



ADAPTİF FİLTERE ME KULLANILARAK SOLUNUM ORANI TESPİT SİSTEM TASARIMI

Mohamed Alfituri Albashir ELHASHMI

**2021
YÜKSEK LİSANS TEZİ
BİYOMEDİKAL MÜHENDİSLİĞİ**

**Tez Danışmanı
Dr. Öğr. Üyesi Ahmet Reşit KAVSAOĞLU**

**ADAPTİF FİLTERE ME KULLANILARAK SOLUNUM ORANI TESPİT
SİSTEM TASARIMI**

Mohamed Alfituri Albashir ELHASHMI

**T.C.
Karabük Üniversitesi
Lisansüstü Eğitim Enstitüsü
Biyomedikal Mühendisliği Anabilim Dalında
Yüksek Lisans Tezi
Olarak Hazırlanmıştır**

**Tez Danışmanı
Dr. Öğr. Üyesi Ahmet Reşit KAVSAOĞLU**

**KARABÜK
Ağustos 2021**

Mohamed Alfituri ELHASHMI tarafından hazırlanan “ADAPTİF FİLTERE ME KULLANILARAK SOLUNUM ORANI TESPİT SİSTEM TASARIMI ” başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Dr. Öğr. Üyesi Ahmet Reşit KAVSAOĞLU
Tez Danışmanı, Biyomedikal Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından Oy Birliği ile Biyomedikal Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 20/08/2021.

<u>Ünvanı, Adı SOYADI (Kurumu)</u>	<u>İmzası</u>
Başkan : Dr. Öğr. Üyesi Sabri ALTUNKAYA (NEÜ)	Çevrimiçi
Üye : Dr. Öğr. Üyesi Hacı Mehmet KAYILI (KBÜ)
Üye : Dr. Öğr. Üyesi Ahmet Reşit KAVSAOĞLU (KBÜ)

KBÜ Lisansüstü Eğitim Enstitüsü Yönetim Kurulu, bu tez ile, Yüksek Lisans derecesini onamıştır.

Prof. Dr. Hasan SOLMAZ
Lisansüstü Eğitim Enstitüsü Müdürü

“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”

Mohamed Alfituri Albashir ELHASHMI

ÖZET

Yüksek Lisans Tezi

ADAPTİF FİLTREME KULLANILARAK SOLUNUM HIZI TESPİT SİSTEM TASARIMI

Mohamed Alfituri Albashir ELHASHMI

Karabük Üniversitesi

Lisansüstü Eğitim Enstitüsü

Biyomedikal Mühendisliği Anabilim Dalı

Tez Danışmanı:

Dr. Öğr. Üyesi Ahmet Reşit KAVSAOĞLU

Ağustos 2021, 55 sayfa

Adaptif filtre sayısal filtreleme tekniğinin bir çeşididir. Klasik filtreleme tekniğine benzemeyen bu filtre biçimi, üç temel elemandan oluşmaktadır. Bu elemanlar toplayıcı eleman, ağırlıklandırıcı (çarpım) elemanı ve bir sayısal filtre yapısıdır. Bu elemanlara sahip bir sistem, çevre ortamına bağlı olarak filtre karakteristiğinde gereken değişmeyi filtre katsayılarını değiştirerek yapabilmektedir.

Solunum, göğüs kafesinin ve akciğerlerin hareketine ve bu organlarda meydana gelen art arda hacim ve basınç değişikliklerine karşılık gelir. Solunum hızı (RR) ise dakikada ki solunum sıklığıdır. (RR) ciddi hastalıkları tespit etmek ve izlemek için kullanıldığından adaptif filtreleme kullanarak bir solunum hızı tespit sisteminin tasarlanması önemli konulardan biri olarak kabul edilir.

Bu alıřmada solunum hızını algılayan bir sistem gerekleřtirilmiřtir. Bu sistemde adaptif filtreleme, ses sinyalinde konuřma sınırını tespit etme algoritması, iki adet ierisine mikrofon yerleřtirilen stetoskop, USB ses adaptörü ve bir MATLAB GUI arayüz tasarımı bulunmaktadır. Mikrofonlardan biri soluk borusunun (trakeal), diğeri de kalbin üzerine yerleřtirilmektedir. Kalp üzerindeki stetoskoptan gelen ses sinyalleri gürültü kaynağı olarak kullanılıp soluk borusu üzerine yerleřtirilen mikrofondan alınan ses sinyallerinin bu gürültüden arındırılması vasıtasıyla bir ön iřlem gerekleřtirilmektedir. Bu ön iřlem sonrası adaptif filtre ıkıřında soluk borusu üzerine yerleřtirilen stetoskoptan kalp seslerinin arındırılmasıyla temiz bir solunum ses sinyaline ulařılmaktadır. Bu temiz solunum ses sinyali üzerinde konuřma sınırlarını tespit etme algoritmasının alıřtırılması ile soluk alma ve soluk verme zaman aralıkları tespit edilebilmektedir. Tespit edilen bu zaman aralıklarının kullanılmasıyla solunum hızı elde edilmektedir.

Anahtar Sözcükler: Adaptif filtreleme, Konuřma sınırı tespiti, Solunum hızı

Bilim Kodu: 92512

ABSTRACT

M. Sc. Thesis

RESPIRATORY RATE DETECTION SYSTEM DESIGN USING ADAPTIVE FILTERING

Mohamed Alfituri Albashir ELHASHMI

Karabük University

Institute of Graduate Programs

The Department of Biomedical Engineering

Thesis Advisor:

Assist. Prof. Dr. Ahmet Reşit KAVSAOĞLU

August 2021, 55 pages

The adaptive filter is a variation of the digital filtering technique. This form of filter, which does not resemble the classical filtering technique, consists of three basic elements. These elements are collector element, weighting (multiplication) element and a digital filter structure. A system, which has these elements, can make the necessary change in the filter characteristics depending on the environmental media by changing the filter coefficients.

Breathing corresponds to the movement of the thorax and lungs and to volume and pressure changes that occur successive in these organs. Respiratory rate (RR) means the respiratory frequency per minute. Since the RR is used to detect and monitor the serious diseases, designing a respiratory rate detection system by means of using adaptive filtering is considered one of the important issues.

In this study, a system that detects respiratory rate has been implemented. In this system, there are adaptive filtering, speech limit detection algorithm in the sound signal, two stethoscopes whose internal part is placed a microphone and a MATLAB GUI interface design. One of the microphones is placed to upper part of the trachea and the other is placed to upper part of the heart. The sound signals coming from the stethoscope on the heart are used as a noise source and a preprocessing is carried out by means of making free from this noise the sound signals received from the microphone placed on the trachea. After this pre-processing, a clean breathing sound signal is reached by means of making free the heart sounds from the stethoscope placed on the trachea at the adaptive filter output. Inhalation and exhalation time intervals can be determined by running the speech limit detection algorithm on this clean breathing sound signal. The respiratory rate is obtained by using these determined time intervals.

Key Words : Adaptive filtering, Speech limit detection, Respiration rate

Science Code : 92512

TEŐEKKÜR

Bu tez alıőmasının planlanmasında, araştırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteęini esirgemeyen, engin bilgi ve tecrübelerinden yararlandıęım, yönlendirme ve bilgilendirmeleriyle alıőmamı bilimsel temeller ışığında őekillendiren sayın hocam Dr. Öğr. Üyesi Ahmet Reőit KAVSAOęLU'a sonsuz teőekkürlerimi sunarım.

Sevgili aileme manevi hiçbir yardımı esirgemedен yanımda oldukları için tüm kalbimle teőekkür ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
KABUL.....	ii
ÖZET.....	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER	ix
ŞEKİLLER DİZİNİ.....	xi
ÇİZELGELER DİZİNİ	xii
SİMGELER VE KISALTMALAR DİZİNİ	xiii
BÖLÜM 1	1
GİRİŞ	1
1.1. LİTERATÜRDE YAPILAN BENZER ÇALIŞMALAR	4
BÖLÜM 2	8
SOLUNUM SİSTEMİ	8
2.1. SOLUNUM YOLLARI.....	9
2.1.1. Yutak (Farenks)	10
2.1.2. Gırtlak (Larnks)	10
2.1.3. Soluk Borusu (Trakea).....	11
BÖLÜM 3	12
MATERYAL VE YÖNTEMLER	12
3.1. ADAPTİF FİLTRELER.....	12
3.1.1. Adaptif Filtre Uygulamaları	15
3.1.1.1. Sistem Tanımlaması.....	15
3.1.1.2. Ters Sistem Tanımlaması.....	16
3.1.1.3. Gürültü Giderme	16
3.1.1.4. Lineer Tahmin.....	17

	<u>Sayfa</u>
3.2. SES SİNYALİNDE KONUŞMA SINIRI TESPİT ETME ALGORİTMASI	17
3.3. MATLAB GUI İLE TASARIM.....	23
BÖLÜM 4	27
DENEYSEL UYGULAMA VE SONUÇLARI	27
4.1. DENEYSEL UYGULAMA	27
4.1.1. Grafiksel Kullanıcı Arayüzü (GUI) Elemanları	29
4.2. ADAPTİF FİLTRELEMENİN GÜRÜLTÜ AZALTMADAKİ ETKİSİ	31
4.3. UYGULAMA SONUÇLARI.....	33
BÖLÜM 5	38
SONUÇ VE TARTIŞMA	38
KAYNAKLAR	40
EK AÇIKLAMALAR A. MATLAB KODLARI	44
ÖZGEÇMİŞ	55

ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 1.1. Sistem blok şeması.....	3
Şekil 2.1. Solunum sistemi [20].	8
Şekil 2.2. Üst solunum yolları [23].	9
Şekil 2.3. Gırtlak anatomisi [24].	10
Şekil 3.1. FIR filtre yapısı [26].	12
Şekil 3.2. Adaptif gürültü yok etme filtre yapısı [27].	13
Şekil 3.3. Adaptif filtre genel blok diyagramı [30].	14
Şekil 3.4. Sistem tanımlaması için adaptif filtre [33].	15
Şekil 3.5. Adaptif filtre ile gürültü önleme [34].	16
Şekil 3.6. Doğrusal tahmin için Adaptif filtre [35].	17
Şekil 3.7. Ses sinyalinde konuşma sınırı algılama algoritması.	18
Şekil 3.8. Spektrogram analizi. a) orijinal solunum sinyali, b) spektrogram, c) solunum sinyalinin tespit edilmesi.	21
Şekil 3.9. Frekans alanında Hanning penceresi.	22
Şekil 3.10. MATLAB GUI bileşenleri [42].	24
Şekil 3.11. GUIDE editörü [42].	25
Şekil 3.12. Programlı olarak kodlanmış bir GUI örneği [43].	26
Şekil 4.1. Kullanılan donanım, a) dizüstü bilgisayar, b) Stetoskop üzerindeki kombine mikروفon.	28
Şekil 4.2. Grafiksel kullanıcı Arayüzü (GUI).	28
Şekil 4.3. "Start_Rec" düğmesine bastıktan sonra ortaya çıkan GUI.	29
Şekil 4.4. "Solunum algılama" düğmesine bastıktan sonra ortaya çıkan GUI.	30
Şekil 4.5. Uygulamanın MATLAB kodları.	31
Şekil 4.6. Gürültülü sinyal, filtrelenmiş sinyal ve gürültü kaynağı sinyali.	32
Şekil 4.7. 3 farklı μ değerinde üretilen hata sinyali.	32
Şekil 4.8. En genç kişiden elde edilen sonuçlar.	34
Şekil 4.9. En yaşlı kişinin sonuçları.	37

ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
Çizelge 4.1. Birinci kişi için solunum hızı sonuçları (en genç kişi).	33
Çizelge 4.2. Son kişinin solunum hızı sonuçları (en yaşlı kişi).	35

SİMGELER VE KISALTMALAR DİZİNİ

KISALTMALAR

RR	: Respiratory Rate (Solunum hızı)
RC	: Respiratory Cycle (Solunum döngüsü)
LOG	: Logarithmic (logaritmik)
IP	: Plethysmography (Pletismografi)
RIP	: Respiratory inductive plethysmography (Solunum indüktif pletismografi)
ECG	: Electrocardiograph (Elektrokardiyografi)
SRLMS	: Signed- Regressor Algorithm (İşaretli-Regresör Algoritması)
SLMS	: Sign Algorithm (İşaret - İmza Algoritması)
SSLMS	: Sign – Sign Algorithm (İşaret - İşaret Algoritması)
RSP	: Recorded Respiratory Signal (Kaydedilmiş Solunum Sinyali)
BLMS	: Block LMS Algorithm (Blok LMS Algoritması)
RRORİG	: Original Tachogram (Orijinal Takogram)
BPM	: Breaths Per minute (Dakikada nefes sayısı)
GUI	: Graphical User Interface (Grafiksel Kullanıcı Arayüzü)
ANC	: Adaptive Noise Cancellation (Adaptif Gürültü Önleme)
HRV	: Heart Rate Variability (Kalp Atış Hızı Değişkenliği)
PSD	: Power Spectral Density (Güç Spektral Yoğunluğu)
PPG	: Photoplethysmography (Fotopletismografi)
PPT	: Pulse Pass Time (Nabız Geçiş Süresi)
AC	: Alternating Current (Alternatif akım)
LMS	: Least Mean Square (En Küçük Kareler Yöntemi)
FIR	: Finite Impulse Response Filter (Sonlu Dürtü Yanıtı filre)
EGG	: Electrogastrogram (elektrogastrogram)
QRS	: QRS heart wave (QRS kalp dalgası)
WHMS	: Wearable Health Monitoring System (Giyilebilir Sağlık İzleme Sistemi)
ACC	: Accelerometer (İvmeölçer)
GALL	: Gradient Adaptive Lattice Laguerre (Gradyan Adaptif Laguerre Kafesi)

- AF : Adaptive Filter (Adaptif Filtre)
PPV : Positive Predicted Value (Pozitif Tahmini Deęer)
EOG : Electrooculography (Elektrookülografi)
NPV : Negative Predicted Value (Negatif Tahmini Deęer)
STFT : Short Time Fourier Transform (Kısa Zamanlı Fourier Dönüşümü)
NLMS : Normalized Least Mean squares filter (Normalleştirilmiş En Küçük Ortalama kareler filtresi)

SİMGELER

- H : hidrojen
CO₂ : karbon dioksit
HCO₃: bikarbonat
O₂ : oksijen

BÖLÜM 1

GİRİŞ

Solunum sisteminin temel işlevi oksijeni almak ve karbondioksiti ortadan kaldırmaktır. Solunan oksijen akciğerlere girer ve alveollere ulaşmaktadır. Alveoller ve çevresindeki kılcal damarları kaplayan hücre katmanlarının her biri yalnızca bir hücre kalınlığındadır ve birbirleriyle çok yakın temas halindedir. Hava ve kan arasındaki katmanların kalınlığı ortalama 1 mikron (1/10000 cm)'e kadardır. Oksijen, bu hava-kan bariyerinden hızla kılcal damarlardaki kana geçmektedir. Benzer şekilde, karbondioksit kandan alveollere geçer ve daha sonra nefesle dışarı verilmektedir [1].

Solunum sistemi, merkezi sinir sistemi, solunumla ilgili motor nöronlar ve solunum kasları arasındaki karmaşık bir etkileşimdir. Merkezi sinir sistemi, çevresel kimyasal reseptörlere dayalı solunum hızını kontrol etmekten sorumludur [2].

Sağlıklı erişkinde solunum sistemin kontrolü H⁺ iyonu ile gerçekleşmektedir. Kanda bulunan CO₂ su ile birleşir ve bilahare H⁺ ve HCO₃⁻ çözünmektedir. H⁺, kan-beyin bariyerini geçer solunum sistemini uyarmaktadır. Korteks, solunum kasları ve tendon iççikleri gibi periferik uyarılardan gelen uyarılarla beraber, önce inspirasyonun başlangıcında üst solunum yolunun açılmasını koruyan solunum kasları ve ardından sakin solunumda diyafram, zorlu solunumda yardımcı solunum kaslarını aktive eden motor nöronları aktive olmaktadır. Toraks kafesi solunum sistemine katılmaktadır, toraks kafesi genişlenerek ve darlanarak ventilasyon gerçekleşmektedir [3].

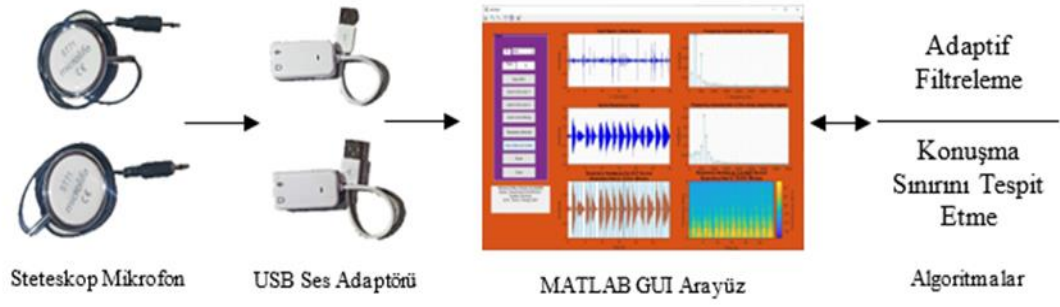
Solunum hızı ve derinliği, beyindeki medulla oblongatada bulunan solunum merkezi tarafından kontrol edilmektedir. Solunumun hızı bireyin yapısına ve yaşına bağlı olarak değişmektedir. Normal solunum sessiz ve düzenli olup hiç bir çaba gerektirmez [3].

Solunum, kendiliğinden gerçekleşir ancak kısmen kontrol edilebilmektedir. Her soluk alma ve verme arasında kısa bir süre vardır (dört kalp atışına karşılık bir kez solunum gerçekleşmektedir). Her nefes almada, akciğerlere giren ya da her soluk vermede akciğerlerden çıkan hava miktarına tidal volüm (solunum havası) denir. Bu hava miktarı yaklaşık 500 santimetre küptür. Dinlenme sırasında her solunum genellikle aynı derinlikte gerçekleşmektedir. Solunum hareketlerinin gözlenmesi ile solunum derinliği değerlendirilmektedir. Solunumun derinliği alınan hava miktarının normalin altında veya üstünde oluşuna göre derin ve yüzeysel olarak tanımlanmaktadır [4].

Solunum sistemi sinyallerini gözlemlemek, uyku bozuklukları, uykusuzluk gibi birçok hastalığı tespit eden önemli şeylerden biridir. Yetişkinler için solunum hızı dakikada 10-20 nefes arasındadır, yeni doğanlarda ise dakikada 44 nefes vardır. Birçok ses, kas hareketi ve kasılması nedeniyle solunum sinyalini etkiler. Gürültüyü ortadan kaldırmak için Adaptif filtre kullanılmaktadır [5].

Adaptif filtre, değişken parametrelerle kontrol edilen bir transfer işlevine ve bu parametreleri bir optimizasyon algoritmasına göre ayarlamak için gerekli bir araca sahip doğrusal filtreli bir sistemdir. Optimizasyon algoritmalarının karmaşıklığından dolayı, neredeyse tüm adaptif filtreler dijital filtrelerdir. Bazı uygulamalar için adaptif filtreler gereklidir çünkü istenen işleme operasyonunun bazı parametreleri (örneğin, yankı yapan bir uzaydaki yansıtıcı yüzeylerin konumları) önceden bilinmemektedir veya değişmektedir. Kapalı döngü adaptif filtre, transfer işlevini daha iyi gerçekleştirmek için bir hata sinyali geri beslemesine sahiptir. Bu geri besleme sayesinde kapalı döngüdeki tekrar eden döngüsel süreçte transfer işlevinin ne şekilde değişime uğraması gerektiğini belirleyen bir algoritma ile desteklenerek transfer işlemi en kısa sürede gerçekleştirilmektedir. Burada esas olan ideal performans için tekrarlanan süreç maliyetlerinin en aza indirilmesidir. En yaygın maliyet fonksiyonu, hata sinyali ortalamasının karesidir [6].

Dijital sinyal işlemcilerin gücü arttıkça, adaptif filtreler çok daha yaygın hale gelmiştir. Cep telefonları ve diğer iletişim cihazları, video kameralar ve dijital kameralar ve tıbbi izleme donanımları gibi cihazlarda rutin dijital sinyal işlemcileri kullanılmaktadır. Bu tez çalışmasında Şekil 1.1'de gösterilen sistem blok şeması gerçekleştirilmektedir.



Şekil 1.1. Sistem blok şeması.

İçerisine dinamik mikrofon eklenerek oluşturulan steteskop mikrofondan gelen sinyaller USB Ses Adaptörü ile MATLAB GUI Arayüz tasarımına aktarılmaktadır. MATLAB GUI arayüzde steteskop mikrofondan gelen kalp üzerindeki gürültü referans sinyali ile soluk borusu üzerine yerleştirilen gürültülü soluk ses sinyalini ayrı ayrı zaman ekseninde ve frekans ekseninde gösterilmektedir. Bu ses sinyalleri adaptif filtrelemede kullanıldıktan sonra filtre çıkışında soluk borusu üzerine yerleştirilen steteskoptan kalp seslerinin arındırılmasıyla temiz bir solunum ses sinyaline ulaşılmaktadır. Bu temiz solunum ses sinyali üzerinde konuşma sınırlarını tespit etme algoritmasının çalıştırılması ile soluk alma ve soluk verme zaman aralıkları tespit edilebilmektedir. Tespit edilen bu zaman aralıklarının kullanılmasıyla solunum hızı elde edilmektedir.

Bu tez çalışması şu şekilde tanzim edilmiştir:

Bölüm 1’de çalışma konusu hakkında bilgi verildikten sonra literatürde benzer çalışmalar özetlenmiş ve sistem blok diyagramına yer verilmiştir. Bölüm 2’de insanlarda solunum sisteminin bölümleri ve solunum yolları bahsedilmiştir. Bölüm 3’te bu çalışmada kullanılan adaptif filtrelerin teknikleri, konuşma sinyalinde ses sınırı belirleme algoritması ve MATLAB ile GUI tasarımı anlatılmıştır. Bölüm 4 ‘te önerilen filtreleme tekniği 10 kişiye uygulanarak uygulamalı deneyler yapılmıştır. Bölüm 5’te projenin sonuçları ve gelecek çalışmalar tartışılmıştır.

1.1. LİTERATÜRDE YAPILAN BENZER ÇALIŞMALAR

2005 yılında Keenan ve Wilhelm, Solunum dışı sensör hareketi tarafından oluşturulan artefakt aralığının ayrıştırılmasını sağlayan ve solunum bileşeninin alınmasını sağlayan sinyal işleme algoritmaları sunmuşlardır. İnsan solunum sinyallerinin daha iyi işlenmesi için bu tekniklerin performansını değerlendirdiler. Çalışmalarında, geleneksel alçak geçiren filtrelerin aksine, bir dizi dalgacık filtresine uygulanan yumuşak bir eşikleme tekniğinin, solunum bilgisini kaybetmeden gürültüyü sinyalden etkili bir şekilde azaltmıştır. Analizleri, geleneksel yüksek ve alçak geçiren filtrelerin, gürültüyü azaltmak için tasarlanmalarına rağmen, sinyaldeki gizli solunum bileşenlerini ciddi şekilde bozabileceğini göstermektedir. Buna karşılık, sunulan lineer ve lineer olmayan sinyal işleme algoritmaları, solunum dışı sensör hareketi tarafından üretilen birkaç kritik gürültünün ayrıştırılmasını sağlamaktadır [7].

Tiinanen vd. 2007 yılında, solunum etkisini Barorefleks duyarlılığının (BRS) tahminlerinden çıkarmak için adaptif bir LMS tabanlı filtre önermişlerdir. Sağlıklı erkeklerde 0.25 Hz'de 5 dakika spontan ve 5 dakika kontrollü solunum sırasında EKG, sürekli kan basıncı ve solunumu ölçmüşlerdir. BRS, filtrelemeli ve filtrelemesiz spektral yöntemlerle hesaplanmıştır. Sonuç olarak, adaptif filtre, solunumun BRS değerleri üzerindeki bozucu etkisini azaltır ve bu da BRS'nin daha doğru tahmin edilmesini sağlamaktadır [8].

Correa vd. 2007 yılındaki çalışmalarında, en küçük ortalama kareler (LMS) algoritmasına dayanan üç adaptif filtre yöntemi önermişlerdir. Birincisi hat girişimini ortadan kaldırır, ikinci adaptif filtre EKG artefaktlarını ortadan kaldırır ve sonuncusu EOG artışlarını iptal etmektedir. Her aşama, EEG'de bulunan artefaktlara benzer bir çıktı üretmek için katsayılarını ayarlayan bir sonlu dürtü yanıtı (FIR) filtresi kullanmaktadır. Önerilen yöntem, polisomnografik çalışmalarda elde edilen beş gerçek EEG kaydında test edilmiştir. Tüm kayıtlarda, hat frekansı, EKG ve EOG artefaktları zayıflatılmıştır. Önerilen filtrenin, bu kayıtlarda yer alan önemli bilgileri kaldırmadan EEG sinyallerinde bulunan artefaktları azalttığı sonucuna varılmıştır [9].

Sancar vd. 2010 yılında, çeşitli adaptif filtre algoritmaları kullanarak sinüzoidal 0.18 Hz solunum sinyalinin hareket artefaktlarını ve 50 Hz Güç kablosunun parazitini filtreleme üzerinde çalışmışlardır. Çalışmalarında, MSE algoritmasının performansının LMS ile karşılaştırıldığında daha iyi olduğu sonucuna varılmıştır. LMS algoritması daha az bellek ve işleme kapasitesi tüketmektedir. Performansın dikkate alınan numune sayısına bağlı olduğunu bulmuşlardır. Bu nedenle bir algoritma seçmek, sistemin çok fazla ilgi duyduğu parametreye bağlıdır [10].

2011 yılında Abdullah ve Bolz, adaptif filtrenin fotopleitizmografi (PPG) ile kullanımını tartışmışlardır, bu sinyal, nabız geçiş süresi (PPT), kan basıncı izleme ve nabız hızı değişkenliği gibi bazı parametreleri algıladığı için önemli bir sinyaldir, bu nedenle kardiyovasküler hastalıkların risk tahmini ve teşhisi için kullanılmaktadır. Ayrıca konsantrasyon, kan oksijen doygunluğu ve diğer kan bileşenlerinin hesaplanması da PPG sinyalinin analizini gerektirebilir. PPG sinyalinin AC bileşeni, teşhis için önemli bilgiler içermektedir, ancak aynı bant genişliğini paylaşan gürültüden etkilenebilir. Abdullah ve Bolz, Kalman Filtresi, optimizasyon algoritmalı Neuro-fuzzy ve dalgacık dönüşümünün bazı durumlarda biyo sinyallerin adaptif filtrenmesi için iyi bir seçenek olabileceğini bulmuşlardır. Acil, yoğun bakım ve evde bakım (non-invaziv ölçümün uygulandığı yerlerde) için adaptif filtre kullanımı büyük önem taşımaktadır. Ayrıca hasta ve tıbbi bakım personeli için rahatsızlıkları azaltacak, maliyetleri düşürecek ve tıbbi sistemleri iyileştirecektir [11].

Cassani vd. 2011 yılında, kalp atış hızı değişkenliği (HRV) sinyal filtreleme için uyarlanabilir bir yöntem tanımlamışlardır. Bu yöntem, HRV bilgileri üzerindeki solunum etkisini azaltmak için en küçük ortalama kareler (LMS) algoritmasıyla birlikte bir sonlu dürtü yanıtı (FIR) filtresi tarafından oluşturulan bir gürültü önleyici kullanılmaktadır. 5 dakikalık deneyler sırasında 240 Hz örnekleme frekansı kullanılarak aynı anda solunum ve elektrokardiyogram (EKG) sinyalleri elde etmişlerdir. Solunum sinyali mekanik yöntemlerle, EKG sinyali ise tek elektrotlu elektrokardiyograf kullanılarak elde edilmiştir. Veri toplamadan sonra, HRV sinyalini elde etmek için EKG ölçümünden bir takogram elde edilmiştir. Daha sonra adaptif gürültü önleme (ANC) filtrelemesi uygulanarak HRV sinyalinden solunumdan kaynaklanan artefaktlar azaltılmıştır. Cassani ve arkadaşları, filtrelemeden önce ve sonra HRV

sinyalinin güç spektral yoğunluğu (PSD) sonuçlarını karşılaştırarak solunum frekansını kontrol etmişlerdir. Sonuçlarda HRV sinyalinin PSD'sinde solunumla ilgili frekans bileşenlerinin iptal edildiği gözlemlenmiştir [12].

Pandey vd. 2011 yılında, solunum indekslerinin tahminini iyileştirmek için solunum artefaktının baskılanması için LMS tabanlı bir adaptif filtreleme tekniği sunmuşlardır. Önerilen teknik bir referans sinyali kullanır, bu referans sinyali bir kardiyogram tarafından elde edilmektedir. Teknik, simüle edilmiş artefaktlı sinyaller ve dokuz sağlıklı denek ve kardiyovasküler bozukluğu olan beş hastadan alınan sinyaller üzerinde test edilmiştir. Deneyle sonuçları, Doppler ekokardiyografiden elde edilen sonuçlarla iyi bir uyum göstermiştir [13].

Foussier vd. 2014 yılında, sinyal ofsetlerini, solunum ve kalp sinyallerini üç farklı sensör kanalından ayıran adaptif kalman filtre prensibine dayalı gerçek zamanlı bir filtreleme sistemi sunmuşlardır. Sistem sürekli olarak solunum ve kalp atış hızlarını ölçmektedir, bu hızlar performansı artırmak için sistem modeline geri beslenmektedir. Sensör ve sistem gürültü kovaryans matrisleri, hedeflenen uygulamaya otomatik olarak uyarlanır, böylece sinyal ayırma özellikleri iyileştirilir. Filtreleri farklı kalp atış hızlarına ve farklı sensör özellikleri olan iki farklı örneğe uyguladılar ve sonuçları aynı filtrenin adaptif olmayan versiyonuyla karşılaştırmışlardır. Sonuç olarak, adaptif kalman filtresi prensibini kullanarak gerçek zamanlı kalp ve solunum hızlarını ölçmek mümkündür. Kalman filtresinin kullanılması sonuçları iyileştirir ve kardiyorespiratuar sinyalleri ölçerken filtreyi evrensel olarak kullanılabilir hale getirmektedir [14].

Zhang vd. 2014 yılında giyilebilir sağlık izleme sistemleri (WHMS) aracılığıyla elde edilen fizyolojik sinyallerde hareket artefaktını azaltmak için adaptif filtrenin kullanımını tartışmışlardır. Çalışmada, bisiklet ergometri egzersizleri sırasında EKG, solunum ve üç eksenli ivmeölçer (ACC) sinyallerini elde etmek için bir (WHMS) kullanmışlardır. Bu sinyallerle, adaptif hareket artefaktını filtrelemesinin performansları hem solunum hem de EKG sinyallerinde test edilmiştir. Hareket artefaktını tahmin etmek için gradyan adaptif laguerre kafesi (GALL) filtreleri kullanmışlardır. GALL filtrelerinin çıktıları ayrıca Kalman filtresi tarafından belirlenen zamanla değişen ağırlıklarla birleştirilmiş. Sonuçlar, solunum sinyalleri için

hareket artefaktın bileşeninin azaltılabileceğini ve sinyal kalitesinin etkin bir şekilde iyileştirilebileceğini göstermektedir. GALL ve Kalman filtrelerinin kombinasyonu, ACC'den referans ekseninin denetimli seçimi olmadan sağlam hareket artefaktın filtrasyonu sağlayabilmektedir [15].

Komorowski vd. 2015 yılında bir referans sinyali kullanan bir adaptif filtreleme yöntemi önermişlerdir. Araştırmalarında, solunum bileşeni standart olmayan EKG (NSECG) sinyalinden elde edilmektedir. Adaptif filtreleme, ayırık kosinüs dönüşüm alanında yapılmaktadır. Daha sonra, yeniden örneklenen elektrogastrografik sinyaller ile küçük solunum bileşenleri, alçak geçiren filtre ile filtrelenmektedir. Bunun sonucunda, EGG sinyali ve sindirim sisteminin iç organlarından gelen sinyal bileşenleri içeren genişletilmiş elektrogastrografik sinyaller elde edilmektedir. Sunulan yöntem, kardiyovasküler, sindirim, solunum veya sinir sistemleri gibi insan vücudundaki farklı sistemler arasındaki etkileşimlerin incelenmesi için umut verici bir teknik göstermektedir. Çalışmalarında, herhangi bir ek sensör veya cihaz olmadan EGG, ECG (HRV) gibi sinyalleri ve solunum sinyalini alma imkânı göstermişlerdir. Ayrıca, bu sinyaller, kayıt cihazındaki sinyalin örnekleme frekansının biraz artırılabilmesi şartıyla, yalnızca yazılım tarafından hesaplanabilmektedir. Komorowski ve arkadaşlarının yöntemi, elektrogastrografik sinyalde QRS kompleksleri altındaki alanın analizi ile elde edilen solunum sinyallerinin etkili bir şekilde yeniden oluşturmasını göstermektedir [16].

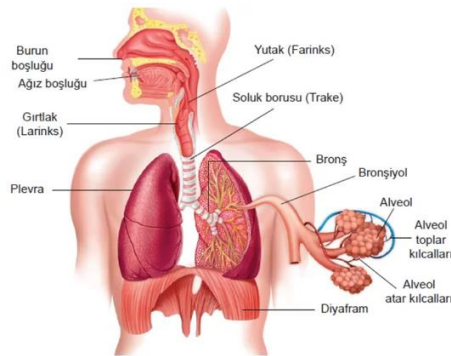
Wu vd. 2020 yılında çalışmalarında trakeal seslerin kalitesini iyileştirmek ve trakeal sesleri kullanarak Apne algılama algoritmasının doğruluğunu incelemek için adaptif filtreleme (AF) algoritmasını kullanmışlardır. Plastik bir parça içine yerleştirilmiş bir mikrofon kullanılarak trakeal sesler elde edilmiştir. Ortam sesleri, sırasıyla sessiz ve gürültülü ortamlarda plastik kasanın dışında duran bir referans mikrofon kullanılarak elde edilmiştir. Apne olaylarını belirlemek için standart sinyaller olarak akış basıncı sinyalleri ve torasik hareket elde edilmiştir. Daha sonra gürültülerle karışan trakea seslerine NLMS algoritması uygulanmıştır. Son olarak, AF ile filtrelenen trakea seslerine apne saptama algoritması uygulanmıştır. Sonuç olarak gürültülü ortamlarda apne tespiti için NLMS AF algoritmasının kullanılması doğru ve güvenilir olduğu kanıtlanmıştır [17].

BÖLÜM 2

SOLUNUM SİSTEMİ

Solunum sistemi, günlük faaliyetlerimizde en çok farkında olduğumuz vücut sistemidir. Zor egzersizler, soğuk algınlığı, havasız odalar ve soğuk sabahlar, solunum sisteminin ve sürekli işleyişinin önemini farkına varmamızı sağlamaktadır. İnsan vücudu, yaşamı desteklemek için gerekli bazı malzemeler (yemek ve su gibi) olmadan uzun süre hayatta kalabilmesine rağmen, oksijen olmadan birkaç dakikadan fazla yaşayamazdır. Oksijen kaynağından mahrum kalan vücut hücreleri yakında ölmeye başlayacaktır. İnsan vücudunda hücrelere oksijen sağlamaktan sorumlu iki sistem vardır: dolaşım sistemi ve solunum sistemidir [18].

Solunum sistemi iki ana bölüme oluşur: iletken organlar ve gaz değişim organlarıdır. Esasen, iletken organlar (üst yol), alt bölgeye hava akışına izin vermek için dış ortama açılan bir tüp oluşturmaktadır. Bu tüp burun boşluğu, farenks, gırtlak ve soluk borusunu içermektedir. Gaz değişimi akciğerlerde gerçekleşmektedir. Alt bölgede dallanmış solunum yolları alveollere ulaşmaktadır. Alveoller, solunum sisteminin fonksiyonel birimleridir. Akciğerlerle ilişkili yapılar da solunum sürecinde önemlidir: plevra, göğüs duvarı, göğüs kafesi, diyafram ve solunumun mekanik sürecine yardımcı olan kaslardır [19]. Şekil 2.1'de solunum sistemi gösterilmektedir.



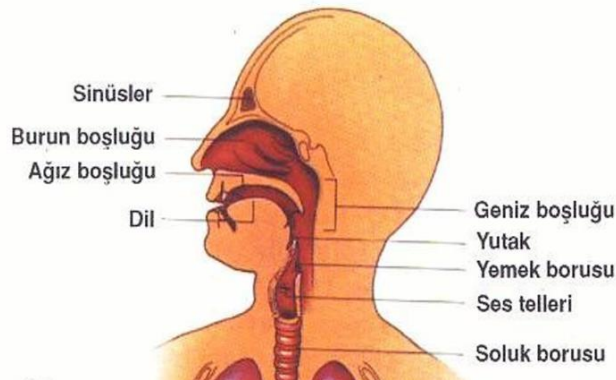
Şekil 2.1. Solunum sistemi [20].

Solunum döngüsü, inspirasyon (nefes alma) ile başlamaktadır. Daha sonra oksijen akciğerlerden kan dolaşımına geçer ve karbondioksit kan dolaşımından akciğerlere geçmektedir. Karbondioksit, normal hücre işleyişinin (metabolizma) bir atık ürünüdür ve vücuttan ekspirasyon (nefes verme) ile atılmalıdır. Aslında, kandaki karbondioksit konsantrasyonu belirli bir seviyeye ulaştığında solunum tetiklenmektedir. İnsan vücudu, karbondioksiti atmak ve oksijeni almak için nefes almaktadır. İspirasyon-ekspirasyon döngüsü (solunum hızı) yetişkinlerde dakikada yaklaşık 10-14 kez ve çocuklarda dakikada yaklaşık 20 kez tetiklenmektedir. Solunum hızı büyük ölçüde vücut ihtiyaçlarına bağlıdır. Vücudun aktivitesi arttıkça oksijen kullanımını ve karbondioksit üretimi de artmaktadır. Tetikleyici karbondioksit seviyesine daha hızlı ulaşıldığında, solunum daha hızlı ve daha derin hale gelmektedir [21].

Solunum sistemi dış ortama (burun ve ağız) açık olduğu için havadaki parçacıklara ve bakterilere karşı hassastır. Bakteri ve mikroorganizmalar havanın her nefesi ile alınmaktadır. Solunum yolu tarafından üretilen salgılar, parçacıkları ve kirleticileri yakalayıp içine alır ve hatta bakterileri yok eden enzimler içermektedir. Bu şekilde solunum sistemi, vücuda gerekli oksijeni sağlama işlemini gerçekleştirirken, vücudu zararlı parçacıklara sürekli maruz kalmaktan korumaktadır [22].

2.1. SOLUNUM YOLLARI

Şekil 2.2, solunum sisteminin üst solunum yollarını göstermektedir.



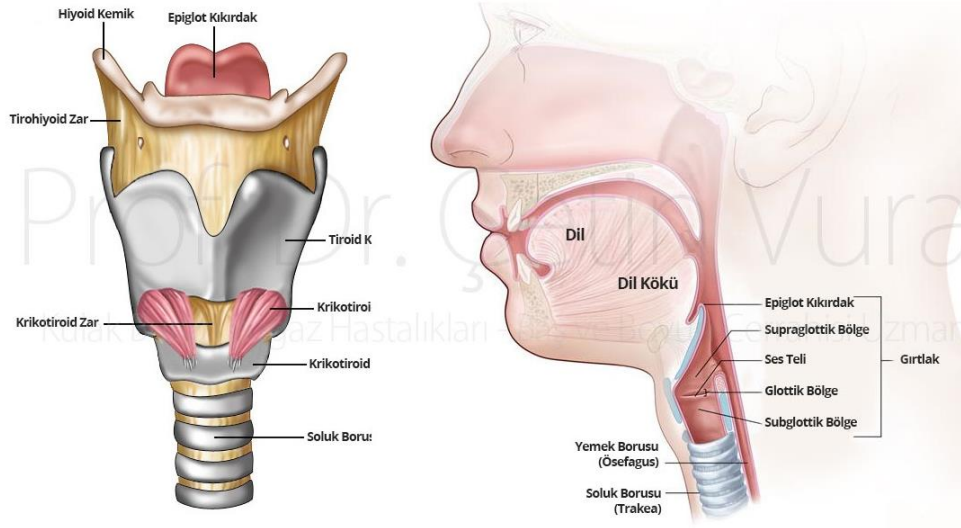
Şekil 2.2. Üst solunum yolları [23].

2.1.1. Yutak (Farenks)

Farenks yutak demektir, burun boşluklarının arkasından gırtlığa kadar uzanan bir boşluktur. Hem sindirim hem de solunum sistemi için ortak bir alan olduğu için farenks, mukosilyer klirens tarafından kendisine gönderilen parçacık-dolu mukusun atılması için idealdir. Yutak, burundan gelen solunan havayı ısıtma ve nemlendirme görevini tamamlamaktadır [18].

2.1.2. Gırtlak (Larnks)

Gırtlak, sağlam kıkırdak halkalarından oluşan sert bir yapıdır, ses tellerini korumaktadır. İnsan ses tellerini kullanarak konuşabilmektedir. Ses tellerinin titreşme ve ses çıkarma özelliği vardır. Ayrıca ses telleri de soluk borusuna giden yolu korumaktadır. Ses telleri arasında, soluk borusuna giden glottis adlı bir yarık bulunmaktadır. Yemek borusuna yönelik herhangi bir şey (yiyecek veya su gibi) yanlışlıkla gırtlığa girerse, glottis alt solunum yolunu kapatmaktadır. Bu şekilde gırtlak solunum yolunu hava dışında herhangi bir şeyi almaktan korumaktadır [18]. Şekil 2.3 Gırtlak anatomisi göstermektedir.



Şekil 2.3. Gırtlak anatomisi [24].

2.1.3. Soluk Borusu (Trakea)

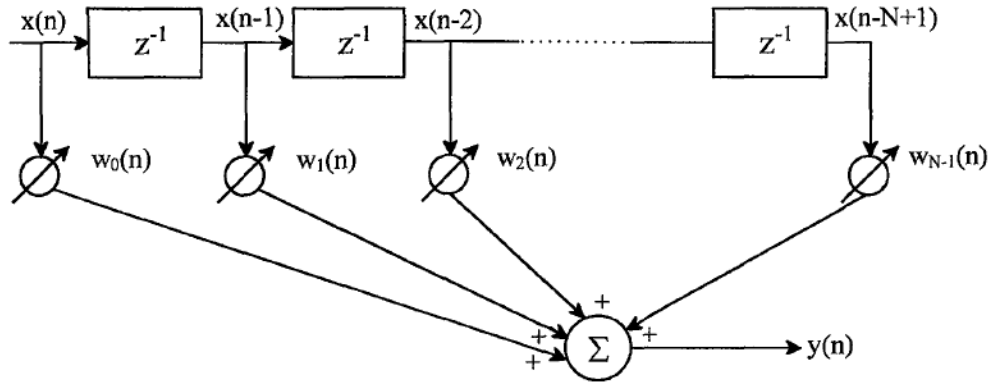
Trakea veya nefes borusu, üst solunum yolunun son kısmıdır. Yaklaşık beş inç uzunluğunda ve bir inç çapında, yutak ve akciğerleri birbirine bağlayan kısa bir tüptür. Trakea, birbiri üzerine yığılmış yaklaşık 20 C-şekilli kıkırdak halkasından oluşmaktadır. Kıkırdak, solunum yolunu açık tutmak için sertlik ve koruma sağlamaktadır. C-şeklindeki kıkırdak sert olmasına rağmen soluk borusunun arka kısmı yumuşaktır. Bu özellik, yemek geçerken yemek borusunun genişlemesini sağlamaktadır. Trakea üst solunum yolunun sonudur. Akciğerlere ve alveollere ulaşmaktadır [18].

BÖLÜM 3

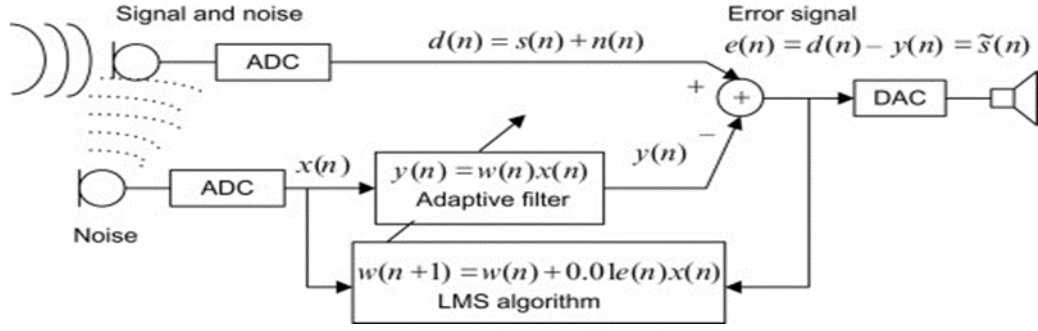
MATERYAL VE YÖNTEMLER

3.1. ADAPTİF FİLTRELER

Bir sistem, sistemin ve çevresinin durumuna bağlı olarak tanımlanmış bir amaca veya hedefe ulaşmak için parametrelerini değiştirebiliyorsa, uyarlanabilir sistem sayılmaktadır. Bu nedenle sistem, çevresindeki bazı olaylara yanıt verecek şekilde kendini ayarlamaktadır. Adaptif filtreler, başlangıç giriş istatistiklerini analiz eden ve zaman içindeki değişimlerini izleyen algoritmayı kullanan filtrelerdir. Bu filtreler deterministik sinyali tahmin eder ve onunla ilgisi olmayan gürültüyü ortadan kaldırmaktadır [25]. Adaptif filtreleme prensibi Şekil 3.1'de gösterilmiştir. Şekil 3.2, adaptif gürültü engelleme filtreleme prensibi göstermektedir.



Şekil 3.1. FIR filtre yapısı [26].



Şekil 3.2. Adaptif gürültü yok etme filtre yapısı [27].

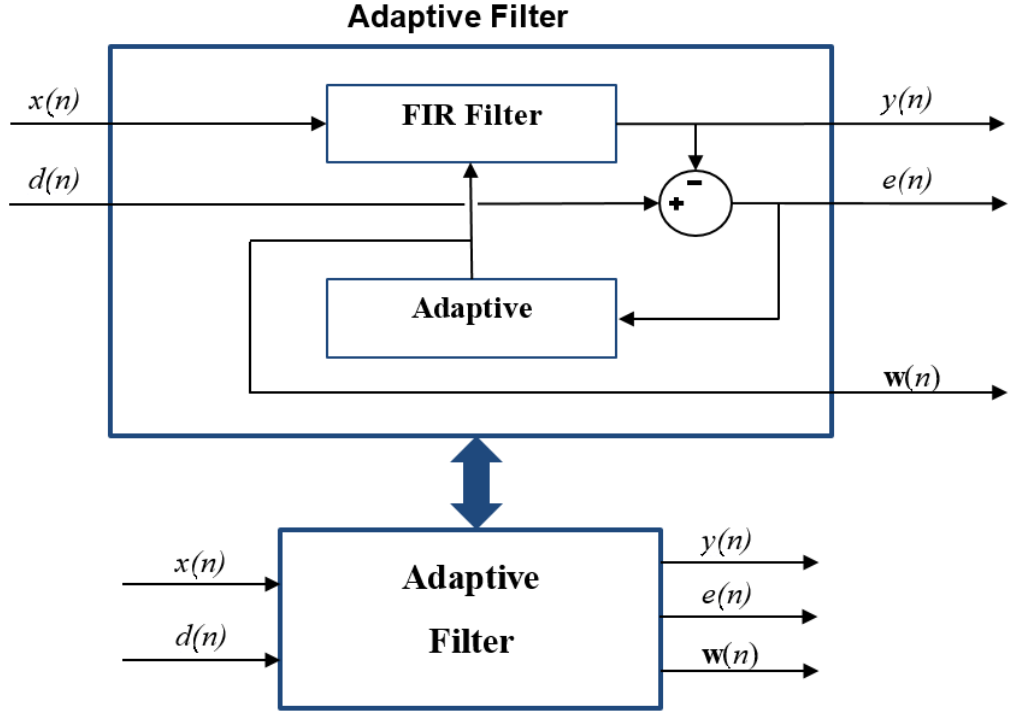
Sensörden elde edilen $d(n)$ sinyali, sadece istenen sinyali $s(n)$ değil, aynı zamanda istenmeyen gürültü sinyali $n(n)$ de içermektedir. Bu nedenle, sensörden ölçülen sinyal gürültü $n(n)$ tarafından bozulmaktadır. O zaman, eğer istenmeyen gürültü sinyali $n(n)$ biliniyorsa, istenen sinyal $s(n)$, bozuk sinyal $d(n)$ 'den gürültü sinyali $n(n)$ çıkarılarak elde edilebilmektedir. Tam gürültü kaynağının elde edilmesi zor olduğundan, nominal gürültü sinyali $n(n)$ kullanılmaktadır. Nominal gürültü sinyali $n(n)$, bazı filtreler ve ölçülebilir bir gürültü kaynağı $X(n)$ tarafından hesaplanmaktadır. Gürültü kaynağı $X(n)$, gürültü sinyali $n(n)$ ile lineer olarak ilişkilidir. Böylece, nominal sinyal $n(n)$ ve gürültülü olan sinyal $d(n)$ kullanılarak istenen sinyal $s(n)$ elde edilebilmektedir. Aktif gürültü filtreleme teorisinde bir adaptif filtre kullanılmaktadır. Adaptif filtre, iki bölüme ayrılmaktadır: adaptif algoritma ve dijital filtredir. Adaptif algoritmanın işlevi, uygun filtre katsayısını yapmaktır. Genel dijital filtrelerde sabit parametreler kullanılmaktadır. Ancak adaptif filtre, giriş sinyalinin, çevrenin ve çıkış sinyalinin özelliklerini dikkate alarak filtre parametrelerini değiştirmektedir [28].

FIR filtre çıkışı $y(n)$, istenen sinyal $d(n)$ ile karşılaştırılır ve ortaya çıkan hata sinyali $e(n)$, daha sonra FIR filtre ağırlık düzeltmelerinin vektörünü hesaplayan adaptasyon algoritmasına geçirilir ve bu filtre katsayılarını güncellemektedir. Adaptasyon algoritması, filtre ağırlıklarını $y(n), x(n)$ ile ilişkili olan $d(n)$ bileşeninin en iyi tahmin edecek şekilde öngörmektedir [28].

Adaptif bir filtre içindeki veri akışı, bir geri besleme kontrol döngüsüne benzemektedir. Bir adaptasyon sistemi $y(n)$ çıkışını, bir hata sinyali $e(n)$ oluşturmak

için bir referans sinyal $d(n)$ ile karşılaştırılmaktadır. Algoritmanın performansı daha sonra hata sinyalinden belirlenir ve adaptasyon algoritması, daha iyi bir çıktı sağlamak için sistemin nasıl değiştirileceğini belirtmektedir [28, 29].

$x(n)$ ve $d(n)$ girişlerinin kaynakları ve çıktılar $y(n)$, $e(n)$ ve ağırlık vektörü $w(n)$ 'nin anlamları adaptif filtre uygulamalarına göre değişmektedir. Sinyal işleme literatüründe birçok farklı tipte adaptif filtre uygulaması mevcuttur, ancak bunların çoğu Şekil 3.3'teki genel blok diyagram ile gösterilebilmektedir. Adaptif filtre türleri arasındaki temel fark, hata sinyalinin özelliklerine yanıt olarak bir filtre ağırlık düzeltme vektörü oluşturan blok "Adaptasyon Algoritması" 'dır [29] .



Şekil 3.3. Adaptif filtre genel blok diyagramı [30].

En Küçük Ortalama Kareler (LMS) yöntemi ile güncellenen doğrusal adaptif bir filtre için LMS adaptasyon algoritması, n mevcut sinyal indeksi, N FIR filtre uzunluğu, μ adım boyutu parametresi, T devrik operatörü ve $*$ karmaşık eşlenik operatörü olmak üzere aşağıdaki eşitlik grubu ile özetlenmiştir [31].

$$x(n) = [x(n), x(n-1), x(n-2), \dots, x(n-N+1)]^T \quad (3.1)$$

$$w(n) = [w_0(n), w_1(n), w_2(n), \dots, w_{N-1}(n)]^T \quad (3.2)$$

$$y(n) = w^T(n-1) \cdot x(n) \quad (3.3)$$

$$e(n) = d(n) - y(n) \quad (3.4)$$

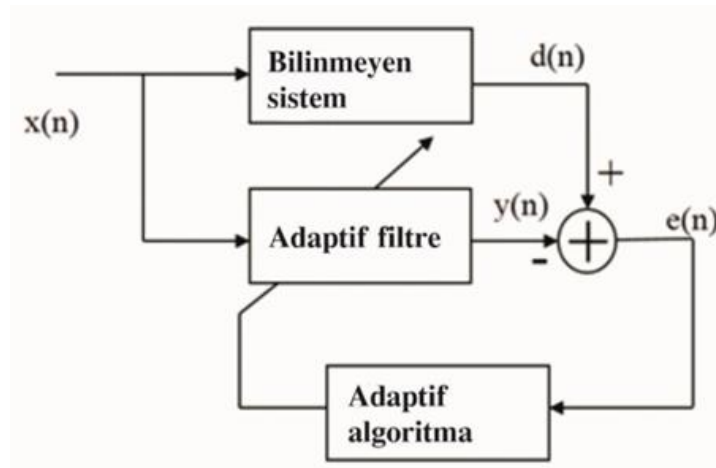
$$w(n) = w(n-1) + \mu e(n) \cdot x^*(n) \quad (3.5)$$

3.1.1. Adaptif Filtre Uygulamaları

Adaptif filtreleme uygulamaları genellikle dört gruba ayrılmaktadır [32].

3.1.1.1. Sistem Tanımlaması

Bilinmeyen bir sistem, bir giriş sinyali ile uyarılmaktadır. Adaptif filtre, bilinmeyen sistemin giriş ve çıkışını karşılaştırır ve frekans yanıtı bilinmeyen sisteminkine yakın olan bir sisteme yakınlaşmaya çalışmaktadır. Sistem tanımlaması için yapılandırılan adaptif bir filtre, bir gezegenin içi gibi sabit bilinmeyen sistemleri veya bir iletim hattı gibi dinamik sistemleri modellemek için kullanılabilir [32]. Adaptif filtre ile sistem tanımlama örneği Şekil 3.4'te gösterilmiştir.



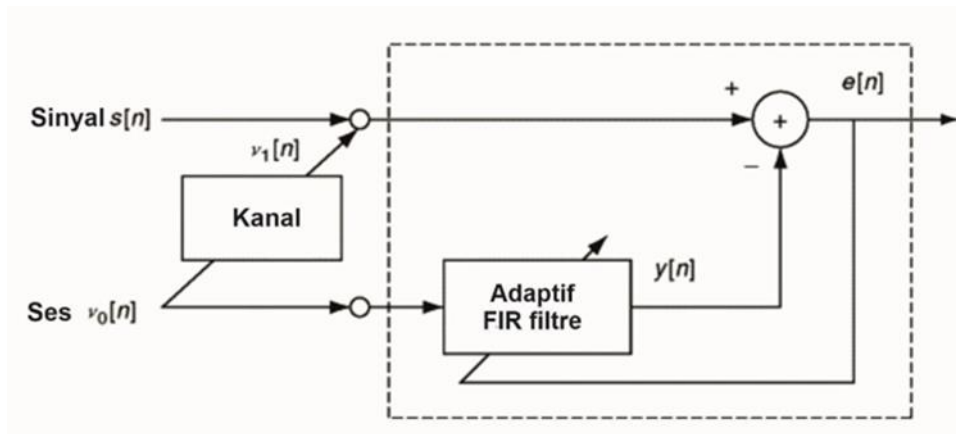
Şekil 3.4. Sistem tanımlaması için adaptif filtre [33].

3.1.1.2. Ters Sistem Tanımlaması

Bilinmeyen sistem giriş ve çıkışının önceki açıklamada rolleri tersine çevrilmektedir, daha sonra adaptif filtre, frekans yanıtı bilinmeyen sistemin tersi olan bir sistem için çözmektedir. Bu mimari bazen bilinmeyen bir sistemin tersinin bilinmeyen sistemin kendisinden daha kolay temsil edilebildiği durumlarda kullanılmaktadır. Ters sistem tanımlama adaptif filtreler, yankı iptal algoritmalarında görünmektedir, bu algoritmalarda bilinmeyen sistem en iyi şekilde tüm kutuplu bir IIR filtresi ile modellenmektedir [32].

3.1.1.3. Gürültü Giderme

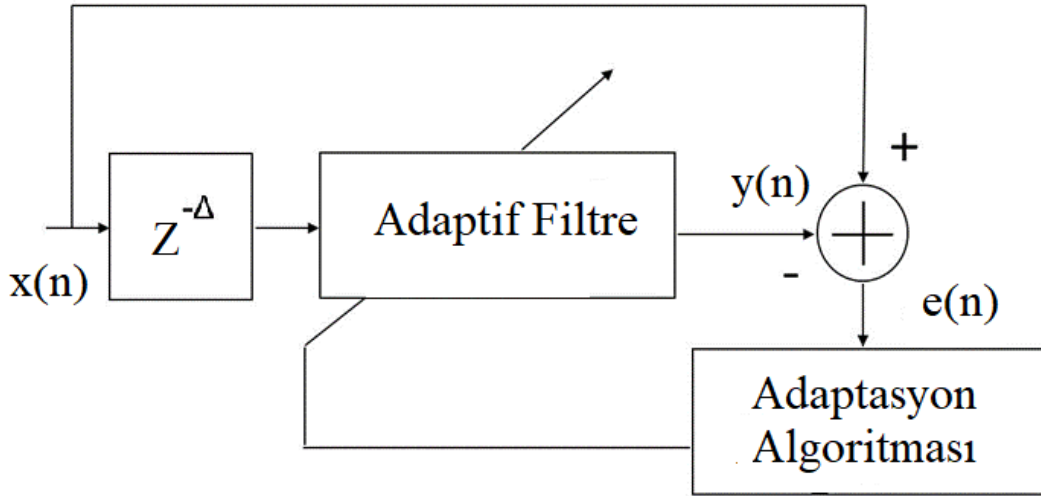
Gürültü ile bozulmuş bir sinyal, bir referans gürültü sinyali ile karşılaştırılmaktadır. Adaptif filtre, referans sinyali ile ilişkili olan gürültülü sinyal bileşenini çözer ve normal sinyalin bir tahminini vermektedir. Gürültü önleyici kulaklıklar ve cep telefonları gibi tüketici elektroniklerde birçok gürültü engelleme adaptif filtre örneği bulunmaktadır. Bu teknik, gürültülü bir ortamda elde edilen verilerin kalitesini iyileştirmek için de yaygındır [32]. Gürültü yok etme teknikleri görüntü işlemede kullanılabilir, örneğin teleskoplarda ve tıbbi görüntülemelerde kullanılmaktadır. Bir gürültü yok etme örneği, Şekil 3.5'te gösterilmiştir.



Şekil 3.5. Adaptif filtre ile gürültü önleme [34].

3.1.1.4. Lineer Tahmin

Lineer tahmin, bir sinyalin gelecekteki bir zamanda değerlerini tahmin etmektedir. Bu model, cep telefonlarında konuşma kodlama, konuşma geliştirme ve konuşma tanıma gibi konuşma işleme uygulamalarında yaygın olarak kullanılmaktadır. Bu yapılandırmada istenen sinyal, adaptif filtre giriş sinyalinin ileri versiyonudur. Lineer tahmin sistemi, Şekil 3.6'da gösterilmiştir [32].

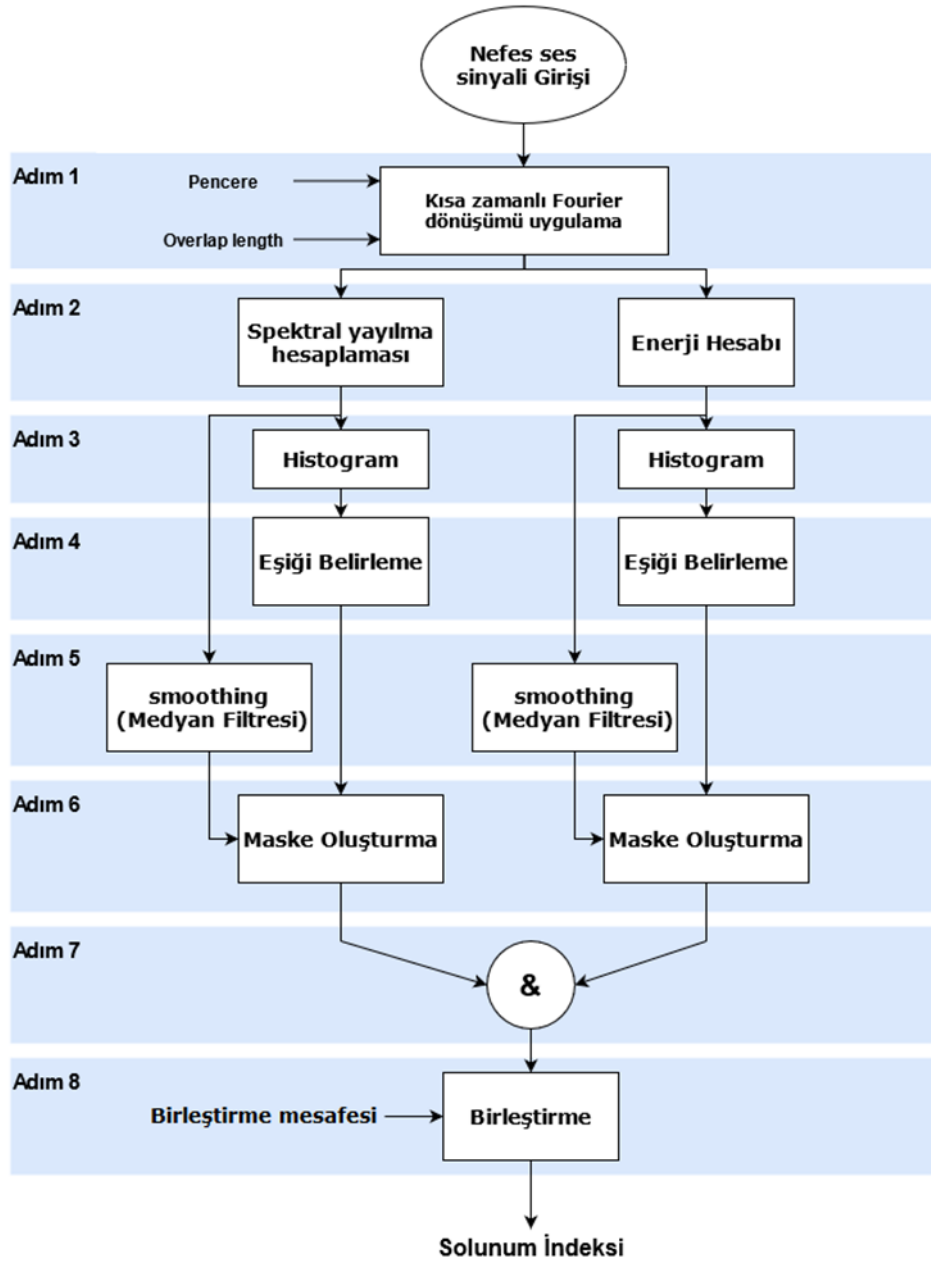


Şekil 3.6. Doğrusal tahmin için Adaptif filtre [35].

3.2. SES SİNYALİNDE KONUŞMA SINIRI TESPİT ETME ALGORİTMASI

Ses sinyalinde konuşma sınırı algılama algoritması Giannakopoulos algoritmasına [36] dayanmaktadır. Bu algoritma, eşik istatistiklerinin kısa vadeli enerji ve spektral merkez yerine kısa vadeli enerji ve spektral yayılma olacak şekilde değiştirilmektedir. Şekil 3.7, ses sinyalinde konuşma sınırı algılama algoritmasını göstermektedir.

Bu algoritma, bir ses sinyalindeki konuşma sinyalinin sınırlarını tespit etmektedir. Bu yöntem, ses sinyalindeki konuşmanın başlangıcına ve sonuna karşılık gelen endeksler vermektedir. Ayrıca ses girişini tek veya çift duyarlıklı tek kanallı sinyal olarak belirtmektedir. Endeksler, $L \times 2$ dizisi olarak döndürülür, burada L , algılanan bireysel konuşma bölgelerinin sayısıdır [37].



Şekil 3.7. Ses sinyalinde konuşma sınırı algılama algoritması.

Şekil 3.7'deki akış şemasından aşağıdaki adımlar anlaşılabilir:

İlk adımda, ses sinyali hızlı Fourier dönüşümü ve belirtilen pencere ve örtüşme uzunluğu kullanılarak bir zaman-frekans temsiline dönüştürülmektedir. İkinci olarak, her parça için kısa süreli enerji ve spektral yayılma hesaplanmaktadır. Spektral yayılım (spectralSpread) MATLAB fonksiyonu ile hesaplanmaktadır. Üçüncü adımda hem kısa süreli enerji hem de spektral yayılma dağılımları için histogramlar

oluşturulmaktadır. Dördüncü adımda, her histogram için Eşitlik 3.6'ya göre bir eşik belirlenmektedir [37]:

$$T = \frac{W \times M_1 + M_2}{W + 1} \quad (3.6)$$

Eşitlik 3.6'da M_1 ve M_2 sırasıyla birinci ve ikinci yerel maksimumlardır. Beşinci adımda hem spektral yayılım hem de kısa süreli enerji, beş-element hareketli medyan filtrelerinden geçirilerek zaman içinde düzleştirilmektedir. Altıncı adımda, kısa süreli enerji ve spektral yayılımın ilgili eşikleriyle karşılaştırılmasıyla maskeler oluşturulmaktadır [37].

Maskeler yedinci adımda birleştirilmektedir. Bir dalgacığın ses sinyalinde konuşma sınırı olarak bildirilmesi için hem kısa süreli enerji hem de spektral yayılma, ilgili eşiklerinin üzerinde olmalıdır. Son adımda, ses sinyalinde konuşma sınırı olarak bildirilen dalgacıklar, aralarındaki mesafe birleştirme mesafesinden küçükse birleştirilmektedir [37].

Ses sinyalinde konuşma sınırı, belirtilen pencere ve örtüşme uzunluğu kullanılarak bir kısa zaman-frekans dönüşümü (STFT) temsiline dönüştürülmektedir. Bir reel $f(t)$ fonksiyonunun STFT dönüşümü Eşitlik 3.7'de gösterilmiştir [38]:

$$\mathcal{G}\{f(t)\} = F(\Omega, \tau) = \int_{-\infty}^{\infty} f(t) \cdot g^*(t - \tau) \cdot e^{-i\Omega t} dt \quad (3.7)$$

Burada $g(t)$ integrallenebilen bir kare fonksiyondur ve * karmaşık eşleniği belirtmektedir. Eşitlik 3.7 ifadesi aşağıdaki gibi de yazılabilmektedir [38]:

$$\mathcal{G}\{f(t)\} = F(\Omega, \tau) = \langle f(t), g(t - \tau) \cdot e^{i\Omega t} \rangle \quad (3.8)$$

Burada $\langle \rangle$ operatörü iki fonksiyonun iç çarpımını gösterir. $g(t - \tau)$ ve $i\Omega t$ fonksiyonlar, problemin Gabor fonksiyonlarıdır. Fonksiyonlar kümesi, τ boyutundaki adımlarla t eksenini boyunca kaydırılan temel bir $g(t)$ fonksiyonundan türetilmektedir.

Bu fonksiyon kümesi, $g(t)$ uygun şekilde seçildiğinde, $f(t)$ fonksiyonunun ayrıştırılması için ortogonal baz oluşturmaktadır. $h(t) = g(t) \cdot e^{i\Omega t}$ çarpımının t değişkenine göre Fourier dönüşümü Eşitlik 3.9 ile verilmektedir [38]:

$$\begin{aligned} \mathcal{F}\{g(t) \cdot e^{i\Omega t}\} &= \hat{H}(\omega) = \int_{-\infty}^{\infty} g(t) \cdot e^{i\Omega t} \cdot e^{-i\omega t} dt = \int_{-\infty}^{\infty} g(t) \cdot e^{-i(\omega-\Omega)t} dt \\ &= \hat{G}(\omega - \Omega) \end{aligned} \quad (3.9)$$

Burada, $\hat{G}(\omega)$ $g(t)$ 'nin Fourier dönüşümüdür. Eşitlik 3.10 ifadesi, frekans alanında, $e^{i\Omega t}$ faktörünün de ötelemelere eşdeğer olduğunu göstermektedir. Zaman düzleminde, bu faktör $f(t)$ 'nin fazının $\theta = \Omega \cdot T$ açısı kadar kaymasına karşılık gelmektedir [38].

$$\mathcal{E}_{g(t)} = \int_{-\infty}^{\infty} |g(t)|^2 dt = 1 \quad (3.10)$$

Eşitlik 3.10, $\mathcal{E}_{g(t)}$ fonksiyonun enerjisidir. $g(t)$ fonksiyonu yerine getirilmiştir, bu nedenle Gabor fonksiyonları bir ortonormal (sadece ortogonal değil) baz oluşturur ve ters dönüşüm Eşitlik 3.11'de verilmektedir [38]:

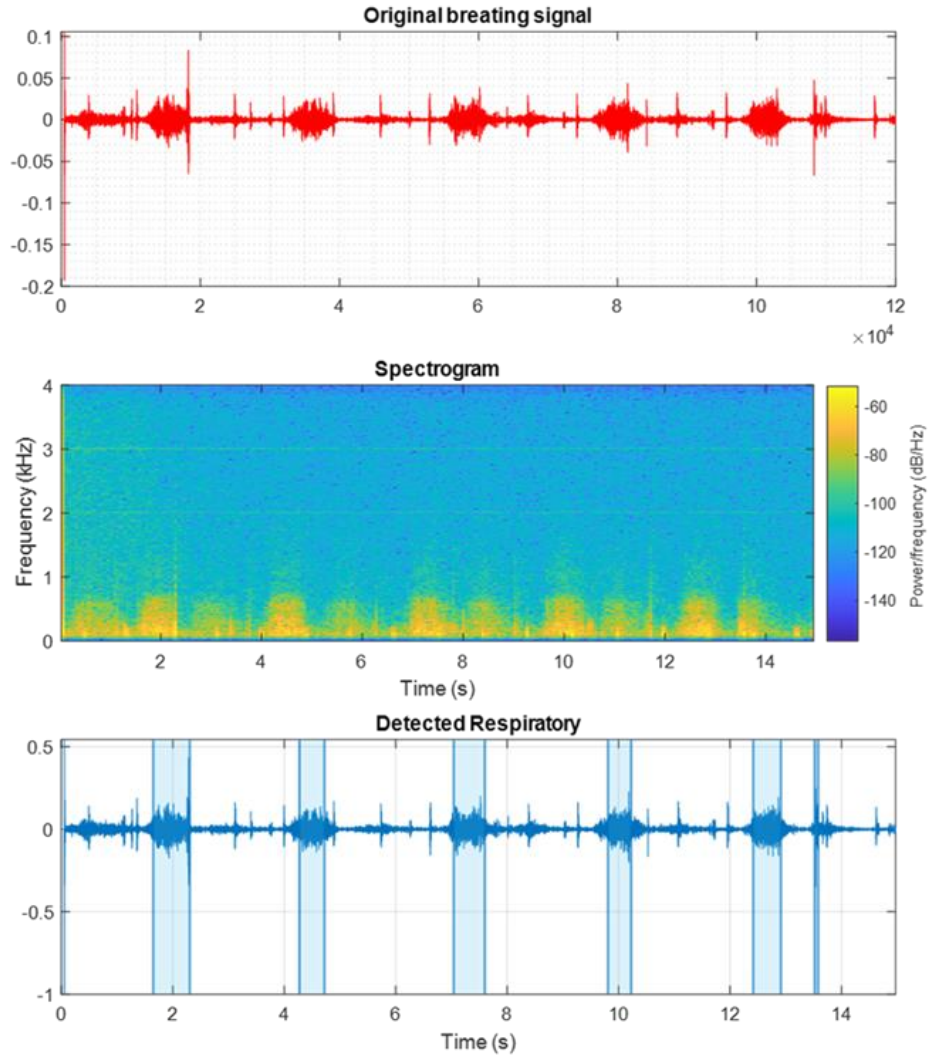
$$f(t) = \mathcal{G}^{-1}\{F(\Omega, \tau)\} = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(\Omega, \tau) \cdot g(t - \tau) \cdot e^{i\Omega t} d\Omega d\tau \quad (3.11)$$

Böylece Eşitlik 3.7 bir konvolüsyon integrali şeklini almaktadır. Sinyal analizinde konvolüsyon işleminin sinyalde bulanıklığa neden olduğu bilinmektedir. Bu durumda girişim, analiz edilen $f(t)$ sinyalinin spektrumları ile $g(t)$ fonksiyonu arasındaki karşılıklı bir girişim olarak tanımlanabilmektedir. Fourier dönüşümünün özelliklerine göre, herhangi bir $h(t)$ fonksiyonu için Eşitlik 3.12 yazılabilmektedir [38]:

$$\Delta t = \frac{\mathcal{M}_2\{|h(t)|^2\}}{\mathcal{M}_0\{|h(t)|^2\}} \Delta\omega = \frac{\mathcal{M}_2\{|\hat{H}(\omega)|^2\}}{\mathcal{M}_0\{|\hat{H}(\omega)|^2\}} \mathcal{M}_n\{h(x)\} = \int_{-\infty}^{\infty} x^n \cdot h(x) dx \quad (3.12)$$

Burada $M_n\{h(x)\}$ n'inci $h(x)$ zaman fonksiyonudur, Δt ve $\Delta\omega$ sırasıyla zaman ve frekans alanlarında taşıyıcı boyut aralığıdır. Eşitlik 3.12, pencerenin uygun τ boyutunu hesaplamamızı sağlamaktadır. Böylece $f(t)$ ve $g(t)$ 'nin spektrumları ayrık olur ve bulanıklığı ortadan kaldırmaktadır [38].

Orjinal sinyal ile spektrogram analizi, Şekil 3.8'de gösterilmiştir.



Şekil 3.8. Spektrogram analizi. a) orijinal solunum sinyali, b) spektrogram, c) solunum sinyalinin tespit edilmesi.

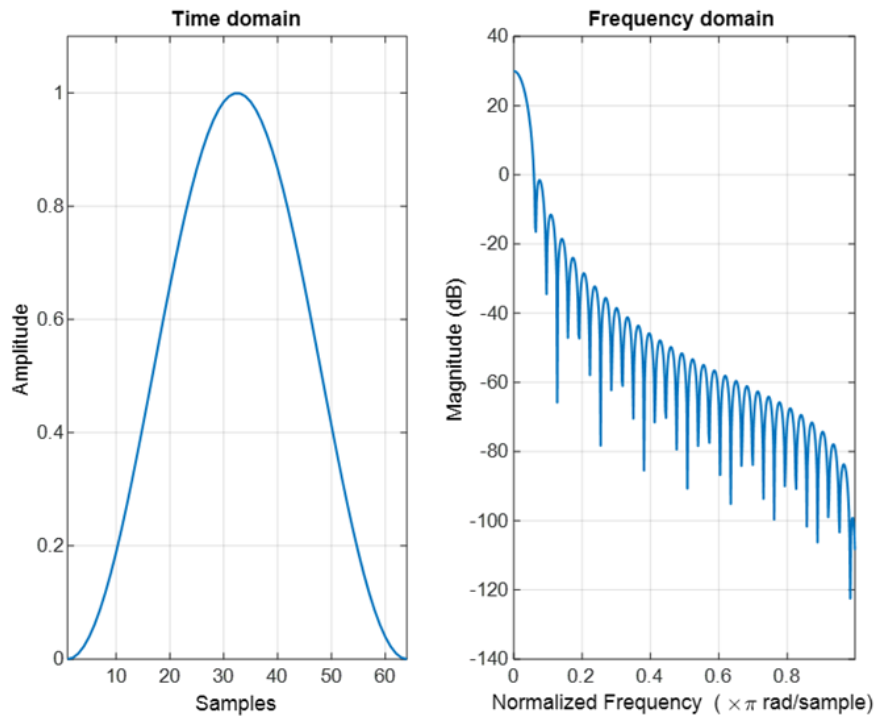
Yöntemde açıklanan spektrogram analizi kullanılarak solunum sinyallerinin çıkarılabileceği ve tespit edilebileceği Şekil 3.8'den anlaşılabilir.

Hanning pencereleri, sinyalin frekansının çıkarılması için kullanılmaktadır. Bu pencerenin şekli Şekil 3.9'da gösterilmiştir. Hanning penceresi, ağırlıklı bir kosinüs kullanılarak oluşturulan bir fonksiyondur. Hanning penceresine yapılan referansların çoğu, değerleri yumuşatmak için birçok pencereleme fonksiyonlarından biri olarak kullanıldığı sinyal işleme literatüründen gelmektedir. Ayrıca apodizasyon (örneklenmiş sinyalin başlangıcındaki ve sonundaki süreksizlikleri yumuşatma) fonksiyonu olarak da bilinmektedir. Bu çalışmada, Şekil 3.7'de gösterildiği gibi, 1. adımda solunum sinyaline Hanning penceresi uygulanmaktadır [39].

Aşağıdaki Eşitlik 3.13, bir Hanning penceresinin katsayılarını üretir [40]:

$$w(n) = 0.5 \left(1 - \cos \left(2\pi \frac{n}{N} \right) \right), \quad 0 \leq n \leq N \quad (3.13)$$

Pencerenin uzunluğu $L = N + 1$.



Şekil 3.9. Frekans alanında Hanning penceresi.

3.3. MATLAB GUI İLE TASARIM

MATLAB, birçok kullanıcı tarafından yalnızca teknik hesaplama için yüksek performanslı bir dil olarak değil, aynı zamanda grafiksel kullanıcı arayüzleri (GUI) oluşturmak için uygun bir ortam olarak görülmektedir. MATLAB'ta veri görselleştirme ve GUI tasarımı "Handle" Grafik Sistemine dayanmaktadır. Handle'de nesnelere bir grafik nesnesi olarak düzenlenir ve çeşitli yüksek ve düşük seviyeli komutlarla etkileşime girebilmektedir [41].

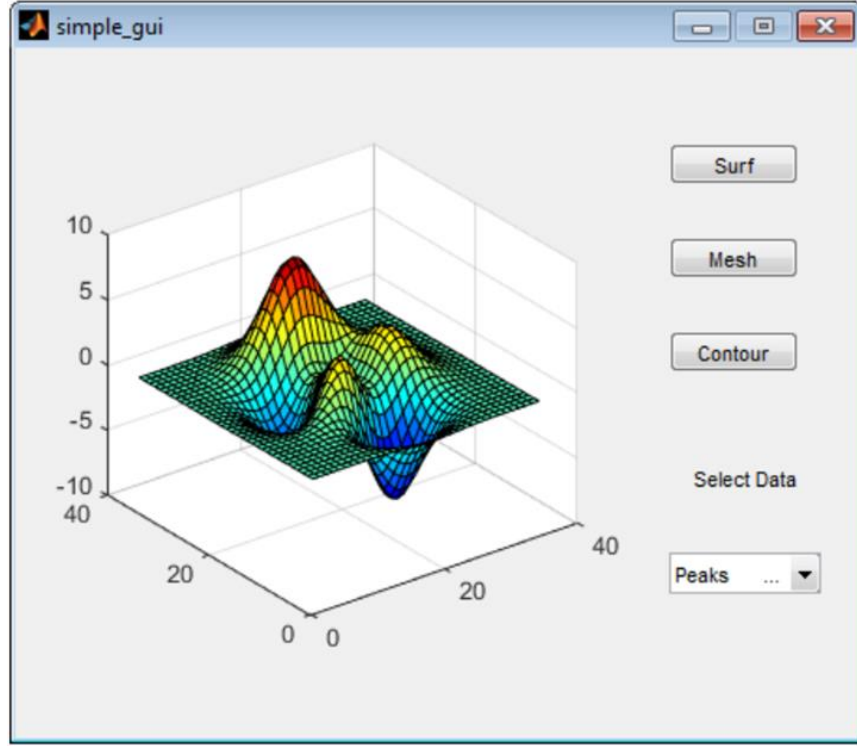
Grafik kullanıcı arayüzü (GUI), kontrolleri içeren bir ya da daha fazla pencereye sahip bir grafik ekrandır. Bu kontroller "bileşenler" olarak adlandırılır. Bileşenler kullanıcının etkileşimli görevleri gerçekleştirmesini sağlamaktadır. Kullanıcının görevleri gerçekleştirmek için bir komut dosyası oluşturması ya da komut satırına komut yazması gerekmez. Belirli görevleri gerçekleştirmek için kod yazmanın aksine, kullanıcının kodun tüm ayrıntılarını bilmesi gerekmemektedir [42].

Şekil 3.10'da gösterildiği gibi GUI bileşenleri menüler, araç çubukları, butonlar, listeler ve diğer nesnelere içerebilmektedir. Ayrıca MATLAB kullanılarak oluşturulan GUI'ler her türlü hesaplamayı gerçekleştirebilir, veri dosyalarını okuyabilir ve yazabilir, diğer GUI'lerle iletişim kurabilir ve verileri tablolar veya grafikler olarak görüntüleyebilmektedir [42].

Genellikle, GUI'ler kullanıcının bir nesneyle etkileşime girmesini bekler ve ardından sırayla her kullanıcı eylemine yanıt vermektedir. Her nesne ve GUI'nin kendisi bir veya daha fazla "Callback" a sahiptir. Bir ekran düğmesine basmak veya imleci bir bileşenin üzerine geçirmek gibi belirli bir kullanıcı eylemi, her "Callback" ın yürütülmesini tetiklemektedir. GUI daha sonra bu olaylara yanıt vermektedir. Kullanıcı, GUI kodlayıcısı olarak, bileşenlerin olayları işlemek için ne yaptığını tanımlayan callback'lar yazmaktadır [42].

Bu tür programlamaya genellikle olay-güdümlü programlama denir. MATLAB GUI'lerinde çoğu olay GUI ile kullanıcı etkileşimleridir, ancak GUI, bir dosyanın

oluřturulması veya bir aygıtın bilgisayara baėlanması gibi bařka tr olaylara da yanıt verebilmektedir [42].



řekil 3.10. MATLAB GUI bileřenleri [42].

Bir MATLAB GUI, programcının kullanıcı tarafından alıřtırılan bileřenleri eklediėi ekrandır. Kullanıcı bu bileřenlerin boyutu ve konumu istediėi gibi deėiřtirebilmektedir. Callback'leri kullanarak kullanıcı, nesnelere tıkladıėında veya nesnelere etkileřime girdiėinde bileřenlerin istediėini yapmasını saėlayabilmektedir [42].

MATLAB GUI'leri iki yntemle oluřturulabilmektedir:

1. "GUIDE" kullanarak GUI oluřturma:

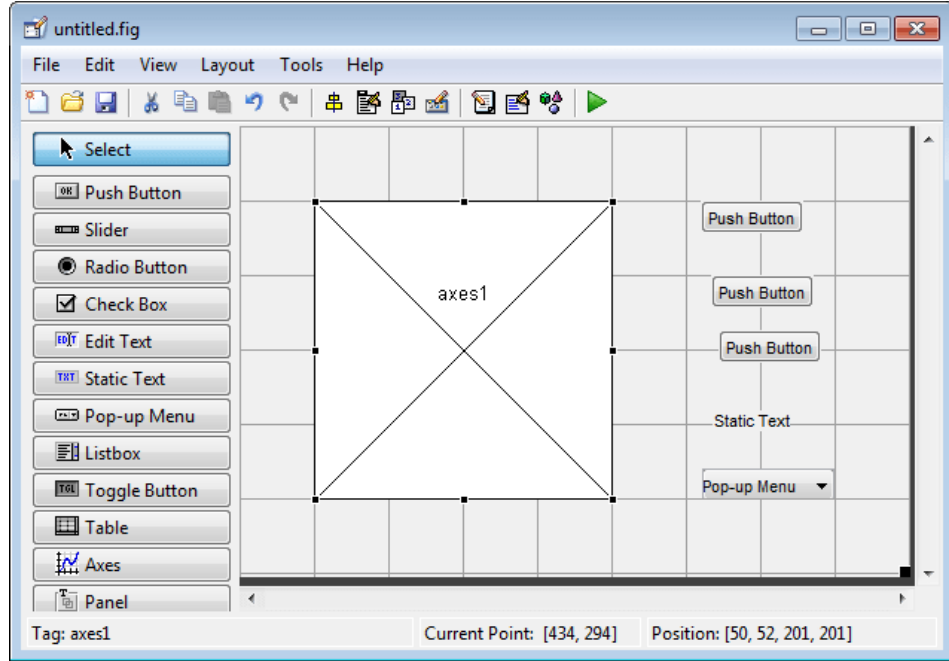
Bu yntem, kullanıcının menden bileřen eklediėi bir pencere ile bařlamaktadır. GUIDE, GUI ve bileřenleri iin callback'ları ieren iliřkili bir kod dosyası oluřturmaktadır. GUIDE hem řekli (FIG dosyası olarak) hem de kod dosyasını kaydetmektedir [42]. GUIDE editr řekil 3.11'de gsterilmiřtir.

2. GUI'yi programlı olarak oluşturma:

Bu yöntemi kullanarak kullanıcı, tüm bileşen özelliklerini ve davranışlarını tanımlayan bir kod dosyası oluşturmaktadır. Bir kullanıcı dosyayı çalıştırıldığında, bir şekil oluşturur, onu bileşenlerle doldurur ve kullanıcı etkileşimlerini beklemektedir [42]. Şekil 3.12'de programlı olarak kodlanmış bir GUI örneği gösterilmiştir.

İki yöntemin kod dosyaları farklı görünmektedir. Programlı yöntem GUI dosyaları genellikle daha uzundur, çünkü şeklin her özelliğini ve kontrollerini ve callback'leri açıkça tanımlarlar. GUIDE GUI, şeklin kendi içindeki özelliklerin çoğunu tanımlamaktadır. Tanımları kod dosyası yerine FIG dosyasında saklamaktadır. Kod dosyası, açıldığında GUI'yi başlatan geri aramaları ve diğer işlevleri içermektedir [42].

Kullanıcı GUIDE ile bir GUI oluşturabilir ve ardından bunu programlı yöntem kullanarak değiştirebilmektedir. Ancak, GUI programlı yöntem kullanılarak oluşturulmuşsa, kullanıcı programı GUIDE kullanarak değiştiremezdir [42].



Şekil 3.11. GUIDE editörü [42].


```

function simple_gui2
% SIMPLE_GUI2 Select a data set from the pop-up menu, then
% click one of the plot-type push buttons. Clicking the button
% plots the selected data in the axes.

% Create and then hide the UI as it is being constructed.
f = figure('Visible','off','Position',[360,500,450,285]);

% Construct the components.
hsurf = uicontrol('Style','pushbutton','String','Surf',...
    'Position',[315,220,70,25]);
hmesh = uicontrol('Style','pushbutton','String','Mesh',...
    'Position',[315,180,70,25]);
hcontour = uicontrol('Style','pushbutton',...
    'String','Contour',...
    'Position',[315,135,70,25]);
htext = uicontrol('Style','text','String','Select Data',...
    'Position',[325,90,60,15]);
hpopup = uicontrol('Style','popupmenu',...
    'String',{'Peaks','Membrane','Sinc'},...
    'Position',[300,50,100,25]);
ha = axes('Units','Pixels','Position',[50,60,200,185]);
align([hsurf,hmesh,hcontour,htext,hpopup],'Center','None');

% Make the UI visible.
f.Visible = 'on';

end

```

Şekil 3.12. Programlı olarak kodlanmış bir GUI örneği [43].

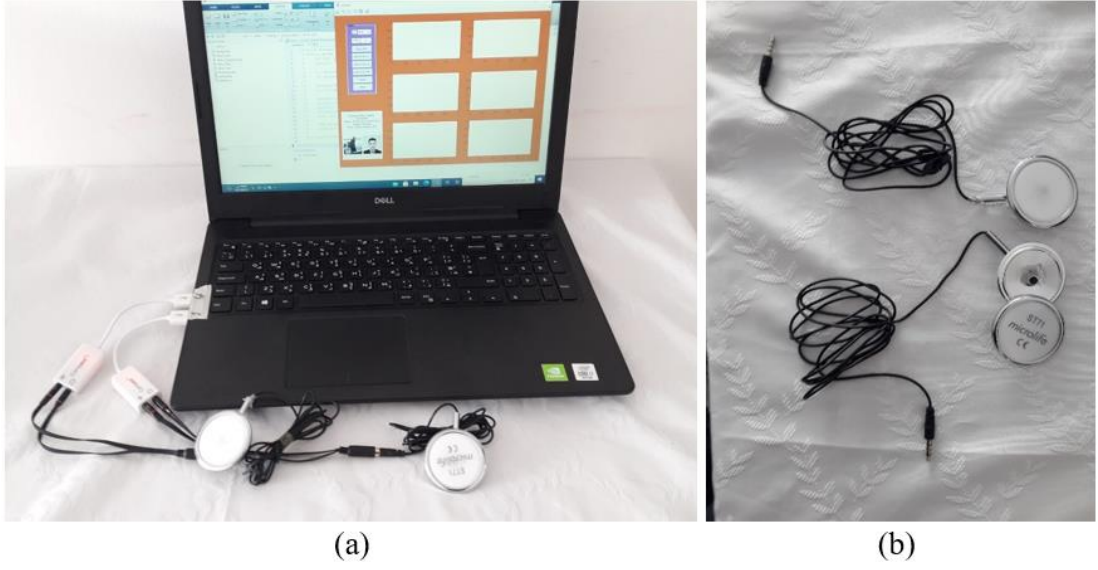
BÖLÜM 4

DENEYSEL UYGULAMA VE SONUÇLARI

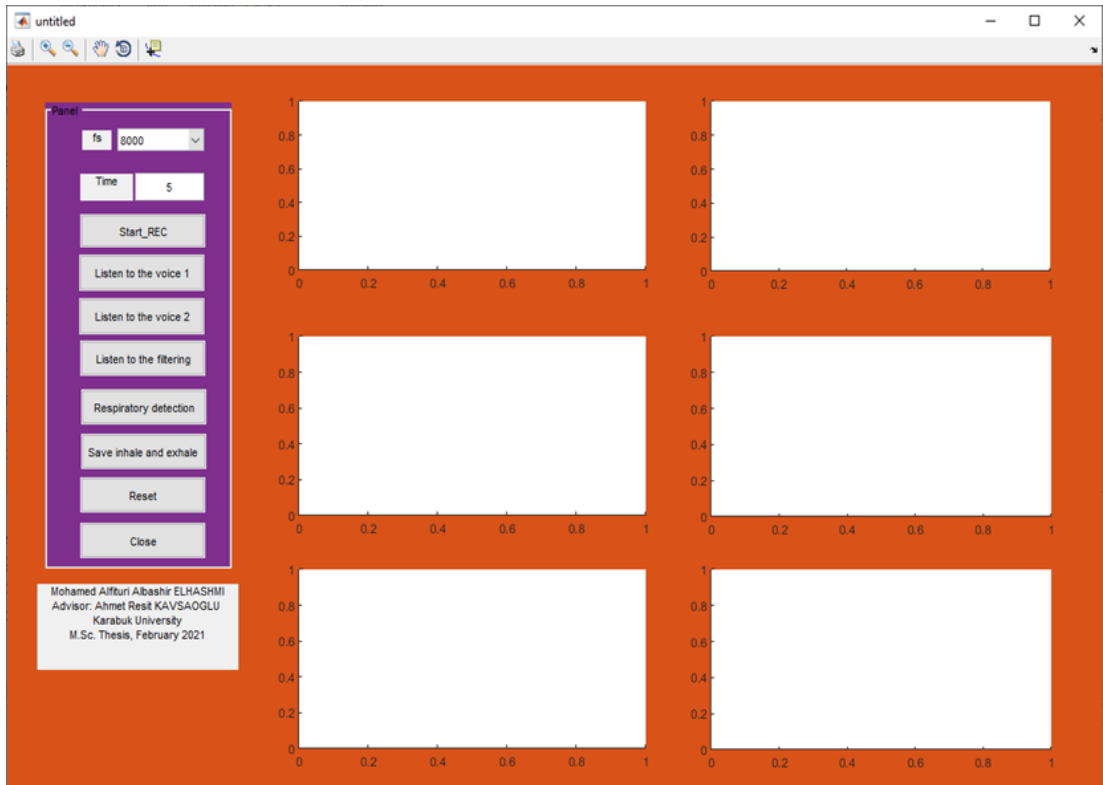
Adaptif filtre klasik filtreleme tekniğinden farklı olarak üç temel elemandan oluşmaktadır. Bu elemanlar toplayıcı eleman, ağırlıklandırıcı (çarpım) elemanı ve bir sayısal filtre yapısıdır. Bu elemanlara sahip bir sistem, çevre ortamına bağlı olarak filtre karakteristiğinde gereken değişmeyi filtre katsayılarını değiştirerek yapabilmektedir. Bu da solunum testlerinde ortaya çıkan gürültü seslerini ayırtmayı sağlayarak gürültüsüz ve doğru sinyal alımını destekleyerek ölçüm sonuçlarının doğruluğunu etkilemektedir. Solunum, göğüs kafesinin ve akciğerlerin hareketine ve bu organlarda meydana gelen art arda hacim ve basınç değişikliklerine sebep olmaktadır. Solunumun tespit edilmesinde bu hareketlerin algılanması ve soluk sayısının belirlenmesi çeşitli hastalıkların doğru tespitinin yapılarak erken teşhis ve tedavi olanaklarının artırılması açısından önemlidir.

4.1. DENEYSEL UYGULAMA

Bu çalışmada dizüstü bilgisayar, stetoskop üzerine yerleştirilen kombine mikrofon kullanılarak hasta ve sağlam bireylerde solunum testleri uygulanmıştır. Adaptif filtre algoritması geliştirilerek 10 kişi ile gerçekleştirilen bu solunum testlerinden elde edilen sinyaller gürültüden arındırılmış ve bir grafik arayüz programı sayesinde verilere dönüştürülmüştür. Elde edilen veriler sonucunda normal testler ile karşılaştırılmıştır. Şekil 4.1 algılama sürecinde kullanılan parçaları göstermektedir. Şekil 4.2, grafiksel kullanıcı arayüzünü (GUI) göstermektedir.



Şekil 4.1. Kullanılan donanım, a) dizüstü bilgisayar, b) Stetoskop üzerindeki kombine mikrofon.



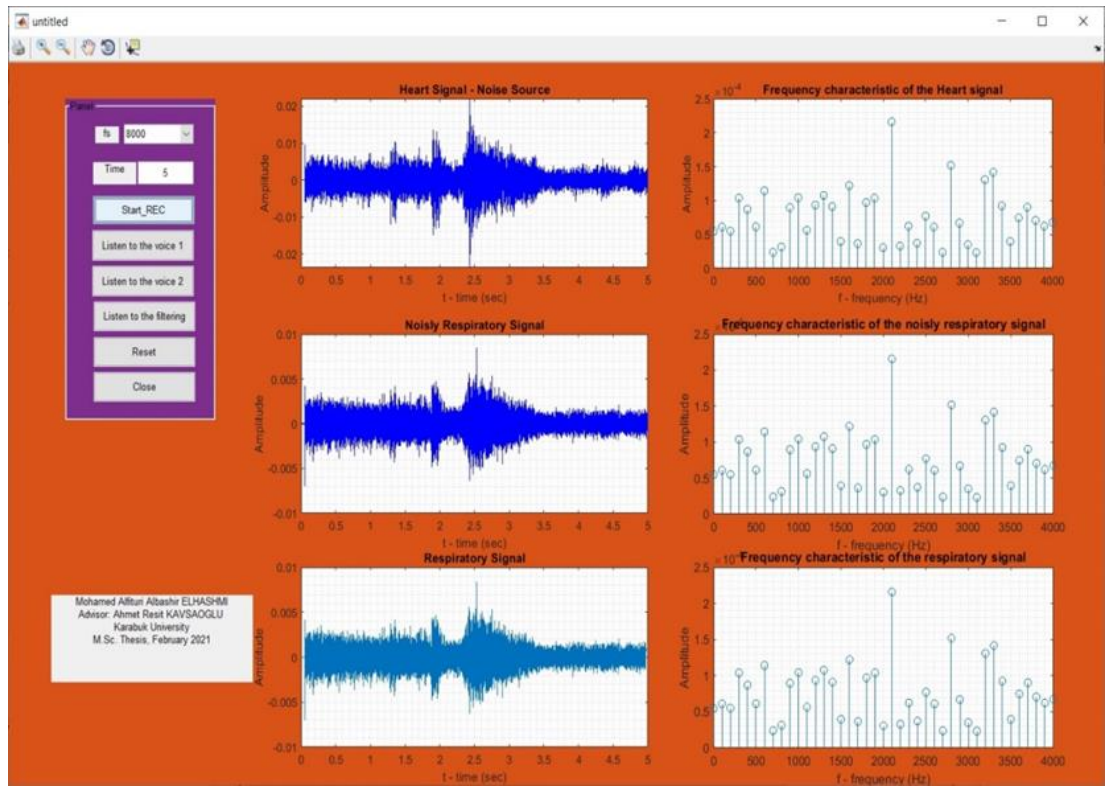
Şekil 4.2. Grafiksel kullanıcı Arayüzü (GUI).

4.1.1. Grafiksel Kullanıcı Arayüzü (GUI) Elemanları

Grafiksel kullanıcı arayüzü birkaç buton içermektedir, her butonun kendine özel bir işlevi vardır. Şekil 4.2’de görüldüğü gibi altı tane buton vardır.

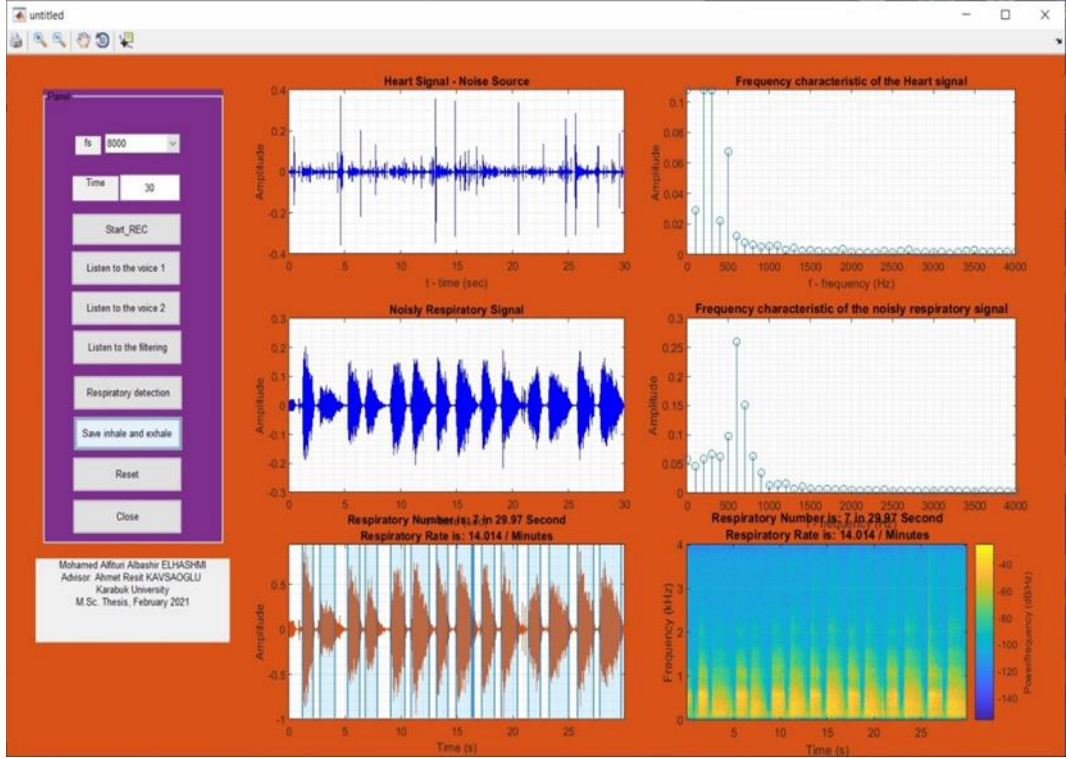
1. "Start_Rec" düğmesi

"Start_Rec" butonuna basıldıktan sonra, Ses kaydı yapılmaktadır. Şekil 4.3’te gösterilmiştir.



Şekil 4.3. "Start_Rec" düğmesine bastıktan sonra ortaya çıkan GUI.

"Solunum algılama" düğmesine bastıktan sonra Şekil 4.4’te gösterilen pencere açılacaktır.



Şekil 4.4. "Solunum algılama" düğmesine bastıktan sonra ortaya çıkan GUI.

2. "Ses 1'i dinle " düğmesi

Bu düğmeye tıkladıktan sonra ilk ses sinyali olan gürültü kaynağı kalp üzerinden kaydedilen sinyal çalmaya başlamaktadır.

3. "Ses 2'yi dinle " düğmesi

Bu düğmeye tıkladıktan sonra ikinci ses sinyali olan gürültülü solunum sinyali çalmaya başlamaktadır.

4. "Filtrelemeyi dinle" düğmesi

Bu düğmeye tıkladıktan sonra, adaptif filtrelenmiş ses sinyali çalar ve GUI'de göstermektedir.

5. "Sıfırla" düğmesi

Bu düğmeye tıkladıktan sonra tüm grafikler ve değerler sıfırlanmaktadır.

6. "Kapat" düğmesi

Bu düğmeye tıkladıktan sonra GUI penceresi kapanmaktadır.

4.2. ADAPTİF FİLTRELEMENİN GÜRÜLTÜ AZALTMADAKİ ETKİSİ

Adaptif filtrelemenin gürültü azaltmadaki etkisini incelemek için 8 kHz'lik bir ses dalgasına 500 Hz'lik genliği ve fazı rasgele değişen bir gürültü örneği eklenerek gürültülü sinyal oluşturmuştur. Gürültülü sinyal ile referans gürültü işareti kullanılarak adaptif filtreleme çıkışı elde edilmektedir. Adaptif filtre çıkışı hata sinyali olarak adlandırılmaktadır. Hata sinyali, sinyale eklenen gürültüyü iptal etme gürültüsüz sinyale yakınsama şeklinde elde edilmektedir. Şekil 4.5'de bu uygulamanın MATLAB kodları göstermektedir. Şekil 4.6'da gürültülü sinyal, filtrelenmiş sinyal ve gürültü kaynağı sinyali gösterilmektedir. Şekil 4.7'de üç farklı μ değeri (0.01, 0.1, 1) için adaptif filtreleme sonucu elde edilen hata sinyalleri gösterilmektedir.

```
clear;clc;close all;

[x,fs] = audioread('intro.wav');
x=x(1:80001);
t=0:1/fs:10;
n= 0.05*rand*sin(2*pi*500*t-rand*30); n=n'; % 500Hz_gürültü
d = x+n; % d = gürültülü Sinyal
t=0:1/fs:10;
mu = 0.01;
lms = dsp.LMSFilter(13,'StepSize',mu); % LMS Filtre Tanımı

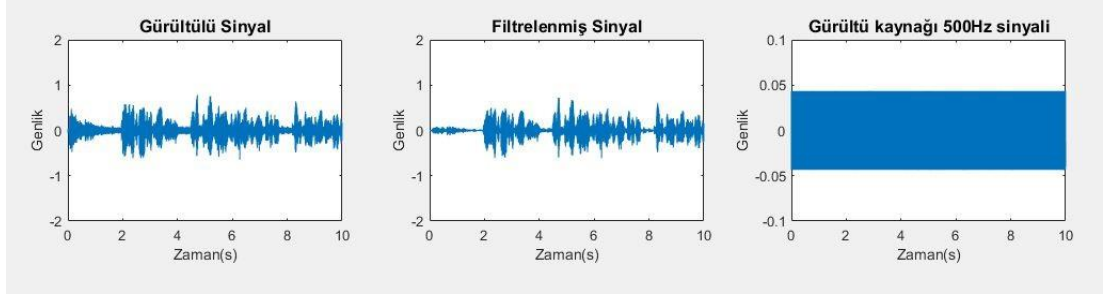
[y,e,~] = lms(x,d); % y: Filtre çıkışı, e=Hata çıkışı

subplot(3,3,1); plot(t,d);ylim([-2 2]);title('Gürültülü Sinyal','FontSize',12);
xlabel('Zaman(s)');ylabel('Genlik');
subplot(3,3,2); plot(t,y);ylim([-2 2]);title('Filtrelenmiş Sinyal','FontSize',12);
xlabel('Zaman(s)');ylabel('Genlik');
subplot(3,3,7); plot(t,e); ylim([-0.5 0.5]);title('Hata  $\mu=0.01$ ','FontSize',12);
xlabel('Zaman(s)');ylabel('Genlik');
subplot(3,3,3); plot(t,n);ylim([-0.1 0.1]);
title('Gürültü kaynağı 500Hz sinyali','FontSize',12);xlabel('Zaman(s)');ylabel('Genlik');

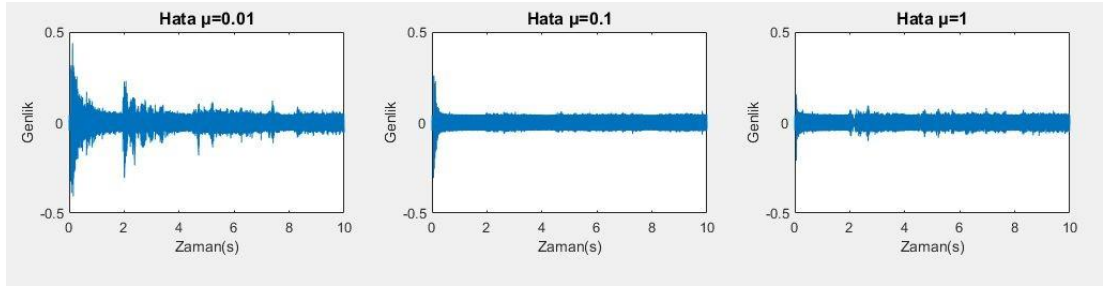
mu = 0.1;
lms = dsp.LMSFilter(13,'StepSize',mu); % LMS Filtre Tanımı
[~,e,~] = lms(x,d);
subplot(3,3,8); plot(t,e); ylim([-0.5 0.5]);title('Hata  $\mu=0.1$ ','FontSize',12);
xlabel('Zaman(s)');ylabel('Genlik');

mu = 1;
lms = dsp.LMSFilter(13,'StepSize',mu); % LMS Filtre Tanımı
[~,e,~] = lms(x,d);
subplot(3,3,9); plot(t,e); ylim([-0.5 0.5]);title('Hata  $\mu=1$ ','FontSize',12);
xlabel('Zaman(s)');ylabel('Genlik');
```

Şekil 4.5. Uygulamanın MATLAB kodları.



Şekil 4.6. Gürültülü sinyal, filtrenmiş sinyal ve gürültü kaynağı sinyali.



Şekil 4.7. 3 farklı μ değerinde üretilen hata sinyali.

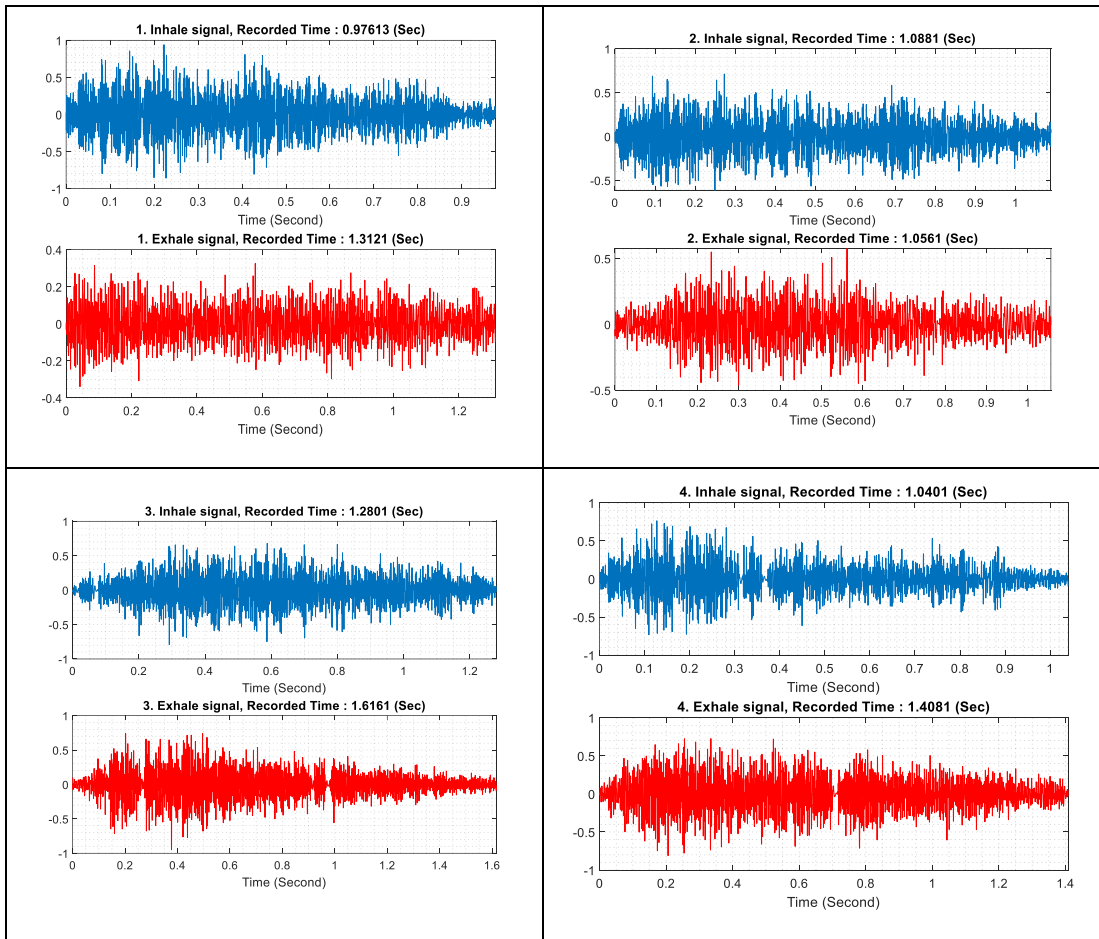
Şekil 4.7'deki deneyimden, adım boyutu μ arttıkça yakınsama hızının arttığı görülmektedir. Fakat frekans bölgesini tamamına yayılmış gürültülü sinyallerde adım boyutu arttıkça istenen değeri yakınsama hatası artabilir. Bu nedenle, filtre katsayılarının yüksek doğrulukta tahminleri isteniyorsa, gürültü biçimine göre uygun bir adım boyutu seçilmelidir. Bu değer uygulamaya göre deneme yanılma ile bulunabilmektedir.

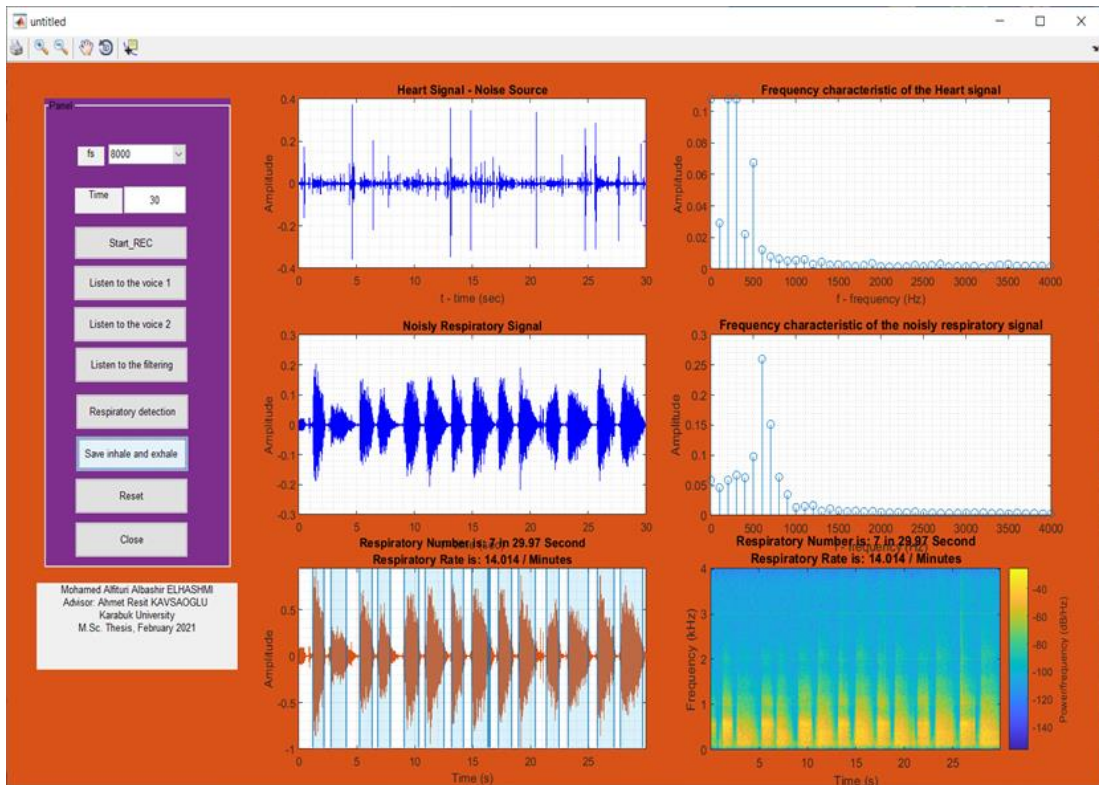
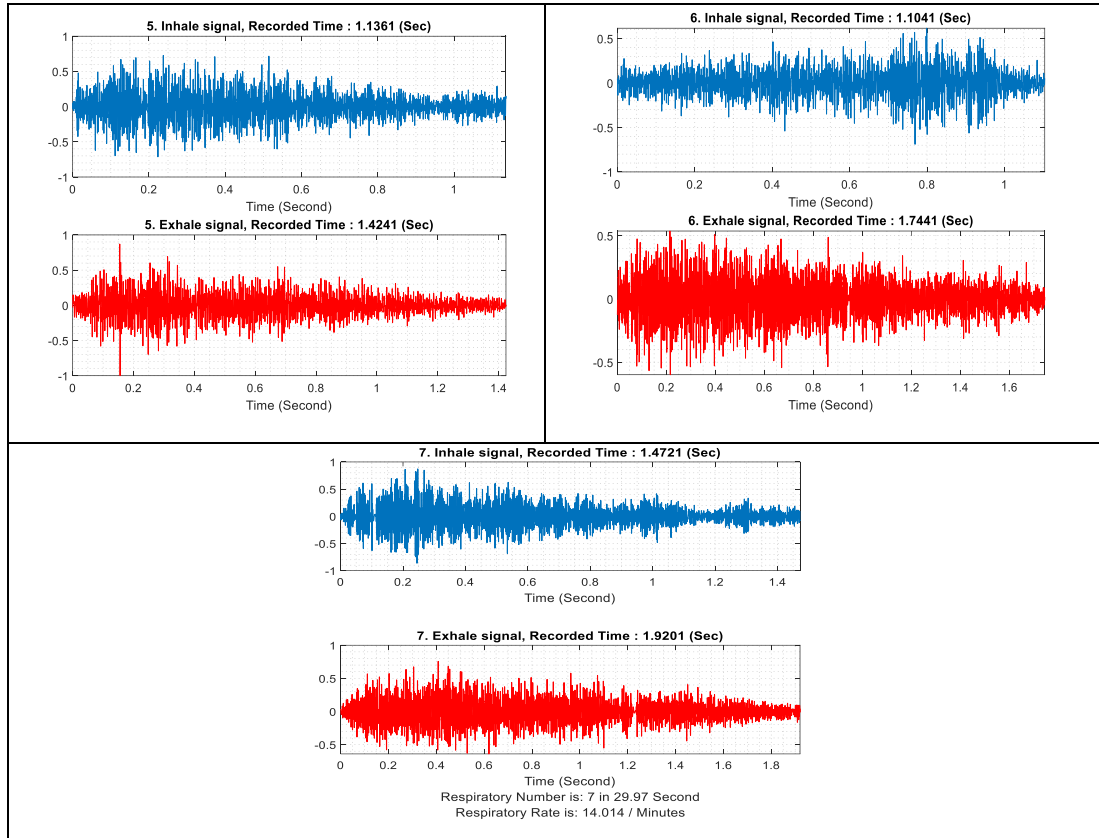
LMS algoritmasının yakınsama süresi, adım boyutuna μ bağlıdır. μ küçükse, uzun bir yakınsama süresi alabilir ve bu, bir LMS filtresi kullanma amacını bozabilir. Ancak μ çok büyükse, algoritma hiçbir zaman yakınsamayabilir. μ değeri, $d(n)$ üzerindeki etkilerine göre bilimsel olarak hesaplanabilmektedir [44]. Optimum μ değerini tespit etmeye yönelik literatürde çeşitli algoritmalar bulunmaktadır [45].

4.3. UYGULAMA SONUÇLARI

Bu bölümde 10 kişiden kaydedilen bazı sonuçlar gösterilmiştir. Aslında burada sadece nefes alma ve nefes verme sinyallerini içeren ilk ve son kişilerin solunum hızı gösterilmektedir. Birinci kişi için solunum sinyali Çizelge 4.1’de gösterilmiştir. Veri kümesindeki ilk kişi en genç kişidir. Şekil 4.8, en genç kişiden alınan verilerin analizini göstermektedir.

Çizelge 4.1. Birinci kişi için solunum hızı sonuçları (en genç kişi).





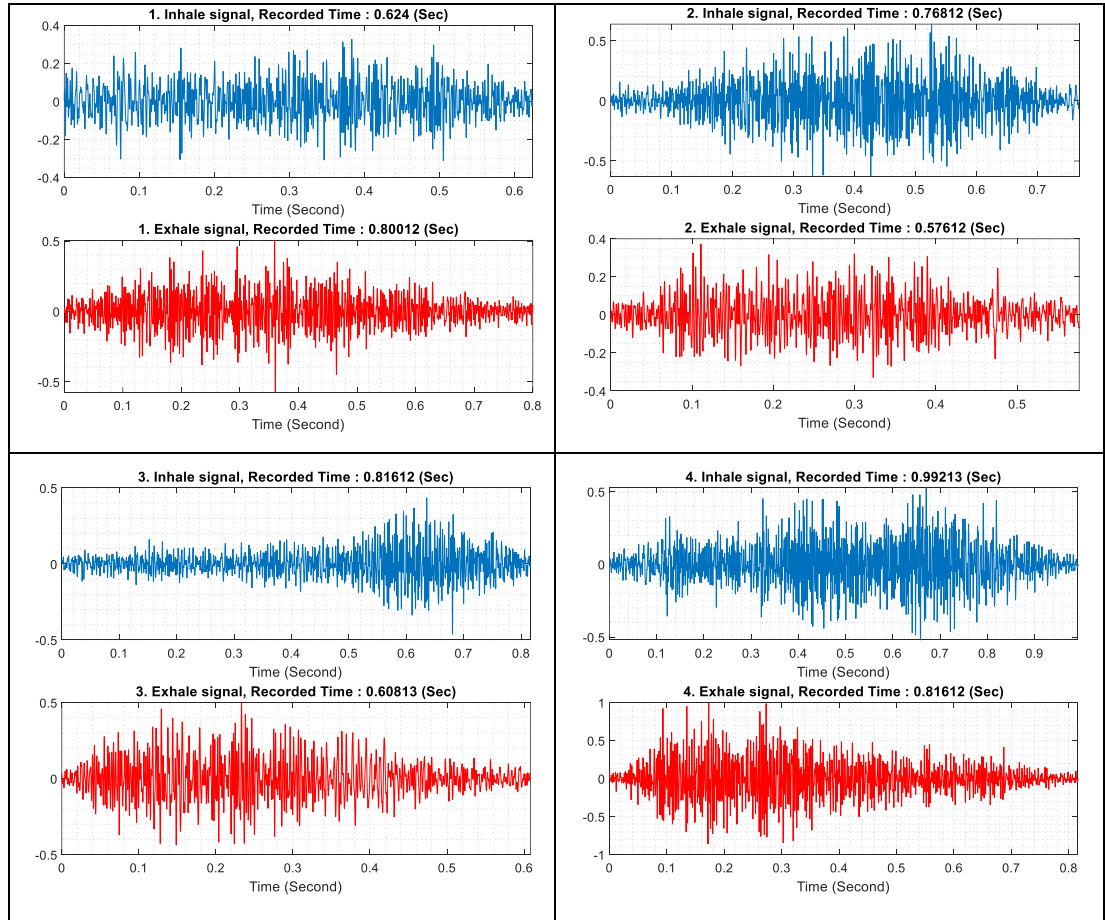
Şekil 4.8. En genç kişiden elde edilen sonuçlar.

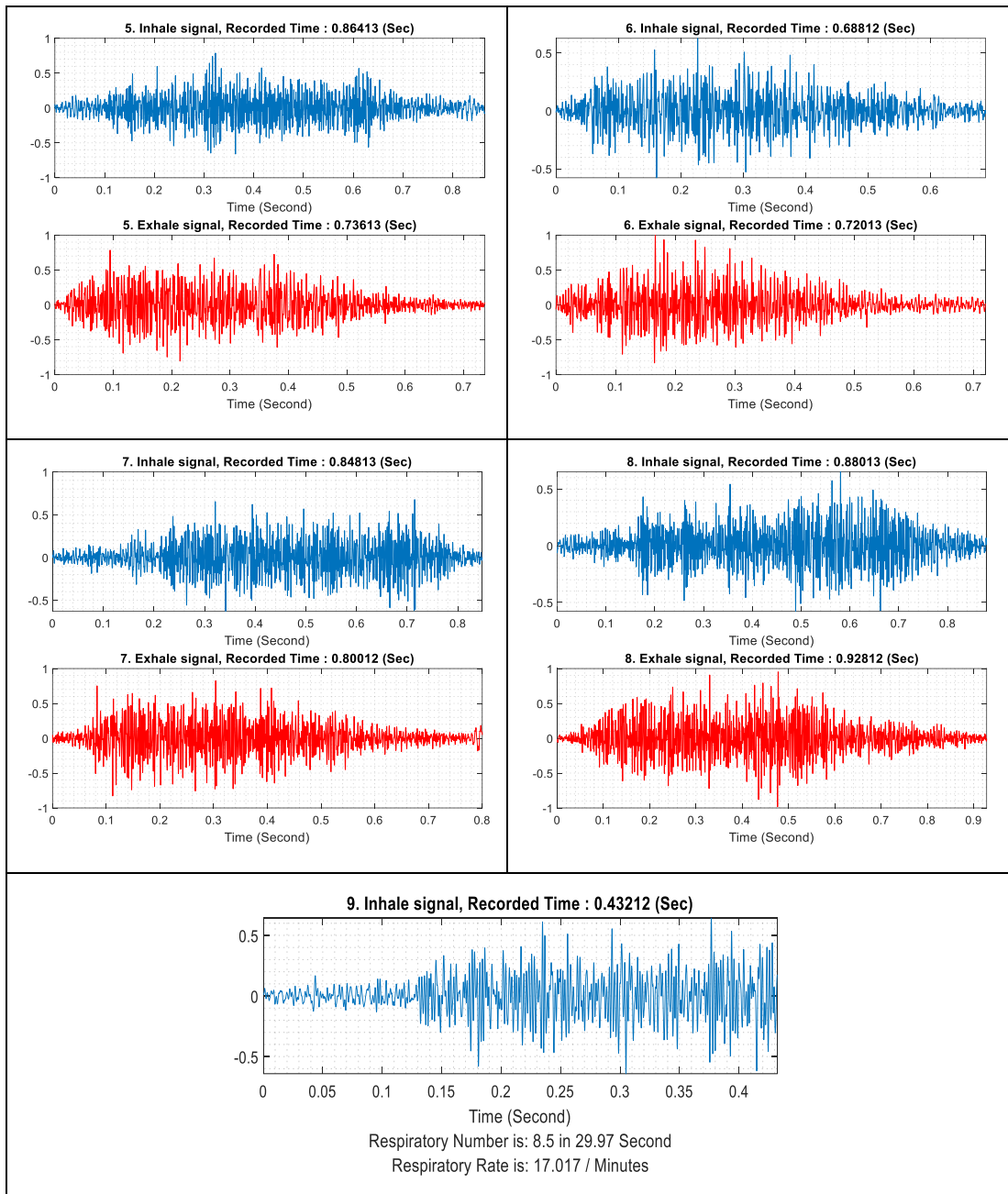
Tıbbi testlere göre yaklaşık 30 saniyelik solunum kaydedilir ve ardından solunum döngüsü (inspirasyon/ekspirasyon) sayısı hesaplanmaktadır. Daha sonra tam dakikadaki solunum hızını hesaplamak için kullanılmaktadır. Şekil 4.8, gruptaki en genç kişi için solunum hızının hesaplanmasını göstermektedir. Solunum sesleri 29.97 saniyelik bir süre boyunca kaydedildi ve sonuç 7 solunum döngüsü (inhalasyon / ekshalasyon) çıkmıştır, solunum hızı Eşitlik 4.1'e göre tam dakika için hesaplandı, sonuç solunum hızı dakikada 14.014 solunum döngüsü olmuştur. Aynı hesaplama Şekil 4.9'deki en yaşlı kişinin sonuçlarına da yapılmıştır.

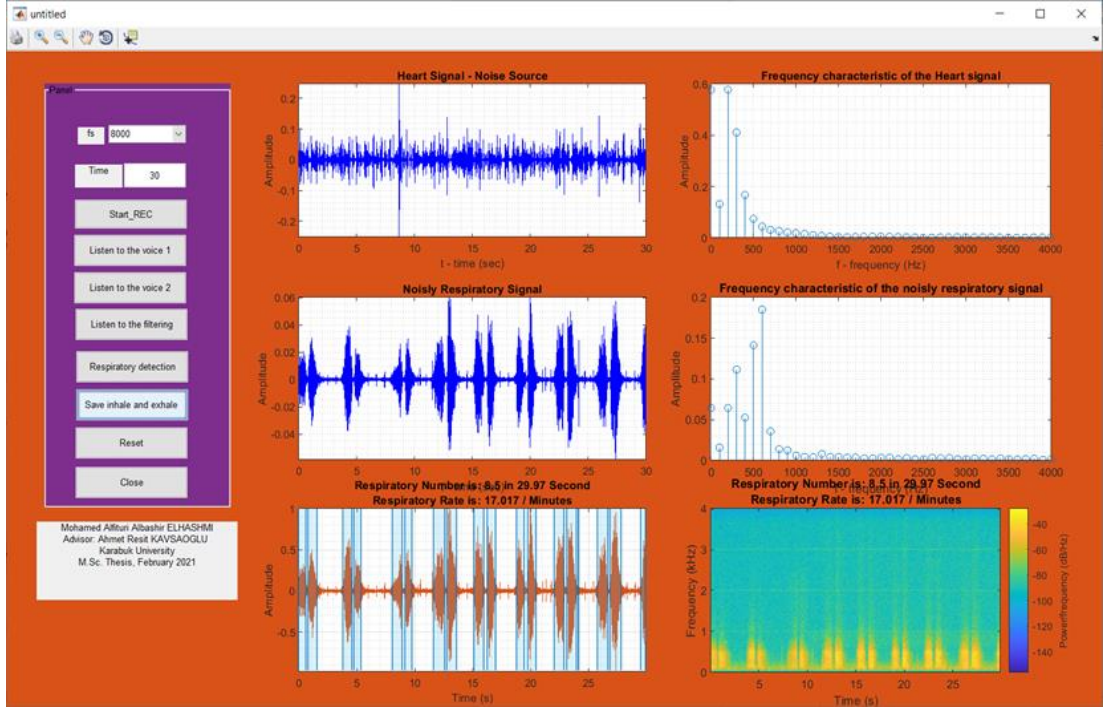
$$\frac{29.97 \text{ saniye}}{7 \text{ döngü}} = \frac{60 \text{ saniye}}{x}, \quad x = 14.014 \text{ solunum döngüsü} \quad (4.1)$$

Son kişinin sonuçları Çizelge 4.2'de gösterilmiştir. Veri kümesindeki son kişi en yaşlı kişidir.

Çizelge 4.2. Son kişinin solunum hızı sonuçları (en yaşlı kişi).







Şekil 4 9. En yaşlı kişinin sonuçları.

BÖLÜM 5

SONUÇ VE TARTIŞMA

Bu çalışmada, solunum sistemin anatomisi ve fizyolojisi, solunum sistemin önemli hastalıkların tespitinde kullanılabileceği vurgulandıktan sonra Adaptif filtreleme ile solunum oranı tespit sistem tasarımı gerçekleştirilmiştir. Adaptif filtreleme, solunum sinyallerini ve diğer biyomedikal sinyalleri işleme ve analiz etmenin en etkili ve en iyi bilinen yöntemlerinden biri haline geldiğinden, bir solunum hızı algılama sistemi tasarlamak için bu yöntem seçilmiştir.

Adaptif filtre uygulamalarından gürültü iptal etme tekniğinin bu çalışmaya dahil edilmesinin amacı, solunum oranının ölçülmesine çalışılırken, solunum sinyalinin kalp atışı sinyalleri ve sindirim sisteminden gelen bazı sinyaller gibi diğer biyolojik sinyallerle çakışması, solunum oranı hesaplamasında bazı hatalara neden olabilmesidir. Adaptif filtrelemenin gürültü azaltmadaki etkisi bir uygulama ile gösterilmiştir. Sonuçlar, bu amaca bu tür filtreler kullanılarak başarıyla ulaşıldığını göstermiştir.

Adaptif filtreleme sonucu elde edilen temiz solunum sinyaline ses sinyalinde konuşma sınırını tespit algoritması gerçekleştirilerek nefes alma ve nefes verme aralıkları tespit edilip parçalar halinde elde edilmiştir. Elde edilen bu nefes alma ve verme parçalarının adeti ve genişlik (süreleri) ve tekrar periyodu kullanılarak solunum oranı tespit edilmiştir. Bu çalışmada nefes alma ve nefes verme parçaları kullanılarak solunum oranı yanında zamana bağlı solunum oranı değişim hızıda tespit edilebilmektedir. Ayrıca, bu çalışmada geliştirilen sistem, kullanımı kolay bir grafik kullanıcı arayüzünü (GUI) sahiptir. Bu arayüz sayesinde solunum oranı takibi ve tanı teşhis amaçlı ürün geliştirme noktasında bilimsel veri toplama amaçlı kullanılabilir.

Tasarım ve programlama aşamasını tamamladıktan sonra, sistemin etkinliğini ve hazırlığını sağlamak için birçok pratik uygulamalar yapılmıştır. Bu uygulamalar on kişi üzerinde gerçekleştirilmiştir. Tez metninde en genç ile en yaşlı bireylerin deneysel uygulama sonuçlarına yer verilmiştir.

KAYNAKLAR

1. Dezube, R., "Exchanging Oxygen and Carbon Dioxide", *Merck Manuals*, (2019).
2. Ludlow, C. L., "Central nervous system control of interactions between vocalization and respiration in mammals", *Head & Neck*, 33 (S1): S21–S25 (2011).
3. Yıldırım, N., "SOLUNUM SİSTEMİ KLİNİK FİZYOLOJİSİ.", *Bulletin Of Thoracic Surgery/Toraks Cerrahisi Bülteni*, 10 (1): (2017).
4. "HEMŞİRELİK Yaşam (Vital) Bulgulari", *Kahramanmaraş Sütçü İmam Üniversitesi*, Ankara, 65-s41 (2012).
5. Costa, T. D. da, Vara, M. de F. F., Cristino, C. S., Zanella, T. Z., Neto, G. N. N., and Nohama, P., "Breathing Monitoring and Pattern Recognition with Wearable Sensors", *Wearable Devices - The Big Wave Of Innovation*, (2019).
6. Kilgour, O. F. G. and Riley, P. D., "Mastering Biology", *Macmillan International Higher Education*, (1999).
7. Barry Keenan, D. and Wilhelm, F. H., "Adaptive and wavelet filtering methods for improving accuracy of respiratory measurement", *Technical Papers Of ISA*, 455: 37–42 (2005).
8. Tiinanen, S., Tulppo, M., and Seppänen, T., "Reducing the effect of respiration in baroreflex sensitivity estimation with adaptive filtering", *IEEE Transactions On Biomedical Engineering*, 55 (1): 51–59 (2008).
9. Correa, A. G., Laciari, E., Patiño, H. D., and Valentinuzzi, M. E., "Artifact removal from EEG signals using adaptive filters in cascade", *Journal Of Physics: Conference Series*, 90 (1): 12081 (2007).
10. Sankar, A. B., Kumar, D., and Seethalakshmi, K., "Performance study of various adaptive filter algorithms for noise cancellation in respiratory signals", *Signal Processing: An International Journal (SPIJ)*, 4 (5): 267–278 (2010).
11. Abdallah, O. and Bolz, A., "Adaptive Filtering by Non-Invasive Vital Signals Monitoring and Diseases Diagnosis", *Adaptive Filtering Applications*, (2011).
12. Cassani, R., Mejia, P., Antonio Tavares, J., Sanchez, J. C., and Martinez, R., "Adaptive filtering for respiration influence reduction on Heart Rate Variability", *CCE 2011 - 2011 8th International Conference On Electrical*

Engineering, Computing Science And Automatic Control, Program And Abstract Book, (2011).

13. Pandey, V. K., Pandey, P. C., Burkule, N. J., and Subramanyan, L. R., "Adaptive filtering for suppression of respiratory artifact in impedance cardiography", ***2011 Annual International Conference Of The IEEE Engineering In Medicine And Biology Society***, 7932–7936 (2011).
14. Foussier, J., Teichmann, D., Jia, J., Misgeld, B., and Leonhardt, S., "An adaptive Kalman filter approach for cardiorespiratory signal extraction and fusion of non-contacting sensors", ***BMC Medical Informatics And Decision Making***, 14 (1): 1–15 (2014).
15. Zhang, Z., Silva, I., Wu, D., Zheng, J., Wu, H., and Wang, W., "Adaptive motion artefact reduction in respiration and ECG signals for wearable healthcare monitoring systems", ***Med Biol Eng Comput***, 52: 1019–1030 (2014).
16. Komorowski, D., Pietraszek, S., Tkacz, E., and Provaznik, I., "The extraction of the new components from electrogastrogram (EGG), using both adaptive filtering and electrocardiographic (ECG) derived respiration signal", ***BioMedical Engineering OnLine***, 14 (1): 1–16 (2015).
17. Wu, Y., Liu, J., He, B., Zhang, X., and Yu, L., "Adaptive Filtering Improved Apnea Detection Performance Using Tracheal Sounds in Noisy Environment: A Simulation Study", ***BioMed Research International***, 2020: (2020).
18. National Evaluation Systems Amherst, MA., I., "The Respiratory System. Instructional Materials in Anatomy and Physiology for Pennsylvania Health Occupations Programs.", (1979).
19. Patwa, A. and Shah, A., "Anatomy and physiology of respiratory system relevant to anaesthesia", ***Indian Journal Of Anaesthesia***, 59 (9): 533 (2015).
20. Internet: Bilgi Al, "Solunum Sistemi Organları ve Görevleri Nelerdir?", **<https://www.bilgial.com/solunum-sistemi-organlari-ve-gorevleri-nelerdir/>** (2021).
21. Macklem, P. T., "The mechanics of breathing", ***American Journal Of Respiratory And Critical Care Medicine***, 157 (4): S88–S94 (1998).
22. Internet: American Lung Association, "Particle Pollution", **<https://www.lung.org/clean-air/outdoors/what-makes-air-unhealthy/particle-pollution>** (2021).
23. AYTEKİN, Ö. and KULUALP, K., "INSAN ANATOMİSİ ders Notları", ***Dokuz Eylül Üniversitesi Anatomi Anabilim Dalı***, .
24. Internet: Çetin Vural, "Gırtlak Nedir?", **<http://cetinvural.com/bas-boyun-kanserleri/girtlak-nedir/>** (2021).

25. Koteeshwari, R. and Suhanya, "Performance Enhancement of Adaptive Filters Using Preprocessing Technique by Wavelet Transform", *International Journal Of Advanced Research In Electrical, Electronics And Instrumentation Engineering*, 3297: (2007).
26. KAVSAOĞLU, A. R., "Adaptif Filtreleme ile Akustik Yankı İptali", *Selçuk Üniversitesi*, (2005).
27. Tan, L. and Jiang, J., "Adaptive filters and applications", *Digital Signal Processing*, *Elsevier*, 421–474 (2019).
28. Chen, J., Benesty, J., Member, S., Huang, Y., and Doclo, S., "New Insights Into the Noise Reduction Wiener Filter", *IEEE TRANSACTIONS ON AUDIO, SPEECH, AND LANGUAGE PROCESSING*, 14 (4): (2006).
29. Jung, S. M., Seo, J.-H., and Park, P. G., "Normalized least-mean-square algorithm with a pseudo-fractional number of orthogonal correction factors", *Journal Of Advances In Computer Networks*, 3 (2): (2015).
30. Mustafa, R., Ali, M., Umat, C., and Al-Asady, D. A., "Design and implementation of least mean square adaptive filter on Altera Cyclone II field programmable gate array for active noise control", *IEEE Symposium On Industrial Electronics And Applications*, 1: 479–484 (2009).
31. Shinde, S. D., Patil, M. A. P., and Belagali, M. P. P., "System identification using adaptive filter algorithms", *IOSR Journal Of Electronics And Communication Engineering*, 54–57 .
32. Avalos, J. G., Sanchez, J. C., and Velazquez, J., "Applications of Adaptive Filtering", *Adaptive Filtering Applications*, (2011).
33. Avalos, J. G., Sanchez, J. C., and Velazquez, J., "Error coded affine projection algorithm", *2013 IEEE 4th Latin American Symposium On Circuits And Systems (LASCAS)*, 1–4 (2013).
34. Kehtarnavaz, N., "Adaptive Filtering", *Digital Signal Processing System Design*, 157–173 (2008).
35. Avalos, J. G., Sanchez, J. C., and Velazquez, J., "Applications of adaptive filtering", *Adaptive Filtering Applications*, 1: 3–20 (2011).
36. Giannakopoulos, T., "A method for silence removal and segmentation of speech signals, implemented in Matlab", *University Of Athens, Athens*, 2: (2009).
37. Internet: MathWorks, "Detect Boundaries of Speech in Audio Signal", <https://www.mathworks.com/help/audio/ref/detectspeech.html> (2021).
38. Benesty, J., Chen, J., and Habets, E. A. P., "Speech Enhancement in the STFT

- Domain", Speech Enhancement in the STFT Domain, *Springer Science & Business Media*, (2012).
39. Internet: Numpy, "Numpy.Hanning", <https://numpy.org/doc/stable/reference/generated/numpy.hanning.html> .
 40. Oppenheim, A. V., Buck, J. R., and Schafer, R. W., "Discrete-Time Signal Processing: United States Edition", 2. Ed., *Pearson*, Upper Saddle River, NJ, 859 (1998).
 41. Simakov, S., "Introduction to MATLAB graphical user interfaces", *DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION EDINBURGH, MARITIME-AUSTRALIA*, (2005).
 42. Mathworks INC, "MATLAB ® Creating Graphical User Interfaces R2015b", 2015b. Ed., *Mathworks*, (2000).
 43. Internet: MathWorks, "Create and Run a Simple Figure-Based App - MATLAB & Simulink", https://www.mathworks.com/help/matlab/creating_guis/about-the-simple-programmatic-gui-example.html (2021).
 44. Internet: Lattice semiconductor, "LMS Adaptive Filter-Reference Design RD1031", https://www.latticesemi.com/-/media/LatticeSemi/Documents/ReferenceDesigns/JM/LMSAdaptiveFilter.ashx?document_id=22982 .
 45. Özbay, Y. and Reşit Kavsaoğlu, A., "An optimum algorithm for adaptive filtering on acoustic echo cancellation using TMS320C6713 DSP", *Digital Signal Processing*, 20 (1): 133–148 (2010).

EK AÇIKLAMALAR A.

MATLAB KODLARI

```

function varargout = untitled(varargin)

% initialization code
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @untitled_OpeningFcn, ...
                  'gui_OutputFcn', @untitled_OutputFcn, ...
                  'gui_LayoutFcn', [] , ...
                  'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function untitled_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;

guidata(hObject, handles);

function varargout = untitled_OutputFcn(hObject, eventdata, handles)

varargout{ 1 } = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)

global fs;
global data;
global t;
global sure;
global sol;
global sag;
global r ;
global p
global q

axes(handles.axes1);
cla reset
axes(handles.axes2);

```

```

cla reset
axes(handles.axes3);
cla reset
axes(handles.axes4);
cla reset
axes(handles.axes8);
cla reset
axes(handles.axes9);
cla reset

sol=0;
sag=0;
fss=str2double(get(handles.edit1,'String'));
fs=fss(get(handles.edit1,'Value'));
recObj_1 = audiorecorder(fs, 16, 1, 2);
recObj_2 = audiorecorder(fs, 16, 1, 1);
sure=str2double(get(handles.edit2,'String'));
t=1/fs:1/fs:sure;
disp('Start speaking. ');
record(recObj_1);
record(recObj_2);
p=0;q=0;
r=1;
tic
while(toc<=sure+1)
    drawnow
    if sol==1
        p=p+1;
        sol_ontime(p)=toc;
        sol=0;
    end
    if sag==1
        q=q+1;
        sag_ontime(q)=toc;
        sag=0;
    end
end
stop(recObj_1);
stop(recObj_2);
drawnow
r=0;

data_1 = getaudiodata(recObj_1);
data_1=data_1(1:length(t));
data_2 = getaudiodata(recObj_2);
data_2=data_2(1:length(t));
% --- Executes on selection change in edit1.
axes(handles.axes1);
plot(t,data_1,'b');

```

```

xlabel('t - time (sec)'); ylabel('Amplitude');title('Heart Signal - Noise Source');
grid minor
hold on
try
ytest1(1:length(sag_ontime))=max(max(data_1));
ytest2(1:length(sag_ontime))=min(min(data_1));
if ytest2>0
    ytest2=-ytest1;
end
ytest3(1:length(sol_ontime))=max(max(data_1));
ytest4(1:length(sol_ontime))=min(min(data_1));
if ytest4>0
ytest4=-ytest3;
end
stem(sag_ontime,ytest1,'-r');
stem(sag_ontime,ytest2,'-r');
stem(sol_ontime,ytest3,'-g');
stem(sol_ontime,ytest4,'-g');
axis([0 t ytest1*1.1 ytest2*1.1]);
catch
end

axes(handles.axes3);
plot(t,data_2,'blue');
xlabel('t - time (sec)'); ylabel('Amplitude');title('Noisy Respiratory Signal');
grid minor
hold on
try
ytest1(1:length(sag_ontime))=max(max(data_2));
ytest2(1:length(sag_ontime))=min(min(data_2));
if ytest2>0
    ytest2=-ytest1;
end
ytest3(1:length(sol_ontime))=max(max(data_2));
ytest4(1:length(sol_ontime))=min(min(data_2));
if ytest4>0
ytest4=-ytest3;
end
stem(sag_ontime,ytest1,'-r');
stem(sag_ontime,ytest2,'-r');
stem(sol_ontime,ytest3,'-g');
stem(sol_ontime,ytest4,'-g');
axis([0 t ytest1*1.1 ytest2*1.1]);

catch
end
axes(handles.axes2);
N=80;

```

```

f = (0:(fs/2)/(N/2):fs/2);
DATA_1 = fft(data_1,N);
DATA_1=fftshift(abs(DATA_1));
stem(f,DATA_1((N/2):N));
xlabel('f - frequency (Hz)');
ylabel('Amplitude');
title('Frequency characteristic of the Heart signal');
grid minor

axes(handles.axes4);
N=80;
f = (0:(fs/2)/(N/2):fs/2);
DATA_2 = fft(data_2,N);
DATA_2=fftshift(abs(DATA_2));
stem(f,DATA_2((N/2):N));
xlabel('f - frequency (Hz)');
ylabel('Amplitude');
title('Frequency characteristic of the noisy respiratory signal');
grid minor

audiowrite('data_1.wav',data_1,fs)
audiowrite('data_2.wav',data_2,fs)

%*****Start of Adaptif Filtering*****
ar=0.01;
error=1;
delays=240;

xn=data_1;
dn=data_2;
h=zeros(1, delays,'double');
n=length(t);

for i=1:(n-delays)

    for j=1:delays
        h(j)=h(j) + ( ar*error*xn(i+j-1));
    end

    sum = 0.0;

    for j=1: delays
        sum = sum + h(j)*xn(i+j);
    end

    yn(i)=sum;

```

```

        error= dn(i) - sum;
        is(i)=error;
end
audiowrite('data_3.wav',is,fs);
data_3=is;

t=t(1:(length(t)-delays));
axes(handles.axes8);
plot(t,data_3');
xlabel('t - time (sec)'); ylabel('Amplitude');
title('Respiratory Signal');
grid minor
hold on
try
ytest1(1:length(sag_ontime))=max(max(data_3));
ytest2(1:length(sag_ontime))=min(min(data_3));
if ytest2>0
    ytest2=-ytest1;
end
ytest3(1:length(sol_ontime))=max(max(data_3));
ytest4(1:length(sol_ontime))=min(min(data_3));
if ytest4>0
ytest4=-ytest3;
end
stem(sag_ontime,ytest1,'-r');
stem(sag_ontime,ytest2,'-r');
stem(sol_ontime,ytest3,'-g');
stem(sol_ontime,ytest4,'-g');
axis([0 t ytest1*1.1 ytest2*1.1]);

catch
end

axes(handles.axes9);
N=80;
f = (0:(fs/2)/(N/2):fs/2);
DATA_3 = fft(data_3,N);
DATA_3=fftshift(abs(DATA_3));
stem(f,DATA_3((N/2):N));
xlabel('f - frequency (Hz)'); ylabel('Amplitude');
title('Frequency characteristic of the respiratory signal');
grid minor

% *****End of Drawing Filtering signal *****

try

```



```

sag_ontime=round(sag_ontime*fs);
sol_ontime=round(sol_ontime*fs);
catch
end
%*****End of Segmentation *****

handles.data_1 = data_1;
handles.data_2 = data_2;
handles.data_3 = data_3;
handles.fs = fs;

guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit2_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pushbutton2_Callback(hObject, eventdata, handles)

data_1 = handles.data_1;
fs = handles.fs;
sound(data_1,fs)

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)

data_2 = handles.data_2;
fs = handles.fs;
sound(data_2,fs)

```

```
function pushbutton4_Callback(hObject, eventdata, handles)
```

```
set(handles.edit2,'String','5');  
axes(handles.axes1);  
cla reset  
axes(handles.axes2);  
cla reset  
axes(handles.axes3);  
cla reset  
axes(handles.axes4);  
cla reset  
axes(handles.axes8);  
cla reset  
axes(handles.axes9);  
cla reset
```

```
function pushbutton5_Callback(hObject, eventdata, handles)
```

```
close all
```

```
function pushbutton6_Callback(hObject, eventdata, handles)
```

```
data_3 = handles.data_3;  
fs = handles.fs;  
sound(data_3,fs)
```

```
function pushbutton1_KeyPressFcn(hObject, eventdata, handles)
```

```
global sol;  
global sag;  
global r ;  
global p ;  
global q ;  
k=get(gcf,'CurrentCharacter');  
if k*1==28  
  
sol=1;  
end  
if k*1==29  
  
sag=1;  
end
```

```

function pushbutton7_Callback(hObject, eventdata, handles)

Fs = 8000;

data_1 = handles.data_1;
data_2 = handles.data_2;

sound(data_2,Fs)

sure = length(data_1)/Fs;

ar=0.01; % This is  $\mu$ 
error=1;
delays=240;

t=1/Fs:1/Fs:sure;

xn=data_1;
dn=data_2;
h=zeros(1, delays,'double');
n=length(t);
%-----LMS ALGORITHM IS STARTING-----
for i=1:(n-delays)

    for j=1:delays
        h(j)=h(j) + ( ar*error*xn(i+j-1));
    end

    sum = 0.0;

    for j=1: delays
        sum = sum + h(j)*xn(i+j);
    end

    yn(i)=sum;
    error= dn(i) - sum;
    is(i)=error;

end

audiowrite('data_3.wav',is,Fs);
Respiratory_Signal = is';

sound(Respiratory_Signal,Fs)

```

```

Respiratory_Signal = Respiratory_Signal/max(abs(Respiratory_Signal));

sound(Respiratory_Signal,Fs)
TimeDuration = length(Respiratory_Signal)/Fs

win = hann(128,'periodic');

axes(handles.axes8);
detectSpeech(Respiratory_Signal,Fs,'Window',win)

title('Detected Respiratory')

[idx,thresholds] = detectSpeech(Respiratory_Signal,Fs,'Window',win);

Dif = idx(:,2) - idx(:,1);

n = length(Dif);
count = 0;
N = 1;
for i = 1:n
    if Dif(i)>=3000
        count = count + 1;
        Signal{N} = Respiratory_Signal(idx(i,1) : idx(i,2));
        N = N + 1;
    else
        if i==n
            break
        elseif abs(Dif(i+1) - Dif(i))<=600 & (Dif(i+1) + Dif(i))>=2000
            count = count + 1;
            Signal{N} = Respiratory_Signal(idx(i,1) : idx(i,2));
            N = N + 1;
            Dif(i+1) = 0;
        end
    end
end
end

RespRateNo = count/2

RespRate = (RespRateNo/TimeDuration)*60

title({'Respiratory Number is: ',num2str(RespRateNo),' in ',
num2str(TimeDuration),' Second';['Respiratory Rate is: ',num2str(RespRate),' /
Minutes']})

axes(handles.axes9);

```

```
spectrogram(Respiratory_Signal,hann(1024,'periodic'),512,1024,Fs,'yaxis')
title(['Respiratory Number is: ',num2str(RespRateNo),' in ',
num2str(TimeDuration),' Second'];['Respiratory Rate is: ',num2str(RespRate),' /
Minutes'])
```

```
handles.Signal = Signal;
handles.RespRateNo = RespRateNo;
handles.RespRate = RespRate;
handles.TimeDuration = TimeDuration;
guidata(hObject, handles);
```

```
function pushbutton8_Callback(hObject, eventdata, handles)
```

```
Signal = handles.Signal;
RespRateNo = handles.RespRateNo;
RespRate = handles.RespRate;
TimeDuration = handles.TimeDuration;
```

```
Len = length(Signal);
Ni = 1;
Ne = 1;
for i = 1:2:Len
    inhale{Ni} = Signal{i};
    Ni = Ni + 1;
    if (i==Len & mod(Len,2)~=0)
        break
    else
        exhale{Ne} = Signal{i+1};
        Ne = Ne + 1;
    end
end
```

```
save Result inhale exhale RespRateNo RespRate TimeDuration
```

ÖZGEÇMİŞ

Mohamed ELHASHMI, İlk ve orta öğrenimini Misrata/ Libya şehrinde tamamladı. Misrata Fen Bilimleri Lisesinden mezun oldu. Eğitimine 2007 yılında Tıp Teknolojisi Fakültesi/ Tıbbi Cihaz Teknolojisi ve Bakımı Anabilim Dalı'nda başlamış ve 2010 yılında mezun olmuştur. 2010 yılında Misrata Adli ve suç Araştırmaları Merkezi'nde tıbbi cihazlar bakım teknisyeni olarak göreve başlamıştır. 2013 yılında Tıbbi Teknoloji Fakültesi/Misrata'da öğretim görevlisi olarak görev yapmıştır. 2019 yılında Türkiye'de Karabük Üniversitesi'nde Biyomedikal Mühendisliği Bölümünde Yüksek Lisans eğitimine başlamıştır.