

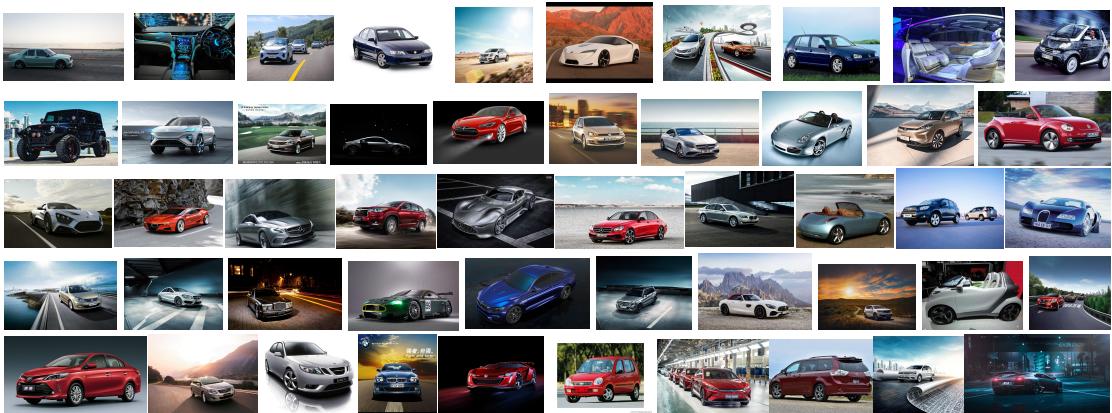
《计算机视觉》（本科，2023）作业 3

- 在搜索引擎上按照某一关键词，搜索 55 张不同的图像，从中选出 5 张作为检索请求，另 50 张作为被检索图像。

【5 张检索请求图像贴这里】



【50 张被检索图像贴这里】



- 以全局 RGB 颜色直方图（每通道 bin 的数量为 8）作为特征，进行图像检索。展示每个检索请求及对应的前 3 个结果。

【特征抽取代码贴这里】

```
def get_hist_feature(image):  
    image_hist = cv2.calcHist([image], [i for i in range(img.shape[2])],  
    None, [8 for i in range(img.shape[2])], [0, 256, 0, 256, 0, 256])  
    image_hist = cv2.normalize(image_hist, image_hist, 0, 255,  
    cv2.NORM_MINMAX).flatten()  
    return image_hist
```

【检索（特征匹配）代码贴这里】

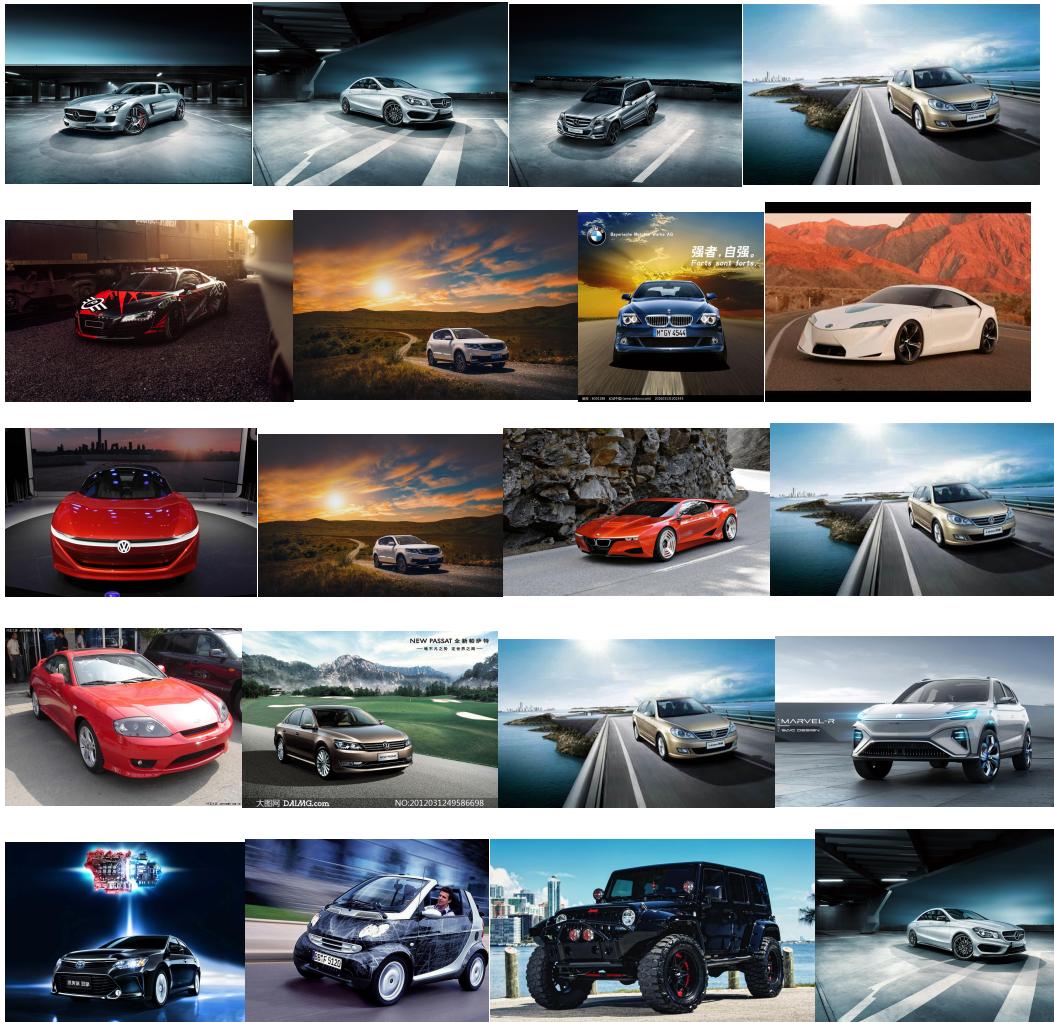
```
path = os.getcwd()  
img_dir_path = path + "\img"  
img_list = os.listdir(img_dir_path)  
assert len(img_list) == 55  
for i in range(5):  
    img_path = img_dir_path + '\\' + img_list[i]  
    print("Test img " + str(i) + " : " + img_path)  
    img = cv2.imread(img_path)  
    img_hist = get_hist_feature(image=img)  
    similarity = []  
    for j in range(5, 55):
```

```

target_path = img_dir_path + '\\' + img_list[j]
target_img = cv2.imread(target_path)
target_hist = get_hist_feature(image=target_img)
compare_coefficient = cv2.compareHist(img_hist, target_hist,
cv2.HISTCMP_INTERSECT)
similarity.append(compare_coefficient)
top_3_similarity = sorted(similarity)[-3:]
top_3_similarity.reverse()
for k in range(len(top_3_similarity)):
    index = similarity.index(top_3_similarity[k])
    print("Similarity rank " + str(k + 1))
    print("Similar image " + img_list[index + 5])
    print("====")

```

【结果（检索请求+前 3 个结果）贴这里】



每行第一个为检索请求，后三个为前三的结果

3. 选择 1-2 种其它特征，重复问题 2。

【特征抽取代码贴这里】

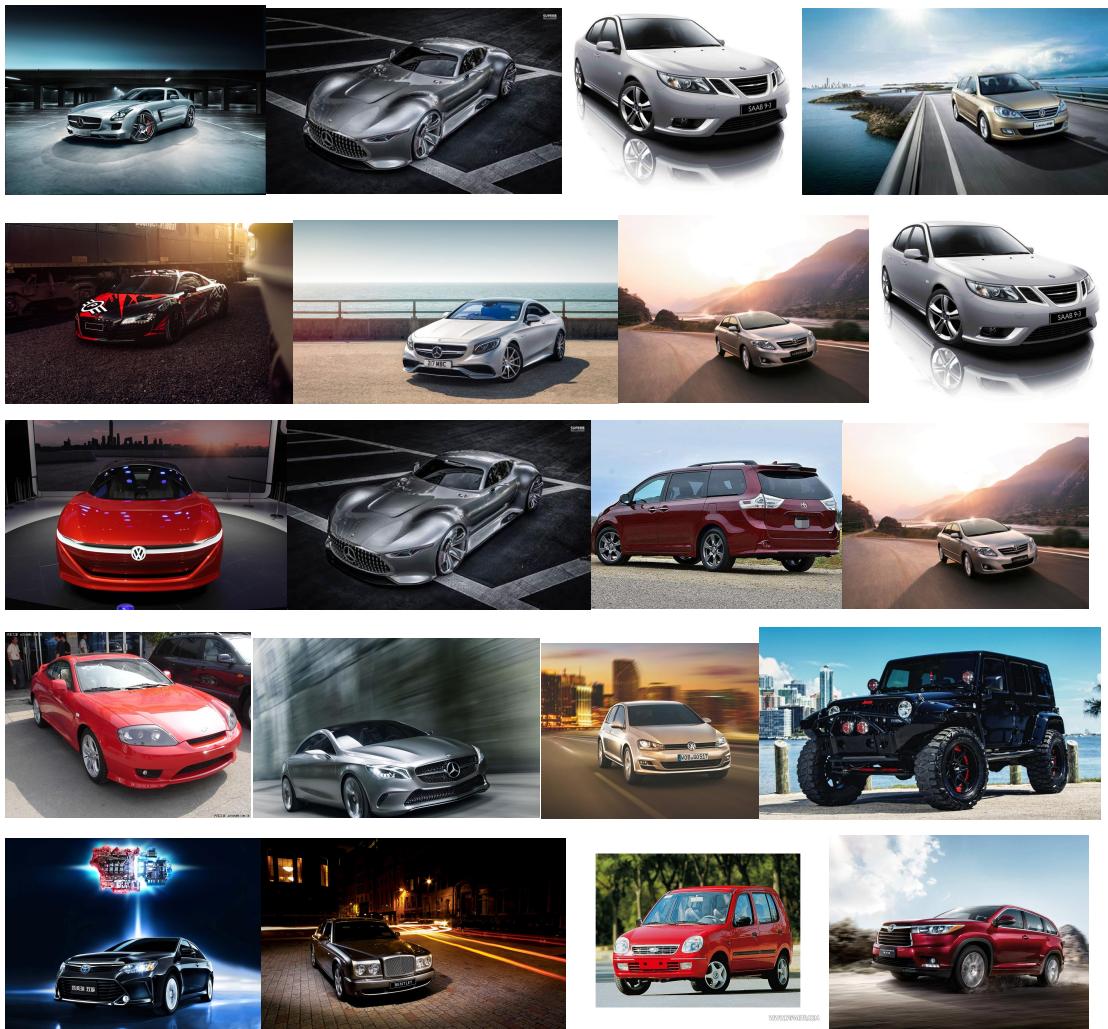
```
orb = cv2.ORB_create()
```

```
img_kp, img_des = orb.detectAndCompute(img, None)
```

【检索（特征匹配）代码贴这里】

```
def get_similarity(des1, des2, threshold=0.75):
    matches = bf.knnMatch(des1, trainDescriptors=des2, k=2)
    good = [m for (m, n) in matches if m.distance < threshold * n.distance]
    return len(good) / len(matches)
```

【结果（检索请求+前 3 个结果）贴这里】



4. 将问题 2 和问题 3 的结果进行比较和分析。

【比较和分析结果贴这里】

两种效果匹配的结果都不尽人意，采用 RGB 颜色直方图的效果，在汽车与背景图片的色调一致时，如第一行结果，可以看到检索结果不错，检索出来的结果是与原图色系接近的，对于第二行结果，可以看到背景色的色调是较为接近的。其余三者的检索结果都不太好，其中第四行的图片，`cv2.compareHist` 结果最大值为 1000+，而其余的最大值仅为 700+，而对于 ORB 检测结果，可以看到，在特征点的选取上，orb 也会识别其背景中的点，因此最终匹配度最高的不一定是汽车曲线相似度最高的，背景曲线也占了一部分。可以看到图片 12 的特征点匹配的是背景的引擎和红色汽车的车轮。

