

الف) همانطور که مشاهده میکنید در خروجی سلول اول، تصویر جدید از لحاظ شدت روشنایی و رنگ ها کاملا شبیه به تصویر reference شده است. اما ساختار تصویر خروجی کاملا شبیه به تصویر source بوده و تغییر نکرده. در خروجی سلول دوم هم همانطور که انتظار داشتیم، cdf تصویر خروجی بر روی cdf تصویر reference منطبق شده. دقت کنید که مجموعه همه شدت روشنایی های تصویر خروجی یک زیرمجموعه از مجموعه همه شدت روشنایی های تصویر reference بوده و cdf هر شدت روشنایی در تصویر خروجی برابر cdf همان شدت روشنایی در تصویر reference است.

ب) مسایلی که دوست داریم از لحاظ رنگی یک تصویر را مشابه یک تصویر دیگر کنیم. مسایلی که به دلیل عکس برداری باعث شده که تصویر ما رنگش را آن طور که باید حفظ نکرده و دوست داریم ترکیب رنگی آن را شبیه به حالت اصلی خودش کنیم.

توضیح کد)
توضیح تابع calc_hist

کل عکس را سرچ کرده و هنگامی که پیکسلی با شدت روشنایی آ دیدیم، مقدار آن را به هیستوگرام مربوطه اضافه کرده. پیچیدگی این الگوریتم $O(w * h)$ میباشد که w و h ابعاد تصویر ورودی است.

توضیح تابع calc_cdf
چون میتوان این قسمت را به صورت بازگشتی حل کرد، آن را با dynamic programming حل کرده ایم. ابتدا هیستوگرام را حساب کرده و سپس با dp تابع توزیع تجمعی عکس را پیدا کرده ایم. پیچیدگی این الگوریتم $O(w * h)$ میباشد که w و h ابعاد تصویر ورودی است.

توضیح تابع hist_matching

اگر بخواهی مقداری درک کد زده شده را ساده تر کنم، باید بگوییم که این قسمت مانند قسمت ادغام الگوریتم merge sort هست. به این صورت که دو پوینتر داریم a و z ، که مبینیم cdf پیکسل a در تصویر اولیه با cdf پیکسل z در تصویر مرجع برابر شده و چون cdf یک تابع صعودی است، به سمت جلو حرکت کرده تا برای همه a ها که در تصویر اولیه موجود هستند، مقداری که نزدیک است cdf ان در تصویر مرجع را بیابیم.