

Ejercicio Práctico 2: Ejemplo sencillo de realidad aumentada

Visión Artificial, 4^o GITT

Con esta práctica, se pretenden reforzar los conocimientos adquiridos, en relación con la calibración de cámaras y el uso de las ecuaciones de proyección de puntos en la imagen. Partiendo de una cámara previamente calibrada y de una plantilla de referencia, correspondiente a un objeto plano sencillo, se pretende sobreponer encima de dicha base otro objeto virtual también sencillo, el cual deberá estar colocado, en todo momento, en una posición y orientación fija, respecto a dicha base de referencia.

1 Calibración de la cámara

En primer lugar se requiere realizar la calibración de la cámara que se vaya a usar, pudiendo ser la cámara de un teléfono móvil. Durante la calibración y durante la realización de las pruebas posteriores, dicha cámara deberá estar ajustada siempre del mismo modo: sin cambio de zoom (evitar también el uso de zoom digital), de resolución, de tamaño de la imagen u otros. Todas las imágenes deben ser tomadas con buenas condiciones de enfoque y luminosidad (iluminación lo más difusa posible, evitando la existencia de reflejos o destellos).

Tenga en cuenta que si, debido a la gran resolución de las imágenes, considera que puede convenir trabajar con las imágenes escaladas por software mediante el comando *imresize()* de MATLAB, resultará conveniente realizar la calibración con imágenes que vengan ya en este mismo formato. Por otra parte, si considera interesante presentar las pruebas finales mediante una secuencia de imágenes procedentes de un vídeo capturado con la cámara, tenga en cuenta que su teléfono móvil probablemente use una resolución diferente en modo vídeo a la que usa en modo fotografía y, por tanto, puede requerirse realizar la calibración utilizando cuadros (frames) extraídos igualmente de fragmentos de vídeo de la plantilla de calibración.

Para realizar la calibración, hágase uso del Toolbox de calibración de cámaras descrito en clase y que se encuentra disponible en el entorno de enseñanza virtual (*BouquetCaltechCalibrationToolbox.zip*).

2 Creación del objeto base de referencia

Se deberá generar un objeto base físico y disponer de un modelo geométrico del mismo. Dicho objeto puede consistir en una plantilla de al menos cuatro puntos no colineales que puedan localizarse fácilmente y ser identificados unívocamente en cada imagen capturada. Véase como ejemplo la plantilla de cuatro puntos mostrada en la figura. Cada alumno o grupo deberá elegir un objeto de referencia particular. En el caso mostrado, los puntos de referencia son los centros de los cuatro círculos. Para favorecer la identificación de cada uno de los puntos, se ha elegido una plantilla asimétrica (claramente rectangular) y con uno de los puntos de tamaño significativamente mayor al resto.

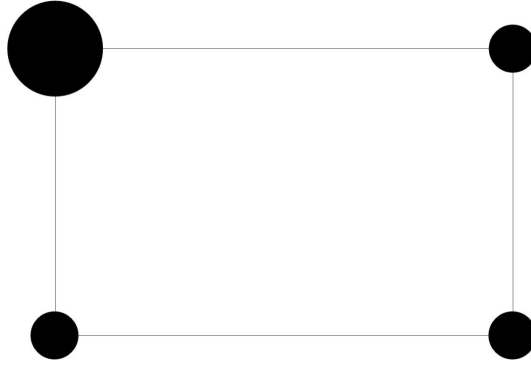


Figure 1: Ejemplo de objeto base a emplear.

3 Localización de los puntos de referencia en la imagen

En cada imagen capturada, deberán extraerse de forma automática los puntos de referencia presentes en la plantilla base. Para ello, se recurrirá a instrucciones que se verán con detalle más adelante en la asignatura, pero que resultan necesarias aquí y se comentarán de forma breve.

Pasos recomendados y funciones útiles:

- Lectura de la imagen en memoria. Puede emplearse la función *imread()*.
- Conversión a escala de grises, en caso de la que imagen capturada sea en color. Para ello puede recurrirse a la función *rgb2gray()*.
- Binarización de la imagen. Pueden usarse las funciones *graythresh()* y *im2bw()*. Idealmente, en la imagen binaria resultante, se deseará que los puntos del objeto de referencia aparezcan en blanco, en lugar de en negro.
- Eliminación de puntos o líneas espúreos. Puede usarse una operación morfológica de *apertura*, mediante la función *imopen()*. Alternativamente, puede ensayarse con filtrado de la mediana, previo o posterior a la binarización, usando la función *medfilt2()*.
- Obtención de las propiedades de las regiones presentes en la imagen binarizada. Puede usarse la función *regionprops()*. De las regiones resultantes, probablemente sea conveniente realizar un filtrado por área mínima y máxima, de forma que se consiga aislar exclusivamente las regiones de interés del objeto de referencia.

Si, en lugar de imágenes individuales, se pretende trabajar con un fragmento de vídeo, pueden usarse las siguientes funciones para la extracción de cuadros (frames) individuales del mismo, en lugar de la función *imread()* indicada anteriormente como primer paso:

- Apertura del objeto vídeo:

```
miVideo = VideoReader ('nombreDelVideo.mp4');
```

- Lectura de un cuadro concreto del vídeo:

```
I = read (miVideo, nroFrame);
```

- Alternativamente, lectura hasta que finaliza el vídeo:

```
while hasFrame (miVideo)
    I = readFrame (miVideo);
    .....
end
```

4 Estimación de la posición y orientación del objeto de referencia respecto a la cámara

Los parámetros extrínsecos de la cámara, es decir, la posición y orientación relativa entre el sistema de referencia ligado al objeto base, $\{O\}$, y el sistema de referencia de la cámara, $\{C\}$, dados mediante el vector de traslación Ct_O y la matriz de rotación CR_O , pueden obtenerse mediante el uso de la función *compute_extrinsic()* del Toolbox de calibración citado más arriba. La sintaxis de dicha función es la siguiente:

```
[~, cto, cRo, ~] = compute_extrinsic (mp, MP, fc, cc, kc, alpha_c)
```

• Parámetros de entrada:

- mp: Matriz conteniendo los puntos 2D en la imagen, dispuestos por columnas.
- MP: Matriz conteniendo los puntos 3D correspondientes a los puntos 2D de la matriz anterior, también organizados por columnas.
- fc: Vector conteniendo los parámetros asociados a la distancia focal: $fc = [f_x, f_y]$.
- cc: Vector conteniendo las coordenadas del punto principal: $cc = [u_0, v_0]$.
- kc: Vector conteniendo los parámetros de distorsión de la lente: $kc = [\kappa_{R_1}, \kappa_{R_2}, \kappa_{T_1}, \kappa_{T_2}, \kappa_{R_3}]$, donde κ_{R_i} son los coeficientes de distorsión radial y κ_{T_i} son los coeficientes de distorsión tangencial.
- alpha_c: Coeficiente de *skewness* del sensor: $alpha_c = s$.

• Parámetros de salida:

Los parámetros primero y último devueltos por la función, no son necesarios y se han ignorado en la llamada anterior.

- cto: Corresponde al vector traslación: Ct_O .
- cRo: Matriz de rotación: CR_O .

5 Definición del objeto virtual y proyección del mismo en la imagen

Será necesario crear un objeto virtual, consistente en un conjunto de puntos, dados por sus coordenadas 3D, expresadas respecto al mismo sistema de coordenadas del objeto de referencia

base. Una vez estimada la posición y orientación del objeto base, este objeto virtual se proyectará sobre la imagen, de forma que aparezca siempre en la misma ubicación en relación al objeto base.

El objeto virtual podrá ser un objeto 3D sencillo formado por un conjunto de líneas, o también se puede pensar en unas letras planas construidas asimismo con segmentos rectos. Un ejemplo del primer caso, se muestra en la figura.

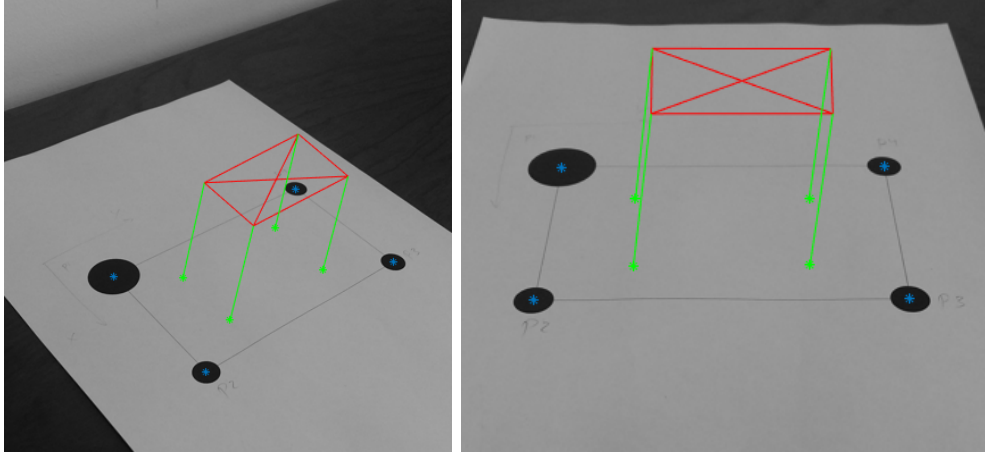


Figure 2: Ejemplos de objeto virtual proyectado sobre plantilla de referencia base.

En este caso, como se ve, el objeto virtual representa un prisma, al que se le ha quitado su base, con la cara superior en rojo y las aristas laterales en verde.

6 Desarrollo del trabajo

En primer lugar, se deberán incluir algunos detalles relevantes del proceso de calibración realizado, así como el juego de parámetros resultante. A continuación, se describirá la geometría precisa de los dos objetos a emplear: objeto de referencia base y objeto virtual. Deberán describirse los pasos seguidos para la extracción de los puntos de referencia pertenecientes al objeto base y finalmente, el procedimiento de proyección del objeto virtual en la imagen. Cada grupo definirá sus propios objetos de referencia y virtuales.

Se recomienda estructurar el código en funciones principales, para mayor organización y claridad. Se deberá grabar un vídeo, o bien una secuencia de fotografías mostrando una secuencia en movimiento encuadrando la plantilla, para demostrar la calidad de la reconstrucción de posición y orientación 3D realizada.