



UNIVERSITAT DE
BARCELONA

Facultat de Matemàtiques
i Informàtica

GRAU DE MATEMÀTIQUES

Treball final de grau

**Creació d'un sistema multicàmera
calibrat per a recol·lecció i anàlisi
de dades sobre la mirada**

Autor: Sergi Clivillé Sánchez

Director: Sergio Escalera Guerrero i Cristina Palmero Cantariño

Realitzat a: Departament de Matemàtiques i Informàtica

Barcelona, 25 de gener de 2021

Abstract

The study of the gaze behavior using images is an ongoing research area of significant relevance in sociology and psychology for its participation in multiple cognitive processes. The use of Convolutional Neural Networks for their study is the standard option, but they need accurate gaze data in order to improve their estimation. The lack of gaze data in some populations, such as children or the elderly, makes the estimations of the devices that use this technology less accurate with them. In this project, and with the objective to solve this lack of available data, we propose a 3-camera system that will allow us to calculate the gaze vectors of any person in a simple, accurate and minimally invasive way, explaining step by step all the mathematical concepts involved in its calculation.

Resum

L'estudi del comportament de la mirada a partir d'imatges és un camp d'investigació especialment rellevant dins la sociologia i psicologia per la seva participació en diversos processos cognitius. L'ús de xarxes neuronals convolucionals pel seu estudi és l'opció estàndard, però requereixen dades precises sobre la mirada pel seu correcte funcionament. La falta de dades sobre la mirada en certes poblacions, com nens o ancians, fa que els dispositius que utilitzen aquesta tecnologia siguin menys precisos amb ells. En aquest treball, i amb l'objectiu de solucionar aquesta manca de dades disponibles, proposem un sistema de 3 càmeres capaç de calcular els vectors mirada d'una persona qualsevol de forma senzilla, precisa i poc invasiva, explicant pas a pas els conceptes matemàtics involucrats en el seu càlcul.

Índex

1	Introducció	1
2	Metodologia	3
2.1	Escena 3D	3
2.1.1	Estímul Visual	3
2.2	Model de Càmera Estenoipeica	4
2.2.1	Generalització del model: Desviació del punt principal	7
2.2.2	Generalització del model: Píxels no quadrats	9
2.2.3	Distorsió	10
2.2.4	Paràmetres Extrínsecs	11
2.3	Calibratge de Càmeres	12
2.3.1	Calibratge Intrínsec	12
2.3.2	Calibratge Càmera-Càmera	14
2.3.3	Calibratge Càmera-Pantalla	15
2.4	Geometria Ocular	18
3	Avaluació	20
3.1	Detalls d'implementació	20
3.1.1	Calibratge Intrínsec	21
3.1.2	Calibratge Càmera-Càmera	22
3.1.3	Calibratge Càmera-Pantalla	23
3.1.4	Estímul visual	24
3.1.5	Càlcul del punt origen	24
3.1.6	Càlcul del Vector Mirada	25
3.2	Resultats	26

1 Introducció

L'estudi de la interacció persona-ordinador (IPO) analitza la forma en què interaccionem amb computadores, robots, dispositius intel·ligents i altres màquines, quan en fem ús, per tal de conèixer la influència personal o social d'aquesta interacció i millorar l'experiència de l'usuari.

Hi ha molts aspectes del comportament d'una persona quan interacciona de forma natural amb una màquina que poden ser modelats i analitzats. Entre ells destaquen el comportament del cos i la cara com, per exemple, en sistemes intel·ligents que s'adapten a l'expressió facial o als gestos i posicions de les mans. La veu i la mirada són altres indicadors del nostre comportament rellevants en l'estudi de la IPO.

En el cas de la mirada, el desenvolupament de dispositius capaços de recollir informació sobre ella és de potencial interès per a aplicacions de màrqueting, entreteniment o, fins i tot, assistència a persones amb discapacitat. De fet, l'estudi del comportament de la mirada té un gran pes dins la sociologia i psicologia, ja que es considera un dels indicadors més importants en l'anàlisi del comportament no verbal per la seva alta participació en molts processos cognitius com la transmissió, conscient o no, de l'estat emocional o cap a on es dirigeix l'atenció [2].

Una de les tasques més complicades, dins l'estudi del comportament de la mirada, és la d'estimar la direcció de la mirada només a partir d'imatges de la cara d'una persona. En aquest cas, és habitual fer ús de Xarxes Neuronals Convolucionals (CNN) per tal d'obtenir una bona estimació.

La precisió de la CNN encarregada d'estimar el vector mirada depèn d'un conjunt de dades, les dades d'entrenament, format per diverses imatges de cares de persones amb el seu vector mirada corresponent adjunt. Aquest conjunt de dades mostra l'estimació perfecta del vector mirada (anomenat *ground truth* del vector mirada) en diversos casos i, mitjançant certs algorismes d'aprenentatge, s'encarrega de millorar la precisió de l'estimació donada per la CNN.

L'objectiu d'aquest treball és proporcionar un mètode que permeti formar aquest conjunt de dades i que sigui apte per recollir i analitzar les dades sobre la mirada pertinents de qualsevol persona, especialment per a persones excloses o poc representades en aquests estudis com ara nens o ancians, ja que la falta de dades conegudes i contrastades d'aquests grups limita l'ús i l'efectivitat de les CNNs en ells.

Hi ha gran diversitat de mètodes que permeten calcular el ground truth del vector mirada i que podem utilitzar per conformar un conjunt de dades d'entrenament adient. Alguns d'aquests mètodes són molt intrusius o restringeixen el seu ús a condicions de laboratori, com aquells que utilitzen electrooculogrames. Altres opcions, disponibles actualment al mercat, requereixen dispositius específics com càmeres infraroges o *eye trackers* que poden ser invasius, costosos o restringir el seu funcionament òptim a certes condicions.

Així doncs, proposem un sistema per a l'obtenció del ground truth del vector mirada basat en càmeres remotes (no muntades al cap) per ser una de les alternatives més flexibles i poc invasives. Aquesta opció requerirà un calibratge de càmeres previ, on estimarem

els paràmetres de la càmera involucrats en el procés de creació d'imatges, amb l'objectiu d'obtenir un vector mirada precís.

El nostre sistema comptarà amb tres càmeres per tal de proporcionar el ground truth del vector mirada des de diferents angles i posicions, augmentant així la diversitat d'imatges que conformaran el conjunt de dades d'entrenament i l'adaptabilitat de la CNN a diferents imatges facials.

En la següent secció, explicarem pas a pas els detalls del funcionament del nostre sistema multicàmera. Començarem per detallar les condicions que ha de reunir el nostre entorn de treball i les característiques de cada sessió de gravació, especificant què haurà de fer el participant durant la prova i per quin motiu.

Tot seguit, especificarem els detalls del model matemàtic considerat per establir una correspondència entre el nostre entorn tridimensional i les imatges bidimensionals obtingudes per les càmeres. Veurem que aquesta correspondència pot expressar-se de forma matricial i dependrà de diversos paràmetres, que dividirem en paràmetres extrínsecs (els que depenen de la posició a l'espai de cada càmera) i paràmetres intrínsecs (els que depenen de les característiques internes de cada càmera). Parlarem de cada un d'aquests paràmetres i de com estimar-los en el procés de calibratge.

Acabarem la següent secció parlant del model geomètric de l'ull, de com estimarem la seva posició a l'espai, i d'on està el punt d'origen del vector mirada que volem calcular.

Per acabar, a la secció final especificarem els detalls de la nostra implementació i mostrarem i explicarem els resultats obtinguts.

2 Metodologia

2.1 Escena 3D

El nostre objectiu és estimar, amb la màxima precisió possible, el ground truth dels vectors mirada d'una persona en un entorn controlat i sota certes condicions.

L'entorn escollit serà una sala ben il·luminada amb un escriptori de treball que contingui un ordinador, una pantalla i una cadira qualsevol que permetin a la persona observar la pantalla directament i sense dificultats mentre roman assegut a la cadira, que situarem a una distància fixa predeterminada de la pantalla. Al voltant de la pantalla col·locarem les 3 càmeres, de manera que puguem gravar des d'elles la cara i els ulls de la persona sota tres angles diferents.

Així doncs, per tal de recol·lectar les dades sobre la mirada que necessitem, durant cada sessió de gravació el participant seurà a la cadira, que haurà de romandre en la mateixa posició durant tota la prova, i observarà un determinat estímul visual que es mostrarà per pantalla mentre les càmeres graven la cara del participant.

L'estímul visual que mostrem per pantalla durant la prova ens donarà el punt final del vector mirada que volem calcular, és a dir, el punt que està observant el participant. Per altra banda, el punt d'origen de cada vector mirada s'estimarà a partir de les imatges de la cara del participant capturades per les càmeres.

Aquests punts s'expressaran en mil·límetres en un sistema de coordenades que definirem a conveniència per tal de representar el nostre entorn de treball tridimensional i conèixer les posicions relatives dels seus elements a l'espai.

En particular, ens interessa conèixer les posicions relatives a l'espai de cada una de les càmeres, la cara del participant i cada punt mostrat per pantalla. Per aquest motiu, és de vital importància que la posició de les càmeres i pantalla quedi prèviament fixat i no es modifiqui durant la prova. A la pràctica, restringir els moviments del cap millora l'estimació de la posició dels ulls, per tant, durant la sessió de gravació es demanarà al participant que mantingui el cap quiet i només mogui els ulls per observar l'estímul visual mostrat.

2.1.1 Estímul Visual

Existeix una forta relació entre el que els usuaris miren i l'aspecte dels seus ulls. Per tal d'estudiar aquesta relació, s'han proposat diversos tipus de sistemes per recol·lectar el ground truth del vector mirada on s'utilitzen diferents estímuls visuals, com ara vídeos, imatges, textos, punts o objectes tridimensionals [2, 3, 4, 5].

Normalment, els estímuls visuals que s'utilitzen es restringeixen a aquells que poden ser mostrats per una pantalla o projectats en ella, ja que són particularment rellevants degudes a les seves implicacions pràctiques en dispositius digitals amb pantalla i, a més, és una opció fàcilment aplicable.

En el nostre cas, utilitzarem com a estímul visual cercles que es generen en posicions aleatòries de la pantalla i van reduint el seu radi fins a convertir-se en un únic punt.

Es demanarà que el participant pressioni una tecla, en el moment en què el cercle es torna un punt, per indicar-nos que està mirant al punt indicat i proporcionar-nos les coordenades en píxels d'aquest punt. Un cop el cercle sigui un punt, el participant tindrà un temps limitat abans que el punt desaparegui i es generi un nou cercle en una nova posició.

Hem escollit aquesta opció perquè la reducció dels cercles i la seva posició aleatòria fa que el participant hagi d'estar atent a la pantalla i ens proporciona punts diferents a cada prova. A més, la limitació del temps disponible per clicar i proporcionar-nos una bona mostra, ens assegura que el participant realment està mirant al punt marcat i, a més, també incita a augmentar l'atenció del participant, ja que la prova acaba quan ens dóna un nombre prefixat de mostres i, per tant, cada error allarga la seva durada.

2.2 Model de Càmera Estenopecica

El paper principal de les càmeres en el nostre sistema és el d'informar-nos de la posició dels ulls del participant durant tota la prova. Per calcular la posició dels ulls, analitzarem les imatges capturades per les càmeres un cop acabada la prova.

Així doncs, ens cal un mètode que ens permeti estimar les coordenades d'un punt del nostre escenari tridimensional a partir de les coordenades d'un punt en una imatge capturada per les nostres càmeres. Aquesta correspondència entre l'espai 3D i la seva projecció al pla imatge, depèn del procés de formació d'imatges de les càmeres.

El model més senzill que descriu el procés de formació d'imatges d'una càmera és el model de càmera estenopecica (*pinhole camera model*) on l'obertura de la càmera, l'orifici per on entra la llum al dispositiu, es descriu com un únic punt, l'estenop. En aquesta simplificació, l'estenop juga el paper de centre de projecció d'una projecció central. Així doncs, els punts de l'espai que queden dintre del camp de visió de la càmera seran projectats, mitjançant feixos de llum que passen per l'estenop, cap a l'interior de la càmera, on es formarà la imatge.

Aquesta imatge es correspondrà amb la dels punts projectats en un pla, el pla imatge, a una certa distància fixada del centre de projecció.

Per tant, si prenem el centre de projecció com a origen del nostre sistema de coordenades de l'espai vectorial Euclidià \mathbb{R}^3 , i definim un dels seus eixos com la recta perpendicular al pla imatge, que ens indica cap a on està enfocada la càmera, obtenim el model:

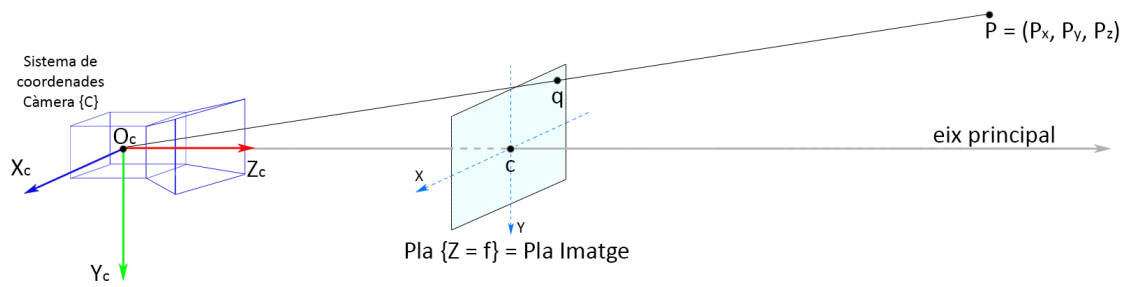


Figura 1: Model de càmera estenopeica

On:

- O_C és el centre de projecció o centre òptic.
- P és un punt qualsevol de l'escenari 3D amb coordenades (P_x, P_y, P_z) en $\{C\}$.
- q és la projecció del punt P en el pla imatge
- c és el punt principal, on l'eix principal (o eix òptic) interseca amb el pla imatge.
- f és la distància focal.

El model descrit simplifica molts aspectes d'una càmera. No només considera l'apertura de la càmera com un únic punt, sinó que també ignora els efectes d'enfocament o el paper de les lents i el seu gruix, que concentren o dispersen els raigs de llum i poden produir una certa distorsió en les imatges o la difuminació d'objectes no enfocats.

El principal problema d'aquest model és el de no tenir en compte la distorsió radial provocada per l'efecte de les lents de la càmera, ja que aquesta distorsió es va fent més gran en els punts més allunyats del centre de la càmera, deformant-la i fent que s'allunyi de la representació que donaria el model ideal. A la pràctica, entre menor sigui la distància focal (i qualitat) de les lents, major serà la distorsió de les imatges.

Per solucionar aquest i altres errors, generalitzarem el model de la càmera estenopeica fins a arribar a un de prou precís per a utilitzar-lo en aquest projecte. Cal tenir en compte que, tot i que el model considerat prové del model general de càmera projectiva, en cap cas arribarem a treballar amb les eines pròpies de la geometria projectiva com els espais projectius, rectes a l'infinit o perspectives, sinó que treballarem sempre dins del marc de la geometria Euclidiana entre espais vectorials de \mathbb{R}^3 o \mathbb{R}^2 .

Així doncs, simplifiquem la projecció de l'espai tridimensional en el pla imatge com a una aplicació lineal entre espais vectorials Euclidians per tal de poder expressar matricialment aquesta transformació, a partir de la matriu associada a l'aplicació, i treballar amb les eines típiques de l'àlgebra lineal. D'aquesta manera, els càlculs involucrats es simplifiquen a problemes matricials on, per tal de treballar l'espai euclidià com a subconjunt de l'espai projectiu, utilitzarem sempre coordenades homogènies on l'última coordenada representarà els punts a l'infinit i, a la pràctica, serà sempre 1.

Tornant al nostre model inicial (Figura 1), podem obtenir les coordenades del punt q

projectat al pla imatge a partir de les coordenades del punt P i la distància focal f. Considerant els eixos en vermell en la següent figura:

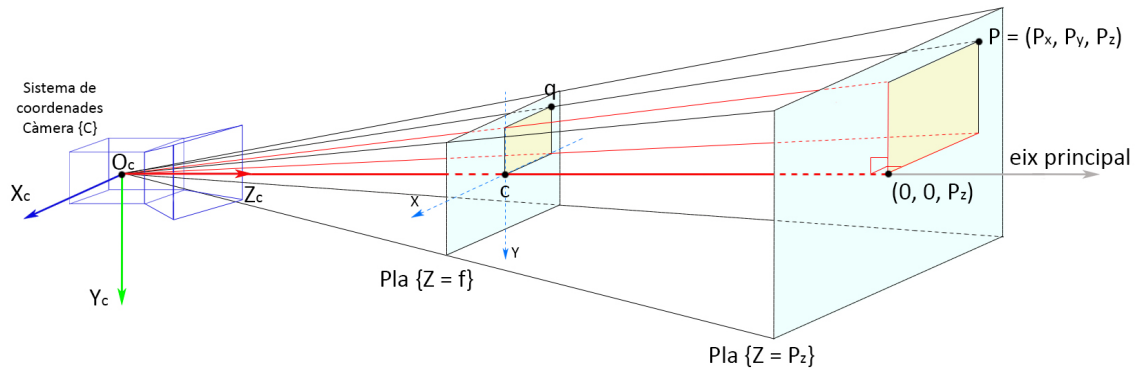


Figura 2: Projecció central del punt P

Observem que formen triangles rectangles semblants (vegeu la Figura 3), és a dir, amb costats proporcionals i mateixos angles interns i, per tant, utilitzant aquesta proporcionalitat (i que $P_z \neq 0$ i $f \neq 0$) podem obtenir les coordenades (q_x, q_y) de q:

$$\frac{q_x}{f} = \frac{P_x}{P_z} \implies q_x = f \frac{P_x}{P_z}$$

$$\frac{q_y}{f} = \frac{P_y}{P_z} \implies q_y = f \frac{P_y}{P_z}$$

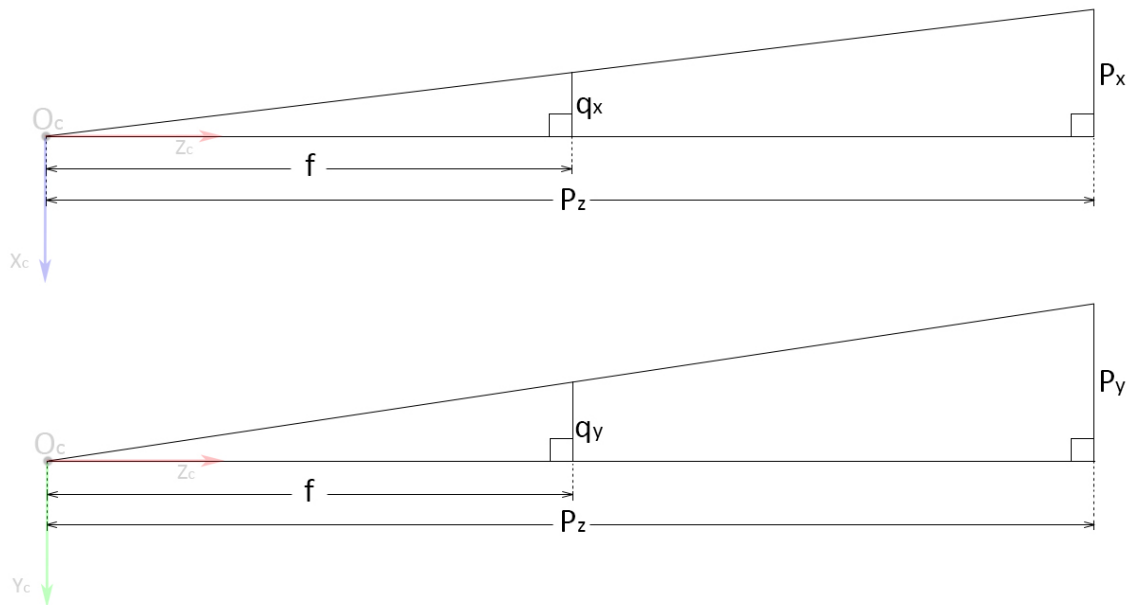


Figura 3: Triangles rectangles de la projecció central

Per tant, en el sistema de coordenades de la Càmera, el punt q queda representat per:

$$q = \left(f \frac{P_x}{P_z}, f \frac{P_y}{P_z}, f \right)$$

I, com sabem que tot punt es projecta al pla $\{Z = f\}$, podem prescindir de l'última coordenada i representar la projecció central al pla imatge mitjançant una aplicació lineal g_1 entre l'espai vectorial Euclidià \mathbb{R}^3 i l'espai vectorial Euclidià \mathbb{R}^2 definida per:

$$(X, Y, Z) \mapsto g_1(X, Y, Z) = \left(f \frac{X}{Z}, f \frac{Y}{Z}\right)$$

O, utilitzant coordenades homogènies:

$$g_1 : \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \left(\begin{array}{ccc|c} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{array} \right) \cdot \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

On, per alleugerir la notació, utilitzarem:

$$K_1 = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad i \quad [K_1 | 0] = \left(\begin{array}{ccc|c} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{array} \right)$$

Per tant, de forma compacta, donat un punt P qualsevol de l'espai amb coordenades homogènies $(P_x, P_y, P_z, 1)^T$ en $\{C\}$, obtenim el seu punt q projectat al pla imatge amb l'expressió:

$$q = [K_1 | 0] \cdot P$$

2.2.1 Generalització del model: Desviació del punt principal

En el model bàsic de càmera estenopèica hem establert una correspondència entre els punts del pla imatge i els de l'espai tridimensional del món real. Aquesta correspondència, representada a través d'un producte matricial, considera que la càmera realitza una projecció central perfectament lineal, de manera que l'eix òptic de la càmera interseca el centre exacte del pla imatge en l'anomenat punt principal (vegeu Figura 1).

Podem observar doncs, que els punts del pla imatge tenen com a centre de coordenades el punt principal, que coincideix amb el centre de la imatge. A la pràctica, veurem que el punt principal acostuma a estar lleugerament desviat del centre de la imatge i que la seva posició depèn de les característiques de cada càmera. A més, les imatges digitals que extraurem acostumen a estar expressades en píxels i a tenir l'origen de coordenades (el punt (0,0)) en la cantonada superior esquerra de la imatge, no en el seu centre.

Així doncs, considerem que les coordenades (\hat{c}_x, \hat{c}_y) del punt principal c expressades en mm (més endavant, quan les expressem en píxels, les denotarem per (c_x, c_y)), de la mateixa manera que la distància focal f, són paràmetres interns de la càmera que haurem d'estimar per tal de trobar la correspondència entre els punts de la imatge i els del món real. A més, per ajustar-nos a la notació habitual, mourem el nostre punt d'origen del pla mitjançant una translació de vector (\hat{c}_x, \hat{c}_y) per tal de situar-lo a la cantonada superior esquerra de la imatge digital obtinguda. Per tant, afegint la translació esmentada, obtenim l'aplicació $g_2 : \mathbb{R}^3 \mapsto \mathbb{R}^2$ definida per:

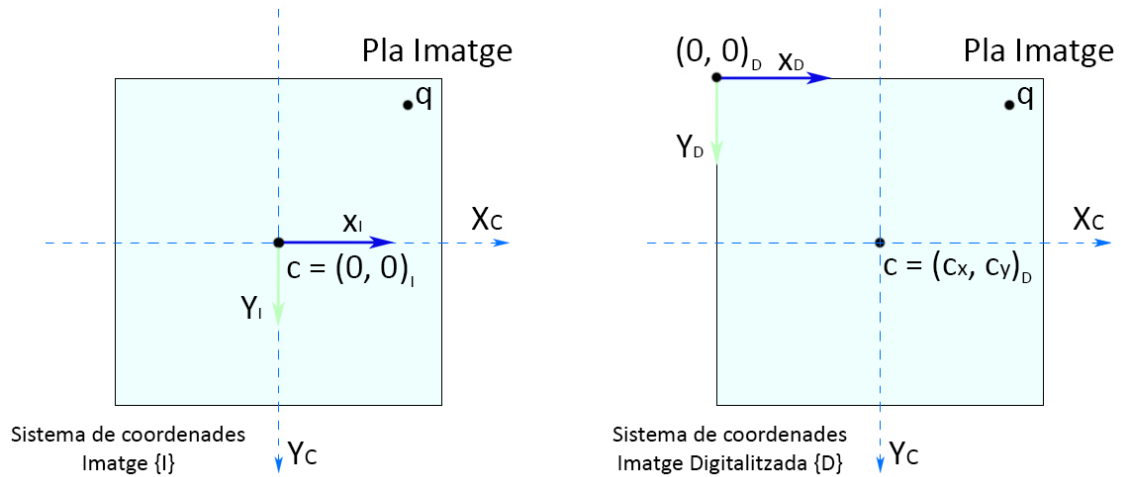


Figura 4: Sistemes de coordenades del pla imatge

$$g_2 : \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX + \hat{c}_x Z \\ fY + \hat{c}_y Z \\ Z \\ 1 \end{pmatrix} = \left(\begin{array}{ccc|c} f & 0 & \hat{c}_x & 0 \\ 0 & f & \hat{c}_y & 0 \\ 0 & 0 & 1 & 0 \end{array} \right) \cdot \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

on:

$$K_2 = \begin{pmatrix} 1 & 0 & \hat{c}_x \\ 0 & 1 & \hat{c}_y \\ 0 & 0 & 1 \end{pmatrix} \quad i \quad K_2 K_1 = \begin{pmatrix} f & 0 & \hat{c}_x \\ 0 & f & \hat{c}_y \\ 0 & 0 & 1 \end{pmatrix}$$

I, com abans, de forma compacta ens queda l'expressió:

$$q = [K_2 K_1 \mid 0] \cdot P$$

Observem que les matrius K_1 i K_2 defineixen aplicacions lineals conegudes. Per una banda la matriu K_1 , que representa la projecció central, és la matriu associada d'una homotècia de centre $O_C = (0, 0, 0)$ i raó f que defineix una relació de semblança entre l'objecte original i el transformat, on cada punt de l'objecte original i el seu punt imatge estan alineats amb el centre de l'homotècia.

Per aquest motiu, hem pogut trobar les coordenades del punt projectat utilitzant semblança de triangles i, intuïtivament, podem entendre aquesta transformació com un redimensionament d'un objecte (ampliació o reducció) per un factor f . De fet, en anglès i en coordenades homogènies, a una matriu amb la forma de K_1 se l'anomena *2D Scale Matrix*, és a dir, matriu de redimensionament 2D.

Per altra banda, com ja havíem comentat, la matriu K_2 defineix una translació de vector (\hat{c}_x, \hat{c}_y) i s'encarrega de moure el punt origen del pla imatge, per tant, l'acció de K_2 es pot interpretar com un canvi de coordenades del pla imatge.

2.2.2 Generalització del model: Píxels no quadrats

El nostre model considera que les coordenades de la imatge són Euclidianes, comportant-se d'igual manera en els seus eixos. En el cas d'algunes càmeres d'ígitals, com aquelles que funcionen amb sensors CCD o CMOS, on les coordenades del pla imatge seran mesurades en píxels, existeix la possibilitat de que els píxels que conformen la imatge no siguin quadrats i, en conseqüència, durant la captura de la imatge s'introdueixin escalats diferents en els dos eixos que conformen l'amplada i l'alçada de la imatge digital.

Si el nombre de píxels per unitat de distància (mil·límetres en el nostre cas) en el pla imatge és m_x per l'eix x i m_y per l'eix y, llavors la transformació de coordenades de l'espai tridimensional (en mm) a coordenades del pla imatge (en píxels) ve donada per l'aplicació $g : \mathbb{R}^3 \mapsto \mathbb{R}^2$ definida per:

$$g : \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} m_x f X + m_x \hat{c}_x Z \\ m_y f Y + m_y \hat{c}_y Z \\ Z \end{pmatrix} = \left(\begin{array}{ccc|c} m_x f & 0 & m_x \hat{c}_x & 0 \\ 0 & m_y f & m_y \hat{c}_y & 0 \\ 0 & 0 & 1 & 0 \end{array} \right) \cdot \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

on:

$$K_3 = \begin{pmatrix} m_x & 0 & 0 \\ 0 & m_y & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad i \quad K_3 K_2 K_1 = \begin{pmatrix} m_x f & 0 & m_x \hat{c}_x \\ 0 & m_y f & m_y \hat{c}_y \\ 0 & 0 & 1 \end{pmatrix}$$

i, utilitzant la notació $K_3 K_2 K_1 = K$, obtenim l'expressió:

$$q = [K \mid 0] \cdot P$$

Observem que, de nou, la matriu K_3 afegida és una matriu d'escalada, però aquest cop utilitza dos factors d'escalada diferents, un per a cada coordenada del pla imatge.

Anomenarem matriu intrínseca de la Càmera a la matriu K obtinguda i la representarem utilitzant la notació següent:

$$K = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \quad o \quad K = \begin{pmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}$$

On:

- $f_x = m_x f$ i $f_y = m_y f$ són les distàncies focals dels eixos X i Y expressades en píxels.
- $c = (c_x, c_y) = (m_x \hat{c}_x, m_y \hat{c}_y)$ és el punt principal expressat en píxels.
- s és un factor de biaix anomenat *skew*.

A la pràctica, f_x i f_y tindran valors molt similars i el punt c serà proper al centre de la imatge. Tot i això, aquestes variacions són bastant significatives i caldrà tenir-les en consideració. En canvi, a la majoria de càmeres podem menysprear el valor s i considerar-lo nul, ja que només apareix quan els angles interns dels píxels no són de 90° .

2.2.3 Distorsió

En definitiva, l'aplicació g amb matriu associada K representarà la nostra projecció. Aquesta transformació és una aplicació lineal, composició de les 3 aplicacions lineals definides per les matrius K_3 , K_2 i K_1 . En particular aquesta simplificació, al ser considerada perfectament lineal, conservarà angles, distàncies i formes. És a dir, les rectes de l'escenari 3D quedaran representades al pla imatge, sota aquesta model, com a rectes amb els mateixos angles i distàncies proporcionals a les reals.

Ara bé, sabem que les imatges obtingudes per la càmera poden representar deformades les rectes de l'escenari 3D, allunyant la imatge obtinguda per la nostra càmera de la imatge que obtindríem seguint el nostre model lineal. Aquesta diferència es deu, principalment, a la distorsió radial provocada per les lents de la càmera.

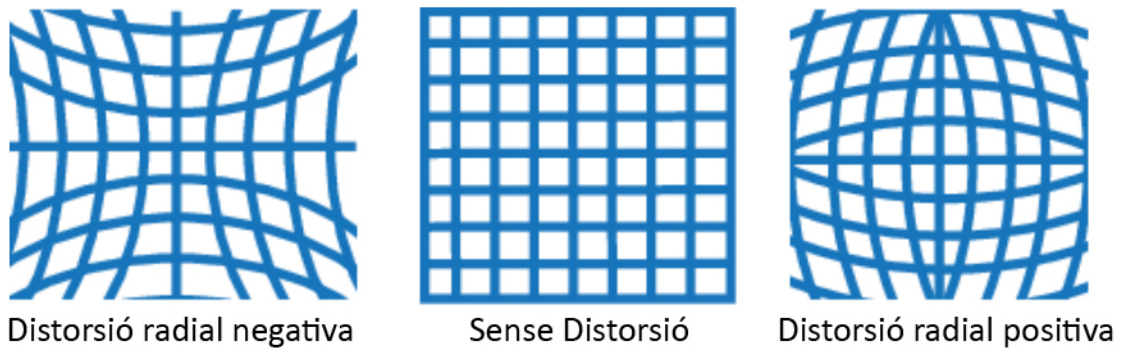


Figura 5: Efectes de la distorsió radial [10]

En comptes d'afegir la distorsió radial al nostre model, corregirem la distorsió de les imatges capturades per les càmeres per tal d'obtenir imatges que s'ajusten al model lineal que utilitzarem en aquest treball.



Figura 6: Exemple de correcció de la distorsió radial d'una imatge

L'efecte de la distorsió radial es pot representar mitjançant una funció $L(r)$ que depèn de la distància r d'un punt respecte al centre de distorsió. Normalment, es considera que el centre de distorsió i el punt principal $c = (c_x, c_y)$ coincideixen. Aquesta distorsió pot expressar-se mitjançant una successió de Taylor:

$$L(r) = 1 + k_1 r + k_2 r^2 + k_3 r^3 + k_4 r^4 \dots$$

On els k_j són els coeficients de correcció radial que volem estimar per tal de corregir la

distorsió radial. En el pla imatge, només tindrem dos coeficients: k_1 i k_2 i, per tant, tenim:

$$\begin{aligned}x &= \hat{x} + \hat{x}L(r)(x - c_x) \\y &= \hat{y} + \hat{y}L(r)(y - c_y)\end{aligned}$$

On (x,y) representa les coordenades ideals (sense distorsió) d'un punt qualsevol al pla imatge en píxels i (\hat{x}, \hat{y}) representa les coordenades del mateix punt, però amb distorsió radial, és a dir, les coordenades del punt tal com apareix representat a la imatge capturada en píxels.

2.2.4 Paràmetres Extrínsecs

Fins ara, hem vist com a partir d'una imatge extreta per una de les nostres càmeres, podem reconstruir l'escena 3D i expressar la posició de punts a l'espai en mil·límetres i respecte a un sistema de coordenades Càmera que té l'origen en el centre de la lent d'aquesta càmera.

Per tant, tenim definida la relació de correspondència entre una imatge i la càmera que l'ha capturat. Tot i això, l'ús de diverses càmeres ens genera la necessitat de trobar un mètode per poder expressar el mateix punt de l'espai en qualsevol dels sistemes de coordenades.

De nou, com estem treballant en l'espai vectorial Euclidià \mathbb{R}^3 , podem expressar els canvis de coordenades de forma matricial, on només intervé una matriu 4x4, anomenada matriu extrínseca, formada per una matriu de rotació R i un vector de translació t , que són els paràmetres extrínsecs de la càmera.

De forma intuïtiva, podem dir que dos sistemes de coordenades a l'espai estan relacionats entre si per una translació t , que desplaça un dels orígens de coordenades cap a l'altre, i una matriu de rotació que rota els eixos d'un sistema de coordenades fins a fer-los coincidir amb el de l'altre.

Així doncs, el canvi de coordenades en un sistema $\{C_1\}$ a coordenades en un sistema $\{C_2\}$ d'un punt $P = (P_x, P_y, P_z, 1)^T$, ve donat pel producte matricial següent:

$$\begin{pmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_1 \\ r_{2,1} & r_{2,2} & r_{2,3} & t_2 \\ r_{3,1} & r_{3,2} & r_{3,3} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} P_x \\ P_y \\ P_z \\ 1 \end{pmatrix} = \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} \cdot P$$

On:

$$R = \begin{pmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{pmatrix} \quad i \quad t = (t_1, t_2, t_3)$$

Observem que el vector de translació t tindrà per components les coordenades del punt origen del sistema $\{C_2\}$ expressades en el sistema $\{C_1\}$, ja que si agafem $P = (0, 0, 0, 1)$, obtenim com a resultat el punt $(t_1, t_2, t_3, 1)$.

Per altra banda, per entendre com es modela la rotació dels tres eixos a partir d'una

única matriu 3x3, hem de recordar que les matrius que exemplifiquen les tres rotacions bàsiques d'angle θ sobre els eixos X, Y i Z són, respectivament:

$$R_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix}, R_y(\theta) = \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix} \text{ i } R_z(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Per tant, la nostra matriu R modelarà un gir d'angle α respecte a l'eix X, un gir d'angle β respecte a l'eix Y i un gir d'angle γ respecte a l'eix Z mitjançant el producte de matrius de rotació bàsiques següent:

$$R = R_x(\alpha)R_y(\beta)R_z(\gamma) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{pmatrix} \begin{pmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

On les rotacions bàsiques $R_x(\alpha)$, $R_y(\beta)$ i $R_z(\gamma)$ es produeixen cap a l'esquerra quan l'eix al voltant del qual estan definides apunta cap a l'observador i el sistema de coordenades és de mà dreta.

Finalment, utilitzant les notacions de les anteriors subseccions, el punt q del pla imatge queda representat per:

$$q = [K \mid 0] \cdot \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} \cdot P$$

2.3 Calibratge de Càmeres

2.3.1 Calibratge Intrínsec

Hem vist que la correspondència entre els punts del nostre escenari 3D i els punts de les imatges obtingudes per les càmeres només depèn, en el nostre model, dels paràmetres f_x , f_y , c_x , c_y i la distorsió radial.

Aquests són els anomenats paràmetres intrínsecs de la càmera, ja que depenen de les característiques internes de cada càmera, i són els paràmetres que necessitem conèixer per tal de poder saber la posició d'un punt a l'escenari 3D a partir de la posició d'aquest punt a la imatge capturada per una de les càmeres.

Per aquest motiu, abans d'iniciar una sessió de gravació, caldrà realitzar un calibratge intrínsec de cada càmera, on s'estimaran els paràmetres intrínsecs. A més, després del calibratge, haurem d'evitar reajustar l'objectiu o el zoom de la càmera, ja que modificaríem els paràmetres intrínsecs de la càmera i, per tant, seria necessari re calibrar.

Existeixen diversos mètodes que permeten estimar els paràmetres intrínsecs d'una càmera. En general, molts d'ells basen el seu funcionament en la captura d'imatges d'un objecte o patró de mides conegudes amb exactitud sota diferents punts de vista. Podem diferenciar aquest tipus de tècniques en funció de les dimensions del patró utilitzat, ja que es poden utilitzar objectes tridimensionals, objectes plans, rectes o, fins i tot, treballar únicament amb punts.

A la pràctica, l'ús de patrons plans és una de les opcions que s'utilitzen més freqüentment per ser la més simple d'aplicar. Es poden utilitzar diferents tipus de patrons, com per exemple patrons en forma de taulell d'escacs, patrons amb cercles o patrons ChArUco (vegeu Figura 7).

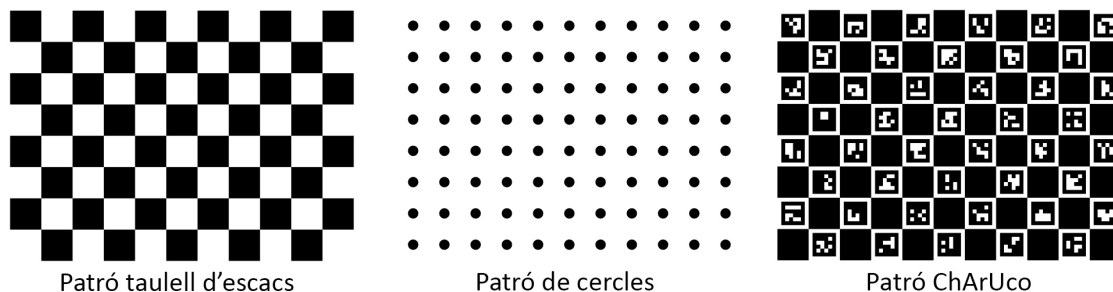


Figura 7: Tipus comuns de patrons bidimensionals per calibrar

Podem elaborar nosaltres mateixos un patró escollint un dels seus models típics i les seves mesures i imprimir-lo i adherir-lo a una superfície rígida i plana que puguem moure amb facilitat.

Un cop tinguem el nostre patró, el mourem davant la càmera que vulguem calibrar, capturant imatges d'ell en diferents angles i orientacions de manera que, el conjunt total de les imatges capturades del patró, cobreixi tot el camp de visió de la càmera. La distància a la qual situarem el patró, respecte a la càmera, serà aproximadament la mateixa a la qual estarà la cara del participant durant la prova i, en totes les imatges que fem servir per calibrar, hi haurà d'aparèixer el patró complet (vegeu Figura 8).

Haurem de detectar els punts característics del patró en les imatges. Per exemple, en el cas del taulell d'escacs, aquests punts són les interseccions de 4 caselles i són fàcilment detectables. La detecció i selecció dels punts pot ser manual o bé utilitzant mètodes automàtics (que detecten intersecció de rectes, per exemple). Un cop tinguem marcats els punts, es refinirà la seva posició fins a obtenir una precisió en l'estimació amb un error inferior a 1 píxel.

Recordem que el model de càmera estenopeica mostrava la relació entre els punts de la imatge i els punts de l'entorn 3D. Així doncs, seguint aquest model i a partir de la posició dels punts característics del patró a les imatges i les mesures del patró en el món real, podrem estimar els paràmetres intrínsecs.

Un dels algorismes més utilitzats per resoldre aquest problema [6] assumeix, sense pèrdua de generalitat, que el patró utilitzat en cada imatge pertany al pla $\{Z = 0\}$ del sistema de coordenades Món que defineix l'entorn 3D. Per tant, si abans teníem:

$$q = [K \mid 0] \cdot \begin{pmatrix} R & t \\ 0 & 1 \end{pmatrix} \cdot P$$

O, equivalentment, utilitzant la notació $R = (r_1, r_2, r_3)$ i $P = (X, Y, Z, 1)^T$:

$$q = [K \mid 0] \cdot \begin{pmatrix} r_1 & r_2 & r_3 & t \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$



Figura 8: Imatges del procés de calibratge mitjançant un patró en forma de taulell d'escacs

Llavors, amb $Z = 0$, podem simplificar-ho a:

$$q = K \cdot \begin{pmatrix} r_1 & r_2 & t \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = H \cdot \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}$$

On la matriu H defineix una homografia amb paràmetres que podem estimar a partir dels valors de q i P obtinguts a partir dels punts del patró.

Per altra banda, si denotem les coordenades d'un punt al pla, en píxels, per $q = (\hat{x}, \hat{y})$. Llavors, les coordenades del punt q en el pla imatge ideal (sense distorsió) i en píxels, venen donades per l'expressió:

$$\begin{aligned} x &= \hat{x} + (\hat{x} - c_x)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2] \\ y &= \hat{y} + (\hat{y} - c_y)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2] \end{aligned}$$

On k_1 i k_2 són els coeficients de distorsió radial que poden ser estimats a partir de la fórmula donada i els punts característics del patró utilitzat.

2.3.2 Calibratge Càmera-Càmera

Anomenem Calibratge extrínsec al procés mitjançant el qual estimem els paràmetres extrínsecs, matriu de rotació R i vector translació t , que ens permeten canviar d'un sistema de coordenades de l'espai vectorial Euclidià \mathbb{R}^3 a un altre sistema de coordenades en el mateix espai.

En el cas del calibratge Càmera-Càmera, ens interessa estimar els paràmetres extrínsecs que ens permeten relacionar dos sistemes de coordenades definits per dues càmeres diferents. Cal recordar, que aquests paràmetres ens donen la rotació d'una càmera respecte a l'altre i la distància de la lent d'una càmera respecte a la de l'altre càmera. Per aquest motiu, és imprescindible que durant el procés de calibratge no es mogui cap de les dues càmeres i que es mantinguin en la mateixa posició durant la prova.

El procés de Calibratge extrínsec, en aquest cas, és molt similar a l'intrínsec. A la pràctica, l'única variació serà que haurem de treure imatges d'un patró des de dues càmeres diferents alhora, de manera que el patró sigui visible en tot moment per les dues càmeres. Així doncs, treballarem amb parelles d'imatges del mateix patró, en la mateixa posició, extretes per les dues càmeres que volem calibrar.

Inicialment, tractarem les imatges de cada càmera de la mateixa manera que en el calibratge intrínsec [6]. Això ens permetrà estimar la matriu de rotació i el vector de translació que relaciona el sistema de coordenades d'una càmera amb el sistema de coordenades definit pel patró, on consideràvem que el patró pertanyia al pla $\{Z = 0\}$.

Per tant, obtindrem una relació entre el sistema de coordenades de cada càmera i el sistema de coordenades definit pel patró, que ens permetran relacionar els sistemes de coordenades de cada càmera entre si.

Recordem que, del calibratge intrínsec doble, obtenim:

$$q_1 = [K_1 \mid 0] \begin{pmatrix} R_1 & t_1 \\ 0 & 1 \end{pmatrix} P \quad i \quad q_2 = [K_2 \mid 0] \begin{pmatrix} R_2 & t_2 \\ 0 & 1 \end{pmatrix} P$$

On P és un punt del taulell en el sistema de coordenades definit pel patró; q_i és un punt del taulell en el sistema de coordenades definit per la càmera C_i ; i K_i , R_i i t_i són els paràmetres de la càmera C_i per a $i=1,2$. Per tant, això ens permet relacionar q_1 amb q_2 , per a cada punt del patró, i estimar la R i t que relacionen les càmeres.

Com a observació final, cal destacar que el mètode de calibratge Càmera-Càmera permet estimar els paràmetres intrínsecs de cada càmera. Tot i això, és recomanable fer un calibratge intrínsec individual per a cada càmera, ja que la necessitat de què el patró sigui totalment visible en dues càmeres alhora pot limitar el tipus de posicions en les que col·locar el taulell i oferir un pitjor calibratge intrínsec. De fet, normalment es realitza un millor calibratge extrínsec si treballem amb paràmetres intrínsecs prèviament estimats durant la calibració intrínseca. Per aquesta raó, el calibratge intrínsec de les càmeres s'acostuma a fer abans de qualsevol altre.

2.3.3 Calibratge Càmera-Pantalla

En el cas del calibratge Càmera-Pantalla, volem trobar la matriu de rotació R i el vector de translació t que ens permeten passar d'un sistema de coordenades definit per una càmera a un sistema de coordenades definit en el pla descrit per la pantalla on es mostrarà l'estímul visual.

Aquest calibratge ens permetrà saber on es troba qualsevol punt de la pantalla respecte al centre de la lent d'una de les nostres càmeres. Per tant, podrem expressar el punt final del vector mirada que volem capturar en qualsevol dels nostres sistemes de coordenades de l'espai tridimensional.

De nou, farem servir un patró amb caselles d'escacs que mostrarem per pantalla. D'aquesta manera, a partir de diverses imatges del patró extretes per una mateixa càmera, podrem estimar la posició del patró, i de la pantalla que el mostra, respecte la càmera.

El problema principal es que, en aquest cas, la pantalla (i per tant el patró) no és visible per cap de les càmeres, ja que aquestes es posicionaran al voltant de la pantalla i enfocaran a la persona que mira cap a ella.

Existeixen varies solucions proposades per a calibrar els paràmetres extrínsecs d'una càmera respecte d'un objecte no visible per ella. La majoria d'ells, posicionen miralls davant la càmera per fer que el reflex de l'objecte entri en el camp de visió de la càmera i calibrar l'objecte a partir d'ell [7, 8, 9].

En el nostre cas, utilitzarem un mètode que permet calcular la posició a l'espai dels punts d'un patró format per caselles d'escacs, respecte al sistema de coordenades definit per la càmera, a partir d'imatges capturades per la càmera on aparegui el reflex del patró en un mirall pla col·locat en més de dues posicions diferents [7].

Aquest mètode, considerarà que el mirall utilitzat no distorsiona la imatge i basarà l'algorisme utilitzat en una condició d'ortogonalitat que haurà de complir cada un dels reflexos del mirall de les diferents imatges. Considerem el següent model:

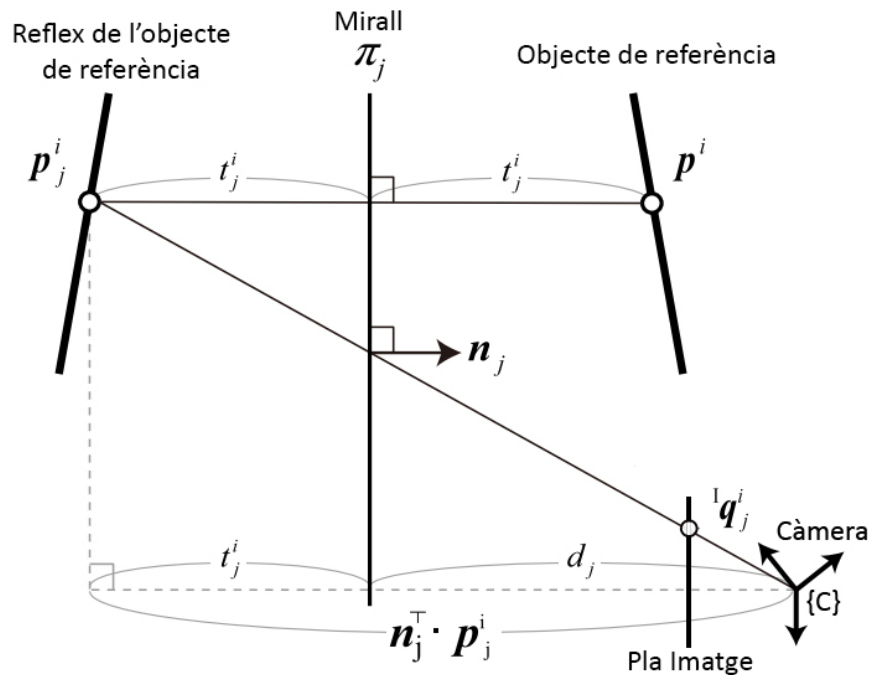


Figura 9: Model del calibratge Càmera-Pantalla [7]

On:

- N_p és el nombre de punts de referència de l'objecte.
- N_m és el nombre de posicions diferents del mirall.
- π_j és el mirall j-èsim, per a $j=1, \dots, N_m$, que ve determinat pel seu vector normal n_j i la seva distància d_j respecte l'origen de $\{C\}$.
- ${}^X p^i = (x_i, y_i, z_i)^T$ és el punt i-èsim de referència, per a $i=1, \dots, N_p$, en el sistema de coordenades $\{X\}$ definit per l'objecte de referència.
- ${}^C p_j^i$ és el reflex del punt p^i en el mirall π_j en $\{C\}$.
- ${}^I q_j^i$ és la projecció del punt ${}^C p_j^i$ en el sistema de coordenades $\{I\}$ definit pel pla imatge.
- t_j^i és la distància entre π_j i p^i , que és igual a la distància entre π_j i p_j^i .

Utilitzant aquesta notació, el nostre problema es redueix a estimar la matriu de rotació R i el vector de translació T a partir dels punts ${}^I q_j^i$.

Sabem que el canvi de coordenades de $\{C\}$ a $\{X\}$ d'un punt s'expressa com a:

$${}^C p^i = R \cdot {}^X p^i + T$$

Per altra banda, a partir del model anterior, observem que:

$$t_j^i + d_j = -n_j^T \cdot {}^C p_j^i \implies t_j^i = -n_j^T \cdot {}^C p_j^i - d_j$$

I, com la relació entre un punt i el seu reflex ve donada per:

$${}^C p^i = 2 \cdot t_j^i \cdot n_j + {}^C p_j^i$$

Obtenim la Transformació de Householder:

$${}^C p^i = 2 \cdot t_j^i \cdot n_j + {}^C p_j^i = -2(n_j^T \cdot {}^C p_j^i + d_j)n_j + {}^C p_j^i$$

Per tant, l'equació fonamental del model serà:

$${}^C p^i = R \cdot {}^X p^i + T = -2(n_j^T \cdot {}^C p_j^i + d_j)n_j + {}^C p_j^i$$

Per altra banda, donat un punt de referència p^i i els seus reflexos p_j^i i $p_{j'}^i$, en dos miralls diferents π_j i $\pi_{j'}$. Considerarem que el vector intersecció $m_{jj'} = n_j \times n_{j'}$ pertanyent a la recta on intersequen els plans definits pels corresponents miralls satisfà la condició d'ortogonalitat següent:

$$(p_j - p_{j'})^T \cdot m_{jj'} = 0$$

Finalment, l'algorisme per estimar R i T consta dels següents passos:

- **Pas 1:** Càlcul dels punts ${}^C p^i$ a partir dels punts ${}^I q_j^i$ resolent el problema PnP (*Perspective-n-Points*) per a $n = N_p$.

- **Pas 2:** Càlcul dels vectors $m_{jj'}$ de cada parella de miralls a partir dels punts ${}^C p^i$, la seva relació amb els seus reflexos i la condició d'ortogonalitat.
- **Pas 3:** Càlcul dels vectors normals n_j a partir dels vectors $m_{jj'}$ utilitzant que $n_j^T \times m_{jj'} = n_{j'}^T \times m_{jj'} = 0$.
- **Pas 4:** Càlcul de R i T mitjançant la resolució d'un sistema d'equacions obtingut a partir de l'equació fonamental del model $R \cdot {}^X p^i + T = -2(n_j^T \cdot {}^C p_j^i + d_j)n_j + p_j^i$.
- **Pas 5:** Optimització no lineal per millorar l'estimació de R i T.

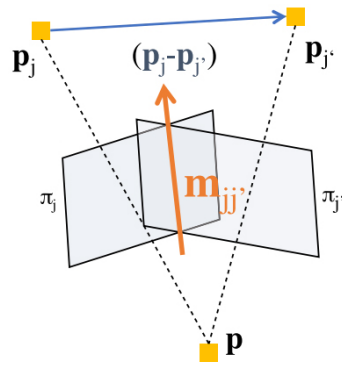


Figura 10: Condició d'ortogonalitat dels miralls [7]

2.4 Geometria Ocular

Per tal d'estimar les coordenades del punt origen del vector mirada en el nostre sistema de coordenades Càmera, considerarem el següent model simplificat de la geometria de l'ull humà [13, 14]:

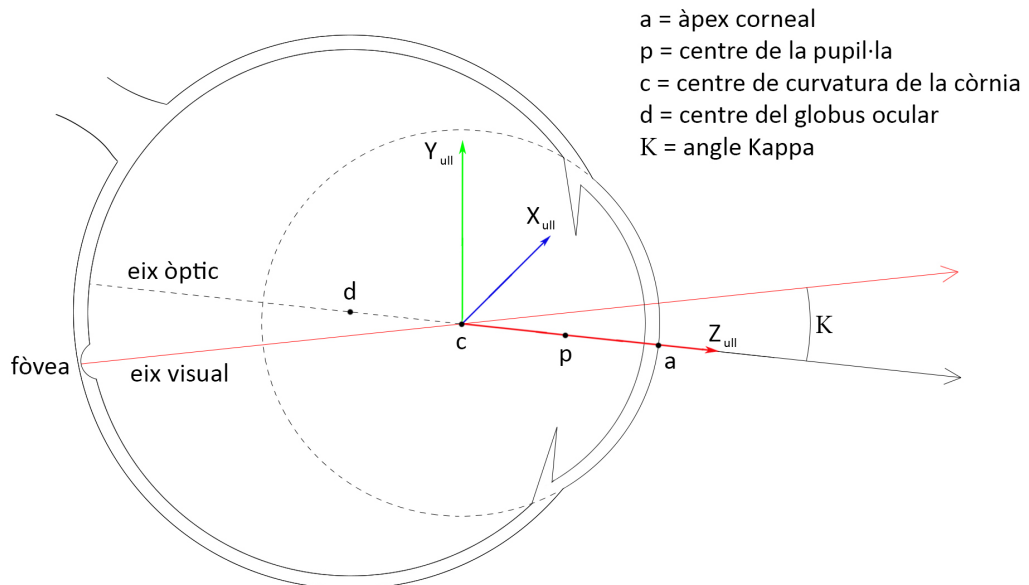


Figura 11: Model de l'estructura ocular

Es defineix l'eix òptic com a aquell eix que passa pel centre pupil·lar i el centre de curvatura de la còrnia. En el nostre model, considerarem que el centre del globus ocular també pertany a aquest eix, ja que la majoria de sistemes òptics presenten simetria rotacional al voltant de l'eix òptic i, per fer que el nostre model també la tingui, cal que l'eix òptic passi pel centre de curvatura de la còrnia, que és una superfície refractora que fa el paper de lent, i el centre del globus ocular.

De fet, l'eix òptic d'un sistema òptic és la recta al voltant de la qual es propaga la llum cap al sistema. En el model de la càmera estenopecica, entra la llum al sistema a través d'un únic punt, l'estenop o centre òptic, que pertanyia a l'eix visual. En el cas de l'ull, l'entrada de llum està regulada per la pupil·la i, per aquesta raó, el seu centre pertany a l'eix òptic.

Tot i això, el punt de la retina on es forma la imatge del punt de fixació, anomenat fòvea, no pertany a l'eix òptic. La fòvea es situa en la retina a aproximadament 1,25mm del punt on interseca l'eix òptic i la retina [12].

Així doncs, anomenarem eix visual a aquell eix on resideix el vector mirada i el definirem com a l'eix que passa per la fòvea i interseca a l'eix òptic en el centre de curvatura de la còrnia.

Per tant, el punt on interseca l'eix òptic i l'eix visual serà el punt d'origen del vector mirada que volem calcular.

Per calcular el centre de la curvatura de la còrnia, utilitzarem un sistema de detecció facial que modela la forma, posició i orientació de la cara. Normalment, els models utilitzats en visió per computador estimen la posició a l'espai de punts representatius del contorn de la cara, celles, nas, boca i ulls. Aquests punts s'anomenen *Facial Landmarks* i, tot i que hi ha diversos models per representar l'estructura facial mitjançant aquests punts, un dels més comuns és el que es basa en 68 landmarks (vegeu la Figura).

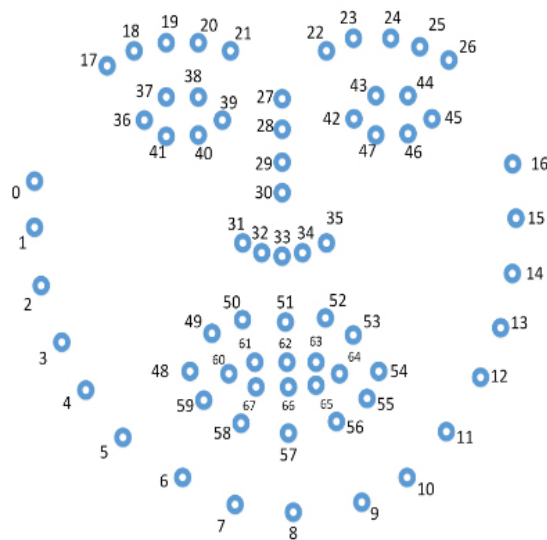


Figura 12: Model Facial format per 68 *Landmarks* [15]

3 Avaluació

3.1 Detalls d'implementació

Els elements principals del nostre sistema multicàmera són:

- 1 Axis M1124 Fixed Network Camera (Càmera Axis)
- 2 Revotech HD Video Security Camera 1712, 1920x1080P (ZC1 i ZC2)
- 1 Pantalla Philips de 21,5", 1920x1080p FullHD

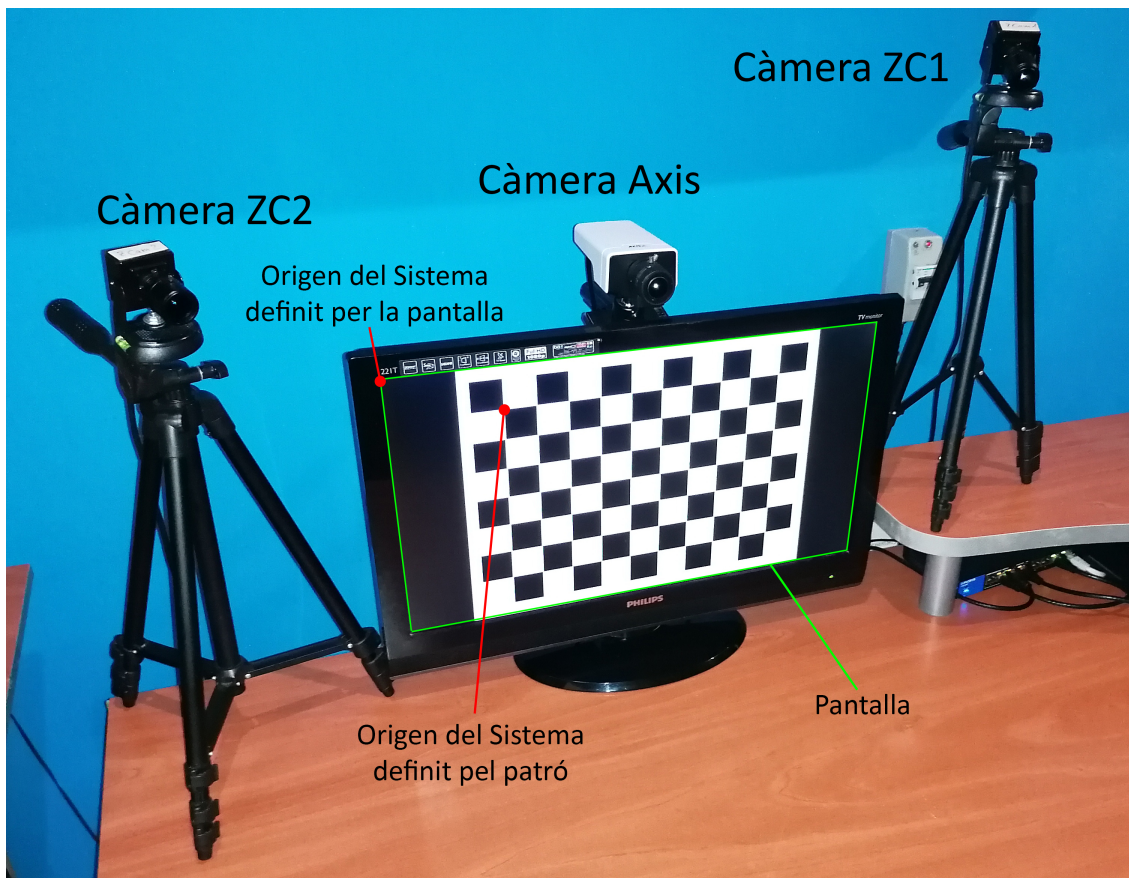


Figura 13: Sistema Multicàmera utilitzat durant la prova

A la figura anterior es mostra la posició en la qual van ser fixades les càmeres i la pantalla durant els calibratges i la prova final. Per tal d'obtenir una idea aproximada sobre els resultats que podíem esperar, vam mesurar les distàncies i rotacions entre els elements del nostre sistema abans de la prova. Cal tenir en compte que les eines utilitzades per aquesta mesura, com cintes mètriques o transportadors d'angles, no ens aporten dades precises, només intuïtives.

Així doncs, les dades inicials obtingudes van ser:

- L'origen del patró està en el punt (135, 102, 0) en mm i respecte al sistema de coordenades definit per la càmera Axis.

- El centre de la lent de la càmera ZC1 està en el punt (-35.5, 16.5, -1) en mm i respecte al sistema de coordenades definit per la càmera Axis.
- El centre de la lent de la càmera ZC2 està en el punt (38, 3, 1.5) en mm respecte al sistema de coordenades definit per la càmera Axis.
- El cap del participant estarà a uns 95cm del centre de la lent de la càmera Axis.

És a dir, vam col·locar la millor càmera de què disponíem de forma que el seu origen estigues, aproximadament, en el pla definit per la pantalla i de forma que la rotació entre els eixos del sistema que defineix la Axis i el sistema que defineix el patró sigui aproximadament la matriu:

$$R = \begin{pmatrix} r_{1,1} & r_{1,2} & r_{1,3} \\ r_{2,1} & r_{2,2} & r_{2,3} \\ r_{3,1} & r_{3,2} & r_{3,3} \end{pmatrix} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

És a dir, el sistema de coordenades definit pel patró tindrà uns eixos molt similars al del sistema de coordenades definit per la Axis si invertim l'eix X i l'eix Z.

De fet, hem col·locat la càmera Axis de manera que la cara del participant intersequi l'eix òptic de la càmera. Això farà que la millor càmera capti imatges frontals i centrades, que acostumen a ser imatges on la detecció facial és més precisa.

Per aquesta raó, i pel fet que és la càmera de major qualitat, considerarem sempre que els resultats més precisos són els que proporciona la càmera Axis i avaluarem la precisió dels resultats de les altres càmeres en funció de quant s'acosten als de la Axis.

3.1.1 Calibratge Intrínsec

Un cop fixades les càmeres i la pantalla, iniciem el calibratge intrínsec. En proves anteriors hem utilitzat dos mètodes diferents. En un d'ells utilitzàvem una llibreria de visió per computador anomenada OpenCV [11] i en l'altre, l'aplicació de Matlab anomenada Matlab Camera Calibration Toolbox [10]. A la pràctica, les dues opcions porten resultats similars, però ens hem decantat per utilitzar Matlab a la prova final per la seva simplicitat, la comoditat de la seva interfície gràfica i les representacions gràfiques que ens permeten veure fàcilment errors en la detecció dels punts dels patrons o zones no cobertes del camp de visió de la càmera.

Per tots els calibratges amb patró, hem utilitzat un patró en forma de taulell d'escacs amb 8x11 caselles amb costats de 30mm generat en <https://calib.io/pages/camera-calibration-pattern-generator>. Per construir el patró, hem imprès en un paper DIN A4 el patró generat i l'hem adherit a una pissarra blanca de 60x40cm utilitzant cola en esprai, procurant que el paper quedés totalment pla i fix a la superfície de la pissarra.

Un cop obtingut el patró, hem capturat les imatges pertinents amb cada càmera, intentant omplir tot el camp de visió de cada càmera. Per obtenir un bon calibratge amb Matlab, hem de procurar capturar imatges nítides, on aparegui tot el patró a una distància

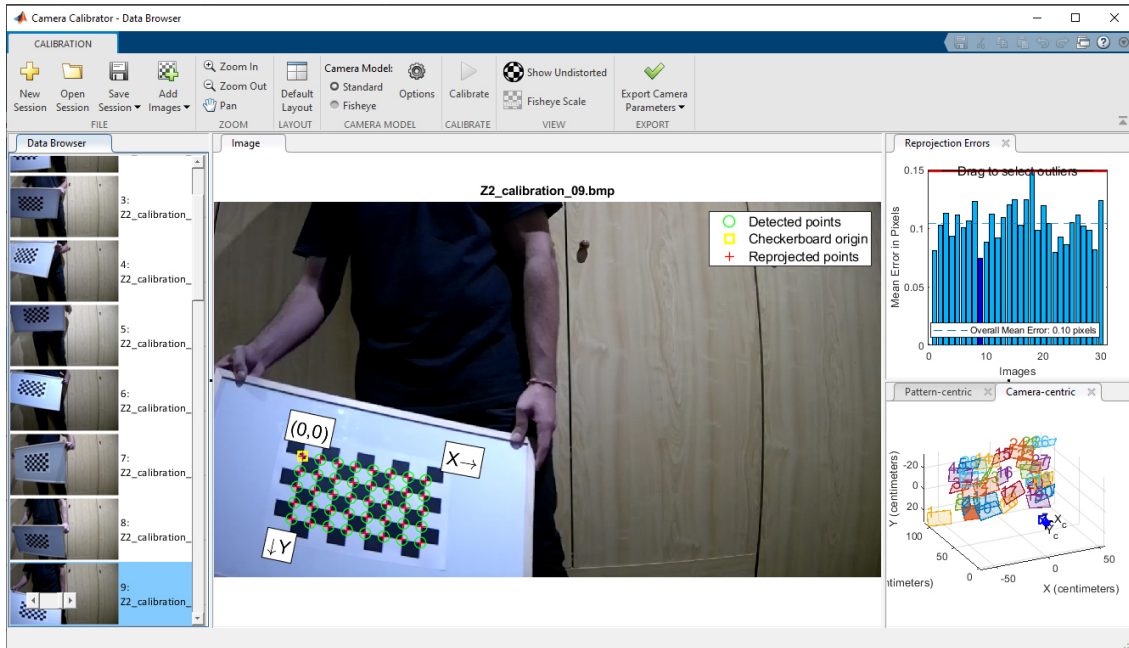


Figura 14: Interfície gràfica de Matlab

de la càmera aproximadament igual a la distància on estarà l'objecte d'interès (el cap del participant en aquest cas) i en diferents angles, però sempre inferiors a 45° respecte del pla de la càmera.

Per tal d'orientar el taulell i considerar sempre el mateix punt d'origen, Matlab no acceptarà patrons quadrats.

Tenint en compte aquestes consideracions al capturar les imatges, el calibratge utilitzant Matlab és molt simple, ja que només ens cal afegir les imatges i especificar les mides del nostre taulell. Un cop calibrada la càmera, Matlab ens mostrarà la gràfica amb els errors de re-projecció de cada patró durant la detecció dels seus punts i l'estimació de la posició dels patrons a l'espai respecte a la càmera.

Finalment, podem exportar els paràmetres intrínsecs i, fins i tot, utilitzar-los per corregir la distorsió de les nostres imatges.

3.1.2 Calibratge Càmera-Càmera

Pel calibratge Càmera-Càmera, hem utilitzat de nou Matlab. En aquest cas, fem servir l'aplicació de Stereo Camera Calibration. El procés és el mateix que el del calibratge intrínsec, però capturant les imatges amb dues càmeres al mateix moment de manera que el patró sigui visible per ambdues alhora.

En el nostre cas, ens interessa obtenir els paràmetres extrínsecs que permeten canviar els punts en el sistema de coordenades de la $ZC1$ o la $ZC2$ al sistema de coordenades definit per la càmera Axis, per tal de comparar els vectors de mirada obtinguts per cada càmera en el sistema de coordenades Axis.

Per tant, només haurem de fer 2 calibratges extrínsecs entre càmeres.

3.1.3 Calibratge Càmera-Pantalla

Pel calibratge càmera-pantalla, aplicarem el mètode basat en miralls plans sense distorsió proposat per Kosuke Takahashi, Shohei Nobuhara i Takashi Matsuyama [7] utilitzant el seu codi disponible en <https://vision.kuee.kyoto-u.ac.jp/takahashi2012cvpr/#download-links>.

Recordem que aquest mètode utilitza un patró d'escacs que haurem de mostrar per pantalla. En el nostre cas, utilitzarem exactament el mateix patró que vam fer servir per a altres calibratges, però aquest cop, en format pdf.

Farem servir la versió del codi de Matlab que utilitza únicament 3 punts de referència del patró de calibratge en forma de taulell d'escacs.

Per utilitzar aquest codi, a la pràctica, només hem de canviar les dades que utilitzen en el seu estudi per les nostres. Així doncs, haurem de proporcionar les següents dades:

- Imatges sense distorsió del patró de la pantalla mostrades a través del mirall en diferents posicions.
- Matriu Intrínseca de la càmera.
- Coordenades en píxels dels 3 punts de referència del patró en cada una de les imatges.

En aquest treball, hem utilitzat un mirall pla sense distorsió de 50x35cm durant el calibratge Càmera-Pantalla amb cada una de les càmeres. El procés de captura d'imatges ha sigut similar al realitzat durant el calibratge intrínsec, però en aquest cas, movent el mirall pel camp de visió de la càmera, en comptes del patró adherit a la pissarra.

Un cop capturades les imatges, corregirem la distorsió radial de cada una d'elles utilitzant Matlab o OpenCV a partir dels paràmetres intrínsecs prèviament calculats.

Quan tinguem les imatges sense distorsió que farem servir durant el calibratge Càmera-Pantalla, farem servir el codi adjunt a aquest treball que pertany a la carpeta "mirror_cal_ext".

Aquest codi, agafa les imatges i detecta les interseccions de les rectes que formen les caselles del patró utilitzant la llibreria OpenCV. Un cop detectades, guarda les coordenades en píxels dels 3 punts de referència que farem servir pel calibratge tenint en compte que, per l'efecte del mirall, els punts del patró apareixen en ordre invers a cada imatge. Finalment, el codi proporcionarà un arxiu de text per a cada imatge on apareixeran escrites les coordenades en píxels dels punts de referència de cada imatge.

Per acabar, ajuntarem les dades que hem obtingut i executarem amb elles el codi que realitza el calibratge Càmera-Pantalla utilitzant el mètode descrit.

3.1.4 Estímul visual

En el codi adjunt al treball, dins la carpeta "gaze_game", trobem el codi utilitzat per a mostrar l'estímul visual per pantalla. Amb aquest codi, accedirem a la IP de les 3 càmeres per tal d'iniciar la captura d'imatges. Les càmeres funcionaran a 30fps i guardaran imatges d'una resolució de 1820x720 píxels. Per tal d'obtenir i guardar totes les imatges que volem a un ritme constant durant tota la prova, aquestes seran emmagatzemades en format de vídeo i utilitzant diferents *threads* per la lectura i escriptura dels fotogrames. Al mateix temps, guardarem en un arxiu de text l'hora que marcava el rellotge intern de l'ordinador en el moment de la captura de cada frame (d'ara en endavant *timestamps*).

Per altra banda, el codi s'encarregarà de mostrar els estímuls visuals, mentre les càmeres graven, seguint el mètode explicat anteriorment. Els cercles que utilitzem com a estímuls visuals tindran un radi inicial de 40 píxels i es reduiran fins a ser un únic punt. En aquest moment, el participant tindrà 1,5 segons per clicar una tecla. Si es fa correctament, guardarem les coordenades en píxels del punt i el timestamp del moment en què s'ha pressionat la tecla.

D'aquesta manera, a l'acabar la prova tindrem les coordenades en píxels del punt final, el timestamp del moment en què la persona mirava al punt, els 3 vídeos del participant observant l'estímul visual i 3 arxius de text amb els timestamps dels fotogrames de cada un dels vídeos.

3.1.5 Càlcul del punt origen

Per tal d'estimar el centre de curvatura de la còrnia que, com hem dit, serà l'origen dels vectors mirada que volem calcular, treballarem amb un Software d'anàlisi automàtic del comportament facial anomenat *OpenFace* [15].

Aquest Software és capaç de detectar cares en una imatge i estimar la posició, en mil·límetres i en el sistema de coordenades definit per la càmera, dels seus punts característics (68 landmarks), que ens mostren l'ubicació i la forma del contorn de la cara, dels ulls, boca o celles.

El procés que seguirem consistirà, primerament, en corregir la distorsió de cada un dels vídeos que hem obtingut utilitzant Matlab i els paràmetres intrínsecs estimats. Després, procesarem el vídeo mitjançant OpenFace i els paràmetres intrínsecs. Aquest software ens proporcionarà un arxiu .csv on, a més dels landmarks, trobarem la posició del cap i l'eix òptic de cada ull, que podem utilitzar per estimar els eixos visuals.

Revisant el codi que utilitza OpenFace [15] podem observar que, tot i que el programa no ens proporciona en cap moment la posició dels centres dels globus oculars, OpenFace els estima internament per, a partir d'ells, calcular l'eix òptic.

OpenFace, basant-se en dades mitjanes de l'anatomia de l'ull calculades empíricament utilitzant el conjunt de dades MPIGaze [15], utilitza que els punts mitjans dels vèrtexs dels ulls (landmarks 36 i 39 per un ull i 41 i 45 per l'altre, en la Figura 12) estan a una distància fixada, en una direcció marcada per la posició del cap, del centre del globus ocular.

Per tant, a partir dels landmarks i la posició del cap que ens dona OpenFace, podem calcular el punt mitjà dels vèrtexs de cada ull, restar el valor que considera OpenFace en el seu codi en la direcció marcada per la posició del cap, i obtenir el centre del globus ocular.

Finalment, com OpenFace també dóna els eixos òptics, podrem calcular fàcilment el centre de curvatura de la còrnia utilitzant que el centre del globus ocular es troba a 5,2mm del centre de curvatura de la còrnia respecte la direcció marcada per l'eix òptic [13].

Els càlculs que hem descrit per obtenir el centre de curvatura de la còrnia a partir de l'arxiu .csv que proporciona OpenFace, apareixen en el codi adjunt al treball que pertany a la carpeta "get_gaze_vector", concretament en el mètode write_origin_points()

Aquest mètode write_origin_points(), l'hem definit de manera que, mitjançant els càlculs que hem descrit, obtingui el punt de curvatura de la còrnia de cada ull per l'estimació de cada vector mirada a partir de les dades de 7 fotogrames del vídeo concrets extrets de l'arxiu .csv que proporciona OpenFace. Un d'aquests fotogrames seleccionats serà aquell que tingui un timestamp que coincideixi, o s'acosti el màxim possible, al timestamp del moment en el qual l'usuari va clicar la tecla (aquests timestamps els vam extreure del codi de gaze_game). Llavors, com les càmeres ZC1 i ZC2 tenen un frame-rate variable que pot provocar pèrdua de fotogrames, també utilitzarem els 3 fotogrames anteriors i els 3 fotogrames posteriors. Un cop calculats el centre de curvatura de la còrnia dels 7 fotogrames, agafarem el punt mitjà com a punt d'origen del vector mirada.

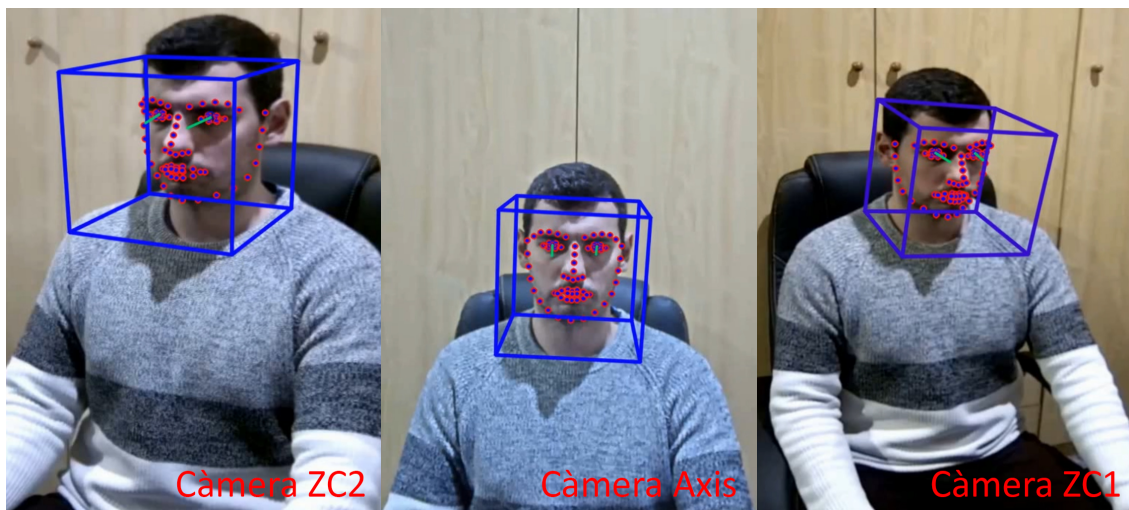


Figura 15: Representació visual de la posició del cap, landmarks facials i eixos òptics donada per OpenFace.

3.1.6 Càlcul del Vector Mirada

En la carpeta "get_gaze_vector" del codi adjunt al treball, s'inclouen els mètodes que ens permeten calcular els vectors de mirada a partir de les dades recollides. Com ja hem dit, el mètode write_origin_points() ens proporcionarà els punts d'origen dels vectors en mm i respecte al sistema de coordenades d'una càmera.

Per altra banda, el mètode `write_final_points()` ens proporcionarà el punt final del vector en mm i respecte al sistema de coordenades d'una càmera. Aquest mètode agafarà les coordenades dels punts finals expressades en píxels i respecte al sistema de coordenades definit per la pantalla, i les canviarà a coordenades expressades en mm i en el sistema de coordenades definit pel patró. Després, a partir dels paràmetres extrínsecs obtinguts en el calibratge Pantalla-Càmera, expressarem aquestes coordenades en el sistema definit per la càmera.

Així doncs, a partir del punt final i punt inicial expressats en el mateix sistema de coordenades, el mètode `compute_gaze_vector()` proporcionarà la parella de vectors mirada normalitzats.

En la carpeta "get_gaze_vector" també s'inclou el mètode `change_cam_coord()` que s'ocuparà de canviar les coordenades dels punts entre sistemes de coordenades definits per diferents càmeres a partir dels paràmetres extrínsecs calculats durant el calibratge Càmera-Càmera.

3.2 Resultats

Resultats del calibratge intrínsec de la Axis Càmera:

- Distància focal: $f = (f_x, f_y) = (8.946344699705428e + 02, 8.990978275357398e + 02)$
- Punt Principal: $c = (c_x, c_y) = (6.644757664960869e + 02, 3.773905809279072e + 02)$
- Coeficients de distorsió radial: $(-0.393389999404963, 0.148434372739964)$
- Mitjana de l'error de reprojecció: 0.114109485868319

Resultats del calibratge intrínsec de la Càmera ZC1:

- Distància focal: $f = (f_x, f_y) = (8.964995612172906e + 02, 8.036395281991663e + 02)$
- Punt Principal: $c = (c_x, c_y) = (6.269146074227509e + 02, 3.878537133539898e + 02)$
- Coeficients de distorsió radial: $(-0.426845712551418, 0.146649624766467)$
- Mitjana de l'error de reprojecció: 0.137231298427994

Resultats del calibratge intrínsec de la Càmera ZC2:

- Distància focal: $f = (f_x, f_y) = (1.101357137210433e + 03, 9.866929521685016e + 02)$
- Punt Principal: $c = (c_x, c_y) = (6.360139277742617e + 02, 3.541052740042482e + 02)$
- Coeficients de distorsió radial: $(-0.415664007537944, 0.179963379840488)$
- Mitjana de l'error de reprojecció: 0.119584246529878

Resultats del calibratge Extrínsec ZC1-Axis:

$$R = \begin{pmatrix} 0.942677550629630 & 0.000399181414108711 & -0.333704774004090 \\ -0.107247241218772 & 0.947310897271111 & -0.301827919786035 \\ 0.316001684789630 & 0.320315320529008 & 0.893051527429696 \end{pmatrix}$$

$$t = (-370.871529153298, -173.083703160387, 10.0396315853324)$$

Resultats del calibratge Extrínsc ZC2-Axis:

$$R = \begin{pmatrix} 0.909871703069329 & 0.013906004501726 & 0.414656613347137 \\ 0.089812035532703 & 0.982341908524189 & -0.164128525950019 \\ -0.405052196916881 & 0.186577055920206 & 0.895054031875706 \end{pmatrix}$$

$$t = (3.845124228301661e + 02, -27.425195850228390, 32.471461015263610)$$

Resultats del calibratge Extrínsc Axis-Pantalla:

$$R = \begin{pmatrix} -0.9999 & -0.0006 & -0.0126 \\ -0.0004 & 0.9999 & -0.0117 \\ 0.0126 & -0.0116 & -0.9999 \end{pmatrix}$$

$$t = (136.5901, 109.5108, -23.0178)$$

Resultats del calibratge Extrínsc ZC1-Pantalla:

$$R = \begin{pmatrix} -0.9374 & -0.0009 & 0.3484 \\ 0.1125 & 0.9456 & 0.3053 \\ -0.3297 & 0.3254 & -0.8862 \end{pmatrix}$$

$$t = (482.9486, 215.9013, 275.9893)$$

Resultats del calibratge Extrínsc ZC2-Pantalla:

$$R = \begin{pmatrix} -0.8978 & -0.0111 & -0.4403 \\ -0.0974 & 0.9800 & 0.1738 \\ 0.4296 & 0.1989 & -0.8809 \end{pmatrix}$$

$$t = (-236.4097, 111.0976, 111.3621)$$

Punts inicials de les 8 parelles de vectors respecte les coordenades definides per la càmera Axis:

punts origen del vector mirada 1:

$$(-43.34396557104435, -33.73447204927673, 947.3718570969259)$$

$$(23.93750780038422, -31.1679990207053, 946.0902905826403)$$

punts origen del vector mirada 2:

$$(-46.766613679067504, -36.90720257188625, 943.1061432272048)$$

$$(20.57224723521821, -34.67269908617197, 941.6115815129192)$$

punts origen del vector mirada 3:

$$(-36.145098011541315, -45.86866095308593, 946.8581218484467)$$

$$(30.795059074172972, -43.27608701022879, 945.5888417913037)$$

punts origen del vector mirada 4:

$$(-37.58649254028544, -46.36602878118935, 946.2957371849866)$$

$$(29.544771459714564, -43.58188020976077, 944.8154790135579)$$

punts origen del vector mirada 5:

(-38.11682235394903, -46.429931556352614, 945.2980756678307)
(28.98687924605097, -44.1344014420669, 943.7872799535451)

punts origen del vector mirada 6:

(-39.93840922384709, -46.13537520436064, 943.6060308045548)
(27.207183290438625, -43.636778118646355, 941.7997470331262)

punts origen del vector mirada 7:

(-35.71352957211156, -44.82925505697278, 942.2982962581207)
(31.839319399317006, -42.144808942687064, 941.5297136866922)

punts origen del vector mirada 8:

(-41.67882134469921, -30.937772751183978, 938.6002095614384)
(26.038432083872223, -28.671943779755402, 937.6194117328668)

Punts finals dels 8 vectors respecte les coordenades definides per la càmera Axis:

Punt final del vector mirada 1:

Coordenades de la pantalla en píxels: (1270, 415)
Coordenades a l'espai en mm: (-74.2748834090277, 167.00024773333334, -26.34245189953704)

Punt final del vector mirada 2:

Coordenades de la pantalla en píxels: (872, 144)
Coordenades a l'espai en mm: (24.425445165000056, 100.06073467333334, -24.322175963333333)

Punt final del vector mirada 3:

Coordenades de la pantalla en píxels:(240, 245)
Coordenades a l'espai en mm: (181.07681454444446, 125.08605355833333, -22.637578207407408)

Punt final del vector mirada 4:

Coordenades de la pantalla en píxels: (229, 622)
Coordenades a l'espai en mm: (183.74769022284724, 218.26454608083333, -23.684183202453706)

Punt final del vector mirada 5:

Coordenades de la pantalla en píxels: (393, 314)
Coordenades a l'espai en mm: (143.1394431509028, 142.12457009083334, -23.313350516712966)

Punt final del vector mirada 6:

Coordenades de la pantalla en píxels: (430, 471)
Coordenades a l'espai en mm: (133.9442365304167, 180.92422208666667, -23.879091836944447)

Punt final del vector mirada 7:

Coordenades de la pantalla en píxels: (1548, 108)
Coordenades a l'espai en mm: (-143.14271506249992, 91.09612175000001, -26.330591005)

Punt final del vector mirada 8:

Coordenades de la pantalla en píxels: (1155, 405)
Coordenades a l'espai en mm: (-45.76607426562495, 164.54010268750002, -25.95455083125)

Vectors mirada normalitzats respecte les coordenades definides per la càmera Axis:

Parella de vectors número 1:

(-0.031096626968759648, 0.20181013487717778, -0.9789308807326599)
(-0.09848152654843925, 0.19871129511705915, -0.9750975387732195)

Parella de vectors número 2:

(0.07266970207939662, 0.13981080693059655, -0.9875080013170057)
(0.003950811576704478, 0.13814665624932917, -0.9904039036953636)

Parella de vectors número 3:

(0.2154694312301748, 0.16957550308097305, -0.9616740991417934)
(0.15116159621133377, 0.1693478348736435, -0.9738950059705875)

Parella de vectors número 4:

(0.21499101258723083, 0.25704658241611594, -0.9421814681763339)
(0.15191604467542574, 0.2579631677112773, -0.9541365308353608)

Parella de vectors número 5:

(0.18065974273535315, 0.18793396004433024, -0.9654236810938771)
(0.11513500921461915, 0.18786199512453075, -0.9754239080732954)

Parella de vectors número 6:

(0.17235386518912124, 0.22506328363958805, -0.9589789692749974)
(0.10703954296771513, 0.22519739959756294, -0.9684155448234802)

Parella de vectors número 7:

(-0.10917585159571103, 0.13813535584980183, -0.984377598735668)
(-0.17629812651604054, 0.13424307545984188, -0.9751398706226801)

Parella de vectors número 8:

(-0.004152987106318325, 0.19862169334103302, -0.9800674342270717)
(-0.07287032084456035, 0.19607994724695438, -0.9778765620606941)

Finalment, podem comparar els resultats obtinguts a partir de les càmeres ZC1 i ZC2 amb els resultats obtinguts a partir de la càmera Axis expressant tots els punts finals i inicials en el sistema de coordenades definit per la càmera Axis:

punts origen de la parella de vectors mirada número 1:

Axis 1: (-43.34396557104435, -33.73447204927673, 947.3718570969259)
Axis 2:(23.93750780038422, -31.1679990207053, 946.0902905826403)
ZC1 1: (-24.036866038400092, -18.14793124936685, 983.8476651029328)
ZC1 2:(46.15015960739703, -13.384151447326701, 990.1031869110627)
ZC2 1: (-57.43749888874146, -21.578576945943922, 963.8551602431586)
ZC2 2:(14.714862856015088, -16.42547348310046, 955.4646204538063)

punts origen de la parella de vectors mirada número 2:

Axis 1: (-46.766613679067504, -36.90720257188625, 943.1061432272048)
Axis 2:(20.57224723521821, -34.67269908617197, 941.6115815129192)
ZC1 1: (-30.9080253367444, -20.983707641036744, 986.5796924481119)

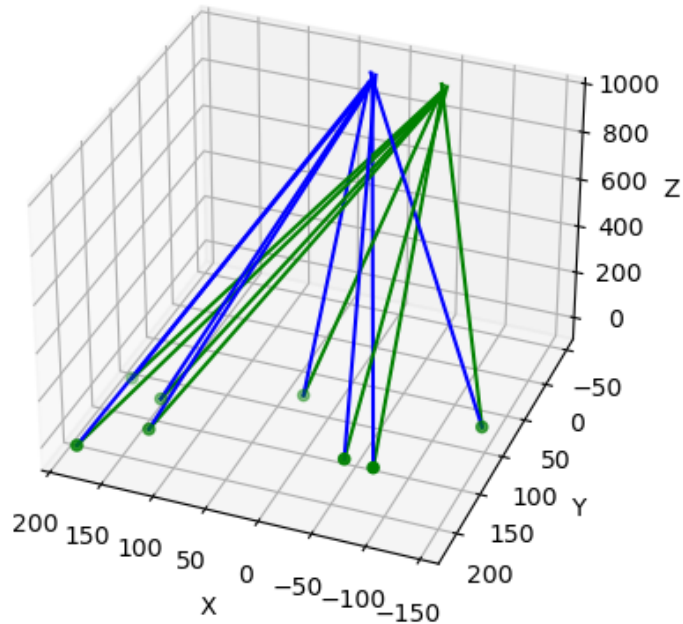


Figura 16: Representació visual dels vectors mirada respecte el sistema de coordenades definit per la càmera Axis.

ZC1 2:(39.680801764313344, -16.58199628584063, 993.0986438386439)
 ZC2 1: (-59.470959395199884, -21.459234556087893, 958.651377023471)
 ZC2 2:(13.31978605985256, -16.432282771338038, 950.0222459052598)

punts origen de la parella de vectors mirada número 3:

Axis 1: (-36.145098011541315, -45.86866095308593, 946.8581218484467)
 Axis 2:(30.795059074172972, -43.27608701022879, 945.5888417913037)
 ZC1 1: (-27.422029753211405, -33.51522820045329, 987.1019582309738)
 ZC1 2:(43.2344306875267, -29.02894537395946, 994.0540721956794)
 ZC2 1: (-49.84815773535121, -20.87945362182316, 956.8679546024748)
 ZC2 2:(22.874166115599564, -16.118123128453846, 949.0648634673894)

punts origen de la parella de vectors mirada número 4:

Axis 1: (-37.58649254028544, -46.36602878118935, 946.2957371849866)
 Axis 2:(29.544771459714564, -43.58188020976077, 944.8154790135579)
 ZC1 1: (-18.057394863694014, -27.684909223322506, 983.3077673085612)
 ZC1 2:(52.46556278771368, -23.32575440534677, 990.2045979308103)
 ZC2 1: (-54.00761841883087, -34.934898939420805, 960.0187556533851)
 ZC2 2:(18.217843360831864, -29.734418684300863, 951.9959018944439)

punts origen de la parella de vectors mirada número 5:

Axis 1: (-38.11682235394903, -46.429931556352614, 945.2980756678307)
Axis 2:(28.98687924605097, -44.1344014420669, 943.7872799535451)
ZC1 1: (-20.312272902930772, -23.45340748509716, 982.0141846843252)
ZC1 2:(50.00553488758777, -19.13952666810951, 988.90839584561)
ZC2 1: (-48.89221284736118, -32.40981463760707, 959.8243621407856)
ZC2 2:(23.779525350666063, -27.554196493547465, 951.9101512905063)

punts origen de la parella de vectors mirada número 6:

Axis 1: (-39.93840922384709, -46.13537520436064, 943.6060308045548)
Axis 2:(27.207183290438625, -43.636778118646355, 941.7997470331262)
ZC1 1: (-21.023299975095654, -22.204978494244216, 981.6083531703354)
ZC1 2:(49.38555192923951, -17.908420340808533, 988.3891346821837)
ZC2 1: (-50.40374018632053, -34.08958887573317, 957.6837629161839)
ZC2 2:(22.259763372005523, -29.38171262539018, 949.6281367761527)

punts origen de la parella de vectors mirada número 7:

Axis 1: (-35.71352957211156, -44.82925505697278, 942.2982962581207)
Axis 2:(31.839319399317006, -42.144808942687064, 941.5297136866922)
ZC1 1: (-33.556708042381956, -25.990051027943537, 987.1366002705842)
ZC1 2:(36.7563366378339, -21.71968617021969, 993.9945829440378)
ZC2 1: (-47.340527796434344, -25.34796756482217, 954.7223955196685)
ZC2 2:(24.509353848359297, -19.937280999729836, 947.2602973026347)

punts origen de la parella de vectors mirada número 8:

Axis 1: (-41.67882134469921, -30.937772751183978, 938.6002095614384)
Axis 2:(26.038432083872223, -28.671943779755402, 937.6194117328668)
ZC1 1: (-20.46089088651081, -31.78374777158919, 987.6997793519058)
ZC1 2:(49.48419447858447, -27.280410392186724, 994.3748522258066)
ZC2 1: (-49.13587397099974, -26.93090592428419, 952.575091139484)
ZC2 2:(23.392034782727308, -22.052534111955225, 945.1019824838336)

punts final del vector mirada número 1:

Axis: (-74.2748834090277, 167.00024773333334, -26.34245189953704)
ZC1: (-62.349918331773665, 177.74278071673066, 27.216122558297187)
ZC2:(-76.64795461084833, 150.85392738545707, 22.558422403200048)

punts final del vector mirada número 2:

Axis: (24.425445165000056, 100.06073467333334, -24.322175963333333)
ZC1: (36.27401326428202, 110.69835626549039, 28.38558249766423)
ZC2:(22.16404271763156, 86.65551901091128, 18.93117677661983)

punts final del vector mirada número 3:

Axis: (181.07681454444446, 125.08605355833333, -22.637578207407408)
ZC1: (192.9571452399676, 135.57119150327813, 30.96374775445132)
ZC2:(178.72367548984735, 116.00083015720654, 14.823200935113185)

punts final del vector mirada número 4:

Axis: (183.74769022284724, 218.26454608083333, -23.684183202453706)
ZC1: (195.73668081023203, 228.7569944531903, 31.517906555934346)

ZC2:(181.2029364809709, 209.23140692948053, 15.91830172001007)

punts final del vector mirada número 5:

Axis: (143.1394431509028, 142.12457009083334, -23.313350516712966)

ZC1: (155.03900765645113, 152.64910414502603, 30.466436716293472)

ZC2:(140.761190065621, 131.98777154676426, 16.108227292082923)

punts final del vector mirada número 6:

Axis: (133.9442365304167, 180.92422208666667, -23.879091836944447)

ZC1: (145.88887666506224, 191.46200928364274, 30.53660840205795)

ZC2:(131.48885696239154, 170.5244812553523, 16.8552104683667)

punts final del vector mirada número 7:

Axis: (-143.14271506249992, 91.09612175000001, -26.330591005)

ZC1: (-131.30736859765656, 101.89875571321105, 25.725687437081902)

ZC2:(-145.34247012145147, 73.06571953099447, 23.549092083860508)

punts final del vector mirada número 8:

Axis: (-45.76607426562495, 164.54010268750002, -25.95455083125)

ZC1: (-33.84350533733351, 175.25413586595417, 27.64673534743183)

ZC2:(-48.14146769072261, 149.18160426309717, 21.722701149291346)

Així doncs, per tal de donar una millor estimació del vector mirada en angles no frontals, s'adjuntarà a cada imatge de la ZC1 o ZC2 els vectors de mirada calculats amb la càmera axis però expressats en el sistema de coordenades definit per la ZC1 o la ZC2, respectivament.

Referències

- [1] CR. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd Edition, Part I: Camera Geometry and Single View Geometry, Cambridge University Press, 2004.
- [2] Funes Mora, K. A.; Monay, F.; Odobez, J. M.: EYEDIAP Database: Data Description and Gaze Tracking Evaluation Benchmarks (Revision 2: 2nd of September 2014)
- [3] Seonwook Park, Emre Aksan, Xucong Zhang, Otmar Hilliges, Towards End-to-end Video-based Eye-Tracking, 26 de Juliol de 2020.
- [4] Zhang, X.; Park, S.; Beeler, T.; Bradley, D.; Tang, S.; Hilliges, O.: ETH-XGaze: A Large Scale Dataset for Gaze Estimation under Extreme Head Pose and Gaze Variation, European Conference on Computer Vision (ECCV), 2020.
- [5] Zhang, X.; Sugano, Y.; Fritz, M.; Bulling, A.: MPIIGaze: Real-World Dataset and Deep Appearance-Based Gaze Estimation, 24 Nov 2017.
- [6] Zhang, Z. “A Flexible New Technique for Camera Calibration”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 22, No. 11, 2000, pp. 1330–1334. <https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr98-71.pdf>
- [7] Kosuke, T.; Nobuhara, S.; Matsuyama, T.: Mirror-based Camera Pose Estimation Using an Orthogonality Constraint, pp. 11-19, 01 Feb 2016.
- [8] Rodrigues, R.; Barreto, J.P.; Nunes, U.: Camera Pose Estimation Using Images of Planar Mirror Reflections, pp. 383-395, 2010.
- [9] Francken, Y.; Hermans, C.; Bekaert, P.: Screen-Camera Calibration using a Spherical Mirror, pp. 11-20, 2007.
- [10] Matlab, “Camera Calibration and 3-D Vision”, [Online]. <https://es.mathworks.com/help/vision/camera-calibration-and-3-d-vision.html>
- [11] K. Sadekar and S. Mallick, “Camera Calibration using OpenCV”, [Online]. <https://www.learnopencv.com/camera-calibration-using-opencv/>
- [12] Puell Marín, M. C.: Óptica Fisiológica: El sistema óptico del ojo y la visión binocular, Capítulo 1: Estructura anatómica del ojo humano.
- [13] Takemura, Kentaro; Yamagishi, Kenta: A hybrid eye-tracking method using a multispectral camera, pp. 1529-1534, 2017.
- [14] Eizenman, M.; Model, D.; Guestrin, E.: Covert monitoring of the point-of-gaze, pp. 551-556m, 2009.
- [15] Baltrušaitis, T.; Zadeh, A.; Lim Y.C.; Morency L.: OpenFace 2.0: Facial Behavior Analysis Toolkit, IEEE International Conference on Automatic Face and Gesture Recognition, 2018. <https://github.com/TadasBaltrusaitis/OpenFace/wiki>
<https://par.nsf.gov/servlets/purl/10099458>