

helper.py فایل

همراه تمامی کدها، یک فایل helper.py قرار دارد که به منظور خوانایی کدهاست. تابع هایی که در فایل اصلی کدها هستند (Qi.py) در فایل helper پیاده سازی میشوند تا کد اصلی کوتاه تر و خواناتر باشد. در توضیحات سوال ها هم کدهای اصلی هم تابع های پیاده سازی شده را تا جای ممکن توضیح میدهم.

در این تمرین برای جداسازی سنجاب ها از graph\_cut استفاده کردم. برای جدا کردن foreground از background توی اینترنت سرچ کردم که به این صفحه رسیدم:

# [https://docs.opencv.org/3.4/d8/d83/tutorial\\_py\\_grabcut.html](https://docs.opencv.org/3.4/d8/d83/tutorial_py_grabcut.html)

ایده اصلی من برای حل این سوال از graph cut از این صفحه میاد.

کل کد در یک for دوتایی قرار داره (هر دو عکس عملیات مشابهی دارن)  
ایمیج رو لود میکنم و با SLIC اون رو over-segment میکنم و ازش عکس لیل میگیرم. دلیل این کار اینه که مقایسه سوپر پیکسل هایی که یک رنگ دارند (میانگین رنگ پیکسل های داخلش) برای graph cut راحت تر خواهد بود تا تمام پیکسل ها با رنگ های متفاوت رو بهش بدیم. در واقع اول عکس کامل رو میدادم ولی چون نتیجه خوبی نمیگرفتم تصمیم گرفتم که اول به over-segment کنم و بعدش روش graph-cut بزنم.  
عکس over-segment شده رو هم خروجی گرفتم.

حالا برای گرف-کات ابتدا به مستطیل مشخص میکنم که یعنی جسم foreground من داخل این مستطیل (به عبارت دیگه source و درون مستطیل و sink رو خارج مستطیل میذاره و الگوریتم flow رو اجرا میکنه).  
این کار باعث میشه یک ماسک درست بشه (همون ماسکی که به grab\_cut\_with\_rectangle دادم) که خارج مستطیل کاملاً ۰ هست. داخل مستطیل و جایی که گراف-کات خارج جسم اصلی تشخیص داده ۲ میشه و داخل جسم ۱. اما خب ممکنه گراف-کات به جاهایی رو اشتباه کرده باشه. جاهایی در جسم اصلی بوده باشه که در نظر نگرفته باشه یا جاهایی رو به اشتباه جسم اصلی در نظر گرفته.

در این مورد، میتونیم ماسک تولید شده رو تغییر بدیم. بدین صورت که جاهایی که جز جسم هستند ولی توی ماسک هستند رو به خط سفید بکشیم تا ۱ بشه. و جاهایی که به اشتباه جز جسم در نظر گرفته رو به خط بکشیم و سیاه کنیم. مثلاً چندین خط بکشیم که فلان جا و فلان جا نبوده ولی الگوریتم به اشتباه تشخیص داده که هست.

و دوباره الگوریتم رو این بار با ماسک اجرا کنیم که این کار با تابع `grab_cut_with_mask` انجام میشه.  
روند کد هم به همین صورته و با توجه به اسم تابع ها میشه راحت فهمید هرکدوم چی کار میکنن.  
( چون کد دوبار ران میشه (برای هر عکس یه بار) خط هایی که برای مرحله دوم گراف-کات روی ماسک میکشیم  
برای هرکدوم از عکسا متفاوتی که این کار رو دو تابع `create_special_mask_for_image_1` و `create_special_mask_for_image_2` انجام میدن )