GSM AĞI ÜZERİNDEN GÜVENLİ SES İLETİMİ

BİTİRME ÖDEVİ
Mehmet Akif Özkan
040070361

Bölümü: Elektronik ve Haberleşme Mühendisliği Bölümü
Programı: Elektronik Mühendisliği

Danışmanı: Yrd. Doc. Dr. Siddika Berna ÖRS YALÇIN

MAYİS 2011
ÖNSÖZ


Ayrıca eğitim hayatım boyunca maddi ve manevi desteklerini hissettiğim aileme sonsuz saygı ve teşekkürü borç bilirim.

Mayıs 2011

M.Akif Özkan
# İçindekiler

<table>
<thead>
<tr>
<th>Sayfa</th>
<th>Önsöz</th>
<th>ii</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>İçindekiler</td>
<td>iii</td>
</tr>
<tr>
<td></td>
<td>Kısaltmalar</td>
<td>v</td>
</tr>
<tr>
<td></td>
<td>Özeti</td>
<td>vi</td>
</tr>
<tr>
<td></td>
<td>Summary</td>
<td>viii</td>
</tr>
<tr>
<td>1.</td>
<td>Giriş</td>
<td>1</td>
</tr>
<tr>
<td>2.</td>
<td>Konuşmanın GSM Üzerinde Şifrelenerek Gönderilmesi</td>
<td>4</td>
</tr>
<tr>
<td>3.</td>
<td>Doğrusal Öngörü Yöntemile Konuşmanın Kodlanması</td>
<td>8</td>
</tr>
<tr>
<td>3.1.</td>
<td>İnsan Konuşmanın Modellenmesi</td>
<td>9</td>
</tr>
<tr>
<td>3.2.</td>
<td>LPC Modeli</td>
<td>10</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Çerçeveleden Konuşmaların Örtüştürülerek İncelenmesi</td>
<td>11</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Çerçeveleden Konuşma İşaretiqin Seslik Sınıflandırılması</td>
<td>12</td>
</tr>
<tr>
<td>3.2.3</td>
<td>LPC Sentez Filtresinin Modellenmesi</td>
<td>20</td>
</tr>
<tr>
<td>3.2.4</td>
<td>LPC Analiz Filtresinin Modellenmesi</td>
<td>21</td>
</tr>
<tr>
<td>3.2.4.1</td>
<td>Korelasyon Yöntemiyle Analiz Filtre Denklemlerinin Elde Edilmesi</td>
<td>23</td>
</tr>
<tr>
<td>3.2.4.2</td>
<td>Otokorelasyon Yöntemiyle Analiz Filtre Denklemlerinin Elde Edilmesi</td>
<td>24</td>
</tr>
<tr>
<td>3.2.5</td>
<td>Levinson-Durbin Algoritması</td>
<td>25</td>
</tr>
<tr>
<td>3.2.6</td>
<td>Kazanç Hesabı</td>
<td>29</td>
</tr>
<tr>
<td>3.2.7</td>
<td>Enerjinin Normalize Değer Gösterimi</td>
<td>30</td>
</tr>
<tr>
<td>3.2.8</td>
<td>Perdenin Tahmin Edilmesi</td>
<td>30</td>
</tr>
<tr>
<td>3.3</td>
<td>LPC Modelinin Matlab Ortamında Tasarlanması</td>
<td>32</td>
</tr>
<tr>
<td>4.</td>
<td>Gelişmiş Şifrelemme Standardı</td>
<td>37</td>
</tr>
<tr>
<td>4.1</td>
<td>Bayt Değiştirme (Sub Bytes)</td>
<td>40</td>
</tr>
</tbody>
</table>
**KISALTMALAR**

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSM</td>
<td>Global System for Mobile Communications</td>
</tr>
<tr>
<td>AES</td>
<td>Advanced Encryption Standard</td>
</tr>
<tr>
<td>FPGA</td>
<td>Field Programmable Gate Array</td>
</tr>
<tr>
<td>LCD</td>
<td>Liquid Crystal Display</td>
</tr>
<tr>
<td>LSF</td>
<td>Linear Spectrum Frequencies</td>
</tr>
<tr>
<td>KBPS</td>
<td>Kilo bit per second</td>
</tr>
<tr>
<td>GSM FR</td>
<td>GSM Full Rate</td>
</tr>
<tr>
<td>LPC</td>
<td>Linear Predictive Coding</td>
</tr>
<tr>
<td>LBG</td>
<td>Linde Buzo Gray Algoritması</td>
</tr>
<tr>
<td>PSTN</td>
<td>Public Switched Telephone Network</td>
</tr>
<tr>
<td>CELP</td>
<td>Code Excited Linear Prediction</td>
</tr>
</tbody>
</table>
GSM AĞI ÜZERİNDEN GÜVENLİ SES İLETİMİ

ÖZET

Mobil iletişim için küresel sistem (Global System for Mobile Communications, GSM) ile haberleşmenin yaygınlaşması hayatımızı kolaylaştırmıştır ancak çok ciddi güvenlik açıklarını da beraberinde getirmiştir. Bu çalışmada, cep telefonları ile haberleşerek GSM üzerinden yapılan görüşmelerin servis sağlayıcıları dahil üçüncü kişilere karşı korunmasını sağlayacak bir sistem geliştirilmiştir.


Çalışma boyunca sayısal verinin şifrelenmesinin sağlanması için, 128 bit gelişmiş kodlama standardı (Advanced Encryption Standard, AES) algoritması sahada programlanabilir kapı dizisi (Field Programmable Gate Array, FPGA) üzerinde gerçekleştirildi. Sayısal verilerden, bozulmadan iletebilecek konuşmaların edilebilmesi için gerekli özelliklere karar verildi ve istenen özelliklere ait parametreler NTIMIT ses veritabanı taranarak elde edildikten sonra LBG (Linde–Buzo–Gray algorithm) algoritması kullanılan kod kitapları tasarlandı. Sayısal verileri, kod kitaplarını kullanarak kodlayan ve doğrusal öngörü yöntemiyle (LPC) konuşmalar sentezleyebilen bir kodlayıcı tasarlandı. Matlab ortamında tasarlanan kodlayıcı ile şifreleme donanımı, seri port kullanılarak haberleştirildi. Cep telefonunu ifade eden 13 kbps (kilo bit per second) GSM FR (Full Rate) kodlayıcısı bilgisayar ortamında modellendi. FPGA de gerçekleştirilen şifreleme donanımına karakter LCD (Liquid Crystal Display) bağlandığı. Sonu colar sistem çalışan olduğu gözlemlendi.
SUMMARY

SECURE VOICE COMMUNICATION OVER GSM

With the spread of Global System for Mobile communications (GSM) usage for communication facilitated our lives, but also brought many serious security vulnerabilities. In this study, a system developed which communicates through GSM mobile phones and provides protection for interviews against third parties including with service providers developed. GSM line is sensitive to human speech to be more efficient and provide more quality for transmission. In addition, a tool should be used to compress speech to transmit speech over GSM. For these reasons, speech cannot be transmitted to the GSM line directly after encrypted. In this study, the encrypted speech which is a data stream, formed speech like waveform by the designed coder to transmit through the GSM line. In addition, in this thesis, a new method is suggested as a trained coder structures.

During the study to provide encryption for data stream, 128-bit advanced encryption standard (AES) algorithm is designed using the field programmable gate array (FPGA). The features needed to achieve to transmit speech with no disruption is decided and desired speech characteristics are obtained by scanning the database of NTIMIT, then LBG (Linde-Buzo-Gray algorithm) algorithm is used to design codebooks which includes speech parameters. A coder using the linear predictive coding method (LPC) and codebooks, is designed to synthesize speech like waveforms from data stream. The coder is designed with MATLAB and communicated with encryption hardware using the serial port. Representing the mobile phone, 13 kbps (kilo bits per second) GSM FR (Full Rate) codec is modeled on computer. AES hardware is connected with character LCD (Liquid Crystal Display) in FPGA. As a result system is observed to work.
1. GİRİŞ

GSM teknolojisi sayesinde, herhangi kablolu bir bağlantıya ihtiyaç duyulmadan dünyanın neredeyse her noktasından haberleşme mümkün hale gelmiştir. Hayatın her alanında önemli bir yere sahip olan GSM üzerinden yapılan konuşmalar; bireylerin şahsi bilgilerinden gizli tutulması gereken devlet bilgilerine kadar, özellik arz eden içeriklere sahip olabilirler. Bu yüzden konuşmaların güvendiği çok önemlidir.


Bu çalışmada, GSM üzerinden yapılan görüşmelerin servis sağlayıcıları dahil üçüncü kişilere karşı korunmasını sağlayacak bir sistem geliştirilmiştir. Tasarlanan sistem, cep telefonları ile haberleşerek konuşmaları güvenli hale getirebilen gerçek zamanlı bir modül tasarımını için gerekli olan teorik altyapıyı ve tasarlanacak modülün temel yapısını içermektedir. Konuşmanın GSM üzerinde tasarlanmasını istenen gerçek zamanlı modül ile güvenli bir şekilde iletimi sırasında; gelen konuşma sayısallaştırılarak bit dizileri elde edilecek, elde edilen bit dizileri şifrelenecek, şifrelenen sayısal veriler kodlanarak sentetik konuşmalar oluşturulacak, üretilen sentetik konuşmalar cep telefonuna gönderilicek; cep telefonunun GSM kodlayıcısıyla konuşma kodlandiktan sonra hatta gönderilicek; hatta gönderilen konuşmanın iletimi...

Tasarlanan modül FPGA üzerinde tasarlanacak bir donanım olması özelliği ile bir ilktir. Ayrıca benzer fikirlerle önerilen kodlayıcı yapıları olmasına rağmen, oluşturulan kod kitapçıkları çalışmaya özgürdür.

Tasarlanacak sistemde ait genel şema ve kodlayıcı yapısı ikinci bölümde, LPC ile konuşmanın analizi ve sentezi üçüncü bölümde, gerçekleştirilen AES şifreleme algoritmaları dördüncü bölümde, kod kitaplarının oluşturulması ve konuşmanın
kodlanması için kullanılması dördüncü bölümde anlatılmış, beşinci bölümde elde edilen sonuçlardan bahsedilmiştir.
2. KONUŞMANIN GSM ÜZERİNDEN ŞİFRELENEREK GÖNDERİLMESİ


GSM iletişiminde hattın verimli kullanılabilmesi için konuşma, temel özelliklerine göre sıkıştırılarak hatta verilir ve alındıktan sonra sıkıştırma açılır. Sıkıştırma işlemi sonucunda oluşturulan sinyal, sıkıştırılmamış haline temel karakteristikleri açısından benzer ancak dalga formları aynı değildir. Konuşma ayrıca GSM iletim hattı üzerinde yeniden kodlanır, bu şekilde iki kere kodlanmasına peş peşe kodlanma (tandeming) denir.

Şekil 2.1: Sesin GSM konuşma kanalı üzerinde iletişimi [5].

Şifreleme işleme, konuşmayı hatta gönderen konuşma kodlayıcısından önce ya da sonra yapılabilir. GSM hattı gelen ses üzerinde duyarlıdır ve konuşma olmayan sinyalleri bastırır. Bu yüzden sesin şifrelenerek iletilebilmesi için şifreledikten sonra elde edilecek sayısal verilerden konuşmaya benzeyen sesler sentezleyeceğ bir
kodlayıcı gerekir. Tasarlanacak kodlayıcı, konuşma kodlayıcısında önce şifrelenen verileri konuşmalara çevirerek konuşma kodlayıcısına gönderecek, konuşma kodlayıcısı ise bu konuşmaları GSM hattına gönderecektir.

Şekil 2.2: Tüm sisteme ait şema [2].


Tasarlanacak kodlayıcı enerji, vokal filtre spektrum eğilimi, perde frekansı gibi konuşmayı ifade eden en önemli parametreleri kullanarak LPC yöntemiyle konuşmalar sentezleyecektir. Bu iş için konuşma kodlayıcısından ve GSM iletimi sırasındaki kodlayıcının bozulmadan geçebilecek konuşmalara ait parametrelerin saklandığı kod kitapları oluşturulacaktır.
Şekil 2.3: Kodlayıcının sayısal verilerden konuşma sentezleyen kısmında[4].

Sentezlenen konuşma GSM FR ses kodlayıcısı ile iletim hattına gönderilerek, karşı taraf iletilen konuşmayı alıktan sonra kod kitaplarına bakarak sayısal veri karşılıklarını elde edecek. Elde edilen şifre kodlarının doğru çözülebilmesi için kodlayıcı tarafından algılanan ve iletilen sayısal verilerin aynı olması gerekir.

Şekil 2.4: Kodlayıcının konuşmadan sayısal verileri sentezleyen kısmında[4].

Sistemden kaynaklanacak hataların giderilebilmesi için kod kitaplarının hataları düzeltecek şekilde tasarlanması gereklidir. Bu çalışmada iletişim, minimum hatalı olması için kod kitaplarına yerleştirilecek konuşmalara ait özellikleri belirlendi. İstenecek özellikteki konuşmalara ait, temel parametrelerin öğrenilmesi için NTIMIT ses veritabanı taramaktaydı. İletimdeki hataların düzelterek, sayısal verilerin değişmeden elde edilebilmesi için veritabanı tarama Suarez parametreler LBG algoritması
kullanılarak birbirine en uzak noktalarda seçildi. Ayrıca girişteki ve çıkıstaki kod kitapları farklı seçilerek, sistemden kaynaklanan hataların düzeltilebileceği gösterilerek eğitilebilir kodlayıcı yapılarına dair yeni bir metot önerildi.
3. DOĞRUSAL ÖNGÖRÜ YÖNTEMİYLE KONUŞMANIN KODLANMASI

Konuşma sinyalinin, kullanılabileceği uygulamaya göre daha az parametre ile ifade edilmesine konuşmanın sıkıştırılması (Speech Compression) yada konuşmanın kodlanması (Speech Coding) denir. Konuşmanın kodlanması için geliştirilen yöntemler genellikle kayıplıdır ancak insan kulağı, konuşmanın kalitesindeki düşüş yeterince algılayamadığı için kabul edilebilirler. Sıkıştırılma işlemi sırasında konuşmadaki sessiz kısımlar iletim için kodlanmaz, bu sayede konuşmayı temsil eden bit sayıısında ciddi bir düşüş sağlanır.

Bant genişliğini artıracak teknolojilerin gelişmesine rağmen, konuşmanın sıkıştırılması ve bant genişliğinin daha verimli kullanılması haberleşme teknolojisinin en temel problemlerindendir. Konuşmanın sıkıştırılması için bu zamana kadar çok çeşitli teknikler geliştirilmiştir ancak bu tekniklerin büyük bir kısmı, konuşmanın 300 Hz ile 3400 Hz arasında sınırlı bir frekans aralığına sahip olması ve konuşmanın yavaş anatomik hareketlerden oluştuğu ilkeleri üzerine kurulmuştur.


LPC algoritması, kodlama (encode) ve kod çözme (decode) olarak adlandırılan iki temel bloktan oluşur. Kodlama bloğu konuşmanın çerçeveleri (frame) halinde incelenir. İncelenen çerçeve analiz edilerek, yeniden sentezlenecek tasarruf edilebilir, konuşmayı ifade eden parametreler kodlama işleminden geçirilerek bit dizileri (bit stream) elde edilir. Kod çözme kısmında ise elde edilen bit dizisi kodlama aralıklarına göre çözülenerek sentezlemeye filtersi ve bu filtreyle uyaran işaret edilerek sentetik ses sentezlenir.

3.1. İnsanın Konuşmanın Modellenmesi

(unvoiced) sesler oluşur. Sesli seslerin enerjileri, sessiz seslere oranla daha fazladır. İnsanda sesin oluşması sistemi modellendiğinde akciğer, kaynak; vokal sistem, konuşmamızı ortaya çıkaran ses çeşitlerini oluşturan filtre olur. LPC konuşma kodlayıcısında da kullanılan bu modellemeye kaynak-filtre modeli (source-filter model) denir.

Şekil 3.1: İnsanda ses oluşumunun modellenmesi [8].

3.2. LPC Modeli

LPC, konuşma analizinde kullanılan en güçlü uygulamalarından biridir. Bu algoritma, konuşmaya ait temel parametrelerin (sesli olup olmaması, formantları, periyot bilgisi (pitch), vb.) bulunmasında ve konuşmanın düşük bit hızında (low bit rate) iletilmesi yada saklanması için modellenmesinde kullanılan temel metotlardan biri haline gelmiştir[7]. Bu metodun en önemli özelliği konuşmayı lineer zamanla değişen bir sisteme ait parametrelerle, güvenli ve hızlı bir şekilde ifade edilmesine imkan vermesidir.

LP algoritması kullanarak konuşmanın analiz edilmesine, LP (Linear predictive) konuşma analizi; analiz sonucu elde edilen parametreler kodlanıp sayısal bir veri elde edilmesine, LPC konuşmanın kodlanması denir. LP konuşmanın analiz edilmesi ve LPC konuşmanın kodlanması, analiz/kodlayıcı (encoder) ve sentez/kod çözücü (decoder) olmak üzere iki temel kısımdan oluşur. Konuşma, yavaş değişen bir işaret olduğundan kısa çerçevelere (20ms, 40ms gibi) bölündükten sonra elde edilen konuşma parçaları durağan sinyaller olarak kabul edilirler.
FS-1015 standardına göre gelen konuşma 8 khz de örneklenir. Analiz kısmında, konuşma kısa çerçevelere parçalar haline ayrıp her bir parça için; sinyalin gücü bulunarak kazancın, sesli (voiced) veya sessiz (unvoiced) olması, sesli parçalar için perde periyodu (pitch period) ve vokal sistemin tasarlanması için LP katsayları (Linear prediction coefficients) bulunur. Sentez kısmında, analizden gelen parametreler incelenerek konuşma sinyali tekrar elde edilir. Sinyal, sesli ise perde periyoduna sahip bir impuls dizisi (impulse train), sessiz ise rastgele gürültü (random noise) elde edilen kazanç ile çarpılarak vokal model bloğuna giriş olarak verilir. LP katsayları kullanılarak vokal model filtresi tasarlanır.

**Şekil 3.2: LPC Konuşma Kodlama/Analizi [9].**

3.2.1. **Çerçevelenen Konuşmaların Örtüştürülerek İncelenmesi**

Konusma sinyalleri analiz edilirken çerçeve eleştirileri ayrınlıkta alınması gerekliliğini konuşuldu. Çerçevelere ayırma işleminde konuşma sinyallerine ait bazı özellikler kaybolabilir. Aynı özellikte sahip konuşma parçasi, yarısı önceki çerçevede yarısı sonrası çerçevede olacak şekilde bölünebilir. Bu durumda iki çerçevede de bu özellik görülemez. Bu kayipten kurtulmak için çerçeve genelde %50 ve %25 arasında bir oranla ortüstürlerek seçilerek konuşmaya ait daha fazla özellik yakalanmaya çalışılır.
%50 oranında örtüştürülerek çerçevelenen bir konuşma aralığı şekil 3.3’de bulunmaktadır.

**Şekil 3.3:** %50 oranında örtüştürülerek seçilen bir konuşma aralığı [10].

Örtüştürme işlemi uygulanarak incelenen konuşmalarda, analizi yapılacak çerçeve sayısı artar; dolayısıyla her bir çerçeve, kendi uzunluğu boyunca sentezlendiğinde konuşma dosyası büyür. Analiz işleminde, örtüştürme yapıldığında sentezlenen konuşma çerçeve uzunluğu örtüştme oranına göre küçülür. Mesela %50 oranında örtüştürme yapılarak incelenen bir konuşmanda, her bir periyot uzunluğundaki çerçeveler yarım periyot uzunluğunda sentezlenir.

### 3.2.2. Çerçevelenen Konuşma İşaretlerinin Seslilik Sınıflandırılması

Konuşma işaretlerinde belirli özelliklerin incelenebilmesi için konuşma belirli zaman aralıklarına bölünür, konuşmanın bu aralıklarda incelenmesine kısa zaman analizi (short time analyse) denir. Bu işlem denklem (3.1) ve (3.2) de gösterilen dündörtgen pencere (rectangular window), hamming penceresi (hamming window) gibi fonksiyonların, zaman domenindeki konvolüsyonu yani frekans domenindeki çapımdır.

\[ w[m] = 1, \quad 0 \leq m \leq N - 1 \begin{cases} \text{diger} & \end{cases} \]  \quad (3.1)

\[ w[m] = 0.54 - 0.46 \cos\left(\frac{2\pi m}{N-1}\right), \quad 0 \leq m \leq N - 1 \begin{cases} \text{diger} & \end{cases} \]  \quad (3.2)

**Şekil 3.4:** Sesli ve Sessiz Konuşma Örnekleri
Ayrık zamanlı bir işaretin enerjisi ve kısa zaman analizi ile elde edilen işaretin enerjisi, denklem (3.3) ve denklem (3.4) de verilmiştir [11].

\[ E = \sum_{m=\infty}^{\infty} x^2(m) \]  
(3.3)

\[ E = \sum_{m=\infty}^{\infty} x^2(m)h(n-m) = \sum_{m=n-N+1}^{\infty} x^2(m) \]  
(3.4)

Konuşma; seslilik açısından sesli (voiced), sessiz (unvoiced) ve suskun (silence) olmak üzere üç temel bölümden oluşur. LPC yönteminde analiz edilecek dalga sesli ve sessiz olmak üzere iki sınıfa ayrılır. Sınıflandırma işleminin kalitesi
sentezlenen konuşmanın kalitesini doğrudan etkiler. Sınıflandırma işleminde sinyalin enerjisi ve enerji dağılımı (Low band to Full band ratio), periyodikliği, tepeli yapıya sahip olması (peakiness), ilk LP tahmin katsayısı, sıfırdan geçme oranı (zero-crossing rate) incelenen başlıca faktörlerdir.

Şekil 3.5: Sesli ve sessiz sesler enerji dağılım grafikleri [12].


\[
\text{Enerji Dağılım Oranı} = \frac{\sum_{i=1}^{N} s_{vhc}(i)^2}{\sum_{i=1}^{N} s(i)^2}
\] (3.5)
Şekil 3.6: Konuşmada 1khz den küçük frekanslardaki enerjinin tüm konuşmadaki enerjiye oran dağılımı, eşik değer 0.85

Şekil 3.5’de verilen sesli ve sessiz konuşma örneklerinde de görüldüğü gibi sesli konuştımlarda bir periyodiklik vardır. Dolayısıyla dalga bu periyot aralıkları ile kendine benzerliği incelenerek sesli ya da sessiz olduğu ortaya çıkartılabilir.

Sesli konuşmalar belirli periyotlarla tekrarlanan darbelerden oluşur. Tepeli (peakiness) yapıya sahip olma özelliğinden dolayı konuşma denklem (3.6) da verilen denklem ile sınıflandırılabilir.

\[ PK = \sqrt{\frac{1}{N} \sum_{i=1}^{N} s(i)^2} \]

\[ PK = \frac{1}{N} \sum_{i=1}^{N} |s(i)| \quad (3.6) \]

Konuşmanın tepelilik özelliğine göre seslilik açısından sınıflandırılması, şekil 3.7’de gözlemlenebileceği gibi çok ayrıntılı bir metot değildir çünkü sessiz kısımlarda da belirli tepeler bulunabilir.
Şekil 3.7: Konuşmanın tepeli yapıya sahip olma özelliğine göre incelenmesi

Sesli konușma sinyallerinde örnekler bir yanındaki örneklerle çok benzerler, bu sayede bir yanındaki örnekle korelasyonları çok yüksektir. Sessiz konuşmalarda ise böyle bir durum söz konusu değildir, daha çok rastgele değerlerden oluşurlar. Bu özelliklerinden dolayı hesaplanan LP tahmin katsayıları; sesli konuşmalarda -1'e yakın bir değer alırken, sessiz konuşmalarda 1 civarında bir değer alır. İlk tahmin katsayısı birinci dereceden normalize otokorelasyon ile ifade edilebilir.

$$
\text{Normalize edilmiş ilk otokorelasyon katsayısı} = \frac{\sum_{i=1}^{N} s(i)s(i-1)}{\sum_{i=1}^{N} s(i)^2}
$$

(3.7)
Bu fark sesli sessiz ayrımının yapımında güvenli ve etkili bir ayrışım sağlar [13]. R.Raibner bir çalışmada dört farklı insana ait konuşmaları inceleyerek Şekil 3.10’da verilen grafiği elde etmiştir.

Şekil 3.9: İlk LPC katsayısının teorik ve ölçülen yoğunluk fonksiyonları [12].

Sıfırdan geçme oranı (zero-crossing rate), konuşma sinyalinin sıfırdan geçme miktarının hızını ifade eder. Sesli konuşmalar daha düşük frekanslarda gerçekleştiği için sessiz konuşmalara nazaran sıfırdan çok daha az geçer. Bu fark bize sesli/sessiz sınıflandırılmasında önemli bir imkan sağlar. R.Raibner bir çalışmada dört farklı insana ait konuşmaları inceleyerek Şekil 3.11’de verilen grafiği elde etmiştir.
Şekil 3.10: Sesli ve sessiz bölgelerin sıfırdan geçme oran yoğunluk fonksiyonlarının teorik ve pratik grafikleri [12].

Sesli konuşmalarında sıfırdan geçme hızının 3 khz in altında olması beklenir [12]. Örnek bir konuşmanın sıfırdan geçme oranına göre sınıflandırılması Şekil 3.12’de gösterilmiştir.

Şekil 3.11: Konuşmanın sıfırdan geçme oranının 3khz eşik değeri ile sınıflandırılması

Sesli ve sessiz konuşmaların ayrımları için kullanılabilecek bazı özelliklerden yukarıda bahsedilmiştir. Sınıflandırılma probleminde incelenen özellikler ve bu özelliklerin karar verme etkileme yoğunlukları hala araştırılan konular arasındadır. Karar verme amacıyla belirli yoğunluk katsaylarının lineer olarak belirlenmesinden,
örnek tanıma (pattern recognition) algoritmalarının kullanılmasına kadar farklı yöntemler önerilmiştir.

3.2.3. LPC Sentez Filtresinin Modellenmesi

Vokal sistem, konuşmanın sınırlı bir aralığında, tüm-kutuplu filtre modelinin (all-pole filter model; autoregressive model, AR) tasarlanması için tahmin katsayları hesaplanarak modellenir. LP analizindeki temel fikir; konuşmanın n anındaki s[n] örneğinin, daha önceki çıkışların lineer kombinasyonu olarak ifade edilebilmesidir.

\[ s[n] = a_1 s[n-1] + a_2 s[n-2] + \ldots + a_p s[n-p] \]

(3.8)

\[ s[n] = \sum_{k=1}^{p} a_k s[n-k] + Gu[n] \]

(3.9)

\[ s[z] = \sum_{k=1}^{p} a_k z^{-k} S[z] + Gu[z] \]

(3.10)

\[ s[z](1 - \sum_{k=1}^{p} a_k z^{-k}) = Gu[z] \]

(3.11)

LPC sentez filtresinin transfer fonksiyonu;

\[ H(z) = \frac{s(z)}{u(z)} = \frac{G}{1 - \sum_{k=1}^{p} a_k z^{-k}} = \frac{1}{A(z)} \]

(3.12)
3.2.4. LPC Analiz Filtresinin Modellenmesi

Konuşmaya ait örneklerin tahmin edilen \( \tilde{s}[n] \) değerleri ile gerçek \( s[n] \) değerleri arasındaki fark hatayı verir.

\[
e[n] = s[n] - \tilde{s}[n] = s[n] - \sum_{k=1}^{p} \alpha_k s[n-k]
\]  
(3.13)

Hata işaret z domeninde incelenirse;

\[
E(z) = S(z) - S(z) \sum_{k=1}^{p} \alpha_k z^{-k} = S(z) \left[ 1 - \sum_{k=1}^{p} \alpha_k z^{-k} \right] = S(z) A(z)
\]  
(3.14)

Denklem (3.14) indirgendiğinde \( A(z) \) hata tahmin filtresi bulunur:

\[
A(z) = 1 - \sum_{k=1}^{p} \alpha_k z^{-k} = 1 - P(z)
\]  
(3.15)

\( A(z) \) filtresinin \( \alpha_k \) kat sayıları, \( s[n] \) ifade eden \( a_k \) kat sayılarına eşit olursa Şekil 3.14 de gösterilen LPC analiz filtresi elde edilir.

![Şekil 3.13: LP analiz filtre modeli](image)

Sistemin ortalama karesel hatası;

\[
E = \sum_{m=-\infty}^{\infty} e^2[m] = \sum_{m=-\infty}^{\infty} (s[m] - \tilde{s}[m])^2 = \left( s_n[m] - \sum_{k=1}^{p} \alpha_k s_n[m-k] \right)^2
\]  
(3.16)

Ortalama karesel hatayı minimum yapan katsayılar, tahmin katsayıları (prediction coefficients) denir. Hatayı minimum yapan \( \alpha_k \) katsayılarını bulmak için ortalama karesel hatanın bütün katsayılar göre kısımı türevleri sıfıra eşitlenerek katsayı adedi kadar denklem elde edilir. Bulunan denklemler çözülen LP katsayıları bulunur.
Denklem (3.16) de verilen ortalama karesel hatanın $\alpha_k$ katsayılarına göre türevleri alırsısa:

$$\frac{\partial E_n}{\partial \alpha_i} = 0, \quad i = 1,2,3,...,p$$

(3.17)

$$\frac{\partial E_n}{\partial \alpha_i} = \frac{\partial}{\partial \alpha_i} \left( \sum_m \left( s_n[m] - \sum_{k=1}^p \alpha_k s_n[m-k] \right)^2 \right)$$

(3.18)

$$\frac{\partial E_n}{\partial \alpha_i} = \sum_m \frac{\partial}{\partial \alpha_i} \left( s_n[m] - \sum_{k=1}^p \alpha_k s_n[m-k] \right)^2$$

(3.19)

$$\frac{\partial E_n}{\partial \alpha_i} = \sum_m 2 \left( s_n[m] - \sum_{k=1}^p \alpha_k s_n[m-k] \right) \left( -\frac{\partial}{\partial \alpha_i} \sum_{k=1}^p \alpha_k s_n[m-k] \right)$$

(3.20)

$$\frac{\partial E_n}{\partial \alpha_i} = \sum_m 2 \left( s_n[m] - \sum_{k=1}^p \alpha_k s_n[m-k] \right) (-s_n[m-i])$$

(3.21)

$$0 = \sum_m 2 \left( -s_n[m]s_n[m-i] + \sum_{k=1}^p \alpha_k s_n[m-k]s_n[m-i] \right)$$

(3.22)

bir yöntemdir. Donanım tasarlanmasında istenen bu sisteme otokorelasyon yönteminin ve levinson-durbin algoritmasının kullanılması tercih edildi.

3.2.4.1. Korelasyon Yöntemiyle Analiz Filtre Denklemlerinin Elde Edilmesi

Denklem (3.22) de verilen eşitlik düzenlenirse denklem (3.23) de verilen eşitlik elde edilir.

\[ \sum_{m} s_{n}[m-i]s_{n}[m] = p \sum_{k=1}^{p} \alpha_{k} \sum_{m} s_{n}[m-i]s_{n}[m-k] \quad 1 \leq i \leq p \]  

(3.23)

Denklem (3.24)’deki gibi bir \( \Phi \) fonksiyonu tanımlanıp denklem (3.23)’e yerleştirildiğinde, daha sade bir gösterim olan denklem (3.25) elde edilir.

\[ \Phi[i,k] = \sum_{m} s_{n}[m-i]s_{n}[m-k] \]  

(3.24)

\[ \sum_{k=1}^{p} \alpha_{k} \Phi[i,k] = \Phi[i,0] \quad i = 1,2,3,\ldots,p \]  

(3.25)

Ortalama karesel hatayı ifade eden denklem (3.16) açılırsa, denklem (3.26) elde edilir.

\[ E_{n} = \sum_{m} \left( s_{n}[m] - \sum_{k=1}^{p} \alpha_{k}s_{n}[m-k] \right)^{2} \]  

\[ E_{n} = \sum_{m} s_{n}^{2}[m] - 2 \sum_{m=\infty}^{\infty} s_{n}[m] \sum_{k=1}^{p} \alpha_{k}s_{n}[m-k] + \sum_{m=\infty}^{\infty} \sum_{k=1}^{p} \alpha_{k}s_{n}[m-k] \sum_{l=1}^{p} \alpha_{l}s_{n}[m-l] \]  

(3.26)

Denklem (3.26) deki son terim, denklem (3.27) deki gibi düzenlenir ve denklem (3.24)’den faydalanılarak denklem (3.28) elde edilir.

\[ \sum_{m} s_{n}^{2}[m] - 2 \sum_{k=1}^{p} \alpha_{k} \sum_{m=\infty}^{\infty} s_{n}[m-k]s_{n}[m-l] + \sum_{m=\infty}^{\infty} \sum_{l=1}^{p} \alpha_{l}s_{n}[m-l]s_{n}[m] \]  

(3.27)

\[ E_{n} = \sum_{m=\infty}^{\infty} s_{n}^{2}[m] - 2 \sum_{k=1}^{p} \alpha_{k} \sum_{m=\infty}^{\infty} s_{n}[m-k]s_{n}[m] + \sum_{l=1}^{p} \alpha_{l} \sum_{m=\infty}^{\infty} s_{n}[m-l]s_{n}[m] \]  

\[ = \sum_{m=\infty}^{\infty} s_{n}^{2}[m] - \sum_{k=1}^{p} \alpha_{k} \sum_{m=\infty}^{\infty} s_{n}[m-k]s_{n}[m] \]  

(3.28)
Denklem (3.25)’deki gösterim denklem (3.28) üzerinde uygulandığında ortalama karesel hata değeri korelasyon cinsinden bulunur.

\[ E_n = \Phi_n[0,0] - \sum_{k=1}^{p} \alpha_k \Phi_n[0,k] \]  

(3.29)

**3.2.4.2. Otokorelasyon Yöntemiyle Analiz Filtre Denklemlerinin Elde Edilmesi**

Otokorelasyon yönteminde seçilen \( N \) genişliğindeki zaman aralığının dışındaki örneklerin sıfır olduğunu düşünlürlür ve ortalama karesel hata \( \pm \infty \) aralığında hesaplanır. Bu işlem, \( 0 \leq m \leq N-1 \) olmak üzere \( s[n] \) konuàma dizisinin denklem (3.30)’da verilen \( N \) genişliğindeki \( w[m] \) dikdörtgen (rectangular) pencere ile pencerelenmiş işaretin sıfıra eşit olmayan ilk elemanının \( m=0 \) olacak şekilde kaydırılmasını.

\[ s_n[m] = \begin{cases} s[m+n]w[m], & 0 \leq m \leq N-1 \\ 0 & \text{diğer} \end{cases} \]  

(3.30)

Denklem (3.24)’de gösterilen korelasyon gösterimi, düzenlenerek otokorelasyon denklemleri elde edilmiştir.

\[ \Phi_n[i,k] = \sum_{m=0}^{N-1-(i-k)} s_n[m]s_n[m+(i-k)], \quad 1 \leq i \leq p, \quad 1 \leq k \leq p \]  

(3.31)

Denklem (3.31)’deki \( \Phi_n \) fonksiyonu sadece değişkenlerin farkı olduğu için bu eşitlik tek değişken ile gösterilebilir. Ayrıca otokorelasyon çift bir fonksiyondur.

\[ r_n[\tau] = r_n[|i-k|] = \Phi_n[i,k] \]  

(3.31)

\( P \) tahmin derecesi olmak üzere, denklem (3.31) ve denklem (3.24) birleştirildiğinde otokorelasyon özelisi katsayıları bulunur.

\[ r_n[i] = \sum_{m=1}^{N-1-i} s_n[m]s_n[m+i] \]  

(3.32)

\( Z \) domeninde yapılan otokorelasyon işlemi, zaman domeninde konvolüsyon işlemine denk gelir ve sinyalin kendine benzerliğinin bir ölçüsüdür. Otokorelasyon sonucunun \( i=0 \) andındaki çıktısı dalganın enerjisini verir ve otokorelasyonun alabileceği en büyük değeri.
\[ s_n[m] \text{ dalgasinin enerjisi } = r_n[0] = \sum_{m=-\infty}^{\infty} |s_n[m]|^2 e^{j\omega} \]  
\hspace{2cm} (3.33)

P tahmin derecesi, \( \alpha \) tahmin katsayıları olduğuna göre denklem (3.25) ve denklem (3.31) kullanılarak denklem (3.34) de verilen eşitlikler elde edilir.

\[ \sum_{k=1}^{p} \alpha_k r_n[i-k] = r_n[i], \quad 1 \leq k \leq p \]  
\hspace{2cm} (3.34)

Denklem (3.34) matris biçiminde gösterildiğinde denklem (3.35)'de verilen toeplitz otokorelasyon matrisi elde edilir. Levinson-Durbin algoritmasıyla bu matrisin çözümü yapıldığında LP tahmin katsayıları bulunur.

\[
\begin{bmatrix}
    r_n[0] & r_n[1] & r_n[2] & \cdots & r_n[p-1] \\
    r_n[1] & r_n[0] & r_n[1] & \cdots & r_n[p-2] \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    r_n[p-1] & r_n[p-2] & r_n[p-3] & \cdots & r_n[0] \\
\end{bmatrix}
\begin{bmatrix}
    \alpha_1 \\
    \alpha_2 \\
    \alpha_3 \\
    \vdots \\
    \alpha_p \\
\end{bmatrix}
=
\begin{bmatrix}
    r_n[1] \\
    r_n[2] \\
    r_n[3] \\
    \vdots \\
    r_n[p] \\
\end{bmatrix}
\hspace{2cm} (3.35)
\]

Denklem (3.29) ve denklem (3.31) kullanılarak ortalama karesel hatanın otokorelasyon sonuçları ile gösterimi bulunur.

\[ E_n = r_n[0] - \sum_{k=0}^{p} \alpha_k r_n[k] \]  
\hspace{2cm} (3.36)

Otokorelasyon yönteminde, hata sadece pencerelelen aralıktı sıfır değildir. Pencerelelen aralığın dışındaki değerler sıfır olduğu için pencerenin en sağında ve en solunda, hata en büyük değerini alır ve buna kenar etkisi (edge effect) denir. Ortalama karesel hatanın, kenar hataları tarafından belirlenesinden kaçınınmak için sinyal, otokorelasyon yöntemiyle incelenmeden önce denklem (3.2)'de verilen hamming penceresi ile pencerelelen.

### 3.2.5. Levinson-Durbin Algoritması

Levinson-Durbin algoritması, toeplitz matrislerin çözümünde kullanılan bilindik en popüler ve en etkili yöntemdir [11]. Çözüm, otokorelasyon matrisinin tersi alınarak \( p^3 \) tane çarpma ve toplama içeren işlemlerle yapılarırken, levinson algoritması sayesinde sonuç \( p^2 \) işleme bulunabilir. Bu algoritma kullanılarak denklem (3.35) de verilen...
otokorelasyon matrisi çözülerek $a_1, a_2, ..., a_{10}$ vokal sistem filtre katsayıları bulunabilir. Levinson-Durbin algoritması aşağıdaki verilen dört adımın yinelenmesinden ve beşinci adından oluşmaktadır:

**Adım 1 (Başlangıç durumu):**

$$E(0) = R(0)$$  \hspace{1cm} (3.37)

**Adım 2:**

$$k_i = \frac{R(i) - \sum_{j=1}^{i-1} a_j^{(i-1)}R(i-j)}{E^{(i-1)}}, \quad 1 \leq i \leq p$$  \hspace{1cm} (3.38)

**Adım 3:**

$$a_i = k_i$$  \hspace{1cm} (3.39)

$$a_j^i = a_j^{(i-1)} - k_i a_{i-j}^{(i-1)}, \quad 1 \leq j \leq i - 1$$  \hspace{1cm} (3.40)

**Adım 4:**

$$E^i = (1 - k_i^2)E^{(i-1)}$$  \hspace{1cm} (3.41)

**Adım 5:**

(2,3,4). adımlar $i=1,2,...,p$ için çözüldükten sonra denklem (3.42) de verilen işlem yapılırak sonuç bulunur.

$$a_i = a_j^p$$  \hspace{1cm} (3.42)

Levinson-Durbin algoritmasında $E^i$ parametreleri, i. yineleme için öngörü hatasını vermektedir. Otokorelasyon matrisindeki $R(i)$ değişkenleri yerine denklem (3.43) de verilen normalize otokorelasyon katsayıları kullanılarak algoritma sonucu elde edilen $a_i$ katsayıları değişmez ve denklem (3.44) de verilen normalize öngörü hatası elde edilebilir [11].

$$r(k) = R(k)/R(0)$$  \hspace{1cm} (3.43)
\[ V^{(i)} = \frac{E^{(i)}}{R(0)} = 1 - \sum_{i=1}^{i} a_i r(k), \quad 0 < V^{(i)} \leq 1 \text{ ve } 0 \leq i \] (3.44)

Denklem 3.44, denklem 3.38 de elde edilen \( k_i \) katsayıları ile gösterilebilir ve \( k_i \) katsayıları denklem (3.45)’de verilen aralığa bulunduğuda hesaplanan LP katsayıları ile tasarlanan filtre kararlı olur [11].

\[ V^{(p)} = \prod_{i=1}^{p} (1 - k_i^2), \quad -1 \leq k_i \leq 1 \] (3.45)

Şekil 3.5 de verilen sesli konuşma dosyasının onuncu dereceden (p=10) analizi ile bulunan sentez ve analiz filtrelerinin frekans cevabı şekil 3.19 da bulunmaktadır.

**Şekil 3.14:** Konuşma aralığına ait LPC sentez ve analiz filtreleri frekans yanıtları

Konusmanın enerji dağılım grafiği ve sentez filtresinin frekans cevabı fonksiyonlarının spektral eğilimlerinin (envelope) uyuştuğu, şekil 3.16 da görülmektedir.
Şekil 3.15: Konuşma aralığına ait LPC sentez ve analiz filtreleri frekans yanıtları

Analiz filtresi alçak geçiren bir filtre tipinde olması sebebiyle, analizfiltresinden geçirilen konuşma işaretinin sadece yüksek frekanslı bileşenleri kalır. Elde edilen hata sinyali şekil 3.17 de görülmektedir.

Şekil 3.16: Vokal filtre çıktısında görülen hata sinyali

Sentez filtresine hata işaret girisi olarak verilirse, konuşma sinyalinin kendisi geri elde edilir.

Sentez filtresinin z domeninde sıfır-kutup dağılımları incelendiğinde filtrenin kararlı bir yapıya sahip olduğu görülür.
3.2.6. Kazanç Hesabı

Sentezlenen ses ile analiz edilen sesin aynı enerjiye sahip olması için filtrenin girişine verilen sinyallerin belirli bir G kazancıyla çarpılması gerekir. (3.9) ve (3.13) denklemleri incelediğinde kazancın, hata sinyaline göre belirlenebileceği görülmektedir. Bu eşitlik denklem (3.46)’da verilmiştir.

\[ G_u[n] = s[n] - \sum_{k=1}^{p} q_k s[n-k] = e(n) \]  

(3.46)

\[ G^2 E_r r = \sum_{k=1}^{p} \alpha_k \beta_k [k] \]  

(3.47)

Şekil 3.17: LPC sentezfiltresinin sıfır-kutup dağılımı

3.2.6. Kazanç Hesabı

Sentezlenen ses ile analiz edilen sesin aynı enerjiye sahip olması için filtrenin girişine verilen sinyallerin belirli bir G kazancıyla çarpılması gerekir. (3.9) ve (3.13) denklemleri incelediğinde kazancın, hata sinyaline göre belirlenebileceği görülmektedir. Bu eşitlik denklem (3.46)’da verilmiştir.

\[ G_u[n] = s[n] - \sum_{k=1}^{p} q_k s[n-k] = e(n) \]  

(3.46)

\[ G^2 E_r r = \sum_{k=1}^{p} \alpha_k \beta_k [k] \]  

(3.47)

Şekil 3.20 de görülen -konuşma işaret, sentezleme filtre cevabı- grafiğinde filtre yanıtının, y ekseninde işaret ile aynı seviyeye gelmesi (dc ofset değerinin belirlenmesi) için filtre frekans yanıtının logaritması normalize hata enerjisinin kökü alınarak hesaplanan kazanç ile çarpılmıştır.
3.2.7. Enerjinin Normalize Değer Gösterimi

Normalize edilmiş ortalama karesel hatanın otokorelasyon yöntemine göre gösterimi denklem (3.48) deki gibi korelasyon gösterime çevirilebilir.

\[
V_n = \frac{\sum_{m=0}^{N-1} e_n(m)^2 - e_n(0)^2}{\sum_{m=0}^{N-1} s_n(m)^2 - s_n(0)^2}
\]  
(3.48)

Hata işaretli denklem (3.49)’da tanımlanmıştır.

\[
\alpha_i = -1 \text{ ise } e_n(m) = \sum_{k=0}^{p} \alpha_i s_n(m-k)
\]  
(3.49)

Denklem (3.49) ve denklem (3.48) birleştirilir ve denklem (3.24) deki gösterime göre denklemler düzenlenirse denklem (3.50) elde edilir.

\[
V_n = \sum_{i=0}^{p} \sum_{j=0}^{p} \alpha_i \frac{\phi_{i,j}}{\phi_{0,0}} \alpha_j
\]  
(3.50)

Denklem (3.25) ile denklem (3.50) birleştirildiğinde ve denklem (3.31) ‘e göre düzenlendiğinde denklem (3.51)’de verilen, hata enerjisinin normalize değerinin otokorelasyon ile ifadesi elde edilir.

\[
V_n = -\sum_{i=0}^{p} \alpha_i \frac{\phi_{i,0}}{\phi_{0,0}} = -\sum_{i=0}^{p} \alpha_i r_n(i)
\]  
(3.51)

Hata sinyali için tanımlanan normalize enerji için başka bir gösterim de denklem (3.45) de verilmiştir.

3.2.8. Perdenin Tahmin Edilmesi

Sesli konuşmalardaki perde periyodunun belirlenmesi ortaya çıkan sesin kalitesinde çok önemli bir etkidir. Konuşmalardaki periyodikliğin mükemmel olmaması, konuşmadaki sesli kısmın başlangıcının tam olarak belirlenememesi ve gerçek hayattaki konuşmalara güriştünün ve yankının eklenmesi başlıca problemlerdir. Genellikle periyot tahmin hassaslığı arttıkça hesaplama yükü de artar.
Otokorelasyon, sinyalin kendine benzerliğinin bir ölçüsü olduğundan periyot tahmini için bu yöntemden yararlanılabilir. Periyodik bir sinyalde gecikme artırılarak otokorelasyon katsayıları hesaplanır. Otokorelasyon değerinin en yüksek olduğu gecikme, periyot bilgisini verir. Şekil 3.19'da periyot değeri seksen gecikmeye (100 hz lik frekansa sahip, 8 khzde örnekleme hızında) eşit olan bir cosinüs işaretinin gecikme miktarına göre otokorelasyon grafiği verilmiştir.

Şekil 3.18: İşaretin otokorelasyon değerinin gecikme miktarına göre değişimi

Her bir gecikme değeri için otokorelasyon işlemi yapmak, ciddi bir işlem yükü doğuracağından perde tespiti için denklem (3.52)'de verilen genlik fark fonksiyonu (AMDF, Average Magnitude Difference Function) tercih edilir. Genlik fark fonksiyonu çarpma işlemi gerektirmemesinden özellikle donanım tasarımına çok daha uygun ancak performans açısından iyi değildir.

\[
AMDF[l] = \sum_{n=0}^{N-1} |S[n] - S[n - l]| \quad (3.52)
\]

İnsan sesi 50 Hz ile 200 Hz arasındadır, dolayısıyla 8 khz ile örneklenmiş bir konuşma işaret perde periyodu tahmini için incelenirken otokorelasyon ve genlik fark fonksiyonları gecikme miktarının (lag) 20 ve 150 arasındaki değerleri için kontrol edilerek işlem yükü azaltılabilir. Eğer periyodiklik bu değerlerin dışında ise incelenen sesin sessiz olduğuna karar verilebilir.

3.3. LPC Modelinin Matlab Ortamında Tasarlanması

Bu çalışmada LPC analiz ve sentez modeli matlab ortamında tasarlanmıştır ve örnek bir konuşmanın analizi ve sentezi yapılmıştır. Yapılan işler maddeler halinde aşağıda verilmiştir:

**Analiz Kısımında:**

- Konuşma 8 kHz de darbe kodlayıcısı (PCM) ile örneklendi.
- 10ms üst üste örtüsecek (overlap) şekilde 20 ms lik çerçevelere ayrılarak sırasıyla aşağıdaki işlemlerden geçildi.
- Hamming penceresi ile pencereleendi.
- Seslilik detektörü (VAD, voice activity detector) sayesinde incelenen çerçeve sesli veya sessiz olarak ayrıldı.
- Otokorelasyon katsayıları oluşturuldu.
- Levinson-Durbin algoritmasıyla LPC tahmin katsayıları hesaplandı.
- Kazanç hesaplandı.
- Sesli çerçeveler için periyot bilgisi bulundu.

**Sentez Kısımında:**

- Konuşma seslilik bilgisi incelenerek 10 ms uzunluğunda uyarı (excitation) sinyali oluşturuldu:
  - Sessiz ise beyaz gürültü,
  - Sesli ise konuşmanın periyot bilgisine bakılarak darbe dizisi oluşturuldu.
- Oluşturulan çerçeve kazanç ile çarpıldı.
- Elde edilen sinyal, LPC tahmin katsayıları ile tasarlanan 1/A(z) sentezleme filtresine giris olarak verilerek sentetik ses oluşturuldu.

Şekil 3.20 ve Şekil 3.23 de analiz edilen ve sentezlenen sesli konuşma çerçevelerinin ve enerji yoğunluk fonksiyon grafipleri bulunmaktadır.
Şekil 3.21: Sesli konuşmanın sentezlenmesi için filtrene girisine verilecek uyarı sinyali ve enerji yoğunluk fonksiyonu grafiği bulunmaktadır. Uyarı sinyalinin hata sinyaline benzemediği açıkça görülmektedir ancak bu iki sinyalin enerji spektrumlarının sınırlarında benzerlik vardır.

Şekil 3.20: Analiz edilen ses ile aynı periyodikliğe sahip kazanç ile çarpmış darbe dizisi

Sentezlenen sesli konuşma

Şekil 3.21: Sentezlenen sesli konuşma

Şekil 3.22: Analizi yapılan sessiz konuşma ve ortalama enerji yoğunluk fonksiyon grafiği

Analizi yapılan ve sentezlenen sessiz konuşmalar karşılaştırıldığında hiçbir ortak örneklerinin olmadığı görülmektedir. Bunun yanında dalgaların spektral eğilimleri birbirine çok yakınır.
Şekil 3.23: Analizi yapılan sessiz konuşma ve ortalama enerji yoğunluk fonksiyon grafiği

Şekil 3.29 de ve şekil 3.30 da LP yöntemiyle incelenen ve sentezlenen konuşmalar verilmiştir.

Şekil 3.24: Analizi yapılan Konuşma
Sıkıştırma işlemi sonucunda elde edilen konuşmanın orijinal konuşmaya göre daha seyrek, ortalama olarak daha düşük genlikli olduğu ancak genliğinin yüksek inişlere ve çıkışlara sahip olduğu bölgelerde daha yüksek genlikler elde edildiği gözlemlendi.

**Şekil 3.25:** Sentezlenen Konuşma

4. GELİŞMİŞ ŞİFRELEME STANDARDI


Bir algoritmada tekrarlanarak yürütülen işlemlerin oluşturduğu yapıya tur denir. AES-128 bit şifreleme işlemi 10 turdan oluşur. Algoritma şifreleme ve şifre çözme olmak üzere iki temel bloktan oluşur. Her turda şifre anahtar üreticisi tarafından değiştirilir. Şifreleme ve şifre çözme blokları; bayt değiştirme (Sub Bytes), satırları kaydırma (Shift Rows), sütun karıştırma (Mix Columns), tur anahtarı eklemesi (Add Round Key) işlemleri ve ters işlemleri yapan dört alt bloktan oluşmaktadır [15].

İlk anahtar toplamasında, algoritmaya girilen veri ve ana anahtarın karşılıklı bitleri arasında özel veya (XOR) işlemi uygulanır ve 16 baytta oluşan ilk durum elde edilir ve bu durum ilk turda işlenir. İlk tur çıktısı, ikincisi turun girişini oluşturur. Bu şekilde N-1 kez tur’dar işlenen veri son olarak final turunda işlenir ve algoritma çıktısı elde edilir [14]. Şekil 4.1 de bu işlem gösterilmiştir.
Şekil 3.3 : AES şifreleyici blok diyagramı [14].

Şifreleme işlemi için fips-197 standardında verilen sözde kod (Pseudo Code) şekil 4.2 de verilmiştir [15].

```plaintext
Cipher(byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])
begin
  byte state[4*Nb]

  state = in
  AddRoundKey(state, w[0, Nb-1])  // See Sec. 5.1.4
  for round = 1 step 1 to Nr-1
    SubBytes(state)  // See Sec. 5.1.1
    ShiftRows(state)  // See Sec. 5.1.2
    MixColumns(state)  // See Sec. 5.1.3
    AddRoundKey(state, w[(round*Nb, (round+1)*Nb-1)]
  end for

  SubBytes(state)
  ShiftRows(state)
  AddRoundKey(state, w[Nr*Nb, (Nr+1)*Nb-1])

  out = state
end
```

Şekil 3.3 : AES şifreleyici sözde kodu [15].

Şifreleyici donanım şematik gösterimi şekil 4.3 de verilmiştir.
Şekil 3.3 : AES şifreleyici donanım şematik gösterimi.

Şifre çözme işlemi için fips-197 standardında verilen sözde kod (Pseudo Code) şekil 4.4 de verilmiştir [15].

```
InvCipher(byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])
begin
    byte state[4,Nb]
    state = in
    AddRoundKey(state, w[Nr*Nb, (Nr+1)*Nb-1]) // See Sec. 5.1.4
    for round = Nr-1 step -1 downto 1
        InvShiftRows(state) // See Sec. 5.3.1
        InvSubBytes(state) // See Sec. 5.3.2
        AddRoundKey(state, w[round*Nb, (round+1)*Nb-1])
        InvMixColumns(state) // See Sec. 5.3.3
    end for
    InvShiftRows(state)
    InvSubBytes(state)
    AddRoundKey(state, w[0, Nb-1])
    out = state
end
```

Şekil 3.3 : AES şifre çözücü sözde kodu [15].

Şifre çözücü donanım şematik gösterimi şekil 4.5’de verilmiştir.
Şekil 3.3 : AES şifre çözücü donanım şematik gösterimi.

Şifreleme ve şifre çözme işlemleri için tasarlanması gereken alt bloklar aşağıda açıklanmıştır.

4.1. Bayt Değiştirme (Sub Bytes)

Bayt değiştirme işlemi sırasında 128 bitlik veri 8 bitlik 16 parça bölünür ve şekil 4.6’da gösterilen 4x4 boylu matris oluşturmaktadır. Girişteki durumun her bir bayt S-kutusu (S-Box) adı verilen değiştirme tablosu kullanılarak farklı baytlara dönüştürülür. Şekil 4.4’de gösterilen bayt değiştirme işlemi, matematiksel olarak giriş baytının indirgeme polinomu kullanılarak çarpma göre tersinin alınması işlemi ve elde edilen sonucun geçiş matrisi olarak adlandırılan bir matrisle çarpılmasıyla bayt değiştirme işleminin yapılmasını sağlar.

şekil 4.1: Durum Matrisi [15].

Ters bayt değiştirme işlemi için aynı model ters S-kutusu kullanılarak yapılabilir.
4.2. Satırları Kaydırma (Shift Rows)

Satırları kaydırma işleminde, durum matrisinin ilk satırı sabit kalır, ikinci satıri bir, üçüncü satıri iki, dördüncü satıri ise üç kere sola kaydırılır. Şekil 4.7’de verilen satır kaydırma işlemi sonucunda elde edilen durum matrisindeki elemanlar birleştirilerek 128 bitlik veri elde edilir.

![ShiftRows Diagram](image)

**Şekil 4.1:** Satırları Kaydırma İşlemi [15].

Satırları kaydırma işleminin tersi şekilde 4.8’de görüldüğü gibi, yukarıda yapılan işlemlerin tersinin yapılmasıdır.

![InvShiftRows Diagram](image)

**Şekil 4.1:** Satırları Kaydırma İşleminin Tersi [15].

4.3. Sütunları Karşıtırma (Mix Columns)

Sütunları karşıtırma işlemiyle giriş durum matrisinin her sütunu şekil 4.9’da gösterilen işlemden geçirilerek, çıkış durum matrisi elde edilir. Her bir sütun
denklem (4.1) de verilen A(x) polinomu ile modülo \( (x^4 + 1) \)’de çarpma işlemi gerçekleştirilir.

\[
A(x) = (03)x^3 + (01)x^2 + (02)
\]

(4.1)

\[\begin{array}{cccc}
02 & 03 & 01 & 01 \\
01 & 02 & 03 & 01 \\
01 & 01 & 02 & 03 \\
03 & 01 & 01 & 02 \\
\end{array}\]

\[\begin{array}{cccc}
\overline{02} & \overline{03} & \overline{01} & \overline{01} \\
\overline{01} & \overline{02} & \overline{03} & \overline{01} \\
\overline{01} & \overline{01} & \overline{02} & \overline{03} \\
\overline{03} & \overline{01} & \overline{01} & \overline{02} \\
\end{array}\times
\]

**Şekil 4.1:** Sütunları Kaydırma İşlemi [14].

Şifrenin çözülmesi sırasında aynı işlem ters denklem için uygulanır.

\[\begin{array}{cccc}
0E & 0B & 0D & 09 \\
09 & 0E & 0B & 0D \\
0D & 09 & 0E & 0B \\
0B & 0D & 09 & 0E \\
\end{array}\]

\[\begin{array}{cccc}
\overline{0E} & \overline{0B} & \overline{0D} & \overline{09} \\
\overline{09} & \overline{0E} & \overline{0B} & \overline{0D} \\
\overline{0D} & \overline{09} & \overline{0E} & \overline{0B} \\
\overline{0B} & \overline{0D} & \overline{09} & \overline{0E} \\
\end{array}\]

**Şekil 4.1:** Sütunları Kaydırma İşleminin Tersi [14].

4.4. **Tur Anahtarı Ekleme**

Durum matrisinin baytları ile tur anahtarında karşılık düşen baytlar arasında özel veya (XOR) işlemi yapılır. Şifreleme ve şifre çözme işlemleri için bu așama aynıdır.
4.5. Anahtar Üreteci

Anahtar üretici, her turun son admında üretilen ara değer ile toplanacak olan tur anahtarlarını üretektedir. Şifreleme veya şifre çözme esnasında tur sayısı kadar tur anahtarı üretilir. 128 bit uzunlukündeki tur anahtarlarının şifreleme işlemi için üretilmesi şekil 4.12’de verilen algoritma ile sağlanır.

Şekil 4.1: Şifreleme işlemi için anahtar üretici [14].
Şifre çözme işlemi için, şekil 4.13’de verilen algoritma ile anahtar üretici tasarlanır.

Şekil 4.1: Şifre çözme işlemi için anahtar üretici [14].

4.6. Gelişmiş Şifreleme Standardının FPGA ile Gerçeklenmesi

Gerçeklenen şifreleme ve şifre çözme donanımlarının sentez raporu şekil 4.14’de ve şekil 4.15’de verilmiştir. Ayrıca donanımların zamanlama rapor sonuçları tablo 4.1 de verilmiştir.

<table>
<thead>
<tr>
<th>Donanım</th>
<th>Minimum Periyot</th>
<th>Maksimum Saat Frekansı</th>
</tr>
</thead>
<tbody>
<tr>
<td>Şifreleme Donanımı</td>
<td>10.515ns</td>
<td>95.098MHz</td>
</tr>
<tr>
<td>Şifre Çözme Donanımı</td>
<td>10.389ns</td>
<td>96.256MHz</td>
</tr>
</tbody>
</table>

Tablo 4.1: Donanımların Zamanlama Rapor Sonuçları
Şekil 4.1: AES Şifreleme Donanımının Sentez Raporu.

Şekil 4.1: AES Şifre Çözme Donanımının Sentez Raporu.

Gerçeklenen şifreleme ve şifre çözme donanımlarının benzetim sonuçları şekil 4.16’da ve şekil 4.17’de verilmiştir.
Şekil 4.1: AES Şifreleme Donanımının Benzetim Sonucu [14].
Şekil 4.1: AES Şifre Çözme Donanının Benzetim Sonucu [14].
5. KODLAYICI TASARIMI ICIN KONUSMANIN KODLANMASI

GSM kodlayıcında, konuşma modellerek iletildiği için iletilen konuşma gerçek konuşmadan farklı olur. Şekil 5.1 de görüldüğü gibi GSM kodlayıcı gerçek konuşmaya karakteristik olarak benzeyen ancak örnek değerleri açısından farklı bir konuşma sentezlemiştir.

Sayısal bir veriden konuşma sentezlenip, GSM kodlayıcıdan geçtikten sonra minimum hata ile sayısal verinin elde edilebilmesi için konuşma, dalga formu olarak değil model parametreleri tabanlı bir yöntem ile sentezlenmelidir. Konuşmaya ait en temel özellikleri ifade eden parametreler LPC filtre katsayıları, enerji ve perde periyodudur.

Güvenli ses iletiminin başarlabilmesi için sayısal verilere karşılık düşen konuşmaların kodlayıcı tarafından sıkıştırılduktan sonra yeniden tanınamılmasi için kabul edilebilir bir hata oranının üzerinde değişmemesi gerekir. Bu tip konuşmalara elde edilebilmesi için GSM kodlayıcının yapısı araştırıldı ve uygun konuşmalara ait özellikler belirlendi. NTIMIT veritabanı içerisindeki konuşmalar taramıp istenen

Şekil 5.27: GSM kodlayıcıya gönderilen ve alınan konuşmalar
koşulları sağlayan çerçeveler seçildi ve bu konuşmalara ait LPC parametreleri elde edildi. Sistemin hatasız çalışabilmesi için bulunan parametreler LBG algoritması ile sınıflara ayrıldı, sayısal verilerden konuşmaların kodlanması için imkan verecek kod kitapları oluşturuldu. Bu kod kitapları sayesinde kodlayıcının girişi ve çıktısı konuşmalar farklı olsalar bile sistem tarafından doğru algılanacaktır.

5.1. GSM FR Kodlayıcısı

GSM-FR kodlayıcısı Avrupa Telekomünikasyon Standartları Enstitüsü (European Telecommunications Standards Institute, ETSI) tarafından mobil cep telefonu uygulamaları için standartlaştırılan 8 kHz örneklemesi frekanslı ve 13 kpbs bit oranına sahip bir konuşma sıkıştırma algoritmasıdır. Bu kodlayıcı CELP (Code Excited Linear Prediction) algoritmalarından biri olan RPE-LTP (Regular Pulse Excitation-Long Term Prediction-Linear Predictive coding) tabanlı bir sıkıştırma tekniğine sahiptir.

**Şekil 5.28:** Celp algoritmasının kodlayıcı kısmı [16].

**Şekil 5.29:** Celp algoritmasının kodlayıcı kısmı [16].

RPE-LTP kodlama yönteminde, CELP kodlama algoritmasındaki rastgele değerlerin bulunduğu kod kitabı düzenli aralıklara sahip darbe dizilerini barındırır. RPE-LTP kodlama yöntemi GSM kodlayıcida kullanıldığı haliley şekil 5.4 de verilmiştir.

**Şekil 5.30:** RPE-LTP, GSM konuşma kodlayıcısı [17].
5.2. Kod Kitaplarında Saklanacak Konuşmaların Belirlenmesi

GSM kodlayıcı incelenirse konuşmanın sessiz kısımları gürültü ile modellenirken, sesli kısımları analiz edilip elde edilen parametreler göre sentezlenir.

Şekil 5.31: GSM kodlayıcı ile kodlanmış konuşmanın sesli bir kısmı.

Şekil 5.32: GSM kodlayıcı ile kodlanmış konuşmanın sessiz bir kısmı.
GSM kodlayıcının yapısı dikkate alındığında ve örnek konuşmalar üzerine yapılan deneySEL incelemeler sonucunda edilen veriler değerlendirilerek, belirlenen model ile GSM üzerinden konuşma iletiminin sağlanması için sayısal verilerin sadece sesli seslerden oluşan konuşmalara çevrilmesi gerektiğini karar verildi.

20 ms uzunluğundaki sesli bir konuşmanın GSM kodlayıcı tarafından kodlanmasıyla elde edilen işaretin ve ilk halinin hamming penceresi ile pencerelendikten sonraki perde periyodu, enerjisi ve LP tahmin katsayları tablo 5.1 de verilmiştir.

<table>
<thead>
<tr>
<th>Konuşma</th>
<th>Perde Periyodu</th>
<th>Enerji</th>
<th>LP Tahmin Katsayları</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analiz Edilen Konuşma</td>
<td>59 örnek</td>
<td>0.8475</td>
<td>-2.1645 -0.6921 -0.6921</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1.3734 -0.2131 -0.6921</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-0.5692 0.1596 0.4616</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0.7886 0.4616 -0.2602</td>
</tr>
<tr>
<td></td>
<td>GSM ile Kodlanan Konuşma</td>
<td>59 örnek</td>
<td>0.6957</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1.3781 -0.2163 -0.2163</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-0.3141 0.4587 -0.0218</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0.5654 -0.0218 -0.0639</td>
</tr>
</tbody>
</table>

Tablo 5.1: GSM kodlayıcı ile analiz edilen ve sentezlenen konuşmaların özellikleri

GSM kodlayıcı ile kodlanan ve analiz edilen konuşmalar şekil 5.7’de, vokal filtre frekans cevapları şekil 5.8’de verilmektedir.
Şekil 5.33: GSM kodlayıcı ile analiz edilen ve sentezlenen konuşmalar

Şekil 5.34: GSM kodlayıcı ile analiz edilen ve sentezlenen konuşmaların filtre frekans yanıtları
LPC katsayıları kodlamaya daha uygun olan LSF (Line Spectrum Frequencies) katsayılarına çevrilecek ve vektörel kuantalama ile oluşturulacak kod kitaplarında saklanacaktır.

5.3. LPC Katsayılarının LSF Parametreleri ile Gösterimi

LP filtro katsayıları, denklem (5.1)’de ve denklem (5.2)’de verilen polinomlarla ifade edilebilir.

\[
P(z) = A(z) + z^{-(p+1)}A(z^{-1}) \tag{5.1}
\]

\[
Q(z) = A(z) - z^{-(p+1)}A(z^{-1}) \tag{5.2}
\]

P(z) ve Q(z) polinomlarının kökleri birim çember üzerinde dizildir ve birbiri içine girmiş (interlaced) durumdadır. Şekil 3.22 de verilen LPC katsayılarının oluşturduğu polinomların kökleri Şekil 5.9 de verilmiştir.

Şekil 5.35: P(z) ve Q(z) polinomlarının kökleri

P(z) ve Q(z) polinomları p/2 tane kompleks konjuge kök setlerinden oluşmaktadır dolayısıyla her polinom beş tane kök ile temsil edilebilir. Polinomların uygun kökleri alınında LSF parametreleri elde edilir. LSF parametreleri sıralanır ve formantlar ile doğrudan bir ilişkiye sahiptirler.

LSF parametrelerinden LPC filtre katsayıları denklem (5.2) de verilen ifade sayesinde geri elde edilebilir.

\[
A(z) = \frac{P(z) + Q(z)}{2} \tag{5.3}
\]
Tablo 5.1 de verilen LPC katsayılarının, LSF parametreleri cinsinden ifadesi tablo 5.2 de verilmiştir.

<table>
<thead>
<tr>
<th></th>
<th>LSF Parametreleri (rad)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analiz Edilen Konuşma</td>
<td>0.1888 0.2785 0.4215 0.5062 0.8091 1.6516</td>
</tr>
<tr>
<td></td>
<td>1.7357 1.9526 2.3579 2.7669</td>
</tr>
<tr>
<td>GSM ile Kodlanan Konuşma</td>
<td>0.1883 0.2816 0.4209 0.5049 0.8124 1.5144</td>
</tr>
<tr>
<td></td>
<td>1.7734 1.9756 2.4102 2.7515</td>
</tr>
</tbody>
</table>

**Tablo 5.2:** GSM kodlayıcı ile analiz edilen ve sentezlenen konuşmaların LSF parametreleri cinsinden gösterilimi

5.4. **LBG Algoritması**


![Şekil 5.36: 2 boyutlu düzlem için vektörel kuantalama](image_url)

İki boyutlu bir düzlem için vektörel kuantalama örneği şekil 5.9’da verilmiştir. Toplam alan on altı bölgeye ayrılmıştır. Kırmızı noktalar kod kelimelerini (code words) ifade eder. On altı kod kelimelerinden oluşan, kod kitabı dört bit ile kodlanabilir.

Sıkıştırma işleminde, tüm ses veya resim içinde birbirine benzer parçalar bulunur ve bunlar kod kelimelerini oluşturur. Sıkıştırma işleminden sonra dosyadaki...
parçalar kod kitabındaki indeksler ile ifade edilir. Bu durum Şekil 5.11 da gösterilmiştir.

![Diagram](image)

**Şekil 5.37:** Vektörel kuantalama ile dosyanın sıkıştırılması [19].


LBG algoritmasında başlangıç olarak, sınıflandırılacak vektörlerden sınıf sayısı kadarı kod kelimesi olarak seçilir. Ardından her bir kod kitabi kendisine en yakın LBG algoritmasında başlangıç değerleri rastgele seçildiği zaman, sonuç ilk seçilen vektörlere göre değişir. Böyle bir hataya maruz kalmamak için bölünen (splitting) LBG algoritması geliştirilmiştir. Bu algoritmanın gerçeklenmesi için uzaklığı ifade eden ölçüt seçildikten sonra (Burada Öklid mesafesi (Euclidean Distance) seçilmiştir) uygulanması gereken adımlar aşağıda sıralanmıştır [20]:

**Algoritmanın işleyişi sırasında kullanılabilecek temel kavramlar:**

Sınıflandırılacak vektörler;

\[ B = \{B_1, B_2, \ldots, B_m \} \] \hspace{1cm} (5.4)

Elde edilecek kod kelimeleri;

\[ C = \{C_1, C_2, \ldots, C_n \} \] \hspace{1cm} (5.5)

Uzaklık kullanılabilecek uygulamaya göre farklı şekillerde belirtilebilir. Bu çalışmada kullanılan öklid mesafesi, vektör boyutu l olmak üzere denklem (5.6)’da verilmiştir.

**Adım 1:**
Başlangıç değerleri atanır.

- Sınıf sayısı, \( N = 1 \).
- Ortalama Hata, \( D^{(0)} = 0 \).
- \( e > 0 \) olmak üzere küçük bir \( e \) sayısı belirlenir.
- Bir sınıfdan oluşan sistem için kod kelimesi;

\[
C_i = \frac{1}{M} \sum_{k=1}^{M} B_m
\]  

**(5.7)**

**Adım 2:**
Bölünme (Splitting) işleminin yapıldığı kısımdır, sınıf sayısı ikiye katma çıkar.

\[
N = 2 \times N;
\]  

\[(5.8)\]

\( i = 1, 2, 3, \ldots, N \) için

\[
C_i^{(k)} = (1 + e \times C_i^{(k)})
\]  

\[(5.9)\]

\[
C_i^{(k)} = (1 - e \times C_i^{(k)})
\]  

\[(5.10)\]
Adım 3:
Her sınıf için en uygun kod kelimesinin arandığı iterasyon işleminin başladığı kısımdır. Her bir vektör için kendisine en yakın kod kelimesi aranır ve vektörler bulunan kod kelimesine ait sınıfa yerleştirilir. P sınıf vektörleri olmak üzere;

\[ P_i^{(k)} = B_n : d(B_n, C_i^{(k)}) < d(B_n, C_j^{(k)}) \quad ; \quad j \neq i \quad ; \quad i=1,2,\ldots,M \]  
(5.11)

Adım 4:
Bütün \( P_i^{(k)} \) sınıflarındaki \( B_i \) vektörleri için denklem (5.7)'de verilen formül ile ağırlık merkezi hesaplanır ve bulunan değer \( C_i^{(k)} \) kod kelimesi olur.

Adım 5:
Yeni belirlenen sınıflar için hata hesaplanır;

\[ D^{(k)} = \frac{1}{M} \sum_{k=1}^{N} D_k, \quad \text{her sınıf için} \quad D_k = \sum_{j=1}^{\text{uzunluk}(P^{(k)})} (C_k - B_j)^2 \]  
(5.8)

Adım 6:
- Hata büyüldüyse ya da \( E \) kabul edilebilir hata değeri için denklem (5.9)'da verilen eşitlik sağlanıyorsa adım 7'ye geçilir.

\[ (D^{(k-1)} - D^{(k)}) / D^{(k)} \leq E \]  
(5.9)
- Hata küçüldüyse ya da denklem (5.9)'da verilen eşitlik sağlanmıyorsa, denklem (5.10)'de verilen işlem yapılarak adım 3'e geçilir.

\[ k = k + 1 \]  
(5.10)

Adım 7:
Sınıf sayısı, istenen sınıf sayısına eşit değil ise adım 2'ye geçilir. İstenen sınıf sayısına ulaşılmış ise denklem (5.11)'de verilen eşitlik sayesinde kod kelimeleri bulunarak algoritma sonlandırılır.

\[ C_i = C_i^{(k)} \]  
(5.11)
Dağınık LBG algoritması kullanılan sınıflandırma işleme ait örnekler 
şekil 5.12 ve şekil 5.13’de verilmiştir.

Şekil 5.38: Dağınık LBG algoritmasıyla noktaların dört sınıfa ayrılması

Şekil 5.39: Rastgele oluşturulan 1000 vektörün dağınık LBG algoritmasıyla on altı 
sınıfa ayrılması
5.5. Kod Kitaplarının Oluşturulması

Sayısal verilerden GSM hattında bastırılmadan geçebilecek konuşmaların sentezlenebilmesi için konuşmaların temel özelliklerini içeren perde periyodu, enerji ve LSF parametrelerinden oluşan kod kitaplarının oluşturulması gerektiği ve oluşturulacak kod kitaplarının konuşmalarının sadece sesli kısımlarını içermesi gerektiğiinde bahsedildi.


Sonuç olarak tasarlanan sistemde kullanılmak üzere (4,6) LSF parametreleri için (5,5) bit sayısal veriler ile ifade edilmek üzere LSF kod kitapları tasarlanmıştır.

<table>
<thead>
<tr>
<th></th>
<th>Enerji</th>
<th>Enerjinin Logaritması</th>
<th>Nominal Enerji</th>
<th>Nominal Enerjinin Logaritması</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Analiz Edilen Konuşma</strong></td>
<td>0.8475</td>
<td>-1.6544</td>
<td>1.6164e-005</td>
<td>-110.3271</td>
</tr>
<tr>
<td><strong>GSM ile Kodlanan Konuşma</strong></td>
<td>0.6957</td>
<td>-3.6290</td>
<td>1.2562e-005</td>
<td>-112.8483</td>
</tr>
</tbody>
</table>

Tablo 5.3: Enerji, nominal enerji ve logaritmalarının incelenmesi

Tablo 5.3 incelenirse GSM kodlayıcı kullanılarak sıkıştırılan konuşma ile analiz edilen konuşmanın enerjinin logaritması kodlama için daha uygundur. Ayrıca
işaretlerin enerjisi incelenirken, denklem 3.51'de verilen nominal enerjinin logaritmasının kullanılması kodlama hatalarının giderilmesi açısından faydalıdır. Nominal enerjinin logaritması 2 bit, perde periyodunun logaritması 4 bit ile kodlanmış, gerekli kod kitapları lineer kuantalama yöntemi ile oluşturulmuştur.

Enerji ve perde periyoduna ilişkin özelliklerin GSM kodlayıcında kabul edilebilir miktarı bozulmasına karşı, LSF parametrelerinin daha iyi kodlanması da da bu parametrelerin tanınması için bir takım hata giderici yapıların kullanılması sistemin gürültü duyarlılığını azaltacaktır.

GSM kodlayıcı konuşmayı analiz edip sentezlediğinde kendi yapısındaki kodlama çözünürlüğünde sonuçlar verecektir. Bu yapı düşünülen GSM kodlayıcıya üretelen kod kitaplarındaki konuşmalar sentezlenerek konuşmalar bozulmayana kadar analiz edildi ve yeniden sentezlendi. Elde edilen yeni konuşmalar LPC yöntemi ile analiz edilerek kod kitaplarındaki LSF parametreleri güncellendi. Şekil 5.7 de verilen konuşma örneği için anlatılan geliştirme yapıldı ve şekil 5.14-5.15, tablo 5.4 de verildiği gibi sistemin çok daha iyi çalıştığı gözlemlendi.

**Şekil 5.40:** Kodlayıcıya Tekrar ederek Verilen Konuşmadaki Değişim
Tablo 5.4: GSM ile 19 defa ve 20 defa kodlanan konuşmaların LSF parametreleri

<table>
<thead>
<tr>
<th></th>
<th>LSF Parametreleri</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSM ile yirmi defa kodlanan konuşma</td>
<td>0.1888 0.2859 0.4506 0.6994 1.4886 1.8211</td>
</tr>
<tr>
<td></td>
<td>1.8826 2.2529 2.4056 2.6704</td>
</tr>
<tr>
<td>GSM ile on dokuz defa kodlanan konuşma</td>
<td>0.1886 0.2859 0.4561 0.7060 1.4823 1.8205</td>
</tr>
<tr>
<td></td>
<td>1.8866 2.2493 2.4087 2.6715</td>
</tr>
</tbody>
</table>

GSM kodlayıcida kullanılan hataların ve hattan kaynaklanan hataların giderilmesi adına kod kitapları en iyi şekilde tasarlanmaya çalışıldı. GSM kodlayıcida değişmeyerek kodlanabilen konuşmalar elde edilmiştir olsa da LPC yöntemi ile analiz ile değişmeyen konuşmalara ait parametreler bulunamadı. Farklı bir yaklaşım olarak, aynı kod kitapları ile konuşmanın sentezlenip analiz edilmesi yerine sentezlenirken kullanılan kod kitapları ile analiz edilirken kullanılan kod kitapları birbirinden farklı tasarlanabilir. Bu sayede kodlayıcından kaynaklanan hata oranı sıfıra indirgenir ve
hattan kaynaklanan gürültü duyarılığı azaltılır. Bir takım yapay sinir ağları veya konuşmacı tanıma algoritmalarının sisteme eklenmesi ile eğitilebilir bir kodlayıcı tasarlanabilir ve bu sayede iletişim kalitesi artar.
6. SONUÇLAR VE TARTIŞMA

Bu bitirme çalışmásında GSM konuşma hattı üzerinden konuşmanın güvenli gönderilebileceği bir sistem gösterildi. Bu sistemde konuşma, sayısallaştırılır ve şifrelenir. Şifrelenen bit dizileri, 13 kpbs GSM FR konuşma kodlayıcısında ve GSM iletim hattında hiç bozulmadan ya da düzeltilebilir bir hata miktarı ile iletilebilecek konuşmalara dönüştürülerek hatta verilir. Alıcı taraf; konuşmayı alır, analiz eder ve ilk konuşma yeniden üretilir.


Bu çalışma gelecekte, gerçek GSM hatlarında gerçek sistemler ile denenip, uygun kod kitapları oluşturulduktan sonra FPGA üzerinde yapılacak bir donanım tasarımının ilk halidir ve teorik bir modellenmesidir.
KAYNAKLAR


ÖZGEÇMİŞ

Adı Soyadı: Mehmet Akif Özkan

Doğum Yeri ve Tarihi: Ankara, 1989


Lisans: İstanbul Teknik Üniversitesi, Elektronik Mühendisliği; 2007-2011