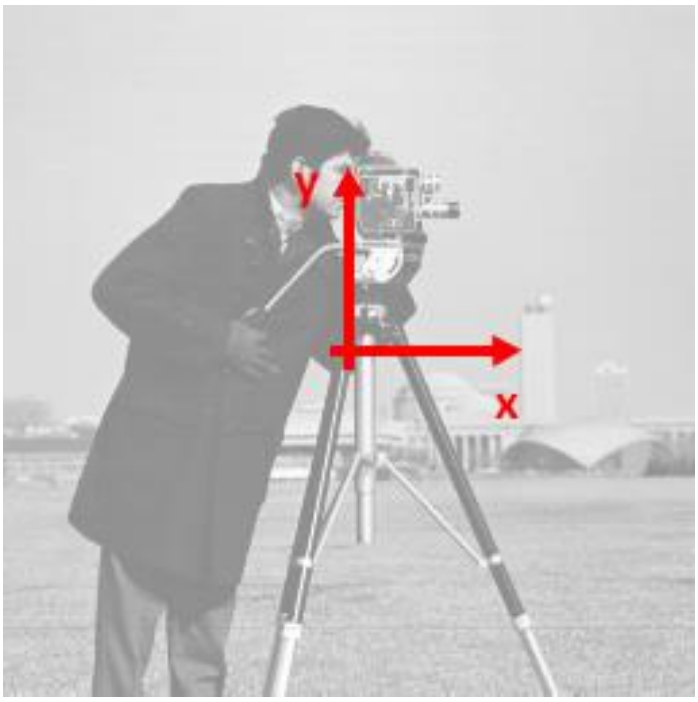
گزارش تمرین سری سوم

سیستم‌های چندرسانه‌ای

رضا اکبریان بافقی - ۹۵۱۰۰۰۶۱

۱ تبدیل تصویر

من در این سوال محور مختصات را مانند شکل یک در تصویر مورد نظر فرض کرده‌ام.



شکل ۱ محور مختصات فرضی در تصویر مورد نظر

برای یافتن تصویر تقارن یافته نسبت به محور x یا همان خط y=0، با استفاده از دستور زیر این‌کار را انجام داده‌ام:

```
res = im(m:-1:1,1:n);
```

که در اینجا در واقع در محور y به‌طور برعکس پیکسل‌ها را جایگذاری می‌کنیم.

برای یافتن تصویر تقارن یافته نسبت به محور y یا همان خط x=0، با استفاده از دستور زیر این‌کار را انجام داده‌ام:

```
res = im(1:m,n:-1:1);
```

که در اینجا در واقع در محور x به‌طور برعکس پیکسل‌ها را جایگذاری می‌کنیم.

برای یافتن تصویر تقارن یافته نسبت به خط y=x، با استفاده از دستور زیر این‌کار را انجام داده‌ام:

```
theta = -90;
tform = affine2d([cosd(theta) sind(theta) 0; -sind(theta)
cosd(theta) 0; 0 0 1]);
res = imwarp(im,tform);
res = res(1:m,n:-1:1);
```

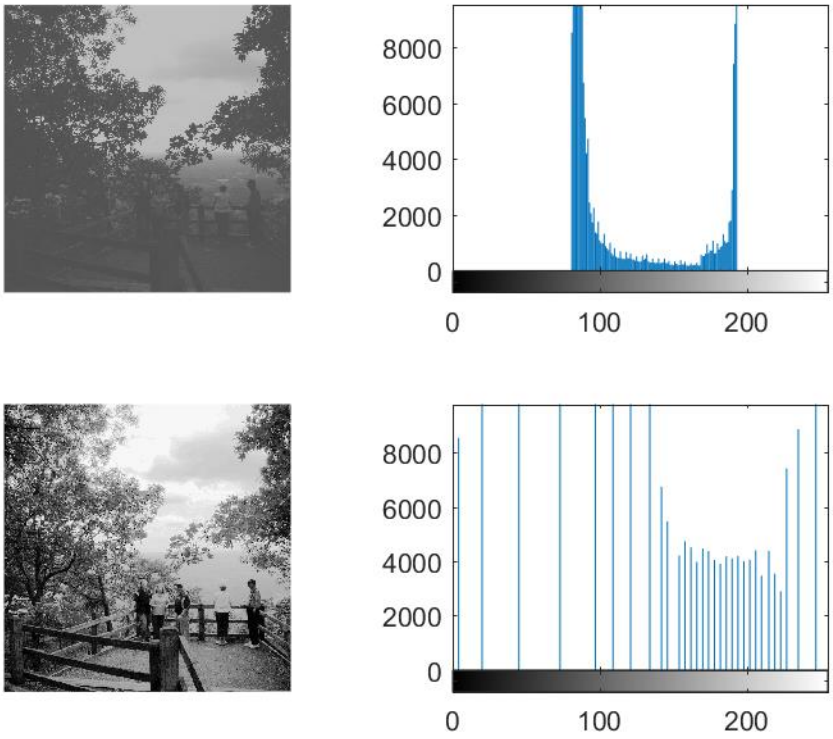
که در اینجا در واقع ابتدا با زاویه ۹۰- درجه عکس را دوران می‌دهیم. این کار با تشکیل ماتریس آفین با زاویه ۹۰- درجه و warp کردن تصویر با آن ماتریس انجام‌پذیر می‌باشد. سپس نسبت به محور y آن را تقارن می‌دهیم. این کار در نهایت مانند این است که نسبت به خط y=x تصویر را تقارن بدهیم.

برای خط y=-x هم همین مراحل مشابه برای y=x را انجام داده‌ام با این تفاوت که ابتدا با زاویه ۹۰ درجه عکس را دوران می‌دهم سپس نسبت به محور x آن را تقارن می‌دهم. نتیجه مانند این است که نسبت به خط y=-x تقارن انجام دهیم.

۲ تغییر سطوح روشنایی تصویر

۱,۲

با اعمال این تابع دامنه روشنایی را گسترش می‌دهیم.



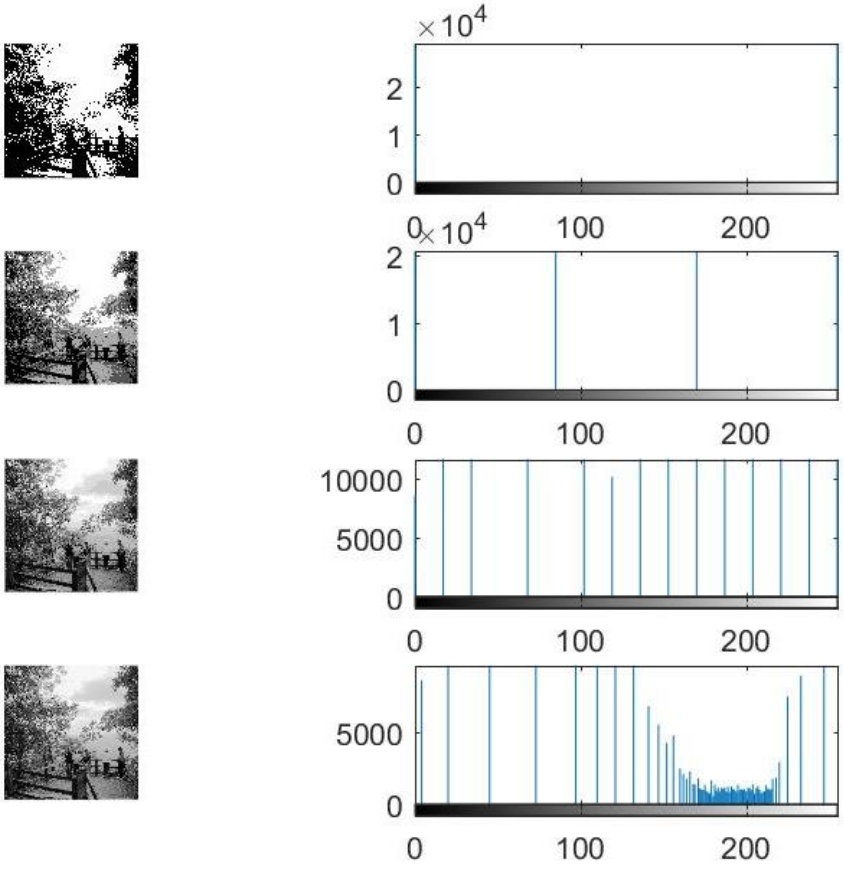
شکل ۲ اعمال تابع histeq روی تصویر به همراه هیستوگرام متناظرشان

۲,۲

هرچه تعداد سطح‌های خاکستری تصویر نهایی بیشتر باشد، تصویر کیفیت بهتری خواهد داشت. چون هر چه هیستوگرام تصویر دامنه بیش‌تری باشد، تفاوت روشنایی در بخش‌های مختلف تصویر مشخص‌تر می‌باشد. در واقع با زیاد کردن n تعداد پیکسل‌های در هر سطح کم‌تر می‌شود و این پیکسل‌ها در دامنه پخش می‌شوند. همچنین نسبت به تصویر اولیه در همه n ها این هیستوگرام پخش‌تر شده است.

$$h(v) = \text{round}\left(\frac{cdf(v) - cdf_{min}}{(M \times N) - cdf_{min}} \times (L-1)\right)$$

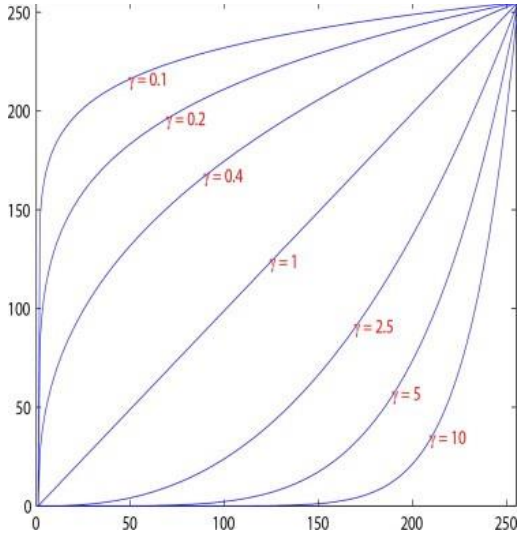
پس طبق فرمول بالا که هیستوگرام هر روشنایی ورودی را مشخص می‌کند، که هر L که همان تعداد سطوح خاکستری بیش‌تر باشد، این اسکیل بهتر می‌شود.



شکل ۳ از بالا به پایین n زیاد تر می شود.

۳,۲

در Gamma Correction با توجه به ضریب گاما روشنایی‌های اولیه را به روشنایی ثانویه‌ای مپ می‌کند. می‌توانید یک نمودار Gamma Correction ساده را در شکل ۴ مشاهده کنید.

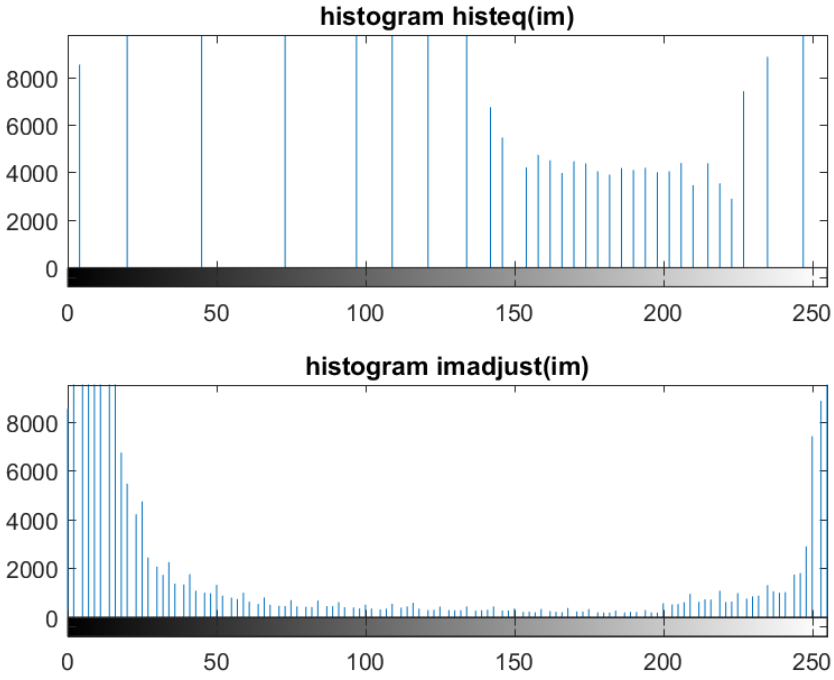


شکل ۴ نمودار یک Gamma Correction

در این‌جا ضریب گاما را برابر ۰,۰۱ قرار داده‌ایم بنابراین زمانی که روی تصویر اعمال می‌کنیم، روشنایی‌های تصویر جدید یا بسیار نزدیک به صفر می‌شوند یا بسیار نزدیک به ۲۵۵ می‌شوند. که مقدار بیشتری از روشنایی‌ها سمت ۰ می‌روند تا این‌که نزدیک ۲۵۵ بروند چون گاما خیلی کوچک‌تر از ۱ است. این اتفاق در شکل ۶ در هیستوگرام مربوط با Gamma Correction قابل مشاهده می‌باشد. با این گاما در نقاط با روشنایی کمتر جزئیات بیش‌تری مشخص می‌باشند نسبت به نقاطی که روشنایی زیادی داشتند.

ولی در Histogram Equalization ما روشنایی که را در طول دامنه پخش می‌کنیم. به همین دلیل جزئیات بیش‌تری در آن مشخص است. هرچند در هر دو عمل ما کنتراست تصویر را زیاد می‌کنیم.

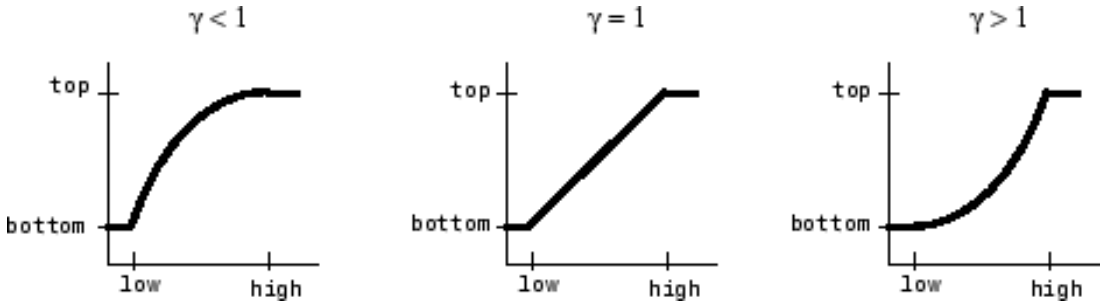
شکل ۵ تصویر حاصل از imadjust و histeq و مقایسه آن با تصویر اصلی



شکل ۶ هیستوگرام تصویر حاصل از imadjust و histeq

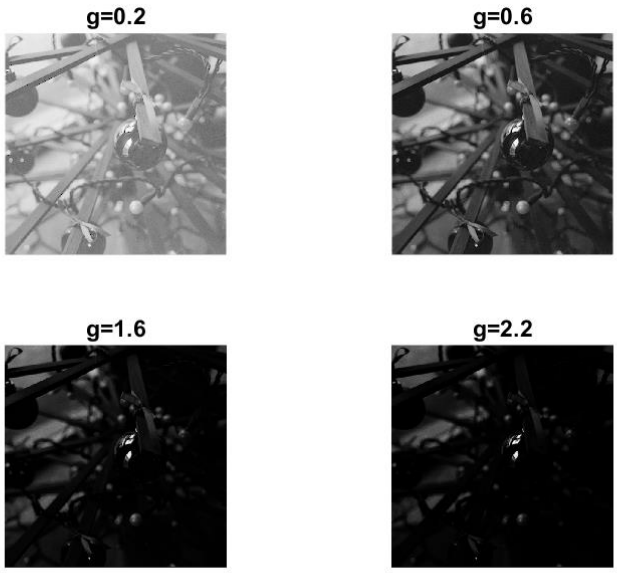
۴,۲

همان‌طور که در شکل ۴ می‌توانید مشاهده کنید، هرچه گاما بیش‌تر از یک باشد این مپینگ بیش‌تر به سمت نقاط تاریک‌تر می‌باشد ولی هر چه گاما کمتر از یک باشد این میپینگ به سمت نقاط روشن‌تر می‌رود. به این معنی که تعداد روشنایی‌های بیشتری به روشنایی‌های بالاتر مپ می‌شوند.



شکل ۷ هر چه گاما بیش‌تر از یک باشد این مپینگ بیش‌تر به سمت نقاط تاریک‌تر می‌باشد ولی هر چه گاما کمتر از یک باشد این میپینگ به سمت نقاط روشن‌تر می‌رود.

با توجه به توضیحات بنابراین در گامای پایین ۰,۲ تصویر روشن‌تر از تصویر اولیه شده‌است ولی هر چه گاما بیش‌تر شده است، تصویر ما هم تاریک‌تر شده‌است. از Gamma Correction برای افزایش کنتراست تصویر و هم‌چنین بهبود کیفیت استفاده می‌شود. این عمل مپینگ به‌صورت خطی عمل می‌کند. با اعمال Gamma Correction هیستوگرام تصاویر نیز دچار تغییر می‌شود.



شکل ۸ تاثیر مقادیر مختلف گاما روی تصویر

۵,۲

در حالت HSV، مقدار Value، همان مقدار متناظر با روشنایی می‌باشد. پس ما ابتدا این آرایه دو بعدی متناظر با Value را جدا می‌کنیم و روی آن توابع histeq و imadjust را اعمال می‌کنیم. سپس آن را به آرایه سه بعدی قبلی اضافه می‌کنیم و آن را تبدیل به RGB می‌کنیم. در شکل ۹ این کار برای فضای رنگی HSV انجام شده است.

شکل ۹ اعمال توابع histeq و imadjust در حالت HSV

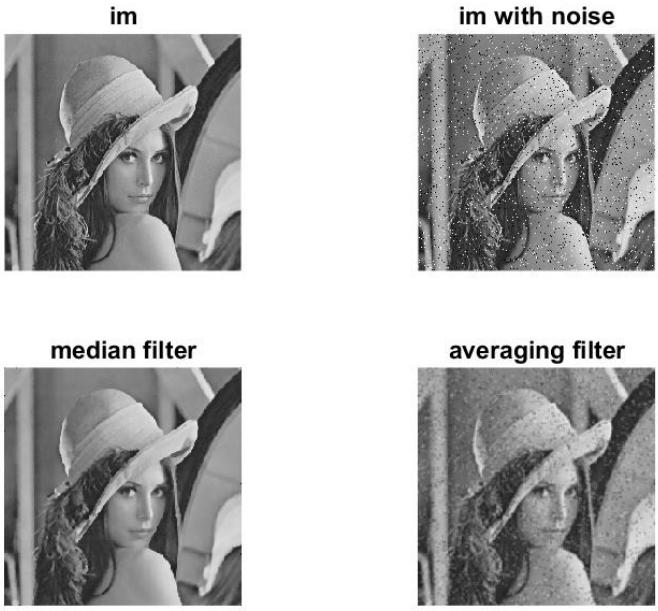
در حالت YCbCr، مقدار Luminance، همان مقدار متناظر با روشنایی می‌باشد. ما همان مراحلی که برای Value در HSV انجام می‌دادیم، برای این مؤلفه انجام می‌دهیم. می‌توانید نتیجه کار را در شکل ۱۰ مشاهده کنید. همان‌طور که می‌توان مشاهده کرد در حالت HSV تصاویر به حالت واقعی نزدیک‌تر شدند. در حالت YCbCr اما اختلاف روشنایی‌ها بیش‌تر می باشند.

شکل ۱۰ اعمال توابع histeq و imadjust در حالت YCbCr

۳ بهبود تصاویر

۱,۳

به‌طور واضح فیلتر median عملکرد بهتری داشته است. دلیل این اتفاق می‌تواند این باشد که چون نویز ایجاد شده با فاصله از هم و به صورت نقطه‌ای ایجاد شده است، پس در میانگین مربع ۳ در ۳ تاثیر می‌گذارد. از این رو بهتر است از فیلتر median استفاده کنیم چون این تفاوت زیاد ایجاد شده توسط نویز توسط فیلتر نادیده گرفته می‌شود. چون بیشتر همسایه های آن پیکسل دست نخورده باقی مانده است. اما در فیلتر averaging یک تغییر زیاد در همسایه‌ها می‌تواند در مقدار آن پیکسل بعد از فیلترینگ اثر بگذارد.



شکل ۱۱ تفاوت حاصل از اعمال فیلتر median و averaging

۲,۳

با استفاده از فیلتر گاوسی، چون نوعی میانگین‌گیری وزن‌دار بین همسایه‌های یک پیکسل انجام می‌شود، تاثیر نویز را کاهش می‌دهد اما هنوز نسبت به تصویر اصلی نویزها مشخص می‌باشند. به نوعی در تصویر حل می‌شود نویزها.

شکل ۱۲ تاثیر فیلتر گاوسی بر نویز ایجاد شده

۳,۳

فیلتر median در این‌جا بهتر عمل کرده است. به همان دلیلی که در ۱,۳ اشاره شد. (فاصله‌ای که در نویزها وجود داشت.) اما gaussian نسبت به averaging بهتر عمل می‌کند. چون در gaussian ما به‌صورت وزن‌دار همسایه‌های یک پیکسل را با هم میانگین می‌گیریم به‌طوری که همسایه‌های نزدیک‌تر به پیکسل ضریب بیش‌تری دارند. اما در averaging همه همسایه‌ها با ضریب یکسان باهم میانگین‌گیری انجام می‌شود.

نوع فیلتر	gaussian	averaging	median
mse	۱۴۳,۳۷۵۳	۲۰۵,۴۰۲۱	۷۵,۱۰۶۸

شکل ۱۳ جدول مقایسه فیلترها

۴,۳

می‌توانیم مشاهده کنیم canny در مقایسه با sobel عملکرد بهتری برای آشکارسازی لبه‌ها داشته‌است. در sobel ما نقاطی را انتخاب می‌کنیم که در دو جهت بیش‌ترین مقدار مشتق را داشته باشد و این مشتق گیری توسط فیلتر sobel انجام می‌شود ولی در canny ما به دنبال ماکسیمم مشتق‌های محلی می‌باشیم و عمل مشتق‌گیری را توسط فیلتر مشتق‌گیر گاوسی انجام می‌دهیم. این کار باعث می‌شود نویزها به عنوان لبه شناخته نشوند.

شکل ۱۴ تفاوت آشکارسازی لبه‌ها توسط canny و sobel

۴ فشرده‌سازی تصویر

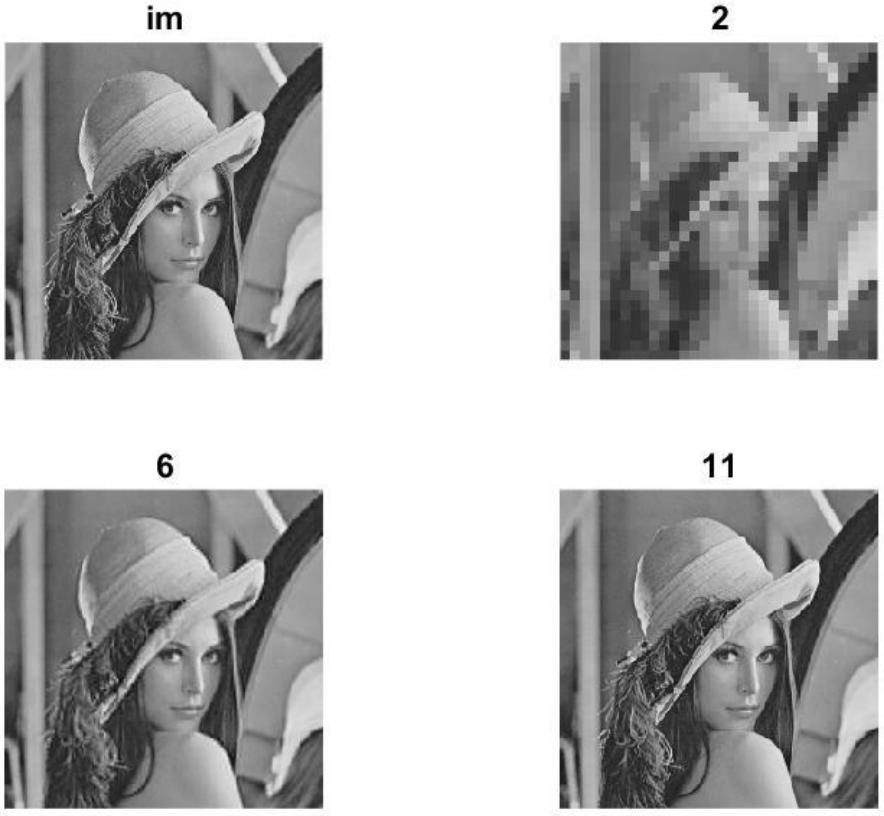
هر چه تعداد بیشتری از ضرایب DCT را صفر کنیم، کیفیت تصویر کاهش پیدا می‌کند. به همین دلیل تصویری که از خط شماره ۱۱ به بعد را صفر کرده‌ایم کیفیت بیش‌تری دارد. می‌توانید در جدول زیر مشاهده کنید که با زیاد کردن n، mse کاهش زیادی داشته است.

N	۲	۶	۱۱
Mse	۰,۰۰۷۹	۰,۰۰۱۲	۰,۰۰۰۱۷۶۷

شکل ۱۵ جدول نمایان‌گر mse تصاویر بازیابی شده با تصویر اصلی در n های مختلف

ما در فشرده‌سازی به هر قسمت از جدول ۸ در ۸ تعداد بیتی را برای نشان دادن عدد آن قسمت نسبت می‌دهیم و هر چه در جدول ۸ در ۸ به سمت پایین و سمت راست می‌رویم آن اعداد برای ما کم اهمیت‌تر می‌شوند و به همین دلیل برای آن‌ها بیت کم‌تری را نسبت می‌دهیم. به این دلیل که این فرکانس‌های بالا در یک مربع ۸ در ۸ کم‌تر رخ می‌دهند، هم‌چنین اگر آن قسمت را دقیقا ذخیره کنیم در موقع نمایشش، چشم انسان قابلیت تشخیص را ندارد. پس برای ما اهمیت زیادی ندارد. حال اگر ما این قسمت‌ها را صفر کنیم انگار جزئیات کم‌تری را در فرکانس‌های بالاتر ذخیره کرده‌ایم ولی همچنان تصویر کیفیت خوبی دارد.
در واقع ما چون از Zigzag ordering برای عمل هافمن کودینگ استفاده می‌کنیم با صفر کردن از شماره خط مشخصی به بعد، مشخص می‌کنیم از کدام خط به بعد یک تعداد صفر پشت سر هم می‌آیند و در واقع آن قسمت‌هایی که دارای فرکانس بالاتر هستند را حذف می‌کنیم.

ما مربع‌های ۸ در ۸ را انتخاب می‌کنیم چون در این مربع‌ها correlation بین پیکسل‌ها بیش‌تر می‌باشد. این عمل DCT گرفتن شبیه همان عملیات FFT می‌باشد.
در این کد من در تابع DCTWithN با n داده شده ماسک مورد نظر را درست می‌کنم و در هر مربع ۸ در ۸ ماتریس حاصل از اجرای blockproc ضرب می‌کنیم. سپس با استفاده از تابع عکس تبدیل DCT آن را تبدیل به ماتریس نهایی تصویر می‌کنیم و mse آن را با تصویر اولیه محاسبه می‌کنیم و در فایلی ذخیره می‌کنیم.



شکل ۱۶ در n های مختلف عمل صفر کردن جدول DCT انجام شده‌است.