

Práctica 1. Calibración de una cámara con la Toolbox de Matlab (Camera Calibration Toolbox)

1. Objetivo

El objetivo de la práctica es que el alumno sea capaz de calibrar una cámara utilizando la Toolbox de Matlab (Camera Calibration Toolbox) de Jean-Yves Bouguet. Para ello y mediante un programa proporcionado, se tomarán imágenes de un patrón de calibración con una cámara USB conectada al ordenador, mostrando los pasos para obtener de forma correcta la matriz de parámetros intrínsecos de la cámara.

NOTA: las versiones modernas de Matlab (desde 2013b) incorporan en la "Computer Vision Toolbox" una aplicación integrada de calibración (**cameraCalibrator**), que realiza automáticamente la localización del patrón en la imagen. Adicionalmente, si se dispone módulo "Matlab Support Package for USB Webcams", se puede hacer la captura de imágenes desde la misma aplicación.

2. Escenarios

Se trabajará con imágenes capturadas por una cámara USB conectada al ordenador. Para ello se utilizará el software proporcionado en la documentación de la práctica para la captura de imágenes.

Se proporcionan un fichero pdf para imprimir una hoja con el patrón de calibración formado por tablero de casillas cuadas blancas y negras (figura 1), utilizando las esquinas como datos de calibración.

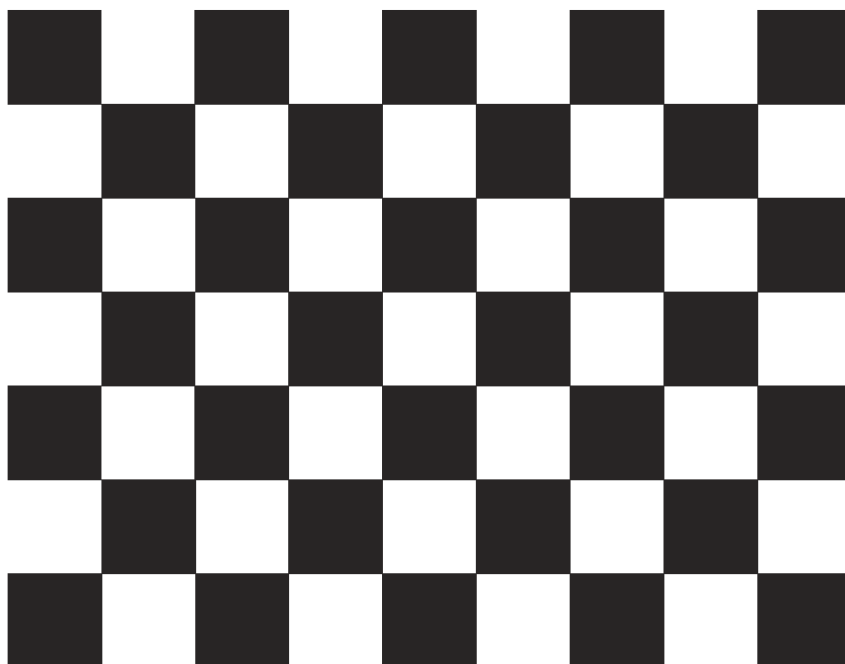


Figura 1. Patrón de calibración

3. Software utilizado

El desarrollo de la práctica se realizará en Matlab. Para ello se utiliza un programa para la captura de imágenes de una cámara conectada al ordenador desarrollado en OpenCV (<http://opencv.org>). El código fuente se encuentra disponible en el blog de la asignatura (<http://umh1782.edu.umh.es/opencv/>).

Programas suministrados:

- **Camera Calibration Toolbox** (http://www.vision.caltech.edu/bouguetj/calib_doc/index.html) Se trata de un paquete para Matlab de código libre desarrollado por de Jean-Yves Bouguet, y que está también implementado en la librería OpenCV. Podemos descargarla directamente desde la web del autor, pero se recomienda descargar el fichero local **toolbox_calib_umh.zip** en el que se han incluido la opción de almacenar ficheros de datos de calibración en formato **YAML** (usado en OpenCV), y se ha modificado para que se pueda ejecutar en **Octave**. Procederemos a descomprimirla y añadiremos la carpeta al **PATH** de Matlab.
- **capture.exe**: ejecutable para capturar imágenes de una cámara (en las versiones de linux o Mac es un ejecutable sin extensión).

Otros ficheros:

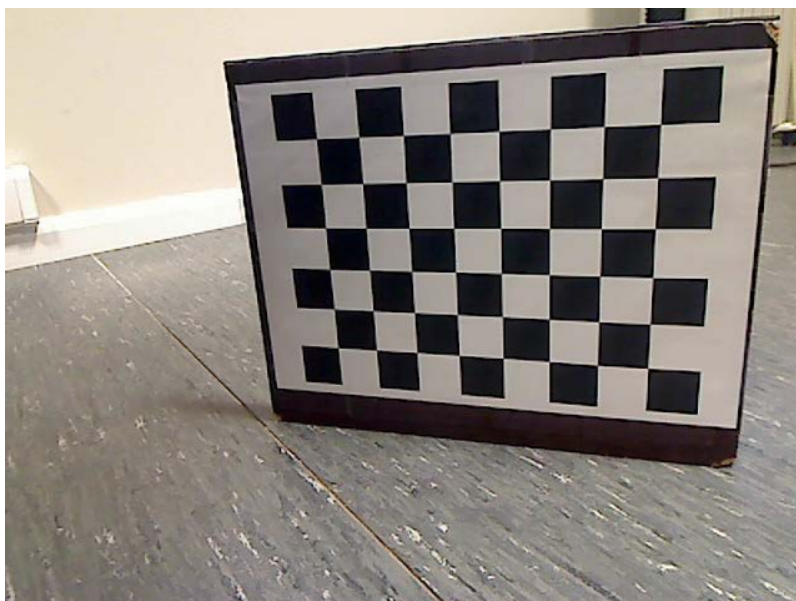
- **pattern.pdf**: fichero con el patrón de calibración

4. Tareas

A continuación se detallan las etapas de las que se compone la ejecución de la práctica

4.1. Captura de imágenes de patrón de calibración.

Se utilizará el ejecutable suministrado que captura y graba una imagen al pulsar la barra **espaciadora** o la tecla **F5**. Para salir del programa se debe pulsar una de las teclas: **Esc,q,Q**. El nombre de los ficheros grabados tienen la siguiente estructura: **ImageXXX.jpg**. Se utiliza una numeración correlativa para los mismos.



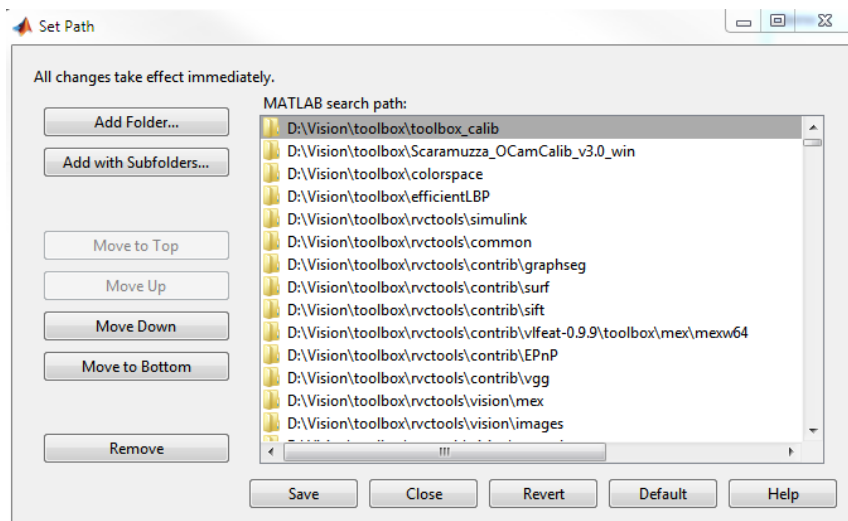
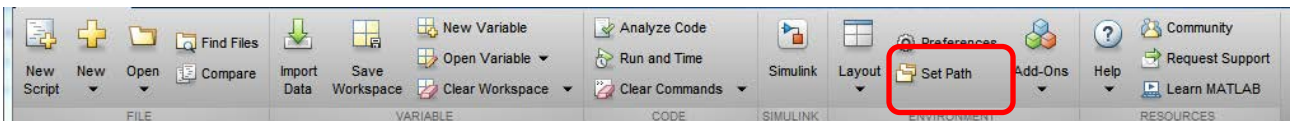
```
Parameters: capture -c id_camera
Capturing images from camera 0 (640,480)
...Hit F5 or Space bar or SHIT+LEFT Mouse click in order to capture and save the
image.
...Hit q/Q or ESC to exit.
Saving image window in file: Image1.jpg
```

Recomendaciones para la captura de imágenes del patrón de calibración:

- La toma de imágenes puede realizarse moviendo el patrón de calibración y dejando fija la cámara, o bien dejando el patrón de calibración en una posición fija y moviendo la cámara.
- El número mínimo teórico de imágenes que se deben capturar es de tres (algoritmo de Zhang), pero en la práctica es necesario disponer de al menos 8 vistas distintas tomadas situando el patrón a al menos dos distancias y cubriendo los cuatro cuadrantes de la imagen para poder obtener datos de calibración significativos y una buena estimación de la distorsión.
- Se debe asegurar que la normal al plano del patrón y el eje óptico de la cámara no son paralelos para evitar el acoplamiento matemático entre varios parámetros de calibración, lo que daría lugar a una estimación incorrecta de los mismos. Se recomienda un ángulo entorno a los **30 grados** entre la orientación del plano del patrón y la cámara.

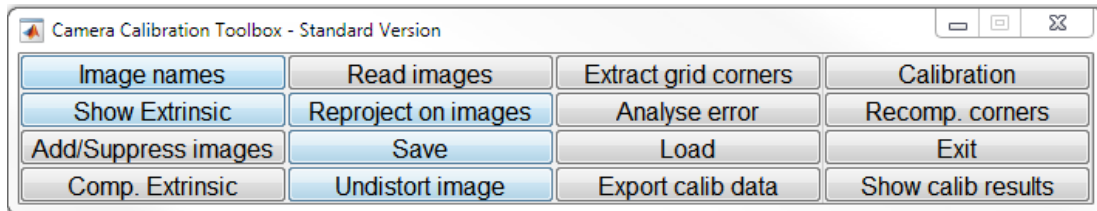
4.2. Manual de uso Toolbox de Calibración

Tras arrancar Matlab comprobaremos que la carpeta con la Toolbox de calibración está incluida en el directorio de búsqueda (PATH). Para ello pulsaremos en la opción **SetPath** y añadiremos el directorio **toolbox_calib** en la ubicación que la hayamos descomprimido.



Se recomienda leer el tutorial de Jean-Yves Bouguet con las instrucciones detalladas de esta toolbox (http://www.vision.caltech.edu/bouguetj/calib_doc/index.html). Aquí resumiremos los aspectos fundamentales para calibrar una cámara:

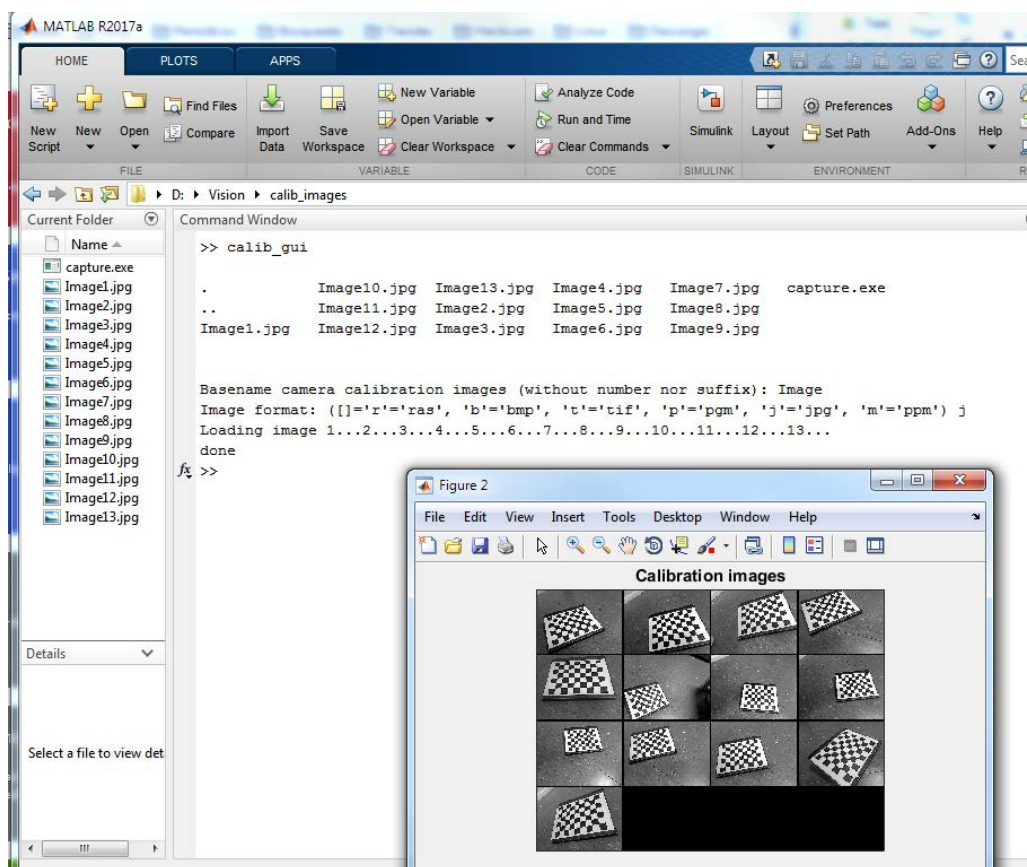
Cambiaremos la carpeta por defecto a la ubicación donde tengamos las imágenes capturadas. Lanzaremos la aplicación tecleando: `calib_gui`. Nos preguntará el modo de gestión de imágenes (memoria o disco), mostrando a continuación la interfaz de la librería de calibración:



Para realizar la calibración de la cámara ejecutaremos las siguientes etapas:

- 1) **Cargar imágenes:** pulsaremos el botón 'Image names'. No pedirá el prefijo de los ficheros de imagen y el formato. De manera alternativa podemos escribir en la consola el comando '`data_calib`'.

(En algunas versiones modernas de Matlab la interfaz de botones no funciona correctamente, no mostrando la entrada de datos en la consola. En ese caso se aconseja ejecutar el comando directo)



- 2) **Extraer las esquinas:** pulsaremos el botón 'Extrac grid corners'. De manera alternativa podemos escribir en la consola el comando 'click_calib'

Nos solicitará varios parámetros con valores por defecto (pulsar INTRO). Parámetros:

- Imágenes a procesar: vector con los índices de las imágenes (por defecto todas)
- Ventana búsqueda alrededor de la esquina a (wintx,winty) (por defecto 5 pixels en radio, con una ventana de 11x11)
- Número de esquinas en el patrón por filas (**X**) y columnas (**Y**) (por defecto las calcula automáticamente). Se utilizan solo las esquinas interiores, no las del contorno exterior.

```
Command Window
>> calib_gui

.      Image10.jpg Image13.jpg Image4.jpg  Image7.jpg  capture.exe
..     Image11.jpg Image2.jpg  Image5.jpg  Image8.jpg
Image1.jpg Image12.jpg Image3.jpg  Image6.jpg  Image9.jpg

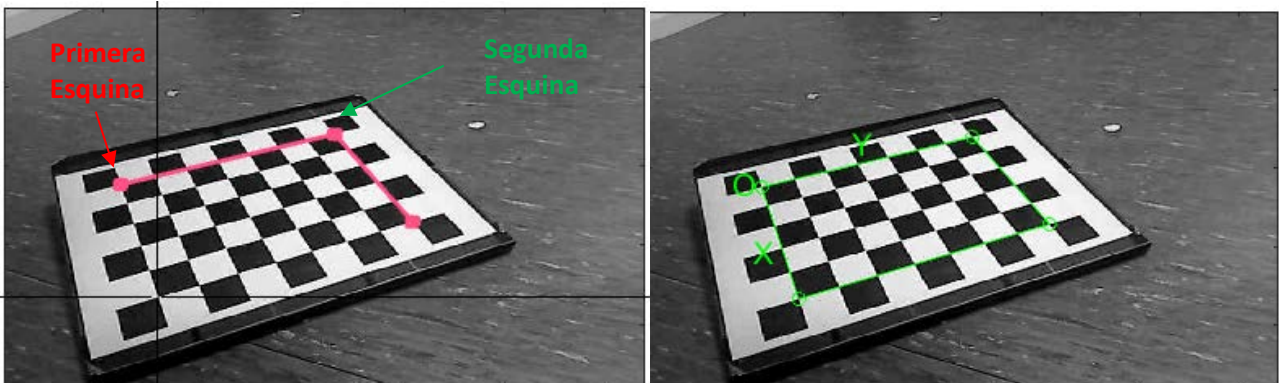
Basename camera calibration images (without number nor suffix): Image
Image format: (['r']='ras', 'b']='bmp', 't']='tif', 'p']='pgm', 'j']='jpg', 'm']='ppm') j
Loading image 1...2...3...4...5...6...7...8...9...10...11...12...13...
done

Extraction of the grid corners on the images
Number(s) of image(s) to process ([] = all images) =
Window size for corner finder (wintx and winty):
wintx ([]) = 5 =
winty ([]) = 5 =
Window size = 11x11
Do you want to use the automatic square counting mechanism (0=[]=default)
or do you always want to enter the number of squares manually (1,other)?

Processing image 1...
Using (wintx,winty)=(5,5) - Window size = 11x11      (Note: To reset the window size, run script clearwin)
Click on the four extreme corners of the rectangular complete pattern (the first clicked corner is the origin)...
fx |
```

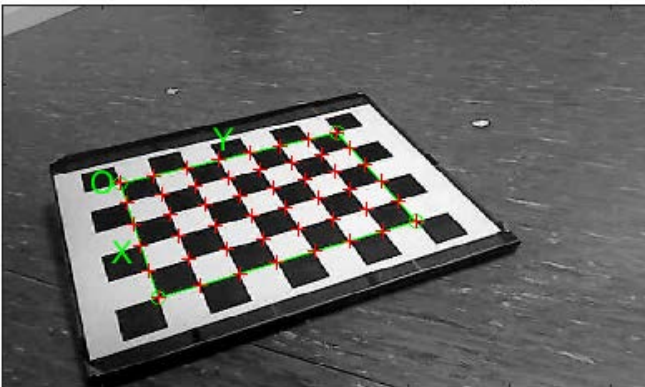
Nos aparecerá una ventana con la primera imagen para que seleccionemos las **4 esquinas** exteriores del patrón para calcular la homografía del patrón en la imagen. La primera esquina será el origen **O** del sistema de coordenadas 3D **(0,0,0)**. Para las 3 esquinas restantes se puede usar cualquier orden mientras sea igual para todas las imágenes. Se recomienda empezar por el extremo largo del patrón y continuar en el mismo sentido para todas las imágenes. En la siguiente imagen se muestra el orden de selección.

Nota: solo se utilizan las esquinas interiores. Las del borde exterior del patrón no deben marcarse (ver la figura siguiente). La posición marcada es aproximada, la aplicación realizará una búsqueda precisa de la esquina en la ventana seleccionada previamente.

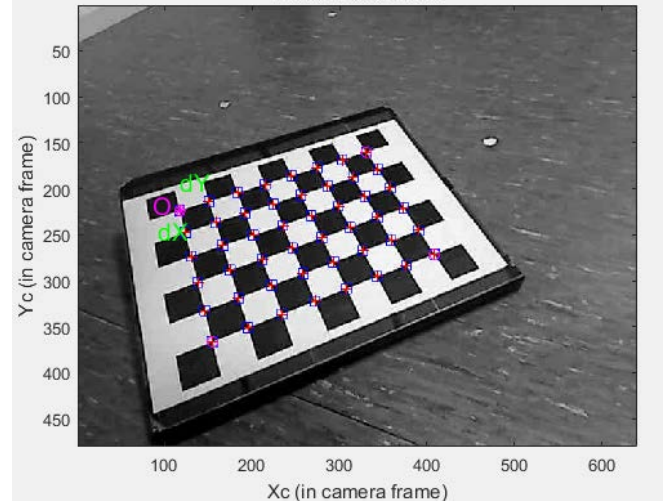


- d. Distancia entre esquinas en el eje **X** (lado corto del patrón). Mediremos el tamaño de un cuadrado del patrón
- e. Distancia entre esquinas en el eje **Y** (lado largo del patrón). Mediremos el tamaño de un cuadrado del patrón
- f. Valor inicial distorsión. A partir de las cuatro esquinas y los datos suministrados se calcula la homografía del patrón de imagen y se muestra en pantalla la estimación de la localización de todas las esquinas del patrón (el error será debido a la distorsión del sistema de captación). Si la aproximación es correcta continuaremos con el valor por defecto.

The red crosses should be close to the image corners



Extracted corners



Command Window

```

Extraction of the grid corners on the images
Number(s) of image(s) to process ([] = all images) =
Window size for corner finder (wintx and winty):
wintx ([] = 5) =
winty ([] = 5) =
Window size = 11x11
Do you want to use the automatic square counting mechanism (0=[]=default)
or do you always want to enter the number of squares manually (1,other)?

Processing image 1...
Using (wintx,winty)=(5,5) - Window size = 11x11      (Note: To reset the window size, run script clearwin)
Click on the four extreme corners of the rectangular complete pattern (the first clicked corner is the origin).
Size dX of each square along the X direction ([]=100mm) = 28
Size dY of each square along the Y direction ([]=100mm) = 28
If the guessed grid corners (red crosses on the image) are not close to the actual corners,
it is necessary to enter an initial guess for the radial distortion factor kc (useful for subpixel detection)
Need of an initial guess for distortion? ([]=no, other=yes)
Corner extraction...

```

Repetiremos el proceso para todas las imágenes, aunque en este caso ya no será necesario volver a indicar de nuevo el tamaño del cuadrado del patrón. Si fallara la detección de esquinas deberemos interrumpir el script y empezar de nuevo.

- 3) **Calibración:** pulsaremos el botón 'Calibration'. De manera alternativa podemos escribir en la consola el comando 'go_calib_optim'

Calculará la estimación de los parámetros intrínsecos de la cámara y la rotación y traslación de cada toma utilizando el algoritmo de Zhang.

```
Command Window
done

Aspect ratio optimized (est_aspect_ratio = 1) -> both components of fc are estimated (DEFAULT).
Principal point optimized (center_optim=1) - (DEFAULT). To reject principal point, set center_optim=0
Skew not optimized (est_alpha=0) - (DEFAULT)
Distortion not fully estimated (defined by the variable est_dist):
    Sixth order distortion not estimated (est_dist(5)=0) - (DEFAULT) .
Initialization of the principal point at the center of the image.
Initialization of the intrinsic parameters using the vanishing points of planar patterns.

Initialization of the intrinsic parameters - Number of images: 13

Calibration parameters after initialization:

Focal Length:      fc = [ 504.71635   504.71635 ]
Principal point:   cc = [ 319.50000   239.50000 ]
Skew:              alpha_c = [ 0.00000 ] => angle of pixel = 90.00000 degrees
Distortion:        kc = [ 0.00000   0.00000   0.00000   0.00000   0.00000 ]

Main calibration optimization procedure - Number of images: 13
Gradient descent iterations: 1...2...3...4...5...6...7...8...9...10...11...12...13...14...15...16...17...18...19...20...21...done
Estimation of uncertainties...done

Calibration results after optimization (with uncertainties):

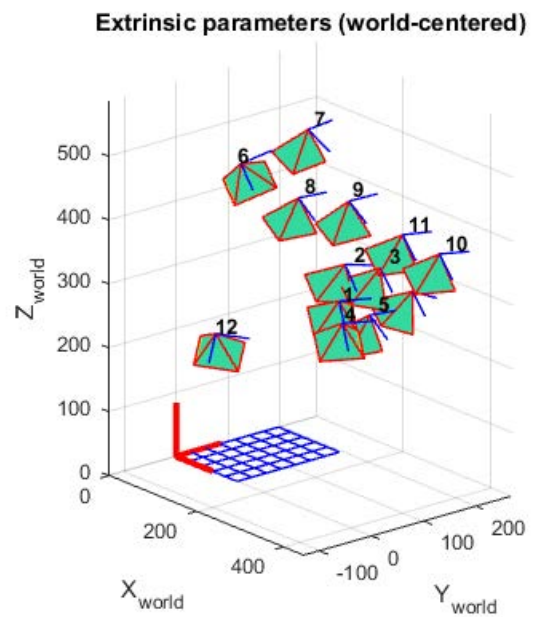
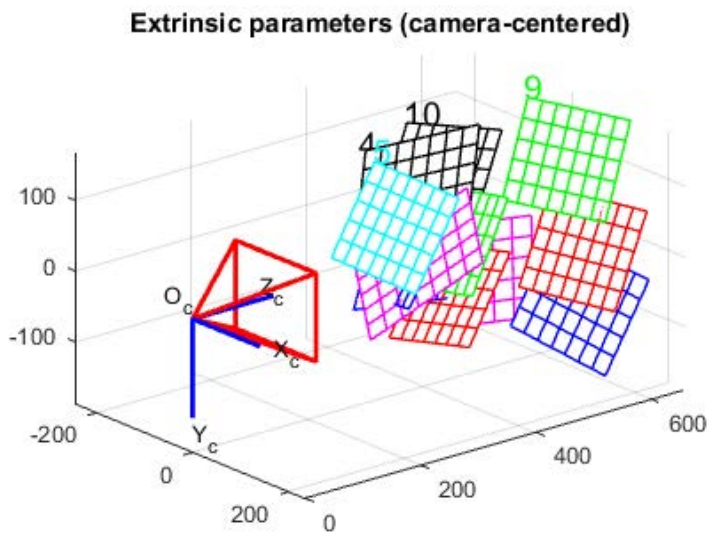
Focal Length:      fc = [ 535.95362   537.35276 ] +/- [ 7.22360   5.22596 ]
Principal point:   cc = [ 299.12515   217.51849 ] +/- [ 4.09906   9.04211 ]
Skew:              alpha_c = [ 0.00000 ] +/- [ 0.00000 ] => angle of pixel axes = 90.00000 +/- 0.00000 degrees
Distortion:        kc = [ 0.07196   -0.18568   -0.00207   -0.00385   0.00000 ] +/- [ 0.01346   0.05244   0.00224   0.00277   0.00000 ]
Pixel error:       err = [ 0.17621   0.20113 ]

Note: The numerical errors are approximately three times the standard deviations (for reference).
```

- 4) **Visualizar Parámetros Extrínsecos:** pulsaremos el botón 'Show Extrinsic'. De manera alternativa podemos escribir en la consola el comando 'ext_calib'

Visualiza la posición de cada una de las tomas del patrón realizadas. Disponemos de dos modos seleccionables con un botón en la esquina inferior derecha.

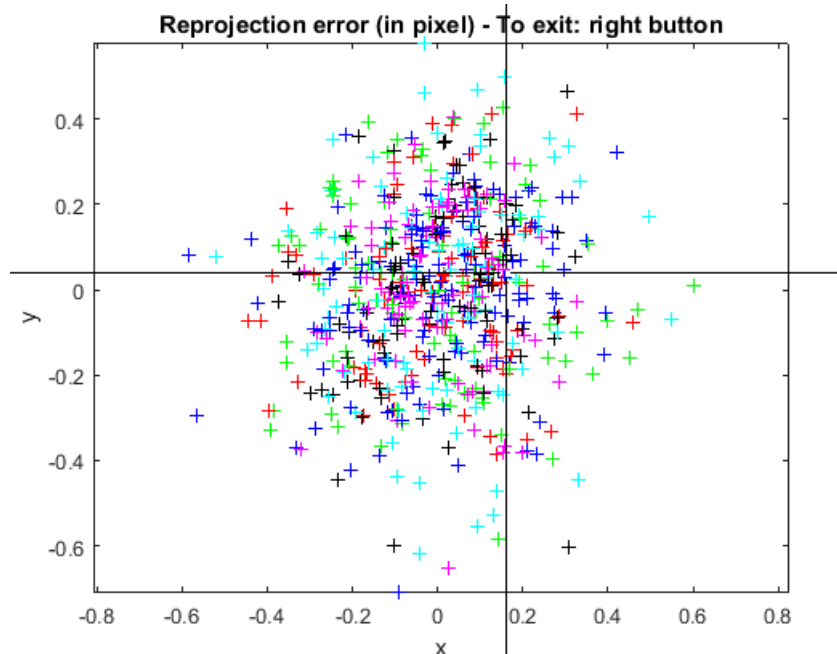
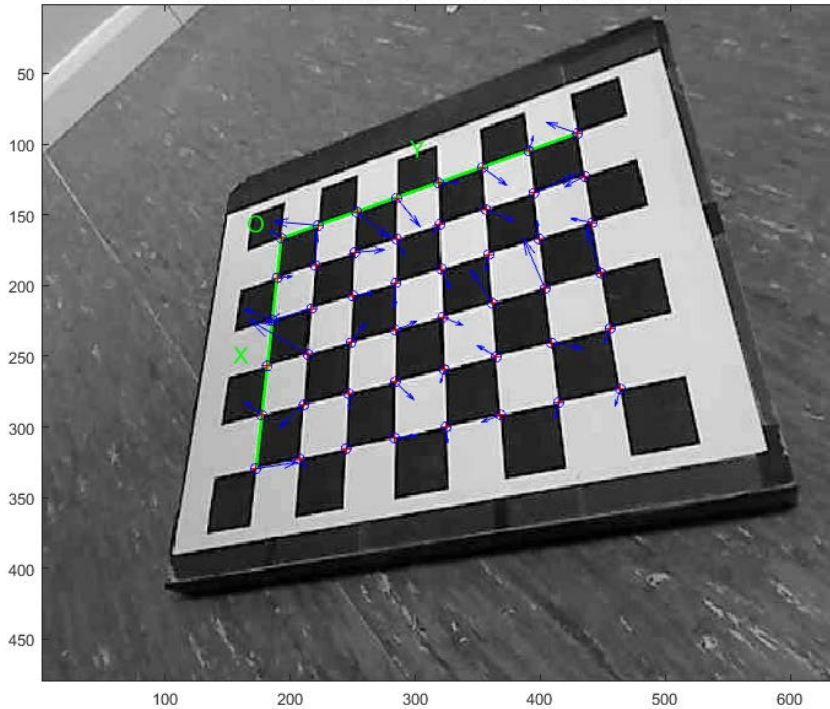
- Centrado en la cámara: la cámara es fija y se mueve el patrón
- Centrado en el patrón. Se mueve la cámara y el patrón es fijo.



- 5) **Visualizar Errores Re-proyección**: pulsaremos el botón **'Reproject on images'**. De manera alternativa podemos escribir en la consola el comando **'reproject_calib'**

Visualiza la diferencia entre las esquinas de la imagen y la proyección de las esquinas del patrón con los datos de calibración estimados (error de re-proyección). Pulsando el botón **'Analyse error'**, nos permite ver dos detalles de cada error (imagen, esquina,...)

Image 13 - Image points (+) and reprojected grid points (o)



```

Selected image: 12
Selected point index: 22
Pattern coordinates (in units of (dX,dY)): (X,Y)=(3,4)
Image coordinates (in pixel): (385.55,207.30)
Pixel error = (-0.15568,0.14645)
Window size: (wintx,winty) = (5,5)

```

- 6) **Eliminar imágenes de calibración:** Si alguna imagen da un error elevado, pulsaremos el botón 'Add/Supress Images'. De manera alternativa podemos escribir en la consola el comando 'add_supress'

Indicaremos la opción '0' (suprimir) y a continuación, entre corchetes, los índices de las imágenes que deseamos eliminar. Volveremos a ejecutar la calibración con el comando go_calib_optim.

```
>> add_suppress

This function is useful to select a subset of images to calibrate

There are currently 13 active images selected for calibration (out of 15):
1, 2, 3, 5, 6, 7, 8, 9, 11, 12, 13, 14 and 15.

Do you want to suppress or add images from that list?
For suppressing images enter 0, for adding images enter 1 ([]=no change): 0
Number(s) of image(s) to suppress ([] = no image) = [2 13]

There is now a total of 11 active images for calibration:
1, 3, 5, 6, 7, 8, 9, 11, 12, 14 and 15.

You may now run 'Calibration' to recalibrate based on this new set of images.
>> go_calib_optim
```

- 7) **Guardar Datos Calibración:** pulsaremos el botón 'Save'. De manera alternativa podemos escribir en la consola el comando 'saving_calib'

Se almacenarán los datos de calibración (parámetros Intrínsecos) con su error de estimación, así como los parámetros extrínsecos de cada toma. Se almacenan en el fichero **Calib_Results** tanto formato **.mat** y en formato **.m**. Si Utilizas la versión propia de la librería, también se almacenan en formato **YAML**.

Para el resto de opciones de la librería se recomienda consultar el manual proporcionado por el autor.

5. Documentos generados y entregados

La realización de la práctica debe realizarse de forma individual. Es necesario entregar **un documento en el que se describa el trabajo realizado** y que incluya los resultados relacionados a continuación, **así como los ficheros de calibración obtenidos**. Se entregará un zip, en el que se indique en nombre del alumno. Por favor, se ruega no mandar las imágenes de entrada, para que el fichero no sea demasiado grande.

Resultados:

- Parámetros de calibración
- Visualización error de re-proyección
- Visualización de parámetros extrínsecos.