

**TÜRKİYE CUMHURİYETİ**  
**YILDIZ TEKNİK ÜNİVERSİTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



**ZAMAN DÜZLEMİNDE ÖTELEME OLAN SİNYALLERİN**  
**HİZALANMASI**

16011043 – Recep Furkan KOÇYİĞİT

**BİLGİSAYAR PROJESİ**

Danışman  
Arş. Gör. Ahmet ELBİR

Kasım, 2019



## TEŞEKKÜR

---

Bu çalışmanın gerçekleştirilmesinde, değerli bilgilerini bizlerle payşalaşan danışman hocam; Arş. Gör. Ahmet Elbir 'e, çalışma süresince tüm zorlukları benimle göğüsleyen ve hayatımın her evresinde bana destek olan aileme teşekkürlerimi sunarım.

Recep Furkan KOÇYİĞİT

# İÇİNDEKİLER

---

<b>KISALTMA LİSTESİ</b>	<b>v</b>
<b>ŞEKİL LİSTESİ</b>	<b>vi</b>
<b>TABLO LİSTESİ</b>	<b>vii</b>
<b>ÖZET</b>	<b>viii</b>
<b>ABSTRACT</b>	<b>ix</b>
<b>1 Giriş</b>	<b>1</b>
1.1 Zaman Ötelemesi Probleminin Tanımlanması . . . . .	1
1.2 Proje Özeti . . . . .	1
<b>2 Ön İnceleme</b>	<b>3</b>
2.1 İlgili Çalışmalar . . . . .	3
<b>3 Fizibilite</b>	<b>4</b>
3.1 Teknik Fizibilite . . . . .	4
3.1.1 Yazılım Fizibilitesi . . . . .	4
3.1.2 Donanım Fizibilitesi . . . . .	4
3.2 İş Gücü ve Kısıtlamalar . . . . .	5
3.3 Yasal Fizibilite . . . . .	5
3.4 Ekonomik Fizibilite . . . . .	5
<b>4 Sistem Analizi</b>	<b>6</b>
4.1 Hedef . . . . .	6
4.2 Gereksinimler . . . . .	6
4.3 Durum Diyagramı . . . . .	6
<b>5 Sistem Tasarımı</b>	<b>8</b>
5.1 Yazılım Tasarımı . . . . .	8
5.1.1 Ön İşleme Yöntemleri . . . . .	8
5.1.2 Hizalama-Eğrilme Metodları . . . . .	11

5.2 Veritabanı Tasarımı . . . . .	16
5.3 Girdi-Çıktı Tasarımı . . . . .	16
<b>6 Uygulama</b>	<b>17</b>
<b>7 Deneysel Sonuçlar</b>	<b>24</b>
<b>8 Performans Analizi</b>	<b>26</b>
<b>9 Sonuç</b>	<b>27</b>
<b>Referanslar</b>	<b>28</b>
<b>Özgeçmiş</b>	<b>30</b>

## KISALTMA LİSTESİ

---

COW	Correlation Optimized Warping
DTW	Dynamic Time Warping
PTW	Parametric Time Warping
HIV	Human Immunodeficiency Virus
HCV	Hepatit C Virus
SHAPE	Selective 2 Hydroxyl Acylation analyzed by Primer Extension

## ŞEKİL LİSTESİ

---

Şekil 3.1	Gantt Diyagramı . . . . .	5
Şekil 4.1	Durum Diyagramı . . . . .	7
Şekil 6.1	PyTimeWarp Uygulaması Ana Penceresi . . . . .	17
Şekil 6.2	Yüklenen Sinyallerin Görselleştirilmesi . . . . .	18
Şekil 6.3	Sinyal Yumuşatma Penceresi . . . . .	18
Şekil 6.4	Yumuşatılmış Sinyallerin Görüntüsü . . . . .	19
Şekil 6.5	Tepe Değerleri Belirginleştirilmiş Sinyallerin Görüntüsü . . . . .	19
Şekil 6.6	Taban Hizaları Düzeltilmiş Sinyallerin Görüntüsü . . . . .	19
Şekil 6.7	DTW Penceresi . . . . .	20
Şekil 6.8	DTW Yönteminde Elde Edilen Eğrilme Yolu . . . . .	20
Şekil 6.9	DTW İle Hizalanmış Sinyaller . . . . .	21
Şekil 6.10	COW Penceresi . . . . .	21
Şekil 6.11	COW Yönteminde Elde Edilen Eğrilme Yolu . . . . .	21
Şekil 6.12	COW İle Hizalanmış Sinyaller . . . . .	22
Şekil 6.13	PTW Penceresi . . . . .	22
Şekil 6.14	PTW İle Hizalanmış Sinyaller . . . . .	23

## TABLO LİSTESİ

---

Tablo 3.1	Minimum donanım gereksinimleri . . . . .	4
Tablo 7.1	Ham Verinin Bazı Sayısal Değerleri . . . . .	24
Tablo 7.2	DTW Yöntemiyle Hizalama Bazı Sayısal Değerleri . . . . .	25
Tablo 7.3	COW Yöntemiyle Hizalama Bazı Sayısal Değerleri . . . . .	25
Tablo 7.4	PTW Yöntemiyle Hizalama Bazı Sayısal Değerleri . . . . .	25



# Zaman Düzleminde Öteleme Olan Sinyallerin Hizalanması

Recep Furkan KOÇYİĞİT

Bilgisayar Mühendisliği Bölümü  
Bilgisayar Projesi

Danışman: Arş. Gör. Ahmet ELBİR

Zaman düzleminde olan sinyallerin hizalanması, geniş uygulama alanı sebebiyle önemli sinyal işleme problemleri arasında yer almaktadır. Bu çalışmada ön işleme teknikleri kullanılarak zaman düzleminde olan sinyallerin gürültüsü azaltılmış, taban hızası düzeltilmiş ve tepe değeri belirginleştirilmiştir. Elde edilen yeni sinyaller dinamik zaman eğrilmesi, polinomal zaman eğrilmesi veya korelasyon eniyilemeli eğrilme yöntemleri kullanılarak hizalanmıştır ve elde edilen sonuçların performans karşılaştırılması yapılmıştır. Çalışma kapsamında GTZAN veri tabanı kullanılmış ve en yüksek başarı PTW yöntemiyle elde edilmiştir.

**Anahtar Kelimeler:** Dinamik Zaman Eğrilmesi, Korelasyon Eniyilemeli Eğrilme, Polinomal Zaman Eğrilmesi, Sinyallerin Hizalanması

# Alignment of Shifted Signals in Time Series

Recep Furkan KOÇYİĞİT

Department of Computer Engineering  
Computer Project

Advisor: Ress. Asst. Ahmet ELBİR

Alignment of signals in the time plane is one of the major signal processing problems due to its wide range of applications. In this study, by using preprocessing techniques, the noise of the signals in the time plane was reduced, the base alignment was corrected and the peak value was determined. The new signals obtained were aligned using dynamic time warp, polynomial time warp or correlation optimized warp methods and the results were compared. The GTZAN database was used in the study and the highest success was achieved with PTW method.

**Keywords:** Dynamic Time Warping, Correlation Optimized Warping, Parametric Time Warping, Alignment of Signals

Bu bölümde, ses ve biyokimyasal türdeki sinyallerde oluşan zaman ötelemesi problemi tanıtılacak ve genel kültür bilgileri verilecektir.

### 1.1 Zaman Ötelemesi Probleminin Tanımlanması

Milyonlarca insanın sağlığını tehdit eden HIV ve HCV gibi virüslerin yapısını öğrenmek ve bu virüslerin insan sağlığını olumsuz etkileyen işleyişini etkisiz hale getirecek ilaç üretimi yapmak ve ayrıca kalp sesinden teşhis koymak için önemli bilimsel çalışmalar yapılmaktadır. Bu çalışmalarda kullanılan virüs RNA sına ve kalpte oluşan ses sinyallerine ait verilerin analiz edilmesi gerekmektedir. Bu verilerin hızlı ve doğru bir şekilde incelenmesi için bazı tekniklerle sinyaller hizalanmalıdır. Bu proje sayesinde, bu konular üzerinde çalışan uzmanlar verileri daha iyi bir şekilde analiz etmesi mümkün olacaktır.

### 1.2 Proje Özeti

Bu çalışmada ses ve biyokimyasal türdeki sinyallerde oluşan zaman ötelemesi probleminin çözümünün hızlı ve etkili bir şekilde yapılması için farklı algoritmalar kullanılması ve bu algoritmaların performanslarının karşılaştırılması, bu algoritmalara farklı yeni özellikler eklenmesi sağlanacak ve tüm yöntemleri kolaylıkla kullanabilecek bir grafik arayüzlü yazılım geliştirilmesi hedeflenmektedir. Kullanıcıdan alınan ham veri Python dili kullanılarak çizdirilmekte ve veriyi işlemesi için yapılan teknikler kullanıcıya sunulmaktadır. İşletim sistemi olarak Windows 10 kullanılmaktadır.

Proje kitabında:

- Ön inceleme bölümünde, projenin yapılacağı alanda daha önce yapılmış olan çalışmalara ve benzer çalışmaların listelenmesini içerir.
- Fizibilite bölümü projenin yapılabilirliğini içerir.

- Teknik Fizibilite bölümü, projenin teknik risklerinin bilgisini içerir.
- İş Gücü ve Zaman Planlaması bölümü, projenin belirlenen süre içinde bitip bitmeyeceğinin bilgisini içerir.
- Yasal Fizibilite bölümü ,projenin yasalara uygun olup olmadığının bilgisini içerir.
- Ekonomik Fizibilite bölümü, projenin mali riskinin bilgisini içerir.
- Sistem Analizi ve Tasarımı bölümünde, projenin detayları,bilgi kaynakları ve gereksinimleri yer alır.
- Referanslar bölümünde, proje kapsamında yararlanılan kaynakların bilgisini içerir.

## 2 Ön İnceleme

---

Biyolojik virüsler ve kalp atışı sesleri, insanların hayatını tehdit eden ölüm gibi tedavisi mümkün olmayan zararlara yol açabilen etkilerinden dolayı bilimsel çalışmalarda büyük bir öneme sahip olan ve ülkeler tarafından çok fazla ekonomik kaynak ayrılan bir konu olmuştur.

### 2.1 İlgili Çalışmalar

Zaman serilerinin analizi ve hizalanması, biyolojik verilerden anlamlı bilgilerin çıkarılmasında önemli rol oynar. Virüslerin yapısını öğrenmek için yapılan çalışmalarda, kullanılan tekniklerden biri de SHAPE tekniğidir. Bu teknik ile virüslerin yapısı öğrenilerek virüslerin yapısına etki edilebilecek kimyasalların bulunması mümkündür. Bu çalışmada SHAPE verilerinin analizi için yeni algoritmalar tasarlanacaktır ve problemin çözümü için dinamik programlama tabanlı Dinamik Zaman Eğrilmesi(Dynamic Time Warping-DTW) yaklaşımı kullanılacaktır [1], [2], [3], [4], [5], [6], [7], [8], [9]. Bu yaklaşımlar dışında yaygın olarak kullanılan Korelasyon Eniyilemeli Eğrilmesi(Correlation Optimized Warping-COW) ve Parametrik Zaman Eğrilmesi(Parametric Time Warping-PTW) yaklaşımları da kullanılacaktır [9], [10], [11], [12]. DTW tabanlı yaklaşımlar zaman serilerinin sıralanması için yaygın olarak kullanılmaktadır. İlk zamanlarda ses sinyallerinin hizalanması için kullanılan klasik DTW algoritması farklı yaklaşımlarla modifiye edilerek bu problemin daha etkin bir şekilde çözülmesini sağlamaktadır.

### 3.1 Teknik Fizibilite

Teknik fizibilite, kendi içinde Yazılım Fizibilitesi ve Donanım Fizibilitesi şeklinde gruplanabilir.

#### 3.1.1 Yazılım Fizibilitesi

Bu proje için Python programlama dili görüntü işleme için gerekli ve yeterli özellikleri taşımaktadır. Python sahip olduğu çok sayıda kütüphaneye geliştiricinin daha kolay ve daha hızlı kod yazmasını sağlamaktadır. Uygulama geliştirme ortamı olarak pythonun kodlaması çok rahat ortam sağlayan Sublime Text Editor tercih edilmiştir. Bunun yanında hata ayıklayıcının bug'ları tespit ederken sağladığı kolaylık, kodu hızlıca düzenleme (code refactoring) konusunda sağladığı kolaylıklardan dolayı PyCharm da kullanılmıştır.

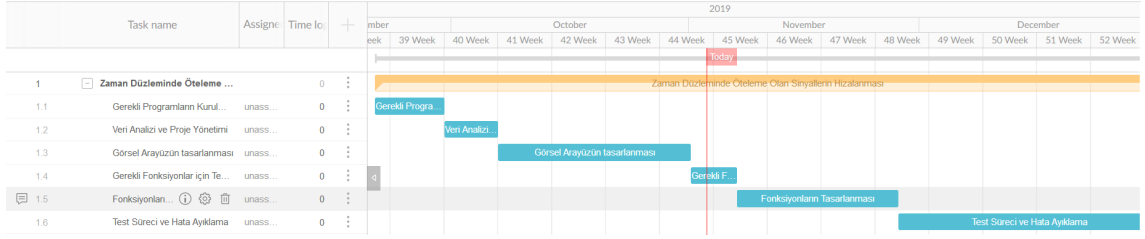
##### 3.1.1.1 İşletim Sistemi

Bu sistemin hedef kitlesinin Windows işletim sistemine daha yatkın olması sebebiyle bu sistem Windows 10 işletim sisteminde çalışacak şekilde geliştirilmiştir.

#### 3.1.2 Donanım Fizibilitesi

**Tablo 3.1** Minimum donanım gereksinimleri

Özellikler	Sistem Gereksinimleri
RAM	1GB
İşlemci Hızı	1.8 GHz
Disk boyutu	2GB



Şekil 3.1 Gantt Diyagramı

## 3.2 İş Gücü ve Kısıtlamalar

Bu projedeki görev dağılımını ve zaman planlamasını içeren Gantt diyagramı aşağıda verilmiştir:

## 3.3 Yasal Fizibilite

Bu projede kullanılan yazılımlar ve donanımların lisanslarıyla alakalı yasal olarak bir engel bulunmamaktadır, ayrıca bu projede gerçekleştirilecek program herhangi bir yasayı, patenti, fikri ve sınai hakları ihlal etmemektedir.

## 3.4 Ekonomik Fizibilite

Bu proje kapsamında yapılan çalışma AR-GE kapsamında ve öğrenciler tarafından gerçekleştirildiği için geliştiriciler herhangi bir ücret almamaktadır. Proje gerçekleştirilirken topluluk (community) sürümü ücretsiz yazılımlar kullanılmıştır. Projenin yapılabilmesi için herhangi bir konuda ücret gerekmemektedir.

## 4 Sistem Analizi

---

Sistem analizinin amacı projede en uygun çözümü bulmak için ana öğeler ve işlevlerin ortaya çıkarılıp tanımlanmasıdır. Ayrıca projenin hedefi belirlenir.

### 4.1 Hedef

Bu projede sinyallerin hizalanması aşamasında Dinamik Zaman Eğrilmesi-DTW, Korelasyon Eniyilemeli Eğrilme-COW ve Parametrik Zaman Eğrilme-PTW algoritmaları kullanılacak, bu yöntemlere ek olarak zaman gecikmesi cezası, tekrarlama gecikmesi cezası gibi bazı yeni modifikasyonlar eklenerek algoritmanın hizalama ve tepe değeri eşleştirme başarısının arttırılması sağlanacaktır. Bu sayede kullanıcıya daha hızlı çalışan ve daha doğru sonuçlar çıkmasını sağlayan bir program sunulmuş olacaktır.

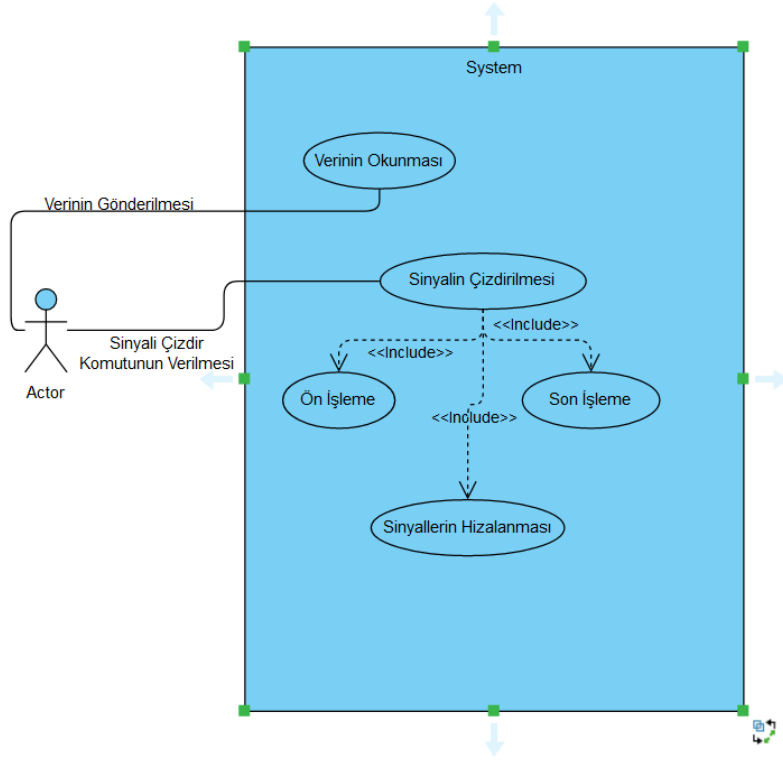
### 4.2 Gereksinimler

Hedefe ulaşabilmek için gereksinimlerin başında kullanışlı bir arayüz sunulmaktası, kullanıcıdan alınan veriyi işleyebilmesi, işlediği sinyali çizdirebilmesi ve kullanıcının isteğine göre değişik hizalama tekniklerinden birini kullanabilmesi işlenen sinyal hakkında bilgiyi verebilmesi gerekir. Bunlar için geliştiricinin Python dilinde “GUI” ve “Ploting Signals” konularını bilmesi, "PyQt5" ve "Figure Canvas" kütüphanelerini kullanabilmesi gerekmektedir.

### 4.3 Durum Diyagramı

Bu sistemin kullanım durum diyagramı aşağıda verilmiştir.





Şekil 4.1 Durum Diyagramı

# 5

## Sistem Tasarımı

---

### 5.1 Yazılım Tasarımı

Projede gerçekleştirilmek istenen sistemdeki tasarım aşamaları, bu bölümde anlatılacaktır. Öncelikle ön tasarım sorularının cevaplandırılması tasarım aşamasındaki işlemleri hızlandıracağından aşağıdaki 3 sorunun cevabının verilmesi gerekmektedir.

- 1-Sistem sıfırdan mı oluşturulacak? Olan sistem üzerinde güncelleme mi yapılacak?
- 2-Satın alındıktan sonra firmanın ihtiyacına göre özelleştirilerek mi kullanılacak?
- 3-Dışarıdan hizmet olarak temin edilebilir mi?

Yapılacak olan sistem şu anda bulunan sistemlere bir alternatif olacağından ve sistemde kullanılacak sinyalleri hizalama, yumuşatma, taban düzeltme, tepe belirginleştirme algoritmaları kodlanacağından sistemin sıfırdan oluşturulacağı söylenebilir.

Gerçekleştirilecek sistem akademik bir çalışma sonucu için kullanılacağından dolayı herhangi bir ücret talep edilmeyerek yazılacak olan kodlar github.com üzerinden açık kaynaklı olarak paylaşılacaktır.

Akademik olarak bir sistem kurulması istendiğinden dışarıdan herhangi bir hizmet temin edilerek kurulabilecek bir sistem değildir.

Sistem akademik amaçla yapılacağından dolayı herhangi bir şifreleme yapısı gerektirmemekte ve virüs kontrolüne ihtiyaç olmamaktadır.

#### 5.1.1 Ön İşleme Yöntemleri

##### 5.1.1.1 Yumuşatma Yöntemleri

Yumuşatma(smoothing) sayısal bir veri dizisindeki gürültünün ya da aykırı değerlerin en aza indirgenmesini sağlayan bir yöntemdir. Bu yöntem sayesinde ham veride

bulunan hatalı tepe değeri gibi bazı aykırı değerler ortadan kaldırılmıştır. Literatürde çok fazla yumuşatma tekniği bulunmasına rağmen yapılan uygulamada üç farklı teknik gerçekleştirilmiştir. Kullanıcı bu üç yöntemden istediğini seçerek farklı yumuşatma işlemleri yapabilmektedir.

**5.1.1.1.1 Aritmetik Ortalama İle Yumuşatma** Bu yumuşatma tekniği matematikte sıklıkla kullanılan  $n$  adet sayısal toplamın  $n$  sayısına bölünmesi şeklinde uygulanır. Bu bölgede aritmetik ortalama sonucunda elde edilen yeni değerler o bölgedeki aykırı değerlerin azaltılmasını sağlar. Pencere boyutu  $w=3$  olan  $N$  elemanlı bir veri setinde yumuşatma işlemi aşağıdaki gibi şekilde gerçekleştirilir.

$$XS[i] = \frac{1}{w}(X[i-1] + X[i] + X[i+1]) \quad (5.1)$$

**5.1.1.1.2 Ağırlıklı Aritmetik Ortalama İle Yumuşatma** Özellikle tepe işaretlerinin olduğu sinyallerde tepe noktalarının daha iyi tespit edilmesi açısından ağırlıklı aritmetik ortalama tercih edilmektedir. Pencere boyutu  $w=3$  olan  $N$  elemanlı bir veri setinde yumuşatma işlemi aşağıdaki gibi şekilde gerçekleştirilir.

$$XS[i] = \frac{1}{(k1 + k2 + k3)}(k1 * X[i-1] + k2 * X[i] + k3 * X[i+1]) \quad (5.2)$$

**5.1.1.1.3 Gauss Yoğunluk Fonksiyonu ile Yumuşatma** Bu yumuşatma tekniğinde Eşitlik 5.2 de kullanılan  $k1$ ,  $k2$  ve  $k3$  gibi belirli katsayılarla sabit olarak vermek yerine bu katsayıların olasılıksal dağılım kullanılarak tespit edilmesi ve yumuşatma işleminin bu katsayılarla yapılması amaçlanmaktadır. Bu sayede daha başarılı bir yumuşatma meydana gelmekte ve orjinal veride bulunan bozucu etkiler giderilebilmektedir. Verilen pencere boyutu kadar elemanın tek boyutlu gauss olasılık fonksiyonu kullanılarak katsayıların bulunması için Eşitlik 5.3 kullanılmaktadır.

$$f(x) = \frac{1}{(\sqrt{2\pi\sigma^2})} \times \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\} \quad (5.3)$$

#### 5.1.1.2 Tepe Değeri Belirginleştirme

Tepe noktası belirginleştirme ya da keskinleştirme (peak shraping) olarak da ifade edilen bu ön işleme adımı sayesinde ham verideki tepe noktalarının belirginleştirilmesi sağlanabilir. Sinyal yumuşatma işlemini yaptıktan sonra orjinal sinyaldeki bazı tepe noktalarında bozulmalar oluşabilir ve bu sebeple tepe değerleri kaybedilebilir.

Meydana gelen bu tür bozulmaları engellemek için tepe değeri belirginleştirme teknikleri kullanılabilir. Genel olarak ikinci ve dördüncü mertebeden türev işlemleri kullanılarak yapılır. Öncelikle ham verinin ikinci veya dördüncü mertebeden türevi alınır ve yeni bir zaman serisi elde edilir. Türetilen bu sinyal bir  $k$  sabiti ile çarpıldıktan sonra ham veriden çıkartılarak tepe noktaları daha belirgin hale getirilmiş yeni bir sinyal elde edilir. Tepe değeri belirginleştirme işlemi Eşitlik 5.4 de belirtilmiştir [2], [3].

$$Y_i = X_i - k.R_i // \quad (5.4)$$

#### 5.1.1.3 Taban Hizası Düzeltme

Zaman düzleminde olan sinyallerin taban hizasında değişimler yaşanabilmektedir. Sinyal hizalamanın başarılı bir şekilde yapılması için hizalanacak sinyallerin aynı taban seviyesine sahip olması gerekmektedir. Bununla birlikte taban hizasının değişmesi eldeki ham verinin tepe noktalarının yükseklik değerinin doğru bir şekilde saptanmasını engelleyen bir durumdur. Taban hizası seviyesinin ayarlanarak ham verinin tepe noktalarının daha belirgin hale getirilmesi sağlanır ve bu tepe noktalarının seviyesinin doğru bir şekilde tespit edilmesi kolaylaştırılır. Taban hizası düzeltme işlemini yapmak için doğrusal interpolasyon işlemi kullanılmıştır [2], [3]. Taban hizası düzeltme işlemini yapabilmek için ham verinin belirli uzunlukta parçalara bölünüp her bir parçadaki en küçük değer bulunmalıdır. Bulunan bu değerlerden doğrusal interpolasyon işlemi yapılarak yeni bir dizi meydana getirilir. Doğrusal interpolasyon işlemi ile yeni dizi elde etmek için Eşitlik 5.5 kullanılır.

$$\frac{Y - Y_1}{X - X_1} = \frac{Y_2 - Y_1}{X_2 - X_1} \quad (5.5)$$

#### 5.1.1.4 Sinyal Genişletme ve Sıkıştırma

DTW ve COW yönteminde eğrilme yolu bulunduktan sonra yeni zaman serilerinin oluşturulması gerekmektedir. Her iki yöntemde de bulunan eğrilme fonksiyonu DTW yönteminde hangi noktanın hangi noktayla eşleşeceğini ifade ederken , COW yönteminde eşleşecek parçaların sınır değerlerini belirtir. Bu aşamadan sonra belirlenen iki nokta arasında interpolasyon yöntemi kullanılarak örnek sinyal yeniden oluşturularak referansla aynı zaman düzlemine getirilir. Bu sayede yeni sinyalleri çizdirme aşamasında sinyaller hizalanmış bir şekilde görünür. Bu işlem kübik şerit(spline) interpolasyon yöntemi ile yapılmıştır. Bu yöntemler sinyali oluşturmak için verinin her bir alt parçasına özel ayrı bir eğri uydurulur ve orjinal sinyal sıkıştırılarak ya da genişletilerek hizalanmış yeni forma getirilir. Verinin her bir noktası

ya da alt segmenti için ayrı bir eğri uydurulması sonucu bilgi kaybı en aza indirgenir. Orjinal verideki salınımlar kaybolmayacağı için sinyalin şeklinde büyük bir değişiklik meydana getirmez [1].

### 5.1.2 Hizalama-Eğrilme Metodları

Eğrilme yöntemleri kullanıcı tarafından yüklenen ham veriden bir tanesinin referans olarak seçilip diğer verilerin bu referans veriye göre hizalanmasını sağlamaktadır. Literatürde dinamik programlama tabanlı, interpolasyon tabanlı, genetik algoritma ve bulanık mantık tabanlı çok fazla hizalama metodu bulunmaktadır. Bu çalışma boyunca dinamik programlama temelli dinamik zaman eğrilmesi, korelasyon iyileştirmeli eğrilme ve parametrik zaman eğrilmesi yöntemleri üzerinde durulmuştur [3],[9],[10],[11],[12].

#### 5.1.2.1 Dinamik Zaman Eğrilmesi

Dinamik zaman eğrilmesi(Dynamic Time Warping-DTW) algoritması iki zaman serisi arasındaki mümkün olan en yüksek değerde eşleşmeyi bulmayı amaçlamaktadır. İki zaman dizisinde, uzaklığı az olduğu bölgeleri hizalayarak ve bu dizide minimum uzaklık elde edilerek doğrusal olmayan bir şekilde eğrilir. DTW iki dizi arasında mümkün olan en fazla benzerlik bulunana kadar zaman eksenleri eğrilerek gerçekleştirilir. Ayrıca bu yöntem zaman ya da hizalama başarısını arttırmak için birçok modifikasyon uygulanabilmektedir. Bunlar; zaman ve eşleşme cezaları, sinyalin birinci dereceden türevinin alınması, farklı uzaklık fonksiyonları olarak verilebilir [2], [3], [4], [5], [9],

Dinamik zaman eğrilmesi algoritmasıyla iki sinyalin hizalanması için elde edilen iki adet sinyalin  $X = (x_1, x_2, \dots, x_N)$ ,  $Y = (y_1, y_2, \dots, y_M)$  ve  $N=M$  olduğu kabul edilir. Öncelikle bu sinyalden  $N*M$  boyutunda uzaklık(distance) veya maliyet(cost) matrisine bu iki serideki her bir elemanın birbirleri ile olan uzaklık değeri bir uzaklık bulma yöntemi ile yerleştirilir. Bu matrisin her bir i. satır ve j. sütun elemanı X dizisinin i. elemanı ile Y dizisinin j. elemanı arasındaki uzaklığın sayısal değerini oluşturur. İki nokta arasındaki uzaklığın sayısal değeri, hizalama işleminin başarısını etkileyeceği için farklı uzaklık fonksiyonları ile bu değer hesaplanması yapılmıştır. Eşitlik 5.6 da matrisin her bir elemanın sayısal değerinin bulunması işlemi gösterilmiştir [2], [3], [5], [8], [9].

$$C(i, j) = \text{Uzaklık}(X[i], Y[j]) \quad \text{her, } i \in [1 \dots N], \quad j \in [1 \dots M] \quad (5.6)$$

Optimum eşleşme noktalarının tespit edilmesi için Eşitlik 5.6 kullanılarak elde edilen maliyet matrisi kullanılarak birikmiş maliyet hesaplanması yapılmalıdır. Birikmiş maliyet matrisi de Eşitlik 5.6 da elde edilen maliyet matrisi ile aynı boyutta olmalıdır. Birikmiş maliyet matrisinin 1.satır ve 1. sütunlarının hesaplanması işlemleri Eşitlik 5.7 ve 5.8 de belirtilmiştir.

$$D(i, j) = \sum_{k=1}^i C(k, 1) \quad \text{her, } i \in [1 \dots N], \quad j = 1 \quad (5.7)$$

$$D(i, j) = \sum_{k=1}^j C(1, k) \quad \text{her, } j \in [1 \dots M], \quad i = 1 \quad (5.8)$$

Eşitlik 5.7 ve 5.8 de birikmiş maliyet matrisinin ilk satır ve sütun değerleri hesaplandıktan sonra matrisin her bir elemanı Eşitlik 5.9 ile hesaplanır [1], [2], [8].

$$D(i, j) = C(i, j) + \min \{D(i-1, j-1), D(i-1, j), D(i, j-1)\} \quad (5.9)$$

her,  $i \in [2 \dots N], \quad j \in [2 \dots M]$

Optimum eğrilme yolu verilen kısıtlamalar göz önünde bulundurularak minimum maliyet olacak şekilde hesaplanır. Eğrilme yolu aşağıdaki verilen ifadeler kullanarak  $WP=(N,M)$  de  $WP(1,1)$  e doğru geri izleme ile bulunur. Bu yöntemler eğrilme yolu yada eğrilme fonksiyonu olarak tanımlanan bu işlem Eşitlik 5.10 daki gibi yapılır.

$$WP = \left\{ \begin{array}{ll} (1, j-1), & i = 1 \text{ için} \\ (i-1, j), & j = 1 \text{ için} \\ \text{argmin} \{D(i-1, j-1), D(i-1, j), D(i, j-1)\}, & \text{diğer} \end{array} \right\} \quad (5.10)$$

**5.1.2.1.1 DTW Modifikasyonları** DTW yöntemine farklı modifikasyonlar yapılarak hesaplama hızı veya hizalama performansı açısından iyileştirmeler yapılabilmektedir. Bu çalışmada hizalama başarımını etkileyen modifikasyonlar özetlenmiş ve geliştirilen yazılım, ilave edilen her bir özelliği kullanıcının interaktif olarak seçip uygulayabileceği şekilde tasarlanmıştır. Bu sayede hizalama performansı en yüksek olan yöntem kullanılarak birbirine en çok benzeyen iki sinyalin bulunması gerçekleştirilebilecektir.

## Farklı Uzaklık Fonksiyonları Kullanarak Maliyet Hesaplama

- **Öklid Uzaklık Fonksiyonu:** L2 norm uzaklığı olarak da bilinen ve en çok kullanılan uzaklık bulma yöntemidir.

$$d_{x,y} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_N - y_N)^2} \quad (5.11)$$

- **Manhattan Uzaklık Fonksiyonu:**Öklid uzaklığı ile birlikte çok sık kullanılan bir başka uzaklık fonksiyonudur. L1 norm uzaklığı olarak da bilinir.

$$d_{x,y} = |x_1 - y_1| + |x_2 - y_2| + \dots + |x_N - y_N| \quad (5.12)$$

- **Chessboard Uzaklık Fonksiyonu:** N boyutlu iki nokta arasındaki uzaklıkların mutlak değerlerinin en büyük olanını verne uzaklık fonksiyonudur.

$$d_{x,y} = \max\{|x_1 - y_1| + |x_2 - y_2| + \dots + |x_N - y_N|\} \quad (5.13)$$

- **Bray-Curtis Uzaklık Fonksiyonu:**Manhattan uzaklık fonksiyonun modifiye edilmesi ile elde edilmiş bir fonksiyondur.

$$d_{x,y} = \frac{|x_1 - y_1| + |x_2 - y_2| + \dots + |x_N - y_N|}{|x_1 + y_1| + |x_2 + y_2| + \dots + |x_N + y_N|} \quad (5.14)$$

**Türev Tabanlı Dinamik Zaman Eğrilmesi** Klasik zaman eğrilmesi yöntemindeki bazı aksaklıklardan dolayı sinyalin birinci mertebeden türevi alınarak oluşan yeni sinyalin hizalanması işlemi hizalama başarısını etkilemektedir. Hizalanacak sinyal istenirse öncelikle türevi alınır ve yukarıda anlatılan farklı uzaklık fonksiyonları da dahil edilerek maliyet matrisi oluşturulur. Birikimli maliyet matrisi ve eğrilme yolu hesaplanması daha önce Eşitlik 5.14 anlatılan şeklide bulunarak hizalama işlemi yapılmaktadır. Bu işlemde N elemanlı X zaman serisinin birinci mertebeden türevinin alınması  $f'(X_i)$  Eşitlik 5.15 deki gibi gerçekleştirilmektedir[2], [4].

$$f'(X_i) = \frac{(X_i - X_{i-1}) + (X_{i+1} - X_{i-1})/2}{2} \quad \text{her } i \in [2 \dots N - 1] \quad (5.15)$$

**Zaman Cezası** Hizalanması istenen iki sinyalin maliyet matrisinin bulunması aşamasında uygulanan ve hizalama işleminin doğru bir şekilde yapılmasını sağlayan diğer bir kısıta zaman cezası(time penalty) uygulamasıdır. Hizalanacak iki sinyalin eşleşme indisleri arasındaki fark ne kadar az ise sinyaller da daha çok birbirine benzeyecek şekilde hizalanır. Örneğin, bir sinyalin 5. noktasınının diğer sinyalin 5. noktası ile hizalanması ideal bir hizalama işlemidir. Fakat 5.nokta ile 50. nokta arasında eşleşme yapılması sinyalin doğru eşleşmemesine neden olabilmektedir. Bu

bozucu etkiyi engelleyebilmek için zaman gecikmesi katsayısı uygulanmalıdır. İki sinyalin i. ve j. noktaları arasında hesaplanan C maliyet değeri için zaman gecikmesi Eşitlik 5.16 daki gibi yapılır.

$$TP(i, j) = \exp\{|i - j| \times T\} \quad (5.16)$$

### 5.1.2.2 Korelasyon Eniyilemeli Eğrilme

Sinyal hizalama işleminin yapılması için gerçekleştirilen yöntemlerden ikincisi korelasyon eniyilemeli eğrilme(Correlation Optimized Warping-COW) yöntemidir. Nielsen ve arkadaşları tarafından kromatogram verilerinin hizalanması için geliştirilen bu yöntem dinamik zaman eğrilmesine(DTW) teknik olarak çok benzemektedir [9],[10]. Temel olarak DTW yönteminde minimum uzaklığın bulunması prensibine karşı COW yönteminde maksimum benzerlik ya da korelasyon değeri kullanılmaktadır. Fakat bu benzerlik değeri iki nokta arasında değil sinyalin belirli uzunluktaki alt parçaları arasında hesaplanmaktadır [9],[10]. COW yöntemi temel olarak iki parametre ile gerçekleştirilmiştir. Bunlarda birincisi, eldeki ham sinyalin küçük parçalara bölünmesi aşamasında kullanılan parça uzunluğudur(m). Bu sayede her iki sinyal n adet m uzunluklu parçaya bölünmekte ve hizalama aşamasında bu parçalar arasındaki en fazla benzerliğin olduğu noktaların sınır değerleri tespit edilerek eğrilme gerçekleştirilmektedir. Bir diğer parametre olan slack(s) parametresi ise karşılıklı olarak bölünen segmentlerin en fazla benzerliğinin bulunması aşamasında sınır değerler arasındaki değişimi göstermektedir. Sinyalin bütün segmentleri arasında en yüksek değerlikli benzerliğin bulunduğu noktalar arasında alt parçalar yeniden oluşturularak hangi parçanın hangi parçayla eşleceği tespit edilmektedir. Nielsen ve arkadaşları tarafından önerilen bu yöntemde benzerlik değeri olarak Pearson Korelasyon katsayısı kullanılmıştır. X ve Y zaman serisi arasında bulunan korelasyon bulunması Eşitlik 5.17 de verilmiştir [9],[10].

$$r = \frac{cov(X, Y)}{\sigma_X \sigma_Y} \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \quad (5.17)$$

Eşitlik 5.17 de elde edilen r katsayısı -1 ile +1 arasındaki pozitif yönlü olduğunu belirtir. Eğer r=0 olması durumunda X ve Y arasında hiçbir benzerliği olmadığı anlaşılır. COW yönteminde istenen ise eldeki bu parçalar arasındaki r değerinin mümkün oldukça +1 e yaklaşması anlamındadır. Bu değerin maksimize edilmesi için sinyalin alt parçaları 1 den başlamak üzere slack parametresi değerine kadar değiştirilir ve yeni parçalar elde edilir. Bu alt parçalardan korelasyon değerinin en yüksek olanı sinyalin doğru bir şekilde hizalanacağını varsayar. Sinyalin her



bir alt parçasının korelasyon değerinin hesaplanması işlemi sürekli olarak yapılarak en yüksek benzerlik bulunan parçaların sınırları arasında bir eğrilme fonksiyonu hesaplanır. Bu fonksiyon çok sayıda korelasyon hesaplaması aşamasından meydana geldiği için maliyetli bir şekilde meydana gelmektedir. Böylece iki zaman serisinin hizalandıktan sonra birbirine benzemesi mümkün olabilmekte ve zaman kayması problemi giderilmektedir [9],[10].

### 5.1.2.3 Parametrik Zaman Eğrilmesi

Zaman ekseninde meydana gelen bozulmayı ortadan kaldırmak için sık kullanılan yöntemlerden bir tanesi de parametrik zaman eğrilmesi yöntemidir. Dinamik programlama yöntemi kullanan diğer yöntemlere göre hızlı çalışabilen ve daha az miktarda bellek tahsis edilen bu yöntem kromatogram verilerini hizalamak için Eiler ve arkadaşları tarafından geliştirilmiştir[9],[12]. Temel olarak iki sinyalin hizalanması için ikinci dereceden bir fonksiyon kullanarak interpolasyon işlemi yapılır ve sinyalin hizalanması gerçekleştirilir.

Zaman ekseninde bozulma olan iki sinyal  $X_i = X(t_i)$  ve  $Y_i = Y(t_i)$  gösterimiyle zamanla değişen olarak tanımlanmaktadır. Bu iki sinyalin hizalanmasını sağlayacak olan eğrilme fonksiyonu ise  $W(t_i)$  olarak belirlenir [9],[12]. Parametrik zaman eğrilmesi yönteminde referans olarak belirlenen X sinyali eğrilme fonksiyonu yardımıyla Y sinyaline  $X(W(t_i))$  işlemi yapılarak hizalanır. Kullanılan  $W(t_i)$  eğrilme fonksiyonu Eşitlik 5.18 de verilmiştir [9],[12].

$$W(t_i) = \sum_{k=0}^K a_k t_i^k = a_0 + a_1 t_i + a_2 t_i^2, \quad K = 2 \quad (5.18)$$

Eşitlik 5.18 de bulunan  $K=2$  alınarak, ikinci dereceden bir eğrilme fonksiyonu meydana getirilir. Bu fonksiyon ile yapılan hizalama işleminde kullanılan  $a_0, a_1$  ve  $a_2$  parametreleri başlangıç değeri olarak  $a_0 = 0, a_1 = 1$  ve  $a_2 = 0$  verilir. Algoritma içerisindeki iterasyonlarda iki sinyalin her bir noktasındaki optimum değer sürekli güncellenmesi gerekmektedir ve bu işlemi gerçekleştirebilmek için belirtilen katsayılara göre birinci dereceden türevi alınarak sıfıra eşitlenmelidir. İki sinyalin birbirine hizalanması için kullanılan bu yöntemde meydana gelen hatanın en aza indirgenmesi için Eşitlik 5.19 da bulunan fonksiyon kullanılır [9],[12].

$$S = \sum [Y_i - X(W(t_i))]^2 \quad (5.19)$$

## **5.2 Veritabanı Tasarımı**

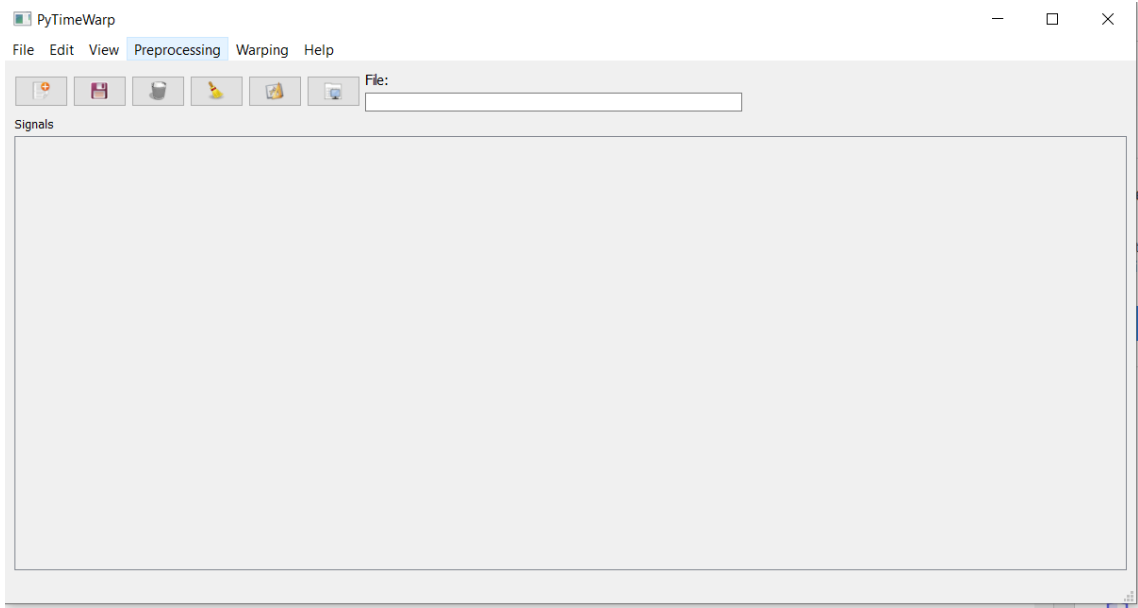
Bu proje kapsamında herhangi bir veritabanı kullanımına ihtiyaç yoktur.

## **5.3 Girdi-Çıktı Tasarımı**


Sistem girdi olarak üzerinde işlem yapmak için ".tsv",".wav" veya ".mp3" uzantılı dosya almaktadır. Bu dosyaların içeriğinde ise zaman serilerine ait nümerik veriler bulunmaktadır. Çıktı olarak ise okunan zaman serilerinin önceki bölümlerde bahsedilen ön işleme veya sinyal hizalama fonksiyonları kullanılması sonucu yeni sinyalleri ekrana çizer ve nümerik verisini dosyanın kendi uzantısında veya ".txt" uzantılı olarak kaydeder.

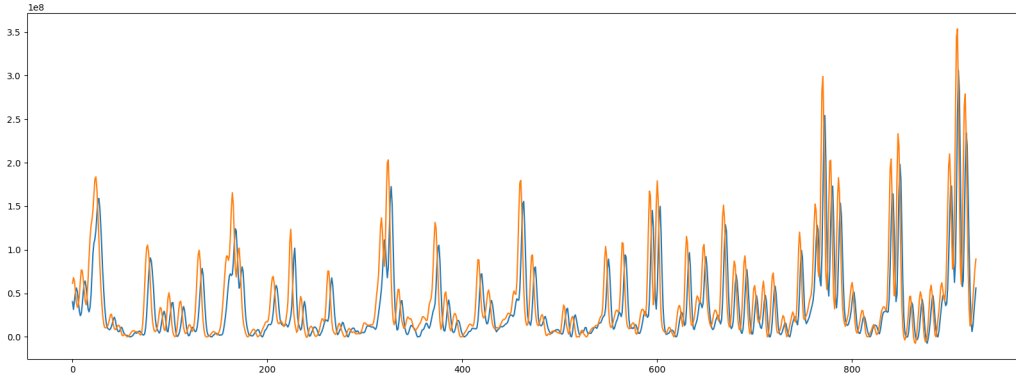
## 6 Uygulama

PyTimeWarp uygulaması bilimsel hesaplamalarda hızla standart haline gelen, son derece kullanışlı, yüksek seviyeli, açık kaynak kodlu, farklı platformlarda (Windows / MacOS / Linux) kullanıma tamamen uyumlu, son derece esnek olan Python dilinde yazılmıştır. Kullanıcı grafik arayüz kullanarak her bir yöntemi ve algoritmayı efektif olarak uygulayabilmekte ve sonuçlarını görebilmektedir. Şekil 6.1 de gerçekleştirilen yazılımın giriş penceresi gösterilmektedir.



**Şekil 6.1** PyTimeWarp Uygulaması Ana Penceresi

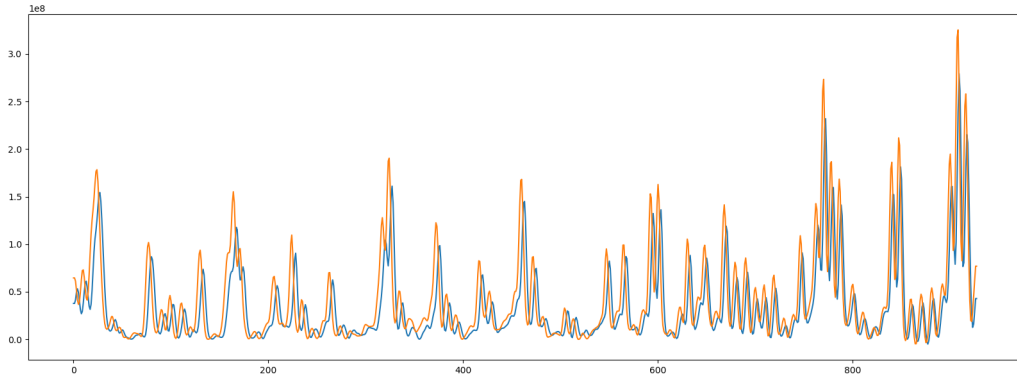
Kullanıcı File menüsünden Open File komutunu kullanarak ya da  simgesini kullanarak bilgisayarında kayıtlı bulunan ham veriyi programa yükleyebilir ve Şekil 6.2 deki gibi verinin görsel halini otomatik olarak görebilmektedir.



Şekil 6.2 Yüklenen Sinyallerin Görselleştirilmesi

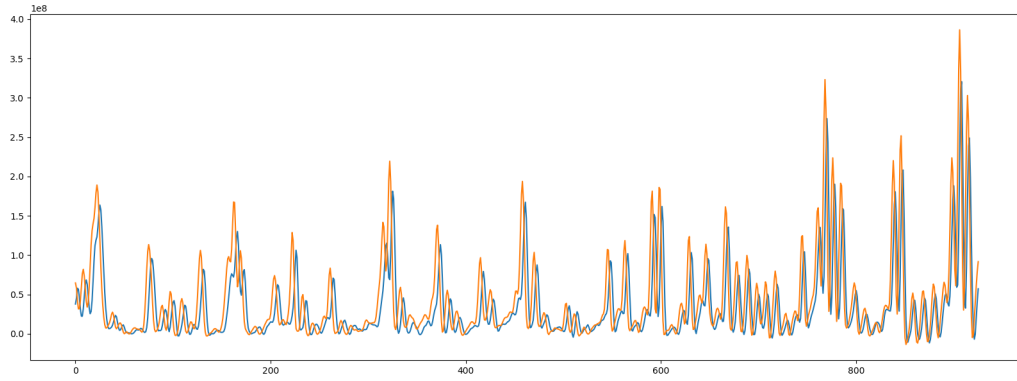
Sinyalleri yuumuşatma işlemi için **Preprocessing** menüsünden **Smoothing** komutu çağrılmaktadır.

Şekil 6.3 Sinyal Yumuşatma Penceresi



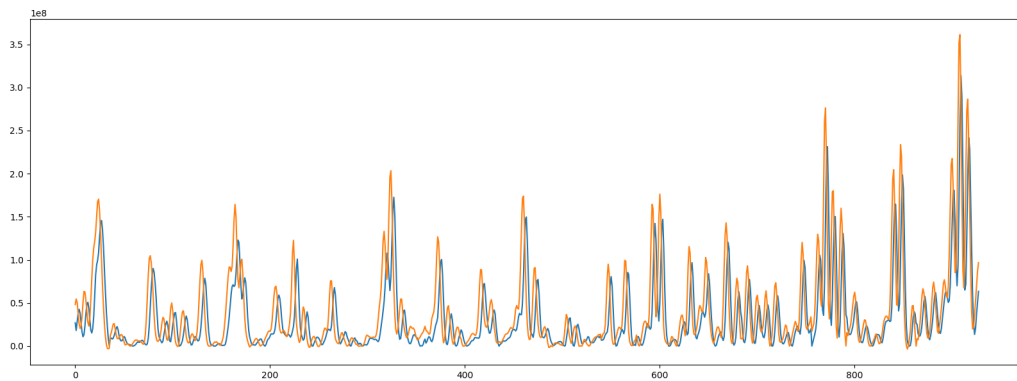
**Şekil 6.4** Yumuşatılmış Sinyallerin Görüntüsü

Sinyallerin tepe değerini belirginleştirmek için **Preprocessing** menüsünden **Resolution Enhancement** komutu çağrılmaktadır.



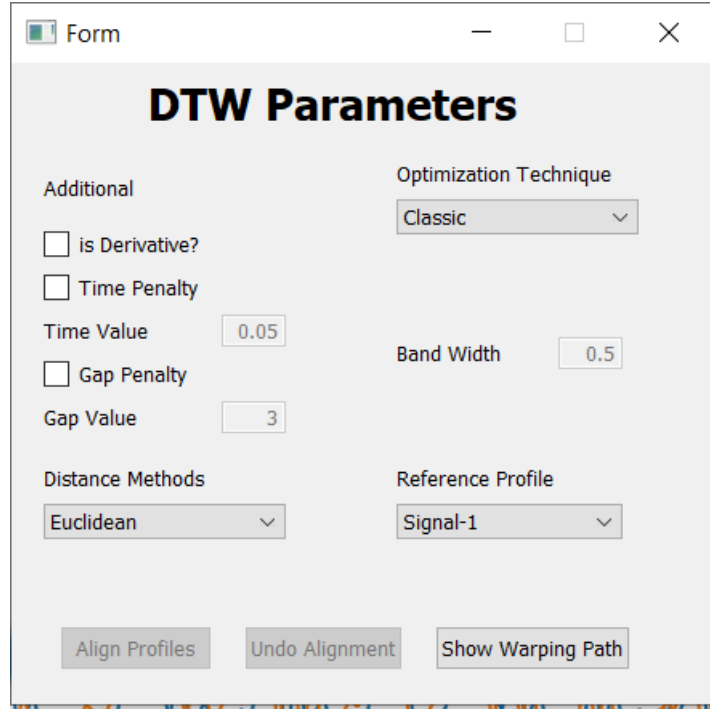
**Şekil 6.5** Tepe Değerleri Belirginleştirilmiş Sinyallerin Görüntüsü

Sinyallerin taban hizasını düzeltmek için **Preprocessing** menüsünden **Baseline Adjustment** komutu çağrılmaktadır.



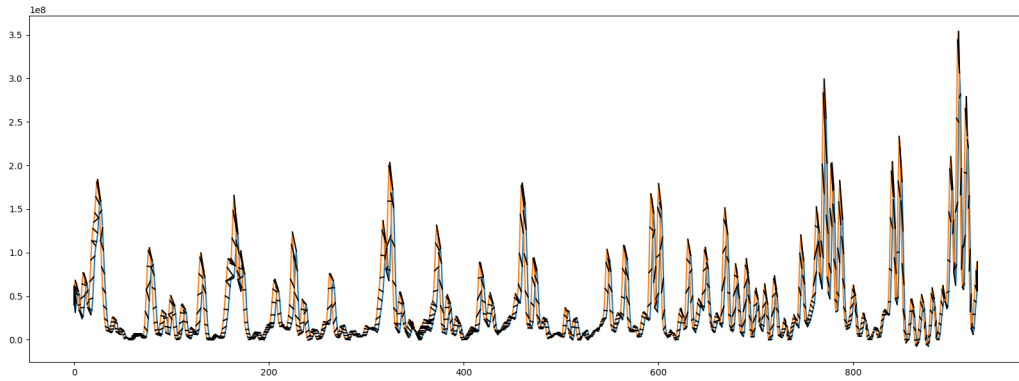
**Şekil 6.6** Taban Hizaları Düzeltilmiş Sinyallerin Görüntüsü

Sinyalleri hizalamak için DTW tekniği kullanılmak isteniyorsa **Warping** menüsünden **Dynamic Time Warping** komutu çağrılmalıdır.



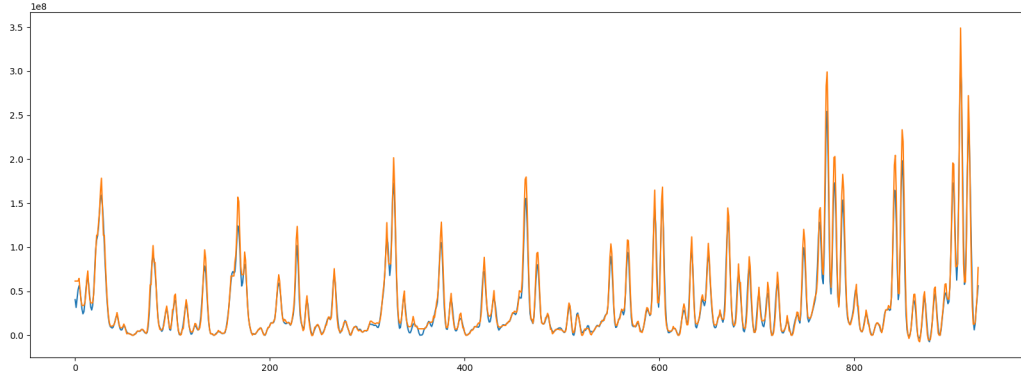
Şekil 6.7 DTW Penceresi

**Show Warping Path** butonuna basıldığında sinyallerin hangi noktaların hangi noktalarla eşleşeceğini bulur ve elde edilen eğrilme yolunun sonucu gösterilir.



Şekil 6.8 DTW Yönteminde Elde Edilen Eğrilme Yolu

**Align Profiles** komutu kullanıldığında sinyallerin hizalanmış görüntüsü elde edilir.

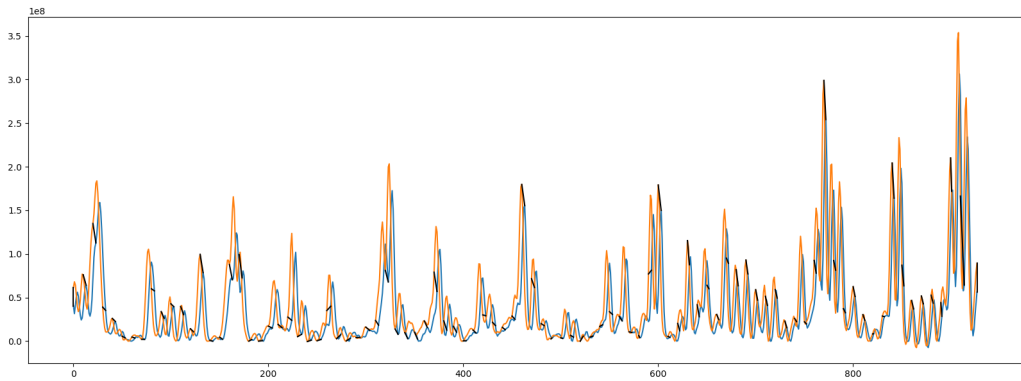


Şekil 6.9 DTW İle Hizalanmış Sinyaller

Sinyalleri hizalamak için COW tekniği kullanılmak isteniyorsa **Warping** menüsünden **COW** komutu çağrılmalıdır.

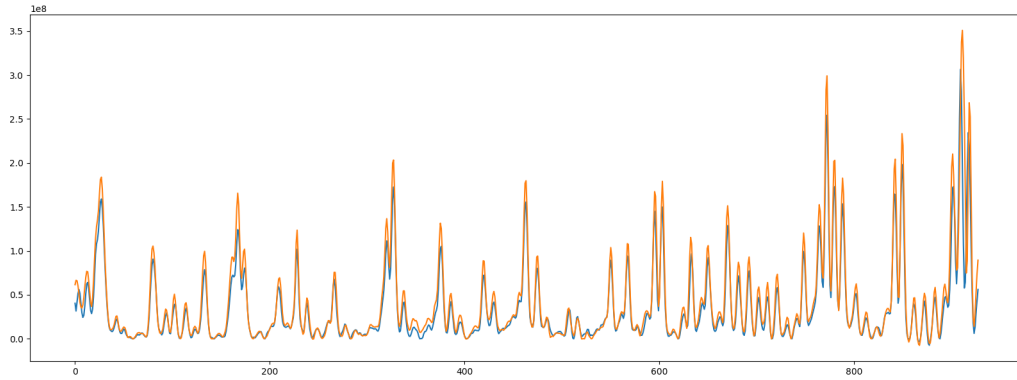
Şekil 6.10 COW Penceresi

**Show Warping Path** butonuna basıldığında sinyallerin hangi noktaların hangi noktalarla eşleşeceğini bulur ve elde edilen eğrilme yolunun sonucu gösterilir.



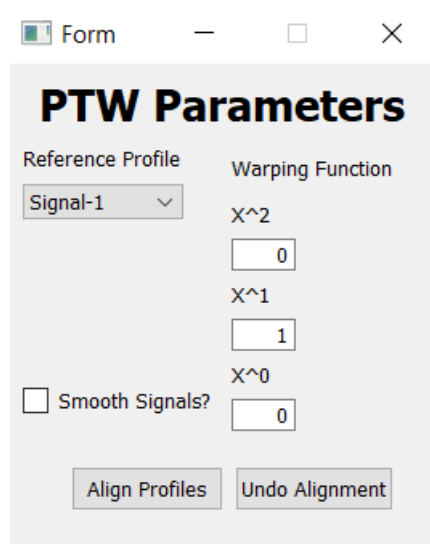
Şekil 6.11 COW Yönteminde Elde Edilen Eğrilme Yolu

**Align Profiles** komutu kullanıldığında sinyallerin hizalanmış görüntüsü elde edilir.



**Şekil 6.12** COW İle Hizalanmış Sinyaller

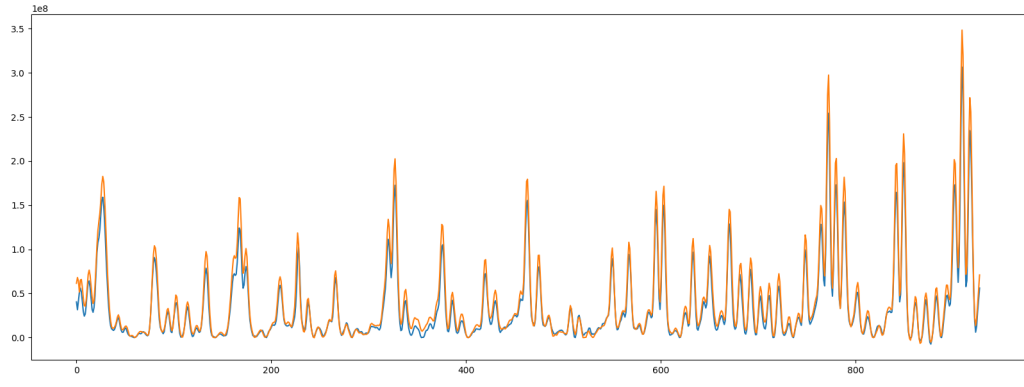
Sinyalleri hizalamak için PTW tekniği kullanılmak isteniyorsa **Warping** menüsünden **PTW** komutu çağrılmalıdır.



**Şekil 6.13** PTW Penceresi

**Align Profiles** komutu kullanıldığında sinyallerin hizalanmış görüntüsü elde edilir.





**Şekil 6.14** PTW İle Hizalanmış Sinyaller

## 7 Deneysel Sonuçlar

Bu bölümde sinyal hizalama işleminde kullanılan eğrilme yöntemlerinin farklı veri setlerine uygulanması sonucu hizalama performans ölçütlerine göre sonuçlar özetlenmiştir. Önceki bölümlerde yer verilen ham veride meydana gelen zaman kayması sorununa çözüm üretecek farklı yöntemler ve parametrenin hangisinin daha iyi sonuç vereceğini belirleyen ölçütler örnek veri setlerinde denenmiş ve aşağıdaki tablolarda özetlenmiştir. Hizalama işlemi ve hizalama işlemini değerlendirmek için PyTimeWarp yazılımı i5 işlemci, 16 GB DDR-4 Ram, Windows 10 İşletim Sistemi ve PyCharm 2019 Edition yazılım geliştirme ortamında derlenmiş, çalıştırılmış ve aşağıdaki tablolarda bulunan sonuçlar alınmıştır.

Bütün yöntemlerin kullanılan ölçütlere göre değerlendirilmesi yapılmış ve bu sonuçlar kullanıcıya tablo şekline yeni pencerede gösterilmiştir. Tablo 7.1 de hizalanmamış örnek veri setinin korelasyon ve ortalama farkların karekökü(RMS) değerleri verilmiştir.

Test Numarası	Korrelasyon	RMS
1	0.67	335.5
2	0.62	12008.35
3	0.82	9955.37
4	0.45	10044.82
5	0.52	8236.2
6	0.61	6900.37
7	0.66	4713.27

**Tablo 7.1** Ham Verinin Bazı Sayısal Değerleri

Tablo 7.2 de veri setinin DTW yöntemiyle hizalandıktan sonraki korelasyon RMS,hizalanan sinyalin tepe noktalarının eşleşme yüzdeleri ve hesaplama süreleri gösterilmektedir.

Tablo 7.1 deki sayısal verilerle Tablo 7.2 deki sayısal veriler karşılaştırıldığında hizalama sonrası elde edilen yeni sinyaller arasındaki korelasyon değeri artmış ve 1

Test Numarası	Korelasyon	RMS	PMS	Time
1	0.99	8159537.93	72.62	3.84
2	0.76	5599247.48	19.0	4.15
3	0.9	13330998.6	30.83	4.02
4	0.84	4888473.11	28.78	4.05
5	0.57	119449441.94	26.49	4.8
6	0.62	10890938.45	22.86	4.0
7	0.71	21786624.45	18.85	3.94

**Tablo 7.2** DTW Yöntemiyle Hizalama Bazı Sayısal Değerleri

numaralı testte tepe noktası eşleşme yüzdesi yüzde 72.62 olarak bulunmuştur. Bu işlemler sırasında boşluk cezası 3 ve zaman cezası 0 seçilmiştir.

Test Numarası	Korelasyon	RMS	PMS	Time
1	0.96	17083118.63	62.92	3.65
2	0.62	7476053.03	23.3	6.09
3	0.85	18709877.25	34.16	6.16
4	0.59	11029956.78	26.53	6.22
5	0.49	125763345.12	22.95	7.42
6	0.46	13172644.47	18.08	6.22
7	0.75	21829318.66	29.02	6.11

**Tablo 7.3** COW Yöntemiyle Hizalama Bazı Sayısal Değerleri

Tablo 7.1 deki sayısal verilerle Tablo 7.3 deki sayısal veriler karşılaştırıldığında hizalama sonrası elde edilen yeni sinyaller arasındaki korelasyon değeri artmış ve 1 numaralı testte tepe noktası eşleşme yüzdesi yüzde 62,92 olarak bulunmuştur. Bu işlemler sırasında segment uzunluğu=10 ve slack=3 seçilmiştir.

Test Numarası	Korelasyon	RMS	PMS	Time
1	1.00	814.13	74.44	0.09
2	0.64	11141.39	17.75	0.44
3	0.8	6487.67	18.45	0.41
4	0.75	2008.92	22.4	0.27
5	0.52	7372.79	20.77	0.44
6	0.67	5801.7	18.64	0.14
7	0.88	8450.08	17.86	0.42

**Tablo 7.4** PTW Yöntemiyle Hizalama Bazı Sayısal Değerleri

Tablo 7.1 deki sayısal verilerle Tablo 7.4 deki sayısal veriler karşılaştırıldığında hizalama sonrası elde edilen yeni sinyaller arasındaki korelasyon değeri artmış ve 1 numaralı testte tepe noktası eşleşme yüzdesi yüzde 74.14 olarak bulunmuştur. Bu işlemler sırasında gap=3 ve kesme fonksiyonu  $x$  seçilmiştir.

## 8 Performans Analizi

---

Tablo 7.4 de görüldüğü üzere en fazla başarılı ve hızlı sonuçlar PTW yönteminden elde edilmiştir. PTW yönteminin bu kadar başarılı olabilmesini ön işleme işlemlerinin yapılmış olmasıdır. Fakat sadece veriler içerisindeki farklı taban hizasından dolayı daha farklı tekniklerin kullanılması da kaçınılmazdır.

Tablo 7.2, 7.3 ve 7.4 deki sonuçlar göz önüne alındığında en başarılı yöntemin PTW olarak gösterilmesine rağmen 20 farklı deney sonucunda ortalama sonuçlar incelendiğinde yöntemlerin birbirine çok yakın sonuçlar verdiği görülmektedir. Bunun nedeni belirli bir yöntemin her tür veriye her zaman en uygun sonuç vermesinin mümkün olmamasından kaynaklanmaktadır. Bu amaca uygun olarak mümkün olan bütün yöntemler uygulanmalı ve en az hatalı sonuç en uygun sonuç olarak kabul edilmelidir.

Biyoenformatikte ve ses dalgalarının saklanması için kullanılan tek boyutlu zaman serilerinde meydana gelen zaman kayması problemini çözmek için geliştirilen yazılımla son kullanıcı için anlaşılır ve hızlı sonuçlar üretebilen bir yazılım geliştirilmiştir. Geliştirilen yazılım Python dilinde yazıldığı için farklı işletim sistemleri üzerinde rahatlıkla çalışabilmektedir. Bu alanda gereksinim duyulan açık kaynak kodlu yazılım ihtiyacının karşılanmasına yönelik tasarlanan bu yazılım birçok türdeki zaman serisi için kolaylıkla kullanılabilir.

Bölüm 8 de özetlenen tüm sonuçlardan anlaşılacağı üzere, sinyal hizalama sonrası en yüksek korelasyon ve tepe noktası eşleme yüzde değerleri tüm yöntemlerde ortalama olarak birbirine çok yakın hesaplanmıştır. Kısa sürede sinyal hizalayabilmesi açısından PTW yöntemi diğer yöntemlere göre öne çıkmaktadır. Ancak, sinyal içerisinde taban hizası ya da gürültü bulunduğunda PTW yöntemi istenilen başarıyı verememiştir. Bu durumlarda DTW ve COW yöntemleri daha başarılı sonuçlar vermiştir. Sinyal içerisindeki taban hizasının ayarlanması için ön işlem olarak önerilen taban hizası ayarlama yöntemi yetersiz kalmıştır. DTW yönteminde kullanılan türev tabanlı hizalama, farklı uzaklık fonksiyonları kullanarak maliyet matrisi hesaplama, zaman cezası ve boşluk cezası modifikasyonları hizalama sonuçlarını hem doğruluk hem de zaman açısından olumlu etkilemiştir. COW yönteminde kullanılan korelasyon işleminin hesaplama karmaşıklığı fazla olduğu için hesaplama süresi veri sayısının daha fazla olduğu durumlarda daha uzayacaktır. Bu sebepten dolayı ilerleyen çalışmalarda kosinüs benzerliği ya da farklı benzerlik fonksiyonları kullanılabilir.

- [1] F. Karabiber, J. L. McGinnis, O. V. Favorov, and K. M. Weeks, "Qushape: Rapid, accurate, and best-practices quantification of nucleic acid probing information, resolved by capillary electrophoresis," *Rna*, vol. 19, no. 1, pp. 63–73, 2013.
- [2] F. Karabiber, "An automated signal alignment algorithm based on dynamic time warping for capillary electrophoresis data," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 21, no. 3, pp. 851–863, 2013.
- [3] F. Karabiber and M. Balcılar, "A signal alingment method based on dtw with new modification," in *2013 21st Signal Processing and Communications Applications Conference (SIU)*, IEEE, 2013, pp. 1–4.
- [4] E. J. Keogh and M. J. Pazzani, "Derivative dynamic time warping," in *Proceedings of the 2001 SIAM international conference on data mining*, SIAM, 2001, pp. 1–11.
- [5] M. Müller, "Dynamic time warping," *Information retrieval for music and motion*, pp. 69–84, 2007.
- [6] V. Niennattrakul and C. A. Ratanamahatana, "On clustering multimedia time series data using k-means and dynamic time warping," in *2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07)*, IEEE, 2007, pp. 733–738.
- [7] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE transactions on acoustics, speech, and signal processing*, vol. 26, no. 1, pp. 43–49, 1978.
- [8] P. Senin, "Dynamic time warping algorithm review," *Information and Computer Science Department University of Hawaii at Manoa Honolulu, USA*, vol. 855, no. 1-23, p. 40, 2008.
- [9] T. G. Bloemberg, J. Gerretzen, A. Lunshof, R. Wehrens, and L. M. Buydens, "Warping methods for spectroscopic and chromatographic signal alignment: A tutorial," *Analytica chimica acta*, vol. 781, pp. 14–32, 2013.
- [10] N.-P. V. Nielsen, J. M. Carstensen, and J. Smedsgaard, "Aligning of single and multiple wavelength chromatographic profiles for chemometric data analysis using correlation optimised warping," *Journal of chromatography A*, vol. 805, no. 1-2, pp. 17–35, 1998.
- [11] G. Tomasi, F. Van Den Berg, and C. Andersson, "Correlation optimized warping and dynamic time warping as preprocessing methods for chromatographic data," *Journal of Chemometrics: A Journal of the Chemometrics Society*, vol. 18, no. 5, pp. 231–241, 2004.

- [12] A. Van Nederkassel, M. Daszykowski, P. Eilers, and Y. Vander Heyden, “A comparison of three algorithms for chromatograms alignment,” *Journal of Chromatography A*, vol. 1118, no. 2, pp. 199–210, 2006.

### BİRİNCİ ÜYE

**İsim-Soyisim:** Recep Furkan KOÇYİĞİT  
**Doğum Tarihi ve Yeri:** 27.10.1998, İstanbul  
**E-mail:** furkankocyigitfk@gmail.com  
**Telefon:** 0541 726 49 61  
**Staj Tecrübeleri:**

### Proje Sistem Bilgileri

**Sistem ve Yazılım:** Windows İşletim Sistemi, Python  
**Gerekli RAM:** 2GB  
**Gerekli Disk:** 256MB