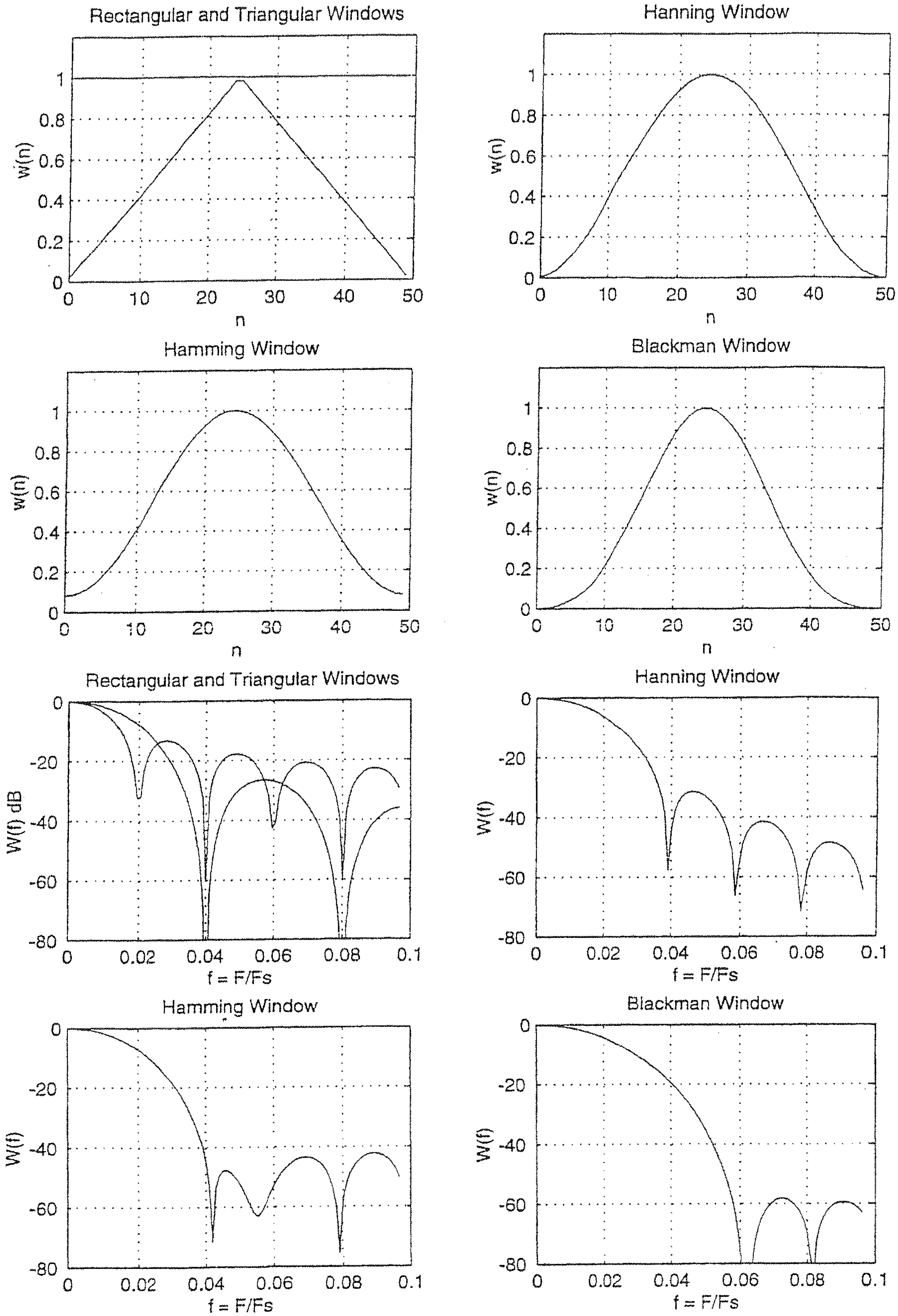

Windows, Spectral Resolution, and Leakage

```
% Script File: Win_Ex1.m
% ------------------------------
% Illustrates the shape of various window types and their spectra
clear,clf
Nn=50;                                % window length in samples
Nk=1024;                              % transform length
w(:,1)=boxcar(Nn);                    % Rectangular window
w(:,2)=triang(Nn);                    % Triangular window
w(:,3)=hanning(Nn);                   % Von Han window
w(:,4)=hamming(Nn);                   % Hamming window
w(:,5)=blackman(Nn);                  % Blackman window
%
W=abs(fft(w,Nk));                     % Matrix of window spectra
for m=1:5
    W(:,m)=W(:,m)/sum(w(:,m));        % normalize peak value to 1.
end
n=[0:Nn-1]';
% window plots
% ----------------
subplot(2,2,1), plot(n,[w(:,2) w(:,1)],'-r'),grid on,xlabel('n'),ylabel('w(n)')
                title('Rectangular and Triangular Windows')
                axis([0 Nn 0 1.2]);
subplot(2,2,2), plot(n,w(:,3),'-r'),grid on,xlabel('n'),ylabel('w(n)')
                title('Hanning Window')
                axis([0 Nn 0 1.2]);
subplot(2,2,3), plot(n,w(:,4),'r'),grid on,xlabel('n'),ylabel('w(n)')
                title('Hamming Window')
                axis([0 Nn 0 1.2]);
subplot(2,2,4), plot(n,w(:,5),'-r'),grid on,xlabel('n'),ylabel('w(n)')
                title('Blackman Window')
                axis([0 Nn 0 1.2]);
subplot, pause; axis;
% Spectral plots
% ----------------
ki=[1:100]';
fi=(ki-1)/Nk;
subplot(2,2,1), plot(fi,20*log10([W(ki,2) W(ki,1)]),'-r'),grid on,
                xlabel('f = F/Fs'),ylabel('W(f) dB'),
                title('Rectangular and Triangular Windows')
                axis([0 0.1 -80 0]);
subplot(2,2,2), plot(fi,20*log10(W(ki,3)),'-r'),grid on,
                xlabel('f = F/Fs'),ylabel('W(f)'), title('Hanning Window')
                axis([0 0.1 -80 0]);
subplot(2,2,3), plot(fi,20*log10(W(ki,4)),'-r'),grid on,
                xlabel('f = F/Fs'),ylabel('W(f)'),
                title('Hamming Window')
                axis([0 0.1 -80 0]);
subplot(2,2,4), plot(fi,20*log10(W(ki,5)),'-r'),grid on,
                xlabel('f = F/Fs'),ylabel('W(f)'),
                title('Blackman Window').
                axis([0 0.1 -80 0]);
subplot, pause; axis;
```

Output from win_Ex1



```
% Script File: win_Ex2.m
% ----------------------------
% Illasutrates the shpae of the Kaiser windows and their spectra
% ------------------------------------------------------------------------
clear, clg
Nn=50;
beta=[0 3 6 9];                          % larger beta ==>wider main lobe
                                         % but larger peak-side lobe level
for m=1:length(beta)
        w(:,m) =kaiser(Nn,beta(m));
end
%
Nk=1024;
W=abs(fft(w,Nk));                        % Matrix of window spectra
for m=1:length(beta)
        W(:,m)=W(:,m)/sum(w(:,m));       % normalize peak value to 1.
        W(:,m)=20*log10(W(:,m));
end
%
% Window Plots
% ----------------
subplot(2,2,1), plot(n, w(:,1),'-r'),grid,xlabel('n'),ylabel('w(n)')
                title('Kaiser Window, beta = 0')
                axis([0 Nn 0 1.2]);
subplot(2,2,2), plot(n,w(:,2),'-r'),grid,xlabel('n'),ylabel('w(n)')
                title('beta = 3')
                axis([0 Nn 0 1.2]);
subplot(2,2,3), plot(n,w(:,3),'r'),grid ,xlabel('n'),ylabel('w(n)')
                title('beta = 6')
                axis([0 Nn 0 1.2]);
subplot(2,2,4), plot(n,w(:,4),'-r'),grid,xlabel('n'),ylabel('w(n)')
                title('beta = 9')
                axis([0 Nn 0 1.2]);
subplot, pause; axis;
%
% Spectral Plots
% ---------------
ki=[1:100]';
fi=(ki-1)/Nk;
subplot(2,2,1), plot(fi,W(ki,1),'-r'),grid,
                xlabel('f = F/Fs'),ylabel('W(f) dB'),
                title('Kaiser Window, beta = 0')
                axis([0 0.1 -80 0]);
subplot(2,2,2), plot(fi,W(ki,2),'-r'),grid,
                xlabel('f = F/Fs'), ylabel('W(f)'),
                title('Kaiser Window, beta = 3')
                axis([0 0.1 -80 0]);
subplot(2,2,3), plot(fi,W(ki,3),'-r'),grid,
                xlabel('f = F/Fs'),ylabel('W(f)'),
                title('Kaiser Window, beta = 6')
                axis([0 0.1 -80 0]);
subplot(2,2,4), plot(fi,W(ki,4),'-r'),grid on,
                xlabel('f = F/Fs'),ylabel('W(f)'),
                title('Kaiser Window, beta = 9')
                axis([0 0.1 -80 0]);
subplot, pause; axis;
%
```

Comments on the output from win_Ex2.m

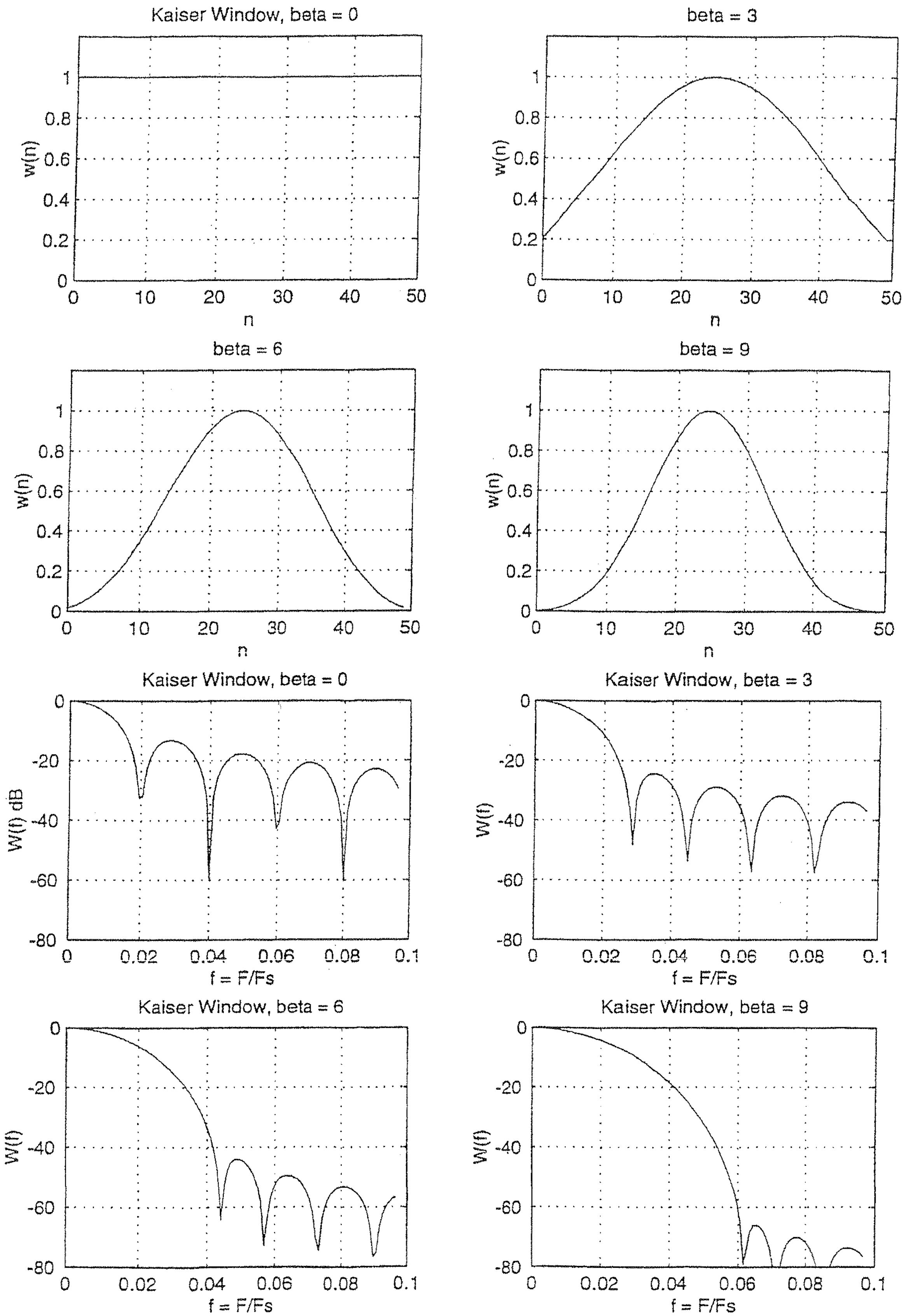
Note that the case beta = 0 is equivalent to a rectangular window. As beta increases the main lobe becomes broader and the absolute value of the peak side-lobe level becomes larger. This degree of freedom is not available with any of the previously examined windows. Note that for beta ~=6, the width of the main-lobe is roughly the same as the width of the hanning and hamming main-lobe, but the peak side lobe level is better (absolute value is larger). Similarly, the width of the main-lobe of the case beta ~= 9 is comparable to that of the Blackman window but the peak side-lobe level is much better. In practice, the Kaiser window is therefore a prferred choice over other types.

```
% Script File: win_Ex3.m
% -----------------------------
% Illustrates  dependence of spectral resolution on signal length
% Single Sinusoid
%
clear,clf
f0=0.2;                              % frequency of sinusoid (F(Hz)/Fs(Hz))
%
% Effect of changing the number of data samples
L=1;
N=256;                               % transform length
for N1=[25  50 100];                 % Rectanguular window width
        n=[0:N1-1];                  % index of time-samples
        v=cos(2*pi*f0*n);            % Sinusoid*rectangular window
        V=abs(fft(v,N));             % magnitude of spectrum spectrum
        k=[1:N/2];                   % positive frequency bins
        fk=(k-1)/N;                  % corresponding F(Hz)/Fs(Hz)
        subplot(3,1,L), stem(fk,V(k))
        xlabel(' f = F/Fs'), ylabel('|V(f)|'),
        title(sprintf('N = 256,  N = %2.0f', N1))
        L=L+1;
end
subplot, pause
% Effect of changing the transform length; number of data samples is constant
L=1;
N1=50;                               % Window (or Signal) Length
n=[0:N1-1];                          % index of time-samples
v=cos(2*pi*f0*n);                    % Sinusoid*rectangular window
for N= [64 128 256];                 % Transform Length
        V=abs(fft(v,N));             % magnitude of spectrum spectrum
        k=[1:N/2];                   % positive frequency bins
        fk=(k-1)/N;                  % corresponding F(Hz)/Fs(Hz)
        subplot(3,1,L), stem(fk,V(k))
        xlabel(' f = F/Fs'), ylabel('|V(f)|'),
        title(sprintf('N = %2.0f,   N1 = 50', N))
        L=L+1;
end
subplot
```

Comments on the output from win_Ex3.m

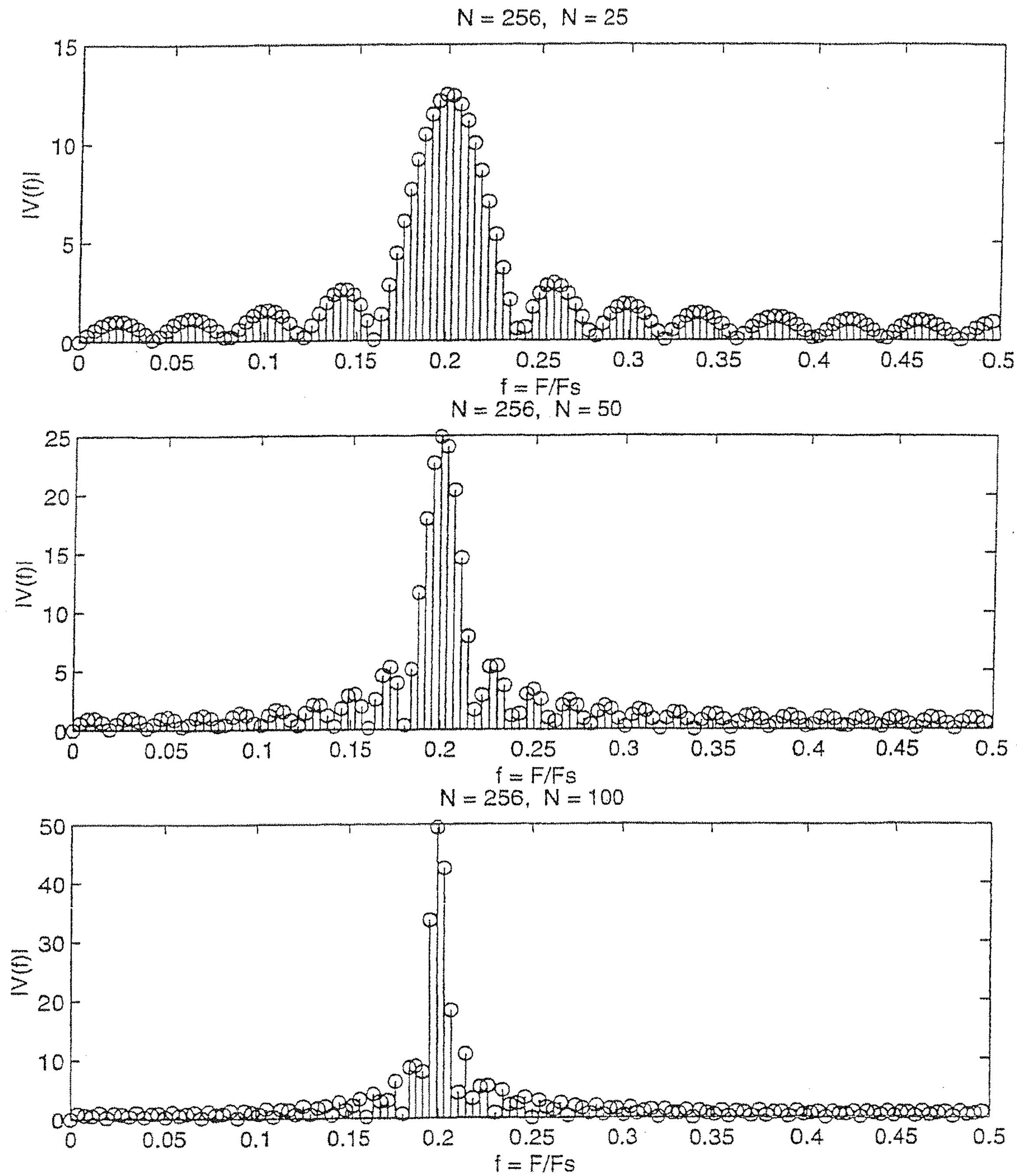
Note the leakage of spectral energy to all frequency bins. Note also that the spectral feature centered at f = 0.2 has a finite width, determined by the width of the rectangular window used, that is, by N1. Increasing the number of transform points N samples the spectrum at more points without changing its shape.

Output from win_Ex2

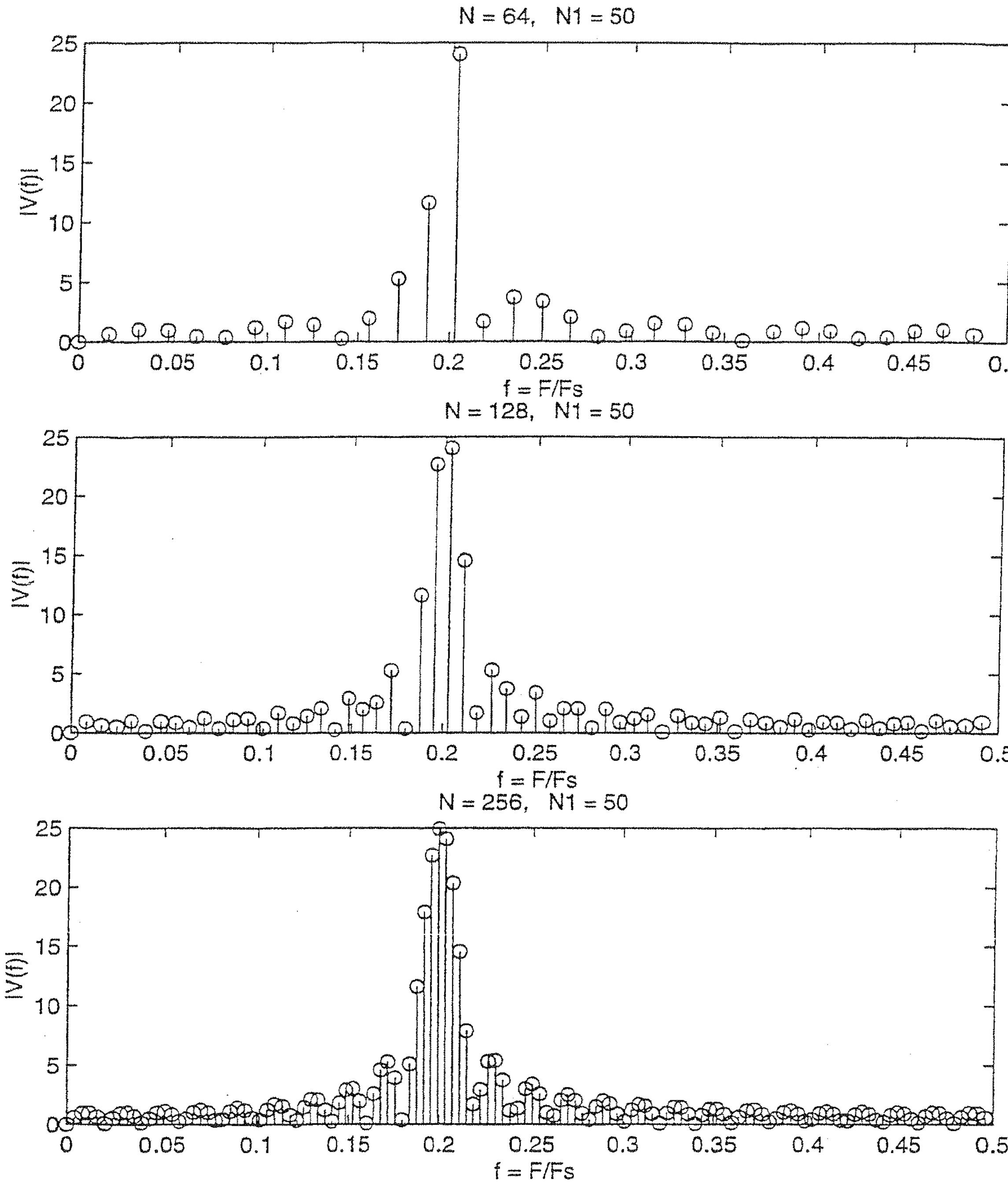


Output from Win_Ex3

variable data segment length, constant transform length
variable spectral resolution, same frequency sample spacing



Output from Win_Ex3
----
constant data segment length, variable transform length
same spectral resolution, variable frequency sample spacing

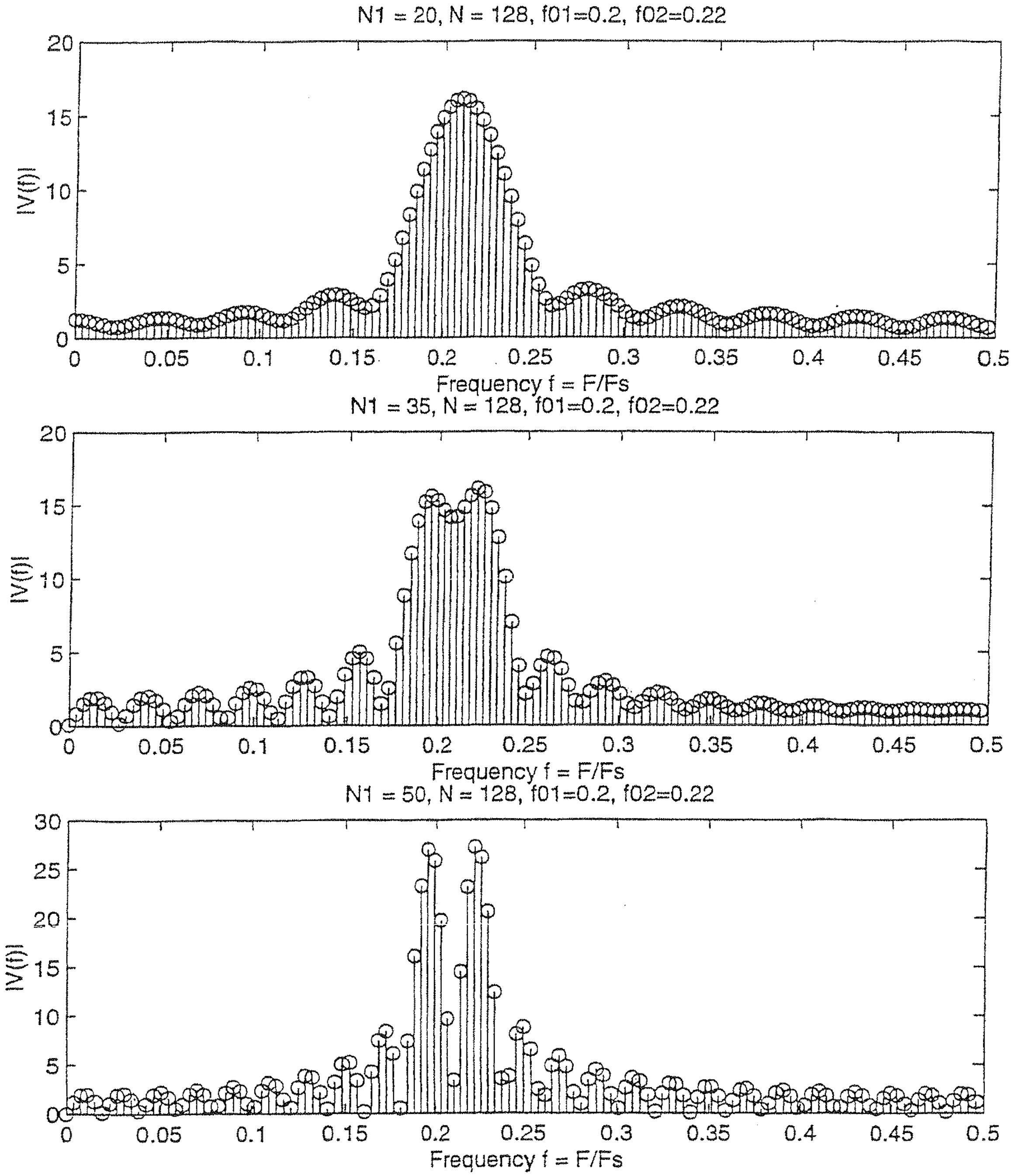


```
% Script File: win_Ex4.m
% ---------------------------
% Illustrates  dependence of spectral resolution on signal length
% Two Sinusoids
%
f01=0.20;
f02=0.22;
N=256;
%
L=1;
for N1=[ 20  35  50];
        n=[0:N1-1];
        v=cos(2*pi*f01*n) + cos(2*pi*f02*n);          % rectangular window
        V=abs(fft(v,N));                                % magnitude of spectrum spectrum
        k=[1:N/2];
        fk=(k-1)/N;
        subplot(3,1,L), stem(fk,V(k)),
        xlabel('Frequency f = F/Fs'), ylabel('|V(f)|'),
        title(sprintf( 'N1 = %2.0f, N = 128, f01=0.2, f02=0.22',N1))
        L=L+1;
end
```

Comments on the output of win_Ex4.m

Two sinusoids of about the same freuency are resolved if their frequency separation exceeds the 3 dB full width of the main lobe of the window used. The rectangular window provides the best frequency resolution; the 3 dB full width ~= 0.9/N1. The rectangular window suffers from a large leakage, however, as illustrated next.

output from Win_Ex4



```
% Script File: win_Ex5.m
% ----------------------------
% Illustrates effect of leakage on detectability of weak sinusoids
% Two Sinusoids, one much weaker than the other
%
f01=0.2;
f02=0.3;
N1=100;
n=[0:N1-1]';
A1=sqrt(2);                                     % strong sinusoid; avge power =1
A2=.01*A1;                                      % 40 dB meaker sinusoid
v=A1*cos(2*pi*f01*n) + A2*cos(2*pi*f02*n);      % strong + weak sinusoids
%
for L=1:3;
        if (L==1), w=boxcar(N1), end;                 % rectangular window
        if (L==1), w=kaiser(N1,6), end;               % Kaiser window
        if (L==3), w=blackman(N1),end;                % Blackman window
        v=v .* w;                                     % window weighted data
        N=512;                                        % large N ==> V(e^jw) ~=
        V(k)
        V=abs(fft(v,N));
        V=V/sum(w);                                   % normalize results
        V=20*log10(V);                                % V in dB
        %
        k=[1:N/2];
        fk=(k-1)/N;
        subplot(3,1,L), plot(fk,V(k),'-r'),grid on,
        axis([0 0.5 -100 0]);
        xlabel('Frequency f = F/Fs'), ylabel('|V(f)|'),
        if (L==1), title('Rectangular Window: N1 = 100, N = 512, f01=0.2, f02=0.3'),end
        if (L==2), title('Kaiser Window (beta=6): N1 = 100, N = 512, f01=0.2, f02=0.3'),end
        if (L==3), title('Blackman Window: N1 = 100, N = 512, f01=0.2, f02=0.3'),end
end
subplot, axis;
```

As you can see, the weak sinusoid is undetectable in the spectrum. Spectral leakage from the stong sinusoid completely masks the spectral contribution of the weaker sinusoid. We need to use a window that provides sufficient dynamic range (more than 40 dB). The Hamming window provides 41 dB; the Blackman window provides 57 dB. A Kaiser window of bata >= 6 (see plots above) aslo provides the required dynamic range. In all case, loss of spectral resolution should be expected, compared to that of the rectangular window. Results for the Blackman and Kaiser (beta = 6) windows are shown.

In real life, of course, you do not know a priori the answers to a particular problem, as in the case above. So you'll have to experiment to find window that gives good spectral reolution and, simultaneously, sufficient dynamic range to yield meaningful spectra for the problem at hand.

Output from Win_Ex5