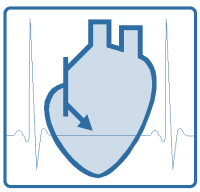


دانشگاه صنعتی امیرکبیر

دانشکده مهندسی پزشکی

گزارش تمرین دوم پردازش تصویر

دانشجو

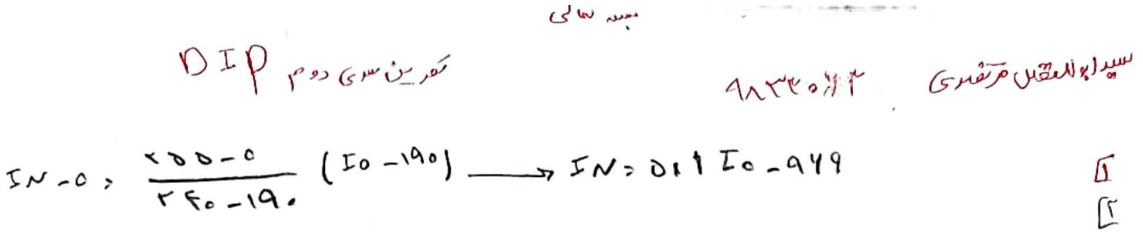
سیدابوالفضل مرتضوی ۹۸۳۳۰۶۳

آبان ۱۴۰۱

فهرست مطالب

تمرین اول

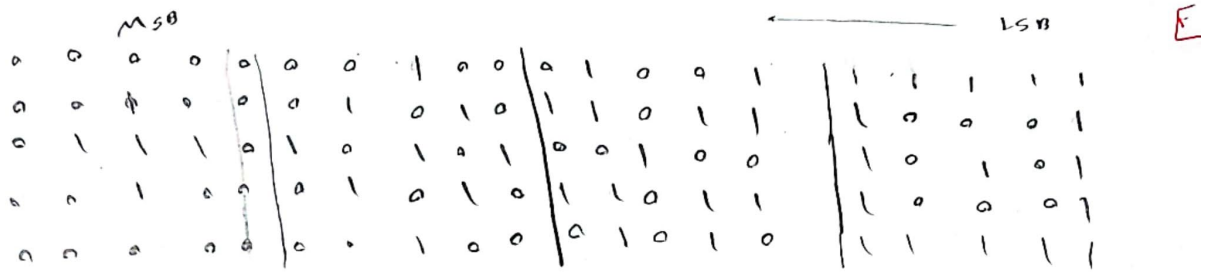
۱-۱ الف



۱-۲ ب

با توجه به اینکه سطح اینتنسیتی تصویر در حد بالا قراردارد باید برای اینکه این محدوده گسترده شود و کنتراست افزایش یابد از γ کوچکتر از ۱ استفاده کنیم. در مورد پارامتر C ، این پارامتر را می‌توان از طریق فرمول $C=(255)^{1-\gamma}$ به دست آورد. این پارامتر معمولا در این مواقع عددی بزرگتر از ۱۰ است.

تمرین دوم



تمرین سوم

۳-۲ بخش دوم

تابع خواسته‌شده به صورت زیر نوشته شد.

```
# transformattion PL and LCS
def transform(img,transformname,gama=1):
    img_MAX=img.max()
    img_MIN=img.min()
    if transformname=='PL':
        for i in range(512):
            for j in range(512):
                img[i,j]=img[i,j]**gama
    elif transformname=='LCS':
        for i in range(512):
            for j in range(512):
                img[i,j]=(255*(img[i,j]-img_MIN))/(img_MAX-img_MIN)
    return img
```

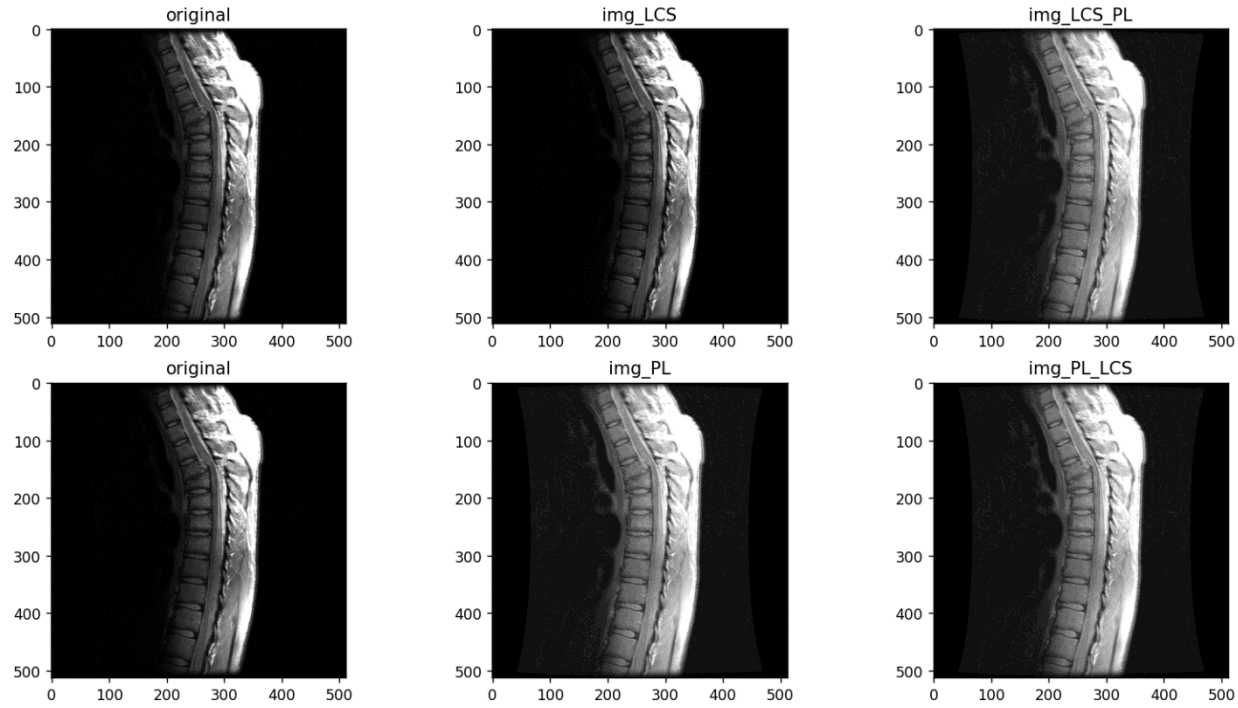
در این تابع از اسم اختصاری PL برای تبدیل توانی و از اسم LCS برای تبدیل خطی استفاده شد. در تبدیل خطی از مقدار ماکزیمم و مینیمم موجود در تصویر استفاده شد و در نهایت فرمول تبدیل خطی به وجود آمد.

۳-۳ بخش سوم

```
img_LCS=transform(img,'LCS').astype('uint8')
img_LCS1=img_LCS.copy()
img_PL=transform(img,'PL',0.5).astype('uint8')
img_PL1=img_PL.copy()
img_LCS_PL=transform(img_LCS1,'PL',0.5).astype('uint8')
img_PL_LCS=transform(img_PL1,'LCS').astype('uint8')
```

در این بخش تبدیل های خواسته شده برروی تصویر اعمال شدند.

۳-۴ بخش چهارم



شکل ۳.۱ خروجی تبدیل های تمرین اول

۳-۵ بخش پنجم

همانطور که درشکل ۳.۱ نشان داده شده است.اگر ابتدا تبدیل توانی برروی تصویر اعمال کنیم باعث افزایش کنتراست می‌شود و سپس با اعمال یک تبدیل خطی بازه اینتنسیتی را در در محدوده۲۵۵-۰ قرار می‌دهیم. بااین کار کنتراست بهتر افزایش می‌یابد. اما اگر ابتدا تبدیل خطی را اعمال کنیم این کار باعث کشیدگی محدوده اینتنسیتی تصویر می‌شود و با اعمال تبدیل توانی برروی این تصویر کنتراست تغییر چندانی نمی‌کند. چرا که ابتدائا بازه اینتنسیستی تصویر کشیده شده است.

تمرین چهارم

۴-۱ بخش اول

```
dimension of 8 bit image is: (706, 320) and 16 bie: (493, 600)
type of 8 : <class 'numpy.uint8'> and 16:  <class 'numpy.uint16'>
```

شکل ۴.۱ خروجی مربوط به ابعاد و نوع دادگان تصویر

۴-۲ بخش دوم

```
def performCLAHE(img):
    img1=img.copy()
    clahe=cv.createCLAHE(clipLimit=40.0, tileGridSize=(3,3))
    CLAHE_IMG=clahe.apply(img1)
    return CLAHE_IMG
```

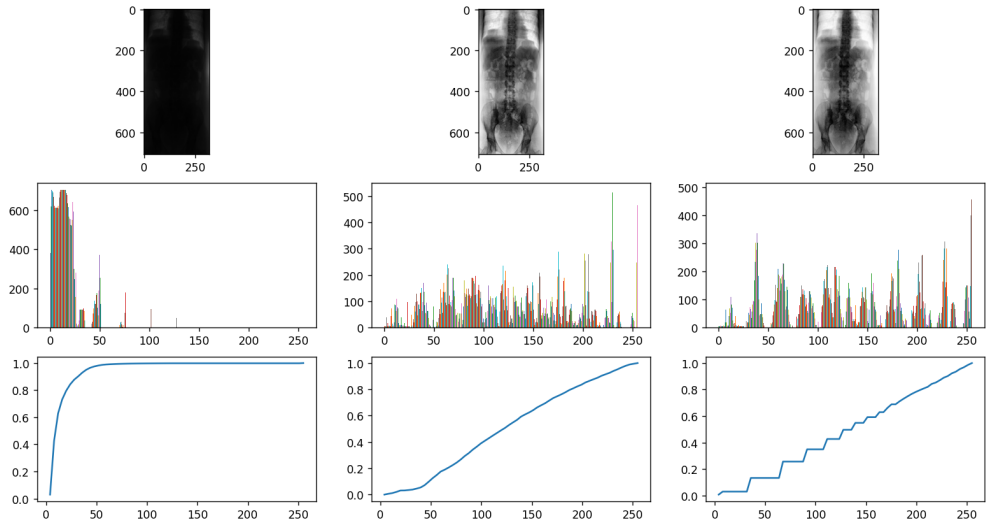
در فیلتر Clahe پارامتر اول سطح آستانه برای محدودیت کنتراست است. مقدار پیش فرض این پارامتر برابر ۴۰ است. پارامتر دوم مربوط به ابعاد پنجره اعمال فیلتر است که به طور پیش فرض ۸*۸ است.

۴-۳ بخش سوم

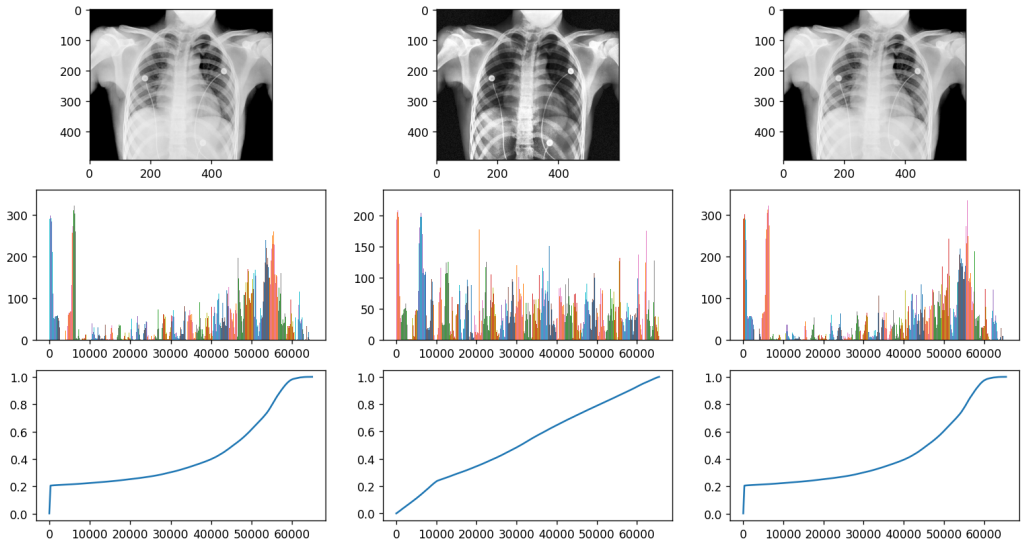
```
def transform(img,bit_depth):
    img2=img.copy()
    if type(img2[1,1])==np.uint8:
        equalized_img=cv.equalizeHist(img2)

    elif type(img2[1,1])==np.uint16:
        row,col=img2.shape
        equalized_img=np.zeros([row,col])
        equalized_img=cv.normalize(img2,equalized_img, 0,
65535,cv.NORM_MINMAX)
    return equalized_img
```

در این تابع برای یکنواخت سازی هیستوگرام بین تصویر ۸ بیتی و ۱۶ بیتی تفاوت قائل شدم. چراکه متد equalizeHist با وجود سریعتر بودن از تصاویر ۱۶ بیتی پشتیبانی نمی‌کند.

۴-۴ بخش چهارم



شکل ۴.۲ خروجی تصویر ۸بیتی



شکل ۴.۳ خروجی تصویر ۱۶ بیتی

۵-۴ بخش پنجم

با توجه به خروجی های ایجاد شده در شکل ۴.۲وشکل ۴.۳ بااستفاده از این روش کنتراست افزایش یافته است. این مورد در تصویر ۸بیتی که هیستوگرام آن فشرده است بهتر قابل مشاهده است. اما در تصویر ۱۶ بیتی که خود تصویر دارای هیستوگرام تقریبا متعادل است زیاد قابل مشاهده نیست.