

ŞİRNAK ÜNİVERSİTESİ BİLGİSAYAR MÜHENDİSLİĞİ

SİNYALLER VE SİSTEMLER EV ÖDEVİ

ÖDEV: Ses Sinyallerinin Frekans Analizi ve Filtreleme

Amaç:

Bu ödevde öğrenciler, bir ses sinyalini analiz etmek, frekans bileşenlerini incelemek ve çeşitli filtreler uygulamak için MATLAB kullanacaklardır. Proje sırasında sinyalin Fourier dönüşümünü ve filtreleme işlemlerini yapacaklar, eksik formülleri ve kodları kendileri tamamlayacaklardır.

Adım 1: Ses Sinyalinin Okunması ve Zaman Domaininde Gösterilmesi

1. Bir ses dosyasını audioread fonksiyonu ile okuyun.
2. Sinyalin zaman eksenini hesaplayın ve zaman domaininde grafiğini çizin.
3. Grafiği yorumlayınız.

Verilen Kod:

```
% Ses dosyasını oku (dosya adını kendiniz belirleyin):
[orjinal_sinyal, Fs] = audioread('_____.wav'); % Ses dosyasını tamamlayın

% Zaman eksenini hesaplayın:
t = (0:length(orjinal_sinyal)-1)/_____; % Örnekleme frekansını kullanarak zamanı hesaplayın

% Orijinal sinyali zaman domaininde çizin:
figure;
subplot(3,1,1);
plot(_____, _____); % Zaman ve sinyal eksenlerini tamamlayın
title('Orijinal Ses Sinyali (Zaman Domaini)');
xlabel('Zaman (s)');
ylabel('Genlik');
```

Teori:

- `audioread`: Ses dosyasını okur.
- Zaman eksenini: $t = \frac{n}{F_s}$, burada n örnek numarası ve F_s örnekleme frekansıdır.

Adım 2: Fourier Dönüşümü ile Frekans Domaini Analizi

1. Sinyalin Fourier dönüşümünü hesaplayın.
2. Frekans spektrumunu çizmek için eksik kodu tamamlayın.
3. Grafikleri yorumlayınız.

Verilen Kod:

```
% Fourier dönüşümünü hesaplayın:
N = length(orjinal_sinyal);
frekans_spektrumu = _____(orjinal_sinyal); % Fourier dönüşümü için eksik fonksiyonu tamamlayın
frekanslar = linspace(0, Fs, _____); % Frekans aralığını hesaplayın

% Frekans domaininde grafiği çizin:
subplot(3,1,2);
plot(_____, _____(1:N/2), abs(_____(1:N/2))); % Frekans ve spektrumu çizin
title('Orijinal Ses Sinyali (Frekans Domaini)');
xlabel('Frekans (Hz)');
ylabel('Genlik');
```

Teori:

- Fourier Dönüşümü (DFT): Sinyalin zaman domaininden frekans domainine geçişini sağlar.

$$X(f) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi \frac{kn}{N}}$$

- MATLAB'de Fourier dönüşümü için `fft` fonksiyonu kullanılır.

Adım 3: Gürültü Ekleme

- Sinyale rastgele bir beyaz gürültü ekleyin.
- Gürültülü sinyali zaman domaininde çizin.
- Grafikleri yorumlayınız.

Verilen Kod:

```
% Gürültü ekleme:
gürültü = 0.02 * _____(size(orjinal_sinyal)); % Rastgele beyaz gürültü ekleyin
gürültülü_sinyal = orjinal_sinyal + _____; % Gürültü ekleyin

% Gürültülü sinyali zaman domaininde çizin:
subplot(3,1,3);
plot(_____, _____); % Eksikleri doldurarak grafiği çizin
title('Gürültülü Ses Sinyali');
xlabel('Zaman (s)');
ylabel('Genlik');
```

Teori:

- Beyaz Gürültü: Sinyal üzerinde rastgele, her frekansta eşit güçte gürültü ekleyen bir sinyaldir. MATLAB'de `randn` fonksiyonu kullanılır.

Adım 4: Filtre Tasarımı ve Uygulama

1. Alçak geçiren, yüksek geçiren ve bant geçiren filtreler tasarlayın ve uygulayın.
2. Filtre kesim frekanslarını ve parametreleri hesaplayarak kodu tamamlayınız.
3. Sonuçları yorumlayınız.

Verilen Kod:

```
% Filtre tasarımı:
fc_low = _____; % Alçak geçiren filtre kesim frekansı (Hz)
fc_high = _____; % Yüksek geçiren filtre kesim frekansı (Hz)

% Alçak geçiren filtre
Wn_low = fc_low / (Fs / 2); % Normalize kesim frekansı
[b_low, a_low] = butter(6, Wn_low, 'low');
filtrelenmis_sinyal_low = filter(_____, _____, gürültülü_sinyal);

% Yüksek geçiren filtre
Wn_high = fc_high / (Fs / 2); % Normalize kesim frekansı
[b_high, a_high] = butter(6, Wn_high, 'high');
filtrelenmis_sinyal_high = filter(_____, _____, gürültülü_sinyal);

% Bant geçiren filtre
Wn_band = [fc_low fc_high] / (Fs / 2);
[b_band, a_band] = butter(6, Wn_band, 'bandpass');
filtrelenmis_sinyal_band = filter(_____, _____, gürültülü_sinyal);
```

Teori:

- Butterworth Filtresi: İdeal bir filtreye yakın performans sağlar ve geçiş bandında düzgün bir yanıt verir.

$$H(s) = \frac{1}{\sqrt{1 + (s/\omega_c)^{2n}}}$$

- MATLAB'de `butter` fonksiyonu ile filtre tasarlanır. Öğrenciler `Wn_low` ve `Wn_high` gibi kesim frekanslarını normalize ederek hesaplayacaklar.

Adım 5: Filtrelenmiş Sinyallerin Zaman Domaininde Gösterimi

1. Alçak geçiren, yüksek geçiren ve bant geçiren filtrelerin sonuçlarını zaman domaininde çizin.
2. Grafikleri Yorumlayınız.

Verilen Kod:

```
% Zaman domaininde filtrelenmiş sinyalleri çizmek:
figure;
subplot(3,1,1);
plot(____, ____); % Alçak geçiren filtrelenmiş sinyali tamamlayın
title('Alçak Geçiren Filtrelenmiş Sinyal');
xlabel('Zaman (s)');
ylabel('Genlik');

subplot(3,1,2);
plot(____, ____); % Yüksek geçiren filtrelenmiş sinyali tamamlayın
title('Yüksek Geçiren Filtrelenmiş Sinyal');
xlabel('Zaman (s)');
ylabel('Genlik');

subplot(3,1,3);
plot(____, ____); % Bant geçiren filtrelenmiş sinyali tamamlayın
title('Bant Geçiren Filtrelenmiş Sinyal');
xlabel('Zaman (s)');
ylabel('Genlik');
```

Adım 6: Sesin Dinlenmesi

1. Filtrelenmiş sinyalleri dinleyin.
2. Çıkan sesi Yorumlayınız.

Verilen Kod:

```
% Sonuçları dinleme:
disp('Orijinal Sinyal');
sound(____, Fs); % Orijinal sinyali tamamlayın
pause(5);

disp('Gürültülü Sinyal');
sound(____, Fs); % Gürültülü sinyali tamamlayın
pause(5);

disp('Alçak Geçiren Filtrelenmiş Sinyal');
sound(____, Fs); % Alçak geçiren filtrelenmiş sinyali tamamlayın
pause(5);

disp('Yüksek Geçiren Filtrelenmiş Sinyal');
sound(____, Fs); % Yüksek geçiren filtrelenmiş sinyali tamamlayın
pause(5);

disp('Bant Geçiren Filtrelenmiş Sinyal');
sound(____, Fs); % Bant geçiren filtrelenmiş sinyali tamamlayın
pause(5);
```