```
import matplotlib.pyplot as plt
import numpy as np
import PIL
import os.path

def make_mask(rows, columns, stripe_width):
    img = PIL.Image.new('RGBA', (columns, rows))
    image = np.array(img)
    for row in range(rows):
        for column in range(columns):
            if (row+column)/stripe_width % 2 == 0:
                #(r+c)/w says how many stripes above/below line y=x
                # The % 2 says whether it is an even or odd stripe

                # Even stripe
                image[row][column] = [255, 127, 127, 0] # pale red,

            else:
                # Odd stripe
                image[row][column] = [255, 0, 255, 255] # magenta, a
    return image

def open_file(filename):
    # Using the os.path library to go to the directory
    directory = os.path.dirname(os.path.abspath(__file__))
    # full path
    fullPath = os.path.join(directory, filename)
    # opens the image
    return plt.imread(fullPath)

def draw_image(the_image):
    figure, ax = plt.subplots(1, 1)
    ax.imshow(the_image, interpolation='none')
    figure.show()

def change(image, brightness_min, brightness_max, color, from_x = 0,
    if to_x == 0:
        to_x = len(image[0])
    if to_y == 0:
        to_y = len(image)
    for r in range(from_y,to_y):
        for c in range(from_x, to_x):
            brightness = sum(image[r][c])
            if brightness>brightness_min and brightness < brightness
                image[r][c]=color

def make_transparency(rgb_img, trans_color):
    rgba_img = PIL.Image.new('RGBA',(len(rgb_img[0]),len(rgb_img)))
    Numpy_rgba_img = np.array(rgba_img)
    for r in range(len(rgb_img)):
        for c in range(len(rgb_img[0])):
            if list(rgb_img[r][c]) == trans_color:
            #if rgb_img[r][c][0] == trans_color[0] and rgb_img[r][c]
                Numpy_rgba_img[r][c]=(0,0,0,255)
    return Numpy_rgba_img

Numpy_img = open_file("minion.jpg")
Numpy_img_change = open_file("minion.jpg")
Numpy_img_mask = open_file("minionTitle.jpg")
Numpy_img_mask2 = make_transparency(Numpy_img_mask,[0,0,0])
change(Numpy_img_change,550,750,(255,0,0),from_x=290, to_x=600,from_
PIL_img = PIL.Image.fromarray(Numpy_img_change)
PIL_img_mask = PIL.Image.fromarray(Numpy_img_mask2)

PIL_img_final = PIL.Image.new('RGBA',(len(Numpy_img_mask[0]),len(Num
PIL_img_final.paste(PIL_img,(0,0),mask = PIL_img_mask)
Numpy_img_final = np.array(PIL_img_final)
fig, ax = plt.subplots(1, 4)
original = PIL_img = PIL.Image.fromarray(Numpy_img)
ax[0].imshow(original)
original.save("original.bmp")
ax[0].set_title('Original')
pixel_by_pixel = PIL_img = PIL.Image.fromarray(Numpy_img_change)
pixel_by_pixel.save("pixel_by_pixel.bmp")
ax[1].imshow(pixel_by_pixel)
ax[1].set_title('Pixel by Pixel')
mask = PIL_img = PIL.Image.fromarray(Numpy_img_mask)
mask.save("mask.bmp")
ax[2].imshow(mask)
ax[2].set_title('Mask')
mask_applied = PIL_img = PIL.Image.fromarray(Numpy_img_final)
mask_applied.save("mask_applied.bmp")
ax[3].imshow(mask_applied)
ax[3].set_title('Mask Applied')
fig.show()
```