

```

%Identify the frame for extracting pitch
S1=S(fix(length(S)/2):374+fix(length(S)/2));
S1=S1.*hamming(length(S1))';
subplot(2,5,i)
plot(S1)
subplot(2,5,i+1)
S2=abs(fft(S1));
plot(S2)
S3=20*log(S2);
subplot(2,5,i+2)
plot(S3)
S4=abs(iff(S3));
subplot(2,5,i+3)
plot(S4)
subplot(2,5,i+4)
S5=smooth(S4,5);
plot(S5)
S5=S5(1:1:188);
[p,q]=max(S5(9:1:length(S5)));
PF=FS/(q+9);

```

3.13 Formant Frequency Estimation

3.13.1 Formant Extraction Using Vocal Tract Model

The resonant frequencies of the vocal tract transfer function $V(Z)$ (refer (3.1)) are known as formant frequencies. Usually, there are three dominant resonant frequencies. They are known as formant-1 (270–730 Hz), formant-2 (840–2290 Hz), and formant-3 (1690–3010 Hz) frequencies. The vocal tract is modeled using LPC as discussed in the Sect. 3.3 and the formant frequencies are estimated by identifying the peak around the formant-1, formant-2, and formant-3 frequencies as demonstrated in Fig. 3.33. The 3-dB bandwidth of the resonant frequency does not increase more than 160, 200, and 300 Hz for formant-1, formant-2, and formant-3, respectively. The peak corresponds to the poles of the vocal tract filter. Hence, roots of the polynomial $A(Z) = \frac{1}{V(Z)}$, that have the bandwidth less than 160, 200, 300 Hz are estimated as the formant-1, formant-2, and formant-3 frequencies, respectively. The formant frequencies are estimated as follows.

1. Multiply the speech segment with the Hamming window.
2. Estimate 11th-order LPC and $A(Z)$ using the technique described in the Sect. 3.3.
3. Compute the roots of the polynomial $A(Z)$.
4. Compute the phase of the roots ranging from 0 to π . Let it be $\omega_1 \dots \omega_n$.
5. Compute the 512-point FFT of the $A(Z)$ and its normalized absolute values are represented as H (maximum amplitude of 1). Note the magnitude of H corresponding to $\omega_1 \dots \omega_n$ is obtained as $H(m_1) \dots H(m_n)$ where $m_i = \text{round}(\frac{\omega_i * 512}{\pi})$.

6. The maximum $\frac{\textit{bandwidth}}{2}$ at the formant frequencies does not exceed 160, 200, and 300 Hz, respectively, for formant-1, formant-2 and formant-3. For formant-1, this is computed in terms of number of samples as $N = \textit{round}(\frac{80*1024}{FS})$. For $FS = 12,500$, the value of N is computed as $N = 7$.
7. Collect the values of H for the range $m_1 - 2 * N$ to $m_1 + 2 * N$ as the vector H_1 . If the maximum value of H_1 is greater than 0.01, compute the index of the maximum value. Let it be p_1 . Compute the ratio $R_1 = \frac{H_1(p_1)}{H_1(p_1-N)}$. If the ratio R_1 is less than the threshold $th = 1.414$, declare \textit{omega}_1 as the formant frequency.
8. Repeat the step 7 for m_2 to m_n to obtain the remaining formant frequencies (if available).
9. If $p_1 < N$, repeat the step 7 with $N = 1$ with the threshold $th = \frac{1.414}{N}$.

```

%endpointdetection.m
[res1, res2, speechsegment, utforste, ltforste, ltforzcr]
    =endpointdetection(S, FS)
%->F1 270 to 730  F2-840 to 2290 F3-1690 to 3010 Hz
%160 HZ 200 Hz and 300 Hz
SEGLEN=FS*30*10^(-3);
S=speechsegment;
S1=S(fix(length(S)/2):SEGLEN-1+fix(length(S)/2));
S1=S1.*hamming(length(S1))';
subplot(2,2,1)
plot(S1)
L=lpc(S1, 11);
[H,W]=freqz(1, L);
subplot(2,2,3)
plot(abs(H))
H=abs(H)/max(abs(H));
R=roots(L);
subplot(2,2,4)
zplane(R)
P=[];
for i=1:1:11
    P=[P phase(R(i))]
end
[T]=find(P>=0);
POSP=P(T);
th1=round((1024*80)/FS);
th2=round((1024*100)/FS);
th3=round((1024*150)/FS);
F1=Forcollect(H,W,POSP,th1,FS);
F2=Forcollect(H,W,POSP,th2,FS);
F3=Forcollect(H,W,POSP,th3,FS);
F=[F1 F2 F3];
F=remdup(F);

function [F]=Forcollect(H,W,POSP,th1,FS)
%Forcollect.m
%Formant frequency collection for the particular formant range.
COLH=[]
COLW=[];
j=1;
for i=1:1:length(POSP)
    [I,J]= find((W-POSP(i))>0)

```

```

if (I>th1)
COLH{j}=H(I(1)-2*th1:1:I(1)+2*th1);
j=j+1;
COLW=[COLW W(I(1)-1)];
end
end
R=[];
F=[];
for i=1:1:length(COLH)
[M,N]=max(COLH{i});
if ((N>th1)&(COLH{i}(N)>0.1))
%R=[R ((COLH{i}(N)/COLH{i}(N-th1)+(COLH{i}(N)-COLH{i}(N+th1)))/2)];
temp=((COLH{i}(N)/COLH{i}(N-th1)+(COLH{i}(N)/COLH{i}(N+th1)))/2);
if(temp<1.414)
F=[F (COLW(i)*FS)/(2*pi)];
else
F=[F 0];
end
else
if(N>1)
temp=((COLH{i}(N)/COLH{i}(N-1)+(COLH{i}(N)/COLH{i}(N+1)))/2);
if(temp<(1.414/th1))
F=[F (COLW(i)*FS)/(2*pi)];
else
F=[F 0];
end
else
F=[F 0];
end
end
end

%remdup.m
function [G]=remdup(F)
%To remove duplicate in the array
F=sort(F);
G=[]
for i=1:1:length(F)-1
if((F(i+1)-F(i))~=0)
G=[G F(i)];
end
end
end
G=[G F(length(F))];
G=nonzeros(G);

```

3.13.2 Formant Extraction Using Homomorphic Filtering

In formant extraction using LPC modeling, the roots of the polynomial $A(Z)$ have to be computed. This increases the computational complexity. The formant extraction using homomorphic filtering has lesser computational complexity compared with LPC modeling technique. The speech signal in Z -domain be represented as $S(Z)$. This is obtained as the product of the excitation source $E(Z)$ and the transfer function of the vocal tract $V(Z)$ (i.e., $S(Z) = E(Z)V(Z)$). For voiced speech segment, $E(Z)$ is assumed to be the periodic impulse stream. The product of $E(Z)$ and $V(Z)$ is converted into additive quantity using log operator as follows.

$$C(Z) = 20 \log_e S(Z) = 20 \log_e E(Z) + 20 \log_e V(Z) \quad (3.26)$$

In $C(Z)$, $20 \log_e E(Z)$ corresponds to the high-frequency component. This is removed by smoothing $C(Z)$ to obtain $C_s(Z) \approx 20 \log_e V(Z)$ term. $V(Z)$ is further obtained as $e^{\frac{C_s(Z)}{20}}$. The peaks of $V(Z)$ around the formant frequency ranges which does not exceed the corresponding formant's bandwidth are declared as the formant frequencies. Figure 3.34 demonstrates the technique of extracting the formant frequencies using homomorphic filtering.

```
%forexthom.m
function [F]=forexthom(S1,FS)
S1=S1.*hamming(length(S1))';
[H,W]=freqz(1,lpc(S1,11),'whole');
figure
```

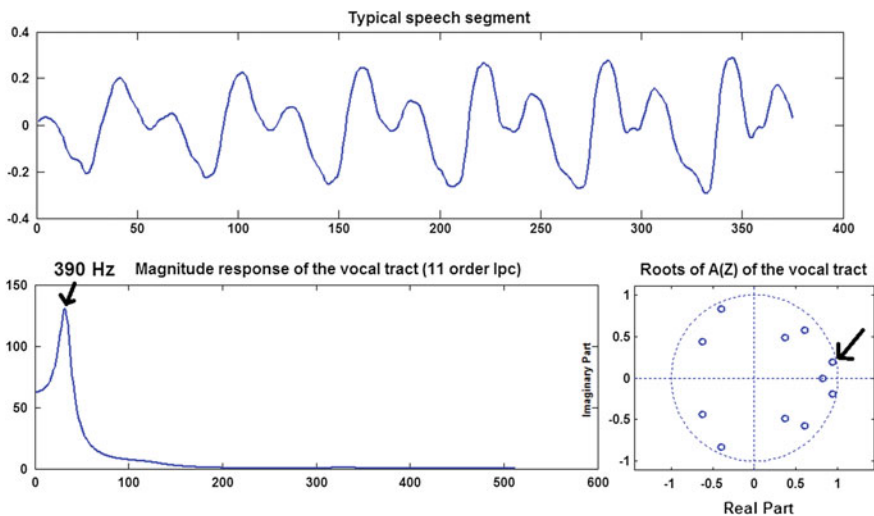


Fig. 3.33 Illustration of the estimation of the formant frequencies using LPC modeling

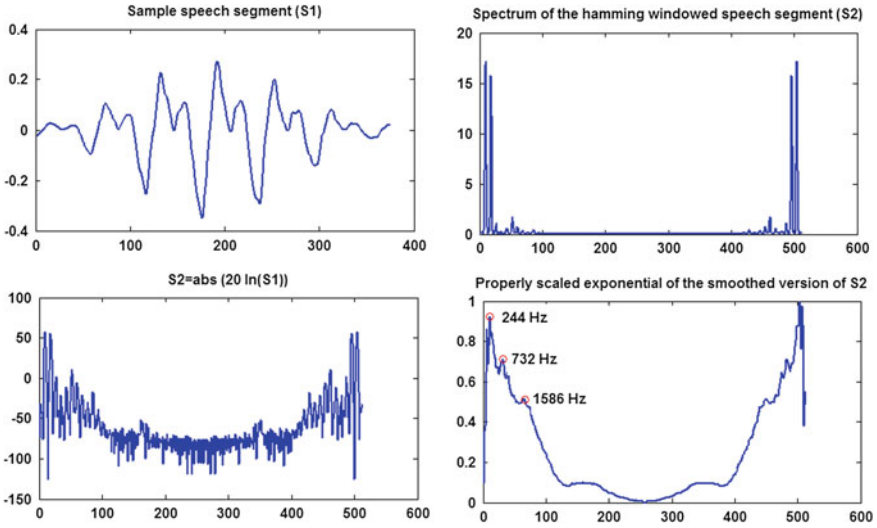


Fig. 3.34 Formant extraction using homomorphic filtering

```

plot(abs(H))
figure
subplot(2,2,1)
plot(S1)
subplot(2,2,2)
plot(abs(fft(S1,512)))
S2=20*log(abs(fft(S1,512)));
subplot(2,2,3)
plot(S2)
S3=smooth(S2,50);
S3=S3+abs(min(S3));
S3=S3/max(S3);
S3=exp(S3)-1;
S3=S3/max(S3);
subplot(2,2,4)
plot(S3)
hold on
th1=round((512*80)/FS);
p=[270/FS 730/FS]*512;
F1=collectf(p,S3,FS,th1);
th2=round((512*100)/FS);
p=[840/FS 2290/FS]*512;
F2=collectf(p,S3,FS,th2);
th3=round((512*150)/FS);
p=[1690/FS 3010/FS]*512;
    
```

```

F3=collectf(p,S3,FS,th1);
F=[F1 F2 F3];
F=remdup(F);

%collectf.m
function [formant1]=collectf(p,S3,FS,th1)
cons=th1;
for i=1:1:2
p1=round(p);
while((p1(i)-cons)<=0)
cons=cons-1;
end
[p2,q2]=max(S3(p1(i)-cons:1:p1(i)+cons));
pos1=p1(i)-cons+q2-1;
colpos(i)=pos1;
format(i)=((pos1-1)*FS)/512;
end

[m,n]=max(S3(colpos(1:2)));
n1=n+colpos(1);
if ((n1-th1)<=0)
bwth=(S3(n1)/S3(n1-1)+S3(n1)/S3(n1+1))/2;
if(bwth<(1.414/th1))
formant1=((colpos(n)-1)*FS)/512;
plot(colpos(n),S3(colpos(n)),'ro')
else
disp('there is no formant ')
end
else
bwth=(S3(n1)/S3(n1-th1)+S3(n1)/S3(n1+th1))/2;
if(bwth<1.4141)
formant1=((colpos(n)-1)*FS)/512;
plot(colpos(n),S3(colpos(n)),'ro')
else
disp('there is no formant ')
end
end
end

```