

```
import numpy as np
```

```
In [1]: import matplotlib.pyplot as plt
```

```
In [2]: from PIL import Image #python imaging Library(PIL)
```

```
In [7]: img=Image.open(r"D:\DEVI (MS-OFFICE -WORD- PAD)\OneDrive\Pictures\room.jpg")
img
```

```
Out[7]:
```



```
In [8]: type(img)
```

```
Out[8]: PIL.JpegImagePlugin.JpegImageFile
```

```
In [9]: img.show()
```

```
In [12]: img6=Image.open(r"D:\DEVI (MS-OFFICE -WORD- PAD)\OneDrive\Pictures\bulb1.webp")
img6
```

Out[12]:

In [1]: `import numpy as np`In [2]: `ones_arr=np.ones((3,3))`In [3]: `ones_arr`Out[3]: `array([[1., 1., 1.],  
 [1., 1., 1.],  
 [1., 1., 1.]])`In [4]: `ones_arr=np.ones((5,5),dtype=int)`In [5]: `ones_arr`Out[5]: `array([[1, 1, 1, 1, 1],  
 [1, 1, 1, 1, 1],  
 [1, 1, 1, 1, 1],  
 [1, 1, 1, 1, 1],  
 [1, 1, 1, 1, 1]])`In [6]: `zeros_arr=np.zeros((3,3),dtype=int)`In [7]: `zeros_arr`

```
Out[7]: array([[0, 0, 0],  
               [0, 0, 0],  
               [0, 0, 0]])
```

```
In [8]: ones_arr
```

```
Out[8]: array([[1, 1, 1, 1, 1],  
               [1, 1, 1, 1, 1],  
               [1, 1, 1, 1, 1],  
               [1, 1, 1, 1, 1],  
               [1, 1, 1, 1, 1]])
```

```
In [9]: ones_arr*225
```

```
Out[9]: array([[225, 225, 225, 225, 225],  
               [225, 225, 225, 225, 225],  
               [225, 225, 225, 225, 225],  
               [225, 225, 225, 225, 225],  
               [225, 225, 225, 225, 225]])
```

```
In [10]: import matplotlib.pyplot as plt
```

```
In [11]: %matplotlib inline
```

```
In [12]: from PIL import Image
```

```
In [14]: horse_img=Image.open(r"D:\DEVI (MS-OFFICE -WORD- PAD)\OneDrive\Pictures\horse.jpg")
```

```
In [15]: horse_img
```

```
Out[15]:
```



```
In [16]: type(horse_img)
```

```
Out[16]: PIL.JpegImagePlugin.JpegImageFile
```

```
In [17]: horse_arr=np.asarray(horse_img)
horse_arr
```

```
Out[17]: array([[[15, 17, 29],
   [15, 17, 29],
   [15, 17, 29],
   ...,
   [23, 38, 35],
   [19, 34, 31],
   [14, 30, 27]],

   [[15, 17, 29],
   [15, 17, 29],
   [15, 17, 29],
   ...,
   [24, 39, 36],
   [22, 37, 34],
   [20, 36, 33]],

   [[15, 17, 29],
   [15, 17, 29],
   [15, 17, 29],
   ...,
   [26, 41, 38],
   [25, 40, 37],
   [24, 40, 37]],

   ...,

   [[49, 50, 44],
   [40, 41, 33],
   [35, 34, 29],
   ...,
   [14, 30, 29],
   [13, 25, 25],
   [11, 23, 23]],

   [[45, 50, 44],
   [38, 43, 36],
   [33, 35, 30],
   ...,
   [11, 25, 25],
   [12, 24, 24],
   [16, 26, 27]],

   [[33, 40, 33],
   [33, 40, 33],
   [33, 38, 32],
   ...,
   [12, 26, 26],
   [16, 26, 27],
   [22, 32, 33]]], dtype=uint8)
```

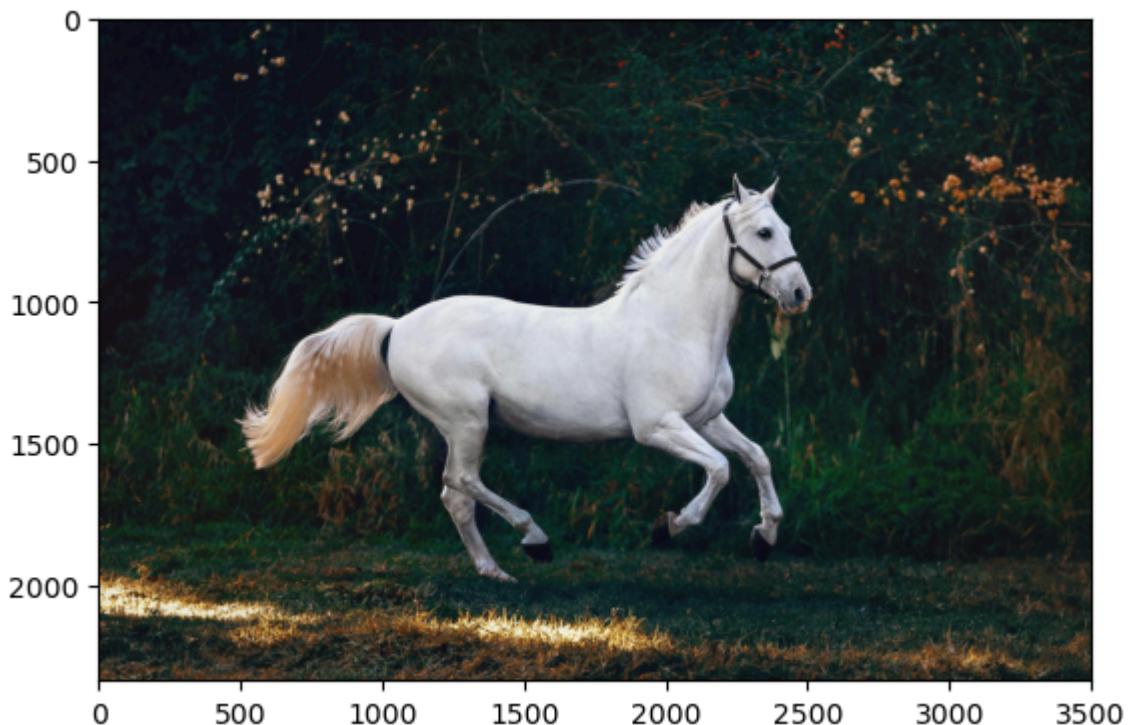
```
In [18]: type(horse_arr)
```

```
Out[18]: numpy.ndarray
```

```
In [19]: horse_arr.shape
```

Out[19]: (2334, 3502, 3)

In [30]: `plt.imshow(horse_arr)  
plt.show()`



In [21]: `horse_red=horse_arr.copy()`

In [22]: `horse_red`

```
Out[22]: array([[[15, 17, 29],  
                 [15, 17, 29],  
                 [15, 17, 29],  
                 ...,  
                 [23, 38, 35],  
                 [19, 34, 31],  
                 [14, 30, 27]],  
  
                [[15, 17, 29],  
                 [15, 17, 29],  
                 [15, 17, 29],  
                 ...,  
                 [24, 39, 36],  
                 [22, 37, 34],  
                 [20, 36, 33]],  
  
                [[15, 17, 29],  
                 [15, 17, 29],  
                 [15, 17, 29],  
                 ...,  
                 [26, 41, 38],  
                 [25, 40, 37],  
                 [24, 40, 37]],  
  
                ...,  
  
                [[49, 50, 44],  
                 [40, 41, 33],  
                 [35, 34, 29],  
                 ...,  
                 [14, 30, 29],  
                 [13, 25, 25],  
                 [11, 23, 23]],  
  
                [[45, 50, 44],  
                 [38, 43, 36],  
                 [33, 35, 30],  
                 ...,  
                 [11, 25, 25],  
                 [12, 24, 24],  
                 [16, 26, 27]],  
  
                [[33, 40, 33],  
                 [33, 40, 33],  
                 [33, 38, 32],  
                 ...,  
                 [12, 26, 26],  
                 [16, 26, 27],  
                 [22, 32, 33]]], dtype=uint8)
```

```
In [24]: horse_arr==horse_red
```

```
Out[24]: array([[[ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True],
   ...,
   [ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True]],

[[ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True],
   ...,
   [ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True]],

[[ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True],
   ...,
   [ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True]],

...,

[[ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True],
   ...,
   [ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True]],

[[ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True],
   ...,
   [ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True]],

[[ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True],
   ...,
   [ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True]],

[[ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True],
   ...,
   [ True,  True,  True],
   [ True,  True,  True],
   [ True,  True,  True]]])
```

```
In [31]: plt.imshow(horse_red)
plt.show()
```



```
In [26]: horse_red.shape
```

```
Out[26]: (2334, 3502, 3)
```

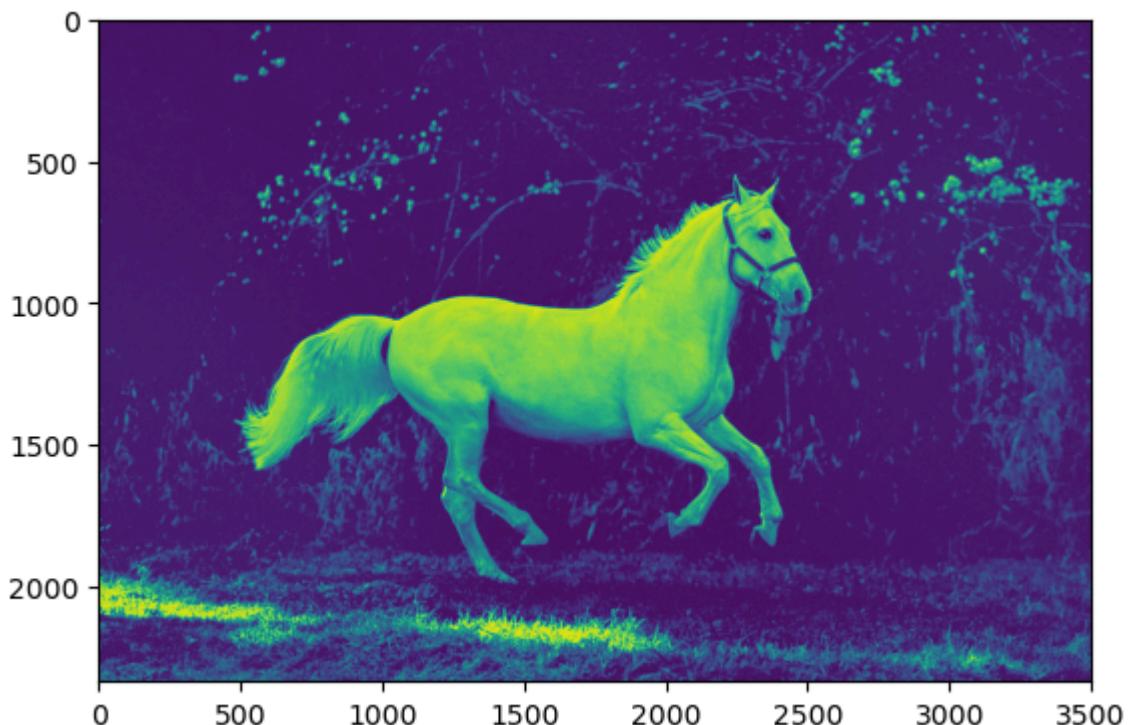
```
In [29]: # R G B  
plt.imshow(horse_red[:, :, 0])  
plt.show()
```



```
In [28]: horse_red[:, :, 0]
```

```
Out[28]: array([[15, 15, 15, ..., 23, 19, 14],  
                 [15, 15, 15, ..., 24, 22, 20],  
                 [15, 15, 15, ..., 26, 25, 24],  
                 ...,  
                 [49, 40, 35, ..., 14, 13, 11],  
                 [45, 38, 33, ..., 11, 12, 16],  
                 [33, 33, 33, ..., 12, 16, 22]], dtype=uint8)
```

```
In [33]: plt.imshow(horse_red[:, :, 0])  
plt.show()
```



```
In [34]: horse_red[:, :, 0]  
plt.show()
```

```
In [35]: plt.imshow(horse_red[:, :, 0], cmap='Greys')
```

```
Out[35]: <matplotlib.image.AxesImage at 0x1ebab3958b0>
```

```
In [36]: plt.show()
```



```
In [40]: plt.imshow(horse_red[:, :, 1], cmap='Greys')
```

```
Out[40]: <matplotlib.image.AxesImage at 0x1ebad4b5fa0>
```

```
In [41]: plt.show()
```



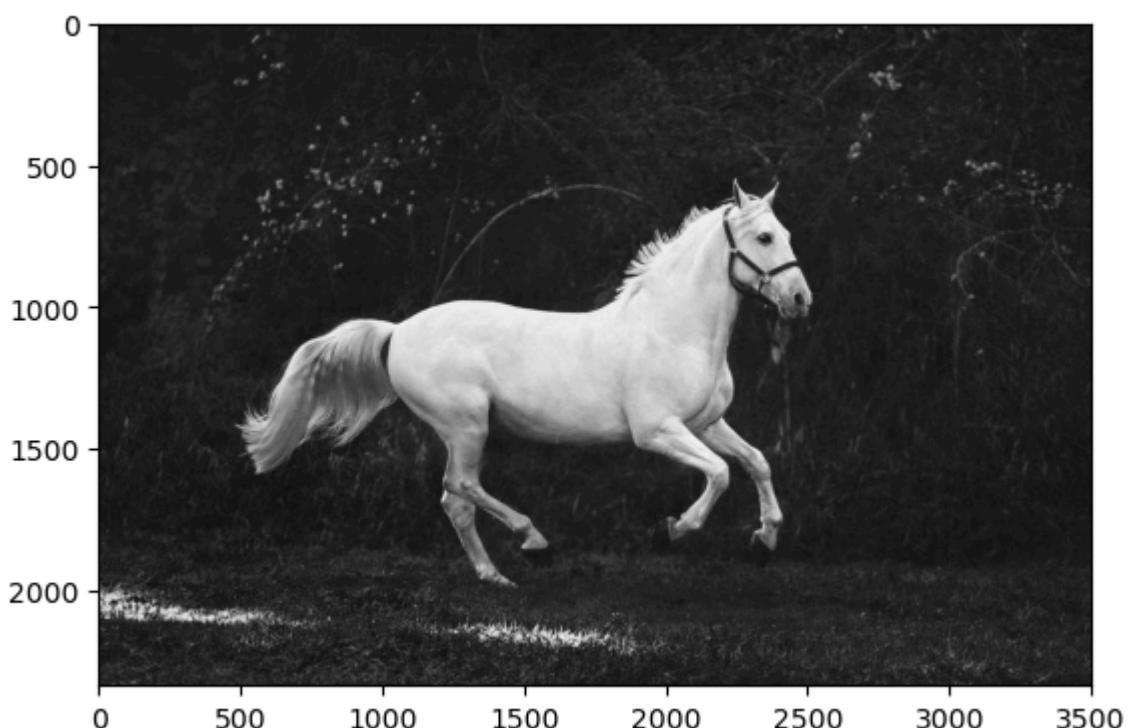
```
In [42]: plt.imshow(horse_red[:, :, 1], cmap='grey')
plt.show()
```



```
In [43]: plt.imshow(horse_red[:, :, 2], cmap='grey')
```

```
Out[43]: <matplotlib.image.AxesImage at 0x1ebb54caff0>
```

```
In [44]: plt.show()
```



```
In [45]: type(horse_red)
```

```
Out[45]: numpy.ndarray
```

```
In [46]: horse_red.shape
```

```
Out[46]: (2334, 3502, 3)
```

```
In [47]: plt.imshow(horse_red)
```

```
Out[47]: <matplotlib.image.AxesImage at 0x1ebb54c8290>
```

```
In [48]: plt.show()
```



```
In [52]: plt.imshow(horse_red[1,1,2],cmap='grey')
```

```

-----
TypeError                                         Traceback (most recent call last)
Cell In[52], line 1
----> 1 plt.imshow(horse_red[1,1,2],cmap='grey')

File ~\anaconda3\Lib\site-packages\matplotlib\pyplot.py:3562, in imshow(X, cmap,
norm, aspect, interpolation, alpha, vmin, vmax, origin, extent, interpolation_st
age, filternorm, filterrad, resample, url, data, **kwargs)
    3541 @_copy_docstring_and_deprecators(Axes.imshow)
    3542 def imshow(
    3543     X: ArrayLike | PIL.Image.Image,
    (...),
    3560     **kwargs,
    3561 ) -> AxesImage:
-> 3562     __ret = gca().imshow(
    3563         X,
    3564         cmap=cmap,
    3565         norm=norm,
    3566         aspect=aspect,
    3567         interpolation=interpolation,
    3568         alpha=alpha,
    3569         vmin=vmin,
    3570         vmax=vmax,
    3571         origin=origin,
    3572         extent=extent,
    3573         interpolation_stage=interpolation_stage,
    3574         filternorm=filternorm,
    3575         filterrad=filterrad,
    3576         resample=resample,
    3577         url=url,
    3578         **({"data": data} if data is not None else {}),
    3579         **kwargs,
    3580     )
    3581     sci(__ret)
    3582     return __ret

File ~\anaconda3\Lib\site-packages\matplotlib\__init__.py:1473, in _preprocess_da
ta.<locals>.inner(ax, data, *args, **kwargs)
    1470 @functools.wraps(func)
    1471 def inner(ax, *args, data=None, **kwargs):
    1472     if data is None:
-> 1473         return func(
    1474             ax,
    1475             *map(sanitize_sequence, args),
    1476             **{k: sanitize_sequence(v) for k, v in kwargs.items()})
    1477     bound = new_sig.bind(ax, *args, **kwargs)
    1478     auto_label = (bound.arguments.get(label_namer)
    1479                   or bound.kwargs.get(label_namer))

File ~\anaconda3\Lib\site-packages\matplotlib\axes\_axes.py:5895, in Axes.imshow
(self, X, cmap, norm, aspect, interpolation, alpha, vmin, vmax, origin, extent, i
nterpolation_stage, filternorm, filterrad, resample, url, **kwargs)
    5892 if aspect is not None:
    5893     self.set_aspect(aspect)
-> 5895 im.set_data(X)
    5896 im.set_alpha(alpha)
    5897 if im.get_clip_path() is None:
    5898     # image does not already have clipping set, clip to Axes patch

File ~\anaconda3\Lib\site-packages\matplotlib\image.py:729, in _ImageBase.set_dat

```

```
a(self, A)
    727 if isinstance(A, PIL.Image.Image):
    728     A = pil_to_array(A) # Needed e.g. to apply png palette.
--> 729 self._A = self._normalize_image_array(A)
    730 self._imcache = None
    731 self.stale = True

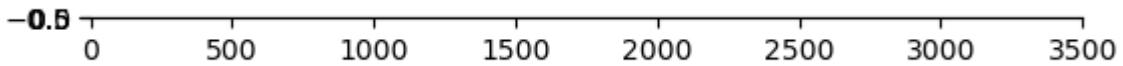
File ~\anaconda3\Lib\site-packages\matplotlib\image.py:697, in _ImageBase._normalize_image_array(A)
    695     A = A.squeeze(-1) # If just (M, N, 1), assume scalar and apply color
map.
    696 if not (A.ndim == 2 or A.ndim == 3 and A.shape[-1] in [3, 4]):
--> 697     raise TypeError(f"Invalid shape {A.shape} for image data")
    698 if A.ndim == 3:
    699     # If the input data has values outside the valid range (after
    700     # normalisation), we issue a warning and then clip X to the bounds
    701     # - otherwise casting wraps extreme values, hiding outliers and
    702     # making reliable interpretation impossible.
    703     high = 255 if np.issubdtype(A.dtype, np.integer) else 1

TypeError: Invalid shape () for image data
```

In [53]: `plt.imshow(horse_red[::1:2], cmap='grey')`

Out[53]: <matplotlib.image.AxesImage at 0x1ebab3e9b20>

In [54]: `plt.show()`



In [55]: `plt.imshow(horse_red[1:,:2], cmap='grey')`

Out[55]: <matplotlib.image.AxesImage at 0x1ebb54c8d10>

In [56]: `plt.show()`



```
In [57]: plt.imshow(horse_red[1:,:,:],cmap='grey')
```

```
Out[57]: <matplotlib.image.AxesImage at 0x1ebba9dd790>
```

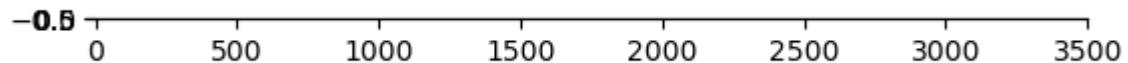
```
In [58]: plt.show()
```



```
In [59]: plt.imshow(horse_red[:1,:,:],cmap='grey')
```

```
Out[59]: <matplotlib.image.AxesImage at 0x1ebad505fa0>
```

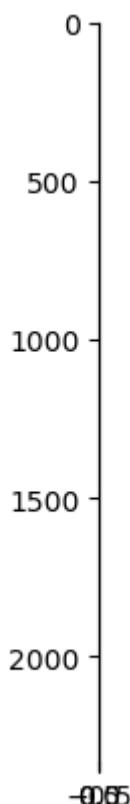
```
In [60]: plt.show()
```



```
In [61]: plt.imshow(horse_red[:, :1], cmap='grey')
```

```
Out[61]: <matplotlib.image.AxesImage at 0x1ebbb5506e0>
```

```
In [62]: plt.show()
```



```
In [63]: plt.imshow(horse_red[1:1:1], cmap='grey')
```

```

-----
ValueError                                     Traceback (most recent call last)
Cell In[63], line 1
----> 1 plt.imshow(horse_red[1:1:1],cmap='grey')

File ~\anaconda3\Lib\site-packages\matplotlib\pyplot.py:3562, in imshow(X, cmap,
norm, aspect, interpolation, alpha, vmin, vmax, origin, extent, interpolation_st
age, filternorm, filterrad, resample, url, data, **kwargs)
    3541 @_copy_docstring_and_deprecators(Axes.imshow)
    3542 def imshow(
    3543     X: ArrayLike | PIL.Image.Image,
    (...),
    3560     **kwargs,
    3561 ) -> AxesImage:
-> 3562     __ret = gca().imshow(
    3563         X,
    3564         cmap=cmap,
    3565         norm=norm,
    3566         aspect=aspect,
    3567         interpolation=interpolation,
    3568         alpha=alpha,
    3569         vmin=vmin,
    3570         vmax=vmax,
    3571         origin=origin,
    3572         extent=extent,
    3573         interpolation_stage=interpolation_stage,
    3574         filternorm=filternorm,
    3575         filterrad=filterrad,
    3576         resample=resample,
    3577         url=url,
    3578         **({"data": data} if data is not None else {}),
    3579         **kwargs,
    3580     )
    3581     sci(__ret)
    3582     return __ret

File ~\anaconda3\Lib\site-packages\matplotlib\__init__.py:1473, in _preprocess_da
ta.<locals>.inner(ax, data, *args, **kwargs)
    1470 @functools.wraps(func)
    1471 def inner(ax, *args, data=None, **kwargs):
    1472     if data is None:
-> 1473         return func(
    1474             ax,
    1475             *map(sanitize_sequence, args),
    1476             **{k: sanitize_sequence(v) for k, v in kwargs.items()})
    1477     bound = new_sig.bind(ax, *args, **kwargs)
    1478     auto_label = (bound.arguments.get(label_namer)
    1479                   or bound.kwargs.get(label_namer))

File ~\anaconda3\Lib\site-packages\matplotlib\axes\_axes.py:5895, in Axes.imshow
(self, X, cmap, norm, aspect, interpolation, alpha, vmin, vmax, origin, extent, i
nterpolation_stage, filternorm, filterrad, resample, url, **kwargs)
    5892 if aspect is not None:
    5893     self.set_aspect(aspect)
-> 5895 im.set_data(X)
    5896 im.set_alpha(alpha)
    5897 if im.get_clip_path() is None:
    5898     # image does not already have clipping set, clip to Axes patch

File ~\anaconda3\Lib\site-packages\matplotlib\image.py:729, in _ImageBase.set_dat

```

```
a(self, A)
    727 if isinstance(A, PIL.Image.Image):
    728     A = pil_to_array(A) # Needed e.g. to apply png palette.
--> 729 self._A = self._normalize_image_array(A)
    730 self._imcache = None
    731 self.stale = True

File ~\anaconda3\Lib\site-packages\matplotlib\image.py:704, in _ImageBase._normalize_image_array(A)
    698 if A.ndim == 3:
    699     # If the input data has values outside the valid range (after
    700     # normalisation), we issue a warning and then clip X to the bounds
    701     # - otherwise casting wraps extreme values, hiding outliers and
    702     # making reliable interpretation impossible.
    703     high = 255 if np.issubdtype(A.dtype, np.integer) else 1
--> 704     if A.min() < 0 or high < A.max():
    705         _log.warning(
    706             'Clipping input data to the valid range for imshow with '
    707             "'RGB data ([0..1] for floats or [0..255] for integers). "
    708             'Got range [%s..%s].',
    709             A.min(), A.max()
    710         )
    711     A = np.clip(A, 0, high)

File ~\anaconda3\Lib\site-packages\numpy\ma\core.py:5833, in MaskedArray.min(self, axis, out, fill_value, keepdims)
    5831 # No explicit output
    5832 if out is None:
--> 5833     result = self.filled(fill_value).min(
    5834         axis=axis, out=out, **kwargs).view(type(self))
    5835     if result.ndim:
    5836         # Set the mask
    5837         result.__setmask__(newmask)

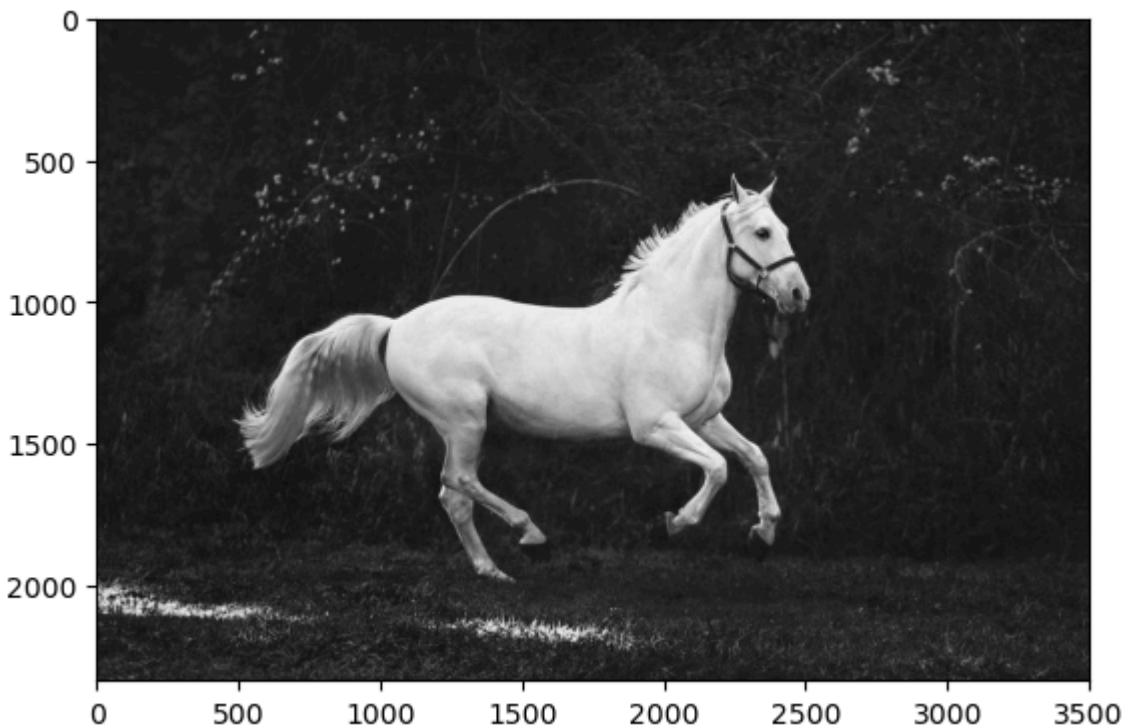
File ~\anaconda3\Lib\site-packages\numpy\core\_methods.py:45, in _amin(a, axis, out, keepdims, initial, where)
    43 def _amin(a, axis=None, out=None, keepdims=False,
    44             initial=_NoValue, where=True):
---> 45     return umr_minimum(a, axis, None, out, keepdims, initial, where)

ValueError: zero-size array to reduction operation minimum which has no identity
```

In [66]: plt.imshow(horse\_red[:, :, 2], cmap='grey')

Out[66]: <matplotlib.image.AxesImage at 0x1ebbe493b90>

In [67]: plt.show()



```
In [70]: horse_red[:, :, 0]
```

```
Out[70]: array([[15, 15, 15, ..., 23, 19, 14],  
                 [15, 15, 15, ..., 24, 22, 20],  
                 [15, 15, 15, ..., 26, 25, 24],  
                 ...,  
                 [49, 40, 35, ..., 14, 13, 11],  
                 [45, 38, 33, ..., 11, 12, 16],  
                 [33, 33, 33, ..., 12, 16, 22]], dtype=uint8)
```

```
In [71]: plt.show()
```

```
In [72]: horse_red[:, :, 1]
```

```
Out[72]: array([[17, 17, 17, ..., 38, 34, 30],  
                 [17, 17, 17, ..., 39, 37, 36],  
                 [17, 17, 17, ..., 41, 40, 40],  
                 ...,  
                 [50, 41, 34, ..., 30, 25, 23],  
                 [50, 43, 35, ..., 25, 24, 26],  
                 [40, 40, 38, ..., 26, 26, 32]], dtype=uint8)
```

```
In [73]: horse_red[:, :, 2]
```

```
Out[73]: array([[29, 29, 29, ..., 35, 31, 27],  
                 [29, 29, 29, ..., 36, 34, 33],  
                 [29, 29, 29, ..., 38, 37, 37],  
                 ...,  
                 [44, 33, 29, ..., 29, 25, 23],  
                 [44, 36, 30, ..., 25, 24, 27],  
                 [33, 33, 32, ..., 26, 27, 33]], dtype=uint8)
```

```
In [74]: plt.imshow(horse_red[:, :, 3], cmap='grey')
```

```
IndexError
```

```
Cell In[74], line 1
```

```
----> 1 plt.imshow(horse_red[:, :, 3], cmap='grey')
```

```
Traceback (most recent call last)
```

```
IndexError: index 3 is out of bounds for axis 2 with size 3
```

```
In [75]: horse_red[:, :, 1] = 0
```

```
In [76]: horse_red[:, :, 1]
```

```
Out[76]: array([[0, 0, 0, ..., 0, 0, 0],
 [0, 0, 0, ..., 0, 0, 0],
 [0, 0, 0, ..., 0, 0, 0],
 ...,
 [0, 0, 0, ..., 0, 0, 0],
 [0, 0, 0, ..., 0, 0, 0],
 [0, 0, 0, ..., 0, 0, 0]], dtype=uint8)
```

```
In [77]: plt.imshow(horse_red)
```

```
Out[77]: <matplotlib.image.AxesImage at 0x1ebbe5099a0>
```

```
In [78]: plt.show()
```



```
In [79]: horse_red[:, :, 2]
```

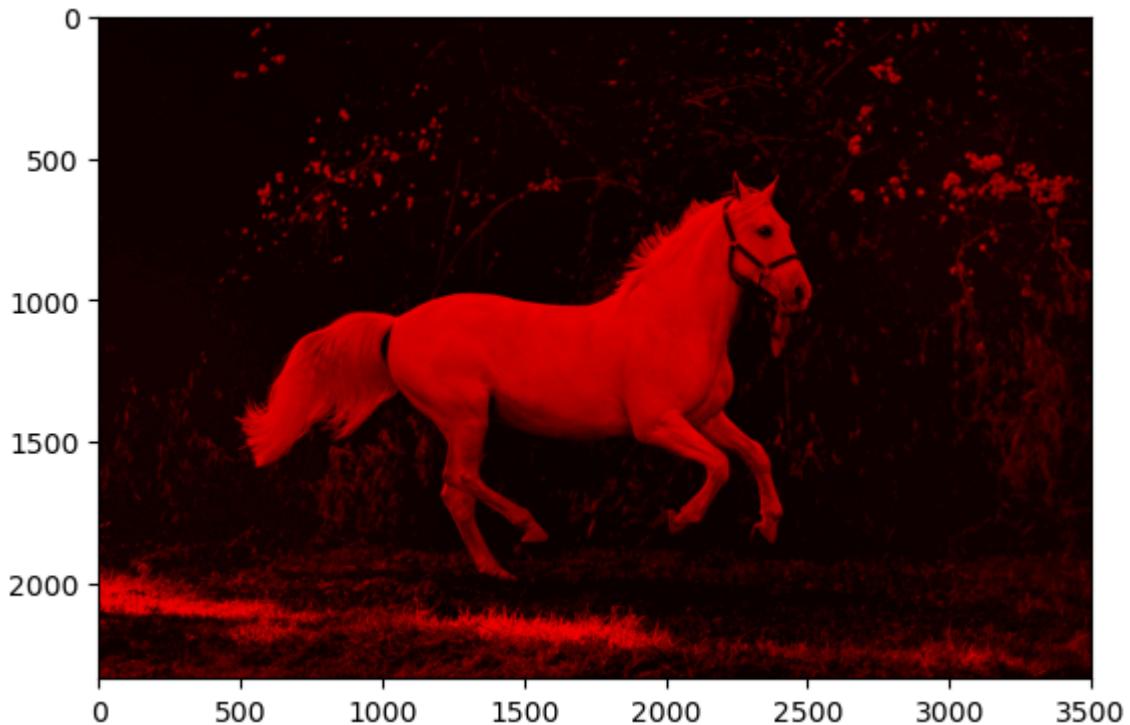
```
Out[79]: array([[29, 29, 29, ..., 35, 31, 27],
 [29, 29, 29, ..., 36, 34, 33],
 [29, 29, 29, ..., 38, 37, 37],
 ...,
 [44, 33, 29, ..., 29, 25, 23],
 [44, 36, 30, ..., 25, 24, 27],
 [33, 33, 32, ..., 26, 27, 33]], dtype=uint8)
```

```
In [80]: horse_red[:, :, 2]=0
```

```
In [81]: plt.imshow(horse_red)
```

```
Out[81]: <matplotlib.image.AxesImage at 0x1ebbeeaa5fa0>
```

```
In [82]: plt.show()
```



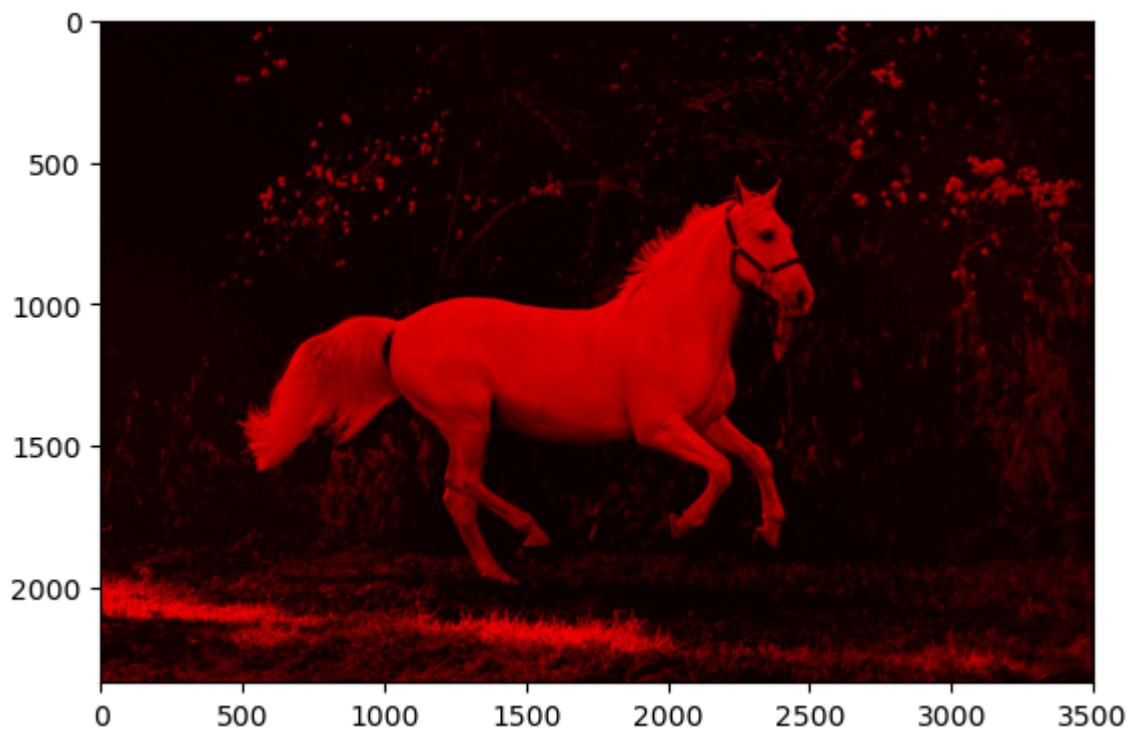
```
In [83]: horse_red[:, :, 2]
```

```
Out[83]: array([[0, 0, 0, ..., 0, 0, 0],  
                 [0, 0, 0, ..., 0, 0, 0],  
                 [0, 0, 0, ..., 0, 0, 0],  
                 ...,  
                 [0, 0, 0, ..., 0, 0, 0],  
                 [0, 0, 0, ..., 0, 0, 0],  
                 [0, 0, 0, ..., 0, 0, 0]], dtype=uint8)
```

```
In [84]: plt.imshow(horse_red)
```

```
Out[84]: <matplotlib.image.AxesImage at 0x1ebbee84e00>
```

```
In [85]: plt.show()
```



```
In [86]: horse_arr
```

```
Out[86]: array([[[15, 17, 29],  
                 [15, 17, 29],  
                 [15, 17, 29],  
                 ...,  
                 [23, 38, 35],  
                 [19, 34, 31],  
                 [14, 30, 27]],  
  
                [[15, 17, 29],  
                 [15, 17, 29],  
                 [15, 17, 29],  
                 ...,  
                 [24, 39, 36],  
                 [22, 37, 34],  
                 [20, 36, 33]],  
  
                [[15, 17, 29],  
                 [15, 17, 29],  
                 [15, 17, 29],  
                 ...,  
                 [26, 41, 38],  
                 [25, 40, 37],  
                 [24, 40, 37]],  
  
                ...,  
  
                [[49, 50, 44],  
                 [40, 41, 33],  
                 [35, 34, 29],  
                 ...,  
                 [14, 30, 29],  
                 [13, 25, 25],  
                 [11, 23, 23]],  
  
                [[45, 50, 44],  
                 [38, 43, 36],  
                 [33, 35, 30],  
                 ...,  
                 [11, 25, 25],  
                 [12, 24, 24],  
                 [16, 26, 27]],  
  
                [[33, 40, 33],  
                 [33, 40, 33],  
                 [33, 38, 32],  
                 ...,  
                 [12, 26, 26],  
                 [16, 26, 27],  
                 [22, 32, 33]]], dtype=uint8)
```

```
In [87]: horse_red
```

```
Out[87]: array([[[15,  0,  0],
   [15,  0,  0],
   [15,  0,  0],
   ...,
   [23,  0,  0],
   [19,  0,  0],
   [14,  0,  0]],

   [[15,  0,  0],
   [15,  0,  0],
   [15,  0,  0],
   ...,
   [24,  0,  0],
   [22,  0,  0],
   [20,  0,  0]],

   [[15,  0,  0],
   [15,  0,  0],
   [15,  0,  0],
   ...,
   [26,  0,  0],
   [25,  0,  0],
   [24,  0,  0]],

   ...,

   [[49,  0,  0],
   [40,  0,  0],
   [35,  0,  0],
   ...,
   [14,  0,  0],
   [13,  0,  0],
   [11,  0,  0]],

   [[45,  0,  0],
   [38,  0,  0],
   [33,  0,  0],
   ...,
   [11,  0,  0],
   [12,  0,  0],
   [16,  0,  0]],

   [[33,  0,  0],
   [33,  0,  0],
   [33,  0,  0],
   ...,
   [12,  0,  0],
   [16,  0,  0],
   [22,  0,  0]]], dtype=uint8)
```

```
In [88]: horse_img
```

Out[88]:



In [89]: arr1=np.asarray(horse\_img)

In [90]: type(arr1)

Out[90]: numpy.ndarray

In [91]: arr1.shape

Out[91]: (2334, 3502, 3)

In [92]: plt.imshow(arr1)

Out[92]: &lt;matplotlib.image.AxesImage at 0x1ebc36fd010&gt;

In [93]: plt.show()



```
In [94]: horse_img1=arr1.copy()
```

```
In [95]: horse_img1
```

```
Out[95]: array([[[15, 17, 29],  
                 [15, 17, 29],  
                 [15, 17, 29],  
                 ...,  
                 [23, 38, 35],  
                 [19, 34, 31],  
                 [14, 30, 27]],  
  
                [[15, 17, 29],  
                 [15, 17, 29],  
                 [15, 17, 29],  
                 ...,  
                 [24, 39, 36],  
                 [22, 37, 34],  
                 [20, 36, 33]],  
  
                [[15, 17, 29],  
                 [15, 17, 29],  
                 [15, 17, 29],  
                 ...,  
                 [26, 41, 38],  
                 [25, 40, 37],  
                 [24, 40, 37]],  
  
                ...,  
  
                [[49, 50, 44],  
                 [40, 41, 33],  
                 [35, 34, 29],  
                 ...,  
                 [14, 30, 29],  
                 [13, 25, 25],  
                 [11, 23, 23]],  
  
                [[45, 50, 44],  
                 [38, 43, 36],  
                 [33, 35, 30],  
                 ...,  
                 [11, 25, 25],  
                 [12, 24, 24],  
                 [16, 26, 27]],  
  
                [[33, 40, 33],  
                 [33, 40, 33],  
                 [33, 38, 32],  
                 ...,  
                 [12, 26, 26],  
                 [16, 26, 27],  
                 [22, 32, 33]]], dtype=uint8)
```

```
In [96]: plt.show()
```

```
In [97]: horse_img1[:, :, 0] = 0
```

```
In [98]: horse_img1
```

```
Out[98]: array([[[ 0, 17, 29],
   [ 0, 17, 29],
   [ 0, 17, 29],
   ...,
   [ 0, 38, 35],
   [ 0, 34, 31],
   [ 0, 30, 27]],

   [[ 0, 17, 29],
   [ 0, 17, 29],
   [ 0, 17, 29],
   ...,
   [ 0, 39, 36],
   [ 0, 37, 34],
   [ 0, 36, 33]],

   [[ 0, 17, 29],
   [ 0, 17, 29],
   [ 0, 17, 29],
   ...,
   [ 0, 41, 38],
   [ 0, 40, 37],
   [ 0, 40, 37]],

   ...,

   [[ 0, 50, 44],
   [ 0, 41, 33],
   [ 0, 34, 29],
   ...,
   [ 0, 30, 29],
   [ 0, 25, 25],
   [ 0, 23, 23]],

   [[ 0, 50, 44],
   [ 0, 43, 36],
   [ 0, 35, 30],
   ...,
   [ 0, 25, 25],
   [ 0, 24, 24],
   [ 0, 26, 27]],

   [[ 0, 40, 33],
   [ 0, 40, 33],
   [ 0, 38, 32],
   ...,
   [ 0, 26, 26],
   [ 0, 26, 27],
   [ 0, 32, 33]]], dtype=uint8)
```

```
In [99]: plt.imshow(horse_img1)
```

```
Out[99]: <matplotlib.image.AxesImage at 0x1ebad545880>
```

```
In [100...]: plt.show()
```



```
In [101]: horse_img1[:, :, 1]
```

```
Out[101]: array([[17, 17, 17, ..., 38, 34, 30],  
                   [17, 17, 17, ..., 39, 37, 36],  
                   [17, 17, 17, ..., 41, 40, 40],  
                   ...,  
                   [50, 41, 34, ..., 30, 25, 23],  
                   [50, 43, 35, ..., 25, 24, 26],  
                   [40, 40, 38, ..., 26, 26, 32]], dtype=uint8)
```

```
In [102]: plt.imshow(horse_img1)  
plt.show()
```



```
In [103... horse_img1[:, :, 1]
```

```
Out[103... array([[17, 17, 17, ..., 38, 34, 30],  
[17, 17, 17, ..., 39, 37, 36],  
[17, 17, 17, ..., 41, 40, 40],  
...,  
[50, 41, 34, ..., 30, 25, 23],  
[50, 43, 35, ..., 25, 24, 26],  
[40, 40, 38, ..., 26, 26, 32]], dtype=uint8)
```

```
In [104... horse_img1[:, :, 1]=0
```

```
In [105... plt.imshow(horse_img1)
```

```
Out[105... <matplotlib.image.AxesImage at 0x1ebc3621b50>
```

```
In [106... plt.show()
```



```
In [107... plt.imshow(horse_img1)
```

```
Out[107... <matplotlib.image.AxesImage at 0x1ebc35e44a0>
```

```
In [108... plt.show()
```



In [ ]: