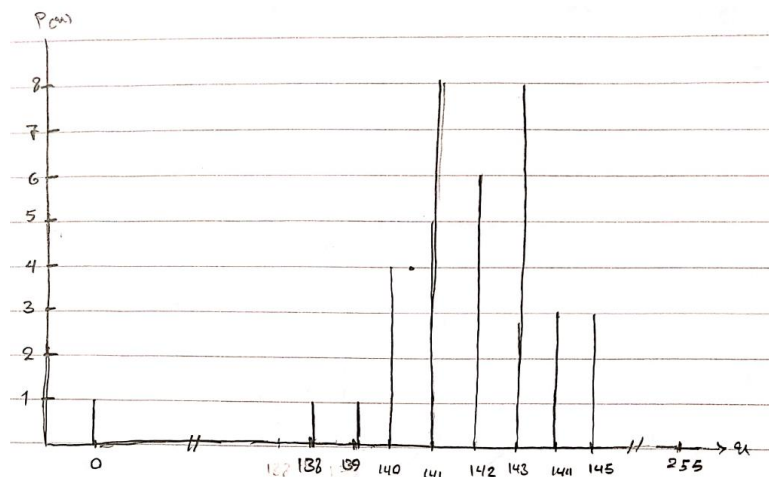


سوال ۱:

بخش الف) برای محاسبه هیستوگرام کافیست برای هر سطح رنگی از ۰ تا ۲۵۵، فرکانس تکرار آن را در تصویر پیدا کنیم و نمودار حاصل را رسم کنیم.

مقدار	۰	۱۳۸	۱۳۹	۱۴۰	۱۴۱	۱۴۲	۱۴۳	۱۴۴	۱۴۵
فرکانس	۱	۱	۱	۴	۸	۶	۸	۳	۳

حال و پس از محاسبه تعداد تکرارها نمودار را رسم می‌کنیم.



- اعمال کشش هیستوگرام: برای این منظور از رابطه زیر که در اسلایدها آورده شده است استفاده می‌کنیم:

$$g(x, y) = \text{stretch}[f(x, y)] = \left(\frac{f(x, y) - f_{\min}}{f_{\max} - f_{\min}} \right) (MAX - MIN) + MIN$$

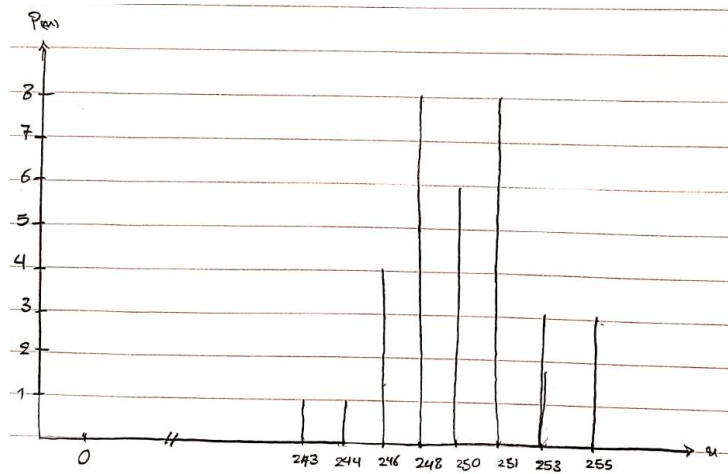
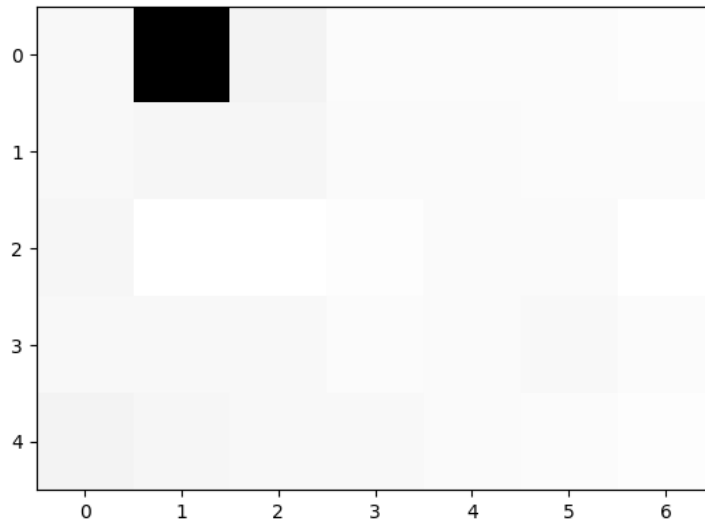
مقدار MIN و MAX به ترتیب عبارتند از بالاترین و بیشترین مقداری که یک پیکسل می‌تواند در اختیار داشته باشد، که در این مثال به ترتیب ۰ و ۲۵۵ هستند. همچنین مقادیر f_{\min} و f_{\max} نیز به ترتیب کوچکترین و بزرگترین مقدار در تصویر داده شده هستند، که به ترتیب برابرند با ۰ و ۱۴۵. بنابراین با جایگذاری این مقادیر به رابطه زیر می‌رسیم:

$$g(x, y) = \frac{(f(x, y))}{145} (255)$$

پس از اعمال این رابطه بر روی هر یک مقادیر $f(x, y)$ به جدول زیر برای مقادیر تصویر تازه می‌رسیم. با توجه به اینکه تعداد از این مقادیر اعشاری هستند و سطوح شدت روشنایی در تصویر بایستی به صورت عدد صحیح باشد، نیاز است که در مرحله نهایی عمل گرد کردن را نیز انجام بدهیم.

$f(x, y)$	۰	۱۳۸	۱۳۹	۱۴۰	۱۴۱	۱۴۲	۱۴۳	۱۴۴	۱۴۵
$g(x, y)$	۰	۲۴۲.۶۸	۲۴۴.۴۴	۲۴۶.۲	۲۴۷.۹۶	۲۴۹.۷۲	۲۵۱.۴۸	۲۵۳.۲۴	۲۵۵
گرد شده	۰	۲۴۳	۲۴۴	۲۴۶	۲۴۸	۲۵۰	۲۵۱	۲۵۳	۲۵۵

در نمودار هیستوگرام فرکانس‌ها تغییر نکرده و صرفاً محور x است که مقادیرش تغییر می‌کنند. به عنوان آخرین مرحله، هیستوگرام و تصویر این بخش را نمایش می‌دهیم. برای این منظور از نمودار را رسم کرده و تصویر را همراه با پیکسل‌هایش می‌نویسیم. همچنین تصویر ایجاد شده در کد را نیز به نمایش می‌گذاریم:



همچنین مقادیر بدست آمده برای پیکسل‌های تصویر نیز به صورت زیر است:

248	0	243	251	251	251	253
248	246	246	250	250	251	251
246	255	255	253	250	250	255
248	248	248	251	250	248	251
244	246	248	248	250	251	253

- اعمال برش هیستوگرام: برای این منظور از رابطه زیر که در اسلایدها آورده شده است استفاده می‌کنیم:

$$g(x, y) = clip[f(x, y)] = \left(\frac{f(x, y) - f_1}{f_{99} - f_1} \right) (MAX - MIN) + MIN$$

مقدار MIN و MAX به ترتیب عبارتند از بالاترین و بیشتری مقداری که یک پیکسل می‌تواند در اختیار داشته باشد، که در این مثال به ترتیب ۰ و ۲۵۵ هستند. همچنین مقادیر f_1 و f_{99} نیز به ترتیب کوچکترین و بزرگترین مقداری هستند که پس از کلیپ کردن تصویر در آن باقی‌مانده‌اند. برای این مثال با توجه به اینکه تنها مقداری که تفاوت قابل توجه‌ای با سایر مقادیر داشت و به گونه‌ای داده پرت محسوب می‌شد، مقدار صفر بود که در کنار گذاشته شد. برای انتخاب حد پایین اگر مطابق آنچه که در کلاس گفته شده است پیش برویم، یعنی پس از کنار گذاشتن چند درصد مقدار اولیه، کوچکترین مقدار بعد از آن را انتخاب کنیم، مقدار ۱۳۸ انتخاب خواهد شد. اما از طرفی می‌توان برای بیشتر کردن کنتراست تصویر مقدار ۱۳۷ را نیز انتخاب کرد چرا که تمام مقادیر غیر از صفر به صورت صعودی و با

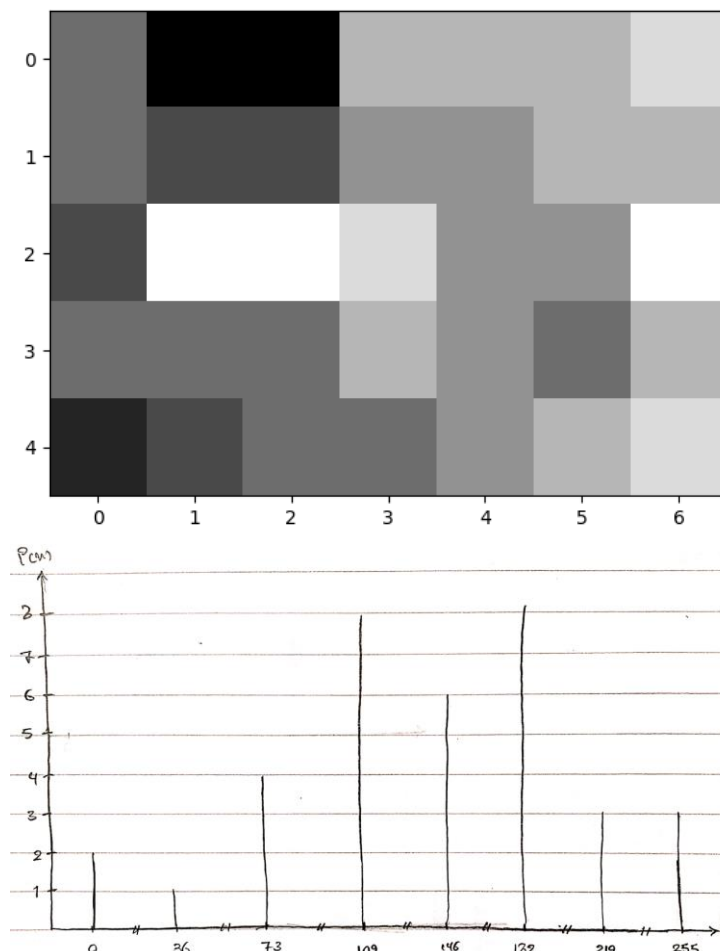
اختلاف یک واحد افزایش می‌یابند. پس می‌توان با کلیپ کردن صفر اولیه و تبدیل آن به ۱۳۷ کنتراست تصویر را بیش از پیش کرد طوری که هیچ دو مقداری بعد از اعمال الگوریتم به یک مقدار یکسان نگاشت نشوند. در این جا هر دو راه‌حل در نظر گرفته می‌شود. در هر دو این‌ها بزرگ‌ترین مقدار ۱۴۵ خواهد بود.
در صورت در اختیار گرفتن مقدار ۱۳۸ به عنوان کوچکترین مقدار، رابطه زیر را خواهیم داشت:

$$g(x,y) = \frac{(f'(x,y) - 138)}{7} (255)$$

پس از اعمال این رابطه بر روی هر یک مقادیر $f(x,y)$ به جدول زیر برای مقادیر تصویر تازه می‌رسیم. با توجه به اینکه تعداد از این مقادیر اعشاری هستند و سطوح شدت روشنایی در تصویر بایستی به صورت عدد صحیح باشد، نیاز است که در مرحله نهایی عمل گرد کردن را نیز انجام بدهیم.

$f(x,y)$	۰	۱۳۸	۱۳۹	۱۴۰	۱۴۱	۱۴۲	۱۴۳	۱۴۴	۱۴۵
$f'(x,y)$	۱۳۸	۱۳۸	۱۳۹	۱۴۰	۱۴۱	۱۴۲	۱۴۳	۱۴۴	۱۴۵
$g(x,y)$	۰	۰	۳۶.۴۲	۷۲.۸۵	۱۰۹.۲۸	۱۴۵.۷۱	۱۸۲.۱۴	۲۱۸.۵۷	۲۵۵
گرد شده	۰	۰	۳۶	۷۳	۱۰۹	۱۴۶	۱۸۲	۲۱۹	۲۵۵

از آنجا که در این مرحله دو مقدار به صفر نگاشت شده‌اند بنابراین فرکانس مقدار صفر حاصل جمع فرکانس این دو مقدار خواهد شد. از طرفی مقادیر مربوط به محور به عنوان آخرین مرحله، هیستوگرام و تصویر این بخش را نمایش می‌دهیم. برای این منظور از نمودار را رسم کرده و تصویر را همراه با پیکسل‌هایش می‌نویسیم. همچنین تصویر ایجاد شده در کد را نیز به نمایش می‌گذاریم:



همچنین مقادیر بدست آمده برای پیکسل‌های تصویر نیز به صورت زیر است:

109	0	0	182	182	182	219
109	73	73	146	146	182	182
73	255	255	219	146	146	255
109	109	109	182	146	109	182
36	73	109	109	146	182	219

به ازای مقدار ۱۳۷ به عنوان حد پایین نیز همان مراحل بالا را به ترتیب طی می‌کنیم:

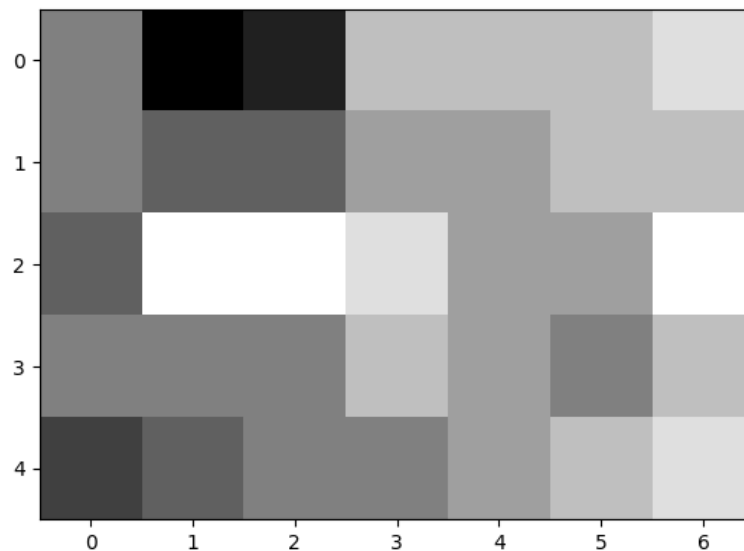
رابطه بدست آمده:

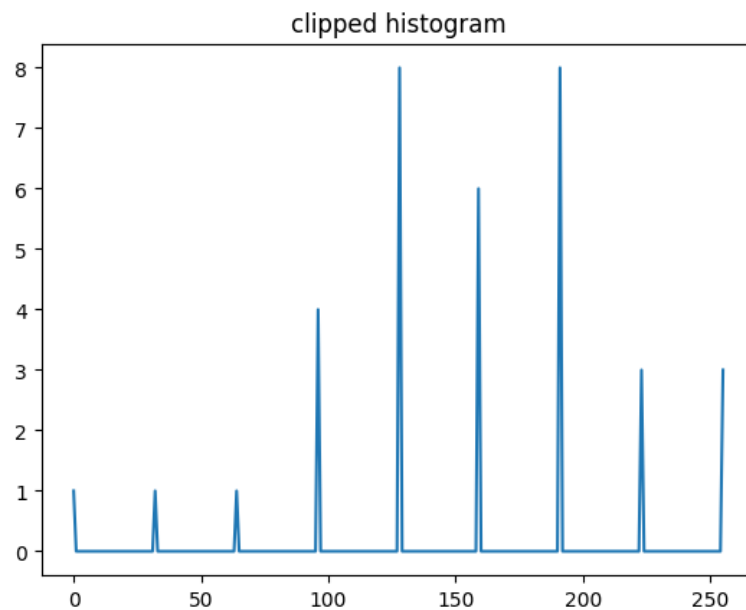
$$g(x,y) = \frac{(f'(x,y) - 137)}{7} (255)$$

بدست آوردن تابع نگاشت:

$f(x,y)$	۰	۱۳۸	۱۳۹	۱۴۰	۱۴۱	۱۴۲	۱۴۳	۱۴۴	۱۴۵
$f'(x,y)$	۱۳۸	۱۳۸	۱۳۹	۱۴۰	۱۴۱	۱۴۲	۱۴۳	۱۴۴	۱۴۵
$g(x,y)$	۰	۳۱.۸۷	۶۳.۷۵	۹۵.۶۲	۱۲۷.۵	۱۵۹.۳۷	۱۹۱.۲۵	۲۲۳.۱۲	۲۵۵
گرد شده	۰	۳۲	۶۴	۹۶	۱۲۸	۱۵۹	۱۹۱	۲۲۳	۲۵۵

رسم هیستوگرام و کشیدن تصویر: از آنجا که پیش از این نمودار برای راه حل اول برش هیستوگرام نمودار کشیده شده است، در این گام صرفاً نمودار موجود در کد نمایش گذاشته می‌شود.





128	0	32	191	191	191	223
128	96	06	159	159	191	191
96	255	255	223	159	159	255
128	128	128	191	159	128	191
64	96	128	128	159	191	223

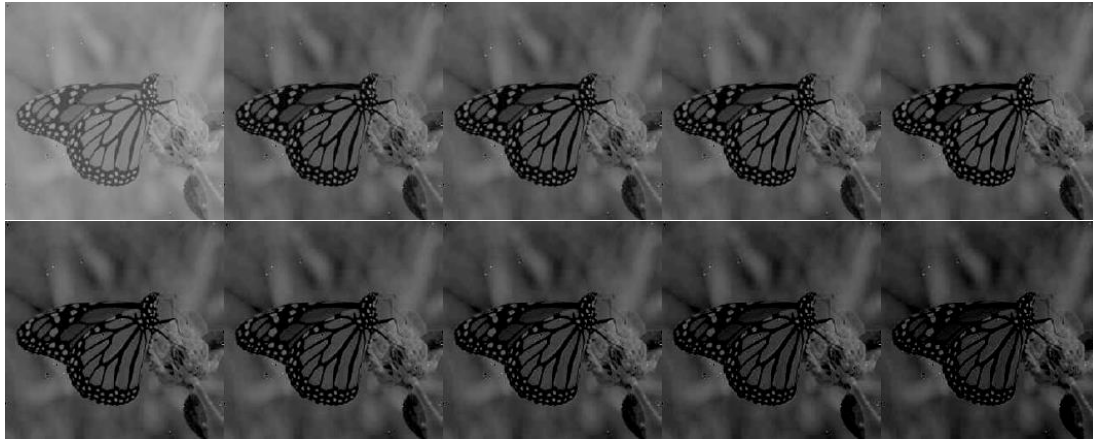
همانطور که مشاهده می‌شود با استفاده از برش هیستوگرام بر خلاف کشش هیستوگرام که به دلیل وجود داده پرت به خوبی عملیات پخش شدن مقدار و افزایش کنتراست صورت نگرفته بود، به خوبی کنتراست تصویر بهبود بخشیده شده است. همچنین در راه حل دوم از برش هیستوگرام که استفاده شده است، این کنتراست حتی بیش از پیش نیز افزایش یافته است.

بخش ج) نتایج به دست آمده برای تصویر به صورت زیر است:

• کشش هیستوگرام:



• برش هیستوگرام:



برای پیاده‌سازی برش هیستوگرام طبق چیزی که در کلاس گفته شده عمل شده است. یعنی یک درصد مشخص از مقادیر بالا و پایین کنار گذاشته شده‌اند و بر اساس سایر مقادیر عملیات برش هیستوگرام انجام شده است. همچنین درصد مورد استفاده از بالا چپ به پایین راست از ۰ شروع شده و با افزایش ۰.۰۱ در هر تصویر به ۰.۱ می‌رسد.

همانطور که واضح است در اثر استفاده از کشش هیستوگرام هیچ تغییری در تصویر مشاهده نمی‌کنیم و به عبارتی هیچ بهبودی نداریم. در حالی که بر اثر استفاده از مقدار مناسبی برای برش هیستوگرام (در این مثال ۰.۰۱) می‌توانیم کنتراست تصویر را افزایش دهیم. دلیل این موضوع نیز وجود نویز در تصویر است که همانطور که مشخص در بعضی از نقاط به صورت نویزهای سفید و سیاه خود را نشان داده است. بنابراین در چنین شرایطی که نویز و داده پرت در تصویر وجود دارد، بهتر است که از برش هیستوگرام استفاده کنیم.

سوال ۲:

بخش الف) برای آنکه عملیات تطبیق هیستوگرام را روی تصویر src اعمال کنیم و شدت رنگ‌های موجود در آن را مشابه با تصویر ref کنیم، نیاز است که در ابتدا cdfهای هر یک از تصاویر را محاسبه کنیم و سپس شدت رنگ‌هایی از تصویر ref را که مقدار CDF برابر با CDF شدت رنگ‌های موجود در تصویر src دارند را در تصویر src جایگزین کنیم. همچنین فرض می‌کنیم که ۸ سطح برای شدت رنگ داریم.

- هیستوگرام و cdf در src:

مقدار	۰	۱	۲	۳	۴	۵	۶	۷
فرکانس	۸	۳۲	۲۴	۰	۰	۰	۰	۰
cdf	۸	۴۰	۶۴	۶۴	۶۴	۶۴	۶۴	۶۴

- هیستوگرام و cdf در ref:

مقدار	۰	۱	۲	۳	۴	۵	۶	۷
فرکانس	۰	۰	۸	۸	۸	۱۶	۸	۱۶
cdf	۰	۰	۸	۱۶	۲۴	۴۰	۴۸	۶۴

حال باید به دنبال Xهایی بگردیم که برای آن‌ها Yهایی وجود داشته باشد، به طوری که:

$$CDF_{src}(x) = CDF_{ref}(y)$$

در نتیجه با بررسی کردن مقادیر به تابع نگاشت زیر دست خواهیم یافت:

x	۰	۱	۲	۳	۴	۵	۶	۷
y	۲	۵	۷	۷	۷	۷	۷	۷

بنابراین تصویر نهایی به صورت زیر می‌شود:

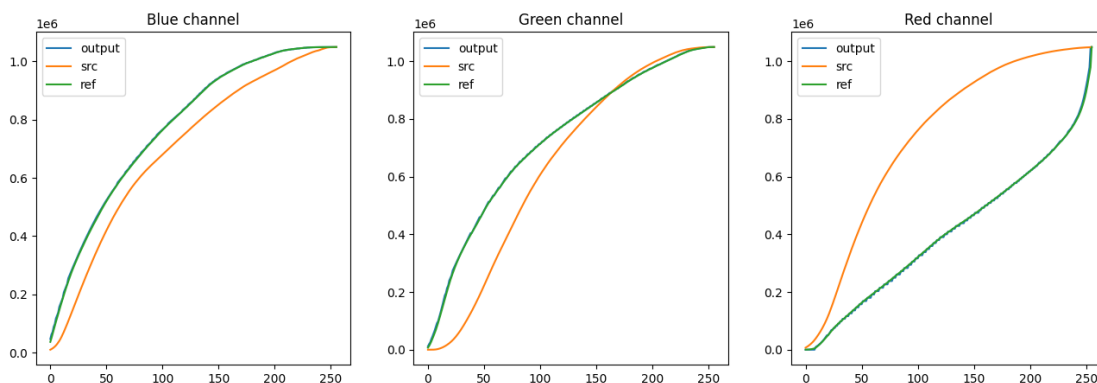
2	2	2	2	2	2	2	2
5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5
7	7	7	7	7	7	7	7
7	7	7	7	7	7	7	7
7	7	7	7	7	7	7	7
5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5

بخش ب) پیاده‌سازی‌ها در نوت‌بوک مربوطه آورده شده است و در این جا صرفاً تحلیل خروجی قرار گرفته است.

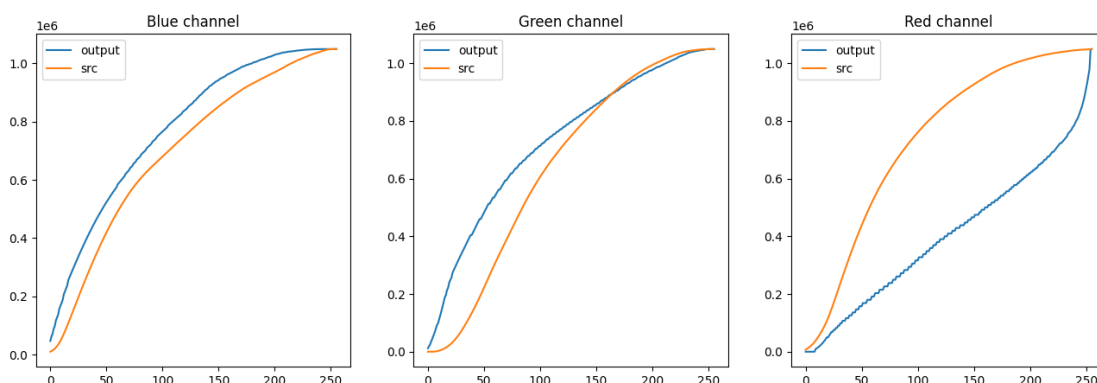


تصویر خروجی حاصل از تبدیل تصویر src در بالا آورده شده است. همانطور که مشخص است تصویر حاصل درست مانند تصویر reference دارای تعداد بیشتری پیکسل با مقدار بالا در کانال قرمز است. به عبارتی با بررسی این سه تصویر، می توان دریافت که به خوبی ترکیب رنگ مشابهی با تصویر reference بدست آورده است در حالی که جزئیات مربوط به تصویر مبدا را حفظ کرده است.

برای بررسی بهتر می توان CDF مربوط به یک از کانال های هر سه تصویر را در کنار یکدیگر رسم کرد. نتیجه حاصل به صورت زیر است:

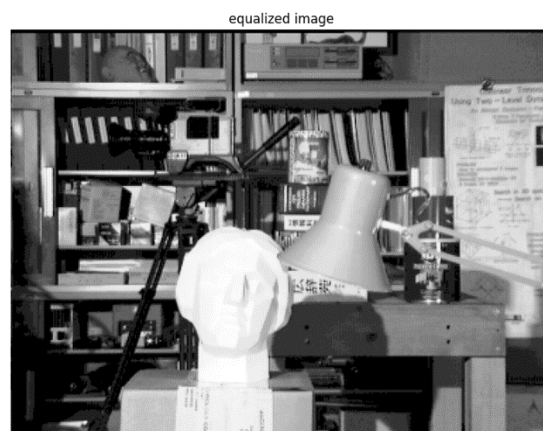
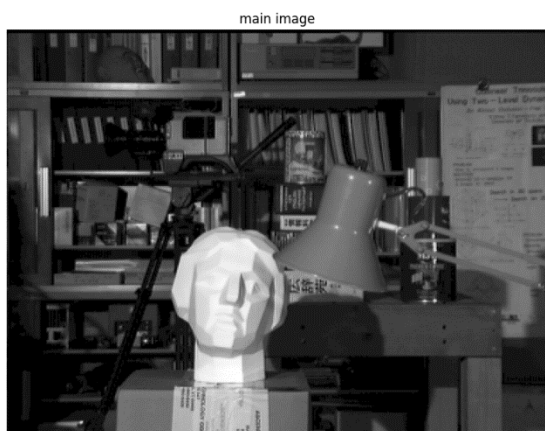


نمودار به رنگ آبی که در تصویر بالا مشخص نیست مربوط به تصویر خروجی است که در زیر نمودار سبز رنگ که مربوط به reference است قرار گرفته است که این نشان دهنده انجام موفقیت آمیز عملیات تطبیق هیستوگرام بوده است. برای تشخیص بهتر این موضوع نمودار زیر را نیز رسم می کنیم که می توان با مقایسه آن با نمودارهای بالا توضیحات گفته شده را بهتر متوجه شد.



سوال ۳:

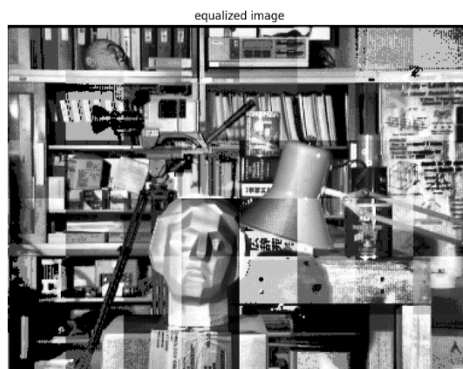
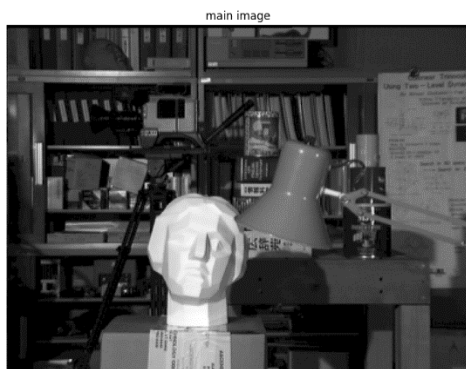
بخش الف) نتیجه بدست آمده به صورت زیر است:



در تصویر اولیه، به طور کلی بخش‌های با شدت رنگ پایین (تیره‌تر) بیشتر هستند نسبت به بخش‌های با شدت رنگ بالا (روشن‌تر). در نتیجه آن و پس از اعمال متعادل‌سازی هیستوگرام تابع تبدیلی که بدست می‌آوریم مربوط به کل تصویر بوده و سعی در روشن‌تر کردن تصویر با توجه به نکته گفته شده می‌کند بدون اینکه جزئیات محلی را در نظر بگیرد. در نتیجه این عمل، نواحی‌ای که به صورت یکنواخت دارای شدت رنگ بالا بوده‌اند (روشن بوده‌اند) روشن‌تر شده و حتی جزئیاتی که پیش از این در آن‌ها قابل مشاهده بوده نیز از دست رفته است. مثالی برای این نوع ناحیه در تصویر بالا مجسمه موجود در پایین تصویر است که با توجه به روشن بودن آن، بعد از اعمال متعادل‌سازی هیستوگرام جزئیات خود را از دست داده است.

بخش ب)

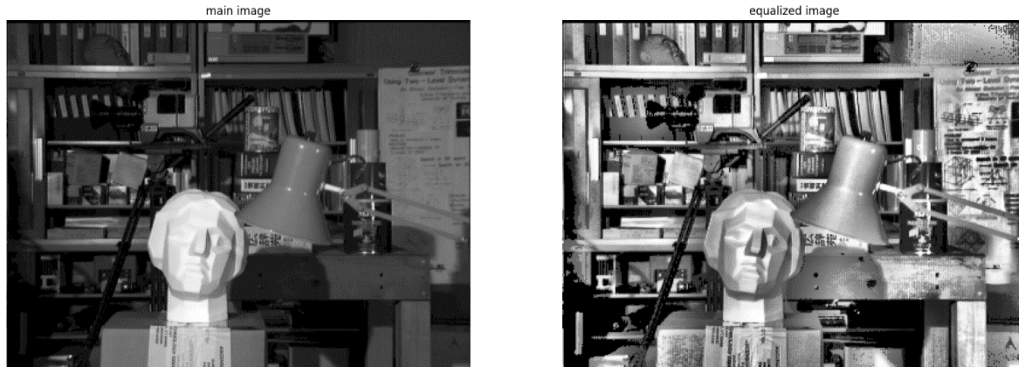
• الگوریتم AHE نوع اول: نتیجه بدست آمده به صورت زیر است:



با استفاده از این روش نیاز است که تعدادی مشبک را تعریف کنیم. هدف از تعریف این مشبک‌ها، اعمال تعدیل‌سازی هیستوگرام در داخل یک مشبک است که در نتیجه آن برای هر مشبک یک نگاشت ایجاد می‌شود. به همین دلیل است که این روش برای پیاده‌سازی ساده بوده و هزینه محاسباتی چندانی نیز ندارد. همچنین در داخل یک مشبک، کنتراست به طور خوبی بهبود پیدا می‌کند در صورتی که نویز در آن بخش وجود نداشته باشد. اما اگر در داخل همین مشبک، تصویر به صورت یکنواخت در حوزه سطوح تیره یا روشن قرار گرفته باشد و از طرفی مقداری نویز در این بخش وجود داشته باشد، همین امر باعث می‌شود که برای آن مشبک بعد از اعمال نگاشت، نویزها بولدتر شده و وضوح

بیشتری پیدا کنند. مانند ناحیه بالا راست در تصویر بالا. همچنین از طرفی اگر که یک مشبک دارای شدت رنگ‌های متفاوت با مشبک‌های همسایه خود باشد، ناحیه مرزی‌ای میان دو این دو در تصویر نهایی ایجاد می‌شود که کیفیت تصویر را تحت تاثیر قرار می‌دهد.

- الگوریتم AHE نوع دوم: نتیجه بدست آمده به صورت زیر است:



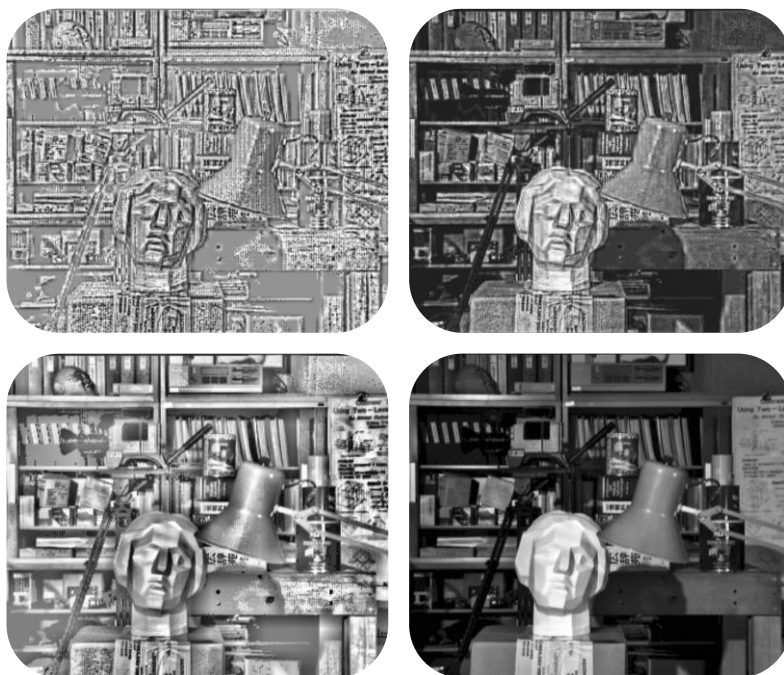
در این روش به جای آنکه یک تابع نگاشت برای هر ناحیه پیدا کنیم، یک تابع نگاشت برای هر پیکسل پیدا می‌کنیم. برای این منظور نیاز است که برای هر پیکسل، مشبکی به مرکز آن را در نظر بگیریم و یک تابع را بدست آوریم و در پایان تابع نگاشت بدست آمده را تنها بر روی همین پیکسل اعمال کنیم. در نتیجه، همین موضوع باعث کند بودن اجرای الگوریتم و نیز دشوارتر شدن پیاده‌سازی آن نسبت به AHE نوع اول می‌شود. ولی مشکل AHE نوع اول یعنی نمایان بودن مرزهای میان مشبک‌ها از میان رفته است. با اینحال همچنان مشکل بولد شدن نویز در نواحی یکنواخت همراه با نویز وجود دارد.

- الگوریتم CLAHE: نتیجه بدست آمده به صورت زیر است:



در این روش مانند روش قبلی برای هر پیکسل یک نگاشت بدست می‌آوریم، اما تفاوتی که در این روش وجود دارد در این است که در هر ناحیه مقداری نرمال‌سازی انجام می‌دهیم؛ به این معنا که در PDF بدست آمده برای هر ناحیه یک عمل محدودسازی و افزودن مقدار مازاد به تمام شدت‌های رنگی را انجام می‌دهیم که در نتیجه این عمل، CDF ای که برای هر ناحیه بدست می‌آوریم به صورت همواره صعودی خواهد بود و شکل تقریباً خطی به خود می‌گیرد. در نتیجه تصویری که بدست می‌آوریم هموارتر شده و نویزها مانند روش‌های قبلی بولد نخواهند شد. درست مانند آنچه که در تصویر بالا بدست آورده‌ایم. همچنین مشاهده می‌شود که کنتراست تصویر نیز به بهبود پیدا کرده است. مشکل این روش مانند روش پیشین، دشوارتر شدن پیاده‌سازی و نیز کاهش سرعت اجرای آن است.

بخش ج) نتایج بدست آمده به صورت زیر هستند:



با بررسی این تصاویر به نتایج زیر می‌رسیم:

- **clip_limit**: این پارامتر آستانه محدودیت کنتراست را در الگوریتم CLAHE تعیین می‌کند و مشخص می‌کند که چقدر کنتراست را می‌توان در تصویر افزایش داد. به عبارتی آستانه‌ای است برای همان تعداد تکرارها در نمودار هیستوگرامی که برای تصاویر بدست آوردیم. هر چه مقدار این پارامتر بیشتر باشد، میزان کنتراست تصویر بیشتر شده و جزئیات بیشتر به خصوص نویزها به چشم می‌خورند. همچنین مقدار پایین آن نیز باعث تقویت شدن نویزها می‌شود چرا که مقداری به CDF آنها نیز اضافه می‌شود.
- **grid_size**: این پارامتر اندازه پنجره مورد نظر جهت انجام پردازش به منظور محاسبه نگاشت مربوط به یک پیکسل را مشخص می‌کند که هر چه بزرگتر باشد تعداد بیشتری پیکسل را در محاسبه هیستوگرام وارد می‌کند و باعث می‌شود که تصویر بدست آمده روشن‌تر شده و همچنین کنتراست بیشتر به چشم بخورد و به نوعی تصویر شارپ‌تر شود.

بنابراین به طور کلی هر چه این دو متغیر بزرگ و بزرگ‌تر شوند، کنتراست تصویر بیش از پیش بیشتر می‌شود و تصویر بدست آمده شارپ‌تر می‌شود. در حالی که با انتخاب اندازه مطلوب این دو پارامتر تصویر بدست آمده، کیفیت مطلوبی خواهد داشت مانند تصویر پایین راست که با سایز پنجره ۱۶ در ۱۶ و آستانه ۲ بدست آمده است.

سوال ۴:

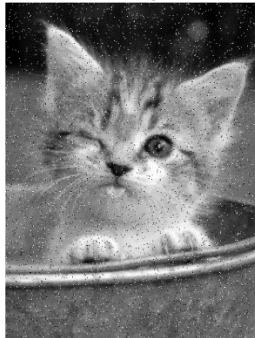
بخش الف) نتیجه اعمال نویز نمک و فلفل، به صورت زیر است:

cat with noise



بخش ب)

main image



Averaging Blurring



Median Blurring



Gaussian Blurring



تحلیل تاثیر هر یک از فیلترها به شرح زیر است:

- **Average Blurring**: این فیلتر به سادگی قابل پیاده سازی است به طوری که در آن کافیست به جای هر پیکسل میانگین پیکسل های همسایه آن را قرار دهیم. اغلب برای کاهش نویزهایی که به طور یکنواخت در تصویر پخش شده اند استفاده می شود اما با اینحال می توان با استفاده از آن نویز نمک و فلفل را نیز کاهش داد. بنابراین مشکلی که این فیلتر دارد از بین بردن جزئیات بوده و همچنین امکان جایگزینی یک پیکسل با مقداری که تا پیش از این در تصویر وجود نداشته است.
- **Median Blurring**: این فیلتر نیز مانند فیلتر پیشین به سادگی و با محاسبه و جایگزینی یک پیکسل با میانه پیکسل های همسایه اش بدست می آید. در نتیجه در این روش برخلاف روش پیشین مقدار جدیدی معرفی نمی شود و مقادیری که جایگزین مقادیر پیشین می شوند، مقادیری اند که از پیش در تصویر وجود داشته اند. با استفاده از این روش می توان نویزها را به ویژه نویز نمک و فلفل را از بین برد اما با اینحال تصویر تا حدودی تیزتر شده و در نواحی ای ممکن است که اطلاعات موجود در تصویر از بین بروند. مانند سیبیل ها و لب گربه که در این مثال در مقایسه با سایر فیلترها تاثیر کمتری از آن ها برجای مانده است.
- **Gaussian Blurring**: این فیلتر از توزیع گاوسی برای محاسبه میانگین وزن دار در یک همسایگی هر پیکسل استفاده می کند به طوری که وزن بیشتری را به مقدار پیکسل های مرکزی نسبت می دهد. متوسط گیری در این روش به طرز نرم تری اتفاق می افتد و اطلاعات مربوط به هر پیکسل بهتر از روش **Average Blurring** حفظ می شود (بدلیل همان وزن دار بودن میان گیری). همچنین لبه ها را بهتر از یک فیلتر متوسط گیر با اندازه مشابه حفظ می کند. درجه هموارسازی نیز با انحراف معیار گاوسی تعیین می شود. این معیار کنترل کننده این امر است که تا چه اندازه شعاع بیشتری نسبت به مرکز فیلتر استفاده شود. به عبارتی وزنی که به همسایگان یک نقطه مرکزی می دهد بیشتر می شود. بنابراین به طور کلی این فیلتر رفتاری مشابه با **Average Blurring** دارد با این تفاوت که در صورت تنظیم مقدار سیگمای آن به خوبی می توان اطلاعات کمتری را از دست داد. در این مثال نیز حاصل هردو فیلتر تا حد خوبی یکسان شده است.

سایز کرنل بر روی هر سه مورد به طور مشابهی تاثیر خود را اعمال می کند که در زیر شرح داده شده است.

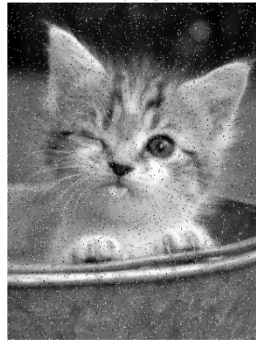
- **سایز کرنل**: هر چه سایز کرنل بیشتر باشد، باعث می شود که محدوده وسیع تری در هنگام اعمال فیلتر مورد استفاده قرار گیرد و در نتیجه آن نویزهای بیشتری از تصویر حذف شوند. اما در عوض در این وضعیت، جزئیات بیشتری از تصویر از دست می روند و تصویر تیزی خود را از دست می دهد. در حالی که با کاهش آن، با اینکه نویزهای کمتری ممکن است که حذف شوند اما با اینحال جزئیات بیشتری در تصویر باقی می ماند و تصویر تیزی خود را مانند حالت اولیه از دست نمی دهد. همچنین از طرفی با بزرگ تر شدن سایز کرنل با توجه به سایز تصویر ممکن است که از نظر محاسباتی نیز هزینه اجرای الگوریتم بیشتر شود.

بخش ج) نتایج بدست آمده از توابع آماده در زیر آورده شده اند. تحلیل نتایج به این شرح است:

- خروجی بدست آمده: خروجی ای که از هر دو نوع از توابع آماده و نیز کتابخانه **OpenCV** بدست آمده است، یکسان می باشد.

- عملکرد برنامه و سرعت اجرا: از این نظر توابع پیاده‌سازی شده از نظر سرعت به شدت کندتر از توابع نظیر خود در کتابخانهٔ OpenCV هستند به طوری که اجرای هر کدام از آن‌ها ۱۰ ثانیه زمان خواهد برد. (با استفاده از سخت‌افزاری که Google Colab در اختیارمان می‌گذارد).
- سادگی استفاده: هر دو نوع از توابع به سادگی قابل فراخوانی هستند چرا که پارامترهای یکسانی برای هر دو قرار گرفته است. بنابراین در صورتی که نیاز به جایگزینی این دو نوع از توابع با یکدیگر وجود داشته باشد، به راحتی می‌توان این کار را با چند تغییر ساده انجام داد.

noisy image



Averaging Blurring



Median Blurring



Gaussian Blurring



سوال ۵:

قَالَ لَهُمْ: طَبِّقُوا فِيهِمَا اَللَّيْلَةَ اَوْ يَوْمَهُ اسْتَأْذِنْتُ لَكُمْ مِنْ صَوْتِ زَمَرٍ اسْتَأْذِنْتُ

$$L^2 P(w) = \begin{array}{ccc} & 0 & 1 & 0 \\ \hline & 1 & -4 & 0 \end{array}$$

با اعمال این مدل بر روی تصویر داده شده نتیجه تینت (تصویر) را خواهیم داشت. همین برای تینت
رایج تصویرهای تغییر یافته از padding از نوع reflect با اندازه 1 و اما از هر طرف استفاده
می‌کنیم.

حال جيت انجام عمليات لايلاسين از الاست جيت اخلاص ليم و م صا لاسين صا و اس صا
ص ليم. بعنوان مثال عنصر $(0,0)$ از سيم لايلاسين م صورت زير خلاصه شد:

$$\begin{aligned} & \vec{r}(0,0) \cdot \vec{a} = 10 \cdot 10 + 10 \cdot 0 + 0 \cdot 1 + 0 \\ & \quad 10 \cdot 10 + 10 \cdot 0 + 0 \cdot 1 = (10 \times 1) + (10 \times 1) + (10 \times -4) \\ & \quad 10 \cdot 10 + 10 \cdot 0 + 0 \cdot 1 + 0 \quad + (10 \times 1) + (10 \times 1) \\ & \quad = 10 + 10 + (-40) + (10) + (10) \\ & \quad = 0 \end{aligned}$$

به ازای α خانه هائیکه مقدار صفر به دست خواهیم آمد مثلاً ۵ خانه مقدار ۱۲ در بالای α است و جمع
راست و مرکز مقدار این خانه ها در فته یازده. به عنوان مثال به ای خانه (۴، ۴) داریم.

$$\begin{array}{ccccccccc} & 10 & 10 & 10 & & 0 & 1 & 0 & \\ (F, G) \text{ is } 0 & 10 & 12 & 10 & \times & 1 & 4 & 0 & = (10 \times 1) + (10 \times 1) + (10 \times 1) \\ & 10 & 10 & 10 & & 0 & 1 & 0 & + (10 \times 1) + (12 \times (-4)) \\ & & & & & & & & = (-8) \end{array}$$

۴ خانه مسکن، یک مغازه و خواهری فقیر، صاحبان مسجون به بی (۲۰۰۰) خانه (۴۳) در زیر اعمده

$$\begin{array}{ccccccc} (F, \varphi) \text{ is } ? & 10 & 10 & 10 & 0 & 1 & 0 \\ & 10 & 10 & 10 & \times & 1 & -4 & 0 & = (10 \times 1) + (10 \times 1) + (10 \times 1) \\ & 10 & 12 & 10 & & 0 & 1 & 0 & + (10 \times -4) + (10 \times 1) \\ & & & & & & & & = 0 \end{array}$$

Subject:

بنا بر این نتیجه لایلاسیون به صورت زیر خواهد بود:

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	2	0	0	0
0	0	0	2	-8	2	0	0
0	0	0	0	2	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

برای بررسی امدن تصویرهای نیز از رابطه $\nabla^2 F(x,y) = F(x,y) + C$ و $g(x,y) = (1-C)$ استفاده می‌کنیم. در نتیجه کافی است که از تصویر اول لایلاسیون را کم کنیم.

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	8	10	10	10
10	10	10	2	18	8	10	10
10	10	10	10	8	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10

سوال ۶:

Subject:

سوال ۶

برای محاسبه تبدیل فوریه از رابطه زیر استفاده می‌کنیم:

$$F(u, v) = \sum_{n=0}^1 \sum_{y=0}^1 f(n, y) e^{-2\pi j (u n/2 + v y/2)}$$

داریم:

$$u=0, v=0 \rightarrow F(0,0) = \sum_{n=0}^1 \sum_{y=0}^1 f(n, y) \times e^{0} = 1 + 2 + 1 + 2 = \boxed{6}$$

$$u=0, v=1 \rightarrow F(0,1) = \sum_{n=0}^1 \sum_{y=0}^1 f(n, y) \times e^{-2\pi j (y/2)}$$

$$\begin{aligned} e^{j\pi} &= -1 \\ e^0 &= 1 \end{aligned} \quad \begin{aligned} &= f(0,0) \times e^0 + f(0,1) \times e^{-j\pi} + f(1,0) \times e^0 + f(1,1) \times e^{-j\pi} \\ &\Rightarrow F(0,1) = 1 + 2 \times (-1) + 2 + 1 \times (-1) = \boxed{0} \end{aligned}$$

$$u=1, v=0 \rightarrow F(1,0) = \sum_{n=0}^1 \sum_{y=0}^1 f(n, y) \times e^{-2\pi j (n/2)}$$

$$\begin{aligned} e^{-j\pi} &= -1 \\ e^0 &= 1 \end{aligned} \quad \begin{aligned} &= f(0,0) \times e^0 + f(0,1) \times e^0 + f(1,0) \times e^{-j\pi} + f(1,1) \times e^{-j\pi} \\ &\Rightarrow F(1,0) = 1 + 2 + 2 \times (-1) + 1 \times (-1) = \boxed{0} \end{aligned}$$

$$u=1, v=1 \rightarrow F(1,1) = \sum_{n=0}^1 \sum_{y=0}^1 f(n, y) \times e^{-2\pi j (n/2 + y/2)}$$

$$\begin{aligned} e^{j\pi} &= -1 \\ e^{-2\pi j} &= 1 \\ e^0 &= 1 \end{aligned} \quad \begin{aligned} &= f(0,0) \times e^0 + f(0,1) \times e^{-j\pi} + f(1,0) \times e^{-j\pi} \\ &\quad + f(1,1) \times e^{-2\pi j} \\ &\Rightarrow F(1,1) = 1 + 2 \times (-1) + 2 \times (-1) + 1 \times (1) \\ &= \boxed{-2} \end{aligned}$$

بنابراین خواهیم داشت:

$$F = \begin{bmatrix} 6 & 0 \\ 0 & -2 \end{bmatrix}$$