

به نام خدا

تمرین سری چهارم پردازش تصویر

نام استاد : دکتر آذرنوش

امیرحسین شریفی صدر

9733044

سوال 1 : بخش آ : ماتریسی قطری با ابعاد 13×13 به طوری که

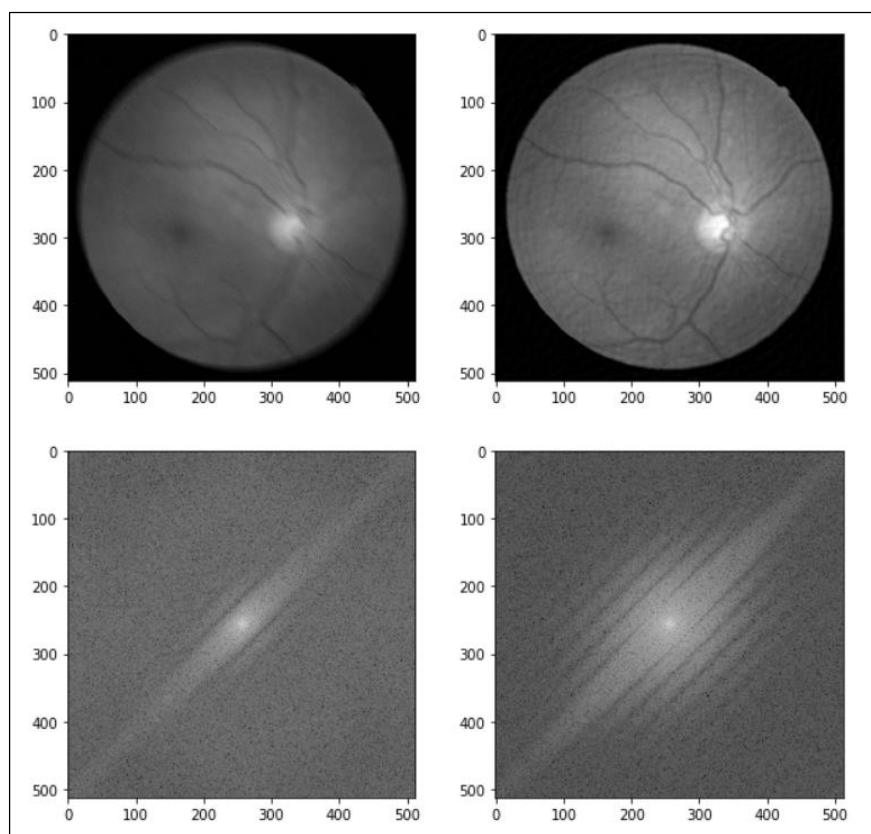
جمع مقادیر آن 1 شود که در همان ماتریسیت که تصویر اصلی به وسیله آن دچار degradation بلور قرار گرفته . و در ادامه نیز از این ماتریس در

restore کردن تصویر استفاده میشود . همان تابع تبدیل ماست .

Degradation
function
 H

$\frac{0.0769}{2308}$	0	0	0	0	0	0	0	0	0	0	0	0
0	$\frac{0.0769}{2308}$	0	0	0	0	0	0	0	0	0	0	0
0	0	$\frac{0.0769}{2308}$	0	0	0	0	0	0	0	0	0	0
0	0	0	$\frac{0.0769}{2308}$	0	0	0	0	0	0	0	0	0
0	0	0	0	$\frac{0.0769}{2308}$	0	0	0	0	0	0	0	0
0	0	0	0	0	$\frac{0.0769}{2308}$	0	0	0	0	0	0	0
0	0	0	0	0	0	$\frac{0.0769}{2308}$	0	0	0	0	0	0
0	0	0	0	0	0	0	$\frac{0.0769}{2308}$	0	0	0	0	0
0	0	0	0	0	0	0	0	$\frac{0.0769}{2308}$	0	0	0	0
0	0	0	0	0	0	0	0	0	$\frac{0.0769}{2308}$	0	0	0
0	0	0	0	0	0	0	0	0	0	$\frac{0.0769}{2308}$	0	0
0	0	0	0	0	0	0	0	0	0	0	$\frac{0.0769}{2308}$	0
0	0	0	0	0	0	0	0	0	0	0	0	$\frac{0.0769}{2308}$

بخش ب : balance انتخاب شده : 0.06 با آزمون خطا به دست آماده چون مقادیر بیشتر یا کمتر از این خیلی به بهبود تصویر کمکی نمیکردند . حال برای دادن تصویر motionblurred به تابع wiener ابتدا آن را داخل تابع normal قرار داده تا intensity هایش به بازی -1 تا 1 مپ شوند و سپس بعد از اعمال تابع wiener توسط تابع n_range آن را به بازه 0 تا 255 و تایپ اینت برمیگردانیم تا نمایش دهیم.



بخش ج :

تصویر بالا سمت چپ همان تصویر retina است که توسط
 motionblur دچار degradation شده است و سمت راست
 تصویر restore شده پس از اعمال تابع wiener روی آن است و در
 پایین نیز طیف فرکانس های هر تصویر قابل مشاهده است

$$y = Hx + n$$

where n is noise, H the PSF and x the unknown original image, the Wiener filter is

$$\hat{x} = F^{\dagger}(|\Lambda_H|^2 + \lambda|\Lambda_D|^2)\Lambda_H^{\dagger}Fy$$

بخش د : در این فرمول

که y همان تصویر دچار degrade و تحت noise قرار گرفته شده
 است و فرمول پایین روش به دست آوردن تصویر restore شده از
 فرمول wiener است که balance در تابع wiener پایداری همان λ
 استفاده شده در فرمول است که نقشش متعادل کردن داده های بیست
 که تمایل به افزایش فرکانس های بالا و همچنین نیز ها دارند پس در
 این تصویر چون بیشتر نویز و degradation در بخش میانی و
 روشن تصویر که فرکانس بالاتری دارند وجود دارد پس باید لاندایی
 استفاده شود که این مقدار کاهش پیدا کند و متعادل شود و اگر کمتر
 از میزانی هم باشد دیگر خیلی کم میشود پس من آن 0.06 در
 نظر گرفتم .

در فضای فرکانسی با توجه به فرمول گفته شده در کلاس :

$$\begin{aligned}\hat{F}(u, v) &= \left[\frac{H^*(u, v)S_f(u, v)}{S_f(u, v)|H(u, v)|^2 + S_\eta(u, v)} \right] G(u, v) \\ &= \left[\frac{H^*(u, v)}{|H(u, v)|^2 + S_\eta(u, v)/S_f(u, v)} \right] G(u, v) \\ &= \left[\frac{1}{H(u, v)} \frac{|H(u, v)|^2}{|H(u, v)|^2 + S_\eta(u, v)/S_f(u, v)} \right] G(u, v)\end{aligned}$$

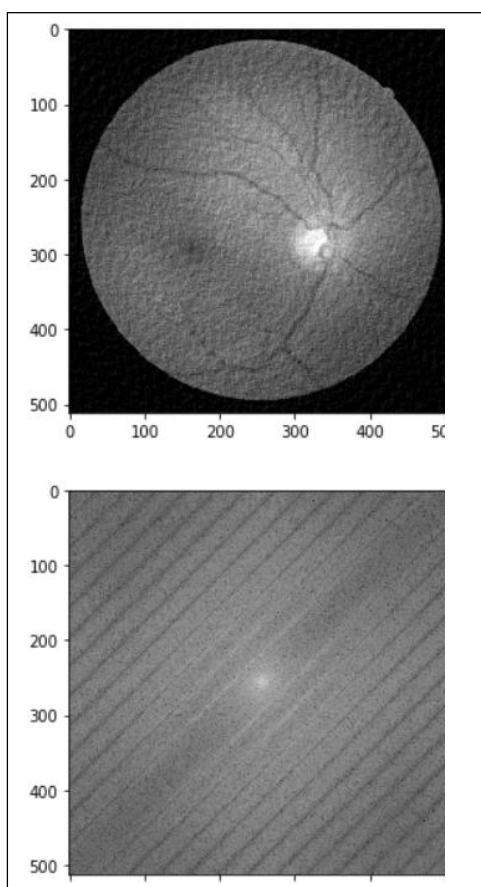
$S_\eta(u, v)$ = Power spectrum of the noise

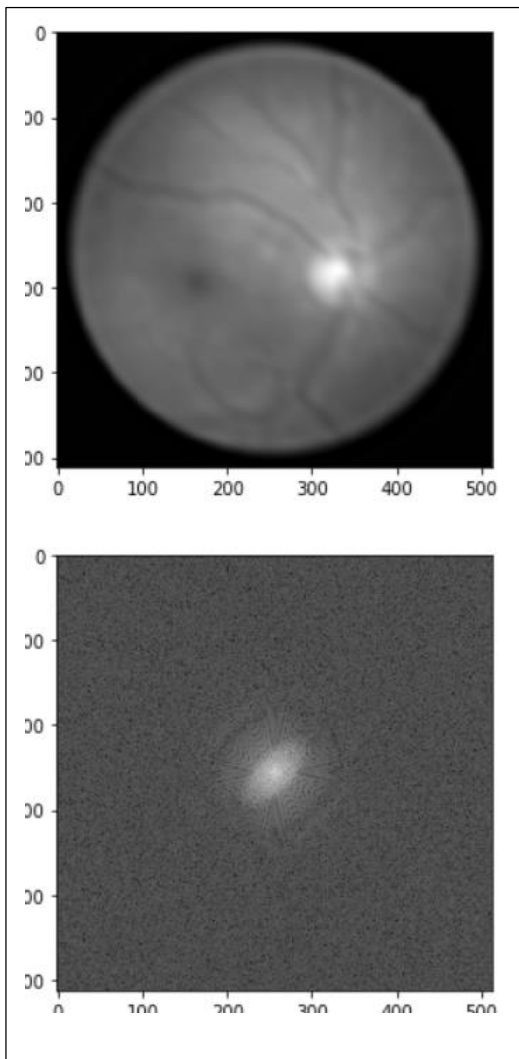
$S_f(u, v)$ = Power spectrum of the undegraded image

balance استفاده شده در تابع wiener احتمالا همین بخش

مشخص شده در فرمول بالا است که چون ما اطلاعاتی از تابع
فرکانسی نویز نداریم باید آن را با آزمون و خطا به دست آوریم
در مقدار بسیار پایین تصویر و طیف فرکانسی تصویر ما به شکل
زیر خواهد بود : در این تصویر

Balance = 0.0006 است





و اگر مقدار balance بالا باشد

مثلا 100 داریم :

میبینیم که تصویر بلور تر شده

در حالت مکانی نیز balance معکوس فرمول مقابل است :

$$SNR = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \hat{f}(x, y)^2}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f(x, y) - \hat{f}(x, y)]^2}$$

که در آن نویز از تفریق

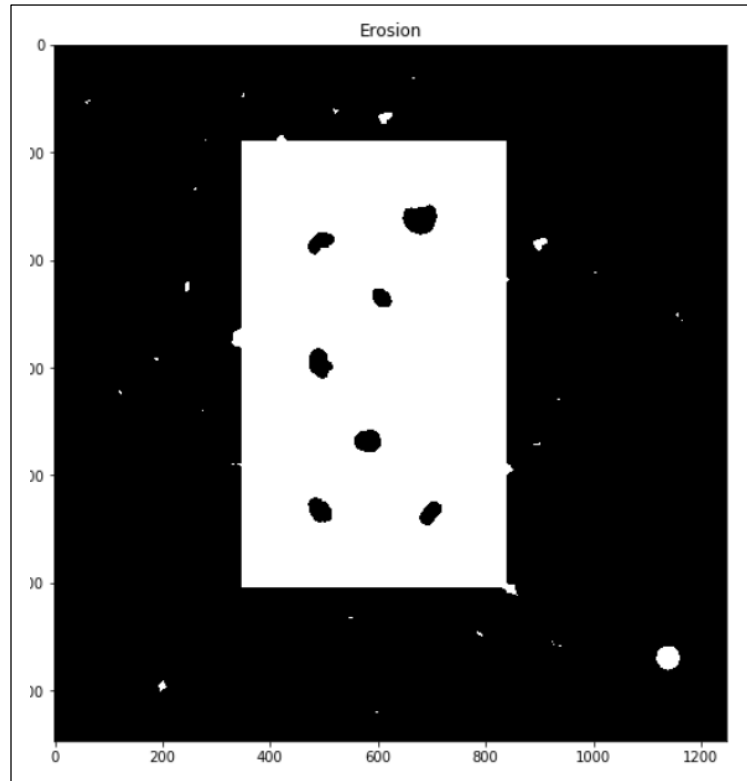
تصویر اصلی از تصویر

Restore شده به دست

می آید .

سوال 2 : بخش الف :

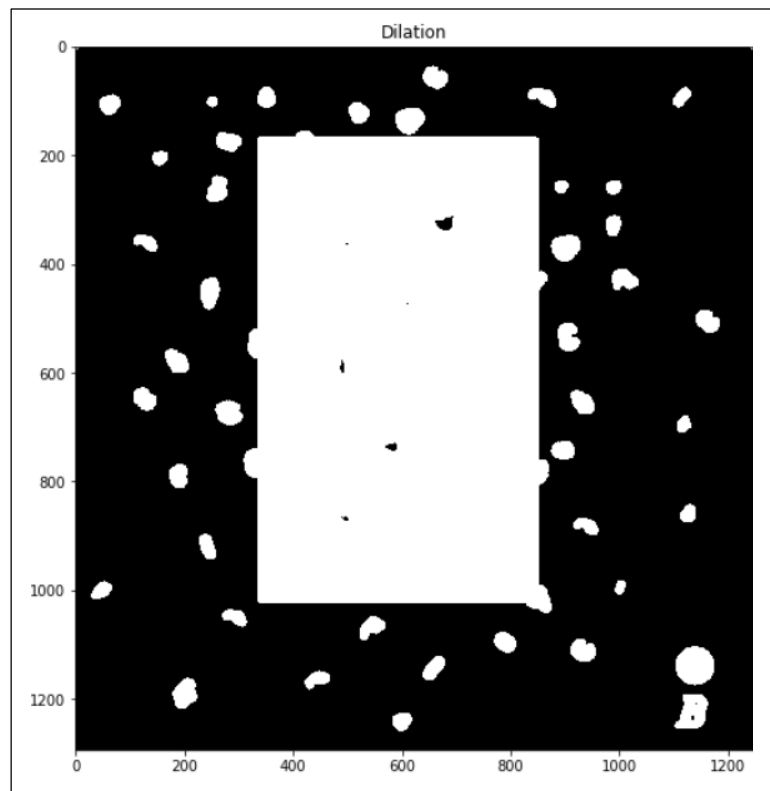
Erosion: $A \ominus B = \{z | (B)_z \subseteq A\}$



Erosion یا خوردگی بدین صورت است که ما پنجره ای را رو تصویر خود حرکت می‌دهیم و هرگاه پنجره تمام محدوده پارامتر مورد نظر در تصویر را پوشش بدهد (زیر مجموعه آن باشد) پیکسلی از تصویر که روی مرکز آن پنجره قرار دارد حفظ می‌شود در غیر این صورت از بین می‌رود. در این شکل مشاهده می‌شود که تصویر حاصل از erosion پیکسل‌های سفید پخش در آن

کوچکتر و دچار خوردگی شده اند و همچنین مستطیل سفید
وسط نیز لبه ها خورده شده و کوچکتر شده اند و هم لبه ها نویز
های سیاه نیز از بین رفته و intensity آنها صفر شده پس
کمی بزرگتر شده اند .

$$\text{Dilation: } A \oplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\}$$



Dilation یا گستردگی بدین صورت است که پنجره ای را روی
تصویر حرکت می‌دهیم و هرگاه آن پنجره کوچکترین اشتراکی با

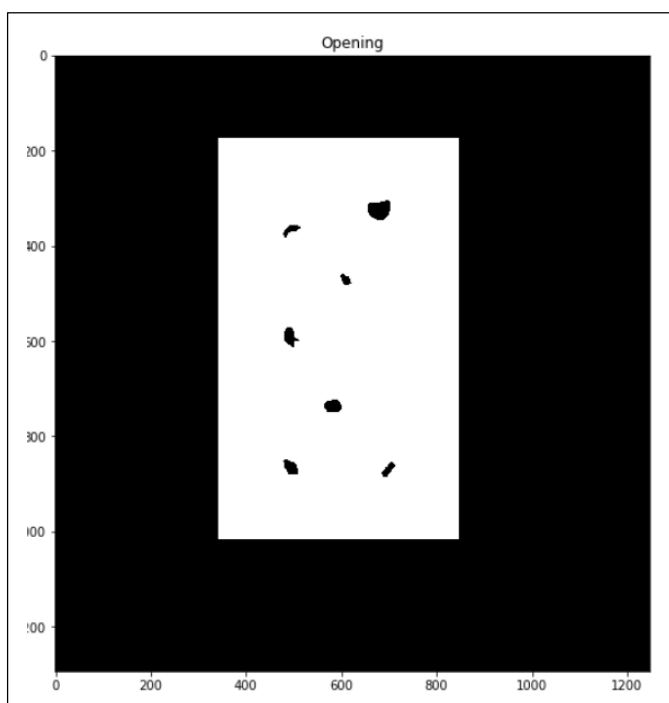
تصویر داشت مرکز آن پنجره را نگه میداریم یا میتوان گفت به تصویر اضافه میکنیم و این سبب گسترده شدن تصویر ما میشود . در شکل بالا مشاهده میشود که بخش هایی از تصویر که سفید هستند گسترده تر شده اند چون به سبب این فیلتر به اطراف آن ها پیکسل هایی اضافه شده (پیکسل های اضافه شده سفید هستند و در ابروشن وقتی پیکسلی حذف میشد به رنگ سیاه درمیآمد) با 255 intensity و همچنین مستطیل سفید وسط نیز کمی گسترده تر شده و نویز های سیاه رنگ وسط تصویر نیز به واسطه گسترده شدن پیکسل های سفید اطرافشان دچار خوردگی شده اند .

پس در نهایت میبینیم که erosion نویز های خارج از مستطیل سفید را از بین برده و dilation نویز های داخل آنرا . بخش ب : opening در اصل همان erosion و dilation پشت هم است . به صورت رو به رو :

Opening: $A \circ B = (A \ominus B) \oplus B$

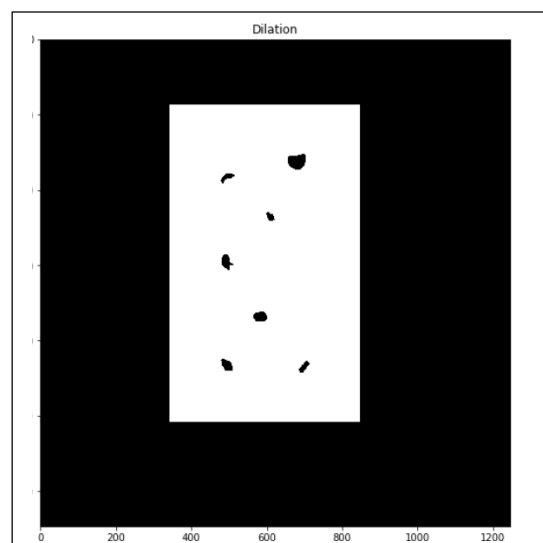
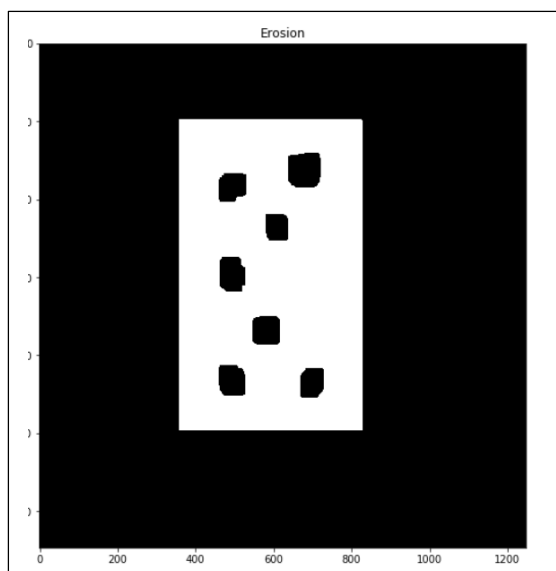
$$A \circ B = \bigcup \{ (B)_z \mid (B)_z \subseteq A \}$$

که پس آزمون و خطا ما برای پنجره مستطیل شکل به جهت اعمال و حرکت دادن آن رو تصویر به حداقل ابعاد (36,47) میرسیم . در رو به رو شکل تحت opening قرار گرفته را



میبینیم :

اگر بخواهیم شیوه این اعمال این تابع را به تفکیک یعنی ابتدا erosion و سپس dilation ببینیم به صورت زیر است :



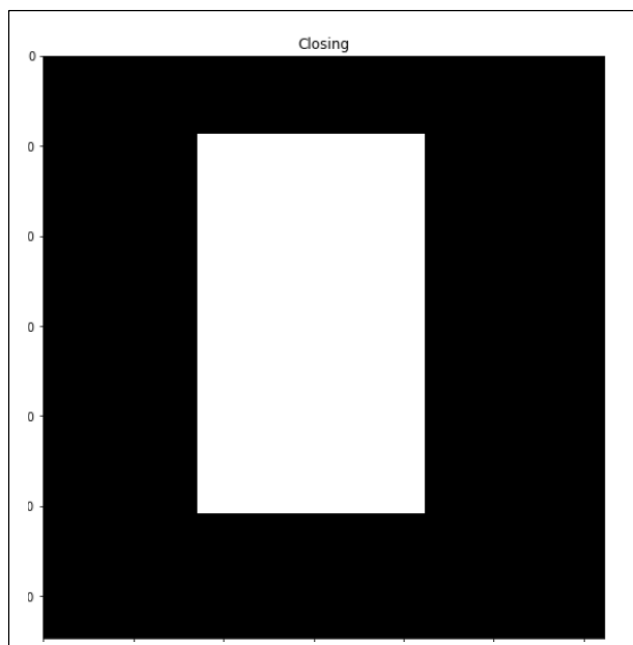
که مشاهده میشود تصویر پس از این دو عمل همان تصویر پس از opening است .

بخش ج : closing برعکس opening توالی انجام dilation و erosion است یعنی ابتدا تصویر دچار دایلیشن و سپس دچار ایزوژن میشود . به صورت رو به رو :

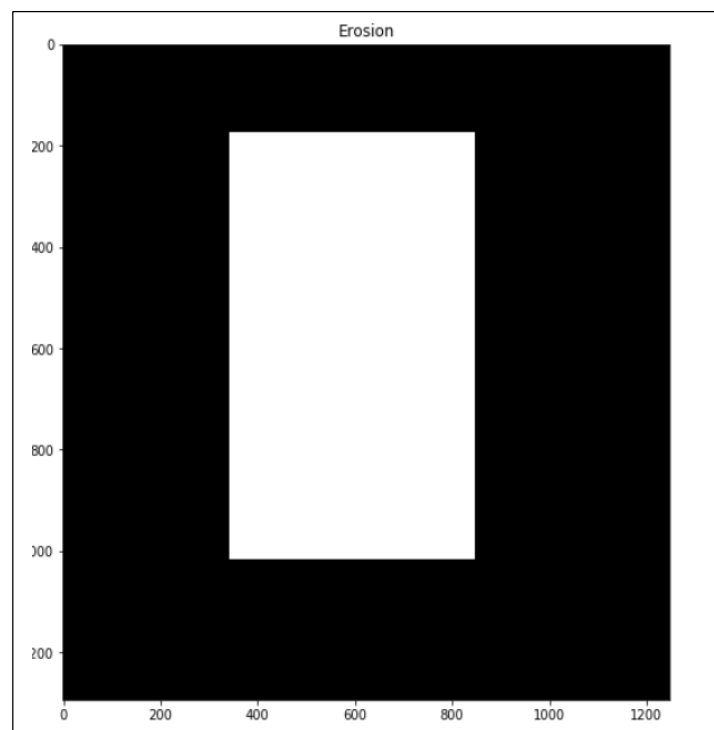
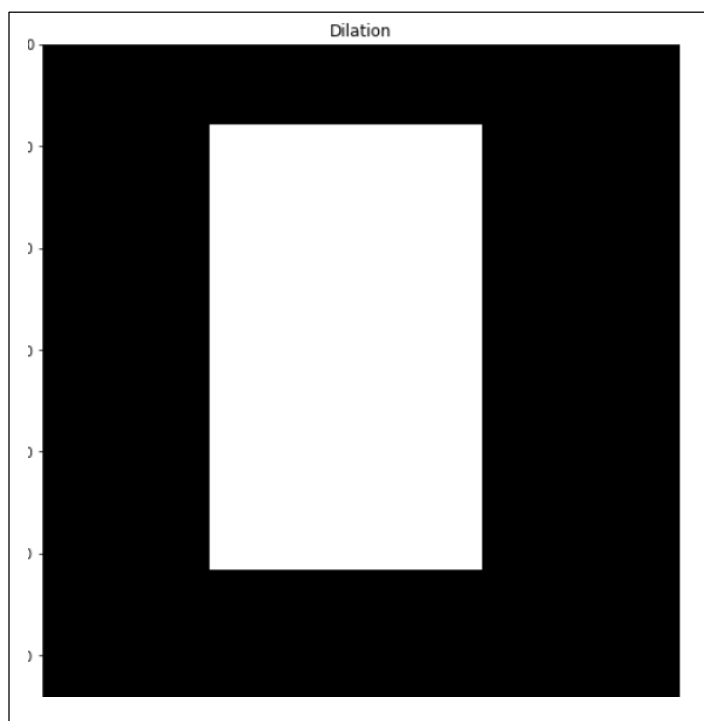
$$\text{Closing: } A \bullet B = (A \oplus B) \ominus B$$

که در این نیز پس از اعمال آزمون و خطا برای پنجره استفاده شده در نهایت به ابعاد (29,32) میرسیم.

و تصویر پس از closing اعمال شده بر روی تصویری که دچار opening شده بود به صورت زیر است که میبینیم تمامی نویز های سفید و سیاه آن از بین رفته است :



حال می‌خواهیم ببینیم که شیوه کار closing به تفکیک erosion و dilation به چه شکل است :



شاید در نگاه اول هیچ تفاوتی بین دو تصویر بالا نباشد ولی شیوه کار به اینگونه است که ابتدا dilation نویزهای سیاه میام تصویر را از بین برده و همچنین باعث بزرگتر شدن مستطیل میانی شده و سپس erosion مستطیل میانی را به سائز قبلی خود برگردانده .

سوال 3 :

بخش آ : از ما خواسته شده تا نویز تصویر را حذف کنیم ، برای اینکار ابتدا فیلتر opening و سپس فیلتر closing را روی تصویر تحت فیلتر opening قرار گرفته اجرا میکنیم ، تصویر اصلی و تصویر denoise شده را مشاهده میکنید :



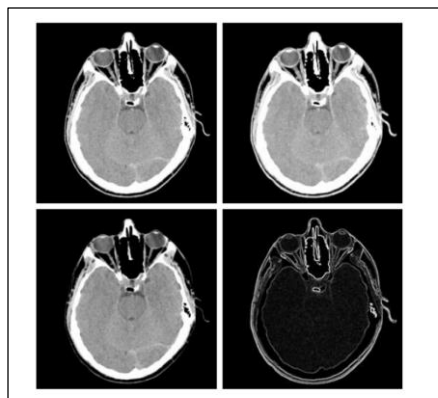
با کرنلی به شکل مربع با ابعاد 3 در 3 این کار را انجام دادیم .

بخش ب : گرادیان در فضا موفولوژی تفریق یا اختلاف تصویر

تحت dilation و تصویر تحت erosion است . $g = (f \oplus b) - (f \ominus b)$

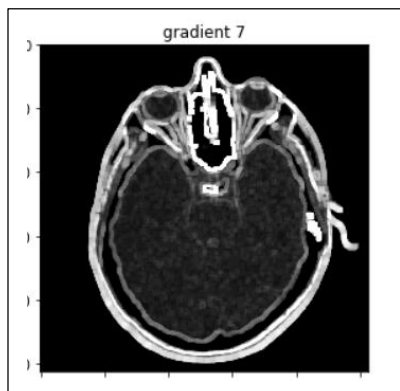
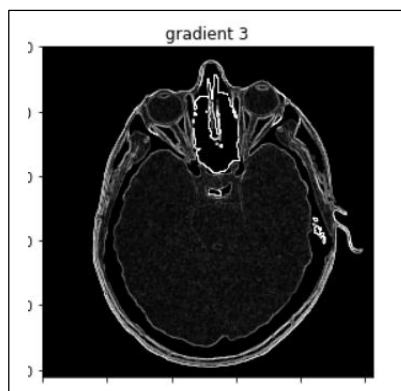
و مراحل این اتفاق نیز در تصویر مقابل که از اسلاید استاد گرفته

شده مشخص است :



حال ما توسط تابع آماده گرادیان در فضای مورفولوژی یک بار با اعمال پنجره ای 3×3 و یک بار با اعمال پنجره ای 7×7 نتایج

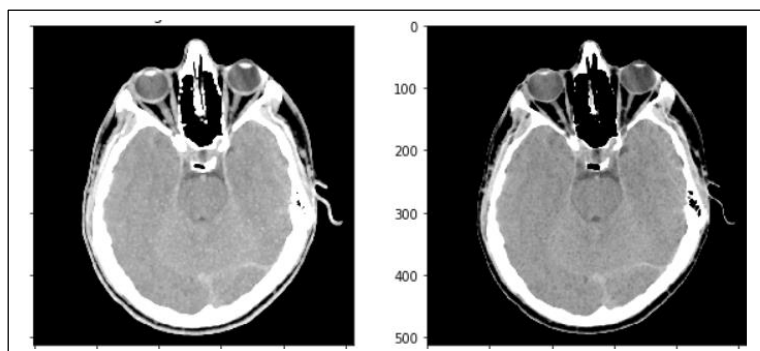
را مشاهده میکنیم :



با دید مورفولوژی و اختلاف dilation و erosion این اختلاف بین دو گرادیان بالا کاملاً قابل توجیه است :

Dilation 3×3

Erosion 3×3



همانطور که دیده میشود

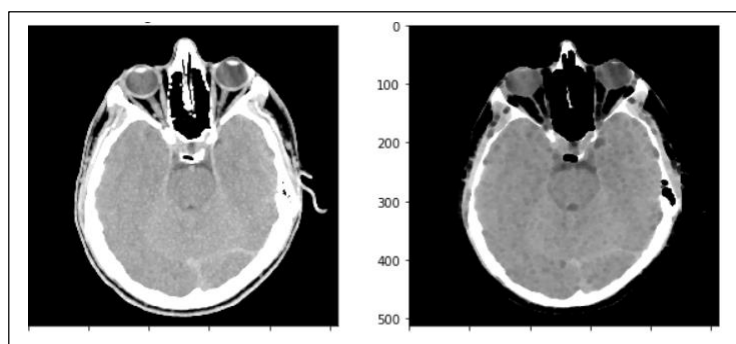
وقتی در شکل پایین تصویر

در dilation دچار

گسترده‌گی بیشتری میشود

Dilation 7×7

Erosion 7×7



و در erosion نیز دچار

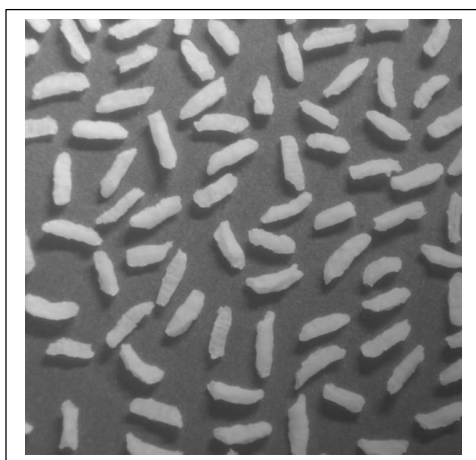
خوردگی بیستری میشود پس

اختلاف آن دو در نهایت بیشتر و لبه های سفید تر بیشتر دارد .

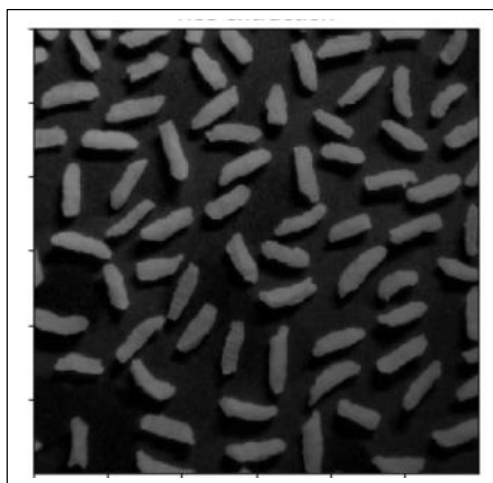
بخش ج :

از روش Tophat برای استخراج برنج استفاده میکنیم ین ورش بدین شکل است که ابتدا توسط opening تمام دانه ها برنج را از تصویر حذف کرده و نور پس زمینه را استخراج میکند و برای اینکار kernel را باید قدری بزرگ بگیریم که مطمئن باشیم تمام دانه های برنج را حذف میکند و سپس با کم کردن تصویر اصلی از نور پس زمینه استخراج شده، دانه های برنج را استخراج میکنیم . این کار را در کد توسط تابع آماده tophat انجام دادم که شکل زیر را به ما میدهد :

Original image



Tophat image with 80*80 kernel



حال برای اینکه دانه های برنج به درستی تفکیک و استخراج شوند تصویر را **threshold** میکنیم و با آزمون بهترین مقدار را برای آن میابیم که نتیجه این است که پیکسل ها 55 به پایین را سیاه کند و بقیه را سفید پس نتیجه نهایی را مشاهده میکنید :

