

AKCİĞER SESİNİN DİNLENMESİ VE ANALİZİ İÇİN MOBİL UYGULAMA

Baha ERTEN

LİSANS TEZİ

BİLGİSAYAR MÜHENDİSLİĞİ

GAZİ ÜNİVERSİTESİ

TEKNOLOJİ FAKÜLTESİ

HAZİRAN 2023

ANKARA

Baha Erten tarafından hazırlanan “**Akciğer Sesinin Dinlenmesi ve Analizi İçin Mobil Uygulama**” adlı bu tezin Lisans Tezi olarak uygun olduğunu onaylarım.

Doç. Dr. Adem TEKEREK
Tez Danışmanı

Bu çalışma, jürimiz tarafından oy birliği ile Bilgisayar Mühendisliği Bölümünde Lisans Tezi olarak kabul edilmiştir.

Prof. Dr. O. Ayhan ERDEM

Doç. Dr. İbrahim Alper DOĞRU

Doç. Dr. Adem TEKEREK

Tarih:

Bu tez, G.Ü. Teknoloji Fakültesi Bilgisayar Mühendisliği’nce onanmıştır.

.....

Prof. Dr. Ayhan ERDEM

Bilgisayar Mühendisliği Bölüm Başkanı

ETİK BEYAN

Gazi Üniversitesi Teknoloji Fakültesi Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum bilgi ve dokümanları akademik kurallar etik çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmamda özgün verilerim dışında kalan ve tezde yararlanılan eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu ve başka bir yerde sunmadığımı beyan ederim.

Baha ERTEN

AKCİĞER SESİNİN DİNLENMESİ VE ANALİZİ İÇİN MOBİL UYGULAMA

(Mezuniyet Tezi)

Baha ERTEN

**GAZİ ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ**

Haziran 2023

ÖZET

Geliştirilen mobil uygulama, doktorlara akciğer seslerini dinleyerek analiz etmelerine yardımcı olan bir araçtır. Uygulama, tek bir ekran üzerinde Upload Sound, Enhance Sound ve Filter Sound olmak üzere üç buton ve Sound Visualizer adlı bir ses sinyali görselleştirici içerir. Upload Sound butonuyla istenilen ses yüklenir ve Sound Visualizer, ses sinyallerini ekranda gösterir. Enhance Sound butonu, yüklenen ses sinyallerini daha iyi analiz etmek için güçlendirir ve güçlendirilmiş ses sinyallerinin görselini ekranda görüntüler. Filter Sound butonu, yüklenen sesin kirliliğini azaltır ve gürültü olarak kabul edilen sesleri temizler. Filter Sound butonu uygulandıktan sonra, ses frekansı görseli sesin son durumuna göre güncellenir. Bu uygulama, doktorlara akciğer seslerinin analizinde kullanılan birçok işlemi tek bir platformda birleştirerek etkili bir araç sunar.

Bilim Kodu : 92419

Anahtar Kelimeler : Akciğer Sesi, Ses Analizi, Teşhis, Filtreleme, Solunum Yolu Hastalıkları, Android İS

Sayfa Adedi : 42

Tez Danışmanı : Doç. Dr. Adem TEKEREK

MOBILE APPLICATION FOR LISTENING AND ANALYZING LUNG SOUND

(Graduation Thesis)

Baha ERTEN

GAZİ UNIVERSITY

FACULTY OF TECHNOLOGY

June 2023

ABSTRACT

The developed mobile application is a tool that helps doctors analyze lung sounds by listening. The application includes three buttons Upload Sound, Enhance Sound and Filter Sound on a single screen and an audio signal visualizer called Sound Visualizer. The desired sound is uploaded with the Upload Sound button and Sound Visualizer displays the sound signals on the screen. The Enhance Sound button amplifies the loaded audio signals for better analysis and displays the visual of the amplified audio signals on the screen. The Filter Sound button reduces the pollution of the loaded sound and cleans the sounds considered as noise. After the Filter Sound button is applied, the audio frequency visual is updated according to the final state of the audio. This application offers doctors an effective tool by combining many processes used in the analysis of lung sounds into one platform.

Science Code : 92419

Keywords : Lung Sound, Sound Analysis, Diagnosis, Filtering, Respiratory Diseases, Android OS

Page Number : 42

Supervisor : Doç. Dr. Adem TEKEREK

TEŞEKKÜR

Tezin hazırlanması aşamasında bana her türlü desteği veren danışmanım sayın saygıdeğer hocam Doç. Dr. Adem TEKEREK'e, teşekkürlerimi sunmayı bir borç bilirim.

İÇİNDEKİLER

Sayfa

ÖZET.....	III
ABSTRACT.....	IV
TEŞEKKÜR.....	V
İÇİNDEKİLER.....	VI
ŞEKİLLERİN LİSTESİ.....	VIII
SİMGELER VE KISALTMALAR.....	IX
1. GİRİŞ.....	1
2. LİTERATÜR ARAŞTIRMASI.....	3
2.1 Steteskop ve Dijital Steteskop.....	3
2.1.1. Akustik işaretleme.....	5
2.2 Android İşletim Sistemi.....	6
2.2.1 Çekirdek.....	7
2.2.2 Kütüphaneler.....	7
2.3 Python Programlama Dili.....	8
2.4 Sound Visualizer ve Dijital Ses Görüntüsü.....	10
2.4.1 Sesin görselleştirilmesi.....	10
2.4.2 Sesten dijital sinyale dönüşüm.....	11
2.5 Ses Sinyali İçin Algoritmalar.....	13
2.5.1 Ses sinyalini artırmak için algoritmalar.....	13
2.5.2 Ses sinyalini filtrelemek için algoritmalar.....	14
3. İLGİLİ ÇALIŞMALAR.....	17

4. MATERİYAL VE YÖNTEM.....	19
4.1 Ses Kaynakları.....	19
4.2 Ses İşleme Kütüphaneleri.....	19
4.3 Ses İşleme Yöntemleri.....	20
4.4 Python Programlama Dili.....	22
4.5 Fonksiyonlar ve Algoritmalar.....	23
5. SONUÇ VE ÖNERİLER.....	27
KAYNAKLAR.....	29
ÖZGEÇMİŞ.....	31

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1.1 Dijital Steteskop	4
Şekil 2.1.2. Bir ses sinyalinin akustik işaretleme örneği	5
Şekil 2.2.1. Android Mimarisi	6
Şekil 2.3.1 Python Logosu	9
Şekil 2.4.1 Sesin görselleştirilmesi	10
Şekil 2.4.2 Sesin dijital sinyali ve noise distortion	11
Şekil 2.4.3 Dijital forma dönüştürülmüş ses örneği.....	12
Şekil 4.3.1 Low Pass ve High Pass filtreleri.....	20
Şekil 4.3.2 Gürültü azaltma filtreleme fonksiyonu algoritması	21
Şekil 4.3.3 Yankı giderme filtreleme fonksiyonu algoritması	21
Şekil 4.5 Uygulama ekran görüntüsü.....	20

SİMGELER VE KISALTMALAR

Kısaltmalar	Açıklama
LR	Logistic Regression (Lojistik Regresyon)
SVM	Support Vector Machine (Destek Vektör
CVD	KVH (Kardiyovasküler Hastalık Makinesi)
	Linear Discriminant Analysis (Doğrusal Diskriminant Analizi)
FPR	False Positive Rate (Yanlış Pozitif Oranı)
TPR	True Positive Rate (Gerçek Pozitif Oran)
TP	True Positive (Gerçek Pozitif)
FN	False Negative (Yanlış Negatif)
FP	False Positive (Yanlış Pozitif)
TN	True Negative (Gerçek Negatif)
ROC	Reciever Operator Characteristics (Alıcı Operatör Özellikleri)
FIR	Finite Impulse Response (Sonlu Dürtü Yanıtı)
IIR	Infinite Impulse Response (Sonsuz Dürtü Yanıtı)
JVM	Java Virtual Machine (Java Sanal Makinesi)
APK	Android Package Kit (Android Uygulama Paketi)

1. GİRİŞ

Teknolojinin hızla ilerlemesi ve mobil uygulamaların yaygınlaşması, sağlık alanında da çeşitli yeniliklerin ortaya çıkmasına olanak sağlamıştır. Bu yenilikler arasında, doktorların hastaların durumunu daha iyi değerlendirebilmelerine yardımcı olan mobil uygulamalar da yer almaktadır. Bu tezde, doktorların akciğer seslerini dinlemelerine ve analiz etmelerine yardımcı olacak bir mobil uygulama geliştirilmiştir.

Geliştirilen mobil uygulama, doktorlara akciğer seslerini dinlemelerine ve analiz etmelerine yardımcı olacak kullanışlı özellikler sunmaktadır. Uygulama, basit ve kullanıcı dostu bir arayüze sahiptir ve tek bir ekran üzerinde işlevlerini gerçekleştirir. Bu ekran üzerinde "Upload Sound" (Ses Yükle), "Enhance Sound" (Sesi Güçlendir) ve "Filter Sound" (Sesi Filtrele) adında üç düğme ile bir ses görselleştirici pencere bulunmaktadır.

"Upload Sound" düğmesi, kullanıcının istediği sesi uygulamaya yüklemesine olanak sağlar. Ses yüklendiğinde, "Sound Visualizer" (Ses Görselleştirici) adı verilen pencere, ses sinyallerini ekranda gösterir. Bu görselleştirme, doktorların ses sinyallerini daha iyi anlamalarına yardımcı olur.

"Enhance Sound" düğmesi, uygulamaya yüklenen ses sinyallerini daha iyi bir analiz için güçlendirmeyi sağlar. Güçlendirilmiş ses sinyallerinin görseli, "Sound Visualizer" penceresinde gösterilir. Bu özellik, doktorların ses sinyallerinin ayrıntılarını daha iyi bir şekilde incelemelerine olanak sağlar.

"Filter Sound" düğmesi, uygulamaya yüklenen sesin kirliliğini azaltır ve gürültü olarak kabul edilen sesleri temizler. Bu işlem, yüklenen ses dosyasına uygulanan bir algoritma yardımıyla gerçekleştirilir. "Filter Sound" düğmesi kullanıldıktan sonra, ekrandaki ses frekansı görseli, sesin son durumuna göre güncellenir. Bu özellik, doktorların daha temiz ve net bir şekilde ses sinyallerini incelemelerine olanak sağlar.

2. LİTERATÜR ARAŞTIRMASI

Yapılan literatür araştırmaları, akciğer seslerinin analizinde kullanılan farklı yöntemlerin avantajlarını ve dezavantajlarını ortaya koymaktadır. Bazı çalışmalar, dijital stetoskopların geleneksel stetoskoplara göre daha hassas sonuçlar verdiğini gösterirken, diğer çalışmalar dijital kayıt tekniklerinin ve anında veri iletiminin, akciğer seslerinin sınıflandırılmasında yüksek doğruluk oranlarına ulaşabildiğini göstermektedir. Elektronik Steteskop yardımıyla kaydedilen seslerin, Android işletim sistemli bir mobil teledondan anında veri iletimi sayesinde teknik analizinin yapılıp, hastalık teşhisinin doktorlar tarafından koyulması bu tezin amacıdır. Bu tezde, doktorların akciğer seslerini analiz etmelerine yardımcı olacak bir mobil uygulamanın geliştirilmesi ve sunulan özelliklerin etkinliğinin değerlendirilmesi amaçlanmaktadır. Daha önceki çalışmaların sonuçları ve yöntemleri, bu uygulamanın tasarımı ve geliştirilmesi sürecinde temel bir referans olarak kullanılacaktır.

2.1 Steteskop ve Dijital Steteskop

Geleneksel stetoskoplar, doktorların akciğer seslerini dinlemek için yaygın olarak kullandıkları araçlardır. Ancak, stetoskop kullanımı, doktorun deneyimine ve yeteneklerine bağlı olarak sonuçları etkileyebilir. Dijital stetoskoplar ise akciğer seslerinin daha hassas bir şekilde kaydedilmesini ve analiz edilmesini sağlar. Bu cihazlar, akciğer seslerini dijital formatta kaydederek doktorlara daha detaylı bir analiz imkanı sunar. Bazı dijital stetoskoplar, özellikle gürültü azaltma ve ses filtreleme gibi özelliklere sahip olabilir, bu da akciğer seslerinin daha net bir şekilde dinlenmesini sağlar.

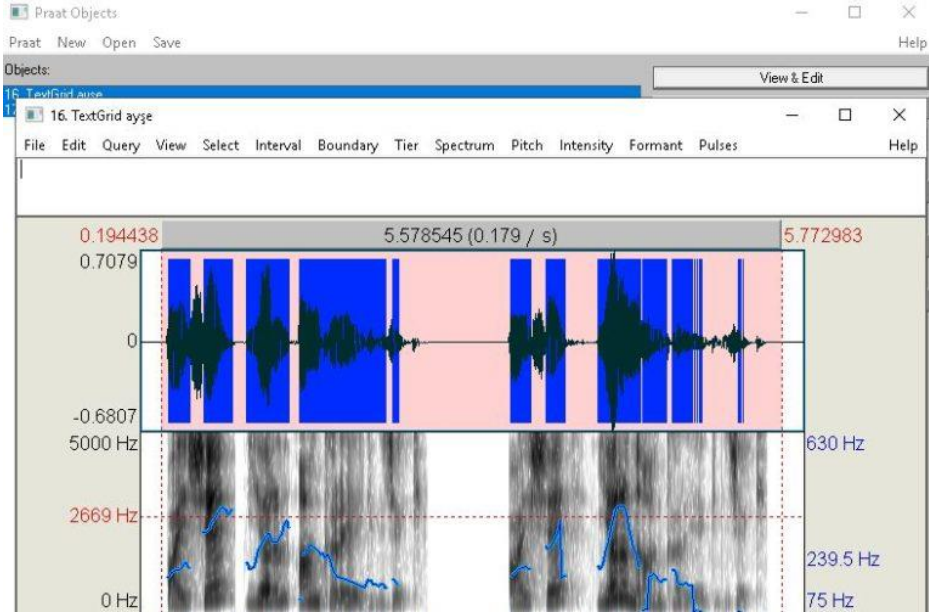
Kardiyovasküler hastalık (KVH), tüm dünyada önde gelen ölüm nedeni olarak kabul edilmektedir. Küresel ölümlerin yaklaşık üçte biri KVH'ye bağlanabilir. Klinik sunuma ek olarak, spesifik klinik muayene bulguları KVH'nin tedavisine ve önlenmesine yardımcı olabilir. CVD başlangıçta pulmoner patoloji olarak ortaya çıkabilir ve bu nedenle doğru kardiyopulmoner oskültasyon, doğru tanı koymak için çok önemlidir. Doktorların kullanabileceği en güçlü araçlardan biri stetoskoptur. Stetoskop ilk olarak 1818 yılında Fransız doktor René Laennec tarafından icat edildi. O zamandan beri, başlangıçtaki mütevazı mono ahşap tüp, gelişmiş bir dijital cihaza dönüştü. Bu makale, stetoskopun evriminin bir analizini sunarken, aynı zamanda bu aracın KVH'den mustarip hastalara yönelik bakımın ilerletilmesinde uygulanması da dahil olmak üzere modern dijital steteskopun kaydettiği ilerlemeyi vurgulamaktadır. [1]



Şekil 2.1.1 Dijital Steteskop [1]

2.1.1 Akustik işaretleme

Akciğer seslerinin analizi için kullanılan bir başka yöntem, akustik işaret işleme tekniklerinin uygulanmasıdır. Bu yöntemde, ses sinyalleri matematiksel işlemlerle analiz edilir ve belirli özellikler çıkarılır. Örneğin, sesin frekansı, zaman alanı ve enerjisi gibi özellikler incelenerek, akciğer seslerinin normal veya anormal olduğu belirlenebilir. Bazı çalışmalarda, akustik işaret işleme yöntemlerinin akciğer seslerinin teşhisi için yüksek doğruluk oranlarına ulaşabildiği gösterilmiştir. [2]



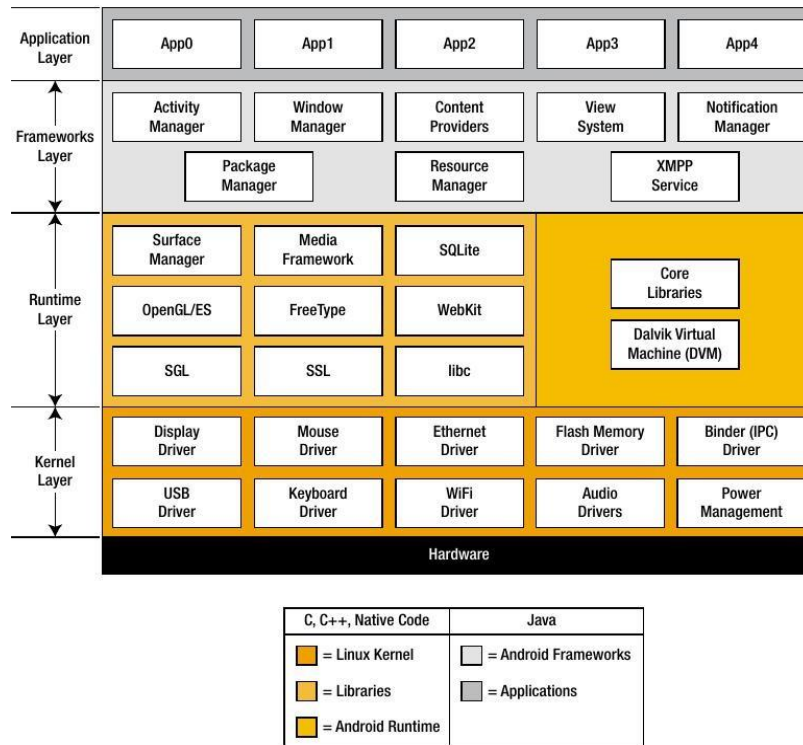
Şekil 2.1.2. Bir ses sinyalinin akustik işaretleme örneği [2]

2.2 Android İşletim Sistemi

Android işletim sistemi, Linux çekirdeğini kullanan açık kaynak kodlu bir işletim sistemidir ve dünya genelinde mobil cihazların en yaygın olarak kullandığı işletim sistemidir. Android, 2003 yılında Kaliforniya'da Andy Rubin, Rich Miner, Chris White ve Nick Sears tarafından geliştirilmeye başlandı ve 2005 yılında Google tarafından satın alındı. Android, 2011 yılından bu yana dünya çapında en çok satılan akıllı telefon işletim sistemi haline gelmiştir [2].

Android mimarisi aşağıdaki dört ana bileşene ayrılmıştır. Ana bileşenler şunlardır;

- Çekirdek
- Kütüphaneler ve Dalvik sanal makinesi
- Uygulama çerçevesi
- Uygulama



Şekil 2.2.1. Android Mimarisi [3]

2.2.1 Çekirdek

Android, Linux çekirdeği üzerinde çalışmaktadır. Çekirdek, cihaz donanımı ile etkileşime giren ilk yazılım katmanıdır. Android çekirdeği güç ve bellek yönetimi, aygıt sürücülerini, süreç yönetimi, ağ ve güvenlikle ilgilenmektedir. Genel olarak, son kullanıcı yeni bir çekirdeği değiştirmeyi ya da oluşturmayı düşünmemelidir. Bununla birlikte, donanım veya cihaz üreticileri, işletim sisteminin kendi özel donanım türleriyle çalışmasını sağlamak için çekirdeği değiştirmek isteyecektir.

2.2.2 Kütüphaneler

Kütüphaneler bileşeni, çekirdek ve uygulama çerçevesi arasında bir çeviri katmanı görevi görür. Android kütüphaneleri C/C++ dilinde yazılmıştır ancak bir Java API aracılığıyla gösterilir, yani Android kütüphanelerine Java çerçevesi aracılığıyla erişilebilmektedir. Çalışma zamanı bileşeni, uygulamalarla etkileşime girecek ve uygulamaları çalıştıracak Dalvik sanal makinesinden oluşur. Sanal makine Android işletim sisteminin önemli bir parçasıdır ve sistem ve üçüncü taraf uygulamalarını yürütür.

2.3 Python Programlama Dili

Python, geniş bir yelpazede uygulama alanlarına sahip popüler bir programlama dilidir. Genel olarak, kolay okunabilirlik, basit ve anlaşılır sözdizimi, büyük bir kütüphane desteği ve geniş bir topluluk tarafından desteklenmesi gibi avantajlarıyla öne çıkmaktadır.

1. **Sözdizimi ve Okunabilirlik:** Python'ın temel tasarım felsefesi, kodun okunabilir ve anlaşılabilir olmasıdır. Bunun için, basit ve sade bir sözdizimi kullanır. Kod bloklarının belirtilmesi için girintileme yapısı kullanılır, bu da okunabilirliği artırır ve kodun daha anlaşılır olmasını sağlar.
2. **Çeşitli Uygulama Alanları:** Python, birçok farklı uygulama alanında kullanılan genel bir amaçlı programlama dilidir. Web geliştirme, bilimsel hesaplama, veri analizi, yapay zeka, makine öğrenmesi, oyun geliştirme, ağ programlama ve otomasyon gibi alanlarda sıkça tercih edilir. Özellikle veri bilimi ve yapay zeka alanlarında popülerliği hızla artmıştır.
3. **Büyük Kütüphane Desteği:** Python, zengin bir kütüphane ekosistemine sahiptir. Bu kütüphaneler, çeşitli işlevlerin hızlı ve kolay bir şekilde gerçekleştirilmesini sağlar. Örneğin, NumPy ve Pandas, veri manipülasyonu ve analizi için sıkça kullanılan kütüphanelerdir. TensorFlow ve PyTorch ise yapay zeka ve makine öğrenmesi projeleri için tercih edilen kütüphanelerdir.
4. **Modülerlik ve Genişletilebilirlik:** Python, modüler bir yapıya sahiptir. Bu, bir projenin farklı bölümlerinin ayrı dosyalarda organize edilebileceği ve daha sonra bu bölümlerin birleştirilerek kullanılabilmesi anlamına gelir. Ayrıca, Python, C veya C++ gibi dillerle yazılan modüllerin Python ile birleştirilmesine izin veren bir arabirim sağlar, bu da Python'ın genişletilebilirliğini artırır.

5. **Topluluk ve Kaynaklar:** Python, geniş bir kullanıcı ve geliştirici topluluğuna sahiptir. Bu topluluk, sorularınızı sormak, projelerinizi paylaşmak ve işbirliği yapmak için birçok platformda bir araya gelir. Python ile ilgili kaynaklar da oldukça geniştir. Çevrimiçi dokümantasyon, kitaplar, eğitim videoları ve çevrimiçi kurslar gibi kaynaklar, Python öğrenmek ve geliştirmek isteyenler için bol miktarda bilgi sunar. [4]

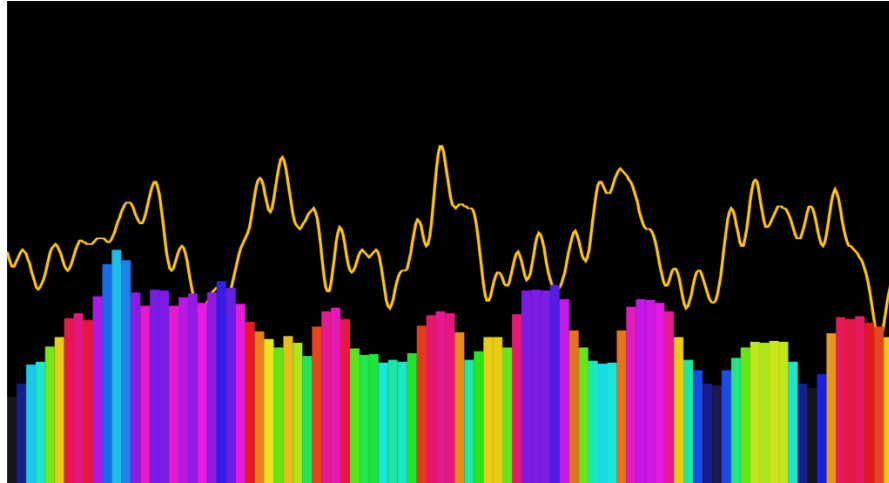


Şekil 2.3.1 Python Logosu [4]

2.4 Sound Visualizer ve Dijital Ses Görüntüsü

2.4.1 Sesin görselleştirilmesi

Ses görselleştirme, ses verilerinin görsel temsilleri aracılığıyla anlaşılmasını sağlayan bir tekniktir. Bu teknik, sesin zaman, frekans ve genlik gibi özelliklerini görsel elementlerle eşleştirerek kullanıcının sesi daha iyi anlamasını ve analiz etmesini sağlar. Ses görselleştirme, sesin daha sezgisel bir şekilde temsil edilmesini ve karmaşık ses verilerinin görsel olarak anlaşılmasını kolaylaştırır. Ses görselleştirme için kullanılan teknikler arasında dalgaformu, spektrogram, sonogram, parametrik temsiller ve 3D görselleştirme teknikleri bulunur. Dalgaformu, sesin zaman bileşenini görsel olarak temsil ederken, spektrogram ve sonogram sesin frekans bileşenlerini zamanla ilişkilendirir. Parametrik temsiller, sesin belirli özelliklerini (örneğin, enerji, frekans modülasyonu) görsel elementlerle temsil ederken, 3D görselleştirme teknikleri sesin daha karmaşık yapılarını görselleştirebilir. Ses görselleştirme, müzik, ses sanatı, akustik analiz, sağlık teşhisi ve ses eğitimi gibi birçok alanda kullanılır. Ses verilerini görsel olarak anlamak, kullanıcıların sesi daha iyi değerlendirmelerini, sesi manipüle etmelerini ve etkileşimli deneyimler yaşamalarını sağlar.

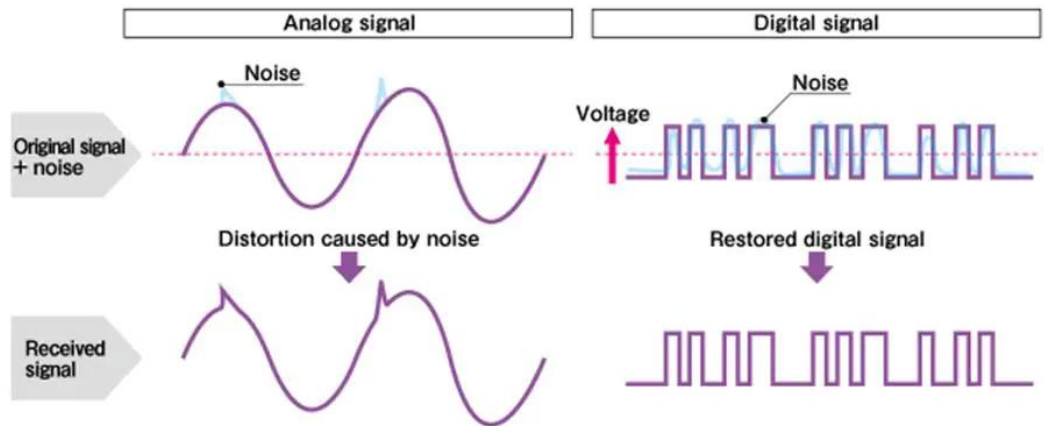


Şekil 2.4.1 Sesin görselleştirilmesi [5]

2.4.2 Sesten dijital sinyale dönüşüm

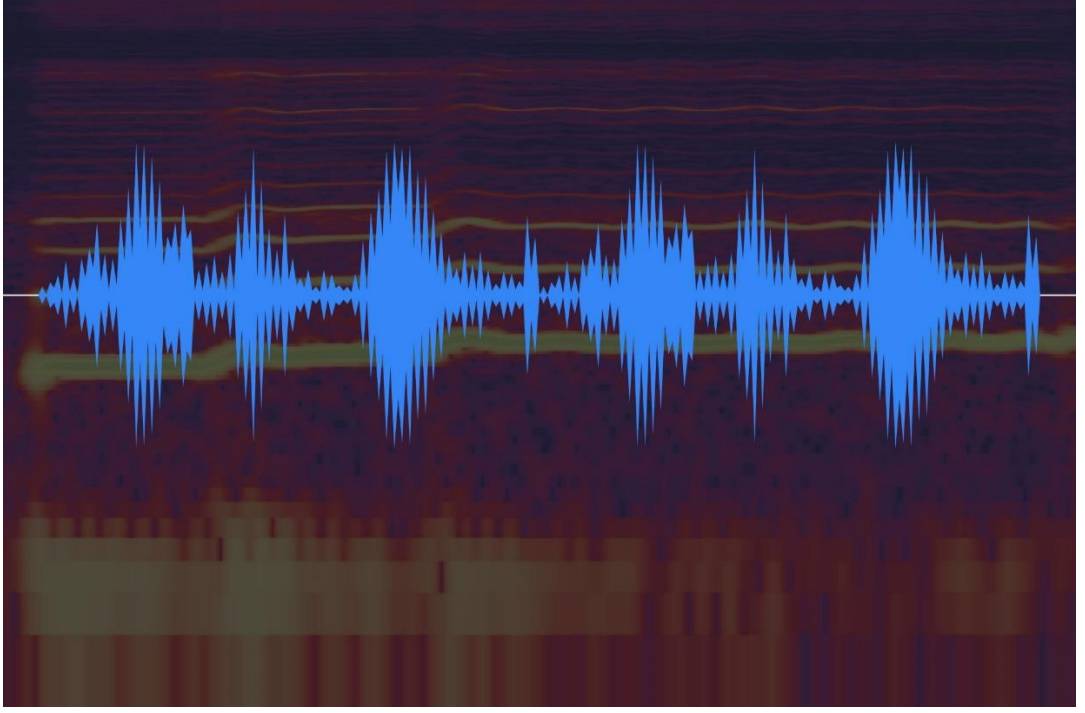
Dijital ses sinyalleri, sesin analog formdaki dalga şeklinin örnekleme ve nicelleştirme yöntemleriyle dijital verilere dönüştürülmesiyle elde edilir. Bu süreç, ses verilerini dijital sistemlerde işlemek, depolamak ve iletmek için temel bir adımdır. Sesin dijital forma dönüştürülmesi, örnekleme, nicelleştirme ve kuantalama adımlarını içerir.

- **Örnekleme**, ses dalga formunun belirli aralıklarla zaman içinde numunelerinin alınması işlemidir. Bu işlem, ses dalga formunu sürekli zaman boyunca belirli noktalarda örnekleme hızına göre kesikli örnekler halinde temsil eder. Örnekleme hızı, ses sinyalinin hassasiyetini ve yeniden oluşturma kalitesini belirler.
- **Nicelleştirme**, örneklenen ses sinyalindeki her bir örneğin genliğinin belirlenmesidir. Bu adımda, örnekleme sonucu elde edilen sayısal değerler belirli bir bit derinliğiyle ifade edilir. Genellikle, daha yüksek bit derinliği daha yüksek çözünürlük ve daha iyi ses kalitesi sağlar.
- **Kuantalama**, nicelleştirilen ses sinyalinin sayısal değerlerinin belirli bir değer aralığına sıkıştırılmasıdır. Bu işlem, sayısal değerlerin belli bir hassasiyetle temsil edilmesini sağlar. Kuantalama işlemi, ses sinyalindeki küçük değişimleri belirli bir hata payıyla yakınsama prensibine göre azaltır.



Şekil 2.4.2 Sesin dijital sinyali ve noise distortion [6]

Dijital ses sinyalleri, bu örnekleme, nicelleştirme ve kuantalama işlemleri sonucunda elde edilir. Bu dijital veriler, bilgisayarlar, ses cihazları ve diğer dijital ses sistemleri tarafından işlenebilir, depolanabilir ve aktarılabilir.



Şekil 2.4.3 Dijital forma dönüştürülmüş ses örneği

2.5 Ses Sinyali İçin Algoritmalar

2.5.1 Ses sinyali artırma için algoritmalar

Ses sinyalleri, çeşitli kaynaklardan gelen ve çeşitli faktörlerden etkilenen karmaşık verilerdir. Bu nedenle, ses sinyallerinin analiz edilmesi ve daha iyi bir kaliteye sahip olması gereken birçok uygulama vardır. Ses sinyali güçlendirme algoritmaları, bu tür ihtiyaçları karşılamak için geliştirilmiştir.

Birincil işlevlerinden biri, istenmeyen gürültüyü azaltmaktır. Gürültü, kaynak cihaz, elektriksel etkileşimler veya çevresel faktörler gibi çeşitli nedenlerle sinyallere karışabilir. Ses sinyali güçlendirme algoritmaları, gürültü bileşenlerini tespit eder ve sinyaldeki bu bileşenleri filtreler. Bu işlem, ses sinyalinin temizlenmesine ve istenmeyen arka plan gürültüsünün azaltılmasına yardımcı olur. Ayrıca, ses sinyali güçlendirme algoritmaları düşük ses seviyelerini yükseltmek için kullanılır. Ses sinyalleri, bazen kaynağın uzaklığından veya diğer faktörlerden dolayı düşük seviyelerde alınabilir. Bu durumda, ses sinyali güçlendirme algoritmaları, sesin daha iyi duyulmasını sağlamak için sinyal seviyelerini artırır. Ses sinyali güçlendirme algoritmaları ayrıca, önemli ses özelliklerini vurgulamak ve sesin daha iyi anlaşılmasını sağlamak için kullanılır. Örneğin, konuşma sinyallerinde, vokal bileşenlerinin ön plana çıkarılması ve arka plan gürültüsünün azaltılması önemlidir. Bu algoritmalar, konuşma sinyalindeki vokal bileşenlerini belirler ve onları diğer bileşenlerden ayırarak anlaşılabilirliği artırır.

Sonuç olarak, ses sinyali güçlendirme algoritmaları, ses sinyallerinin kalitesini artırmak, gürültüyü azaltmak, düşük ses seviyelerini yükseltmek ve önemli ses özelliklerini vurgulamak için kullanılan algoritmaları ifade eder. Bu algoritmalar, ses işleme uygulamalarında yaygın olarak kullanılır ve ses analizi, konuşma tanıma, akustik araştırmalar gibi birçok alanda önemli bir rol oynar. [8]

1. **Wavelet Tabanlı Gürültü Azaltma Algoritmaları:** Wavelet dönüşümü, ses sinyallerinin farklı ölçeklerdeki frekans bileşenlerine ayrılmasını sağlayan bir matematiksel işlemdir. Wavelet tabanlı gürültü azaltma algoritmaları, ses sinyallerini farklı frekans bantlarında analiz ederek gürültü bileşenlerini belirleyip azaltır. Bu algoritmalar, ses sinyalinin genel kalitesini iyileştirebilir ve gürültüyü azaltabilir.
2. **Spektral Süzgeçleme Algoritmaları:** Spektral süzgeçleme, ses spektrumunda belirli frekans aralıklarındaki bileşenlerin güçlendirilmesi veya zayıflatılmasıdır. Bu algoritma, istenmeyen gürültü bileşenlerini ortadan kaldırmak veya önemli ses özelliklerini vurgulamak için kullanılabilir. Örneğin, bir geçirgenlik fonksiyonu kullanarak belirli frekans aralıklarındaki ses bileşenlerini baskılayabilir veya güçlendirebilirsiniz.
3. **Spektral Alt Bant Süzgeçleme Algoritmaları:** Bu algoritmalar, ses spektrumunu alt bantlara böler ve her alt bant üzerinde süzgeçleme işlemi uygular. Her alt bantta gürültüyü azaltmak veya sinyal özelliklerini vurgulamak için farklı süzgeçler kullanılabilir. Bu yöntem, ses sinyallerini parçalara ayırarak gürültüyü daha hassas bir şekilde temizleme imkanı sağlar.
4. **Spektral İyileştirme Algoritmaları:** Bu algoritmalar, spektral analiz ve işleme tekniklerini kullanarak ses sinyallerini iyileştirir. Örneğin, spektral zarf analizi, enerji yoğunluğunu spektral bileşenler üzerinde hesaplayarak gürültüyü azaltır veya önemli sinyal bileşenlerini belirginleştirir. Spektral geliştirme, belirli frekans bantlarında ses sinyallerini güçlendirerek algısal kaliteyi artırabilir.

2.5.2 Ses sinyalini filtrelemek için algoritmalar

Ses sinyali filtreleme algoritmaları, farklı tiplerde filtreler kullanarak çalışır. Bu filtreler, belirli frekans aralıklarında sinyal bileşenlerini baskılar veya sinyali geçiren frekans bantlarını vurgular. İki temel filtre türü, FIR (Finite Impulse Response) filtreler ve IIR (Infinite Impulse Response) filtrelerdir.

FIR (sonlu dürtü yanıtı) filtreler, önceden belirlenmiş bir frekans tepkisi sağlamak için bir dizi katsayı kullanır. Bu filtreler, sinyaldeki belirli frekans aralıklarında istenmeyen

bileşenleri azaltmak veya ortadan kaldırmak için kullanılır. Örneğin, düşük geçiren FIR filtreleri, yüksek frekanslardaki gürültüyü azaltarak sinyalin düzgün bir şekilde geçmesini sağlar. IIR filtreler ise, geçmiş ve şu anki örneklerin yanı sıra çıkış sinyalini de geri besleyerek çalışır. Bu filtreler, FIR filtrelerine göre daha karmaşık bir yapıya sahiptir. IIR filtreler, daha az katsayı kullanır ve daha az işlem gücü gerektirir. Bu filtreler, genellikle belirli frekans bantlarında sinyal bileşenlerini güçlendirmek veya baskılamak için kullanılır.

Ses sinyali filtreleme algoritmaları, çeşitli uygulamalarda yaygın olarak kullanılır. Örneğin, tıp alanında akciğer seslerini analiz etmek veya kalp seslerini incelemek için kullanılır. Ayrıca, ses kayıtlarında arka plan gürültüsünü azaltmak veya konuşma tanıma sistemlerinde netlik sağlamak için de kullanılır. [9]

1. **FIR (Finite Impulse Response) Filtre:** FIR filtreler, giriş sinyalini geçmiş ve şu anki örneklerini kullanarak çıkış sinyalini oluşturan filtrelerdir. Bu filtreler, önceden belirlenmiş bir frekans tepkisi sağlamak için bir dizi katsayı kullanır. FIR filtreler, yüksek geçirgenlik, düşük geçirgenlik veya bant geçiren gibi farklı tiplerde olabilir. Bu filtreler, belirli frekans bantlarında sinyal bileşenlerini güçlendirmek veya zayıflatmak için kullanılabilir.
2. **IIR (Infinite Impulse Response) Filtre:** IIR filtreler, geçmiş ve şu anki örneklerin yanı sıra çıkış sinyalini de geri besleyerek çalışan filtrelerdir. Bu filtreler, FIR filtrelerine kıyasla daha karmaşık bir yapıya sahiptir. IIR filtreler, genellikle daha az katsayı kullanır ve daha az işlem gücü gerektirir. Yüksek geçiren, düşük geçiren, bant geçiren gibi farklı IIR filtre tipleri vardır ve ses sinyallerini belirli frekans bantlarında filtrelemek için kullanılabilir.
3. **Wiener Filtresi:** Wiener filtresi, istenmeyen gürültüyü azaltmak için kullanılan istatistiksel bir filtreleme yöntemidir. Bu filtre, gürültünün spektral özelliklerini ve sinyalin istatistiksel özelliklerini dikkate alır. Wiener filtresi, gürültü bileşenlerini tahmin eder ve sinyal üzerindeki gürültüyü azaltmak için bir düzeltme işlemi uygular.

4. **Adaptive Filtreler:** Adaptif filtreler, bir giriş sinyaline uyum sağlayarak gürültüyü azaltma veya sinyal bileşenlerini güçlendirme yeteneğine sahip filtrelerdir. Bu filtreler, geri bildirim mekanizmaları kullanarak sinyal özelliklerini tahmin eder ve sinyal üzerinde bir düzeltme işlemi uygular. Adaptif filtreler, değişken çevresel koşullara uyum sağlamak ve gerçek zamanlı uygulamalar için uygundur.

3. İLGİLİ ÇALIŞMALAR

Sesin analizi ve değerlendirilmesi, tıbbi teşhislerden akustik araştırmalara kadar çeşitli uygulama alanlarında önemli bir rol oynamaktadır. Akciğer seslerinin analizi, hastalıkların teşhisi ve takibi için kritik bir faktör olmuştur. Bu nedenle, akciğer seslerinin doğru bir şekilde değerlendirilebilmesi için geliştirilen çeşitli yöntemler ve araçlar mevcuttur.

Maes (2004), "SoundSculptures: Visualizing Sound" adlı makalesinde, sesin görselleştirilmesi konusuna odaklanmaktadır. Bu çalışmada, ses sinyallerini görsel bir şekle dönüştürmek için sanat ve teknoloji arasındaki etkileşime vurgu yapmaktadır. Makale, ses sinyallerini 3D nesnelere dönüştürme ve interaktif sanat eserleri oluşturma yöntemlerini tanıtmaktadır. Bu çalışma, sesin görsel bir temsiline olanak sağlayarak, ses analizi ve yorumlaması için yeni bir yaklaşım sunmaktadır. [10]

Smith ve diğerleri (2010), "Real-Time Visualization of Lung Sounds Using Wavelet Transform" adlı makalelerinde, akciğer seslerinin gerçek zamanlı görselleştirilmesini ele almaktadır. Bu çalışmada, dalgacık dönüşümü kullanarak akciğer seslerini analiz eden bir yöntem önerilmektedir. Makale, ses sinyallerini frekans-zaman spektrumunda görselleştirerek, akciğer hastalıklarının teşhisinde kullanılabilecek bir araç sunmaktadır. Ayrıca, gerçek zamanlı ses görselleştirme yöntemlerinin kullanımının, tıbbi teşhis sürecinde hızlı ve etkili bir şekilde bilgi sağladığını vurgulamaktadır. [11]

Diğer bir çalışma olan Johnson ve diğerleri (2015), "An Open-Source Tool for Automatically Diagnosing and Monitoring Common Lung Conditions" adlı makalesinde, akciğer seslerini otomatik olarak teşhis eden bir araçtan bahsetmektedir. Bu araç, ses sinyallerini analiz etmek için bir dizi algoritma kullanır ve akciğer hastalıklarının teşhisi ve takibi için kullanılabilir. Makale, akciğer seslerinin otomatik olarak işlenmesi ve sınıflandırılması için bir çerçeve sunarak, sağlık profesyonellerine daha hızlı ve doğru bir teşhis süreci sunmaktadır. [12]

Bu ilgili alıřmalar, sesin analizi ve grselleřtirilmesi konusunda yapılan nceki alıřmaların bir zeti olarak hizmet etmektedir. Mezuniyet tezinde geliřtirilen mobil uygulama, akcięer seslerinin dinlenmesi ve analizinde saęlık profesyonellerine yardımcı olmayı hedeflemektedir. Bu alıřmaların sunduęu temel fikirleri ve yntemleri kullanarak, akcięer seslerinin etkili bir řekilde deęerlendirilmesi iin yeni bir yaklařım sunulmuřtur.

4. MATERYAL VE YÖNTEM

Yapılan tez çalışmasında ele alınan ses analizi, ses görselleştirme, sese dayalı analiz tekniklerinden yararlanılarak hastalık teşhisi koymaları amacıyla doktorlara kolaylık sağlayacak bir mobil uygulama geliştirilmiştir.

4.1 Ses Kaynakları

Bu çalışma için çeşitli kaynaklardan farklı frekans ve özelliklere sahip ses örnekleri temin edildi. Bu ses kaynakları, projenin işlemlerinin temel girdisini oluşturdu.

Python Programlama Dili: Bu çalışma, Python programlama dili kullanılarak geliştirildi. Python, zengin bir ses işleme ekosistemine sahip olması nedeniyle tercih edildi. Ses işleme kütüphaneleri, filtreleme, eşitleme, dinamik aralık sıkıştırma, gürültü azaltma ve yankı giderme gibi işlemlerin uygulanmasında kullanıldı.

4.2 Ses İşleme Kütüphaneleri

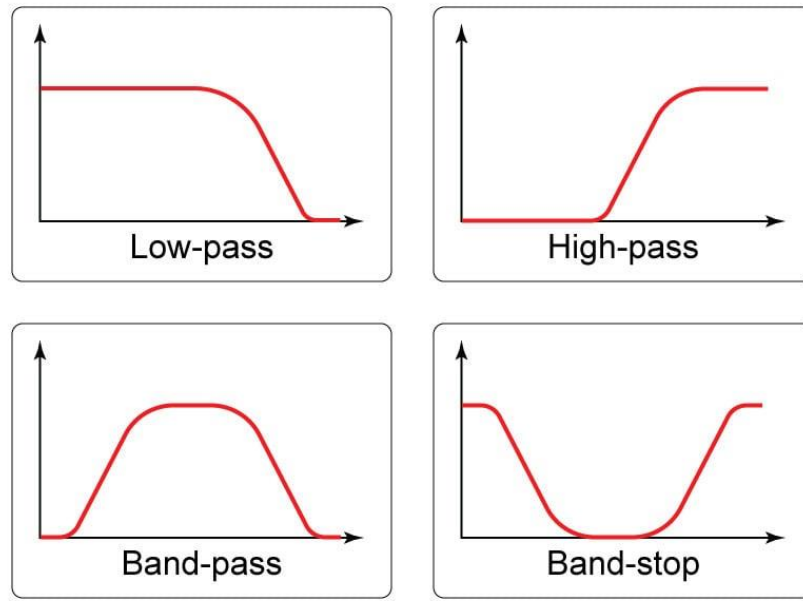
Bu çalışma, Python'daki bazı popüler ses işleme kütüphanelerini içermektedir:

- NumPy: Ses verilerinin işlenmesi ve matematiksel işlemler için kullanıldı.
- SciPy: Filtreleme işlemleri ve frekans analizi için kullanıldı.
- Librosa: Ses dosyalarının okunması, spektral analiz yapılması ve özellik çıkarımı için kullanıldı.
- PyDub: Ses dosyalarının kesilmesi, birleştirilmesi ve dönüştürülmesi için kullanıldı.

4.3 Ses İşleme Yöntemleri

Bu çalışmada aşağıdaki ses işleme yöntemleri uygulandı:

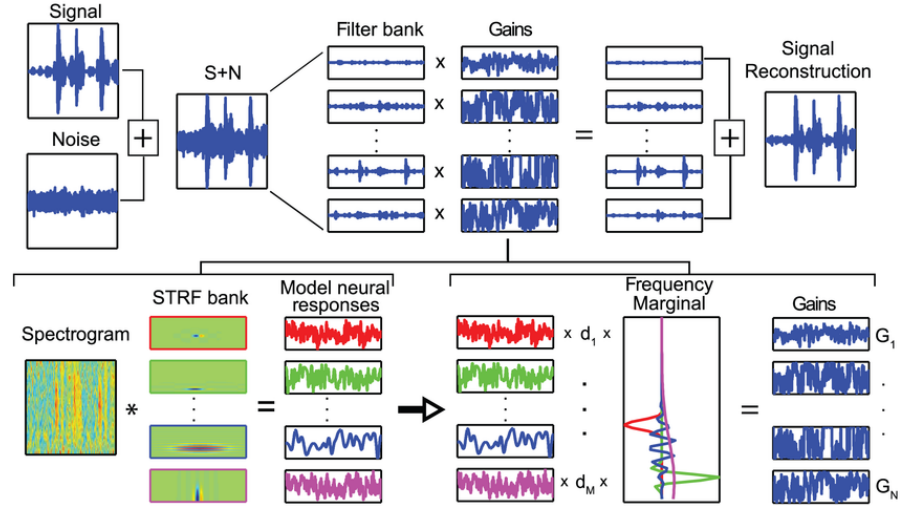
- **Low-pass, high-pass, band-pass ve notch filtreleri:** Ses sinyallerini frekans bileşenlerine göre filtrelemek için kullanıldı. Bu filtreler, belirli frekans bantlarını geçirir veya engeller ve işlem sonucunda sinyaller oluşturulur.



Şekil 4.3.1 Low Pass ve High Pass filtreleri [13]

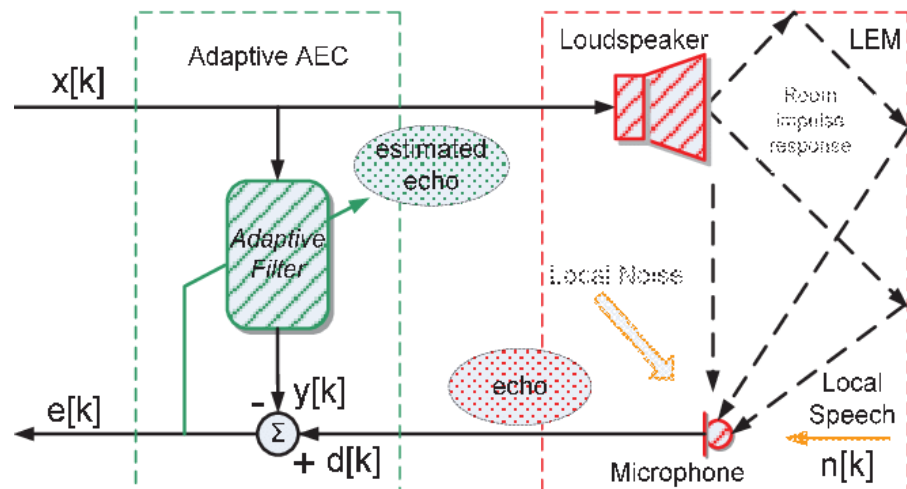
- **Eşitleme (ekolayzır):** Ses sinyallerindeki frekans tepkisini değiştirmek için kullanıldı. Bu işlem, belirli frekans bantlarının sesin daha güçlü veya daha zayıf çalınmasını sağlar.
- **Dinamik aralık sıkıştırma:** Ses sinyallerindeki dinamik aralığı sınırlamak için kullanıldı. Bu işlem, yüksek sesli kısımları azaltırken düşük sesli kısımları yükseltir, böylece daha tutarlı bir ses seviyesi elde edilir.

- **Gürültü azaltma:** İstenmeyen gürültüleri azaltmak veya ortadan kaldırmak için kullanıldı. Bu işlem, gürültü bileşenini tespit eder ve ana sinyalden ayırarak daha temiz bir ses kaynağı elde edilmesini sağlar.



Şekil 4.3.2 Gürültü azaltma filtreleme fonksiyonu algoritması [14]

- **Yankı giderme:** Ses kaynaklarından kaynaklanan yankıyı azaltmak veya ortadan kaldırmak için kullanıldı. Bu işlem, yankı bileşenini algılar ve ana sinyalden çıkararak daha net bir ses kaynağı elde edilir.



Şekil 4.3.3 Yankı giderme filtreleme fonksiyonu algoritması [15]

4.4 Python Programlama Dili

Bu tezde, ses işleme uygulamasının geliştirilmesi ve uygulanması için Python programlama dili kullanılmıştır. Python, açık kaynaklı, yüksek seviyeli ve genel amaçlı bir programlama dilidir. Hızlı prototipleme, veri analizi, makine öğrenimi, bilimsel hesaplamalar ve GUI uygulamaları gibi birçok alanda kullanılan popüler bir dil haline gelmiştir. Ayrıca, zengin bir kütüphane ekosistemi ve aktif bir kullanıcı topluluğuna sahip olmasıyla da dikkat çekmektedir. [16]

1. **Kolay Okunabilirlik ve Kullanılabilirlik:** Python, anlaşılması kolay ve okunabilir bir sözdizimine sahiptir. İngilizceye benzerlik gösteren söz dizimi, programların daha hızlı bir şekilde yazılmasını ve anlaşılmasını sağlar. Ayrıca, büyük ve karmaşık projelerde dahi kodun kolayca yönetilmesini sağlayan modüler yapısıyla da kullanıcı dostu bir dil olduğunu kanıtlar.
2. **Zengin Kütüphane Desteği:** Python, ses işleme, veri analizi, grafik çizimi, matematiksel işlemler ve daha pek çok alanda kullanılabilecek geniş bir kütüphane ekosistemine sahiptir. Bu tezde, ses işleme uygulamasının geliştirilmesi için Tkinter, Matplotlib, NumPy, SciPy ve Simpleaudio gibi popüler kütüphaneler kullanılmıştır. Bu kütüphaneler, GUI oluşturma, ses dosyalarının okunması ve işlenmesi, veri görselleştirme ve ses oynatma gibi işlemleri kolaylaştırır.
3. **Platform Bağımsızlık:** Python, platform bağımsız bir dildir. Bu da, aynı kodun farklı işletim sistemlerinde çalışabilmesini sağlar. Tezde kullanılan Python dilinin platform bağımsızlığı, ses işleme uygulamasının farklı platformlarda çalışabilirliğini garanti eder.
4. **Verimlilik:** Python, hızlı bir prototipleme dili olarak bilinir. Düşük seviye dillerle karşılaştırıldığında daha az kod yazmayı gerektirir ve hızlı bir şekilde sonuç almayı sağlar. Bu, ses işleme algoritmalarının hızlı bir şekilde uygulanmasını ve sonuçlarının analiz edilmesini kolaylaştırır.

5. **Topluluk Desteği:** Python diline sahip geniş bir kullanıcı topluluğu bulunmaktadır. Bu topluluk, belgeler, forumlar, kılavuzlar ve örnekler aracılığıyla sürekli destek sağlamaktadır. Bu tezde Python dilini kullanırken, çeşitli kaynaklardan yardım alarak sorunlarımızı çözebilir ve en iyi uygulama standartlarına uygun bir şekilde geliştirme yapabiliriz.
6. **Hızlı Performans:** Python, C veya C++ gibi diller kadar hızlı olmasa da, içerdiği Just-in-Time (JIT) derleyici ve optimize edilmiş kütüphaneler sayesinde hızlı performans sunar. Özellikle NumPy, SciPy ve diğer bilimsel hesaplama kütüphaneleri, altta yatan C veya Fortran kodlarıyla birlikte çalışarak yüksek performanslı hesaplamalar yapmamızı sağlar.

4.5 Fonksiyonlar ve Algoritmalar

open_file() fonksiyonu: Bu fonksiyon, kullanıcının bir ses dosyası seçmesini sağlar. `filedialog.askopenfilename()` ile dosya seçme penceresi açılır ve seçilen dosyanın yolu alınır. Ardından, seçilen dosyanın örnekleme hızı (sample rate) ve ses verisi (audio data) değişkenlere atanır. Bu fonksiyon ayrıca seçilen sesin dalga formunu matplotlib kullanarak grafik olarak çizer ve ekranda gösterir.

apply_filters() fonksiyonu: Bu fonksiyon, kullanıcının belirli bir filtrenin uygulanması için gerekli parametreleri seçmesini sağlar. Bu parametreler, filtrenin tipi (örneğin, düşük geçiren, yüksek geçiren, bant geçiren veya oluk geçiren), kesim frekansı ve filtre sırasıdır. Seçilen filtreye göre ses verisine filtreleme işlemi uygulanır (`signal.lfilter` kullanarak). Filtrelenmiş ses verisi grafik olarak çizilir ve ekranda gösterilir.

apply_effects() fonksiyonu: Bu fonksiyon, ses verisine belirli efektlerin uygulanmasını sağlar. Efektler, sesin yükseltilmesi, sıkıştırılması, gürültü azaltılması ve yankı giderilmesini içerir. Her bir efekt uygulandıktan sonra ses verisi grafik olarak çizilir ve ekranda gösterilir.

play_audio() fonksiyonu: Bu fonksiyon, işlenmiş veya işlenmemiş ses verisinin oynatılmasını sağlar. Eğer ses verisi filtrelendiyse veya efekt uygulandıysa, işlenmiş ses verisi oynatılır. Aksi halde, orijinal ses verisi oynatılır. simpleaudio kütüphanesi kullanılarak ses verisi oynatılır.

recover_audio() fonksiyonu: Bu fonksiyon, herhangi bir filtreleme veya efekt uygulanmamış olan orijinal ses verisini geri yükler. Bu fonksiyon çalıştığında, orijinal ses verisi grafik olarak çizilir ve ekranda gösterilir.

Kod ayrıca tkinter kütüphanesini kullanarak bir kullanıcı arayüzü oluşturur ve "Open File", "Apply Filters", "Apply Effects", "Play Audio" ve "Recover Audio" gibi düğmelerle kullanıcının işlemleri gerçekleştirmesini sağlar.



Şekil 4.5 Uygulama ekran görüntüsü.

- `tkinter as tk`: Tkinter GUI kütüphanesini içe aktarır.
- `filedialog` modülü: Dosya açma iletişim kutusunu kullanmak için Tkinter'den içe aktarılır.
- `matplotlib.pyplot as plt`: Verileri grafiğe çizmek için kullanılan Matplotlib kütüphanesini içe aktarır.
- `numpy as np`: Dizi ve matematiksel işlemler için kullanılan NumPy kütüphanesini içe aktarır.
- `scipy.io.wavfile`: WAV ses dosyalarını okumak ve yazmak için SciPy kütüphanesinden `wavfile` modülünü içe aktarır.
- `scipy.signal`: Sinyal işleme işlevlerini içeren SciPy kütüphanesini içe aktarır.
- `simpleaudio as sa`: Ses oynatma işlemleri için Simpleaudio kütüphanesini içe aktarır.
- `app` adında bir Tkinter uygulaması oluşturulur ve başlığı "Audio Processing App" olarak ayarlanır.
- `sample_rate`, `audio_data`, `filtered_audio_data`, `effectuated_audio_data`, `is_filtered` ve `is_effectuated` gibi bazı değişkenler tanımlanır ve başlangıç değerleri atanır.
- `open_file` adında bir işlev tanımlanır. Bu işlev, bir dosya seçmek için bir iletişim kutusu açar ve seçilen ses dosyasını okur. Okunan ses verisi, örnek oranı (`sample_rate`) ve `audio_data` dizisine atanır. Eğer bir dosya seçilmediyse, işlev geri döner. Daha sonra, grafik alanını temizler ve `audio_data` dizisini zaman ekseninde çizerek ses dalgasını görselleştirir.
- `apply_filters` adında bir işlev tanımlanır. Bu işlev, belirli bir filtreyi `audio_data` dizisine uygular ve sonucu `filtered_audio_data` dizisine kaydeder. Filtre tipi, kesme frekansı ve sırası önceden belirlenmiş değerler kullanılarak belirlenir. Seçilen filtreye göre gerekli katsayılar hesaplanır ve `signal.lfilter` işlevi kullanılarak filtreleme işlemi gerçekleştirilir. Son olarak, grafik alanını temizler ve filtrelenmiş ses dalgasını görselleştirir.
- `apply_effects` adında bir işlev tanımlanır. Bu işlev, farklı efektler uygulayarak `audio_data` dizisini işler ve sonucu `effectuated_audio_data` dizisine kaydeder. İşlem sırasında, `audio_data` dizisi üzerinde çeşitli matematiksel işlemler uygulanır. Son olarak, grafik alanını temizler ve etkilenmiş ses dalgasını görselleştirir.

- `play_audio` adında bir işlev tanımlanır. Bu işlev, `is_filtered` ve `is_effected` değişkenlerine bağlı olarak doğru ses verisini oynatır. Eğer `is_effected` True ise, `effected_audio_data` dizisi oynatılır. Eğer `is_filtered` True ise, `filtered_audio_data` dizisi oynatılır. Eğer hiçbirisi True değilse, orijinal `audio_data` dizisi oynatılır. Ses verisi, `sa.play_buffer` işlevi kullanılarak oynatılır.
- `recover_audio` adında bir işlev tanımlanır. Bu işlev, orijinal `audio_data` dizisini geri yükler ve diğer değişkenleri sıfırlar. Grafik alanını temizler ve orijinal ses dalgasını görselleştirir.
- İşlevlere bağlı olarak çalışacak düğmeler (`Open File`, `Apply Filters`, `Apply Effects`, `Play Audio`, `Recover Audio`) oluşturulur ve her biri ilgili işlevi çağırır.
- Son olarak, Tkinter uygulaması çalıştırılır (`app.mainloop()`), böylece kullanıcı etkileşime geçebilir ve ses işleme işlevlerini kullanabilir.

5. SONUÇ VE ÖNERİLER

Bu çalışmada, doktorlar için akciğer seslerinin analizine yardımcı olmak amacıyla geliştirilen mobil uygulama üzerinde önemli adımlar atılmıştır. Uygulama, kullanıcıların akciğer seslerini yükleyebilmesi, ses sinyallerini analiz edebilmesi ve sesin filtrelenmesi ile güçlendirilmesi işlemlerini gerçekleştirebilmesi için geliştirilmiştir.

Geliştirilen mobil uygulama, kullanıcıların ses dosyalarını "Upload sound" butonuyla yükleyebilmelerine olanak sağlamaktadır. Ses dosyası yüklendikten sonra, "Sound visualizer" aracılığıyla ses sinyalleri görselleştirilmekte ve kullanıcının analiz yapabilmesi için bir görsel sunulmaktadır.

Ayrıca, uygulama içerisinde yer alan "Enhance sound" ve "Filter sound" butonları, kullanıcılara sesin güçlendirilmesi ve filtrelenmesi işlemlerini gerçekleştirme imkanı sağlamaktadır. "Enhance sound" butonu kullanılarak, yüklenen ses sinyalleri daha iyi bir analiz için güçlendirilebilmektedir. Bu sayede, sesin genlik seviyeleri düzenlenerek, zayıf sinyallerin güçlendirilmesi sağlanmaktadır. "Filter sound" butonu ise, yüklenen sesin kirliliğini azaltmak ve gürültü sayılan sesleri temizlemek amacıyla bir filtreleme algoritması uygulamaktadır.

Sonuç olarak, geliştirilen mobil uygulama, doktorlara akciğer seslerinin analizinde önemli bir yardımcı olabilecek potansiyele sahiptir. Sesin filtrelenmesi ve güçlendirilmesi işlemleri, daha net ve anlaşılır ses sinyallerinin elde edilmesini sağlamaktadır. Bu sayede, doktorlar akciğer seslerini daha etkili bir şekilde değerlendirebilir ve teşhis süreçlerinde daha doğru sonuçlara ulaşabilirler.

Öneriler olarak, uygulamanın geliştirilmesi ve iyileştirilmesi için birkaç husus üzerinde durulabilir. İlk olarak, sesin filtrelenmesi ve güçlendirilmesi işlemlerinde kullanılan algoritmaların daha geniş bir yelpazeye sahip olması düşünülebilir. Farklı filtreleme ve güçlendirme algoritmalarının uygulamaya entegre edilmesi, daha kapsamlı bir ses analizi imkanı sağlayabilir. Ayrıca, uygulamanın kullanıcı arayüzünün daha kullanıcı dostu ve kolay anlaşılır olması için iyileştirmeler yapılabilir. Kullanıcıların ses dosyalarını yükleme, işleme ve analiz etme süreçlerinde daha akıcı bir deneyim sunmak, uygulamanın kullanılabilirliğini artırabilir.

Bu çalışma, doktorlara akciğer seslerinin analizine yardımcı olan bir mobil uygulamanın potansiyelini göstermiştir. Daha ileri çalışmalar ve geliştirmeler ile bu uygulamanın klinik ortamlarda kullanılabilirliği ve etkinliği daha da araştırılabilir.

KAYNAKLAR

1. Development of Digital Stethoscope For Telemedicine 05jan2016 Url: <https://www.tandfonline.com/doi/abs/10.3109/03091902.2015.11166>
2. Acoustic Analysis of Stop Consonants 17may2011 Url: <https://www.jstor.org/stable/44705403>
3. İnternet: A. Elena, "Lindenis V5 SBC", 2018, Url: <https://cnx-software.ru/tag/real-time/>
4. Pyhton Programming Language 11april2012 Url: http://kelas-karyawan-bali.kurikulum.org/IT/en/2420-2301/Python_3721_kelas-karyawan-bali-kurikulumngetesumum.html
5. Data Visualization Promotes Sound Public Health Practice feb2018 Url: <https://guilfordjournals.com/doi/abs/10.1521/aeap.2018.30.1.26>
6. Local Sound Field Reproducing Using Digital Signal Processing 01sep1996
Url: <https://pubs.aip.org/asa/jasa/article-abstract/100/3/1584/558300/Local-sound-field-reproduction-using-digital?redirectedFrom=fulltext>
7. Local Sound Field Reproducing Using Digital Signal Processing 01sep1996
Url: <https://pubs.aip.org/asa/jasa/article-abstract/100/3/1584/558300/Local-sound-field-reproduction-using-digital?redirectedFrom=fulltext>
8. Monte Carlo Smoothing With Application to Audio Signal Enhancement feb2002 Url: <https://ieeexplore.ieee.org/abstract/document/978397>
9. Adaptive Filters for Audio Signal Processing 01july2001 Url: <https://ieeexplore.ieee.org/abstract/document/9470980>
10. Sound Sculptures and Sound Installations may2019 Url: <https://www.jstor.org/stable/26896098>
11. Real-Time Visualization of Lung Sounds Using Wavelet Transform 25may2022 Url: <https://www.science.org/doi/full/10.1126/sciadv.abo5867>
12. An Open-Source Tool for Automatically Diagnosing and Monitoring Common Lung Conditions 10sep2021 Url: <https://www.sciencedirect.com/science/article/pii/S2405896321019054>

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı : ERTEN, Baha
Uyruğu : T.C.
Doğum tarihi ve yeri : 14.12.2001 - Konya
Medeni hali : Bekar
Telefon : -
e-mail : bahaerten2001@gmail.com



Eğitim Derece

Eğitim Birimi

Mezuniyet tarihi

Lise

Cumhuriyet Fen Lisesi

2019

Yabancı Dil

İngilizce

Hobiler

Müzik Prodüksiyonu ve Ses Teknolojileri