```
Name : Dhadge Yash Kailas
Roll No : 9019
Class : BE-IT
Course : Information Storage and Retrieval

Group:B (Assignment-03)
Problem Statement :
    Implement a program for feature extraction in 2D color images (any features
like color, texture etc. and
    to extract features from input image and plot histogram for the features.
```

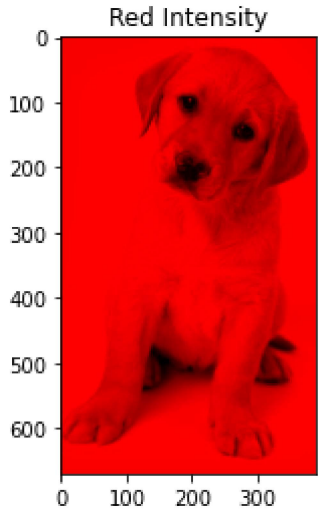
Importing an Image :-

In [1]:
```
import matplotlib.pyplot as plt
from matplotlib import image as mpimg
image = mpimg.imread("puppy_image.jpeg")
plt.imshow(image)
plt.show()
```

Red and Yellow image :-

In [2]:
```
red,yellow=image.copy(),image.copy()
red[:,:,(1,2)]=0
yellow[:,:,2]=0
```

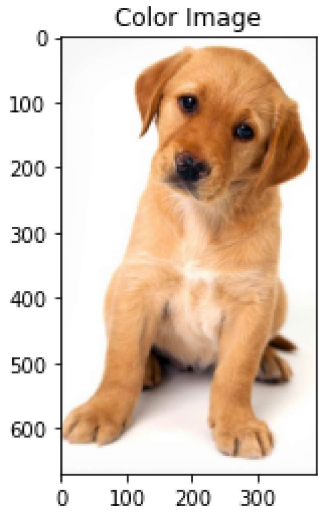
In [3]:
```
plt.title('Red Intensity')
plt.imshow(red)
plt.show()
```

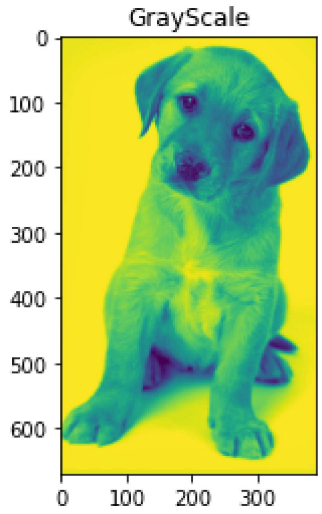


In [4]:
```
plt.title('Yellow Intensity')
plt.imshow(yellow)
plt.show()
```

Converting Images to a 2-D matrix (RGB TO Gray) :-

In [5]:
```
from skimage.color import rgb2gray
gray_image=rgb2gray(image)
plt.title('Color Image')
plt.imshow(image)
plt.show()
plt.title('GrayScale')
plt.imshow(gray_image)
plt.show()
print ("Colored image shape: ",image.shape)
print ("Gray image shape: ",gray_image.shape)
```

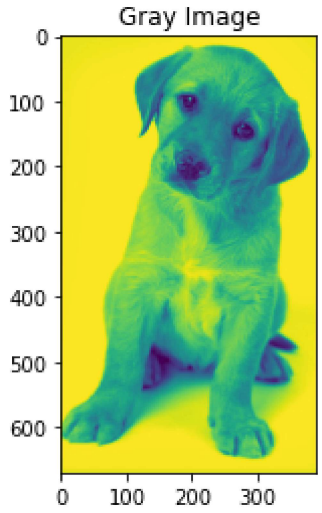


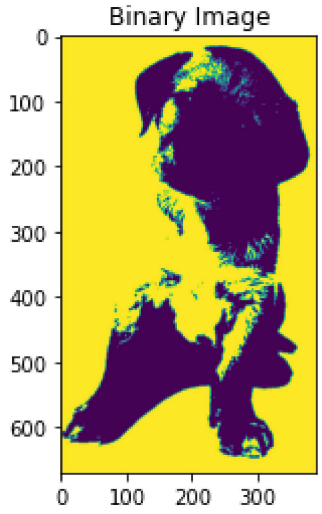



```
Colored image shape:  (670, 391, 3)
Gray image shape:  (670, 391)
```

Binarization of gray scale image:-

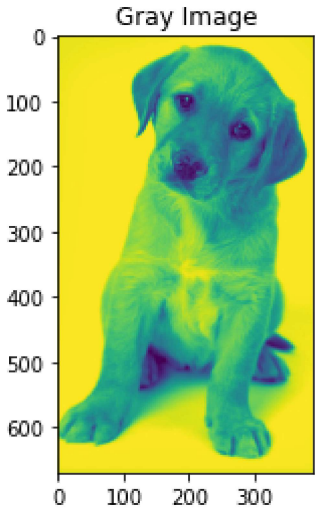
In [6]:
```
from skimage.filters import threshold_otsu

thresh = threshold_otsu(gray_image)

binary = gray_image > thresh
plt.title('Gray Image')
plt.imshow(gray_image)
plt.show()
plt.title('Binary Image')
plt.imshow(binary)
plt.show()
```

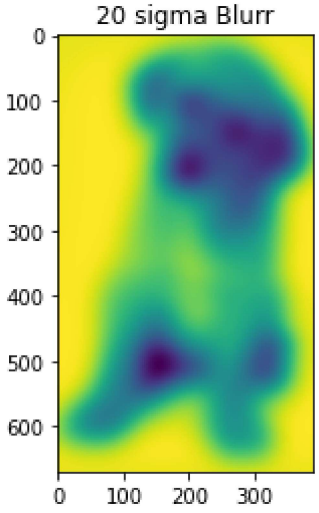




Blurring an Image:-

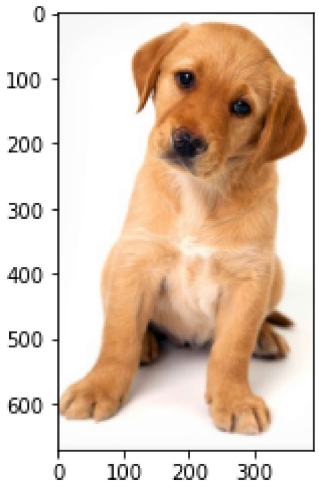
In [7]:
```
from skimage.filters import gaussian
blurred_image=gaussian(gray_image,sigma=20)
plt.title('Gray Image')
plt.imshow(gray_image)
plt.show()
plt.title('20 sigma Blurr')
plt.imshow(blurred_image)
plt.show()
```





Histogram creation using numpy array:-

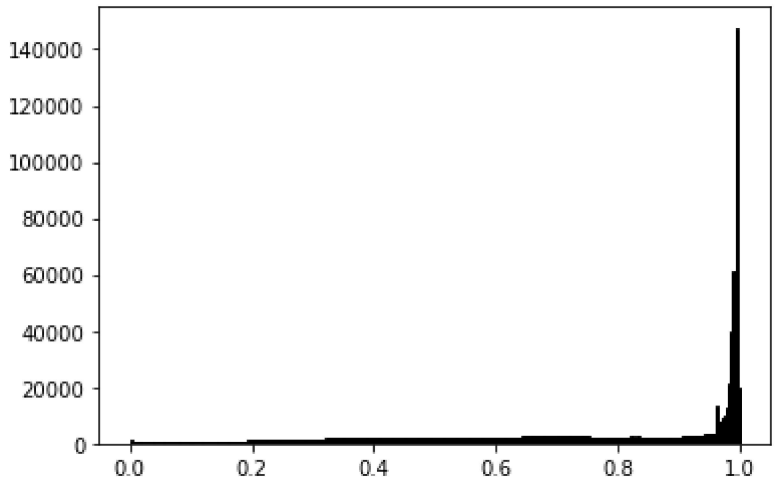
In [8]:
```
image1 = mpimg.imread("puppy_image-Copy1.png")
plt.imshow(image1)
plt.show()
```



In [9]:
```
plt.hist(image1.ravel(),bins=256,range=(0.0,1.0),fc='k',ec='k')
```

Out[9]:
```
(array([   994.,    293.,    307.,    306.,    255.,    296.,    324.,
           323.,    314.,    347.,    380.,    418.,    475.,    473.,
           455.,    497.,    524.,    576.,    639.,    610.,    672.,
           644.,    624.,    652.,    664.,    685.,    739.,    689.,
           721.,    719.,    765.,    728.,    774.,    783.,    761.,
           788.,    786.,    826.,    777.,    778.,    851.,    794.,
           812.,    852.,    866.,    871.,    860.,    855.,    881.,
           939.,    959.,    963.,    974.,   1031.,   1074.,   1102.,
          1114.,   1101.,   1196.,   1142.,   1160.,   1214.,   1186.,
          1179.,   1234.,   1198.,   1230.,   1294.,   1313.,   1328.,
          1336.,   1339.,   1414.,   1417.,   1498.,   1529.,   1431.,
          1559.,   1561.,   1634.,   1614.,   1624.,   1704.,   1747.,
          1781.,   1712.,   1759.,   1773.,   1718.,   1760.,   1740.,
          1811.,   1814.,   1754.,   1852.,   1826.,   1771.,   1785.,
          1796.,   1831.,   1829.,   1899.,   1817.,   1851.,   1842.,
          1913.,   1932.,   1978.,   2066.,   2068.,   2078.,   2210.,
          2140.,   2168.,   2264.,   2256.,   2232.,   2215.,   2305.,
          2253.,   2256.,   2261.,   2153.,   2183.,   2142.,   2174.,
          2146.,   2233.,   2212.,   2217.,   2279.,   2262.,   2174.,
          2112.,   2281.,   2232.,   2212.,   2161.,   2197.,   2165.,
          2239.,   2208.,   2278.,   2149.,   2123.,   2046.,   2215.,
          2044.,   2169.,   2048.,   2063.,   1978.,   2073.,   2035.,
          2072.,   2131.,   2042.,   2128.,   2186.,   2193.,   2082.,
          2233.,   2260.,   2323.,   2423.,   2371.,   2429.,   2401.,
          2595.,   2548.,   2620.,   2537.,   2582.,   2441.,   2405.,
          2566.,   2555.,   2540.,   2524.,   2587.,   2565.,   2571.,
          2608.,   2569.,   2622.,   2553.,   2574.,   2551.,   2410.,
          2429.,   2503.,   2421.,   2396.,   2235.,   2295.,   2203.,
          2245.,   2246.,   2212.,   2294.,   2171.,   2181.,   2038.,
          2071.,   2172.,   2222.,   2217.,   2222.,   2296.,   2312.,
          2383.,   2353.,   2470.,   2356.,   2258.,   2241.,   2270.,
          2233.,   2148.,   2224.,   2142.,   2224.,   2190.,   2274.,
          2280.,   2175.,   2308.,   2219.,   2157.,   2168.,   2254.,
          2318.,   2436.,   2814.,   2774.,   2675.,   2605.,   2888.,
          2780.,   2857.,   3023.,   3185.,   3394.,   3405.,   3093.,
          3478.,  13684.,   7831.,   9296.,   9578.,  12472.,  21112.,
         39604.,  61435., 147663.,  20125.]),
 array([0.        , 0.00390625, 0.0078125 , 0.01171875, 0.015625  ,
        0.01953125, 0.0234375 , 0.02734375, 0.03125   , 0.03515625,
        0.0390625 , 0.04296875, 0.046875  , 0.05078125, 0.0546875 ,
        0.05859375, 0.0625    , 0.06640625, 0.0703125 , 0.07421875,
        0.078125  , 0.08203125, 0.0859375 , 0.08984375, 0.09375   ,
        0.09765625, 0.1015625 , 0.10546875, 0.109375  , 0.11328125,
        0.1171875 , 0.12109375, 0.125     , 0.12890625, 0.1328125 ,
        0.13671875, 0.140625  , 0.14453125, 0.1484375 , 0.15234375,
        0.15625   , 0.16015625, 0.1640625 , 0.16796875, 0.171875  ,
        0.17578125, 0.1796875 , 0.18359375, 0.1875    , 0.19140625,
        0.1953125 , 0.19921875, 0.203125  , 0.20703125, 0.2109375 ,
        0.21484375, 0.21875   , 0.22265625, 0.2265625 , 0.23046875,
        0.234375  , 0.23828125, 0.2421875 , 0.24609375, 0.25      ,
        0.25390625, 0.2578125 , 0.26171875, 0.265625  , 0.26953125,
        0.2734375 , 0.27734375, 0.28125   , 0.28515625, 0.2890625 ,
        0.29296875, 0.296875  , 0.30078125, 0.3046875 , 0.30859375,
        0.3125    , 0.31640625, 0.3203125 , 0.32421875, 0.328125  ,
```

```
       0.33203125, 0.3359375 , 0.33984375, 0.34375   , 0.34765625,
       0.3515625 , 0.35546875, 0.359375  , 0.36328125, 0.3671875 ,
       0.37109375, 0.375     , 0.37890625, 0.3828125 , 0.38671875,
       0.390625  , 0.39453125, 0.3984375 , 0.40234375, 0.40625   ,
       0.41015625, 0.4140625 , 0.41796875, 0.421875  , 0.42578125,
       0.4296875 , 0.43359375, 0.4375    , 0.44140625, 0.4453125 ,
       0.44921875, 0.453125  , 0.45703125, 0.4609375 , 0.46484375,
       0.46875   , 0.47265625, 0.4765625 , 0.48046875, 0.484375  ,
       0.48828125, 0.4921875 , 0.49609375, 0.5       , 0.50390625,
       0.5078125 , 0.51171875, 0.515625  , 0.51953125, 0.5234375 ,
       0.52734375, 0.53125   , 0.53515625, 0.5390625 , 0.54296875,
       0.546875  , 0.55078125, 0.5546875 , 0.55859375, 0.5625    ,
       0.56640625, 0.5703125 , 0.57421875, 0.578125  , 0.58203125,
       0.5859375 , 0.58984375, 0.59375   , 0.59765625, 0.6015625 ,
       0.60546875, 0.609375  , 0.61328125, 0.6171875 , 0.62109375,
       0.625     , 0.62890625, 0.6328125 , 0.63671875, 0.640625  ,
       0.64453125, 0.6484375 , 0.65234375, 0.65625   , 0.66015625,
       0.6640625 , 0.66796875, 0.671875  , 0.67578125, 0.6796875 ,
       0.68359375, 0.6875    , 0.69140625, 0.6953125 , 0.69921875,
       0.703125  , 0.70703125, 0.7109375 , 0.71484375, 0.71875   ,
       0.72265625, 0.7265625 , 0.73046875, 0.734375  , 0.73828125,
       0.7421875 , 0.74609375, 0.75      , 0.75390625, 0.7578125 ,
       0.76171875, 0.765625  , 0.76953125, 0.7734375 , 0.77734375,
       0.78125   , 0.78515625, 0.7890625 , 0.79296875, 0.796875  ,
       0.80078125, 0.8046875 , 0.80859375, 0.8125    , 0.81640625,
       0.8203125 , 0.82421875, 0.828125  , 0.83203125, 0.8359375 ,
       0.83984375, 0.84375   , 0.84765625, 0.8515625 , 0.85546875,
       0.859375  , 0.86328125, 0.8671875 , 0.87109375, 0.875     ,
       0.87890625, 0.8828125 , 0.88671875, 0.890625  , 0.89453125,
       0.8984375 , 0.90234375, 0.90625   , 0.91015625, 0.9140625 ,
       0.91796875, 0.921875  , 0.92578125, 0.9296875 , 0.93359375,
       0.9375    , 0.94140625, 0.9453125 , 0.94921875, 0.953125  ,
       0.95703125, 0.9609375 , 0.96484375, 0.96875   , 0.97265625,
       0.9765625 , 0.98046875, 0.984375  , 0.98828125, 0.9921875 ,
       0.99609375, 1.        ], dtype=float32),
<BarContainer object of 256 artists>)
```



OpenCV in built function for Histogram:-

In [10]:
```
import cv2
histr = cv2.calcHist([image],[0],None,[256],[0,256])
plt.plot(histr)
plt.show()
```