

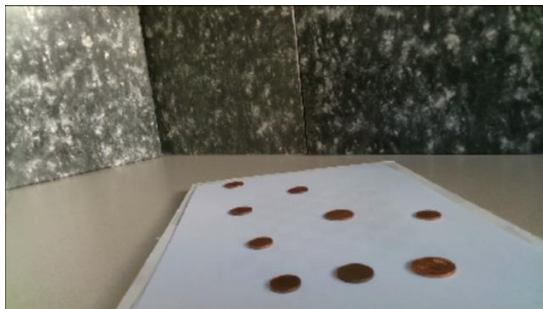
Guia Pràctica 3 – Projectió Perspectiva + Calibratge Càmera

Curs 2021-2022 – Dept. Ciències de la Computació, UAB.

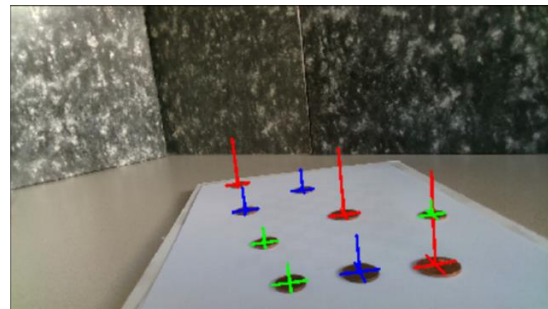
1 Context

Igual que a la pràctica anterior, el treball a realitzar en aquesta pràctica té com a objectiu implementar un sistema de inspecció basat en visió per computador. En aquest cas, t'han demanat un sistema perquè classifiquis 3 tipus de peces circulars en base a la seva dimensió (àrea, radi, diàmetre,... t'ho deixen al teu criteri).

A la zona d'inspecció s'hi presenten sempre 3 peces de 3 mides diferents (9 peces en total), i es vol assignar cadascuna d'elles la seva classe segons mida (petita, mitjana, gran). Per restriccions en el muntatge, la zona d'inspecció es captura des d'una càmera situada de manera obliqua respecte la zona de treball. Es vol que s'indiqui sobre la imatge captada la classe de cada element, representant-ho utilitzant la idea de *realitat augmentada* (veure Figura 1). Això facilitarà a un operari interpretar fàcilment la classe de cada peça, per processar-les degudament.



Imatge d'entrada



Imatge de sortida, mostrant la classificació de les peces amb un possible estil de realitat augmentada

FIGURA 1: POSSIBLE REPRESENTACIÓ DE LA CLASSE DE CADA PEÇA USANT REALITAT AUGMENTADA.

2 Enunciat

En aquesta pràctica has de crear un Jupyter notebook per resoldre el problema de classificació de peces plantejat, i mostrar el resultat obtingut mitjançant realitat augmentada. La situació de partida és molt similar al de la pràctica 2. El punt de vista de les peces no és el desitjable, i una visió zenital de l'escena et facilitarà molt la feina. Com a bon enginyer, reutilitza el codi de la pràctica 2 per resoldre el problema de visió.

Per mostrar la classificació resultat via realitat augmentada et caldrà caracteritzar com el món 3D projecta sobre les imatges. Prenent el model de càmera lineal mostrat a classe, openCV proveeix de funcions per estimar-ne els seus paràmetres a partir de diverses imatges d'un patró de calibratge planar. D'aquest procés en diem calibratge de la càmera. Amb la matriu de calibratge de la càmera, i coneixen la posició 3D de la càmera respecte la zona d'inspecció, podem projectar objectes 3D sobre la imatge que informin de la classe de cada peça.

2.1 Repte 1: Detecció i caracterització de les peces circulars

Descarrega i descomprimeix l'arxiu imatges.zip del campus virtual. Veuràs que hi ha diferents imatges organitzades en 3 directoris:

- Calibratge: conjunt de 22 imatges, que s'empraran per calibrar la càmera.
- Cas1 i Cas2: directoris contenint respectivament 2 imatges:
 - Patro.png: Permet caracteritzar la posició de l'àrea d'inspecció respecte la càmera.
 - Peces.png: Imatge mostrant les peces a classificar.

En aquest repte has d'implementar el codi per processar les imatges als directoris Cas1 i Cas2. Per a fer-ho, replica l'aproximació seguida a la pràctica 2:

- Càlcul de l'homografia entre el patró i la seva vista zenital (Figura 2)
- Transformació de la imatge Peces.png a la seva vista zenital
- Procés de la imatge zenital de Peces.png, obtenint la coordenada píxel on es troba el centre de cada peça, i un indicador de la seva dimensió (radi, diàmetre, àrea,el que consideris)

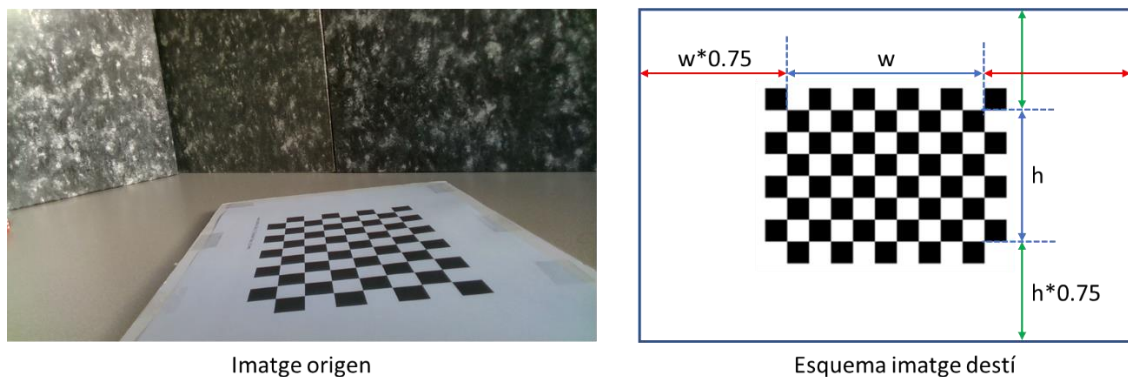


FIGURA 2: TRANSFORMACIÓ A REALITZAR

2.2 Repte 2: Calibratge de la càmera.

Adapta el següent tutorial d'opencv per calibrar la càmera usada a la pràctica.

- https://docs.opencv.org/3.4/dc/dbb/tutorial_py_calibration.html

De manera similar a com has determinat l'homografia, per calibrar la càmera s'ha d'emparellar els punts de creuament que es veuen a les imatges amb uns punts associats. En aquest cas, els punts associats són tridimensionals, i s'estableixen a partir de situar el sistema de referència de les coordenades món sobre el patró de calibratge, en el punt de creuament de la cantonada superior esquerra (veure Figura 3). Els punts sobre el patró tenen tots la coordenada Z a zero.

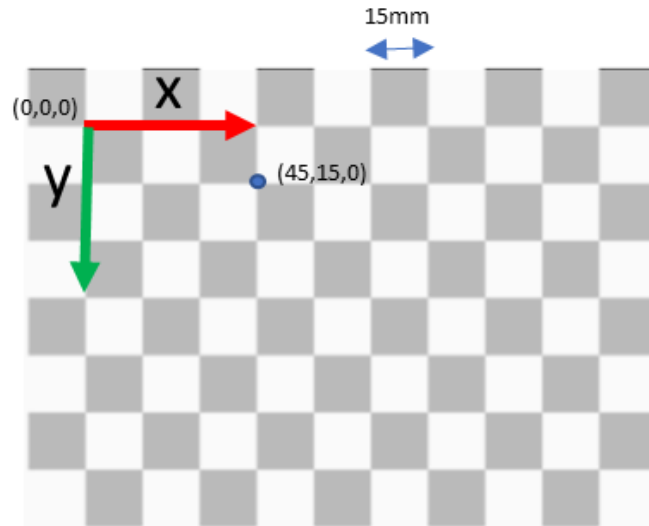


FIGURA 3: SISTEMA DE COORDENADES MÓN SITUAT SOBRE EL PATRÓ DE CALIBRATGE

La funció de calibratge d'openCV caracteritza la càmera retornant la informació en diferents paràmetres:

- **mtx**: matriu de calibratge, amb els paràmetres intrínsecs de focals efectives en x i y, i la coordenada del punt principal.
- **dist**: paràmetres modelant la distorsió no lineal de la lent.
- **rvecs**, **tvecs**: paràmetres extrínsecs corresponents a cada imatge usada en el procés de calibrar. Permeten projectar punts expressats en coordenades món cap a un sistema de coordenades càmera.

2.3 Repte 3: Mostrar classificació de les peces amb realitat augmentada.

En el repte 1 has desenvolupat un mètode per localitzar i mesurar les peces en la imatge zenital de la zona d'inspecció. Aquestes coordenades en píxels les podem passar a coordenades món, a través de la relació que mostra la figura 4.

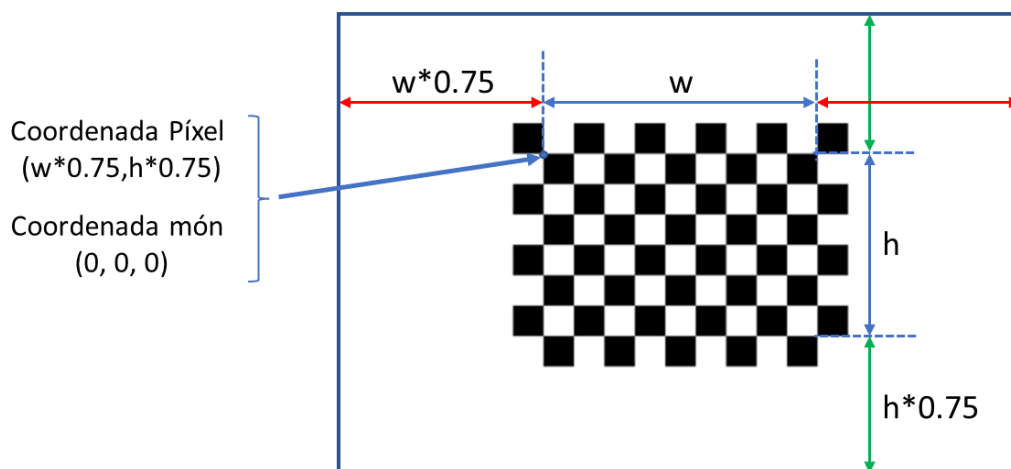


FIGURA 4: RELACIÓ DE LES COORDENADES PÍXEL DE LA IMATGE ZENITAL AMB LES COORDENADES MÓN

Podem situar doncs hipotètics objectes 3D en el sistema de coordenades món, en les posicions on s'ubiquen les peces detectades. Cal tenir en compte que el patró assumim que està a la posició $z=0$ del món, i que openCV estableix que l'eix Z creix "en profunditat". És a dir, per situar un punt 3D per sobre del patró de referència, la seva coordenada Z haurà de ser negativa.

Un cop es tinguin definits uns objectes 3D en el món, representant les peces detectades, el següent pas és projectar aquests objectes 3D sobre la imatge inicial (amb l'efecte de la perspectiva). Per resoldre aquesta part convé adaptar el que es mostra en aquest tutorial:

- https://docs.opencv.org/3.4/d7/d53/tutorial_py_pose.html

La idea és la següent:

- Amb els punts de creuament detectats a la imatge Patro.png i els paràmetres del calibratge de la càmera **mtx** i **dst** podem estimar la posició de la càmera respecte el sistema de coordenades món emprant la funció *cv2.solvePnP*. Ens la dona en un vector de rotació de Rodrigues **rvecs** i un vector de translació **tvecs**.
- La funció *cv2.projectPoints* utilitza els paràmetres **mtx**, **dst**, i la posició de la càmera a **rvecs** i **tvecs** per projectar punts 3D en coordenades món cap el píxel de la imatge que els correspon. S'ua doncs aquesta funció per determinar els píxels on projecten els objectes 3D dissenyats per informar de la classe de cada peça.
- Les coordenades píxel obtingudes s'utilitzen per dibuixar sobre la imatge de sortida, emprant les funcions que proporciona openCV: *cv2.line*, *cv2.drawContours*, ...