BCA 607 Hareket Analizi Sistemleri

Kinematik Analiz - Filtreleme

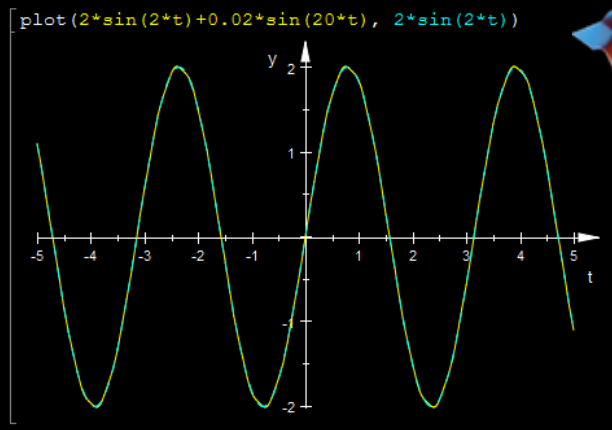
De Motu Animalium G.Borelli (1680)

SERDAR ARITAN

serdar.aritan@hacettepe.edu.tr

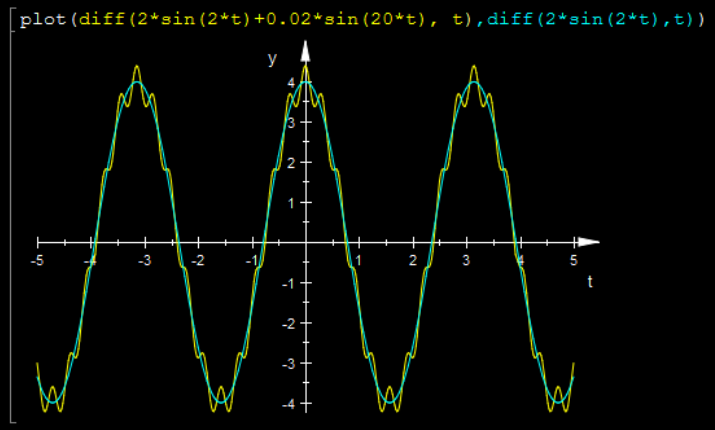
Biyomekanik Araştırma Grubu
www.biomech.hacettepe.edu.tr
Spor Bilimleri Fakültesi
www.sbt.hacettepe.edu.tr
Hacettepe Universitesi, Ankara, Türkiye
www.hacettepe.edu.tr

İvme hesaplamaları gürültüye karşı hassastır



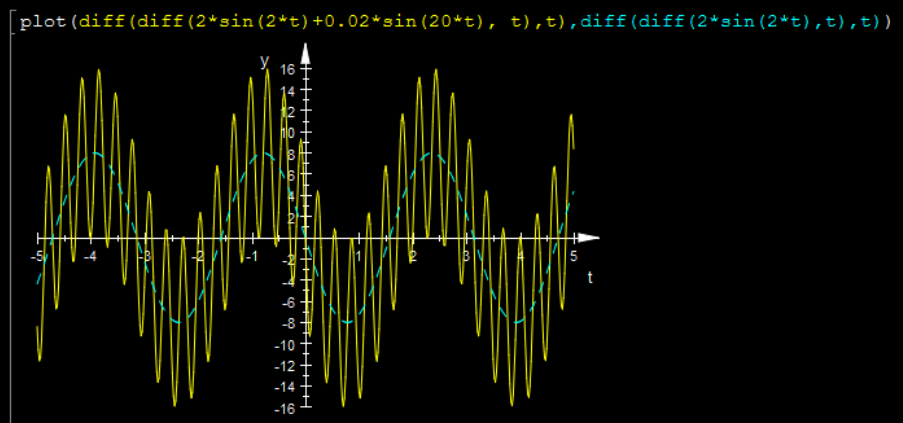
$$r = \underbrace{2\sin(2t)}_{\text{Sinyal}} + \underbrace{0.02\sin(20t)}_{\text{Gürültü}}$$

%1 gürültü



$$\mathbf{dr/dt} = 4\cos(2t) + 0.4\cos(20t)$$

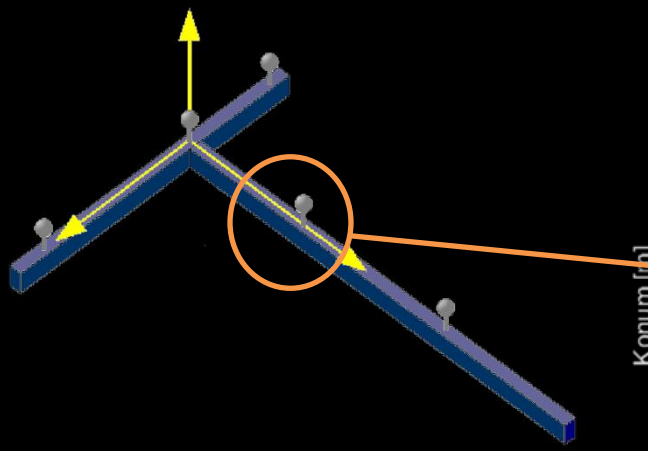
% 10 gürültü

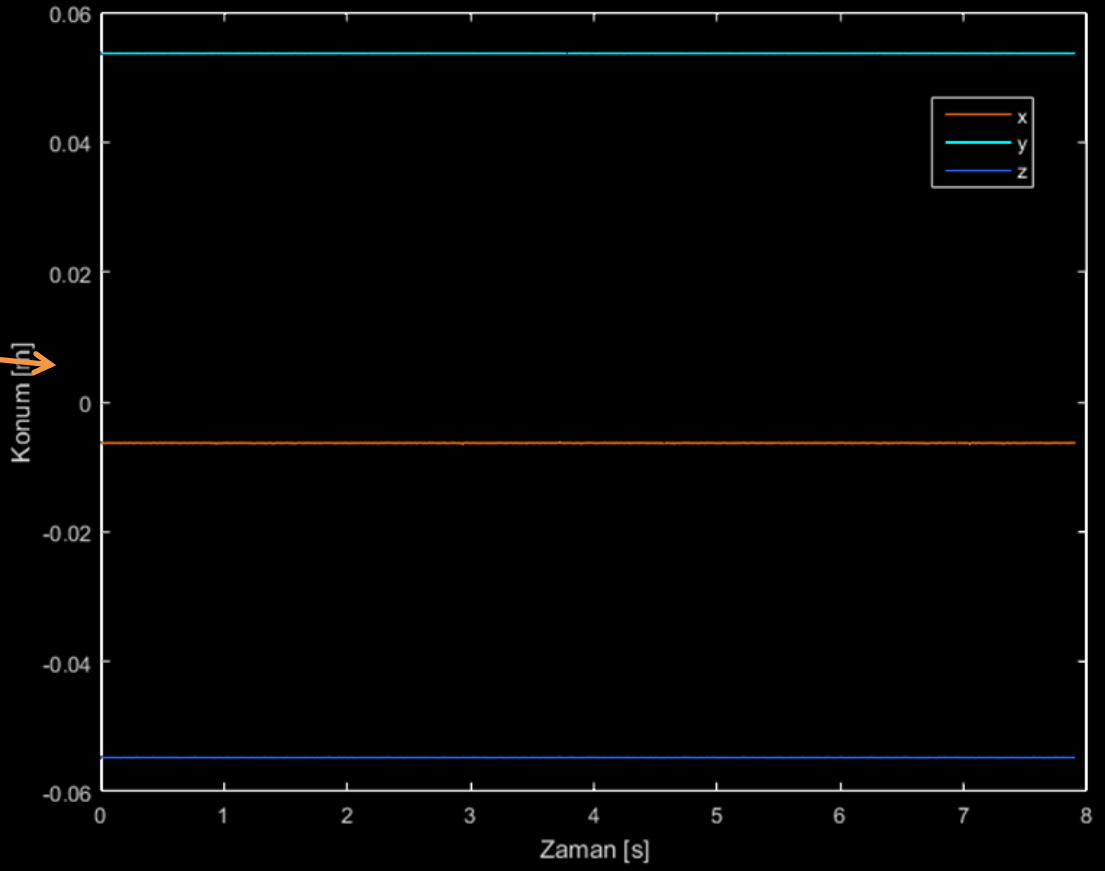


$$\mathbf{d^2r/dt^2} = -8\sin(2t) - 8\sin(20t)$$

%100 gürültü

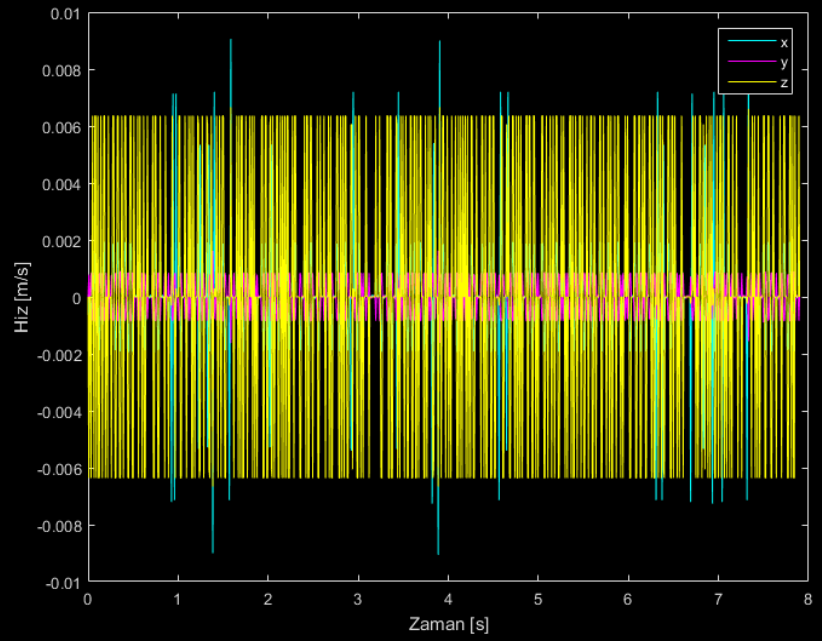
Vicon Kalibrasyon Çubuğu: hareketsiz



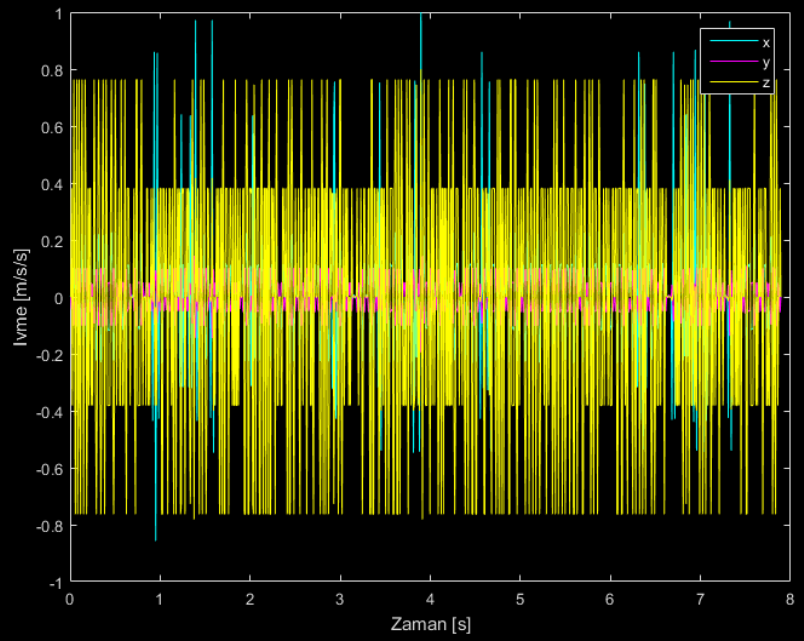


Vicon Kalibrasyon Çubuğu: hareketsiz

hız



ivme



Süzgeçler (Filters)

Hareket yakalama tamamlandıktan sonra yapılacak ilk iş verinin temizlenmesi olacaktır. Süzgeç (Filter), verideki yüksek veya alçak frekansları azaltan matematiksel bir uygulamadır. Süzgeç verinin bütünlüğünü bozmadan ani değişimleri veya sürüklenmeleri veriden ayıklar. Günümüzde hareket analizi verilerinde birçok farklı süzgeç algoritması kullanılmaktadır. Hareket yakalama sistemi herhangi bir sebeple yansıtıcıları göremediğinde yanlış bir tahmin yaparak veride ani bir değişime (spike) sebep olur. Alçak geçiren süzgeçler (Low-pass filters) bu ani yükselişleri hızlı bir şekilde ortadan kaldırır. Ayrıça, kötü kalibrasyon yapıldığı durumlarda yüksek frekanslı gürültüler hareket analizi verilerine de eklenir. Butterworth filter alçak geçiren süzgeç için çok güzel bir örnek olacaktır

Süzgeçler (Butterworth Filter)

Matlab da butterworth analog ve sayısal süzgeç tasarımı için `butter` komutu kullanılır.

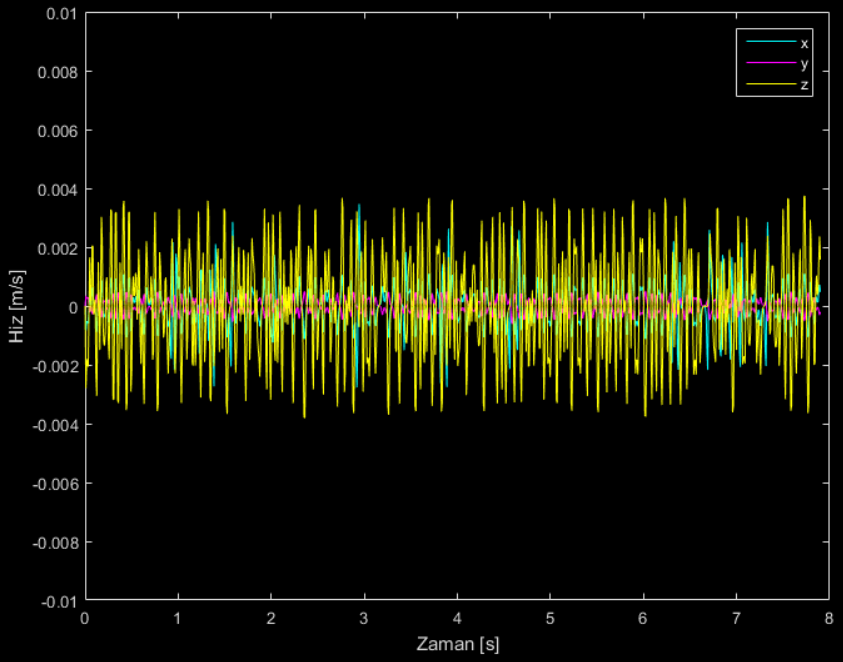
[b, a] = butter(n, Wn, 'ftype') **`Butter`** komutu alçak geçiren, band geçiren, yüksek geçiren ve band durduran **n**. dereceden butterworth süzgeç tasarlamak için kullanılır. Buradaki n filtrenin derecesidir. **`Butter`** komutu n+1 uzunluğunda olan ve filtre katsayılarını içeren b (numerator), a (denominator) vektörlerini geri döndürür. **Wn** kesim frekansıdır ve **Wn** değeri normalize edilmelidir. Örneğin kinematik verideki gibi 50 Hz lik bir sinyalde 6 Hz i filtre ederek almak için 50 Hzn yarısı alınarak 6 ya bölünür. Wn = 6/ 25 = 0.24 olmalıdır.

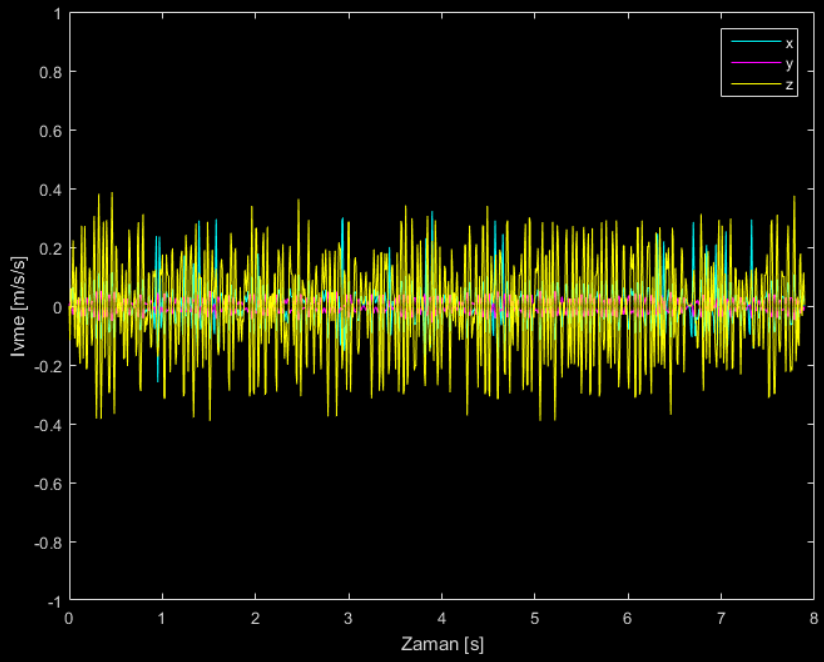
```
% Butterworth süzgeç parametrelerini belirle.
[b,a]= butter(2,6/25,'low');
% Verileri suzgecten gecir.
filterData = filtfilt(b,a,rawData);

% y = filtfilt(b,a,x) performs zero-phase digital filtering
by processing the input data, x, in both the forward and
reverse directions
```

Süzgeçler (Butterworth Filter)

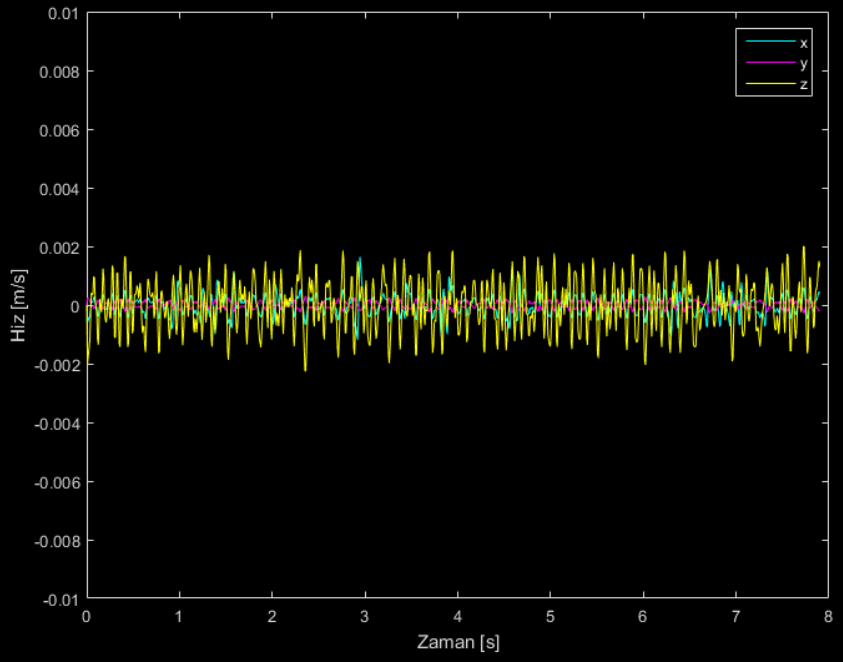
```
% Butterworth süzgeç parametrelerini belirle.
[b,a]= butter(1,20/60,'low');
% Verileri suzgecten gecir.
filterData = filtfilt(b,a,rawData);
```

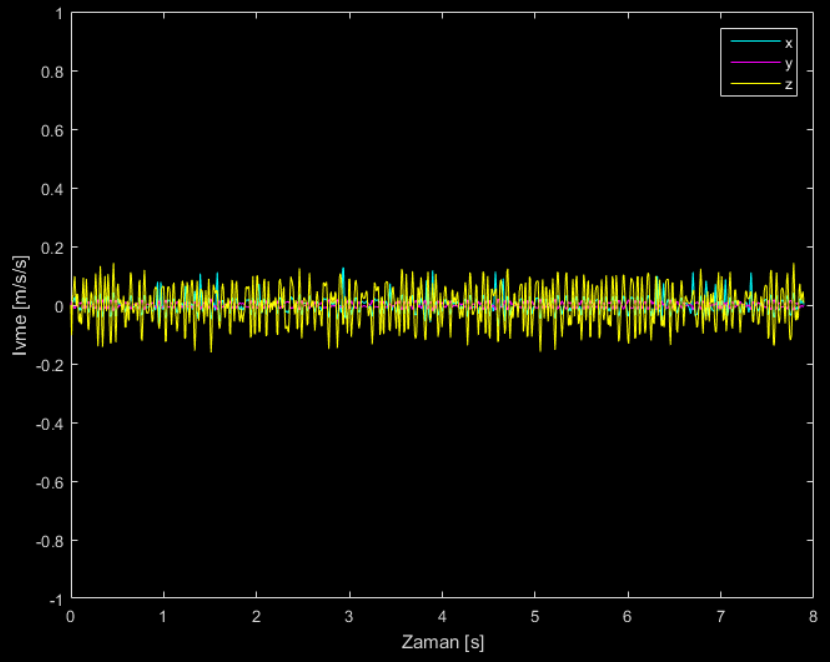




Süzgeçler (Butterworth Filter)

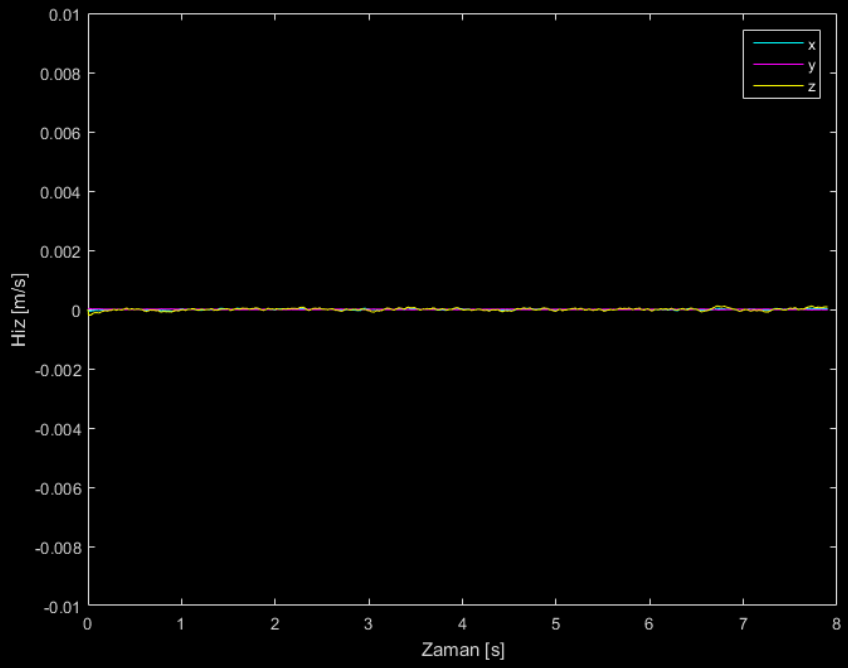
```
% Butterworth süzgeç parametrelerini belirle.
[b,a]= butter(1,10/60,'low');
% Verileri suzgecten gecir.
filterData = filtfilt(b,a,rawData);
```

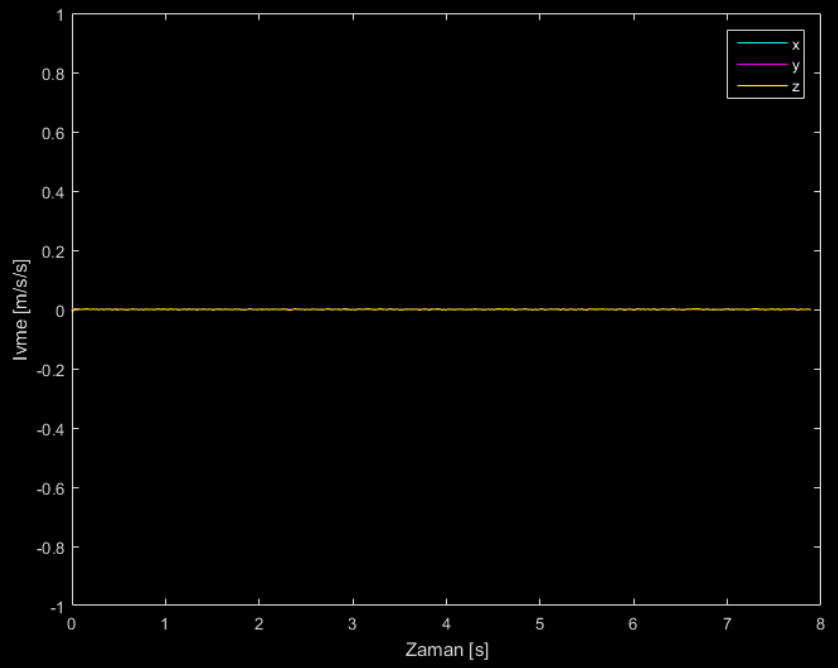




Süzgeçler (Butterworth Filter)

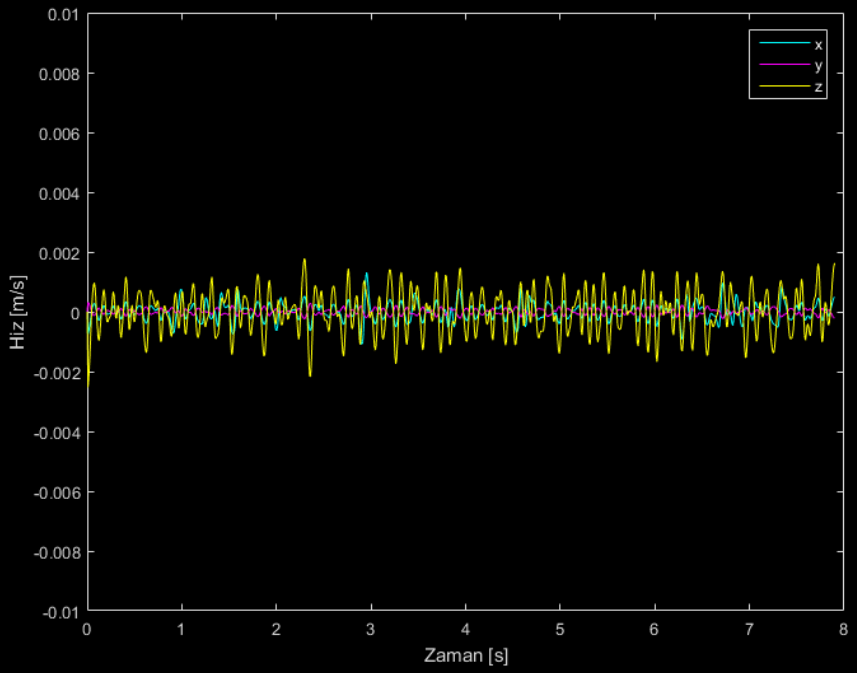
```
% Buuterworth süzgeç parametrelerini belirle.
[b,a]= butter(1,1/60,'low');
% Verileri suzgecten gecir.
filterData = filtfilt(b,a,rawData);
```

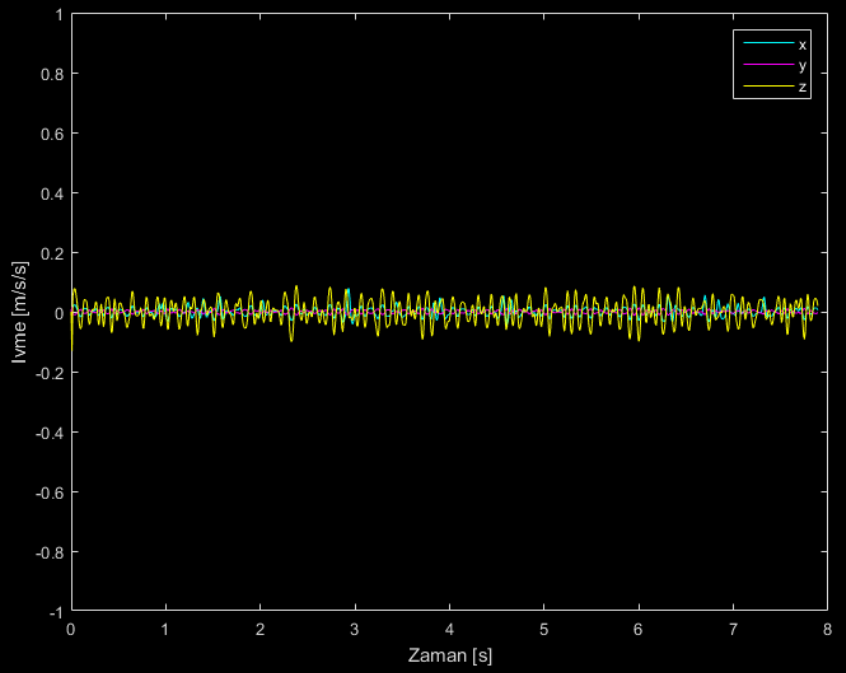




Süzgeçler (Butterworth Filter)

```
% Buuterworth süzgeç parametrelerini belirle.
[b,a]= butter(2,10/60,'low');
% Verileri suzgecten gecir.
filterData = filtfilt(b,a,rawData);
```





Süzgeçler (Butterworth Filter)

Örnek : El Hareketi için Farklı Süzgeç Kesme Frekansları

5 Hz Ağır Süzgeç (Heavy Filtering):
Hızlı ve ani hareketleri oldukca fazla yumuşataçaktır. Ancak normal kontrollü el hareketinin ana özelliklerini izlenebilecektir.

10 Hz Orta Süzgeçleme (Medium Filtering):
Elin normal ve orta hızlı hareketlerinin ana özelliklerini kaybolmaz iken bazı yüksek frekanslarda aralıklı bozulmalar olabilir.

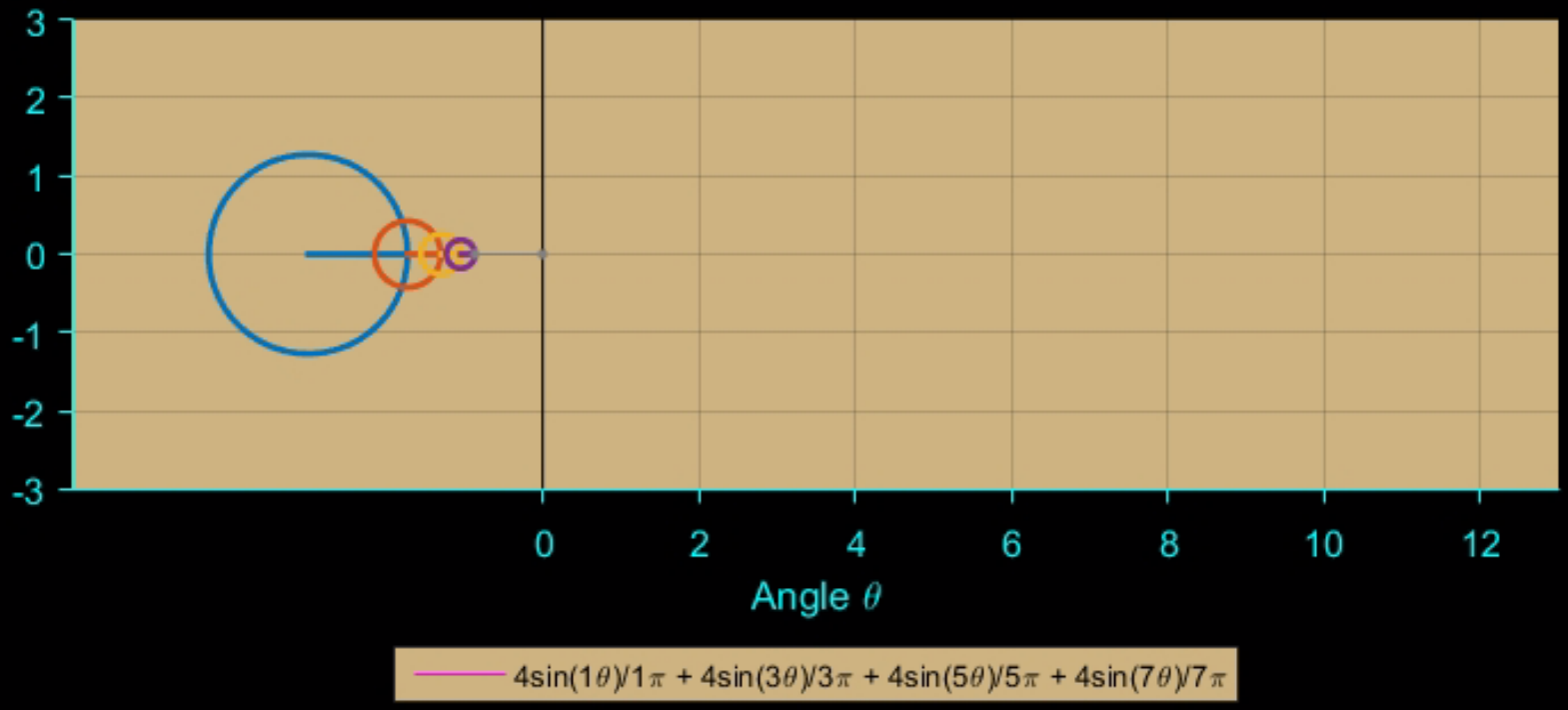
15 Hz Hafif Süzgeçleme (Light Filtering):
Elin normal ve hızlı hareketlerinin ana özelliklerini korumaktadır. Sadece çok yüksek hızlardaki ani hareketlerde bölgesel bozulmalar olabilir.

Süzgeçler (Butterworth Filter)

- Genellikle iki-yönlü filtre kullanılmalıdır, (Matlab'da **`filtfilt`**), aksi durumda süzgeçlenmiş veride faz farkı olacaktır.
- İki-yönlü filtre kullanıldığında süzgeçin derecesinin (order) 2 katı arttığı unutulmamalıdır.
- Süzgeçlenmiş ve süzgeçlenmemiş veriler mutlaka cizilerek uygunluğu kontrol edilmelidir.
- Çalışma rapor edildiğinde kullanılan süzgeçin tipi (alçak-yüksek), kesme frekansı ve katsayısı belirtilmelidr. *Örneğin; kesme frekansı 20Hz olan 2. dereceden iki-yönlü alçak geçiren Butterworth süzgeç konum verisine uygulanmıştır.*
- Hız ve ivme hesaplamalarında süzgeçler konum verisine uygulanmalıdır.

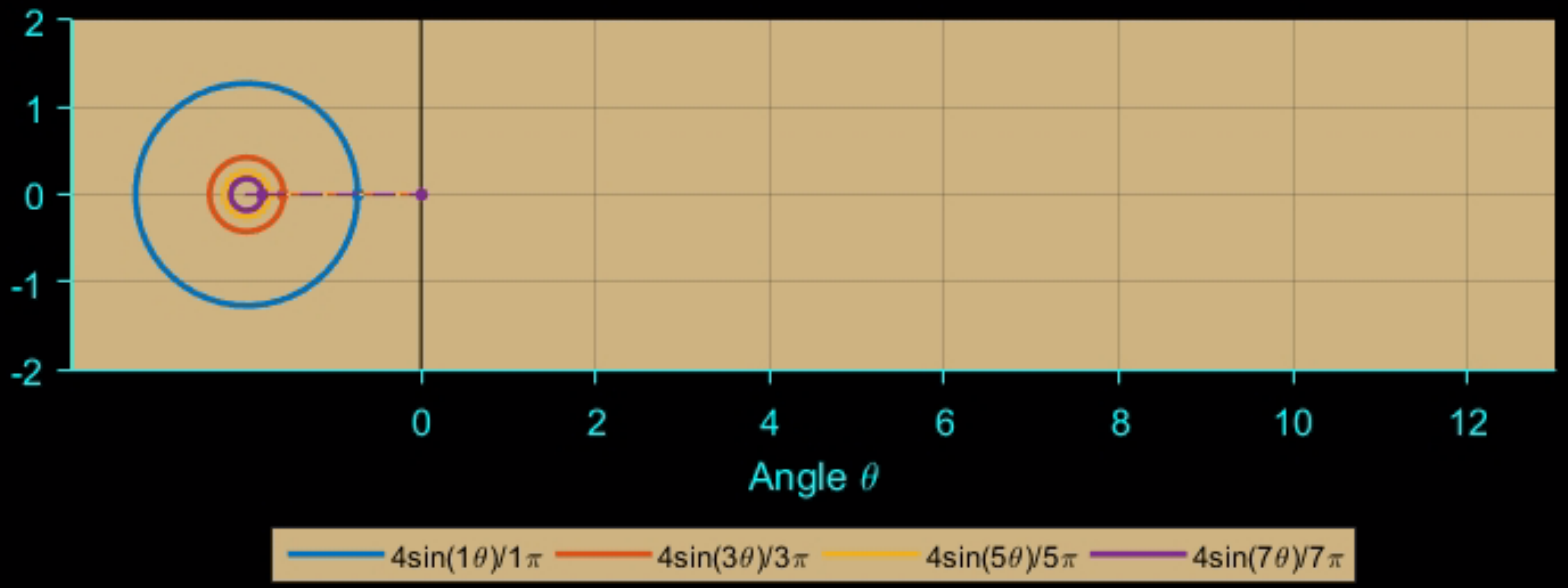
İzge Analizi

Fast Fourier Transformation (FFT)



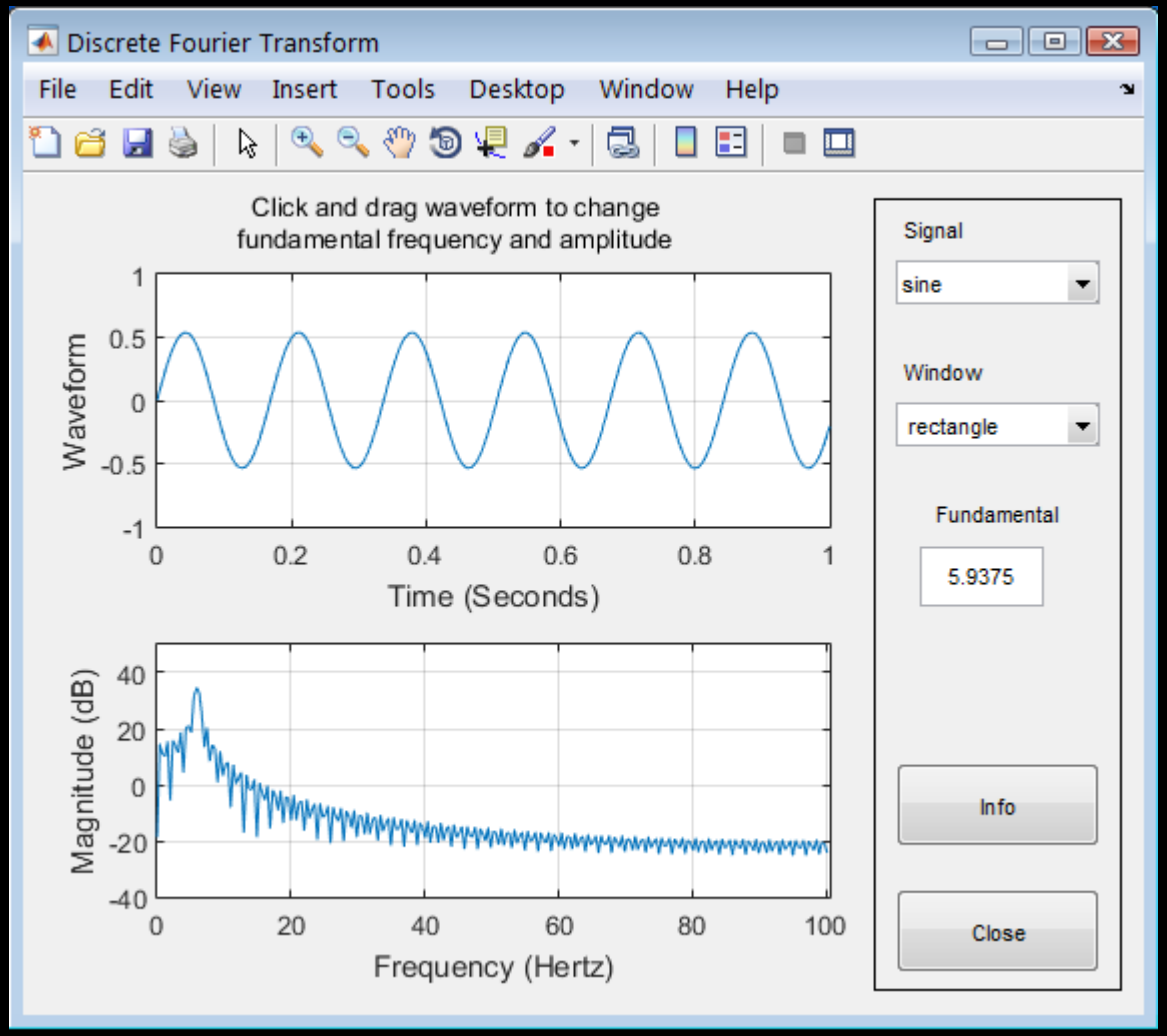
İzge Analizi

Fast Fourier Transformation (FFT)

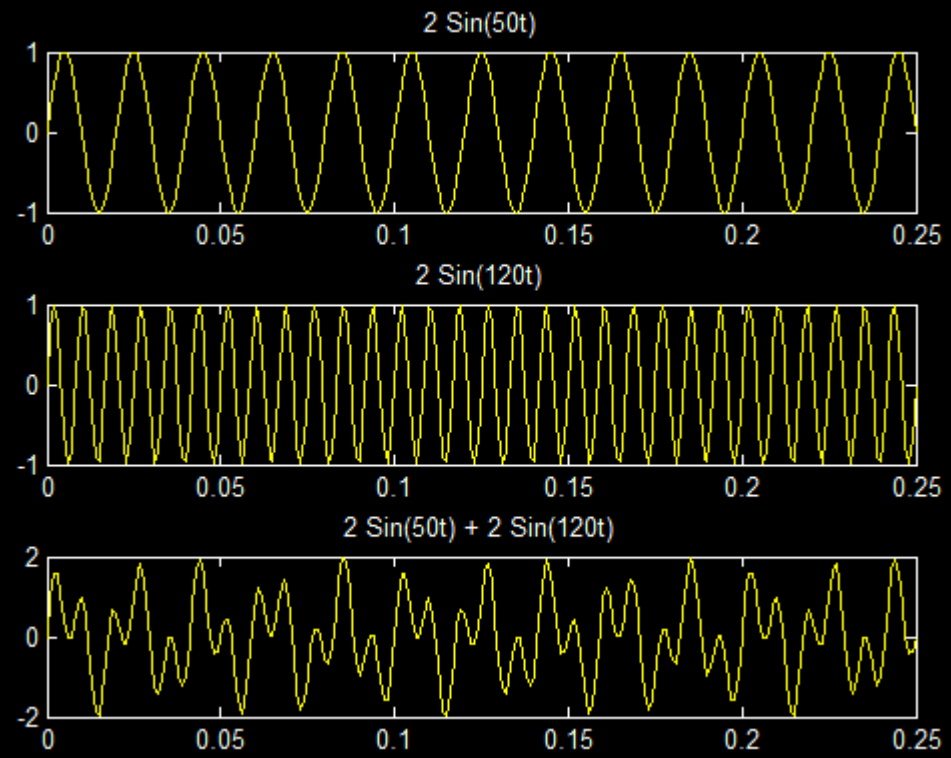


Ayrık Zamanlı Fourier Dönüşümü

>> DFTExample

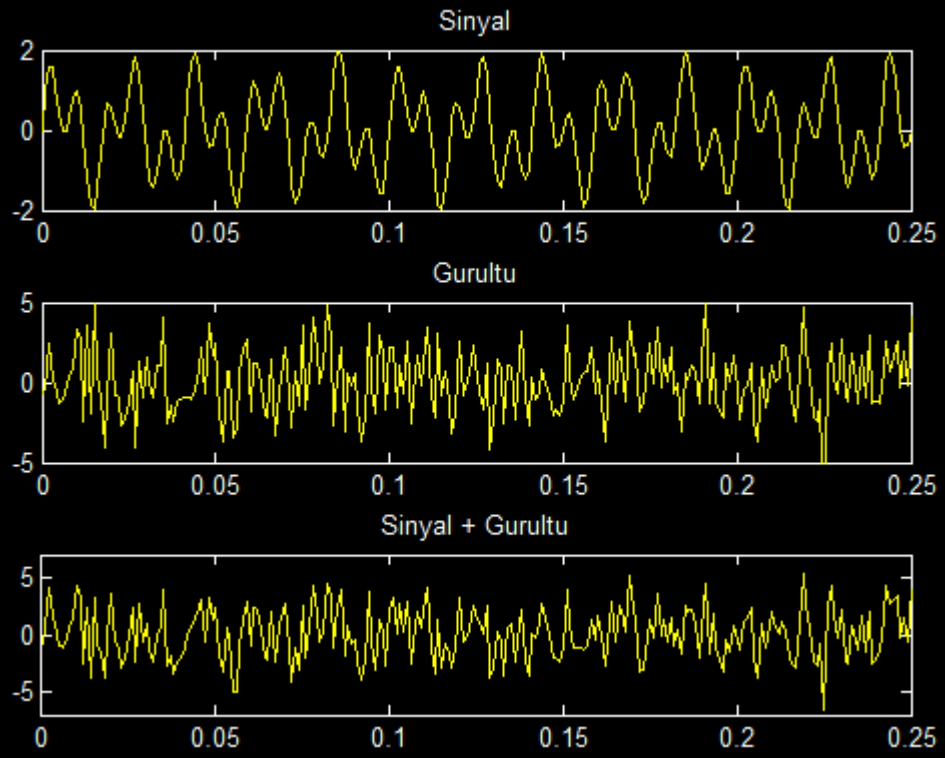


Ayrık Zamanlı Fourier Dönüşümü



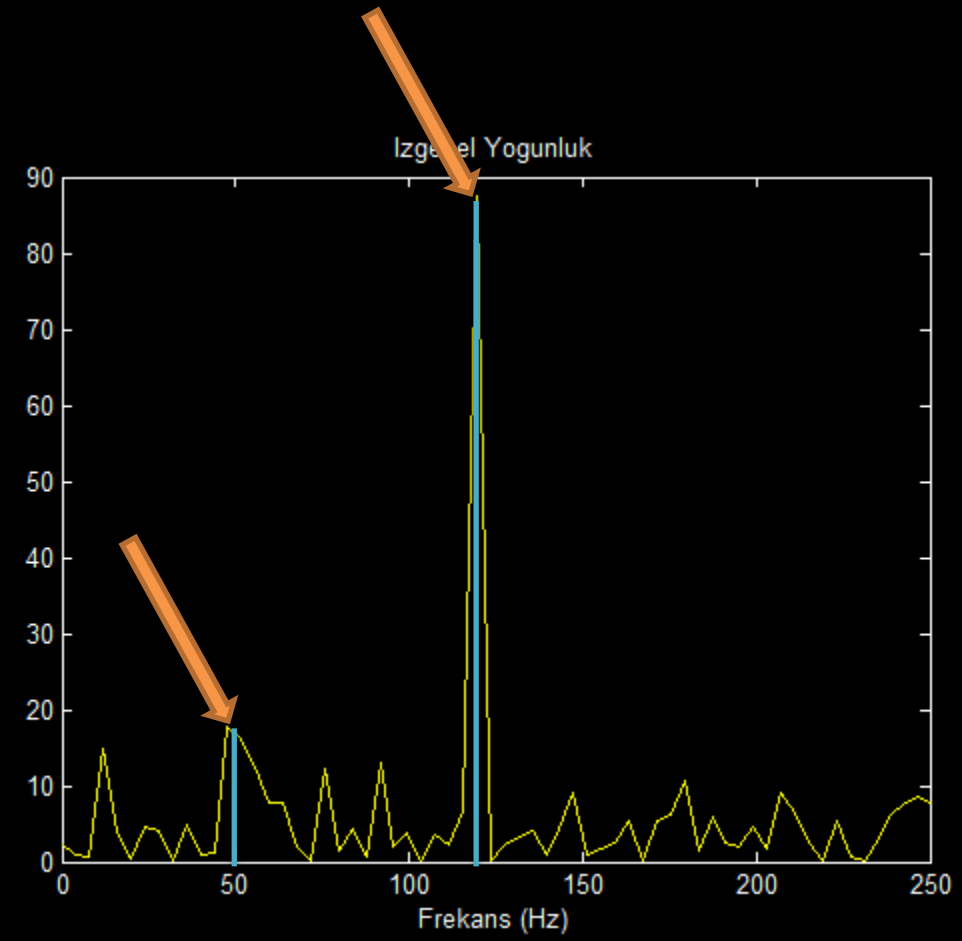

```
t = 0:.001:.25;
x1 = sin(2*pi*50*t);
x2 = sin(2*pi*120*t);
x3 = x1 + x2;
figure,
subplot(3,1,1);
plot(t, x1), title('2 Sin(50t)')
subplot(3,1,2);
plot(t, x2),title('2 Sin(120t)')
subplot(3,1,3);
plot(t, x3), title('2 Sin(50t) + 2
Sin(120t)')
```

Ayrık Zamanlı Fourier Dönüşümü




```
gurultu = 2*randn(size(t));
sinyal = x3 + gurultu;
figure,
subplot(3,1,1);
plot(t, x3), title('Sinyal')
subplot(3,1,2);
plot(t, gurultu), ylim([-5 5]),
title('Gurultu')
subplot(3,1,3);
plot(t, sinyal), ylim([-7 7]),
title('Sinyal + Gurultu')
```

Ayrık Zamanlı Fourier Dönüşümü




```
Y = fft(sinyal, 251);
Pyy = Y.*conj(Y)/251;
f = 1000/251*(0:127);
figure, plot(f, Pyy(1:128)),
xlim([0 250])
title('Izgesel Yogunluk')
xlabel('Frekans (Hz)')
```

Y = fft(x) returns the discrete Fourier transform (DFT) of vector x, computed with a fast Fourier transform (FFT) algorithm.

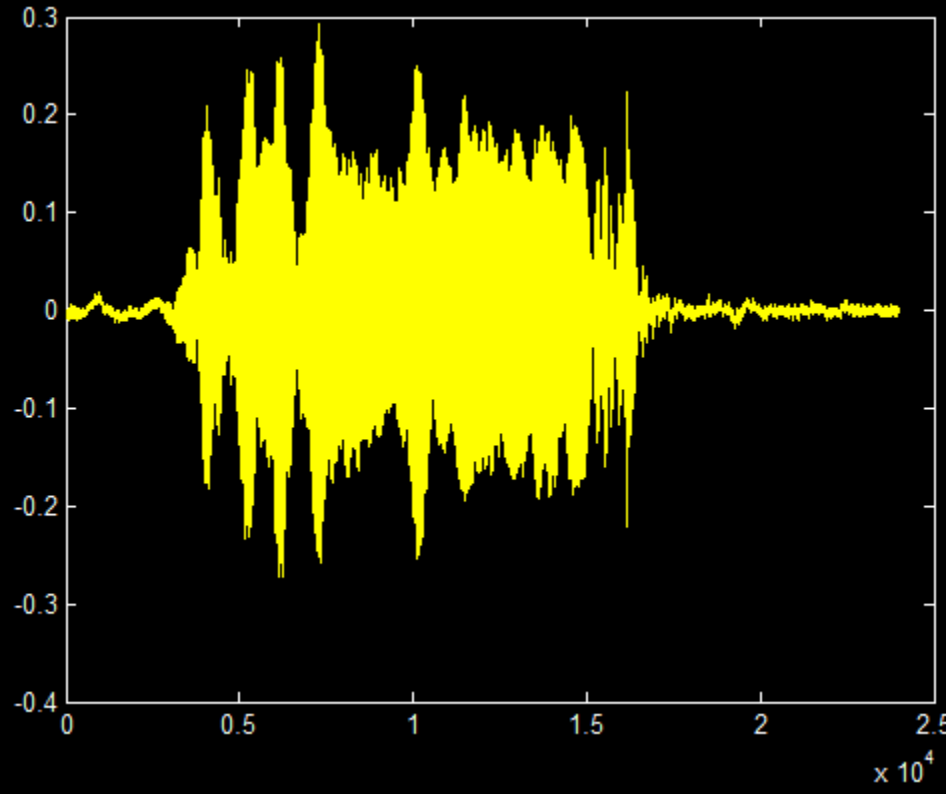
Konum Verisinin İzge Analizi

Fast Fourier Transformation (FFT)

```
function Y = calculate_fft(rawdata, freq)
Fs = freq;                          % Sampling frequency
T = 1/Fs;                           % Sample time
L = length(rawdata);                % Length of signal
t = (0:L-1)*T;
NFFT = 2^nextpow2(L); % Next power of 2 from length of y
Y = fft(rawdata,NFFT)/L;
f = Fs/2*linspace(0,1,NFFT/2);
% Plot single-sided amplitude spectrum.
figure, plot(f,2*abs(Y(1:NFFT/2)),'k:'),grid on;
title(['Amplitude Spectrum of Ch1'])
xlabel('Frequency (Hz)')
ylabel('|Y(f)|');
```

Hızlı Fourier Dönüşümü

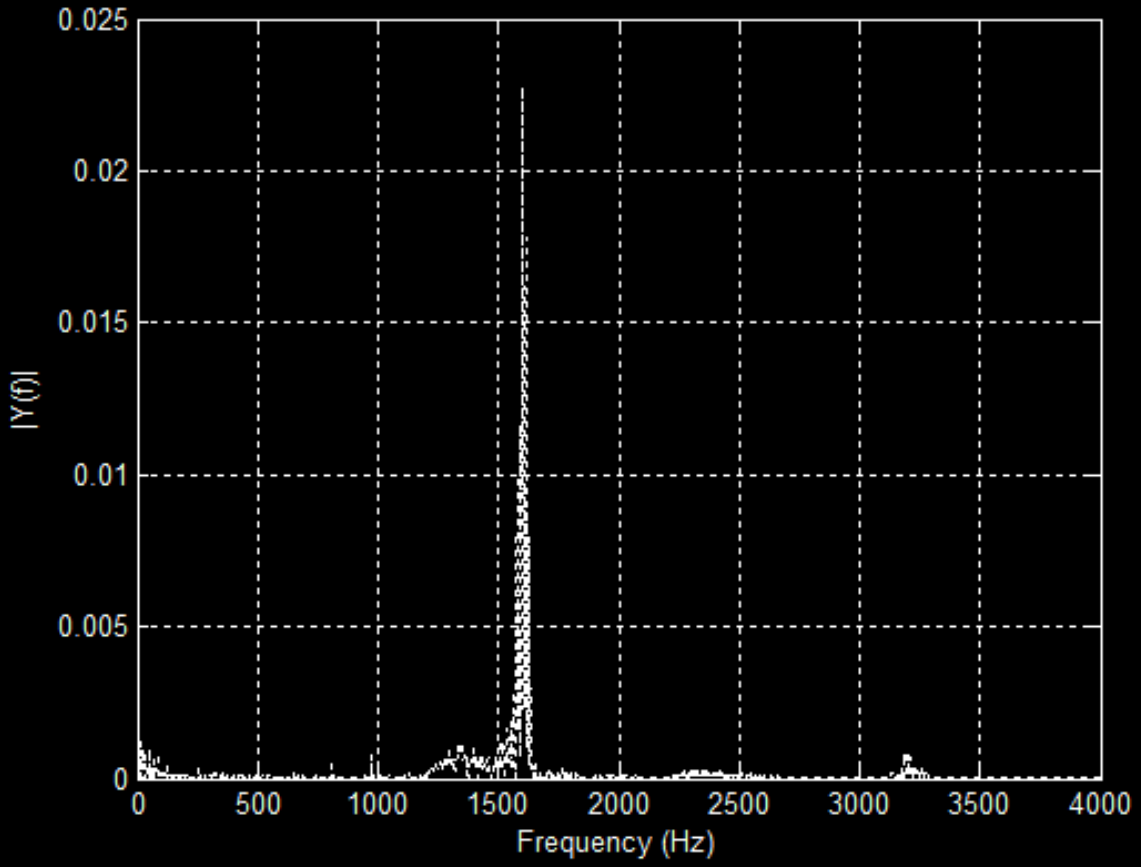
Gürültü Temizleme




```
% Isligimizi 3 saniye kayit edelim
% 8000Hz orneklemede, 16 bitlik 1 kanal
freq = 8000;
recObj = audiorecorder(freq, 16, 1);
disp('Islik Calin')
recordblocking(recObj, 3);
disp('Kayit Sona Erdi');
% Kaydi dinleyelim.
play(recObj);
% Kayiti sayisal degere cevirelim.
myRecording = getaudiodata(recObj);
% Kayiti gorelim.
figure, plot(myRecording);
%%
calculate_fft(myRecording, freq);
```

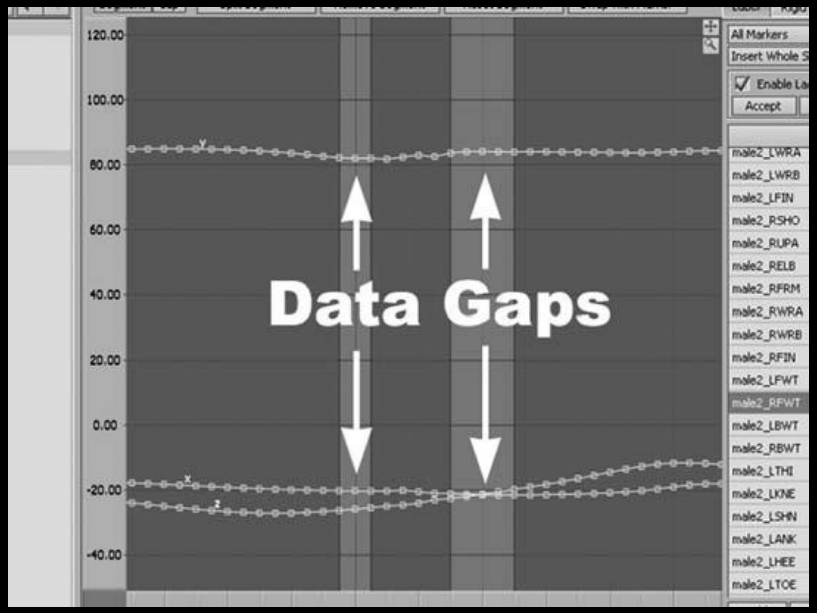
Hızlı Fourier Dönüşümü

Gürültü Temizleme



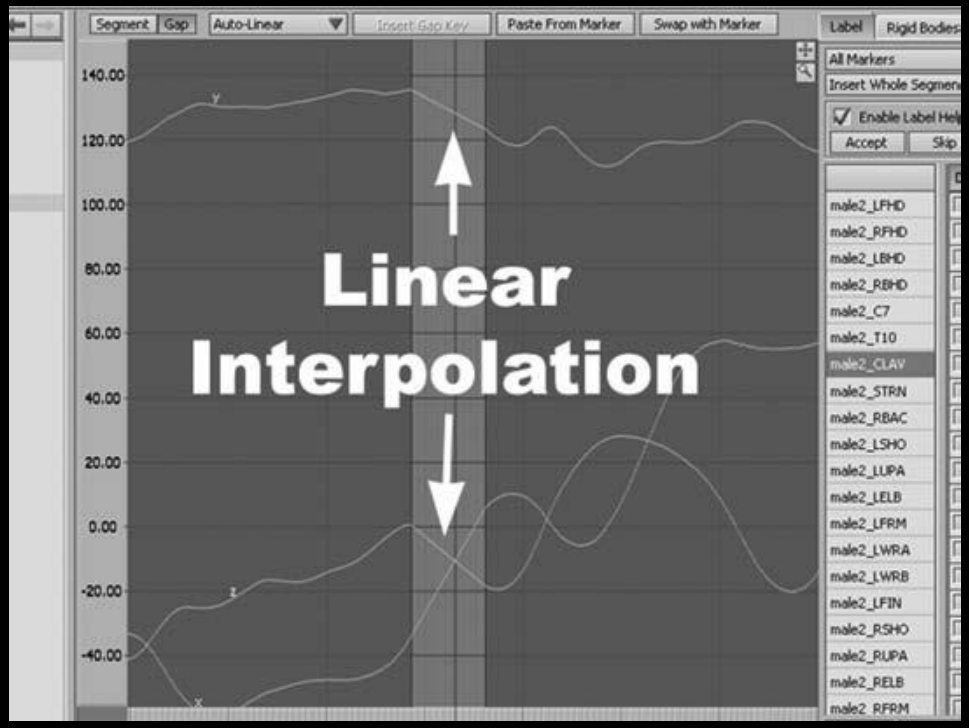
Kayıp Veri (Eliminating gaps)

The most common problem with optical mocap data is the gap caused by an absence of data or by removing an irregular peak that is a result of incorrect solution for computing marker locations. Gaps caused by the absence of data can occur for many different reasons, but occlusion is probably the most familiar one. One of the places where markers suffer from occlusion most frequently is the hands. A gap in the data looks like in the Figure below.



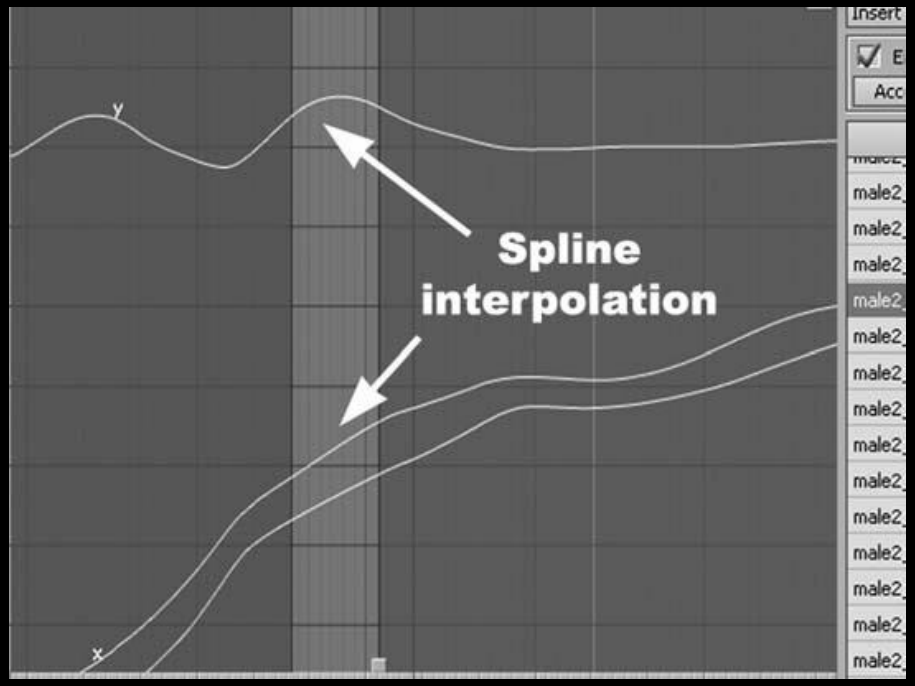
Kayıp Veri (Eliminating gaps)

The quickest and most direct way to fill the gap is using linear interpolation (Figure). Linear interpolation is connecting the last good data point before the gap and the first good data point after the gap with a straight line between them. There is no ease in or ease out but a straight line filling the gap. So, the resulting motion may look mechanical for that section.

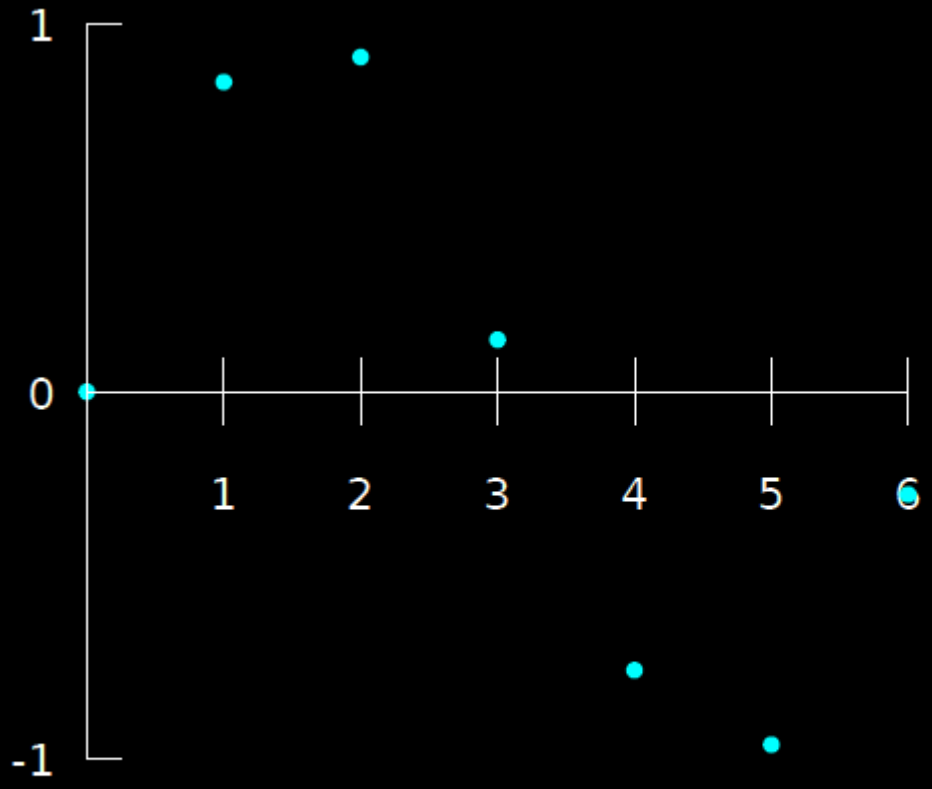


Kayıp Veri (Eliminating gaps)

Another way to fill the gap is using spline interpolation (Figure 4.3). The type of spline most commonly implemented for a graphical data editing tool is a cubic spline defined by the positions and tangents of two end points. Spline tools normally match the tangent of the curve's beginning point to the tangent of the last data point before the gap and match the tangent of the curve's end point to the tangent of the first data point after the gap.



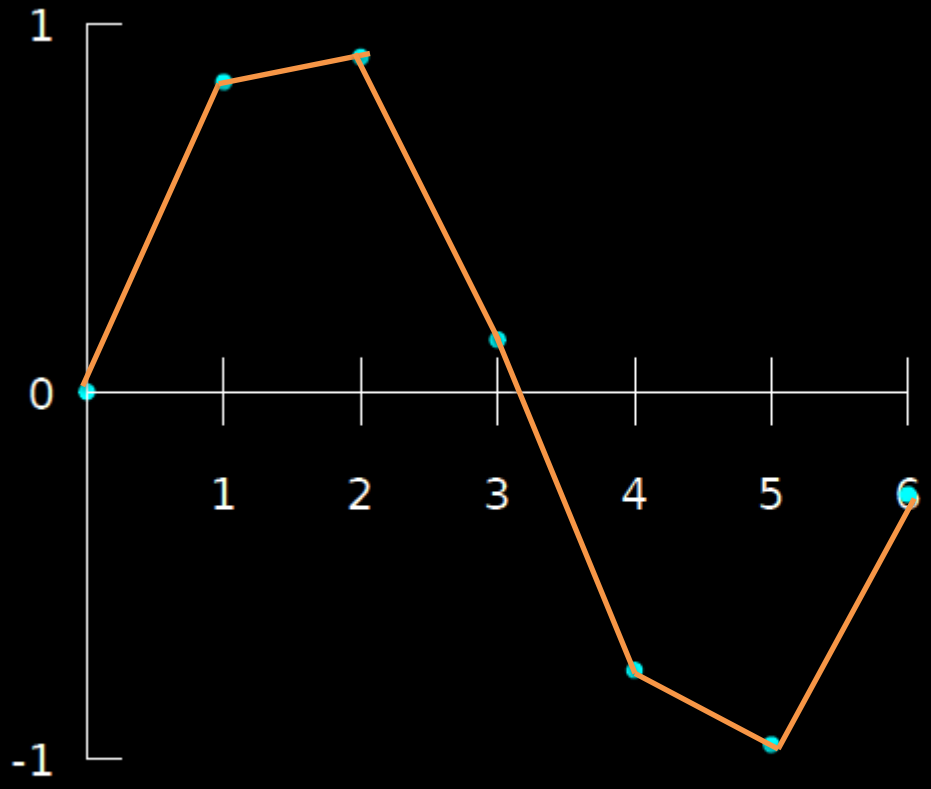
Kayıp Veri (Eliminating gaps)



- Interpolation is a way of filling in the gaps

Kayıp Veri (Eliminating gaps)

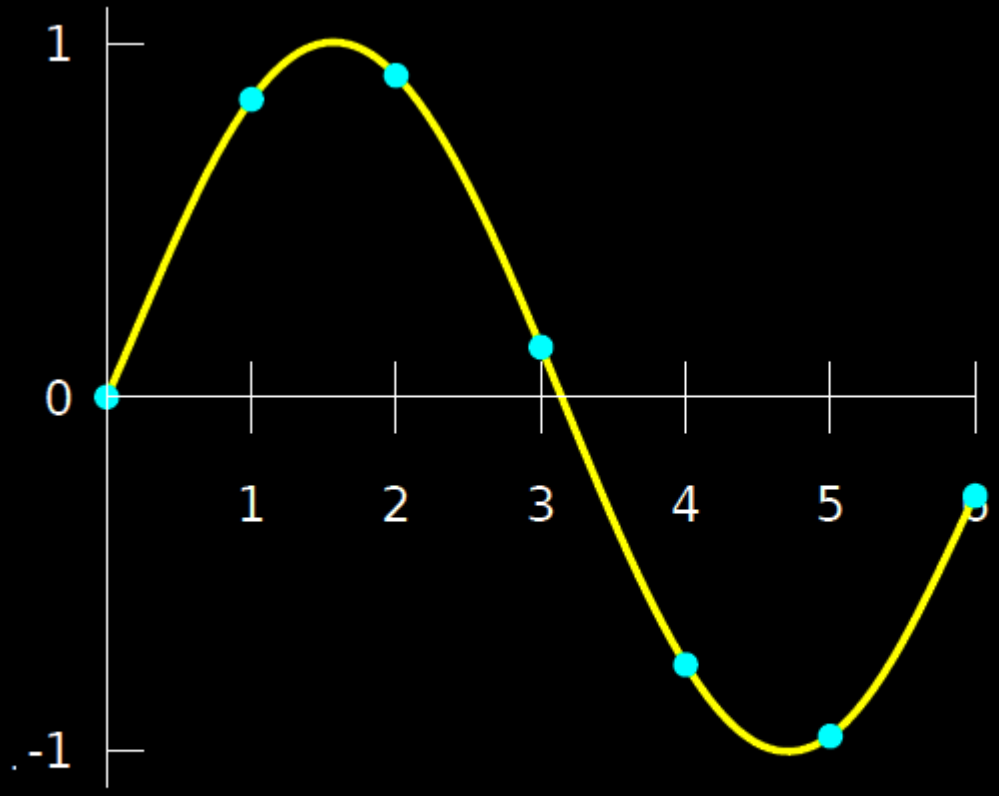
Linear interpolation: Basically draw a line between the two nearest points



- Interpolation is a way of filling in the gaps

Kayıp Veri (Eliminating gaps)

Polynomial interpolation



- Interpolation is a way of filling in the gaps

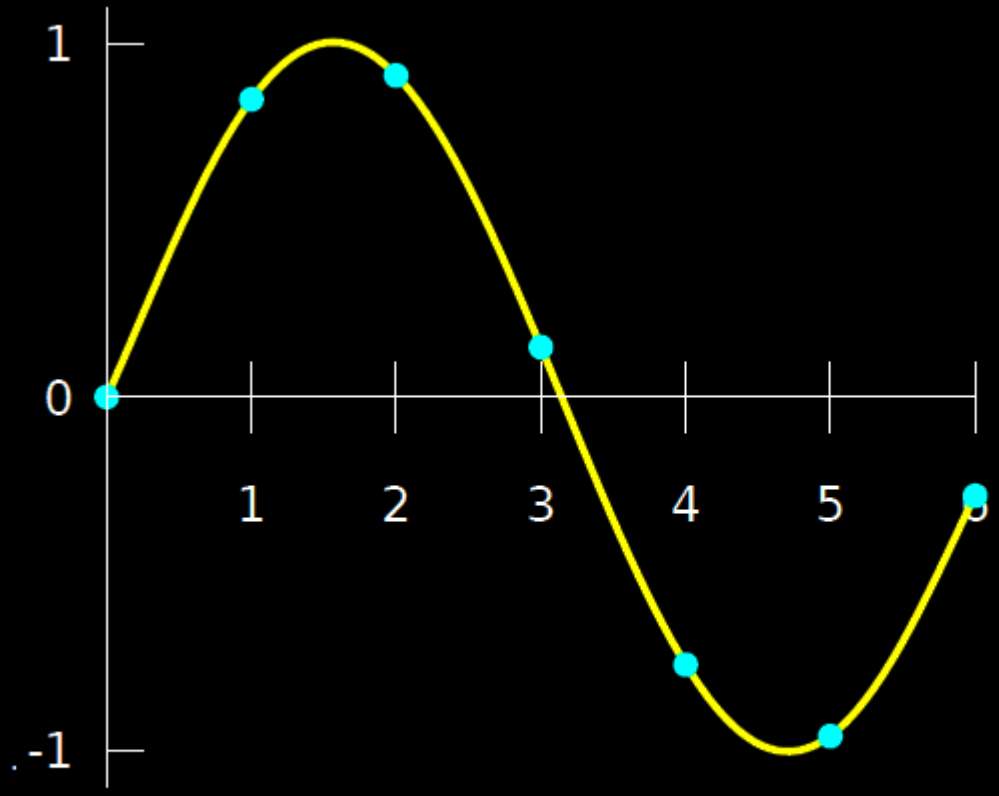
Kayıp Veri (Eliminating gaps)

Polynomial interpolation

- Draw a polynomial through all the data points
- Need a polynomial of degree one less than the number of points
- In general is problematic - noisy data can cause very strange results, so usually not used for movement data

Kayıp Veri (Eliminating gaps)

Spline interpolation



- Interpolation is a way of filling in the gaps

Kayıp Veri (Eliminating gaps)

Spline interpolation

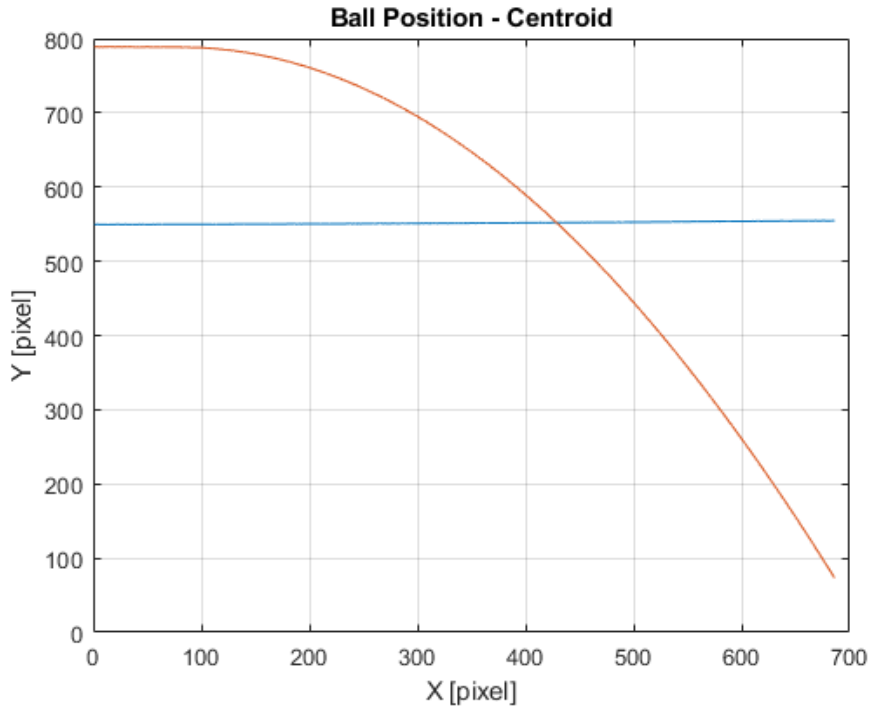
- Use low order polynomials (typically 3rd order - cubic) between each pair of points, and match the gradient at the data points (so it will be smooth)
- Missing samples can be estimated using spline interpolation
- Splines can be used also to smooth the data
- If less splines are used than there are data points, the splines will not go through every point.
- Can use the Matlab function `cpaps` from the spline / curve fitting toolbox or the function `splinefit` (free from the Matlab central file exchange)
- Can use this technique to interpolate and smooth data in one step

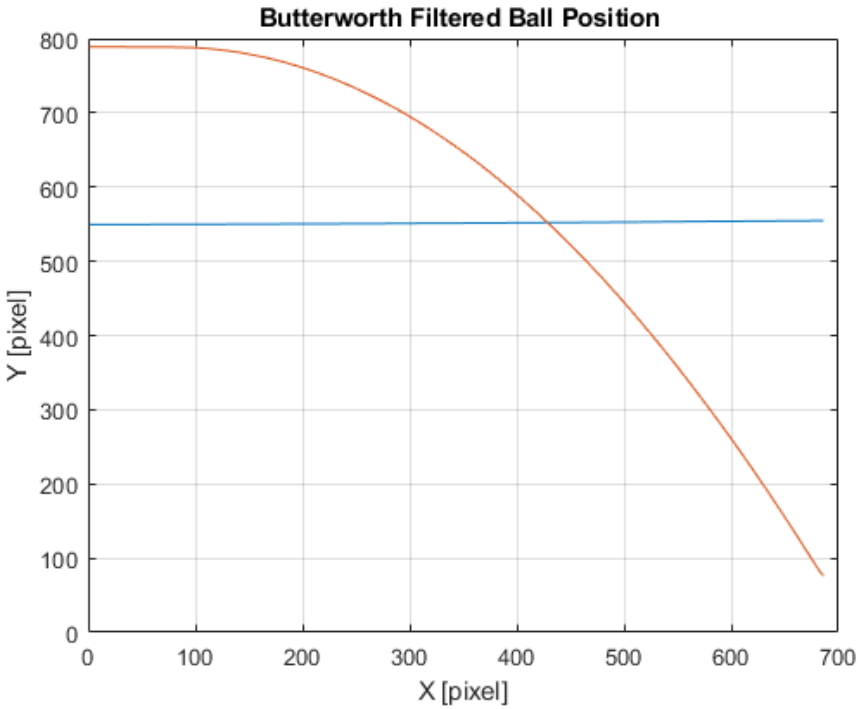
Serbest Düşme Deneyi

Veri toplama hızı : 1000Hz

Serbest Düşme Deneyi

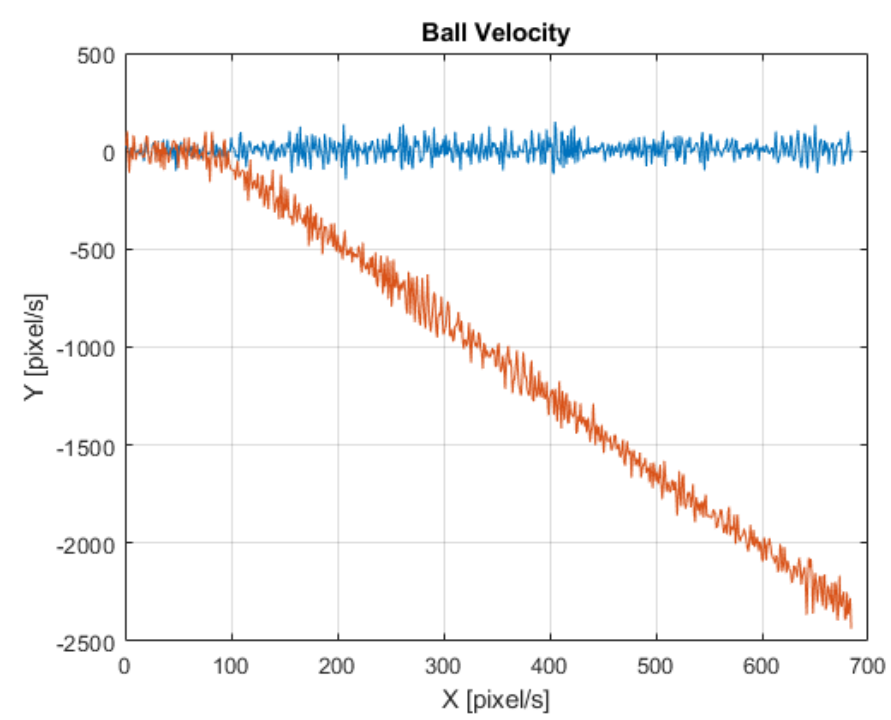
Konum Verisi

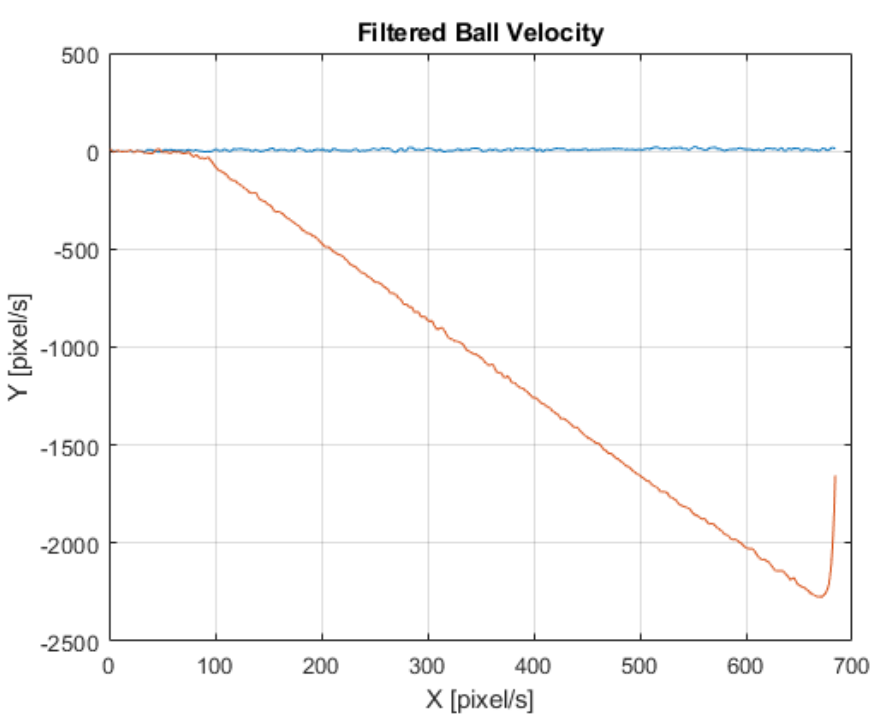




Serbest Düşme Deneyi

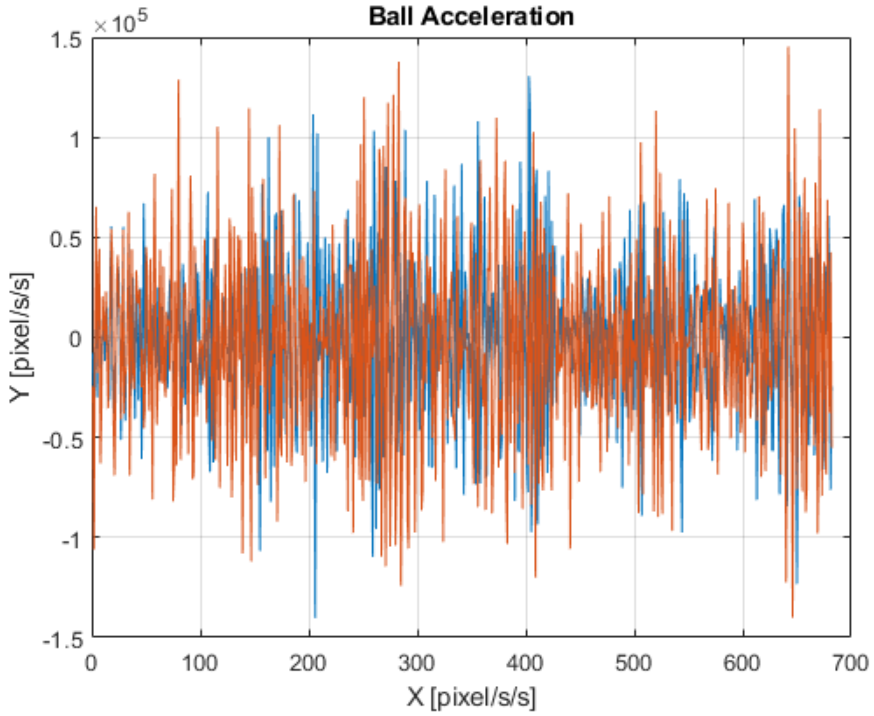
Hız Verisi

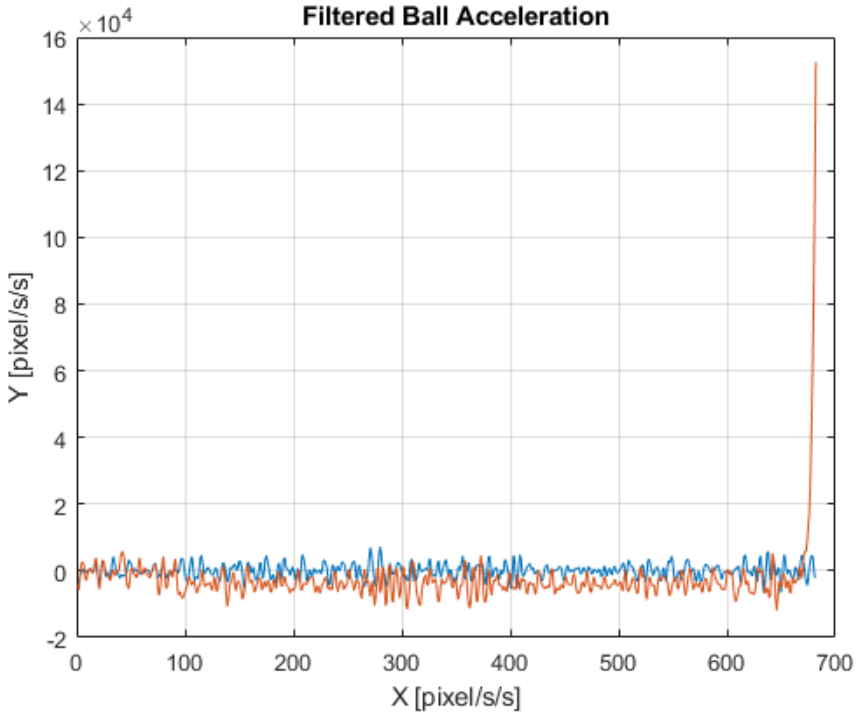




Serbest Düşme Deneyi

Veri toplama hızı : 1000Hz





1000 Hz veri toplama hızı ile elde edilmiş top bırak görüntülerindeki topun merkezinin hız ve ivmesini ham veri ve filtre uygulayarak hesaplayınız ve grafik olarak gösteriniz.

Teslim Tarihi : 29 Kasım 2017 Perşembe
Saat 10:00