

فایل helper.py

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.

بعد از خواندن عکس، ابتدا تغییراتی رو آن اعمال میکنیم. تابع normalize که در helper.py پیاده سازی شده است این وظیفه را بر عهده دارد. دلیل این تغییرات سایه ای که در عکس وجود دارد است. این سایه در گرادیان تصویر تاثیر منفی میگذاشت. سعی کردم با این تابع تاثیر سایه رو توی عکس از بین ببرم.

سپس عکس را سیاه سفید میکنیم (برای گرادیان گرفتن)

حال میخواهیم یک دایره از نقاط دور تسبیح ایجاد کنیم. این کار میتواند دستی انجام شود که اگر نقاط به خوبی انتخاب شوند، نتیجه بهتری هم خواهیم گرفت. اما من یک دایره از نقاط رو دور تسبیح قرار دادم. شعاع و مرکز دایره دستی انتخاب شده اند و کدی که این دایره از نقاط رو به وجود میاره رو هم از اینترنت کپی کردم.

سپس گرادیان عکس را محاسبه میکنیم. توابعی که در تمرین سری ۲ پیاده سازی کرده بودم را کپی کردم و از همان ها برای محاسبه گرادیان عکس در دو راستای x و y استفاده کردم. عکس گرادیان ها را هم خروجی گرفتم.

برای active_contour یک for ۵۰۰ تایی قرار دادم. در هر بار، تابع calculate_move روی نقاط اجرا میشود. سپس نقاط را روی عکس نشان می دهیم و عکس را برای ساختن فیلم ذخیره میکنم. در انتها برای ساختن فیلم تابع make_video را پیاده سازی کردم که از یک کد آماده در اینترنت گرفته شده است و فریم هایی که ذخیره کردم را پشت سر هم میگذارد و یک فیلم خروجی میدهد.

هسته اصلی کد تابع calculate_move است که در helper.py پیاده سازی شده است.

این تابع به ازای هر نقطه ، تابع `move_one_point` را روی آن نقطه فراخوانی میکند. این تابع، بهترین حرکت برای آن نقطه را بررسی میکند و آن نقطه را حرکت میدهد. این تابع دقیقا مشابه اسلایدها پیاده سازی شده اند. فرمول هایی که در اسلایدها آمده بودند پیاده سازی شده اند. با محاسبه انرژی درونی (دو ترم) و انرژی خارجی، مینیمم انرژی را برای حرکت بعدی نقطه مورد نظر پیدا میکنیم و آن را حرکت می دهیم. این کار را برای تمام نقاط انجام میدهیم تا یک دور تمام شود. این کار ۵۰۰ دور انجام میشود.