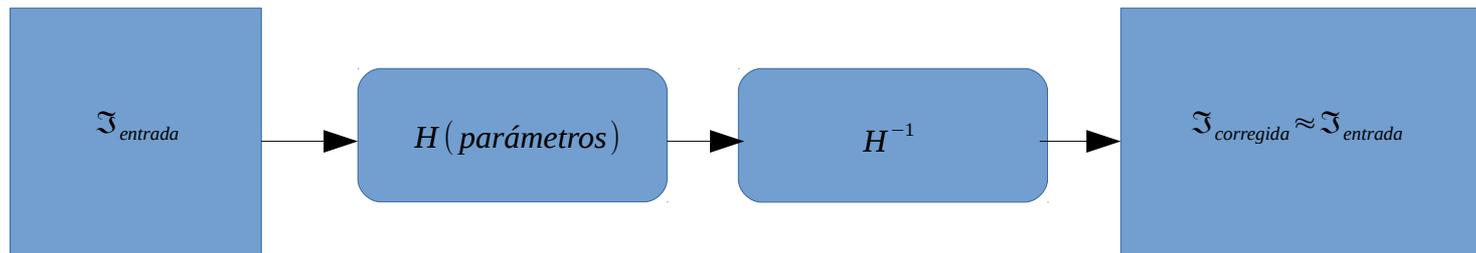


# Restauración de imágenes

- Tenemos una o más imágenes
- Un modelo de la degradación.
- Suponemos la degradación estable en el tiempo.
- Se busca identificar los parámetros del modelo a fin de corregir la distorsión producida.

# Idea general

- Se supone cierta degradación cuyo modelo es conocido y estable en el tiempo.
- Se trata de estimar los parámetros de dicho modelo a fin de revertir su efecto e intentar “recuperar” la imagen original.



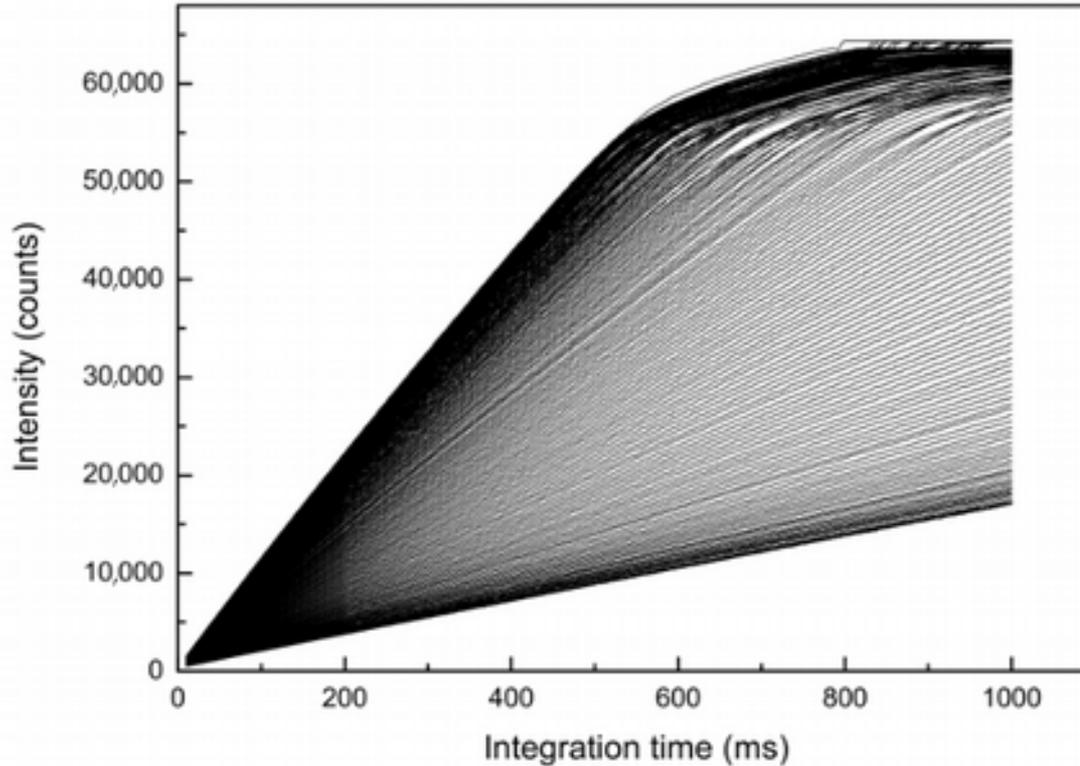
# Ejemplos

- Iluminación no uniforme
- Aberración óptica
- Calibración de cámaras

# Iluminación no uniforme

- En el CCD: Variaciones en la respuesta de cada pixel a la iluminación.
- Se supone que la respuesta de un pixel a la iluminación es uniforme. En la práctica esto es aproximado y hay errores de menos del 10%.
- Suponiendo esa respuesta lineal, sucede que la respuesta de cada pixel en un sensor es diferente.
- Función lineal caracterizada por 2 parámetros por pixel.

# Iluminación no uniforme



Cada traza es la respuesta de un pixel. El comportamiento es lineal hasta unos 50000 counts.

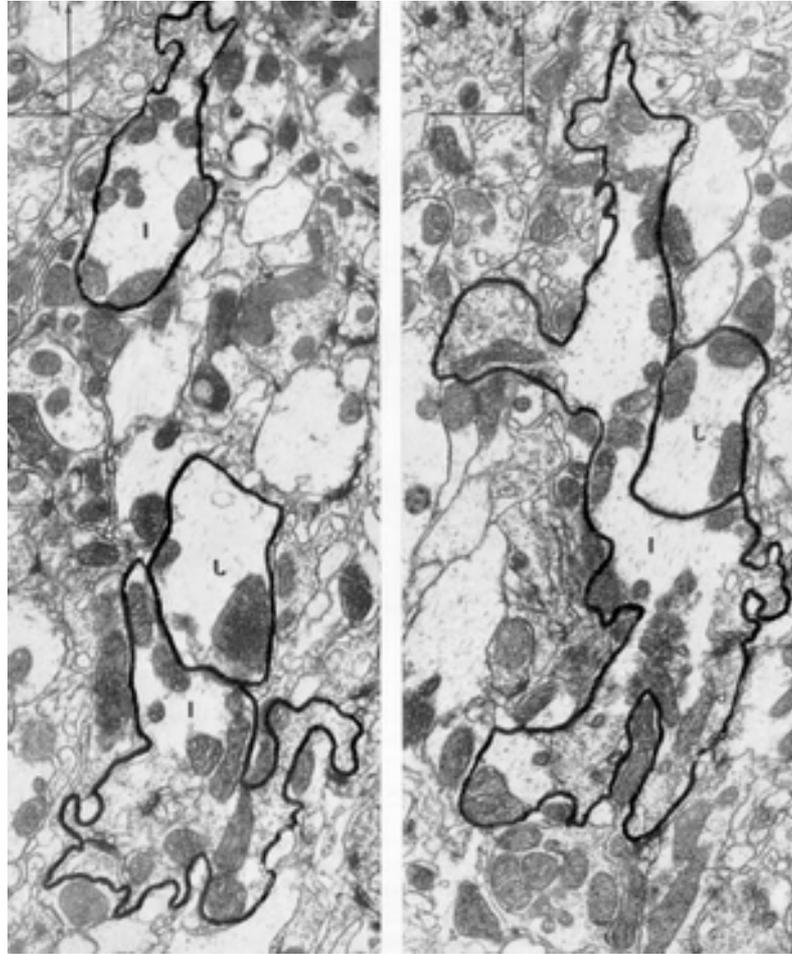
De: Improving Optical Measurements: Non-Linearity Compensation of Compact Charge-Coupled Device (CCD) Spectrometers, Nehir et al. Sensors (Basel), 2019 Jun 25;19(12):2833.doi: 10.3390/s19122833

# Método de corrección

- Iluminar el CCD con luz uniforme controlada  
[https://en.wikipedia.org/wiki/Flat-field\\_correction](https://en.wikipedia.org/wiki/Flat-field_correction)
- Necesitamos dos valores para definir una recta  $y=mx+b$ .
- Iluminamos el sensor con una iluminación uniforme y la imagen negra (dark frame) correspondiente a la corriente de señal para no iluminación
- Corrección:
  - C (matriz) imagen corregida
  - R (matriz) imagen RAW
  - F (matriz) Flat field image
  - D (matriz) dark current image
  - m (escalar) valor medio de la imagen (F-D)

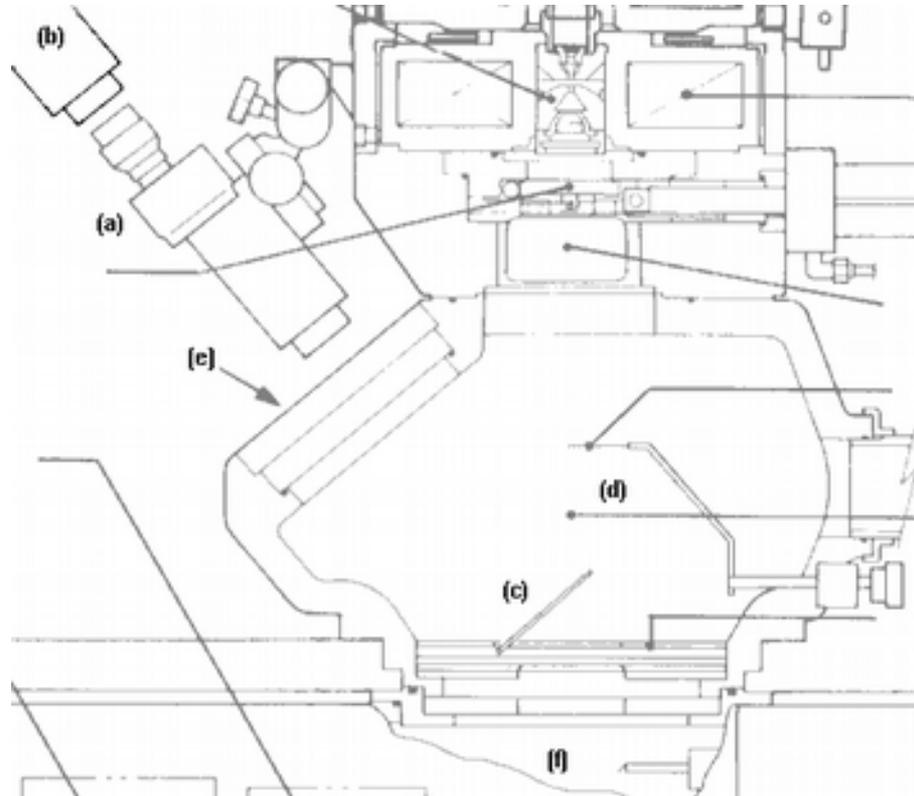
$$c = \frac{(R-D)m}{F-D}$$

# Neuro3D



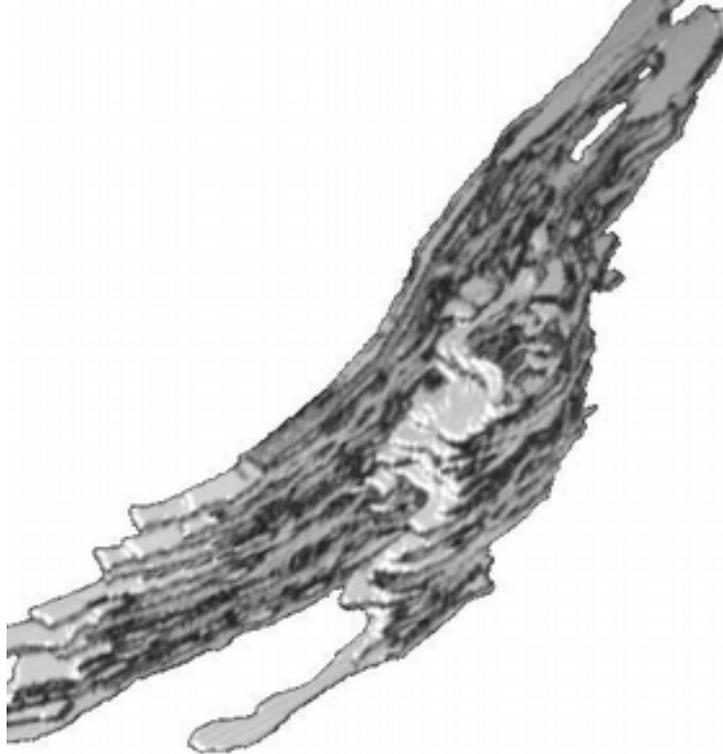
Neuro3d: Software de Reconstrucción Tridimensional de Neuronas. Randall et al, CACIC 97.

# Iluminación no uniforme. Neuro3D



Neuro3d: Software de Reconstrucción Tridimensional de Neuronas. Randall et al, CACIC 97.

# Neuro3D



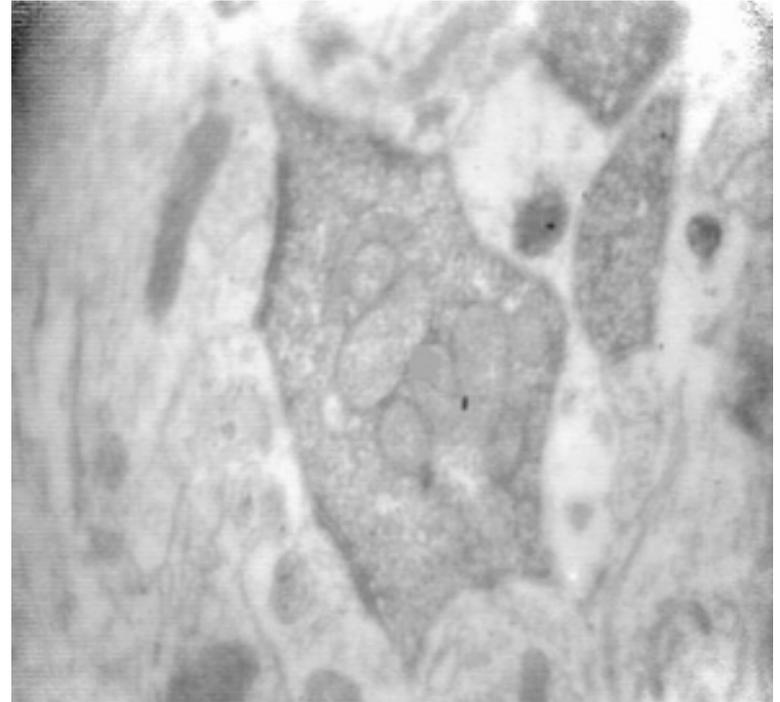
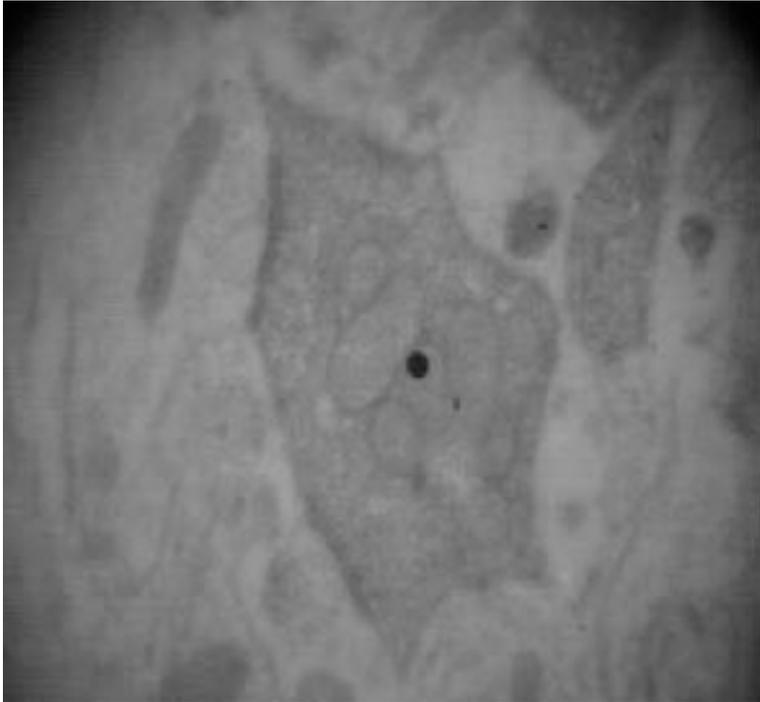
Neuro3d: Software de Reconstrucción Tridimensional de Neuronas. Randall et al, CACIC 97.

# Iluminación no uniforme.

## Neuro3d

- Iluminación media no uniforme: intensa en el centro y bajando hacia los bordes.
- Sobre ese valor medio una señal útil
- Estimar la función de la no uniformidad: una superficie de revolución de la que hay que determinar el centro y la forma.
- Conocida la función se pueden estimar los parámetros mediante mínimos cuadrados tomando  $n$  puntos en la imagen observada (valor medio en un patch suficientemente grande).
- La imagen corregida es la original dividida por la función estimada y una ecualización de histograma.

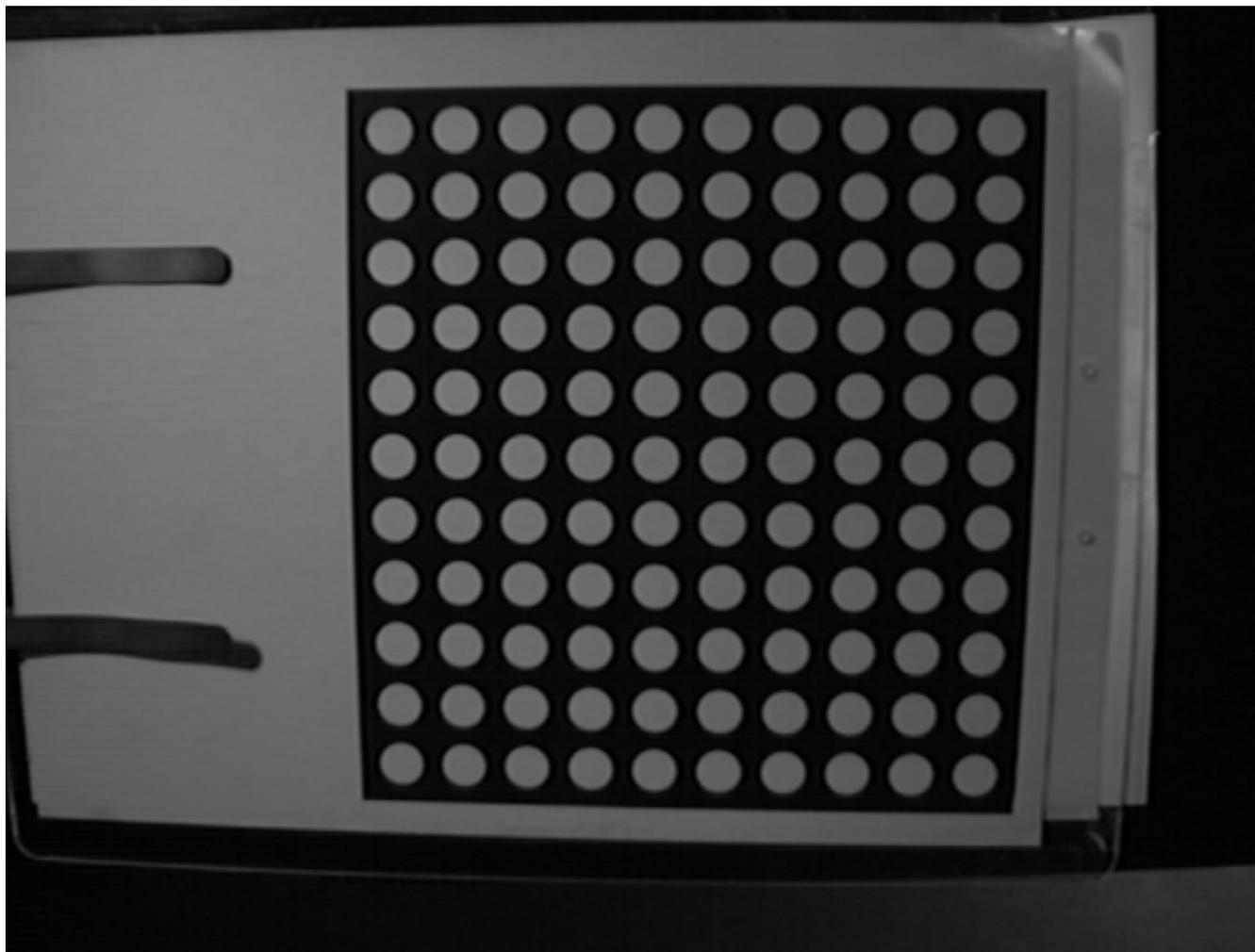
# Iluminación no uniforme. Neuro3D



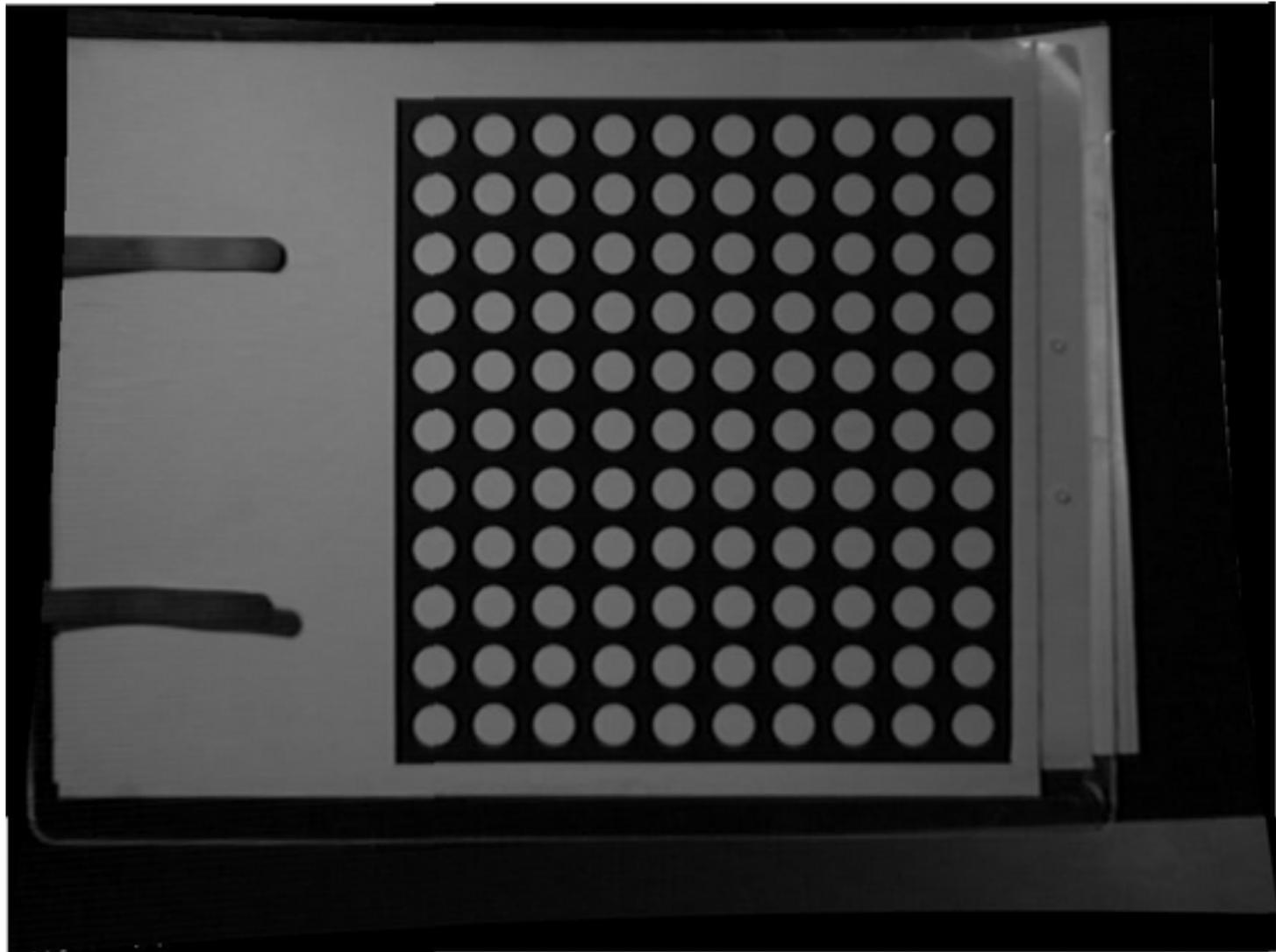
# Calibración de cámaras

- Se observa una imagen conocida. Generalmente un patrón de calibración cuidadosamente construido.
- La imagen adquirida puede ponerla en relación con la que se fue observada para determinar los parámetros de la degradación.
- Con los parámetros del modelo de degradación aplico la transformación inversa y genero una imagen reconstruida que debe ser similar a la observada.

# Imagen original



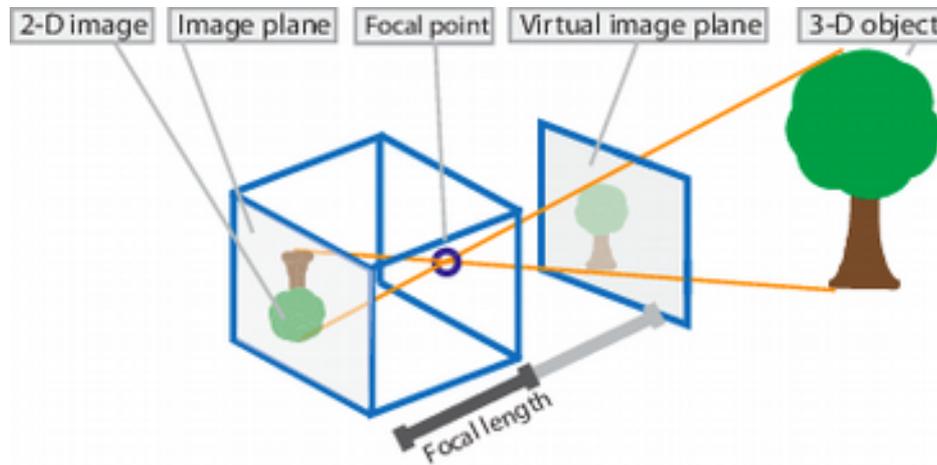
# Inversión del modelo completo



# Modelo Pin Hole

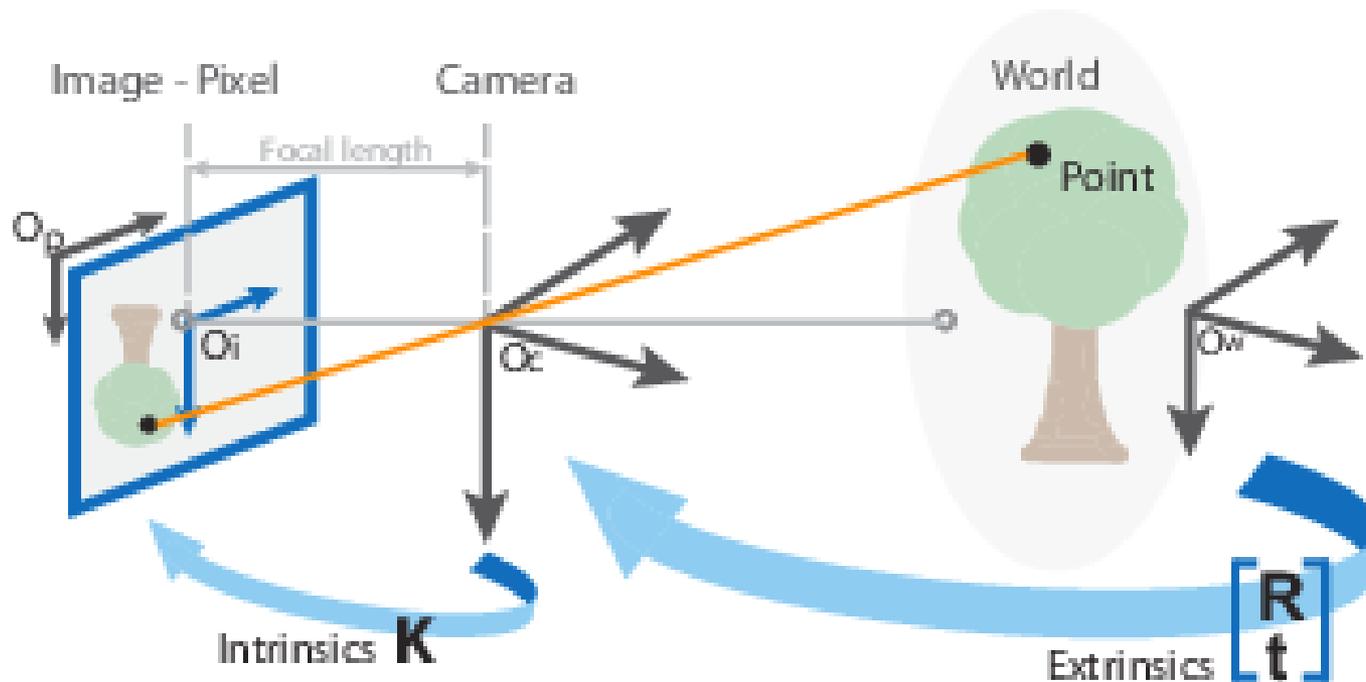
- Dos referenciales:
  - Rabs: mundo de los objetos
  - Rcam: mundo de la cámara.
- Se aproxima la lente de la cámara. Por ejemplo, una aproximación de primer orden.
- La cámara es una transformación rígida compuesta por una proyección geométrica y una discretización espacial.

# Modelo Pin Hole

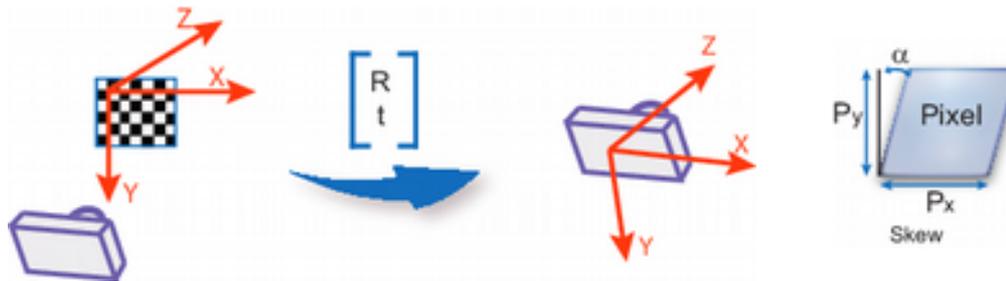
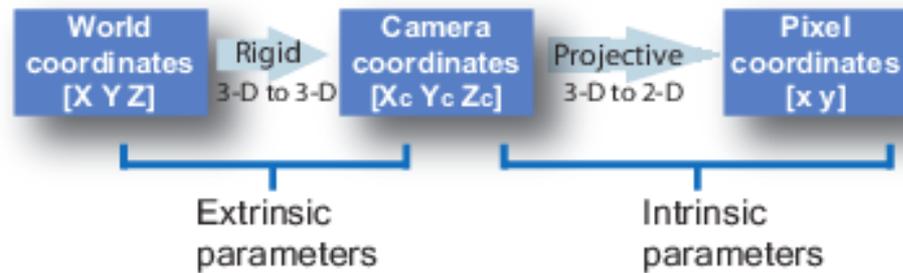


<https://www.mathworks.com/help/vision/ug/camera-calibration.html>

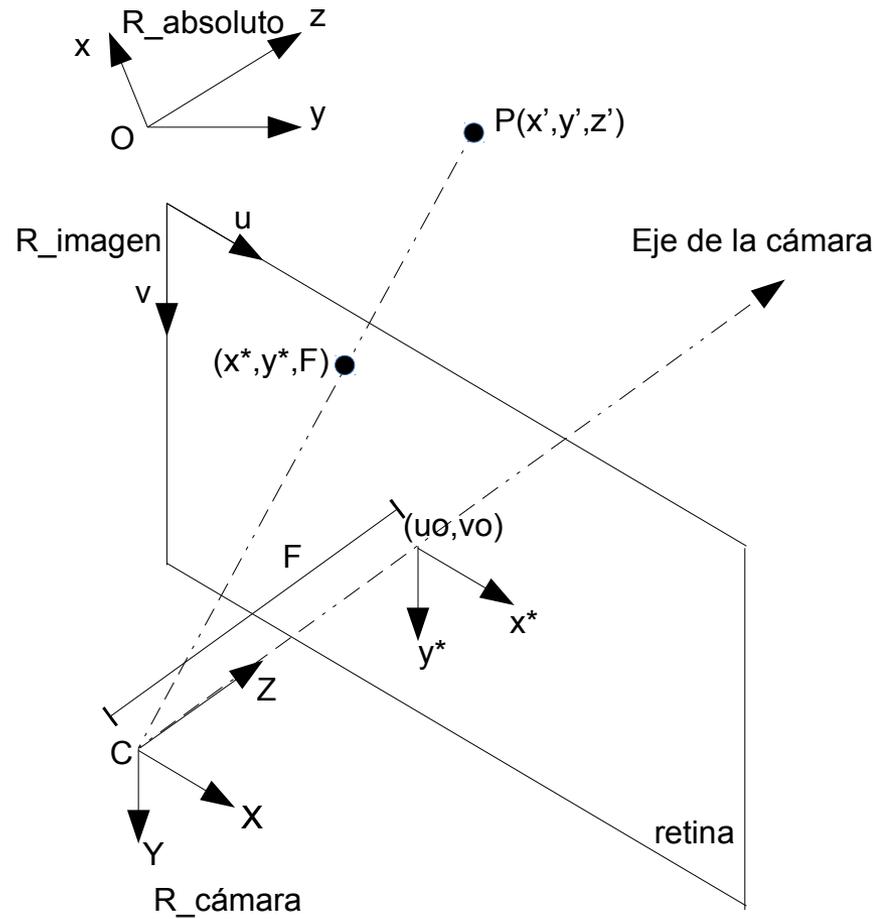
# Parametros extrínsecos e intrínsecos



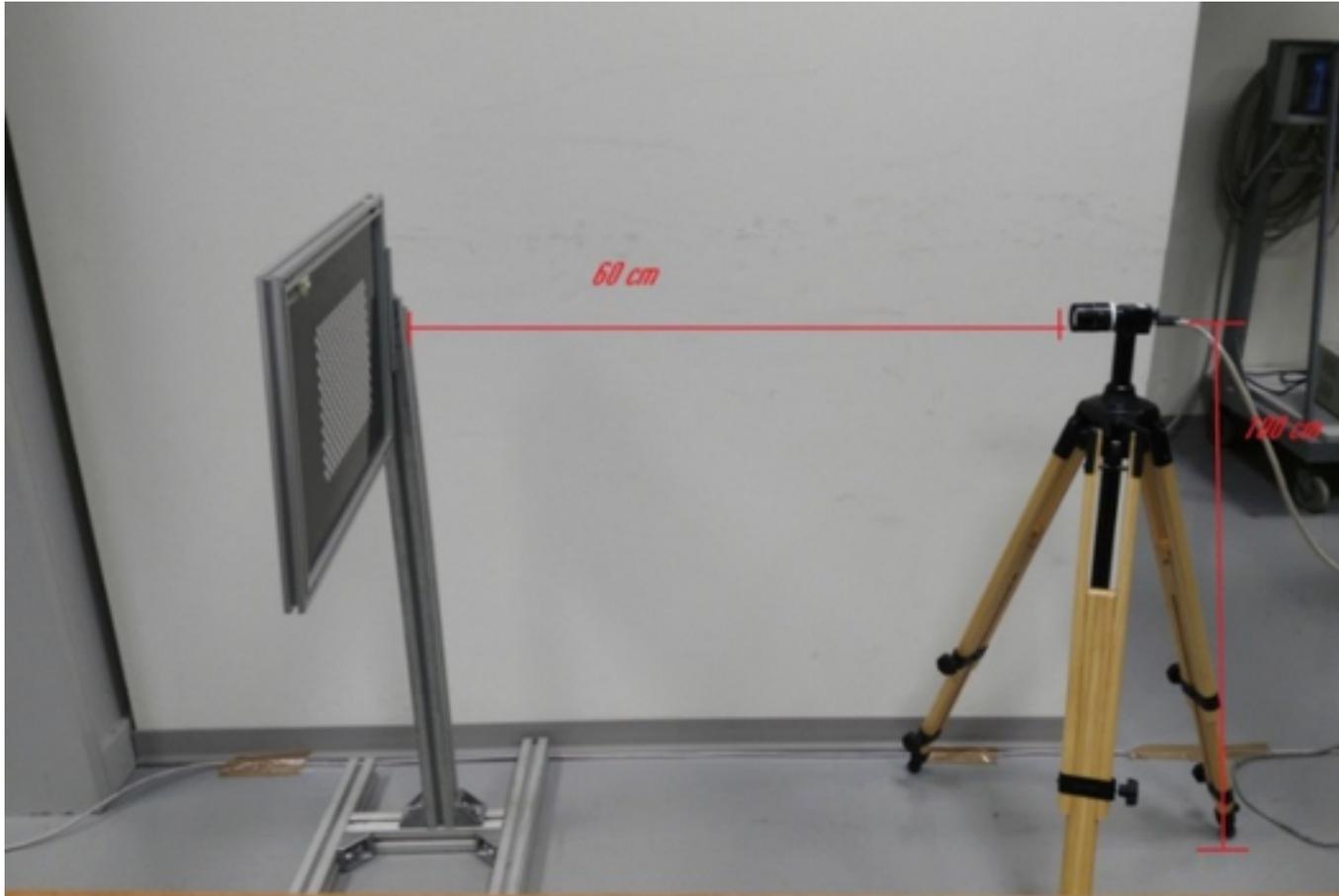
# Del mundo 3d al plano imagen.



# Modelo simplificado de la cámara



# Montaje experimental



Review of techniques for 2D camera calibration suitable for industrial vision systems.  
G D'Emilia and D Di Gasbarro 2017 J. Phys.: Conf. Ser. 841 012030

# Formulación

- Movimiento rígido en coordenadas homogéneas, de R a Rcam.
- Todos los objetos referidos al referencial de la cámara.

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} R_{3 \times 3} & t_{1 \times 3} \\ 0_{3 \times 1} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

# Formulación

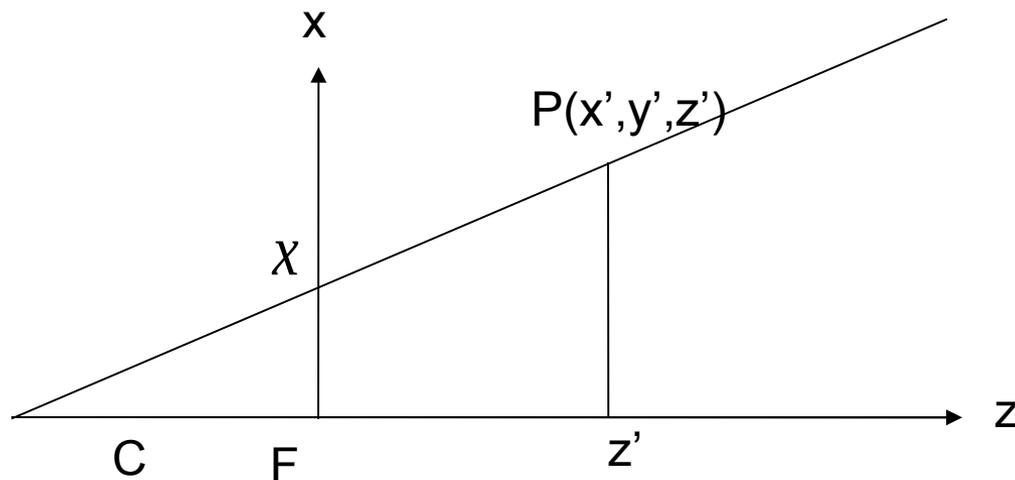
- Proyección. Del mundo 3D al 2D.

$$\begin{bmatrix} x_M \\ y_M \\ s \end{bmatrix} = [P] \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} \quad P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/F & 0 \end{bmatrix}$$

# Formulación

- La matriz de proyección  $P$  depende de la distancia focal  $F$ , entre el plano imagen y el foco  $C$ .

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} x_M / S \\ y_M / S \end{bmatrix}$$



$$\frac{x}{F} = \frac{x'}{z'}$$
$$x = \frac{F x'}{z'}$$

# Discretización de la imagen

- Retina rectangular de pasos  $K_v$  y  $K_u$  en las direcciones vertical y horizontal
- Traslación  $(U_0, V_0)$  para referir todos los puntos al origen en el referencial de la imagen.
- El punto entonces queda:

$$u = u_0 + k_u x''$$
$$v = v_0 + k_v y''$$

# discretización y proyección

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} K_u & 0 & u_0 \\ 0 & K_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/F & 0 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z \\ 1 \end{bmatrix}$$

$$sF \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} K_u F & 0 & u_0 & 0 \\ 0 & K_v F & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix}$$

$$a_u = K_u F$$

$$a_v = K_v F$$

# Parámetros extrínsecos e intrínsecos

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} a_u & 0 & u_0 & 0 \\ 0 & a_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

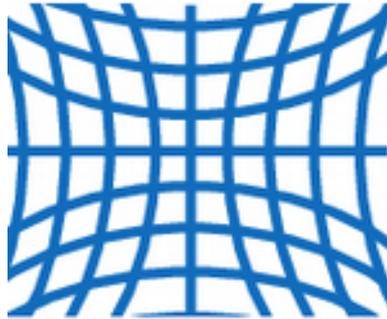
Los parámetros extrínsecos son  $\theta_x, \theta_y, \theta_z$  que definen la matriz de rotación  $R$  y el vector  $[t_x, t_y, t_z]^T$  que definen la traslación

Los parámetros intrínsecos son  $a_u$  y  $a_v$  que contienen la distancia focal y el paso de discretización de manera combinada y el centro de la imagen  $u_0$  y  $v_0$ .

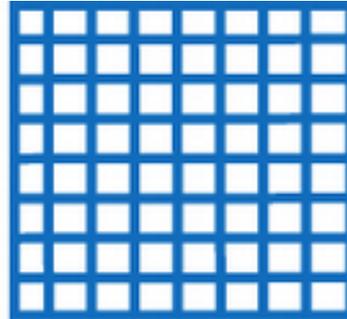
# Se puede refinar el modelo

- Aberración óptica: Un “punto objeto” va a  $N$  puntos en la imagen. Es decir que cada punto de la imagen depende de  $N$  vecinos en el “punto objeto”.
- Distorsión óptica: Suponiendo correspondencia punto a punto. Distorsión radial a partir del eje de la lente.
- Más general: distorsiones locales.

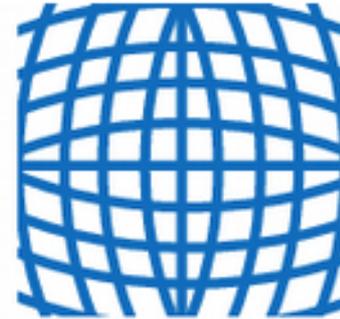
# Distorsión óptica



Negative radial distortion  
"pincushion"

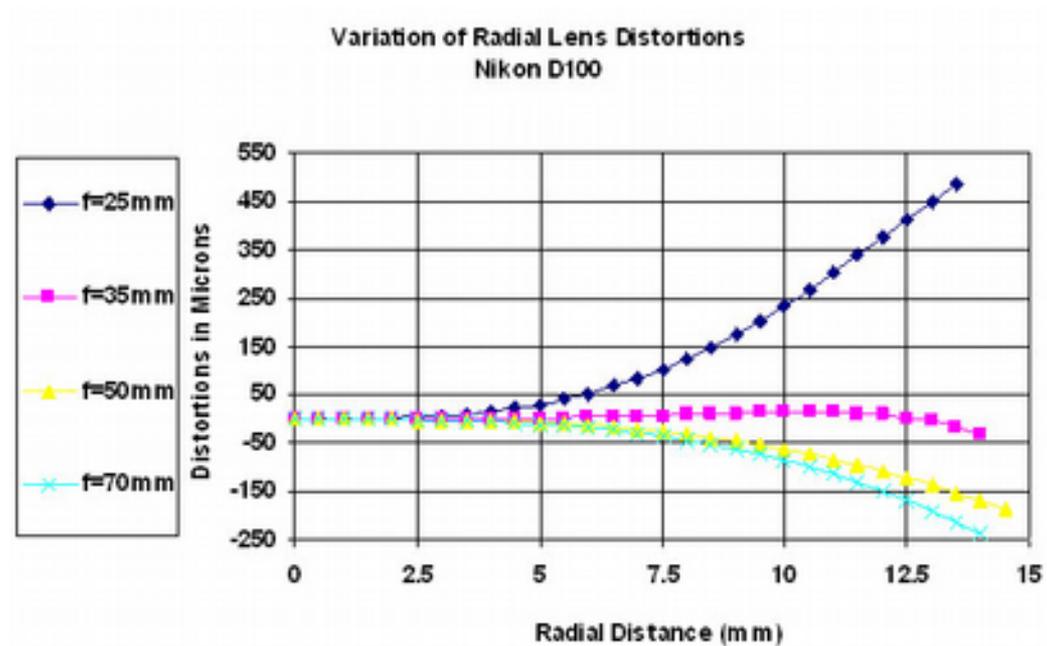


No distortion



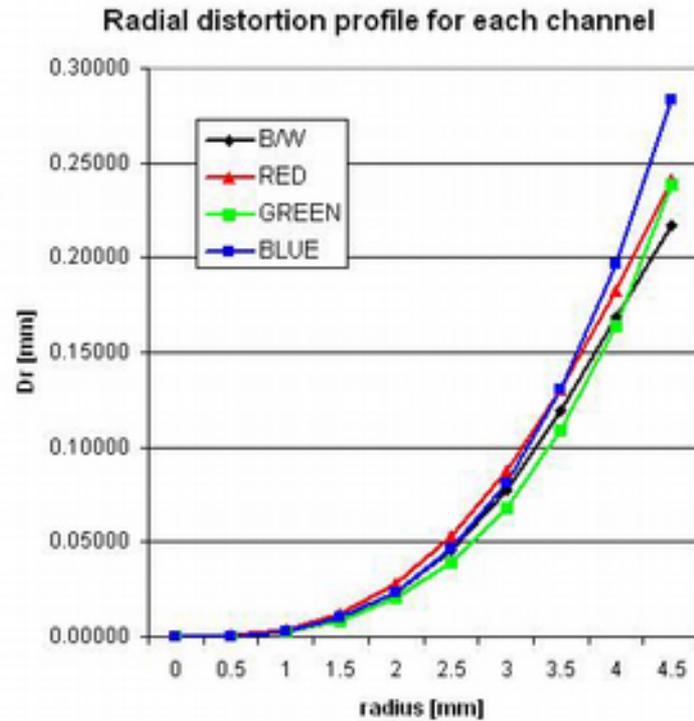
Positive radial distortion  
"barrel"

# Distorsión radial



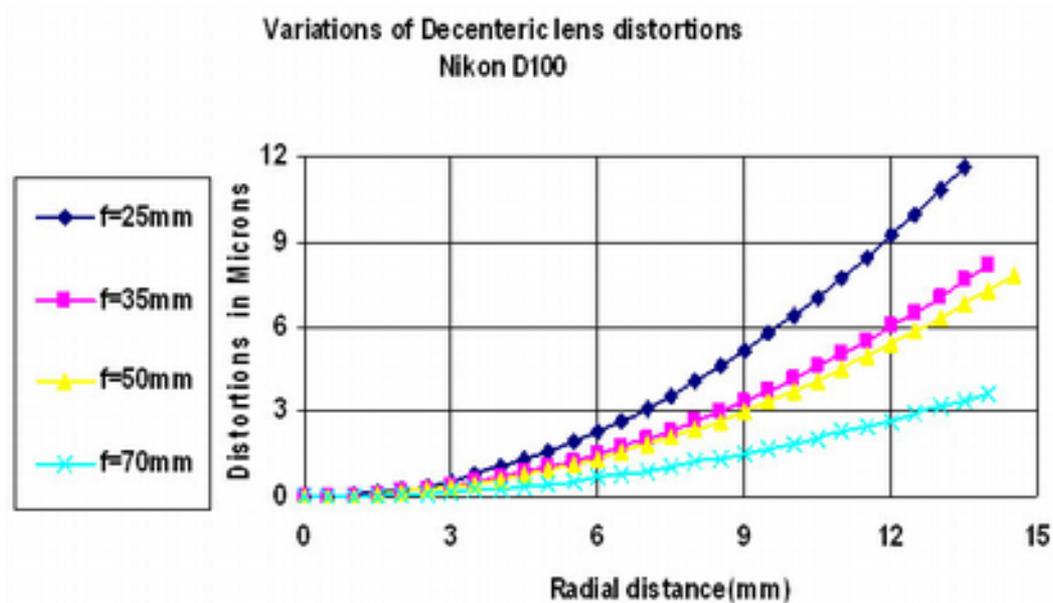
"Digital camera calibration methods: considerations and comparisons" F. Remondino y C. Fraser.  
ISPRS Commission V Symposium 'Image Engineering and Vision Metrology'

# Distorsión radial



“Digital camera calibration methods: considerations and comparisons” F. Remondino y C. Fraser.  
ISPRS Commission V Symposium 'Image Engineering and Vision Metrology'

# Distorsión por descentraje



"Digital camera calibration methods: considerations and comparisons" F. Remondino y C. Fraser.  
ISPRS Commission V Symposium 'Image Engineering and Vision Metrology'

# Distorsión radial

- Las coordenadas  $x$  e  $y$  normalizadas, trasladadas al centro imagen y divididas por la distancia focal

$$x_d = x(1 + \alpha_1 r^2 + \alpha_2 r^4 + \alpha_3 r^6)$$

$$y_d = y(1 + \alpha_1 r^2 + \alpha_2 r^4 + \alpha_3 r^6)$$

$$r = \sqrt{(x^2 + y^2)}$$

# Descentraje del lente

$$x^d = x'' + \beta(3x''^2 + y''^2) + 2\gamma x'' y''$$

$$y^d = y'' + \gamma(x''^2 + 3y''^2) + 2\beta x'' y''$$

# Orden de las transformaciones

- Para obtener la relación entre punto observado y real se deben componer las transformaciones de la siguiente manera:
  - Proyección central a través de la matriz  $P$
  - Distorsión radial o con descentraje
  - discretización

# Calibración

Conociendo la posición de  $n$  puntos de un patrón y la observación de los mismos es posible estimar los parámetros del modelo.

Patrones: coplanares o no, círculos, cuadrados, etc.

Estimar los parámetros es un problema de minimización bajo restricciones.

A veces se desacopla el problema para simplificarlo. Por ejemplo estimar los parámetros intrínsecos y los extrínsecos.

# Patrones de calibración



<https://calib.io/blogs/knowledge-base/calibration-patterns-explained>

# Construcción del patrón de calibración

Es un tema complejo.

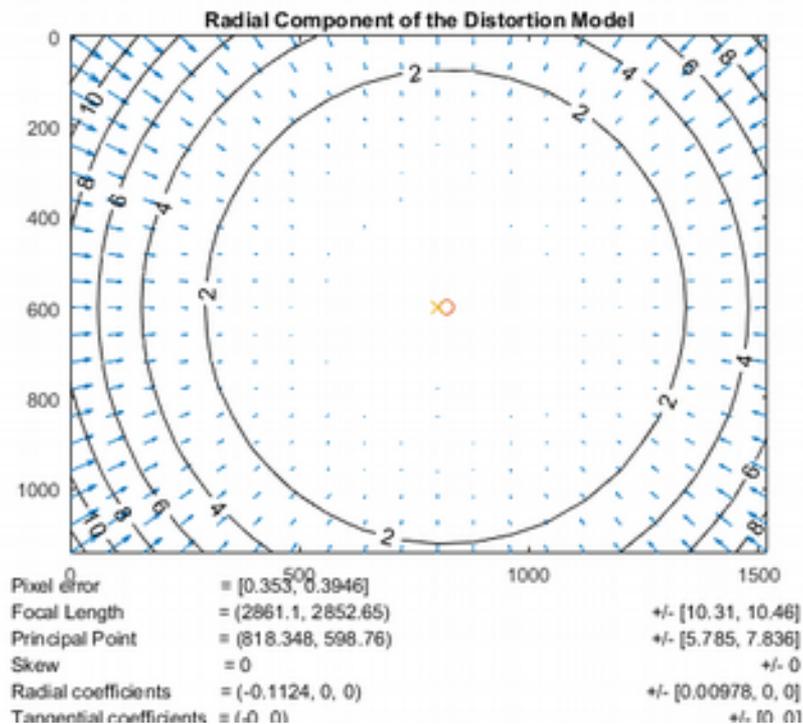
El diseño debe garantizar:

- Detección de los objetos con gran precisión: círculos, elipses, corner detectors, etc.
- Un número de puntos suficientemente grande para que el error sea pequeño.

La construcción debe garantizar:

- La fidelidad al diseño: impresión, mecánica.

# Resultados del proceso de calibración



# Referencias

[https://boofcv.org/index.php?title=Tutorial\\_Camera\\_Calibration](https://boofcv.org/index.php?title=Tutorial_Camera_Calibration)

[https://docs.opencv.org/2.4/doc/tutorials/calib3d/camera\\_calibration/camera\\_calibration.html](https://docs.opencv.org/2.4/doc/tutorials/calib3d/camera_calibration/camera_calibration.html)

## Literatura sobre 3d pero con elementos de calibración:

Three-Dimensional Computer Vision, A Geometric Viewpoint, Olivier Faugeras.

Una versión muy reducida en:

[https://www.researchgate.net/publication/30869787\\_Three-dimensional\\_computer\\_vision\\_a\\_geometric\\_viewpoint](https://www.researchgate.net/publication/30869787_Three-dimensional_computer_vision_a_geometric_viewpoint)

Multiple View Geometry in Computer Vision by Richard Hartley (Author), Andrew Zisserman