University: Sharif University of Technology

Department: Electrical Engineering

Course Name: Medical Signal and Image Processing Lab

Lab 7 Report

Student Name: Ali Shahbazi, Zahra Kavian, MohammadReza Safavi

Student ID: 98101866, 98102121, 98106701

Instructor: Dr. Sepideh Hajipour

Academic Semester: 2023 Spring

فهرست مطالب

۱ بخش تئوری ۱

۲ بخش شبیه‌سازی ۲

فهرست تصاویر

۱ تصویر t1 . . . ۲

۲ اندازه و فاز تبدیل فوریه سطر ۱۲۸ام . . . ۲

۳ تصویر و تبدیل فوریه دوبعدی . . . ۳

۴ ماتریس‌های G و F و کانولوشن آن‌ها . . . ۳

۵ کانولوشن تصویر pd و ماتریس G . . . ۴

۶ تصویر ct و zoom-in شده‌ی آن . . . ۴

۷ تصویر اصلی و شیفت یافته در حوزه فرکانس . . . ۵

۸ مقایسه اندازه و فاز تصویر قبل و بعد از شیفت فرکانسی در حوزه زمان . . . ۵

۹ اندازه تبدیل فوریه کرنل . . . ۶

۱۰ مقایسه تصویر اصلی و چرخش یافته . . . ۶

۱۱ مقایسه تبدیل فوریه تصویر اصلی و تصویر چرخش یافته . . . ۷

۱۲ چرخش حوزه فرکانس . . . ۷

۱۳ ترسیم تصویر اصلی به همراه مشتق عمودی و افقی و گرادیان تصویر . . . ۸

۱۴ یافتن لبه‌های تصویر به کمک الگوریتم‌های sobel و canny . . . ۹

۱ بخش تئوری

اثبات.

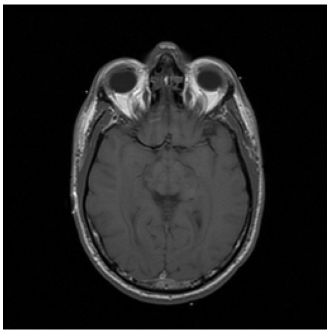
$$
\begin{aligned}
f(x) &= \sum_{k=-m}^{m} c_k \left(\cos\left(\frac{2\pi kx}{N}\right) + i\sin\left(\frac{2\pi kx}{N}\right)\right) \\
&= \sum_{k=-m}^{-1} c_k \left(\cos\left(\frac{2\pi kx}{N}\right) + i\sin\left(\frac{2\pi kx}{N}\right)\right) + c_0 + \sum_{k=1}^{m} c_k \left(\cos\left(\frac{2\pi kx}{N}\right) + i\sin\left(\frac{2\pi kx}{N}\right)\right) \\
&= \underbrace{c_0}_{a_0} + \sum_{k=1}^{m} (\underbrace{c_k + c_{-k}}_{a_k}) \cos\left(\frac{2\pi kx}{N}\right) + \sum_{k=1}^{m} \underbrace{i(c_k - c_{-k})}_{b_k} \sin\left(\frac{2\pi kx}{N}\right)
\end{aligned}
$$

$$
\left.\begin{aligned} a_k &= c_k + c_{-k} \\ b_k &= ic_k - ic_{-k} \end{aligned}\right\} \rightarrow \left.\begin{aligned} a_k &= c_k + c_{-k} \\ ib_k &= c_{-k} - c_k \end{aligned}\right\} \rightarrow \begin{aligned} c_k &= \frac{a_k - ib_k}{2} \\ c_{-k} &= \frac{a_k + ib_k}{2} \end{aligned}
$$

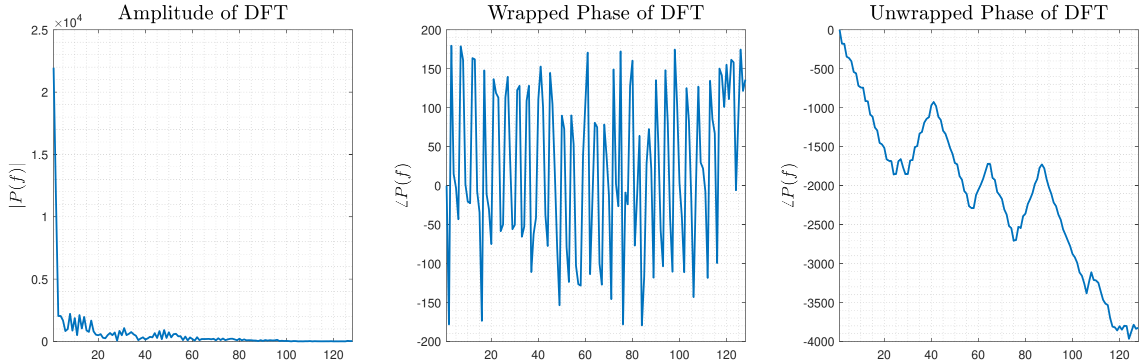
□

۲ بخش شبیه‌سازی

۱) تصویر مربوطه در شکل ۱ آمده است. برای هر پیکسل این تصویر تنها یک عدد نمایانگر روشنایی آن است و به اصطلاح تک‌کاناله است. همچنین با دستور *fft()* تبدیل فوریه گسسته را محاسبه می‌کنیم و سپس اندازه *abs()*، و فاز *angle()*، آن را رسم می‌کنیم (شکل ۲).



شکل ۱: تصویر t1

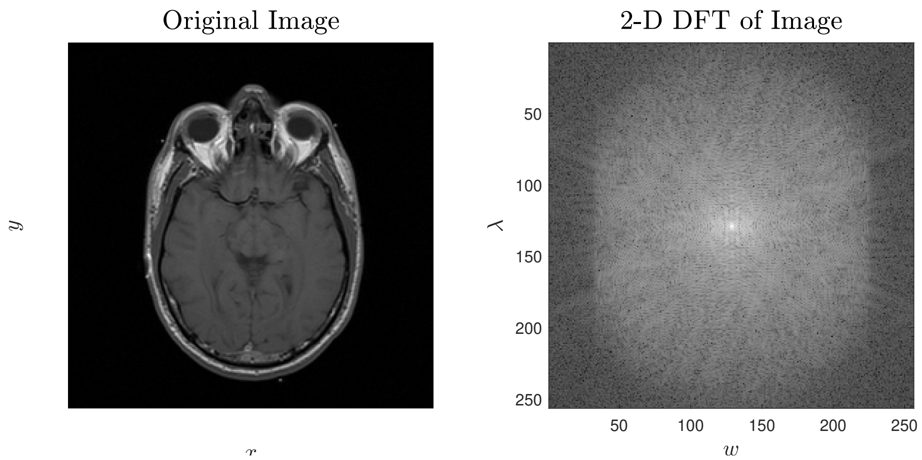


شکل ۲: اندازه و فاز تبدیل فوریه سطر ۱۲۸ام

از آنجایی که سطر ۱۲۸ام شامل مقادیر زیادی صفر (نقطه‌ی سیاه) پشت سر هم است یعنی تغییرات کم در سیگنال دیده می‌شود. همچنین مقادیر روشنایی همگی مثبت هستند که منجر به ایجاد یک مولفه DC در تبدیل فوریه می‌شود. بنابراین اندازه‌ی تبدیل فوریه باید اولا در فرکانس نزدیک به صفر پیک بزند و دوما در فرکانس‌های کم مقدار بیشتری داشته باشد که همینطور نیز مشاهده می‌شود. درباره فاز تبدیل فوریه می‌توان گفت که شیفت مولفه‌های سینوسی تصویر را تعیین می‌کند. هنگامی که فاز صفر باشد، تمام سینوسی‌ها در یک مکان align می‌شوند و تصویر حاصل ساختاری کاملا متقارن بدست می‌آید که هیچ شباهتی به تصویر اولیه ندارد. بنابراین اندازه‌ی تبدیل فوریه قدرت سیگنال و مقدار روشنایی را تعیین می‌کند و فاز تبدیل فوریه چگونگی کنار هم قرار گرفتن سینوسی‌ها و پدیدآورنده‌ی تغییرات در روشنایی در تصویر است.

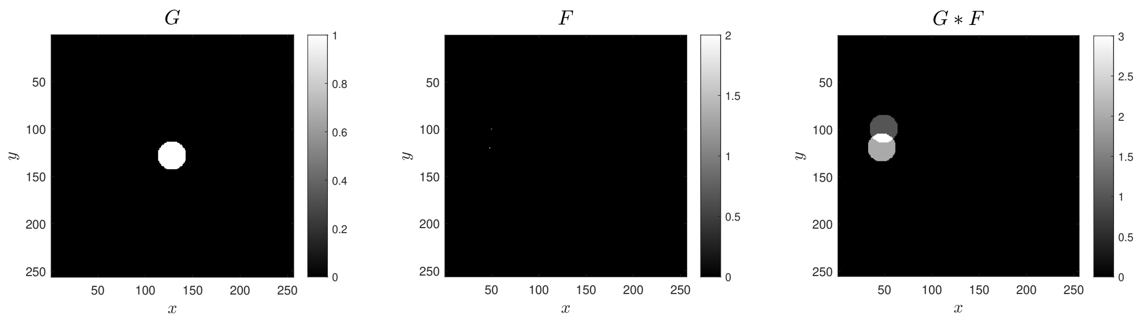
همچنین با استفاده از دستور *fft2()* تبدیل فوریه دوبعدی را حساب می‌کنیم. تابع *fftshift()* برای ورودی دوبعدی به این صورت عمل می‌کند که اگر تصویر را از نقطه‌ی وسط به چهار ناحیه ۱ و ۲ و ۳ و ۴ مطابق نواحی مثلثاتی تقسیم کنیم، جای ناحیه ۱ و ۳ و همچنین جای ناحیه ۲ و ۴ را عوض می‌کند (شکل ۳، دقت شود که مقادیر تبدیل فوریه در مقیاس لگاریتمی نمایش داده شده است).

همانند توضیحات یک‌بعدی، می‌توان تعمیم داد که مولفه‌ی DC در این تصویر بیشترین دامنه را دارد و هر چه به سمت فرکانس‌های بیشتر می‌رویم، قدرت تبدیل فوریه کمتر می‌شود. این اثر را می‌توان در هر دو بعد مشاهده کرد. نقطه‌ی وسط تبدیل فوریه دوبعدی نشانگر مولفه‌ی نزدیک به DC، و هر چه از مرکز دور می‌شویم مربوط به فرکانس‌های بیشتر است.



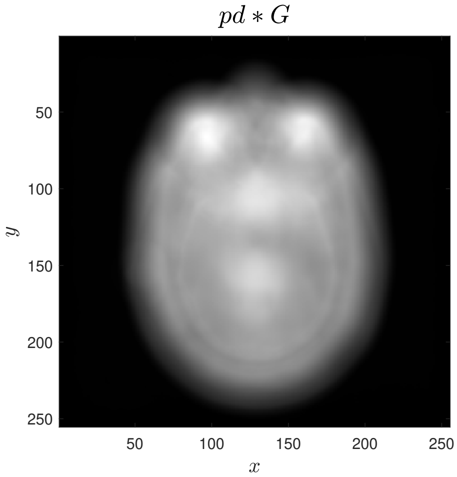
شکل ۳: تصویر و تبدیل فوریه دوبعدی

۲) با استفاده از معادله‌ی دایره به مرکز وسط تصویر، ماتریس G را می‌سازیم. همچنین ماتریس F نیز به سادگی ساخته می‌شود. به کمک دستور *conv2()* متلب، کانولوشن دوبعدی دو تصویر را محاسبه می‌کنیم. شکل ۴ این عملیات را نشان می‌دهد که مطابق آن، هر نقطه در ماتریس F مشابه یک تابع δ عمل کرده است و باعث شیفت دایره به آن نقطه شده. به همین صورت می‌توان تمام تصویر را مجموعی از پیکسل‌ها (همان تابع‌های δ) در نظر گرفت و در واقع هنگام کانولوشن ماتریس G با تصویر pd، تعدادی دایره با شعاع ۱۵ با یکدیگر جمع می‌شوند تا تصویر نهایی را بسازند. اگر شعاع دایره را بیشتر کنیم، مقدار blurr بیشتر می‌شود و اگر کمتر کنیم و به سمت ۱ میل دهیم، همان تصویر اولیه حاصل می‌شود (شکل ۵).

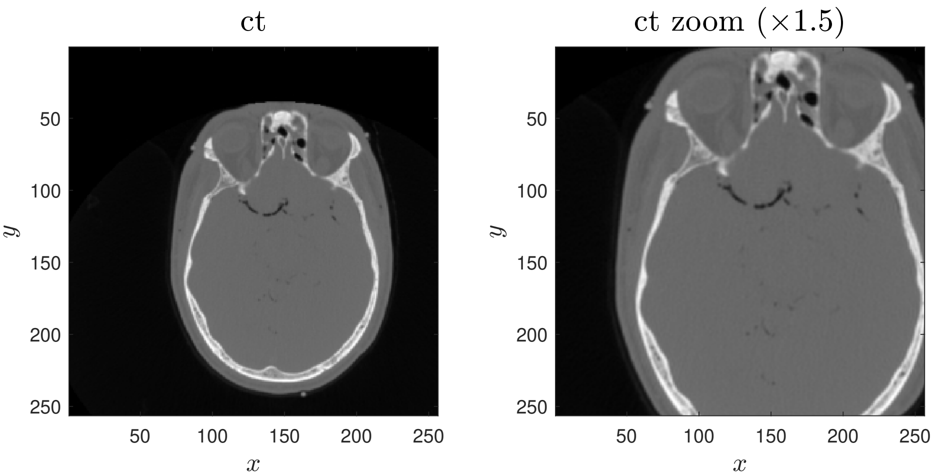


شکل ۴: ماتریس‌های G و F و کانولوشن آن‌ها

۳) این عملیات در واقع استفاده از همان خاصیت scaling تبدیل فوریه است. یعنی اگر یک سیگنال را در حوزه فرکانس فشرده کنیم، در حوزه زمان باز می‌شود. بنابراین اگر در حوزه‌ی فرکانس به اطراف تصویر مقادیر صفر اضافه کنیم (zero padding)، و بعد به حوزه زمان برگردیم تصویر ما scale شده است (شکل ۶).



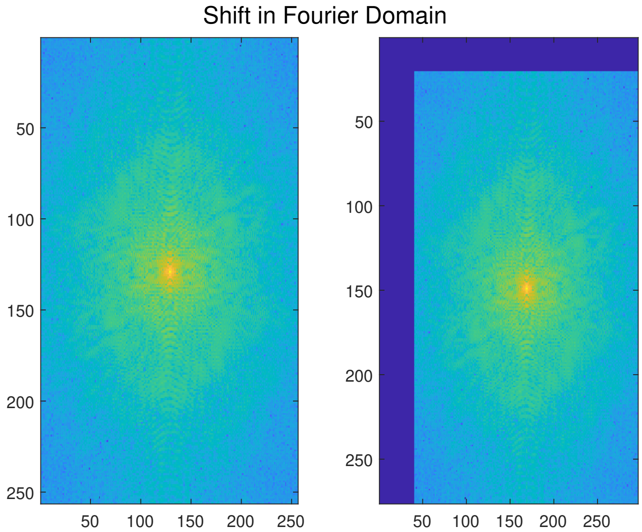
شکل ۵: کانولوشن تصویر pd و ماتریس G



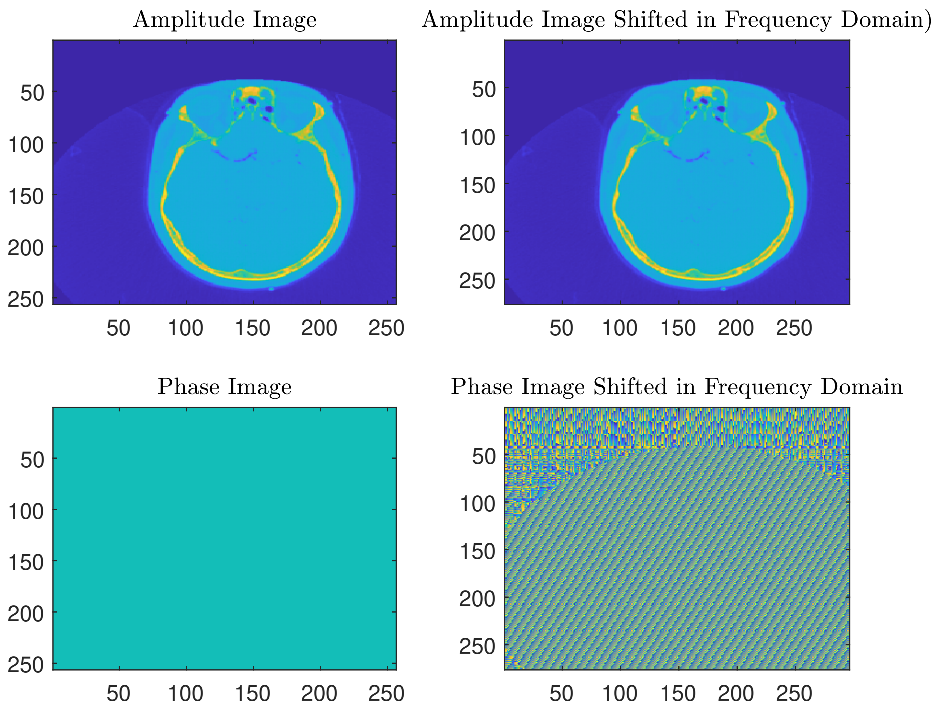
شکل ۶: تصویر ct و zoom-in شده‌ی آن

۴) بعد از تبدیل فوریه گرفتن از تصویر، در حوزه فرکانس ۲۰ واحد به سمت راست و ۴۰ واحد به پایین شیفت می دهیم (سمت چپ و بالای تصویر صفر قرار داده شده است، شکل ۷).

با تبدیل فوریه معکوس از تصویر حاصل شده به حوزه مکان برگشته، فاز و اندازه تصویر در شکل ۸ مشاهده می شود. طبق خواص تبدیل فوریه، شیفت حوزه فوریه معادل تغییر فاز در حوزه مکان است. بنابراین انتظار داریم اندازه تصویر بعد از شیفت فرکانسی تغییر نکند اما فاز تصویر به هم بریزد (۸).



شکل ۷: تصویر اصلی و شیفت یافته در حوزه فرکانس

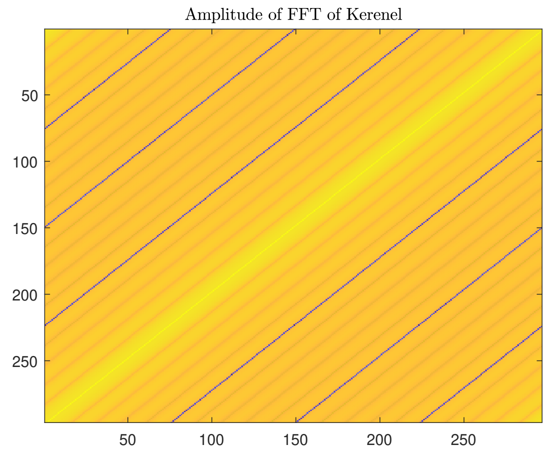


شکل ۸: مقایسه اندازه و فاز تصویر قبل و بعد از شیفت فرکانسی در حوزه زمان

کرنل شیفت دهنده به راست و پایین به شکل زیر تعریف می شود:

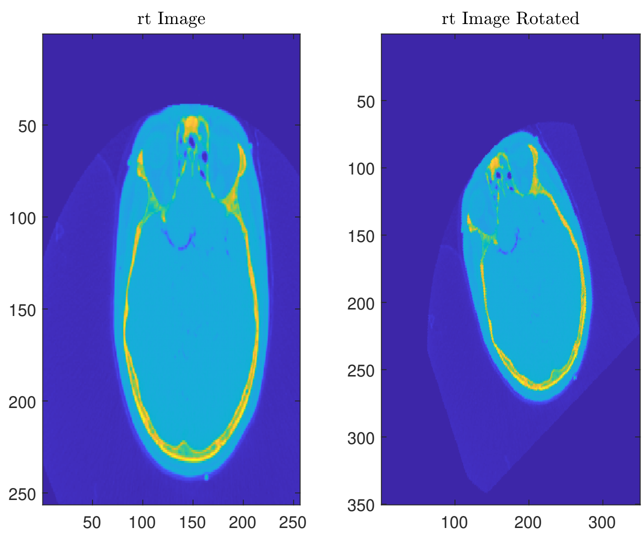
$$\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}^{20} \times \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}^{40} \tag{۱}$$

با تبدیل فوریه گرفتن از این کرنل، اندازه آن نشان داده شده است (شکل ۹).



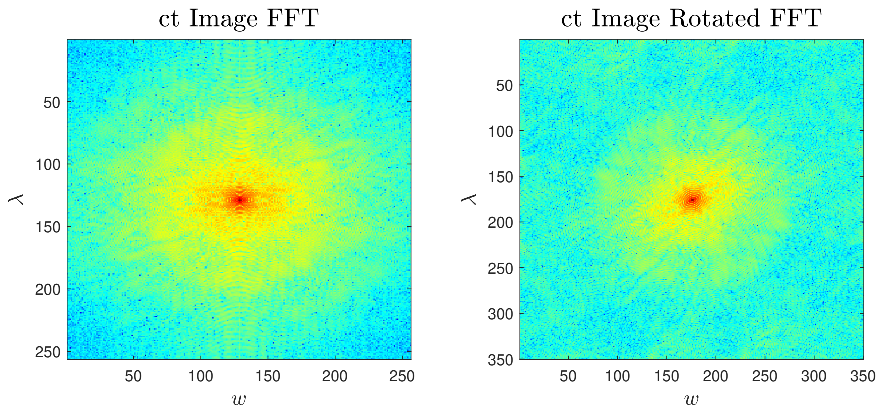
شکل ۹: اندازه تبدیل فوریه کرنل

۴-۲) با استفاده از دستور imrotate اسلایس اول تصویر به اندازه ۳۰ درجه در جهت خلاف عقربه های ساعت می چرخانیم. در شکل ۱۰ تصویر اصلی و چرخش یافته نشان داده می شود.



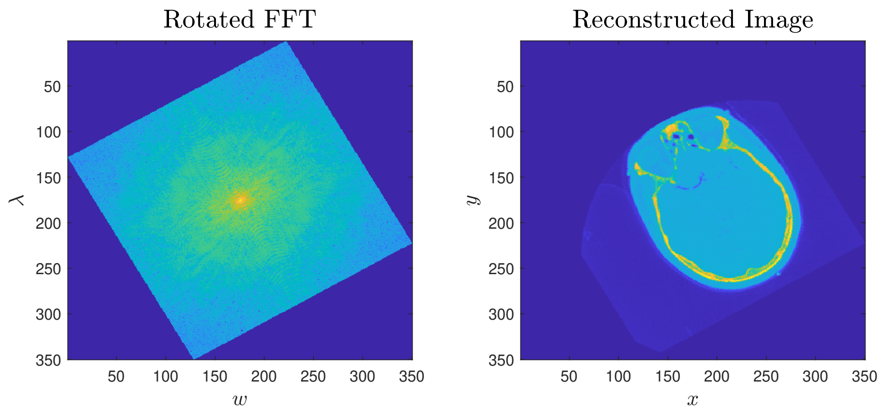
شکل ۱۰: مقایسه تصویر اصلی و چرخش یافته

صویر اصلی و چرخش یافته در شکل ۱۱ نشان داده شده است. چرخش در حوزه مکان معادل با ضرب شدن کرنل چرخش در حوزه فرکانس است. لذا اندازه تصویر چرخش یافته در حوزه فرکانس به اندازه کرنل چرخش تغییر می کند. اما تبدیل فوریه تصویر نیز به همان اندازه در حوزه مکان چرخیده است.



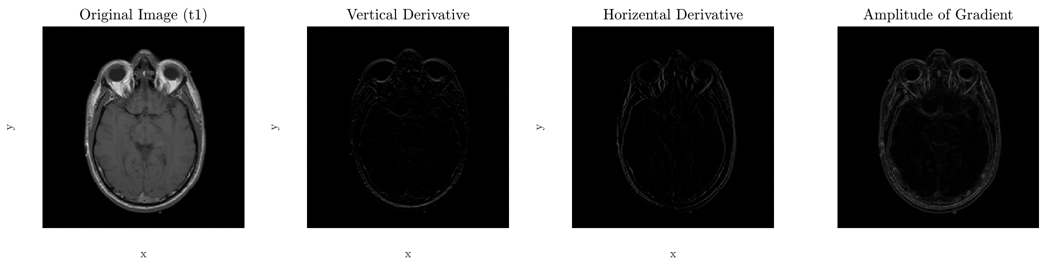
شکل ۱۱: مقایسه تبدیل فوریه تصویر اصلی و تصویر چرخش یافته

اگر تصویر در حوزه فرکانس به همان اندازه بچرخانیم و تبدیل فوریه معکوس بگیریم، معادلا تصویر در حوزه مکان به همان اندازه می چرخد (شکل ۱۲).



شکل ۱۲: چرخش حوزه فرکانس

۵) برای انجام این بخش، پس از خواندن عکس‌ها و تبدیل آن‌ها به ماتریس از جنس double ، با استفاده از تابع circshift ، تصاویر را یک بار یک واحد به بالا، یک بار یک واحد به پایین و بار دیگر به چپ و نهایتا به راست شیفت داده و نسخه‌های شیفت یافته را از هم کم می‌کنیم و حاصل را تقسیم بر ۲ می‌کنیم تا تقریب ما از مشتق central عمودی و افقی عکس به دست آید. در نهایت ماتریس‌ها را به uint۸ تبدیل کرده و عکس حاصل را ترسیم می‌کنیم. برای محاسبه گرادیان تصویر نیز مشتق عمودی و افقی را به صورت element-wise به توان ۲ رسانده، با هم جمع می‌کنیم و سپس ریشه دوم آن را می‌گیریم. نتیجه انجام این کارها در تصویر ۱۳ مشخص است. مشاهده می‌شود در تصویر مشتق عمودی، لبه‌های عمودی (یعنی مثلا خطوط افقی) آشکار شده‌اند و در تصویر مشتق افقی، لبه‌های افقی (مثلا خطوط عمودی) عکس را می‌بینیم. علت این موضوع هم آن است که اگر به طور مثال در عکس یک خط عمودی داشته باشیم، شیفت این تصویر به بالا و پایین چندان تصویر را تغییر نمی‌دهد و بنابراین اگر عکس شیفت یافته به بالا و عکس شیفت یافته به پایین از هم کم شوند، عملا خروجی صفر است پس لبه‌ای تشخیص داده نمی‌شود. به عکس اما اگر یک خط افقی داشته باشیم، شیفت به بالا و پایین باعث می‌شود این خط به بالا و پایین شیفت پیدا کند و تصویر حاصل تفاضل شیفت‌ها، یک خط افقی در خود داشته باشد پس در مشتق عمودی، خطوط افقی کشف می‌شوند که متناظر با لبه‌های عمودی هستند. در تصویر گرادیان نیز به نوعی کل لبه‌ها کشف شده‌اند چون اثر مشتق عمودی و افقی با هم جمع شده‌اند. پس در تصویر گرادیان، به نوعی لبه‌های مایل هم مشخص شده‌اند.



شکل ۱۳: ترسیم تصویر اصلی به همراه مشتق عمودی و افقی و گرادیان تصویر

۶) برای انجام این بخش، با استفاده از تابع edge ، که یک بار پارامتر ورودی آن را sobel و بار دیگر canny تعیین می کنیم، با استفاده از دو الگوریتم فوق لبه‌های تصویر را میابیم.

نتیجه انجام این کارها در تصویر ۱۴ مشخص است. مشاهده می‌شود هنگام استفاده از روش canny ، لبه‌های

بیشتری پیدا شده است. در واقع الگوریتم sobel مبتنی بر کانولوشن دو ماتریس با تصویر است که آن ماتریس‌ها یکی نقش مشتق عمودی و دیگری نقش مشتق افقی را دارد و نهایتا از این دو تصویر حاصل، تصویر نهایی تولید می‌شود. در مقابل الگوریتم canny الگوریتم پیچیده‌تری است و در چندین گام انجام می‌شود و به طور خلاصه ابتدا تا جایی که می‌تواند لبه‌های تصویر را پیدا می‌کند، سپس پیکسل‌ها را با پیکسل‌های اطراف مقایسه می‌کند و سپس پیکسل‌ها را با تمام تصویر مقایسه می‌کند و نهایتا تصمیم می‌گیرد کدام لبه‌ها مهم هستند و باید حفظ شوند و کدام لبه‌ها باید حذف شوند. در نهایت باید گفت الگوریتم canny به مراتب پیچیدگی محاسباتی بیشتری دارد اما مشاهده می‌شود لبه‌های بسیار بیشتری را در عکس یافته است. همچنین مشاهده می‌شود لبه‌هایی که الگوریتم canny پیدا کرده، انسجام بیشتری دارند و گویی واقعا یک سری خط به عنوان لبه یافته است. در حالی که الگوریتم sobel بعضا لبه‌هایی که پیدا کرده در حد چند پیکسل هستند و پیوستگی ندارند.

شکل ۱۴: یافتن لبه‌های تصویر به کمک الگوریتم‌های sobel و canny