

HELA2018, Instituto Tecnológico de Hermosillo

TALLER DE PROCESAMIENTO DIGITAL DE IMÁGENES

Por M. En R. Karen Flores, M. En TA. Sandra de la Fuente, e Ing. Ángel Hernández.



INVESTIGACIÓN Y VINCULACIÓN

Proyectos de valor agregado
en el IPN CICATA Unidad Querétaro

WWW.CICATAQRO.IPN.MX



LÍNEAS DE
INVESTIGACIÓN

CONTENIDO

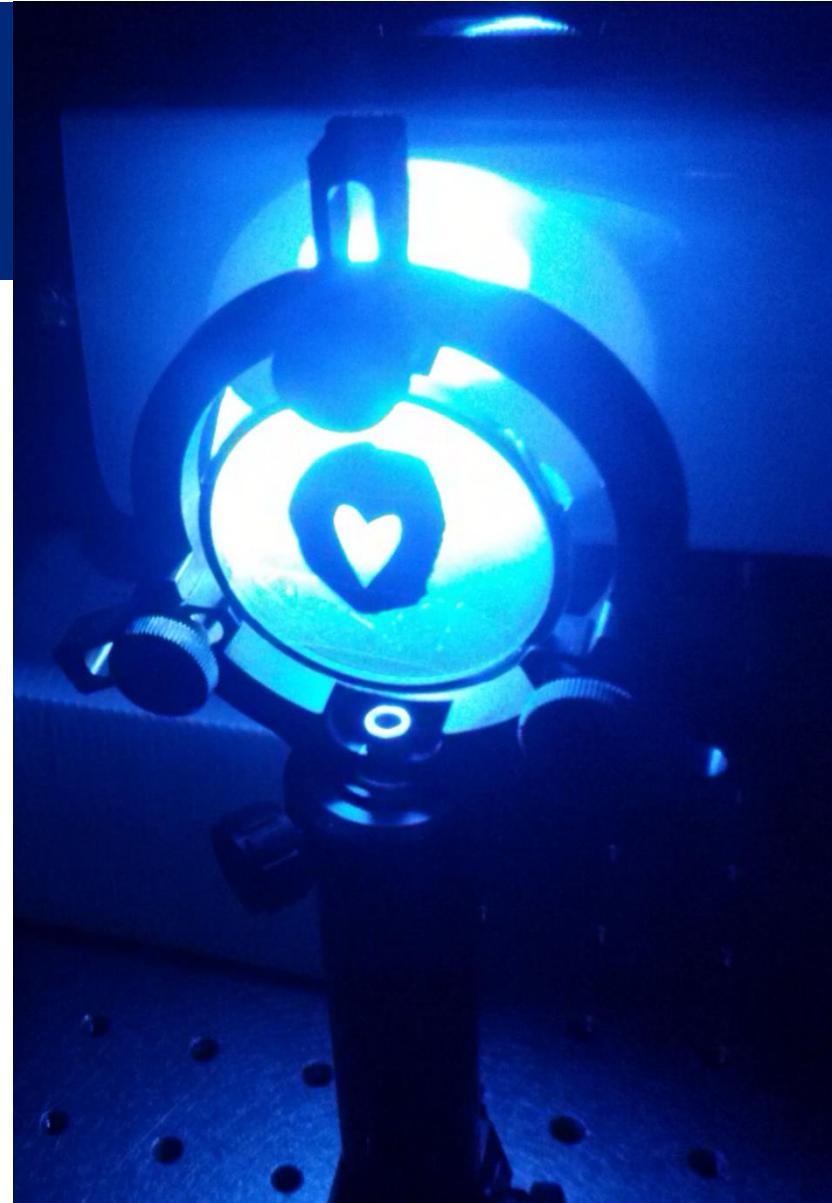
Introducción

Sistemas ópticos

Calibración de cámara

Introducción a las imágenes digitales

Procesamiento digital de imágenes



Taller de Procesamiento Digital de Imágenes









Para la formación de una imagen se necesita:

Objeto

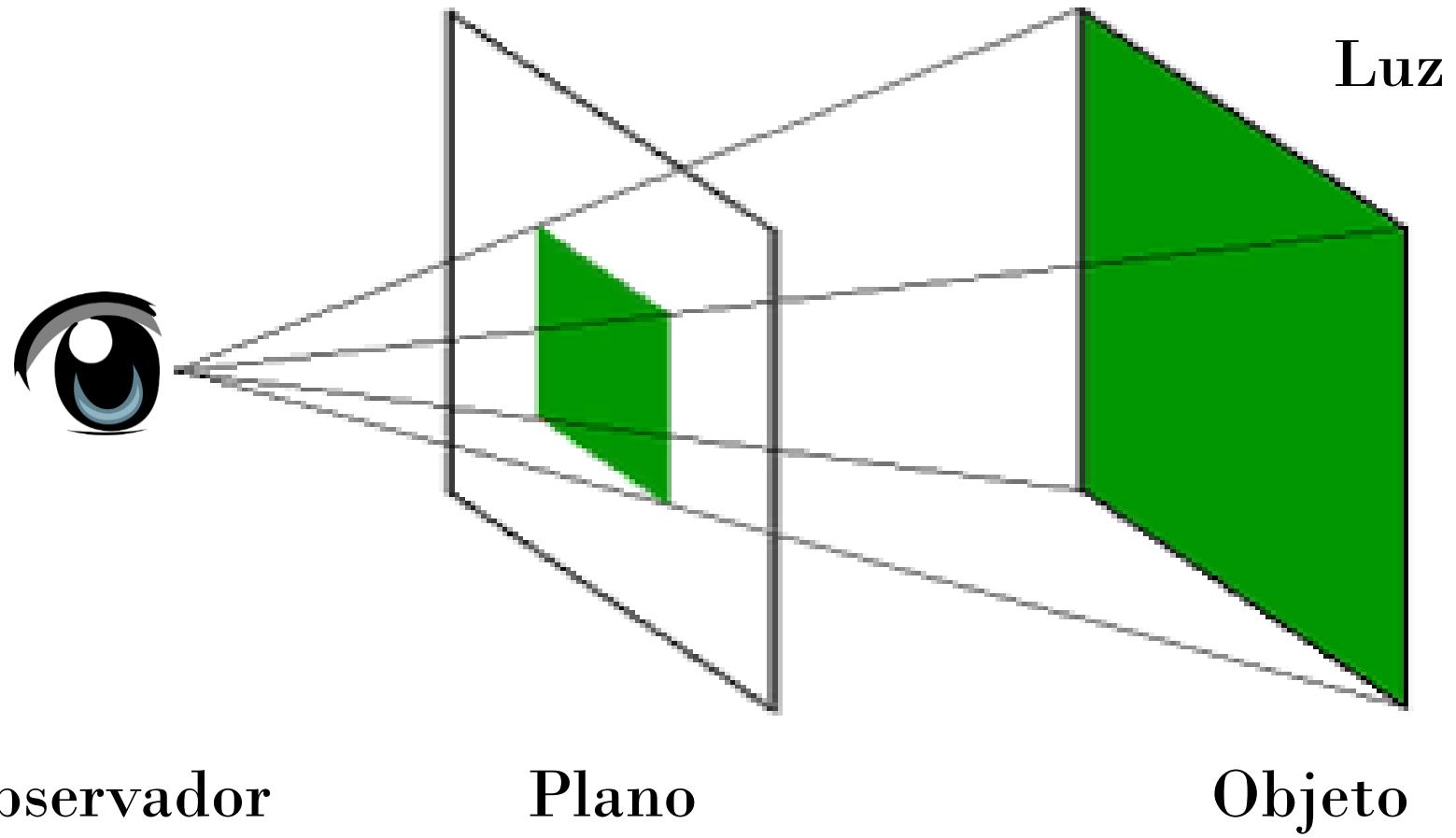
Objeto del cual se generará la imagen.

Fuente de luz

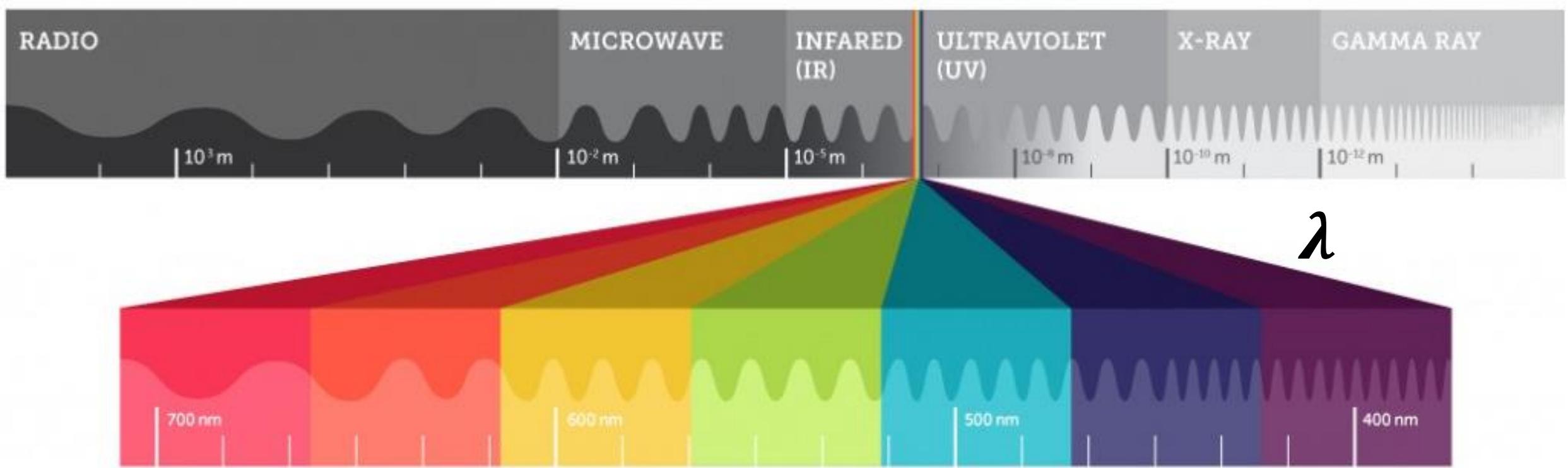
Para iluminar la superficie del objeto en el medio.

Plano

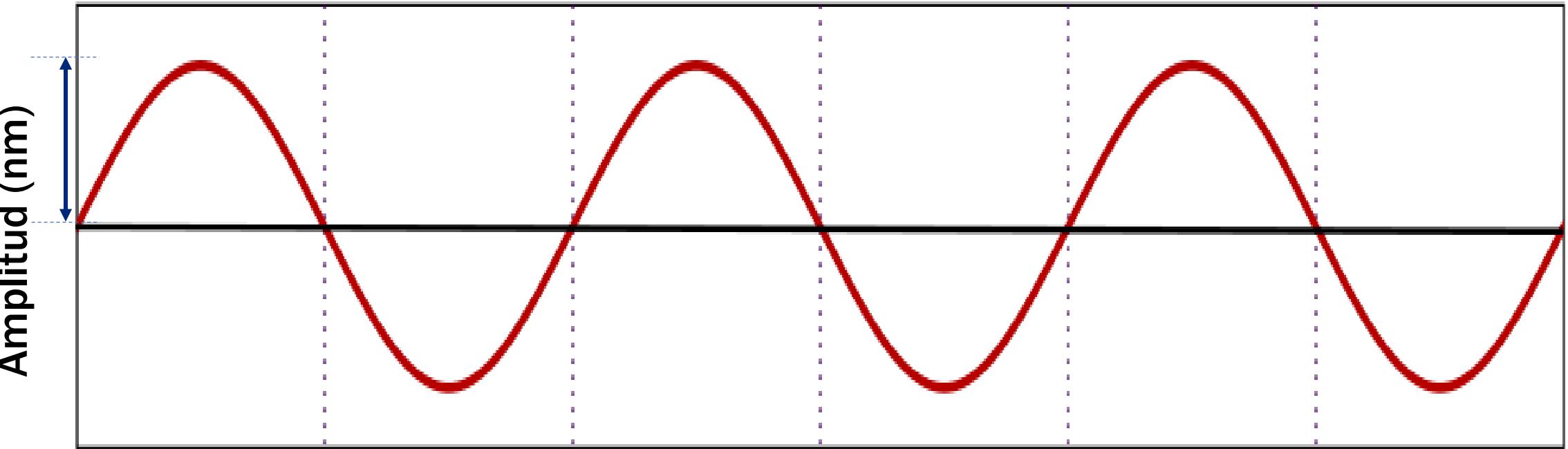
Plano de proyección de la imagen.



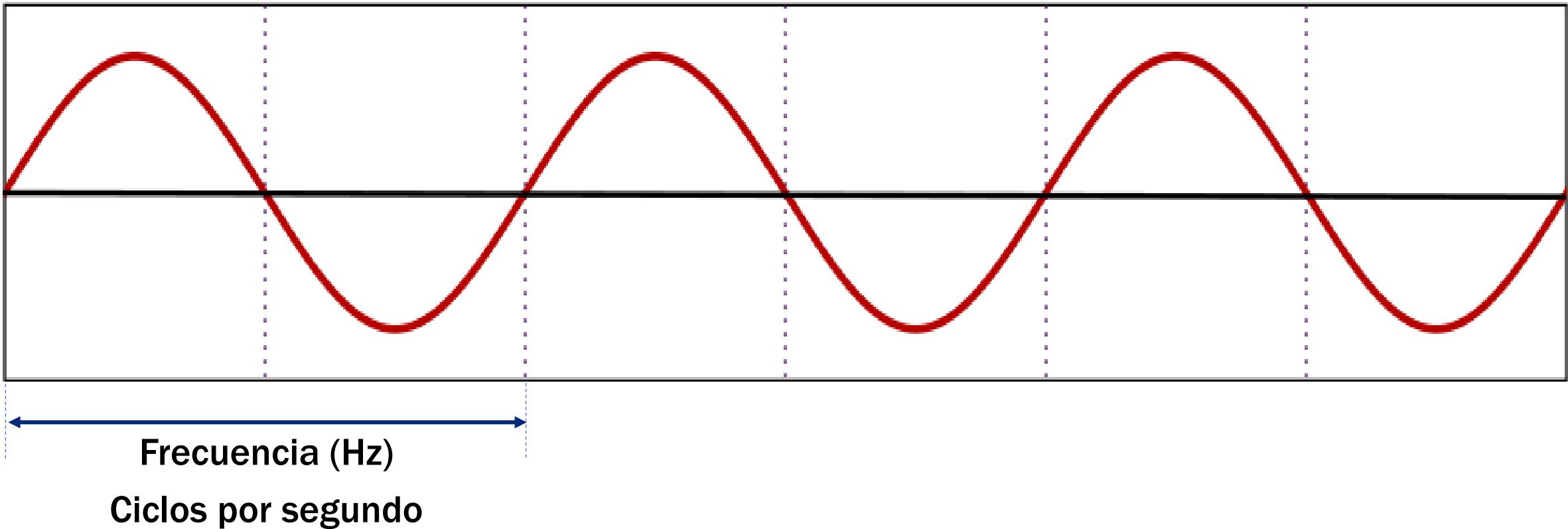
FUENTE DE LUZ



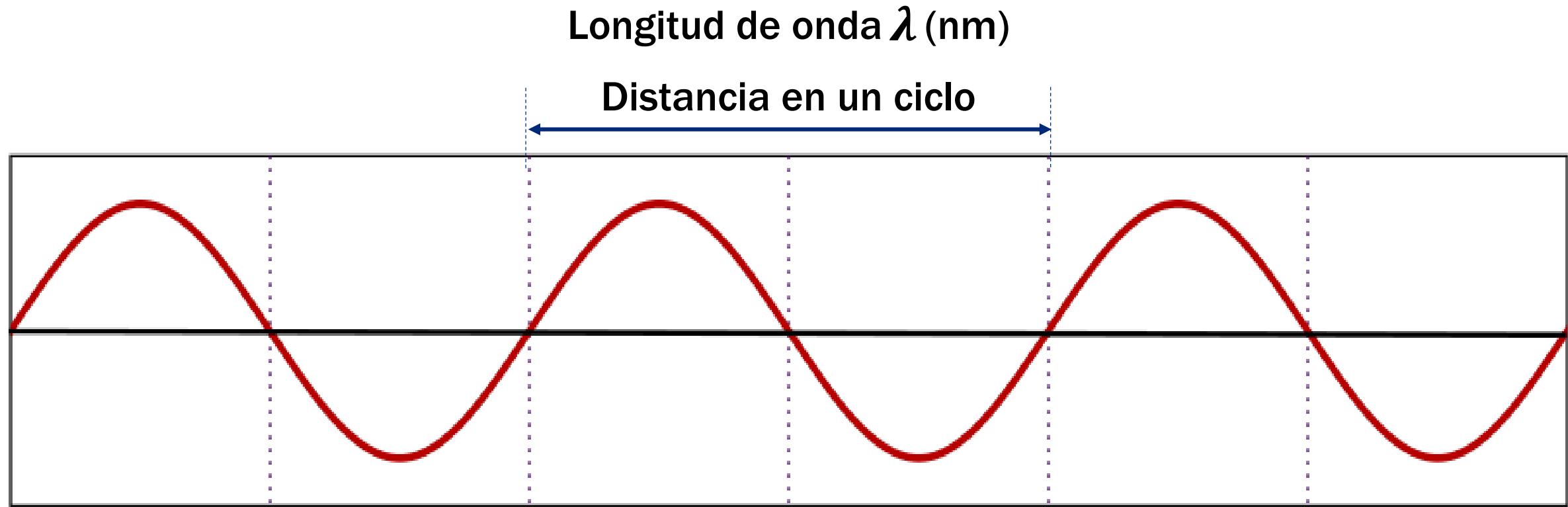
COMPONENTES DE UNA SEÑAL



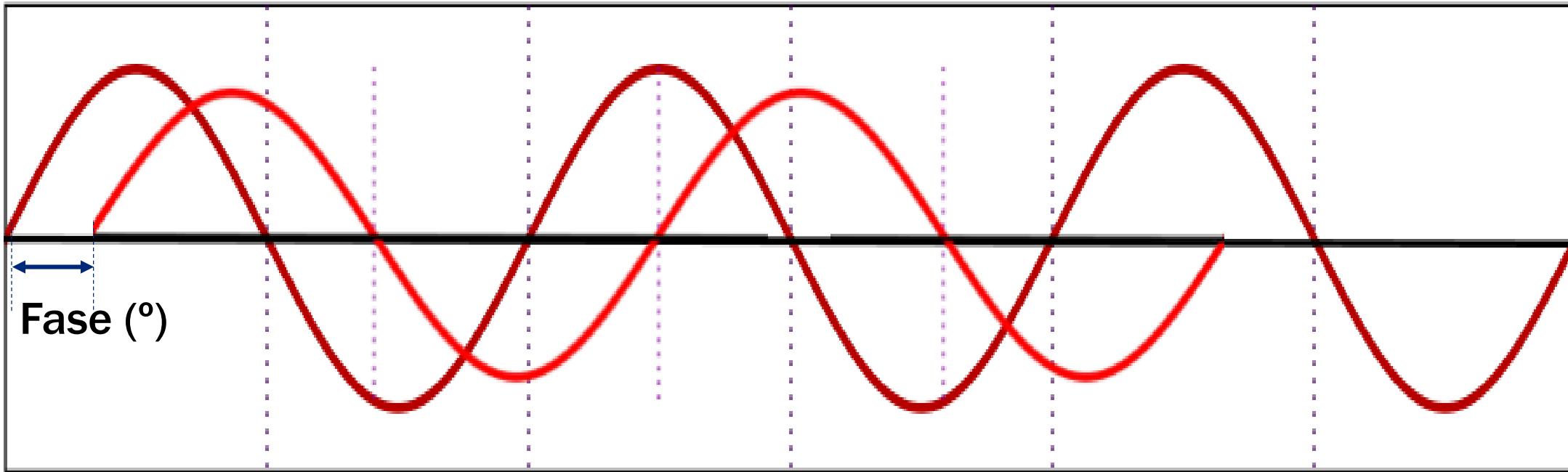
COMPONENTES DE UNA SEÑAL



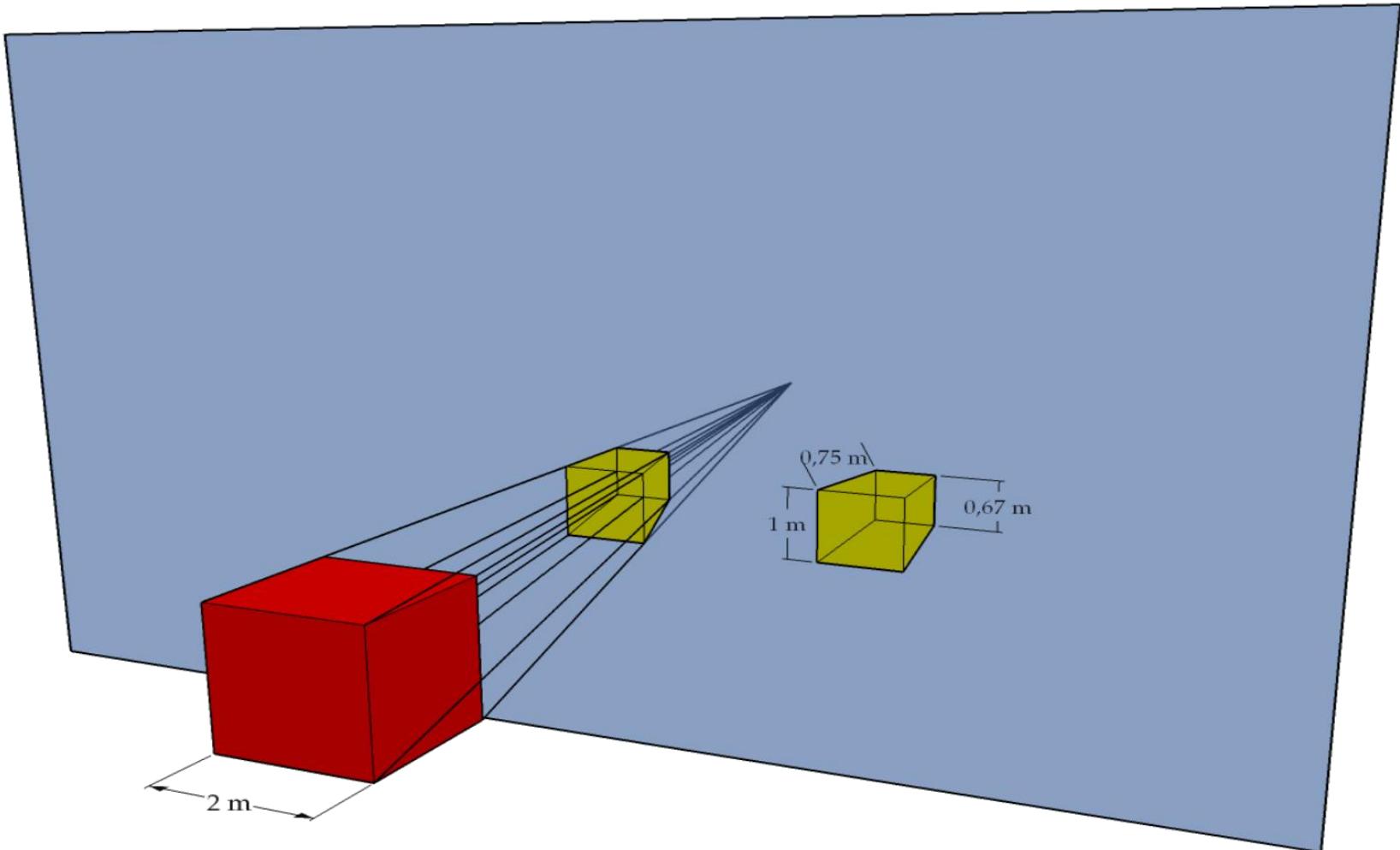
COMPONENTES DE UNA SEÑAL

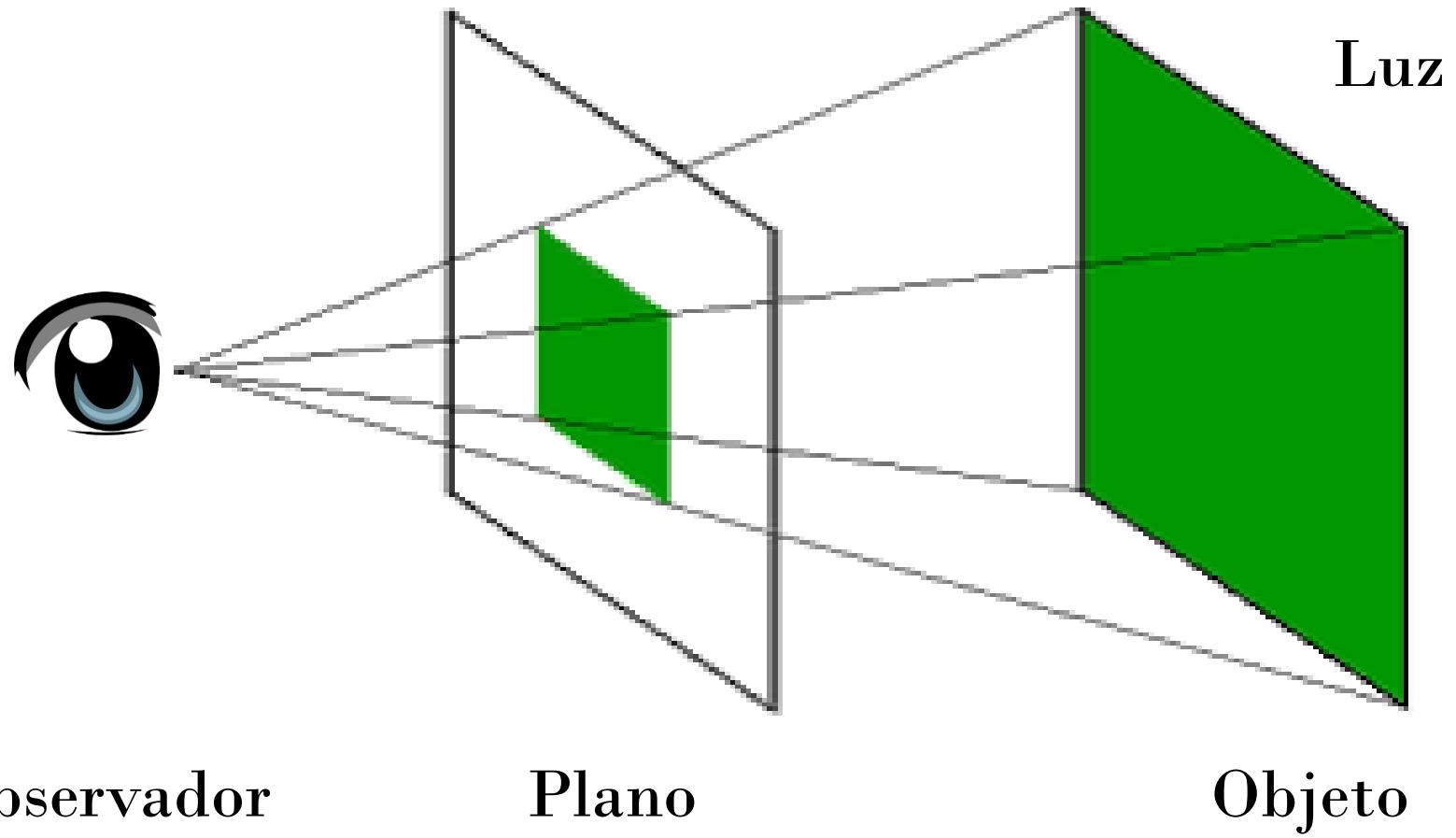


COMPONENTES DE UNA SEÑAL

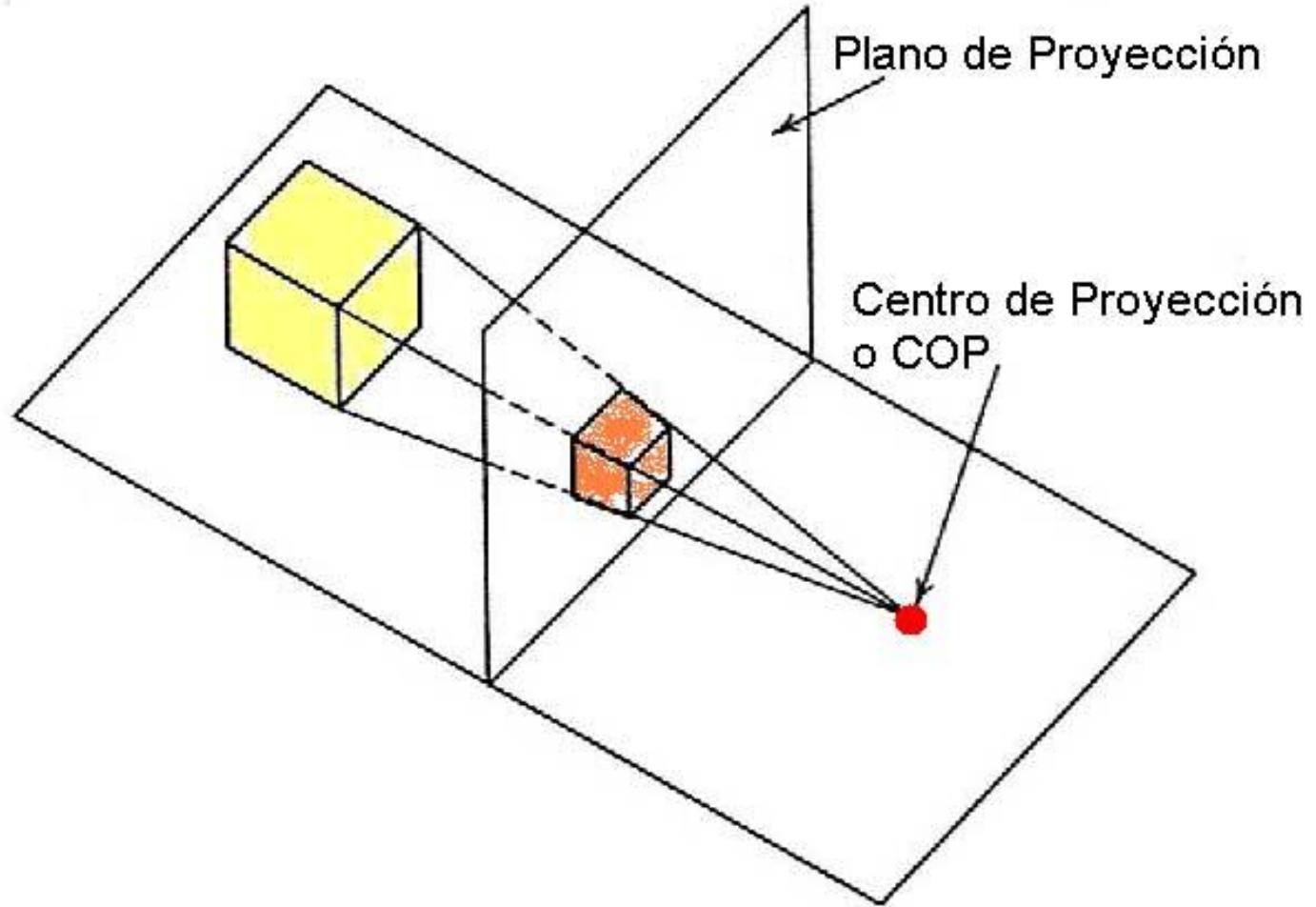


PLANO

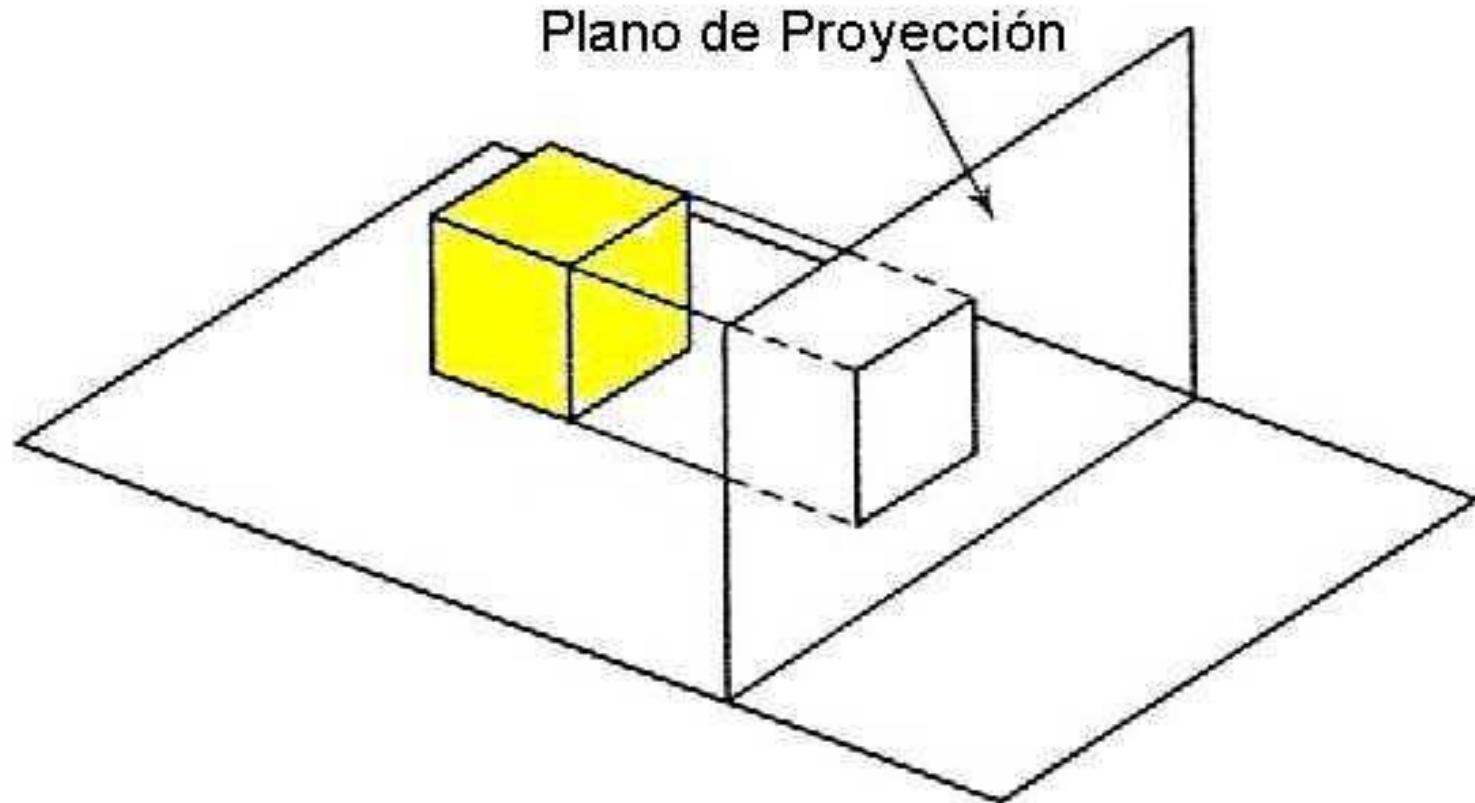




PROYECCIÓN PERSPECTIVA



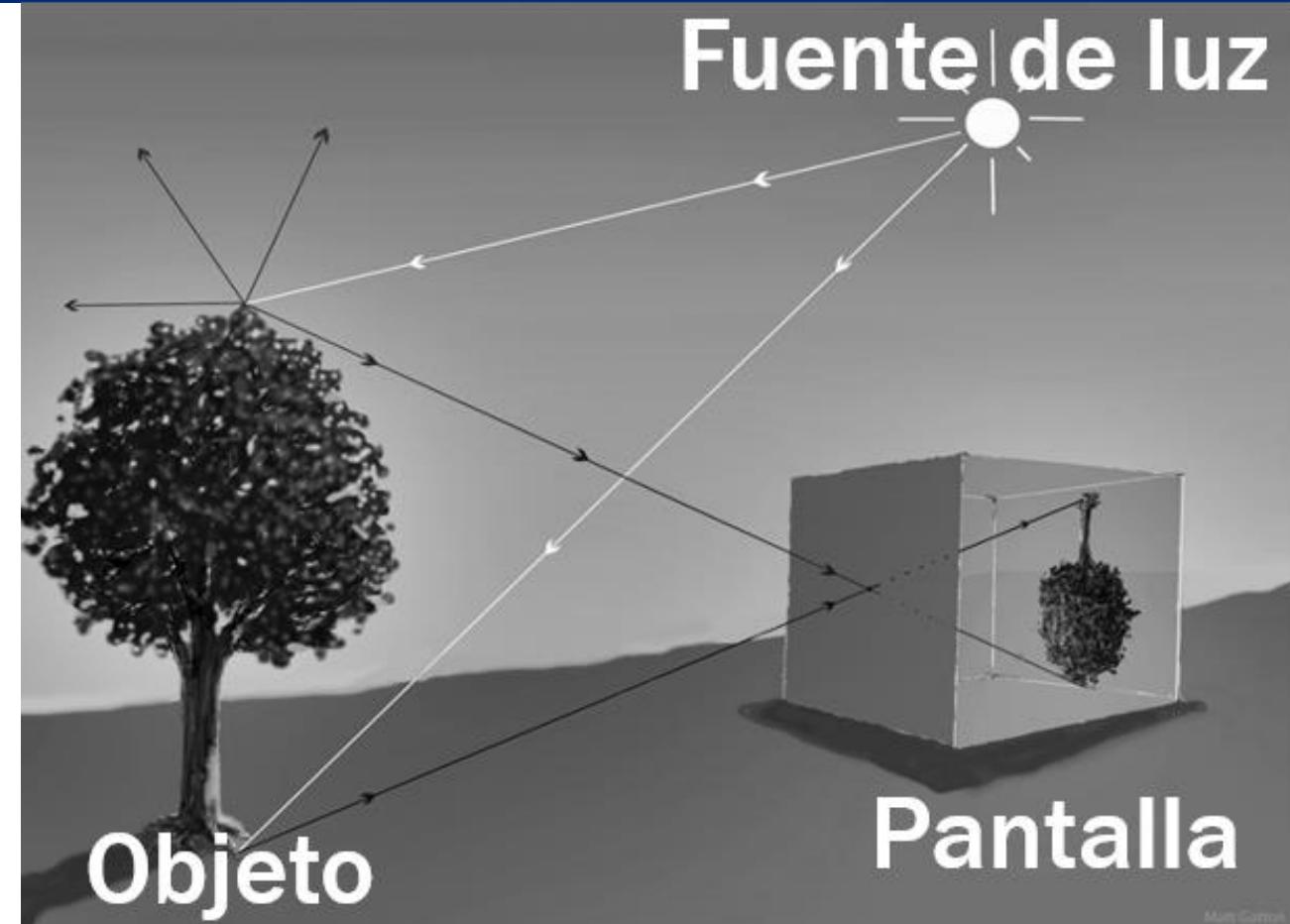
PROYECCIÓN ORTOGRÁFICA



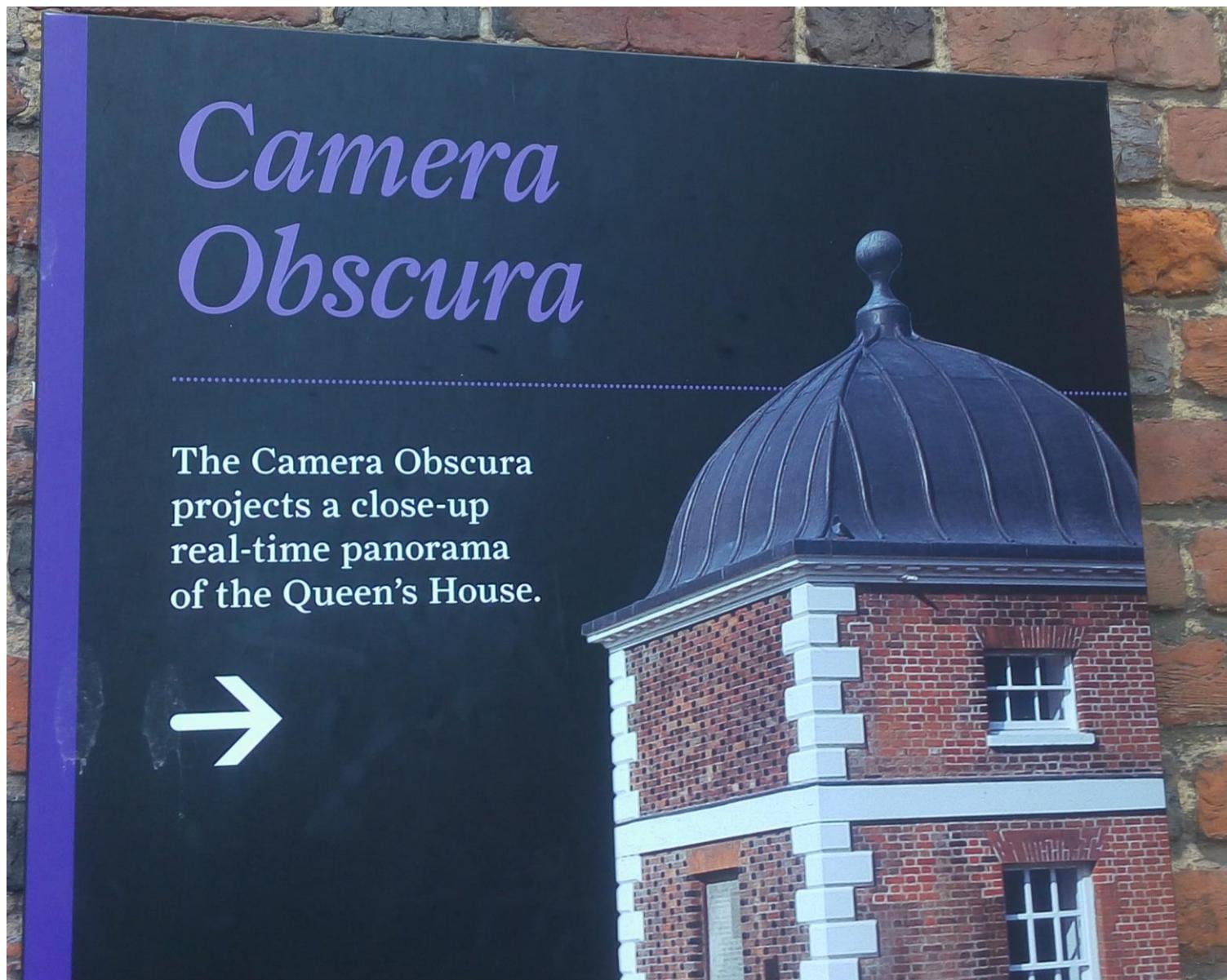
EJEMPLO 1 CÁMARA OSCURA

¿Qué es una cámara oscura?

En una habitación oscura con un pin hole en la pared, se proyecta una imagen del mundo en la pared opuesta. Los avances tecnológicos han permitido convertir una cámara oscura de una habitación a un instrumento portable.



John Flamsteed, el primer Astrónomo Real, observó el sol utilizando una cámara oscura que proyecta la imagen en una pantalla. En esta habitación usted puede utilizar esta cámara oscura para ver la casa de la Reyna (the Queen's House) en Greenwich, UK y el río Támesis.





PIN HOLE

PANTALLA

Taller de Procesamiento Digital de Imágenes

16 de Noviembre de 2018

21



IMAGEN DEL MUNDO

EJEMPLO 2

ARREGLO DE LENTES

Óptica geométrica paraxial

Se asume que todos los rayos de luz pasan a través de los lentes y la normal a las superficies refractivas y esféricas de los lentes es pequeña. También se asume que los lentes son rotacionalmente simétricos con respecto al eje óptico.



CONTENIDO

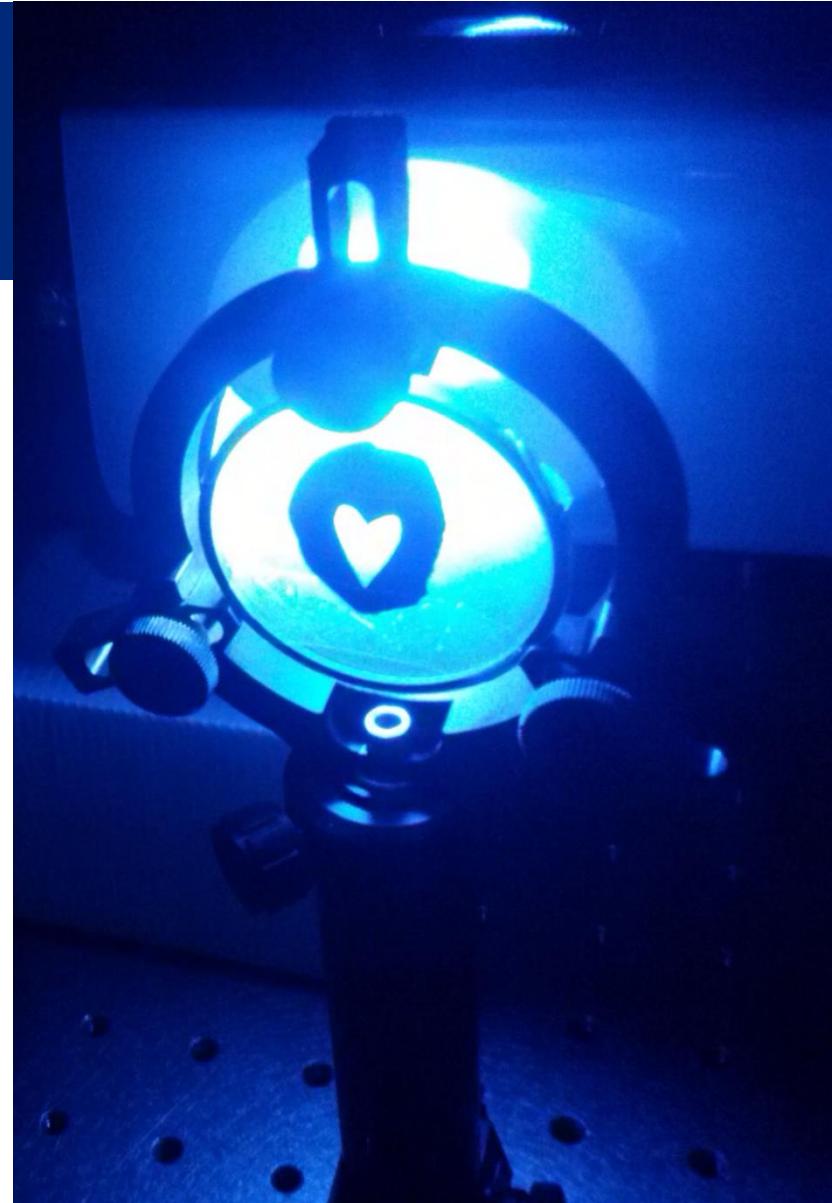
Introducción

Sistemas ópticos

Calibración de cámara

Introducción a las imágenes digitales

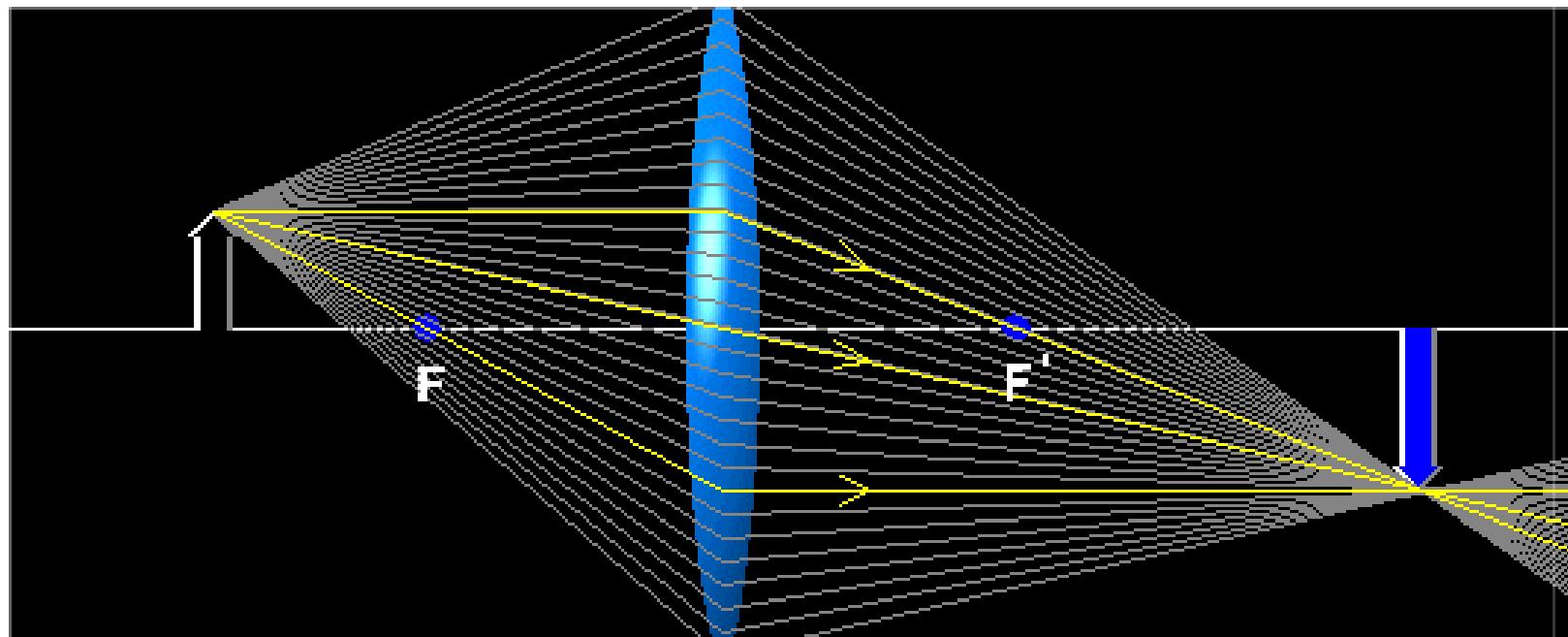
Procesamiento digital de imágenes

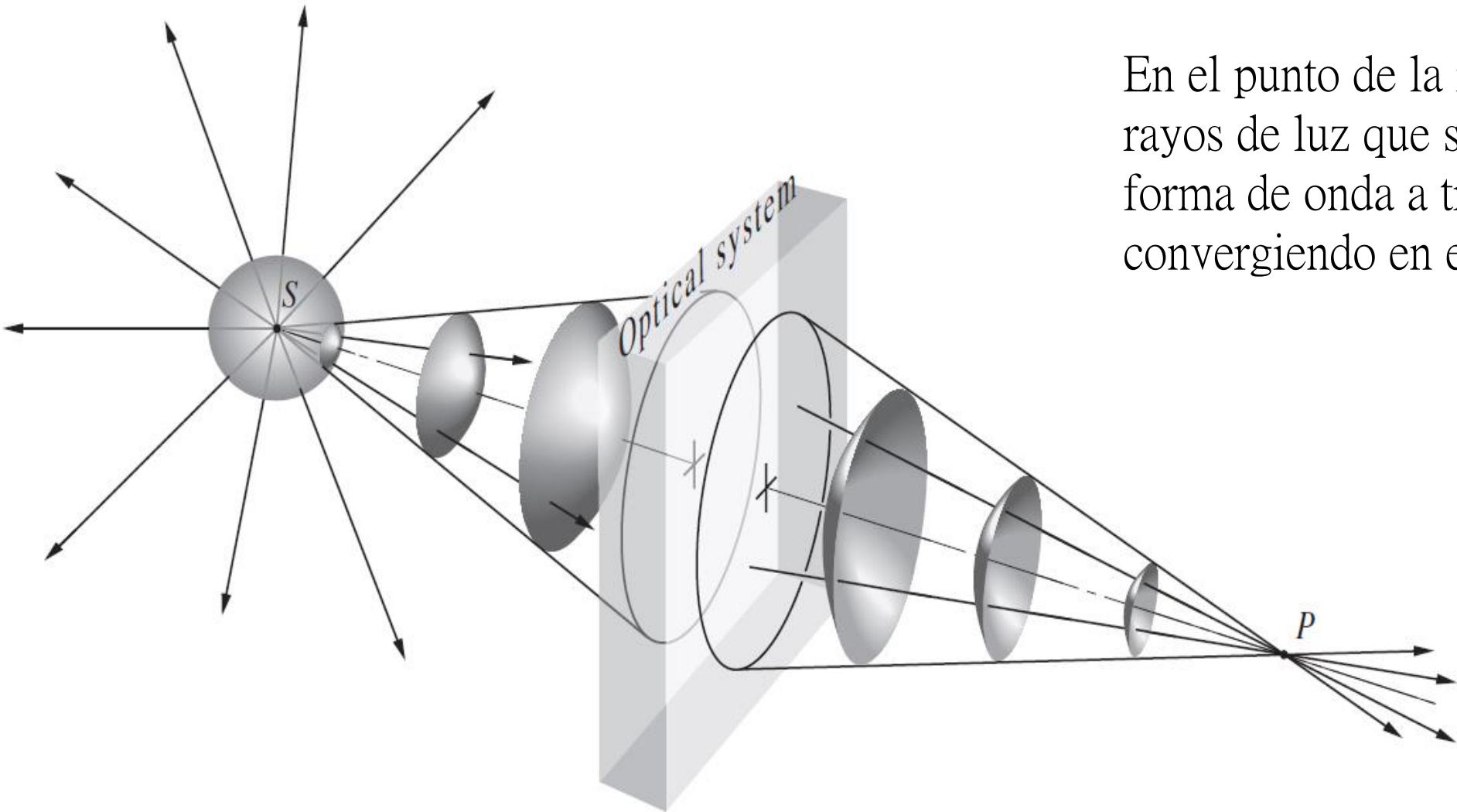


Taller de Procesamiento Digital de Imágenes

LENTES

Una lente concentra los rayos de luz provenientes de los objetos a visualizar, en un plano donde se formará su imagen para ser capturada por un sensor.

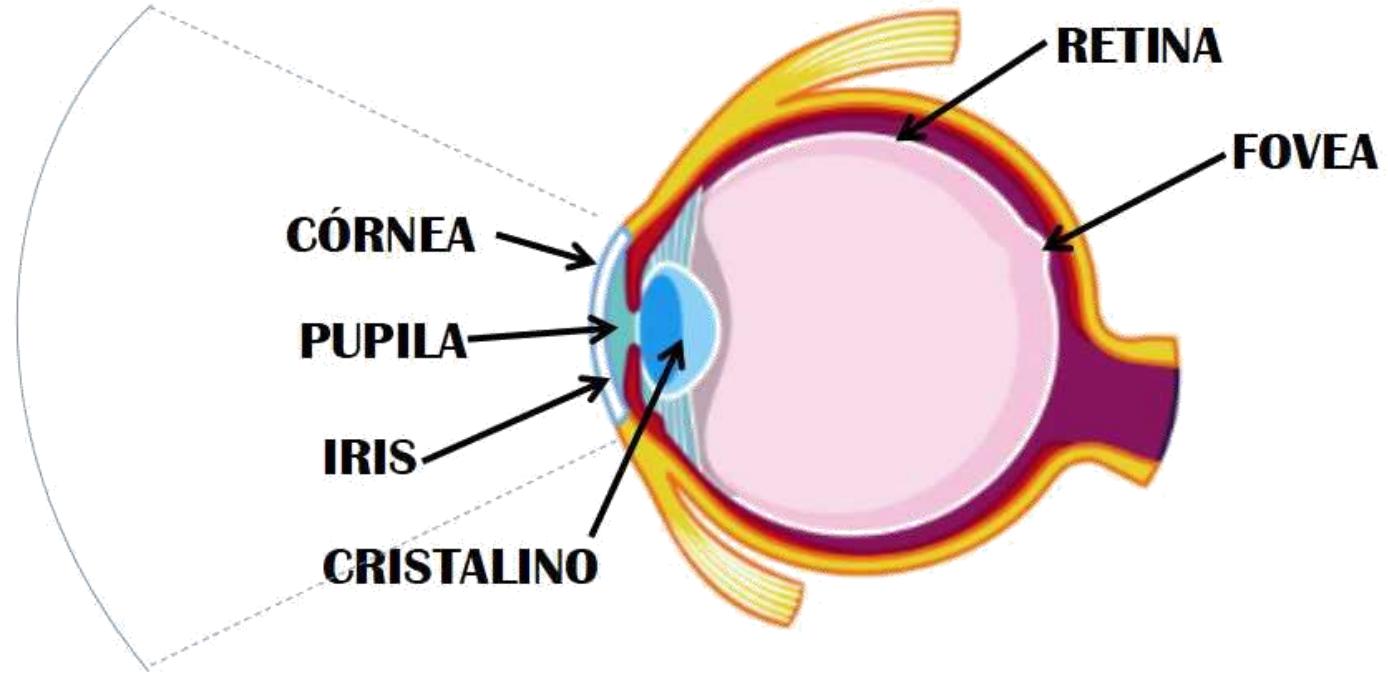




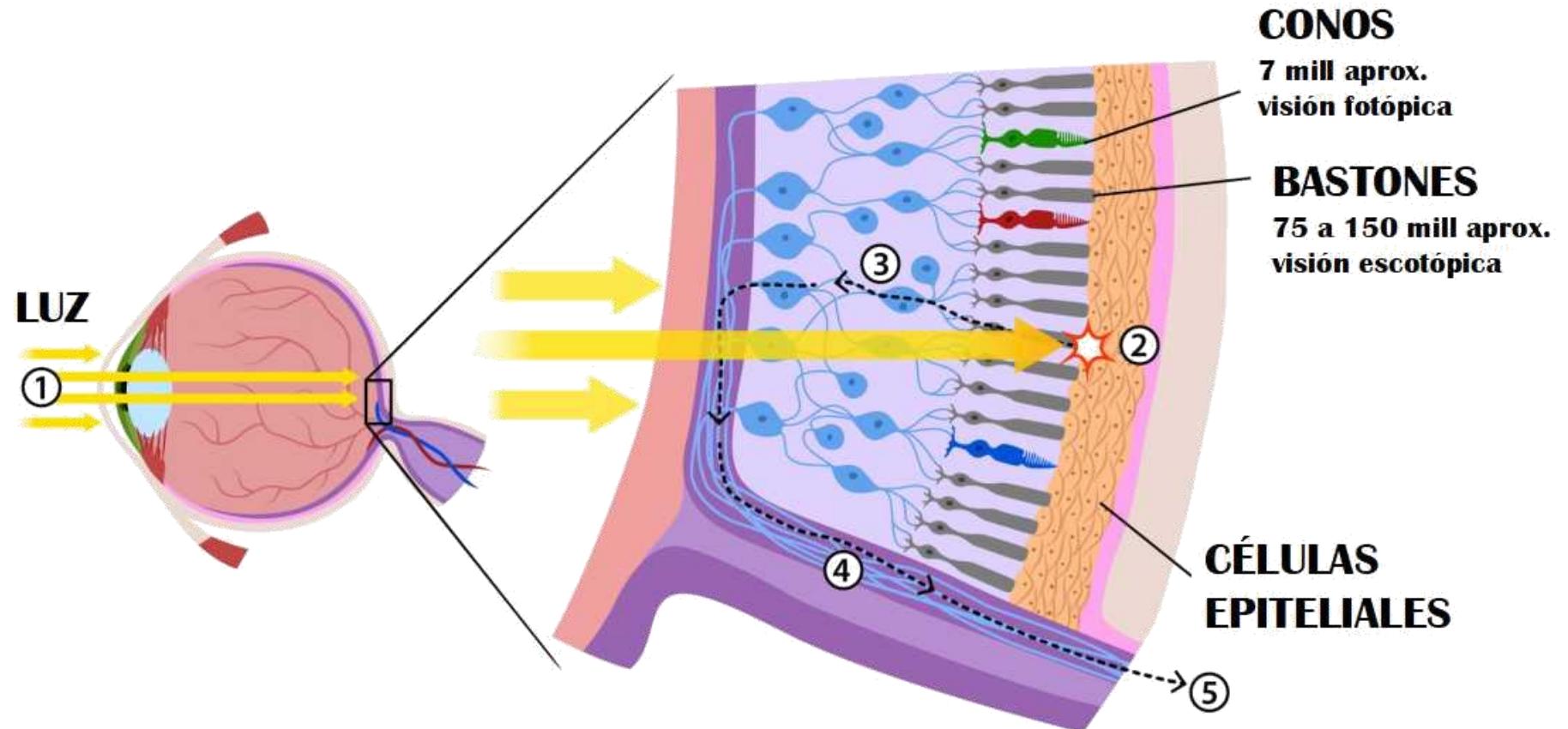
En el punto de la imagen S refleja rayos de luz que se transmiten en forma de onda a través de una lente convergiendo en el punto P .

ANATOMÍA OJO

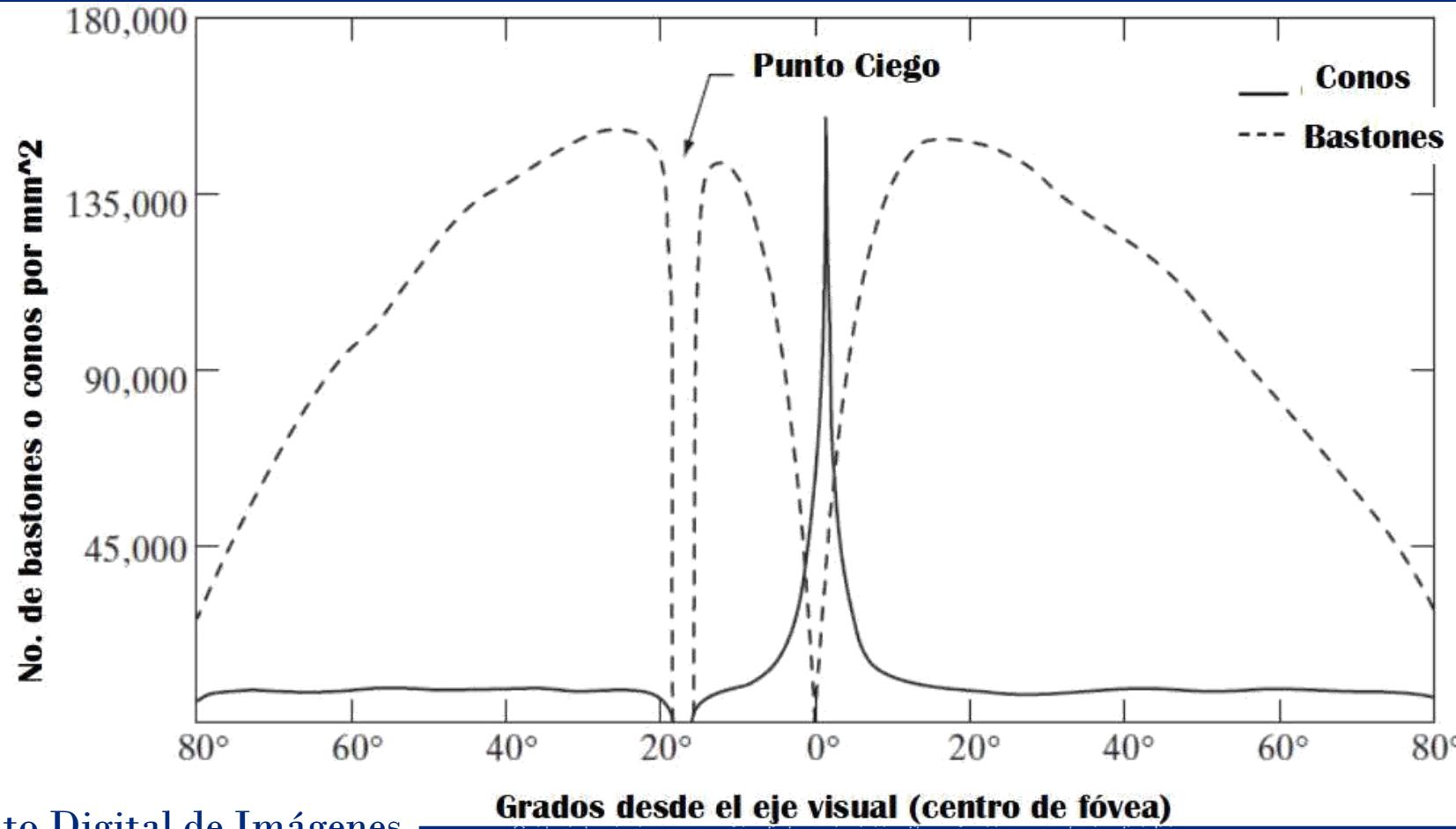
**135° ALTO
160° ANCHO**



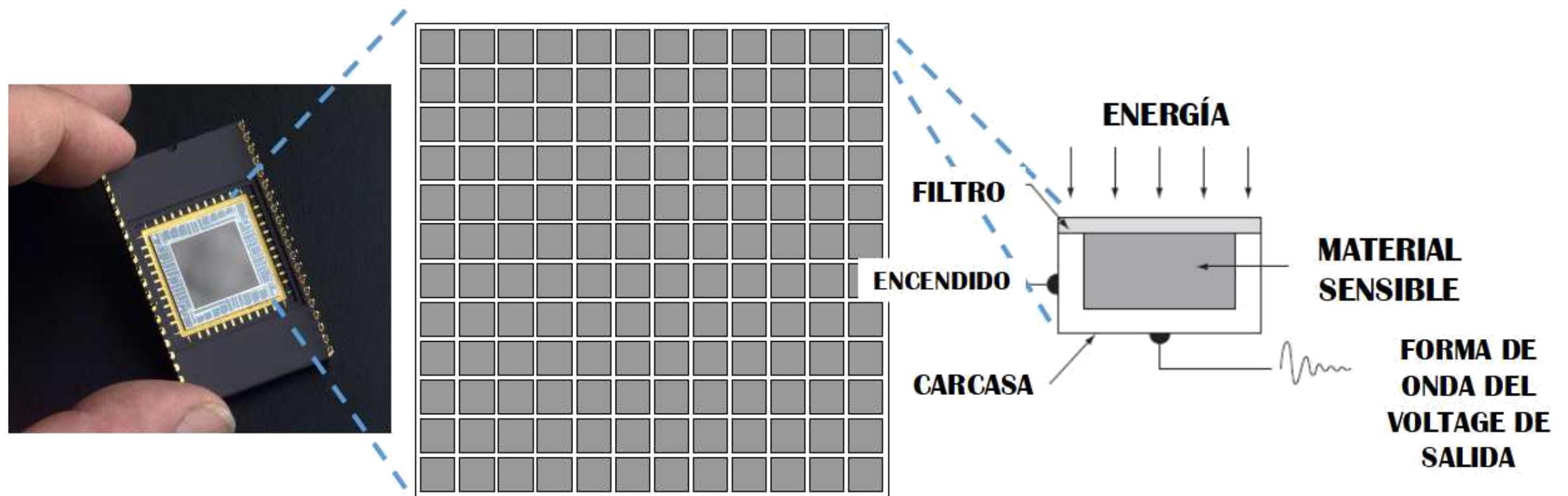
ANATOMÍA OJO



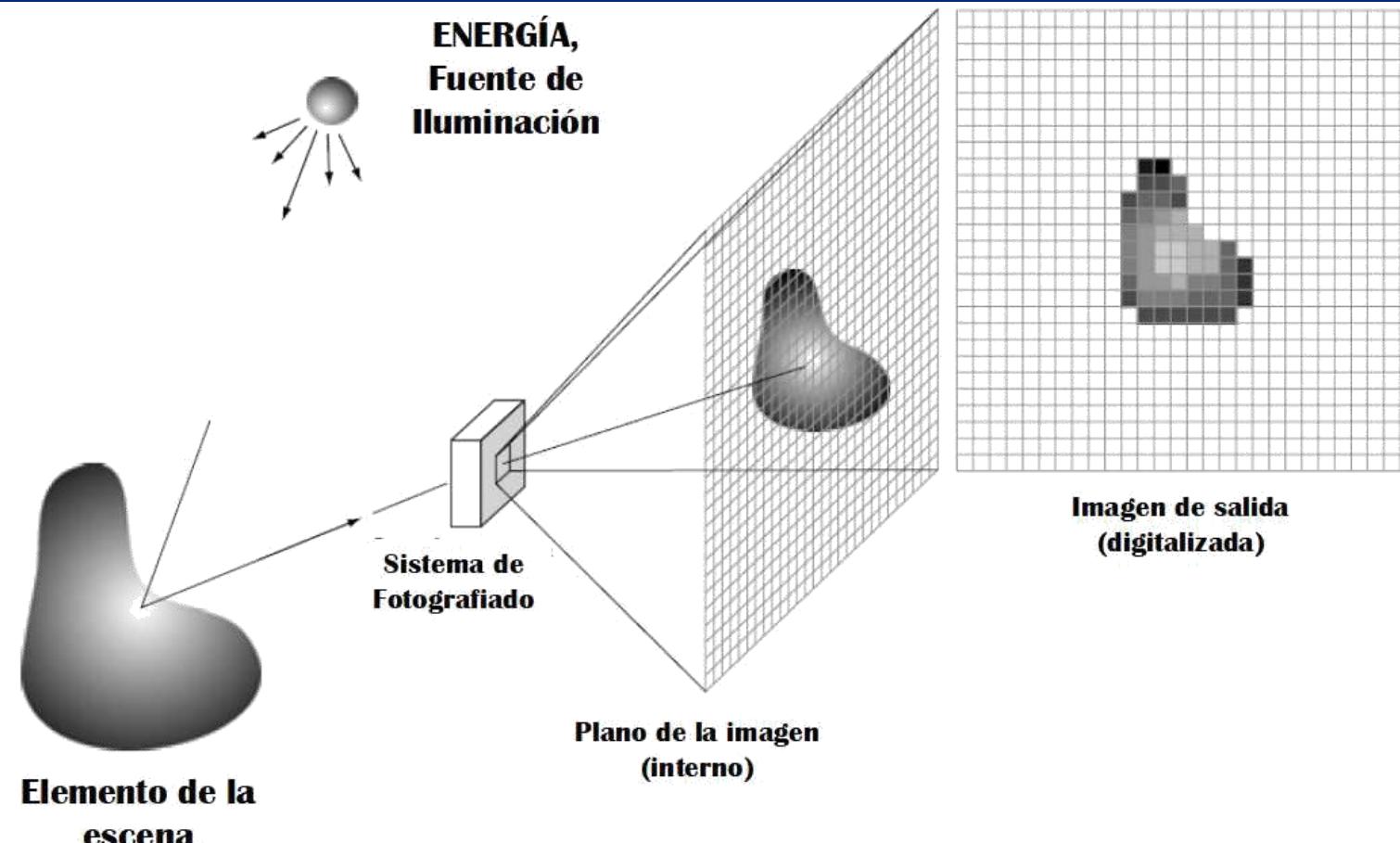
ANATOMÍA OJO



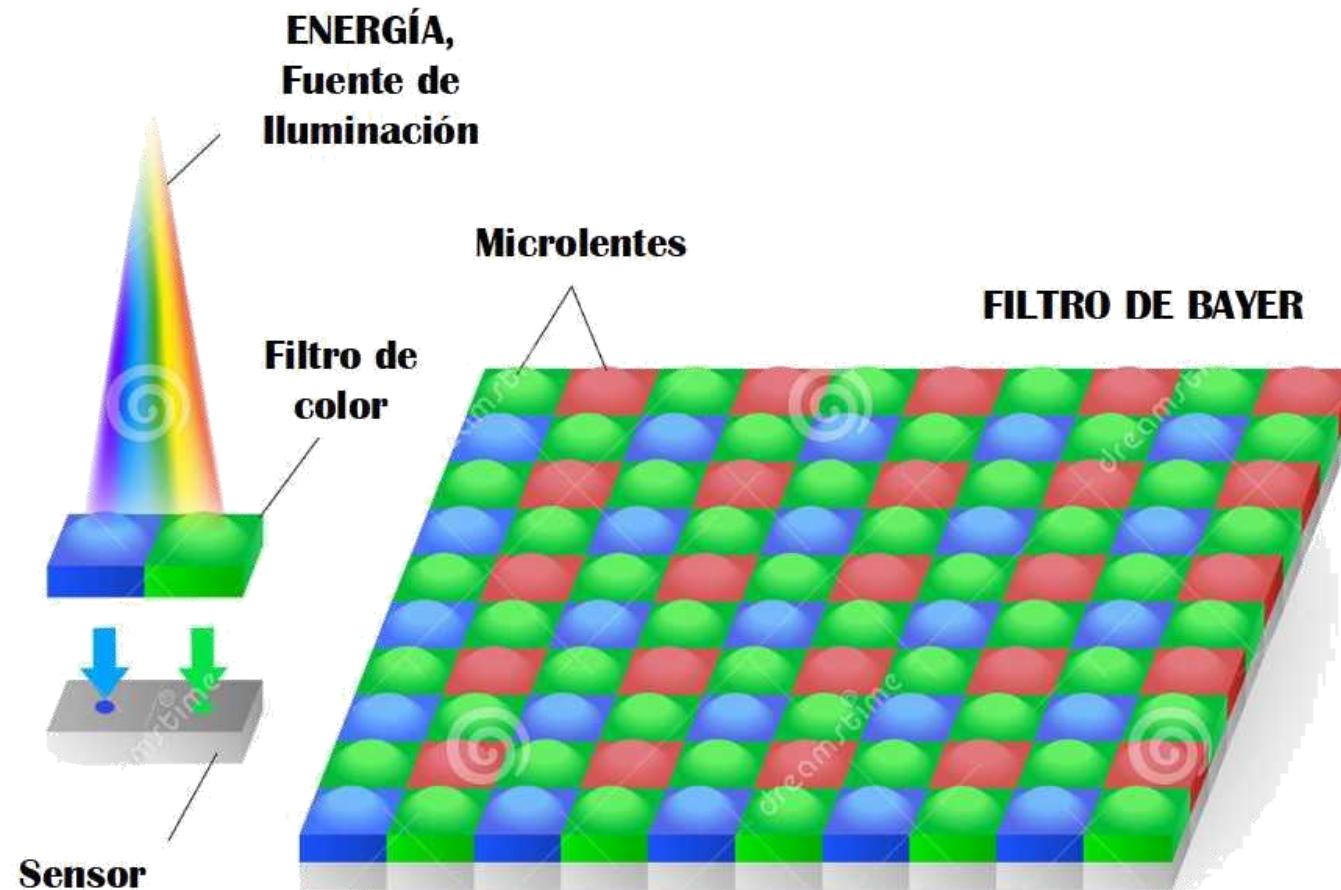
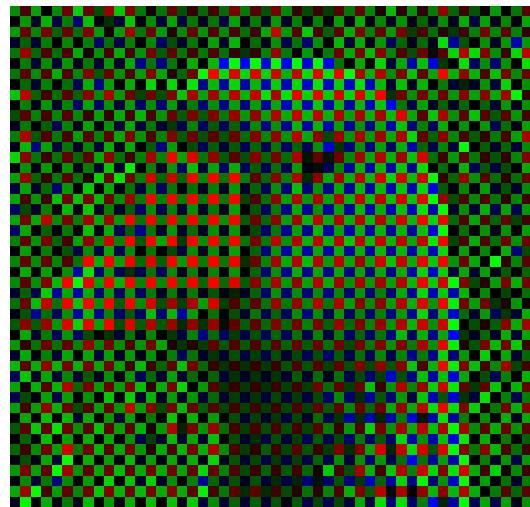
CCD CHARGED COUPLED DEVICE



CCD CHARGED COUPLED DEVICE



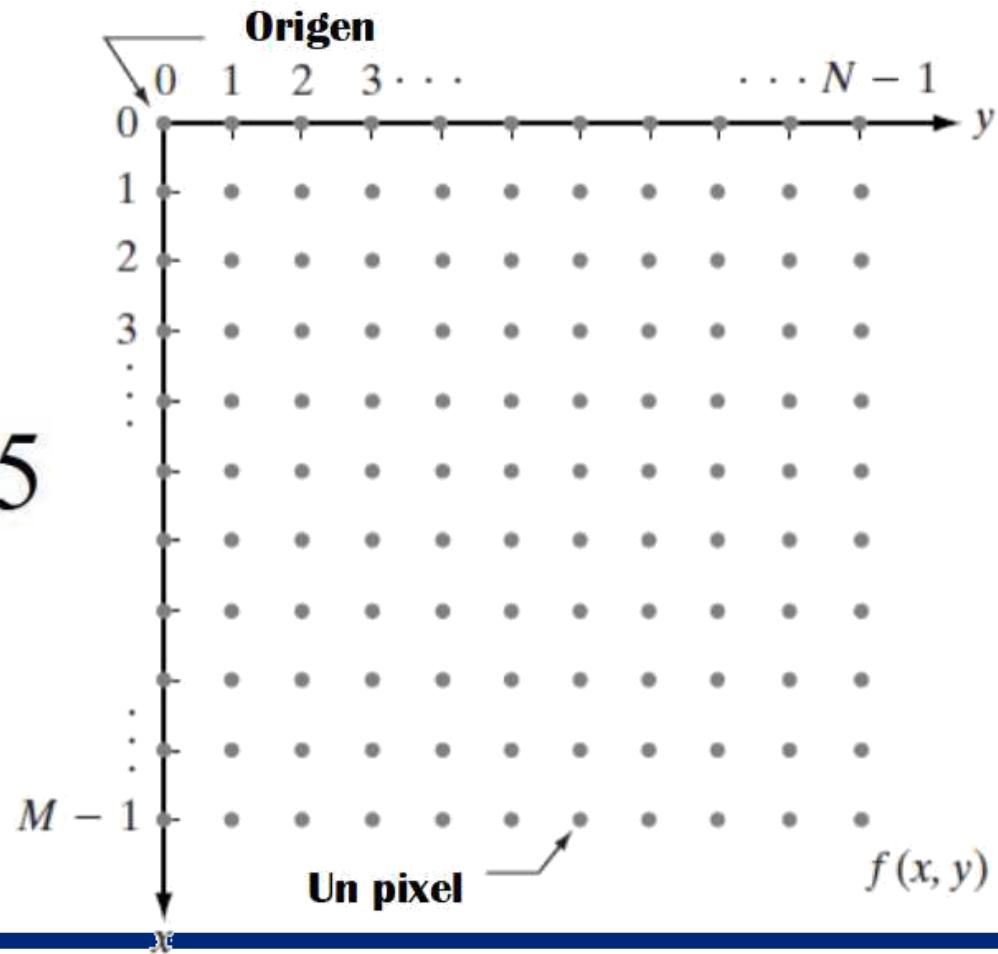
CCD CHARGED COUPLED DEVICE



Taller de Procesamiento Digital de Imágenes

CCD CHARGED COUPLED DEVICE

$$0 < f(x, y) < 255$$



CCD CHARGED COUPLED DEVICE

$$f(x, y) = \begin{bmatrix} f(0, 0) & f(0, 1) & \cdots & f(0, N - 1) \\ f(1, 0) & f(1, 1) & \cdots & f(1, N - 1) \\ \vdots & \vdots & & \vdots \\ f(M - 1, 0) & f(M - 1, 1) & \cdots & f(M - 1, N - 1) \end{bmatrix}.$$

$$\mathbf{A} = \begin{bmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{bmatrix}.$$

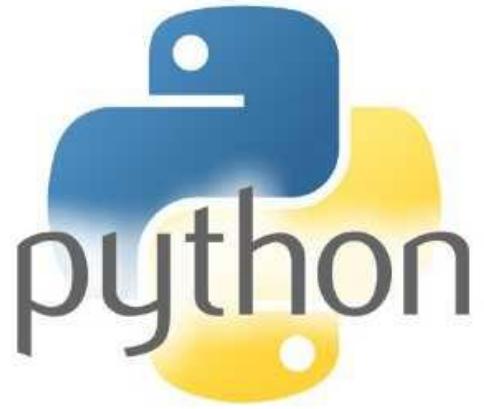
HERRAMIENTAS OPEN SOURCE

Taller de Procesamiento Digital de Imágenes

16 de Noviembre de 2018

35





- *The mission: ... to promote, protect, and advance the Python programming language, and to support and facilitate the growth of a diverse and international community of Python programmers.*
- —from the *Mission Statement* page

Función factorial en C (indentación opcional)

```
int factorial(int x)
{
    if (x < 0 || x % 1 != 0) {
        printf("x debe ser un numero entero mayor o igual a 0");
        return -1; //Error
    }
    if (x == 0) {
        return 1;
    }
    return x * factorial(x - 1);
}
```

Función factorial en Python (indentación obligatoria)

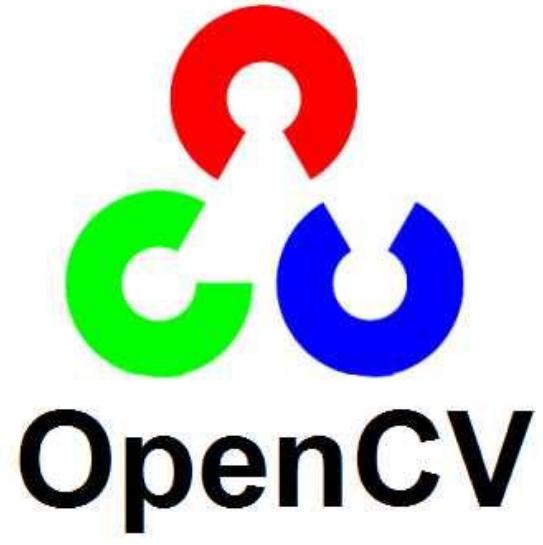
```
def factorial(x):
    assert x >= 0 and x % 1 == 0, "x debe ser un entero mayor o
    igual a 0."
    if x == 0:
        return 1
    else:
        return x * factorial(x - 1)
```



With over 6 million users, the open source Anaconda Distribution is the fastest and easiest way to do Python and R data science and machine learning on Linux, Windows, and Mac OS X. It's the industry standard for developing, testing, and training on a single machine.



Project Jupyter exists to develop open-source software, open-standards, and services for interactive computing across dozens of programming languages.



OpenCV (Open Source Computer Vision Library)

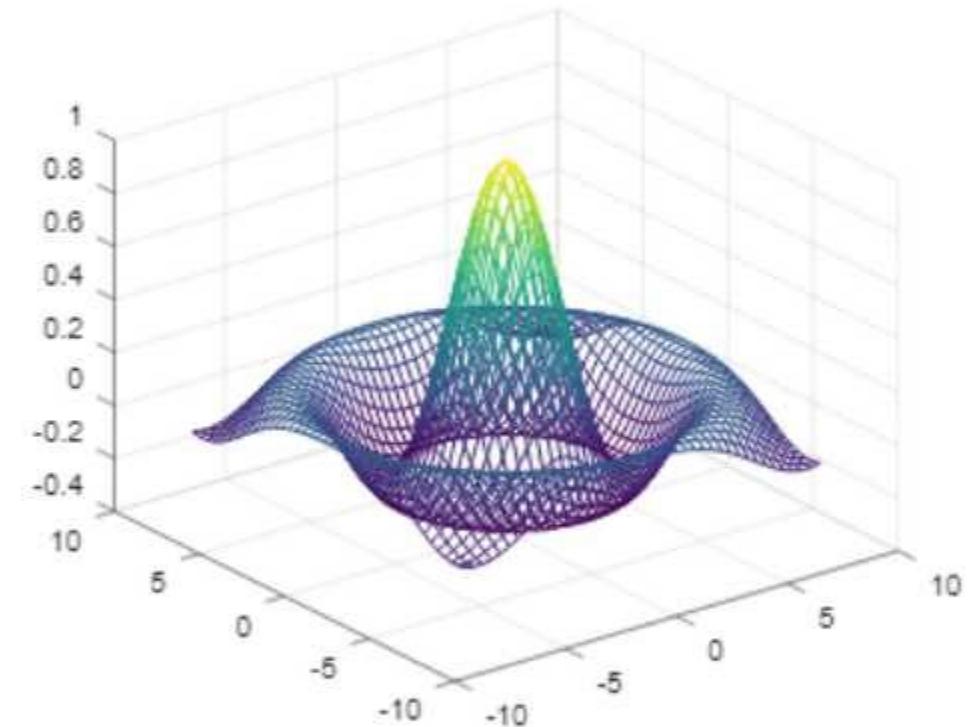
It has C++, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android.

OpenCV was designed for computational efficiency and with a strong focus on real-time applications.



Scientific Programming Language

- Powerful mathematics-oriented syntax with built-in plotting and visualization tools
- Free software, runs on GNU/Linux, macOS, BSD, and Windows
- Drop-in compatible with many Matlab scripts



CONTENIDO

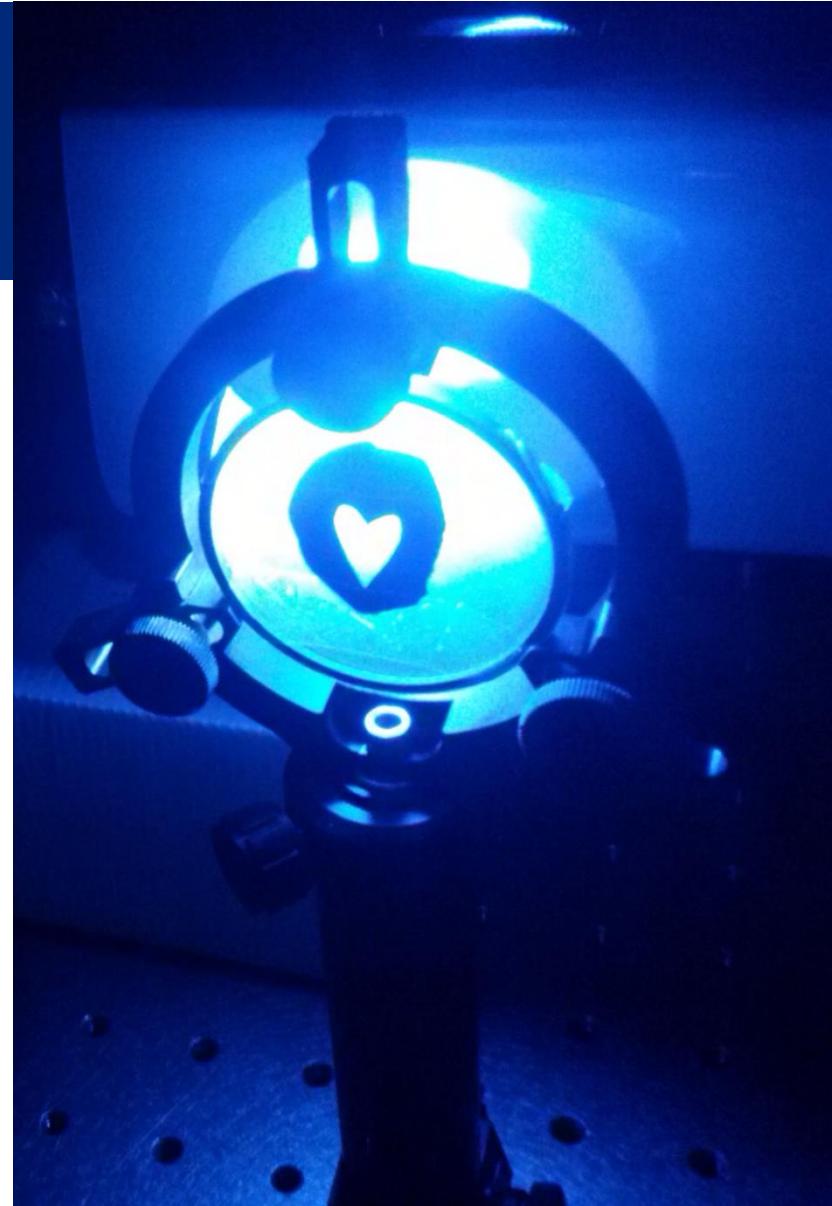
Introducción

Sistemas ópticos

Calibración de cámara

Introducción a las imágenes digitales

Procesamiento digital de imágenes



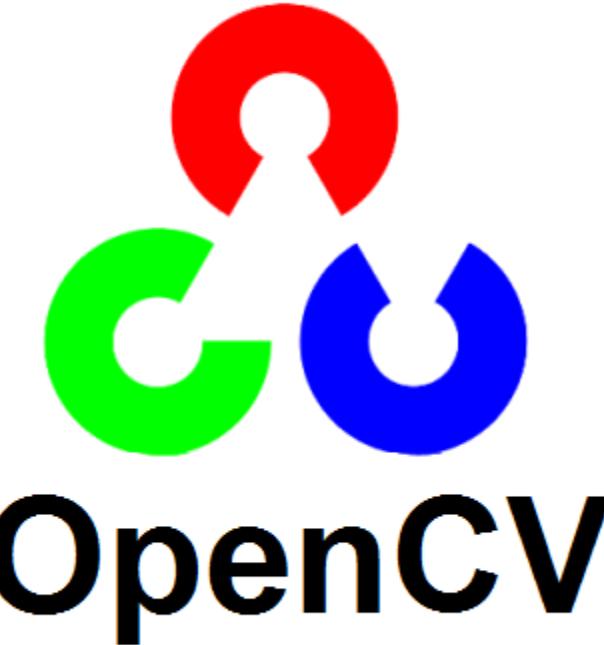
Taller de Procesamiento Digital de Imágenes

PRÁCTICA

```
mirror_object to mirror_mod.mirror_object
operation == "MIRROR_X":
    mirror_mod.use_x = True
    mirror_mod.use_y = False
    mirror_mod.use_z = False
operation == "MIRROR_Y":
    mirror_mod.use_x = False
    mirror_mod.use_y = True
    mirror_mod.use_z = False
operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end - add line
    _ob.select= 1
    mirr_ob.select=1
    bpy.context.scene.objects.active = eval(
        ("Selected" + str(modifier)))
    mirror_ob.select = 0
    bpy.context.selected_objects = eval(
        "data.objects[" + one.name + "].select")
    print("please select exactly one object")

OPERATOR CLASSES ---
```



Abre la práctica
camera_calibration.ipynb

CONTENIDO

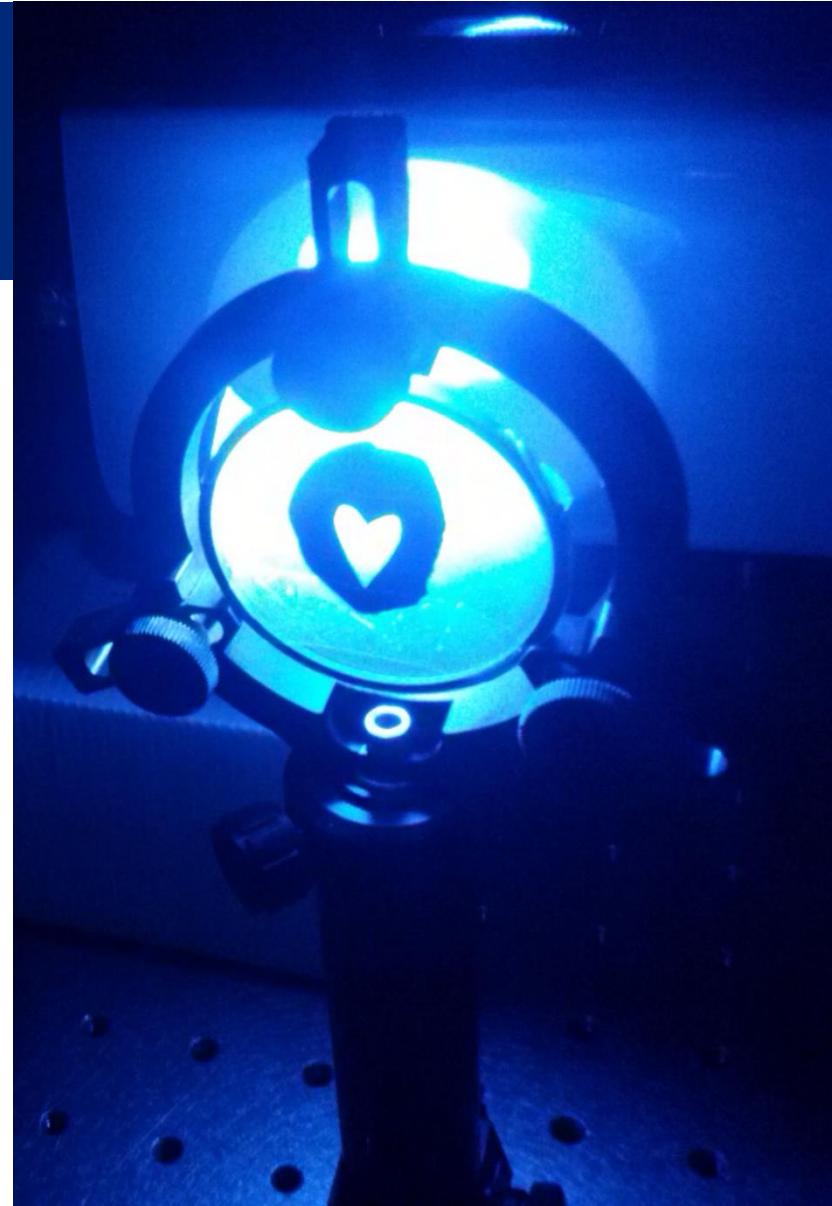
Introducción

Sistemas ópticos

Calibración de cámara

Introducción a las imágenes digitales

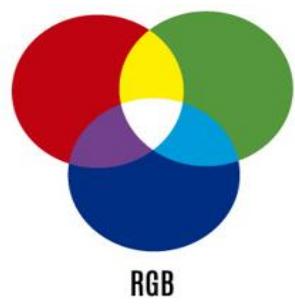
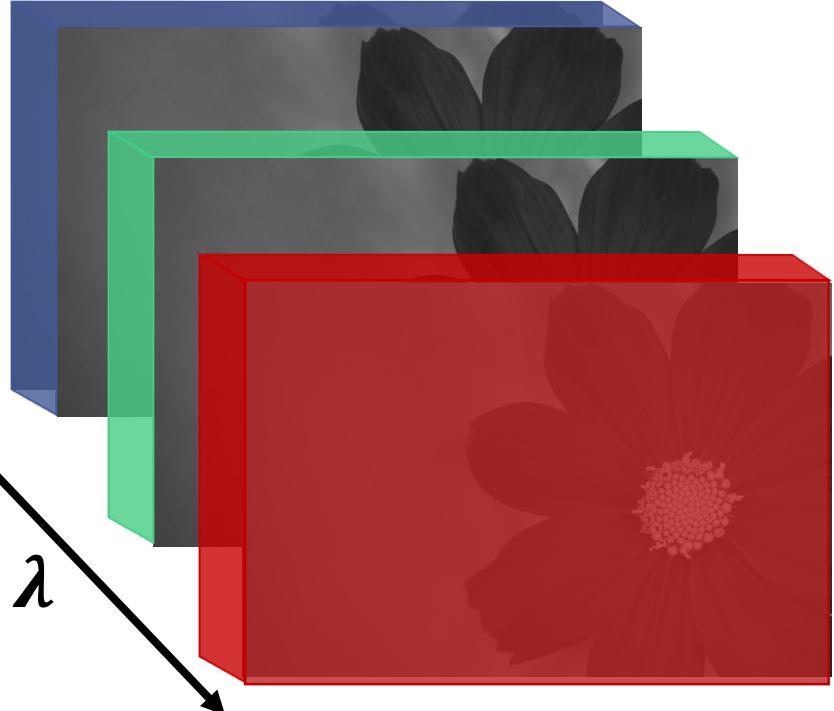
Procesamiento digital de imágenes



Taller de Procesamiento Digital de Imágenes

COLOR

RGB



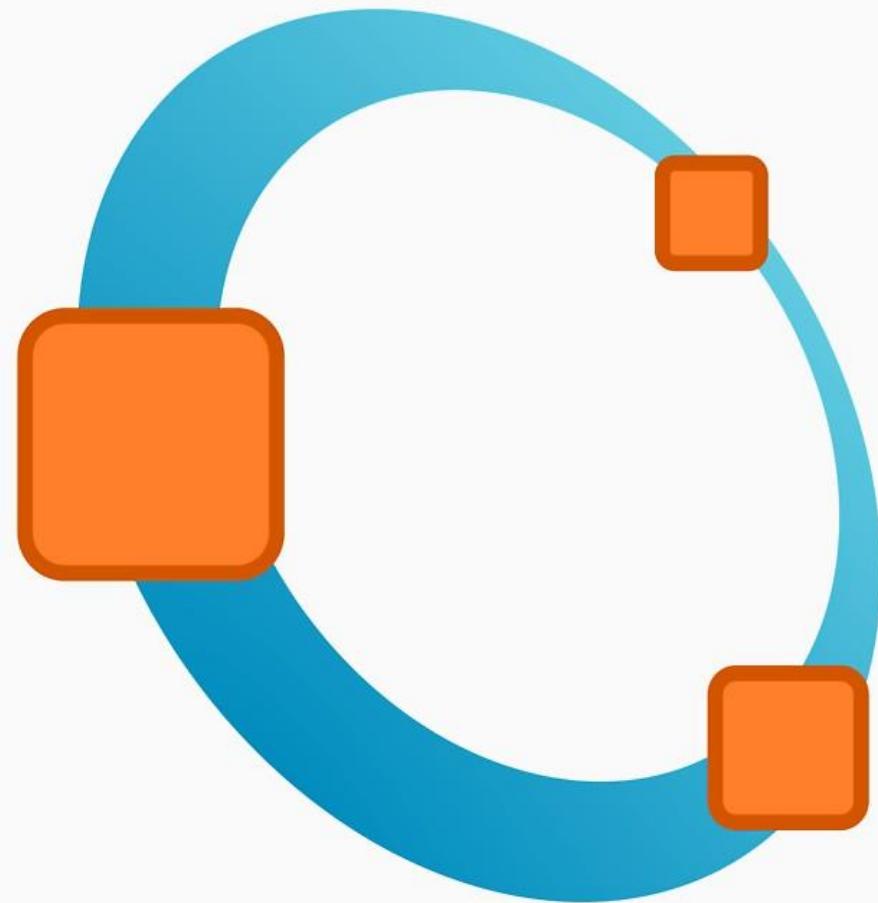
PRÁCTICA



```
mirror_mod.mirror_object = None
if mirror_mod.mirror_object == None:
    if mirror_mod.operation == "MIRROR_X":
        mirror_mod.use_x = True
        mirror_mod.use_y = False
        mirror_mod.use_z = False
    elif mirror_mod.operation == "MIRROR_Y":
        mirror_mod.use_x = False
        mirror_mod.use_y = True
        mirror_mod.use_z = False
    elif mirror_mod.operation == "MIRROR_Z":
        mirror_mod.use_x = False
        mirror_mod.use_y = False
        mirror_mod.use_z = True

#selection at the end - add last selected
    _ob.select= 1
    mirr_ob.select=1
    context.scene.objects.active = eval("Selected" + str(modifier))
    mirror_ob.select = 0
    bpy.context.selected_objects = []
    data.objects[one.name].select = 1
    print("please select exactly one object")
else:
    print("mirror object to mirror")
    mirror_mod.mirror_object = mirror_mod.mirror_object
```

pkg load image



Ejecuta la práctica
ImageRGB.m

TAMAÑO

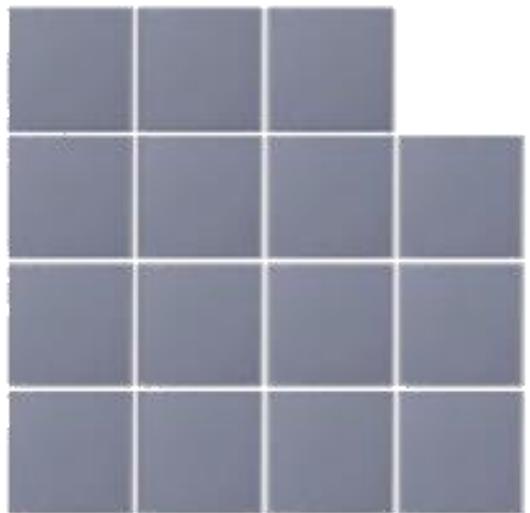


Taller de Procesamiento Digital de Imágenes

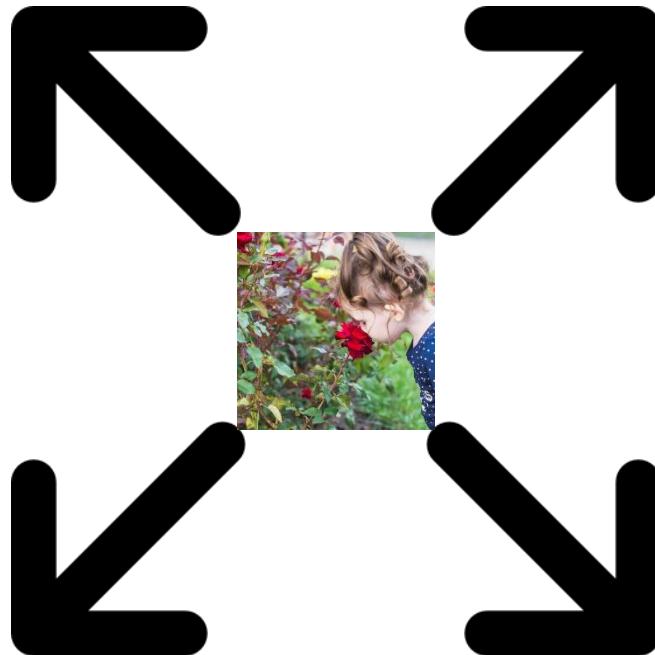
16 de Noviembre de 2018

TAMAÑO

2D



$m \times n$ pixeles



16	45	36	39	32	33	39	34	51	51	76	97	100	121	126	147	149	150	146	146	145	153
10	39	41	48	52	56	74	81	102	106	163	194	201	210	221	199	199	152	152	91	95	61
36	32	42	42	54	54	77	74	104	102	163	198	192	211	211	205	203	150	155	92	94	60
93	95	117	118	140	140	166	166	186	195	213	220	226	210	217	176	166	117	108	55	45	21
37	91	116	117	142	142	164	166	189	181	216	223	221	215	213	178	177	111	118	44	53	18
71	172	188	184	203	200	216	211	215	215	191	186	180	167	166	143	141	112	105	81	75	72
70	172	187	185	201	199	213	214	215	214	195	183	180	168	168	143	143	110	109	78	80	65
73	166	164	158	166	166	177	176	182	188	195	205	207	214	218	204	207	179	184	144	148	115
71	167	163	161	164	166	172	178	185	185	197	209	203	214	212	212	210	183	182	144	150	115
70	165	162	159	164	167	174	174	184	185	194	208	206	219	214	204	202	184	179	152	146	117
36	137	148	150	167	171	185	188	194	195	196	202	201	206	204	211	212	197	198	186	183	172
37	136	153	151	167	165	183	186	190	195	200	202	203	204	211	202	209	197	203	192	186	2X75

TAMAÑO



Taller de Procesamiento Digital de Imágenes

TAMAÑO



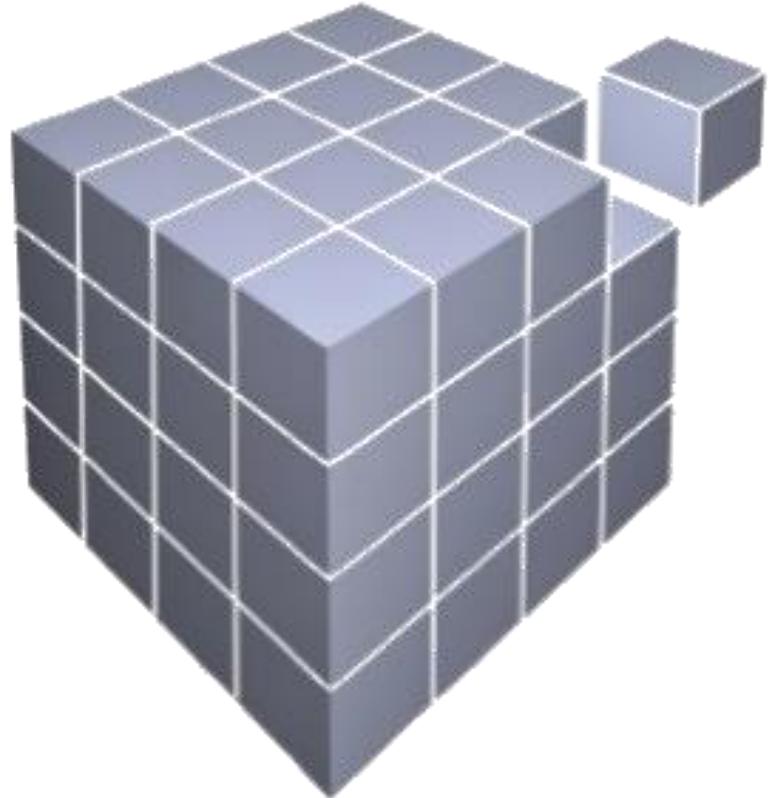
Taller de Procesamiento Digital de Imágenes

16 de Noviembre de 2018

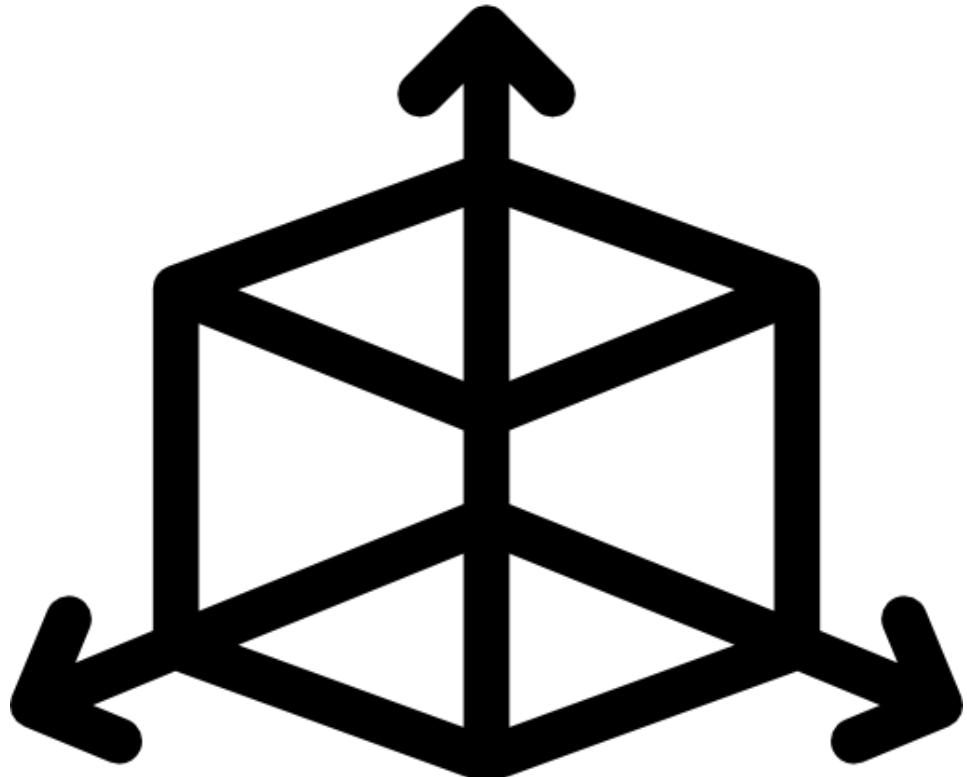
51

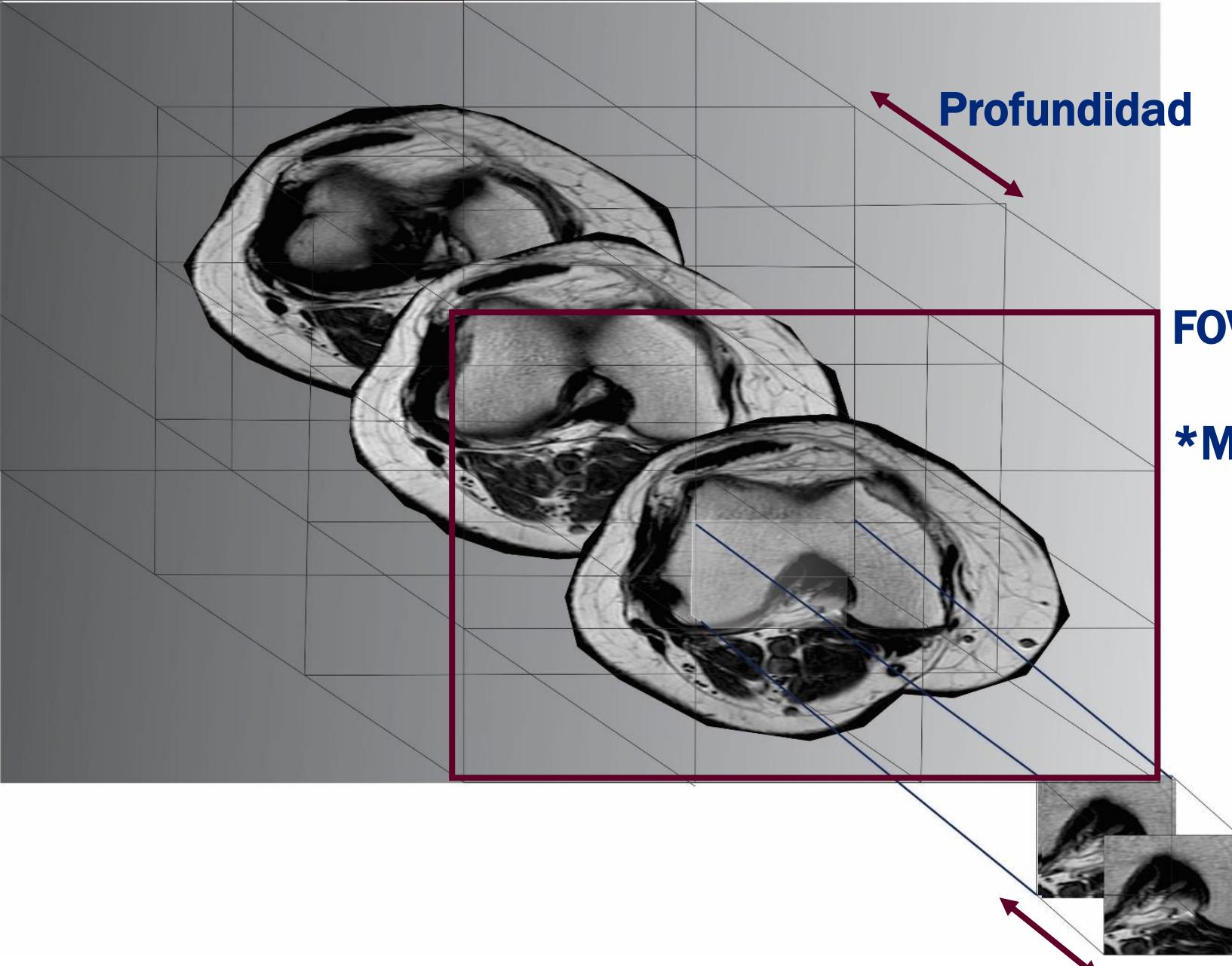
TAMAÑO

3D



$m \times n \times q$ voxels





Profundidad

FOV

*Matriz

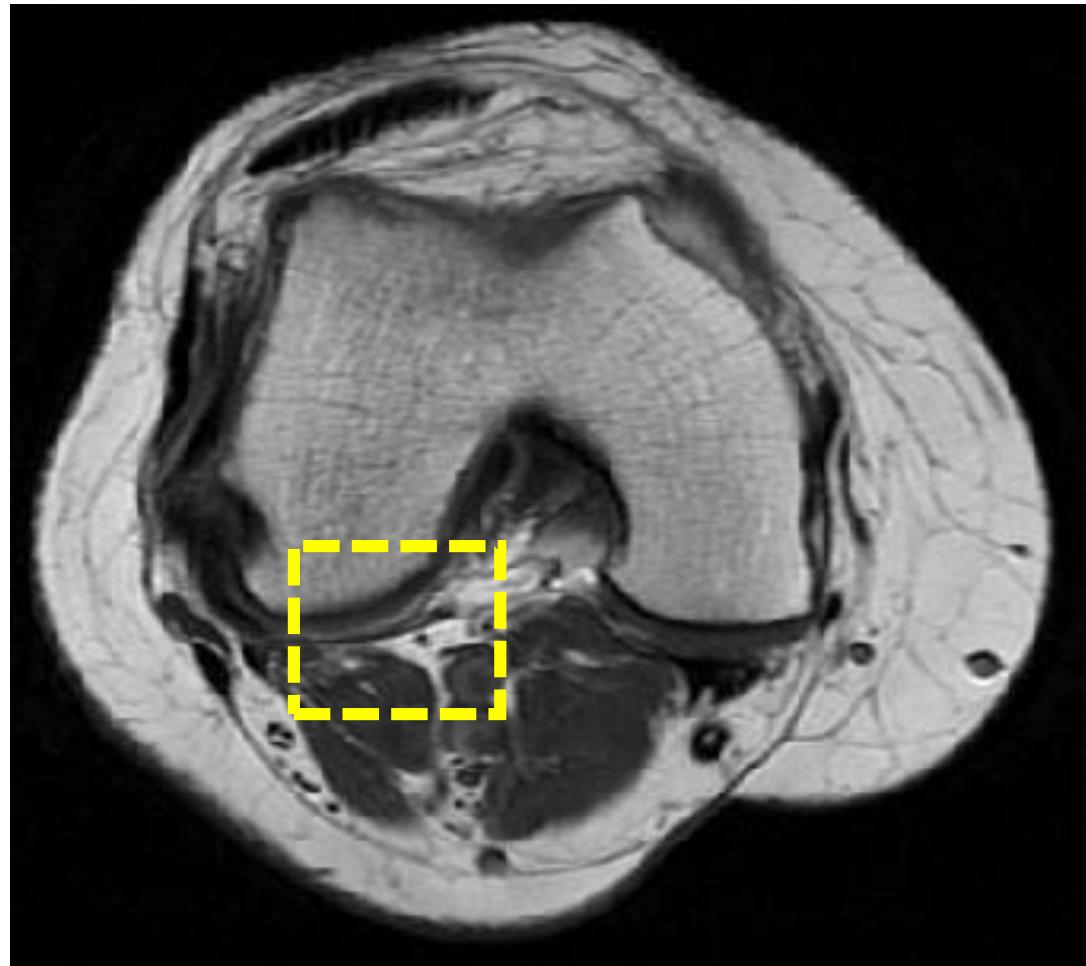
Voxel

RESOLUCIÓN

La resolución de una imagen indica la cantidad de detalles que puede observarse en esta.



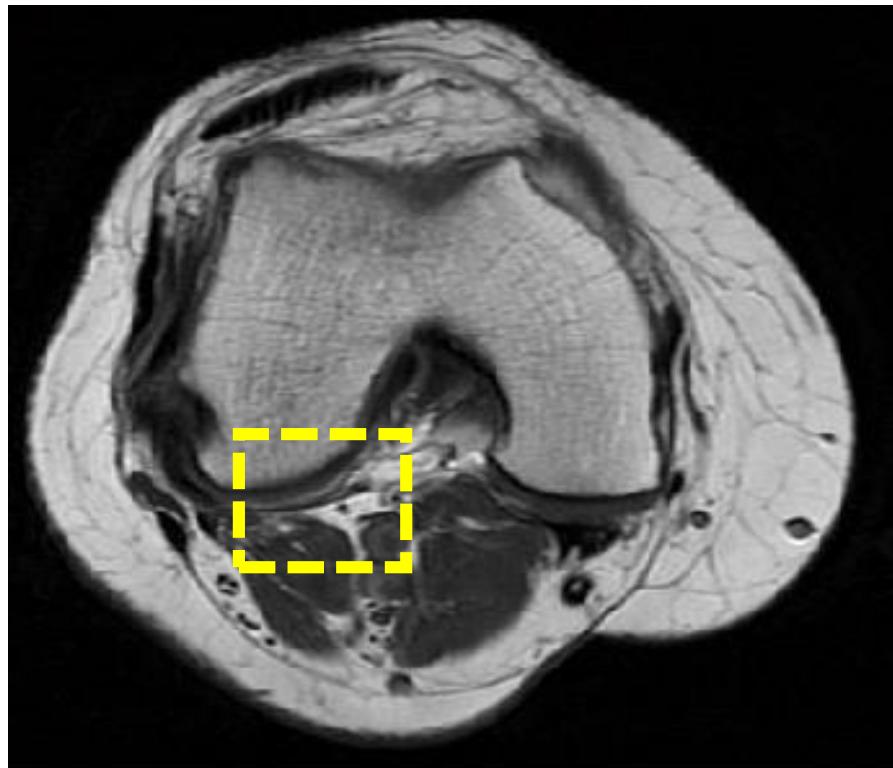
RESOLUCIÓN



Taller de Procesamiento Digital de Imágenes

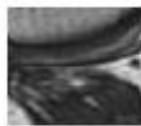
16 de Noviembre de 2018

RESOLUCIÓN

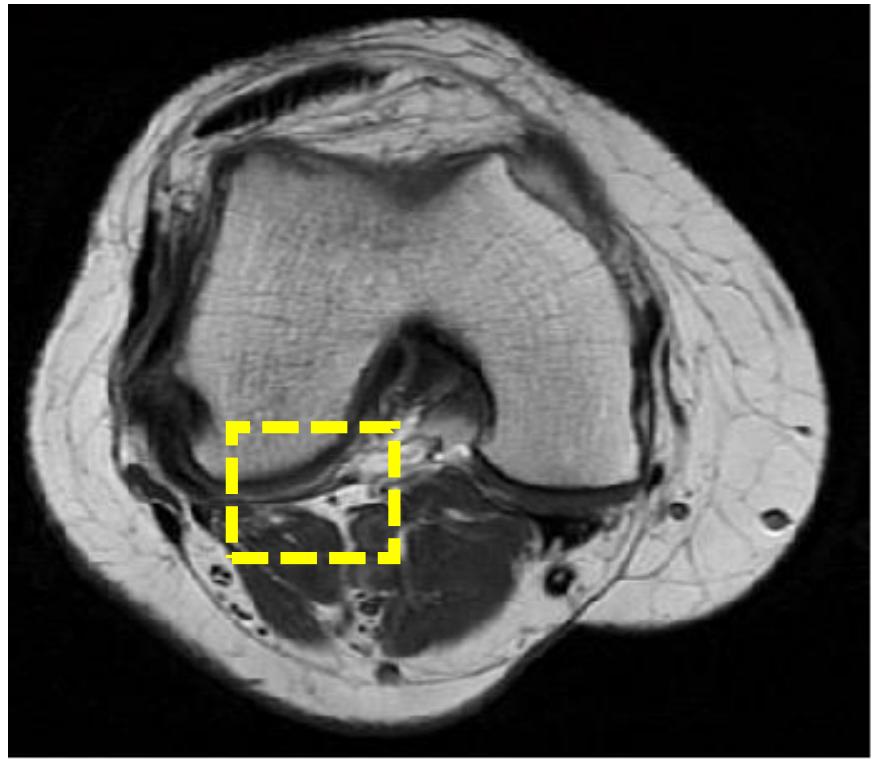


Resolución

1x



RESOLUCIÓN

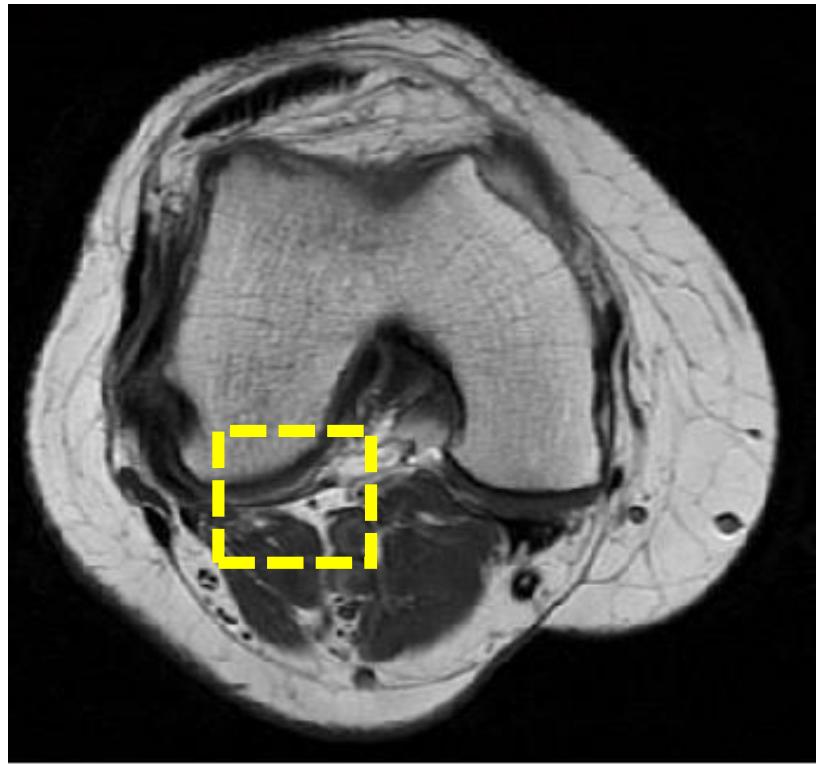


Resolución

2x



RESOLUCIÓN

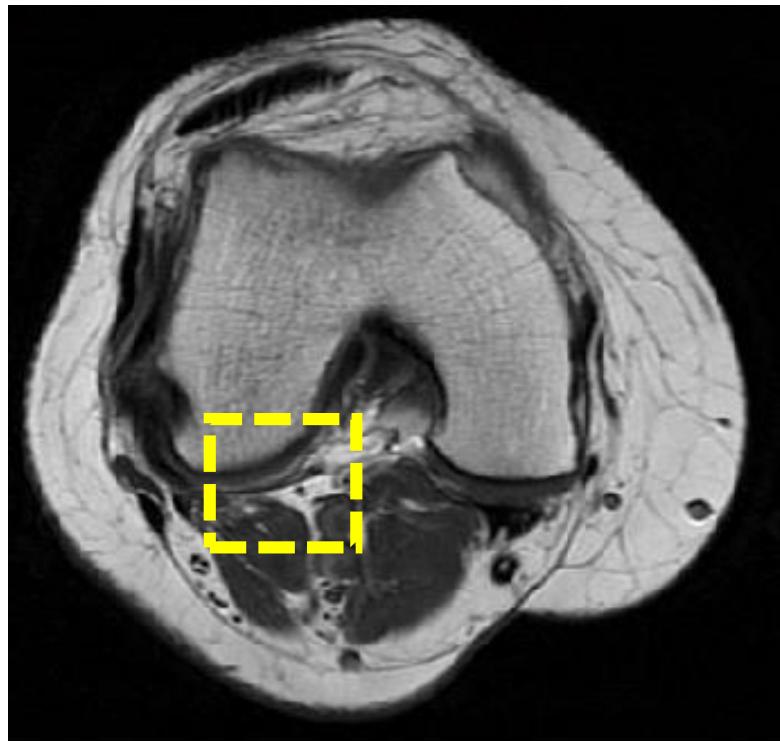


Resolución

3x



RESOLUCIÓN

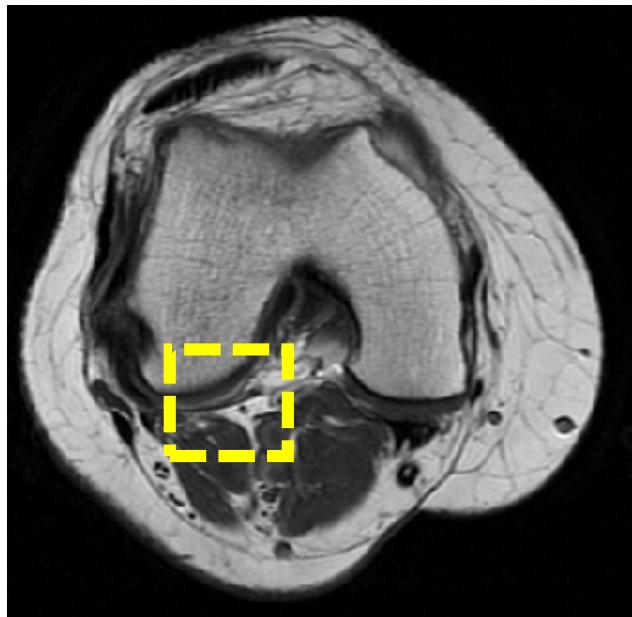


Resolución



4x

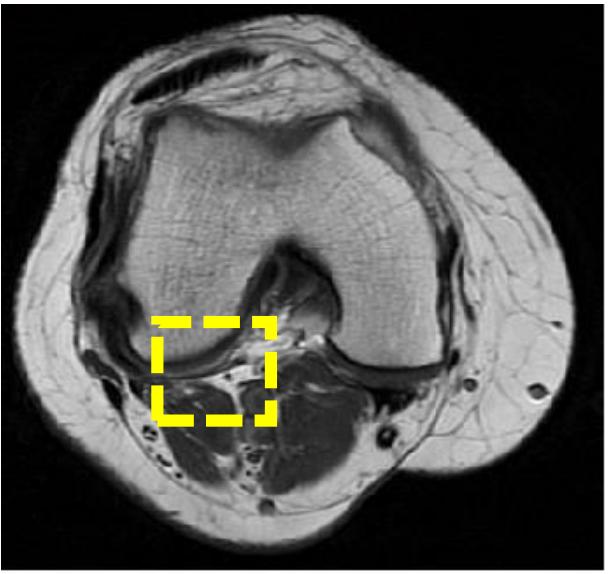
RESOLUCIÓN



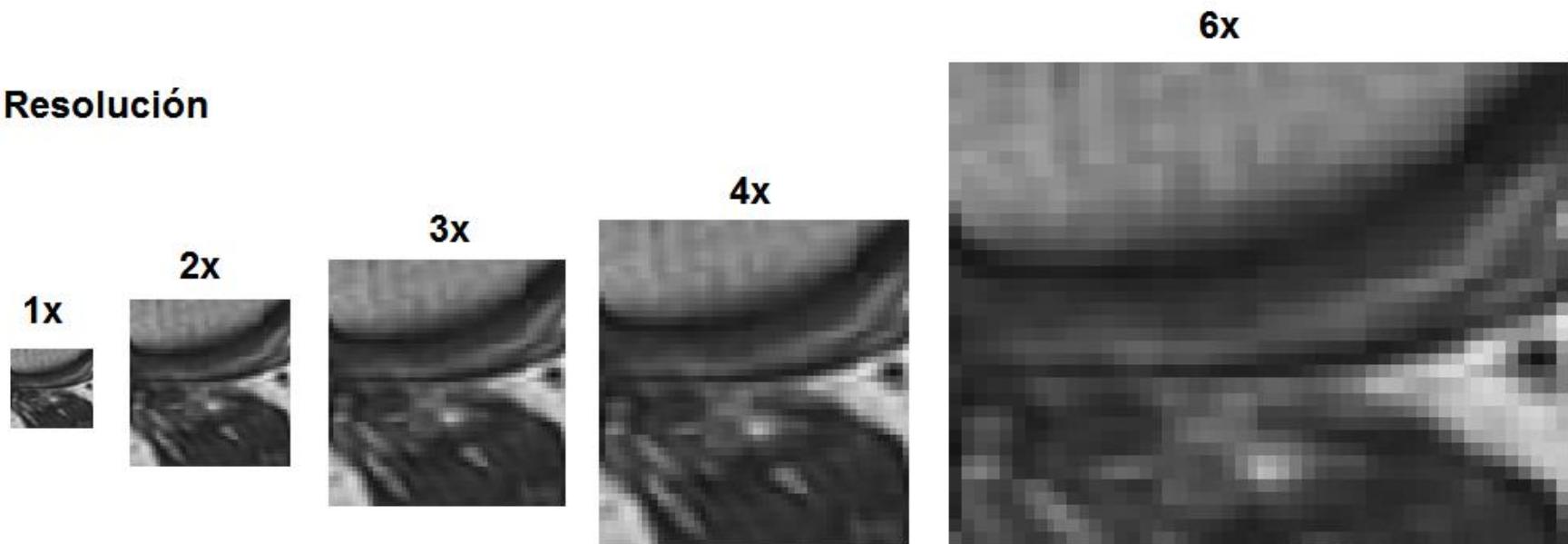
Resolución

6x

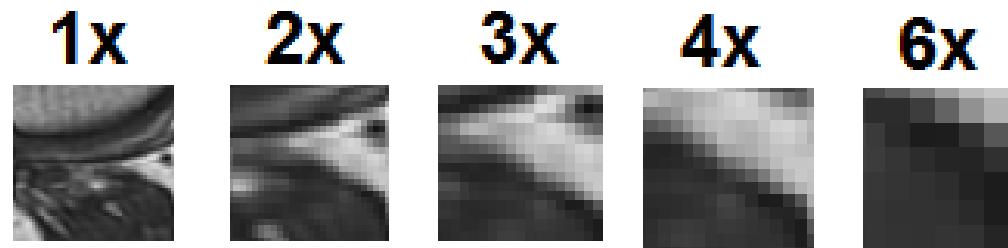




Resolución



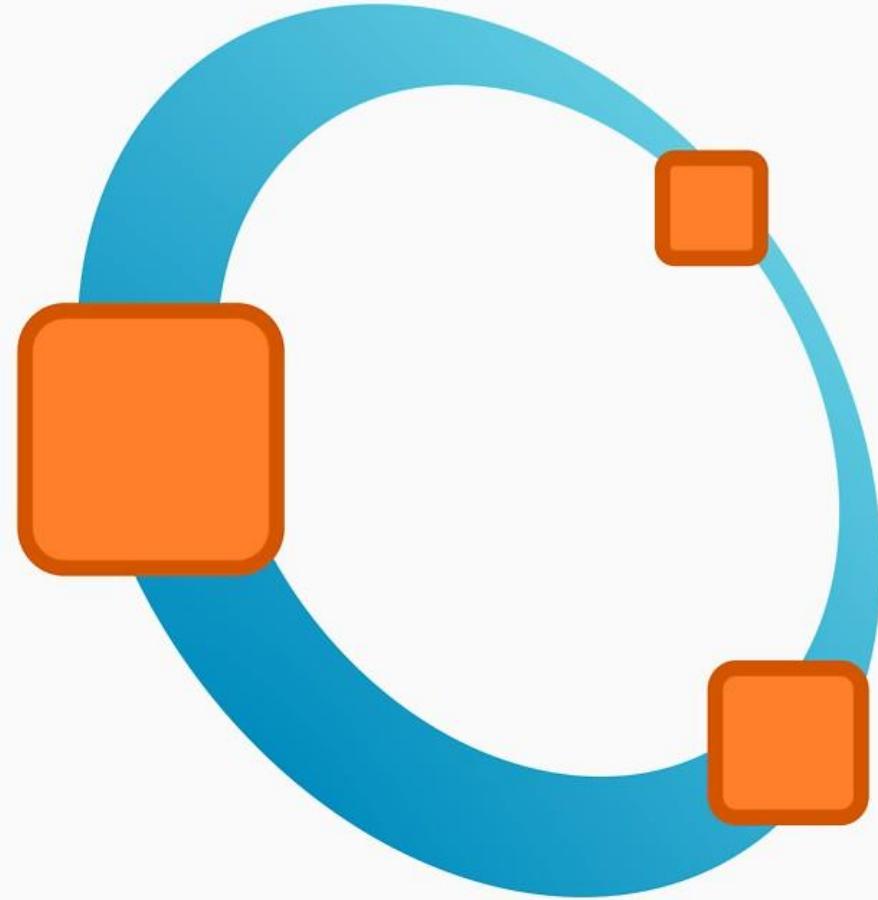
FOV



PRÁCTICA

```
    mirror_object to mirror
mirror_mod.mirror_object
    _operation = "MIRROR_X":
mirror_mod.use_x = True
mirror_mod.use_y = False
mirror_mod.use_z = False
    _operation = "MIRROR_Y":
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
    _operation = "MIRROR_Z":
mirror_mod.use_x = False
mirror_mod.use_y = False
mirror_mod.use_z = True

#selection at the end - add here
_ob.select= 1
mirr_ob.select=1
bpy.context.scene.objects.active = 
    ("Selected" + str(modifier))
mirror_ob.select = 0
bpy.context.selected_objects = 
    data.objects[one.name].select
int("please select exactly one object")
```



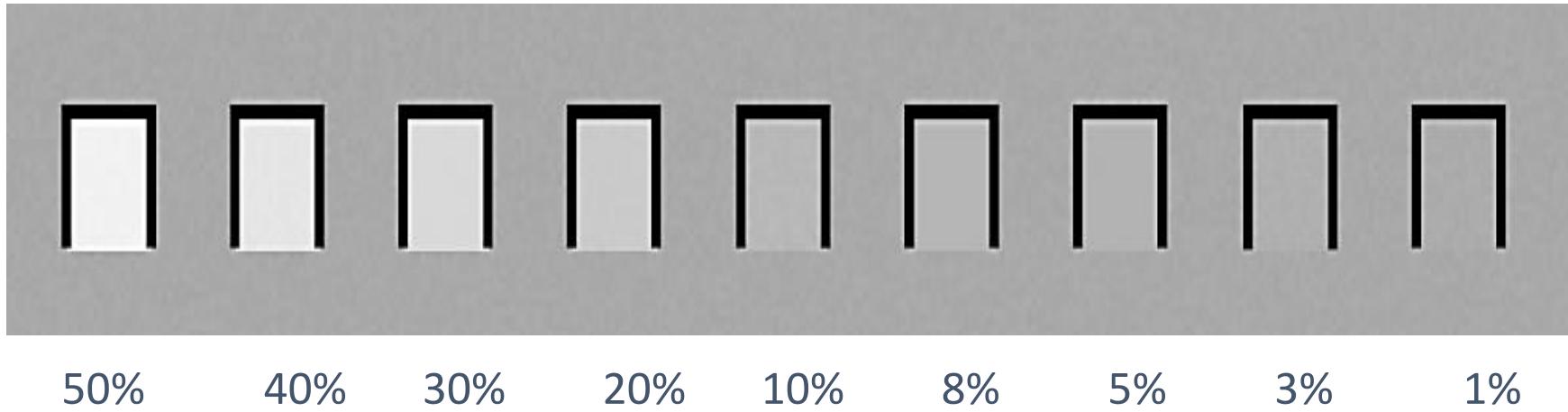
Ejecuta la práctica
ImageResize.m

CONTRASTE

Contraste de color es la diferencia en las características de dos o más colores que interactúan en un diseño afectando al modo en que se perciben.



CONTRASTE

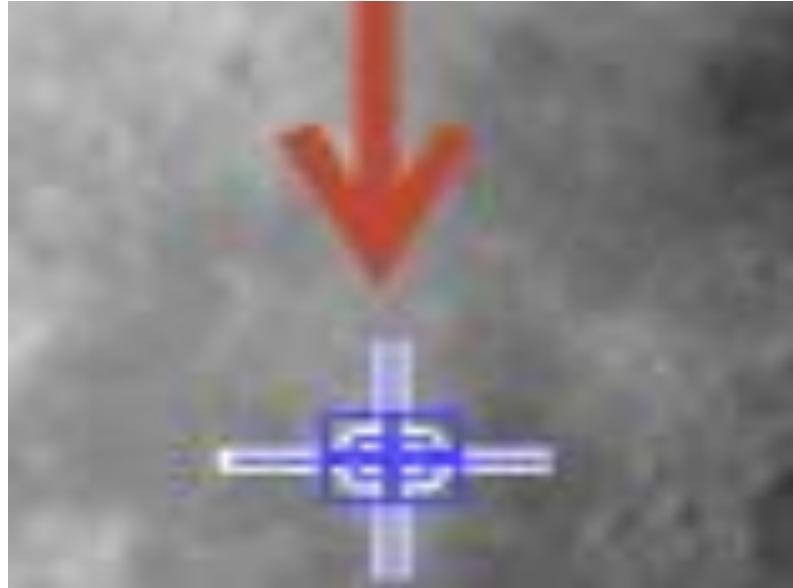


INTENSIDAD



Taller de Procesamiento Digital de Imágenes

INTENSIDAD



134	127	124
136	133	130
138	139	130

8 bits



$2^8 = 256$ bits

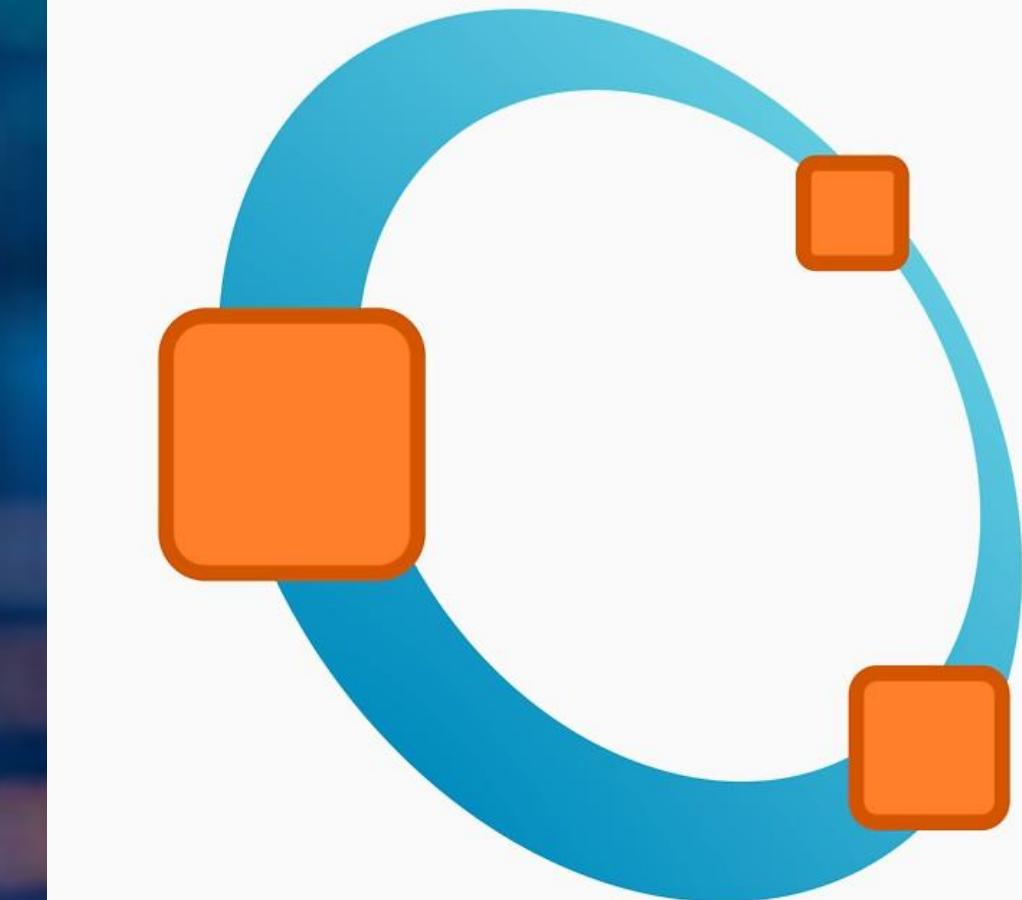
58	63	70	77	85	89	94	97	99	97	89	80	67	57	56	75	86	91	86	73	58	49	73
73	77	82	88	95	95	89	82	78	73	65	58	60	59	67	78	74	68	64	58	64	90	152
84	85	87	91	93	90	82	74	72	66	57	52	54	53	53	51	56	71	92	111	146	176	230
82	80	78	81	83	77	63	55	53	45	35	32	37	51	76	97	126	145	149	148	151	167	212
63	57	51	51	49	43	32	30	37	41	45	55	74	105	161	198	217	199	146	92	62	77	150
43	38	32	30	36	37	37	48	70	90	115	141	169	182	217	223	211	178	115	48	18	36	138
42	48	59	71	82	91	120	137	155	169	187	202	214	213	197	182	167	144	112	77	69	73	126
93	88	91	101	111	129	165	181	181	168	161	163	176	185	193	206	215	206	180	148	115	104	131
83	76	75	81	88	93	111	122	131	136	151	168	185	193	198	204	209	207	199	186	171	159	172
62	66	69	77	84	80	81	83	90	100	122	145	171	185	193	187	179	184	200	211	216	195	185
57	57	57	61	62	63	60	59	56	58	74	96	128	158	187	187	176	176	187	200	210	193	175
71	60	47	41	36	41	37	40	38	35	39	51	77	109	149	166	169	175	184	194	191	192	188
101	87	70	61	57	55	38	37	38	39	41	48	49	64	91	110	124	144	166	188	198	200	196

PRÁCTICA



```
mirror_mod.mirror_object = None
if mirror_mod.mirror_object == None:
    if mirror_mod.use_x == True:
        mirror_mod.use_y = False
        mirror_mod.use_z = False
    if mirror_mod.use_y == True:
        mirror_mod.use_x = False
        mirror_mod.use_z = False
    if mirror_mod.use_z == True:
        mirror_mod.use_x = False
        mirror_mod.use_y = False
else:
    if mirror_mod.use_x == True:
        mirror_mod.use_y = False
        mirror_mod.use_z = False
    if mirror_mod.use_y == True:
        mirror_mod.use_x = False
        mirror_mod.use_z = False
    if mirror_mod.use_z == True:
        mirror_mod.use_x = False
        mirror_mod.use_y = False

#selection at the end - add line
#_ob.select=1
#mirror_ob.select=1
context.scene.objects.active = eval("Selected" + str(modifier))
mirror_ob.select = 0
bpy.context.selected_objects = []
data.objects[one.name].select = 1
print("please select exactly one object")
#OPERATOR CLASSES
```



Ejecuta la práctica
ROI.m

RUIDO



Se considera un ruido a aquel elemento que se encuentre en la imagen que no forme parte de la muestra.

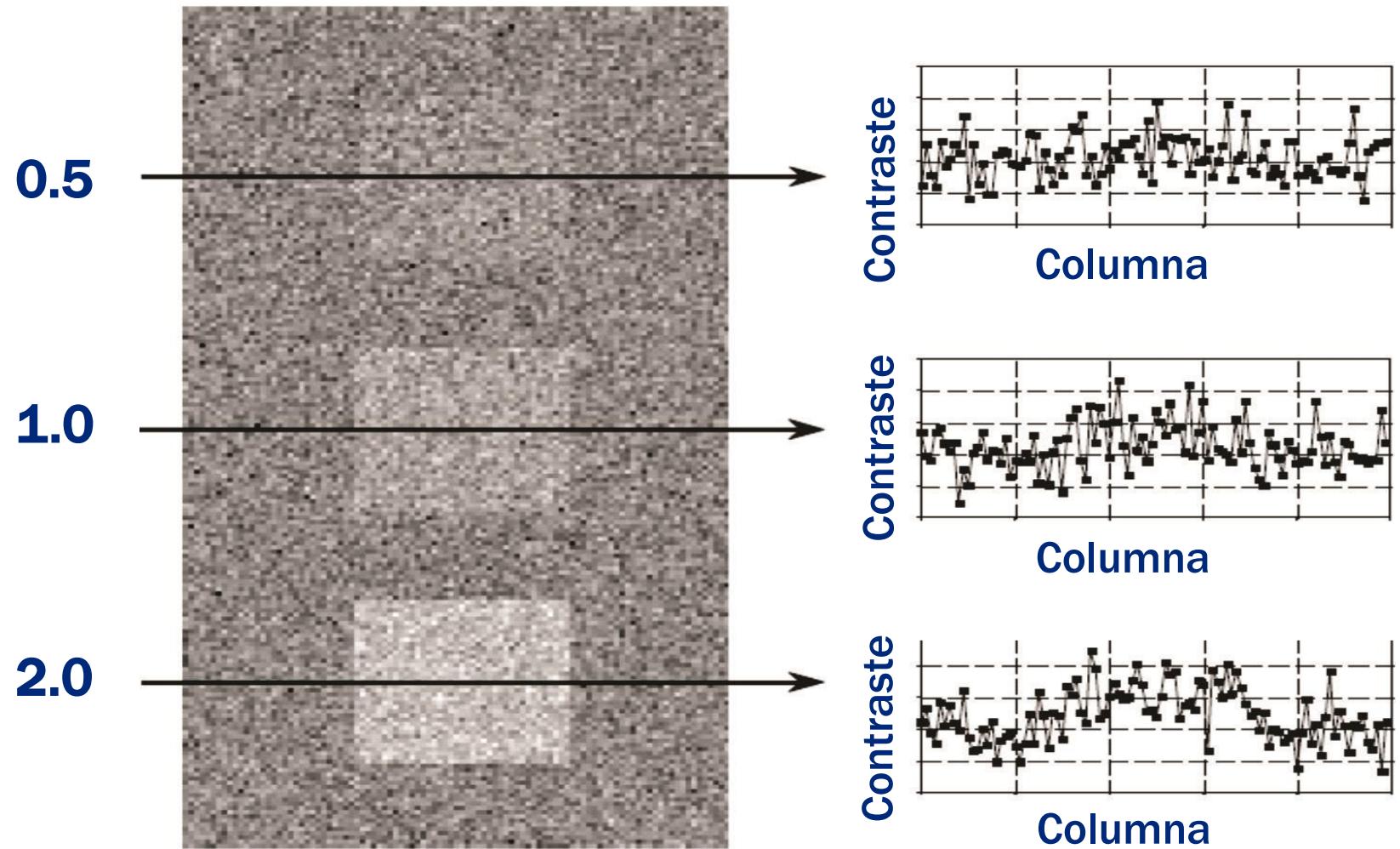




El SNR permite medir la relación de la señal y ruido en la imagen.

$$SNR = \frac{Señal}{Ruido}$$

SNR



CONTENIDO

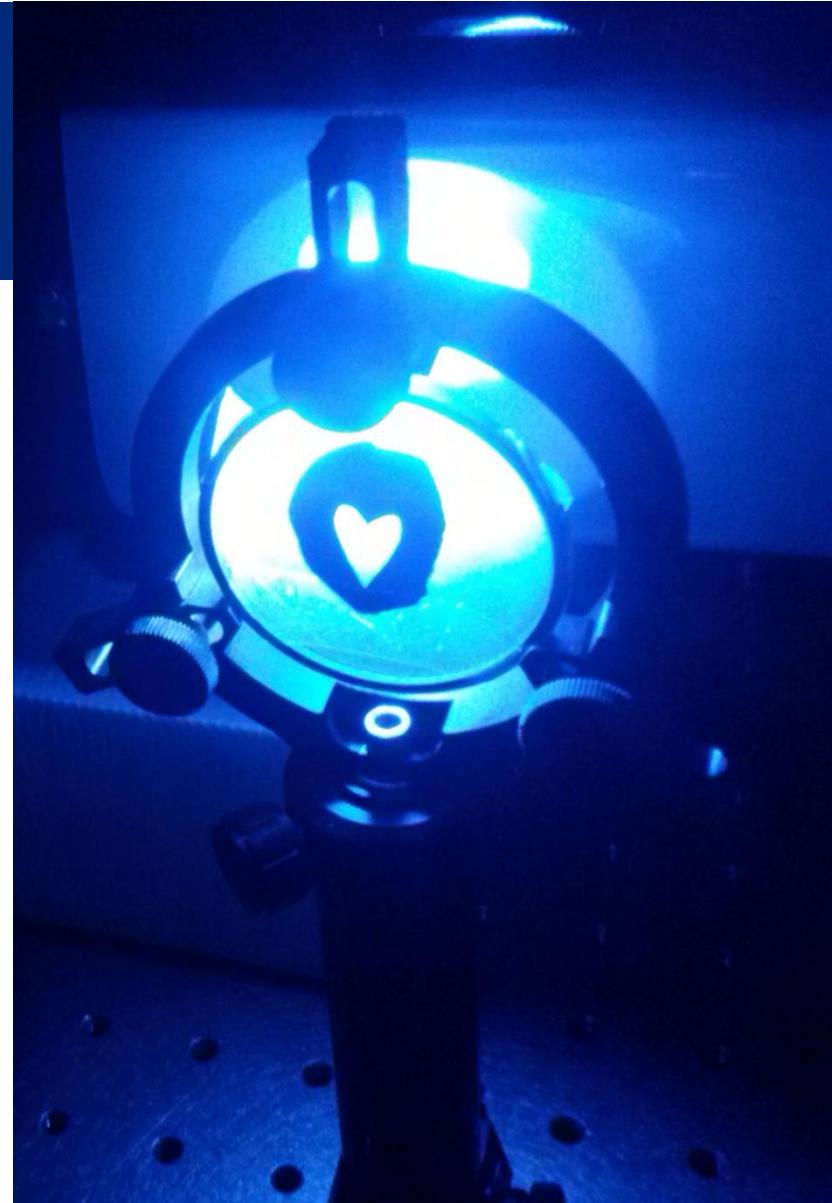
Introducción

Sistemas ópticos

Calibración de cámara

Introducción a las imágenes digitales

Procesamiento digital de imágenes



Taller de Procesamiento Digital de Imágenes

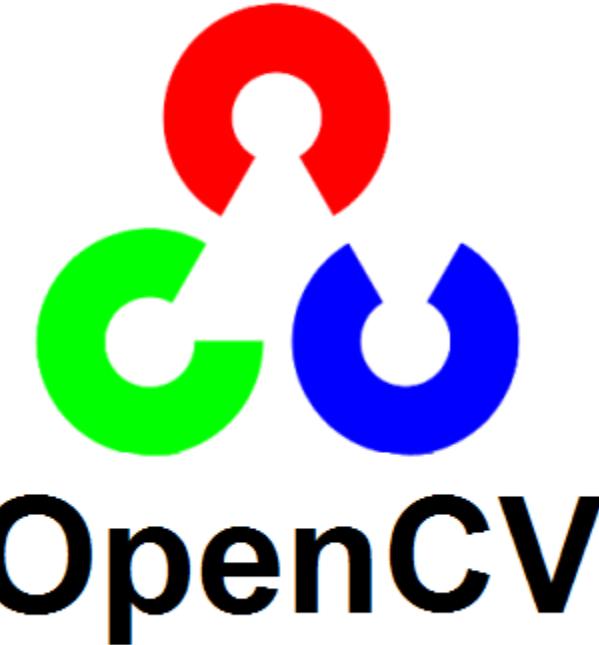
PRÁCTICA



```
    mirror_object to mirror_mod.mirror_object
    operation == "MIRROR_X":
        mirror_mod.use_x = True
        mirror_mod.use_y = False
        mirror_mod.use_z = False
    operation == "MIRROR_Y":
        mirror_mod.use_x = False
        mirror_mod.use_y = True
        mirror_mod.use_z = False
    operation == "MIRROR_Z":
        mirror_mod.use_x = False
        mirror_mod.use_y = False
        mirror_mod.use_z = True

    #selection at the end - add last
    _ob.select= 1
    mirr_ob.select=1
    context.scene.objects.active = eval(
        ("Selected" + str(modifier)))
    mirror_ob.select = 0
    bpy.context.selected_objects = []
    data.objects[one.name].select = 1
    print("please select exactly one object")
    return{'FINISHED'}
```

OPERATOR CLASSES



Ejecuta las prácticas.

REFERENCIAS

Gonzalez, R. C., & Woods, R. E. (2002). Digital image processing second edition. Beijing: Publishing House of Electronics Industry, 455.

Ponce, J., Forsyth, D., Willow, E. P., Antipolis-Méditerranée, S., d' activité

RAweb, R., Inria, L., & Alumni, I. (2011). Computer vision: a modern approach. Computer, 16(11).

Hecht, E. (2002). Optics. 2002. Pearson Education, Inc.,, 360, 366-367.

Apuntes personales.

Referencias diversas de Octave, Python y OpenCV.

Thank you!



Ing. Ángel Hernández, ahernandezp1800@alumno.ipn.mx

M. En R. Karen Flores, kfloresr1800@alumno.ipn.mx

M. En TA. Sandra de la Fuente, sdelafuenteb1400@alumno.ipn.mx

CICATA IPN Unidad Querétaro
Tel. (442) 2290804 Ext. 81021/ 81047
Cerro Blanco No. 141, Col. Colinas del Cimatario. C.P.
76090. Querétaro, Qro.
www.cicataqroipn.mx

