ELM368 – DÖNEM PROJESİ
SES TANIYICI

Ebru İrem Doğan, Muhammed Baki Karagöz, Şafak Ferhat Kaya
151024021, 161024041, 1801022092
eidogan@gtu.edu.tr, muhammet.karagoz2016@gtu.edu.tr, sfkaya@gtu.edu.tr

ABSTRACT (ÖZET)

Bilindiği üzere her insanın sesi eşsiz bir yapıdadır. Bir konferanstaki konuşmacıları ayırt etmenin yanı sıra daha kompleks bir sistem olarak niteleyebileceğimiz güvenlik sistemlerinde de bu eşsiz yapıyı birbirinden ayırt etmek bir problem olarak karşımıza çıkar.

Bu çalışma ise bu problemi konu edinerek, konuşan kişiyi ayırt etmeyi amaçlar. Problemin çözümünde üç ekip üyesinin sesleri ile üç saniyelik ses dosyalarından veri tabanı oluşturulmuştur. Ardından bu ses dosyaları raporun ilerleyen kısımlarında açıklanacak olan matematiksel işlemlerle nümerik olarak elde edilmiştir, ardından yeni kaydedilen ses verisinin veri tabanındaki verilerle karşılaştırılması sonucu, ses sahibinin grup üyeleri arasından hangisine ait olduğu tespit edilmiştir.

ANAHTAR KELİMELER

Fourier Dönüşümü, Ses Analizi, FIR, AGF, MFCC, Kendall Korelasyon Katsayısı

SEMBOLLER VE KISALTMALAR

MFCC : Mel-Frequency Cepstral Coefficients (Mel-Frekans Kepstral Katsayıları)

FFT : Fast Fourier Transform

τ : Kendall's Tau

FIR : Finite Impulse Response (Sonlu Dürtü Tepkisi)

IIR : Infinite Impulse Response (Sonsuz Dürtü Tepkisi)

AGF : Alçak Geçiren Filtre

1. Giriş

İnsan sesi, konuşma, şarkı söyleme, gülme, ağlama, çığlık atma ve bağırma olmak üzere birçok farklı çeşitte olabilir. İnsan sesi frekansı, özellikle ses kıvrımlarının (ses telleri) birincil ses kaynağı olduğu insan sesi üretiminin bir parçasıdır ve her ses eşsizdir.

Bu özelliklerinden dolayı, günümüzde gerek IoT nesnelerinde gerekse güvenlik sistemlerinde ses tanıma işlemi vazgeçilmez bir adım olmuştur. Örneğin; müşteri, işlem yapmak üzere bankayı aradığında kimliğinden emin olmak için kullanılan ses imzası ya da ev ortamında kullanılan bir sanal asistanın sizin sesinizi tanıyıp tercihlerinizi anlaması önemlidir.

Her ne kadar yüz tanıma, parmak izi tanıma, göz retinası tanıma gibi sistemlere kıyasla doğruluk oranı tartışılabilir olsa da yeterli veri tabanı oluşturulduğunda ses tanıma işlemi de bu sistemler kadar doğruluk sağlayabilmektedir.

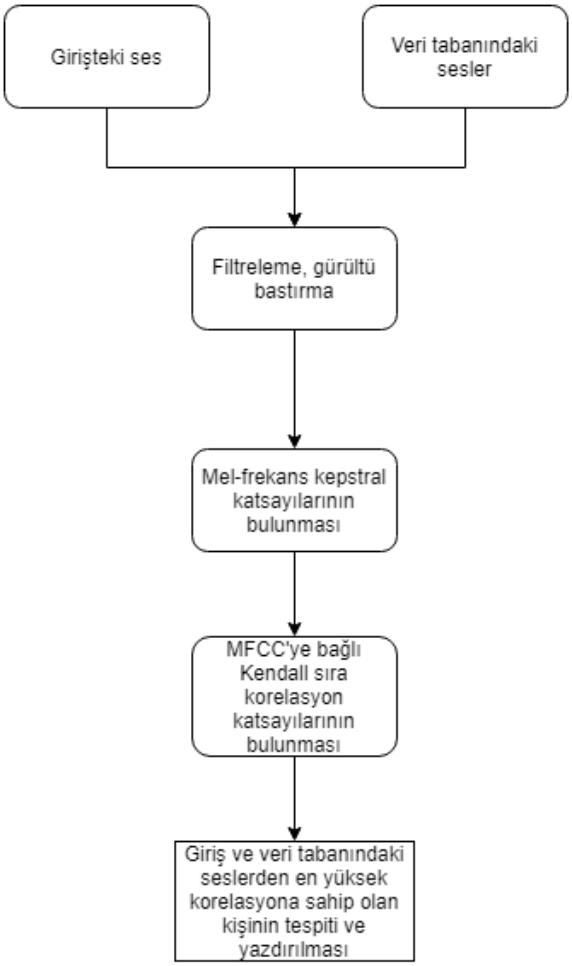
Ses tanıma işlemini bir "sorun" haline getiren ve yukarıda bahsi geçen alanlardan ayıran kısım ise insan sesinin sürekli aynı olan bir sinyal olmamasından kaynaklıdır. İnsanın parmak izinin her an değişemeyeceği gibi insan suratı veya göz retinası da her gün farklı bir şekilde belirmez. Ancak konu sese geldiğinde artık bambaşka bir anlayış mevcuttur. Çünkü günlük, haftalık, yıllık değişimlerin yanı sıra anlık ses değişimleri bile çok olasıdır. Günlük iletişimde kullanılan ses tonu ile şarkı söylerken, gülerken veya bağırırken kullanılan ses tonu arasında hem genlik hem frekans düzeyinde büyük değişiklikler mevcuttur. İki kişinin sesi arasında ayrım yapmak ve bu ayrımı belirgin bir şekilde ortaya koyabilmek için sesin nümerik özellikleri ortaya konulmalı ve bunun üzerinden yakınsamalar yapılmalıdır.

Bu çalışma, her grup üyesi için üçer saniyelik üç ses kaydı oluşturulmuş olması, yani yeterli bir veri tabanı oluşturulmaması, makine öğrenmesi ve yapay zeka algoritmaları içermemesi gibi nedenlerden dolayı ses tanıma işlemlerine giriş niteliği taşımaktadır. Buna rağmen grup üyelerinin seslerini veri tabanında bulunan örneklerle girdi

olarak verilen sesi karşılaştırarak birbirinden ayırt edebilmektedir.

2. Deneyler ve Analiz

Öncelikle problemin çözümünde büyük öneme sahip üç yöntem açıklanacaktır. Bu yöntemler projedeki verilere uygulanan işlem sırasına göre açıklanacaktır. Bu yöntemlerden ilki filtreleme, ikincisi MFCC ve son olarak Kendall rank correlation coefficient (Kendall korelasyon katsayısı) yöntemleri ve projenin akışında nasıl kullanıldığı açıklanacaktır.



Şekil 1. Akış Diyagramı

2.1.1 Filtreleme

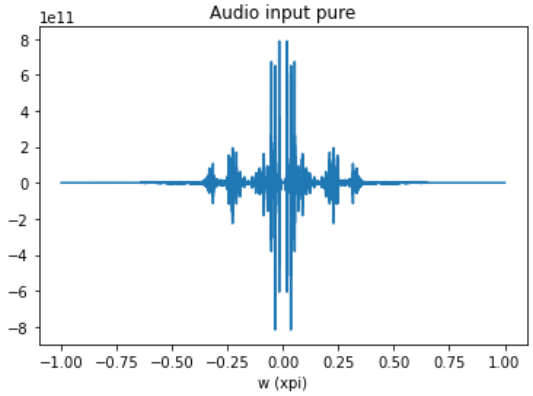
Ses tanımada önemli problemlerden biri de ortamdaki gürültünün sesin karakteristiğine etkimesi sonucunda yanlış çıkarımlar yapılabilmesidir. Sinyal işlemede gürültü, bir sinyalin yakalama, depolama, iletim, işleme veya dönüştürme sırasında maruz kalabileceği istenmeyen (ve genel olarak bilinmeyen) değişiklikler için genel bir terim olarak tanımlanabilir.[1]

Bu çalışmada gürültü genel olarak bilgisayarın fan sesi veya arka plandan gelen istenmeyen başka bir insan, araba sesi olarak karşımıza çıkmaktadır. Gürültü azaltma ses analizinin doğruluğunu arttırabilmek için önemli bir adımdır ve ses sinyalinin gürültüsünü azaltmanın yolu, ses verisine sonraki adımlarda anlatılacak olan işlemleri uygulamadan önce frekans analizi yaptıktan sonra filtrelemekten geçer.

Dijital filtreler dürtü cevabına göre iki türe ayrılır. Bunlar Infinite Impulse Response (Sonsuz Dürtü Cevabı) ve Finite Impulse Response (Sonlu Dürtü Cevabı) olarak sınıflandırılır. Adlarından da anlaşılacağı gibi, her filtre türü, dürtü yanıtının uzunluğuna göre sınıflandırılmaktadır.

Bu projede kaydedilen sesin ve veri tabanındaki seslerin gürültüsünün filtrelenmesinde FIR filtre kullanılmıştır. Sebebi ise FIR filtrelerin IIR filtrelere göre lineer faz, sabit grup gecikmesine sahip olması sonucunda göreceli harmonik ilişkileri koruması ve kararlılık açısından daha başarılı olmasıdır.[2]

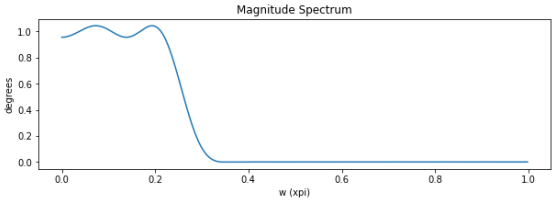
Filtrenin türünün dürtü cevabına bağlı olarak seçilmesinden sonraki adım, filtrenin kesim frekansının ve tipinin (alçak geçiren, yüksek geçiren vs.) belirlenmesidir. Bunun için sesler Fourier Uzayında incelenerek gürültünün hangi frekanslarda yoğunlaştığı tespit edilmiştir.



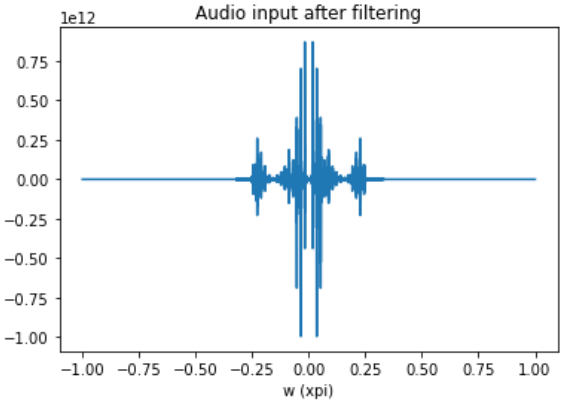
Şekil 2. Kaydedilen Sesin Filtreleme İşleminden Önce Frekans Spektrumu

Kaydedilen seslerdeki gürültünün genel olarak bilgisayar fanından kaynaklı olduğu bilinmektedir ve bu sesin de yüksek frekanslarda bir ses olduğu gözlemlenmektedir, ayrıca frekans spektrumunda alçak frekanslı bölgelerdeki yüksek genlikli işaretlerin insan sesi olduğu belirlenmiştir. Bunun sonucunda, tasarlanması gereken filtre tipinin alçak geçiren filtre olduğu belirlenmiştir.

Yapılan analizler sonunca *"pyfda"* ortamında tasarlanan filtrenin optimum verime sahip olabilmesi için, filtre derecesi (ideale yakın olması için) *N=38* ve kesim frekansı *0.2 π* olan alçak geçiren Equiripple filtre tasarlandı.



Şekil 3. Gürültü Azaltıcı (AGF) Filtrenin Genlik Spektrumu



Şekil 4. Kaydedilen Sesin Filtreleme İşleminde Sonra Frekans Spektrumu

Şekil 4'te ses sinyalinin, alçak geçiren filtrenin dürtü cevabının katsayılarıyla konvolüsyon işlemine alınması suretiyle filtrelenmiş ses sinyali elde edilmiştir. Kaydedilen sesteki gürültünün olabildiğince bastırıldığı gözlemlenmektedir. Gürültü bastırma işlemi veri tabanındaki sesler için de uygulanmıştır.

2.1.2 Mel-Frekans Kepstral Katsayıları

Mel-Frekans Cepstrum (MFC), lineer olmayan bir Mel frekans ölçeğinde bir log güç spektrumunun lineer kosinüs dönüşümüne dayanan bir sesin, kısa vadeli güç spektrumunun bir temsilidir. MFCC, güvenlik amacıyla havayolu rezervasyonlarında, telefonda konuşulan numaraları ve ses tanıma sistemlerinde ve tür sınıflandırması, ses benzerliği ölçümleri vb. gibi uygulamalarda giderek daha fazla kullanılmaktadır.

Bir ses sinyali sürekli değişir, bu yüzden işleri basitleştirmek için kısa zaman ölçeklerinde ses sinyalinin fazla değişmediği varsayılır (yani istatistiksel olarak durağan). Bu yüzden sinyal 10-25ms (fonksiyonda tanımlı) ile çerçevelenir.

MFC katsayılarının elde edilmesinde kullanılan algoritma şu şekildedir:

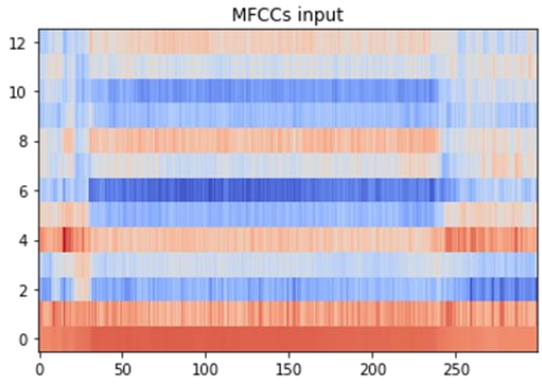
- Bir sinyalin (pencereli bir alıntısının) Fourier dönüşümü alınır.
- Fourier dönüşümünden elde edilen spektrumun güçlerini, üçgen örtüşen pencereler veya alternatif olarak kosinüs örtüşen pencereler kullanarak Mel ölçeğine eşlenir.
- Her bir Mel frekansındaki güçlerin logaritması alınır.
- Mel log güçleri listesinin ayrık kosinüs dönüşümü alınır.
- MFCC'ler, elde edilen spektrumun genlikleri olarak karşımıza çıkar.[3]

MFCC'nin Python ortamında uygulanışı aşağıda görüldüğü gibidir:

```
sesin_ornekleme_frekansi, ses_verisi =
wav.read("database/ses_verisi.wav")

ses_verisi_mfcc =  mfcc(ses_verisi,
sesin_ornekleme_frekansi)
```

Yukarıdaki kod parçasında .wav formatındaki ses verisinin okunması ve MFCC katsayılarının elde ediliş şekli gösterilmiştir. Bu yöntem ile veri tabanındaki seslerin ve girdi olarak, kime ait olduğu bulunmak üzere, kaydedilmiş olan sesin filtrelenmesinin ardından MFCC katsayıları bulunmuştur.



Şekil 5. Girdi Olarak Alınan Sesin Filtrelendikten Sonraki MFCC Grafiği

Şekil 5'te MFCC uygulandıktan sonraki çıktı gözükmektedir ve bu tüm çerçevelerden çıkarılan özellik vektörlerine sahip bir matristir. Bu çıktı matrisinde satırlar karşılık gelen çerçeve numaralarını ve sütunlar karşılık gelen özellik vektör katsayılarını [1-4] temsil eder. Son olarak bu çıktı matrisi korelasyon işlemi için kullanacaktır.

2.1.3 Kendall Korelasyon Katsayısı

İstatistikte, korelasyon veya bağımlılık, iki rastgele değişken veya iki değişkenli veri arasındaki nedensel olsun veya olmasın herhangi bir istatistiksel ilişkidir. En geniş anlamda korelasyon, herhangi bir istatistiksel ilişkidir, ancak genellikle bir çift değişkenin doğrusal olarak ilişkili olma derecesini ifade eder.
Bu projede, MFCC'de elde edilen katsayıların korelasyon yöntemi ile benzerliğinin ölçümü sağlanarak, analiz yapılmıştır. Bunun için korelasyon katsayılarına değinilmesi gerekmektedir.

Korelasyon katsayısı, iki değişken arasındaki istatistiksel ilişkiyi ifade eden, korelasyonun bir tür sayısal ölçüsüdür. Değişkenler, genellikle örnek olarak adlandırılan belirli bir gözlem veri setinin iki sütunu veya bilinen bir dağılıma sahip çok değişkenli bir rastgele değişkenin iki bileşeni olabilir. Bu projede değişkenler, veri tabanında bulunan kayıtlı sesler ve kimin sesi olduğu bulunmak üzere girdi olarak uygulanan ses olarak karşımıza çıkmaktadır.[4]

Kendall'ın sıra katsayısı için açık bir ifade şöyledir:

$$\tau = \frac{2}{n(n-1)} \sum_{i<j} \operatorname{sgn}\left(x_i - x_j\right) \operatorname{sgn}\left(y_i - y_j\right)$$

Kendall'ın sıra korelasyon katsayısı (τ) gibi sıra korelasyon katsayıları; bir değişken arttıkça, diğer değişkenin, bu artışın doğrusal bir ilişki ile temsil edilmesini gerektirmeden artma eğilimini ölçer. Değişkenlerden biri artarken diğeri azalırsa sıra korelasyon katsayıları negatif olacaktır.

Veri tabanında seslerin ve girdi olarak alınan sesin MFCC katsayılarının korelasyon katsayılarının bulunması şu şekilde ifade edilebilir; veri tabanındaki sesin MFCC katsayıları artarken, girdi olarak alınan sesinki de artıyorsa, korelasyon katsayısı pozitif bulunur. Eğer girdinin MFCC katsayıları azalıyor, veri tabanındaki sesin MFCC katsayıları artıyorsa negatif korelasyon bulunacaktır. Böylece iki veri arasındaki MFCC katsayılarının benzerliği elde edilecektir.

Bu yöntem veri tabanındaki her sese uygulanarak τ değerleri elde edilmiştir ve projede amaç, ses sahibi olan "kişiyi" tanımak olduğundan; girdinin ve veri tabanındaki o kişiye ait tüm seslerin korelasyonu kümülatif olarak toplanmıştır. Böylece girdinin, kişinin veri tabanındaki "aaa", "bee" ve "cee" seslerine olan toplam benzerliği elde edilmiştir.
Projede Kendall'ın korelasyon katsayısı yöntemi seçilmesinin nedeni, kısıtlı veri tabanında diğer korelasyon katsayısı yöntemlerine göre daha sağlıklı sonuçlar vermesidir.

2.2 Analiz

```
Cumulative Correlations
{'Baki': 1.1852757274105152, 'Ebru': 1.4680359634719427, 'Safak': 1.5805203906922278}

Owner of the audio Safak
```

Şekil 6. Kod Çıktısı

- Şekil 6'te projeden örnek kod çıktısı görülmektedir. Bu çıktıda sonuç beklendiği gibidir. Sonuçlar veri tabanın sınırlı veri olmasından ve "bee", "cee" seslerinde "e" sesinin benzerliğinden ve uzunluğundan kaynaklı hatalı çıktılar verebilmektedir. Kodun girdilerle yapılan testler sonucunda %80 doğrulukta çalıştığı gözlemlenmiştir. Korelasyon katsayısının aynı veri seti için 1 olduğu bilinmektedir, böylece veri tabanındaki seslerin girdi olarak kullanılması durumunda çıktıların %100 doğrulukta olduğu gözlemlenmiştir.

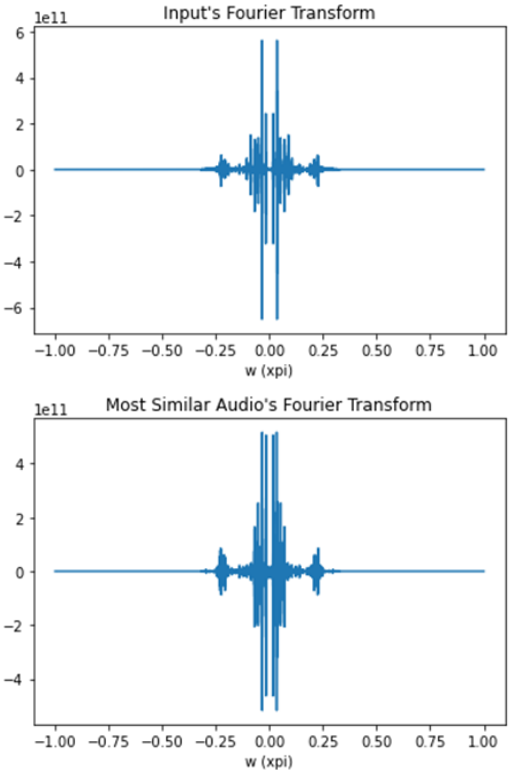
- Kendall korelasyonu O(n^2) hesaplama karmaşıklığına sahiptir, Spearman korelasyonu ise O(n logn) karmaşıklığına sahiptir. Burada n örnek sayısını ifade eder ve bilinmektedir ki örnek sayısı çok büyük olmadığında O(n^2) karmaşıklığı daha verimlidir.

- Projede ilk olarak, işaretlerin Fourier dönüşümleri üzerinden korelasyon

katsayıları bulunarak işlem yapılmaya çalışıldı; fakat istenen doğrulukta sonuçlar elde edilemedi ve önce MFCC sonra korelasyon katsayılarının elde edilmesiyle algoritma geliştirildi.

- Seslerin benzerliklerini kıyaslamada karşılaşılan bir diğer algoritma Dynamic Time Warping olarak bilinen algoritmadır. DTW'nin O(n) karmaşıklığı ile çalıştığı bilinmektedir, yani Kendall'ın korelasyon katsayıları yöntemine göre daha verimlidir; fakat iki dizinin farkını alarak çalışan bu yöntemin projede tekrar eden "e" sesleri için doğruluğunun düşük olacağı düşünüldüğünden korelasyon yöntemi tercih edildi.

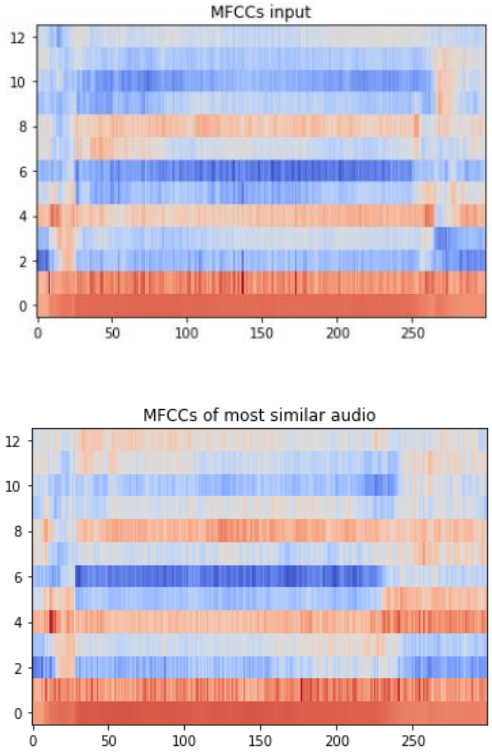
- Alçak geçiren filtre tasarlanırken çok düşük bir kesim frekansı seçilirse sesin karakteristiğinin (incelik, kalınlık) kaybolduğu gözlemlendi, çok yüksek bir kesim frekansının seçilmesi durumunda ise gürültü bastırılamıyordu. Yapılan analizler sonucunda alçak geçiren filtre için uygun kesim frekansı 0.2 π olarak ayarlandı.



Şekil 7. Girdi Olarak Test Edilen Sesin ve Veri Tabanında Ona En Benzer Bulunan Sesin Fourier Dönüşümlerinin Kıyaslanması

Şekil 7'de girdi olarak test edilen sesin ve veri tabanındaki seslerin, MFCC'lerinin korelasyonlarının hesaplanması sonucunda, en yüksek korelasyona sahip, yani en büyük benzerliğin gözlemlendiği (a18_safak_b.wav) sesinin ve girdinin frekans uzayında karşılaştırılması görülmektedir. Frekans uzayında seslerin benzerlik gösterdiği açıkça gözlemlenmektedir.

Şekil 8'de girdi olarak test edilen sesin ve veri tabanındaki seslerden, girdi ile en yüksek korelasyona sahip olanının MFCC katsayılarının grafiklerinin karşılaştırılması görülmektedir. Korelasyonun benzerlik yakalamadaki verimi gözlemlenmektedir.



Şekil 8. Girdi Olarak Test Edilen Sesin ve Veri Tabanında Ona En Benzer Bulunan Sesin MFCC Katsayılarının Kıyaslanması

3. Sonuç ve Yorum

Bu çalışmada ilk başta konu hakkında pratik bilgimiz olmamasına rağmen araştırmalarımız ve denemelerimiz sonucunda ekip üyeleri arasındaki sesler ayırt edildi. Araştırmalar esnasında konu hakkında yapılan çalışmaların genellikle "voice/speaker recognition" değil de "speech recognition" üzerine olduğu gözlemlendi. Bundan dolayı genellikle "speech recognition" üzerine yapılan çalışmalar ve yayımlanan makalelerden faydalanıldı ve öğrenilen bilgiler projeye uyumlu hale getirildi. Sesin öznitelik analizinde kullanılan LPCC, MFCC gibi birçok teknik, araştırmalar esnasında öğrenildi ardından seslerin MFCC katsayılarını elde edilerek çalışma detaylandırıldı. Ve kıyaslama için korelasyon, çapraz korelasyon gibi işlemlerin yanı sıra Dynamic Time Warping (DTW) işlemleri ile de deneme yapıldı. Yapılan bu denemeler esnasında doğruluğu test edebilmek için ekip üyeleri tarafından her farklı yöntem için yetmişin üzerinde deneme yapılarak sonuçların çetelesi tutuldu. Hangi yöntem, hangi kişi ve hangi harf için en çok doğruluğa sahip olduğu belirlenerek çalışma detaylandırıldı ve raporda aktarılmış olan sonuçlara ulaşıldı.

Kullanılan çözüm, normal ses tonu ile konuşulduğunda doğru bir çözüm veriyor; ancak inceltme veya kalınlaştırma gibi denemeler yapıldığında yetersiz kalıyor. Bunun sebebi veri tabanında bulunan örneklerin yetersiz olmasıdır.

Kullanılan yöntemi değiştirmeden, doğruluk oranı daha yüksek bir çalışma elde edebilmek için öncelikle veri tabanı güncellenmeli ve örnekler arttırılmalıdır. Sadece normal ton ile harfleri söylemek yerine günlük konuşmada değişen ses tonu aralığımız ve genliğimiz dikkate alınarak detaylı bir veri tabanı oluşturulduğu takdirde programın yanılma oranının bir hayli azalacağı, ekip üyelerinin araştırmaları ve deneyimleri sonucunda ön görülmektedir.

Çözümü geliştirmek için ise makine öğrenmesi algoritmaları kullanılabilir. Detaylandırmak gerekirse seslerin, MFCC ve LPCC yöntemleri ile öz niteliklerinin elde edilmesinin ardından bu diziler birbirleri ile kıyaslanıp sınıflandırma işlemi yürütebilmek için Vektör Kuantalama (VQ/Vector Quantization) yöntemi kullanılabilir. Bu yöntemle veri tabanından okunan veriler bir düzlem üzerinde nokta şeklinde gösterilecek olursa; noktaların üç farklı bölgede yoğunlaştığı görülecektir ki bu bölgeler de ekip üyelerini temsil eder ve sınıflandırma daha hızlı bir şekilde yapılmış olur. Aynı zamanda bu sınıflandırma sayesinde daha yüksek bir doğruluk oranı da elde edilebilir.

Birçok yönüyle bu çalışma öğrenilenleri ve araştırmaları pratiğe dökme konusunda deneyim sağlamıştır. Öğrenilen teknik bilgilerin, denemelerin yanı sıra farklı yöntemler arasından optimizasyon işlemleri yapılması, çözümde maksimum doğruluk seviyesine ulaşmanın hangi yöntemlerle mümkün olacağının araştırılması, öğrenilmesi ve en önemlisi ekip üyeleri ile uyumlu çalışabilmenin projeye olan olumlu yansıması bizler için iyi bir tecrübe olmuştur.

Kaynakça

Kod için kaynakça:
https://python-speech-features.readthedocs.io/en/latest/

https://realpython.com/playing-and-recording-sound-python/

Rapor için kaynakça:
[1] https://en.wikipedia.org/wiki/Noise

[2]https://en.wikipedia.org/wiki/Filter_(signal_processing)

[3] https://en.wikipedia.org/wiki/Mel-frequency_cepstrum

[4]https://en.wikipedia.org/wiki/Kendall_rank_correlation_coefficient