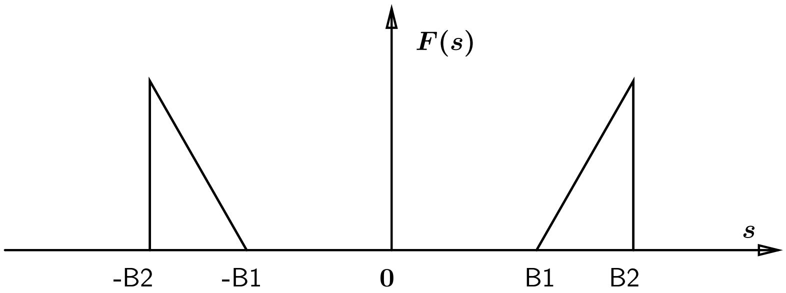
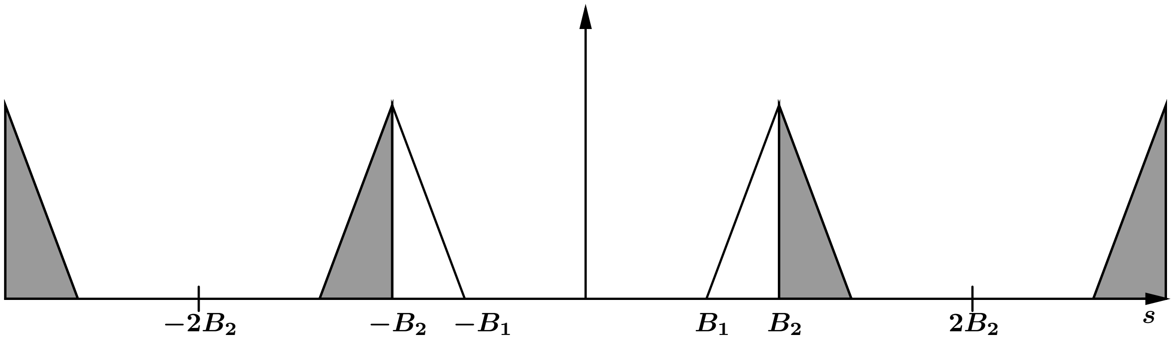
EE 261 The Fourier Transform and its Applications
Fall 2007
Solutions to Problem Set 6

1. (20 points) *Nyquist rate.* The signal $f(t)$ has the Fourier transform $F(s)$ as shown below.

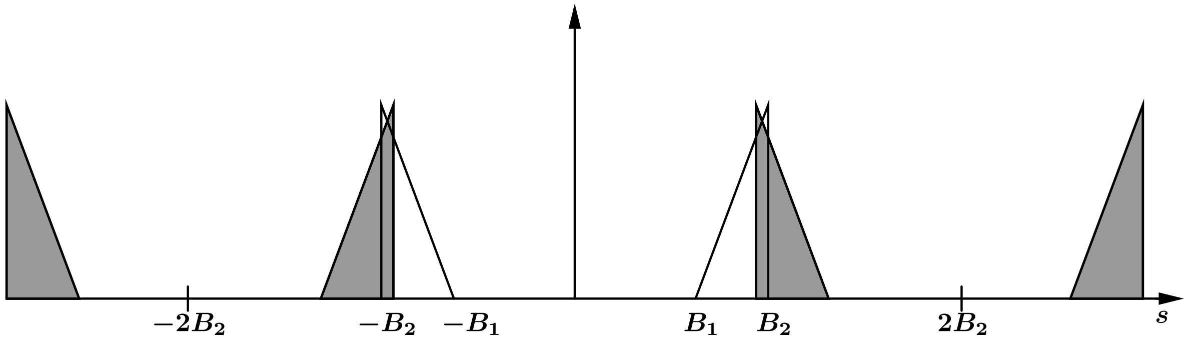


The Nyquist frequency is $2B_2$ since the highest frequency in the signal is B_2. The Sampling Theorem tells us that if we sample above the Nyquist rate, no aliasing will occur. Is it possible, however, to sample at a lower frequency in this case and not get aliasing effects? If it is possible, then explain how it can be done and specify at least one range of valid sampling frequencies below the Nyquist rate that will not result in aliasing. If it is not possible, explain why not.

Solution: The idea here is to somehow exploit the gap between $-B_1$ and B_1. First of all, let us consider what happens when we sample at the Nyquist rate. The spectrum gets replicated at integer multiples of the sampling rate, *i.e.*, $2B_2$ as shown below:

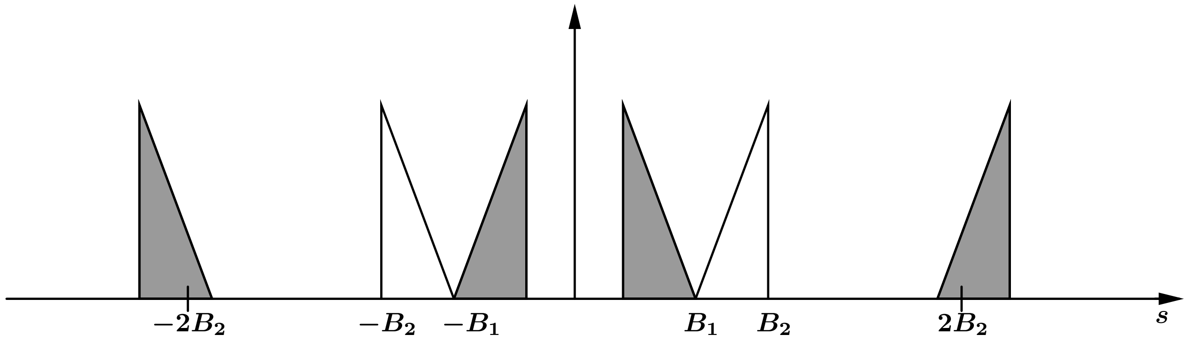


Now, what happens as we sample a little bit below the Nyquist rate? Well, the spectrum still gets replicated at integer multiples of the sampling rate, but there is now overlap, *i.e.*, aliasing:

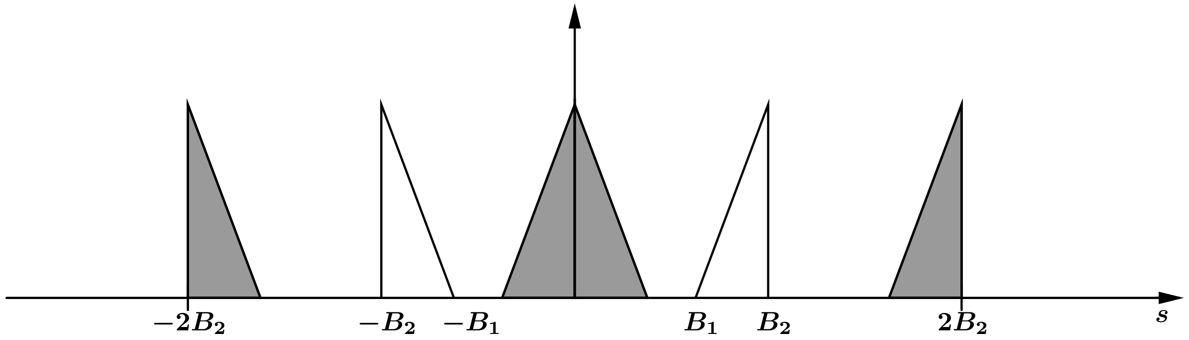


However, what happens as we decrease the sampling frequency even further? If $B_2 \geq 2B_1$, then we clearly will continue to have overlap for all sampling rates all the way down to 0. A simple way to think about it is that the gap between the two spectral islands is simply not big enough to contain another two spectral islands of the same width. Therefore, in that case, the Nyquist rate *is* indeed the slowest sampling rate allowed for perfect reconstruction.

However, if $B_2 < 2B_1$, then we can find a sampling rate lower than the Nyquist frequency and still avoid aliasing. For example, consider the case when the sampling rate is $2B_1$:



This can continue until the sampling frequency reaches B_2 and we are on the verge of yet another overlap:



So, clearly one possible range for f_s is $\boxed{B_2 < f_s < 2B_1.}$
Note that depending on what B_2, B_1 ultimately are, we may be able to exploit the gap further.

2. (20 points) *Natural sampling.* Suppose the signal $f(t)$ is band-limited with $\mathcal{F}f(s) = 0$ for $|s| \geq B$. Instead of sampling with a train of δ's we sample $f(t)$ with a train of very narrow pulses. The pulse is given by a function $p(t)$, we sample at a rate T, and the sampled signal

then has the form

$$g(t) = f(t)\left(\sum_{k=-\infty}^{\infty} Tp(t-kT)\right)$$

(a) Is it possible to recover the original signal f from the signal g?

(b) If not, why not. If it is possible, what conditions on the parameters T and B, and on the pulse $p(x)$ make it possible.

Solution: It's generally easier to think about sampling in the frequency domain, so we'll begin by finding the Fourier transform of $g(t)$. The trick to this problem is noting that our natural sampling function is periodization of the pulse and can be expressed as a convolution of the pulse with a Shah function.

$$\begin{aligned}
g(t) &= f(t)T\sum_{n=-\infty}^{\infty} p(t-nT) \\
&= f(t)T\sum_{n=-\infty}^{\infty} \delta(t-nT) * p(t) \\
&= f(t)T\left(p(t) * \sum_{n=-\infty}^{\infty} \delta(t-nT)\right) \\
&= f(t)T\big(p(t) * \mathrm{III}_T(t)\big)
\end{aligned}$$

Now using the transform pair for $\mathrm{III}_T(t)$, along with the convolution theorem and its dual we can easily find the Fourier transform.

$$\begin{aligned}
\mathcal{F}g(s) &= \mathcal{F}f(s) * T\left(\mathcal{F}p(s)\frac{1}{T}\mathrm{III}_{\frac{1}{T}}(s)\right) \\
&= \mathcal{F}f(s) * \left(\mathcal{F}p(s)\sum_{n=-\infty}^{\infty} \delta\left(s-\frac{n}{T}\right)\right)
\end{aligned}$$

Using the multiplication property of the delta function

$$\mathcal{F}g(s) = \mathcal{F}f(s) * \sum_{n=-\infty}^{\infty} \mathcal{F}p\left(\frac{n}{T}\right)\delta\left(s-\frac{n}{T}\right)$$

And, finally, linearity of convolution and the sift property of delta function

$$\mathcal{F}g(s) = \sum_{n=-\infty}^{\infty} \mathcal{F}p\left(\frac{n}{T}\right)\mathcal{F}f\left(s-\frac{n}{T}\right)$$

(a) So we see that, just like with ideal sampling, the Fourier transform of the sampled signal is an infinite sum of shifted copies of the original signal's Fourier transform. The only difference in the case when realistic pulses are used is that each of these copies is scaled by the value of the pulse's Fourier transform at the center frequency of that copy. But a scaling factor can easily be taken out by adjusting the gain of the low-pass filter we use to reconstruct $f(t)$ from its samples. So $\boxed{\text{yes,}}$ under certain conditions it is possible to recover $f(t)$ from $g(t)$.

(b) As with ideal sampling, we need to sample above the Nyquist rate so that there is no aliasing. So, we need $\boxed{T < \frac{1}{2B}.}$ Surprisingly, there are no conditions on the pulse $p(t)$ as long as it has a Fourier transform so that all the steps we took before make sense. As long as $\mathcal{F}p(0) \neq 0$ we can reconstruct $f(t)$ by lowpass filtering with the proper scaling, and if $\mathcal{F}p(0) = 0$ then we can recover by bandpass filtering to use one of the other copies.

3. (15 points) Let $f(t) = \cos 2\pi t$. Suppose we sample $f(t)$ at a rate 2/3 Hz and then interpolate using a lowpass filter with cut-off frequency 2/3. What signal, $g(t)$, is the result? Sketch $f(t)$ and $g(t)$ on the same axes and comment on what you see. Is $g(t)$ an alias of $f(t)$ for this sampling rate?

The process illustrated in this problem is the basis of the 'sampling oscilloscope'.

Solution:

Start with the Fourier transform of $f(t)$:

$$\mathcal{F}f(s) = \frac{1}{2}(\delta(s-1) + \delta(s+1)).$$

To sample $f(t)$ at 2/3 Hz is, in the frequency domain, to convolve $\mathcal{F}f$ with $\mathrm{III}_{2/3}$. This gives

$$\mathcal{F}f * \mathrm{III}_{2/3} = \frac{1}{2}(\delta_1 + \delta_{-1}) * \sum_{k=-\infty}^{\infty} \delta_{2k/3} = \frac{1}{2}\sum_{k=-\infty}^{\infty}(\delta_{1+2k/3} + \delta_{-1+2k/3}).$$

We're then to cut off above 2/3 Hz (and below $-2/3$ Hz) so the question is what δ's in the convolution above are between $-2/3$ and $+2/3$. For the δ's at the points $1 + 2k/3$ we want to know the integer k's for which

$$\begin{aligned}
&-\frac{2}{3} < 1 + \frac{2k}{3} < \frac{2}{3} \quad \text{so} \\
&-\frac{5}{3} < \frac{2k}{3} < -\frac{1}{3} \\
&-5 < 2k < -1
\end{aligned}$$

and that gives $k = -1$ and $k = -2$ with corresponding δ's (including the factor 1/2 that's in front):

$$\frac{1}{2}(\delta_{1-2/3} + \delta_{1-4/3}) = \frac{1}{2}(\delta_{1/3} + \delta_{-1/3}).$$

For the δ's at the points $-1 + 2k/3$ we want to know the integer k's for which

$$\begin{aligned}
&-\frac{2}{3} < -1 + \frac{2k}{3} < \frac{2}{3} \quad \text{so} \\
&\frac{1}{3} < \frac{2k}{3} < \frac{5}{3} \\
&1 < 2k < 5
\end{aligned}$$

and that gives $k = 1$ and $k = 2$ with corresponding δ's

$$\frac{1}{2}(\delta_{-1+2/3} + \delta_{-1+4/3}) = \frac{1}{2}(\delta_{-1/3} + \delta_{1/3}).$$

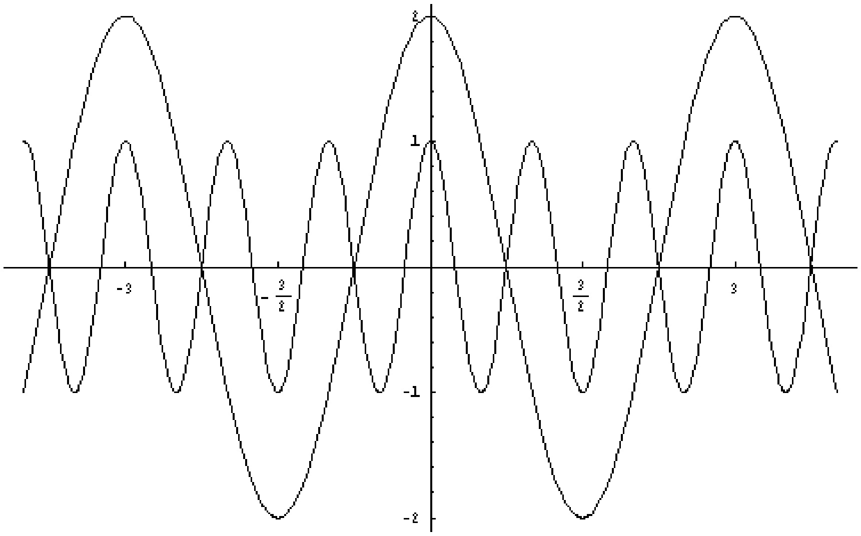
Therefore, cutting off $\mathcal{F}f * \text{III}_{2/3}$ by $\Pi_{4/3}$ (note that $\Pi_{4/3}$ cuts off outside of $\pm 2/3$ results in

$$\frac{1}{2}(\delta_{1/3} + \delta_{-1/3}) + \frac{1}{2}(\delta_{-1/3} + \delta_{1/3}) = \delta_{-1/3} + \delta_{1/3}.$$

The inverse Fourier transform of this is our function $g(t)$, and that's

$$g(t) = 2\cos\frac{2\pi t}{3}.$$

Here's a plot of $f(t)$ and $g(t)$ together:



I'd say that $g(t)$ has the same shape as $f(t)$ but it's stretched in both the vertical and horizontal directions. $f(t)$ has period 1 and amplitude 1, while $g(t)$ has period 3 and amplitude 2.

Using the sampling rate 2/3 the sample points (according to the sampling theorem) are $3k/2$. The values of $f(t)$ at the sample points are

$$f(3k/2) = \cos 2\pi\frac{3k}{2} = \cos 3k\pi = (-1)^k.$$

The values of $g(t)$ at the sample points are

$$g(3k/2) = 2\cos\frac{2\pi}{3}\frac{3k}{2} = 2\cos\pi k = 2(-1)^k.$$

$g(t)$ is 'tracking' $f(t)$ at the sample points but the functions are not equal at those points; $g(t)$ is twice the size. I wouldn't call $g(t)$ an alias of $f(t)$ in the standard usage of the term.

4. (15 points) *More on sampling.* Let $g(t)$ be an integrable, band-limited signal with $\mathcal{F}g(s) = 0$ for $|s| \geq p/2$. Express the integral

$$\int_{-\infty}^{\infty} g(t)\, dt$$

in terms of sample values of g. What is the minimum sampling rate that will allow us to do this?

Solution: Again, it's easiest to consider this problem in the frequency domain. First we recognize that the area under the function is just the Fourier transform evaluated at $s = 0$.

$$\int_{-\infty}^{\infty} g(t)\, dt = \left. \int_{-\infty}^{\infty} g(t) e^{-2\pi i s t}\, dt \right|_{s=0} = \mathcal{F}g(s)|_{s=0}$$

This means we only need to recover one value of the Fourier transform in terms of the sampled signal, call it h, where

$$h(t) = g(t) \text{III}_{\frac{1}{s_{min}}}(t)$$

This means that the 'frequency islands' can overlap as long as they don't interfere at $s = 0$. Since the signal is bandlimited to $s_0 = p/2$, and the spectral copies or islands are spaced s_{min} apart, none of the other copies will interfere at $s = 0$ if

$$\boxed{s_{min} > s_0}$$

As long as our sampling rate is greater than s_0 we know that

$$\mathcal{F}h(s)|_{s=0} \propto \mathcal{F}g(s)|_{s=0} = \int_{-\infty}^{\infty} g(t)\, dt$$

Specifically

$$\mathcal{F}h(s) = \mathcal{F}g(s) * s_{min} \text{III}_{s_{min}}(s)$$

So each copy of $\mathcal{F}g(s)$ is scaled by s_{min} and therefore

$$\frac{1}{s_{min}} \mathcal{F}h(s)|_{s=0} = \mathcal{F}g(s)|_{s=0} = \int_{-\infty}^{\infty} g(t)\, dt$$

Let's find an expression $\mathcal{F}h(s)$ in terms of the samples.

$$\begin{aligned} h(t) &= g(t) \sum_{n=-\infty}^{\infty} \delta\left(t - \frac{n}{s_{min}}\right) \\ &= \sum_{n=-\infty}^{\infty} g\left(\frac{n}{s_{min}}\right) \delta\left(t - \frac{n}{s_{min}}\right) \end{aligned}$$

Let g_n denote the samples of g:

$$h(t) = \sum_{n=-\infty}^{\infty} g_n \delta\left(t - \frac{n}{s_{min}}\right)$$

Taking the Fourier transform

$$\mathcal{F}h(s) = \sum_{n=-\infty}^{\infty} g_n e^{-2\pi i \frac{n}{s_{min}} s}$$

If we evaluate this at $s = 0$, we get

$$\mathcal{F}h(s)|_{s=0} = \sum_{n=-\infty}^{\infty} g_n$$

Plugging this into our previous result, we conclude

$$\boxed{\int_{-\infty}^{\infty} g(t)\,dt = \frac{1}{s_{min}} \sum_{n=-\infty}^{\infty} g_n}$$

The area is a scaled sum of the samples.

5. (15 points) *Matlab Exercise on Sampling:* Obtain the image
http://see.stanford.edu/materials/lsoftaee261/man.gif

   (a) Load the 8-level gray-scale 256x256 image in Matlab using 'imread' command. Convert the matrix image into a vector of length 256^2 (using Matlab's ':' command) and call it the 'time-domain' signal $x(n)$. Change the values in $x(n)$ from type 'uint8' to 'double'. Normalize it such that the maximum signal value is 1.

   (b) Find the Fourier transform $X(f)$ of $x(n)$ using the FFT command, where $X(f)$ has length the same length as $x(n)$. Plot the magnitude response of $X(f)$ centered at 0 i.e. on the X-axis, f ranges from $-\frac{256^2}{2}$ to $\left(\frac{256^2}{2} - 1\right)$ (total 256^2 points).

   (c) Define the 'bandwidth', $N_B(\alpha)$, of the signal $x(n)$ as the frequency values that contain fraction α of the total energy. Mathematically, this is equivalent to

$$\begin{aligned} E_{\text{total}} &\triangleq \sum_{f=-\frac{256^2}{2}}^{f=\frac{256^2}{2}-1} |X(f)|^2 \\ E_m &\triangleq \sum_{f=-m}^{f=m} |X(f)|^2 \\ N_B(\alpha) &\triangleq 2p \quad \text{s.t.} \quad \frac{E_p}{E_{\text{total}}} = \alpha \end{aligned}$$

   For the following α values, $\alpha = 0.9, 0.95, 0.99$, sample the signal $x(n)$ at the corresponding Nyquist rate. In order to sample $x(n)$ at a rate $\frac{1}{r}$, simply pick every rth value in $x(n)$, so that the resulting sampled-signal $x_s(n)$ will be of length $\frac{256^2}{r}$. Find the re-constructed signal $\hat{x}(n)$ of length 256^2 using interpolation on the sampled-signal $x_s(n)$. For each value of α, the methods to be used for interpolation are (i) replication by nearest-neighbor and (ii) linear. In the two interpolation schemes, reconstruction of any point has contributions from 1 and 2 nearest neighbors respectively. Display the signal $\hat{x}(n)$ as a 256x256 image. Submit the original image, the 6 reconstructed images and Matlab code. Briefly explain the artifacts seen in the reconstructed images. Matlab commands

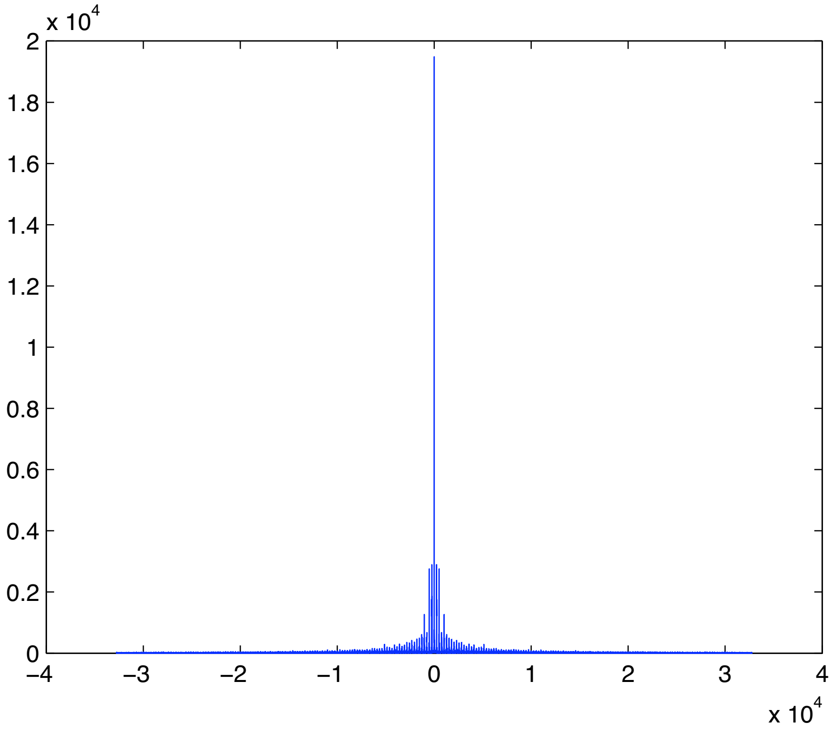
```
imagesc(); colormap('gray')
```

can be used to display the image.

Solution:

(a)
```
man_img=imread('man256.gif');
x_n=man_img(:);
x_n=double(x_n);
max_value=max(x_n);
x_n=x_n/max_value;
```

(b)
```
X_f=fft(x_n);
len_X_f=length(X_f);
X_f_abs=abs(X_f);
X_f_centered=[X_f_abs(len_X_f/2+1:end); X_f_abs(1:len_X_f/2)];
plot([-(len_X_f/2):(len_X_f/2)-1], X_f_centered);
```



(c)
```
% Compute energy
total_energy=sum(X_f_abs.^2);
partial_energy(1)=X_f_abs(1)^2;
for i=2:(len_X_f/2)-1;
   partial_energy(i)=partial_energy(i-1)+2*X_f_abs(i)^2;
end
ratio_energy=partial_energy/total_energy;
```

```
% Compute bandwidth
cut_off_freq=length(ratio_energy(find(ratio_energy<alpha)))

% Sample signal
sampling_rate=(2*cut_off_freq)/length(x_n);
sampling_interval=floor(1/sampling_rate);
x_s_n=x_n(1:sampling_interval:end);

% Interpolation method 1 -- replace by nearest neighbor
x_hat_1=zeros(1,length(x_n));
for i=1:length(x_n)
    if mod(i-1,sampling_interval)==0
        x_hat_1(i)=x_s_n(1+((i-1)/sampling_interval));
    elseif (1+ceil((i-1)/sampling_interval))> length(x_s_n) || ...
            (1+floor((i-1)/sampling_interval))> length(x_s_n)
        x_hat_1(i)=x_s_n(end);
    else
        if (mod(i-1,sampling_interval) > sampling_interval/2)
            x_hat_1(i)=x_s_n(1+ceil((i-1)/sampling_interval));
        else
            x_hat_1(i)=x_s_n(1+floor((i-1)/sampling_interval));
        end
    end
end
hat_image_1=uint8(max_value*reshape(x_hat_1,sqrt(length(x_n)),sqrt(length(x_n))));

% Interpolation method 2 -- linear interpolation
x_hat_2=zeros(1,length(x_n));
for i=1:length(x_n)
    w_2=((i-1)-floor((i-1)/sampling_interval)*sampling_interval)/sampling_interval;
    w_1=1-w_2;
    if (1+ceil((i-1)/sampling_interval))> length(x_s_n) || ...
            (1+floor((i-1)/sampling_interval))> length(x_s_n)
        x_hat_2(i)=x_s_n(end);
    else
        x_hat_2(i)=w_1*x_s_n(1+floor((i-1)/sampling_interval))+w_2*x_s_n(1+ceil((i-1)
    end
end
hat_image_2=uint8(max_value*reshape(x_hat_2,sqrt(length(x_n)),sqrt(length(x_n))));

% Diplay the image
figure; imagesc(man_img); colormap('gray');
figure; imagesc(hat_image_1); colormap('gray');
figure; imagesc(hat_image_2); colormap('gray');
```

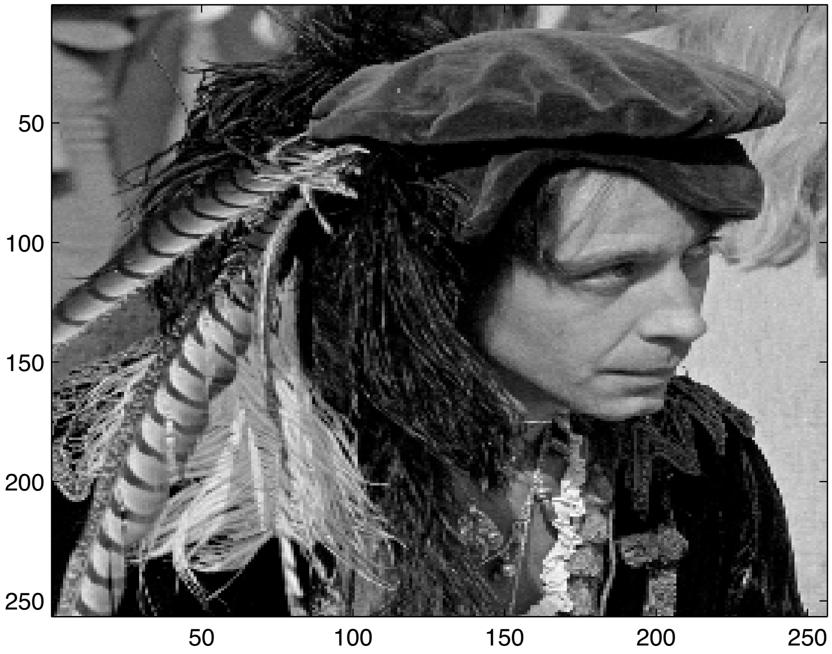


Figure 1: Original Image

Nearest neighbor reconstruction results in block-effect, whereas linear reconstruction is more smooth.

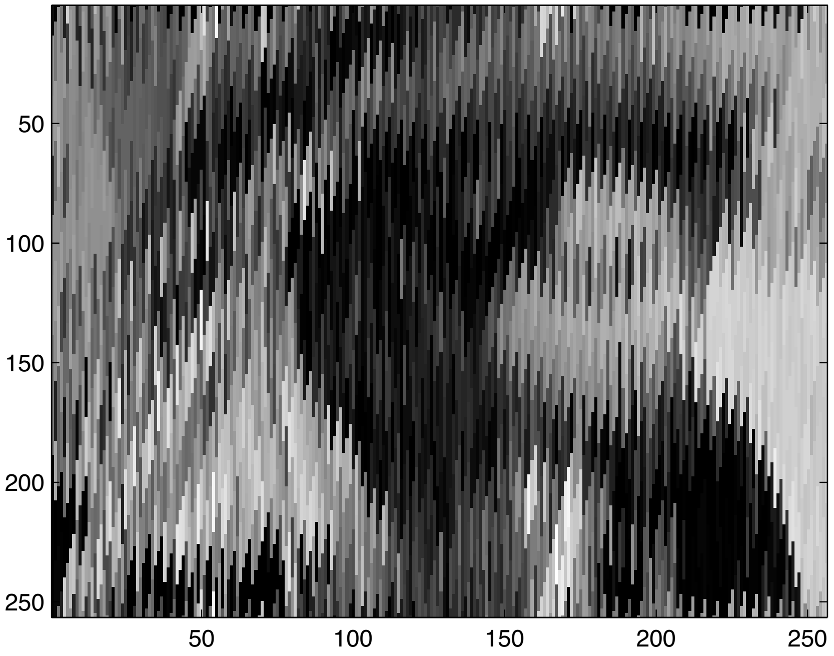


Figure 2: Nearest Neighbor $\alpha = 0.9$

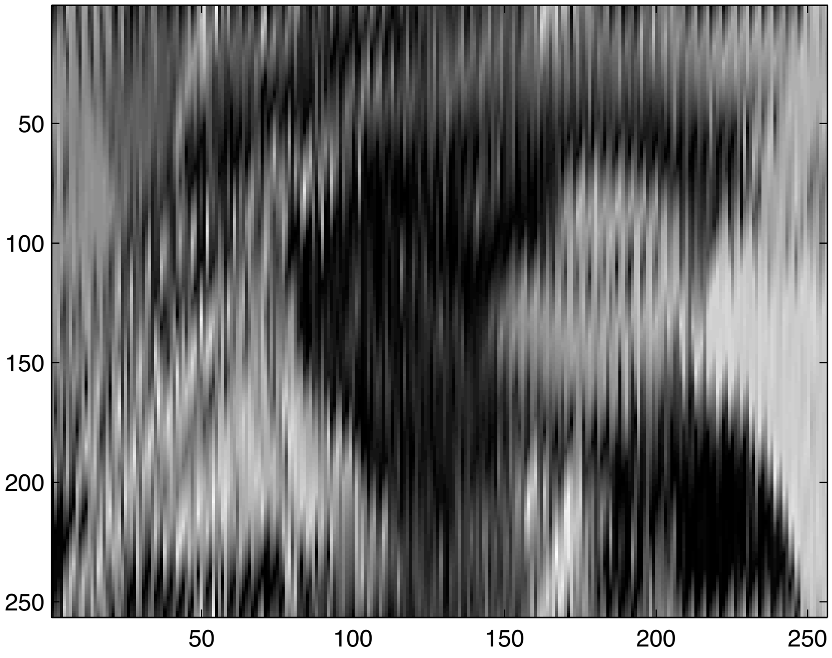


Figure 3: Linear $\alpha = 0.9$

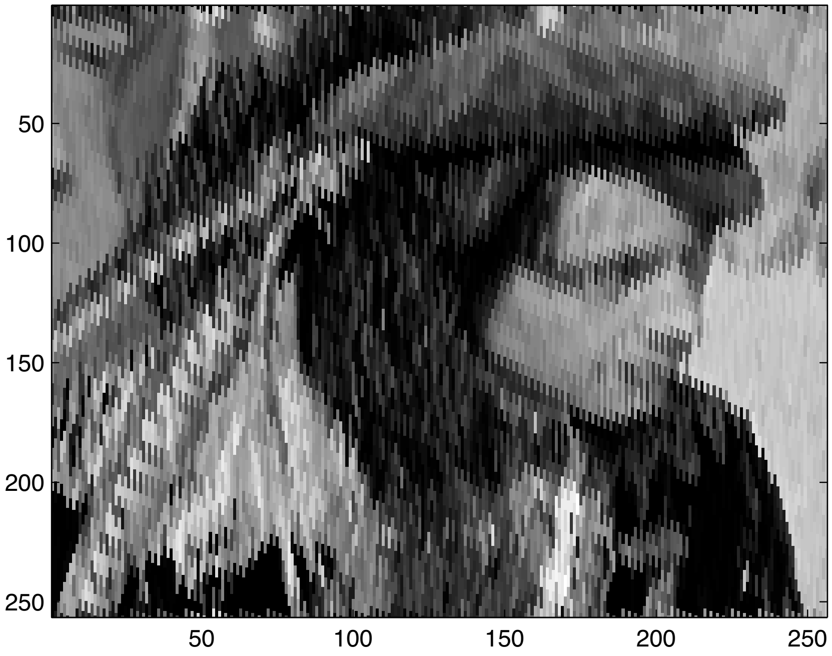


Figure 4: Nearest Neighbor $\alpha = 0.95$

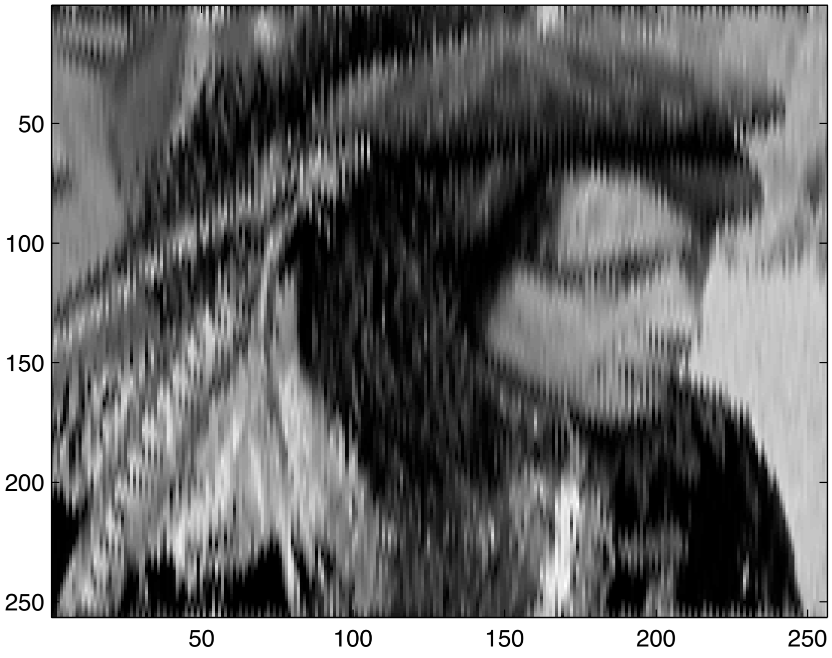


Figure 5: Linear $\alpha = 0.95$

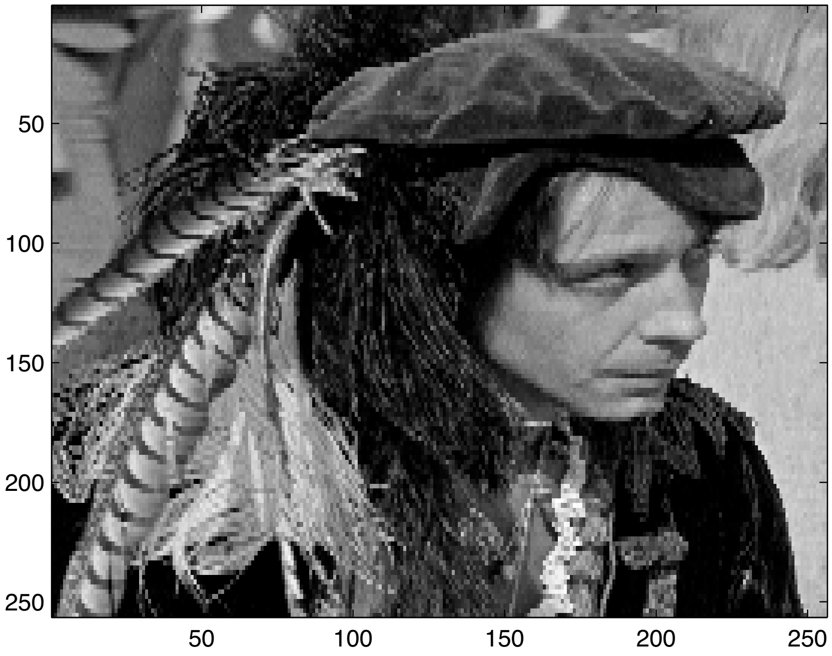


Figure 6: Nearest Neighbor $\alpha = 0.99$

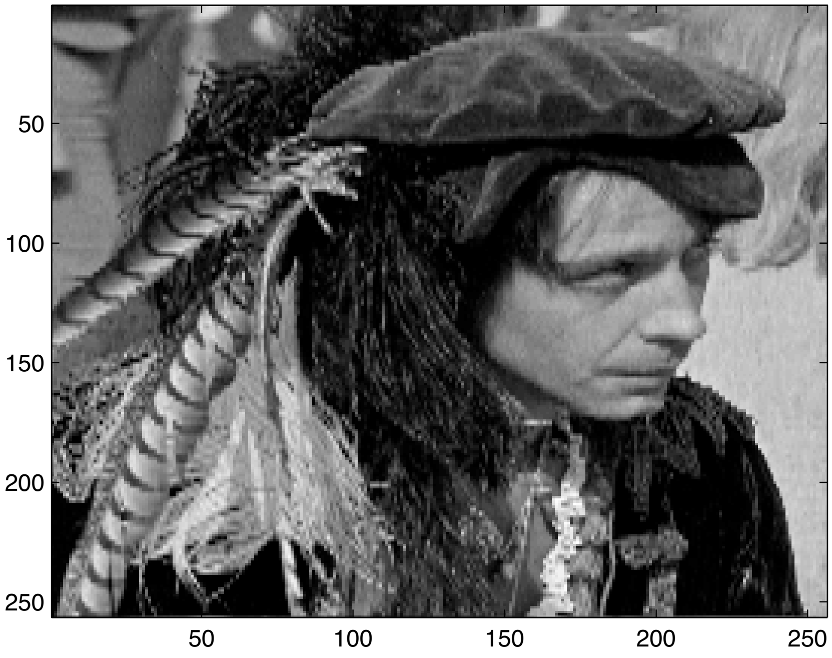


Figure 7: Linear $\alpha = 0.99$