RGB (BGR sous opencv) sur **16bits**.
- Un fichier contient les noms des N images déjà mentionnées. (filenames.txt)
- Un fichier contient les positions des sources lumineuses. (light\_directions.txt)
- Un fichier contient les intensités des sources lumineuses. (light\_intensities.txt)
- Une image contient le masque de l'objet (mask.jpg).

**L'objectif de cette étape consiste à :**

- 1- Créer une fonction **load\_lightSources** qui return une matrice (Nx3) qui représente les positions des sources lumineuses (chaque ligne représente une position x,y,z).
- 2- Créer une fonction **load\_intensSources** qui return une matrice (N\*3) qui représente les intensités des sources lumineuses (chaque ligne représente l'intensité d'un pixel **R,G,B**).
- 3- Créer une fonction **load\_objMask** qui retourne une matrice (image) binaire tel que : **1** représente un pixel de l'objet et **0** : un pixel du fond.
- 4- Créer une fonction **load\_images** qui permet de charger les N images (les images sont sauvegardées sur 16bits non signés).
  - a. Changer l'intervalle des valeurs de **uint16** [0 ,  $2^{16}-1$ ] à **float32** [0 , 1]
  - b. Diviser chaque pixel sur l'intensité de la source ( **$B/intB, G/intG, R/intR$** )
  - c. Convertir les images en niveau de gris ( **$NVG = 0.3 * R + 0.59 * G + 0.11 * B$** ).
  - d. Redimensionner l'image telle que chaque image est représentée dans une seule ligne.
  - e. Ajouter les images dans un tableau (pour former une matrice de N lignes et (h\*w) colonnes où chaque ligne représente une image).
  - f. Retourner la matrice des images.
- 5- Le dataset est disponible sur le lien suivant :  
[http://www.vision.lyabada.com/projet21/objet1PNG\\_SII\\_VISION.zip](http://www.vision.lyabada.com/projet21/objet1PNG_SII_VISION.zip)

## Etape 2. Calcul des normales (4 pts) :

Créer une fonction ***calcul\_needle\_map*** qui calcule le champ de normales (le vecteur normal pour chaque pixel). Cette fonction permet de :

- Charger les N images dans une variable ***obj\_images*** (à l'aide de la fonction : **load\_images**)
- charger les positions des sources lumineuses dans une variable ***light\_sources*** (à l'aide de la fonction : **load\_lightSources**)
- charger le masque de l'objet dans une variable ***obj\_masques*** (à l'aide de la fonction : **load\_objMask**)

Basant sur la modélisation suivant (voir le cours) :

$$E = S.N$$

E : un vecteur (matrice à une seule colonne) qui représentent les niveaux de gris d'un seul pixel (chaque colonne de la matrice ***obj\_images*** )

S : la matrice ***light\_sources***

N : le vecteur normal d'un seul pixel (x,y,z)

1- Calculer les normales de l'objet (une matrice de h lignes, w colonnes et trois composantes x,y,z) (3tps).

2- Afficher les normales dans une image (x,y,z au lieu de BGR) (1pts).

## Etape 3. Intégration des normales pour déterminer Z (1.5 pts) :

Proposer une méthode simple pour calculer la profondeur Z à partir des normales N et afficher l'objet 3D.

## Rapport (2.5 points) :

Rédiger un rapport en détaillant l'application et le code source (max 6 pages).

## Remarques :

- La présentation de l'application est importante.
- La programmation se fera avec python.
- Les commentaires et l'organisation du code source sont importants.
- Un seul fichier compressé (zip) contient le rapport (PDF) et le code source seul (sans données) doit être envoyé à l'adresse [abada.lyes@gmail.com](mailto:abada.lyes@gmail.com) avant le 12/12/2021 (à 00h00)
- La partie 1 est notée sur 5 points et la partie 2 est notée sur 12 points.