

00-Opening-Image-Files-in-a-Notebook

August 12, 2025

```
[1]: import numpy as np
import cv2
import matplotlib.pyplot as plt
%matplotlib inline
```

WARNING! Make sure to provide the correct path like we do in our notebooks!
ALWAYS confirm that your file path is correct before posting a question on this.

```
[2]: # WONT GIVE ERROR! GIVES NONE INSTEAD!!!
img = cv2.imread("some/wrong/path.jpg")
```

```
[ WARN:001.999] global loadsave.cpp:248 findDecoder
imread_('some/wrong/path.jpg'): can't open/read file: check file path/integrity
```

```
[3]: print(img)
```

None

```
[4]: img = cv2.imread('puppy.jpg')
```

```
[5]: img
```

```
[5]: array([[78, 81, 95],
           [80, 83, 97],
           [81, 84, 98],
           ...,
           [22, 27, 25],
           [22, 27, 25],
           [22, 27, 25]],

           [[78, 81, 95],
           [79, 82, 96],
           [79, 82, 96],
           ...,
           [22, 27, 25],
           [22, 27, 25],
           [22, 27, 25]])
```

```

[[78, 81, 95],
 [77, 80, 94],
 [77, 80, 94],
 ...,
 [22, 27, 25],
 [22, 27, 25],
 [22, 27, 25]],

...,

[[20, 29, 19],
 [21, 30, 20],
 [21, 30, 20],
 ...,
 [22, 30, 23],
 [23, 31, 24],
 [23, 31, 24]],

[[21, 30, 20],
 [21, 30, 20],
 [20, 29, 19],
 ...,
 [22, 30, 23],
 [23, 31, 24],
 [23, 31, 24]],

[[21, 30, 20],
 [20, 29, 19],
 [20, 29, 19],
 ...,
 [22, 30, 23],
 [23, 31, 24],
 [23, 31, 24]]], dtype=uint8)

```

```

[6]: img_bgr = cv2.imread('puppy.jpg')
     plt.imshow(img_bgr)

```

```

[6]: <matplotlib.image.AxesImage at 0x73440598c410>

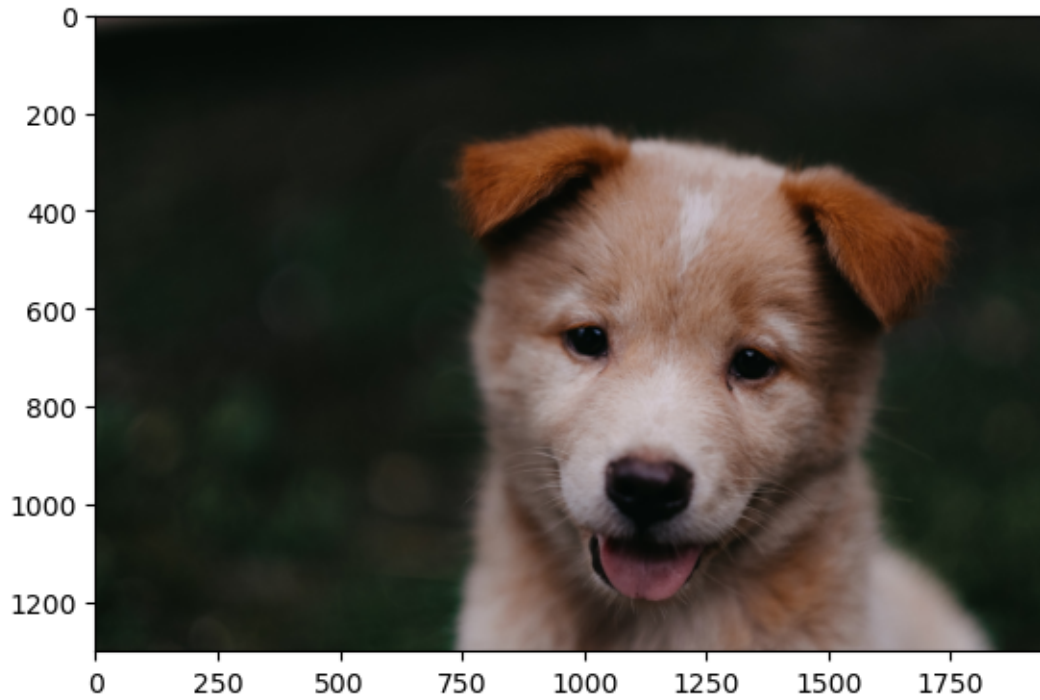
```



The image has been correctly loaded by openCV as a numpy array, but the color of each pixel has been sorted as BGR. Matplotlib's plot expects an RGB image so, for a correct display of the image, it is necessary to swap those channels. This operation can be done either by using openCV conversion functions `cv2.cvtColor()` or by working directly with the numpy array.

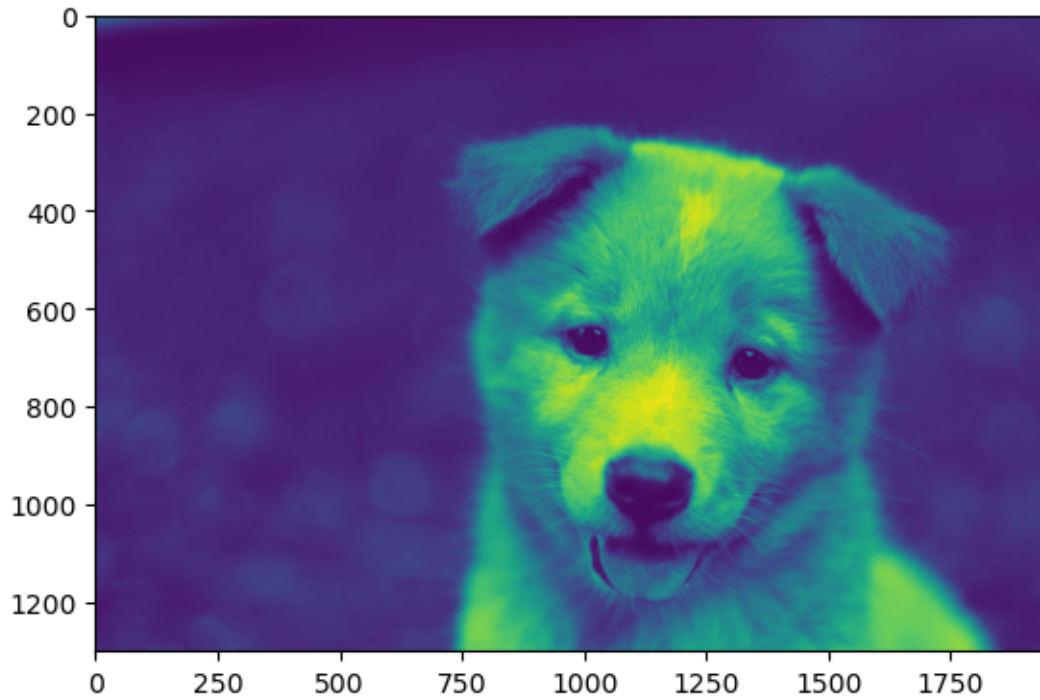
```
[7]: img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
plt.imshow(img_rgb)
```

```
[7]: <matplotlib.image.AxesImage at 0x734404990cd0>
```



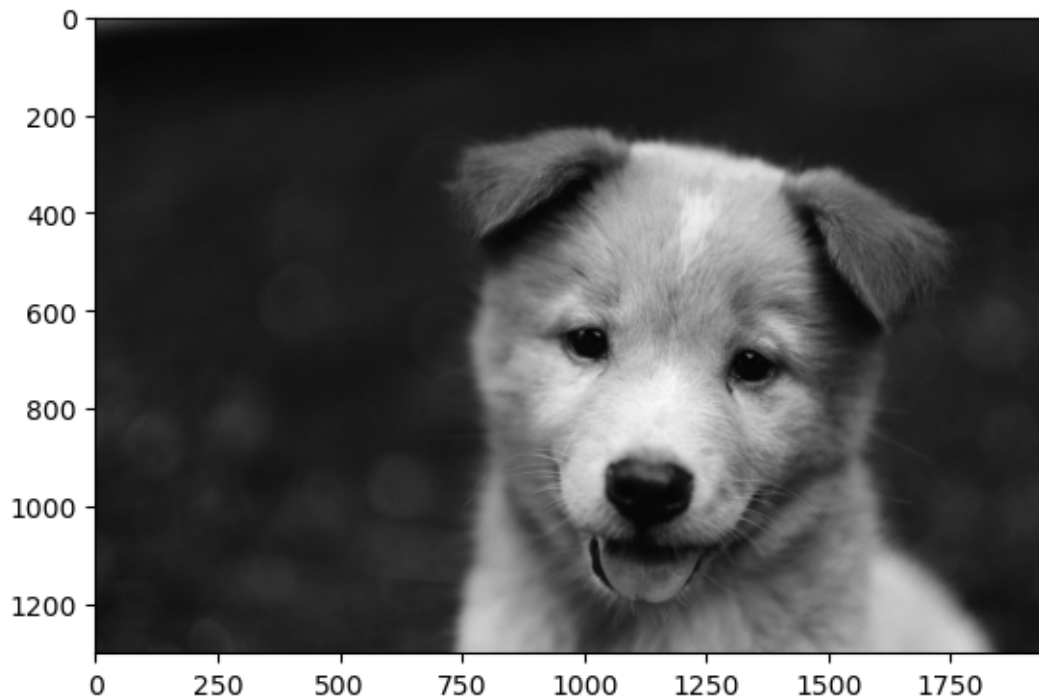
```
[8]: img_gray = cv2.imread('puppy.jpg',cv2.IMREAD_GRAYSCALE)
plt.imshow(img_gray)
```

```
[8]: <matplotlib.image.AxesImage at 0x734404084410>
```



```
[9]: img_gray = cv2.imread('puppy.jpg',cv2.IMREAD_GRAYSCALE)
plt.imshow(img_gray,cmap='gray')
```

```
[9]: <matplotlib.image.AxesImage at 0x7344040f5890>
```



0.1 Resize Images

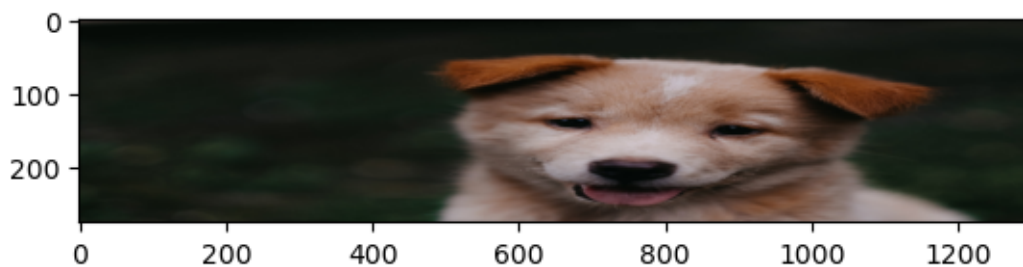
```
[10]: img_rgb.shape  
      # width, height, color channels
```

```
[10]: (1300, 1950, 3)
```

```
[11]: img =cv2.resize(img_rgb,(1300,275))
```

```
[12]: plt.imshow(img)
```

```
[12]: <matplotlib.image.AxesImage at 0x734404128cd0>
```



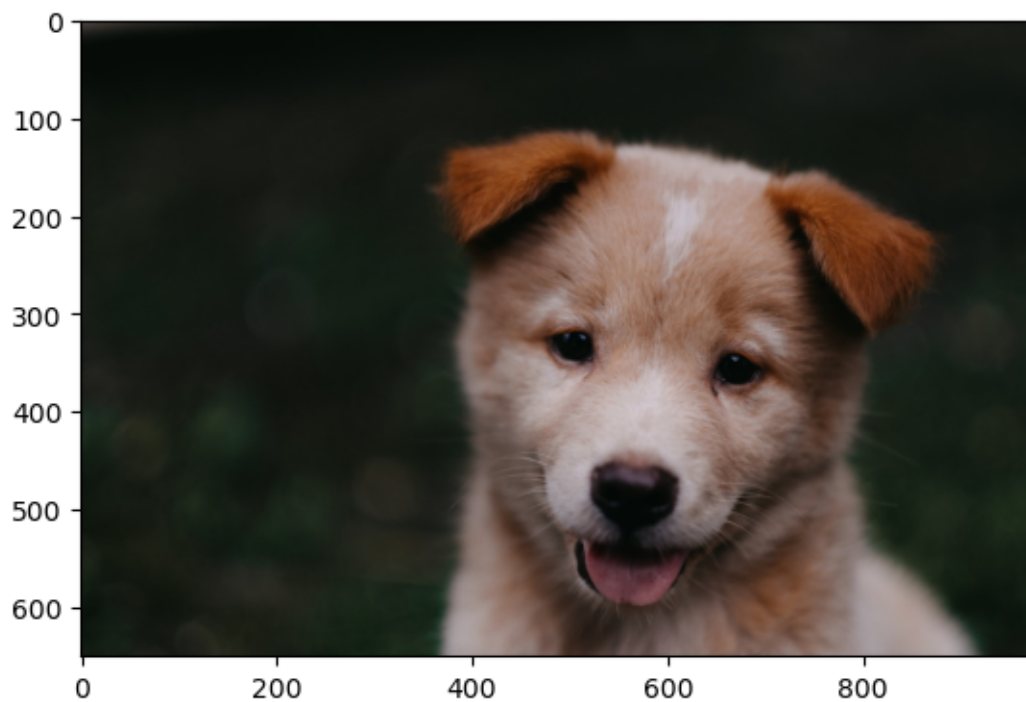
0.1.1 By ratio

```
[13]: w_ratio = 0.5  
      h_ratio = 0.5
```

```
[14]: new_img = cv2.resize(img_rgb,(0,0),img,w_ratio,h_ratio)
```

```
[15]: plt.imshow(new_img)
```

```
[15]: <matplotlib.image.AxesImage at 0x7343fd74d890>
```



```
[16]: plt.imshow(new_img)
```

```
[16]: <matplotlib.image.AxesImage at 0x7343fd7b8e10>
```

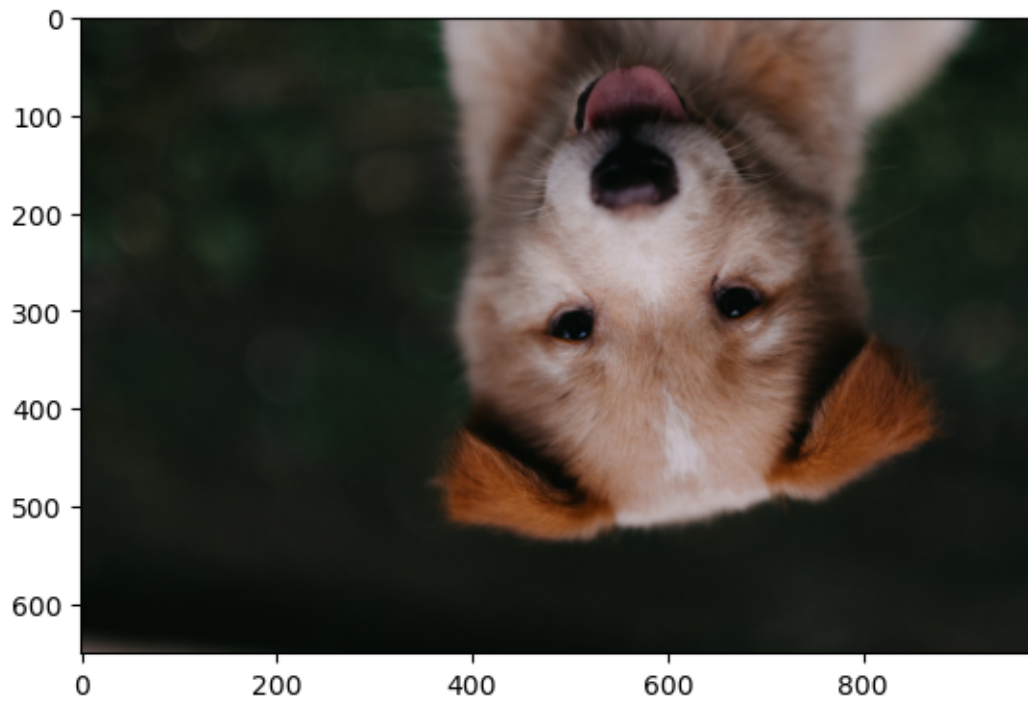


0.1.2 Flipping Images

```
[17]: %matplotlib inline
```

```
[18]: # Along central x axis  
new_img = cv2.flip(new_img,0)  
plt.imshow(new_img)
```

```
[18]: <matplotlib.image.AxesImage at 0x7343fd62c410>
```

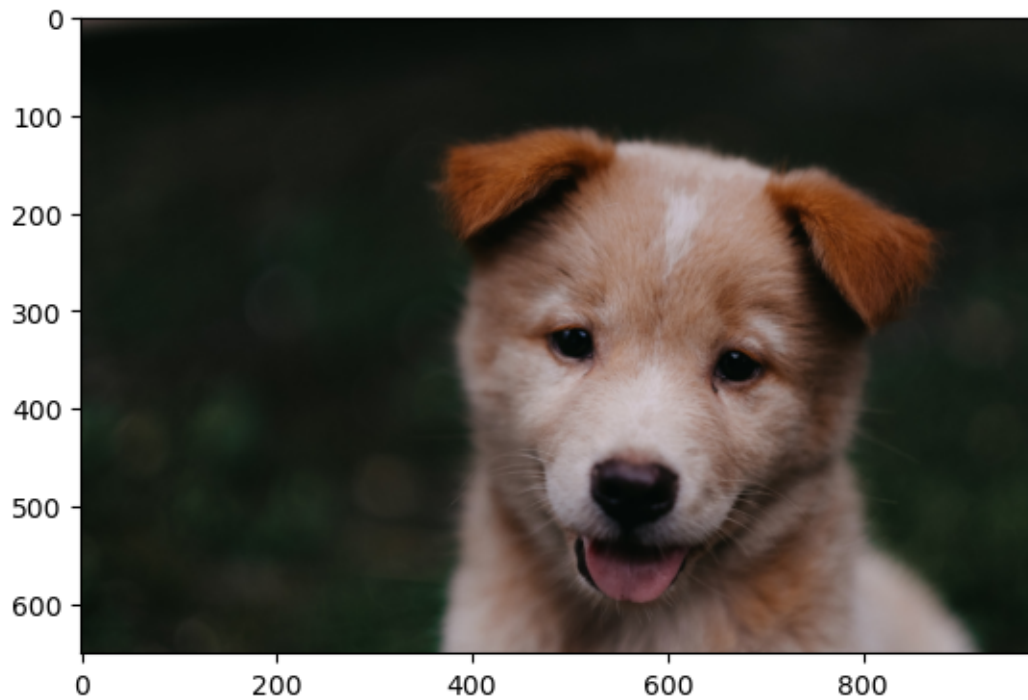
```
[19]: # Along central y axis  
new_img = cv2.flip(new_img,1)  
plt.imshow(new_img)
```

```
[19]: <matplotlib.image.AxesImage at 0x7343fd6ac410>
```



```
[20]: # Along both axis  
new_img = cv2.flip(new_img,-1)  
plt.imshow(new_img)
```

```
[20]: <matplotlib.image.AxesImage at 0x7343fd534410>
```



1 Saving Image Files

```
[21]: type(new_img)
```

```
[21]: numpy.ndarray
```

```
[22]: cv2.imwrite('my_new_picture.jpg',new_img)
```

```
[22]: True
```

Keep in mind, the above stored the BGR version of the image.

1.1 Larger Displays in the Notebook

```
[23]: fig = plt.figure(figsize=(10,8))  
      ax = fig.add_subplot(111)  
      ax.imshow(new_img)
```

```
[23]: <matplotlib.image.AxesImage at 0x7343fd5a7310>
```



[]: