

«به نام خدا»

استاد: دکتر محمدرضا محمدی
درس مبانی بینایی کامپیوتر

نام: فاطمه زهرا بخشنده
شماره دانشجویی: 98522157

گزارش تمرین 3:

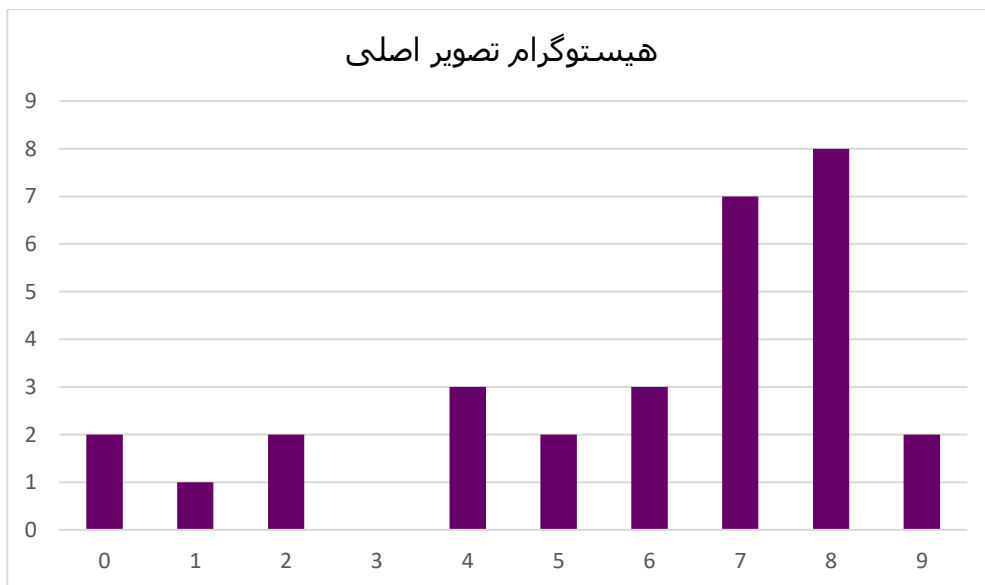
سوال اول:

کشیدن هیستوگرام:

هیستوگرام برای یک تصویر دیجیتال با سطوح روشنایی در محدوده $[0, L - 1]$ تابعی است گسسته که به صورت زیر تعریف میشود:

$$h(r_k) = n_k$$

که r_k یک سطح روشنایی در محدوده مورد نظر است و n_k تعداد پیکسل هایی است که دارای آن سطح روشنایی هستند.



بخش اول:

برش و کشش هیستوگرام:

ساده ترین راه برای استفاده از تمام سطوح روشنایی، کشش هیستوگرام است. که فرمول آن به صورت زیر است:

$$g(x, y) = stretch[f(x, y)] = \left(\frac{f(x, y) - f_{min}}{f_{max} - f_{min}} \right) (Max - Min) + Min$$

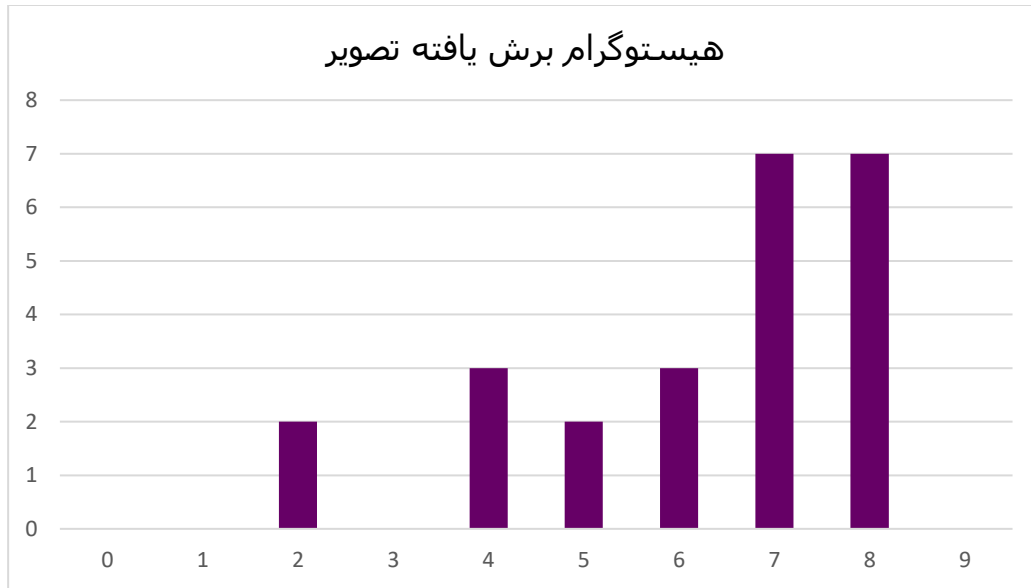
حالا اگر بخواهیم ابتدا آن را 10% برش داده، سپس بکشیم، از فرمول زیر استفاده می کنیم که در واقع بجای f_{min} ، f_{10} و بجای f_{max} ، f_{90} را قرار داده ایم. یعنی تصویر را با برش 10 درصدی می کشیم.

$$g(x, y) = clip[f(x, y)] = \left(\frac{f(x, y) - f_{10}}{f_{90} - f_{10}} \right) (Max - Min) + Min$$

در برش هیستوگرام، بخشی از مولفه های پایین و بالا در نمودار هیستوگرام را قطع میکنیم. تعداد پیکسل ها 30 تاست.

$$0.1 \times 30 = 3$$

پس 3 مولفه از هر طرف را حذف می کنیم. از سمت راست شروع می کنیم. 2 تا رنگ 0 داریم که حذف می کنیم. 1 رنگ 1 داریم که حذف می کنیم. پس 3 مولفه از اول آن حذف شد. یک بار هم از آخر شروع می کنیم. 2 تا رنگ 9 داریم که حذف می کنیم. 8 تا رنگ 8 داریم که 1 ای از آن را حذف می کنیم. پس 3 مولفه از آخر هیستوگرام حذف شد.



حالا روشن ترین رنگ (آخرین جایی که f صفر نیست) رنگ 8 و تیره ترین رنگ (اولین جایی که f صفر نیست) رنگ 2 است. پس $f_{90} = 8$ و $f_{10} = 2$. Min و Max هم که در این مثال به ترتیب 9 و 0 هستند. در فرمول قرار می دهیم:

$$g(x, y) = clip[f(x, y)] = \left(\frac{f(x, y) - 2}{8 - 2} \right) (9 - 0) + 0 = 1.5f(x, y) - 3$$

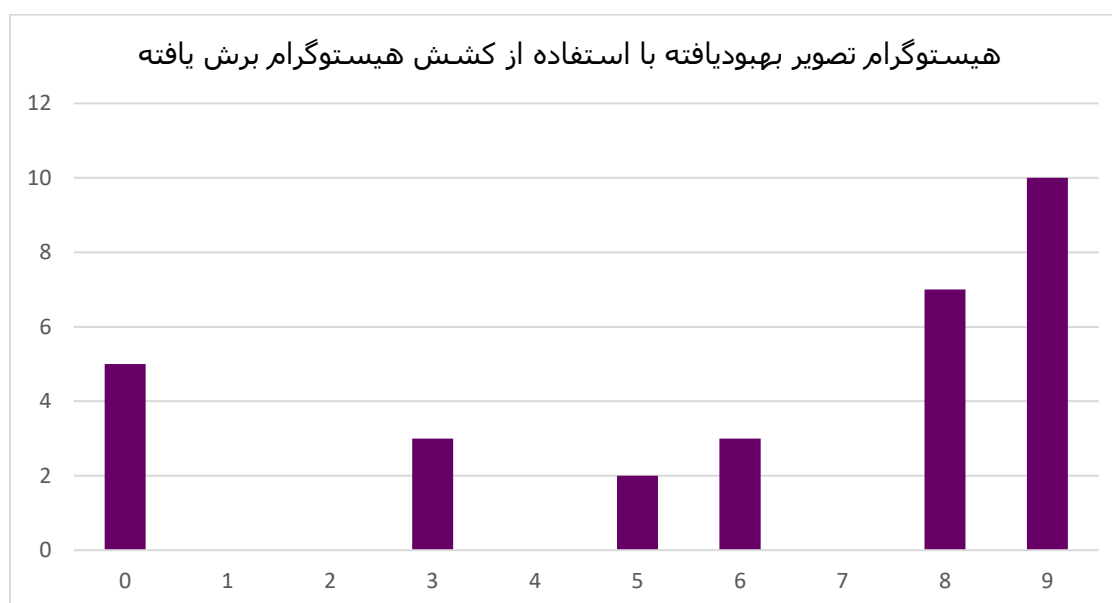
این تبدیل خطی را روی تمام رنگ ها اعمال می کنیم.

$r = f(x, y)$	0	1	2	3	4	5	6	7	8	9
$s = 1.5f(x, y) - 3$	-3	-1.5	0	1.5	3	4.5	6	7.5	9	10.5
Round(s)	0	0	0	2	3	5	6	8	9	9

تصویر بهبود یافته با کشش به صورت زیر است:

8	8	9	9	9	9
0	0	3	3	3	9
8	0	5	5	0	9
9	0	6	9	9	8
9	8	6	6	8	8

هیستوگرام تصویر بهبود یافته نیز به صورت زیر است:



می‌بینیم که کشش روی هیستوگرام اعمال شده و مقادیر آن پراکنده تر شده اند و از 0 تا 9 کشیده شده اند.

بخش دوم:

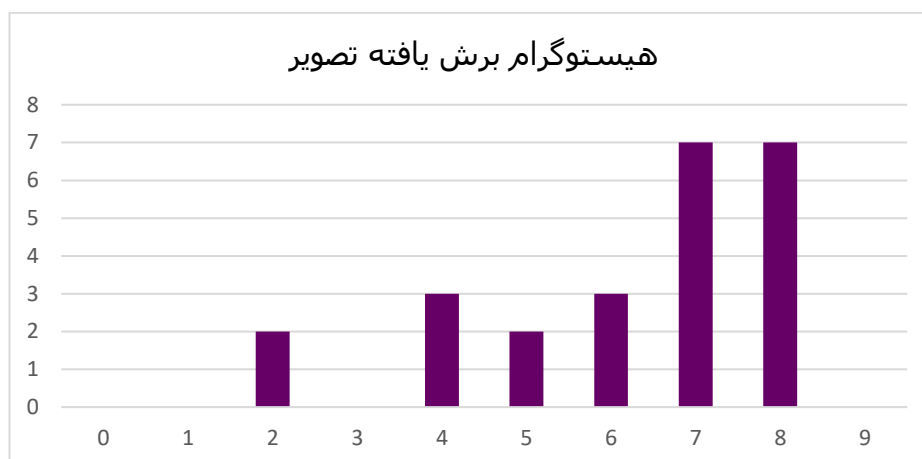
برش هیستوگرام:

در این بخش قبل از متعادل سازی، ابتدا تنها برش هیستوگرام بدون کشش را روی تصویر اصلی انجام می دهیم. تعداد پیکسل ها 30 تاست.

$$0.1 \times 30 = 3$$

پس 3 مولفه از هر طرف را حذف می کنیم. از سمت راست شروع می کنیم. 2 تا رنگ 0 داریم که حذف می کنیم. 1 رنگ 1 داریم که حذف می کنیم. پس 3 مولفه از اول آن حذف شد.

یک بار هم از آخر شروع می کنیم. 2 تا رنگ 9 داریم که حذف می کنیم. 8 تا رنگ 8 داریم که 1 ای از آن را حذف می کنیم. پس 3 مولفه از آخر هیستوگرام حذف شد.



متعادل سازی هیستوگرام:

متعادل سازی هیستوگرام (Equalization Histogram) پردازشی است که هیستوگرام تصویر را تا حد امکان مسطح میکند. اساس متعادل سازی هیستوگرام مبتنی بر تئوری احتمالات است که در آن هیستوگرام به عنوان تابع توزیع احتمال سطوح روشنایی تصویر در نظر گرفته میشود. متعادل سازی هیستوگرام برابر با تابعی است که این توزیع احتمال را به توزیع احتمال یکنواخت تبدیل کند.

$$T(r) = (L - 1) p_r(r), \quad s_k = \frac{L - 1}{n} \sum_{j=0}^k n_j$$

طبق این فرمول مراحل متعادل سازی هیستوگرام را انجام می دهیم.

k	0	1	2	3	4	5	6	7	8	9
n_k تصویر اصلی	2	1	2	0	3	2	3	7	8	2
n_k برش یافته	0	0	2	0	3	2	3	7	7	0
$\sum_{j=0}^k n_j$	0	0	2	2	5	7	10	17	24	24
$\sum_{j=0}^k \frac{n_j}{n}$	0	0	$\frac{2}{30}$	$\frac{2}{30}$	$\frac{5}{30}$	$\frac{7}{30}$	$\frac{10}{30}$	$\frac{17}{30}$	$\frac{24}{30}$	$\frac{24}{30}$
$(L - 1) \sum_{j=0}^k \frac{n_j}{n}$	0	0	0.6	0.6	1.5	2.1	3	5.1	7.2	7.2
Round	0	0	0	0	2	2	3	5	7	7

تصویر بهبود یافته به صورت زیر است:

5	5	7	7	7	7
0	0	2	2	2	7
5	0	2	2	0	7
7	0	3	7	7	5
7	5	3	3	5	5

هیستوگرام تصویر بهبود یافته نیز به صورت زیر است:



روش کشش هیستوگرام در واقع یک تبدیل خطی به ما می دهد. اگر تصویر ما مقادیر روشن و تیره بیشتری داشته باشد و مقادیر میانی خاکستری کمی داشته باشد، کشش هیستوگرام خیلی مناسب نیست. اما متعادل سازی هیستوگرام تابع ارتقای کنتراست را متناسب با نوع تصویر ارائه می دهد که می تواند غیر خطی باشد و اگر رنگ های میانی هم نداشته باشیم هیستوگرام آن قسمت را متعادل می کند.

برای این مثال دیدیم که تقریباً هردو عملکرد خوبی داشتند، اما روش متعادل سازی هیستوگرام همانطور که انتظار می رفت در قسمت میانی هم متعادل سازی بهتری انجام داده است و روش کشش، رنگ هارا بیشتر به دو طرف کشیده است.

منابع: اسلاید ها

سوال دوم:

الف) برای بهبود تصویر با استفاده از متعادل سازی هیستوگرام، ابتدا هیستوگرام تصویر را بدست می آوریم. برای اینکار، باید تعداد دفعات تکرار هر رنگ را در عکس بدست آوریم. با استفاده از تابع unique کتابخانه numpy میتوان برای رنج رنگ 0 تا 255 این مقدار را بدست آورد.

سپس برای متعادل سازی هیستوگرام، فرمول زیر را مرحله به مرحله بر روی تصویر اعمال کنیم تا نمودار هیستوگرام نرمالسازی شود و سپس سطوح روشنایی متعادل سازی شده را جایگزین سطوح روشنایی قبلی میکنیم. هرمرحله جداگانه با کامنت در کد مشخص شده است.

$$S_k = (L - 1) \sum_{j=0}^k \frac{n_j}{n}$$

n_j : هیستوگرام رنگ های تصویر از 0 تا 255

L : 256 سطح روشنایی

n : تعداد کل پیکسل های تصویر

نتیجه:



میبینیم که کنتراست تصویر افزایش یافته است و وضوح آن بیشتر شده است.

حالا از تابع آماده متعادل سازی استفاده می کنیم. نتیجه:



نتیجه یکسانی گرفتیم پس متعادل سازی هیستوگرام را درست پیاده سازی کرده ایم.

ب) برای متعادل سازی هیستوگرام، کنتراست کلی تصویر در نظر گرفته میشود و با شدت نور تمامی نقاط کار داریم. در حالت کلی این روش تصویر را واضح میکند ولی اگر مانند ورودی این سوال تصویر به گونه ای باشد که سوژه و بک گراند عکس اختلاف شدت رنگ داشته باشند، این روش مناسب نیست. متعادل سازی بک گراند عکس را واضحتر کرده ولی چون تونل چوبی را تیره تر کرده، جزئیات آن مشخص نیست و دیگر ساختار چوب را به خوبی نمی بینیم. همچنین درخت هایی که بک گراندشان آسمان است هنوز واضح نیستند. به همین دلیل سراغ روش Adaptive Histogram Equalization می رویم که همسایه های نزدیک پیکسل ها را در نظر می گیرد و در آن ناحیه متعادل سازی را انجام میدهد.

این روش جایی مشکل ساز میشود که در تصویر تکه یکنواخت (از لحاظ شدت روشنایی، مثلا یکدست سیاه یا سفید) داشته باشیم زیرا مثلا شدت روشنایی 2 تا 5 را به 0 تا 255 مپ میکند و این موضوع باعث ایجاد نویز در آن نواحی میشود. برای جلوگیری از این اشکال سراغ Contrast Limitation می رویم که برای تقویت کنتراست محدودیت بگذاریم. ترکیب این دو روش منجر به CLAHE میشود که با اعمال آن روی تصویر ورودی این سوال، متوجه میشویم متعادل سازی بسیار بهتر اتفاق افتاده است.

برای پیدا کردن بهترین مقادیر clip_limit می توانیم مقادیر مختلف را برای آن امتحان کنیم. مقدار دیفالت آن 40 است که اگر با همین مقدار، clahe انجام شود نتیجه زیر حاصل می شود:



که برای این تصویر مطلوب ما نیست. با امتحان مقادیر مختلف بنظرم مقادیر 2 تا 4 برای آن، نمای بهتری از تصویر را ارائه می دهد. مقدار 3 را به آن می دهیم. نتیجه:



حالا نتایج histogram equalization و clahe را کنار هم قرار می دهیم و مقایسه می کنیم.





می بینیم که برای این تصویر از clahe نتیجه بهتری گرفتیم. که دلیل آن در بالا ذکر شد.

(پ) مراحل را برای تصویر city تکرار می کنیم. نتیجه histogram equalization که پیاده سازی کردیم:



نتیجه histogram equalization موجود در openCV:



که این دو یکسان هستند.

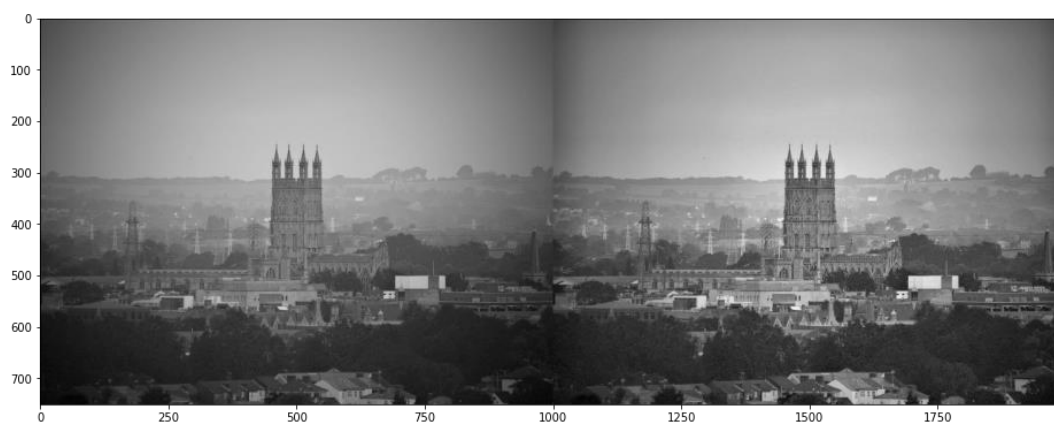
نتیجه clahe:

در بخش هایی از این تصویر تکه یکنواخت وجود دارد (از لحاظ شدت روشنایی، مثلا آسمان تقریبا یکدست است). طبق چیزی که قبلا گفتیم، این روش در این مواقع مشکل ساز می شود. چون مثلا شدت روشنایی 2 تا 5 که شاید بخاطر خطای دوربین باهم برابر نشدند را به 0 تا 255 مپ میکند و این موضوع باعث ایجاد نویز بیشتر در آن نواحی میشود. مثلا نتیجه با $\text{clip_limit} = 10$ به صورت زیر است:



همانطور که گفتیم برای بهبود آن clip_limit را تنظیم می کنیم. برای این تصویر بنظرم باید آن را 1 یا نهایتا 2 بگذاریم. (با 2 هم کمی آسمان نویزی می شود).

نتیجه با $\text{clip_limit} = 1$:



می بینیم که نتیجه برای این تصویر تقریبا نزدیک به histogram equalization است.

ت) histogram equalization یک فرآیند غیر خطی است. تقسیم کانال ها و equalize هر کانال به طور جداگانه راه مناسبی برای equalize کردن کنتراست نیست. equalization مقادیر Intensity تصویر را شامل می شود نه اجزای رنگ. بنابراین این روش تغییرات چشمگیری در تعادل رنگ تصویر ایجاد می کند. پس برای یک تصویر رنگی RGB ساده، HE نباید به صورت جداگانه در هر کانال اعمال شود. بلکه باید طوری اعمال شود که مقادیر Intensity بدون برهم زدن تعادل رنگ تصویر برابر شوند. در واقع این روش ارتباط 3 کانال را در نظر نمی گیرد و تصویر را خراب می کند.

یک روش دیگر استفاده از histogram equalization بر اساس میانگین کانال های رنگی است. در واقع هنگام محاسبه هیستوگرام، می توانیم سه کانال را با هم در نظر بگیریم. این روش ارتباط کانال های R، G و B را در نظر می گیرد و نتایج بهتری به دست می آورد. فرق آن با روش قبلی استفاده از هیستوگرام میانگین هیستوگرام سه کانال است.

روش بهتر، تبدیل فضای رنگی تصویر از RGB به یکی از فضاهای رنگی است که مقادیر شدت را از اجزای رنگ جدا می کند. مثل HSV/HLS، YUV و YCbCr. ابتدا تصویر را از RGB به یکی از فضاهای رنگی ذکر شده در بالا تبدیل می کنیم. از بین این ها YCbCr ترجیح داده می شود زیرا برای تصاویر دیجیتال طراحی شده است. برای مثال ابتدا تصویر R به فضای رنگی HSI تبدیل می کنیم.

$$H = \begin{cases} \theta & \text{if } B \leq G \\ 360 - \theta & \text{if } B > G \end{cases} \quad \text{with } \theta = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R-G) + (R-B)]}{[(R-G)^2 + (R-B)(G-B)]^{\frac{1}{2}}} \right\}$$

$$S = 1 - \frac{3}{(R+G+B)}[\min(R, G, B)]$$

$$I = \frac{1}{3}[R+G+B]$$

سپس روش grayscale را بدون تغییر در رنگ و اشباع تصویر روی درخشندگی اعمال می کنیم. سپس تصویر RGB خروجی را از سه کانال پردازش شده بازسازی می کنیم.

RG sector ($0 \leq H < 120$)

$$B = I(1-S)$$

$$R = I \left[1 + \frac{S \cos H}{\cos(60-H)} \right]$$

$$G = 1 - (R+B)$$

GB sector ($120 \leq H < 240$)

$$R = I(1-S)$$

$$G = I \left[1 + \frac{S \cos(H-120)}{\cos(60-(H-120))} \right]$$

$$B = 1 - (R+G)$$

BR sector ($240 \leq H < 360$)

$$G = I(1-S)$$

$$B = I \left[1 + \frac{S \cos(H-240)}{\cos(60-(H-240))} \right]$$

$$R = 1 - (G+B)$$

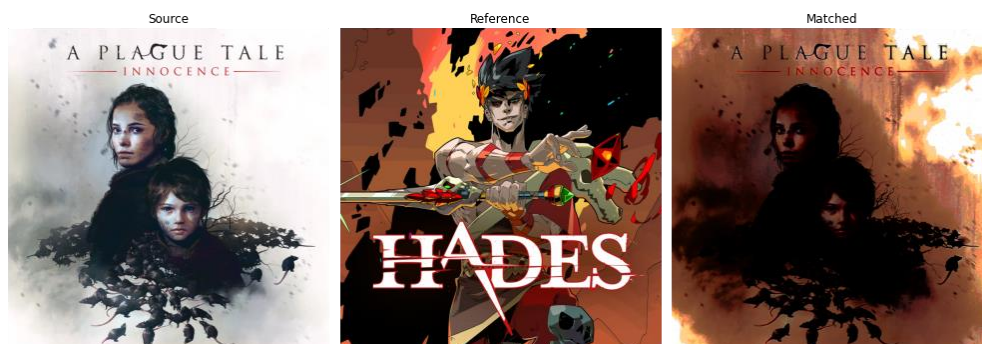
این روش اغلب نتایج خوبی دارد، اما گاهی اوقات ممکن است اینطور نباشد.

منابع: لینک و لینک

سوال سوم:

الف) کاربردهایی وجود دارد که ارتقاء تصویر به روش یکنواخت ساختن هیستوگرام بهترین راه حل نیست. در برخی موارد لازم است که هیستوگرام تصویر مورد پردازش مشابه با یک هیستوگرام از پیش تعیین شده باشد.

میتوان ابتدا تابع متعادل سازی هیستوگرام تصویر ورودی را اعمال کرد و سپس معکوس تابع متعادل سازی تصویر مرجع را بر آن اعمال نمود. نتیجه:



ب) چون تصاویر ما رنگی هستند، برای پیاده سازی آن، باید cdf هیستوگرام های سه کانال قرمز و سبز و آبی تصویر ورودی و تصویر مرجع را بدست آورد و سپس به ازای cdf های متناظر یکسان در تصویر ورودی و مرجع، matching را انجام دهیم. ممکن است برای مقادیری cdf یکسان موجود نباشد که در این صورت اولین cdf مرجع بزرگتر از cdf تصویر را جایگذاری میکنیم.

مراحل مختلف در کد با کامنت مشخص شده است.

نتیجه:



نتیجه مشابه خروجی تابع آماده تطبیق هیستوگرام شد. پس این تابع را درست پیاده سازی کردیم.

(پ) نتیجه با تابع آماده:



نتیجه با تابعی که پیاده سازی کردم:



نتایج مشابهند. هیستوگرام تصویر ورودی A Plague Tale را بر هیستوگرام تصویر ورودی Hades تطبیق دادیم.

منابع: اسلاید ها