Green Screen Car

June 10, 2019

1 Color Masking, Green Screen

1.0.1 Import resources

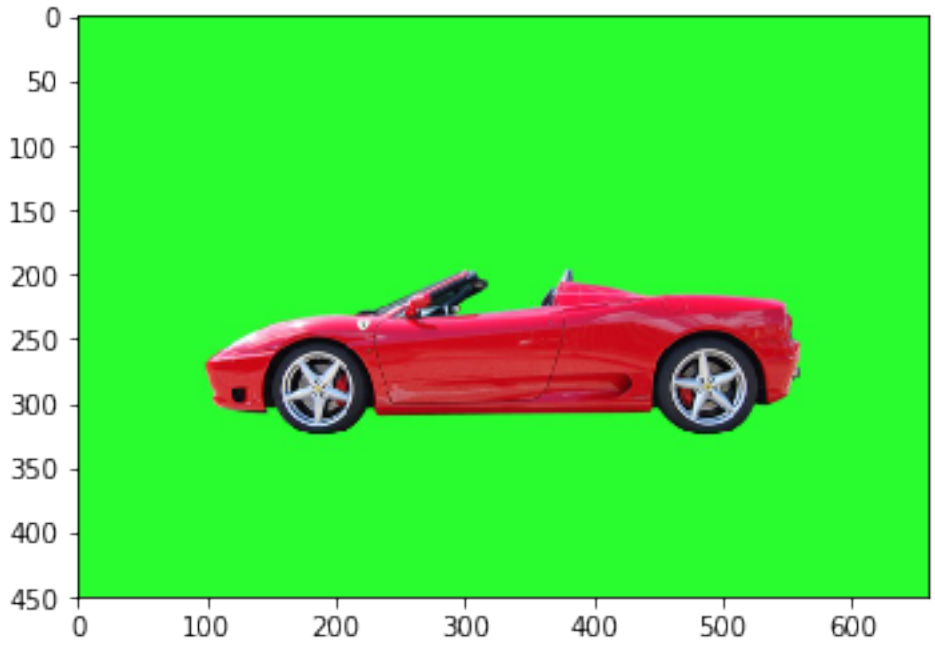
```
In [1]: import matplotlib.pyplot as plt
        import matplotlib.image as mpimg

        import numpy as np
        import cv2

        %matplotlib inline
```

1.0.2 Read in and display the image

```
In [2]: # Read in the image
        image = mpimg.imread('images/car_green_screen.jpg')

        # Print out the image dimensions (height, width, and depth (color))
        print('Image dimensions:', image.shape)

        # Display the image
        plt.imshow(image)
```

```
Image dimensions: (450, 660, 3)
```

```
Out[2]: <matplotlib.image.AxesImage at 0x7f27f1f79208>
```



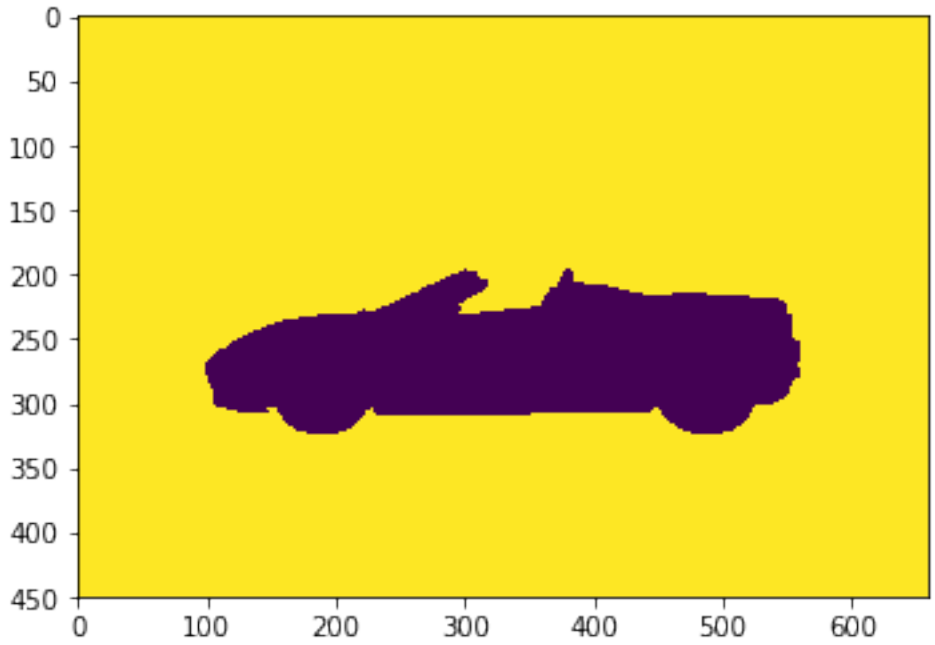
1.0.3 Define the color threshold

```
In [3]: # Define our color selection boundaries in RGB values
        lower_green = np.array([0,180,0])
        upper_green = np.array([100,255,100])
```

1.0.4 Create a mask

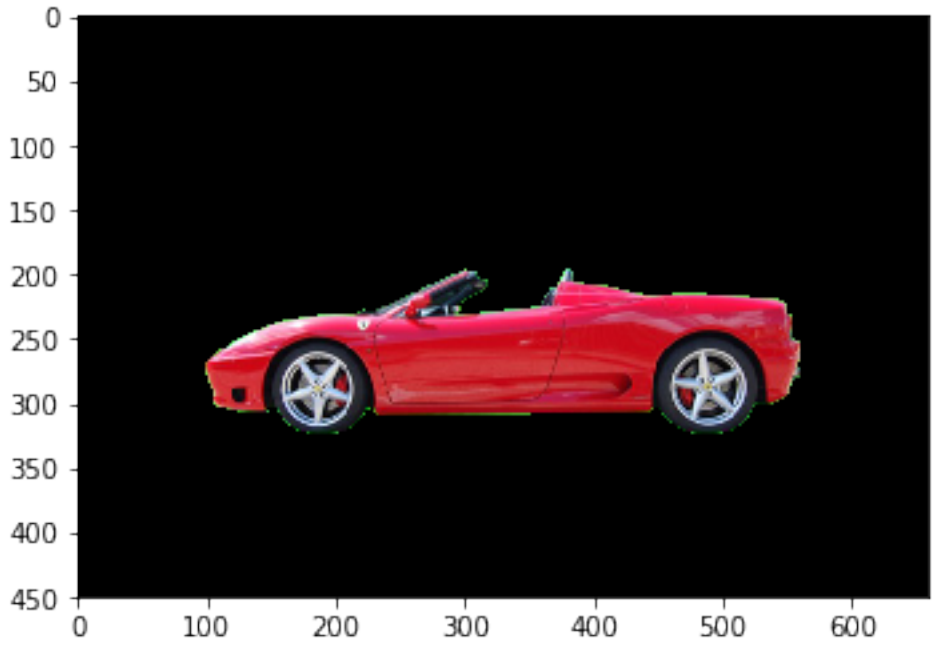
```
In [8]: # Define the masked area
        mask = cv2.inRange(image, lower_green, upper_green)

        # Vizualize the mask
        plt.imshow(mask)
```

```
Out[8]: <matplotlib.image.AxesImage at 0x7f27ba5e0080>
```



In [9]:
```
# Mask the image to let the car show through
masked_image = np.copy(image)

masked_image[mask != 0] = [0, 0, 0]

# Display it!
plt.imshow(masked_image)
```

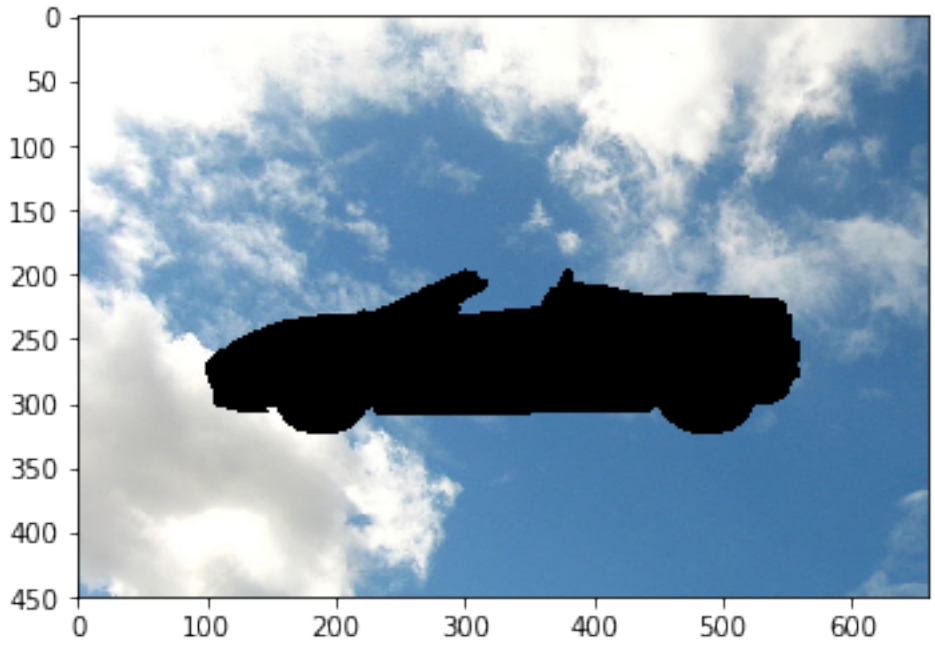
Out[9]:
```
<matplotlib.image.AxesImage at 0x7f27ba546198>
```



1.1 TODO: Mask and add a background image

In [27]:
```
# Load in a background image, and convert it to RGB
background_image = mpimg.imread('images/sky.jpg')

## TODO: Crop it or resize the background to be the right size (450x660)
# Hint: Make sure the dimensions are in the correct order!

im_bg_copy = np.copy(background_image)
im_bg_copy_croped = im_bg_copy[0:450, 0:660]



## TODO: Mask the cropped background so that the car area is blocked
# Hint: mask the opposite area of the previous image

im_bg_copy_croped[mask == 0] = [0, 0, 0]


## TODO: Display the background and make sure
plt.imshow(im_bg_copy_croped)
```

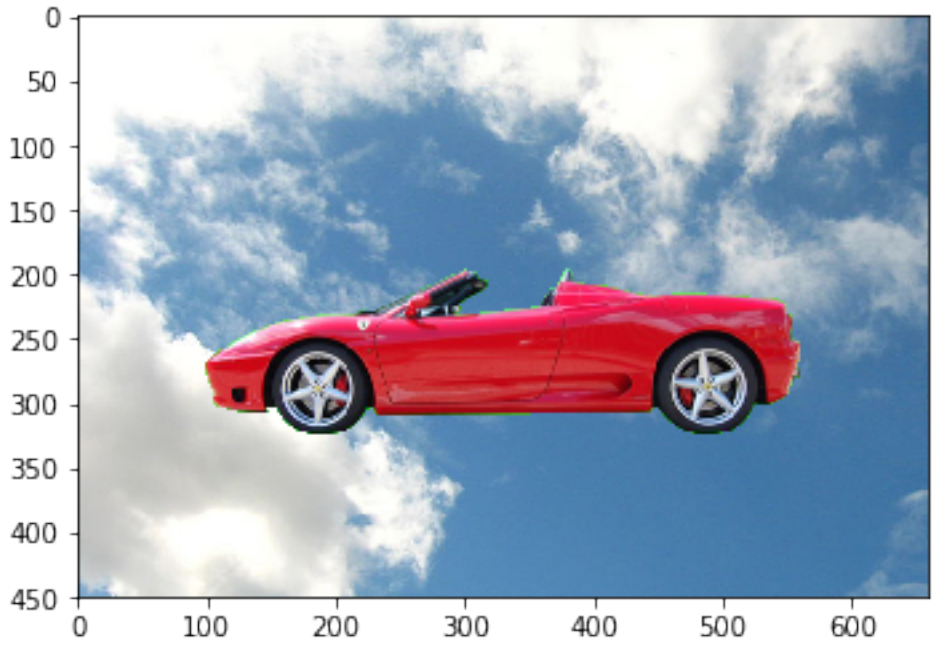
Out[27]:
```
<matplotlib.image.AxesImage at 0x7f27b87da470>
```



1.1.1 TODO: Create a complete image

In [20]:
```
## TODO: Add the two images together to create a complete image!
# complete_image = masked_image + crop_background

final = im_bg_copy_croped + masked_image


plt.imshow(final)
```

Out[20]:
```
<matplotlib.image.AxesImage at 0x7f27b8a0add8>
```



In [ ]:

In [ ]: