TD2 – Placement objet virtuel et recalage “naïf” - 1 séance

Objectif : placement d’un objet virtuel sur l’image de référence, et illustrer les limites d’un recalage de l’objet uniquement basé sur la détection des coins du damier.

1 Barycentre du damier dans le repère caméra 3D

Exercice 1
L’objectif final de cet exercice est de placer virtuellement un cercle au centre de coins détectés sur la feuille. Le résultat attendu est illustrée par la figure 1.

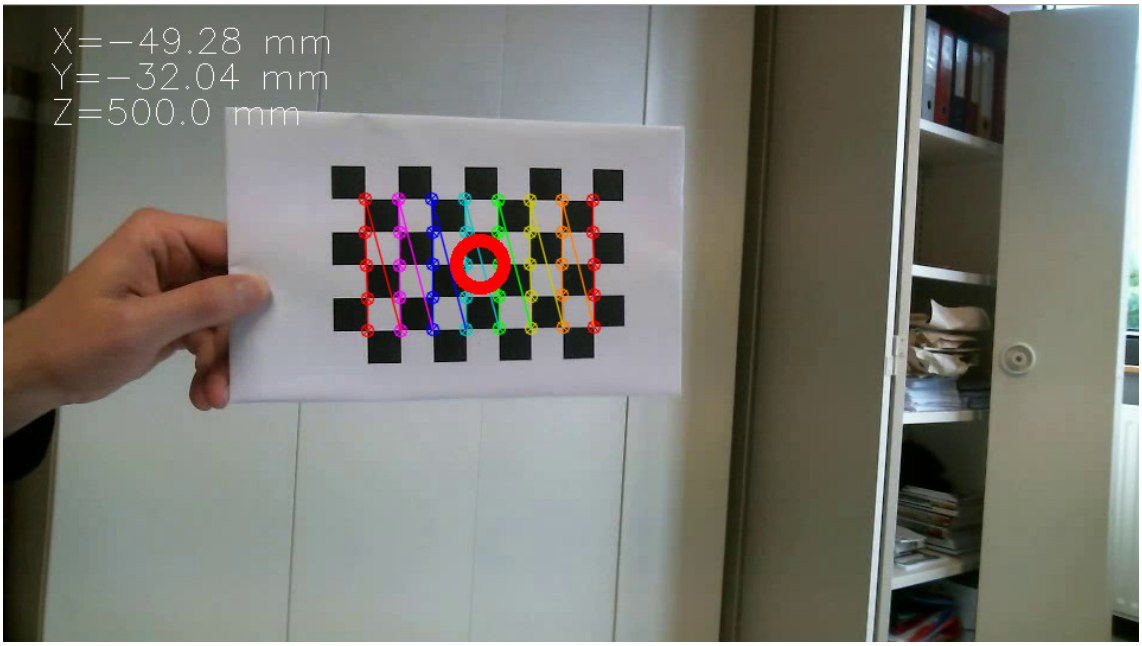

FIGURE 1 – Résultat attendu à l’issue de l’exercice 1.

Les différentes questions suivante forment des étapes vous permettant d’atteindre ce résultat.

1. Détectez les coins `corners_2d` afin d’obtenir leurs coordonnées “pixels” dans l’image.
2. En déduire leurs coordonnées 3D (`corners_3d`) dans le repère caméra (en millimètres), sachant que la dernière coordonnée Z est connu ($Z = 500$ mm).
3. Calculez ensuite le barycentre `b` de ces points en 3D `chessboard_center`, et afficher la valeur des deux premières coordonnées superposées à l’image.
   Indication : Les points stockées dans le tableau numpy `corners_3d` sont organisés selon les lignes (les 3 colonnes sont les coordonnées en X, Y et Z). La moyenne entre les lignes

(i.e. ou chaque colonne) d'un tableau numpy est obtenue par "numpy.mean(tab,0)" (0 désigne que la moyenne se fait entre les lignes).

4. Dessinez un cercle sur l'image, dont le centre est le projeté de `b` dans le plan image. *Remarque* : on pourra utiliser la fonction `world2pixels` fournie avec la correction du TP précédent et gérant le cas d'un point (`if len(...shape)==1`), et pas seulement le cas d'un ensemble de points (tableau à plusieurs lignes), sinon vous aurez un message d'erreur.

Exercice 2

1. Utilisez le code exemple `TD2_caneva_video.py` fourni pour la lecture et l'affichage de la vidéo, incluant le tracking des cases du damier.
2. Généralisez la procédure de l'exercice précédent à l'ensemble de la vidéo `video_damier.m4v`.

Il est à noter que ces coordonnées sont très approximatives (voire fausses) car les calculs font l'hypothèse que le damier est toujours à une distance de Z=500 mm, ce qui n'est évidemment pas le cas.

2 Placement d'un objet virtuel (carré plaqué sur le damier).

Exercice 3

L'objectif de cet exercice est placer un carré de taille fixe dans le monde de la caméra comme illustré par la figure 2.

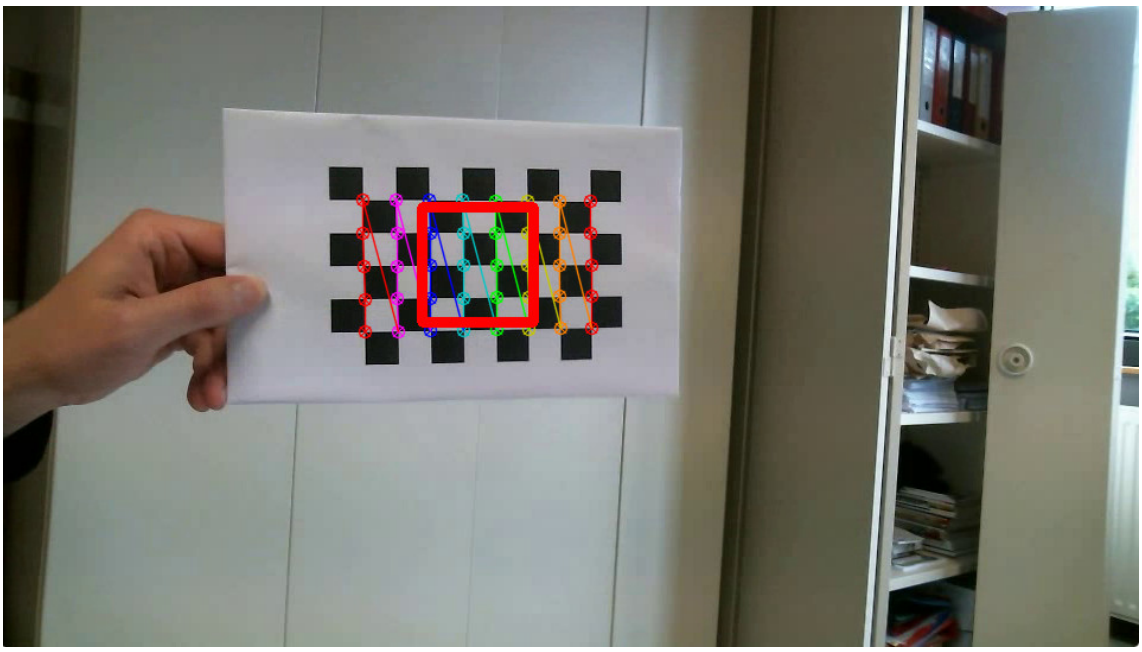

FIGURE 2 – Résultat attendu à l'issue de l'exercice 3.

On travaillera sur l'image de référence.

1. Ajoutez au fichier `helper.py` une fonction `create_square(points3d, size)` dont le rôle est de calculer les coordonnées des 4 coins d'un carré dans le repère caméra, centré sur

le centre du damier (barycentre 3D des coins du damier), parallèle à celui-ci (i.e. selon X,Y) et d'arête égale à 60 millimètres.
Cette fonction retourne le carré sous forme d'une liste de 4 points correspondant aux coins du carré : `numpy.array([[x0,y0,z],[x1,y1,z],....])`.

2. Projetez ce carré dans le repère 2D pixels, puis superposez-le à l'image affichée à l'aide de la fonction opencv `polylines`.
Indication : La fonction opencv `polylines` prend en second paramètre une liste de tableaux convertis en entiers (`[tab.astype(np.int32)]`). Chaque tableau correspond à un polygone et contient les coordonnées (x,y) des points constituant le polygone associé.

Exercice 4

1. Comme dans l'exercice 2, appliquer le code de l'exerice précédent sur chaque image de la vidéo.
2. Pour quelle raison, le résultat n'est-il pas satisfaisant ?