

گزارش تمرین اول multimedia

**دانشجو:**

پویا شریفی

**استاد:**

دکتر شریفیان

سوال 1:

**اولین ماتریس ,دارد رنگ ها را invert میکنند**:

که با استفاده از کتابخانه open cv در پایتون نوشته شده است. هدف از این کد، ایجاد تصویر جدید با انجام برخی عملیات بر روی پیکسل‌های تصویر ورودی است.

این کد با ایجاد یک آرایه جدید از NumPy به نام new\_im\_2 با ابعاد مشابه تصویر ورودی gray شروع می‌شود. این آرایه تصویر جدیدی را که ساخته خواهد شد، نگهداری می‌کند.

سپس کد با استفاده از دو حلقه تو در تو بر روی پیکسل‌های تصویر ورودی حرکت می‌کند. برای هر پیکسل، کد مقدار پیکسل gray[i,j] را بررسی کرده و بر اساس مقدار آن، برخی عملیات را انجام می‌دهد.

برای هر نمودار به نحو مختلفی if ها را تعیین میکنیم ,مثلا برای دومی:

اگر مقدار پیکسل کمتر از ۰.۲ باشد، پیکسل متناظر در تصویر جدید برابر با ۲۵۵ بار مقدار پیکسل gray[i,j] قرار داده می‌شود. این عملیات باعث افزایش نواحی تیره تصویر می‌شود.

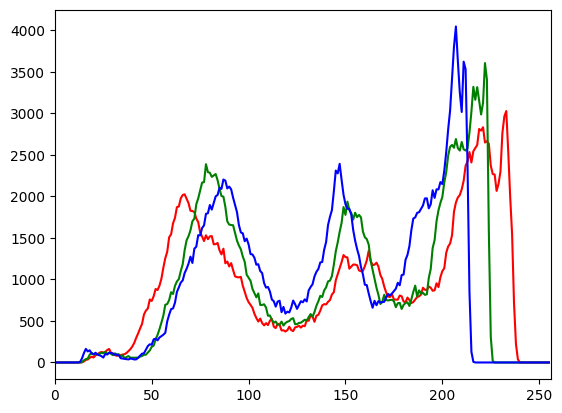
اگر مقدار پیکسل بین ۰.۲ و ۰.۵۵ باشد، پیکسل متناظر در تصویر جدید برابر با صفر قرار داده می‌شود. این عملیات باعث حذف پیکسل‌هایی با شدت متوسط می‌شود و تصویر دودویی ایجاد می‌شود.

اگر مقدار پیکسل بیشتر از ۰.۵۵ باشد، پیکسل متناظر در تصویر جدید برابر با ۲۵۵ بار مقدار پیکسل gray[i,j] قرار داده می‌شود. این عملیات باعث افزایش نواحی روشن تصویر می‌شود.

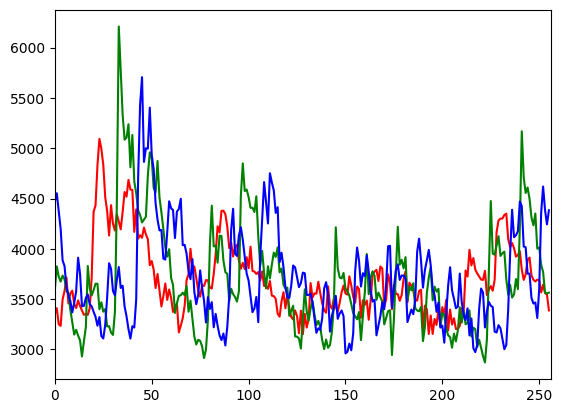
**پس در کل تصویر دو بیشتر نزدیک به سیاه و سفید میشود و خاکستری های بین از بین میروند.**

**سوال2:**

**در این قسمت** ما اول کانال های مختلف رنگی را جدا کرده ,سپس تغییر اندازه دادیم ,جمع کردیم و هستوگرام آن را کشیدیم.



و سپس پس از این تغییرات



دلیل این تغییرات این است که در هنگام resize کردن برای پیدا کردن پیکسل های میانی,تغریب هایی زده میشود.مانند آن است که این دیاگرام را از یک فیلتر رد کرده باشیم.

سوال 3:

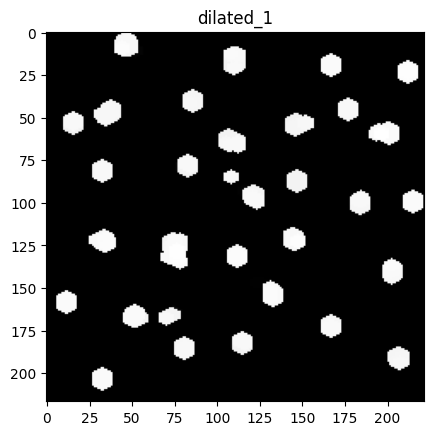
برای جدا کردن این خط ها از ترکیب دو فانکشن استفاده کردیم.

اول

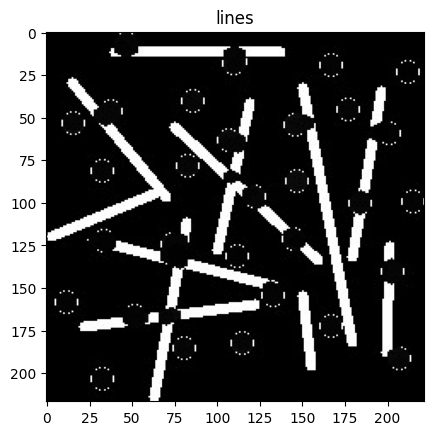
eroded\_img = cv2.erode(img, kernel, iterations=2)

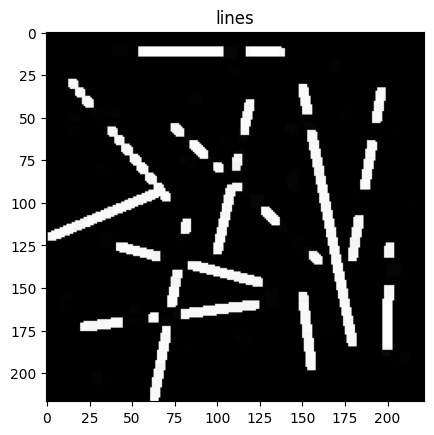
که باعث میشود خط ها محو شوند و دایره ها کوچک شوند ,سپس دایره ها را بزرگ میکنیم:

dilated = cv2.dilate(eroded\_img, kernel, iterations = 2)



برای خط ها نیز از کم کردن دو تا عکس استفاده کردیم:





سوال 4:

با استفاده از پکیج OpenCV، می‌توان به راحتی عملیات پردازش تصویر را انجام داد. در این میان، فیلترهای مختلفی وجود دارند که می‌توان از آن‌ها برای بهبود کیفیت تصویر استفاده کرد. در این گزارش به معرفی سه نوع فیلتر از جمله Median، Bilateral و Gaussian پرداخته می‌شود.

فیلتر Median:

در فیلتر Median، برای هر پیکسل، مقدار میانگین پیکسل‌های همسایه به عنوان مقدار جدید برای آن پیکسل انتخاب می‌شود. این فیلتر برای حذف نویز و افزایش وضوح تصویر بسیار مفید است.

فیلتر Bilateral:

فیلتر Bilateral، یکی از محبوب‌ترین فیلترهای پردازش تصویر است که در بسیاری از برنامه‌های کاربردی استفاده می‌شود. این فیلتر برای کاهش نویز و حفظ لبه‌های تصویر بسیار مناسب است. در این فیلتر، به جای میانگین پیکسل‌های همسایه، میانگین وزن‌دار برای هر پیکسل محاسبه می‌شود. برای محاسبه این میانگین وزن‌دار، وزنی برای هر پیکسل در نظر گرفته می‌شود که برابر با فاصله پیکسل با پیکسل مورد نظر است.

فیلتر Gaussian:

فیلتر Gaussian برای حذف نویز و افزایش وضوح تصویر مورد استفاده قرار می‌گیرد. در این فیلتر، برای هر پیکسل، مقدار میانگین وزن‌دار پیکسل‌های همسایه به عنوان مقدار جدید آن پیکسل در نظر گرفته می‌شود. وزن‌دهی در این فیلتر بر اساس توزیع Gaussian انجام می‌شود.