



**T.C.**  
**KONYA TEKNİK ÜNİVERSİTESİ**  
**LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**



**TEMPOROMANDİBULAR EKLEM  
BOZUKLUKLARININ BELİRLENMESİNDE  
SİNYAL İŞLEME VE YAPAY ZEKÂ  
TEKNİKLERİ KULLANILMASI**

**Uğur TAŞKIRAN**

**DOKTORA TEZİ**

**Elektrik-Elektronik Mühendisliği Anabilim Dalı**

**Eylül-2019**  
**KONYA**  
**Her Hakkı Saklıdır**

## TEZ KABUL VE ONAYI

Uğur TAŞKIRAN tarafından hazırlanan “Temporomandibular Eklem Bozukluklarının Belirlenmesinde Sinyal İşleme ve Yapay Zekâ Tekniklerinin Kullanılması” adlı tez çalışması 12/09/2019 tarihinde aşağıdaki jüri tarafından oy birliği ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı’nda DOKTORA TEZİ olarak kabul edilmiştir.

### Jüri Üyeleri

#### Başkan

Prof. Dr. Serhan AKMAN

#### Danışman

Prof.Dr. Mehmet ÇUNKAŞ

#### Üye

Doç.Dr. Murat CEYLAN

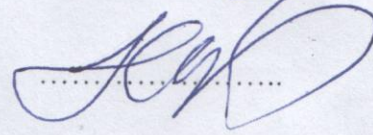
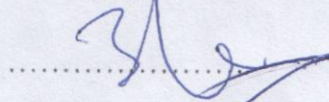
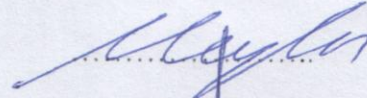
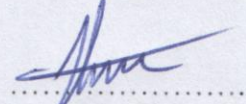
#### Üye

Doç.Dr. Bayram AKDEMİR

#### Üye

Doç.Dr.Hasan Erdiñç KOÇER

### İmza



Yukarıdaki sonucu onaylarım.

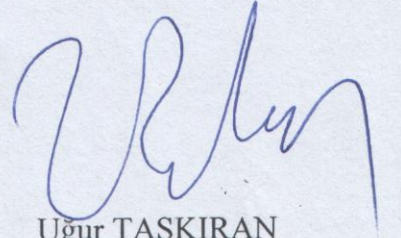
Prof. Dr. Hakan KARABÖRK  
Enstitü Müdürü

## TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

## DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.



Uğur TAŞKIRAN

Tarih: 12.09.2019

# ÖZET

## DOKTORA TEZİ

### TEMPOROMANDİBULAR EKLEM BOZUKLUKLARININ BELİRLENMESİNDE SİNYAL İŞLEME VE YAPAY ZEKÂ TEKNİKLERİ KULLANILMASI

Uğur TAŞKIRAN

Konya Teknik Üniversitesi  
Lisansüstü Eğitim Enstitüsü  
Elektrik Elektronik Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Mehmet ÇUNKAŞ

2019, 91 Sayfa

Jüri

Prof. Dr. Serhan AKMAN  
Prof. Dr. Mehmet ÇUNKAŞ  
Doç. Dr. Murat CEYLAN  
Doç. Dr. Bayram AKDEMİR  
Doç. Dr. Hasan Erdiñ KOÇER

Temporomandibular Eklem (TME) alt çene kemiği (mandibula) kondil ile temporal kemikteki mandibularfossa arasında meydana gelen bir eklemdir. TME çok karmaşık hareketler yapabilmeye kabiliyetine sahiptir. Bu eklemden meydana gelen her türlü bozukluğa Temporomandibular Bozukluk (TMB) denir. TMB oldukça yaygın bir rahatsızlık olup genel nüfusun %75 inde görülmektedir. TMB teşhisinin klasik yöntemlerinden biri hekimin hasta bireyin klinik muayenesi esnasında eklem seslerinin stetoskop vasıtasıyla dinlemesidir. Bu sesler; krepitasyon, kliking ve popping olarak üç sınıfa ayrılmıştır. Bu çalışmada yapay zekâ yöntemleriyle TME seslerini işleyerek sağlıklı ve hasta olarak sınıflandıran bir yöntem geliştirilmiştir. İlk aşamada seslerin kaydı için girişimsel olmayan (non-invasive) bir cihaz tasarlanmış ve hasta ve sağlıklı bireylere ait sesler kaydedilmiştir. Seslerin gürültüden arındırılması ve bilgi içermeyen bölümlerin atılması için ses verilerine sayısal sinyal işleme uygulanmış ve daha sonra her bir veriye ait 100 adet frekans tabanlı özellik çıkarımı yapılmıştır. Bu veriler Yapay Sinir Ağı (YSA) ile sınıflandırılmış ve %78 civarında başarı oranı elde edilmiştir. İkinci aşamada ses veri dizisinden istatistiksel veriler çıkartılmıştır. Bu özellikler kullanılarak YSA ile sınıflandırma yapılmış ve %87-89 civarında ortalama bir başarı elde edilmiştir. Üçüncü aşamada ise derin öğrenme yöntemi kullanılarak sınıflandırma yapılmıştır. Seçilen verilere bir bant geçiren filtre uygulanmış, herhangi bir bilgi taşımayan yüksek frekans bant aralıkları örnekleme frekansı düşürülerek atılmıştır. Derin öğrenme ağlarının resim sınıflamadaki başarısı bilindiğinden işlenen veriler kısa zamanlı Fourier dönüşümü tabanlı spektrogram resim verilerine çevrilmiştir. Bu resim verilerine derin öğrenme algoritması uygulanmıştır. Parametre ve ağ yapısı ayarları yapılarak ağın tutarlılığı artırılmıştır. Derin öğrenme ağının sınıflandırma başarısı tutarlı olarak %90'nın üzerine çıkarılmıştır. Sonuçlar karşılaştırıldığında derin öğrenmeye dayalı sınıflandırma yönteminin önceki iki yönteme göre daha başarılı olduğu görülmüştür.

**Anahtar Kelimeler:** Derin öğrenme, frekans tabanlı özellik çıkarma, istatistiksel tabanlı özellik çıkarma, sayısal sinyal işleme, temporomandibular eklem, temporomandibular eklem bozukluğu, yapay sinir ağları

## ABSTRACT

### PhD THESIS

# DETERMINATION OF TEMPOROMANDIBULAR JOINT DISORDER BY USING SIGNAL PROCESSING AND ARTIFICIAL INTELLIGENCE TECHNIQUES

Uğur TAŞKIRAN

Konya Technical University  
Institute of Graduate Studies  
Department of Electrical and Electronics Engineering

Advisor: Prof. Dr. Mehmet ÇUNKAŞ

2019, 91 Pages

#### Jury

Prof. Dr. Serhan AKMAN  
Prof. Dr. Mehmet ÇUNKAŞ  
Assoc. Prof. Dr. Murat CEYLAN  
Assoc. Prof. Dr. Bayram AKDEMİR  
Assoc. Prof. Dr. Hasan Erdiñ KOÇER

Tempromandibular Joint (TMJ) is the joint between lower jaw bone (mandible) condyle and mandibular fossa of temporal bone. TMJ has ability to perform complex movements. Any health problem arising from TMJ is called Temporomandibular Joint Disorder (TMD). TMD is a frequent health problem and %75 of the population may suffer from some degree of TMD. One of the classical diagnose method of TMD is listening TMJ sound during the clinical examination of the patient by the dentist using a stethoscope. TMD sound are grouped into three main categories known as crepitation, clicking and popping. In this study, a method is developed to classify the TMJ sounds as healthy and not-healthy by using artificial intelligence techniques. First a non-invasive device is designed and TMJ sounds of healthy and non-healthy people are recorded. In first phase, to remove noise and insignificant parts digital signal processing is applied to sound data and then 100 frequency based features are extracted from each data set. Data is classified by an Artificial Neural Network (ANN) and a success rate of around 78% is obtained. In second phase, statistical features are extracted from sound data set. Extracted features are used to classify the sound data set by means of ANN. A success rate of around %87 to 89 is obtained. In the third phase deep learning methods are used for classification. A band pass filter is applied to chosen data and insignificant parts of the higher frequency parts are removed by decreasing sampling rate. Known success of deep learning methods classifying picture data direct the transforming of sound data to spectrogram picture data by using Short Time Fourier Transform. Deep learning algorithms are applied to picture data. Parameter and network structure adjustments are made to increase the network consistency. The success rate of deep learning algorithm is regularly increased over 90%. If the results are compared, it is observed that classification method based on deep learning is more successful then previous two methods.

**Keywords:** artificial neural networks, deep learning, digital signal processing, frequency based feature extraction, statistical based feature extraction, temporomandibular joint, temporomandibular joint disorder,

## ÖNSÖZ

Bu çalışmanın yürütülmesi sırasında yardım ve desteğini esirgemeyen değerli hocam ve danışmanım Prof. Dr. Mehmet ÇUNKAŞ'a, tez izleme komitesinde yer alan ve tavsiyeleriyle bana yardımcı olan hocalarım Prof. Dr. Serhan AKMAN ve Doç. Dr. Murat CEYLAN'a, doktora çalışma süreç boyunca yardımlarını esirgemeyen Prof. Dr. Saadettin HERDEM ve Prof. Dr. Filiz AYKENT'e, Diş Hekimliği Fakültesinde verilerin toplanmasında yardımcı olan Dr. Öğretim Üyesi Tuba SAVAŞ YILMAZ'a, yapay zekâ ve derin öğrenme süreçleri ile ilgili karşılıklı bilgi alışverişinde bulunduğum Doç. Dr. Hasan Erdinç KOÇER'e teşekkürlerimi sunarım.

Yoğun çalışmalarım esnasında her türlü desteğini eksik etmeyen sevgili eşim Nurhan TAŞKIRAN'a, bana YSA ve derin öğrenme konularında yeni bir ufuk açan kızım Salimkan Fatma TAŞKIRAN'a, sevimliliği ve güler yüzüyle bana daima neşe kaynağı olan kızım Nilüfer Canan TAŞKIRAN'a sonsuz teşekkür ederim.

Uğur TAŞKIRAN  
KONYA-2019

## İÇİNDEKİLER

|   |            |
|---|------------|
| <b>ÖZET .....</b>   | <b>iv</b>  |
| <b>ABSTRACT.....</b>  | <b>v</b>   |
| <b>ÖNSÖZ .....</b>  | <b>vi</b>  |
| <b>İÇİNDEKİLER .....</b>  | <b>vii</b> |
| <b>SİMGELER VE KISALTMALAR.....</b>   | <b>ix</b>  |
| <b>1. GİRİŞ .....</b>   | <b>1</b>   |
| 1.1. TMB Probleminin Önemi, Yaygınlığı ve Semptomları.....  | 1          |
| 1.2. TMB Semptomları ve Eklem Sesleri Sınıflandırması .....   | 2          |
| 1.3. Tezin Amaç ve Önemi .....  | 2          |
| 1.4. Tezin Organizasyonu .....  | 4          |
| <b>2. KAYNAK ARAŞTIRMASI .....</b>  | <b>6</b>   |
| 2.1. Genel TMB ve Hastalığın Popülasyonda Görülme Sıklığı.....  | 6          |
| 2.2. Frekans Analizine Dayanan Sınıflandırma Çalışmaları.....   | 6          |
| 2.3. İstatistik Özellik Çıkarımına Dayalı Sınıflandırma Çalışmaları.....  | 9          |
| 2.4. Derin Öğrenmeye Dayalı Sınıflandırma Çalışmaları.....  | 10         |
| 2.5. Literatürün Genel Değerlendirmesi .....  | 12         |
| <b>3. MATERYAL VE YÖNTEM.....</b>   | <b>14</b>  |
| 3.1. TME Ses kayıtlarını Yapmak için Girişimsel Olmayan (non-invasive) Cihaz<br>Tasarımı ve Gerçeklenmesi ..... | 15         |
| 3.2. Zamanda Örneklenmiş Sinyalin Frekans ve Zaman-Frekans Bileşenleri.....                                     | 23         |
| 3.2.1. Ayrık Fourier Dönüşümü ve Hızlı Fourier Dönüşümü (DFT ve FFT).....                                       | 23         |
| 3.2.2. Zaman Frekans Dönüşümü - Spektrogram .....   | 24         |
| 3.3. Bir dizinin istatistiksel özellikleri .....  | 24         |
| 3.4. Yapay Sinir Ağları .....   | 26         |
| 3.4.1. Giriş .....  | 26         |
| 3.4.2. Yapay sinir ağlarında öğrenme .....  | 28         |
| 3.5. Derin Öğrenme ve Konvolüsyon Sinir Ağları .....  | 29         |
| 3.5.1. Giriş .....  | 29         |
| 3.5.2. Konvolüsyon sinir ağları.....  | 30         |
| 3.5.3. Derin ağ katmanları ve özellikleri .....   | 32         |
| <b>4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....</b>  | <b>36</b>  |
| 4.1. Ses Verilerinin Toplanması ve Kayıt Altına Alınması .....  | 36         |

|  |           |
|--|-----------|
| 4.2. Ses Verilerinin Frekans İçeriğine Dayalı Özellik Çıkarımı.....                    | 42        |
| 4.3. Ses Verilerinin YSA Kullanarak Sınıflandırılması.....                             | 47        |
| 4.4. İstatistiksel Tabanlı Özelliklerin Çıkarımı ve YSA ile Sınıflandırılması .....    | 49        |
| 4.4.1. Ses verilerinin istatistiksel özellik çıkarımı .....                            | 49        |
| 4.4.2. İstatistiksel özellik çıkarımı yapılan verilerinin YSA ile sınıflandırılması .. | 50        |
| 4.5. Derin Öğrenme Tabanlı Sınıflandırma .....   | 53        |
| 4.5.1. Kullanılan verilerin seçimi.....  | 54        |
| 4.5.2. Verilerin işlenmesi ve sınıflandırılması.....                                   | 56        |
| 4.4. Sınıflandırma Yöntemlerinin Karşılaştırılması.....                                | 61        |
| <b>5. SONUÇLAR VE ÖNERİLER .....</b>   | <b>65</b> |
| 5.1 Sonuçlar .....   | 65        |
| 5.2 Öneriler .....   | 67        |
| <b>KAYNAKLAR .....</b>   | <b>69</b> |
| <b>EKLER .....</b>   | <b>73</b> |



## SİMGELER VE KISALTMALAR

### Simgeler

Hz: Hertz, frekans birimi, 1 saniyedeki tekrar sayısı.

KHz: KiloHertz=  $10^3$  Hz

sn: Saniye

msn: milisaniye=  $10^{-3}$  sn

### Kısaltmalar

AGF: Alçak Geçiren Filtre

AVR: Aort Valv Replasmanı

BGF: Bant Geçiren Filtre

CNN: Convolutional Neural Networks, Evrişimsel/Konvolüsyon Sinir Ağları

DFT: Discrete Fourier Transform, Ayrık Fourier Dönüşümü

DSP: Digital Signal Processing, Sayısal Sinyal İşleme

ESA: Evrişimsel/Konvolüsyon Sinir Ağları

FFT: Fast Fourier Transform, Hızlı Fourier Dönüşümü

HMM: Hidden Markov Models, Saklı Markov Modelleri

MR: Manyetik Rezonans

MVR: Mitral Valv Replasmanı

RID: Reduced Interference Distribution, Azaltılmış Enterferans/Karışım Dağılımı

TMB: Temporomandibular Bozukluk

TME: Temporomandibular Eklem

TMD: Temporomandibular Disorder, Temporomandibular Eklem Bozukluğu

TMJ: Temporomandibular Joint, Temporomandibular Eklem

YSA: Yapay Sinir Ağları

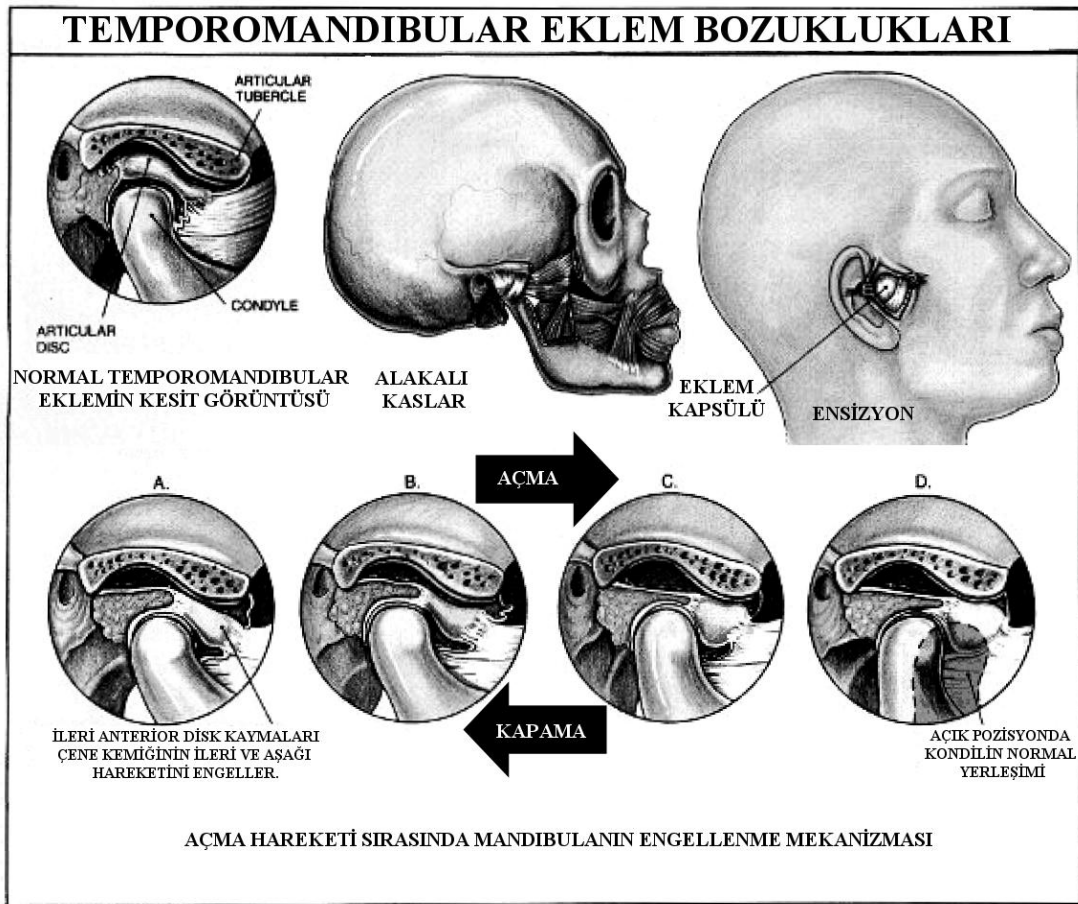
SNR: Signal to Noise Ratio, Sinyal Gürültü Oranı

SVM: Support Vector Machines, Destek Vektör Makineleri

RGB: Red-Green-Blue, Kırmızı Yeşil Mavi ana renkler

## 1. GİRİŞ

Temporomandibular eklem (TME) alt çene kemiğindeki (mandibula) kondil ile temporal kemikteki mandibular fossa arasında meydana gelen bir eklemdir. Bu iki kemik arasında artiküler disk vardır. Bu nedenle TME çok karmaşık hareketler yapabilme kabiliyetine sahiptir. TME’de meydana gelen her türlü bozukluğa Temporomandibular Bozukluk (Temporomandibular Disorder) denir. Şekil 1.1’de TME yapısı ve bozukluklarının grafiksel bir gösterimi vardır.



Şekil 1.1 Temporomandibular eklem ve eklem bozukluğu mekanizması.

### 1.1. TMB Probleminin Önemi, Yaygınlığı ve Semptomları

TME ve ilgili bozukluklar genel popülasyonda oldukça yaygındır. Araştırmalara göre TME rahatsızlığı %75 oranında yaygınlık gösterirken, popülasyonun en az %33’ü TME bozukluk semptomlarından birini göstermektedir (Mc Neil 1993; Okeson 1985). Bu oran ülkemizde de çok farklı olmayıp, Özan ve ark. (2007) ’nın araştırmalarına göre

daha da fazladır. Hastalığın semptomlarının başında gelen ağrı şikâyeti çoğu zaman yansıyan ağrılar şeklinde görülebilir. Hastanın yanlış algılamasından kaynaklanan baş ağrıları (genellikle migreni andıran), kulak ağrıları ve gözlerde ağrı ile karıştırılabilir. Ağrılara ilaveten ağzın açılması veya kapatılması esnasında ses, çene hareketlerinde kısıtlanması ve çenenin yana doğru kaymaları (deviasyon ve defleksiyon) diğer semptomlar arasındadır. Ağrılara ilaveten ağzınızı açtığınızda veya kapadığınızda tıklama veya çıtlama sesi, esneme, ağzınızı geniş bir şekilde açma veya çiğneme esnasında ağrı, “takılan”, “kilitlenen” çene, çene kaslarında hassasiyet, üst ve alt dişlerimizin birleşme şeklinde ani değişiklik vs. diğer semptomlar arasında sayılabilir (Mc Neil 1993).

### **1.2. TMB Semptomları ve Eklem Sesleri Sınıflandırması**

Temporomandibular eklem (TME) bozuklukları popülasyonun yaklaşık %75’inde görülen yaygın bir rahatsızlıktır. Bu rahatsızlığın sebebi çoğu zaman net olarak tespit edilemez. Hastalığın semptomları arasında kulak ağrısı, göz ağrısı ve migren ağrısı şeklinde yansıyan ağrılar, konuşma ve çiğneme esnasında gelen eklem sesleri, çene hareketlerinin kısıtlanması ve çenede ortaya çıkan deviasyon ve defleksiyon kaymaları sayılabilir. Yine rahatsızlığın belirlenmesinde hekimin klinik olarak eklem muayenesinde eklemdeki seslerin tespiti TME rahatsızlığının belirlenmesinde önemli bir bulgudur. Eklemde görülen sesler krepitasyon, kliking veya popping kliking şeklinde olabilir. Ancak hekimin bu sesleri tam teşhis edememesi ya da ortamdaki başka sesler ile karıştırması mümkün olabilmektedir.

### **1.3. Tezin Amaç ve Önemi**

TME rahatsızlıklarında ortaya çıkan semptomlar arasında yer alan ağrı, eklem sesi, eklemde kısıtlılık ve çenenin deviasyonu ya da defleksiyonu hekimin rahatsızlığı tespit etmesindeki önemli bulgulardır. Klinikteki muayene sonucunda tespit edilen bu bulgular TME’in radyografik görüntüleri hastalığın tespitinde hekime yardımcıdır. Ancak TME’in ve özellikle de diskin görüntüsü ve yapılarında meydana gelen deformasyonlar her radyolojik görüntülerde net olarak görülmez. Bu konuda manyetik rezonans (MR) görüntüleri eklemde net olarak görülmesine yardımcı olurlar. Ancak MR yöntemi pahalıdır ve zaman alıcı bir yöntemdir. Ulaşılması her bölgede kısa zamanda

mümkün olmayabilir. Bu çalışma TME seslerinden elde edilen ham verileri kullanarak bir kişinin TMD hastası olup olmadığını araştırmayı ve belirlemeyi amaçlamıştır. Ses tabanlı sınıflandırma, özellikle tıbbi teşhislerde olmak üzere pek çok alanda kullanılmaktadır. Hasta vücudundan gelen sesleri dinlemek pek çok tıbbi alanda ana teşhis metodu olarak yer alır. Ancak sesleri dinleyerek doğru tanı koymak pek çok kez tecrübeli kulaklara ihtiyaç duyar. Sonuç olarak bu çalışma, en azından, tıbbi personele hastalıkları teşhis ederken yardımcı olmayı hedeflemektedir.

TME rahatsızlığının teşhisinde ses sınıflandırmaları yapılmış ve literatürde bununla ilgili bazı sinyal işleme ve yapay zekâ teknikleri kullanılmıştır. Bu araştırmada bu tekniklere ilaveten farklı ses sinyali işleme yöntemleri ve yapay zekâ teknikleri kullanarak TME rahatsızlık derecesi sınıflandırmaları yapılmaya çalışılmıştır. En önemli katkılardan biri ise literatürde pek rastlanmayan derin öğrenme tabanlı konvolüsyon ağlarının TME ses verilerinin sınıflandırılması ve hastalık teşhisinde kullanılmasıdır.

Bunun için ilk başta TME seslerini klinik ortamda kaydetmek için bir cihaz geliştirilmiş, birçok hastadan ve sağlam gönüllülerden ölçümler alınmıştır. Daha sonra bu ölçümleri sınıflandıracak sinyal işleme ve yapay zekâ algoritmaları kullanılmıştır. Elde edilen sonuçlar karşılaştırılmıştır.

Sonuç olarak sinyal işleme ve yapay zekâ algoritmaları kullanarak ve insan kulağının duyma sınırının dışındaki frekans bileşenlerini ya da insan kulağının fark edemeyeceği ses değişikliklerini algılayarak hastalığın teşhisi sağlanmıştır. Eklem seslerinin net olarak kaydedilmesi tasarlanan cihaz sayesinde mümkün olmuştur. İlaveten cihazla birlikte bir ölçüm algoritması geliştirilmiş ve verilerin bilgisayara kaydedilmesi amacıyla kullanılmıştır. Bu araştırma sonuçlarıyla birlikte hastalığın tedavisi esnasında uygulanan tedavi yöntemlerinin (ilaç tedavisi ve/veya splint uygulaması) etkinliği konusunda hekime fikir verilebilecektir. Ayrıca ileride tedavi esnasında TME bozukluğundaki önemli semptomlar arasında yer alan eklemde duyulan seslerdeki azalmanın tespit edilebilme imkânı olacaktır. Sistemin en önemli özelliği ölçüm yapımını genellemesi yüzünden üzerinde yazılımsal birçok ayarlamamanın yapılmasına izin vermektedir. Bu durum, sistemin ileri versiyonlarının yazılımsal olarak gerçekleştirilmesine izin verecektir. Örnek olarak sistem yalnızca TME temel alınarak geliştirilmesine rağmen, teknolojsi benzer eklem rahatsızlarında da kullanılabilir şekilde uyarlanabilecek temel bilimsel alt yapıyı oluşturacaktır. Bu çalışma ilerisi için gömülü sisteme aktarılabilir ve/veya bilgisayara direk bağlanabilecek basit bir yapıya

dönüştürülebilirse yaygın olarak muayenehanelerde hekime yardımcı bir araç olarak kullanılabilir.

#### 1.4. Tezin Organizasyonu

Bu Tez çalışmasının ana hatları aşağıda özetlenmiştir.

Birinci bölüm tezin giriş bölümüdür. Bu bölüme TME'in fizyolojik yapısı hakkında kısa bir özet yapılarak başlanmıştır. TME nedir, işlevi, yapısı ve ortaya çıkan TMD'nin tanımı yapılmıştır. TMD probleminin sebepleri, teşhisi, semptomları gibi konular burada kısaca anlatılmıştır. TMD çeşitlerinden, eklem steteskop ile dinlenmesi esnasında ortaya çıkan seslerin sınıflandırılması ve rahatsızlıkla olan bağlantıları açıklanmıştır. Bu bölümde yine tezin önemi ve amacı anlatılmıştır ve tez organizasyonun bir özeti de eklenmiştir.

İkinci bölümde literatürde yer almış çalışmalara ait bir kaynak taraması yer almaktadır. Bu literatür özetinde konu ile doğrudan alakalı çalışmaların yanı sıra tez süreci esnasında yapılan tercih, tasarım, geliştirme süreçlerini dolaylı olarak etkileyen ve ilgilendiren çalışmalarda yer almıştır. Söz konusu yapılan çalışmalar TME ve yaygınlığı, frekans tabanlı sınıflandırma, istatistiksel tabanlı sınıflandırma ve derin öğrenme tabanlı sınıflandırma olarak dört alt başlıkta incelenmiştir.

Üçüncü bölüm ise materyal ve yöntemle ayrılmıştır. Bir giriş bölümünün ardından TME seslerinin kaydedilmesi için geliştirilen cihaz tanıtılmıştır. Bu cihazın tasarım aşamalarından bahsedilmiştir. Özellikle cihazın müdahalesiz yani non-invasive olması gerektiğine vurgu yapılmıştır. Cihazın teknik özellikleri ve gereksinimleri ve cihazın ses kaydı yapabilmesi için geliştirilen yazılımın özelliklerinden bahsedilmiştir. Takip eden bölümde kaydedilen seslerin sınıflandırılması için gerekli teknik ve uygulamaların özetleri ve uygulama biçimleri anlatılmıştır. Bir zamanda örneklenmiş sinyalin frekans bileşenlerinin elde edilmesi, Ayrık Fourier Dönüşümü ve Hızlı Fourier Dönüşümü, zaman frekans dönüşümü ve spektrogram kavramları açıklanmıştır. Bir dizinin istatistiksel özelliklerinin genel ve kısa bir özeti verilmiştir. Yapay sinir ağlarının bir özeti bu bölüme ilave edilmiştir. Bu bölümün en son konusu derin öğrenmedir. Burada derin öğrenme konusu geniş bir şekilde özetlenmiş ve derin öğrenme süreçlerinin verilerin sınıflandırılmasında kullanılabileceği anlatılmıştır.

Dördüncü bölümde sonuçlar sunulmuş ve yorumlanmıştır. İlk kısımda frekans tabanlı özellik çıkarımına dayanan YSA sınıflandırması incelenmiştir. Çıkan sonuçlar

rapor edilmiş ve yorum yapılmıştır. İkinci kısımda istatistiksel özellik çıkarımı ve YSA sınıflandırması incelenmiş, raporlanmış ve yorumlanmıştır. Son kısımda ise spektrogram çıkarım sürecine dayalı derin öğrenme ağları tartışılmıştır. Ortaya çıkan sonuçlar raporlanmış ve yorumlanmıştır.

Beşinci bölümde ise sonuçların geniş bir tartışması ve yorumu bulunmaktadır. İleri çalışmalar için öneriler de bu bölümde yer almaktadır. Bu bölümü takiben referanslar ve ekler bölümü yer almaktadır.



## 2. KAYNAK ARAŞTIRMASI

### 2.1. Genel TMB ve Hastalığın Popülasyonda Görülme Sıklığı

Editörlüğü Charles McNeil tarafından yapılan ve The American Academy of Orofacial Pain grubunun yayınladığı “Temporomandibular Disorders, Guidelines for Classification, Assessment and Management” (McNeil 1993) adlı kitap konunun temel fizyolojik kapsamı hakkındadır. Orofacial pain, ağız ve yüzle ilgili ağrılar anlamındadır. Genel anlamda TME bozuklukları üzerinedir. TME bozukluğunun tanımı, tanı tarihi, kas ve anatomisi, epidemiyolojisi, etiyolojisi, teşhis sınıflandırması, değerlendirilmesi ve yönetimi üzerine hazırlanmış oldukça ayrıntılı bir başvuru ve referans kitabıdır.

Özan ve ark. (2007) ise Türk popülasyonundaki TME bozukluk hastalarının yaygınlığını incelemiş ve araştırmışlardır. TME bozukluklarıyla alakalı olarak 15 ila 72 yaş arası 792 Türk için TME semptomları ve işaretlerini değerlendirilmiştir. Önceki çalışmalara benzer olarak anket ve fiziki muayene ile veriler toplanmıştır. En önemli sonuçlarından birisi TME bozukluklarının kadın nüfusunda erkek nüfusuna oranla daha yaygın olduğudur. Genel olarak Türk nüfusundaki TME rahatsızlık oranının diğerlerine göre daha fazla olduğu görülmüştür.

### 2.2. Frekans Analizine Dayanan Sınıflandırma Çalışmaları

Konunun en temel yayınlarından biri Wildmalm ve ark.(1996a) çalışmasıdır. Bu çalışmada yazarlar TME eklem tıklama ve çıtırdama seslerinin fiziksel büyüklüklerinin araştırmasını yapmışlardır. Bu bilgi doğal olarak tasarlanacak ve kullanılacak sayısal sistemler için önemlidir. Yapılan araştırma ilk önce 51 hasta ve 9 sağlam kişi üzerinde 44100Hz örnekleme frekansında yapılmıştır. Tıklama seslerinin bir tipi 2 ila 20 ms süreye sahiptir. Diğer bir tipi ise 1ms den bile az olup genel olarak 0-2ms arasında değişmektedir. Eğer örnekleme frekansı 3000 Hzden düşükse sinyalin ayrıntıları kaybolmaktadır. Bu iki tipin karakteristikleri de farklıdır. Kısa süreli olan tipi genelde çok yüksek bir perdeye sahiptir. Uzun süreli olan daha düşük bir perdeye sahiptir. Çıtırdama sesleri ise bir dizi kısa süreli seslerden (10ms den az) oluşmaktadır. Sonuç olarak mikروفon bant aralığı en az 20Hz ila 20KHz arasında ve 3000Hzden çok yüksek

bir frekansta örnekleme yapılmalıdır. Örnekleme hızı tercihen 10KHz üzerinde olmalıdır.

Wildmalm ve ark.(1996b) diğer bir çalışmalarında 98 hastanın TME ses kayıtlarını alarak stetoskop, hasta mülakatı yöntemi ile karşılaştırmıştır. Yeni geliştirilen yer ve enerji tepe sayısına bağlı karışımı azaltılmış dağılım metoduyla (Reduced Interference Distribution - RID) 5 tip sinyal ayrımı yapmışlar ve RID1 ile RID5 şeklinde sınıflandırılmışlardır. RID 1-3 tipi zamanda birbirine yakın birkaç enerji yükselmesinden oluşurken RID 4-5 ise tipik çıtırdama sesi olup zamanda birbirine yakın 20 – 300ms süren birçok enerji yükselmesinden oluşmaktadır. Sonuç olarak TME ses dalga sınıflandırılması RID'sinin gözlenmesinin dinlenmesine oranla daha detaylı bir sınıflandırma yapmaya olanak sağlamaktadır. Böylece RID sınıflandırma metodu farklı patolojileri gösterme açısından kullanılabilir bir sistem olarak kullanılabilir.

Zheng ve ark.(1999) Motor Birim Aksiyon Potansiyeli ve TME sesleri gibi durağan olmayan sinyallerin zaman frekans analizlerini yeni bir yöntem kullanarak yapmışlardır. Bu dağılımın diğer zaman-frekans teknikleri ile karşılaştırıldığında daha kesin olduğu ve daha fazla bilgi verdiği görülmüştür. Teknik Azaltılmış Karışım Dağılımı (Reduced Interference Distribution - RID) olarak adlandırılmıştır.

Sano ve ark. (1999) TME rahatsızlığı olan hastalarda en sık kullanılan tanılama işareti olan TME seslerinin genellikle hasta olmayan kişilerde de görülmekte olduğuna dikkat çeker. Bu şekliyle ses kaydı ancak sağlıklı ve hasta bireylerin TME seslerinde bir farklılık olması durumunda işe yarayacaktır. TME seslerinin genlik ve güç spektrumu, semptomları olan ve olmayan hastalarda kaydedilerek karşılaştırılmıştır. Hasta bireylerde sesler daha yüksek bir genliğe sahiptir. Seslerin önemli bir bölümü 2KHz ile 3KHz arasına düşmektedir. Bu şekliyle TME sesleri mümkün bir teşhis işareti olarak çalışılmaya devam edilecektir. TME ses kaydının bant genişliği en az 3KHz olmalıdır.

Niedzwiecki (2000), "Identification of Time Varying Processes" adlı referans kitabında zaman serilerinin sınıflandırılması ve modellemesi üzerine çalışmıştır. Bu kitabın Modelleme Esasları, Durağan Olmayan Süreçlerin Modellenmesi, Süreç Parçalama, Ağırlıklı En Küçük Kareler, En Küçük Ortalama Kareler, Taban Fonksiyonları, Kalman Filtreleri gibi konuları içeren alt bölümleriyle modelleme ve sınıflandırmanın teorik arka planı üzerine ayrıntılı bir sunum yapmaktadır. İlaveten bazı pratik problemler çözümlerini uygulamalı olarak göstermiştir.

Djurdjanovic ve ark. (2000) TME'den kaynaklanan tıklama ve çıtırdama sesleri çoğu zaman patoloji belirttiğini not etmiştir. Bunların RID zaman-frekans dağılımı



tanısal bir değere sahiptir. Görsel inceleme hatalara yol açabildiğinden seslerin otomatik olarak tanınması önem kazanmıştır. Çalışmanın amacı TME sesleri için optimal işleyebilecek ve örüntü tanıma teknikleri bulmaktır. Otomatik analiz öncekilerin benzer bir sonucunu ortaya çıkarmıştır. Skala değişmezliği atıldığında çok daha iyi sonuçlar alındığı görülmüştür.

Akan ve Başar Ünsal (2000) evrimsel spektrum tabanlı TME seslerinin zaman frekans analizi ve sınıflandırmasını yapmıştır. Çalışmada bir akselerometre ile TME titreşimleri çene açma veya kapama sürecinde kaydedilmiştir. Dört farklı tip sinyal sınıflandırılmıştır. TME seslerini sınıflandırmak için ortak zaman-frekans momentlerini ve sinir ağlarını kullanan bir metot sunmuşlardır.

Akan ve ark.(2006) TME sesleri ile diş birleşmesi rahatsızlıkları arasında bir ilişki olup olmadığını araştırmışlardır. Bir akselerometre ile TME titreşimleri çene açma veya kapama sürecinde kaydedilmiştir. Sinyal ayrık evrimsel dönüşüm (Discrete Evolutionary Transform) yöntemi ile analiz edilmiştir. Sinyaller tedavinin farklı aşamalarında kaydedilmiş ve sonuçlar tartışılmıştır.

Took ve ark.(2006) kesin kaynak yeri tespit edilememiş TME seslerinin ayrıştırılması için bir filtreleme yaklaşımı önermişlerdir. FastICA algoritması ismi verilen algoritmanın bir uzantısı olarak hazırlanmış yazılım ile altta yatan iki kaynağın kürtosis farklılığından yola çıkarak karışım matrisinin elde edilmesi dolayısıyla kaynak ayırımının gerçekleştirilmesi sağlanmıştır. Yazarlar burada FastICA algoritmasının nasıl kaynakları ayırdığını göstermişler ve bunu TME seslerini ayrışmasında kullanmışlardır.

Ghodsı ve ark. (2008) tekli spektrum analiz tabanlı yeni bir yöntem sunmuştur. Bu metotta birçok denek yüzlerine hareket veri işaretleyicileri konularak izlenmiş ve analiz edilmiştir. TME rahatsızlığı olan ve olmayan deneklerin verileri sinyal bileşenleri ve orijinal zaman serileri incelenerek gürültü ayrılmıştır. Sonuçlar hem simülasyon hem de gerçek verilerde etkin olduğu görülmüştür.

Took ve ark. (2008) TME’inde ortaya çıkan seslerin kaynaklarının ayırt edilebilmesi için özellikle klik sesleri üzerine yoğunlaşmışlardır. Burada kaydedilen sinyallerin bir tanesi bozuk TME’den biri ise sağlam TME’den geldiği varsayılmıştır. TME sinyallerinin istatistiksel durağan olmama özelliğinden faydalanan bir zaman-frekans yaklaşımı olan bozulmuş ayrıştırma kestirim tekniği (Degenerate Unmixing Estimation Technique, DUET) algoritması sayesinde ayrıştırma gerçekleştirilmiştir. DUET algoritması birbirine yakın iki sensor gerektirdiğinden zaman gecikmesi problemi ortaya çıkmıştır. Dolayısıyla zaman gecikmesi probleminin üstesinden gelmek

gerekmıştır. Burada DUET bir eklenti ile geliştirilmiş ve konvolüsyonsal Infomax algoritmasından da daha iyi performans sergilediği görülmüştür. Sonuçta klik sesi üreten TME sesi sağlıklı TME sesinden belirgin bir şekilde ayırt edilebilmiştir. Ayrıca otomatik olarak kaynak lokalizasyonu da gerçekleştirmiştir.

Kim ve ark. (2009) tekli değer ayrışımı (Singular Value Decomposition) kullanarak gürültü temizleme (denoising), ve dinamik zaman eğme (Dynamic Time Warping) kullanarak parçalama yapmışlar, sağlam ve hasarlı insan diz ekleminden elde edilen veriler arasındaki farklılığı tespit etmişlerdir. Sonuç olarak ortalama  $91.4 \pm 1.7$  sınıflandırma başarısı elde edilmiştir.

Oldukça sık görülen bir rahatsızlık olan temporomandibular eklem bozukluğuna tanı koymada en yaygın metotlardan biri de eklem seslerinin stetoskop ile dinlenmesidir. Dagar ve ark.(2014) geleneksel stetoskopları, temporomandibular eklem seslerini daha net alabilecek şekilde modifiye etmiş ve eklem seslerini kaydedip dijital ortamda analiz etmek için bu stetoskoba uygun bir yazılım geliştirmişlerdir. Modifiye edilmiş stetoskop ile hastalar ağızlarını açıp kaparken temporomandibular eklem sesleri dinlenmektedir. Yazılım dinlenen sesi, bir ses dalgası grafiği olarak kaydeder ve bu grafikten spektrum analiz ve ses uzunluğu grafiklerini çıkarır. Bu şekilde yazılım ile birlikte kullanılan cihaz hastalardan, herhangi bir rahatsızlığa sebebiyet vermeden daha net ses kayıtları alınmasına olanak sağlayarak, TME hastalıklarına tanı konulmasına yardımcı olur.

### **2.3. İstatistik Özellik Çıkarımına Dayalı Sınıflandırma Çalışmaları**

Wold ve ark.(1996) sesleri algısal ve akustik özelliklerine indirgeyen bir ses analiz, arama ve sınıflandırma motoru geliştirmişlerdir. Sözü edilen motor kullanıcıların seslerin bir veya daha fazla özelliklerini kullanarak aramalarını veya bir referans sesi seçerek veya yükleyerek motora bu seslere benzer veya benzemez olanların bulunmasını sağlar.

Altunkaya ve ark.(2010a) doğal kalp kapakları yerine ameliyatla takılan mekanik kalp kapaklarının durumunu, kalp kapaklarından alınan ses kayıtları ile analiz etmişlerdir. Bu amaçla AVR ve MVR operasyonu geçirmiş hastalardan alınan kalp seslerinin S1 ve S2 bileşenleri EKG yardımı ile seçilmiş daha sonra güç spektrum yoğunlukları AR yöntemi ile bulunmuştur. Daha sonra sonuçlardan altı farklı özellik çıkarılmıştır. Mitral odak ve aort odağı olmak üzere iki farklı odaktan alınan kayıtlar

istatistiksel olarak karşılaştırılmış, mitral odak kayıtlarının S1 ve S2 sesleri, aort odak kayıtlarının S1 ve S2 sesleri ve son olarak da MVR ve AVR'li hastaların aynı odağından alınan seslerin aralarındaki farklılıklar ortaya konmuştur.

Altunkaya ve ark. (2010b) mekanik kalp kapaklarında tromboz oluşumuna tanı koymak için, kalp ses kayıtlarından elde edilen istatistiksel özelliklerin kullanılmasını önermişlerdir. Veri olarak düzgün çalışan mekanik kalp kapağı olan beş hastadan ve mekanik kalp kapağı trombozu olan bir hastadan, tedavi öncesi ve sonrası kaydedilen kalp sesleri kullanılmıştır. Kalp seslerinin S1 ve S2 bileşenleri EKG ile bulunmuş ve bu ses bileşenleri için eğrilik (skewness) ve basıklık (kurtosis) istatistiksel özellikleri çıkarılmıştır. Tromboz tedavi öncesi ve sonrası trombozlu kalp kapağı seslerini karşılaştırmak için bağımlı t-testi, normal kalp kapağı sesleri ile trombozlu kalp kapağı seslerini karşılaştırmak için ise bağımsız t-testi kullanılmış, sonuç olarak S1 bileşeninin eğrilik özelliğinin düzgün çalıştığı ve problemlili kalp kapağını ayırt etmede iyi sonuçlar verdiği görülmüştür.

Arka plan sesleri, konuşma sinyallerinin kalitesini düşüren ve anlaşılabilirliğini bozan seslerdir ve konuşmayı doğru tanımlamak için gürültülü sinyallerin etkisini ortadan kaldırmak veya azaltmak önemlidir. Gürültülü sinyalleri sınıflandırırken, sinyallerde gürültünün yerini tespit etmek için çeşitli istatistiksel özelliklerden yararlanılır. Prabhavathi ve Ravikumar (2014), gürültü içeren konuşma sinyallerinin, ortalama, varyans, ko-varyans özelliklerini karşılaştırmışlardır. Otokorelasyon fonksiyonu kullanarak gürültülerin sabit sinyaller olup olmadığını tespit etmişlerdir. Farklı yaş aralığında ve cinsiyetteki kişilerin konuşma kayıtlarına dört farklı gürültü çeşidi ekleyerek oluşturulan verilerin içerdiği gürültü sinyallerini sınıflandırmışlar ve ses sinyallerindeki sinyal/gürültü oranı (SNR) bularak karşılaştırmışlardır.

#### **2.4. Derin Öğrenmeye Dayalı Sınıflandırma Çalışmaları**

Salaomon ve Bello (2015) çalışmasında kentsel sesleri sınıflandırırken özellik çıkarma aşamasında mel-spektrogram ve dönüşümü kullanılmasının sonuçlarını karşılaştırarak analiz etmişlerdir. Verilerden, mel-spektrogram parçaları ve saçılma dönüşümü çerçeveleri elde edildikten sonra, küresel k-means kümeleme algoritması kullanılarak özellikler çıkarılmışlardır. Mel-spektrogram için rastgele orman, saçılma dönüşümü için ise destek vektör makinesi sınıflandırıcıları kullanılarak, iki temsil

modelinin sınıflandırma başarısı karşılaştırılmış ve saçılma dağılımı kullanılarak çıkarılan özelliklerin daha iyi sonuçlar verdiği görülmüştür.

McLoughlin ve ark. (2015) gerçek dünya seslerinin tanınması üzerine çalışmışlardır. Yazarlar, spektrogram tabanlı, işitsel görüntü analizi kullanarak özellik çıkarma ile derin sinir ağları sınıflandırıcıları kullanarak bir ses tanıma yöntemi önermişlerdir. Veri olarak RWCP'den alınan, her biri 80 kayıt içeren 50 sınıflık bir veri seti kullanılmıştır. Önerilen metodun performansını analiz etmek için ilk olarak Google-SAI sistemindeki gibi SAI özellikleri kullanılarak destek vektör makineleri ile sınıflandırma yapılmıştır. Bu modele en uygun parametreler bulunduktan sonra sınıflandırıcı derin sinir ağları ile değiştirilerek tekrar test edilmiştir. Son olarak da spektrogram tabanlı SIF özellikleri kullanılarak derin sinir ağları sınıflandırıcısı eğitilmiş ve test sonuçları değerlendirilmiştir. Aynı zamanda modeller verilere farklı oranlarda gürültü eklenerek de test edilmiştir. Çalışmanın sonucunda SIF özellikleri kullanılarak yapılan sınıflandırmanın farklı gürültü seviyelerinde de diğer iki modelden daha iyi performans verdiği, alınan %90 üzeri sınıflandırma başarısı ile görülmüştür.

İnik ve Ülker (2017) derin öğrenme ile ilgili çeşitli fonksiyonların matematik arka planını vermişler, özellikle ESA (Evrişimsel/Konvolüsyon Sinir Ağları) hakkında ayrıntılı teorik bilgiler sunmuşlardır. AlexNet, ZFNet, GoogLeNet, Microsoft Rest Net ve R-CNN gibi birçok ünlü derin ağları da kısaca tanıtmışlardır.

Salamon ve Bello (2017) çevresel seslerin sınıflandırılması için bir derin konvolüsyon sinir ağı modeli önermişlerdir. Burada aynı zamanda etiketlenmiş verilerin azlığı sorunu için de ses verilerini artıracak farklı metotlar kullanılmışlar ve bu metotların CNN ile birlikte ortaya koyduğu performansı analiz etmişlerdir. Sınıflandırma için üç konvolüsyon katmanı, iki ortaklama katmanı ve son olarak da iki adet tam bağlantı katmanından oluşan bir derin konvolüsyon sinir ağı oluşturmuşlardır. On farklı sınıfa sahip, çevresel seslerden oluşan veri seti, beş farklı metot kullanılarak metodu genişletmişlerdir. Konvolüsyon sinir ağı modeli ve genişletme metodları kullanarak yapılan sınıflandırma son derece iyi sonuçlar vermiş, farklı veri genişletme metotlarının performanslarının verilerin sınıflarına göre değiştiği gözlemlenmiştir.

Beale ve ark. (2018) derin ağlar, derin ağ tasarımı, bulut alanında derin öğrenme, sinirsel ağ nesnelere, veri ve eğitim sınıfları, çok katmanlı sığ sinir ağları ve geriye yayılım eğitimi, dinamik sinir ağları, kontrol sistemleri, radyal tabanlı sinir ağları, kendini organize eden ve öğrenen vektör değerlendirme ağları, adaptif filtreler ve adaptif öğrenme gibi konular hakkında detaylı bilgi vermektedirler.

Smales (2018) çevresel seslerin sınıflandırılması için, aslen görüntü sınıflandırma alanında kullanılan derin öğrenme yöntemlerinin kullanılabileceğini araştırmıştır. Oluşturulan modele verilecek ses örneğinin, daha önceki belirlenmiş sınıflardan hangisine ne kadar yüzde ile ait olduğu belirleyebilmek için on farklı sınıf ve yaklaşık 8700 ses kaydı içeren bir veri seti seçmiştir. Ses kayıtlarının, ses kanalı sayısı, örnekleme hızı ve bit derinliği özellikleri standardize ederek özellik çıkarma adımına hazır hale getirmiştir. Daha sonra ses kayıtlarının görsel bir temsilini, Mel Frekanslı Cepstral Katsayıları (Mel-Frequency Cepstral Coefficients- MFCC) yöntemi ile bulmuştur. Elde edilen bu görseller eğitim ve test verileri olarak ikiye bölündükten sonra, eğitim verileri ile dört adet konvolüsyon katman ve bir adet yoğun katmandan oluşan bir konvolüsyon sinir ağı modeli eğitilmiş, test verileri ile de modelin doğruluğu test edilmiştir. Model eğitim aşamasında ~%98, test aşamasında ise ~%91 doğruluk oranı ile çalışmıştır. Bu durum yüksek doğruluk oranı gibi görünmektedir ancak karmaşıklık matrisine bakıldığında modelin bazı sınıfları birbiriyle karıştırdığı ve yanlış sonuçlar verdiği açıkça görülmektedir.

## 2.5. Literatürün Genel Değerlendirmesi

TME sesi sınıflandırmaları ve fiziksel büyüklükleri (frekans, perde, genlik, süre vs.) birçok araştırmanın konusu olmuştur (Widmalm ve ark. 1996a; 1996b; Sano ve ark. 1999; Zheng ve ark 1999). TME ses sinyallerinin genel özellikleri ve yapıları bahsi geçen çalışmalarda ayrıntısıyla irdelenmiş ve çıkarılmıştır. Fakat TME ses sinyalin temel özelliklerinden birisi sinyalin durağan olmamasıdır (non-stationary). Durağan olmayan sinyallerde farklı sinyaller aynı güç spektrumuna sahip olabilirler. Sınıflandırılması üzerine çeşitli araştırmalar özellikle zaman frekans analizi tabanlıdır (Akan ve Başar Ünsal. 2000; Akan ve ark. 2006). Bunların bir kısmında yapay zekâ kullanıldığı belirtilmekle beraber ayrıntı verilmemiştir. Diğer ses tanıma tabanlı olmayan bazı metotlarda literatürde yer almaktadır (Drurdjanovic ve ark. 2000).

Yukarıdaki çalışmalarında belirttiği gibi TME sinyalleri durağan olmayan sinyallerdir. Durağan sinyal bir olasılıklı (stochastic) sinyal olup birleşik momentleri (ortalama ve standart sapma gibi) zamanla ya da uzayla değişmeyen sinyaldir. Durağan olmayan sinyaller ise bunların tam tersi olup istatistiksel özellikleri zamanla değişen, belki bu değişimi tahmin edilebilir veya tespit edilebilir sinyallerdir. Bu sinyallerin özellik tespiti ve sınıflandırılması hayli problemlidir ve yeni teknikler gerektirmektedir

(Niedzwiecki 2000). Bu teknikler içerisinde yer alan evrimsel dönüşüm yöntemi (Evolutionary Transform) daha önceki çalışmalarda kullanılmıştır (Akan ve ark. 2006). Bu teknikler temel olarak durağan olmayan sinyallerin sınıflandırılması için kullanılan çeşitli tekniklerdir. Bunların içerisinde Dalgacık Dönüşümü (Wavelet Transform), Saklı Markov Modelleri (Hidden Markov Models-HMM), bulanık HMM, Support Vector Machines (SVM), SVM-HMM, ayrık evrimsel dönüşüm (Discrete Evolutionary Transform), yapay sinir ağları (YSA), bulanık mantık, uzman sistemler gibi sinyal işleme ve yapay zekâ teknikleri sayılabilir.

Yukarıda sayılan tekniklere ilaveten Altunkaya ve ark. (2010), mekanik kalp kapaklarında tromboz oluşumuna tanı koymak için, kalp ses kayıtlarından elde edilen istatistiksel özelliklerin kullanılmasını önermişlerdir. Kalp seslerinin EKG ile kaydedilmiş temel ses bileşenleri için eğrilik (skewness) ve basıklık (kurtosis) istatistiksel özellikleri çıkarılmıştır. Tromboz tedavi öncesi ve sonrası trombozlu kalp kapağı seslerini karşılaştırmak t-testleri kullanılmış, sonuç olarak düzgün çalışan ve problemlili kalp kapağını ayırt etmede iyi sonuçlar verdiği görülmüştür. Bu çalışma ve benzer diğer çalışmalar, Wold ve ark. (1996) ses sınıflaması yapmak için özellik vektörlerinde ortalama, varyans ve otokorelasyonun kullandığı çalışması, Sengupta ve ark. (2016) ortalama ve standart sapma gibi istatistiksel özellikleri temel aldığı “Lung sound classification using cepstral-based statistical features” adlı çalışması gibi yöntemler istatistiksel özelliklerin ses sınıflandırılmasında kullanıldığını göstermektedir.

En son olarak derin öğrenme tabanlı sınıflandırma algoritmaları ele alınmıştır. Son yıllarda yaygınlaşmış olan derin öğrenmeye dayalı ses sınıflandırmaları popülerite kazanmaktadır. Bu konuda konunun yeni olmasından dolayı çok fazla yayın bulunmamakla beraber yayın sayısı gün geçtikçe artmaktadır. Araştırmaya temel oluşturacak yayınlar arasında McLoughlin ve ark. (McLoughlin 2015) derin sinirsel ağların kullanılarak ses tabanlı olay sınıflandırmaya ait çalışmaları, Salamon ve Bello'nun (Salamon 2015) şehir ortamında kaydedilmiş seslerin analizinde derin saçılma ile özellik öğrenme çalışması ve yine aynı yazarlara ait (Salamon 2017) “Feature learning with deep scatterin for Urban Sound Analysis” ve “Deep Convolutional Neural Networks and Data Augmentatiton for Environmental Sound Classification” adlı çalışmaları dikkati çekmektedir.

### 3. MATERYAL VE YÖNTEM

Bu çalışmada TME seslerini dinlemek, kaydetmek ve üzerinde işlem yapmak için bilgisayar destekli bir cihaz tasarlanmış, kaydedilen veriler üzerinde işlem yapmak için sinyal işleme ve yapay zekâ algoritmalarına ilaveten diş hekimine kolaylık sağlayacak bir ara yüz geliştirilmiştir. Toplanan veriler içerisinde hastanın sol ve sağ TME sesi, ortam gürültüsü sesi, hastaya ait klinik veriler, diş hekiminin hasta hakkındaki notları, teşhis ve tedavisi (EK.1) vs. vardır. Ses verileri sinyal işleme yöntemleriyle işlenerek sinyalin çeşitli özellikleri üzerinde istenen değişiklikler yapılmış, ayrıca sinyaller hastalıklara göre YSA ve derin öğrenme algoritmaları ile sınıflandırılarak hastanın durumu hakkında karar veren yöntem geliştirilmiştir.

Amaçlanan yöntemin aşamaları aşağıda sıralanmıştır:

1. Cihazın Tasarım ve Kurulum aşaması: İlk aşamada cihaz uygun bir şekilde ses ve hekim notlarını kaydedecek şekilde bir araya getirilip bazı ek elektronik devreler tasarlanarak gerçekleştirilmiştir. Cihazın veri toplaması ve minimum düzeyde görsel analiz yapabilmesi için basit bir kayıt ve görüntüleme yazılımı geliştirilmiştir. Söz konusu yazılıma ait kodlar EK.2’de verilmiştir.

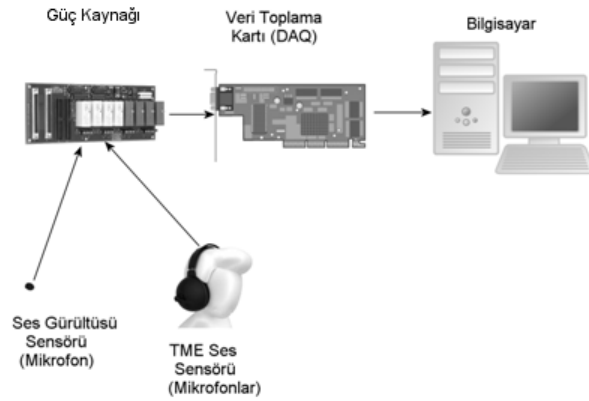
2. Veri Toplama Aşaması: Cihaz daha sonra Selçuk Üniversitesi Diş Hekimliği Fakültesine yerleştirilerek veri toplama safhasına geçilmiş ve hastalardan alınan veriler kaydedilmiştir. Diş Hekimliği Fakültesinden Etik Kurul oluru alınmıştır. (EK.3). Hastalara ilaveten gönüllü sağlam deneklerden de sağlam verileri toplanarak sınıflandırmaya katılmıştır.

3. Algoritma Geliştirme ve Raporlama: Toplanan veriler incelenerek çeşitli sinyal işleme ve yapay zekâ algoritmalarına tabi tutulmuştur. Araştırmanın çok yönlü olarak devam ettirilebilmesi için ortaya çıkan sonuçlar alınmış ve raporlanmıştır. Bu rapor ile araştırmanın ana klinik veri toplama ve temel algoritma geliştirme aşaması bitirilmiş ve sonuçlandırılmıştır. Süreç içerisinde konuda uzman diş hekimlerinin devamlı olarak görüşleri alınmıştır. Diş hekimleri sürece hastadan veri alınması ve hastalığın sınıflandırılması aşamasında aktif olarak katılmış ve uygun klinik muayeneler gerçekleştirmişlerdir. Uygulanan teknikler daha sonra deneylere tabi tutularak karar verme doğruluğu test edilmiştir.

Haziran 2011 ile Mayıs 2014 tarihleri arasında 3 sene süresince toplam 48 farklı gönüllüden veri toplamıştır. Bu toplanan verilerin 21 âdeti sağlıklı, 27 âdeti hasta yani TME rahatsızlığı olan kişilere aittir.

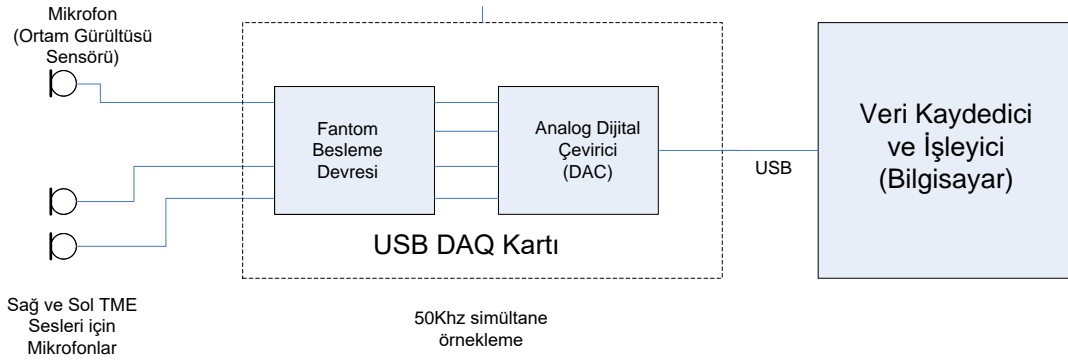
### 3.1. TME Ses kayıtlarını Yapmak için Girişimsel Olmayan (non-invasive) Cihaz Tasarımı ve Gerçeklenmesi

Cihazın yapısı Şekil 3.1 ve blok diyagramı Şekil 3.2’de verilmiştir. Tasarlanan sistemde en az üç adet ses sensörü yani mikrofon bulunmaktadır. Mikrofonlar olabildiğince geniş ses spektrumuna duyarlı olacak şekilde tercih edilmiştir. Bu spektrum aralığı literatürde (Sano ve ark. 1999) önerildiği gibi en az 3KHz olmalıdır, fakat tezde spektrum daha geniş tutularak 16Hz ile 20KHz arasında bir banda sahip olanlardan seçilmiştir. Mikrofonlar Earthworks M23 olarak belirlenmiştir. Earthworks M23 karakteristikleri Çizelge 3.1’de verilmiştir.



Şekil 3.1 Tasarlanan sistem prototipi





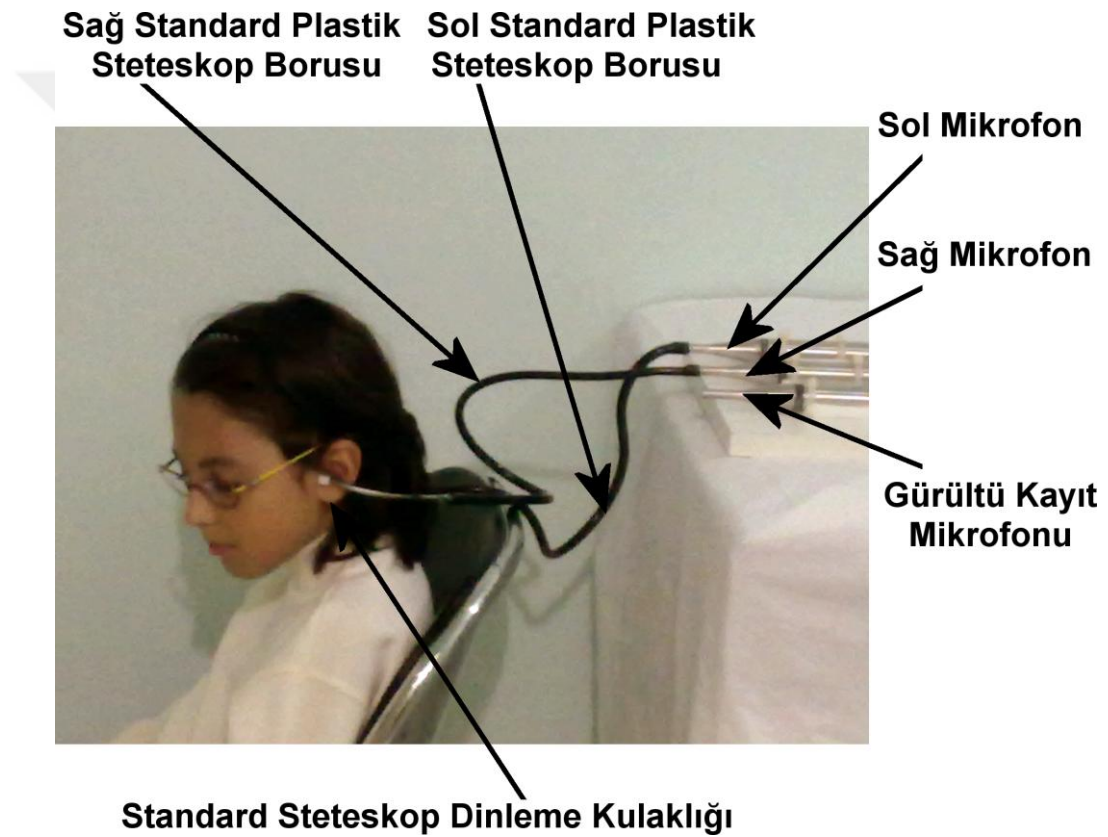
Şekil 3.2 Tasarlanan sistem blok diyagramı

Çizelge 3.1 Earthworks M23 karakteristikleri ve özellikleri

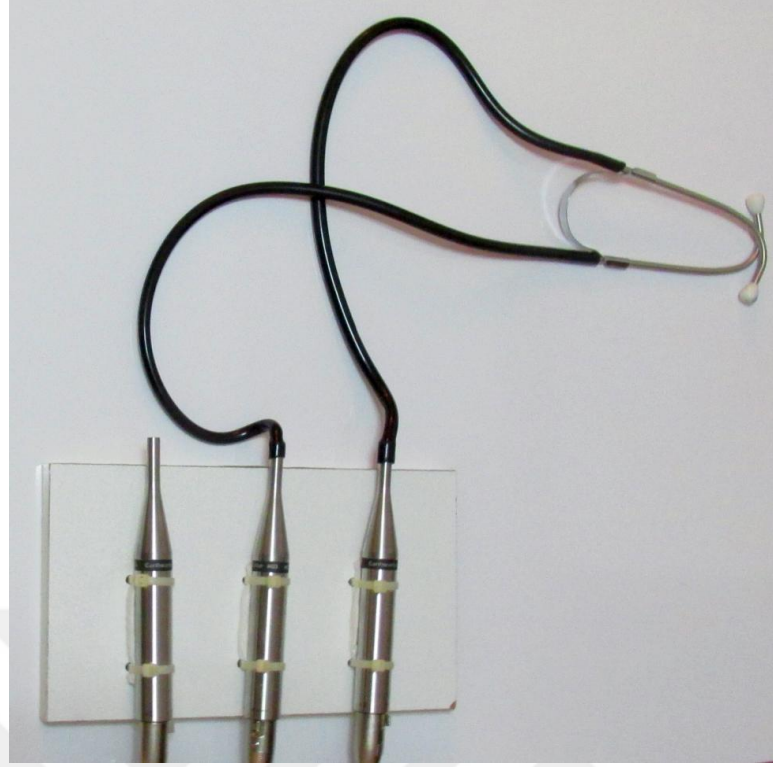
|                        |                                  |
|------------------------|----------------------------------|
| Frekans Cevabı         | 9Hz – 23KHz, +1 ile -3dB         |
| Kutupsal alım örüntüsü | Yönsüz                           |
| Duyarlılık             | 30mv/Pa, Tipik                   |
| Güç kaynağı            | 48v Fantom, 10mA                 |
| Tepe Akustik Giriş     | 42dB SPL                         |
| Çıkış                  | XLR:<br>                         |
| En düşük yük:          | 600Ω, 2 ve 3 nolu uçlar arasında |
| Gürültü                | 22dB, A eşdeğeri                 |

Bu mikrofonlar TME eklemine yakın olan TME seslerini ölçmek için kullanılan bir yöntem olan dış kulak yoluna içi boş boru ile bağlanmıştır. Burada bu plastik boru esnekliğinden ve hekimler tarafından tanınan ve hasta tarafından yadırganmasından dolayı stetoskop borusu olarak seçilmiştir. Hatta sesler hastanın dış kulak boşluğundan stetoskop vasıtasıyla toplanmıştır. Fakat burada sanki bir hekim gibi stetoskopun dinleme uçlarını kulağına oturtmuştur. Stetoskopun dinleme uçları burada hekim tarafından kullanıldığı şeklin tam aksi şekilde kullanılmıştır. Yani hekim stetoskopu ses dinlemek için kullanırken burada stetoskop hastanın kulak boşluğundan sesleri toplamak amacıyla kullanılmıştır. İki ayrı boru vasıtasıyla iki kulaktan toplanan sesler iki ayrı mikrofona aktarılarak hem sağ hem de sol TME sesleri muayene esnasında kaydedilmiştir. Ayrıca uygun bir yere yerleştirilecek bir başka mikrofon vasıtasıyla da ortam gürültüsü TME sesleriyle aynı anda algılanmış ve kaydedilmiştir. Dolayısıyla her

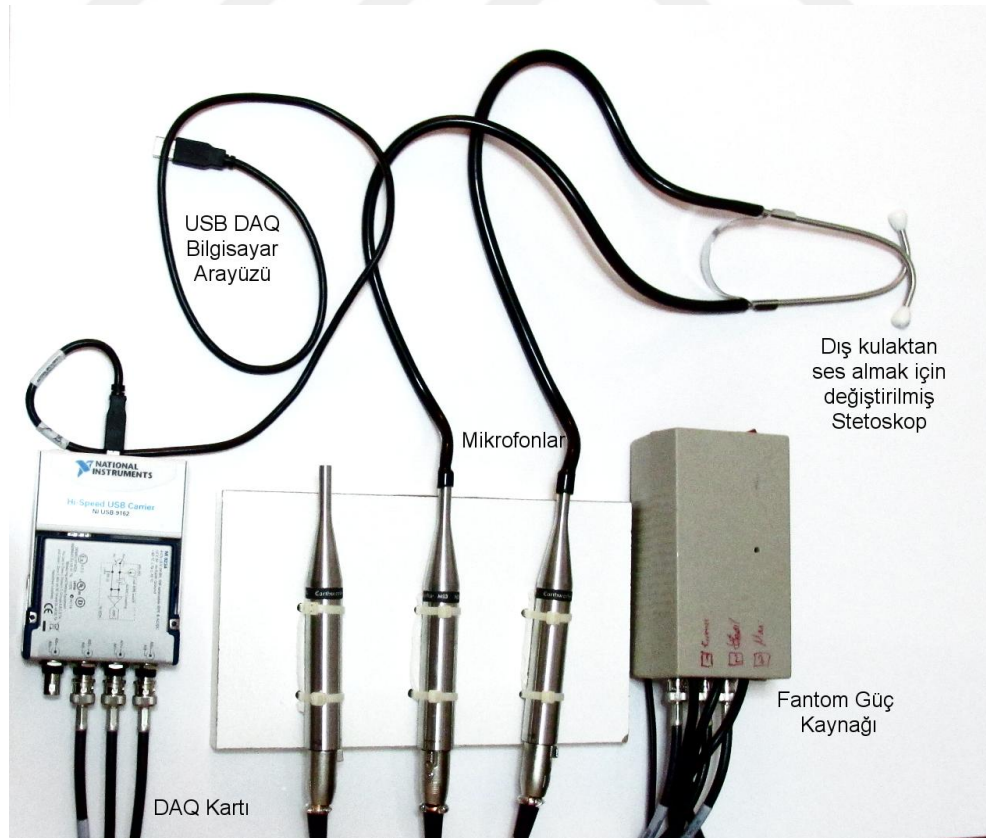
muayene esnasında iki adet eklem ve bir adet gürültü kanalı dâhil olmak üzere en az üç kanaldan veriler elde edilmiştir. Verilerin toplanması bir kez makine çalışır hale geldiğinde yani bilgisayar başlatılıp makine sıfırlandığında arka arkaya ölçüm yapabilecek duruma gelmiştir. Her hasta kaydı beş saniye sürmekte aynı hasta için bir kaç dakikada arka arkaya onlarca veri alınabilmektedir. Farklı hastalar için ise yalnızca hastaların muayene koltuğuna oturması ve steteskop takma süresi kadar bir süre geçmektedir. Bu süre bir dakikadan az olup ölçümlere hemen başlanabilmektedir. Sistemin giriş bölümünün çeşitli görüntüleri Şekil 3.3, Şekil 3.4 ve Şekil 3.5’de verilmiştir.



Şekil 3.3 Tasarlanan sistemin ölçüm esnasında kullanımı.

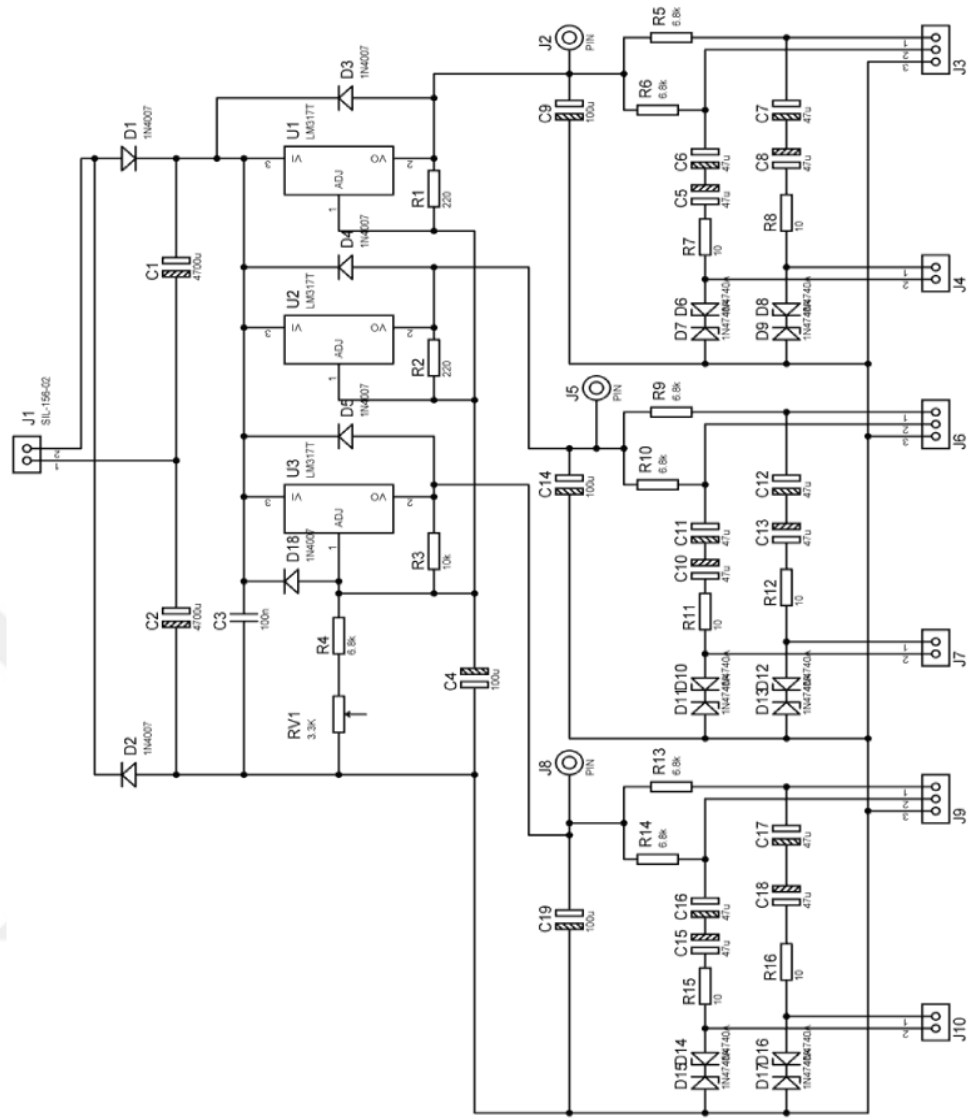


Şekil 3.4 Tasarlanan sistemin giriş kısmının stetoskop bağlantısı

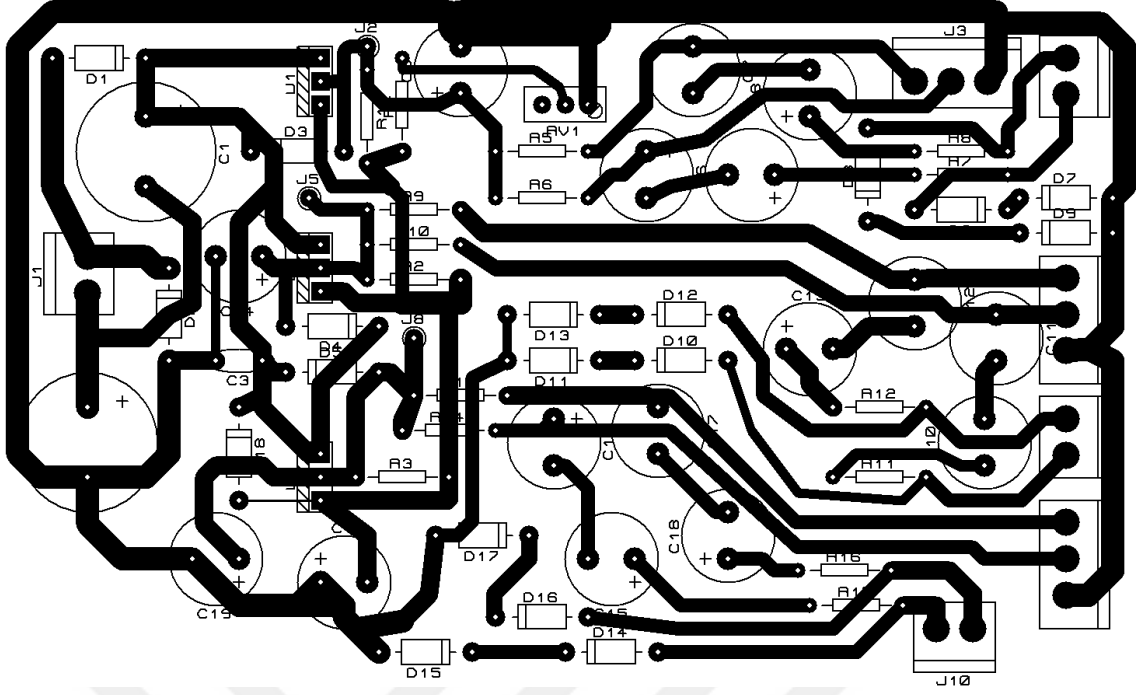


Şekil 3.5 Tasarlanan sistemin bilgisayar harici bütün donanımı

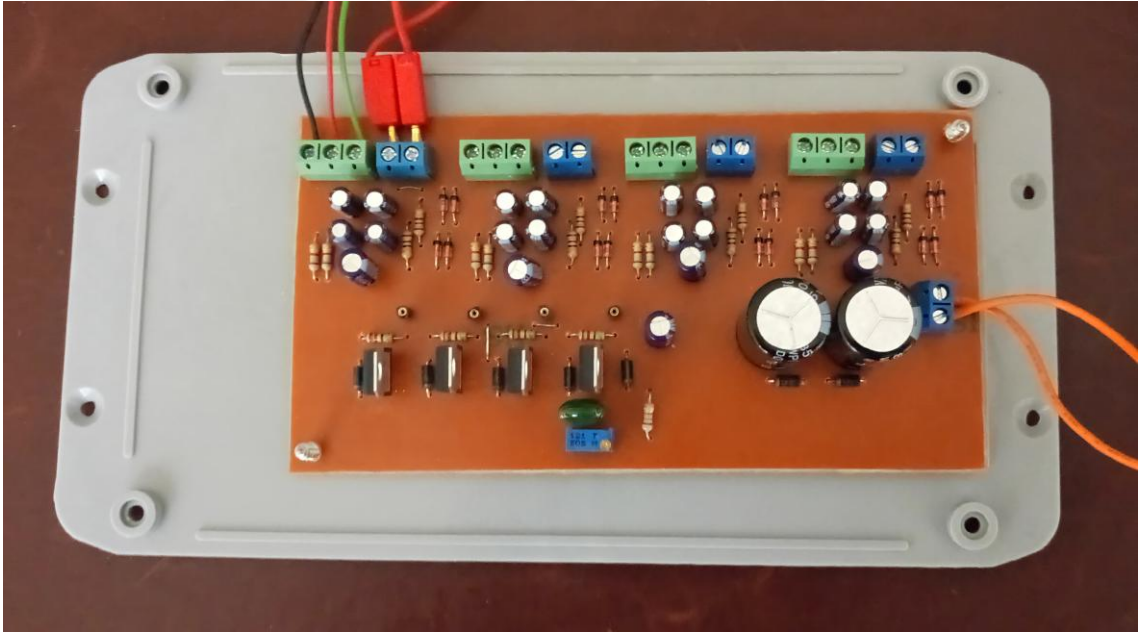
Şekil 3.6'da tasarlanan üç kanallı fantom güç kaynağının devre şeması görülmektedir. Fantom güç kaynaklı mikrofonların avantajlarından biri sinyal aktarımı sırasında simetrik bağlantı yapılmasıdır. Bu özellik uzun mesafe sinyal aktarımı sırasında ortaya çıkacak elektromanyetik gürültüyü azaltmak için oldukça faydalıdır. Fantom güç kaynakları olmadan bu tip mikrofonlar çalışmazlar. Fantom mikrofonlar 48v D.C. beslemeye ihtiyaç duyarlar. Devrede her bir ses kanalından yansıyan gerilim değişimlerini diğer kanallardan ayırmak için her bir kanalın kendine ait üç ayrı D.C. regülatörü tasarlanmıştır. Simetrik bağlantılar hem sinyal aktarımı hem de +48v D.C. güç aktarımı yaparlar. Şase bağlantısı ise D.C. gerilimin toprağına bağlanır. Fantom mikrofonlar bu nedenle üç ayrı bağlantıya ihtiyaç duyarlar. İki adet simetrik bağlantı uçları DAQ kartı girişlerine gönderilir. Burada yönsüz kondansatör elde etmek için iki kondansatör seri bağlanmıştır. Kullanılan bütün dirençler simetrisinin bozulmaması ve elektromanyetik gürültünün bastırılması amacıyla %1 hassasiyetinde seçilmiştir. Zener diyotlar ise DAQ kartını aşırı gerilim yüklemelerinden korumak amacı ile her iki simetrik uç ile toprak arasına konmuştur. Bu devreye ait baskı devre Şekil 3.7'de görülmektedir. Gürültü bastırılması ve elemanların yerleşimi arttırmak için daha sonra optimize edilmiştir. Kanal sayısı dörde çıkarılmış ve optimize edilmiş güç kaynağı ve aktarım devresinin gerçekleştirilmiş hali Şekil 3.8'de görülmektedir



Şekil 3.6 Fantom Güç Kaynağı Devresi



Şekil 3.7 Fantom güç kaynağı baskı devresi



Şekil 3.8 Fantom güç kaynağının gerçekleştirilmiş hali

Mikrofon tarafından toplanan sinyaller hem besleme hem de kuplaj ara yüzü olarak kullanılan ve gürültü seviyesinin en aza indirilmesi ve sinyal kalitesinin artırılması için fantom güç kaynağından geçirilir ve sinyal daha sonra analogtan sayısala dönüştürücü (DAQ) veri toplama kartına transfer edilir. DAQ olarak ise

National Instrument NI USB-9234 DAQ bağlantısı kullanılmıştır. NI USB9234 DAQ cihazının özellikleri Çizelge 3.2’de verilmiştir.

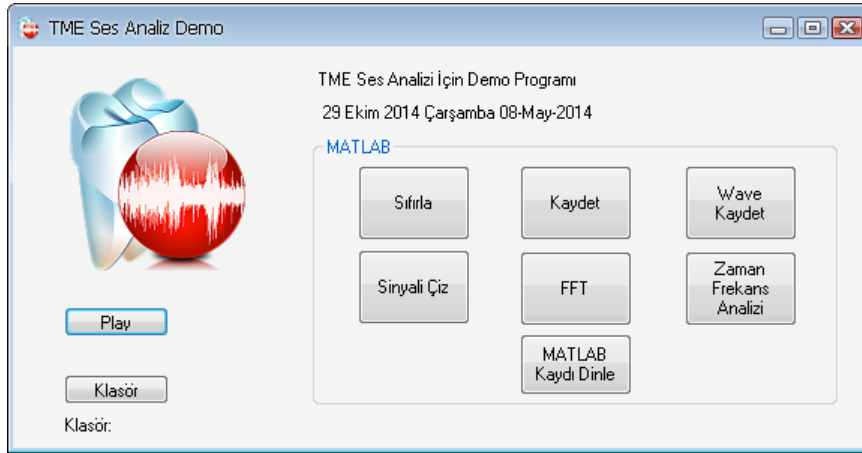
**Çizelge 3.2** NI-USB USB-DAQ teknik özellikleri

|                              |                                   |
|------------------------------|-----------------------------------|
| Kanal Sayısı                 | 4                                 |
| Örnekleme oranı              | 51200 Hz                          |
| Çözünürlük                   | 24 bit                            |
| Simültane Örnekleme          | Evet                              |
| Giriş Gerilim Aralığı        | +5v ila -5v                       |
| Duyarlılık                   | 176nv                             |
| Ön Sinyal İşleme             | Akım sürücülü anti-alising filtre |
| Bilgisayar bağlantı ara yüzü | USB                               |

DAQ cihazı analog verileri 51200Hz ve 24 bit çözünürlükte örnekleyerek bilgisayara USB ara yüzü ile transfer eder. NI USB-9234 DAQ özelliği 4 kanal ses sinyalleri için tasarlanmış olması ve en önemlisi bütün kanalları aynı anda örnekleyebilmesidir. Aynı anda örnekleme özelliği verilerin zaman bölgesi karşılaştırması, korelasyon analizi, gürültü azaltma algoritmaları geliştirmek için gereklidir. DAQ kartı  $51200 \times 4 \times 24 = 4.9152$  Gbit/sn veriyi bilgisayara aktarabilecek kapasitededir.

Verilerin direkt olarak bilgisayara kaydı mümkün olmadığından bu verilerin kaydı için bir yazılım geliştirilmiştir. Bu yazılım MATLAB programının veri toplama ara yüz fonksiyonlarının VisualC# programından çağrılarak gerçekleştirilmiştir. Bu yazılım MATLAB programında veri toplama fonksiyonları, grafik çizme fonksiyonları, veri kayıt fonksiyonları, veri işleme fonksiyonları gibi fonksiyonlar kullanılarak kodlanmıştır. Bu programlar daha sonra MATLAB derleme fonksiyonları kullanılarak dinamik bağlantı kütüphanelerine (DLL) çevrilmiştir. DLL kütüphaneleri daha sonra VisualC# kullanılarak görsel bir ara yüzle çağırılmıştır. MATLAB yazılımının sinyal işleme ve matematik hesaplama kapasitesi ile VisualC# programının hem nesneye yönelik programlama hem de görsel programlama özelliklerinin bir araya getirilerek hibrid yazılım gerçekleşmiştir. Program sinyal kaydetme, basit sinyal işleme, grafik çizme, kataloglama fonksiyonlarını başarıyla yerine getirmiş ve verileri MATLAB programının anlayacağı .mat formatında başarıyla kaydetmiştir. Bu formatın seçilmesinin sebebi matematik ve sinyal işleme analizlerinin MATLAB programıyla yapılacağından bir veri çevirmeye ihtiyaç duymadan direk olarak MATLAB programına

transfer edilebilmesidir. Geliştirilen yazılımın VisualC#'da hazırlanan ara yüzü Şekil 3.9'da görülmektedir.



**Şekil 3.9** TME Ses Analizi için tasarlanan ve gerçekleştirilen hibrid veri toplama ve ön işleme programının ara yüzü

Programın Sıfırla butonu DAQ donanımının ve yazılımının sıfırlanmasını sağlamaktadır. Kaydet butonu ise programın esas fonksiyonlitesi olan verilerin kayıt işlemini gerçekleştirir. Bu butona tıklandıktan sonra 5sn için mikrofonlardan gelen veriler, verilerin kaydedildiği tarih ve zaman bilgisiyle beraber kaydedilir. Wave Kaydet butonu kaydedilen verilerin .wav dosyası olarak kaydedilmesini Play butonu ise dinlenmesini sağlar. Sinyali Çiz, FFT ve Zaman Frekans Analizi butonları kaydedilen verileri sırasıyla zaman bölgesinde, frekans bölgesinde ve zaman frekans analiz grafiklerini çizmek için kullanılır. MATLAB Kaydı Dinle ise doğrudan MATLAB üzerinden kaydedilen verilerin dinlenmesini sağlar.

## 3.2. Zamanda Örneklenmiş Sinyalin Frekans ve Zaman-Frekans Bileşenleri

### 3.2.1. Ayrık Fourier Dönüşümü ve Hızlı Fourier Dönüşümü (DFT ve FFT)

Bir sinyalin bilgisayara kaydedilebilmesi için hem zamanda hem de uzamda sonlu ve örneklenmiş olmalıdır. Bu tip sinyallere sayısal sinyaller denir. Sayısal sinyaller zamanda sınırlı olduğu gibi frekansta da sınırlı olmalıdır. Zamanda örneklenmiş bir sinyalin zamanda örnekleme hızı Nyquist teoreminde belirtildiği üzere içerisinde bulunan en yüksek frekans bileşeninin en az iki katı olmalıdır. Bu tez çalışmasında kullanılan NI USB-9234 DAQ cihazı 51200 Hz örnekleme frekansına



sahiptir. Bu cihazın kabul edeceği bant genişliği 0 ile maksimum  $51200/2=25600\text{Hz}$  olabilir. Buradan da anlaşılacağı üzere elde edilen sinyallerin sınırlı bir bant genişliği vardır. Zamanda ve uzamda sınırlı bir sinyalin frekans bileşenleri DFT (Discrete Fourier Transform – Ayrık Fourier Dönüşümü) olarak bilinen bir matematiksel işlem ve algoritma ile hesaplanabilir. Verilmiş bir  $N$  adet örneğe sahip  $x[n]$  sayısal sinyalinin DFT'si aşağıdaki şekilde hesaplanır.

$$X(k) = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi kn}{N}} \quad (3.1)$$

Yukarıdaki Eşitlik 3.1 bilgisayarda hesaplamaya uygundur. Fakat eşitliğin hesaplama yükü ve müddeti  $N$ 'ye bağlı olarak asimptotik bir şekilde artmaktadır. Bundan kaçınmak için FFT (Fast Fourier Transform – Hızlı Fourier Dönüşümü) algoritmaları geliştirilmiştir. DFT algoritmalarının hesaplama yükü  $N^2$  oranında artmakta iken FFT algoritmalarının yükü  $N\log(N)$  oranında artmaktadır. FFT ve DFT'nin sonuçları temelde aynı olup, FFT'nin örnek sayısı  $N=2^j$  (burada  $j$  tamsayı) olmalıdır. Eğer  $N=2^j$  değil ise  $x[n]$  sayısal sinyalinin sonuna  $N=2^j$ 'yi sağlayan en küçük  $2^j$  değerine kadar sıfır ilave edilir. FFT algoritmaları MATLAB benzeri hazır program paketlerinde bulunmaktadır.

### 3.2.2. Zaman Frekans Dönüşümü - Spektrogram

Bir sinyalin frekans bileşeni zamanla değişkenlik göstermektir. FFT gibi algoritmalar ancak sinyalin bütünü için frekans bileşenlerini hesaplarlar. Bir spektrogram bir sinyalin frekans tayfının görsel bir temsilidir. Şekil 3.10'da sistem kimliklendirme ve radar uygulamalarında kullanılan genliği sabit, frekansı zamanla değişen bir sinyal olan cıvıltı fonksiyonunun spektrogramı görülmektedir. Siyah beyaz veya farklı renklerde gösterimler de mevcuttur. Örnekte dikey eksen zamanı, yatay eksen ise frekansı göstermektedir. Bu örnekteki spektrogram short time Fourier transform (kısa süreli Fourier dönüşümü) kullanılarak elde edilmiştir.

### 3.3. Bir dizinin istatistiksel özellikleri

Literatürdeki birçok yayında verilerin istatistiksel özellikleri kullanılarak sınıflandırma işlemleri yapılmıştır. Bir dizinin bazı istatistikî özellikleri kısaca aşağıda özetlenmiştir.

Ortalama (aritmetik ortalama): Dizi içinde yer alan bütün değerlerin toplanıp dizi uzunluğuna bölünmesi ile elde edilir.

Medyan: Dizinin en büyük değeri ile en küçük değerinin ortasında kalan değerdir.

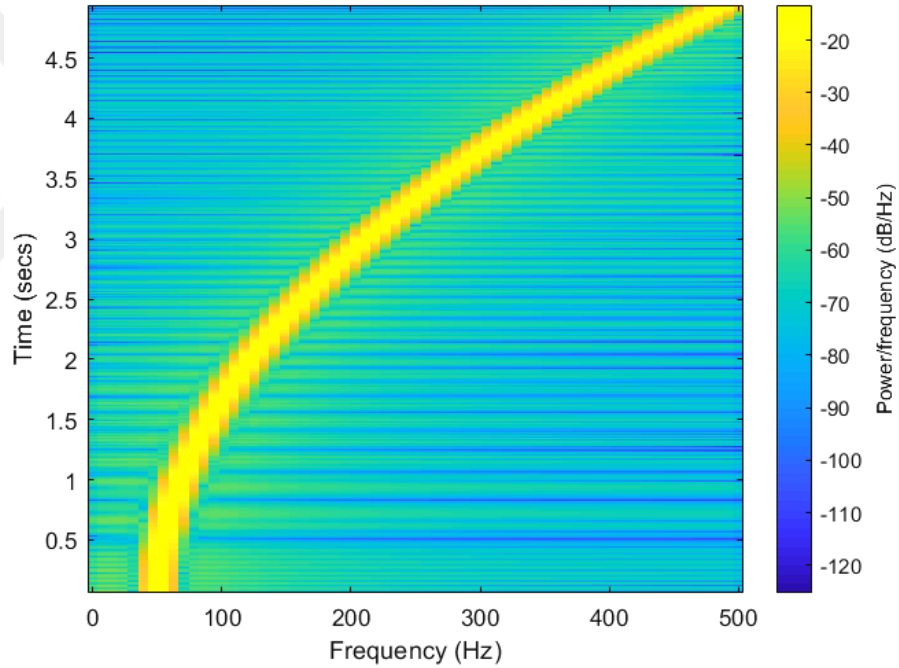
Mod: Dizinin en çok tekrarlayan değeridir.

Varyans: Bir dizinin varyansı Eşitlik 3.2 deki formülle hesaplanır.

$$Var = \sigma^2 = \frac{\sum_{n=0}^{N-1} (x[n] - \bar{x})^2}{N} \quad (3.2)$$

Burada  $x[n]$  dizinin  $n$  elemanın değeri,  $\bar{x}$ ,  $x[n]$  dizisinin aritmetik ortalaması,  $N$  dizinin eleman sayısıdır.

Standart Sapma: Varyansın pozitif kareköküne standart sapma denir. Yani Eşitlik 3.2'deki  $\sigma$  değeridir.



Şekil 3.10 Spektrogram Örneği

Momentler: Bir  $x[n]$  dizisinin  $r$ 'inci momenti Eşitlik 3.3 teki gibi hesaplanır.

$$\bar{x}^r = \frac{\sum_{n=0}^{N-1} x[n]^r}{N} \quad (3.3)$$

Ayrıca aritmetik ortalamaya göre momentler vardır. Bir  $x[n]$  dizisinin aritmetik ortalamaya göre  $r$ 'inci momenti yani  $m_r$  Eşitlik 3.4te verilmiştir.

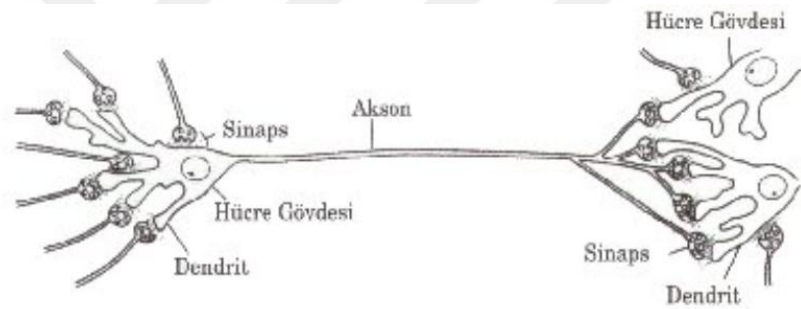
$$m_r = \frac{\sum_{n=0}^{N-1} (x[n] - \bar{x})^r}{N} \quad (3.4)$$

### 3.4. Yapay Sinir Ağları

#### 3.4.1. Giriş

Yapay sinir ağları, beyni oluşturan biyolojik sinir ağlarından ilham alınarak tasarlanmış hesaplama modelleridir. Beyindeki nöronların çalışması ve birbiriyle iletişimini temel alan modeller, regresyon analizi, yakınsama, sınıflandırma, filtreleme ve kümeleme gibi veri işleme problemlerinde kullanılır.

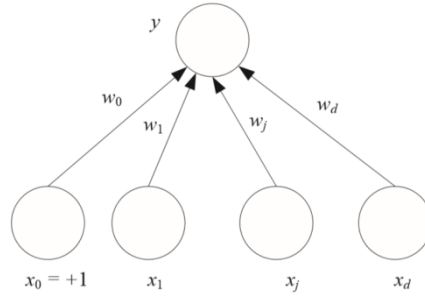
Biyolojik sinir sisteminin yapı taşları olan nöronlar, elektriksel ve kimyasal sinyallerle bir nörondan diğerine bilgi aktaran özel hücrelerdir. Bir sinir hücresi, gövde (soma), akson ve dentritlerden oluşur. Nöronlar sinyalleri dentritler yoluyla alır ve gelen sinyal akson boyunca taşınıp sinapslarda bölünerek diğer nöronlara aktarılır. Şekil 3.11’de bir biyolojik nöron yapısı görülmektedir.



Şekil 3.11 Biyolojik nöron yapısı

Biyolojik nöronlar matematiksel olarak ilk kez 1943 yılında Warren McCulloch ve Walter Pitts tarafından bir eşik mantık birimi (logic threshold unit) şeklinde ifade edilmiştir. Yapay nöronlar olarak adlandırılan bu modeller bir veya birden fazla girdi alır ve girdilerin toplamını bir aktivasyon fonksiyonundan geçirerek çıkış üretir (Anonim 2019; Dacombe 2017).

Tek bir yapay nöronun oluşun sinir ağı modeli 1958 yılında Frank Rosenblatt tarafından perceptron olarak tanımlanmıştır. Daha önce McCulloch–Pitts tarafından önerilen modele her girdi için belirli değişken ağırlıklar eklenmiştir. Bu model Şekil 3.12’de görülmektedir (Alpaydın 2014). Dış dünyadan  $x$  verilerini alan perceptron, her bir veriyi ilgili  $w$  ağırlığı çarpar ve ağırlıklı çarpımların toplamını çıktı olarak verir. Eşitlik 3.5 perceptron çıkış fonksiyonudur.



Şekil 3.12 Perceptron modeli

$$f(x) = \begin{cases} 1, & \sum_{i=1}^d w_i \cdot x_i + w_0 < 0 \\ 0, & \text{diğer durumda} \end{cases} \quad (3.5)$$

Aktivasyon fonksiyonları, modelin girdi ve çıktılar arasındaki ilişkiyi doğrusal olmayan bir şekilde tanımlamasına olanak vererek daha esnek modeller oluşturulmasını sağlar. Zamanla farklı problemler için yeni aktivasyon fonksiyonları önerilmiştir. Bunların en bilinenleri:

- Birim basamak fonksiyonu (unit step function): Bu fonksiyonun çıkışı girdilerin ağırlıklı toplamının belirli bir değeri sağlayıp sağlayamamasına göre 1 veya 0 değerini alır.
- Lineer birleşim: Bu durumda çıkış değeri girdilerin ağırlıklı toplamı artı bias değeridir.
- Sigmoid: Sigmoidler s şeklindeki eğrilere sahip, lineer olmayan fonksiyonlardır. Ancak genellikle aşağıdaki formülle tanımlanan lojistik fonksiyonun yerine kullanılır:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3.6)$$

- Rectifier Linear Unit (ReLU): Şu an en çok kullanılan aktivasyon fonksiyonlarından biri olan ReLU aşağıdaki gibi tanımlanır.

$$f(x) = \max(0, x) \quad (3.7)$$

Birden fazla yapay nöronun aralarında tek yönlü bağlantılar olan katmanlar halinde birleşmesiyle yapay sinir ağı modelleri elde edilir. Bu modeller, doğrusal olmayan verilerin sınıflandırılması tek katmanlı perceptron tarafından yapılamadığı için oluşturulmuştur.

Klasik ileri beslemeli yapay sinir ağı giriş katmanı, gizli katman ve çıktı katmanı olmak üzere en az üç katmandan oluşur.

- Giriş katmanı hariç tüm katmanlardaki nöronlar bir aktivasyon fonksiyonu içerir. Bu katmanda hesaplama yapılmaz yalnızca dış dünyadan gelen verileri modelde ileri taşımak için kullanılır
- Gizli katman verinin oluşturulan matematiksel modele göre işlendiği katmalardır. Modelin ihtiyacına göre birden fazla olabilir.
- Çıktı katmanı gizli katmandan aldığı işlenmiş verileri, çözülmeye çalışılan probleme uygun bir halde dış dünyaya yansıtır.

İleri beslemeli YSA modellerinde tek yönlü bir veri akışı vardır. Yapay sinir ağlarının çalışması basitçe girdi katmanından alınan verilerin gizli katmana iletilmesi ve buradaki aktivasyon fonksiyonlarından geçirilerek çıktı katmanına aktarılması şeklinde olur.

### 3.4.2. Yapay sinir ağlarında öğrenme

Tek katmanlı sinir ağlarında öğrenme, rastgele seçilen ağırlıkların hatayı en aza indirgeyecek şekilde değiştirilmesi ile olur. Ağırlıklar değiştirildikçe yeni çıktı değerleri hesaplanır ve hata oranı olabildiğince düşünceye kadar bu şekilde iteratif olarak ağırlıklar değiştirilir.

- Rastgele  $w$  ağırlıkları ve eşik değeri seçilir.
- Eğitim setindeki her veri  $j$  için  $t$  iterasyonunda perceptrondan gelen çıktı değeri hesaplanır (Eşitlik 3.8).

$$y = f[w(t) \cdot x_j] \quad (3.8)$$

- Hata miktarı gerçek değerden tahmin edilen değer çıkarılarak hesaplanır. Bir sonraki iterasyon için yeni ağırlıklar Eşitlik 3.9'daki gibi güncellenir.

$$w(t + 1) = w(t) - \text{öğrenme oranı} \cdot (\text{gerçek değer}_j - \text{tahmini değer}_j(t)) \cdot x_j \quad (3.9)$$

Ağırlıklar hata miktarı belirlenen eşik değerine düşünceye kadar güncellenir, bu şekilde en az hata ile gerçek sonuçlara en yakın sonuçları elde edebilecek bir sınıflandırma modeli elde edilmiş olur.

Çok katmanlı yapay sinir ağlarında da öğrenme ağırlıkların optimize edilmesi ile gerçekleştirilir. Ağırlıklar gradient descent yöntemine göre, hata oranı global minimuma ulaşılmaya çalışılarak güncellenir.

- Başlangıçta rastgele seçilen ağırlıklar ve girdi verileri modelden geçirilerek çıktılar hesaplanır.
- Ağırlıklardaki değişimi bulmak için önce toplam hatanın ağırlıklara göre kısmi türevi bulunur. Hata değeri Eşitlik 3.10 kullanılarak hesaplanır. Eşitlik 3.11 kullanılarak ta ağırlık değişimleri hesaplanır

$$E = \frac{1}{2} \sum_j (\text{gerçek değer} - \text{tahmini değer})^2 \quad (3.10)$$

$$\Delta w(t) = - \text{öğrenme oranı} \cdot \left[ \frac{\partial E}{\partial w} \right] \quad (3.11)$$

- Ağırlıklar ilgili ağırlık değişimleri ile Eşitlik 3.12'deki gibi toplanarak güncellenir.

$$w(t + 1) = w(t) + \Delta w(t) \quad (3.12)$$

Bu şekilde geri yayılma yöntemi ile güncellenen ağırlıklarla tekrar ileri besleme yapılarak yeni sonuçlara ve yeni hata oranlarına ulaşılır. Ağırlıklar tekrar tekrar hedeflenen hata oranına ulaşıncaya kadar optimize edilir.

Yapay sinir ağları, eğitim verileri kullanılarak eğitilip optimum ağırlık değerleri hesaplandıktan sonra sınıflandırma için kullanılabilir. Modelin veri setindeki sınıf sayısı kadar çıkışı olur. Yeni gelen bir verinin hangi sınıfa ait olduğunu bulunmak istendiğinde, veri eğitilmiş modelden geçirilir ve çıkış sayısı kadar sonuç alınır. Sonuçlar verinin ilgili sınıfa ne kadar olasılıkla ait olabileceğini söyler. Veri olasılığı en yüksek olan sınıfa atanarak sınıflandırılmış olur.

Bilgisayarların hesaplama güçleri arttıkça yapay sinir ağlarının da daha etkin versiyonları ortaya çıkmıştır. Özellikle son zamanlarda popüleritesi oldukça yüksek olan derin öğrenme modellerinin temelleri yapay sinir ağlarına dayanmaktadır. Bilgisayarlı görü, konuşma tanıma, doğal dil işleme, ses tanıma, makine çevirisi gibi alanlarda kullanılan derin öğrenme mimarileri insanlarla yarışabilecek hatta bazı durumlarda insanlara üstün gelen sonuçlar üretmişlerdir.

### 3.5. Derin Öğrenme ve Konvolüsyon Sinir Ağları

#### 3.5.1. Giriş

Derin öğrenme son yıllarda oldukça yaygınlaşmış ve sıkça adını duyduğumuz bir kavram haline gelmiştir. Derin öğrenme dediği zaman daha çok akla ilk gelen şey

bir bilgisayar vasıtasıyla verilmiş bir grup verinin bilgisayar tarafından bir çeşit algoritma kullanarak sınıflandırılması anlaşılmaktadır. Derin öğrenme aslen bundan çok daha fazlasıdır. Derin öğrenme temelde bilgisayarın insanın doğal olarak sahip olduğu öğrenme yetisinin bilgisayar tarafından edinilmesidir. Bu temel olarak makine öğrenmesi algoritmalarının bilgiyi önceden belirlenmiş herhangi bir model olmaksızın aynen insanların öğrenme sürecinde olduğu gibi tecrübe ile doğrudan veri grubundan öğrenilmesidir. Derin öğrenme bir veri grubunun içerisinde derinlerde saklı olan bilgiyi söz konusu veri grubunu kullanarak öğrenmektir (Beale ve ark. 2018).

Derin öğrenme özellikle resim tanıma gibi problemlere daha çok uygundur. Bu uygulamalar içerisinde yüz tanıma, hareket fark etme ve otonom sürüş, şerit bulma, yaya fark etme ve otonom park etme ve benzeri ileri sürüş yardımcılığı gibi uygulamalarda girmektedir.

### 3.5.2. Konvolüsyon sinir ağları

Konvolüsyon Sinir Ağları (Convolutional Neural Networks, CNN) oldukça yaygın olarak kullanılan bir derin öğrenme aracıdır. Konvolüsyon Sinir Ağları özellikle resim girişli uygulamalar için uygunsuz da metinler, sinyaller ve diğer sürekli tepkisel süreçler için de kullanılmaktadır.

Basit ve karmaşık hücrelerin bir karışımı olan görsel korteksin biyolojik yapısı Konvolüsyon Sinir Ağlarının geliştirilmesinde bir zemin oluşturmuştur. Bu hücreler görsel alanın alt bölgelerindeki durumlarına göre aktif hale gelmektedirler. Bu alt bölgeler algı alanları olarak adlandırılırlar. Görsel korteksin bu yapısından yola çıkarak bir konvolüsyon kattaki nöronlar diğer tipteki sinirsel ağlara bütün olarak bağlanmak yerine ilk önce katmanın alt bölgelerine bağlanırlar. Bu nöronlar bu alt bölgenin dışındaki alanlardaki resme bir tepki vermezler. Bazen bu alt bölgeler birbiri ile üst üste gelerek Konvolüsyon Sinir Ağlarının mekânsal korelasyonlu çıkışlar üretebilirken, diğer bazı sinirsel ağlarda nöronlar hiçbir bağlantı paylaşmadıklarından bağımsız çıkışlar üretirler.

Tam bağlantılı nöronlardan oluşan sinirsel ağlarda parametre sayısı yani bağlantı ağırlıkları girişin sayısının artması ile asimptotik olarak artar. Bir konvolüsyon sinir ağı ise azaltılmış bağlantı sayısı, paylaşılmış ağırlıklar ve örnek azaltma ile parametre sayısını azaltmaktadır. Bu hesaplama yükünü hafiflettiğinden konvolüsyon sinir ağlarının önemli bir avantajıdır.

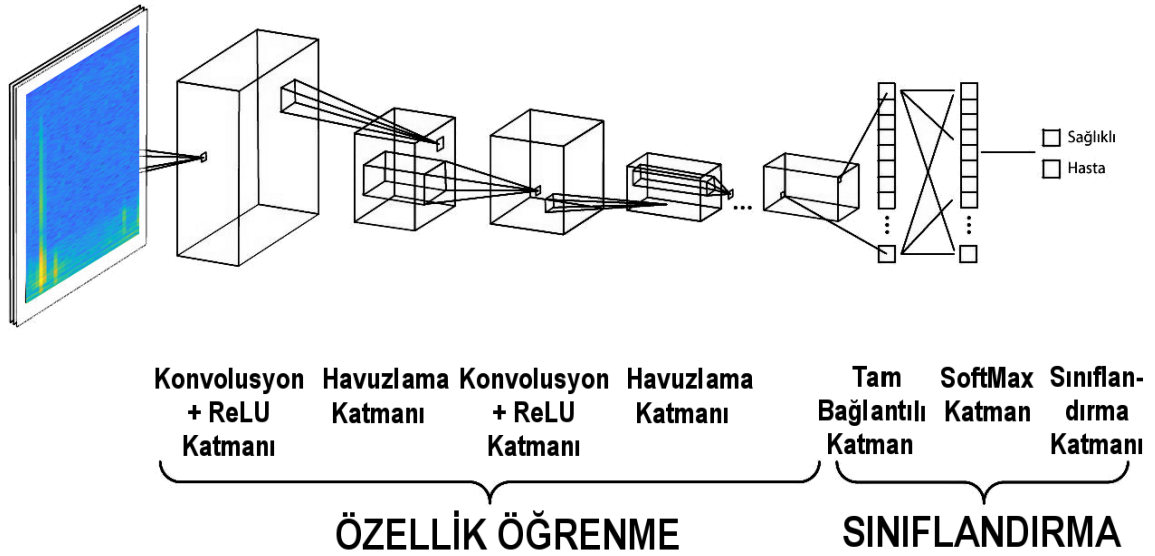
Bir Konvolüsyon Sinir Ağı, konvolüsyon katmanı (convolutional layer), en büyüğü seçen havuzlama katmanı (max-pooling layer) veya ortalama seçen havuzlama katmanı (average pooling layer) ve tam bağlantılı katman (fully connected layer) gibi birçok katmandan oluşmaktadır.

Resim tanımda kullanılan konvolüsyonel Sinir Ağında her bir katmandaki nöronlar 3 boyutlu giriş ve 3 boyutlu çıkışı olan 3 boyutlu nöronlar olarak düzenlenmiştir. Bu üç boyut, resimlerin uzunluğu, genişliği ve renk kanalları olarak belirlenmiştir. İlk konvolüsyon katmanı da bu resimlerin bölgelerine bağlıdır ve bunları 3 boyuta dönüştürmektedir. Her bir katmandaki gizli nöronlar özellik çıkarma olarak bilinen girişin doğrusal olmayan birleşimi öğrenme işlemini gerçekleştirmektedir. Aktivasyonlar olarak bilinen söz konusu öğrenilen özellikler bir katmandan diğer katmana giriş olarak belirlenmiştir. Son olarak öğrenilmiş özellikler ağın en ucunda yer alan bir sınıflandırıcıya ya da regresyon fonksiyonuna giriş olurlar.

Konvolüsyonel Sinir Ağlarının yapısı katman tipi ve sayısına bağlı olarak değişiklik göstermektedir. Bu katman tipleri ve sayıları uygulamanın özelliklerine ve veri tipine bağlı olarak farklılık arz eder. Eğer kategorik tepkiler var ise, bir sınıflandırma fonksiyonuna ve sınıflandırma katmanı yeterlidir. Bir başka durumda eğer tepki sürekli ise ağın sonunda bir regresyon katmanı yer almalıdır. Az sayıda siyah-beyaz resimleri öğrenmek için birkaç konvolüsyon katmanlı küçük bir ağ yeterli olmakla beraber, milyonlarca renkli resimden oluşan karmaşık veriler için birçok konvolüsyon katmanlı ağlardan ve tam bağlantılı katmanlardan oluşan çok daha karmaşık ağlara ihtiyaç duyulabilir.

Bu tez çalışmasında kullanılan derin öğrenme ağı MATLAB yazılımıyla geliştirilmiş bir Konvolüsyonel Sinir Ağı olup bir resim giriş katı, üç adet çok katmanlı konvolüsyon katmanı ve bir adet çıkış katmanından oluşmaktadır. Her bir konvolüsyon katmanı birbirini takip eden bir adet Convolution2dLayer, bir adet reLULayer ve bir adet Maxpooling2dLayer'den meydana gelir. Çıkış katmanı ise yine birbirini takip eden bir adet fullyConnectedLayer, bir adet softmaxLayer ve bir adet classificationLayer'den oluşmaktadır. Şekil 3.13'te tipik resim sınıflandırması için kullanılan konvolüsyon sinir ağları tabanlı derin öğrenme sistemi görülmektedir.





Şekil 3.13 Tipik Resim Sınıflandırması Yapan Derin Öğrenme Ağı Yapısı

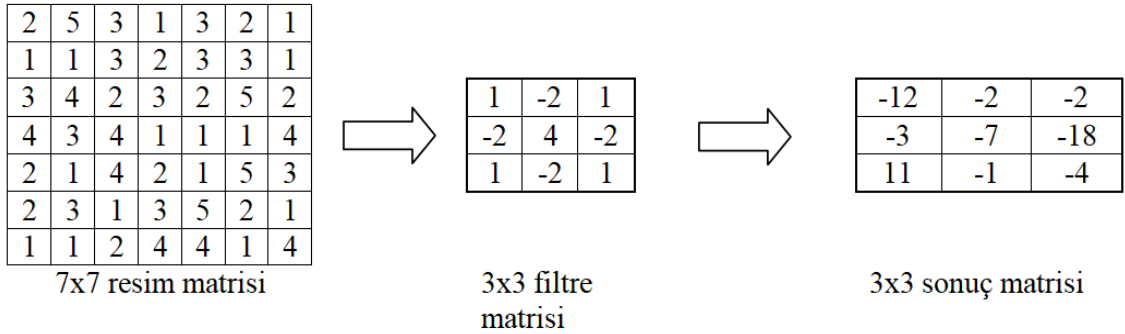
### 3.5.3. Derin ağ katmanları ve özellikleri

**Resim Giriş Katmanı (Image Input Layer):** Bu katmanın temel amacı resimleri giriş olarak kabul etmek ve normalizasyon uygulamaktır. Resim giriş katmanı resimleri giriş olarak alırken resimlerin özelliklerinin de ilaveten belirtilmesi gerekmektedir. Bu resim özellikleri resim boyutları ve renk sayısıdır. Örneğin 32 32 1 parametresi 32x32 büyüklüğünde 1 renkli (siyah-beyaz) resim demektir.

**Konvolüsyon Katmanı (Convolutional Layer):** Bir iki boyutlu kayan konvolüsyon filtresi girişe uygulanır. Konvolüsyon katmanı özellikle MATLAB® yazılımı için birçok farklı alt bileşen ve parametreden oluşur. Bunlar filtreleme ve adımlama, genişletilmiş konvolüsyon, özellik haritaları, sıfır ekleme, çıkış büyüklüğü, nöron sayısı, öğrenme parametreleri ve katman sayısıdır.

Temel konvolüsyon katmanı aynı zamanda dönüşüm katmanı olarak ta bilinir. İşlem 2 boyutlu bir filtrenin tüm görüntü üzerinde gezdirilmesi şeklinde olup çoğu zaman filtreleme olarak adlandırılır. Filtreler genellikle kare 2x2, 3x3, 5x5 büyüklüğünde iki boyutlu filtrelerden oluşmaktadır. Kare filtre resmin en üst sol köşe pozisyonundan başlayarak bütün resim boyunca kaydırılır. Bu kaydırma işlemi aslen iki boyutlu bir konvolüsyonu temsil ettiğinden katman isminin kökenini oluşturmaktadır. Filtreleme işlemi sonucu giriş matris boyutları küçülmektedir. Filtre matrisinin her bir adımda ne kadar kayması gerektiğini düzenleyen parametreye adım (stride) adı verilir. Aşağıdaki Şekil 3.14'te basit bir 7x7 boyutlu tek renk katmanına uygulanan 3x3 boyutlu

2 adımlı bir filtre uygulanması ve süreç sonucunda ise 3x3 boyutlu bir çıkış elde edilmesinin bir örneği görülmektedir.



**Şekil 3.14** 7x7 boyutlu renk katmanına uygulanan 3x3 boyutlu 2 adımlı bir filtre uygulaması

Her adım boyunca filtre matrisi üzerine geldiği 3x3 alt resim parçasının her birimi ile bire bir çarpılır ki buradaki örnek için toplam  $3 \times 3 = 9$  sonuç elde edilmektedir. En son elde edilen bu sonuçlar toplanarak çıkış değeri oluşturulur. Bu filtre matrisleri her bir renk kanalı için farklı olabilir ve uygulanan filtreleme işlemi her renk kanalı için ayrı ayrı yapılır ve bu değerlerin toplamı aktivasyon haritasını oluşturur. Daha sonra filtre kat sayılarından yola çıkarak bir normalizasyon işlemi yapılır. Yukarıdaki örnekte filtre katsayıları toplamı sıfır olduğundan bir normalizasyon uygulamaya gerek yoktur.

Bazen filtreler dilation (genişletme) ile genişletilerek alışı alanının büyüklüğü artırılabilir. Genişletme parametresine dilation factor denir. Genişletme ile örneğin 3x3 bir filtre 5x5 filtreye genişletilebilir. Genişlemiş filtrenin tanımlanmamış elemanları sıfıra eşitlenerek yapılır. Bu işlemin en büyük avantajı filtre alanını genişletirken temel filtre parametrelerini ve hesaplama yükünü arttırmamasıdır.

Eğer giriş resim büyüklüğü, filtre boyutu ve adım parametreleri ile tam olarak kaplanmıyorsa yani kaydırma sonucunda resim dışına taşma oluyorsa taşma alanı kadar bölüm giriş resmine eklenir ve her eklenen veri sıfıra eşitlenir. Buna zero-padding yani sıfır ekleme denir.

Süreç sonucunda elde edilen çıkış boyutu aşağıdaki formül kullanılarak hesaplanır;

$$CB = \frac{\{GB - [(FB - 1) \times GF + 1] + 2 \times SE\}}{A} + 1 \quad (3.13)$$

Burada  $CB$  Çıkış Boyutu,  $GB$  Giriş Boyutu,  $FB$  Filtre Boyutu,  $GF$  Genişleme Faktörü,  $SE$  eklenen sıfır sayısı,  $A$  Filtre Adım Aralığıdır.

Çıkış boyutunun genişlik ve boyunun çarpılması ile elde edilen sayı özellik haritasındaki nöron sayısını verir. Buradan elde edilen nöron sayısı ile filtre sayısının çarpılması ile konvolüsyon katmanının toplam çıkış nöron sayısı bulunur.

ReLU Katmanı (ReLU Layer): ReLU rectified linear unit'in kısaltmasıdır ve düzeltilmiş doğrusal birim anlamındadır. Bir ReLU katmanı girişin her bir parçası için bir eşik operasyonu uygular. Bu katmanın çıkış fonksiyonu aşağıdaki gibi tanımlanmıştır.

$$f(x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (3.14)$$

ReLU katmanı giriş boyutlarında bir değişiklik yapmaz. Standard ReLU'ya ilaveten leaky ReLU (sızdıran ReLU) ve clipped ReLU (kırpılmış ReLU) gibi varyasyonlarda mevcuttur.

Maksimum ve Ortalama Havuzlama Katmanı (Max Pooling Layer ve Average Pooling Layer): Bir max pooling katmanı girişi dikdörtgen seçme havuzlarına ayırır ve bu havuzun en büyük sayısını seçerek bir çeşit örnek azaltma işlemi yapar. Average pooling katmanının yaptığı ise dikdörtgen bölgelerin en büyük sayısını seçmek yerine dikdörtgen bölgenin ortalamasını alarak örnek azaltma işlemi yapar. Havuzlama katmanları örnek azaltma katmanları olduğu için havuzlama katman çıkışları girişlerine göre daha küçük boyuttadır. Bu katmanlar herhangi bir öğrenme işlemi yapmazlar ama kendisini izleyen katmanın giriş boyutunu küçülterek öğrenilmesi gereken parametre sayısını azaltmış olurlar.

Tam Bağlantılı Katman (Fully Connected Layer): Bir Tam Bağlantılı Katman girişi, bir ağırlık matrisi ile çarpar ve sabit bir değer (bias) ilave eder. Bu katmanlar genellikle bir veya daha fazla olarak konvolüsyonel katmanların sonuna eklenir. Adından da anlaşılacağı üzere bir önceki katmandaki bütün nöronlara bağlantı kurar. Bu katman önceki katmanlarda öğrenilmiş yerel özellikleri ve bilgileri birleştirerek bütün bir resim boyunca daha büyük desenlerin bulunmasını sağlar. Bu katmanların en son çıkışı sınıf sayısı kadar olacak şekilde ayarlanır. Bu katmanda ayrıca öğrenme hızı ve düzenleme parametrelerini ayarlamak da mümkündür.

Yumuşak Maksimum ve Sınıflandırma Katmanı (Softmax and Classification Layer): Softmax katmanı ve sınıflandırma katmanı sınıflandırma yapmak için Tamamen Bağlı Katmandan sonra birbirini izleyen katmanlar olmak zorundadır. Softmax katmanı girişe softmax fonksiyonunu uygular. Softmax çıkış aktivasyon fonksiyonu;

$$y_r(x) = \frac{\exp(a_r(x))}{\sum_{j=1}^k \exp(a_j(x))} \quad (3.15)$$

Yukarıda  $0 \leq y_r \leq 1$  ve  $\sum_{j=1}^k y_j = 1$ .

Softmax fonksiyonu çok sınıflı sınıflandırma problemleri için en son fully connected katmandan sonraki çıkış birim aktivasyon fonksiyonudur.

$$P(c_r|x, \theta) = \frac{P(x, \theta|c_r)P(c_r)}{\sum_{j=1}^k P(x, \theta|c_j)P(c_j)} = \frac{\exp(a_r(x, \theta))}{\sum_{j=1}^k \exp(a_j(x, \theta))} \quad (3.16)$$

Yukarıda  $0 \leq P(c_r|x, \theta) \leq 1$  ve  $\sum_{j=1}^k P(c_r|x, \theta)_j = 1$ , ek olarak  $a_r = \ln(P(x, \theta|c_r)P(c_r)) - P(x, \theta|c_r), P(c_r)$  sınıfın öncel olasılığı olmak kaydıyla verilmiş bir  $r$  sınıfının şartlı olasılığıdır.

Softmax fonksiyonu aynı zamanda genelleştirilmiş üssel olarakta bilinir ve mantıksal sigmoid fonksiyonun çok sınıflı bir genelleştirmesi olarak kabul edilir.

Tipik bir sınıflandırma ağı için Softmax katmanından sonra sınıflandırma katmanı eklenir. Bu katman softmax katmanından çıkışları alır ve aşağıda verilmiş ve  $1$  in  $K$ 'ı şeması olarak bilinen eşitlik bağlamında çapraz entropi fonksiyonu kullanarak her girişi  $K$  adet birbirini dışlayan sınıfa eşler. Kayıp fonksiyonu;

$$\text{loss} = \sum_{i=1, j=1}^{N, K} t_{ij} \ln y_{i,j} \quad (3.17)$$

Yukarıda  $N$  örnek sayısı,  $K$  sınıf sayısı,  $t_{ij}$   $i$ nci örneğin  $j$ inci sınıfa ait olduğunu gösteren gösterge ve  $y_{i,j}$  örnek  $i$  için  $j$ inci sınıfın çıkış örneğidir ve aynı zamanda softmax fonksiyonundan gelen değerdir (Beale ve ark. 2018; İnik ve Ülker 2017).

#### 4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Bu bölümde sınıflandırma yöntemlerin uygulanması ve sonuçların tartışılması yapılacaktır.

##### 4.1. Ses Verilerinin Toplanması ve Kayıt Altına Alınması

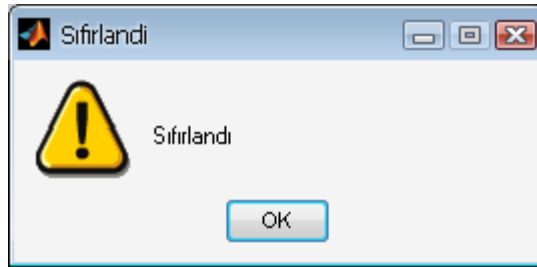
Veri toplama aşaması S.Ü. Diş Hekimliği Fakültesinin işbirliği ile gerçekleştirilmiştir. Bu aşamada gönüllü hasta ve sağlam kişilerden gerekli veriler toplanmıştır.

Bu aşamaya geçmeden önce S.Ü Diş Hekimliği Fakültesi Etik Kurulundan klinik deneyler için izin alınmıştır. Alınan iznin kopyası EK.3'te bulunmaktadır.

Veriler toplanırken hekim muayenesi ile birlikte veriler kayıt cihazı ile alınmıştır. Hekim muayenesinin bir özeti EK.1'de verilen hasta bilgi formuna işlenmiştir. Bu şekilde her bir hastaya ait klinik muayene bilgileri ve alınan verilerin hangi tarih ve saatte alındığı ve kime ait olduğu açık bir şekilde kataloglanmıştır. Bu aşama, çalışmanın en uzun süresini almış, hasta sayısının az olması ve benzeri problemler yüzünden verilerin toplanması oldukça uzun sürmüştür.

Haziran 2011 ile Mayıs 2014 tarihleri arasındaki 3 senelik süreç boyunca 48 farklı kişiden veriler toplanmış, bu gönüllüler içerisinde 21 sağlıklı kişiyi verileri vardır. Veriler toplanırken muayene olan hasta ve gönüllüler daha sonra veri toplama odasına alınmış, sistem kurulup sıfırlandıktan sonra hastaların verileri toplanmıştır. Veriler toplanırken dış gürültü etkilerini azaltabilmek için klinik dışı bir oda seçilmiştir. Odaya sistem Şekil 3.3'te görüldüğü gibi kurulmuş ve bilgisayar başlatılmıştır. Sistemin taşındığı çantadan çıkarılıp kurulma süresi 5dk civarında olup bilgisayarın hızına ve işlem kapasitesine bağlı olarak bu süre düşmektedir. Daha sonra veri kayıt programının Sıfırla düğmesi tıklanarak donanımın sıfırlanması beklenmektedir. Bu süre 30 ile 60 saniye civarında olup sıfırlama program tarafından açık bir şekilde Şekil 4.1'de görüldüğü gibi kullanıcıya bildirilmektedir. Fakat sıfırlama işlemi bir kez yapıldıktan sonra veriler hızlıca arka arkaya toplanabilmektedir, verilerin toplanması en fazla bir dakika sürmektedir. Şekil 3.3'te görüldüğü gibi hasta sandalyeye sırtı cihaza dönük olarak rahatça oturtulduktan sonra hastaya verilerin sağlıklı kaydedilmesi için 5sn'lik kayıt esansında sessiz olması ve komutlar harici hareket etmemesi istenmiştir. Hastalar

oturtulduktan sonra arkalarından stetoskop bir hekimin taktığı gibi aynen kulaklarına takılmıştır. Gönüllüler ve hastalar kendilerini minimum derecede rahatsızlık verdiğini düşündüğümüz durum için herhangi bir rahatsızlık şikâyetinde bulunmamışlardır. Aynı zamanda hastaların oldukça rahat olduğu gözlenmiştir. Bunun sebebinin ise hastaları rahatsız etmeden günlük hayatta kolaylıkla yapabilecekleri hareket ve durumların istenmesi, verilerin herhangi bir iç müdahaleye gerek görmeden ve 1 dakika gibi çok kısa sürede alınabilmesidir. Gönüllülerden cihazı kullananın işaretiyle ağızlarını bir defalığına olabildiğince geniş açıp ve kapatması istenmiştir. Bu esnada kullanıcı veri kaydı için geliştirilen programı kullanarak Kayıt düğmesini tıklayarak verilerin kaydı başlatılır. Hasta ağzını kayıt süresi olan 5sn içerisinde açıp kapatmalıdır. Bunu çok hızlı veya çok yavaş yapmamalı, doğal olan normal çene hareketini tekrarlamalıdır. Bu aşamada hastalar bazen komutlara uymadığından veri kaydı hatalı olmaktadır. Fakat bir sonraki ölçüm komutların tekrarlanması ile birlikte en fazla 1dk içerisinde tamamlanabilmektedir. Verilerin kaydı ve hasta formunun doldurulması ile bir örnek ölçüm tamamlanarak hard diske kaydedilmektedir. Kayıt esnasında kayıtlar tekrar dinlenebilmekte, grafikleri çizilerek özellikle hastanın çene hareketi izlenebilmekte ve bazı durumlarda çene hareketinin kayıt başlangıcından önce veya sonrasında yapıldığı fark edilerek ölçümler tekrar edilebilmektedir.



Şekil 4.1 Sistemin sıfırlandığını bildiren mesaj

Alınan verilerin sınıflandırılması için kullanılan kısaltmalar ve sınıflandırma formatının ayrıntıları Çizelge 4.1’de sunulmuş ve bu çizelgeye uygun klinik muayene teşhis ve sonuçları, verilerin sınıflandırılması, verilerin hangi tarihte alındığı gibi benzeri verilerin yer aldığı bilgiler Çizelge 4.2’de verilmiştir. Çizelge 4.2’de toplanan verilerin grafiklerinin çizdirilerek basit bir görsel inceleme sonucunda bazı verilerin hekim teşhisiyle uyuşmadığı görülmüştür. Hekim teşhisiyle çeliştiği görülen veriler şüpheli olarak işaretlenmiş, eğer ölçüm esnasında hastanın komutları anlamaması, yanlış zamanda çenesini oynatması, ortamın beklenenden gürültülü olması, hastanın

ölçüm esnasında konuşması gibi durumlarda ölçümler kullanıcı tarafından doğrudan geçersiz olarak not edilmektedir. Çizelgelerde şüpheli ve geçersiz veriler renkli olarak kodlanmıştır ve Çizelge 4.3 elde edilmiştir. Çizelge 4.3'e ait renk kodları Çizelge 4.4'te açıklanmıştır. Çizelge 4.2 ve Çizelge 4.3'te kaydedilen veriler aşağıdaki dört harfli (ABCD) formatı kullanarak sınıflandırılmıştır. Sınıflandırma formatının ayrıntıları Çizelge 4.1'de verilmiştir:

**Çizelge 4.1** Veri sınıflandırılması

| <b>ABCD</b>  |                              |
|--|------------------------------|
| <b>A ve C</b><br>Eklem yeri  | <b>R:</b> Sağ eklem verisi   |
|  | <b>L:</b> Sol eklem verisi   |
| <b>B ve D</b><br>Ses türü  | <b>H:</b> Sağlıklı, ses yok  |
|  | <b>C:</b> Klik               |
|  | <b>K:</b> Krepitasyon        |
|  | <b>P:</b> Pop                |
|  | <b>D:</b> Pop +Klik          |
|  | <b>E:</b> Krepitasyon + Klik |
|  | <b>F:</b> Pop + Krepitasyon  |
| Örnek; <i>RHLF</i> , sağ eklem sağlıklı ses yok, sol eklemde pop ve krepitasyon sesi duyulmuştur. Bu sınıflandırma hekim tarafından yapılmış, kullanıcı tarafından teyit edilmiştir. |                              |

Çizelge. 4.2 Verileri toplanan hasta bilgileri teşhis çizelgesi

| Hasta No:   | C/Y(*) | Bulgular ve Notlar   |
|-------------|--------|--|
| 20131223_01 | E/19   | Sağlıklı   |
| 20111223_01 | E/21   | Sağlıklı   |
| 20111223_02 | E/21   | Sağ_Klik Sol_Pop   |
| 20111223_03 | E/21   | Belirgin olmayan ses   |
| 20120106_01 | E/21   | Sağlıklı   |
| 20120106_02 | E/22   | Ses Yok, eklem problemi, gece diş sıkma  |
| 20120106_03 | E/22   | Sol kapatırken klik  |
| 20140506_01 | K/25   | Çift taraf klik  |
| 20131212_01 | K/21   | Sağ masseter prob. Belirgin ses yok  |
| 20140306_01 | E/21   | Klinik olarak her şey normal. Fakat Her iki tarafta da tanımlanamayan ama klik benzeri ses var |
| 20140306_02 | E/19   | Ses muayenede yok ama ölçümde solda klik   |
| 20140306_03 | E/20   | Ses yok, ağrı yok ça 41 mm sınır   |
| 20140306_04 | E/18   | Sağ deviasyon 49mm Normal  |
| 20140306_05 | E/20   | Alt çene sola deviasyon, her iki taraf klik  |
| 20140306_06 | E/21   | Açıklık 57mm, solda ses var. (BT:Sol Krepitasyon)  |
| 20140306_07 | E/19   | Diş sıkıyor, Açıklık 48mm Sol hafif klik   |
| 20140306_08 | K/19   | Açıklık 39mm, iki şiddetli poppin klik   |
| 20140410_01 | E/22   | Sola deviasyon, her iki tarafta click (BT:Pop?)  |
| 20140508_01 | K/20   | Kaparken klik  |
| 20140508_02 | E/19   | Sağlıklı   |
| 20140508_03 | E/21   | Sağlıklı-Ses yok   |
| 20111215_01 | K/21   | AA 19mm Sağa deviasyon   |
| 20131216_11 | E/21   | AA42mm (Düşük) sol pop klik (Çift)   |
| 20110926_01 | K      | Sağda rahtsızlık Sağ popping   |
| 20110921_01 | K/15   | Daha çok solda çok hafif ses (BT C + P?)   |
| 20110926_02 | K      | Solda hafif klik   |
| 20131212_02 | E/20   | Normal Ses yok. BT: Solda Klik+Pop   |
| 20131212_03 | E/20   | Disk kayması sağ problem Tedavi olmuş, Sağlıklı  |
| 20131212_04 | E/20   | Sağ çene ağrısı, dişsel olabilir, eklem normal   |
| 20131212_05 | E/20   | Kas muayenesi normal sola hafif dev Sol klik   |
| 20131212_06 | E/20   | Sola deviasyon ara sıra şiddetli sağ yansıma pop solda   |
| 20131216_01 | E/19   | Nadiren kramp Hafif deviasyon Sol klik   |
| 20131216_02 | E/19   | Belirgin bir durum yok (BT: Solda klik)  |
| 20131216_03 | E/19   | Her şey Normal, Sağlıklı   |



|             |      |  |
|-------------|------|--|
| 20131216_04 | E/20 | TME'de hafif ağrı, hafif klik iki tarafta                                      |
| 20131216_05 | E/19 | Sağlıklı (Ölçüm Problemleri)   |
| 20131216_06 | E/18 | Sağlıklı Ses yok   |
| 20131216_07 | E/19 | Şikayeti yok (Ölçümlerde problem)  |
| 20131216_08 | E/20 | Arada çene kitlen. Muayene normal. (Ölçümlerde problem)                        |
| 20131216_09 | E/19 | LP hassasiyet Hs Başlangıcı, Ölçüm problemleri                                 |
| 20131216_10 | E/20 | Sola deviasyon, iki tarafata klik sağda siddetli                               |
| 20110825_01 | K/16 | AA 38 mm D- Sol Sağ krepitasyon+klik   |
| 20110620_01 | K    | Sağda Disk Deformasyonu Sol Sağ Pop+Krepitasyon                                |
| 20110706_01 | K    | AA42mm redüksiyonlu antdisk deplasm Sağ Sol Pop+Klik                           |
| 20110706_02 | K/19 | AA29mm redüksiyonsuz antdisk deplasm Sağ Sol Pop+Krepitasyon                   |
| 20110706_03 | E/23 | AA30mm redüksiyonsuz antdisk deplasm Sağ Sol Pop                               |
| 20110810_01 | K/23 | AA28mm sağa deviasyon redüksiyonlu antdisk dep Sağ Klik Sol Klık + Krepitasyon |
| 20110905_01 | K/30 | AA48mm redüksiyonlu antdisk deplasm Sağ Sol Pop                                |

(\*) C/Y: Cinsiyet ve Yaş

**Çizelge. 4.3** Verileri toplanan hasta bilgileri değerlendirme, sınıflandırma ve bazı seçilmiş veriler çizelgesi

| Hasta No:   | Sınıflandırma | Veri Adı | Orijinal Veri Adı  | Ek Veriler           |                      |
|-------------|---------------|----------|--------------------|----------------------|----------------------|
| 20131223_01 | RHLH          | 0001RHLH | 23-Dec-2013-162306 | 23-Dec-2013-162342   | 23-Dec-2013-162402   |
| 20111223_01 | RHLH          | 0002RHLH | 23-Dec-2011-162437 | 23-Dec-2011-162453   |                      |
| 20111223_02 | RC LP         | 0003RCLP | 23-Dec-2011-162755 | 23-Dec-2011-162728   | 23-Dec-2011-162708   |
| 20111223_03 | RHLH          | 0004RHLH | 23-Dec-2011-160635 | 23-Dec-2011-160515   | 23-Dec-2011-160540   |
| 20120106_01 | RHLH          | 0005RHLH | 06-Jan-2012-163436 | 06-Jan-2012-163457   |                      |
| 20120106_02 | RHLH          | 0006RHLH | 06-Jan-2012-165823 | 06-Jan-2012-165914   | 06-Jan-2012-165840   |
| 20120106_03 | RHLC          | 0007RHLC | 06-Jan-2012-163211 | 06-Jan-2012-163148   |                      |
| 20140506_01 | RCLC          | 0008RCLC | 06-May-2014-111852 |                      |                      |
| 20131212_01 | RHLH?         | 0009RHLH | 12-Dec-2012-202242 | 12-Dec-2012-202331   | 12-Dec-2012-202226   |
| 20140306_01 | RHLH          | 0010RHLH | 06-Mar-2014-125214 | 06-Mar-2014-125258   | 06-Mar-2014-125320   |
| 20140306_02 | RHLC          | 0011RHLC | 10-Apr-2014-110037 | 10-Apr-2014-110112   | 10-Apr-2014-110205   |
| 20140306_03 | RHLH          | 0012RHLH | 06-Mar-2014-130555 | 06-Mar-2014-10141    | 06-Mar-2014-130215   |
| 20140306_04 | RHLH          | 0013RHLH | 06-Mar-2014-130754 | 06-Mar-2014-131449   | 06-Mar-2014-131517   |
| 20140306_05 | RCLC          | 0014RCLC | 06-Mar-2014-131416 | 06-Mar-2014-130821   |                      |
| 20140306_06 | RHLK          | 0015RHLK | 06-Mar-2014-131945 | 06-Mar-2014-131850   |                      |
| 20140306_07 | RHLC          | 0016RHLC | 06-Mar-2014-131156 | 06-Mar-2014-131229   | 06-Mar-2014-131259   |
| 20140306_08 | RPLP          | 0017RPLP | 06-Mar-2014-183832 | 06-Mar-2014-183802   | 06-Mar-2014-183903   |
| 20140410_01 | RCLC          | 0018RCLC | 10-Apr-2014-192328 | 10-Apr-2014-19:21:43 | 10-Apr-2014-19:22:13 |
| 20140508_01 | RHLC          | 0019RHLC | 08-May-2014-181745 | 08-May-2014-181851   | 08-May-2014-181917   |
| 20140508_02 | RHLH          | 0020RHLH | 08-May-2014-181536 | 08-May-2014-181632   |                      |
| 20140508_03 | RHLH          | 0021RHLH | 08-May-2014-181349 | 08-May-2014-181417   | 08-May-2014-181307   |
| 20111215_01 | RPLP          | 0022RPLP | 15-Dec-2011-111127 | 15-Dec-2011-111427   | 15-Dec-2011-111005   |
| 20131216_11 | RHLD          | 0023RHLD | 16-Dec-2013-161036 | 16-Dec-2013-161118   |                      |
| 20110926_01 | RPLH          | 0024RPLH | 26-Sep-2011-120428 |                      |                      |
| 20110921_01 | RHLD          | 0025RHLD | 21-Sep-2011-110333 | 21-Sep-2011-110256   | 21-Sep-2011-110208   |
| 20110926_02 | RHLC          | 0026RHLC | 26-Sep-2011-121110 | 26-sep_2011-121206   |                      |
| 20131212_02 | RHLH          | 0027RHLH | 12-Dec-2013-202942 | 12-Dec-2013-202645   | 12-Dec-2013-202822   |
| 20131212_03 | RHLH          | 0028RHLH | 12-Dec-2013-204012 | 12-Dec-2013-203827   |                      |
| 20131212_04 | RHLH          | 0029RHLH | 12-Dec-2013-203148 | 12-Dec-2013-203119   | 12-Dec-2013-203034   |
| 20131212_05 | RHLC          | 0030RHLC | 12-Dec-2013-204221 | 12-Dec-2013-204243   | 12-Dec-2013-204304   |
| 20131212_06 | RHLP          | 0031RHLP | 12-Dec-2013-205036 | 12-Dec-2013-204653   | 12-Dec-2013-204930   |
| 20131216_01 | RHLC          | 0032RHLC | 16-Dec-2013-160114 | 16-Dec-2013-160148   | 16-Dec-2013-160013   |
| 20131216_02 | RHLH          | 0033RHLH | 16-Dec-2013-162407 | 16-Dec-2013-162436   | 16-Dec-2013-162501   |
| 20131216_03 | RHLH          | 0034RHLH | 16-Dec-2013-153739 | 16-Dec-2013-153958   |                      |

|             |      |          |                    |                    |                     |
|-------------|------|----------|--------------------|--------------------|---------------------|
| 20131216_04 | RCLC | 0035RCLC | 16-Dec-2013-161510 | 16-Dec-2013-161542 | 16-Dec-2013-161637  |
| 20131216_05 | RHLH | 0036RHLH | 16-Dec-2013-154801 | 16-Dec-2013-154841 | 16-Dec-2013-154912  |
| 20131216_06 | RHLH | 0037RHLH | 16-Dec-2013-160501 | 16-Dec-2013-160327 |                     |
| 20131216_07 | RHLH | 0038RHLH | 16-Dec-2013-161941 | 16-Dec-2013-161843 | 16-Dec-2013-1612006 |
| 20131216_08 | RHLH | 0039RHLH | 16-Dec-2013-163122 | 16-Dec-2013-163043 | 16-Dec-2013-163205  |
| 20131216_09 | RHLH | 0040RHLH | 16-Dec-2013-155429 | 16-Dec-2013-155812 | 16-Dec-2013-155128  |
| 20131216_10 | RCLC | 0041RCLC | 16-Dec-2013-160704 | 16-Dec-2013-160740 | 16-Dec-2013-160820  |
| 20110825_01 | RELE | 0042RELE | 25-Aug-2011-120739 | 05-Sep-2011-114951 | 05-Sep-2011-115215  |
| 20110620_01 | RFLF | 0043RFLF | 20-Jun-2011-121551 | 28-Jun-2011-110935 | 12-Aug-2011-122303  |
| 20110706_01 | RDLD | 0044RDLD | 06-Jul-2011-141151 |                    |                     |
| 20110706_02 | RFLF | 0045RFLF | 06-Jul-2011-141501 | 05-Sep-2011-114635 |                     |
| 20110706_03 | RPLP | 0046RPLP | 06-Jul-2011-141649 |                    |                     |
| 20110810_01 | RCLE | 0047RCLE | 10-Aug-2011-154111 | 16-Aug-2011-165840 |                     |
| 20110905_01 | RPLP | 0048RPLP | 05-Sep-2011-155641 | 13-Sep-2011-142554 | 05-Sep-2011-155454  |

**Çizelge 4.4 Verilerin Renk Kodlamaları:**

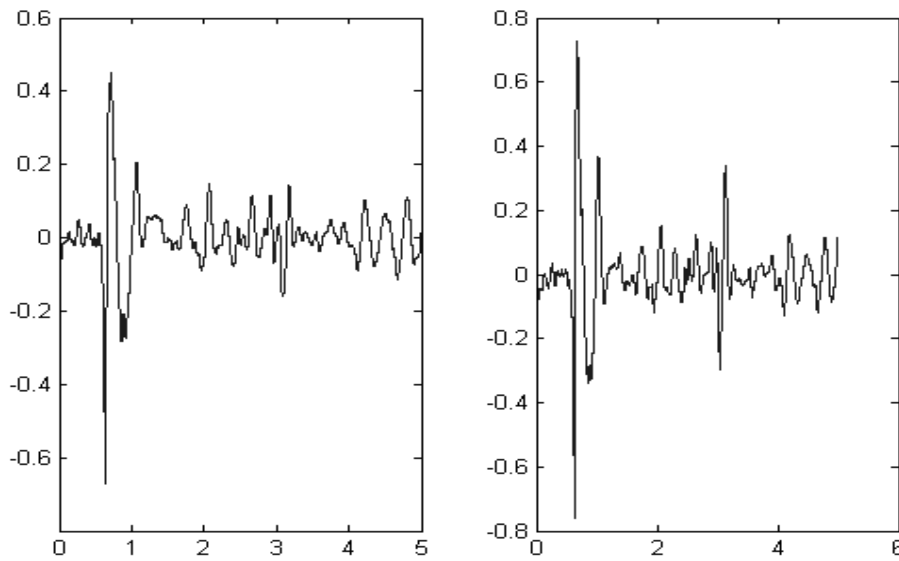
|  |  |  |
|--|--|--|
| Beyaz:<br>Örnek: 06-Jul-2011-141649<br>Geçerli veriler | Sarı:<br>Örnek: 13-Sep-2011-14255<br>Şüpheli Bulunan Veriler | Kırmızı:<br>05-Sep-2011-155454<br>Geçersiz Veriler |
|--|--|--|

## 4.2. Ses Verilerinin Frekans İçeriğine Dayalı Özellik Çıkarımı

Hali hazırda .mat formatında kaydedilmiş veriler MATLAB kullanılarak işlenmiş ve sonuçlar raporlanmıştır. Bu aşama tamamen klinikten uzak bilgisayar ortamında gerçekleştirilmiş ve herhangi bir hekim müdahalesi olmadan sonuçlar elde edilmiştir.

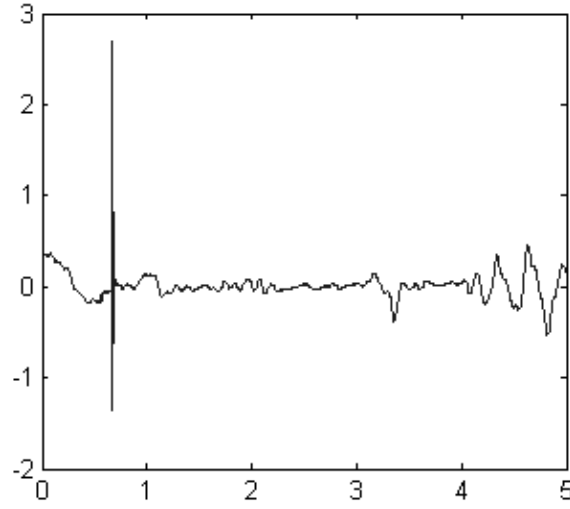
Veri toplama sürecinin çok uzun ve zor olması algoritma geliştirme sürecini kısıtlamış, düşük veri sayısı kullanılarak en iyi sonucun elde edilmesi amaçlanmıştır. Bu süreçte hekim teşhisi ve kaydedilen veriler arasında bazı uyumsuzluklar tespit edilmiş bu geçerli veri sayısını daha da düşürmüştür. Örnek olarak hasta 20131212\_01 için sağ masseteer kasında problem tespit edilmiş ama eklem için sağlam teşhisi konulmuştur. Fakat Şekil 4.2'teki grafikler incelendiğinde bu teşhisin problemlili olduğu tespit edilmiştir. Bu grafiklerde soldaki grafik sağ eklem verisi, sağdaki grafik sol eklem verisi olmak üzere sol eklem grafiklerinin genliğinin yüksek olması sol eklem daha problemlili olduğu sonucunu ortaya çıkarmıştır. Özellikle 3.5sn civarındaki sol eklem ses

genliğinin sağ eklem ses verilerinde görülmemesinden dolayı sağ eklem sesinde görülen yükselmenin sol eklem sesinin yansıması olduğu düşünülmektedir. Sol eklemden hem açarken (yaklaşık 1sn civarı), hem de kapatırken (yaklaşık 3.5sn civarı) bir ses tespiti genliklerin artmasından kolaylıkla anlaşılmaktadır. Grafik krepitasyon gibi durmaktadır. Bu sonuçlar elde edilen verinin hekim teşhisiyle uyumunun olduğu durumu ortadan kaldırmakta, bu da verinin elimine edilmesini gerekli kılmaktadır. Verilerin şüpheli olanlarının elenmesi ve yeterli sayıda veri olabilmesi açısından daha yaygın olan eklem klik sesi sınıflandırma için seçilmiştir.



**Şekil 4.2** hasta 20131212\_01 için eklem ses grafikleri. Yatay eksen zaman (sn) dikey eksen gerilim (volt)

Klik sesi grafiksel olarak görsel yolla ayırt edilmesi oldukça kolaydır. Bu özellikle Şekil 4.3'te 20140306\_02 nolu hastaya ait olan sol eklem verisinin veri kaydında açıkça görülmektedir. Bir saniyeden hemen önce ortaya çıkan ani hızlı yükselme yüksek frekans bileşenini göstermektedir. Bu klik sesinin açıkça ortaya çıkması olarak yorumlanmıştır. Verilerin sayısının az olması ve veri kaydetme aşamasının uzun sürmesinden dolayı algoritma geliştirme ve raporlama aşamasına çok geç geçilmiş bu yüzden hızlı bir sonuca ulaşabilmek için en net veriler seçilerek bunlar üzerinde analiz uygulanmıştır. Seçilen veriler sağlıklı eklemler ve klik sesi çıkaran eklemler şeklinde iki grup olarak Çizelge 4.2 ve 4.3'teki veriler arasından seçilmiş ve Çizelge 4.5'te listelenen 19 veri elde edilmiştir. Bu verilerin 12 tanesi sağlıklı eklemlerden, 7 tanesi klik sesi çıkaran eklemlerden elde edilmiştir. Böylece Çizelge 4.5'te bulunan veriler sınıflandırma için seçilmiştir.



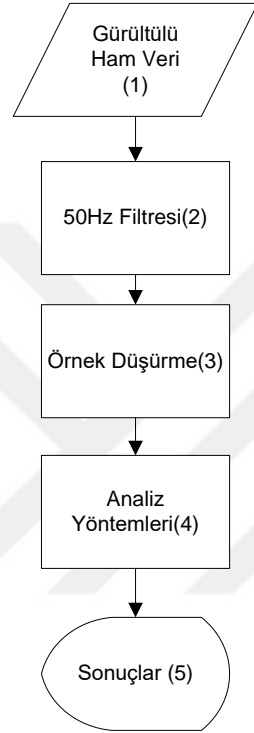
**Şekil 4.3** Hasta 20140306\_02 için sol eklem ses grafikleri.  
Yatay eksen zaman (sn) dikey eksen gerilim (volt)

**Çizelge 4.5** Analiz için seçilen veriler  
(R: sağ L: sol eklem verisi, H: Sağlıklı, C: Klik Sesi)

|                        |   |   |
|------------------------|---|---|
| 23-Dec-2013-162306.mat | R | H |
| 23-Dec-2011-162437.mat | L | H |
| 23-Dec-2011-160635.mat | R | H |
| 06-Jan-2012-163436.mat | L | H |
| 06-Jan-2012-165823.mat | R | H |
| 06-Mar-2014-130754.mat | L | H |
| 08-May-2014-181536.mat | R | H |
| 12-Dec-2013-204012.mat | L | H |
| 12-Dec-2013-203148.mat | R | H |
| 16-Dec-2013-153739.mat | L | H |
| 16-Dec-2013-160501.mat | R | H |
| 08-May-2014-181349.mat | L | H |
| 06-Jan-2012-163211.mat | L | C |
| 10-Apr-2014-110037.mat | L | C |
| 06-Mar-2014-131156.mat | L | C |
| 08-May-2014-181745.mat | L | C |
| 26-Sep-2011-121110.mat | L | C |
| 12-Dec-2013-204221.mat | L | C |
| 16-Dec-2013-160114.mat | L | C |

Bu veriler MATLAB programına alınarak sinyal işleme algoritmaları vasıtasıyla işlenmiş ve daha sonra YSA kullanılarak sınıflandırılmıştır. Sürecin geliştirilmiş

akış diyagramı Şekil 4.4'te gösterilmiştir. Bu aşamalar, şekilden de görüleceği gibi; gürültülü ham verinin MATLAB ortamına alınması, veriler 50Hz şebeke gürültüsünden kurtulmak için notch filtreden geçirilmesi, örnek sayısı düşürülerek gereksiz frekans bileşenleri atılması, verilerin uygun analiz yöntemi için ön işlemden geçirilerek sınıflandırılması, sonuçların elde edilmesi ve raporlanması olarak belirlenmiştir.



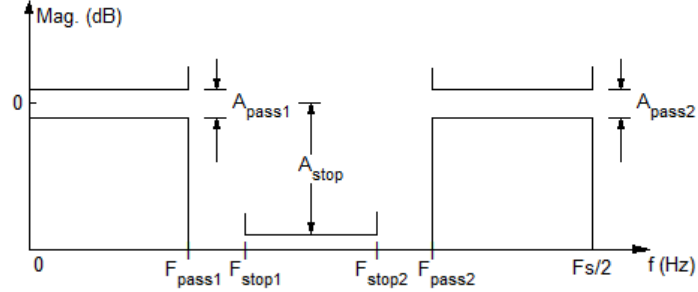
1. Gürültülü ham veri MATLAB ortamına alınır.
2. Veriler 50Hz hat gürültüsünden kurtulmak için notch filtreden geçirilir
3. Örnek sayısı düşürülerek gereksiz frekans bileşenleri atılır
4. Veriler uygun analiz yöntemi için ön işlemden geçirilerek sınıflandırılır
5. Sonuçlar elde edilir ve raporlanır.

Şekil 4.4 Veri değerlendirme sürecin geliştirilmiş akış diyagramı

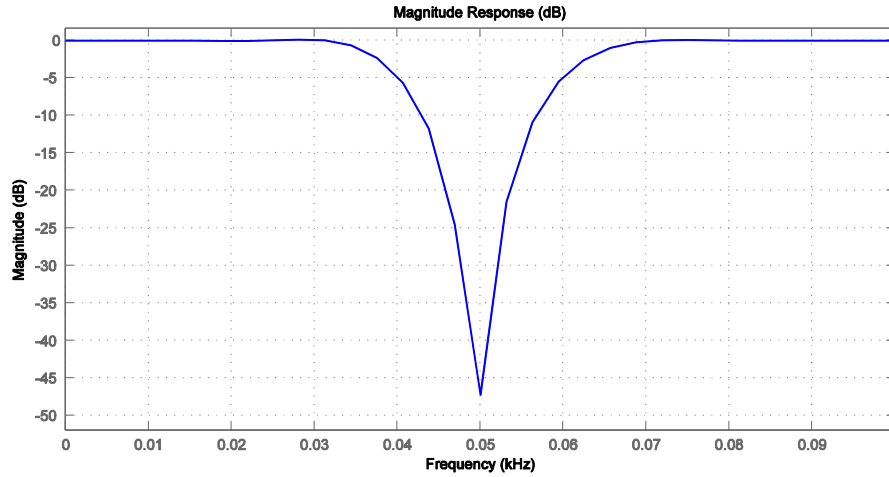
Sürecin ilk aşaması zaten .mat formatında olan verilerin doğrudan MATLAB ortamına atılmasıyla başlamaktadır. Bu aşamadan sonra Şekil 4.5'te kriterleri verilen 50Hz Notch filtresi tasarlanmıştır. Şekil 4.6'da notch filtrenin 50Hz civarı büyütülmüş frekans cevabı görülmektedir. Bu filtrenin kriterlerinin belirlenmiş teknik özellikleri aşağıdaki gibidir:

- $F_s=51200\text{Hz}$ : Örnekleme frekansı
- $F_{pass1}=35\text{Hz}$  1. Bant geçirme sınır frekansı
- $F_{stop1}=45\text{Hz}$  1. Bant durdurma frekansı
- $F_{stop2}=55.56\text{Hz}$  2. Bant durdurma frekansı
- $F_{pass2}=71.43\text{Hz}$  2. Bant geçirme sınır frekansı
- $A_{pass1}=0.057501127785$  1. Bant geçirme dalgalanması

- $A_{stop} = 0.022387211386$  Bant durdurma azaltması.
- $A_{pass2} = 0.057501127785$  2. Bant geçirme dalgalanması
- $dens = 20$  Yoğunluk faktörü



Şekil 4.5 50Hz notch filtre kriterleri



Şekil 4.6 Tasarlanmış 50Hz notch filtrenin 50Hz civarı büyütülmüş frekans cevabı  
Yatay eksen frekans (KHz) dikey eksen genlik (dB)

Bu filtre esas itibariyle 50Hz ana şebeke geriliminin oluşturduğu gürültüyü -16.5dB civarında bir azaltma yapmak amacıyla kullanılmıştır. Bundan sonra örnek sayısını düşürme yoluna gidilmiştir. Örnek sayısı her bir veri dizisi için  $51200 \times 5 = 256000$  veriden oluşmaktadır. Bu verilerin 5KHz'ten yüksek frekans kısımları kesilerek atılmış ve böylece 0-25KHz arası veriler, 0-5KHz arasına düşürülmüştür. Burada alçak geçiren filtre yerine örnek azaltılarak bu işlem gerçekleştirilmiştir. Bu işlem hem yüksek frekans gürültülerini yok etmiş, hem de veri sayısını  $5 \times 10000 = 50000$  örneğe indirmiştir.

Daha sonra analiz ve sınıflandırma aşamasına geçilmiştir. Örnek sayıları kullanılarak veriler önce zaman bölgesinde YSA'ya tabi tutulmuş, veri sayıları

düşürülerek denemeler devam etmiştir. Verilerin zaman bölgesi YSA sınıflandırmalarının başarısı oldukça düşük olduğundan bu kez de örneklere Fourier dönüşümü uygulanarak frekans bölgesine geçilmiştir. Bu işlem yapılırken örnek yine 50000 örneklik serilerden başlanmış ve DFT uygulanarak frekans bölgesine geçilmiştir. Frekans bölgesi verileri tekrar örnek azaltma işlemine tâbi tutulup en son aşama için 100 örneklik bir veri dizisine kadar indirilmiştir. Bu veri dizisine YSA uygulanmış ve sonuçlar raporlanmıştır.

### 4.3. Ses Verilerinin YSA Kullanarak Sınıflandırılması

Veriler programa girilmiş YSA programı ilk analiz kısmında verilerin 3 tanesini tekrar elemiştir. Bu veriler katılmadan YSA eğitim ve test uygulanmıştır. Verilerin iki tanesi test için ayrılmıştır. YSA aynı verilerle 7 defa eğitilmiş ve her eğitim için farklı veri setleri test olarak seçilmiştir. YSA sınıflandırma sonuçları aşağıdaki Çizelge 4.6'da verilmiştir.

Bu sonuçlara göre YSA'nın frekans spektrumu 100 örneklik verileri sınıflandırma özeti aşağıdaki gibidir.

- 10 adet seçilmiş sağlıklı sınıfına ait verinin doğru sınıflandırılma sayısı: 8  
Başarı yüzdesi %80
- 10 adet seçilmiş sağlıklı sınıfına ait verinin yanlış sınıflandırılma sayısı: 2  
Başarısızlık yüzdesi %20
- 4 adet seçilmiş klik sınıfına ait verinin doğru sınıflandırılma sayısı: 3  
Başarı yüzdesi %75
- 4 adet seçilmiş sağlıklı sınıfına ait verinin yanlış sınıflandırılma sayısı: 1  
Başarısızlık yüzdesi %25
- 14 seçilmiş her iki gruba da ait verinin doğru sınıflandırılma sayısı: 11  
Başarı yüzdesi %78.6
- 14 seçilmiş her iki gruba da ait verinin yanlış sınıflandırılma sayısı: 3  
Başarısızlık yüzdesi %21.4

Bu sonuçlara göre uygulanan analiz ve sınıflandırma yöntemi %78.6 oranında başarılı olmuştur. Sonuçların Karmaşıklık Matrisi Şekil 4.7'de verilmiştir. Bu başarı oranı uygulanan yöntemin özellikle gürültülü ortamlar içerisinde ve gerçek muayene ortamında alınan yani laboratuvar dışında toplanan verilerin sınıflandırılması açısından



başarılı sayılabilir. Fakat diğer bazı sınıflandırma algoritmaları ile karşılaştırıldığında ne yazık ki yeterli bir başarı oranı olarak kabul edilmemektedir. Buna göre veri işlemede kullanılan YSA algoritmaları bu uygulamaya özel olarak sınıflandırma açısından yetersiz kalmaktadır. Bunun en temel sebepleri arasında normalde YSA algoritmalarının doğru eğitim sürecinden geçebilmeleri için bol miktarda eğitim verisine ihtiyaç duymalarıdır. Maalesef veri toplama aşaması sırasında yaşanan sıkıntılar toplanan verilerin sayısının düşük kalmasına sebep olmuştur. Bu da YSA performansını düşürmüştür. Bu uygulama için YSA kullanımının yeteri kadar başarılı olmadığı sonucuna varılabilir.

**Çizelge 4.6** YSA eğitim sonuçları

| Deneme Sayısı | Veri Numarası | Sınıf Çeşidi | Sınıflandırma | Veri Numarası | Sınıf Çeşidi | Sınıflandırma |
|---------------|---------------|--------------|---------------|---------------|--------------|---------------|
| 1             | 3             | Sağlıklı     | Doğru         | 15            | Klik         | Yanlış        |
| 2             | 2             | Sağlıklı     | Doğru         | 10            | Sağlıklı     | Doğru         |
| 3             | 2             | Sağlıklı     | Doğru         | 15            | Klik         | Doğru         |
| 4             | 3             | Sağlıklı     | Yanlış        | 11            | Sağlıklı     | Doğru         |
| 5             | 7             | Sağlıklı     | Yanlış        | 18            | Klik         | Doğru         |
| 6             | 6             | Sağlıklı     | Doğru         | 18            | Klik         | Doğru         |
| 7             | 8             | Sağlıklı     | Doğru         | 10            | Sağlıklı     | Doğru         |

|              |   |            |            |                |
|--------------|---|------------|------------|----------------|
| Çıkış Sınıfı | H | 3<br>%21.4 | 2<br>%14.3 | %60<br>%40     |
|              | S | 1<br>%7.14 | 8<br>%57.1 | %88.9<br>%11.1 |
|              |   | %75<br>%25 | %20<br>%80 | %78.6<br>%21.4 |
|              |   | H          | S          | Hedef Sınıf    |

**Şekil 4.7** Frekans tabanlı özellik çıkarımına dayanan yöntemlerin tipik karmaşıklık matrisi

Sınıflandırma başarısının artırılması için yapılabilecek işler arasında ilk sırayı veri sayısının artırılması gelmektedir. Bu iki türlü olabilmektedir, birincisi direkt veri artırma yöntemidir. İkincisi ise verilerin korelasyon analizi ve ileri gürültü bastırma

yöntemleri gibi yöntemlerle hali hazırda mevcut elimine edilmiş veriler içerisinde temizlenerek ayrıştırılmasıdır. Veri sayısının yetersizliği durumunda istatistiksel ayrıştırma ve sınıflandırma yöntemleri veya hibrid yöntemler tercih edilebilir. Diğer bir çözüm ise yukarıdaki sınıflandırma yöntemlerinde özellik çıkarım sürecinin değiştirilmesi ki bu çözüm Bölüm 4.4'te irdelenmiştir. Bölüm 4.5'te ise derin öğrenme tabanlı sınıflandırma süreci incelenmiştir.

#### **4.4. İstatistiksel Tabanlı Özelliklerin Çıkarımı ve YSA ile Sınıflandırılması**

##### **4.4.1. Ses verilerinin istatistiksel özellik çıkarımı**

TME seslerinin istatistiksel öznitelik çıkarımı yapılmış ve yapay sinir ağları kullanarak sınıflandırma işlemi gerçekleştirilmiştir. Veriler, tecrübeli bir diş hekiminin gözetimi altında, gerçek klinik ortamında toplanmıştır. Ses verileri bir önceki bölümde kullanılan verilerle aynıdır. Veri toplama ve çıkarma işlemi önceki bölümlerde detaylı bir şekilde anlatılmıştır.

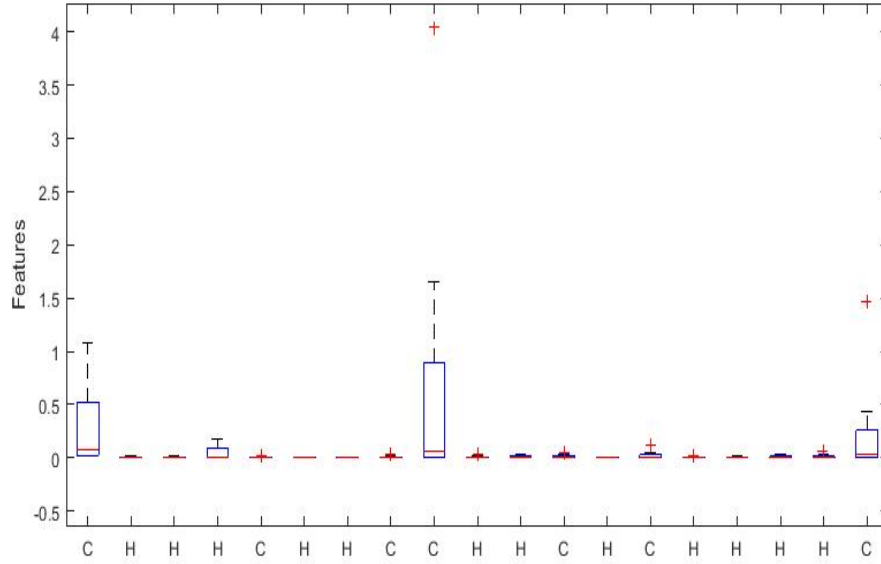
Yapay sinir ağını eğitmek ve test etmek için istatistiksel öznitelik çıkarımı kullanılmıştır. İstatistiksel öznitelikler, insan vücudundan çıkan seslerin sınıflandırılması için Altunkaya ve ark. (2010b) gibi bazı çalışmalarda kullanılmıştır. Takip eden kısımda sunulan metot, veri ön işleme olmaksızın oldukça iyi sonuçlar vermiştir.

Veri kümesi önceki bölümden alınmıştır. Verilerin kaydedilme süreci ve araçlarının detaylı açıklaması önceki bölümlerde ayrıntılı olarak verilmiştir. Veri kümesindeki her veri 5 saniyelik, 24 bit, 51200 KHz örnekleme hızı özellikli verilerdir, bu da toplam 256000 örnek oluşturur.

Şekil 4.8'de görüldüğü üzere verilerin istatistiksel kutu grafiği sağlıklı ve hasta verilerin arasındaki farkı daha açık göstermektedir. Yukarıda bahsedildiği gibi verilerin istatistiksel özelliklerinin hastanın durumunun sınıflandırılmasında kullanılmasına karar verilmiştir. Örneklenmiş verileri kullanarak TME ses kayıtlarının temel istatistiksel özellikleri çıkarılmıştır:

- Ortalama
- Standart sapma
- Varyans
- Örneklenmiş verinin maksimum ve minimum değerleri

Her kayıt için, sağ ve sol eklemlerden alınan örnekler kullanılarak, 8 öznelik elde edilmiştir. Özellik çıkarma işlemi, MATLAB yazılımı kullanılarak gerçekleştirilmiştir.

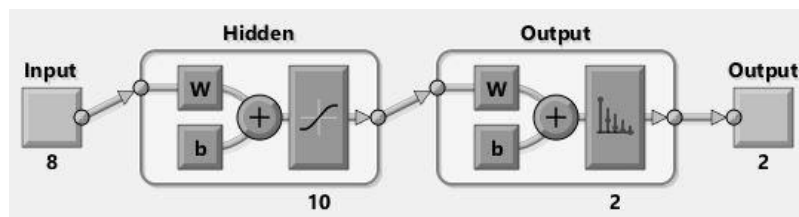


Şekil 4.8: Temel veri sınıflarının kutu grafiği (C: sağlıklı, H: hasta)

#### 4.4.2. İstatistiksel özellik çıkarımı yapılan verilerinin YSA ile sınıflandırılması

Gereken öznelikler çıkartıldıktan sonra MATLAB YSA aracı kullanılmıştır. Sınıflandırma yapılırken bu tür problemlerin birçoğunda daha verimli olduğu için yapay sinir ağları tercih edilmiştir.

Şekil 4.9’da görüldüğü gibi çok katmanlı, ileri beslemeli bir yapay sinir ağı, MATLAB’ın örüntü tanıma programı aracılığı ile oluşturulmuştur. MATLAB’ın örüntü tanıma programında saklı katmanlar için aktivasyon fonksiyonu olarak sigmoid ve çıktı vermeden önce softmax katmanı bulunur. Kullanılan sınıflandırıcıda 10 gizli katman bulunmaktadır, ancak daha az sayıda gizli katman kullanarak da istenen sonuca ulaşılmıştır. Çizelge 4.7’de sağlıklı ve hasta veri dağılımı görülmektedir.



Şekil 4.9 MATLAB’ın çok katmanlı ileri beslemeli YSA diyagramı

**Çizelge 4.7** Sağlıklı ve hasta veri dağılımı

|               | Veri sayısı |
|---------------|-------------|
| TMD hastası   | 28          |
| Sağlıklı kişi | 48          |
| Toplam        | 76          |

Daha sonra ağ, önceden hazırlanmış istatistiksel öznitelikler kullanılarak eğitilmiştir. Mevcut 76 verinin 54'ü sınıflandırıcıyı eğitmek için kullanılmış, kalan 22 veride doğrulama ve test aşaması için eşit şekilde ayrılmıştır. Çizelge 4.8'de veri seti dağılımı verilmiştir.

**Çizelge 4.8** Her aşama için veri dağılımı

|           | Kullanılan veri sayısı |
|-----------|------------------------|
| Eğitim    | 54                     |
| Doğrulama | 11                     |
| Test      | 11                     |

Her eğitim süreci için öznitelikler rastgele bölündüğünden her eğitim süreci sonunda farklı sonuçlar alınmıştır. Şekil 4.10'daki karmaşıklık (hata) matrisi, Sınıf1 hasta ve Sınıf2 sağlıklı etiketi olmak üzere, en fazla elde edilen sonuçlardan birini özetlemektedir:

- Matrisin satırları yapay sinir ağının tahmin ettiği sınıfları gösterirken, sütunları verilerin gerçek sınıflarına karşılık gelir.
- Matrisin ortasından geçen çapraz çizgi, doğru sınıflandırılmış özelliklerin yüzdesini ve miktarını gösterir. En alttaki hücre ise genel doğruluk ve hata oranını gösterir.
- Son sütun doğru ve yanlış sınıflandırılmış tahminlerin yüzdesini göstermektedir.
- En alt satır doğru veya yanlış sınıflandırılmış özelliklerin yüzdesini ve duyarlılık oranını gösterir.

Bu durum şu şekilde yorumlanabilir: 1. Sütunda 28 TMD hastasından 20'si doğru bir şekilde hasta olarak sınıflandırılmış, 8 tanesi ise sağlıklı olarak

sınıflandırılmıştır. 2. Sütunda 48 verinin veya 1. sınıfa ait tüm verilerin hepsi doğru; 2. sınıf etiketini taşıyan 56 verinin 8'i yanlış, 48'i doğru sınıflandırılmıştır.

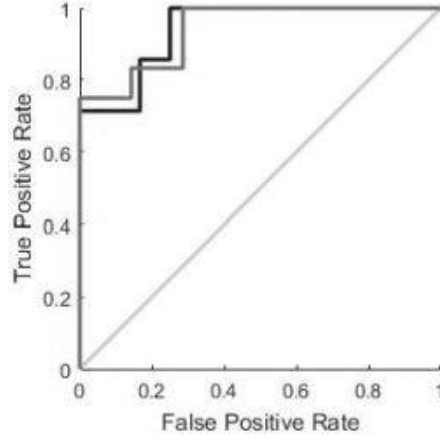
|              |   |                |              |                |
|--------------|---|----------------|--------------|----------------|
| Çıkış Sınıfı | H | 20<br>%26.3    | 0<br>%0.0    | %100<br>%0.0   |
|              | S | 8<br>%10.5     | 48<br>%63.2  | %85.7<br>%14.3 |
|              |   | %71.4<br>%28.6 | %100<br>%0.0 | %89.5<br>%10.5 |
|              | H | S              | Hedef Sınıf  |                |

Şekil 4.10 Karmaşıklık matrisi

Şekil 4.11'deki ROC eğrisinde görüldüğü gibi grafik, TMD hastalarını saptama olasılığının yüksek olduğu, doğru pozitiflik oranına yaklaştığı için eğitilen sınıflandırıcının hata oranının az olduğunu söyleyebiliriz.

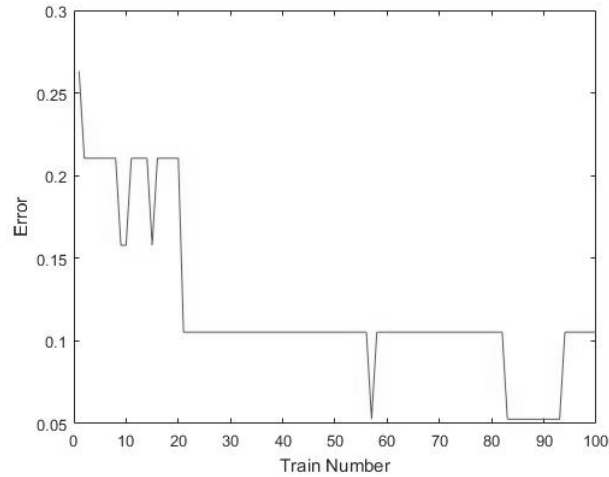
Genel sonuçlar, eğitim ve test süreçlerinin birçok kez tekrarlanması ile elde edilmiştir. Sınıflandırma ve test işlemi 100 kere tekrarlanmış, her tekrarda eğitim, test ve doğrulama verileri rastgele seçilmiş ve sonuçlar not edilmiştir. Eğitim sürecinin tekrarlanmasının amacı YSA modeli için gerçekçi sonuçlar elde etmektir. Bazı YSA modelleri, eğitim algoritması tarafından seçilen veriler, veri setinin tamamından çok küçük miktardaki ideal koşulları temsil edeceğinden gerçek olamayacak kadar iyi sonuçlar vermiştir. Bu şekilde alınan sonuçlar, eğitim ve test aşamasının birçok kez tekrarlanması ile ortadan kaldırılmış ve uzun vadede daha gerçekçi modeller elde edilmiş ve sonuçlar alınmıştır.

Şekil 4.12'de, YSA'nın 100 kez çalışması sonucu alınan hata yüzdelerinin grafiği verilmiştir. Burada bazı durumlarda rastgele seçilen verilerin çok küçük miktardaki ideal koşulları temsil etmesinden dolayı sonuçların gerçek dışı bir şekilde düşük hata yüzdesine sahip olduğu görülmektedir. Daha iyi sonuçlar elde etmek için sınıflandırıcı çok kez eğitimden geçirilmiş ve alınan doğruluk yüzdelerinin ortalaması ~ %87-89, genel doğru sınıflandırma oranı olarak kabul edilmiştir.



Şekil 4.11 ROC Eğrisi

Önceki bölümlerde anlatılan veri toplama ve özellik çıkarma aşamasından sonra veriler için ilk önce, YSA modeli eğitilerek sınıflandırıcı oluşturulmuştur. Ses verileri için ön işleme yapılmamasına rağmen gelecek vadeden sonuçlar alınmıştır. Daha doğru sonuçlar elde etmek için eğitim süreçleri 100 kez tekrarlanmış ve doğru sınıflandırma yüzdesi kaydedilmiştir. Bu aşamadan sonra ortalama sınıflandırma yüzdesi yaklaşık %87-89 olarak hesaplanmıştır. Bu sonuç bir önceki frekans tabanlı sınıflandırma ile elde edilen %78.6'lık sonuçtan daha yüksektir.



Şekil 4.12 100 farklı eğitilmiş YSA modelinin hata yüzdeleri

#### 4.5. Derin Öğrenme Tabanlı Sınıflandırma

Derin öğrenmenin resim tanıma başarısından dolayı araştırmamızın bu bölümünde derin öğrenme tabanlı sınıflandırma kullanılmıştır (Beale ve ark. 2018; İnik ve Ülker 2017). Tek boyutlu ses verileri spektrogram dönüşümü yapılarak iki boyutlu renkli

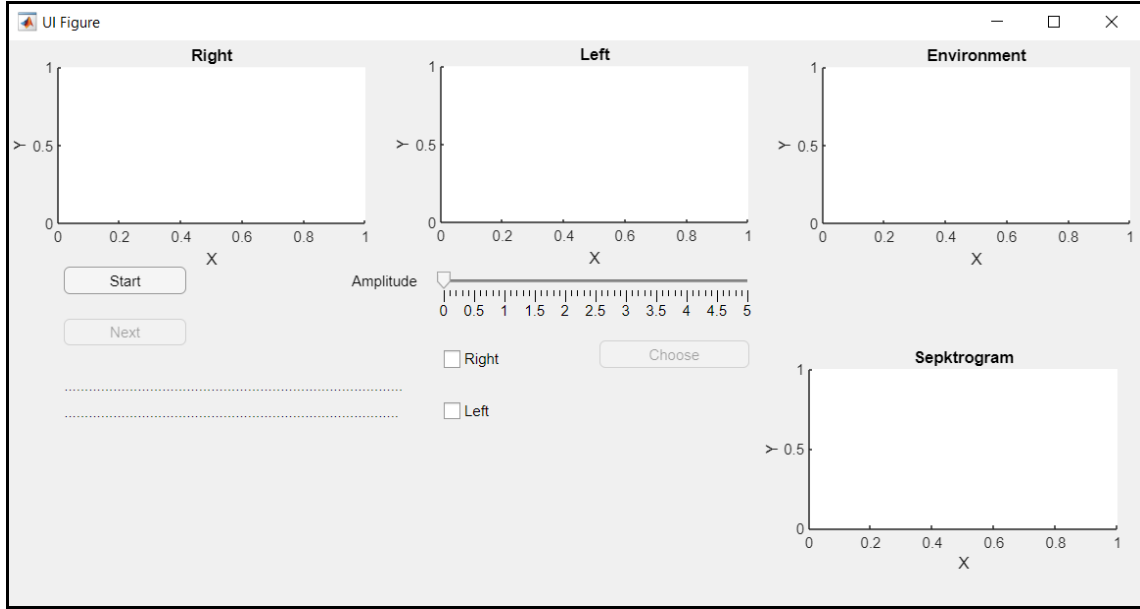
(RGB) resim verilerine dönüştürülmüş ve bu resim verileri kullanılarak ses sınıflandırılması yapılmıştır. Bu yapılırken MATLAB® 2018b Deep Learning Toolbox kullanılmıştır.

#### 4.5.1. Kullanılan verilerin seçimi

Derin öğrenme algoritmaları sağlıklı çalışabilmeleri için yüksek miktarda veriye ihtiyaç duymaktadır (Beale ve ark. 2018; İnik ve Ülker 2017). Daha önceden kaydedilmiş ses verilerinin kısıtlı sayıda olmasından dolayı veri sayısının artırılması gerekmektedir. Bunu yapmanın birçok yolu vardır. Bunun en sağlıklı olan yolu hasta sayısını arttırmak ve bu kişilere ait verilerin kaydedilerek hastalıklı insanlara ait veri sayısını arttırmaktır. Bu durum birkaç ay gibi kısa bir sürede üstesinden gelinecek bir durum olmayıp birkaç yıl süren uzun bir süre alacaktır. Bunun yerine kaydedilmiş verilerin tekrar gözden geçirilmesi bir çözüm olarak sunulmuştur. Veri kayıtlarının hem sağ hem de sol kulak boşluğundan alınması aslen veri sayısının artırılmasında oldukça faydalı olmuştur. Bütün veriler baştan aşağı incelenmiş, önceden yapılan sınıflandırmada hatalı olduğu düşünülen verilerin bir kısmı tekrar gözden geçirilerek geçerli veri sınıflandırılmasına sokulmuştur. Böylece veri sayısında bir miktar artma meydana gelmiştir.

Ses verilerinin 5 saniyelik olduğu, verilerin alınması ile ilgili Bölüm 4.1’de not edilmişti. Ham veriler incelendiğinde ses kaydının 5 saniyelik olduğu kolaylıkla görülebilir. Bu verilerde birkaç durum ortaya çıkmaktadır. Birinci durum yalnızca çenenin açılma veya kapanması esnasında ortaya çıkan eklem seslerinin olduğu verilerden, ikinci durum hem açılırken hem de kapanırken ortaya çıkan eklem seslerinin olduğu verilerden oluşmaktadır. Bu durum veri sayısının artırılması için bir başka taban oluşturmuş ve hem açma hem de kapama esnasında oluşan her iki sesin ayrılarak iki farklı veriye dönüştürülmesi mümkün olmuştur. Buna ilaveten aktif ses süreleri incelendiğinde bu sürelerin hiçbir zaman 1 saniyeyi geçmediği gözlemlenmiştir. Bu gözlemden yola çıkarak her bir verinin 2 saniyelik maksimum üç mümkün parçaya bölünebileceği görülmüştür. Her bir veri aktif ses süreci başta, ortada ve sonda olmak üzere bir ila üç parçaya ayrılmıştır. Bu durumdan hareketle MATLAB® yazılımı kullanılarak basit bir ara yüz programı geliştirilmiştir. Her ne kadar bu işi otomatik olarak yapmak mümkün olsa da program manüel olarak yani gözlemci müdahalesiyle çalışmaktadır. Bu aşamasında veri ayrıştırmanın elle müdahaleye ihtiyaç duymaktadır. Bu programın ara yüzü Şekil. 4.13’de görülmektedir. Geliştirilen program sayesinde

mevcut verilerden bir ila üç arasında yeni veriler oluşturmak mümkün olmuştur. Böylece kısıtlı miktardaki veri sayısından derin öğrenme için gerekli veri miktarını elde etmek mümkün hale gelmiştir. Sonuç olarak 168 sağlıklı sınıf verisi ve 73 hasta sınıf verisi elde etmek mümkün olmuştur.



Şekil 4.13 MATLAB ile veri kırpmaya amacı ile geliştirilen programın ara yüzü

Bir sonraki adım verilerin bant genişliğinin azaltılmasıdır. Verilerin spektrogramlar ve tayflar incelendiğinde yüksek frekans bileşenlerinin çok fazla olmadığı görülmektedir. Buradan yola çıkarak yüksek frekans bilgilerinin atılmasının sonucu çok fazla etkilemeyeceği düşünülmüştür. Daha sonra 100 Hz - 10 KHz bant geçiren filtre vasıtasıyla 51200 KHz'de örneklenmiş ses verilerinin yüksek frekans bileşenleri ve 50 Hz şebeke gürültüsünden etkilenebilecek alçak frekans bileşenleri filtrelenerek atılmıştır. Takip eden adım ise örnekleme frekansının 20 KHz'e düşürülmesidir. Çünkü 10000 Hz ila 25600 Hz arası frekans bandı filtrelemeden dolayı hiçbir veri içermemektedir. Bu aynı zamanda bilgisayar ve algoritmaya girilecek veri büyüklüğünün azaltılmasına yol açacağından algoritmanın daha hızlı çalışmasını da sağlayacaktır. Örnekleme frekansının 20KHz'e düşürülmesiyle verinin tek boyutlu işlem süreci noktalanmış olmaktadır. Bu süreç anlatıldığı üzere birçok sayısal sinyal işleme algoritmasını ve sürecini içermektedir.

Gelinen bu aşamada verilerin içeriğini temsil eden iki boyutlu resimlere dönüştürülmesi gerekmektedir. Bununla ilgili olarak literatürde derin öğrenme



algoritmalarının resim sınıflandırma süreçlerinde daha başarılı olduğu bilgisi yer almaktadır (Beale ve ark. 2018; İnik ve Ülker 2017). Bu yüzden sınıflandırma işleminin resim üzerinden yapılabileceği düşünülmüş ve benzer uygulamaların olup olmadığına dair literatür taraması yapılmıştır. Bunların içinde McLoughlin ve ark. (2015), Salamon ve Bello (2015) ve Salamon ve Bello (2017) çalışmaları incelenmiştir. Özellikle bazı kaynaklarda (Graves ve ark. 2013; Smales 2018) uygulanması planlanan sisteme benzer çalışmalar yapıldığı görülmüştür. Bu çalışmada daha önce işlenmiş ses verilerinin spektrogramının hesaplanarak resim verilerine çevrilmesi, çevrilen verinin png veya jpg formatında kaydedilmesi ve daha sonra bu resimlerin sınıflandırması işlemleri yapılmıştır.

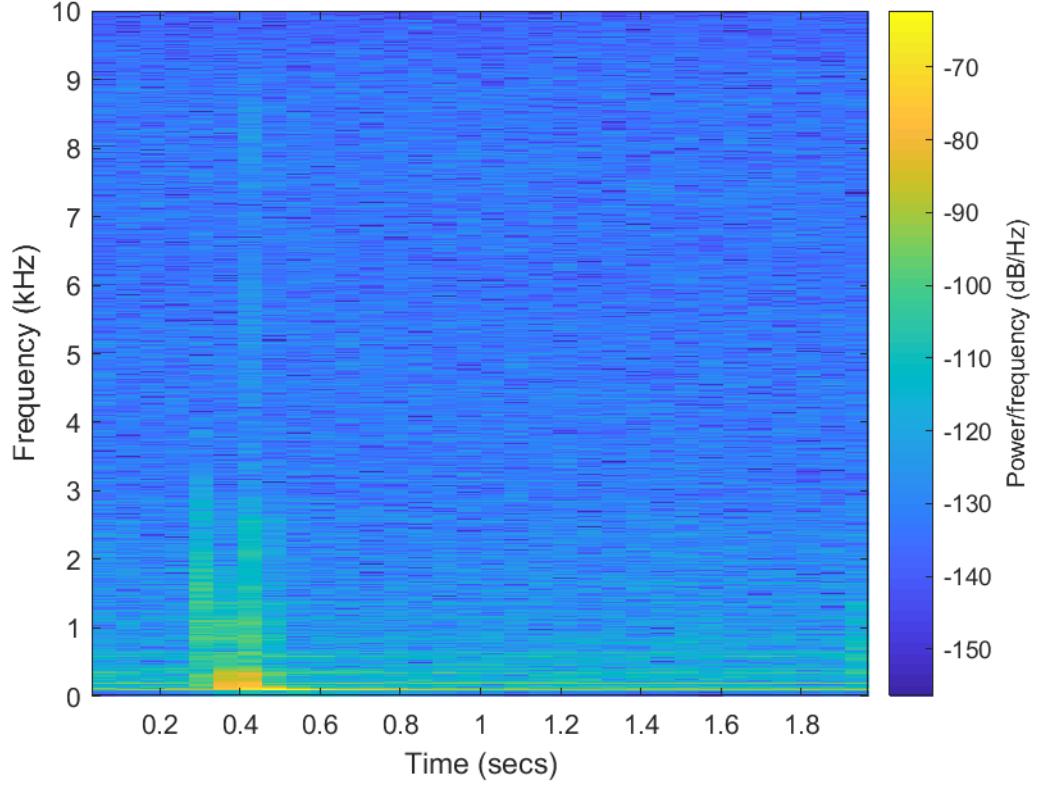
Bu süreç boyunca her verinin MATLAB yazılımı kullanılarak spektrogram verileri hesaplanmış ve daha sonra resim formatına çevrilerek kaydedilmiştir. Bu aşamada resim formatı Şekil 4.14'te görüldüğü gibi 656x875 boyutunda RGB tabanlı renkli resim şeklindedir. Fakat resmin esas verileri orta alanda olan renkli alan olduğundan kenarlardan kesme işlemi uygulanarak resim yalnızca veri kısmını içeren Şekil 4.15'te tipik örnekleri görülen sağlıklı ve hasta bireylere ait resme dönüştürülmüş hali görülmektedir. Bu aşamada resim PNG formatında 521x593 boyutlu RGB tabanlı renkli resme dönüştürülmüştür. Bu işlemin yapılmasındaki en önemli amaç yine derin öğrenme ağının girişine uygulanacak veri boyutunun küçültülmesi ve algoritma ve bilgisayar yükünün azaltılmaya çalışılmasıdır.

Bu şekilde 168 sağlıklı bireylere ve 73 adet hasta bireylere ait temel bir veri tabanı oluşturulduktan sonra derin öğrenme ağları vasıtasıyla sınıflandırma yapılmıştır.

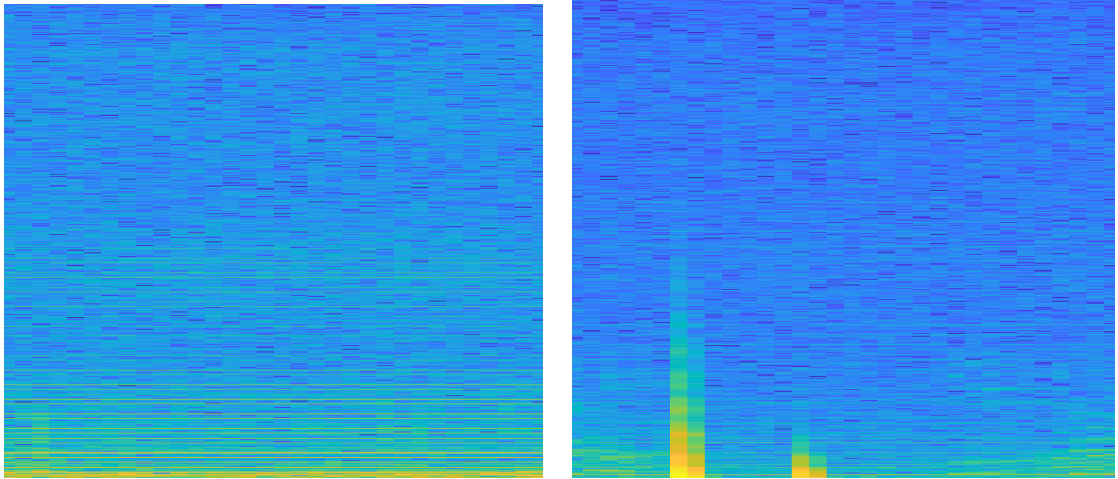
#### **4.5.2. Verilerin işlenmesi ve sınıflandırılması**

Önceki bölümlerde ses verilerini spektrogram tabanlı yani verilerin hem zaman hem de frekans değişimini gösteren resimlere çevrildiği anlatılmıştır. Bu şekilde bir temel küme oluşturulduktan sonra Şekil 4.16'te görülen ağa benzer bir resim sınıflandırma ağı geliştirilmiştir. Bu ağın giriş katmanında bir adet resim giriş katmanı (Image Input Layer) kullanılmıştır. Giriş katmanının çıkışına 3 adet ara katman birbirini takip edecek şekilde bağlanmıştır. Her bir ara katmanda sırasıyla bir adet 2D konvolüsyon katmanı (convolution 2D layer), bir adet doğrultulmuş doğrusal birim katmanı (Rectified Linear Unit Layer, ReLU Layer) ve bir adet 2D maksimum havuzlama katmanı (maxpooling2D layer) bulunmaktadır. Ara katmanları çıkış

katmanları olan tam bağlantılı katman (fully connected layer), yumuşak maksimum katman (softmax layer) ve sınıflandırma çıkış katmanı (classoutput layer) izlemektedir.



Şekil 4.14 656x875 boyutunda MATLAB spektrogram sonucu PNG formatlı hasta verisi

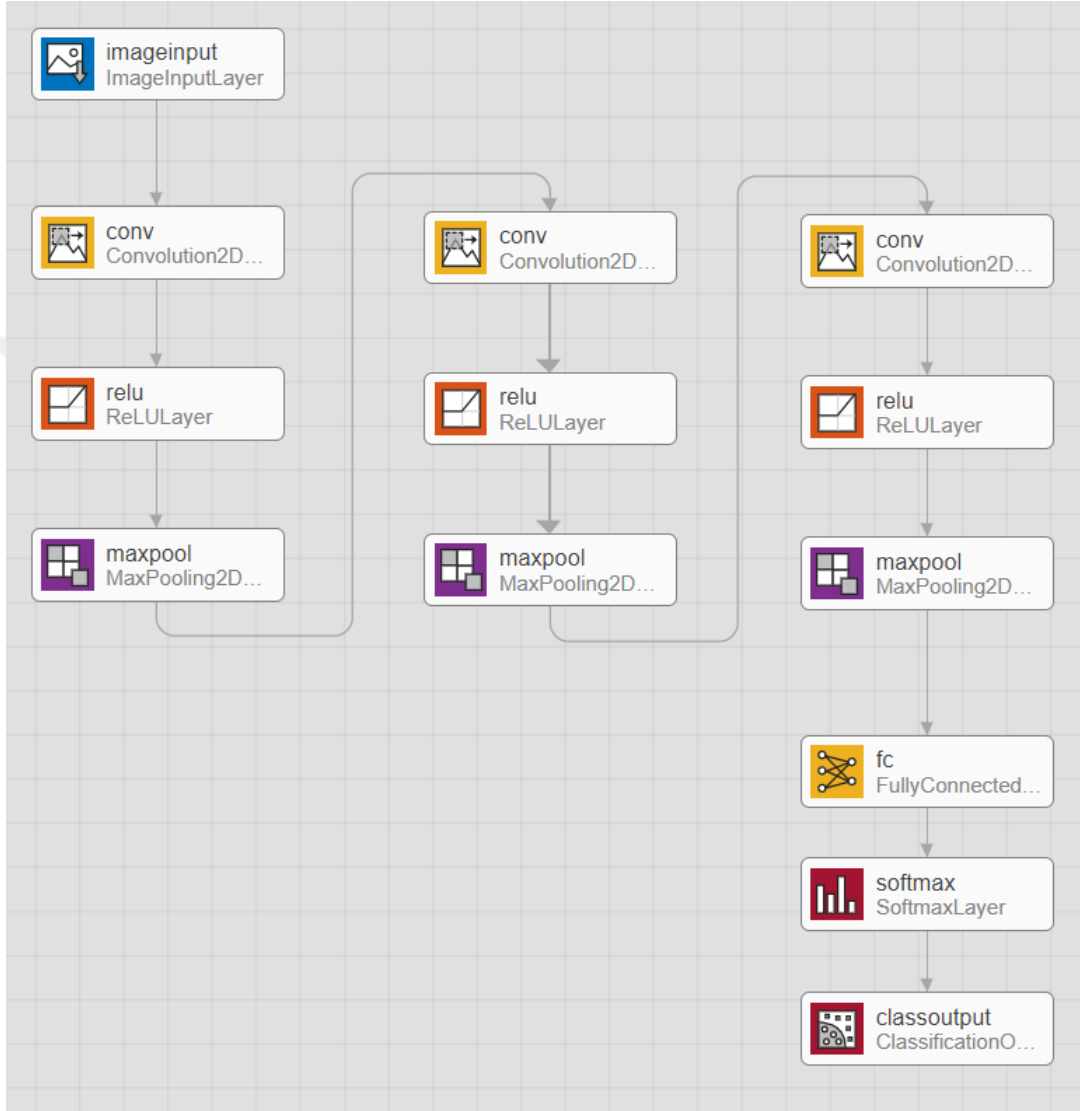


Şekil 4.15. a) Tipik Sağlıklı Birey Verisi

b) Tipik Hasta Birey Verisi

MATLAB 2018 deep learning toolbox üzerinden bir derin öğrenme algoritması geliştirilmiş ve ağ derlenmiştir. Seçilen verilerle ve varsayılan parametrelerle bir eğitim

süreci başlatılmıştır. İlk başlarda derin öğrenme ağı %60 civarında bir başarı yakalamıştır. Eğitimin başarı yüzdesini arttırmak için derin öğrenme ağının çeşitli parametreleri değiştirilmiş, başlangıçta düşük olan sınıflama başarısı zamanla ve onlarca denemenin ardından kademeli olarak %90'nın üzerine çıkarılabilmektedir.



Şekil 4.16 Tipik Resim Sınıflandırma için Derin Öğrenme Ağı

Geliştirilen derleme algoritması Şekil 4.17’de verilmiştir. Giriş parametresi giriş resminin 521x593 boyutlarında 3 renkli (RGB) resim olduğunu belirtmektedir. İlk 2 boyutlu konvolüsyon katmanının parametreleri 3x3 boyutlu 8 filtreden ve her bir kenarına 1 adet sıfır eklemekten oluştuğunu göstermektedir. Dikkat edilirse bir birini takip eden katmanlarda 2 boyutlu konvolüsyon katmanının parametreleri ilk önce 5x5 boyutlu 16 filtreden, en sondaki konvolüsyon katmanının parametreleri 7x7 boyutlu 32

filtreden oluşmaktadır. 2 boyutlu büyük seçen havuzlama katman parametreleri 3x3 boyutlu ve adım (Stride) ebadı 3x3 olarak seçilmiştir. Tamamen bağlantılı katman çıkış parametresi 2 olarak seçilmiştir, çünkü çıkış iki sınıftan oluşmaktadır.

Şekil 4.17’de verilmiş özellikteki parametrelerle çalışan ağ onlarca kez çalıştırılmıştır. Bu algoritmaya giriş olarak 521x593 ebatlarında 168 sağlıklı bireye ait ve 73 adet hasta bireye ait PNG spektrogram verisi taşıyan RGB resimler kullanılmıştır. Algoritmanın oldukça yüksek sınıflandırma başarısı verdiği görülmüştür. Bu sınıflandırma başarısı bazen %100’e kadar çıkmakta ve genellikle %90 üstünde olmaktadır. Başarı yüzdesi yani doğruluk oranı yüzde olarak aşağıdaki gibi hesaplanır;

$$\text{Doğruluk} = \% \frac{\text{Öngörülerle test verilerinin eşitlik olma durumlarının sayısı}}{\text{Test verileri sayısı}} \times 100 \quad (4.1)$$

Şekil 4.18’de tipik bir eğitim süreci görülmektedir. Başarı oranı %80’in altındaki durumlar incelendiğinde hasta ve sağlıklı birey verilerinin birbirine yakın sayıda olmadıklarından problem ortaya çıktığı görülmüştür. Sağlıklı bireylere ait veri sayısı hasta bireylere ait veri sayısının iki katından fazladır. Sistemin algoritma koşturma esnasında bazen bu eğilime takıldığı görülmektedir. Bu durumun ortadan kaldırılması için hasta bireylere ait veri sayısını arttırmak gerekmektedir. Bu aşamada bu mümkün olmadığından sağlıklı bireylere ait veri sayısını hasta bireylere veri sayısına yaklaştırmak amacıyla sağlıklı birey sayısı 100’e düşürülmüştür (Yan ve ark. 2015). İlk başta aynı deneğe ait benzer veriler silinmiştir. Devamında sınıflandırma algoritmasını zorlamak amacıyla sınıflandırması kolay veriler sağlıklı birey havuzundan atılmıştır. Böylece hem eğilimi ortadan kaldırma hem de sınıflandırma algoritmasını zorlama amaçlı 100 adet sağlıklı birey verisinden oluşan bir havuz oluşturulmuştur. Algoritmayı hızlandırmak ve yakınsamasını sağlamak amacıyla her reLU katmanının önüne bir batchNormalizationLayer (Yığın Normalizasyon Katmanı) eklenmiştir. Bu ağa ait katman tanımı Şekil 4.19’de görülmektedir. Bu şekilde algoritma birçok defa koşturulmuş ve ortalamada %94.5 sınıflandırma başarısı elde edilmiştir. Şekil 4.20’de tipik bir eğitim süreci görülmektedir. Dikkat edilirse yakınsama Şekil 4.18’e göre daha düzenli ve hızlıdır.

Yeniden tasarlanmış derin öğrenme ağının ortalama başarısı %90’ların üzerine çıkmıştır. Bu aşamada elde edilen derin öğrenme sınıflandırıcısı daha tutarlı ve kararlı sonuçlar üretmiştir. Böylece hem oldukça yüksek bir başarı oranı elde edilmiştir hem de koşturma esnasında elde edilen sonuçların tutarlılığı artmıştır. Başarı oranını etkileyen

faktörlerden birinin de Spektrogram verilerinin hem zaman hem de frekans bilgisini içerdiğini not etmek gerekmektedir.

```

clear all

layers_matlab = [ ...
    imageInputLayer([521 593 3])
    convolution2dLayer(3,8,'Padding',1)
    reluLayer
    maxPooling2dLayer(3,'Stride',3)
    convolution2dLayer(5,16,'Padding',1)
    reluLayer
    maxPooling2dLayer(3,'Stride',3)
    convolution2dLayer(7,32,'Padding',1)
    reluLayer
    maxPooling2dLayer(3,'Stride',3)
    fullyConnectedLayer(2)
    softmaxLayer
    classificationLayer];

imds_sd = imageDatastore(fullfile(...
    'E:\UGUR\MyDocuments\MATLAB\Doktora_DeepLearning'), ...
    'IncludeSubfolders',true,'FileExtensions','.png', ...
    'LabelSource','foldernames')
numTrainingFiles =50;
[imdsTrain,imdsTest] =
splitEachLabel(imds_sd,numTrainingFiles,'randomize');

options_matlab = trainingOptions('sgdm', ...
    'MaxEpochs',12,...
    'InitialLearnRate',1e-3, ...
    'Verbose',0, ...
    'Plots','training-progress','MiniBatchSize',4,
    'ExecutionEnvironment','gpu');

trainedNet = trainNetwork(imds_sd,layers_matlab,options_matlab)

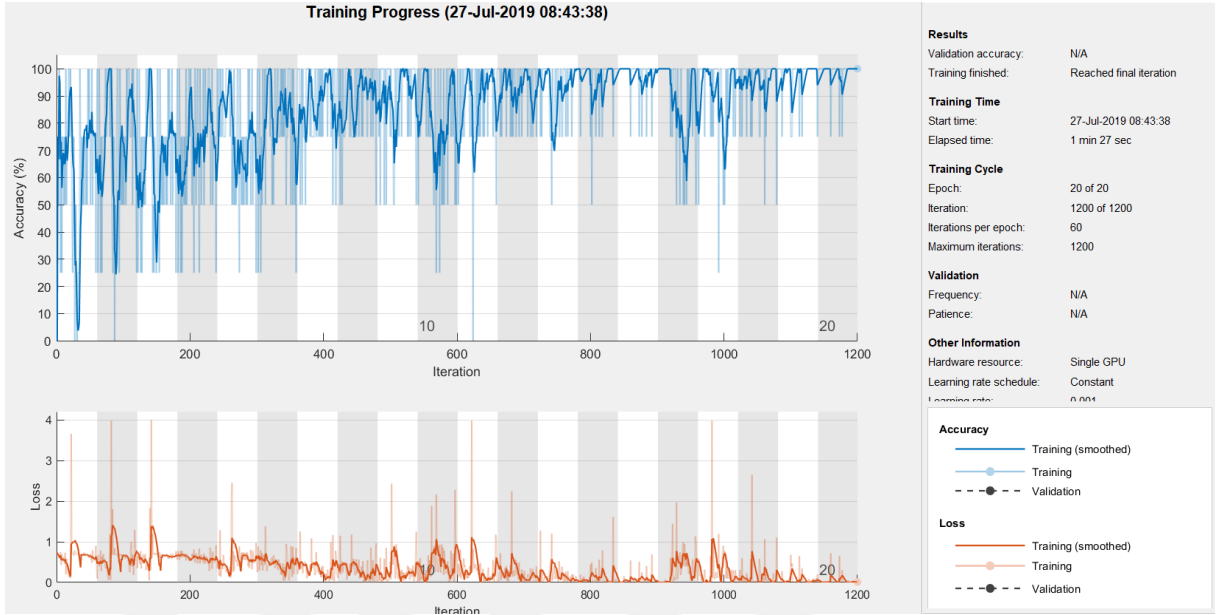
YPred = classify(trainedNet,imdsTest);
YTest = imdsTest.Labels;
cf = confusionmat(YTest, YPred)
% Calculate the accuracy.
% Accuracy is the ratio of the number of true labels
% in the test data matching the classifications from
% |classify|, to the number
% of images in the test data.
accuracy = sum(YPred == YTest)/numel(YTest)

```

Şekil 4.17 İlk aşamada kullanılan derleme algoritması ve parametreleri

Elde edilen sonuçlar daha önceki aşamalarda elde edilen sonuçlardan daha tutarlı ve başarılıdır. Bu durum derin öğrenme ağlarının resim sınıflandırmadaki başarılarının ses verilerinde de elde edilebileceğini göstermektedir. Ses verilerinin frekans bölgesinde işlenmesi, fazlalık frekansların atılması ve bilginin belli bir frekans bandında yoğunlaştırılması bu sürece katkı sağlamıştır. Derin öğrenme ağlarını ses verilerinde

kullanmak için basitçe ses verileri içerisinde hem zaman hem de frekans bilgilerinin yer aldığı spektrogram çevrimi yeterli olmuştur.



Şekil 4.18 Tipik başarılı bir derin öğrenme ağının eğitim süreci grafiği

```

layers_matlab = [ ...
    imageInputLayer([521 593 3])
    convolution2dLayer(5,8,'Padding',1)
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3,'Stride',3)
    convolution2dLayer(7,16,'Padding',1)
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3,'Stride',3)
    convolution2dLayer(9,32,'Padding',1)
    batchNormalizationLayer
    reluLayer
    maxPooling2dLayer(3,'Stride',3)
    fullyConnectedLayer(2)
    softmaxLayer
    classificationLayer];

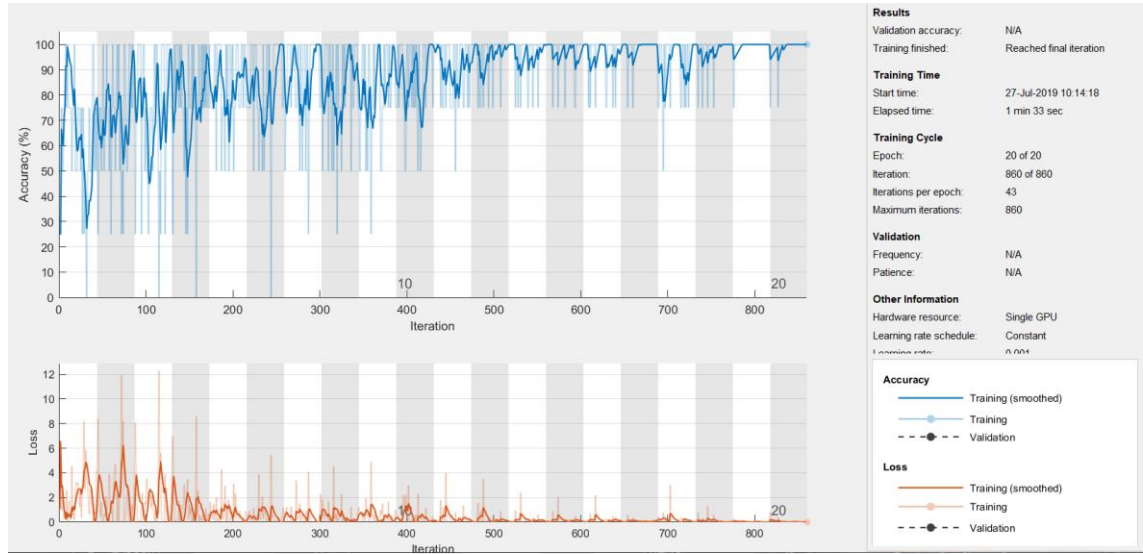
```

Şekil 4.19 batchNormalizationLayer eklenerek iyileştirilmiş derin öğrenme ağı parametreleri

#### 4.4. Sınıflandırma Yöntemlerinin Karşılaştırılması

Buraya kadar sunulmuş üç farklı yöntemin geniş tartışmasının ardından bu bölümde bu yöntemlerin kıyaslanması yapılmıştır. Frekans tabanlı özellik çıkarımına dayanan YSA sınıflandırma başarısı ortalama %78.6 olmuştur. Bu yöntemin uzun dönem kararlılığı yeteri kadar incelenememiştir ama başarı oranının %80'nin üzerine çıkmadığı gözlemlenmiştir. İstatistiksel tabanlı özellik çıkarımına dayanan YSA

sınıflandırmaya başarısı genelde %80'nin üzerindedir. Bu yöntemin uzun dönem kararlılığı incelenmiş ve ortalamada %87 ile %89 başarılı olduğu görülmüştür. Derin öğrenme yöntemlerine dayanan ilk çalışma ağının sınıflandırma başarısı ise genel olarak %70 ile %100 arasındadır. Bu yaygınlığının sebebinin eğilimleme olduğu yukarıda anlatılmıştır. Bu yöntemin eğilimlemeye karşı iyileştirilmesinin ardından başarı yüzdeleri ortalamada %94.5 olmuştur.



**Şekil 4.20** batchNormalizationLayer eklenerek iyileştirilmiş derin öğrenme ağına ait tipik eğitim süreci

Frekans tabanlı özellik çıkarımına dayanan YSA yöntemlerin tipik karmaşıklık matrisi Şekil 4.20de verilmiştir. İstatistiksel tabanlı özellik çıkarımına dayanan YSA yöntemlerin tipik karmaşıklık matrisi ise Şekil 4.21de verilmiştir. Şekil 4.22'de ise derin öğrenme tabanlı sınıflandırma yönteminin tipik karmaşıklık matrisi verilmiştir. Karmaşıklık matrisleri karşılaştırıldığında ve bir önceki paragrafta göz önüne alındığında derin öğrenme tabanlı sınıflandırma yönteminin daha başarılı ve tutarlı olduğu görülmektedir.

Yöntemlere ait sonuçların kesinlik (precision), duyarlılık (recall), özgüllük (specificity) ve F-Skoru (F-Measure) değerleri karşılaştırma amaçlı olarak Çizelge 4.9'da verilmiştir. Bu parametrelere göre derin öğrenme tabanlı sınıflandırma yöntemi diğer yöntemlere göre 4 ölçüt kategorisinin 3'ünde üstünlük göstermektedir.

|              |   |            |            |                |
|--------------|---|------------|------------|----------------|
| Çıkış Sınıfı | H | 3<br>%21.4 | 2<br>%14.3 | %60<br>%40     |
|              | S | 1<br>%7.14 | 8<br>%57.1 | %88.9<br>%11.1 |
|              |   | %75<br>%25 | %20<br>%80 | %78.6<br>%21.4 |
|              |   | H          | S          | Hedef Sınıf    |

Şekil 4.20 Frekans tabanlı özellik çıkarımına dayanan yöntemlerin tipik karmaşıklık matrisi

|              |   |                |              |                |
|--------------|---|----------------|--------------|----------------|
| Çıkış Sınıfı | H | 20<br>%26.3    | 0<br>%0.0    | %100<br>%0.0   |
|              | S | 8<br>%10.5     | 48<br>%63.2  | %85.7<br>%14.3 |
|              |   | %71.4<br>%28.6 | %100<br>%0.0 | %89.5<br>%10.5 |
|              |   | H              | S            | Hedef Sınıf    |

Şekil 4.21 İstatistiksel tabanlı özellik çıkarımına dayanan yönteminin tipik karmaşıklık matrisi

|              |   |              |               |                |
|--------------|---|--------------|---------------|----------------|
| Çıkış Sınıfı | H | 19<br>%26    | 4<br>%5.5     | %82.6<br>%16.7 |
|              | S | 0<br>%0.0    | 50<br>%68     | %100<br>%0.0   |
|              |   | %100<br>%0.0 | %92.6<br>%7.4 | %94.5<br>%5.5  |
|              |   | H            | S             | Hedef Sınıf    |

Şekil 4.22 Derin öğrenme tabanlı yöntemin karmaşıklık matrisi



**Çizelge 4.9** Karşılaştırma amaçlı parametre çizelgesi

| Yöntem                                 | Kesinlik | Duyarlılık | Özgüllük | F-Skoru |
|--|----------|------------|----------|---------|
| Frekans tabanlı özellik çıkarımı       | 0.75     | 0.60       | 0.89     | 0.67    |
| İstatistiksel tabanlı özellik çıkarımı | 0.71     | 1.00       | 0.86     | 0.83    |
| Derin öğrenme tabanlı yöntem           | 1.00     | 0.83       | 1.00     | 0.90    |

## 5. SONUÇLAR VE ÖNERİLER

### 5.1 Sonuçlar

Bu tez çalışmasında TME bozukluklarını tespit etmek amacıyla sinyal işleme ve yapay zekâ tekniklerine dayalı bir teşhis yöntemi geliştirilmiştir. Deneklere ait TME ses verileri toplanmış, veriler işlenerek yapay zekâ algoritmaları vasıtasıyla sınıflandırılmış ve deneklerin hasta veya sağlıklı olduğu belirlenmiştir.

Veri toplama amacıyla hastayı minimum derecede rahatsız eden müdahalesiz (non-invasive), taşınması ve kurulumu kolay, ara yüzü basit ve temporamandibular eklem rahatsızlıklarıyla ilgilenen bir diş hekiminin kolaylıkla kurup kullanabileceği, taşınabilir bir cihaz imal edilmiştir.

Tasarlanan cihaz sayesinde veriler toplandıktan sonra veri işleme sürecine geçilmiştir. Kullanılan verilerin çok olması geliştirilen sınıflandırma algoritmalarının daha iyi çalışacağı anlamına gelmektedir. Daha iyi sonuçlar her zaman daha fazla veri olduğu durumlarda ortaya çıkmaktadır. Veri sayısındaki sınırlılıktan dolayı hastalık tiplerine sınıflandırma yapmak mümkün olmadığından doğrudan sağlıklı ve hasta birey ayrıştırılmasına gidilmiştir. Veri sayısının az olduğu durumlarda mümkün olduğu kadar veri sayısının çeşitli yöntemlerle artırılması sağlanmıştır.

İlk uygulanan yöntem frekans tabanlı bir sınıflandırmadır. Bu yöntemde saniyede 51200 defa örneklenmiş ve toplamda 256000 adet örnekten oluşan zaman dizisi (5sn kaydedilmiş ses verisi) elde edilmiştir. Bu aşamadan sonra analiz ve sınıflandırma kısmına geçilmiştir. Örnek sayıları kullanılarak önce veriler zaman bölgesinde YSA'ya tabi tutulmuş, veri sayıları düşürülerek denemeler devam etmiştir. Verilerin zaman bölgesi YSA sınıflandırmalarının başarısı oldukça düşük olduğundan örneklere Fourier dönüşümü uygulanarak frekans bölgesine geçilmiştir. Bu işlem yapılırken 50000 örneklilik serilerden başlanmış ve DFT uygulanarak frekans bölgesine geçilmiştir. Frekans bölgesi verileri tekrar örnek azaltma işlemine tâbi tutulup en son aşama için 100 örneklilik bir veri dizisine kadar indirilmiştir. Bu veri dizisine YSA uygulanmış ve sonuçlar raporlanmıştır.

Bu veriler 50Hz çentik filtresi ve 5KHz AGF ile filtrelenmiştir. Daha sonra 256000 örneğe sahip dönüşüm verisinin 5KHz üstü bölümü atılmış yani örnekleme frekansı sayısal olarak 10000'e çekilerek düşürülmüştür. Toplamda 50000 örneklilik bir

veri elde edilmişti. Bir sonraki aşamada FFT kullanılarak frekans dönüşümü yapılmış ve bu veriler tekrar örnek azaltma yöntemi ile her bir bireye ait veri 100 örneğe düşürülmüştür. Bu işlem aslen frekans bölgesinde özellik çıkarma işlemidir. Seçilmiş 12 sağlıklı bireye ait ve 7 hasta bireye ait verilere YSA uygulanarak %78 civarında bir sınıflandırma başarısı elde edilmiştir. Başarı yüzdesinin düşük olmasından dolayı bu yöntem kullanılabilir bir yöntem olarak görülmemiştir.

Bir sonraki aşamada özellik sayısının düşürülmesi amaçlanmış ve zaman serisi şeklinde olan verilerin istatistiksel özellikleri kullanılarak bir sınıflandırma yapılmıştır. Bu amaçla önceki seçilen sağlıklı ve hasta bireylere ait olan veri sayıları artırılarak kullanılmış ve 28 hasta birey ve 48 sağlıklı birey verisi elde edilmiştir. Her bir veri için, verilerin ortalaması, standart sapması, varyansı ve maksimum ve minimum değerleri olacak şekilde 5 adet özellik çıkarılmış ve bu özellikler kullanılarak bir YSA tasarlanmış ve eğitimi yapılmıştır. Süreç yüzlerce kez tekrar edilmiş ve ortalama %87 ile %89 arasında bir sınıflandırma başarısı elde edilmiştir.

En son aşamada ise derin öğrenme yöntemleri kullanılmıştır. Burada spektrogram tabanlı sınıflandırma tercih edilmiştir. Spektrogramlar zaman serilerinin frekans ve zaman bilgisini içerecek şekilde resme çevrilmiş halidir. Veriler tekrar gözden geçirilerek hasta bireylere ait veri sayısı 73'e ve sağlıklı bireylere ait veri sayısı 168'e çıkarılmış ve derin öğrenme ağına uygulanmıştır. Eğilimlenme nedeniyle istenen başarı sürekliliği tam olarak yakalanamadığı için hasta ve sağlıklı bireylere ait veri sayılarını birbirine yaklaştırmak amacıyla bir kısım sağlıklı birey verisi elenerek sayı 100'e düşürülmüştür. Derin öğrenme ağı tekrar tasarlanarak ortalama başarı oranı %90'ların üzerine çıkarılarak tutarlı sonuç elde edilmiştir. Böylece oldukça yüksek bir başarı oranı yakalanmıştır. Burada başarı oranını etkileyen faktörlerden birinin de Spektrogram verilerinin hem zaman hem de frekans bilgisini içerdiğini not etmek gerekmektedir.

Denenen üç yöntemden derin öğrenme ağları ile sınıflandırmanın diğerlerine oranla daha tutarlı ve başarılı olduğu görülmüştür. Sonuç olarak cihaz ve yazılım tasarımı, veri toplama, verilerin işlenmesi ve sonuçların yorumlanması ile ilgili belirlenen amaçların birçoğu başarı ile gerçekleştirilmiştir. Hekim gözetiminde toplanan verilerin sinyal işleme süreçleriyle işlenmesi ve yapay zekâ teknikleri vasıtasıyla sınıflandırılma işlemi yapılmıştır. Sınıflandırma sonucunda ise muayene edilen bireyin hasta veya sağlıklı olduğu şeklinde bir sonuç ortaya konulmuştur.

## 5.2 Öneriler

Sınıflandırma başarı oranının artırılması, kararlılığın ve tutarlılığın yükseltilmesi ileride bu konuda yapılacak araştırmanın temel amaçlarından olmalıdır. Başarı oranının yüksekliği büyük oranda eldeki mevcut verilere, tasarlanan yapının ve kullanılan yöntemlerin kabiliyetlerine dayanmaktadır. Bunlar göz önüne alınarak gelecekte yapılacak araştırmalar için aşağıdaki öneriler sıralanabilir.

Araştırma süreci ve sonuçlardan da anlaşılacağı üzere veri sayısı oldukça önemli bir başarı ve kararlılık gereksinimidir. Bu açıdan veri sayısının artırılması önem taşımaktadır. Burada tavsiye edilebilecek uygulama veri sayısının olabildiğince artırılmasıdır. Veri sayısının sınıf başına en az 100 civarı olması tavsiye edilir. Veri sayısının yeterli olmadığı durumlarda aynı hastaya ait birden fazla kayıt alma, verilerin bazı yöntemlerle çoğaltılması gibi süreçler dikkate alınabilir. Eğilimlemenin önüne geçmek içinde her sınıfa ait veri sayısının olabildiğince birbirine yakın olması tercih edilmelidir.

Verilerin ilk sınıflandırılma sürecinde aynı derecede başarıyı etkilemektedir. Veri kayıtları sırasında alınan verilerin hangi sınıflandırma gruplarına ait olduğu mümkünse hekimin teşhisinin yanı sıra en azından araştırma süreci boyunca MR, röntgen ve benzeri resim, bilgi, rapor gibi ekstra süreçlerle desteklenmelidir. Bu veri kesinliğini arttıracak ve yoruma ait bazı durumlarda hatalı teşhis oranını da düşürecektir.

Bir başka öneri özellikle tasarlanan derin öğrenme ağlarının yapısı ve parametreleri ile ilgilidir. Açıktır ki derin öğrenme ağı ne kadar büyük ve yaygın o kadar başarılı olmaktadır. Tabi ki burada ağın yaygınlığı ve genişliği hesaplama yükü olarak ortaya çıkmaktadır. Parametrelerin düzgün seçilmesi, ağ yaygınlığının ve büyüklüğünün makul sınırlar içerisinde tutulması gerekmektedir. Ağ parametrelerinin seçimi maalesef bazen yalnızca deneme yanılma yöntemi ile yapılabilmektedir. Bu özellikle başarı oranının artırılması sürecinde parametrelerin ince ayarını gerektirmektedir. Ağın yaygınlığı ve büyüklüğü de makul ölçülerde tutulmalıdır. Ağın büyüklüğünün artırılması, başarı oranında aynı miktarda bir iyileşme anlamına gelmemektedir. Yine burada ağın tasarımı ve büyüklüğü deneme yanılma yöntemi kullanılarak makul bir seviyede tutulmalıdır.

Sınıflandırma başarısını ve sayısını arttırmak için bu araştırmada kullanılmayan ama literatürde yer almış diğer ses sınıflandırma yöntemleri kullanılabilir. TME sinyalleri durağan olmayan sinyaller olmasından dolayı bu sinyallerin özellik tespiti ve

sınıflandırılması hayli problemlidir olmakta ve yeni teknikler gerektirmektedir. Her ne kadar derin öğrenmenin burada sayılan problemlerin bir kısmının üstesinden geldiği açıksa da evrimsel dönüşüm yöntemi (Evolutionary Transform), Dalgacık Dönüşümü (Wavelet Transform), Saklı Markov Modelleri (Hidden Markov Models-HMM), bulanık HMM, Destek Vektör Makineleri (Support Vector Machines, SVM), SVM-HMM, ayrık evrimsel dönüşüm (Discrete Evolutionary Transform), bulanık mantık gibi sinyal işleme ve yapay zekâ teknikleri başarı oranını yükseltmek için kullanılabilir.



## KAYNAKLAR

- Alpaydm E., 2015, Introduction To Machine Learning 3rd Edition, *The MIT Press*, Cambridge, Massachusetts
- Altunkaya S., Kara S., Görmüş N., Herdem S., 2010a, Aort ve Mitral Kapak Replasmanı Sonrası Mekanik Kalp Kapak Seslerinin Spektral Özelliklerinin Analizi, *2010 15th National Biomedical Engineering Meeting*, Antalya, 2010, pp. 1-4.
- Altunkaya S., Kara S., Gormus, N., Herdem S.. 2010b. Statistically Evaluation of Mechanical Heart Valve Thrombosis Using Heart Sounds. *Proceedings of the World Congress on Engineering 2010 Vol I WCE 2010*, June 30 - July 2, 2010, London, U.K.
- Akan A., Ergin A., Yildirim M., Öztaş, E. 2006, Analysis of Temporomandibular joint sounds in orthodontic patients. *Computers and Electrical Engineering* 32, 312-321, 2006.
- Akan A., Başar Ünsal R. ,2000, Time-frequency analysis and classification of Temporomandibular sounds. *Journal of The Franklin Institute*, 337 (2000) 437-451.
- Anonim 2019, Beyin ve bilgisayar ilişkisi, <http://bagem.tripod.com/bagem/yz2.html> [Ziyaret Tarihi: 5 Ağustos 2019].
- Beale M. H., Hagan M. T., Demuth H. B., 2018, MATLAB® Deep Learning Toolbox™ User Guide, Eylül 2018, Yalnızca Çevrimiçi, MATLAB® version 2018b, sürüm 12.0 için revize edilmiş.
- Dacombe J., 2017, An introduction to Artificial Neural Networks (with example), <https://medium.com/@jamesdacombe/an-introduction-to-artificial-neural-networks-with-example-ad459bb6941b> [Ziyaret Tarihi: 5 Ağustos 2019].
- Dagar S. R. S., Turakiya V, Pakhan A. J., Jaggi N., Kalra A., Vaidya V., 2014, Modified stethoscope for auscultation of temporomandibular joint sounds, *Journal of International Oral Health* 2014; 6(2):40-44
- Davy M., Doucet A. & Rayner P. J. W., 2002, Optimized Support Vector Machines for non-stationary signal classification. *IEEE Signal Processing Letters*, vol. 9, no. 12, Aralık 2002.
- Djurdjanovic D., Widmalm S. E., William J. W., Christopher K. H. Koh, and Kok Pin Yang, 2000. Computerized Classification of Temporomandibular Joint Sounds, *IEEE Transactions on Biomedical Engineering*. Vol. 47, No. 8, August 2000.
- Ghods M., Hassani H., Sanei S., Hicks Y., 2008, The use of Noise Information for detection of Temporomandibular disorder. *Biomed. Signal Process. and Control* (2008), doi:10.1016/j.bspc.2008.10.001. (Article in Press )

- Graves A., Mohamed A., Hinton G., 2013, Speech recognition with deep recurrent neural networks, *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, 2013, pp. 6645-6649. doi: 10.1109/ICASSP.2013.6638947
- Honda, K., Natsumi Y. & Urade M., 2008, Correlation between MRI evidence of degenerative condylar surface changes, induction of articular disc displacement and pathological joint sound in Temporomandibular joint. *Gerodontology* 2008; 25; 251-257, 2008.
- İnik Ö. ve Ülker E., 2017, Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri, *GaziOsmanPaşa Bilimsel Araştırmalar Dergisi, GaziOsmanPaşa Journal of Scientific Research*. ISSN 2146-81 68, Cilt:6 Sayı: 3, 2017, sf. 85-104
- Kim, K. S., Seo J. H., Kang J. U. & Song C. G., 2009, An enhanced algorithm for knee joint sound classification using feature extraction based on time-frequency analyses. *Computer Methods and Programs in Biomedicine* 94, 198-206, 2009.
- McLoughlin I., Zhang H., Xie Z., Song Y. and Xiao W., 2015, Robust Sound Event Classification Using Deep Neural Networks, *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 540-552, March 2015. doi: 10.1109/TASLP.2015.2389618
- Mc Neil C., 1993, The American Academy of Orofacial Pain, Edited by Charles McNeil, Temporomandibular Disorders, Guidelines for Classification, Assessment and Management. *Quintessence Books*, Illinois 1993.
- Niedzwiecki M., 2000, Identification of Time Varying Processes, *John Wiley & Sons*, Londra 2000
- Okeson J. P., 1985, Fundamentals of Occlusion and Temporomandibular Disorders; critical reviewers, James G. Burch, John T. Kemper, Jr., Williams S. Seilby ; Alison Lucas, illustrator; St. Louis : Mosby, 1985.
- Özan F., Polat S., Kara İ., Küçük D., Polat H. B., 2007, Prevalence Study of Signs and Symptoms of Temporomandibular Disorders in a Turkish Population, *The Journal of Contemporary Dental Practice*, vol: 8, no: 4, May 2007
- Prabhavathi C. N. ve Ravikumar K. M., 2014, Acoustic Noise Classification and Characterization Using Statistical Properties, *International Journal of Emerging Technology and Advanced Engineering*, Volume 4, Issue 6, June 2014
- Salamon J. ve Bello J. P., 2015, Feature learning with deep scattering for urban sound analysis, *2015 23rd European Signal Processing Conference (EUSIPCO)*, Nice, 2015, pp. 724-728. doi: 10.1109/EUSIPCO.2015.7362478
- Salamon J. ve Bello J. P., 2017, Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification, *IEEE Signal Processing*

*Letters*, vol. 24, no. 3, pp. 279-283, March 2017. doi: 10.1109/LSP.2017.2657381

- Sandberg F., Stridth M., Sörnmo L., 2008, Frequency tracking of atrial fibrillation using hidden Markov models. *IEEE Transactions on Medical Engineering*. Vol. 55, no. 2, Şubat 2008.
- Sano T., Widmalm S. E., Westesson P. L., Takahashi K., Yoshida H., Michi K., & Okanao T., 1999, Amplitude and frequency spectrum of Temporomandibular joint sounds from subjects with and without other signs/symptoms of Temporomandibular disorders. *Journal of Oral rehabilitation* 26; 145-150, 1999.
- Santana-Mora U., Cudeiro J., Mora-Bermudez M. J., Rilo-Pausa B., Ferreira-Pinho J. C., Otero-Cepeda J. L., Santana-Penin U., 2008, Changes in EMG activity during clenching in chronic pain patients with unilateral Temporomandibular disorders. *Journal of Electromyography and Kinesiology*, (Article in Press) 2008.
- Smales M., 2018, Classifying Urban sounds using Deep Learning, <https://medium.com/@mikesmales/sound-classification-using-deep-learning-8bc2aa1990b7> [Ziyaret Tarihi: 30 Temmuz 2019], yazarın , *Classifying Urban sounds using Deep Learning, UDACITY Machine Learning Nanodegree 2018 Capstone Project. December 12, 2018* adlı proje raporunun çevrimiçi özeti
- Taşkıran U., Herdem S., Çunkaş M., Aykent F., Savaş T. Y., Koçer H. E., 2015, A Sample Reduction and ANN Based Classification of Temporomandibular Joint Sounds *International Research Journal of Electronics and Computer Engineering*, Vol 1 No 3 (2015). (A pressed version of paper presented at *Third International Conference on Science, Technology, Engineering and Management (3rd ICSTEM 2015)*)
- Taşkıran U., Taşkıran S. F., Çunkaş M., 2018, Statistical Feature Extraction and ANN Based Classification of Temporomandibular Joint Sounds. *International Conference on Engineering Technologies (ICENTE'18)* Oct, 26-28 2018 Konya, TURKEY
- Took C. C., Sanei S., Chambers J., Dunne S., 2006, Underdetermined blind source separation of Temporomandibular joint sounds. *IEEE Transactions on Biomedical Engineering* vol. 53, No. 10 Ekim 2006
- Took C. C., Sanei S., Rickard S., Chambers J., Dunne S, 2008. Fractional delay estimation for blind source separation and localization of Temporomandibular joint sounds. *IEEE Transactions on Biomedical Engineering*. Vol. 55No. 3, Mart 2008.
- Widmalm S. E., Williams W. J., Adams B. S., 1996a, The waveforms of Temporomandibular joint sound clicking and crepitation. *Journal of Oral rehabilitation* 23; 44-49, 1996.



- Widmalm S. E., Williams W. J., Christiansen R. L., Gunn S. M., Park D. K., 1996b, Classification of Temporomandibular joint sounds based upon their reduced interference distribution. *Journal of Oral rehabilitation* 23; 35-43, 1996.
- Wold E., Blum T., Keislar D., Wheaton J., 1996 , Content-based classification, search, and retrieval of audio, *IEEE MultiMedia* (Volume: 3 , Issue: 3 , Fall 1996 ), pages: 27 - 36
- Yan Y., Chen M., Shyu M., Chen S., 2015, Deep Learning for Imbalanced Multimedia Data Classification, *2015 IEEE International Symposium on Multimedia (ISM)*, Miami, FL, 2015, pp. 483-488.
- Zheng C., Widmalm S. E., Williams W. J., 1999, New Time-Frequency Analyses of EMG and TMJ sound signals. *IEEE Engineering in Medicine & Biology Society 11th Annual International Conference*. 1999.



**EKLER****EK-1 Hasta İzlemede Kullanılan Form****HASTA FORMU****Hasta No:****Adı ve Soyadı:****Cinsiyet/ Yaş:****Tel:****Hastanın Şikayeti: .****Klinik Muayene:****Bulgular:****SAĞ TME****SOL TME****Teşhis:****Tedavi:****Sinyal Kayıt Tarihi:****Kontrol:**

## EK-2-A Geliştirilen Ses Kayıt Yazılımın Kodları

### program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;

namespace WindowsFormsApplication1
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new TME_Demo_Form());
        }
    }
}
```

## EK-2-B Geliştirilen Ses Kayıt Yazılımın Kodları

### TME DSP Analizi.cs

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Media;
using MathWorks.MATLAB.NET.Utility;
using MathWorks.MATLAB.NET.Arrays;
using TME_DSP_Analiz;
using System.IO;

namespace WindowsFormsApplication1
{
    public partial class TME_Demo_Form : Form
    {
        public TME_Demo_Form()
        {
            InitializeComponent();
        }

        private void Form1_Load(object sender, EventArgs e)
        {
        }

        private void button1_Click(object sender, EventArgs e)
        {
            SoundPlayer simpleSound = new SoundPlayer(@"D:\System
Files\Ugur\Documents\MATLAB\dene2.wav");
            simpleSound.Play();

            /*
            //To return a Stream object associated with an embedded
            //resource, you must prepend the namespace to the original
            //name of the file in the project.
            private void btnEmbedded_Click(object sender, System.EventArgs e) {
                Sound sound = new Sound
(Assembly.GetExecutingAssembly().GetManifestResourceStream("SoundSample.chimes.wav"));
                sound.Play();
            }

            private void btnFile_Click(object sender, System.EventArgs e) {
                Sound sound = new Sound ("Program Files\SoundSample\chord.wav");
                sound.Play();
            }
            */
            //Sound sound = new Sound("D:\\System
Files\\Ugur\\Documents\\MATLAB\\dene2.wav");
            //sound.Play();
        }

        private void button1_Click_1(object sender, EventArgs e)
        {
            // labell_Information.Text = "Sıfırlanıyor";
            TME_DSP_Analiz_Class initializeclass = new TME_DSP_Analiz_Class();
            MWNumericArray error_type = null;
            error_type = (MWNumericArray)initializeclass.seskayit((MWNumericArray)0,
(MWNumericArray)5, (MWNumericArray)44100);
            MWNumericArray item1 = (MWNumericArray)error_type[1];
            labell_Information.Text = "Sıfırlandı. Hata Kodu : " + item1;
        }

        private void button_Record_Sound_Click(object sender, EventArgs e)
        {
            // labell_Information.Text = "Kaydediyor";
            TME_DSP_Analiz_Class initializeclass = new TME_DSP_Analiz_Class();
            MWNumericArray error_type = null;

```

```

        error_type = (MWNumericArray)initializeclass.seskayit((MWNumericArray)1,
(MWNumericArray)5, (MWNumericArray)44100);
        MWNumericArray item1 = (MWNumericArray)error_type[1];
        labell_Information.Text = "Kaydediyor. Hata Kodu : " + item1;
    }

    private void button_Wav_Save_Click(object sender, EventArgs e)
    {
        TME_DSP_Analiz_Class initializeclass = new TME_DSP_Analiz_Class();
        MWNumericArray error_type = null;
        error_type = (MWNumericArray)initializeclass.seskayit((MWNumericArray)2,
(MWNumericArray)5, (MWNumericArray)44100);
        MWNumericArray item1 = (MWNumericArray)error_type[1];
        labell_Information.Text = "Wave Dosyası Kaydediyor. Hata Kodu : " + item1;
    }

    private void button_Plot_Time_Click(object sender, EventArgs e)
    {
        TME_DSP_Analiz_Class initializeclass = new TME_DSP_Analiz_Class();
        MWNumericArray error_type = null;
        error_type = (MWNumericArray)initializeclass.seskayit((MWNumericArray)3,
(MWNumericArray)5, (MWNumericArray)44100);
        MWNumericArray item1 = (MWNumericArray)error_type[1];
        labell_Information.Text = "Zaman Domeni Grafiği. Hata Kodu : " + item1;
    }

    private void button_Plot_FFT_Click(object sender, EventArgs e)
    {
        TME_DSP_Analiz_Class initializeclass = new TME_DSP_Analiz_Class();
        MWNumericArray error_type = null;
        error_type = (MWNumericArray)initializeclass.seskayit((MWNumericArray)4,
(MWNumericArray)5, (MWNumericArray)44100);
        MWNumericArray item1 = (MWNumericArray)error_type[1];
        labell_Information.Text = "Frekans Domeni Grafiği. Hata Kodu : " + item1;
    }

    private void button_Spectrogram_Click(object sender, EventArgs e)
    {
        TME_DSP_Analiz_Class initializeclass = new TME_DSP_Analiz_Class();
        MWNumericArray error_type = null;
        error_type = (MWNumericArray)initializeclass.seskayit((MWNumericArray)5,
(MWNumericArray)5, (MWNumericArray)44100);
        MWNumericArray item1 = (MWNumericArray)error_type[1];
        labell_Information.Text = "Zaman Frekans Analizi (Hamming Short Term FFT).
Hata Kodu : " + item1;
    }

    private void button_Listen_in_MATLAB_Click(object sender, EventArgs e)
    {
        TME_DSP_Analiz_Class initializeclass = new TME_DSP_Analiz_Class();
        MWNumericArray error_type = null;
        error_type = (MWNumericArray)initializeclass.seskayit((MWNumericArray)6,
(MWNumericArray)5, (MWNumericArray)44100);
        MWNumericArray item1 = (MWNumericArray)error_type[1];
        labell_Information.Text = "MATLAB Kaydı. Hata Kodu : " + item1;
    }

    private void button_PWD_Click(object sender, EventArgs e)
    {
        // EventArgs e problematic try'da hata veriyor
        try
        {
            // Get the current directory.
            string path = Directory.GetCurrentDirectory();
            label_PWD.Text = path;
        }
        catch (Exception ex)
        {
            label_PWD.Text = "The process failed: {0}" + ex.ToString();
        }
    }
}
}
}

```

## EK-2-C Geliştirilen Ses Kayıt Yazılımın Kodları

### TME DSP Analizi.Designer.cs

```

namespace WindowsFormsApplication1
{
    partial class TME_Demo_Form
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed;
        otherwise, false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            System.ComponentModel.ComponentResourceManager resources = new
            System.ComponentModel.ComponentResourceManager(typeof(TME_Demo_Form));
            this.Play_Button = new System.Windows.Forms.Button();
            this.button_MATLAB_Initialize = new System.Windows.Forms.Button();
            this.groupBox_MATLAB_1 = new System.Windows.Forms.GroupBox();
            this.button_Listen_in_MATLAB = new System.Windows.Forms.Button();
            this.button_Spectrogram = new System.Windows.Forms.Button();
            this.button_Plot_FFT = new System.Windows.Forms.Button();
            this.button_Plot_Time = new System.Windows.Forms.Button();
            this.button_Wav_Save = new System.Windows.Forms.Button();
            this.button_Record_Sound = new System.Windows.Forms.Button();
            this.pictureBox_Icon = new System.Windows.Forms.PictureBox();
            this.label1_Information = new System.Windows.Forms.Label();
            this.button_PWD = new System.Windows.Forms.Button();
            this.label_PWD = new System.Windows.Forms.Label();
            this.groupBox_MATLAB_1.SuspendLayout();

            ((System.ComponentModel.ISupportInitialize)(this.pictureBox_Icon)).BeginInit();
            this.SuspendLayout();
            //
            // Play_Button
            //
            this.Play_Button.Location = new System.Drawing.Point(49, 173);
            this.Play_Button.Name = "Play_Button";
            this.Play_Button.Size = new System.Drawing.Size(75, 23);
            this.Play_Button.TabIndex = 0;
            this.Play_Button.Text = "Play";
            this.Play_Button.UseVisualStyleBackColor = true;
            this.Play_Button.Click += new System.EventHandler(this.button1_Click);
            //
            // button_MATLAB_Initialize
            //
            this.button_MATLAB_Initialize.Location = new System.Drawing.Point(30, 19);
            this.button_MATLAB_Initialize.Name = "button_MATLAB_Initialize";
            this.button_MATLAB_Initialize.Size = new System.Drawing.Size(75, 50);
            this.button_MATLAB_Initialize.TabIndex = 1;
            this.button_MATLAB_Initialize.Text = "Sıfırla";
            this.button_MATLAB_Initialize.UseVisualStyleBackColor = true;
            this.button_MATLAB_Initialize.Click += new
            System.EventHandler(this.button1_Click_1);
        }
    }
}

```

```

//
// groupBox_MATLAB_1
//
this.groupBox_MATLAB_1.Controls.Add(this.button_Listen_in_MATLAB);
this.groupBox_MATLAB_1.Controls.Add(this.button_Spectrogram);
this.groupBox_MATLAB_1.Controls.Add(this.button_Plot_FFT);
this.groupBox_MATLAB_1.Controls.Add(this.button_Plot_Time);
this.groupBox_MATLAB_1.Controls.Add(this.button_Wav_Save);
this.groupBox_MATLAB_1.Controls.Add(this.button_Record_Sound);
this.groupBox_MATLAB_1.Controls.Add(this.button_MATLAB_Initialize);
this.groupBox_MATLAB_1.Location = new System.Drawing.Point(200, 60);
this.groupBox_MATLAB_1.Name = "groupBox_MATLAB_1";
this.groupBox_MATLAB_1.Size = new System.Drawing.Size(350, 200);
this.groupBox_MATLAB_1.TabIndex = 2;
this.groupBox_MATLAB_1.TabStop = false;
this.groupBox_MATLAB_1.Text = "MATLAB Demo";
//
// button_Listen_in_MATLAB
//
this.button_Listen_in_MATLAB.Location = new System.Drawing.Point(138, 131);
this.button_Listen_in_MATLAB.Name = "button_Listen_in_MATLAB";
this.button_Listen_in_MATLAB.Size = new System.Drawing.Size(75, 50);
this.button_Listen_in_MATLAB.TabIndex = 7;
this.button_Listen_in_MATLAB.Text = "MATLAB Kaydı Dinle";
this.button_Listen_in_MATLAB.UseVisualStyleBackColor = true;
this.button_Listen_in_MATLAB.Click += new
System.EventHandler(this.button_Listen_in_MATLAB_Click);
//
// button_Spectrogram
//
this.button_Spectrogram.Location = new System.Drawing.Point(248, 76);
this.button_Spectrogram.Name = "button_Spectrogram";
this.button_Spectrogram.Size = new System.Drawing.Size(75, 50);
this.button_Spectrogram.TabIndex = 6;
this.button_Spectrogram.Text = "Zaman Frekans Analizi";
this.button_Spectrogram.UseVisualStyleBackColor = true;
this.button_Spectrogram.Click += new
System.EventHandler(this.button_Spectrogram_Click);
//
// button_Plot_FFT
//
this.button_Plot_FFT.Location = new System.Drawing.Point(138, 76);
this.button_Plot_FFT.Name = "button_Plot_FFT";
this.button_Plot_FFT.Size = new System.Drawing.Size(75, 50);
this.button_Plot_FFT.TabIndex = 5;
this.button_Plot_FFT.Text = "FFT";
this.button_Plot_FFT.UseVisualStyleBackColor = true;
this.button_Plot_FFT.Click += new
System.EventHandler(this.button_Plot_FFT_Click);
//
// button_Plot_Time
//
this.button_Plot_Time.Location = new System.Drawing.Point(30, 75);
this.button_Plot_Time.Name = "button_Plot_Time";
this.button_Plot_Time.Size = new System.Drawing.Size(75, 50);
this.button_Plot_Time.TabIndex = 4;
this.button_Plot_Time.Text = "Sinyali Çiz";
this.button_Plot_Time.UseVisualStyleBackColor = true;
this.button_Plot_Time.Click += new
System.EventHandler(this.button_Plot_Time_Click);
//
// button_Wav_Save
//
this.button_Wav_Save.Location = new System.Drawing.Point(248, 19);
this.button_Wav_Save.Name = "button_Wav_Save";
this.button_Wav_Save.Size = new System.Drawing.Size(75, 50);
this.button_Wav_Save.TabIndex = 3;
this.button_Wav_Save.Text = "Wave Kaydet";
this.button_Wav_Save.UseVisualStyleBackColor = true;
this.button_Wav_Save.Click += new
System.EventHandler(this.button_Wav_Save_Click);
//
// button_Record_Sound
//
this.button_Record_Sound.Location = new System.Drawing.Point(138, 19);
this.button_Record_Sound.Name = "button_Record_Sound";
this.button_Record_Sound.Size = new System.Drawing.Size(75, 50);

```

```

        this.button_Record_Sound.TabIndex = 2;
        this.button_Record_Sound.Text = "Kaydet";
        this.button_Record_Sound.UseVisualStyleBackColor = true;
        this.button_Record_Sound.Click += new
System.EventHandler(this.button_Record_Sound_Click);
        //
        // pictureBox_Icon
        //
        this.pictureBox_Icon.Image =
((System.Drawing.Image)(resources.GetObject("pictureBox_Icon.Image")));
        this.pictureBox_Icon.Location = new System.Drawing.Point(34, 21);
        this.pictureBox_Icon.Name = "pictureBox_Icon";
        this.pictureBox_Icon.Size = new System.Drawing.Size(128, 128);
        this.pictureBox_Icon.TabIndex = 3;
        this.pictureBox_Icon.TabStop = false;
        //
        // labell_Information
        //
        this.labell_Information.AutoSize = true;
        this.labell_Information.Location = new System.Drawing.Point(200, 15);
        this.labell_Information.MinimumSize = new System.Drawing.Size(350, 40);
        this.labell_Information.Name = "labell_Information";
        this.labell_Information.Size = new System.Drawing.Size(350, 40);
        this.labell_Information.TabIndex = 4;
        this.labell_Information.Text = "TME Ses Analizi İçin Demo Programı";
        //
        // button_PWD
        //
        this.button_PWD.Location = new System.Drawing.Point(49, 202);
        this.button_PWD.Name = "button_PWD";
        this.button_PWD.Size = new System.Drawing.Size(75, 30);
        this.button_PWD.TabIndex = 5;
        this.button_PWD.Text = "Klasör";
        this.button_PWD.UseVisualStyleBackColor = true;
        this.button_PWD.Click += new System.EventHandler(this.button_PWD_Click);
        //
        // label_PWD
        //
        this.label_PWD.AutoSize = true;
        this.label_PWD.Location = new System.Drawing.Point(5, 235);
        this.label_PWD.MaximumSize = new System.Drawing.Size(300, 20);
        this.label_PWD.MinimumSize = new System.Drawing.Size(300, 20);
        this.label_PWD.Name = "label_PWD";
        this.label_PWD.Size = new System.Drawing.Size(300, 20);
        this.label_PWD.TabIndex = 6;
        this.label_PWD.Text = "Klasör:";
        this.label_PWD.UseWaitCursor = true;
        //
        // TME_Demo_Form
        //
        this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
        this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
        this.ClientSize = new System.Drawing.Size(569, 264);
        this.Controls.Add(this.label_PWD);
        this.Controls.Add(this.button_PWD);
        this.Controls.Add(this.labell_Information);
        this.Controls.Add(this.pictureBox_Icon);
        this.Controls.Add(this.groupBox_MATLAB_1);
        this.Controls.Add(this.Play_Button);
        this.Icon = ((System.Drawing.Icon)(resources.GetObject("$this.Icon")));
        this.Name = "TME_Demo_Form";
        this.Text = "TME Ses Analiz Demo";
        this.Load += new System.EventHandler(this.Form1_Load);
        this.groupBox_MATLAB_1.ResumeLayout(false);

        ((System.ComponentModel.ISupportInitialize)(this.pictureBox_Icon)).EndInit();
        this.ResumeLayout(false);
        this.PerformLayout();

    }

#endregion

private System.Windows.Forms.Button Play_Button;
private System.Windows.Forms.Button button_MATLAB_Initialize;
private System.Windows.Forms.GroupBox groupBox_MATLAB_1;
private System.Windows.Forms.PictureBox pictureBox_Icon;

```



```
private System.Windows.Forms.Label label1_Information;  
private System.Windows.Forms.Button button_Record_Sound;  
private System.Windows.Forms.Button button_Wav_Save;  
private System.Windows.Forms.Button button_Plot_Time;  
private System.Windows.Forms.Button button_Plot_FFT;  
private System.Windows.Forms.Button button_Spectrogram;  
private System.Windows.Forms.Button button_Listen_in_MATLAB;  
private System.Windows.Forms.Button button_PWD;  
private System.Windows.Forms.Label label_PWD;  
}  
}
```



**EK-2-D****TME\_DSP\_Analiz\_Class.cs**

```

/*
 * MATLAB Compiler: 4.9 (R2008b)
 * Date: Mon Dec 28 10:33:01 2009
 * Arguments: "-B" "macro_default" "-W"
 * "dotnet:TME_DSP_Analiz,TME_DSP_Analiz_Class,0.0,private" "-d" "D:\System
 * Files\Ugur\Documents\MATLAB\TME_DSP_Analiz\src" "-T" "link:lib" "-v"
 * "class{TME_DSP_Analiz_Class:D:\System Files\Ugur\Documents\MATLAB\seskayit.m}"
 */

using System;
using System.Reflection;
using System.IO;
using MathWorks.MATLAB.NET.Arrays;
using MathWorks.MATLAB.NET.Utility;
using MathWorks.MATLAB.NET.ComponentData;
#if SHARED
[assembly: System.Reflection.AssemblyKeyFile(@"")]
#endif

namespace TME_DSP_Analiz
{
    /// <summary>
    /// The TME_DSP_Analiz_Class class provides a CLS compliant, MWArray interface to the
    /// M-functions contained in the files:
    /// <newpara></newpara>
    /// D:\System Files\Ugur\Documents\MATLAB\seskayit.m
    /// <newpara></newpara>
    /// deployprint.m
    /// <newpara></newpara>
    /// printdlg.m
    /// </summary>
    /// <remarks>
    /// @Version 0.0
    /// </remarks>
    public class TME_DSP_Analiz_Class : IDisposable
    {
        #region Constructors

        /// <summary internal= "true">
        /// The static constructor instantiates and initializes the MATLAB Component
        /// Runtime instance.
        /// </summary>
        static TME_DSP_Analiz_Class()
        {
            if (MWArray.MCRAppInitialized)
            {
                Assembly assembly= Assembly.GetExecutingAssembly();

                string ctfilePath= assembly.Location;

                int lastDelimiter= ctfilePath.LastIndexOf(@"\");

                ctfilePath= ctfilePath.Remove(lastDelimiter, (ctfilePath.Length -
lastDelimiter));

                string ctfileName = MCRComponentState.MCC_TME_DSP_Analiz_name_data +
".ctf";

                Stream embeddedCtfStream = null;

                String[] resourceStrings = assembly.GetManifestResourceNames();

                foreach (String name in resourceStrings)
                {
                    if (name.Contains(ctfileName))
                    {
                        embeddedCtfStream = assembly.GetManifestResourceStream(name);
                        break;
                    }
                }
            }
        }
    }
}

```

```

    }
    mcr= new MWMCR(MCRComponentState.MCC_TME_DSP_Analiz_name_data,
                  MCRComponentState.MCC_TME_DSP_Analiz_root_data,
                  MCRComponentState.MCC_TME_DSP_Analiz_public_data,
                  MCRComponentState.MCC_TME_DSP_Analiz_session_data,
                  MCRComponentState.MCC_TME_DSP_Analiz_matlabpath_data,
                  MCRComponentState.MCC_TME_DSP_Analiz_classpath_data,
                  MCRComponentState.MCC_TME_DSP_Analiz_libpath_data,
                  MCRComponentState.MCC_TME_DSP_Analiz_mcr_application_options,
                  MCRComponentState.MCC_TME_DSP_Analiz_mcr_runtime_options,
                  MCRComponentState.MCC_TME_DSP_Analiz_mcr_pref_dir,
                  MCRComponentState.MCC_TME_DSP_Analiz_set_warning_state,
                  ctfFilePath, embeddedCtfStream, true);
    }
    else
    {
        throw new ApplicationException("MWArray assembly could not be
initialized");
    }
}

/// <summary>
/// Constructs a new instance of the TME_DSP_Analiz_Class class.
/// </summary>
public TME_DSP_Analiz_Class()
{
}

#endregion Constructors

#region Finalize

/// <summary internal= "true">
/// Class destructor called by the CLR garbage collector.
/// </summary>
~TME_DSP_Analiz_Class()
{
    Dispose(false);
}

/// <summary>
/// Frees the native resources associated with this object
/// </summary>
public void Dispose()
{
    Dispose(true);

    GC.SuppressFinalize(this);
}

/// <summary internal= "true">
/// Internal dispose function
/// </summary>
protected virtual void Dispose(bool disposing)
{
    if (!disposed)
    {
        disposed= true;

        if (disposing)
        {
            // Free managed resources;
        }

        // Free native resources
    }
}

#endregion Finalize

#region Methods

```

```

    /// <summary>
    /// Provides a single output, 0-input mxArray interface to the seskayit M-
function.
    /// </summary>
    /// <remarks>
    /// M-Documentation:
    /// srs: Sound Record Request
    /// 0: Initilize
    /// 1: record and background saving
    /// 2: save as a wave file
    /// 3: Graph the recording
    /// 4: Show FFT of the recording
    /// 5: Show PSD of the recording
    /// 6: Listen Recording in MATLAB
    /// sure: recording lenght in seconds
    /// Fs: Sampling Frequency
    /// </remarks>
    /// <returns>An mxArray containing the first output argument.</returns>
    ///
public mxArray seskayit()
{
    return mcr.EvaluateFunction("seskayit", new mxArray[]{});
}

    /// <summary>
    /// Provides a single output, 1-input mxArray interface to the seskayit M-
function.
    /// </summary>
    /// <remarks>
    /// M-Documentation:
    /// srs: Sound Record Request
    /// 0: Initilize
    /// 1: record and background saving
    /// 2: save as a wave file
    /// 3: Graph the recording
    /// 4: Show FFT of the recording
    /// 5: Show PSD of the recording
    /// 6: Listen Recording in MATLAB
    /// sure: recording lenght in seconds
    /// Fs: Sampling Frequency
    /// </remarks>
    /// <param name="srs">Input argument #1</param>
    /// <returns>An mxArray containing the first output argument.</returns>
    ///
public mxArray seskayit(mxArray srs)
{
    return mcr.EvaluateFunction("seskayit", srs);
}

    /// <summary>
    /// Provides a single output, 2-input mxArray interface to the seskayit M-
function.
    /// </summary>
    /// <remarks>
    /// M-Documentation:
    /// srs: Sound Record Request
    /// 0: Initilize
    /// 1: record and background saving
    /// 2: save as a wave file
    /// 3: Graph the recording
    /// 4: Show FFT of the recording
    /// 5: Show PSD of the recording
    /// 6: Listen Recording in MATLAB
    /// sure: recording lenght in seconds
    /// Fs: Sampling Frequency
    /// </remarks>
    /// <param name="srs">Input argument #1</param>
    /// <param name="sure">Input argument #2</param>
    /// <returns>An mxArray containing the first output argument.</returns>
    ///
public mxArray seskayit(mxArray srs, mxArray sure)
{
    return mcr.EvaluateFunction("seskayit", srs, sure);
}

```

```

    /// <summary>
    /// Provides a single output, 3-input MWArray interface to the seskayit M-
function.
    /// </summary>
    /// <remarks>
    /// M-Documentation:
    /// srs: Sound Record Request
    /// 0: Initilize
    /// 1: record and background saving
    /// 2: save as a wave file
    /// 3: Graph the recording
    /// 4: Show FFT of the recording
    /// 5: Show PSD of the recording
    /// 6: Listen Recording in MATLAB
    /// sure: recording lenght in seconds
    /// Fs: Sampling Frequency
    /// </remarks>
    /// <param name="srs">Input argument #1</param>
    /// <param name="sure">Input argument #2</param>
    /// <param name="Fs">Input argument #3</param>
    /// <returns>An MWArray containing the first output argument.</returns>
    ///
public MWArray seskayit(MWArray srs, MWArray sure, MWArray Fs)
{
    return mcr.EvaluateFunction("seskayit", srs, sure, Fs);
}

    /// <summary>
    /// Provides a single output, 4-input MWArray interface to the seskayit M-
function.
    /// </summary>
    /// <remarks>
    /// M-Documentation:
    /// srs: Sound Record Request
    /// 0: Initilize
    /// 1: record and background saving
    /// 2: save as a wave file
    /// 3: Graph the recording
    /// 4: Show FFT of the recording
    /// 5: Show PSD of the recording
    /// 6: Listen Recording in MATLAB
    /// sure: recording lenght in seconds
    /// Fs: Sampling Frequency
    /// </remarks>
    /// <param name="srs">Input argument #1</param>
    /// <param name="sure">Input argument #2</param>
    /// <param name="Fs">Input argument #3</param>
    /// <param name="path">Input argument #4</param>
    /// <returns>An MWArray containing the first output argument.</returns>
    ///
public MWArray seskayit(MWArray srs, MWArray sure,
    MWArray Fs, MWArray path)
{
    return mcr.EvaluateFunction("seskayit", srs, sure, Fs, path);
}

    /// <summary>
    /// Provides the standard 0-input MWArray interface to the seskayit M-function.
    /// </summary>
    /// <remarks>
    /// M-Documentation:
    /// srs: Sound Record Request
    /// 0: Initilize
    /// 1: record and background saving
    /// 2: save as a wave file
    /// 3: Graph the recording
    /// 4: Show FFT of the recording
    /// 5: Show PSD of the recording
    /// 6: Listen Recording in MATLAB
    /// sure: recording lenght in seconds
    /// Fs: Sampling Frequency
    /// </remarks>
    /// <param name="numArgsOut">The number of output arguments to return.</param>

```

```

/// <returns>An Array of length "numArgsOut" containing the output
/// arguments.</returns>
///
public MWArray[] seskayit(int numArgsOut)
{
    return mcr.EvaluateFunction(numArgsOut, "seskayit", new MWArray[]{});
}

/// <summary>
/// Provides the standard 1-input MWArray interface to the seskayit M-function.
/// </summary>
/// <remarks>
/// M-Documentation:
/// srs: Sound Record Request
/// 0: Initilize
/// 1: record and background saving
/// 2: save as a wave file
/// 3: Graph the recording
/// 4: Show FFT of the recording
/// 5: Show PSD of the recording
/// 6: Listen Recording in MATLAB
/// sure: recording lenght in seconds
/// Fs: Sampling Frequency
/// </remarks>
/// <param name="numArgsOut">The number of output arguments to return.</param>
/// <param name="srs">Input argument #1</param>
/// <returns>An Array of length "numArgsOut" containing the output
/// arguments.</returns>
///
public MWArray[] seskayit(int numArgsOut, MWArray srs)
{
    return mcr.EvaluateFunction(numArgsOut, "seskayit", srs);
}

/// <summary>
/// Provides the standard 2-input MWArray interface to the seskayit M-function.
/// </summary>
/// <remarks>
/// M-Documentation:
/// srs: Sound Record Request
/// 0: Initilize
/// 1: record and background saving
/// 2: save as a wave file
/// 3: Graph the recording
/// 4: Show FFT of the recording
/// 5: Show PSD of the recording
/// 6: Listen Recording in MATLAB
/// sure: recording lenght in seconds
/// Fs: Sampling Frequency
/// </remarks>
/// <param name="numArgsOut">The number of output arguments to return.</param>
/// <param name="srs">Input argument #1</param>
/// <param name="sure">Input argument #2</param>
/// <returns>An Array of length "numArgsOut" containing the output
/// arguments.</returns>
///
public MWArray[] seskayit(int numArgsOut, MWArray srs, MWArray sure)
{
    return mcr.EvaluateFunction(numArgsOut, "seskayit", srs, sure);
}

/// <summary>
/// Provides the standard 3-input MWArray interface to the seskayit M-function.
/// </summary>
/// <remarks>
/// M-Documentation:
/// srs: Sound Record Request
/// 0: Initilize
/// 1: record and background saving
/// 2: save as a wave file
/// 3: Graph the recording
/// 4: Show FFT of the recording
/// 5: Show PSD of the recording
/// 6: Listen Recording in MATLAB

```

```

/// sure: recording lenght in seconds
/// Fs: Sampling Frequency
/// </remarks>
/// <param name="numArgsOut">The number of output arguments to return.</param>
/// <param name="srs">Input argument #1</param>
/// <param name="sure">Input argument #2</param>
/// <param name="Fs">Input argument #3</param>
/// <returns>An Array of length "numArgsOut" containing the output
/// arguments.</returns>
///
public MArray[] seskayit(int numArgsOut, MArray srs,
                        MArray sure, MArray Fs)
{
    return mcr.EvaluateFunction(numArgsOut, "seskayit", srs, sure, Fs);
}

/// <summary>
/// Provides the standard 4-input MArray interface to the seskayit M-function.
/// </summary>
/// <remarks>
/// M-Documentation:
/// srs: Sound Record Request
/// 0: Initilize
/// 1: record and background saving
/// 2: save as a wave file
/// 3: Graph the recording
/// 4: Show FFT of the recording
/// 5: Show PSD of the recording
/// 6: Listen Recording in MATLAB
/// sure: recording lenght in seconds
/// Fs: Sampling Frequency
/// </remarks>
/// <param name="numArgsOut">The number of output arguments to return.</param>
/// <param name="srs">Input argument #1</param>
/// <param name="sure">Input argument #2</param>
/// <param name="Fs">Input argument #3</param>
/// <param name="path">Input argument #4</param>
/// <returns>An Array of length "numArgsOut" containing the output
/// arguments.</returns>
///
public MArray[] seskayit(int numArgsOut, MArray srs,
                        MArray sure, MArray Fs, MArray path)
{
    return mcr.EvaluateFunction(numArgsOut, "seskayit",
                               srs, sure, Fs, path);
}

/// <summary>
/// Provides an interface for the seskayit function in which the input and output
/// arguments are specified as an array of MArrays.
/// </summary>
/// <remarks>
/// This method will allocate and return by reference the output argument
/// array.<newpara></newpara>
/// M-Documentation:
/// srs: Sound Record Request
/// 0: Initilize
/// 1: record and background saving
/// 2: save as a wave file
/// 3: Graph the recording
/// 4: Show FFT of the recording
/// 5: Show PSD of the recording
/// 6: Listen Recording in MATLAB
/// sure: recording lenght in seconds
/// Fs: Sampling Frequency
/// </remarks>
/// <param name="numArgsOut">The number of output arguments to return</param>
/// <param name="argsOut">Array of MArray output arguments</param>
/// <param name="argsIn">Array of MArray input arguments</param>
///
public void seskayit(int numArgsOut, ref MArray[] argsOut, MArray[] argsIn)
{
    mcr.EvaluateFunction("seskayit", numArgsOut, ref argsOut, argsIn);
}

```

```
/// <summary>
/// This method will cause a MATLAB figure window to behave as a modal dialog box.
/// The method will not return until all the figure windows associated with this
/// component have been closed.
/// </summary>
/// <remarks>
/// An application should only call this method when required to keep the
/// MATLAB figure window from disappearing. Other techniques, such as calling
/// Console.ReadLine() from the application should be considered where
/// possible.</remarks>
///
public void WaitForFiguresToDie()
{
    mcr.WaitForFiguresToDie();
}

#endregion Methods

#region Class Members



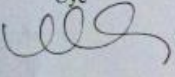
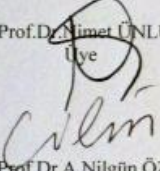
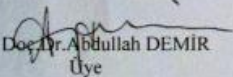
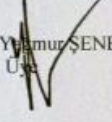
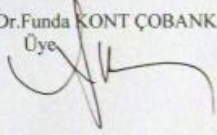
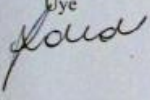
private static MWMCR mcr= null;

private bool disposed= false;

#endregion Class Members
}
}
```



## EK-3 Etik Kurul İzni

|   |   |   |
|---|---|---|
|     |   |   |
| <b>SELÇUK ÜNİVERSİTESİ</b><br><b>DIŞ HEKİMLİĞİ FAKÜLTESİ</b><br><b>GİRİŞİMSEL OLMAYAN KLİNİK ARAŞTIRMALAR</b><br><b>DEĞERLENDİRME KOMİSYONU</b>   |   |   |
| Toplantı sayısı : 2011/05   | Toplantı tarihi : 05.05.2011  |   |
| <p>Selçuk Üniversitesi Mimarlık Mühendislik Fakültesi Elektrik Elektronik Mühendisliği Bölümünden Prof.Dr.Saadetdin HERDEM, aynı bölümden Öğr.Gör.Uğur TAŞKIRAN ve Diş Hekimliği Fakültesi Protetik Diş Tedavisi Anabilim Dalından Prof.Dr.Filiz AYKENT tarafından sunulan <b>“Temporomandibular Eklem Bozukluklarının Belirlenmesinde Sinyal İşleme ve Yapay Zeka Teknikleri Kullanılması”</b> başlıklı araştırma projesi 7 üyenin katılımı ile değerlendirildi.</p> <p>Değerlendirme sonucunda, Projenin, Girişimsel Olmayan Klinik Araştırmalar Değerlendirme Yönergesi İlkelerine uygun olduğundan <b>“kabulüne”</b> oybirliği ile karar verildi.</p> |   |   |
| Prof.Dr.Faruk Ayhan BAŞÇİFTÇİ<br>katılmadı  |   |   |
| Prof.Dr.Mihtikar GÜRSÉL<br>Üye<br>   | Prof.Dr.Nimet ÜNLÜ<br>Üye<br>   | Prof.Dr.Doğan DOLANMAZ<br>katılmadı   |
| Prof.Dr.Duygu FINDIK<br>katılmadı   | Prof.Dr.A.Nilgün ÖZTÜRK<br>Üye  | Prof.Dr.Ender ERDOĞAN<br>katılmadı  |
| Doç.Dr.Abdullah DEMİR<br>Üye<br>   | Prof.Dr.Yamur ŞENER<br>Üye<br> | Doç.Dr.Funda KONT ÇOBANKARA<br>Üye<br> |
| Yrd.Doç.Dr.Füsün YAŞAR<br>Üye<br>  | Yrd.Doç.Dr.Hüsamettin VATANSEV<br>katılmadı   | Yrd.Doç.Dr.Fatih KARA<br>katılmadı  |

## ÖZGEÇMİŞ

### KİŞİSEL BİLGİLER

**Adı Soyadı** : Uğur TAŞKIRAN  
**Uyruğu** : TC  
**Doğum Yeri ve Tarihi** : Kızıltepe – 04.07.1968  
**Telefon** : +90 (536) 669 31 86  
**Faks** :  
**E-Posta** : [utaskiran@selcuk.edu.tr](mailto:utaskiran@selcuk.edu.tr)

### EĞİTİM

| Derece        | Adı   | İlçe | İl | Bitirme Yılı |
|---------------|---|------|----|--------------|
| Lise          | : Fatih Endüstri Meslek Lisesi, Selçuklu, KONYA   |      |    | 1985         |
| Üniversite    | : Hacettepe Üniversitesi Mühendislik Fakültesi  |      |    |              |
|               | : Elektrik-Elektronik Mühendisliği, Beytepe, ANKARA   |      |    | 1990         |
| Yüksek Lisans | : The Univesity of Florida, Graduate School, Department of Electrical Enginnering, Florida, ABD           |      |    | 1993         |
| Doktora       | : Konya Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Elektrik-Elektronik Müh. A.B.D. Selçuklu, KONYA |      |    | 2019         |

### İŞ DENEYİMLERİ

| Yıl       | Kurum   | Görevi                  |
|-----------|---|-------------------------|
| 1990-1991 | Hacettepe Üniversitesi Mühendislik Fakültesi Elektrik-Elektronik Mühendisliği | Araştırma Görevlisi     |
| 1991-1999 | T.C. Milli Eğitim Bakanlığı   | Yurtdışı Burslu Öğrenci |
| 2000-2001 | K.K.K. Işıklar Askeri Lisesi  | İngilizce Öğretmeni     |
| 2001-2016 | Selçuk Üniversitesi Teknik Eğitim Fakültesi                                   | Öğretim Görevlisi       |
| 2016-     | Selçuk Üniversitesi Teknoloji Fakültesi                                       | Öğretim Görevlisi       |

### UZMANLIK ALANI

Devreler, Sinyaller ve Sistemler, Sayısal Sinyal İşleme, Programlama, Haberleşme Sistemleri, İletişim Ağları, ATM Ağları, Yapay Zeka Sistemleri, Derin Öğrenme

### YABANCI DİLLER

İngilizce

## YAYINLAR

### Uluslararası SCI, SCI-Exp İndekslerinde Taranan Dergilerdeki Yayınlar

Çunkaş M., Taşkiran U. Electricity Consumption Forecasting using Genetic Programming. Energy Sources Part B, (2011).

Şahman M.A., Çunkaş M., İnal Ş. İnal F., Coşkun B., Taşkiran U., Cost Optimization of Feed Mixes by Genetic Algorithms, Advances in Engineering Software, 40(10), 965-974, (2009)

### Uluslararası Diğer İndekslerde Taranan Dergilerdeki Yayınlar

Taşkiran U., Herdem S., Çunkaş M., Aykent F., Savaş T.Y., Koçer H. E., A Sample Reduction and ANN Based Classification of Temporomandibular Joint Sounds, International Research Journal of Electronics and Computer Engineering, Vol 1 No 3 (2015). (A pressed version of paper presented at Third International Conference on Science, Technology, Engineering and Management (3rd ICSTEM 2015) (Doktora Tezinden Yapılmıştır.)

Kuş R., Taşkiran U., Bayrakçeken H., Modeling And Experimental Investigation Of Effect Of Steering Angle On Braking Force, Journal of International Research Publications, 1, 1, 12 - 21, (2006)

### Uluslararası Konferanslarda Sunulan Yayınlar

Taşkiran U., Taşkiran S. F., Çunkaş M., Statistical Feature Extraction and ANN Based Classification of Temporomandibular Joint Sounds. International Conference on Engineering Technologies (ICENTE'18) Oct, 26-28 2018 Konya, TURKEY (Doktora Tezinden Yapılmıştır.)

Taşkiran U., Herdem S., Çunkaş M., Aykent F., Savaş T.Y., Koçer H. E., A Sample Reduction and ANN Based Classification of Temporomandibular Joint Sounds, Third International Conference on Science, Technology, Engineering and Management (3rd ICSTEM 2015), pp.21-24, Dubai, BAE,01-04 Aralık 2015. (Doktora Tezinden Yapılmıştır.)

Kus R., Taskiran U., Bayrakçeken H., Neural Network Modeling of Breaking Force in Variable Road and Slope Conditions, Proceedings of 2012 Hong Kong International Conference on Engineering and Applied Science., Hong Kong, Aralık 2012

Ünal Y., Köklü M., Taşkiran U., Application of Data Mining Techniques on Backpatient Data, ISTEC 2010, International Science and Technology Conference, 27-29 Ekim 2010, KKTC:

Köklü M., Taşkiran U., Ünal Y., Application of Web Aided Education on Laboratory of Introduction to Electronics Technology Course, ICITS 2010 4th International Computer and Instructional Technologies Symposium 24-26 Eylül 2010, Konya, Türkiye

Ünal Y., Köklü M., Taşkiran U., Examining the Factors that Affect the University Student's Attendance to the Socio-Cultural Activities by Using Data Mining, ICITS 2010 4th International Computer and Instructional Technologies Symposium 24-26 Eylül 2010, Konya, Türkiye

Taşkiran U., Köklü M., Ünal Y., Web Aided Training of Measurement Devices and Circuit Elements Used in Electronics Laboratories, IETC 2010, 10th International Educational Technology Conference & Exhibition, 26-28 Nisan 2010, İstanbul, Türkiye.

Köklü M., Taşkiran U., Ünal Y., Web Adided Teaching of Circuit Analysis Laboratory Courses, IETC 2010, 10th International Educational Technology Conference & Exhibition, 26-28 Nisan 2010, İstanbul, Türkiye.

Mehmet Çunkaş ve Uğur Taşkiran, Application of Genetic Programming to Turkey's Long Term Electrical Consumption Forecast, IMS'2008, The 6th International Symposium On Intelligent & Manufacturing Systems 2008.

Asiltürk I., Taşkiran U., Ünüvar A., Applications of Intelligent Control and Monitoring at Manufacturing. Int. Conf. On Modelling and Simulation, 2006

Sarıtaş İ., Taşkiran U., Ünal Y., Bulanık Yaklaşım Metodu İle Ameliyathanelerin Isı Kontrolü, 4th International Advanced Technologies Symposium, 2005

Arslan C., Işık H., Taşkiran U., Bir Geçici Pacemaker Tasarımı ve Yapımı, 4th International Advanced Technologies Symposium, 2005

## **Projeler**

Temporomandibular Eklem Bozukluklarının Belirlenmesinde Sinyal İşleme ve Yapay Zekâ Teknikleri Kullanılması, Selçuk Üniversitesi Araştırma Fonu, 2010, *Yardımcı Araştırmacı*

Konya İlinde Mesleki ve Teknik Eğitim Kurumları Eğitimcilerinin Otomotiv Mekatroniği Konusunda Bilgi ve Becerilerinin Arttırılması Projesi, Avrupa Birliği, Proje *Teknik İşler Koordinatör Yardımcısı*, MEGEP /SVET DELTUR/2005/99512

## **Uluslararası Ödüller**

### Best Presentation:

Taşkiran U, Herdem S, Çunkaş M, Aykent F, Yılmaz Savas T, Koçer H. E., A Sample Reduction and ANN Based Classification of Temporomandibular Joint Sounds, Third International Conference on Science, Technology, Engineering and Management (3rd ICSTEM 2015), pp.21-24, Dubai, BAE,01-04 Aralık 2015.