

به نام خدا

پردازش تصویر

آزمون میانترم

امین سخایی

۹۷۳۳۰۳۶

استاد درس

دکتر حامد آذرنوش

سوال شماره ۱:

ورودی برنامه تصویر retina.jpg است که به صورت خاکستری ۸ بیتی آنرا فراخوانی می کنیم.

تصویر ورودی از نوع تک کانال (خاکستری) با ابعاد ۲۹۱۲*۲۹۱۲، نوع داده ی هر پیکسل نیز ۸بیتی بدون علامت می باشد.

بخش الف) فضای حافظه ای برای ذخیره تصویر (بدون فشرده سازی) از فرمول زیر بدست می آید:

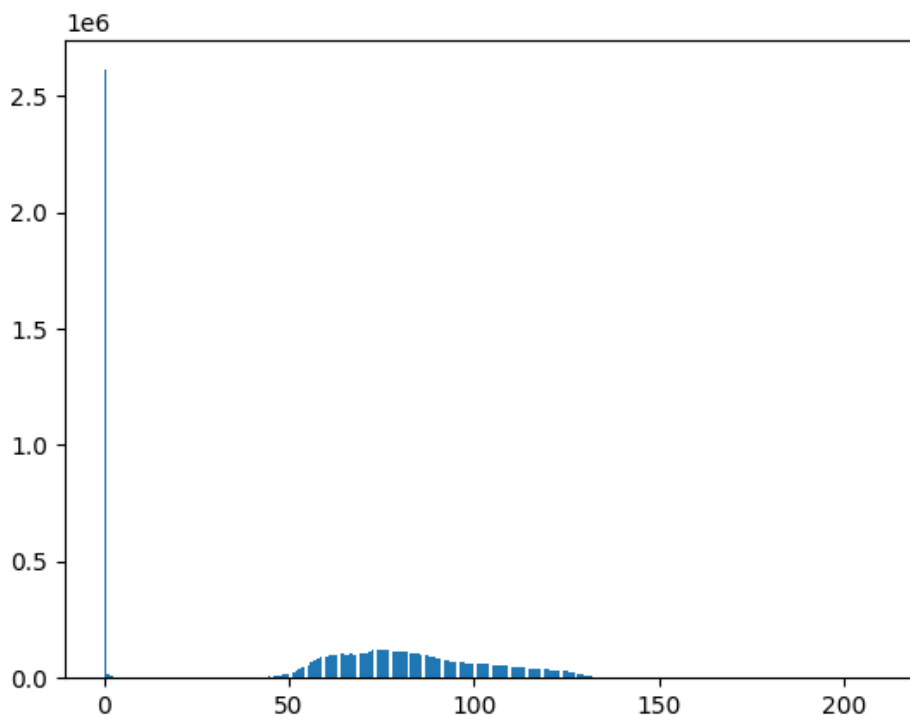
$$b = MNK$$

تعداد بیت $K =$ ابعاد تصویر $M, N =$

```
(2912, 2912)
uint8
size= 67837952
```

بخش ب) هیستوگرام تصویر نموداری است که توسط آن تعداد پیکسل های هر سطح روشنایی در تصویر ورودی مشخص می شود. نمودار هیستوگرام دارای دو مولفه افقی و عمودی است. مولفه افقی سطوح خاکستری را نشان می دهد که معمولاً بین ۰ (سیاه) و ۲۵۵ (سفید) می باشد. مولفه عمودی نیز تعداد پیکسلهای دارای سطح خاکستری مشخص در تصویر است.

کنتراست به اختلاف بین بیشترین و کمترین شدت روشنایی گفته می شود بنابراین با استفاده از هیستوگرام کنتراست تصویر را می توان تعیین کرد. (به دلیل حاشیه سیاه دور تصویر، هیستوگرام آن کنتراست بالایی را نشان می دهد).



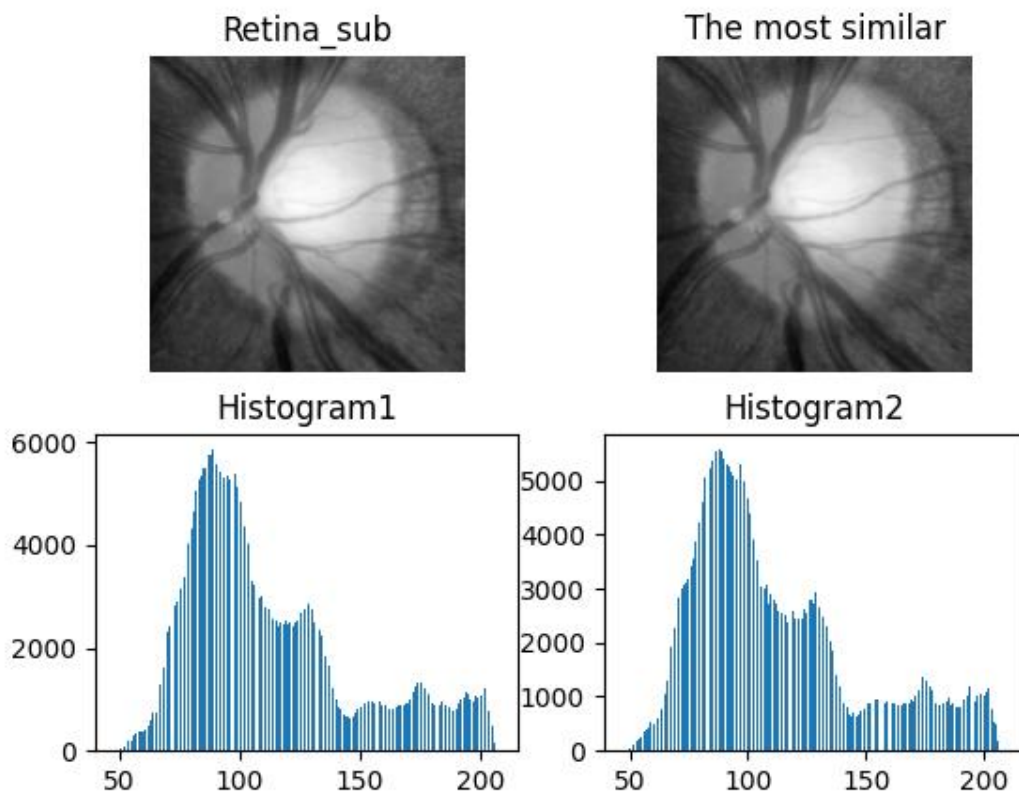
بخش ج) برای مقایسه هیستوگرام ها ابتدا معیار موردنظر را انتخاب میکنیم.

معیار مورد استفاده در این برنامه تابع Chi-Square است که به صورت زیر تعیر می شود:

$$d(H_1, H_2) = \sum_I \frac{(H_1(I) - H_2(I))^2}{H_1(I)}$$

هرچقدر d کوچکتر باشد هیستوگرام دو تصویر مشابه تر هستند.

حال یک حلقه که اندازه آن اختلاف ابعاد دو تصویر ورودی می باشد ایجاد می کنیم.در هر تکرار یک پنجره با ابعاد تصویر retina_sub به اندازه یک پیکسل(برای تسریع اجرای برنامه تعداد گام افزایش داده شده است) جلو می رود و هیستوگرام آن با تصویر retina_sub مقایسه می شود و با استفاده از یک دستور مقایسه ای مقدار مینیمم و شماره ی حلقه ی آن را ذخیره می کنیم. در آخر با استفاده از شماره حلقه پنجره ی مورد نظر را از تصویر اصلی کات کرده و آن را نمایش می دهیم.

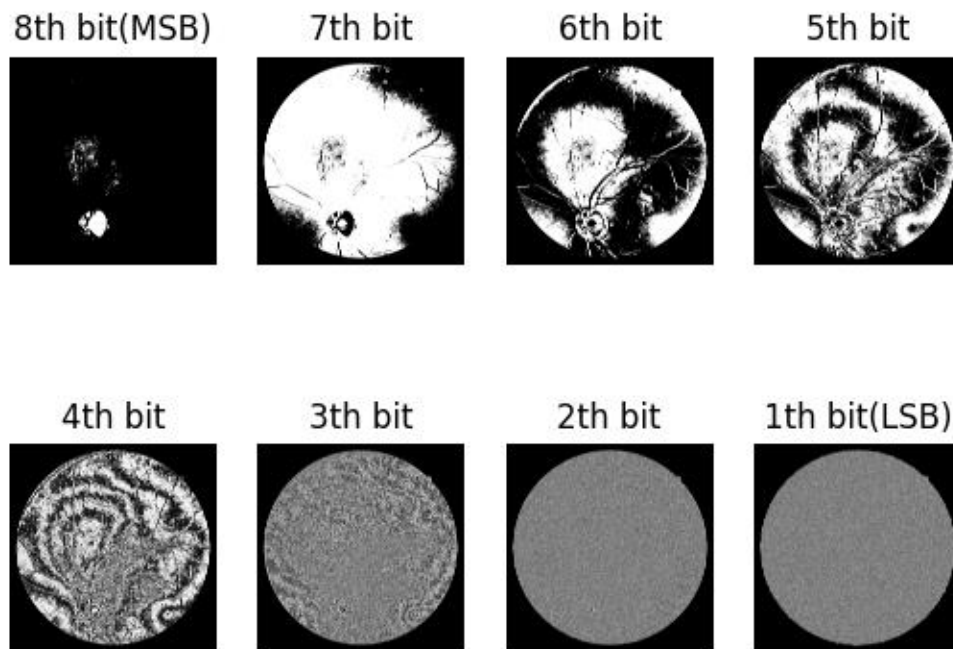


کمترین مقدار اختلاف دو تصویر به صورت زیر است:

d= [3.9085085]

بخش د) bit plane slicing برای نمایش یک تصویر با یک یا چند بیت استفاده شده در هر بایت است. در grayscale=۲۵۶ پیکسل ها در ۸ بیت ذخیره می شوند.

برای استخراج صفحات بیت های مختلف به شکل زیر عمل می کنیم:
ابتدا مقادیر پیکسل های تصویر دریافتی را از دسیمال به باینری تبدیل کرده و آن را در 2^{bit-1} ضرب می کنیم.



همانطور که مشاهده می شود شکل کلی تصویر در صفحات بیتی با ارزش تر شکل می گیرد و صفحات با بیت های کمتر ارزش تر مربوط به جزئیات تصویر هستند.

سوال شماره ۲:

بخش الف)

$$s(r) = (L-1) \sin(\omega r) \quad \text{domain} = [0, L-1] \quad \text{range} = [0, L-1]$$

$$L-1 = (L-1) \sin(\omega(L-1)) \rightarrow \sin(\omega(L-1)) = 1$$

$$\omega(L-1) = \sin^{-1}(1) = \frac{\pi}{2}$$

$$\omega = \frac{\pi}{2(L-1)} \xrightarrow{L=256} \omega = \frac{\pi}{510}$$

CS Scanned with CamScanner

بخش ب) دو تصویر CT_۱.tif و CT_۲.tif را به صورت خاکستری ۸ بیتی فراخوانی می کنیم.

تابع موردنظر رابطه ی بدست آمده از بخش الف را برای هر پیکسل تکرار و با مقدار جدید جایگزین می کند.

$$s(r) = (L-1) * \sin\left(\frac{\pi}{510} * r\right)$$

CT_1



CT_2



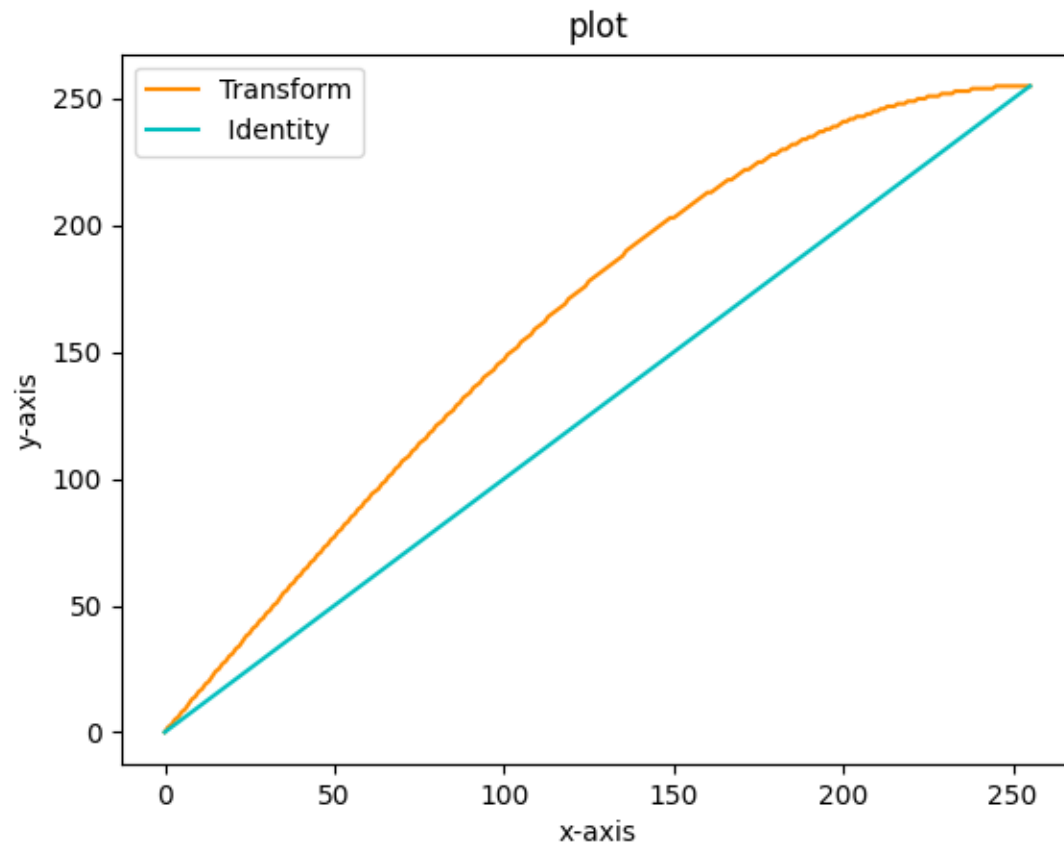
CT_1 transformed



CT_1 transformed



بخش ج) ابتدا با استفاده از تابع `np.linspace` گروهی از اعداد صحیح تولید می کنیم و مقدار آنها را در تابع بخش ب و تابع $y=x$ محاسبه و نمودار آنها را رسم می کنیم.



بخش د) تابع $S(r)$ محدوده ی باریک از مقادیر با شدت خاکستری پایین را به سطح وسیع تر از خاکستری نگاشت می کند و برای گسترش مقادیر تاریک در یک تصویر به کار می رود. این تبدیل برای تصویر ورودی با شدت های خاکستری بالا (تاریک) مناسب است.

سوال شماره ی ۳:

بخش الف)

- ۱- تصویر به صورت خاکستری فراخوانی نشده است.
- ۲- فیلتر اعمال شده بر روی تصویر نمی تواند نویز نمک فلفلی را به طور کامل حذف کند. بنابراین از فیلتر median استفاده می کنیم تا تصویر به شکل غیرخطی فیلتر شود و اثر نویز نیز به طور کامل از بین برود.
- ۳- فیلتر sobel لبه ی بالارونده تصویر را استخراج می کند اما برای دیده شدن لبه های عمودی مقدار $dy=1$ و مقدار $dx=0$ قرار می دهیم.

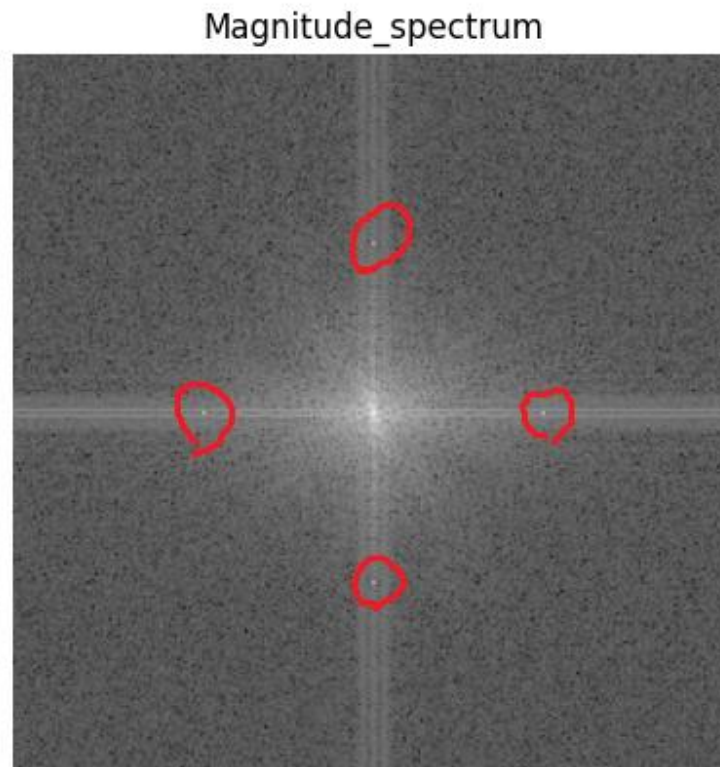
خروجی :

Problem 3 Figure



سوال شماره ۴:

ورودی برنامه تصویر xray_checkered.png است که به صورت خاکستری ۸ بیتی آنرا فراخوانی می کنیم.
ابتدا از تصویر تبدیل فوریه گرفته و فرکانس صفر را به مرکز منتقل می کنیم و *magnitude spectrum* آن را حساب کرده و نمایش می دهیم.



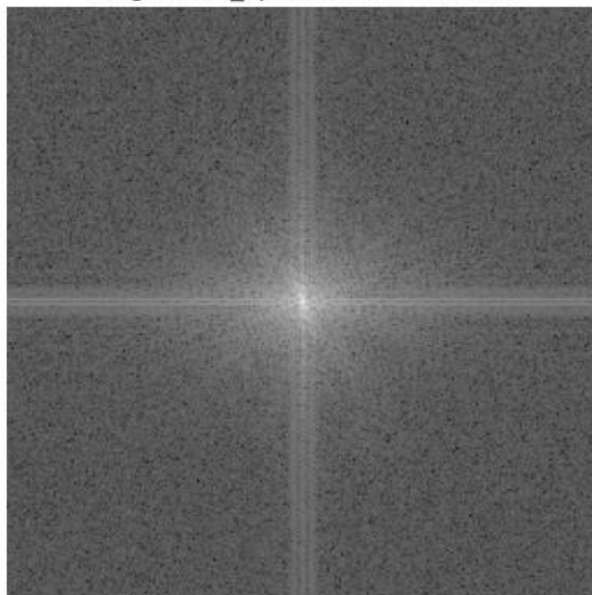
فرکانس های اعوجاج به صورت نقاط سفید(فرکانس های با شدت بالا) در تصویر مشاهده می شوند. برای بدست آوردن مختصات این فرکانس ها به صورت زیر عمل می کنیم:

- ۱- این فرکانس ها دو خاصیت دارند که با پیدا یک فرکانس می توانیم فرکانس مقابل آن را بدست آوریم:
(الف)فرکانس دوم قرینه فرکانس اول در محور می باشد. (ب)شدت دو فرکانس با یکدیگر برابر است.
- ۲- تصویر را به گونه ای کات می کنیم که یک بار بالاترین فرکانس همان فرکانس اعوجاج سمت چپ باشد. ماکسیمم را بدست می آوریم. به طور مشابه برای فرکانس بالایی نیز این عمل را تکرار می کنیم. حال مقدار دو فرکانس اعوجاج را داریم.

۳- با یک حلقه و دستور شرطی هرگاه مقدار طیف اندازه با مقدار یکی از فرکانس هایی که در قسمت قبل بدست آوردیم برابر شد آن را با چهار پیکسل همسایه جایگزین می کنیم.

magnitude spectrum پس از اعمال تغییرات به شکل زیر است:

Magnitude_spectrum(Denoised)



همانطور که مشاهده می شود اثر فرکانس های اعوجاج از بین رفته اند. اکنون فرکانس صفر را به محل اصلی باز می گردانیم و از تصویر تبدیل فوریه معکوس می گیریم.

خروجی برنامه:

