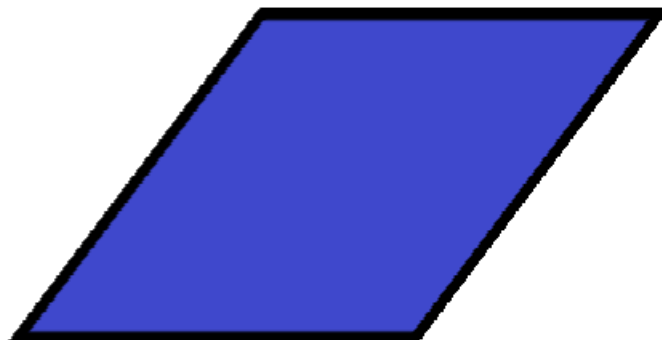


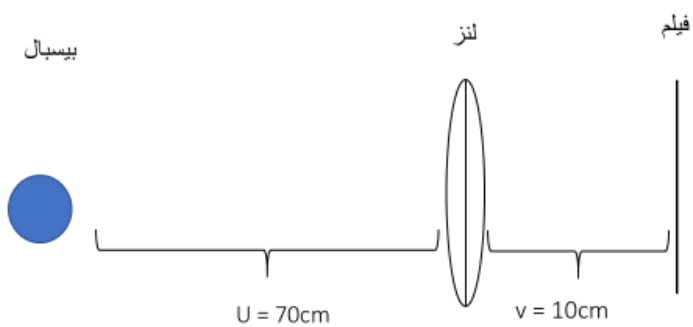
سوال 1) الف) در rolling shutter ای که داریم شاتر از بالا به پایین میرود بنابراین هنگام با حرکت جسم به سمت چپ تصویر به شکل زیر ایجاد میشود. دلیل تشکیل این تویر این است که شاتر که به سمت پایین حرکت میکند و هر سطر را ثبت میکند، مستطیل مقداری به سمت چپ حرکت کرده و آرایه ی شاتر در ردیف های بعدی به سمت چپ متمایل تر مستطیل را ثبت میکنند البته این در صورتی است که مستطیل با سرعت تقریبا ثابتی حرکت کند.



ب) در global shutter شاتر به صورت آرایه ای است یعنی نور کل تصویر را ثبت میکند. که این باعث میشود با حرکت تصویر به سمت راست تصویر کشیده شود و تار شود. چون در طول حرکت مستطیل نور های موجود را ثبت میکنیم و با هم دیگر جمع میشوند. البته در تصویر زیر صرفا کشیدگی نشان داده شده و در نظر گرفته ایم خطوط مشکی وجود ندارد چون پیچیدگی رنگ هم ایجاد میشود.

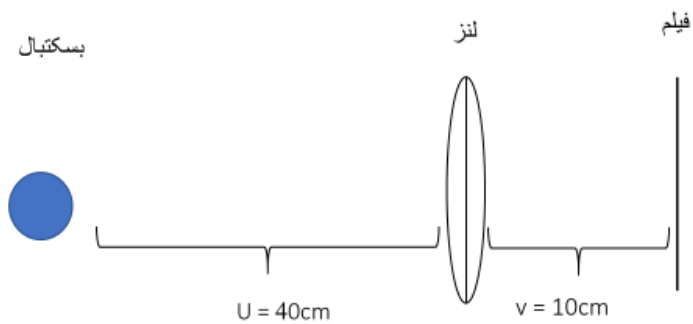


(2 الف)



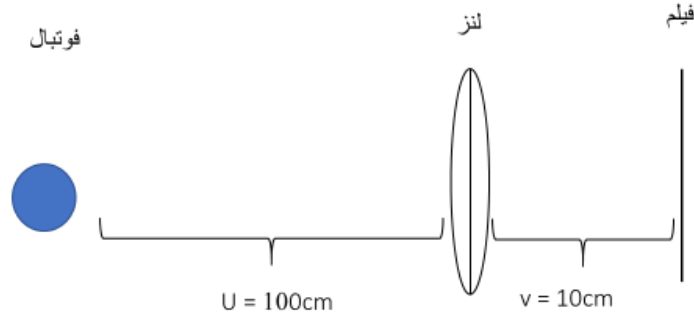
با توجه به فرمول $1/f = 1/u + 1/v$ داریم $f = 8.75$

قسمت اول :



با توجه به فرمول بالا داریم $f = 8$ پس باید فاصله کانونی را 0.75 کم کنیم..

قسمت دوم :

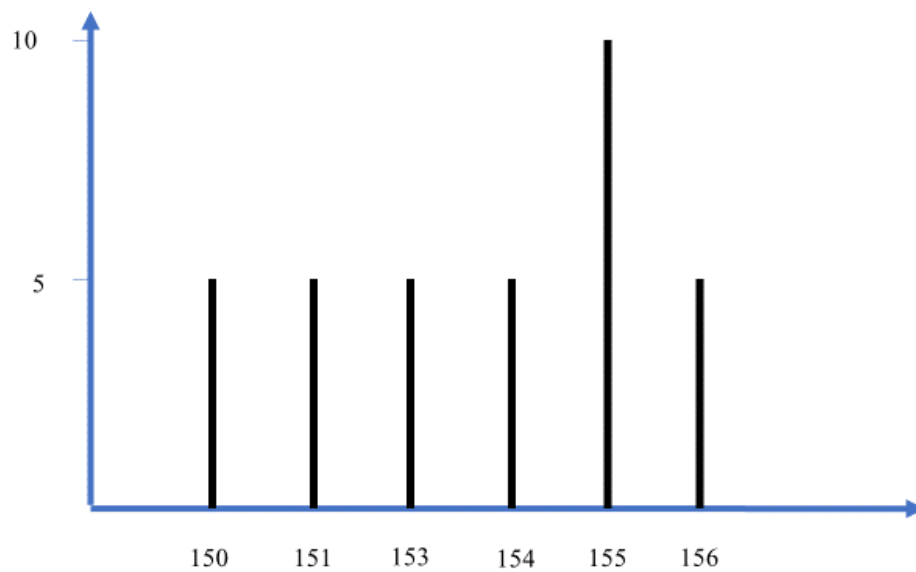


با توجه به فرمول بالا داریم $f = 9.09$ پس باید فاصله کانونی را 0.34 افزایش دهیم .

ب) با افزایش اندازه دریچه فاصله‌ای که عکس‌های واضح تولید می‌کند، کوچکتر می‌شود و عمق عکس کمتر می‌شود زیرا میزان نور وارد شده بیشتر است. و برعکس با کوچکتر شدن دیافراگم لنز، بازه‌ی فاصله‌ای که عکس باکیفیت تولید می‌کند، بزرگتر خواهد شد و عمق عکس بیشتر می‌شود زیرا میزان نور وارد شده کمتر است.

(3)

هیستوگرام اولیه



برای کشش با توجه به فرمول زیر برای هر یک از رنگ ها محاسبات را انجام می‌دهیم.

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

$$G(150) = (150 - 150) / (156 - 150) * 255 = 0$$

$$G(151) = ((151 - 150) / (156 - 150)) * 255 = 43$$

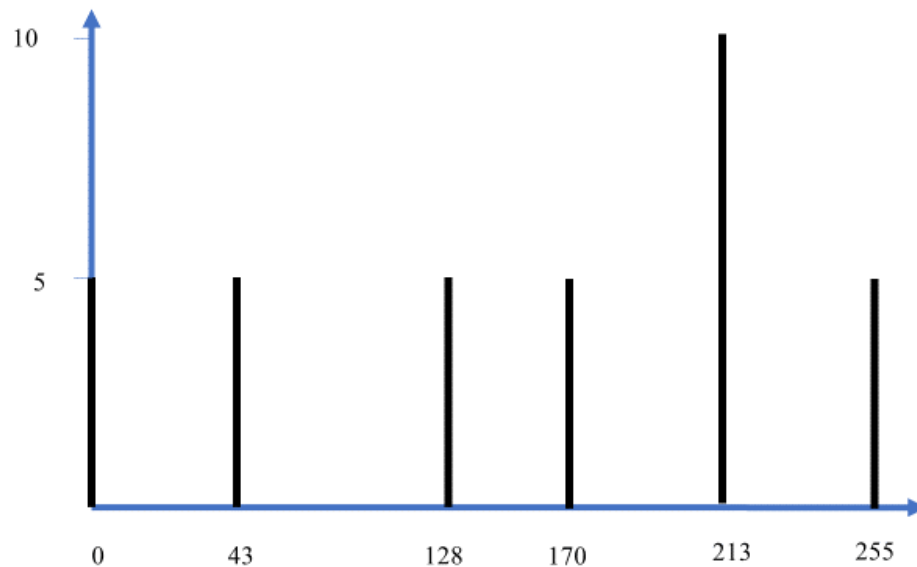
$$G(153) = ((153 - 150) / (156 - 150)) * 255 = 128$$

$$G(154) = ((154 - 150) / (156 - 150)) * 255 = 170$$

$$G(155) = ((155 - 150) / (156 - 150)) * 255 = 213$$

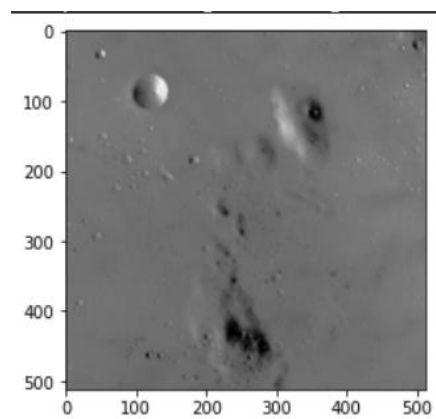
$$G(156) = ((156 - 150) / (156 - 150)) * 255 = 255$$

هیستوگرام زیر بدست می آید.

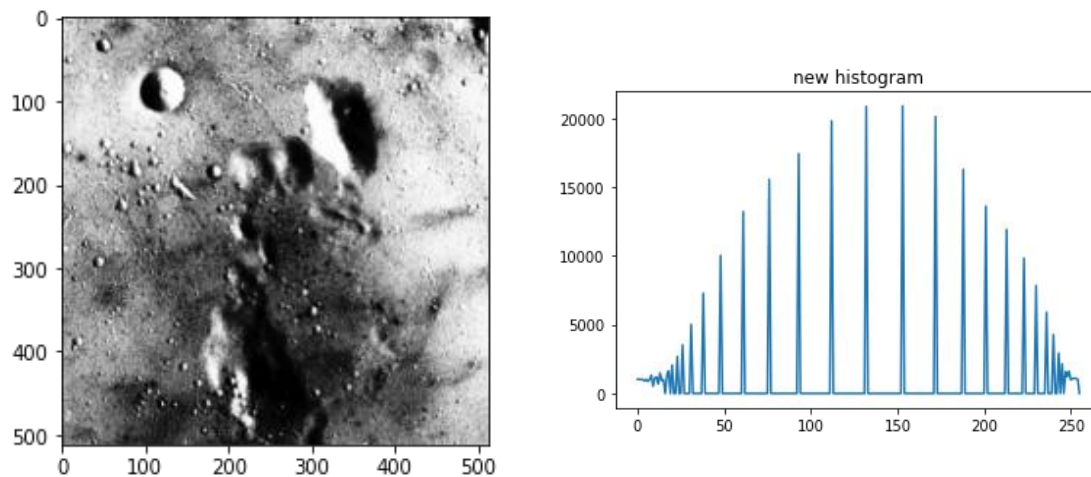


ب) در این قسمت پس از تعریف آرایه چون قسمت $(MAX - MIN) / (Fmax - Fmin)$ در فرمول بالا ثابت است آن را بدست آورده و ضربدر آرایه و از مقدار مینیمم آرایه کم میکنیم که دقیقاً همان فرمول بالاست.

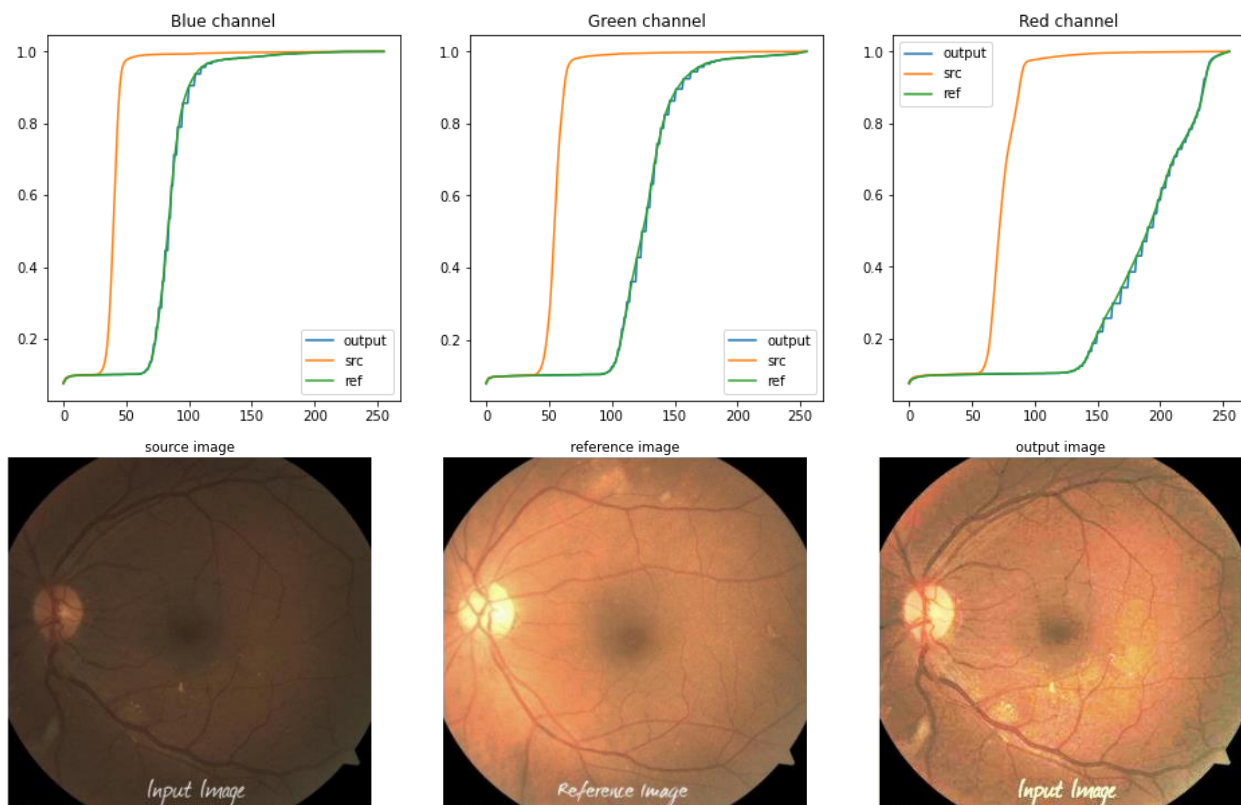
ج) همان طور که در تصویر زیر میبینیم خروجی تغییر چندانی نکرده زیرا ما در هیستوگرام اولیه میبینیم که داده های پرت زیاد داریم یعنی مثلاً ممکن است تعداد بسیار کمی از پیکسل ها 0 تعدادی 255 باشند در این صورت کشش هستوگرام تغییری ایجاد نمی کند. تصویر نهایی :



د) در این قسمت از equalization استفاده میکنیم. به این صورت که ابتدا با استفاده از `cumsum` مقدار `cdf` هیستوگرام را که در تابع `calc_hist` بدست آوردیم، محاسبه میکنیم. بعد آن را با تقسیم کردن بر مجموع همه ی مقادیر نرمال میکنیم. این مقدار بین 0 و 1 و یا بیشتر از 1 است پس در یک 255 ضرب می کنیم تا اعداد بین 0 تا 255 بشوند. سپس با استفاده از `interp()` این `cdf` را با تصویر اصلی میچ میکنیم یعنی میبینیم هر کدام از پیکسل ها چه مقداری در `cdf` پیدا کرده اند. و بعد چون ممکن است اعداد از 255 بزرگ تر شوند با تابع `clip()` این اعداد را تبدیل به 255 می کنیم. خروجی:



4الف) در این قسمت با تعریف تابع `calc_hist` و `calc_cdf` که از توابع آماده کتابخانه `numpy` برای آن ها استفاده میکنیم. بعد در `hist_matching` برای هر کانال اول `cdf` عکس مورد نظر بعد `cdf` عکس سورس را بدست می آوریم. سپس با استفاده از تابع `interp` مقدار هر عدد در `cdf` عکس مورد نظر را در `cdf` سورس پیدا کرده و آن را در لیست به نام `table` قرار میدهم. بعد هم در عکس خروجی به ازای هر پیکسل مقدار میچ شده در `table` را قرار میدهم. خروجی:



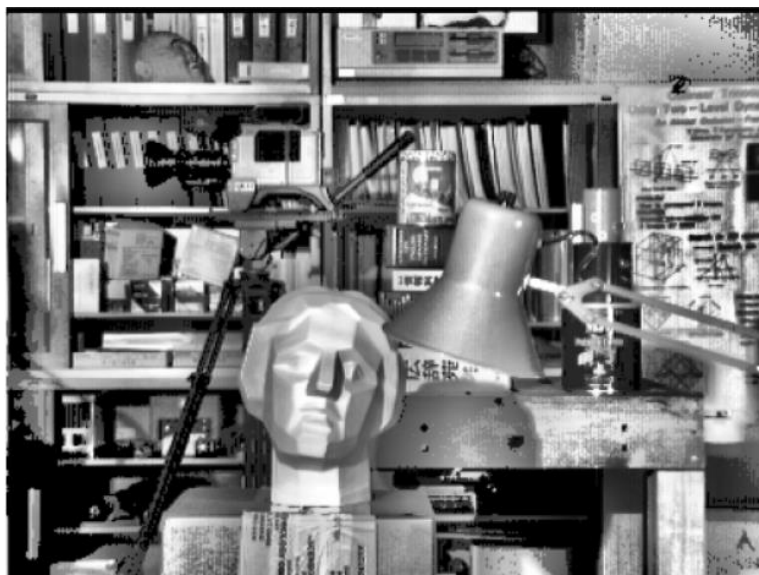
(ب) از این روش زمانی که می خواهیم سطح کنتراست گروهی از تصاویر را یکسان کنیم استفاده میکنیم.

5(الف) در قسمت های پشتی و تیره تصویر بهبود یافته چون جزییات تصویر دیده نمیشود البته این در قسمت روشن تصویر هم همینطور هست اما در قسمت هایی مثل مجسمه کنتراست تصویری پایین آمده چون خیلی روشن شده و جزییات تصویر دیده نمیشود. اما به صورت کلی کنتراست تصویر بهتر شده چون اکثر صفحه تیره است.

(ب) در این روش هر ناحیه به صورت جداگانه تغییر پیدا میکند پیکسل هایشان و در خروجی هم میتوان دید که تصویر کاشی کاشی شده. مثلاً دو پیکسل کنار هم چون تابع های متفاوتی روی آن ها صدا شده مقدار رنگ آن ها متفاوت است. خروجی:



(ج) در این قسمت ابتدا اطراف عکس را با reflect برادر میگذاریم تا نقاط کنار هم را بتوان به خوبی کنتراستشان را بالا برد. بعد با استفاده از تابع equalizeHist مقدار را برای نقاط اطراف grid بدست می آوریم و در خروجی قرار میدهیم. در این روش قسمت هایی که پیکسل ها مثلاً همه روشن یا همه تیره هستند عکس نویزی میشود و در واقع این روش نویز ها را تقویت می کند.



د) در این قسمت برای حد گذاری روی هیستوگراممان از clip استفاده میکنیم و بقیه ی موارد را در قسمت های بالا تر توضیح داده ایم. این کار باعث متعادل سازی هیستوگرام می شود و نویز هارا از بین میبرد. در رابطه با سایز فیلتر هم انتخاب این سایز بسیار مهم است برای قسمت های نویزی. یعنی ممکن است افزایش این سایز فیلتر در کنار بالا بردن کنتراست باعث ایجاد نویز شود. برای مثال اگر مقدار آن 255 باشد باعث میشود که equalization کار خودش را انجام دهد و در واقع هیستوگرام را پخش نمیکند. و با 1 بودن آن، تصویر اصلی را خواهیم داشت . خروجی:

