



# Computer vision in the new era of Artificial Intelligence and Deep Learning

## Visión por computador en la nueva era de la Inteligencia Artificial y el Deep Learning

**Rubén Usamentiaga\*, Alberto Fernández°**

**\* University of Oviedo**

**° TSK**

Gijón (Spain)  
5 – 16 April 2021



<https://github.com/albertofernandezvillan/computer-vision-and-deep-learning-course>

# Pillow



**Notebook: pil\_introduction\_python.ipynb**



[computer-vision-and-deep-learning-course/pil\\_introduction\\_python.ipynb at main · albertofernandezvillan/computer-vision-and-deep-learning-course \(github.com\)](https://github.com/albertofernandezvillan/computer-vision-and-deep-learning-course/blob/main/pil_introduction_python.ipynb)



Open in Colab

[pil\\_introduction\\_python.ipynb - Colaboratory \(google.com\)](https://colab.research.google.com/github/albertofernandezvillan/computer-vision-and-deep-learning-course/blob/main/pil_introduction_python.ipynb)



<https://github.com/albertofernandezvillan/computer-vision-and-deep-learning-course>

# Pillow introduction

- Pillow is the friendly PIL fork by Alex Clark and Contributors. PIL is the Python Imaging Library by Fredrik Lundh and Contributors. As of 2019, Pillow development is supported by Tidelift.
- Use the following import convention

```
import PIL
```

# Pillow main features

To load an image use `Image.open()`

Note that PIL uses **RGB** mode

```
from PIL import Image

img = Image.open("/content/face_test.png")
print(img)
print("Format: '{}', size: '{}', mode: '{}'"
      .format(img.format,
               img.size, img.mode))
```

```
<PIL.PngImagePlugin.PngImageFile image mode=RGB
size=482x504 at 0x7F3A5722DC50> Format: 'PNG', size:
'(482, 504)', mode: 'RGB'
```

# Pillow main features

To convert images between different pixel representations use `img.convert()`.

```
from PIL import Image

# This converts RGB image to grayscale
img = Image.open("/content/face_test.png")
img_1 = img.convert("L")
```

```
from PIL import Image, ImageOps

# For this purpose, we can also use ImageOps module
img_2 = ImageOps.grayscale(img)
```



# Pillow main features

To save an image on disk: `img.save()`

You can specify a different extension from its original, and the saved image will be converted to the specified format.

```
from PIL import Image

# Load image:
img = Image.open("/content/face_test.png")

# Save it on disk:
img.save("face_test.jpg", quality=95)
```

# Pillow main features

## Convert PIL Image to OpenCV format

```
import cv2
from PIL import Image
import numpy as np

img = Image.open("/content/face_test.png")

img_bgr = cv2.cvtColor(np.asarray(img), cv2.COLOR_RGB2BGR)
```

## Convert OpenCV format to PIL Image

```
from PIL import Image

pil_image = Image.fromarray(img_rgb)
```

# Pillow modules

[Reference — Pillow \(PIL Fork\) 3.0.0 documentation](#)

Image Module

ImageChops

ImageDraw Module

ImageEnhance Module

ImageFont Module

ImageMorph Module

ImagePalette Module

ImageStat Module

ImageCms Module

ImageOps Module

ImageColor Module

ExifTags Module

ImageFile Module

ImageFilter Module



# Pillow modules

[Reference — Pillow \(PIL Fork\) 3.0.0 documentation](#)

## ImageFilter Module

```
img.filter(ImageFilter.UnsharpMask)
```



# Pillow



## Recommended lectures

- PIL Modules:
  - <https://pillow.readthedocs.io/en/3.0.x/reference/index.html>



<https://github.com/albertofernandezvillan/computer-vision-and-deep-learning-course>