

# Python程序设计

案例：图像处理  
NumPy和PIL

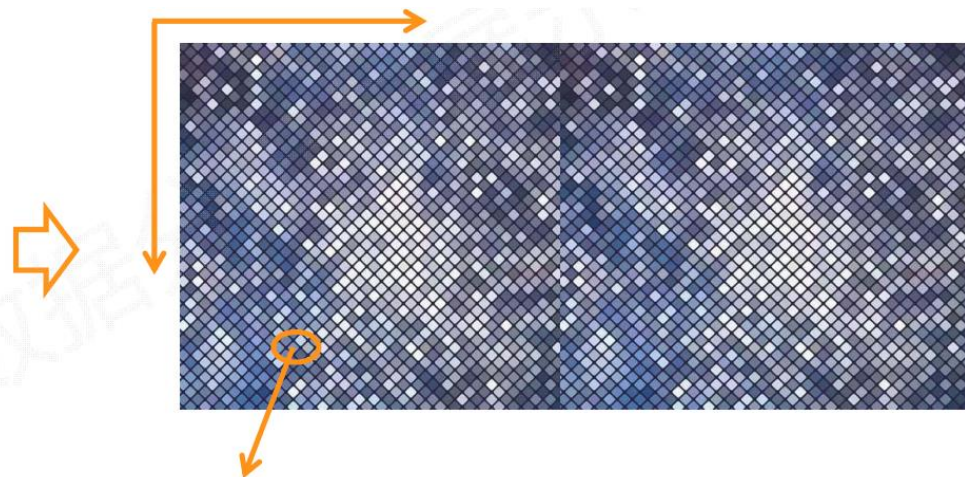


张 华

# 图像处理

## 图像的数组表示

- 图像是一个由像素组成的二维矩阵，每个元素是一个RGB值。



RGB值: (R, G, B)

# 图像处理

## 图像的数组表示

### PIL (Python Image Library)

- PIL库是一个具有强大图像处理能力的第三方库。
- Anaconda已集成该库。

**from PIL import Image**

- Image是PIL库中代表一个图像的类（对象）。

# 图像处理

## 图像的数组表示

- ✿ 读入图像并创建`ndarray`对象，图像就是一个三维数组，维度分别是高度、宽度和像素RGB值。

```
>>> from PIL import Image
>>> import numpy as np
>>> im = np.array(Image.open('beijing.jpg'))
>>> print(im.shape, im.dtype)
(669, 1012, 3) uint8
>>> im
array([[[ 79, 129, 226],
        [ 74, 124, 235],
        [ 72, 125, 227],
        ...,
        [ 63,  94, 125],
        [ 66,  95, 129],
        [ 69,  96, 126]]], dtype=uint8)
```

# 图像处理

## 图像的数组表示

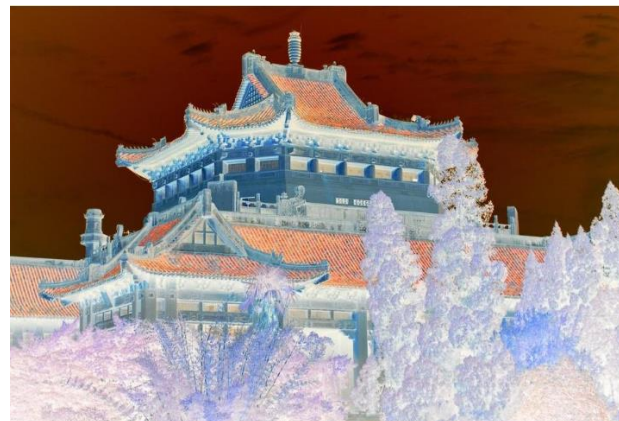
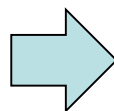
- ✱ 图像一般使用**RGB**色彩模式，即每个像素点的颜色由红(R)、绿(G)、蓝(B)组成。
- ✱ **RGB**三个颜色通道的变化和叠加得到各种颜色，其中
  - **R** 红色，取值范围，**0-255**
  - **G** 绿色，取值范围，**0-255**
  - **B** 蓝色，取值范围，**0-255**

# 图像处理

## 图像的变换

✿ 读入图像后，获得像素RGB值，修改后保存为新的文件。

```
>>> from PIL import Image  
>>> import numpy as np  
>>> whu = np.array(Image.open('whu.jpg'))  
>>> whutrans = [255, 255, 255] - whu  
>>> im = Image.fromarray(whutrans.astype('uint8'))  
>>> im.save('whutrans.jpg')
```



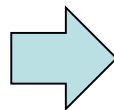


# 图像处理

## 图像的变换

✿ 读入图像后，获得像素RGB值，修改后保存为新的文件。

```
>>> from PIL import Image  
>>> import numpy as np  
>>> whu = np.array(Image.open('whu.jpg').convert('L'))  
>>> im = Image.fromarray(whu.astype('uint8'))  
>>> im.save('whutrans2.jpg')
```



# 图像处理

## 图像的变换

✿ 读入图像后，获得像素RGB值，修改后保存为新的文件。

```
>>> from PIL import Image  
>>> import numpy as np  
>>> whu = np.array(Image.open('whu.jpg').convert('L'))  
>>> whutrans = 255 - whu  
>>> im = Image.fromarray(whutrans.astype('uint8'))  
>>> im.save('whutrans3.jpg')
```

