

```

# CHALLENGE 5: Image analysis using OpenCV.
import numpy as np
import matplotlib.pyplot as plt
import cv2,sys,os
from matplotlib import cm

# 1. Load and show an image.
aScale = 0.1
WD = os.path.dirname(sys.argv[0])
image = cv2.imread(os.path.join(WD,
                                'bright-canada-day-celebrate-1697902.jpg'))
smImage = cv2.resize(image,(0,0),
                     fx=aScale,fy=aScale,interpolation=cv2.INTER_AREA)
cv2.imshow("Image",smImage)
cv2.waitKey(0)
cv2.destroyAllWindows()

# 2. Plot the negative of the image.
imNegative = 255 - smImage
cv2.imshow("Image Negative",imNegative)
cv2.waitKey(0)
cv2.destroyAllWindows()

# 3. Plot a histogram of the colors of the image.
color = ('b','g','r')
for i,col in enumerate(color):
    histr = cv2.calcHist([image],[i],None,[256],[0,256])
    plt.step(np.arange(len(histr)),histr,color = col)

plt.xlim([-1,256])
plt.show()

# E1. Remove a feature from your image.
smImageMask = np.zeros(smImage.shape[0:2],dtype='uint8')
smImageMask[240:355,200:210] = 1 #the stick
smImageMask[30:150,240:346] = 1# fireworks R
smImageMask[100:250,280:300] = 1# fireworks R

imIP = cv2.inpaint(smImage,smImageMask,5,cv2.INPAINT_NS)
cv2.imshow("Inpainting the image", imIP)
cv2.waitKey(0)
cv2.destroyAllWindows()

# E2. Plot the laplacian of the image.
laplIm = cv2.Laplacian(smImage, cv2.CV_8U,ksize=5)
cv2.imshow("Image Laplacian",laplIm)
cv2.waitKey(0)
cv2.destroyAllWindows()

# E3: find edges using Canny edge detector
edgeIm = cv2.Canny(smImage, 35,100)
plt.imshow(edgeIm,cmap=cm.Blues_r); plt.axis('off'); plt.show()

```