TASK1.2：图像就是矩阵（或者矢量，Tensor）

TASK 目标：本次任务的目的就是帮助你熟悉图像的各种代数运算：（1）找一张个人大头像，并通过各种奇葩数学变幻，发现一个新的自我！

```
from PIL import Image
photo = Image.open('C:/Users/hp/OneDrive/图片/刘嘉玲/一张严肃的照片.jpg')
photo
```

```
print(photo.size)
photo=photo.resize([128,128])
print(photo.size)
photo
```

```
(1080, 1440)
(128, 128)
```

```
import numpy as np
Im=np.array(photo)
print(Im.shape)
Im[:,:,0]
```

```
(128, 128, 3)

array([[213, 213, 213, ..., 239, 239, 239],
       [205, 206, 208, ..., 239, 239, 239],
       [147, 152, 156, ..., 240, 240, 240],
       ...,
       [247, 247, 248, ...,  13,  13,  13],
       [247, 247, 247, ...,  13,  13,  11],
       [247, 247, 247, ...,  13,  13,  11]], dtype=uint8)
```

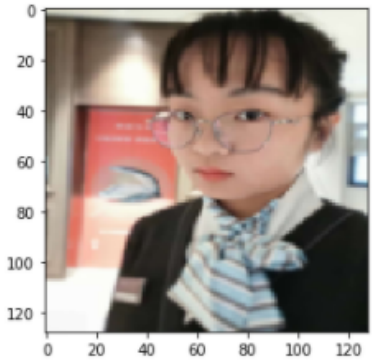
```
Im=Im/255
print(Im[:,:,0])
```

```
[[0.83529412 0.83529412 0.83529412 ... 0.9372549  0.9372549  0.9372549 ]
 [0.80392157 0.80784314 0.81568627 ... 0.9372549  0.9372549  0.9372549 ]
 [0.57647059 0.59607843 0.61176471 ... 0.94117647 0.94117647 0.94117647]
 ...
 [0.96862745 0.96862745 0.97254902 ... 0.05098039 0.05098039 0.05098039]
 [0.96862745 0.96862745 0.96862745 ... 0.05098039 0.05098039 0.04313725]
 [0.96862745 0.96862745 0.96862745 ... 0.05098039 0.05098039 0.04313725]]
```

```
from matplotlib import pyplot as plt
plt.imshow(Im)
```

```
Bad key "text.kerning_factor" on line 4 in
E:\Anaconda3\lib\site-packages\matplotlib\mpl-data\stylelib\_classic_test_patch.mplstyle.
You probably need to get an updated matplotlibrc file from
https://github.com/matplotlib/matplotlib/blob/v3.1.3/matplotlibrc.template
or from the matplotlib source distribution
```

```
<matplotlib.image.AxesImage at 0x1e341574dc8>
```

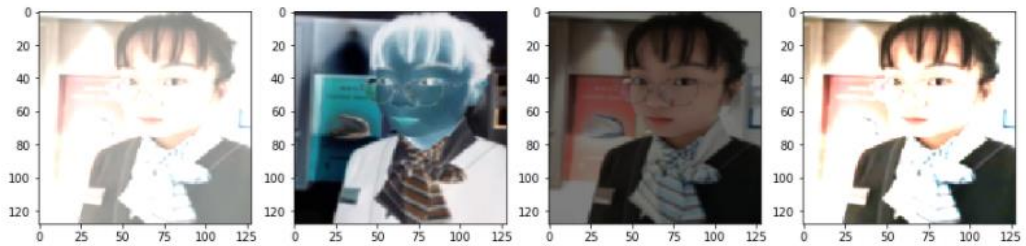



```
Im1=Im+0.5
Im2=1-Im
Im3=0.5*Im
Im4=Im/0.5
plt.figure()
fig,ax=plt.subplots(1,4)
fig.set_figwidth(15)
ax[0].imshow(Im1)
ax[1].imshow(Im2)
ax[2].imshow(Im3)
ax[3].imshow(Im4)
```

```
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
```

```
<matplotlib.image.AxesImage at 0x1e343223e48>
```

```
<Figure size 432x288 with 0 Axes>
```



第 1 张图像将 Im 图像加上 0.5 得到 Im1。

第 2 张图像用 1 减去 Im 图像得到 Im2。

第 3 张图像用 0.5 乘以 Im 图像得到 Im3。

第 4 张图像将 Im 图像除以 0.5 得到 Im4。

将这 4 张图像（Im1，Im2，Im3 和 Im4）展示在一张画板上，.figure 函数用于初始化画板，.subplots 函数用于切分画板，这里切分成了 1 行 4 列。切分之后会有两个输出，其中 fig 可以控制画板的长和宽。

第2讲：图像数据与运算基础

图像数据读入与展示 ¶

```
In [2]: from PIL import Image
        photo=Image.open('./photos/xiongda.jpg')
        photo
```

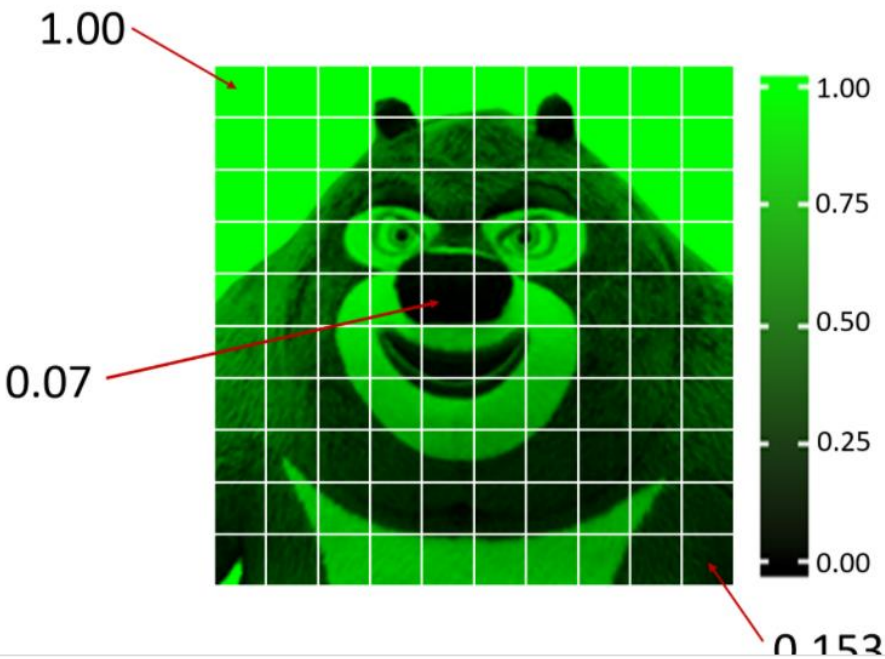
Out[2]:

```
In [3]: Image.open('./photos/xiongda3color.png')
```

Out[3]:

In [4]:
```
Image.open('./photos/xiongdaresolution.png')
```

Out[4]:



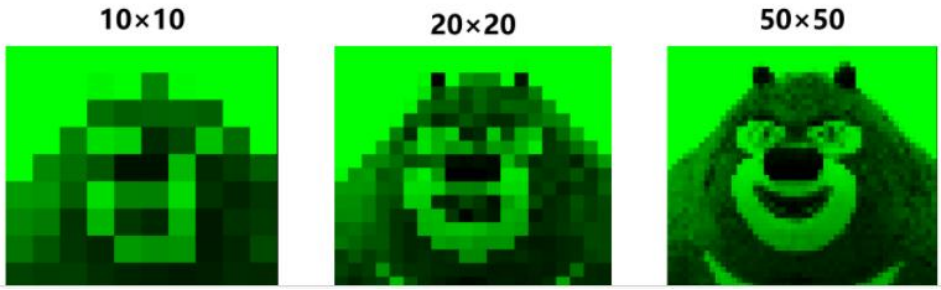
In [5]:
```
Image.open('./photos/xiongdapixelmatrix.png')
```

Out[5]:

1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
1.000	1.000	1.000	0.956	0.997	0.520	0.982	0.999	1.000	1.000
1.000	1.000	1.000	0.440	0.364	0.384	0.398	0.368	1.000	1.000
1.000	1.000	0.501	0.871	0.821	0.166	0.277	0.755	0.419	0.994
1.000	0.532	0.431	0.337	0.070	0.074	0.724	0.162	0.228	0.365
0.600	0.581	0.369	0.864	0.511	0.502	0.852	0.198	0.139	0.205
0.527	0.477	0.201	0.709	0.221	0.158	0.678	0.099	0.194	0.228
0.449	0.338	0.204	0.153	0.604	0.562	0.578	0.101	0.148	0.277
0.261	0.236	0.238	0.182	0.142	0.126	0.106	0.118	0.160	0.153
0.153	0.137	0.671	0.709	0.271	0.177	0.212	0.626	0.315	0.153

In [6]:
```
Image.open('./photos/xiongdamany.png')
```

Out[6]:



In [7]:
```
print(photo.size)
photo=photo.resize([128,128])
print(photo.size)
photo
```

```
(301, 296)
(128, 128)
```

Out[7]:

In [8]:
```
import numpy as np
Im=np.array(photo)
print(Im.shape)
Im[:,:,0]
```

```
(128, 128, 3)
```

Out[8]:
```
array([[255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       [255, 255, 255, ..., 255, 255, 255],
       ...,
       [ 34,  32,  58, ...,  54,  50, 129],
       [ 27,  48, 180, ...,  53,  46, 127],
       [ 37, 168, 255, ...,  54,  43, 122]], dtype=uint8)
```

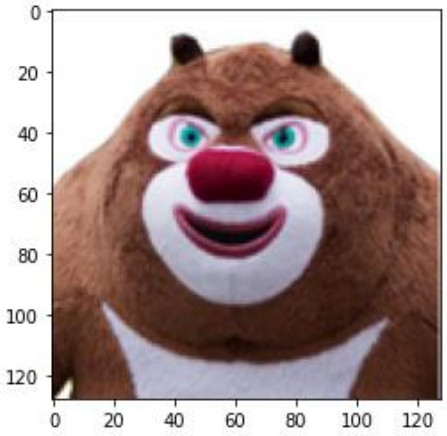
In [9]:
```
Im=Im/255
print(Im[:,:,0])
```

```
[[1.         1.         1.         ... 1.         1.         1.        ]
 [1.         1.         1.         ... 1.         1.         1.        ]
 [1.         1.         1.         ... 1.         1.         1.        ]
 ...
 [0.13333333 0.1254902  0.22745098 ... 0.21176471 0.19607843 0.50588235]
 [0.10588235 0.18823529 0.70588235 ... 0.20784314 0.18039216 0.49803922]
 [0.14509804 0.65882353 1.         ... 0.21176471 0.16862745 0.47843137]]
```

图像展示

In [10]:
```
from matplotlib import pyplot as plt
plt.imshow(Im)
```

Out[10]:
```
<matplotlib.image.AxesImage at 0x7f12be1d8828>
```



单色图片

[11]:
```
Im0=1.0*Im;Im0[:,:,1]=0;Im0[:,:,2]=0
Im1=1.0*Im;Im1[:,:,0]=0;Im1[:,:,2]=0
Im2=1.0*Im;Im2[:,:,0]=0;Im2[:,:,1]=0
plt.figure()
fig,ax=plt.subplots(1,3)
fig.set_figwidth(15)
ax[0].imshow(Im0)
ax[1].imshow(Im1)
ax[2].imshow(Im2)

Im0=1.0*Im;Im0[:,:,1]=0;Im0[:,:,2]=0
Im1=1.0*Im;Im1[:,:,0]=0;Im1[:,:,2]=0
Im2=1.0*Im;Im2[:,:,0]=0;Im2[:,:,1]=0
plt.figure()
fig,ax=plt.subplots(1,3)
fig.set_figwidth(15)

ax[0].imshow(Im0)
ax[1].imshow(Im1)
ax[2].imshow(Im2)
```

ıt[11]:
```
<matplotlib.image.AxesImage at 0x7f12bdbe1780>

<Figure size 432x288 with 0 Axes>
```

代数运算

[12]:
```
Im0=0.5*Im
Im1=0.5*Im+0.5
Im2=1-Im
plt.figure()
fig,ax=plt.subplots(1,3)
fig.set_figwidth(15)
ax[0].imshow(Im0)
ax[1].imshow(Im1)
ax[2].imshow(Im2)

Im0=0.5*Im
Im1=0.5*Im+0.5
Im2=1-Im
plt.figure()
fig,ax
```

t[12]:
```
(<Figure size 1080x288 with 3 Axes>,
 array([<matplotlib.axes._subplots.AxesSubplot object at 0x7f12bda7fda0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7f12bda969e8>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7f12bda45c18>],
       dtype=object))

<Figure size 432x288 with 0 Axes>
```

思考问题：还可以对图像做哪些有趣的计算？能否把计算结果以图像的方式展示出来？

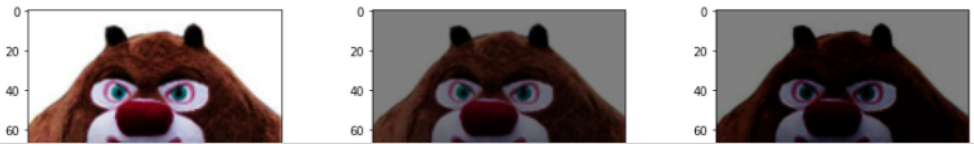
n [13]:
```
Im3=Im**2
Im4=0.5*Im**2
Im5=0.5*Im**4

plt.figure()
fig1,ax1=plt.subplots(1,3)
fig1.set_figwidth(15)
ax1[0].imshow(Im3)
ax1[1].imshow(Im4)
ax1[2].imshow(Im5)

Im3=Im**2
Im4=0.5*Im**2
Im5=0.5*Im**4
plt.figure()
fig1,ax1
```

Out[13]:
```
(<Figure size 1080x288 with 3 Axes>,
 array([<matplotlib.axes._subplots.AxesSubplot object at 0x7f12bd990208>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7f12bd93dc50>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7f12bd96ee80>],
       dtype=object))

<Figure size 432x288 with 0 Axes>
```



In [21]:
```
Im6=2*Im;Im6[:,:,2]=0;Im6[:,:,2]=0
Im7=2*Im;Im7[:,:,1]=0;Im7[:,:,2]=0
Im8=2*Im;Im8[:,:,1]=0;Im8[:,:,1]=0
plt.figure()
fig,ax=plt.subplots(1,3)
fig.set_figwidth(15)

ax[0].imshow(Im6)
ax[1].imshow(Im7)
ax[2].imshow(Im8)
```

```
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).
```

Out[21]:
```
<matplotlib.image.AxesImage at 0x7f12bd35b550>

<Figure size 432x288 with 0 Axes>
```