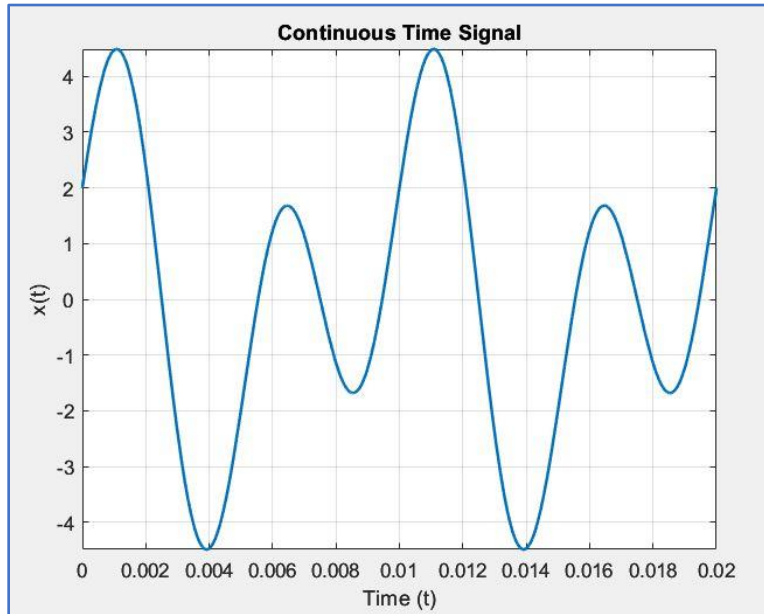


1. کمی نمونه برداری

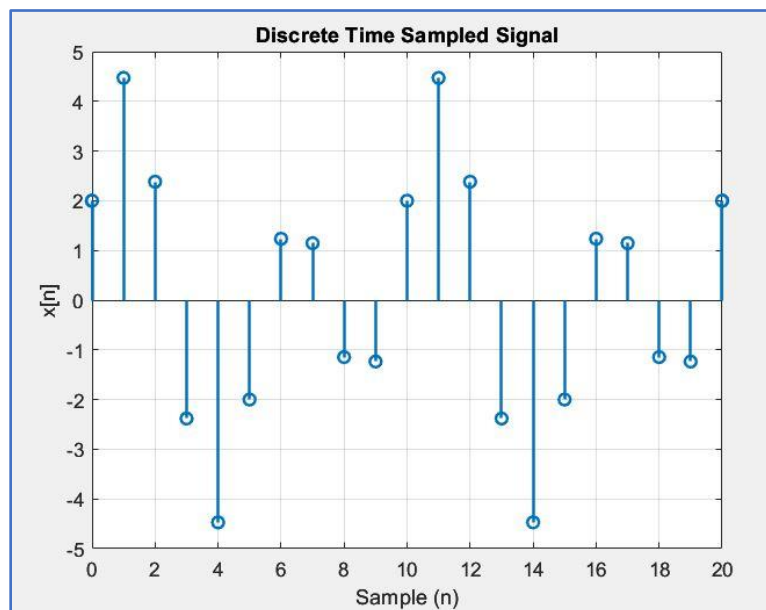
1.1. نمونه برداری از سیگنال پیوسته در زمان

الف) ابتدا سیگنال پیوسته در زمان داده شده را با استفاده از متغیرهای سمبولیک در متلب تعریف کرده و آن را نمایش می دهیم:



شکل 1 - نمایش سیگنال پیوسته در زمان

ب) حال، این سیگنال را با دوره تناوب 0.001 ثانیه یا همان فرکانس 1 کیلوهرتز نمونه برداری می کنیم و به صورت یک سیگنال گسسته در زمان نمایش می دهیم:



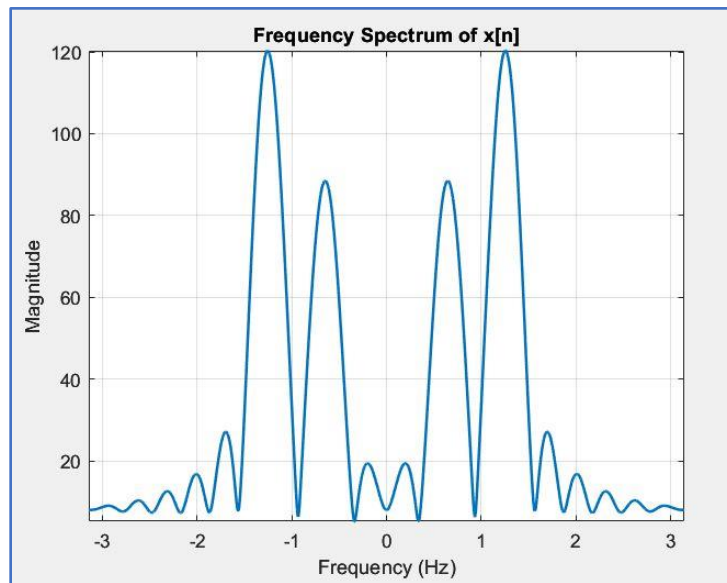
شکل 2 - سیگنال نمونه برداری شده با فرکانس نمونه برداری 1 کیلوهرتز

ج) اکنون، از دو سیگنال پیوسته و گسسته ای که در اختیار داریم، تبدیل فوریه می گیریم تا طیف فرکانسی آن ها را بررسی کنیم؛ جهت انجام این کار، برای سیگنال پیوسته در زمان از دستور **fourier** استفاده می کنیم و برای سیگنال گسسته در زمان، رابطه تبدیل فوریه را با استفاده از یک متغیر سمبولیک و یک حلقه **for** پیاده سازی می کنیم و در نهایت اندازه تبدیل فوریه به دست آمده را نمایش می دهیم؛ تبدیل فوریه سیگنال پیوسته در زمان داده شده، مجموع ضرایبی از تابع دلتای دیراک می باشد و از آنجایی که نمایش تابع دلتای دیراک در نرم افزار متلب امکان پذیر نیست، ما به نمایش ضابطه تبدیل فوریه این سیگنال اکتفا می کنیم:

$$2\pi i * (\text{dirac}(W - 200\pi) + \text{dirac}(W + 200\pi)) - \pi i * (\text{dirac}(W - 400\pi) - \text{dirac}(W + 400\pi)) * 3i$$

شکل 3- تبدیل فوریه سیگنال پیوسته در زمان

همانطور که دیده می شود محتوای فرکانسی این سیگنال، فقط شامل فرکانس های $\pm 200\pi$ و $\pm 400\pi$ است.



شکل 4- اندازه تبدیل فوریه سیگنال گسسته در زمان

همانطور در سیگنال گسسته نمونه برداری شده که دوره تناوب 10 دارد، هارمونیک های دوم و سوم، بیش تر از بقیه تحریک شده اند؛ انتظار می رود که این دو هارمونیک، متناظر با همان فرکانس های 200π و $\pm 400\pi$ در سیگنال پیوسته باشند. همچنین دیده می شود، که با انجام نمونه برداری، فرکانس های دیگری نیز علاوه بر دو فرکانس غالب ایجاد شده اند.

2.1. انجام کاهش نمونه بر روی سیگنال نمونه برداری شده $x[n]$ با ضریب $M=2$

با تعریف یک ضریب M و برابر قرار دادن آن با 2، نرخ نمونه برداری را کاهش می دهیم؛ به بیان دیگر فرکانس نمونه برداری را به 500 هرتز تغییر می دهیم. تصویر سیگنال نمونه برداری شده را در صفحه بعد به همراه سیگنال نمونه برداری شده اصلی آورده ایم (شکل 5). همانطور که دیده می شود نرخ نمونه برداری کاهش یافته و انگار نمونه های سیگنال گسسته اولیه را یکی در میان حذف کرده ایم. با توجه به اینکه فرکانس نمونه برداری 500 هرتز و بالاترین فرکانس موجود در سیگنال اصلی 200 هرتز است، نتیجه می گیریم که رابطه زیر برقرار است:

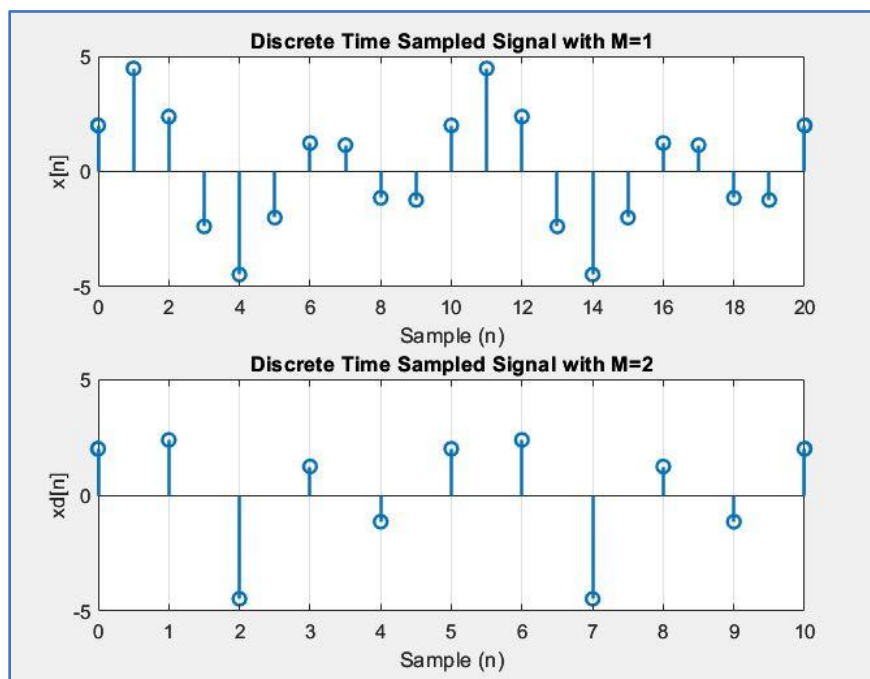
$$\omega_s > 2\omega_m$$

به این معنی که فرکانس نمونه برداری از Nyquist Sampling Rate بیشتر است. پس می توان انتظار داشت که با استفاده از سیگنال نمونه برداری شده جدید نیز می توان سیگنال اصلی را بازیابی کرد.

3.1. اجرای فیلتر anti-aliasing جهت جلوگیری از اثرات انعکاسی در فرآیند نمونه برداری

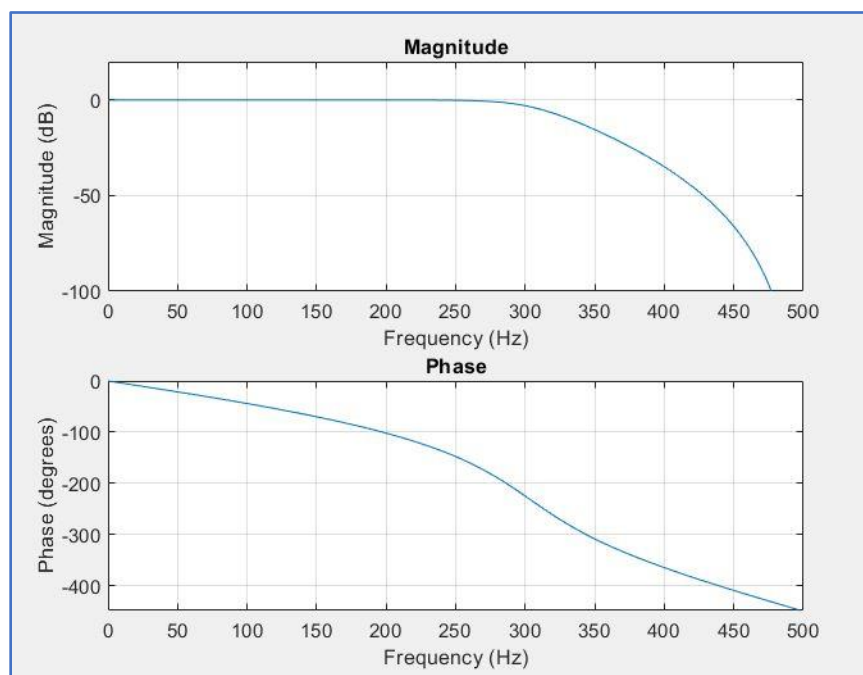
در این مرحله با استفاده از دستور **butter** یک فیلتر پایین گذر از نوع **butterworth** ایجاد می کنیم. برای این کار فرکانس نمونه برداری را برابر با 1000 هرتز می گیریم.

همچنین چون بالاترین فرکانس موجود در سیگنال 200 هرتز است، فرکانس قطع فیلتر را به طور دل به خواه 300 هرتز در نظر می گیریم.



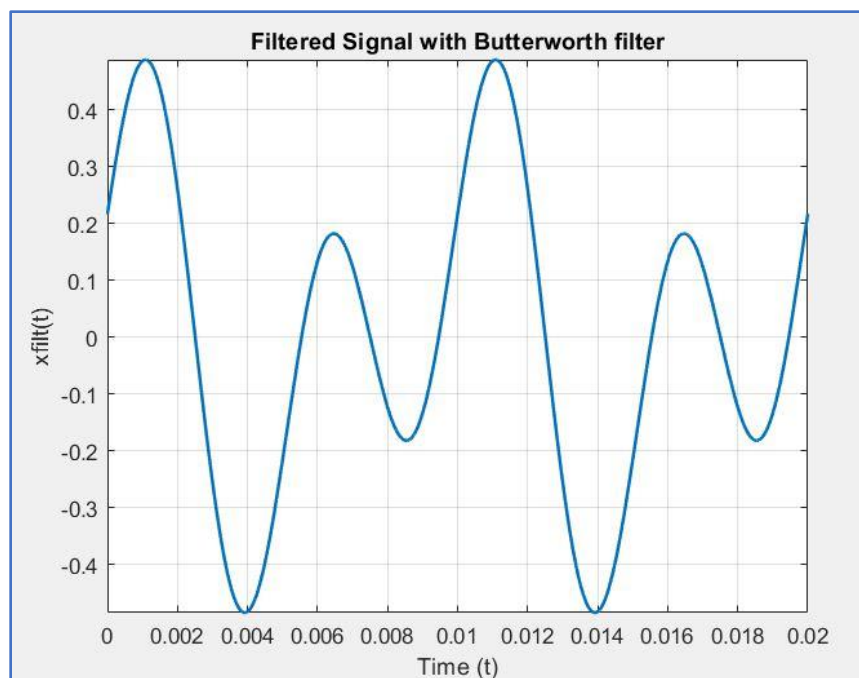
شکل 5 - سیگنال های نمونه برداری شده با ضرایب $M=2$ و $M=1$

اندازه و فاز فیلتر butterworth تولید شده به صورت زیر خواهد بود:



شکل 6 - اندازه و فاز فیلتر butterworth تولید شده

حال می دانیم برای اعمال این فیلتر به سیگنال اصلی مان، باید آن ها را در حوزه زمان کانوالو کنیم؛ این معادلا به این معنی است که باید آن ها را در حوزه s در هم ضرب کنیم. بنابراین با استفاده از دستور `poly2sym` تبدیل لاپلاس فیلتر تولید شده و با استفاده از دستور `laplace` تبدیل لاپلاس سیگنال پیوسته در زمان را بدست می آوریم و با ضرب آن ها در هم و تبدیل لاپلاس معکوس گرفتن، سیگنال زمانی فیلتر شده را بدست می آوریم. تصویر این سیگنال در صفحه بعد آورده شده است (شکل 7).



شکل 7 - سیگنال فیلتر شده با استفاده از فیلتر butterworth

همانطور که دیده می شود، سیگنال بدست آمده با سیگنال اصلی مطابقت دارد چرا که تمام محتوای فرکانسی سیگنال درون بازه فرکانس قطع فیلتر قرار گرفته و چیزی فیلتر نمی شود، بنابراین نیازی به نگرانی راجع به اثرات انعکاسی در نمونه برداری نیست. اما اگر فرکانس نمونه برداری را کاهش دهیم، اثرات انعکاسی افزایش می یابد و آن گاه نیاز است که از فیلتر anti-aliasing استفاده کنیم.
* توجه: اجرای کد این بخش، به خاطر پیچیده بودن تبدیل لاپلاس نهایی و ضرورت تبدیل لاپلاس معکوس گرفتن از آن، کمی زمانبر است.

2. حذف نویز

1.2. شناخت انواع نویز

در این بخش به معرفی مختصری از چندین نویز معروف می پردازیم.

• Salt and Pepper

این نویز به عنوان impulse noise نیز شناخته می شود. با این که تعداد پیکسل های آغشته شده به این نویز در تصویر، محدود است اما هرکدام از این پیکسل ها intensity بالایی خواهند داشت و به صورت پیکسل های تمام سیاه یا تمام سفید خواهند بود. یک متود موثر برای حذف این نویز از تصاویر، استفاده از فیلتر median است. فیلتر های دیگری نیز برای حذف پیکسل های سفید یا پیکسل های سیاه به صورت جداگانه، وجود دارد.

• Gaussian

نویز گوسی، نوعی از نویز است که یک تابع چگالی احتمال دارد، که این تابع چگالی احتمال، همان تابع چگالی توزیع نرمال معروف است. به عبارت دیگر، مقادیری که هر پیکسل آغشته به نویز می تواند بگیرد، یک توزیع گوسی با میانگین و انحراف معیار مشخص دارد. روش های موثر برای حذف این نویز از تصاویر، شامل استفاده از فیلتر های median و gaussian می شود.

• Poisson

نام دیگر این نویز، shot noise می باشد که نوعی نویز است که می توان آن را با فرآیند پواسون مدل کرد. مقایسه جالبی که بین این نویز و نویز گوسی وجود دارد این است که مقدار نویز پواسون با مقدار intensity هر پیکسل در تصویر، همبستگی دارد، در حالی که مقدار نویز گوسی، مستقل از مقدار intensity هر پیکسل است.
فیلتر median، فیلتر مناسبی برای حذف این نویز از تصاویر می باشد.

• Speckle

این نویز، رایج ترین نوع از نویز در تصاویر پزشکی می باشد. مقدار شدت نویز در یک ناحیه مشخص از تصویر، وابسته به مقدار مقیاس خاکستری آن تصویر در همان ناحیه در حالت بدون نویز است.

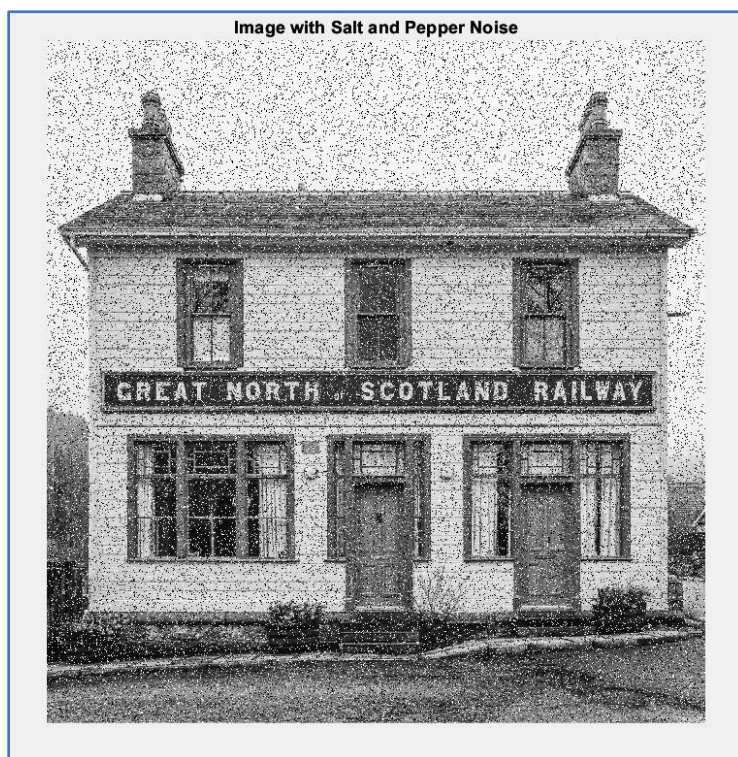
فیلتر median، می تواند فیلتر مناسبی برای حذف این نویز از تصاویر باشد.

اکنون که آشنایی مختصری با این نویز ها یافته ایم، تصویری را به طور دلخواه انتخاب می کنیم و این نویز ها را به صورت جداگانه به تصویر اضافه می کنیم.



شکل 8 - تصویر اصلی که به مقیاس خاکستری برده شده

حال، یکی یکی نویز ها را به تصویر اضافه می کنیم و نمایش می دهیم.



شکل 9 - تصویر آغشته به نویز Salt and Pepper با چگالی نویز 0.15

Image with Gaussian Noise

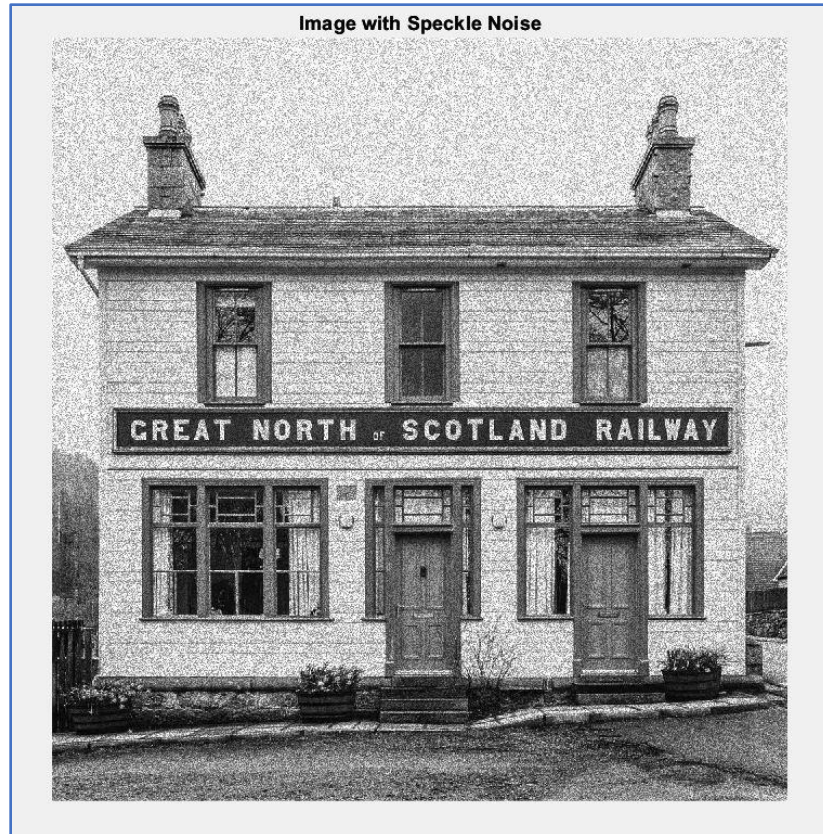


شکل 10 - تصویر آغشته به نویز Gaussian با میانگین 0 و واریانس 0.05

Image with Poisson Noise



شکل 11 - تصویر آغشته به نویز Poisson



شکل 12 - تصویر آغشته به نویز Speckle با میانگین 0 و واریانس 0.05

2.2. آشنایی با فیلتر های Gaussian و Median

در این بخش، آشنایی مختصری با فیلتر های Gaussian و Median خواهیم داشت.

• فیلتر Median

این فیلتر، یک فیلتر غیرخطی است؛ به این معنی که در فیلتر کردن عکس، عمل کانولوشن انجام نمی گیرد و ما کرنل تعریف شده و معینی برای انجام کانولوشن نداریم.

نحوه فیلتر کردن عکس به روش median به این صورت است که یک پنجره $n \times n$ در اطراف هر پیکسل در نظر می گیریم به طوری که n عددی فرد و پیکسل مورد نظر در مرکز پنجره باشد. سپس اعداد درون پنجره را به ترتیب کوچک به بزرگ یا بزرگ به کوچک سورت می کنیم و عدد میانه را در بین این اعداد، به عنوان مقدار نهایی پیکسل مورد نظر انتخاب می کنیم.

• فیلتر Gaussian

برعکس فیلتر median، این فیلتر، یک فیلتر خطی است و با تعریف کرنل و انجام عمل کانولوشن، بر روی تصویر اعمال می شود. نحوه فیلتر کردن عکس به روش median به این صورت است که یک کرنل $n \times n$ تعریف کرده و مختصات درایه مرکزی را به عنوان مبدا در نظر می گیریم؛ سپس با استفاده از توزیع گوسی و میانگین و واریانس که در اختیار داریم، مقادیر درایه های مختلف را تعیین می کنیم. در نهایت با اعمال کردن کرنل به تصویر، آن را فیلتر می کنیم.

بهترین انتخاب برای n در این فیلتر به صورت $n \approx 2\pi\sigma$ است که در آن σ همان انحراف معیار توزیع گوسی می باشد.

3.2. تعریف توابع مربوط به فیلتر ها و اعمال کردن فیلتر ها به تصاویر آغشته به نویز

با توجه به توضیحات داده شده توابع مربوط به فیلتر کردن تصاویر به روش gaussian و median را تعریف می کنیم. همچنین مشکل پیکسل های مرزی را نیز با استفاده از دستور padarray برطرف می کنیم.

اکنون، تصاویر آغشته به نویز های مختلف را با استفاده از این دو تابع فیلتر می کنیم و نمایش می دهیم.

The Noisy Picture with Salt and Pepper Noise, Filtered Using Median Filter with K=3



شکل 13- تصویر آغشته به نویز Salt and Pepper فیلتر شده به روش median

The Noisy Picture with Salt and Pepper Noise, Filtered Using Gaussian Filter with K=7 and Std=1.11



شکل 14- تصویر آغشته به نویز Salt and Pepper فیلتر شده به روش gaussian

در توضیحات تصاویر بالا و تصاویری که در پی می آید، K سایز کرنل و Std همان انحراف معیار است.

The Noisy Picture with Gaussian Noise, Filtered Using Median Filter with $K=5$



شکل 15- تصویر آغشته به نویز Gaussian فیلتر شده به روش median

The Noisy Picture with Gaussian Noise, Filtered Using Gaussian Filter with $K=9$ and $\text{Std}=1.43$



شکل 16- تصویر آغشته به نویز Gaussian فیلتر شده به روش gaussian

The Noisy Picture with Poisson Noise, Filtered Using Median Filter with $K=3$



شکل 17- تصویر آغشته به نویز Poisson فیلتر شده به روش median

The Noisy Picture with Poisson Noise, Filtered Using Gaussian Filter with $K=5$ and $\text{Std}=0.8$



شکل 18- تصویر آغشته به نویز Poisson فیلتر شده به روش gaussian

The Noisy Picture with Speckle Noise, Filtered Using Median Filter with K=5



شکل 19- تصویر آغشته به نویز Speckle فیلتر شده به روش median

The Noisy Picture with Speckle Noise, Filtered Using Gaussian Filter with K=7 and Std=1.11

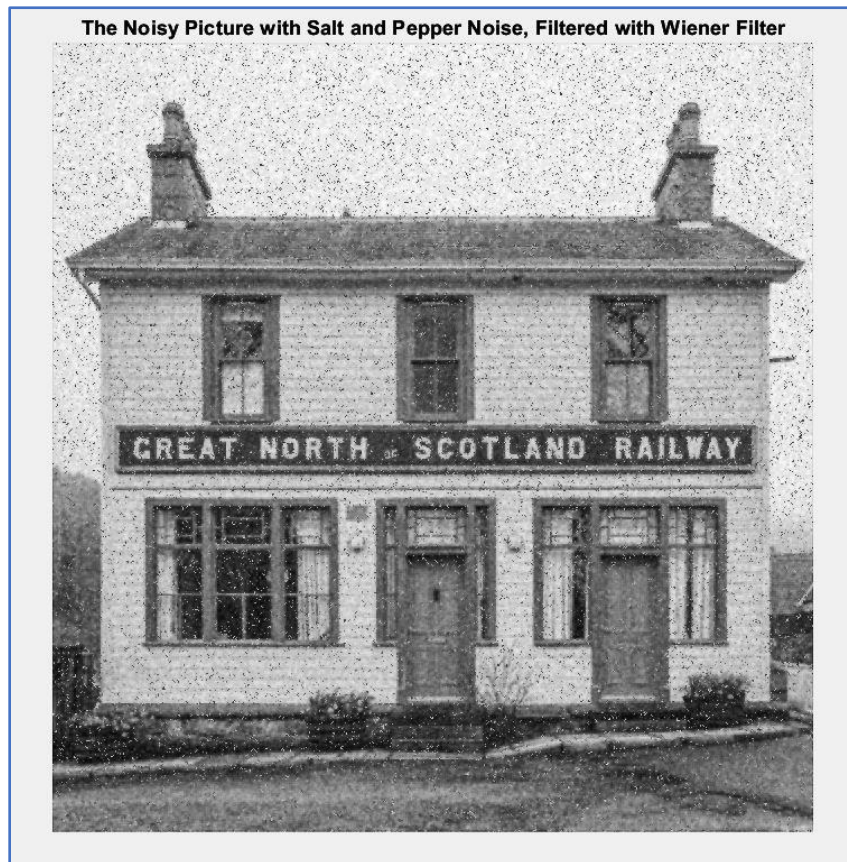


شکل 20- تصویر آغشته به نویز Speckle فیلتر شده به روش gaussian

همانطور که دیده می شود، در برخی از موارد توانسته ایم نویز را تا حد خوبی از تصویر حذف کنیم؛ با این حال در بعضی موارد دیگر بخش جزئی و کوچکی از نویز از بین رفته و با زیاد تر کردن پارامتر های فیلتر اطلاعات اصلی تصویر را از دست می دهیم.

4.2. (امتیازی) فیلتر Wiener

در این بخش آشنایی مختصری با فیلتر wiener خواهیم داشت و در این فیلتر را روی تصاویر آغشته به نویزمان اعمال خواهیم کرد. فیلتر وینر به صورت یک فیلتر بهینه پایین گذر به منظور برطرف کردن اثرات نویزهای جمع شونده طراحی شده است. این فیلتر با تخمین محلی مقادیر میانگین و واریانس پیرامون هر پیکسل، فاصله میانگین مربعی بین سیگنال مشاهده شده و سیگنال بدون نویز را حداقل می کند. در واقع با تعریف یک پنجره $m \times n$ برای این فیلتر، مقادیر میانگین و واریانس برای هر پیکسل در ناحیه حول آن محاسبه شده و با استفاده از تابع چگالی احتمال توزیع گوسی، مقدار آن پیکسل به دست می آید. حال با استفاده از تابع wiener2 این فیلتر را بر روی تصاویرمان اعمال می کنیم. این تابع تصویر مورد نظر و ابعاد پنجره دلخواه برای محاسبه مقادیر پیکسل ها را به عنوان ورودی دریافت کرده و تصویر فیلتر شده را به عنوان خروجی تحویل می دهد. ابعاد پنجره دلخواهمان را برای همه تصاویر، 5×5 تعریف می کنیم.



شکل 21 - تصویر آغشته به نویز Salt and Pepper فیلتر شده به روش wiener

همانطور که دیده می شود بخشی از نویز تصویر حذف شده و کیفیت بهتری نسبت به قبل پیدا کرده. در صفحات بعد، تصاویر آغشته به بقیه نویز ها بعد از فیلتر شدن با استفاده از این فیلتر را آورده ایم. همانطور که دیده می شود این فیلتر برای نویز های gaussian، poisson و speckle تا حد بسیار خوبی موثر است و بخش اعظمی از نویز را از تصاویر آغشته به این نویز ها حذف می کند.

The Noisy Picture with Gaussian Noise, Filtered with Wiener Filter

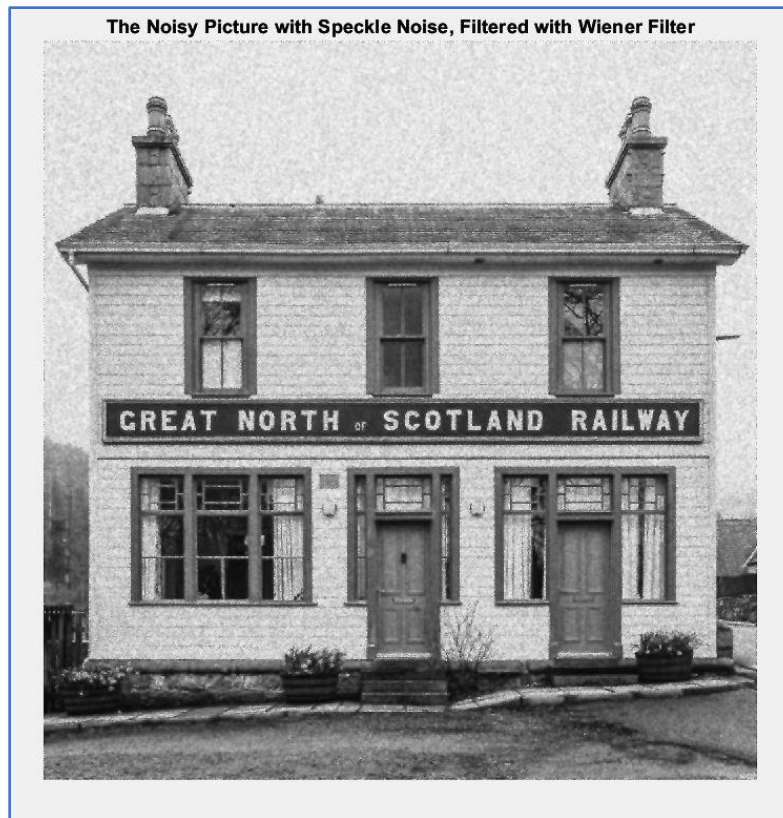


شکل 22- تصویر آغشته به نویز Gaussian فیلتر شده به روش wiener

The Noisy Picture with Poisson Noise, Filtered with Wiener Filter



شکل 23- تصویر آغشته به نویز Poisson فیلتر شده به روش wiener



شکل 24- تصویر آغشته به نویز Speckle فیلتر شده به روش wiener

3. طراحی مقدماتی matched filter

1.3. آشنایی با مفاهیم matched filter و SNR

Matched filter یا فیلتر منطبق، فیلتری است که برای تشخیص یک الگوی خاص با طول محدود، در یک سیگنال اصلی استفاده می شود.

نحوه پیاده سازی این فیلتر، با استفاده از انجام عمل کانولوشن بین سیگنال اصلی و نسخه time-inversed و مزدوج مختلط سیگنال الگوست؛ به بیان ساده تر مانند این است که بین مزدوج مختلط سیگنال الگو و سیگنال اصلی، عمل کانولوشن انجام دهیم با این تفاوت که سیگنال الگو یا سیگنال اصلی را در زمان قرینه نکنیم.

شرط امکان پیاده سازی این فیلتر این است که نویز های موجود درون سیگنال، نویز سفید باشند.

نویز سفید: نویز سفید به سیگنالی گفته می شود که توان آن در همه بسامدها، یکنواخت توزیع شده است. به تعریف دیگر، نویز سفید به هر سیگنال گسسته در زمان که نمونه های آن دنباله ای از متغیرهای تصادفی ناهمبسته که دارای میانگین صفر و واریانس متناهی هستند، گفته می شود.

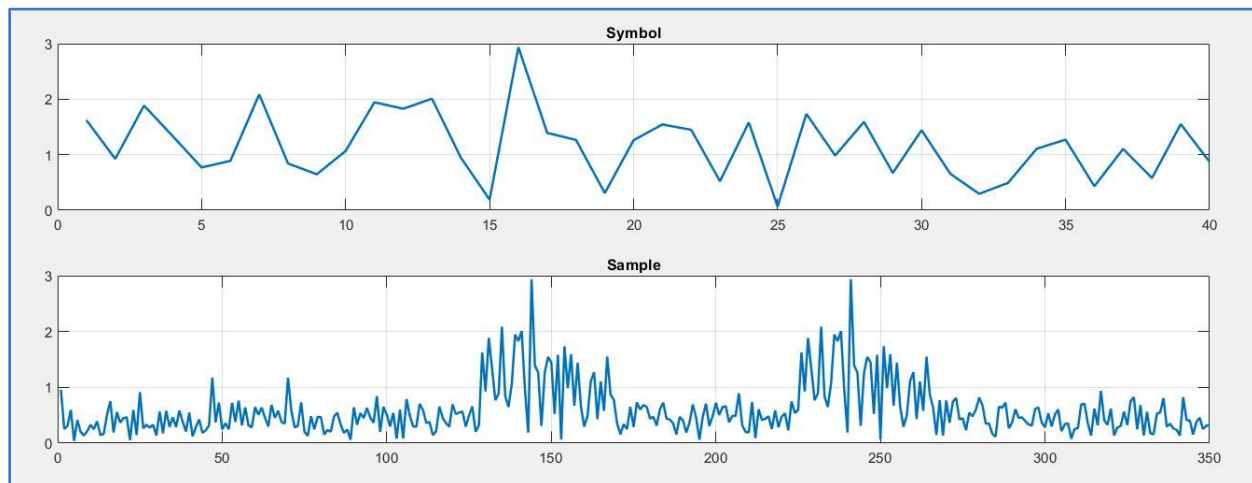
فیلتر منطبق، یک فیلتر خطی ایده آل برای بیشینه کردن SNR در یک سیگنال است.

SNR : نسبت سیگنال به نویز (Signal to Noise ratio) معیاری برای نمایش میزان سیگنال مفید در مقابل سیگنال مزاحم یا نویز در سیستم های الکتریکی است. این عدد، نسبت توان سیگنال به توان نویز است، و آن را بر حسب دسی بل (dB) بیان می کنند. نسبت بالاتر از 1:1 (بیشتر از 0 دسی بل) نشانگر سیگنال بیشتر از نویز است. معمولاً مقدار کمتر از 12 dB نشان دهنده مشکل جدی نویز در خطوط انتقال است، مقدار بالاتر از 20 dB رضایت بخش، و مقدار بالاتر از 30 dB مناسب است. در واقع این شاخص هرچه بیشتر باشد، وضعیت بهتر بوده و نشان دهنده شدت سیگنال مفید بیشتری است.

$$SNR \triangleq \frac{P_{signal}}{P_{noise}}$$

2.3. پیاده سازی تابع تولید سیگنال الگو و جایگذاری آن در یک سیگنال نویزی

با توجه خواسته های ذکر شده تابع مورد نیاز را تعریف می کنیم. در ابتدا با استفاده تابع `randn` یک سیگنال الگو و یک سیگنال نمونه به عنوان سیگنال اصلی تولید می کنیم. سپس سیگنال الگو را به طور رندم در دو بخش از سیگنال نمونه جایگذاری می کنیم و به این شرط هم توجه می کنیم که دو الگوی جایگذاری شده در سیگنال نمونه با هم تداخل نداشته باشند. همچنین به طور دلخواه، دامنه الگو را 3 برابر دامنه سیگنال نمونه آغشته به نویز در نظر می گیریم تا شرط حاکم بر SNR برقرار باشد. مانند تصویر آورده شده در فایل تمرین، طول سیگنال نمونه را برابر با 350 و طول سیگنال الگو را 40 در نظر می گیریم و آن ها را تولید کرده و نمایش می دهیم:

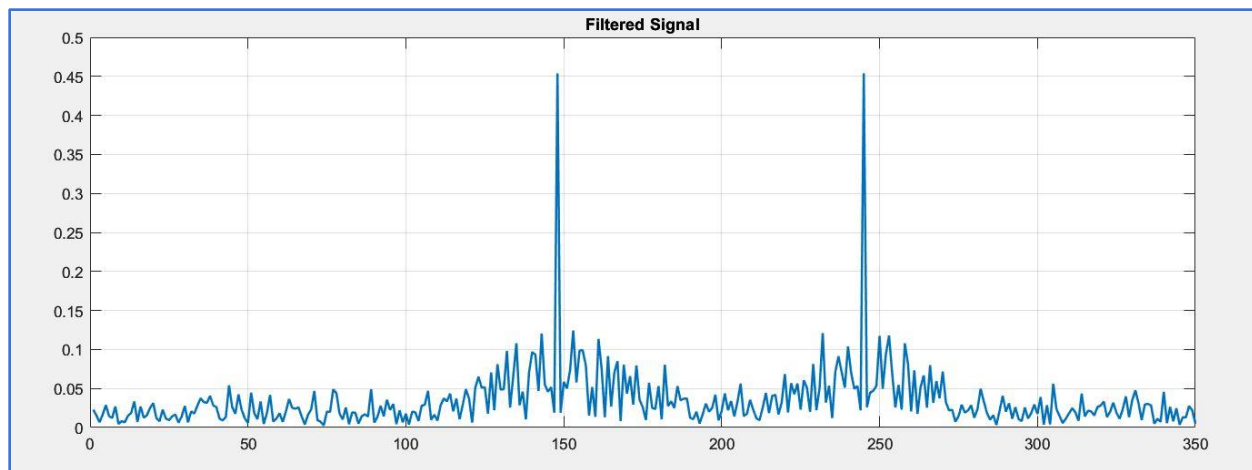


شکل 25 - نمایش سیگنال نمونه و سیگنال الگو

همانطور که مشاهده می شود، سیگنال الگو در دو بخش از سیگنال نمونه جایگذاری شده است.

3.3. پیاده سازی تابع `matched filter` و اعمال آن بر روی `sample` تولید شده

اکنون، تابع `matched filter` را تعریف می کنیم که در آن نسخه `Time-inversed` و مزدوج مختلط سیگنال الگو با سیگنال نمونه کانالو می شود و سیگنال حاصل پس از `normalize` شدن، نمایش داده می شود. با اعمال این تابع بر روی سیگنال بالا خروجی زیر را دریافت می کنیم:



شکل 26 - خروجی `matched filter`

همانطور که دیده می شود، سیگنال خروجی در محل هایی که سیگنال الگو قرار داشته، دارای پیک های تیز شده است.

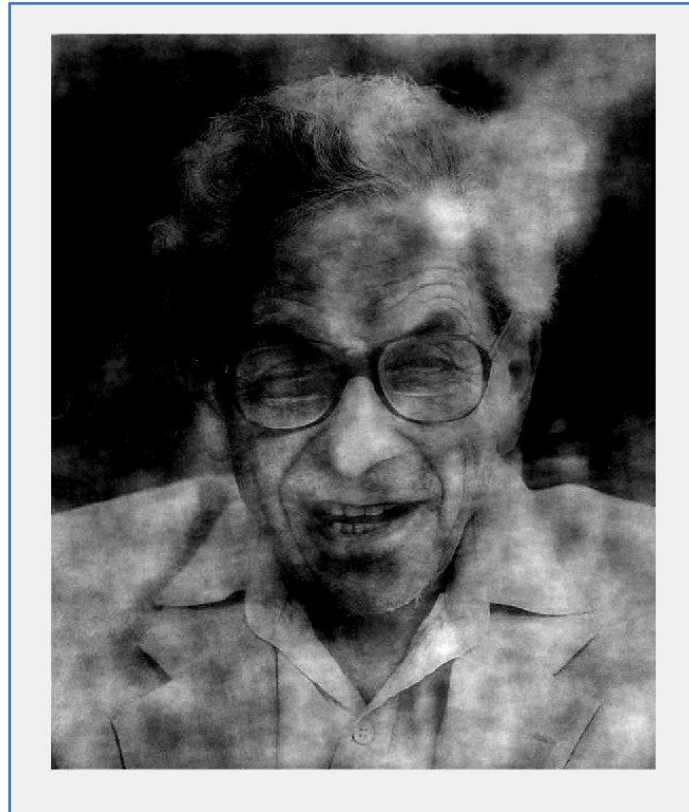
4. اهمیت اندازه و فاز تبدیل فوریه در سیگنال تصویری

دو تصویر ضمیمه شده در فایل تمرین را در متلب لود می کنیم و از آن ها تبدیل فوریه می گیریم.

حال با استفاده از دستورات **abs** و **angle** ، اندازه و فاز هر کدام را به دست آورده و در آرایه هایی ذخیره می کنیم.
می دانیم:

$$F(j\omega) = |F(j\omega)| e^{j\angle F(j\omega)}$$

با توجه به این رابطه تصویری می سازیم که فاز تبدیل فوریه آن، برابر با فاز تبدیل فوریه تصویر اول و اندازه تبدیل فوریه تصویر دوم باشد. تصویر خروجی به شکل زیر است:



شکل 27 – تصویر تولید شده از ترکیب فاز تبدیل فوریه تصویر اول و اندازه تبدیل فوریه تصویر دوم

همانطور که دیده می شود، اطلاعات تصویر اول در تصویر بدست آمده با اختلاف بسیار فاحشی مشهود تر است؛ بنابراین می توان تاثیر فاز تبدیل بر تاثیر اندازه تبدیل فوریه غالب است.

دلیل این امر به طور شهودی، این است که مکان هندسی نقاط و اطلاعات موجود در یک تصویر در فاز آن ذخیره شده اند. با تغییر اندازه صرفاً **intensity** آن ها را تغییر داده ایم. اما اگر فاز را تغییر دهیم، مکان این اطلاعات نسبت به هم تغییر می کند و در نتیجه تصویر جدیدی حاصل خواهد شد.