

Projet : Détection, Calibrage et Calcul de Position d'un Objet à l'aide de Deux Caméras

Partie 1 : Détection d'objet

1. Proposer une méthode pour détecter un objet ponctuel connu, permettant de retourner sa position (x, y) ou celle de son centre.
2. Décrire en détail la méthode proposée dans le rapport (2 pages maximum).

Partie 2 : Calibrage de la caméra

1. Développer une méthode de calibrage de caméra qui génère les matrices de paramètres intrinsèques et extrinsèques.
2. Présenter le principe théorique et la technique de cette méthode dans le rapport (2 pages maximum).

Partie 3 : Calcul de la position et de la distance d'un objet par rapport à deux caméras

1. Fixer deux caméras orientées dans la même direction avec un écart horizontal de b , comme illustré dans la figure ci-dessous (vue en cours).
2. Calibrer les deux caméras en utilisant la méthode proposée dans la Partie 2.
3. Déterminer la position (par pixel) d'un objet ponctuel dans chacune des deux caméras, en s'assurant que la coordonnée y est identique dans les deux images. En utilisant la méthode proposée dans la Partie 1.
4. Calculer et afficher en temps réel la position 3D de l'objet dans les coordonnées du monde réel.
5. Calculer et afficher les coordonnées du monde réel du point situé à mi-distance entre les centres de projection des deux caméras.
6. Détailler les calculs théoriques dans le rapport (2 pages maximum).

Partie 4 : Amélioration

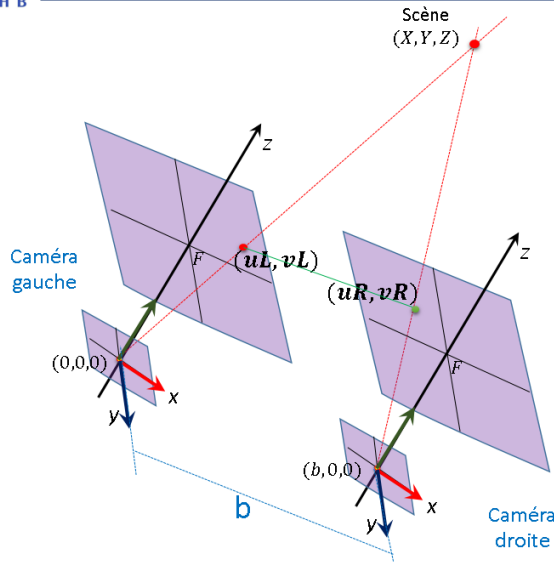
1. Proposer une fonctionnalité ou une amélioration de votre choix.

Remarque 1:

1. Si la caméra est fixe, un seul calibrage est nécessaire. Dans le cas contraire, on peut placer l'outil de calibrage (par exemple, une mire de type checkboard) à une distance fixe par rapport à l'objet.
2. Un checkboard bidimensionnel (feuille) ou tridimensionnel (cube) peut être utilisé pour le calibrage.
3. Les tests sont réalisés dans deux configurations :
 - Objet fixe avec des caméras en mouvement.
 - Objet en mouvement avec des caméras fixes.
4. La partie 3 repose sur des calculs matriciels **sans utilisation de fonctions prédéfinies**.
5. Le calcul illustré dans la figure suivante est un exemple simplifié ; des méthodes de calcul **plus avancées** peuvent également être utilisées.



Calibration et Stéréovision



$$\text{3D vers 2D} \quad \begin{cases} u_L = f_x \frac{x}{z} + o_x \\ v_L = f_y \frac{y}{z} + o_y \end{cases}$$

$$\text{3D vers 2D} \quad \begin{cases} u_R = f_x \frac{x-b}{z} + o_x \\ v_R = f_y \frac{y}{z} + o_y \end{cases}$$

Solution

$$\begin{cases} x = \frac{b(u_L - o_x)}{(u_L - u_R)} \\ y = \frac{b f_x (v_L - o_y)}{f_y (u_L - u_R)} \\ z = \frac{b f_x}{(u_L - u_R)} \end{cases}$$

Remarque 2 :

1. Le code source doit être soigneusement commenté ;
2. L'interface graphique est requise ;
3. Lors de la démonstration, la modification du code source est interdite ; toutes les fonctionnalités doivent être lancées exclusivement depuis l'interface graphique ;
4. La programmation doit être réalisée en Python. L'utilisation de VSCode est fortement recommandée. Si un autre environnement est utilisé, le programme remis doit être indépendant de toute dépendance externe ;
5. Le rapport ne doit pas excéder 8 pages, en respectant la limite de pages pour chaque section, et inclure les parties supplémentaires (introduction, interface utilisateur, etc.) .

La préparation et l'envoi du projet :

- Le code source et le rapport doivent être compressés dans **un seul** fichier portant **les noms des étudiants** ;
- Envoyez le fichier à l'adresse suivante : abada.lyes@gmail.com;
- Pour plus d'informations, veuillez utiliser l'adresse suivante : lyes_abada@yahoo.fr;
- Le dernier délai pour envoyer le rapport et le code est le : **08 décembre 2024** ;
- La démonstration se fera le **10 et le 12 décembre 2023** ;
- Vous devez mentionner les configurations nécessaires pour exécuter le code dans le rapport (bibliothèques à téléchargées...) ;

بالتوفيق