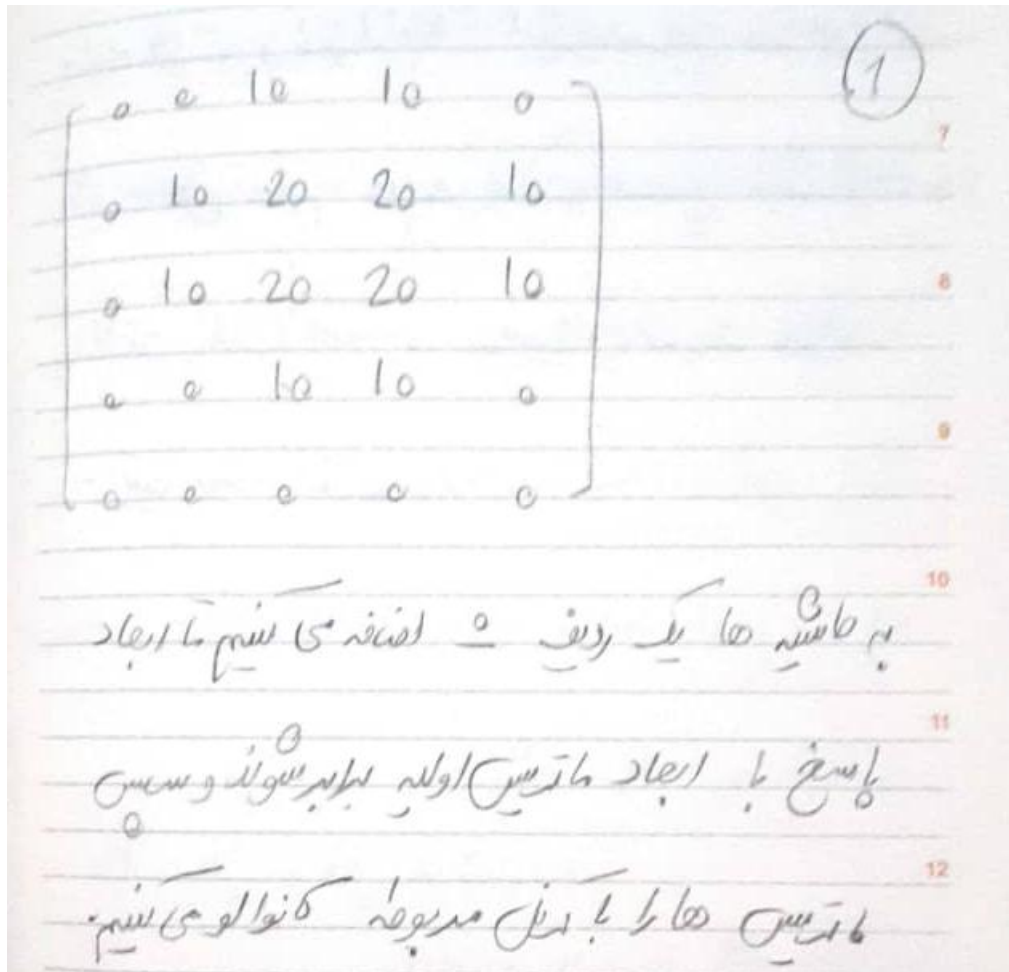


" به نام خدا "

مروارید لعل نور 9623097

گزارش تمرین سری سوم پردازش تصویر

سوال اول :



اولین فیلتر نرم کردن که استفاده شده به همراه ماتریس خروجی به این صورت هستند :

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{70}{9} & \frac{40}{3} & \frac{40}{3} & 0 \\ 0 & \frac{70}{9} & \frac{40}{3} & \frac{40}{3} & 0 \\ 0 & \frac{40}{9} & \frac{70}{9} & \frac{70}{9} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

دومین فیلتر نرم کردن :

$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{130}{16} & 15 & 15 & 0 \\ 0 & \frac{130}{16} & 15 & 15 & 0 \\ 0 & \frac{15}{4} & \frac{130}{16} & \frac{130}{16} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

همانطور که مشاهده میشود اختلاف مقدار پیکسل ها کمتر شده و اعداد به هم نزدیکتر شده اند که این باعث نرم تر شدن تصویر و کمرنگ شدن تضاد ها میشود

اولین فیلتر تیز کردن :

0	-1	0
-1	5	-1
0	-1	0

0	0	0	0	0
0	20	40	40	0
0	20	40	40	0
0	-20	20	20	0
0	0	0	0	0

دومین فیلتر تیز کردن :

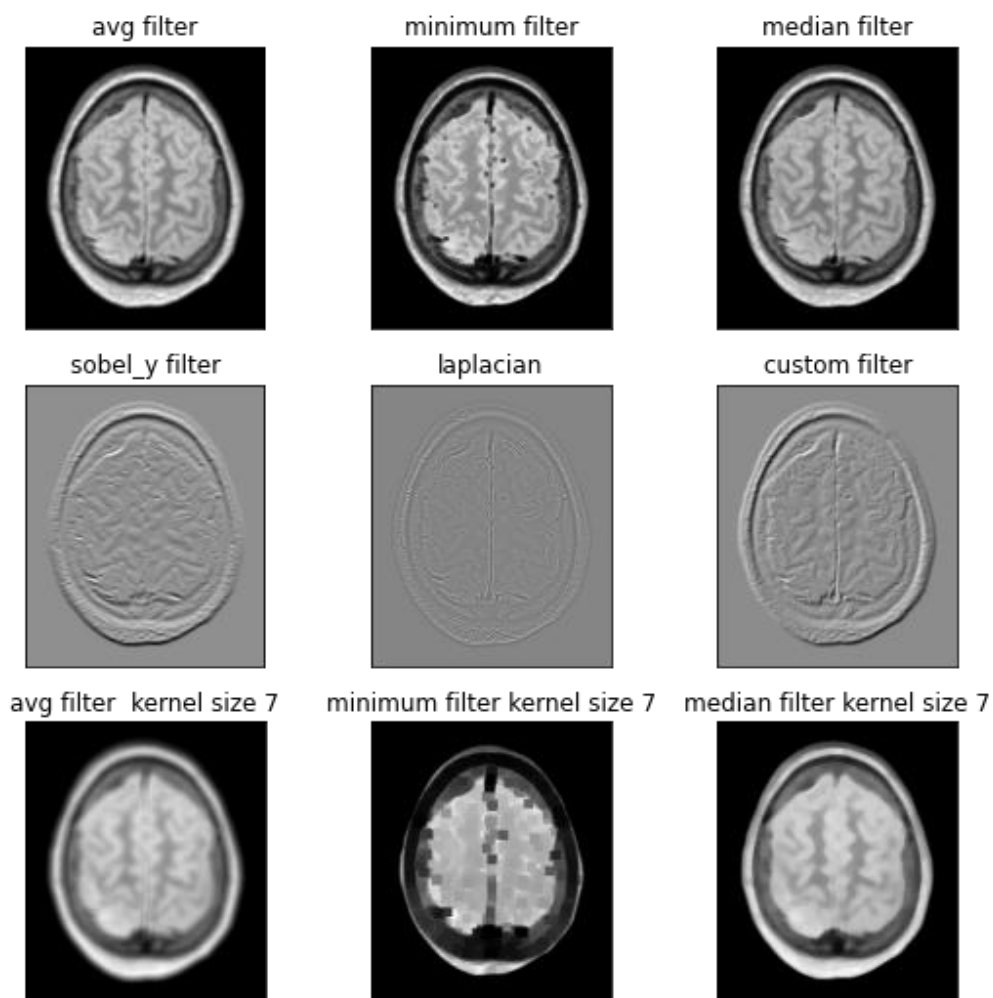
-1	-1	-1
-1	9	-1
-1	-1	-1

0	0	0	0	0
0	30	80	80	0
0	30	80	80	0
0	-40	30	30	0
0	0	0	0	0

در این نوع فیلتر ها نیز مشاهده میشود که اختلاف مقدار پیکسل های مجاور هم تشدید شده تا تضاد ها مشخص تر باشند و در واقع تصویر sharp شود .

سوال دوم :

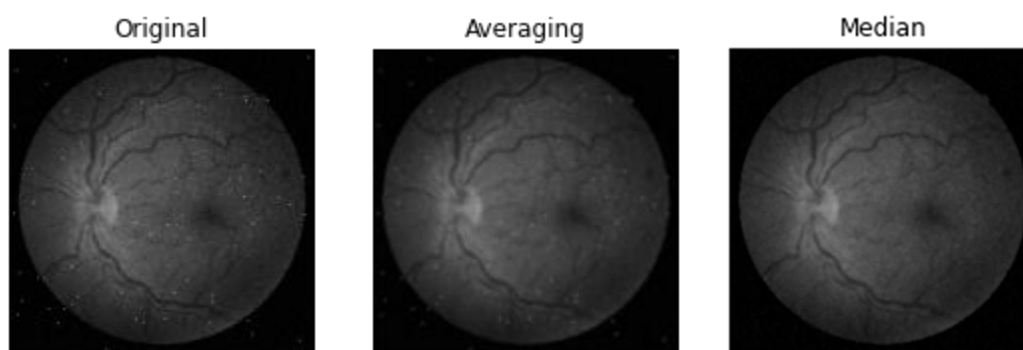
نتیجه اعمال کرنل ها به صورت گفته شده در سوال و با استفاده از تابع نوشته شده به این صورت است :



فیلتر پیشنهاد شده توسط صورت سوال را به نام **embossing filter** درون برنامه استفاده کرده ایم و با توجه به تصاویر میتوان گفت این فیلتر با قرار دادن اختلاف پیکسل های دو طرف قطر کرنل باعث افزایش کنتراست در قسمت هایی از تصویر که دارای اختلاف در دو طرف (خط مورب) است شده و در نتیجه خطوط موجود در تصویر را به صورت برجستگی هایی مشخص تر میسازد (و در واقع انگار تصویر مثبت کاری شده از تصویر اولیه میدهد)

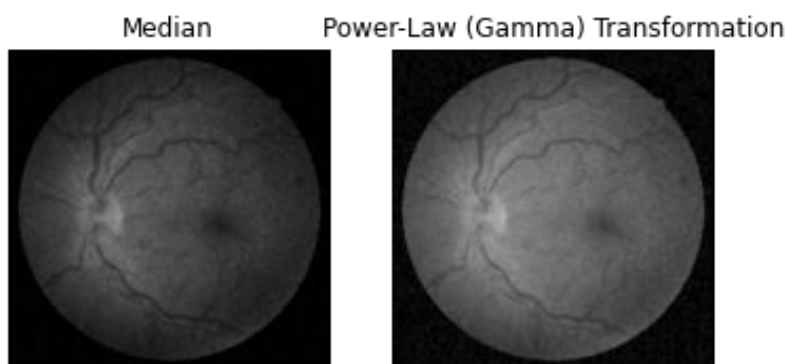
سوال سوم :

(آ)



با توجه به آنکه فلتر میانگین گیریتنها تصویر را تارتر میکند و از همان پیکسل های قبلی (هر چند با مقدار بی ربط) برای ساختن تصویر جدید استفاده میکند میتوان مشاهده کرده که تصویر هنوز نویز دارد و کامل ترمیم نشده است . اما فیلتر میانه گیری از مقادیر داده های پرت در پیکسل ها فاکتور میگیرد و آنها را حذف میکند پس مشاهده میشود که نویز ها تقریبا به طور کامل از بین رفته اند .

ب و ج)

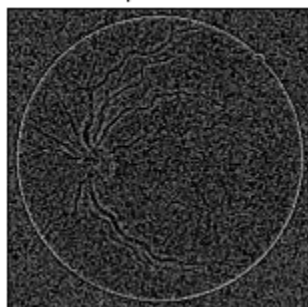


تابع تبدیل توانی را نوشتیم به طوری که ابتدا مقدار تمامی پیکسل ها را بین 0 تا 1 نرمالایز میکند ، آنها را به توان گاما میرساند و دوباره در 255 ضرب میکند و آنرا بر روی تصویر بدست آمده از فیلتر میانه گیری اعمال کردیم و همانطور که گفته شده گاما را برابر $2/3$ قرار دادیم . همانطور که مشاهده میشود در قسمت قبل تصویر خروجی فیلتر میانه را تقریبا بدون نویز در نظر گرفتیم اما با اعمال تبدیل توانی در و به توان $2/3$ رساندن پیکسل ها در واقع تفاوت بین پیکسل های کم و زیاد را بهتر میتوان مشاهده کرد و نتیجه گرفت تصویر هنوز نویز دارد و کاملا پاک نشده .

(د)

با اعمال فیلتر لاپلاسین و مشاهده تصویر تبدیل توانی شده با گامای $1/3$:

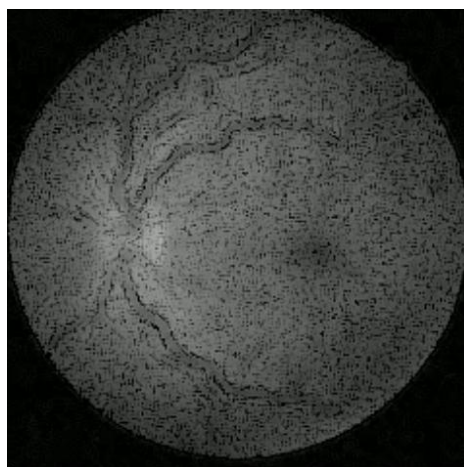
laplacian



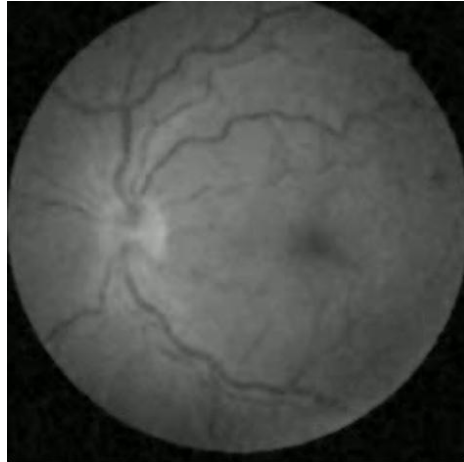
با توجه به تمیز نبودن تصویر ورودی و وجود پیکسل های نویزی درون آن و اینکه فیلتر لاپلاسین مانند یک فیلتر تشخیص لبه به کار میرود ، نویز های موجود در تصویر ورودی آشکار تر شده اند و به طور کلی تفاوت های میان پیکسل ها آشکار تر شده است .

(ه)

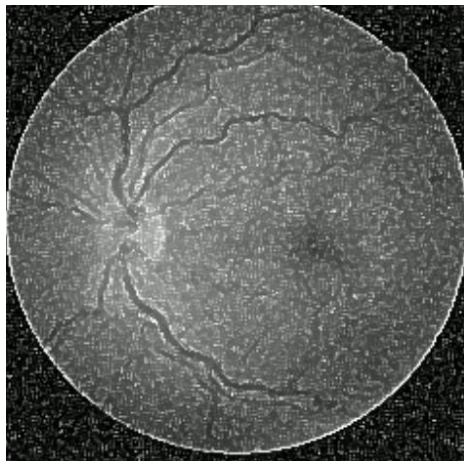
در این قسمت از ما خواسته شده یک ویدیو با استفاده از عکس هایی که طبق صورت سوال ساخته شده اند بسازیم . ویدیو ابتدا با این تصویر شروع میشود :



با این تصویر ادامه میابد :



و با این تصویر تمام میشود :



اینطور به نظر میرسد که در نیمه ی اول ویدیو چون عکس هایی که با ضرایب منفی ماسک جمع شده اند (c_i بین -2 تا 0 است) فریم های اولیه را تشکیل میدهند مثل آن است که تمامی نقاطی که قبلا لبه میدانستیم حالا کم شده و تبدیل به 0 شده اند و در میانه ویدیو (ثانیه 13 ام) ضریب ماسک 0 بوده و دقیقاً همان تصویر $image_c$ (خروجی بخش ج) را داریم اما در نیمه دوم که ضرایب مقادیر مثبت دارند (c_i بین 0 تا 2 است) لبه ها و نواحی شارپ شده (و همچنین نویز ها) شروع به روشن تر شدن (افزایش مقدار پیکسل نسبت به قبل) میکنند تا فریم آخر که تقریباً رنگ این مناطق سفید شده است .

به نظر میرسد c_i مناسب برای بهبود تصویر مقداری مثبت است زیرا هرچه این ضریب به سمت مثبت تر می رود تصویر سارپ تر شده و جزئیات مشخص ترند که میتوان در پایان ویدیو آنرا مشاهده کرد .

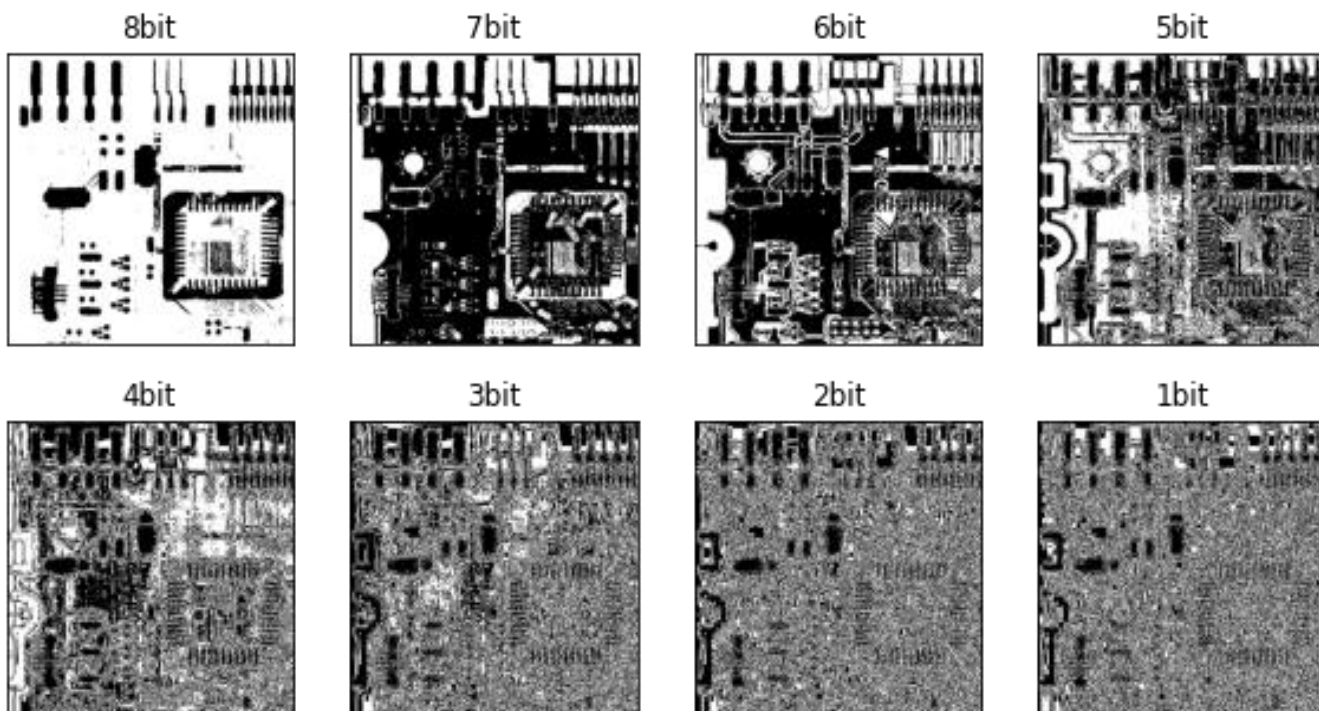
اگر از همان ابتدا به جای استفاده از مقادیر اعشاری این عملیات را روی داده های با مقدار 8 بیتی بی علامت انجام دهیم از آنجایی که مانند داده های اعشاری مقادیر کوچکتر از یک را در نظر نگرفته ایم ممکن است بسیاری از تغییرات را مشاهده نکنیم و تعداد زیادی عکس یکسان داشته باشیم که مقادیر صحیح یکسانی دارند اما در واقع مقادیر اعشاری آنها متفاوت بوده است .

سوال چهارم :

(آ)

این تابع را به این صورت پیاده کردیم که ابتدا تمامی پیکسل های تصویر را به صورت باینری در یک لیست ذخیره میکند و سپس براساس آنکه چه عددی به عنوان n را دریافت کرده است بیت n ام همه ی اعضای لیست را در قالب ماتریسی با همان ابعاد تصویر اولیه بازمیگرداند.

(ب) نتیجه اعمال تابع روی تصویر مورد نظر :



(ج) تصاویر بدست آمده از عملیات مربوط به تشخیص حرکت :

