



**DERİN ÖĞRENME MİMARİLERİ KULLANILARAK AYRIK VİDEO
GÖRÜNTÜLERİ ÜZERİNDEN İŞARET DİLİ TANIMA**

Cemil GÜNDÜZ

**DOKTORA TEZİ
BİLİŞİM SİSTEMLERİ ANA BİLİM DALI**

**GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ**

ARALIK 2021

ETİK BEYAN

Gazi Üniversitesi Bilişim Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Cemil GÜNDÜZ

08/12/2021

DERİN ÖĞRENME MİMARİLERİ KULLANILARAK AYRIK VIDEO GÖRÜNTÜLERİ ÜZERİNDEN İŞARET DİLİ TANIMA

(Doktora Tezi)

Cemil GÜNDÜZ

GAZİ ÜNİVERSİTESİ

BİLİŞİM ENSTİTÜSÜ

Aralık 2021

ÖZET

İşaret dilleri, işitme ve konuşma engelli bireylerin günlük yaşamda kullandıkları ana iletişim ortamları olan görsel dillerdir. Çok sayıda kanal üzerinden aktarılan işaretlerin bilgisayarlı tanınması sayesinde, işitme ve konuşma engelli bireyler hem diğer bireylerle hem de makineler ile iletişimlerini doğal şekilde yapabileceklerdir. Bu tez çalışmasında, derin öğrenme kullanılarak ayrik işaret dili videoları üzerinden işaret dili tanıma gerçekleştirilmiştir. BosphorusSign veri kümesinin “genel” isimli alt kümesi kullanılarak yapılan çalışmada, öncelikle veri artırma ve ön işleme parametrelerinin belirlenmesi için çalışmalar yürütülmüştür. Ardından çeşitli derin öğrenme modelleri kullanılarak yapılan deneyler sonucunda işaret dili tanıma için kullanılabilecek uygun bir model belirlenmiştir. Daha sonra işaret dilindeki çeşitli kanalları ifade etmek üzere çıkarılan farklı veri kiplerinin tek başlarına ve çeşitli birleşimlerle başarımları değerlendirilmiştir. Bu sayede çok kipli bir işaret dili tanıma için kullanılacak en uygun veri kipi kombinasyonu elde edilmiştir. Son olarak, deneyler sonucunda elde edilen parametreler ve veri kiplerini kullanan çok kipli bir işaret dili tanıma modeli önerilmiştir. Önerilen model, RGB, eklem ve optik akış kiplerinde toplamda 6 farklı veri akışını bir arada girdi olarak almaktadır. Model bünyesindeki birleştirme mekanizması ile veri akışlarından çıkarılan öz nitelikler birleştirilmiş ve derin öğrenme tabanlı sınıflandırıcı katmanlara aktarılmıştır. Uçtan uca bir yöntemle eğitilen bütünsel işaret dili tanıma modeli, kullanılan veri setinde görülen en yüksek başarımla olan %89,3 doğruluk sunmuştur. Önerilen çok kipli işaret dili tanıma modelinin işaret dili tanıma başarımını iyileştirme konusunda geliştirilebilir bir potansiyeli vardır.

Bilim Kodu : 92419

Anahtar Kelimeler : İnsan-bilgisayar etkileşimi, işaret dili tanıma, derin öğrenme, tekrarlayan sinir ağları, evrimsel sinir ağları,

Sayfa Adedi : 100

Danışman : Doç. Dr. Hüseyin POLAT

ISOLATED SIGN LANGUAGE RECOGNITION USING DEEP LEARNING

ARCHITECTURES

(PhD. Thesis)

Cemil GÜNDÜZ

GAZİ UNIVERSITY

INSTITUTE OF INFORMATICS

December 2021

ABSTRACT

Sign languages are visual languages, which are the main communication medium used by hearing and speech impaired individuals in their daily life. Thanks to the computer recognition of the signs transmitted over many channels, individuals with hearing and speech disabilities will be able to communicate naturally with both other individuals and machines. In this thesis, sign language recognition was carried out through isolated sign language videos using deep learning. In the study, first experiments were carried out to determine the data augmentation and preprocessing parameters using the “general” subset of the BosphorusSign dataset. Then, as a result of experiments using various deep learning models, a suitable model for sign language recognition was determined. After that, a set of studies were conducted using different data modalities. The performances of data modalities that extracted to express various channels in sign language were evaluated by themselves and with various combinations. In this way, the most suitable data modality combination to be used for a multimodal sign language recognition has been obtained. Lastly, a multimodal sign language recognition model is proposed which uses the parameters and data modalities obtained by the experiments. The proposed model takes a total of 6 different data streams as input in RGB, joint and optical flow modalities. The features extracted from the data streams are combined and transferred to the deep learning-based classifier layers with the help of a fusion mechanism in the model. The holistic sign language recognition model, trained with an end-to-end method, provided 89.3% accuracy which is the highest performance seen in the data set used. The proposed multimodal sign language recognition model has strong potential to improve sign language recognition performance.

Science Code : 92419

Key Words : Human computer interaction, sign language recognition, deep learning, convolutional neural networks

Page Number : 100

Supervisor : Assoc. Prof. Hüseyin POLAT

TEŞEKKÜR

Tezimin her aşamasında destekleri ve tavsiyelerinden dolayı tez danışmanım Doç. Dr. Hüseyin POLAT'a, değerli görüşlerini paylaşan ve tezin bu noktaya gelmesinde katkılarını esirgemeyen tez izleme jürimin üyeleri Prof. Dr. Suat ÖZDEMİR ve Doç. Dr. Uçman ERGÜN'e saygılarımı ve teşekkürlerimi sunarım. Doktora eğitimim süresince yanımda olan ve zorlukları benimle paylaşan değerli aileme ayrıca teşekkür ederim.



İÇİNDEKİLER

	Sayfa
ÖZET	iv
ABSTRACT.....	v
TEŞEKKÜR.....	vi
İÇİNDEKİLER	vii
ÇİZELGELERİN LİSTESİ.....	ix
ŞEKİLLERİN LİSTESİ.....	x
RESİMLERİN LİSTESİ.....	xii
SİMGELER VE KISALTMALAR.....	xiii
1. GİRİŞ.....	1
2. İLGİLİ ÇALIŞMALAR	9
2.1. Geleneksel Makine Öğrenmesi Tabanlı İşaret Dili Tanıma.....	11
2.1.1. Veri toplama	13
2.1.2. Veri ön işleme.....	14
2.1.3. Öznitelik çıkarımı	15
2.1.4. Sınıflandırma	16
2.2. Derin Öğrenme Tabanlı İşaret Dili Tanıma	16
3. MATERYAL VE METODLAR.....	21
3.1. Makine Öğrenmesi	21
3.2. Yapay Sinir Ağları	24
3.3. Derin Öğrenme.....	30
3.4. Evrişimsel Sinir Ağları.....	32
3.5. Tekrarlayan Sinir Ağları	34
4. VERİ KÜMESİ VE VERİ ÖNİŞLEME	39
4.1. Veri Kümesi	39

	Sayfa
4.2. Veri Önişleme ve Veri Artırma.....	40
4.2.1. Resimlerde veri artırma yöntemleri	41
4.2.2. Videolarda veri artırma yöntemleri	43
5. DENEYSEL BULGULAR	47
5.1. Veri Artırma Deneyleri ve Bulgular	48
5.2. Çeşitli Derin Öğrenme Modelleri ile Gerçekleştirilen Deneyler ve Bulgular.....	56
5.3. Çeşitli Veri Tipleri ile Deneyler ve Bulgular	64
5.3.1. Çok veri kipli işaret dili tanıma çalışmaları.....	64
5.3.2. Çok veri kipli deneyler için veri önişleme.....	68
5.3.3. Veri kiplerinin tek başına değerlendirildiği deneylerde elde edilen bulgular	70
5.3.4. Veri kiplerinin birleşimleri ile yürütülen deneyler ve bulgular	72
5.4. Önerilen Çok Kipli İşaret Dili Tanıma Modeli ve Bulgular	74
6. SONUÇ VE ÖNERİLER	81
7. KAYNAKLAR.....	85
EKLER.....	95
EK-1. Veri kümesindeki sınıflar	96
ÖZGEÇMİŞ	99

ÇİZELGELERİN LİSTESİ

Çizelge	Sayfa
Çizelge 3.1. Yapay sinir ağlarında kullanılan aktivasyon fonksiyonları	27
Çizelge 5.1. İkili bir karmaşıklık matrisi	47
Çizelge 5.2. Örneklem boyutları ve önişleme sonrasında elde edilen sayıları	50
Çizelge 5.3. Test verisindeki sınıflar ve örnek sayıları.....	50
Çizelge 5.4. 1. grup deneylerde kullanılan C3D modelinin katmanları.....	52
Çizelge 5.5. Deneyler sonucunda elde edilen doğruluk oranları	53
Çizelge 5.6. En yüksek doğruluk veren modelin sınıflandırma sonuçları	55
Çizelge 5.7. İkinci grup deneylerde test verisindeki sonuçlar	60
Çizelge 5.8. Tüm veri kümesi üzerinde yapılan eğitimlerin sonuçları	62
Çizelge 5.9. Her bir akışın tek başına kullanıldığı deneylerde sınıflandırma başarımları ..	71
Çizelge 5.10. Akışların farklı şekillerde birleştirilmesi ile elde edilen sınıflandırma başarımları	73
Çizelge 5.11. BosphorusSign veri kümesi kullanan çalışmalarda elde edilen sonuçlarla karşılaştırma	79
Çizelge 5.12. LSA64 veri kümesinde edilen sonuçların karşılaştırması	79

ŞEKİLLERİN LİSTESİ

Şekil	Sayfa
Şekil 2.1. Geleneksel makine öğrenmesi tabanlı işaret dili tanıma aşamaları	12
Şekil 3.1. Biyolojik sinir hücresi.....	24
Şekil 3.2. Yapay sinir hücresi	25
Şekil 3.3. Çok katmanlı bir yapay sinir ağı mimarisi	28
Şekil 3.4. İleri beslemeli ve tekrarlayan sinir ağı mimarileri.....	30
Şekil 3.5. Temel bir evrişimsel sinir ağının katmanları	33
Şekil 3.6. Bir tekrarlayan sinir ağının genişletilmesi sonucu ortak ağırlıkların gösterimi	35
Şekil 3.7. Çeşitli TSA mimarileri A: diziden diziye, B: diziden vektöre, C: Vektörden diziye, D: Kodlayıcı çözücü	36
Şekil 3.8. UKSB birimi ve kapılar	37
Şekil 4.1. Videolardaki kare sayılarını ifade eden histogram	40
Şekil 5.1. İki ve üç boyutlu evrişim işlemleri	51
Şekil 5.2: Kare sayısı - doğruluk grafiği.....	54
Şekil 5.3. Deneyler sonucunda en yüksek doğruluk veren parametrelerin eğitim ve doğrulama süreci	54
Şekil 5.4. Deneyler sonucunda en yüksek doğruluk veren modelin karmaşıklık matrisi	56
Şekil 5.5. I3D model mimarisi.....	57
Şekil 5.6. Resnet modelinde kısayol bağlantıları.....	58
Şekil 5.7. Transfer öğrenme yoluyla deney veri kümesinde yapılan eğitimin doğruluk oranı değişimi.....	60
Şekil 5.8. 15 sınıflık test verisinde I3D modeli ile çıkarılan özneliliklerin t-SNE görselleştirmesi.....	61
Şekil 5.9. I3D modelinin transfer öğrenme yöntemi ile eğitim süreci.....	63
Şekil 5.10. I3D modelinin 154 sınıflık veri kümesindeki karmaşıklık matrisi.....	63
Şekil 5.11. Optik akış renk dönüşümü	69

Şekil	Sayfa
Şekil 5.12. Önişleme sonrası elde edilen veriler: a) RGB beden videosu, b) optik akış videosu, c) beden eklem verisi, d) ve e) ellerin eklem verisi, kafa ve eller için kırpılmış video görüntüleri(f, g ve h).....	70
Şekil 5.13. Her bir veri kipindeki akışlar ve öznitelik çıkarımında kullanılan derin mimariler.....	71
Şekil 5.14. Geliştirilen çok kipli işaret dili tanıma modeli akışları	75
Şekil 5.15. Çok akışlı modelin eğitim sürecince doğruluk ve kayıp değişimi.....	76
Şekil 5.16. Çok akışlı modelin karmaşıklık matrisi.....	77
Şekil 5.17. Modelin yanlış sınıflandırdığı işaretler. Emeklemek (a) ve koşmak (b), Vermek (c) ve Cevap (d), Yıl (e),Kadın (f) ve Kız (g) işaretlerinin gösterimi.....	78

RESİMLERİN LİSTESİ

Resim	Sayfa
Resim 3.1. Bir evrişimsel sinir ağının farklı katmanlardaki öznitelik haritaları.....	34
Resim 4.1. Resimler üzerinde yapılan veri artırma örnekleri	42
Resim 4.2. Videonun ilk karesinden (A) YOLO ile işaretçi konumunun bulunması (B) ve uzamsal veri artırımı sonrası oluşan kareler (C).....	45



SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Kısaltmalar	Açıklamalar
-------------	-------------

ESA	Evrişimsel sinir ağı (Convolutional neural network - CNN)
GİB	Grafik işlem birimi (Graphics processing unit - GPU)
LMK	Leap Motion kontrolcüsü (Leap motion controller - LMC)
UKSB	Uzun kısa süreli bellek (Long short-term memory - LSTM)
RGB	Kırmızı, yeşil, mavi renk uzayı
TSA	Tekrarlayan sinir ağı (Recurrent neural network - RNN)
SMM	Saklı Markov modeli
YSA	Yapay sinir ağı

1. GİRİŞ

İletişim, duygu, düşünce, bilgi ve haber gibi iletilerin (mesaj) karşıdaki kişi veya topluluğa aktarılması olarak ifade edilebilir. Shannon'un bilgi teorisine göre iletişimin kaynak, şifreleme, kanal, çözme ve alıcı olmak üzere beş bileşeni vardır [1]. Bu teoriye göre bir iletiyi karşı tarafa aktarmak isteyen kaynak, öncelikle bunu belirli simgeleri kullanarak kodlar. Ardından kodlanmış olan ileti, alıcının da sahip olduğu bir kanal üzerinden aktarılır. Karşı taraftaki alıcı, bu iletiyi kendi simgeleri ile çözümleyerek yorumlar. Alıcı, iletiyi aldığına dair bir geri dönüşü yine kaynağın da sahip olduğu bir kanalı kullanarak gönderir. Böylelikle bir iletişim döngüsü tamamlanmış olur.

Engelsiz bireyler vermek istedikleri iletileri sözlü olarak iletirler. İletiyi gönderen bireyler kaynak konumunda olur ve iletileri doğal dillerin gramer kuralları uyarınca (kodlayıcı) kodlar ve ses tellerinin yardımı ile ses dalgalarına (kanal) dönüştürür. Karşıdaki muhatap bu ses dalgalarını kulakları yoluyla alır ve yine dil kuralları gereğince (çözücü) anlamlandırır. Alıcının iletiye cevap olarak söylediği sözler veya gösterdiği tepkiler (geri dönüş) kaynağa ulaştığında döngü tamamlanır. İnsanlar günlük yaşamlarında fark etmeden bu iletişim döngüsünü sürekli olarak gerçekleştirirler.

İletişim döngüsünün sorunsuz sağlanabilmesi için hem kaynak hem de alıcının aynı kanallara ve bu kanallardan gelecek iletileri kodlayıp çözebilmek için ortak değerlere sahip olmaları gerekir. Kaynak ve alıcıda, gelen iletileri alacak duyu organları sağlıklı olsa da, iletinin çözümü için ortak değerlerin bulunmaması iletişimin sağlıklı bir şekilde sağlanmasına engel olacaktır. Konuşulan dil bu değerlerden biridir. Aynı dili konuşmayan bireyler bu ortak değere sahip olmadığından daha evrensel olan jest ve mimikler yoluyla iletişim kurmaya çalışırlar. Jest ve mimikler kullanıldığında bile kültürel farklılıklardan kaynaklanan bazı yanlış anlaşılmalarda oluşabilir.

Engelsiz bireyler sözlü iletişimde vermek istedikleri iletileri konuşma dilleri yoluyla aktarırlar. Bunun yanı sıra jest ve mimikler de görsel yardımcı kanallar olarak kişiler arası iletişimde ileti aktarımında önemli bir yer teşkil eder [2]. Bu yardımcı kanallar kullanılarak konuşmaya farklı anlamlar katılabilir veya anlam güçlendirilebilir. Sözel kanaldan gönderilen bir ileti, farklı jest ve mimikler ile ifade edildiğinde tamamen zıt anlamlar

ortaya konulabilir. Örneğin ironi yaparken kullanılan jest ve mimikler sayesinde sözlü olarak verilenden daha farklı anlamlar ifade edilebilir.

Sözlü iletişim olmadan sadece jest ve mimikler yoluyla iletişim kurmak hem kaynak hem de alıcı açısından oldukça güçtür. Engelsiz bireyler her iki kanalda da iletileri alabildiklerinden iletişim döngüsünü sorunsuz gerçekleştirirler. Ancak görme, işitme veya konuşma yetilerinde herhangi bir engeli olan bireyler için iletişim bu kadar pürüzsüz ilerlememektedir.

İşitme ve konuşma engelli bireyler konuşma dilini iletişimde kullanamazlar. Bu bireyler engelsiz bireylerin konuşmanın anlamını güçlendirmek için kullandıkları jest ve mimikler gibi görsel kanallar üzerinden iletişim kurarlar. Dünya sağırler federasyonu verilerine göre dünyada 70 milyonun üzerinde işitme engelli birey bulunmaktadır [3]. Ülkemizde ise Engellilerin Sorun ve Beklentileri Araştırması sonuçlarına göre işitme engelli bireyler toplam engelli nüfusun %5,9'unu teşkil etmekte, toplam nüfusun %1.1'i işitme; 0,7'si ise konuşma güçlüğü çekmektedir [4].

İşaret dilleri, işitme ve konuşma engelli bireylerin iletişimde kullandıkları birincil ortamlar ve günlük yaşamlarında kendilerini ifade ettikleri yazılı veya sözlü olmayan görsel dillerdir. İşaret dillerinde kaynak veya alıcıdan en az birisinin sözel kanala sahip olmaması nedeni ile tüm iletiler görsel kanallar üzerinden aktarılır.

İşaret dillerinde iletiler birden çok kanal üzerinden eş zamanlı olarak aktarılırlar [5, 6]. İşaret dillerindeki iletişim kanalları el ile gösterilen işaretler dışında, yüz ifadeleri, vücudun duruşu ve hareketleri de olabilir. Sözlü iletişimde, konuşma, jestler ve mimikler gibi tüm kanalları sürekli kullanmak gerekli olmadığı gibi, işaret dillerinde de bir işaretin gösteriminde tüm bu kanalların gösterimi gerekmebilir. Bir işaretin anlamlı olması için sadece el ile yapılan işaret yeterli olurken, bir diğer işaret için birden fazla kanaldan ileti aktarımı yapılabilir.

İşaret dilleri, toplumda bilinen genel kanının aksine, evrensel diller değildir [7-10]. Nasıl ki ülkeler arasında konuşma dilleri arasında farklılıklar varsa, farklı ülkelerin işaret dilleri de farklılık gösterir. Benzer şekilde, bir ülkenin farklı coğrafi bölgeleri arasında konuşma dilinde farklı ağızlar görüldüğü gibi, işaret dillerinde de yerel farklılıklar gözlenebilir [11].

Sözel dillere benzer olarak işaret dilleri de ihtiyaca yönelik olarak doğal yollarla oluşmuş dillerdir ve her işaret dilinin kendine özgü bir gramer yapısı vardır. İşaretin gösteriminde kişisel farklılıklar olmasına rağmen, işaretler belli gramer kuralları dâhilinde gerçekleştirilir. Bu kurallar, işaret dilinin kullanıldığı bölgedeki sözel dilden farklıdır [10, 12]. Örneğin Türkçe konuşma dilinde kullanılan kelime ekleri, Türk işaret dilinin dilbilimsel özelliklerinden değildir.

İşaret dillerinde işaretler çok sayıda bileşenden oluşur. Bazı çalışmalarda bu bileşenler el işaretleri, el dışı işaretler [13] ve parmak alfabesi [5] olarak gruplandırılmıştır. El işaretleri parmakların aldığı şekil, elin yönü, konumu ve hareketleri ile oluşturulur. El dışı işaretler yüz ifadeleri, bedenin duruşu ve hareketleridir. Parmak alfabesinde ise işaret dilinde karşılığı olmayan kavramlar parmakla gösterilen harflerle hecelenir. Bunların yanı sıra, yapılan işaretin yönü, hızı, konumu gibi değişkenler de bazı hareketlerde anlamı değiştirmede veya kuvvetlendirmede kullanılabilir [5].

Birçok işaret dilinde işaretler iki elin birlikte kullanımı ile gösterilirler. Bir işaret dilinde tek elle ve çift elle yapılan işaretler bulunabilir. İki elin birden kullanıldığı işaretlerde asıl gösterilmek istenen işaret baskın el ile gösterilirken, ikinci el işaretin gösterimine yardımcı rolündedir. Baskın el genellikle işaretçinin günlük yaşamda işlerini yürütürken kullandığı elidir [13].

Bazı durumlarda işaretçinin ifade etmek istediği kavramın işaret dilinde bir karşılığı bulunmayabilir veya iletişim esnasında bir özel isim anılmak istenebilir. Bu durumda işaretçiler parmak alfabesini kullanırlar. Parmak alfabesi, anlatılmak istenen kavramın veya özel ismin parmakla gösterilen harfler yoluyla hecelenmesi olarak ifade edilebilir. Parmak alfabesi, işitme engelli bireylerin günlük yaşamda iletişimlerinde sıklıkla başvurdukları bir yöntemdir.

İşaret dillerinin çok sayıda bileşenden oluşması ve günlük hayatta işaret dillerinin sadece işitme ve konuşma engelli bireyler tarafından kullanılması gibi nedenlerden dolayı engelsiz bireylerin işaret dillerini doğal yollarla öğrenebilmeleri oldukça zordur. Toplumda işitme ve konuşma engelli bireyler ve yakın aileleri dışında işaret dili bilen kişi sayısı oldukça sınırlıdır. Bu durum iletişimin başarılı bir şekilde gerçekleştirilebilmesi için kaynak ve

alıcıda aynı kanalın kullanılması ve ortak kuralların bulunması gerekliliğinin sağlanamamasına neden olmaktadır.

Sağlıklı bir iletişim kuramamanın sonucu olarak işitme ve konuşma engelli bireylerin topluma adaptasyonları güçleşmekte ve bu kişiler toplumdan soyutlanmaktadır. İşitme engelli bireylerin günlük yaşamda ve sosyal çevrelerinde karşılaştıkları en büyük zorluklardan biri, toplumun yeterli desteği göstermemesi ve yardımcı olamaması olarak belirtilmiştir [4].

İletişimde karşılaştıkları sorunları aşmak için işitme ve konuşma engelli bireyler beraberlerinde bir refakatçi ile topluma katılmak zorunda kalmaktadırlar. Bu durum, işitme ve konuşma engelli bireylerin öz yeterliklerinin önünde büyük bir engel olarak karşımıza çıkmaktadır. Bazı durumlarda bu refakatçiler ücretle hizmet veren işaret dili çevirmenleri de olabilmekte ve bu da birey için maddi yükleri de beraberinde getirmektedir. İşaret dili tanıma, işitme ve konuşma engelli bireylerin karşılaştıkları iletişim kaynaklı sorunları aşmak için gelecek vaat eden bir uygulama alanıdır.

İnsan bilgisayar etkileşimi alanının alt çalışma alanlarından birisi olan işaret dili tanıma, sürekli veya aralıklı olarak gerçekleştirilen işaretlerin bilgisayarlar yardımı ile tercüme edilebilmesini amaçlar. İşaret dili tanıma sistemleri, bilgisayar teknolojisi kullanılarak işaret dilinin, işaret dili bilmeyen kişilere metin, resim veya sesli olarak çevirisini yapan sistemler olarak tanımlanabilir [14]. Günümüzde neredeyse her bireyin üzerine taşıdığı akıllı telefonların yüksek işlem kapasitesine sahip olması, yaygın olarak kullanılabilecek işaret dili tanıma uygulamalarının geliştirilebilmesine imkân tanımaktadır. Geliştirilecek bir işaret dili tanıma uygulaması, işitme engelli bireylerin günlük yaşamlarını kolaylaştıracak ve bu bireyler ile engelsiz bireyler arasındaki iletişim bariyerinin aşılmasına yardımcı olacaktır.

İşaret dillerinin bilgisayarlar yardımı ile tanınması, işitme ve konuşma engelli bireylerin sosyal yaşama katılmalarına yardımcı olacaktır. Fakat bu problemin çözümü için aşılması gereken bazı sorunlar bulunmaktadır. Bunlar; işaret dilinde bir işaret farklı kişilerce yapıldığında bazı kişisel farklılıkların bulunması ve işaretin çözümü için birden çok kanalın birlikte değerlendirilmesi gerekliliğidir. Eğer amaç sürekli işaret dili tanıma ise, geliştirilecek sistem hem işaretleri, hem parmak alfabesini tanıyabilmeli ve bu işaretler

arasındaki geiş hareketlerini göz ardı edip sadece anlamı olan kısımları algılayabilmelidir. İřaret dili tanıma alıřmaları bu amaçla hareket tanıma (*gesture recognition*) alıřmalarındaki gelişmelerden sıklıkla yararlanır.

Saėlık, eėlence gibi alanlarda uzun süredir kullanılan hareket tanıma uygulamaları yakın zamanda son kullanıcıların hizmetine de sunulmuş ve televizyon gibi aletlerin kontrolünde de kullanılmaya başlanmıştır. Hareket tanıma uygulamaları, görüntü işleme yöntemleri kullanılarak jestlerin tanınmasını amaçlar ve bu jestler aracılığı ile donanımların kontrolü sağlanabilir. İřaret dili tanıma alıřmaları ile hareket tanıma alıřmaları benzerlik göstermektedir. Ancak iřaret dili tanıma, hareket tanıma alıřmalarına göre daha zor bir alandır. İřaret dillerinin karmařık ve katı kurallarının olması, iřaretlerin birden ok kanal yoluyla ifade edilmesinin yanı sıra iřareti bulunmayan bir kavramı ifade ederken parmak alfabesinin kullanılması, iřaretlerin gösteriminde ortamdaki nesnelerin de kullanılması gibi durumlar iřaret dili tanıma sistemlerinin geliştirilmesini zorlařtırmaktadır. Ancak derin öğrenme gibi güncel yöntemler bu zorlukların daha kolay aşılabilmesini sağlayabilir.

Son yıllarda görüntü işleme alıřmalarında sıklıkla kullanılan derin öğrenme yöntemleri, iřaret dili tanıma alıřmalarında da yoğun olarak kullanılmaya başlanmıştır. Görüntülerdeki ilişkileri modelleyip öznitelik çıkarımını kullanıcı müdahalesine ihtiyaç duymadan yapabilen evriřimsel sinir aėları (ESA) (*Convolutional Neural Networks - CNN*) aracılığı ile geleneksel makine öğrenmesi tabanlı iřaret dili tanıma alıřmalarındaki manuel öznitelik çıkarımına gerek kalmamıştır. 3 boyutlu evriřimsel sinir aėı modelleri, videolardaki hem uzamsal hem de zamana baėlı veriyi birlikte modelleyebilmekte ve daha yüksek tanıma başarımı gösterebilmektedir. Sıralı veriler üzerinde başarılı bir şekilde modelleme gerekleřtirebilen diziden diziye (seq2seq) derin öğrenme modelleri yardımı ile sürekli iřaret dili tanıma uygulamalarındaki hata oranları da giderek düşmektedir.

Son yıllarda derinlik algılayan kameraların yaygınlařması ve cep telefonlarında birden ok kameraya yer verilmesi gibi gelişmeler iřaretin gösterimindeki 3. boyutun da deėerlendirilebilmesine imkân sağlamaktadır. Standart kameralar ile elde edilen renkli görüntülerin (*RGB*) yanı sıra farklı algılayıcılar ile toplanan derinlik gibi veri türleri ok kipli (*multimodal*) iřaret dili tanıma sistemlerinin geliştirilmesine imkân sağlamak ve bu sayede iřaret dili tanıma başarımı artırılabilir.

Bu tez çalışmasında, derin öğrenme modelleri kullanılarak ayrık video görüntüleri üzerinden çok kipli Türkçe işaret dili tanıma gerçekleştirilmiştir. Bunun için ilk olarak veri ön işleme ve veri artırma yöntemlerinde çeşitli parametreler ile deneyler yapılmıştır. Ardından günümüzde hareket tanıma alanında yüksek başarımlar sunan çeşitli derin öğrenme modellerinin işaret dili tanımadaki sonuçları değerlendirilmiştir. Ardından, Türk işaret dili özelinde olmak üzere işaret dilindeki farklı kanalların işaret dili tanımaya etkisi incelenmiş ve en yüksek başarımları veren kanalları kullanan çok kipli bir uçtan uca işaret dili tanıma sistemi önerilmiştir.

Bu tez çalışmasının amacı, Türk işaret dili kullanan bireylerin günlük yaşamlarını kolaylaştırmak için kullanılabilecek bir işaret dili tanıma sistemi geliştirmektir. Geliştirilen sistemin ekstra donanımlara ihtiyaç olmadan standart RGB kamera görüntüleri üzerinden çalışabilmesi hedeflenmiştir. Bu amaçla;

- İşaret dili videolarının en iyi şekilde temsilini sağlayan parametreleri belirlemek üzere deneyler yürütülmüş,
- Çeşitli derin öğrenme modelleri kullanılarak bu modellerin işaret dili tanımadaki başarımları incelenmiş,
- Türk işaret dilindeki farklı iletişim kanallarının işaret dili tanıma başarımına etkileri değerlendirilmiş,
- RGB görüntülerden çıkarılabilecek çeşitli veri kiplerinin işaret dili tanımadaki verimlilikleri değerlendirilmiş,
- Farklı veri kiplerini birlikte girdi olarak alabilen derin öğrenme modelleri ve birleştirme şekilleri incelenmiştir.

Yürütülen çalışmalar sonucunda bu tezin bilime yaptığı katkılar şunlardır;

- Türk işaret dilindeki çeşitli iletişim kanallarının etkilerini değerlendiren ilk çalışma olması,
- Çok kipli uçtan uca bir işaret dili tanıma sistemi önerilmesidir.

Bu tez çalışması şu şekilde biçimlendirilmiştir:

2. Bölümde işaret dili tanıma konusunda yapılmış çalışmalara değinilmiştir. Farklı dillerde yapılmış işaret dili tanıma çalışmalarında kullanılan yöntemler açıklanmıştır. İşaret dili

tanıma çalışmaları geleneksel makine öğrenmesi tabanlı ve derin öğrenme tabanlı çalışmalar olarak sınıflandırılmış ve her bir sınıftaki süreçler detaylandırılmıştır.

3. bölümde işaret dili tanıma konusunda yapılmış çalışmalara temel teşkil eden yöntemler hakkında bilgiler verilmiştir. 4. bölümde, bu tez çalışmasında kullanılan veri kümesinin özelliklerine değinilmiş ve veri ön işleme aşamaları detaylandırılmıştır.

5. bölümde ayırık işaret dili videoları üzerinden işaret dili tanıma için kullanılacak parametrelerin ve derin öğrenme modellerinin belirlenmesi amacı ile çeşitli gruplar halinde yapılan deneylerin sonuçları sunulmuştur. Bu deneyler üç grup halinde yürütülmüş ve ilk grup deneylerde videolardan alınacak örneklemelerin kare sayıları ve boyutlarının belirlenmesi hedeflenmiştir. İkinci grup deneylerde çeşitli derin öğrenme modelleri ile deneyler yürütülerek başarımları değerlendirilmiştir. 3. grupta ise farklı veri kipleri ile deneyler yürütülmüştür. Ardından önerilen çok veri kipli uçtan uca işaret dili tanıma sisteminin detaylarına yer verilmiştir. Bölüm sonunda önerilen çok kipli işaret dili tanıma modelinin işaret dili tanıma sonuçları sunulmuştur.

6. bölümde ise tez sürecinde yapılanların genel bir özeti sunulmuş ve önerilen işaret dili tanıma modeli ile ilgili değerlendirmelere yer verilmiştir. Bölüm sonunda, işaret dili tanıma üzerine yapılabilecek diğer çalışmalara ışık tutmak adına önerilerde bulunulmuştur.



2. İLGİLİ ÇALIŞMALAR

İnsan bilgisayar etkileşimi, kişilerin günlük iletişimdekine benzer şekilde doğal yollarla ve sezgisel olarak bilgisayarlarla iletişim kurabilmesini sağlayan bir çalışma alanıdır [2]. Günümüz bilgisayar etkileşiminde grafik arayüzler en sık kullanılan arabirimdir. Ancak grafik arayüzler sundukları girdi ve çıktı imkânları bakımından hem de kullanıcı deneyimi açısından doğal bir etkileşim sunmaktan uzaktır. Bu ortamlar, zaten kişiler arası iletişimde çok sayıda güçlükle karşılaşan işitme engelli bireyler için ise daha sınırlayıcı olmaktadır. Geliştirilecek bir işaret dili tanıma sistemi, bu kişilerin hem diğer bireylerle hem de makineler ile olan iletişimlerini kolaylaştırması açısından önemlidir [15].

İşaret dili tanıma, insan bilgisayar etkileşimi alanının alt çalışma alanlarından birisidir. İşaret dili konusunda insan bilgisayar etkileşimi alanındaki çalışmalar sadece işaret dili tanıma ile sınırlı değildir. [10, 15, 16]’da yürütülen çalışmalar işaretlerin tanındıktan sonra bu kavramların doğal dile dönüştürülmesi veya bunun tersini amaçlamaktadır. Bazı çalışmalarda, doğal dil metinlerinden 3 boyutlu bilgisayar grafikleri ve robotlar yardımı ile işaret dili gösterimi ve işaret dili eğitimi hedeflenmiştir [17, 18]. Bunun yanı sıra, işaret dili tanıma araştırmacıları ve dilbilimcilerin kullanımı için veri kümesi geliştirme [7, 12, 19, 20] gibi konularda da yapılmış çalışmalar bulunmaktadır. İşaret dili çalışmaları, görüntü işleme, doğal dil işleme, görüntü sentezleme, dilbilim gibi birçok alt çalışma alanı ile de etkileşim halindedir.

İşaret dili tanıma konusundaki çalışmalar, işaretlerin devamlı olması bakımından ses tanıma, el hareketlerinin tanınması bakımından ise hareket tanıma konusundaki gelişmelerden yararlanmaktadır. Hareket tanıma çalışmalarında kullanıcılar tarafından yapılacak hareketler araştırmacılar tarafından belirlenebilir ve işaret dillerinde karşılaşılan kapatma (occlusion) gibi birçok problemin önüne geçilebilir [2]. Ancak işaret dillerinin katı kurallara sahip olması, işaretlerin belli kurallar dâhilinde gerçekleştirilmesi, birden çok kanaldan eş zamanlı olarak gösterim yapılması gibi durumlar işaret dili tanıma çalışmalarının hareket tanıma çalışmalarına göre daha zor olmasına neden olmaktadır [5].

İşaret dili tanıma çalışmaları algılayıcı tabanlı ve görsel işaret dili tanıma olmak üzere iki grupta incelenebilir [5, 21-23]. Algılayıcı (sensör) tabanlı işaret dili çalışmalarında engelli

birey (iřaretçi), bir eldiven, giysi veya bařlık giymektedir. Bu giyilebilir cihazlar üzerinde EMG (elektromiyografi), ivmeölçer, jiroskop gibi algılayıcılar bulunur. İřaretçiler iřaretleri icra ederken, algılayıcılar kiřinin hareketlerini sayısallařtırır. İřaretçinin ellerinin konumu, yönü, řekli, parmaklarının řekli ve duruřu gibi birçok parametre algılayıcılar yardımı ile toplanabilir. Toplanan veriler sınıflandırıcılara dođrudan aktarılarak tanıma iřlemi gerçekteřirilir. Bu yöntemin görme tabanlı iřaret dili tanıma sistemlerine göre daha yüksek bařarım gösterdiđi ifade edilmiřtir. Ancak iřaretçinin, iřaretler yapılırken bu giysileri giymesi gerekliliđi kiřinin hareket özgürlüğünü kısıtlamaktadır [5]. Bu nedenle algılayıcı tabanlı iřaret dili tanıma sistemleri günlük yařamda kullanım için çok uygulanabilir görülmemiřtir [13].

Görsel veya görme tabanlı (bazı kaynaklarda dıřsal) iřaret dili tanıma sistemlerinde ise iřaretler bir kamera yardımı ile kaydedilir. Bu kameralar, standart RGB kameralar olabileceđi gibi derinlik algılayan kameralar [24] veya *Kinect* [25, 26] ve Leap Motion Kontrolcüsü (LMK) (*Leap Motion Controller - LMC*) [27, 28] gibi özel donanımlar da olabilir. Görme tabanlı iřaret dili tanıma çalıřmalarında algılayıcı tabanlı çalıřmaların aksine, kameralardan elde edilen görüntüler dođrudan sınıflandırmada kullanılamaz. Kaydedilen görüntüler, sürecin iřlem yükünü artıran bir dizi öniřlemden geçirilmeli ve sınıflandırıcıların kullanabilmesi için öznitelik çıkarımı gerçekteřirilmelidir. Görme tabanlı iřaret dili tanımada farklı ten rengine sahip iřaretçiler, kayıt ortamındaki ıřıklandırma ve kayıt açısı, arka plan gürültüsü gibi durumlar tanıma bařarımına olumsuz etki edebilir.

Görme tabanlı iřaret dili tanıma çalıřmaları statik ve hareketli görüntüler üzerinden gerçekteřirilebilir. Statik iřaret dili tanımada, statik görüntüler üzerinden parmak alfabesi harflerinin veya hareketsiz iřaretlerin tanınması hedeflenmektedir. Hareketli iřaret dili çalıřmalarında ise videolar kullanılır. İřaret dili tanımada kullanılan videolarda eller serbest bir durumdan bařlayıp tek bir iřaret gerçekteřirildikten sonra serbest haline geri dönüyorsa bu türdeki çalıřmalar ayrıık iřaret dili çalıřmalarına dâhil edilir. Sürekli iřaret dili çalıřmalarında ise videolarda art arda birden fazla iřaret gösterilir. Ayrıık ve sürekli iřaret dili arasındaki fark, ayrııkta bir kelime, sürekli cümle gösterimi yapılmaktadır gibi örneklendirilebilir.

Sürekli işaret dili videolarında bir işaret tamamlanıp diğerinin gösterimine başlanırken arada hareket geçişleri (*movement epenthesis*) olmaktadır. Hareket geçişleri ne önceki işarete ne de sonraki işarete dâhil edilebilir. Sürekli işaret dili çalışmalarında aradaki bu geçiş kısımlarının göz ardı edilebilmesi işaret dili tanımanın performansı için önemlidir. Bu nedenle sürekli işaret dili tanımanın daha karmaşık ve zor ancak günlük yaşamda kullanmak için daha yararlı olduğu ifade edilmektedir [5].

Görme tabanlı işaret dili tanıma çalışmaları geleneksel makine öğrenmesi tabanlı ve derin öğrenme tabanlı işaret dili tanıma olmak üzere iki sınıfta incelenebilir [22]. Geleneksel makine öğrenmesi tabanlı işaret dili tanıma sistemlerinde görüntü üzerinde yapılan ön işleme ile işaret dili tanıma yararlı olan alanların tespiti, ayrıştırılması, takibi sağlanır. Ardından bu alanları temsil edecek öznitelikler çıkarılır. Son aşamada ise çıkarılan özniteliklerden sınıflandırıcılar yardımı ile işaret dili tanıma gerçekleştirilir. Geleneksel makine öğrenmesi tabanlı yöntemlerin dezavantajları, sınıflandırmanın doğrudan öznitelik çıkarımına başarımına bağlı olması, çıkarılan özniteliklerin temsil yeteneğinin sınırlı olması nedeniyle karmaşık işaretlerin tespitinde yetersiz kalması ve adım adım yapılan işlemlerin gerçek zamanlı uygulamalarda düşük başarımla göstermesidir [22].

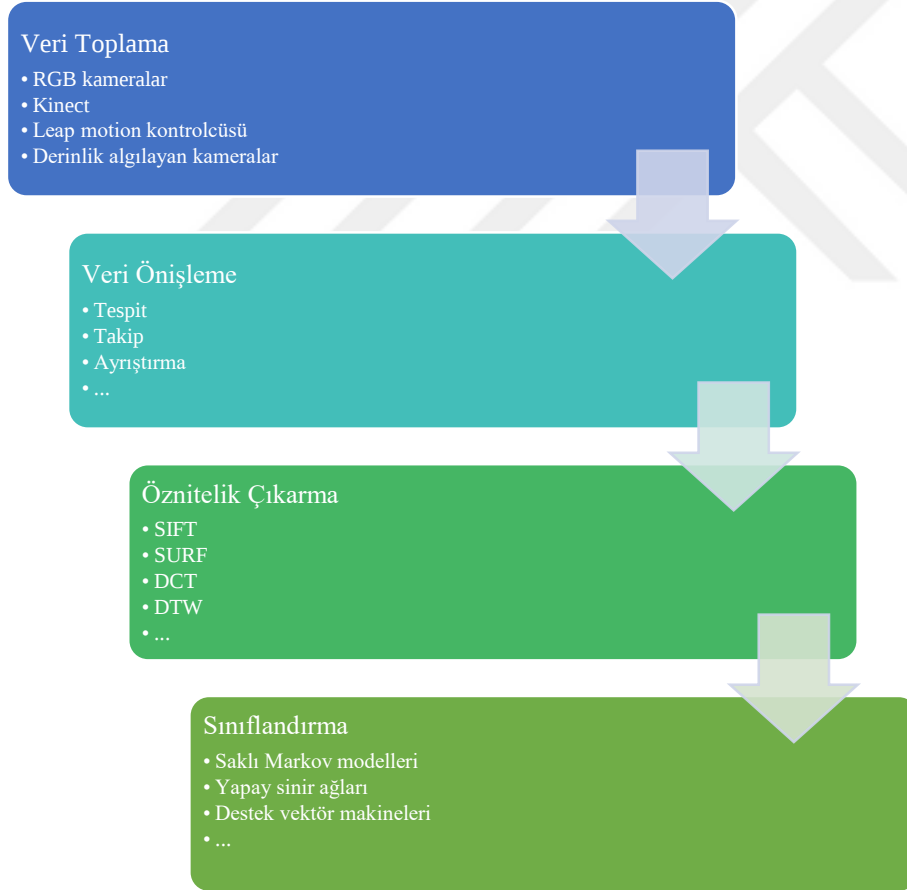
Derin öğrenme tabanlı işaret dili tanıma çalışmalarında ise sıklıkla evrimsel sinir ağlarından yararlanılır ve öznitelik çıkarımı için kullanıcı müdahalesine ihtiyaç duyulmaz. Öznitelik çıkarımı evrim katmanları aracılığı ile otomatik olarak gerçekleştirilir. Öznitelik çıkarımı ve sınıflandırma eş zamanlı olarak yapıldığından gerçek zamanlı uygulamalarda geleneksel makine öğrenmesi tabanlı yöntemlere göre daha yüksek başarımla sunulabilir.

2.1. Geleneksel Makine Öğrenmesi Tabanlı İşaret Dili Tanıma

Geleneksel makine öğrenmesi tabanlı işaret dili tanıma çalışmaları, görüntülerden öznitelik çıkarımının farklı algoritmalar yardımı ile araştırmacıların müdahalesi ile yapıldığı işaret dili tanıma çalışmalarını ifade eder.

Geleneksel makine öğrenmesi tabanlı işaret dili tanıma çalışmalarının ilk aşaması veri toplama aşamasıdır. Ardından, toplanan veriden sınıflandırmada kullanılacak özniteliklerin çıkarımı gerçekleştirilir. Geleneksel makine öğrenmesi tabanlı çalışmalarda öznitelik

çıkartım aşaması iki şekilde ele alınmıştır. Eğer öznitelik çıkartımı öncesinde önışleme yoluyla eller ve yüzün tespiti ve takibi yapılmışsa bu durumda takip tabanlı öznitelik çıkartımından bahsedilir [5]. Takip tabanlı yöntemde önışleme aşamasında ilgi duyulan alanlar belirlenerek bu kısımlardan öznitelik çıkartımı yapılır. Bu aşamada elde edilen öznitelikler sınıflandırmanın sonucunu doğrudan etkilemesi nedeni ile tespit ve takip işlemi mümkün olan en yüksek başarımla yapılmalıdır. Diğer yöntem olan takipsiz öznitelik çıkartımında ise görüntünün tamamı üzerinden öznitelik çıkartımı yapılır. Bu işlem bazı çalışmalarda evrensel (*global*) öznitelik çıkartımı olarak ifade edilmiştir [29]. Öznitelik çıkartma aşamasını sınıflandırma takip eder. Bu aşamada çıkartılan öznitelikler sınıflandırma algoritmaları yardımı ile en uygun sınıfa atanırlar. Şekil 2.1’de geleneksel makine öğrenmesi tabanlı işaret dili tanıma süreçleri gösterilmiştir.



Şekil 2.1. Geleneksel makine öğrenmesi tabanlı işaret dili tanıma aşamaları

2.1.1. Veri toplama

Görme tabanlı işaret dili tanıma çalışmalarında veri toplama için kameralardan yararlanılır. Kamera karşısında işaretleri gerçekleştiren işaretçilerin hareketleri, farklı görüntü algılayıcıları yardımı ile toplanarak sayısallaştırılır. İşaret dilinde işaretlerin eksiksiz bir şekilde değerlendirilebilmesi için işaretçinin belden üstünün kaydedilmesi gerektiği ifade edilmiştir [23]. Görme tabanlı çalışmalarda yoğun olarak sadece RGB görüntü kaydeden standart kameralar kullanılır [30].

Tek bir kamera yardımı ile elde edilen görüntüler 2 boyutlu olacağından işaretin gösteriminde yapılan ileri-geri hareketin algılanması mümkün olmayabilir. Bu sorunu aşmak için bazı çalışmalarda farklı açılarda kayıt yapan birden çok kamera kullanılarak işaretin gösterimindeki derinlik boyutunun da yakalanması amaçlanmıştır. Wang ve diğerleri [30], 0-30° arasında değişen açılarda videolar çekerek oluşturdukları Çin İşaret Dilindeki 100 işaretlik veri kümesinde çekim açısından bağımsız işaret dili tanıma hedeflemişlerdir. [31] işaretteki derinliği algılayabilmek ve giyilebilir bir işaret dili tanıma sistemi geliştirebilmek için şapka üzerine yerleştirdikleri bir kameradan yararlanmışlardır. Şapka üzerindeki kamera ile birinci şahıs görüşünde çekilmiş görüntülerin karşıdan çekilmiş videolara göre daha yüksek başarımlı sunduğunu ifade etmişlerdir. [32]'de ön, üst ve yandan olmak üzere üç farklı kamera açısından veri toplanmış ve bu görüntülerden 3 boyutlu öznitelikler çıkarılmıştır. Çalışma sonucunda, çıkarılan 3 boyutlu özniteliklerin 2 boyutlu olanlara göre daha yüksek sınırlandırma başarımlı sunduğu belirtilmiştir.

Önişlemeye gerek duymadan işaretlerin gösterimindeki derinlik verisini sunabildiği için işaret dili tanıma çalışmalarında Kinect kamerası sıklıkla kullanılmıştır [14, 26, 33]. Bu kamera, uçuş zamanı (*Time of Flight - ToF*) yöntemini kullanarak derinlik bilgisi de sunabilmektedir. Derinlik bilgisinin elde edilmesi, işaretin gösterimindeki ileri geri hareketin algılanmasını sağlamakla birlikte, işaretçinin arka plandan ayrıştırılmasına da yardımcı olur.

İşaret dili tanıma için sadece el hareketlerinin kullanılacağı durumlarda Leap Motion Kontrolcüsü adında el takibi ve hareket tanıma için özel olarak geliştirilmiş donanımlardan yararlanıldığı da olmuştur [27, 28, 34]. İşaretçiler işaretleri bu donanım karşısında gerçekleştirdiklerinde, parmak eklemlerinin konumları 3 boyutlu olarak tespit

edilebilmektedir. Tespit edilen konumları ifade eden öznitelikler, algılayıcı tabanlı işaret tanımadada olduğu gibi doğrudan sınıflandırmada kullanılabilir [35]. [28]'de LMK kullanarak daha az veri ile RGB görüntüler kullanan yöntemlere yakın doğruluk elde edebildiklerini ifade etmişlerdir. Ancak [34], özellikle yüz ve beden teması gerektiren hareketlerde eller uzaklaştığı ve algılayıcının çalışma alanından çıktığı için günlük işaretlerde LMK kullanımının sınırlı olduğunu ifade etmiştir.

2.1.2. Veri önileme

Geleneksel makine öğrenmesi tabanlı işaret dili tanımadada, görüntünün tamamının sınıflandırmaya aktarılması başarımı düşürebileceği ve işlem yükünü artıracığı için toplanan veri bir önilemeden geçirilir. Takip tabanlı çalışmalarda yürütülen önileme aşamasında, gösterilen işareti temsil edecek en uygun kısımlar (ilgi duyulan alanlar) belirlenir. Çalışmalarda bu bölgeler genellikle el ve yüz olmuştur [36, 37].

Eller ve yüzün tespiti için ten rengi modelleme (*skin color modelling*) yönteminden yararlanılabilir [2, 31, 38]. Bu durumda işaretçiler genellikle koyu bir arkaplan önünde koyu ve uzun kollu giysiler ile işaretleri gerçekleştirmişlerdir [2, 23]. Kişilerin ten renginin farklı tonlarda olması, çekimin yapıldığı ortamın ışıklandırma koşulları gibi durumlar, tespit işini zorlaştırmaktadır.

Ellerin tespiti için hareket, renk, şekil, büyüklük, konum, yön, diğer uzuvlara olan uzaklık gibi farklı görsel işaretçilerden de yararlanılabilir. Bu işaretçiler kullanılarak yapılan tespit işlemi, farklı sınırlamaları da beraberinde getirilmiştir. Örneğin [31]'de tespit edilen el konumlarından en soldaki her zaman sol el, en sağdaki her zaman sağ el olarak ifade edilmiş, ellerin çaprazlanabileceği durumu göz ardı edilmiştir. Benzer şekilde, işaretçinin ellerinin sürekli hareket halinde olması [38], kafanın sabit bir konumda tutulması [39], ellerin ve yüzün birbirlerini kapatmasının önüne geçmek için işaretlerin doğal olmayan şekillerde yapılması gibi sınırlamalar da görülmüştür.

Kinect kamerası, RGB görüntülerin yanı sıra derinlik ve eklemlerin konumunu gösteren iskelet (*skeleton - pose*) verisi de sunabilmektedir. Bu durum, araştırmacıların önileme aşamasında ilgi duyulan alanları tespit ve takip etmesini kolaylaştırmıştır. [26], Kinect tarafından sunulan iskelet verisini elleri ve yüzü tespit etmede kullanmıştır.

Ellerin şeklinin belirlenmesi *görünüm tabanlı* olarak 2 boyutlu yapılabileceği gibi, farklı açılarda kayıt yapan kameralar yardımı ile *model tabanlı* 3 boyutlu olarak da gerçekleştirilebilir. Çalışmalarda işleme kolaylığı, basitlik ve hesaplama maliyetinin düşüklüğü nedenlerinden ötürü görünüm tabanlı metotlara daha sık başvurulmuştur [21].

Tespit ve takip işini kolaylaştırmak ve parmakların birbirlerini ve yüzü kapamaları durumunda daha başarılı bir ayırıştırma yapabilmek için bazı çalışmalarda işaretçilerin eldivenler giydikleri görülmüştür. Burada kullanılan eldivenler, algılayıcı tabanlı işaret dili çalışmalarında kullanılanlardan farklı olarak üzerinde algılayıcı bulundurmeyen, renkli eldivenlerdir. Bazı çalışmalarda sağ ve sol el için iki farklı renkte eldiven kullanılırken [7], bazılarında her parmak için farklı renkler kullanılmıştır [13, 30].

2.1.3. Öznitelik çıkarımı

Öznitelik çıkarımı, geleneksel makine öğrenmesi tabanlı işaret dili tanımanın önemli bir aşamasıdır. İlgi duyulan bölgeler tespit edilip ayırıştırıldıktan sonra bu bölgeleri temsil edecek en uygun öznitelik kümesinin belirlenmesi işine öznitelik çıkarımı adı verilir.

Öznitelik çıkarımı için ön işleme sonrasında elde edilen görüntüye göre bölge tabanlı ve yüzey tabanlı niteleyiciler kullanılabilir. Bölge tabanlı niteleyicilerde ön işleme sonrası tespit edilen ellerin dış sınırlarını içeren görüntüleri kullanarak bölgenin alanı, orta noktası, ağırlık merkezi, sınırlayıcı kutunun genişlik ve yüksekliği gibi bilgiler kullanılır [2]. Ancak bu niteleyiciler gürültüden daha çok etkilendikleri ve parmakların şekli, elin yönü gibi bilgileri sunamadıklarından yüzey tabanlı niteleyiciler daha çok tercih edilmiştir [21].

Yüzey tabanlı yöntemde, görüntünün tamamı ya da ön işleme sonrasında ayırıştırılan bölümler üzerinden öznitelik çıkarımı gerçekleştirilir. Ölçekten bağımsız öznitelik dönüşümü (SIFT) (*Shift-invariant feature transform*) [40], LBP (*Local binary patterns*), DCT (*Discrete cosine transform*), DTW (*Dynamic time warping*) [30, 41], Fourier descriptors [38, 39] gibi niteleyiciler yardımı ile görüntüleri temsil edecek olan öznitelikler çıkarılır.

Öznitelik çıkarımında kullanılan metodun başarımı sınıflandırıcının başarımına da etki edecektir. [40], SURF (*Speeded up robust features* - Hızlandırılmış gürbüz öznitelikler) ile daha hızlı öznitelik çıkarımı yapılabildiğini ancak SIFT ile çıkarılan özniteliklerin ışıklandırma ve kamera açısı gibi etkenlerden daha az etkilendiğini ifade etmiştir.

Çalışmalarda PCA (*Principal Component Analysis*) [42] gibi yöntemler yardımı ile öznitelik azaltma yoluna gidildiği de görülmüştür. Öznitelik azaltma, sınıflandırıcının işlem yükünü azaltırken aynı zamanda işareti temsil eden en uygun öznitelik kümesinin seçilmesini de sağlar.

2.1.4. Sınıflandırma

Sınıflandırma, bir algoritmanın kendisine verilen girdinin bu girdiye en uygun çıktı sınıfına atanmasını sağlayan fonksiyonun belirlenmesi olarak tanımlanabilir. İşaret dili tanımada sınıflandırıcılar, kendisine verilen öznitelikler doğrultusunda bu örneklerin hangi işarete ait olduğunu belirlemede kullanılırlar.

Görme tabanlı işaret dili tanımının ilk çalışmalarında ve algılayıcı tabanlı işaret dili tanımada yapay sinir ağlarına (YSA) daha yoğun olarak başvurulduğu görülebilir [39]. Dinamik görme tabanlı işaret dili çalışmalarında (videolar üzerinden işaret dili tanımada) ise en çok karşılaşılan sınıflandırıcı saklı Markov modelleridir (SMM) [2, 38, 42, 43]. Çalışmalarda bunun nedeni olarak saklı Markov modellerinin sıralı veriyi modellemede başarılı olmaları gösterilmiştir [13, 31]. Bu iki sınıflandırıcının yanı sıra Destek Vektör Makineleri (*Support Vector Machines* - SVM) [41], Rastgele Orman (*Random Forests*) [24, 28] gibi sınıflandırıcılar kullanılarak yapılmış çalışmalar da bulunur.

2.2. Derin Öğrenme Tabanlı İşaret Dili Tanıma

Geleneksel makine öğrenmesi tabanlı işaret dili tanıma çalışmalarına benzer olarak derin öğrenme tabanlı görsel işaret dili tanıma çalışmalarının da ilk aşaması veri toplama ve ön işleme aşamalarıdır. Ancak derin öğrenme tabanlı işaret dili tanıma çalışmalarında toplanan veri genellikle evrişimsel sinir ağlarına aktarılır ve öznitelik çıkarımı kullanıcı müdahalesi olmadan

gerçekleştirilir. Toplanan verinin kipine göre farklı derin öğrenme mimarilerinden yararlanılarak tanıma yapılabilir.

[44]'ün ImageNet görüntü sınıflandırma yarışmasında evrimsel sinir ağları kullanarak elde ettikleri başarımlar sonrasında görüntü işleme çalışmalarında derin öğrenme kullanımı yaygınlaşmaya başlamıştır. Büyük miktarda işaretlenmiş veri ve yüksek işlem gücüne sahip grafik işlem birimlerinin (GİB) (*Graphical Processing Unit - GPU*) yanı sıra, başarımlar ve beraberinde gelen yazılımsal gelişmeler derin öğrenme çalışmalarını hızlandırmıştır.

Evrimsel sinir ağlarının resim sınıflandırmadaki başarısı üzerine, benzer modeller statik işaret dili tanıma amacı ile de kullanılmaya başlanmıştır. [8], derinlik algılayan bir kamera aracılığı ile topladıkları Amerikan işaret dilindeki parmak harfleri ve rakamlardan oluşan statik 31 işareti AlexNet [44] derin öğrenme mimarisine benzer bir evrimsel sinir ağı ile % 85,5 doğrulukla sınıflandırmışlardır.

Bir diğer çalışmada, [45], işaret dilinde kullanılan ağız şekillerini tanımak için evrimsel sinir ağlarını videolar üzerinden zayıf gözetimli bir biçimde eğitmiş ve kareleri tek tek işaretlemeye gerek kalmadan eğitim yapmışlardır. Çalışmada, Imagenet üzerinde eğitilmiş ağırlıkları kullanan GoogleNet derin öğrenme mimarisindeki [46] bir evrimsel sinir ağı ile 12 farklı ağız hareketini en yüksek başarımla (*State of the Art*) ile tanıyabilmişlerdir.

Statik görüntülerdeki uzamsal veriden öznitelik çıkarımı için evrimsel sinir ağları yüksek başarımla vermektedir. Ancak videolarda, resimlerdeki 2 boyutun (uzamsal) yanı sıra zaman boyutunun da değerlendirilmesi gerekir. Aksiyon tanıma gibi bazı alanlarda, videonun herhangi bir karesindeki veri, sınıfın belirlenmesi yeterli olabilmektedir. Örneğin; spor alanında geliştirilmiş bir veri kümesinde, bir karedeki futbol topu, yapılan sporun tespit edilmesinde yeterli olabilir. Ancak işaret dili gibi alanlarda, tek bir kare üzerinden sınıflandırma mümkün olmayacaktır. Bu gibi uygulamalarda uzamsal veri kadar zamansal veri de değerlidir.

Videolar üzerinde derin öğrenme kullanılan ilk çalışmalarda, zamansal verinin değerlendirilmesi için farklı yöntemlerle video kareleri arasındaki değişimler modellenmeye çalışılmıştır. İki akışlı uzam-zamansal ağlarda zamansal verinin

modellenmesi için optik akıştan (*optical flow*) yararlanılmıştır [47]. [48]'de, iki evrişimsel sinir ağından birinde videolardaki uzamsal öznitelikleri, diğerinde optik akış ile video kareleri arasındaki değişimi ifade eden öznitelikleri çıkarmışlardır. UFC-101 hareket tanıma veri kümesinde uzamsal öznitelikler ile %73, zamansal öznitelikler ile %83,7 doğruluk elde etmişler, iki ağın özniteliklerini birleştirdiklerinde ise %88 doğruluğa ulaşmışlardır.

[49]'da araştırmacılar işaret dili videolarındaki zamansal veriyi değerlendirebilmek için video karelerini yan yana birleştirip 2 boyutlu bir görüntü elde etmişler ve bir evrişimsel sinir ağı ile öznitelik çıkarımı yapıp işaret dili tanıma gerçekleştirmişlerdir. Çalışma sonucunda kendi geliştirdikleri 6 sınıflık işaret dili veri kümesinde %86'lık doğruluk elde etmişlerdir. Bu yöntemde her ne kadar videodaki çok sayıda kare kullanılmış olsa da zamansal verinin modellenmesinde uygun bir yöntem olarak görülmemiştir.

[50]'de video karelerinden çıkarılan öznitelikler farklı şekillerde birleştirilerek zamansal verinin modellenmesi hedeflenmiştir. Çalışma sonucunda öznitelikleri birkaç katmanda birleştiren yavaş birleştirme şeklinin en iyi sonucu verdiğini ve videolar üzerinden aksiyon tanıma işi için evrişimsel sinir ağlarının kullanılabileceğini ifade etmişlerdir. Bu çalışmayı temel alan [33], evrişimsel sinir ağları kullanarak İtalyan işaret dilinde geliştirilmiş veri kümesinde işaret dili tanıma gerçekleştirmişlerdir. Çalışma sonunda %91,7 kullanıcı bağımsız, %95,68 kullanıcı bağımlı doğrulukla işaret dili tanıma yapabilmiş ve evrişimsel sinir ağlarının uzam-zamansal veriyi modelleme ve genelleme yeteneğinin işaret dili tanımaya katkı sağlayacağını ifade etmişlerdir.

Zamansal verinin değerlendirilmesi için bir diğer yaklaşımda sürekli işaret dili tanıma amacı ile geleneksel makine öğrenmesi tabanlı işaret dilinde sıklıkla kullanılan saklı Markov modelleri ile evrişimsel sinir ağları birlikte kullanılmıştır [51]. Video karelerinden evrişimsel katmanlar yardımı ile çıkarılan uzamsal öznitelikler bir saklı Markov modeline aktarılmış ve zamansal verinin modellenmesi sağlanmıştır. Bu sayede birden çok sürekli işaret dili tanıma veri kümesinde en yüksek başarımlar elde edilebilmiştir.

Saklı Markov modellerinin sıralı verideki başarısı, popüler bir derin öğrenme mimarisi olan uzun kısa-süreli bellek (UKSB) (*LSTM - Long Short Term Memory*) [52] ile de sağlanabilir. Çin işaret dilindeki 40 işareten oluşan bir veri kümesini kullanan [53] de

evrişimsel katmanlar ile elde edilen uzamsal öznitelikler UKSB katmanlarına aktararak zamansal verinin modellenmesi gerçekleştirilmiştir. Ayırık işaret dili tanıma amaçlanan çalışmada, %96,52 doğruluk elde edilmiştir.

[54], Kinect kamerası ile elde ettikleri Çin işaret dilindeki 100 ve 500 sınıflık veri setlerinde bilek ve dirsek eklem noktalarını UKSB ile değerlendirerek sırası ile %86,2 ve %63,3 doğruluk elde etmişlerdir. Çalışma sonucunda UKSB'lerin sıralı veriyi modellemede SMM'lere göre daha başarılı olduğunu ifade etmişlerdir.

Uzam-zamansal verinin modellenebilmesi için geliştirilen bir diğer yöntem de 3 boyutlu evrişim modelleridir [55-57]. Bu modellerde 2 boyutlu evrişim işleminde kullanılan 2 boyutlu filtreler yerine 3 boyutlu filtreler kullanılarak zaman boyutunda da evrişim gerçekleştirilir. [14]'de araştırmacılar, manuel olarak çıkarılan öznitelikleri sınıflandırmak için kullandıkları saklı Markov modelini, geliştirdikleri 3 boyutlu evrişim yapabilen bir sinir ağı modeli ile karşılaştırmışlardır. SMM ile elde ettikleri en yüksek doğruluk %90,8 iken, 3 boyutlu evrişimsel sinir ağı ile %94,2 doğruluğa erişebilmişlerdir.

Derin öğrenme modelleri büyük veri kümelerine ihtiyaç duyarlar. Ancak işaret dili tanıma için yakın zamana kadar çok büyük veri kümelerinin toplanması mümkün olmamıştır. Bu nedenle, çok derin modellerin eğitilebilmesi için farklı veri setlerinde eğitilmiş modellerin ağırlıklarından transfer öğrenme (*transfer learning*) yapılmış ya da yarı gözetimli eğitim (*semi-supervised learning*) yoluna gidilmiştir.

[12]'deki araştırmacılar, ayırık işaret dili tanıma amacı ile I3D modelini [56] kullanarak yaptıkları çalışmalarında bu modelin işaret dili tanıma için uygun olduğunu ifade etmişlerdir. Çalışmada yaptıkları deneylerde I3D ile çıkarılan uzam-zamansal öznitelikler önceden eğitilmiş VGG modeli [58] ile çıkardıkları uzamsal öznitelikler ve UKSB ile çıkarılan zamansal özniteliklere büyük oranda üstünlük göstermektedir. Çalışma sonunda, eğitilen modelin ağırlıklarını da transfer öğrenmede kullanılması için sunmuşlardır.

Geleneksel sürekli işaret dili çalışmalarında işaretlerin tanınabilmesi için öncelikle videodaki anlamlı işaretlerin belirlenmesi ve hareket geçişlerinin elenmesi gerekmektedir. Ancak diziden diziye (*sequence-to-sequence, seq2seq*) derin öğrenme modelleri ile birlikte işaretler arasındaki hareket geçişlerinin belirlenmesine gerek kalmamaktadır. Bu modeller

işaret dili videosu ile birlikte işaretlerin ne anlama geldiğini belirten metni birlikte girdi olarak alarak işaret dili tanıma ve çevirisi yapabilirler. [59]'da araştırmacılar makine çevirisi amacı ile geliştirilen bir diziden diziye mimari [60] kullanarak işaret dili çevirisi yapmışlardır. Çalışma sonundaki nitel değerlendirmede, işaret dili videolarından metne yapılan çevirinin birebir olmasa da verilmek istenen bilgiyi doğru bir şekilde ilettiğini ifade etmişlerdir.

Sürekli işaret dili tanıma hedefleyen bir diğer çalışmada, evrimsel katmanlar ve iki yönlü bir uzun kısa süreli bellek (UKSB) mimarisi (Bi-LSTM) kullanarak uçtan uca bir işaret dili tanıma sistemi geliştirilmiş ve yüksek başarımla elde edilebilmiştir [61].



3. MATERİYAL VE METODLAR

İşaret dili tanıma için geçmiş yıllarda geleneksel makine öğrenmesi yöntemleri kullanılırken günümüzde sıklıkla derin öğrenme modellerine başvurulmaktadır. Dolayısıyla bu bölümde, makine öğrenmesi ve derin öğrenme yöntemleri açıklanmıştır. Günümüzde kullanılan derin öğrenme modellerinin temelini teşkil eden yapay sinir ağları ve bu ağların bileşenleri irdelenmiştir. Derin öğrenme yöntemlerinin kullanıldığı çalışmalardaki ortak yönler değerlendirildikten sonra tez kapsamında kullanılan evrimsel sinir ağları ve UKSB mimarileri detaylandırılmıştır.

3.1. Makine Öğrenmesi

Makine öğrenmesi T. Mitchell tarafından; bir bilgisayar yazılımının T görevini yapmadaki P performansını E deneyimlerini kullanarak iyileştirmesi, bu yazılımın adı geçen T görevini, P performans ölçütleri doğrultusunda E deneyimlerinden öğrenmesi olarak tanımlanmıştır [62]. İşaret dili tanıma bağlamında bakılacak olursa, T: işaret dili işaretlerini ayırt etme ve bir işaretin ne anlama geldiğini ifade etme görevini, E işaret dili videoları veya algılayıcı verilerini, P ise bu görevin hangi doğruluk oranında yapıldığını ifade eder. Makine öğrenmesi için yapılmış bir diğer tanım da bilgisayarların örnek veri veya geçmiş deneyimleri kullanarak bir performans ölçütünü optimize etmek üzere programlanması olarak ifade edilmiştir [63].

Makine öğrenmesi özellikle programlaması uzun sürecek veya ince ayarlar gerektiren problemlerin çözümünde, geleneksel programlama ile çözmesi çok karmaşık olan problemleri çözmede, çözümün sürekli güncellenmesi gereken durumlarda ve çok büyük verinin olduğu durumlarda verideki karmaşık desenleri bulma ve bu desenlerden öngörüler çıkarmada oldukça yararlıdır [64].

Makine öğrenmesi algoritmaları, sahip oldukları parametreler yardımı ile öğrenme gerçekleştirirler. Algoritmanın kendisine verilen veriyi kullanarak parametrelerini tekrarlı olarak değiştirmesi ve iyileştirme algoritmalarını kullanarak performansını artırması işine *eğitim* adı verilir. Eğitim aşamasından sonra *değerlendirme aşamasına* geçilir ve ne

düzeyde bir öğrenme gerçekleştirildiği incelenir. Son aşama olan *test aşamasında* ise makine öğrenmesinin daha önce karşılaşmadığı veriler üzerinde testler yapılır.

Öğrenmenin sağlanabilmesi için uygun verilerin makine öğrenmesi algoritmasına sunulması gerekir. Eğitim ve değerlendirme aşamalarında farklı, test aşamalarında farklı veriler kullanılır. Eğitim ve değerlendirme aşamalarında kullanılacak veri, eğitim aşamasında daha fazla olacak şekilde bölünür, kalan kısmı değerlendirmede kullanılır. Test aşamasında ise eğitim ve değerlendirmede kullanılmamış olan farklı bir veri kümesi kullanılır. Test verisinin amacı makine öğrenmesinin öğrenme düzeyini ve farklı veriler üzerindeki genelleme düzeyini belirlemektir. Çalışmalarda sıklıkla uygulanan yöntem, veri kümesinin tümünü %70, %20, %10 veya %60, %20, %20 gibi oranlarda sırası ile eğitim, doğrulama ve test verisi olarak bölmektir [65].

Makine öğrenmesi algoritmalarının başarılı bir şekilde sınıflandırma ve tahmin yapabilmesi için veri kümelerindeki örneklerden öznitelik çıkarımı gerçekleştirilir. Veri setinin toplandığı alandaki uzmanların alan bilgisi kullanılarak, veriler üzerinde birleştirme, dönüştürme, normalleştirme gibi işlemler gerçekleştirilebilir. Veya birden çok niteliğin değerleri kullanılarak yeni bir öznitelik üretilebilir. Eldeki veri kümesinde önemli olan niteliklerin belirlenmesi, önemsizlerin atılması, öznitelik üretme gibi işlemlere *öznitelik mühendisliği* adı verilir.

Genelleme, makine öğrenmesi algoritmasının (modelin) yeni ve görülmemiş veriler üzerinde doğru tahminler yapabilme yeteneği olarak ifade edilir [6]. Eğer eğitilmiş bir model eğitim ve değerlendirme aşamalarında çok iyi performans gösteriyor ancak test aşamasındaki performansı çok düşükse bu durumda *aşırı öğrenmeden (overfitting)* bahsedilebilir. Aşırı öğrenmenin nedeni, veri kümesindeki örnek sayısının azlığı ve veri kümesindeki örneklerin tüm durumları kapsamaması olabileceği gibi, modelin çok fazla parametre içermesi yani kapasitesinin çok büyük olması da olabilir. Modelin kapasitesinin düşük olması durumunda ise veri kümesindeki tüm ilişkiler ve desenler öğrenilemez ve hem eğitim ve doğrulama hem de test aşamasında düşük performans gözlenir. Bu duruma ise *yetersiz (eksik) öğrenme (underfitting)* adı verilir. Optimal bir modelde iki durumun da olması istenmez. Yani hem öğrenmenin hem de genellemenin yüksek olması beklenir [64].

Makine öğrenmesi, yapay zekânın alt alanlarından bir tanesidir. Günümüzde birçok problemin çözümünde makine öğrenmesi ana veya yan rollerde yer alır.

Makine öğrenmesi, öğrenmenin şekline göre *gözetimli (supervised)*, *yarı gözetimli (semi supervised)*, *gözetimsiz (unsupervised)* ve *pekiştirmeli (reinforcement)* öğrenme olmak üzere 4 alt kategoride incelenmektedir.

Gözetimli öğrenmede, veri kümesindeki her bir örnek, örneğin özelliklerini ifade eden özniteliklerin yanı sıra hangi sınıfa ait olduğunu belirten sınıf etiketini de barındırır. Böylelikle örneklerin sınıfları öğrenilerek daha sonra makine öğrenmesi yöntemine verilen bir örneğin hangi sınıfa ait olduğu veya olabileceği belirlenir. Bu yöntemde makine öğrenmesi yöntemine verilen öznitelik-sınıf etiketi ikilileri incelenerek özniteliklerden sınıf etiketini elde edecek bir fonksiyon elde edilmesi amaçlanmaktadır [65]. Sınıflandırma ve regresyon uygulamaları gözetimli öğrenmeye örnek olarak verilebilir. K-en yakın komşu, lineer ve lojistik regresyon, destek vektör makineleri, yapay sinir ağları gözetimli öğrenme algoritmalarıdır. İşaret dili tanımada bir gözetimli öğrenme yöntemi olan sınıflandırma algoritmaları kullanılır.

Gözetimsiz öğrenmede ise veri kümesindeki örnekler etiketsizdir. Buradaki amaç, makine öğrenmesi algoritmasına verilen bir öznitelik vektörünü farklı bir şekilde ifade etmek veya bir problemi çözmek amacı ile gruplandırmaktır. Kümeleme, aykırılık tespiti, görselleştirme ve öznitelik azaltma, ilişki kuralları oluşturma uygulamaları gözetimsiz öğrenmeye örnektir. Kümeleme uygulamalarının amacı, örnekleri belirli özelliklerine göre gruplandırmak iken, öznitelik azaltma uygulamalarında veriyi en iyi şekilde ve mümkün olan en az öznitelikle ifade etmek amaçlanır. DBScan, K-means, PCA, apriori sıklıkla karşılaşılan gözetimsiz öğrenme algoritmalarıdır.

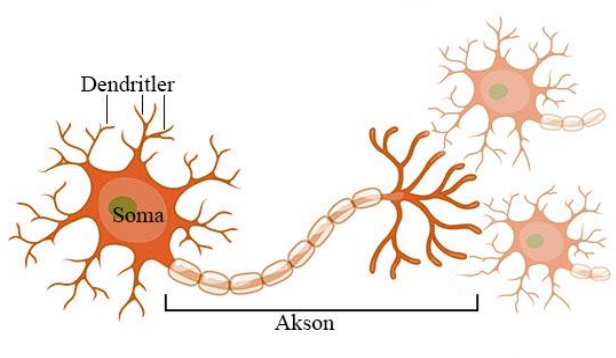
Özellikle sağlık gibi bazı alanlarda makine öğrenmesinde kullanılacak verinin etiketlenmesi güç ve maliyetli olabilir veya çok fazla verinin olduğu durumlarda her bir veriyi etiketlemek zaman alabilir. Bu gibi durumlarda yarı gözetimli öğrenme algoritmalarına başvurulur. Bu algoritmalar etiketsiz veriyi kullanarak bir süre eğitilir ve verideki belli desenleri öğrenirler. Ardından etiketlenmiş veriyi de kullanarak önceki öğrendikleri desenleri etiketler ve öğrenimi devam ettirirler. Bu sayede az sayıdaki etiketlenmiş veri ile daha hızlı ve daha iyi modelleme sağlayabilirler.

Pekiştirmeli makine öğrenmesi yöntemlerinde, makine öğrenmesi algoritması bulunduğu çevredeki durumları inceler ve belli aksiyonlarda bulunur. Bu aksiyon doğrultusunda algoritmaya ödülleri (veya negatif ödül olarak cezalar) verilir ve bir sonraki duruma geçilir. Makine bulunduğu duruma göre en uygun aksiyonu yapmayı öğrenmeye çalışır. Pekiştirmeli öğrenme, problemi çözmek için sıralı bir karar sürecinin gerekli olduğu ve uzun süreli hedeflerin olduğu durumda kullanılır. Robotik uygulamalarında robotun yürümeyi öğrenmesinde ve Go gibi masa oyunlarında kullanımı ile popülerleşen bir yöntemdir.

3.2. Yapay Sinir Ağları

Yapay sinir ağları (YSA), kendisine verilen örnekleri kullanarak olayları öğrenebilen ve sonrasında da çevreden gelen etkilere nasıl tepki verebileceğini hesaplayıp belirleyebilen yöntemler olarak tanımlanmıştır [66].

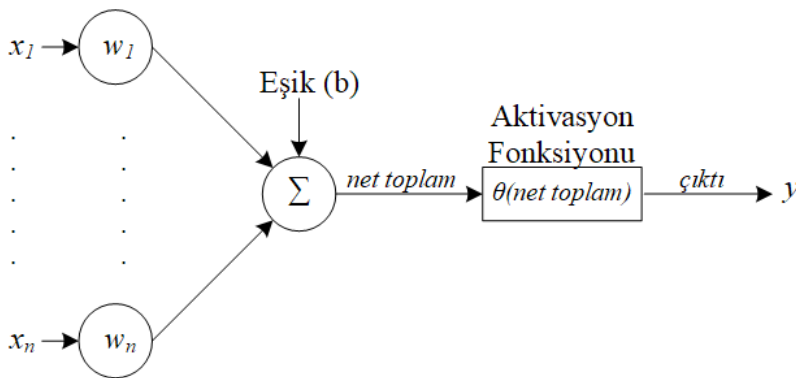
Yapay sinir ağları ile ilgili ilk çalışmalar biyolojik sinir hücrelerinden esinlenerek temellendirilmiştir. Biyolojik sinirler, içerisinde çekirdeği içeren bir sinir hücresi gövdesi (soma), diğer hücrelerden gelen sinyalleri alan dal benzeri yapılar olan dendritler ve hücredeki işlemin sonucunu diğer hücrelere aktaran aksondan oluşur. Biyolojik sinir hücresinin yapısı Şekil 3.1’de verilmiştir. Bir sinir hücresi akson uçları yardımı ile binlerce diğer sinir hücresi ile iletişim halindedir. Sinir hücresine gelen sinyal, belli bir eşik değerin üzerinde ise sonraki sinirlere iletim sağlanır. İletim, sinir hücresinin aksonu ile diğer sinir hücrelerinin dendritleri arasındaki sinaptik boşluklardan aktarılan kimyasallar aracılığı ile gerçekleştirilir.



Şekil 3.1. Biyolojik sinir hücresi

Biyolojik sinir sistemleri, milyarlarca sinir hücresinden meydana gelmektedir. Örnek olarak insan beyninde yaklaşık 10^{11} sinir hücresi bulunmaktadır ve her bir sinir hücresi ortalama 10^4 sinir hücresi ile bağlantılıdır. Her ne kadar biyolojik sinirlerin tepkime süreleri bilgisayarların tepki sürelerinden daha yavaş olsa da, paralel işlem yapabilme yetenekleri nedeni ile biyolojik sinir ağları çok hızlı bir şekilde işlemleri sonuçlandırabilmektedirler. Yapay sinir ağları ile amaçlanan da sinir hücrelerine benzer şekilde dağıtık bir hafızadan yararlanan ve yüksek paralel işlem kapasitesine sahip modeller geliştirebilmektir [67].

Yapay sinir ağlarının temel elemanları olan yapay sinir hücreleri (nöronlar) 5 temel bileşenden oluşur. Bunlar: girdiler, ağırlıklar, toplama fonksiyonu, aktivasyon fonksiyonu ve çıktılarıdır. Biyolojik hücrelerdeki hesaplama ve iletim sürecinin benzeri yapay olarak modellenmiş sinir hücrelerinde de gerçekleştirilir. Yapay sinir ağlarında sinir hücreleri, girdileri alır, toplama fonksiyonları yardımı ile bu girdileri bir hesaplama tabi tutarak hücrenin net girdisi hesaplanır. Hesaplanan net girdi aktivasyon fonksiyonlarına aktararak hücrenin çıktısı belirlenir. Burada, girdiler dendritlerden alınan sinyale, aktivasyon fonksiyonu çekirdekte yapılan işlemlere, toplama fonksiyonu biyolojik hücrede görülen kimyasal eşik değerine, çıktıları ise hücrenin aksonlardan diğer sinir hücrelerine aktarılan sinyale benzetilebilir [66].



Şekil 3.2. Yapay sinir hücresi

Bir yapay sinir hücresi, Şekil 3.2’de verildiği üzere girdileri (x) alır, bunları ağırlıklar (w) ile bir toplama fonksiyonundan geçirir ve net toplam elde edilir. Toplam fonksiyonu, Eş. 3.1’de verilen ağırlıklı toplam şeklinde hesaplanabileceği gibi, çarpım, maksimum,

minimum, çoğunluk veya kümülatif toplam şeklinde de hesaplanabilir [66]. Eşitlikte x_i , sinir hücresinin i . sıradaki girdisini, w_i bu girdiye ait ağırlık değerini ifade eder.

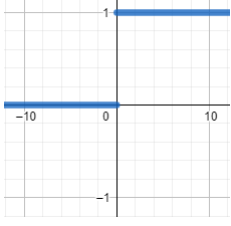
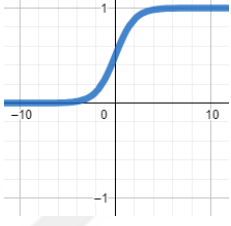
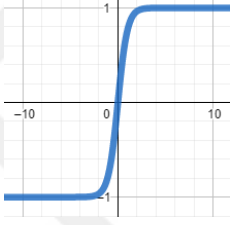
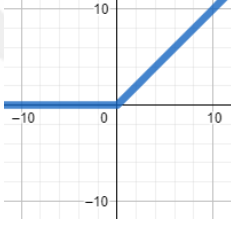
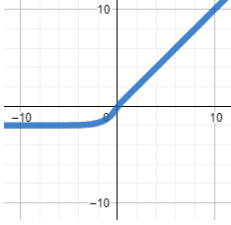
$$net\ toplam = \sum_{i=1}^n x_i w_i = x_1 w_1 + x_2 w_2 + \dots + x_n w_n \quad (3.1)$$

Elde edilen net toplam, nöronun çıktısını hesaplamak üzere bir aktivasyon fonksiyonuna aktarılır. Yapay sinir ağlarında sigmoid, hiperbolik tanjant, adım fonksiyonu, doğrultulmuş lineer birim (*ReLU*), üstel lineer birim (*ELU*) veya sızıntılı doğrultulmuş lineer birim (*Leaky ReLU*) gibi aktivasyon fonksiyonları kullanılır. Yapay nöronun çıktısı, Eş. 3.2'deki gibi hesaplanır. Eşitlikte $y_{nöron}$ nöronun çıktısını, θ aktivasyon fonksiyonunu, b eşik değeri, $\sum net$ ise toplam fonksiyonunun sonucunu ifade eder.

$$y_{nöron} = \theta (b + \sum net) \quad (3.2)$$

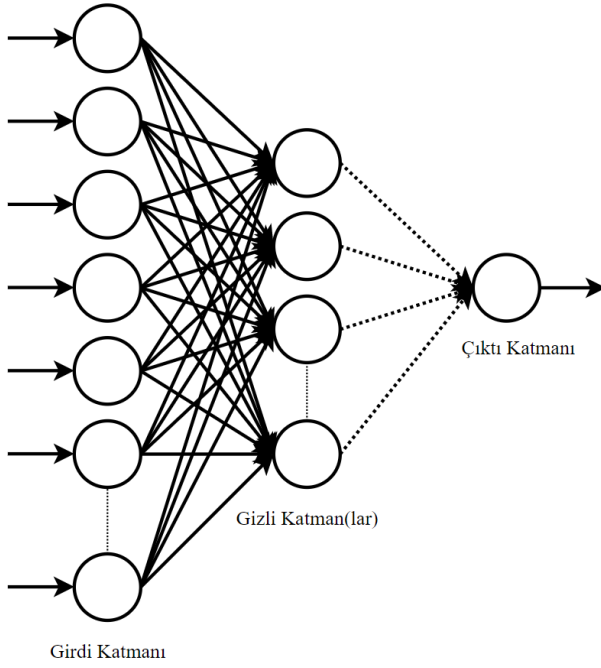
Ağdaki tüm nöronların aynı toplama ve aktivasyon fonksiyonunu kullanması zorunlu değildir [66]. Çizelge 3.1'de yapay sinir ağlarında kullanılan aktivasyon fonksiyonlarına örnekler verilmiştir.

Çizelge 3.1. Yapay sinir ağlarında kullanılan aktivasyon fonksiyonları

Aktivasyon Fonksiyonu	Formülü	Grafiği
Adım	$\theta(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$	
Sigmoid	$\theta(x) = \frac{1}{1 + e^{-x}}$	
Hiperbolik Tanjant	$\theta(x) = \tanh(x)$	
ReLU	$\theta(x) = \max(0, x)$	
Sızıntılı ReLU	$\theta(x) = \max(0.1x, x)$	
ELU	$\theta(x) = \begin{cases} x & x \geq 0 \\ \alpha(e^x - 1) & x < 0 \end{cases}$	

Biyolojik sinir sisteminde olduğu gibi, yapay sinir hücreleri de tek başlarına karmaşık fonksiyonları çözmek için yeterli değildir. Karmaşık problemlerin çözümü için çok katmanlı sinir ağlarından (*MLP – Multilayer Perceptron*) yararlanılır. Çok katmanlı sinir

ağları, çok sayıda yapay sinir hücresinin, katmanlar halinde bir araya getirilmesiyle oluşturulur. Çok katmanlı bir yapay sinir ağı, ağ dışından bilgileri alan bir girdi katmanı, öğrenmeyi gerçekleştiren ve işlemlerin gerçekleştirildiği gizli (ara) katmanlar ve işlemlerin sonucunu beklenen şekilde ağ dışına sunan çıktı katmanından oluşur. Her bir katmandaki yapay sinir hücrelerinin sayısı farklı olabilir. İleri beslemeli çok katmanlı bir yapay sinir ağı yapısı Şekil 3.3’de verilmiştir.



Şekil 3.3. Çok katmanlı bir yapay sinir ağı mimarisi

Tam bağlı (*fully connected*) yapay sinir ağlarında girdi katmanı dışındaki katmanlarda bulunan her yapay sinir, kendinden önceki katmandaki tüm nöronların çıktısını girdi olarak alır. Yapay sinir ağlarında gizli katman sayısı bir veya daha fazla olabilir. Çok katmanlı bir sinir ağının her bir (j) gizli katmanının çıktısı $h_{(j)}(x)$ Eş. 3.3’de formüle edilmiştir.

$$h_{(j)}(x) = \begin{cases} \theta(b_{(j)} + w_{(j)}x) & , j = 1 \\ \theta[(b_{(j)} + w_{(j)}h_{(j-1)}(x))] & , j > 1 \end{cases} \quad (3.3)$$

Yapay sinir ağlarında gizli katmanların ve bu katmanlardaki nöronların sayısı bir yapay sinir ağının hiperparametrelerindendir. Hiperparametre, ağın eğitim esnasında öğrenme algoritması tarafından değiştirilemeyen parametreleri olarak tanımlanmıştır [6].

Hiperparametreler algoritmanın zaman ve bellek karmaşıklığının yanı sıra, öğrenme süresince davranışları ve yeni veri üzerindeki çıkarım performansı gibi birçok duruma etki eder.

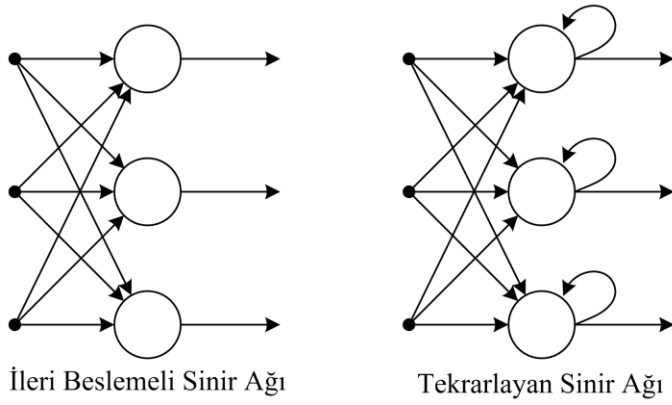
Yapay sinir ağlarının davranışı, ağdaki sinir hücrelerinin sahip olduğu değiştirilebilir, gerçek değerli parametreler (ağırlıklar) tarafından belirlenir. Yapay sinir ağlarının eğitime başlamadan önce genellikle ağırlıklara rastgele küçük değerler atanır. Yapay sinir ağlarında eğitim aşamasının amacı, beklenen sonucu verecek şekilde ağ içerisindeki yapay nöronların ağırlıklarını düzenlemektir. Eğitim aşaması, ağın karmaşıklığına ve probleme bağlı olarak uzun sürebilir [68].

Eğitim aşamasında, ağırlıkların güncellenmesi için ağa bir girdi sunulur ve var olan ağırlıklar kullanılarak ağın çıktısı elde edilir. Bu çıktı ile olması gereken sonuç arasındaki fark bir maliyet fonksiyonu (*cost function*) kullanılarak hesaplanır. Ardından ağın çıktısı ile olması gereken çıktı arasındaki fark ağdaki tüm ağırlıklara yayılır ve başka bir giriş örneği kullanılarak ağırlık güncellemesi işlemine baştan başlanır.

Ağırlıkların güncellenmesinde dereceli azalma (*gradient descent*) algoritması kullanılır. Dereceli azalma algoritmasında amaç, maliyet fonksiyonu ile hesaplanan farkı minimize etmektir. Ağın hiperparametrelerinden biri olan öğrenme hızı (*learning rate*) kullanılarak ağırlık güncelleştirmeleri gerçekleştirilir. Öğrenme hızı, ağırlık güncellemesindeki adım büyüklüğünü belirleyen pozitif bir sayı (0 ile 1 arasında) ile ifade edilir [64]. Öğrenme hızının küçük bir değer seçilmesi ağın yakınsamasını yavaşlatır. Yüksek bir değer seçilmesi durumunda ise adım büyüklüğü fazla olacağından optimal durumun etrafında dönülüp uygun sonuca erişmek mümkün bile olmayabilir. Öğrenme hızının ağın çıktısına göre değiştirilip uygun bir değer belirlenebilmesi için Momentum, Adadelta, Adam gibi uyarlanabilir öğrenme hızı (*adaptive learning rate*) yöntemleri kullanılır.

Ağırlık güncellemesi her bir giriş örneği için yapılabileceği gibi, yığınlar (*batch*) için de yapılabilir. Her bir yığındaki örnek sayısı ağın hiperparametrelerinden bir diğeridir. Eğitim sürecinde ağırlık güncellemesi belli bir tekrar sayısınca (*epoch*) veya belli bir başarıım elde edilene kadar devam ettirilir [68].

Çok katmanlı sinir ağlarının eğitimi için girdi ile birlikte bu girdinin beklenen çıktısı da kullanılır yani gözetimli öğrenme yapısıdır. Bu öğrenme ileri doğru yönlü olabileceği gibi, tekrarlayan (*recurrent*) bir yapıda da olabilir. İleri beslemeli ağlarda yapay sinirlerin çıktısı bir sonraki katmana aktarılırken, tekrarlayan sinir ağlarında nöron çıktısı bir sonraki gelen girdiye eklenerek aynı katmana girdi olarak verilir. Şekil 3.4’de tekrarlayan ve ileri beslemeli çok katmanlı sinir ağı yapıları gösterilmiştir.



Şekil 3.4. İleri beslemeli ve tekrarlayan sinir ağı mimarileri

3.3. Derin Öğrenme

Günlük yaşamda kullanılan algılayıcıların çoğalması ile birlikte her geçen yıl katlanarak artan miktarlarda veri üretilmektedir. Bu verinin büyük bir kısmı yapılandırılmamış durumdadır. Geleneksel makine öğrenmesi yöntemleri, metin, ses, resim, video gibi bu yapılandırılmamış verileri ham hali ile işlemekte yetersiz kalmaktadır. Ham verilerin geleneksel makine öğrenmesi yöntemlerinde kullanılabilmesi için alan uzmanlığı ve mühendislik yeteneklerine sahip bireyler tarafından büyük vakitler ayrılarak bir öznitelik çıkarımı gerçekleştirilmelidir. Makine öğrenmesi yöntemlerinin performansı da bu çıkarılmış özniteliklere doğrudan bağlı olmaktadır [69].

Derin öğrenme, temsili öğrenme olarak isimlendirilen yöntemleri kullanarak amaca uygun öznitelikleri hiyerarşik ve otomatik olarak yapabilen algoritmaları ifade etmektedir. Temsili öğrenme yöntemleri ham veriyi girdi olarak alarak verideki yararlı ilişkileri ve benzerlikleri otomatik olarak çıkarabilirler. Derin öğrenme, makine öğrenmesinin bir alt dalıdır ve günümüzde görüntü işleme, konuşma tanıma, doğal dil işleme, doğal dil çevirisi,

aksiyon tanıma gibi birçok alanda yüksek başarımlar sunmakta, birçok problemin çözümünde kullanılmaktadır [69].

Derin öğrenme ile ilgili farklı tanımlar olsa da tanımların ortak noktası, derin öğrenme modellerinin çok sayıda katmana sahip olması ve bu katmanlar sayesinde büyük miktardaki ham veriden hiyerarşik olarak öznitelikler çıkarabilmesidir [64, 68, 69]. Metin, ses, video gibi doğal veriler hiyerarşik olarak bir araya gelmiş bileşenlerden oluşurlar. Örneğin görüntüler özelinde bakıldığında, işlenmemiş görüntüler piksellerden oluşur. Pikseller bir araya gelerek kenarları, kenarlar nesneleri, nesneler görüntünün genelini oluştururlar. Derin öğrenme modelleri hiyerarşik katmanları yardımı ile pikseller, kenarlar, nesneleri temsil edecek olan öznitelikleri doğrudan veri üzerinden çıkarabilmektedirler.

Bir derin öğrenme mimarisi, öğrenme yeteneğine sahip çok sayıdaki küçük birimin katmanlı olarak bir araya getirilmesi ile oluşturulur. Sıralı olarak bir araya getirilen katmanlar, amaca uygun öznitelikleri belirlerken hiyerarşik olarak aktive olurlar. 5 ile 20 arasında katmana sahip derin öğrenme mimarileri oldukça karmaşık fonksiyonları ifade ederken, verideki en küçük farklılıkları tespit edebilirler [69].

Birçok derin öğrenme mimarisi, sabit boyutlu bir veri örneğini (örneğin bir resim) girdi olarak alıp sabit boyutlu bir çıktı (örneğin bir sınıfa ait olma olasılıkları) veren modellerdir. Derin öğrenme modellerinde aktivasyon fonksiyonu olarak hızlı öğrenme yeteneği nedeni ile genellikle ReLU aktivasyon fonksiyonu veya çeşitleri tercih edilir.

Son 10 yılda, daha çok katmana sahip derin ağların eğitimini mümkün kılan GİB teknolojisinin gelişmesi, daha iyi genelleme özelliği sağlayan algoritmalar ve artan veri miktarı ile birlikte derin öğrenme modelleri daha popüler olmuştur [69, 70]. Özellikle konuşma tanıma konusunda yapılmış çalışmalarda elde edilen başarılı sonuçlar, derin öğrenme konusuna ilgiyi artırmıştır. Bu dönemde evrimsel sinir ağları ile dikkat çeken performanslar elde edilebilmiştir [69].

Derin öğrenme çalışmalarında sıklıkla görülen bir diğer durum da gözetimsiz ve transfer öğrenmedir. Özellikle verinin az olduğu durumlarda, eğitime başlanmadan önce farklı bir veri kümesi kullanılarak gözetimsiz ön öğrenme ile ağın belirli bir kısmı eğitildiğinde, sadece sahip olunan veri kümesi ile yapılan eğitimden daha yüksek performans elde

edildiği görülmüştür [69]. Transfer öğrenme ile de bir uygulama için öğrenilmiş ağırlıklar benzer başka bir uygulamada kullanılabilir. Böylelikle öğrenme daha kısa sürede tamamlanabilir ve daha yüksek başarımlar elde edilebilir.

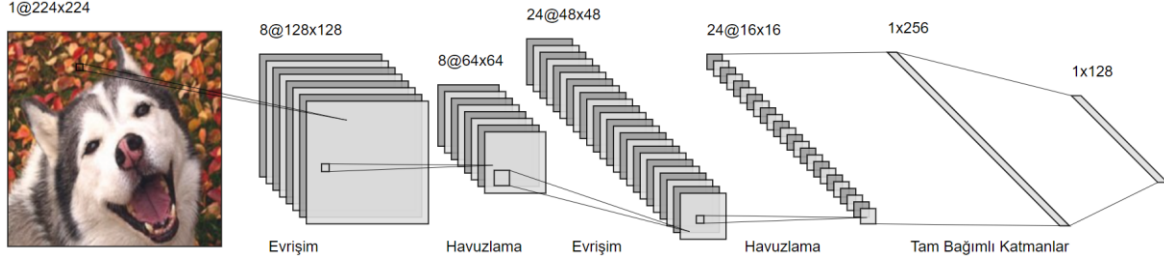
Derin öğrenme modelleri içerisinde Deep Boltzman Machines (DBM), Deep Belief Networks (DBN) gibi farklı mimariler olsa da, bu tez kapsamında görüntü işleme ve işaret dili tanıma alanlarında sıklıkla kullanılan evrişimsel ve tekrarlayan sinir ağlarından bahsedilecektir.

3.4. Evrişimsel Sinir Ağları

Evrişimsel sinir ağları, sesler, resimler, videolar ve volümetrik görüntüler gibi çok dizili verilerin işlenmesi için tasarlanmış derin öğrenme modelleridir. Evrişimsel sinir ağları, doğal dünyadan aktarılmış sinyalleri başarılı bir şekilde modelleme yeteneğine sahip mimarilerdir [69]. Özellikle görüntü sınıflandırma, görüntüde nesne tanıma ve optik karakter tanıma gibi uygulamalarda başarılı sonuçlar vermektedir [71].

Geleneksel yapay sinir ağları veriyi tek boyutlu olarak girdi alırlar. Örneğin 100*100 piksel boyutlarındaki bir görselin sınıflandırılması için geleneksel sinir ağları kullanıldığında girdi 10000 pikselden oluşan bir vektör olacaktır. Bu durumda gizli katmandaki tüm nöronlar tüm girdilerle ilişkili olacağından, verideki uzamsal ilişkiler büyük oranda kaybolmaktadır [63]. Bu nedenle görüntü işlemede öncelikle amaca uygun öznitelik çıkarımı gerçekleştirilir ve çıkarılan öznitelikler sınıflandırma için yapay sinir ağlarına aktarılır. Görüntü işlemede, ana sorunlardan birisi görsel dünyayı başarılı bir şekilde ifade edecek özniteliklerin nasıl çıkarılabileceği olmuştur [71].

Konuşma, görüntü ve dil gibi doğal veriler, hiyerarşik özniteliklerden oluşur. Konuşmamız seslerden kelimelere sonra cümlelere evrilir; görüntüler nesnelerden, nesneler parçalardan, parçalar kenarlardan, kenarlar piksellerden oluşur. Evrişimsel sinir ağları sahip olduğu farklı katmanlar sayesinde, doğal verideki hiyerarşik öznitelikleri ve komşuluk ilişkilerini modelleyebilir ve bu sayede tanıma veya sınıflandırma gerçekleştirebilir [69]. Evrişimsel sinir ağları, biyolojiden esinlenilmiş, veriden değişken hiyerarşik öznitelikleri çıkarabilen eğitilebilir mimariler olarak tanımlanmıştır [71].

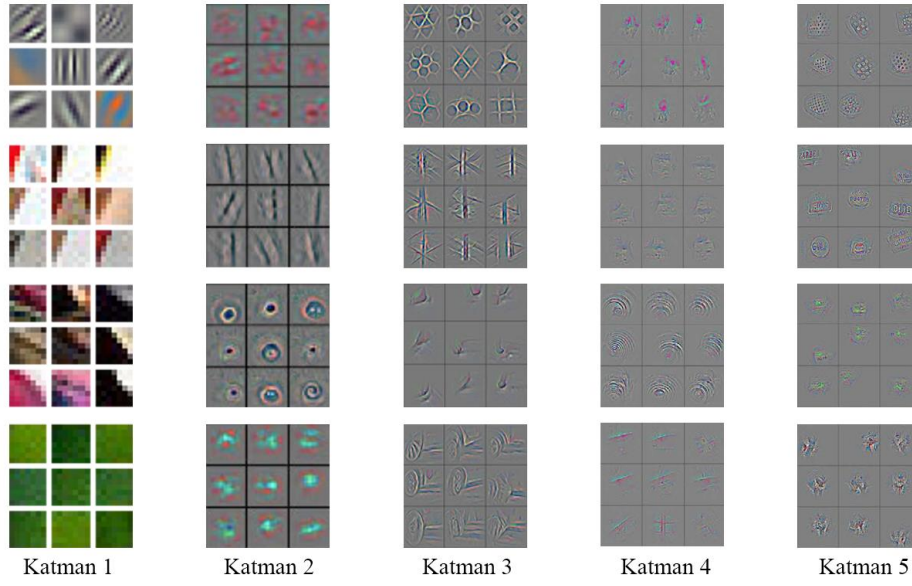


Şekil 3.5. Temel bir evrişimsel sinir ağının katmanları

Bir evrişimsel sinir ağı mimarisi, Şekil 3.5’de sunulduğu üzere çeşitli amaçları gerçekleştirmek için bir araya gelmiş farklı türdeki katmanlardan oluşur. İlk katmanlar iki çeşit katman içerir. Bunlar evrişim (*convolution*) ve havuzlama (*pooling*) katmanlarıdır. Evrişimsel katmanda, bir önceki katmandan alınan girdinin tamamı üzerinde yerel alıcı bölgeler (*local receptive fields, filters*) yardımı ile evrişim yapılır ve evrişim işleminin sonucunda öznitelik haritaları oluşturulur. Bir öznitelik haritasındaki tüm filtreler ortak bir ağırlık havuzunu paylaşırlar, aynı katmandaki farklı öznitelik haritalarının farklı ağırlık havuzları vardır. Bu sayede yerel ilişkiler ortaya çıkarılır, birbirine yakın veri parçaları algılanması kolay gruplar oluşturur ve bulundukları konumdan bağımsız olarak tespit edilebilirler [69].

Evrişim katmanlarında amaç, bir önceki katmanın çıktısındaki yerel ilişkileri tespit etmek iken, havuzlama katmanlarının amacı öznitelik havuzlarını daha küçük boyutta ifade edebilmektir. Bu katmanda her bir öznitelik havuzu kendi başına ele alınır. Havuzlama filtresi ilgilendiği alanın maksimumunu (veya ortalamasını) alarak o bölgeyi ifade edebilir. Havuzlama filtresi tüm girdi üzerinde kaydırılarak girdiden daha küçük bir öznitelik haritası çıkarılır. Yeni öznitelik haritasının boyutu, havuzlama filtresinin boyutuna ve kayma adımına bağlıdır.

Evrişim ve havuzlama katmanları tekrarlı olarak bir araya getirilerek hiyerarşik öznitelik haritaları oluşturulur. Resim 3.1’de görüldüğü üzere her bir katman bir öncekinden daha karmaşık şekilleri ifade edebilmektedir [70]. Görüntü sınıflandırma amacı ile eğitilen bir evrişimsel sinir ağında evrişim ve havuzlama katmanlarını tam bağlı sınıflandırıcı katman veya katmanlar takip eder. Eğitim sürecinde tüm ağ geri besleme ile eğitilerek tüm öznitelik haritalarının ağırlıkları güncellenir [69].



Resim 3.1. Bir evrişimsel sinir ağının farklı katmanlardaki öznitelik haritaları

3.5. Tekrarlayan Sinir Ağları

Evrişimsel sinir ağları verideki yerel ilişkili hiyerarşik özniteliklerin çıkarılmasında etkilidir. Ancak bazı uygulamalarda veriyi sadece uzamsal boyutu üzerinden değerlendirmek ve sınıflandırma yapmak yeterli olmayacaktır. Videonun tek bir karesinden hareketin yönünü çıkarmak mümkün olmayacağı gibi, cümle içindeki bir kelimenin tahmin edilebilmesi için öncesi ve sonrasındaki kelimelerin de bilinmesi gereklidir.

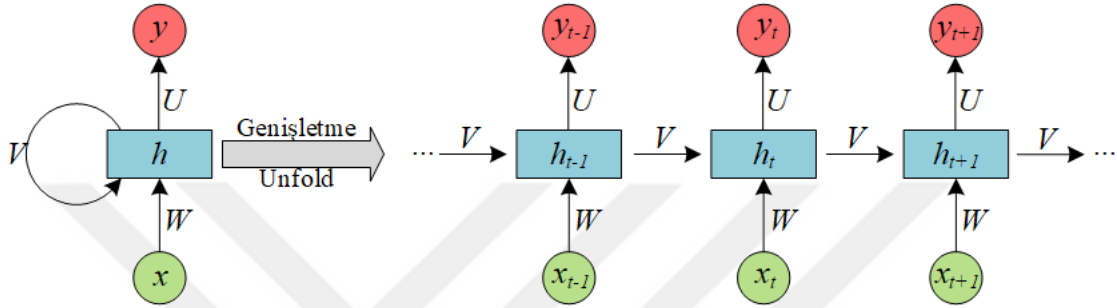
Tekrarlayan sinir ağları (TSA) (*Recurrent Neural Networks - RNN*), konuşma veya zaman serileri gibi sıralı veriler üzerinde etkili bir şekilde çalışabilen sinir ağı mimarileridir. Her katmanda bağımsız işlemler yapılan ileri beslemeli yapay sinir ağlarından farklı olarak, tekrarlayan sinir ağlarında katman içinde bir bilgi alışverişi vardır. Tekrarlayan sinir ağlarında, bir katmanın çıktısı, bir sonraki girdi ile birlikte yeniden aynı katmana girdi olarak aktarılır. Bu sayede, ağıdaki gizli katman, geçmiş tüm veri noktaları arasındaki ilişkiyi tutan bir durum vektörüne sahip olmaktadır [69]. Bu durum, tekrarlayan sinir ağlarına zaman sıralı verideki ilişkileri başarılı bir şekilde modelleme özelliği verir.

$x = (x_1, \dots, x_T)$, şeklinde verilmiş sıralı bir veri için standart bir TSA'da saklı durum vektörü dizisi $h = (h_1, \dots, h_T)$ ve çıktı vektörü dizisi $y = (y_1, \dots, y_T)$, $t = 1$ den T 'ye kadar aşağıdaki eşitlikler aracılığı ile hesaplanır.

$$h_t = \theta (w_{xh} x_t + w_{hh} h_{t-1} + b_h) \quad (3.4)$$

$$y_t = \theta (w_{hy} h_t + b_y) \quad (3.5)$$

Eş. 3.4 ve Eş. 3.5’de w ifadeleri ağırlıkları, b ifadeleri eşik vektörlerini ve θ ise aktivasyon fonksiyonlarını temsil eder.



Şekil 3.6. Bir tekrarlayan sinir ağının genişletilmesi sonucu ortak ağırlıkların gösterimi

Bir TSA’nın tekrarlı işlemleri zaman boyutunda genişletilerek görselleştirildiğinde Şekil 3.6’daki gibi bir yapı elde edilir. Bu nedenle tekrarlayan sinir ağları ortak ağırlıkları paylaşan çok derin ileri beslemeli bir ağ olarak değerlendirilebilirler [63, 69]. Şekildeki W , V ve U ağırlıkları katmanlar arasında paylaşılan ortak ağırlıkları ifade etmektedir.

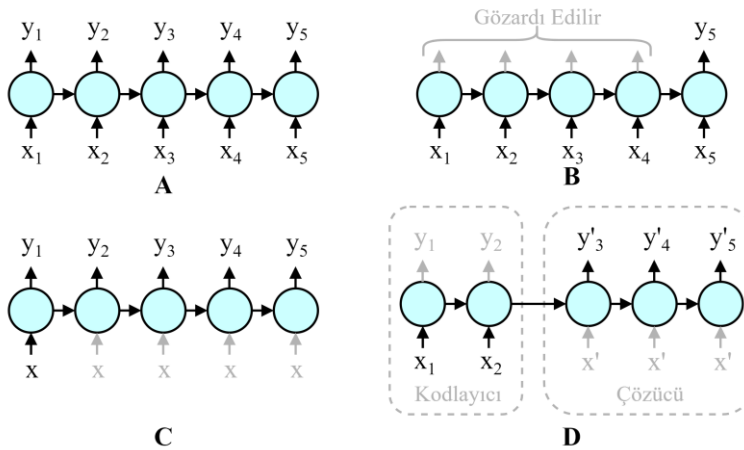
Tekrarlayan sinir ağlarının ileri beslemeli ağlara göre avantajlarından birisi ortak ağırlıkların kullanımı ve bu sayede hesaplamalarda geçmiş verileri de değerlendirebilmesidir [63]. Aynı zamanda birçok tekrarlayan sinir ağı, değişken uzunluktaki girdi boyutları ile de çalışabilir [64]. TSA’ların avantajlarından bir diğeri girdi boyutu değişse de model parametrelerinin sayısının sabit kalmasıdır.

Tekrarlayan sinir ağları girdi ve çıktı türlerine göre farklı mimarilerde oluşturulabilirler [72]. Örneğin diziden diziye (*sequence-to-sequence*) tekrarlı sinir ağları girdi olarak bir dizi alıp çıktı olarak da bir dizi sunarlar. Bu türdeki ağlar, doğal dil işleme çalışmalarında sıklıkla kullanılır ve verilen bir yarım cümlelin tamamlanmasını veya cümle içerisindeki boşlukların doldurulmasını sağlayabilirler.

Girdi olarak bir dizi sunulup son çıktı dışındaki diğer çıktılar göz ardı edilerek diziden vektöre (*sequence-to-vector*) bir tekrarlı sinir ağı mimarisi de oluşturulabilir. Bu sayede örneğin girdi olarak alınan bir metin sonucunda çıktı olarak metindeki duygu belirlenebilir.

Resim yazısı oluşturma (*image captioning*) uygulamalarında ise tekrarlayan sinir ağına girdi olarak bir vektör (görüntü) tekrarlı olarak verilir. Çıktı olarak ise görüntüdeki olayı ifade eden bir metin alınır. Bu tip bir mimari ise vektörden diziye (*vector-to-sequence*) tekrarlayan sinir ağına örnektir.

Tekrarlayan sinir ağlarının bir türevi de kodlayıcı-çözücü (*encoder-decoder*) ağlarıdır. Bu ağlarda, öncelikle diziden vektöre bir ağ (*encoder*) yardımı ile dizi şeklindeki girdilerin vektörle ifadesi sağlanır. Ardından elde edilen vektör bir vektörden diziye ağa (*decoder*) aktarılarak vektörün farklı bir dizi şeklinde ifadesi gerçekleştirilebilir. Burada dönüşümün çıktısı girdi olarak verilen bir metin için başka bir dildeki çevirisi olabileceği gibi, video şeklindeki bir girdi için videodaki olup biteni anlatan metinler de olabilir. Kodlayıcı-çözücü mimarileri, girdi-çıkı ilişkisi bakımından diziden diziye mimariye benzemekle birlikte birçok uygulamada bu mimariden daha yüksek performans sunmaktadır [72]. Girdi ve çıktılar bakımından çeşitli TSA mimarileri Şekil 3.7’de sunulmuştur.

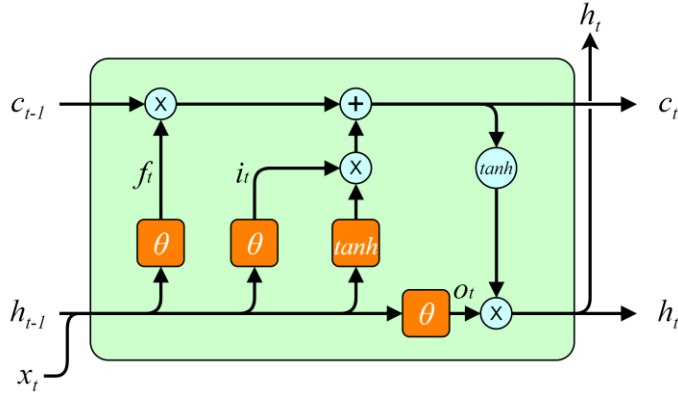


Şekil 3.7. Çeşitli TSA mimarileri A: diziden diziye, B: diziden vektöre, C: Vektörden diziye, D: Kodlayıcı çözücü

TSA’larda ortak ağırlıklar kullanılarak uzun dönemli ilişkileri modellemek hedeflenmiştir. Teorik olarak bu mümkün olsa da, eğitim esnasında, özellikle çok uzun sıralı veriler kullanıldığında geri beslemede hesaplanan gradyanların giderek büyüdüğü ve şiştiği

(*exploding gradients problem*) veya zamanla sifira yaklaşılarak kaybolduğu (*vanishing gradients problem*) görülmüştür [69]. Bu durum verideki uzun süreli bağımlılıkların tespit edilmesinin önüne geçmektedir.

TSA'ların eğitiminde görülen sorunları çözmek ve daha uzun sıralı verileri işleyebilmek için geliştirilen yöntemlerden en bilineni Uzun-Kısa Süreli Bellek (UKSB) (*Long Short Term Memory - LSTM*) modelleridir [73]. UKSB mimarisinin TSA'lara göre ana katkısı, durumunu zamanla koruyabilen bir bellek hücresi ve hücreye giren ve çıkan bilgi akışını düzenleyen doğrusal olmayan geçit birimlerine (*gating units*) sahip olmasıdır.



Şekil 3.8. UKSB birimi ve kapılar

UKSB'ler sahip oldukları özelleştirilmiş hafıza birimleri sayesinde bilgileri daha uzun süreyle saklayıp sıralı verideki geniş aralıklı ilişkileri tespit edebilirler. Bir UKSB birimi, bir hafıza birimi (*cell - c*), bir girdi kapısı (*input gate - i*), çıktı kapısı (*output gate - o*) ve bir unutma kapısından (*forget gate - f*) oluşur [52]. Bir UKSB birimi ve kapıları Şekil 3.8'de sunulmuştur. Verilen bir girdi sonrasında UKSB içerisinde Eş. 3.6 - Eş. 3.10'da verilen hesaplamalar gerçekleştirilir.

$$i_t = \theta_i(w_{xi}x_t + w_{hi}h_{t-1} + w_{ci}c_{t-1}), \quad (3.6)$$

$$f_t = \theta_f(w_{xf}x_t + w_{hf}h_{t-1} + w_{cf}c_{t-1}), \quad (3.7)$$

$$c_t = f_t c_{t-1} + i_t \theta_c(w_{xc}x_t + w_{hc}h_{t-1}), \quad (3.8)$$

$$o_t = \theta_o(w_{xo}x_t + w_{ho}h_{t-1} + w_{co}c_t), \quad (3.9)$$

$$h_t = o_t \theta_t(c_t), \quad (3.10)$$

Eşitliklerde, w iki UKSB birimi arasındaki ağırlıkları, θ aktivasyon fonksiyonunu, i_t , f_t , c_t , o_t ve h_t ise sırası ile t zamanındaki girdi kapısı, unutma kapısı, hafıza birimi, çıktı kapısı ve UKSB biriminin çıktısını ifade eder.

UKSB'ler kullanılarak günümüzde konuşma tanıma, müzik üretimi, makine çevirisi, hareket tanıma gibi alanlarda başarılı sonuçlar elde edilmektedir.



4. VERİ KÜMESİ VE VERİ ÖNİŞLEME

Bu bölümde, tez çalışmasında kullanılan işaret dili tanıma veri kümesinin özellikleri, önışleme ve veri artırma işlemlerinin detayları açıklanmıştır.

4.1. Veri Kümesi

Bu tez çalışmasında, veri kümesi olarak BosphorusSign [19] veri kümesinin ayrıık işaretler içeren bir alt kümesi kullanılmıştır. BosphorusSign veri kümesi, finans, sağlık ve genel olmak üzere farklı amaçları yerine getirmek için geliştirilmiş üç alt veri kümesinden oluşmaktadır.

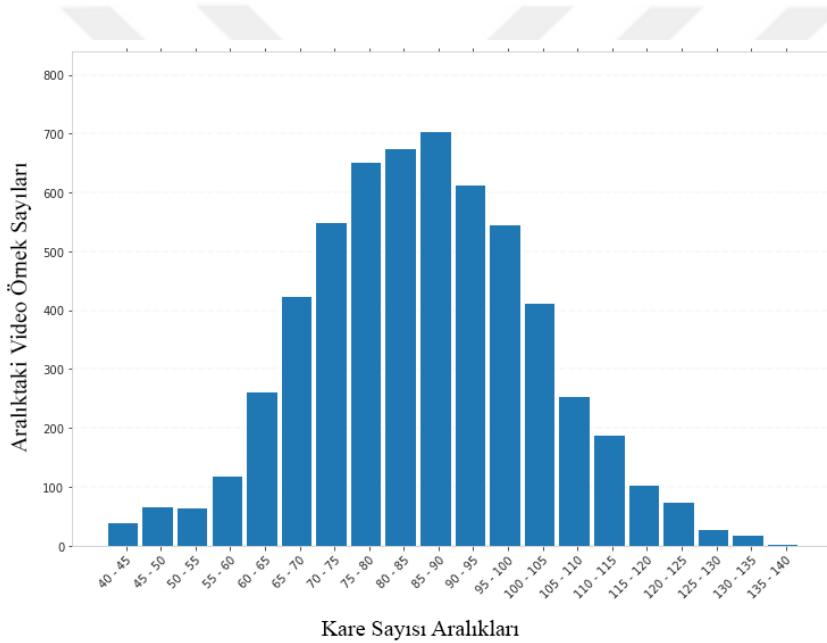
Sağlık alt kümesi, hastaneler ve doktor randevularında; finans veri kümesi bankalarda; genel veri kümesi ise günlük işleri yürütürken sıklıkla kullanılan işaretlerden oluşmaktadır. İşaretlerin belirlenmesinde, işitme engelli bireyler, dil bilimciler ve alan uzmanlarından yararlanılmıştır. Finans alt kümesinde 140, sağlık alt kümesinde 301 ve genel alt kümede 154 sınıf bulunmaktadır. Bu tez çalışmasında, BosphorusSign veri kümesinin 154 sınıflık “genel” alt kümesi kullanılmıştır.

Genel veri kümesi, işitme engelli bireylerin günlük yaşamda daha sık kullandıkları fiil, isim ve ifadelerden oluşmaktadır. Tüm işaretleri içeren liste tez çalışmasının sonunda ek olarak verilmiştir. Genel veri kümesinde toplam 5829 işaret dili videosu bulunmaktadır. Videolar, sabit bir arka plan önünde, stüdyo ortamında 1920x1080 çözünürlük ve saniyede 30 kare içerecek şekilde çekilmiştir. Kayıt cihazı olarak Kinectv2 kullanılmıştır.

Veri kümesinde her bir işaret 6 işaretçi tarafından farklı sayılarda tekrar edilmiştir. Aynı işaretin farklı tekrarlarında işaretçilerin farklı giysiler giydikleri, farklı saç stilleri kullandıkları da görülmüştür. Videolarda, işaretçiler eller yanda olacak şekilde serbest bir durumdan başlayarak işareti gerçekleştirmektedir. İşaretin gösterimi bittiğinde yeniden serbest duruma dönülerek kayıt sonlandırılmıştır.

Veri kümesini oluşturan videolar, her bir işarete en az 29 tekrar içermektedir. Video uzunlukları ise 2 ile 5 saniye arasında değişen uzunluklardadır. Videoların kare sayılarını ifade eden histogram Şekil 4.1’de verilmiştir.

Kullanılan veri kümesi işaret dili tanıma çalışmalarında kullanılmak üzere tasarlandığından, eğitim ve test aşamalarında kullanılacak kısımların belirlenmesi verileri sağlayan Boğaziçi Üniversitesi Algısal Zeka Laboratuvarı tarafından yapılmıştır. Testlerin yapılması için 4. işaretçinin videolarının kullanılması, kalan kısmın eğitim ve doğrulama için kullanılması veri sağlayıcı tarafından önerilmiştir. Bu çalışmadaki tüm deneylerin eğitim ve test aşamaları veri sağlayıcının bu önerileri doğrultusunda gerçekleştirilmiştir.



Şekil 4.1. Videolardaki kare sayılarını ifade eden histogram

4.2. Veri Önışleme ve Veri Artırma

Veri önışleme, verinin makine öğrenmesi modeline girdi olarak sunulabilmesi için verinin temizlenmesi, boyutunun değıştirilmesi, renk dönüşümü yapılması, kırılması, öznetelik mühendisliğı gibi süreçleri ifade eder. Evrişimsel sinir ağıları, belirli bir girdi boyutu ile çalışır. Bunu sağlamak için de çalışmalarda, veri setinde bulunan görüntüler üzerinde veri önışleme işlemleri yürütölür.

Derin öğrenme modelleri, fazla sayıdaki model parametreleri nedeni ile aşırı öğrenmeye (*overfitting*) müsaittirler. Bunun önüne geçmek için büyük miktarlarda veri ile birlikte seyreltme (*dropout*) [74] ve yığın normalizasyonu (*batch normalization*) [75] gibi düzenleme (*regularization*) teknikleri kullanılır.

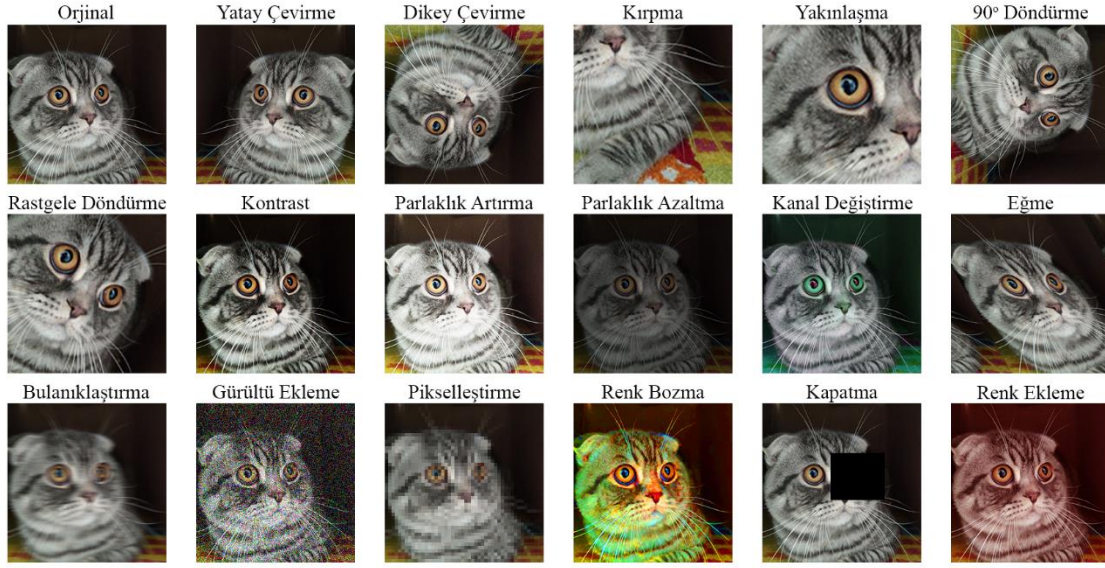
Birçok durumda, büyük miktarda veri toplamak ve bunları işlemek mümkün olmayabilir. Örneğin sağlık alanında, verinin temin edilebilmesi için yüksek maliyetli cihazların çalıştırılması, elde edilen örneklerin doktorlar veya radyoloji uzmanlarınca incelenmesi ve etiketlenmesi gerekir. Maliyetlerin yanı sıra nadir görülen hastalıklarda veri sağlayacak hastaların bulunması, hasta mahremiyeti gibi sorunlarla da karşılaşılabilir. Bu gibi çok sayıda örnek toplamanın mümkün olmadığı durumlarda veri artırma yöntemlerine başvurulur. Veri artırma, veri kümesindeki örnekler üzerinde örneğin sınıf etiketini farklılaştırmayacak değişikliklere gidilerek, modelin eğitiminde kullanılacak örnek sayısının artırılması işlemini ifade eder [35].

Bir görüntü sınıflandırma uygulaması gerçek dünya kullanımında, kamera açısı, ışıklandırma, kamera kalitesi, nesnenin başka nesnelerce kapatılması, farklı arka planlar, kamera uzaklığı gibi birçok farklı problemi aşabilmelidir. Ancak makine öğrenmesi yönteminde kullanılacak verinin toplanması esnasında tüm bu durumlar dikkate alınmamış olabilir. Veri artırma yardımı ile gerçek dünyada görülebilecek farklı senaryolar yapay olarak veri kümesine dâhil edilebilir.

[44]'de araştırmacılar, veri artırma sayesinde veri kümesi büyüklüğünü 2048 katına çıkardıklarını, bu sayede modelde değişikliğe gidilmeden aşırı öğrenmenin önüne geçilebildiğini ifade etmişlerdir.

4.2.1. Resimlerde veri artırma yöntemleri

Derin öğrenme ile görüntü sınıflandırma çalışmalarında resimler üzerinde basit morfolojik düzenlemeler ile yapılan veri artırma sıklıkla kullanılan bir yöntemdir. Statik görüntülerde, kaydırma (*shift*), döndürme (*rotate*), yamultma (*shear*), yakınlaştırma (*zoom*), çevirme (*flip*), renk yoğunluğunu ve parlaklığı değiştirme gibi işlemler kolayca ve yüksek hesaplama maliyeti gerektirmeden yapılabilir.



Resim 4.1. Resimler üzerinde yapılan veri artırma örnekleri

Statik görüntüler üzerinde yapılmış derin öğrenme çalışmaları incelendiğinde görüntüye gürültü ekleme, karıncalandırma, bulanıklaştırma, renkleri ters çevirme, yamultma, rastgele bölge karartma, renk uzayı dönüşümleri gibi görüntü işleme yöntemlerinin kullanıldığı da görülmüştür. Örnek bir görüntü üzerinde veri artırımı teknikleri uygulanması ile elde edilen sonuçlar Resim 4.1’de verilmiştir.

Çalışmalarda daha ileri veri artırma yöntemi olarak derin öğrenme mimarileri kullanılarak yapay veriler oluşturma yoluna gidildiği de görülmüştür. [76]’de bir derin öğrenme mimarisi olan çekişmeli üretici ağlar (*Generative Adversarial Networks - GAN*) kullanılarak oluşturulan görüntüler duygu tespiti amacı ile kullanılmış ve bu görüntülerin doğruluğa %5-%10 arasında olumlu etki gösterdiği ifade edilmiştir.

Bazı uygulama alanlarında verinin yapay olarak oluşturulması daha kolay olabilir. [77]’de araştırmacılar, bilgisayar destekli tasarım yöntemleri kullanarak ürettikleri veriyi nesne tespiti amacı ile kullanmışlardır. Bilgisayar destekli tasarım yardımı ile nesneyi görüntünün istenen bir noktasına farklı açılar ve farklı arka planlar kullanarak yerleştirebildikleri için ayrıca bir etiketlemeye ihtiyaç kalmadığını ifade etmişlerdir. Çalışma sonucunda yapay veriler ile farklı alanlardan gelen gerçek resimlere göre daha iyi sonuçlar elde ettiklerini, nesne tanıma uygulamalarında eğer yeni bir etiket sınıfı eklenecekse bu yöntemin daha zahmetsiz olduğunu belirtmişlerdir [77].

4.2.2. Videolarda veri artırma yöntemleri

RGB kameralar ile elde edilen statik görüntülerde veri, görüntünün genişlik ve yüksekliği olmak üzere 2 boyutla ifade edilir. Bu 2 boyutlu veri, çalışmalarda uzamsal (*spatial*) veri olarak adlandırılmaktadır. Ancak videolarda verinin 3. boyutu olarak zaman boyutu da vardır. Videodaki karelerin zamanla değişimi, videonun zamansal verisini oluşturur.

2 boyutlu statik görüntülerde kullanılan uzamsal veri artırma yöntemleri videolar üzerinde de kullanılabilir. Video üzerinde uzamsal veri artırma yöntemleri uygulanırken dikkat edilmesi gereken bir durum, kaydırma, yaklaşma, döndürme açısı gibi veri artırma parametrelerinin videonun tüm karelerinde eşit olması gerektiğidir. Örneğin videonun ilk karesinde 10 derecelik bir açı ile döndürme işlemi yapılmış ise sonraki karelerde de aynı açı ile döndürülmesi, bir kaydırma veya yakınlaştırma işlemi yapılmışsa yine aynı değerlerle yapılması gerekir. Aksi durumda, videodaki zamansal veri kaybolacak veya bozulacaktır.

Hareket tanıma gibi bazı uygulamalarda videolardaki uzamsal veri artırma yöntemlerinin seçimi daha özenli yapılmalıdır. Bu gibi çalışmalarda yön belirten hareketlerde görüntünün çevrilmesi veya döndürülmesi hareketin anlamını değiştirebilir. Örneğin sola kaydırma işaretini yatayda ters çevirdiğinizde işaret sağa kaydırma olarak değişecektir. Bu durumda işaretin etiketini değiştirmek gerekebilir.

Videolarda, uzamsal veri artırımının yanı sıra, videonun zamansal verisi üzerinde de değişiklikler yaparak veri artırımına gidilebilir. Videonun saniyedeki kare hızının değiştirilmesi, zaman boyutu üzerinde kırpma yapılarak gidilmesi, karelerin tekrar edilmesi gibi işlemler videoda zamansal veri artırma yöntemlerindendir.

İşaret dili çalışmalarında veri artırma konusunda en sık karşılaşılan veri artırma yöntemleri kırpma ve zamansal örnekleme olarak ifade edilmiştir [29]. Bu tez çalışmasında videoda uzamsal veri artırma yöntemi olarak görüntünün kaydırılması ve döndürülmesi, zamansal veri artırma yöntemi olarak da kareler üzerinde eşit aralıklı örnekleme yöntemi kullanılmıştır.

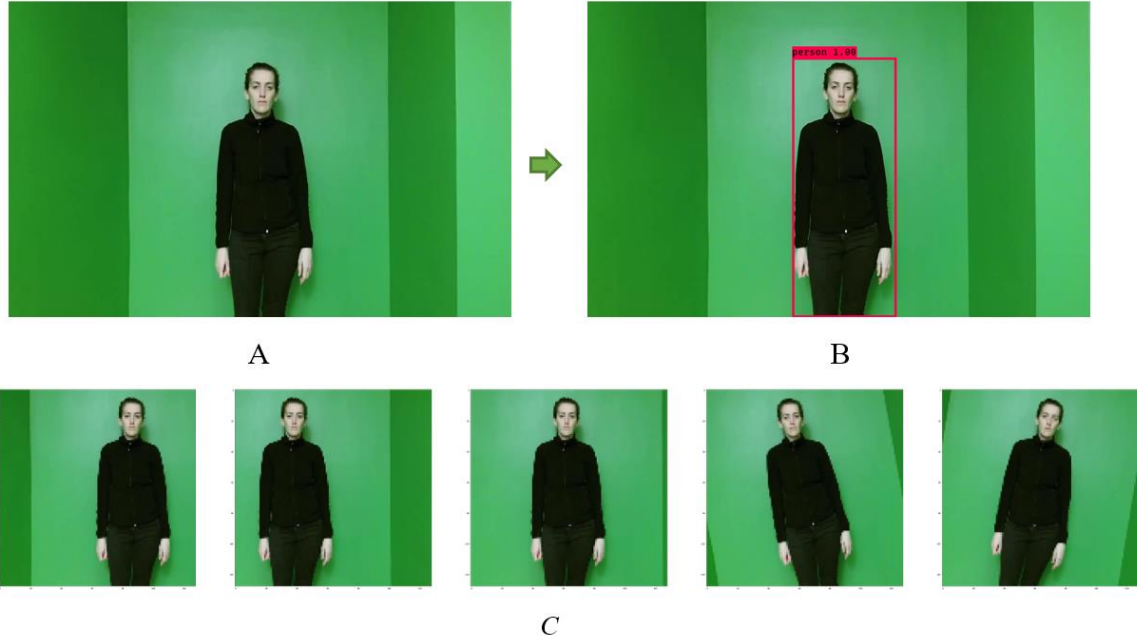
Kullanılan veri kümesindeki videolar incelendiğinde bazı işaretçilerin kamera görüntüsünün tam ortasında olmadığı görülmüştür. Veri artırma öncesinde uzamsal veri kaybı olmaması açısından işaretçi videonun tam ortasında olacak şekilde kırpmalar yapılması hedeflenmiştir. Bunun sağlanabilmesi için işaretçinin video karesinin neresinde olduğunun belirlenmesi gereklidir. Bu tez çalışmasında işaretçinin video görüntüsündeki konumunun belirlenmesi için YOLO nesne tanıma algoritması kullanılmıştır.

YOLO, nesne tanıma işini bir regresyon problemi olarak ele alan bir derin öğrenme algoritmasıdır [78]. Verilen görüntü üzerinde nesne tanıma, çevreleyen kutu belirleme ve nesne sınıfını belirleme işlemlerini tek bir taramada yapabilir. Çevre belirleme ve sınıflandırma işlemini uçtan uca tek seferde yapabildiği için diğer nesne tanıma uygulamalarına göre daha hızlıdır. 53 tanesi nesne algılama 53 tanesi sınıflandırmada kullanılmak üzere toplamda 106 katmandan oluşmaktadır.

Veri kümesindeki videolarda işaretçinin görüntüdeki konumunu bulmak için COCO veri kümesi üzerinde önceden eğitilmiş YOLOv3 modeli kullanılmıştır. Bu veri kümesi ile eğitilmiş model, 80 farklı nesneyi tanıyabilmektedir. Bu nesne sınıfları arasında kişi (*person*) sınıfı bulunduğundan yeniden eğitmeye gerek görülmemiş, önceden eğitilmiş ağırlıklar ile devam edilmiştir.

İşaretçilerin video karesindeki konumunun bulunması için tüm karelerde nesne tanıma yapılmasına gerek görülmemiştir. İşaretçiler videolarda konum değiştirmedikleri ve kamera hareket etmediği için videonun sadece ilk karelerinde YOLO ile nesne tanıma yapılmış ve işaretçinin bu karedeki konumu tespit edilmiş ve kaydedilmiştir. Bu esnada, veri kümesinde kırılmamış videolar veya işaret barındırmayan videolar gibi bazı problemli örnekler olduğu görülmüş ve problemli örnekler veri kümesinden çıkarılmıştır.

İşaretçinin video karesindeki konumu tespit edildikten sonra uzamsal veri artırma yapılmıştır. YOLO ile tespit edilen işaretçinin videonun tam ortasında, sağında ve solunda olacağı şekilde video kareleri kırılmıştır. Yapılan bir diğer uzamsal veri artırma şekli de belli açı ile döndürme işlemidir. İşaretçinin resim karesinin ortasında olduğu görüntü 10° ar derecelik açılarla saat yönünde ve saat yönünün tersine döndürülmüş ve uzamsal veri artırma işlemleri sonrasında toplamda 5 video elde edilmiştir. Uzamsal veri artırma sonucu oluşan görüntüler Resim 4.2’de verilmiştir.



Resim 4.2. Videonun ilk karesinden (A) YOLO ile işaretçi konumunun bulunması (B) ve uzamsal veri artırımı sonrası oluşan kareler (C)

Yapılan uzamsal veri artırma sonrasında elde edilen 5 farklı video görüntüsünden zamansal veri artırma metotları ile de veri artırma gerçekleştirilmiştir. Zamansal veri artırma için videolardan eşit aralıklı olarak belirli sayıda kare alınarak örnekleme yoluna gidilmiştir. Belirlenen örneklem büyüklüğüne bağlı olarak zamansal ve uzamsal veri artırma sonrasında bir ham işaret dili videosundan elde edilen örnek sayısı 40'a kadar çıkabilmektedir. Bu nedenle veri artırmanın yeterli olduğu düşünülmüş ve daha karmaşık veri artırma yöntemlerine başvurulmamıştır.

İşaret dili tanıma çalışmalarında bir videodan alınan kare sayılarında çeşitli değerler kullanıldığı görülmüştür. Bu tez çalışması kapsamında örneklem kare sayısı ve büyüklüğünün işaret dili tanıma başarımına etkisini incelemek için bir dizi deney yürütülmüştür. Hiperparametre olarak değerlendirilen kare sayısı ve boyutları ile ilgili deneyler Bölüm 5.1'de verilmiştir.



5. DENEYSEL BULGULAR

Bu tez çalışmasında derin öğrenme yardımıyla işaret dili tanıma deneyleri üç başlık altında gerçekleştirilmiştir.

- İlk grup deneyler, işaret dili tanıma amacı ile kullanılacak videolardan alınacak örneklemelerdeki kare sayıları ve boyutları ile ilgilidir. Bu deneylerde önişleme parametreleri belirlenmiştir.
- 2. grup deneyler, 1. grup deneylerde en yüksek doğruluğu veren parametreler kullanılarak farklı derin öğrenme modellerinin işaret dili tanıma başarımını belirlemek üzere yürütülmüştür.
- 3. grup deneylerde ise birden fazla veri kipi (*data modality*) işaret dili tanıma amacı ile kullanılmış ve işaret dilindeki çeşitli kanalları ifade eden verilerin tanıma başarımına etkisi değerlendirilmiştir.

Deneylerin başarımlarını değerlendirme ölçütü olarak sınıflandırma doğruluğu (*accuracy*) dikkate alınmıştır. Bunun yanı sıra bazı modellerin sınıflandırma sonuçlarında kesinlik (*precision*), duyarlılık (*sensitivity*) ve f1-skorları da verilmiştir. Ayrıca veri kümesinin eğitimi sonrasında test aşamasında elde edilen karmaşıklık matrisleri (*confusion matrix*) verilerek sonuçlar görselleştirilmiştir. Çizelge 5.1’deki gibi verilen ikili karmaşıklık matrisi için doğruluk, kesinlik, duyarlılık ve f1-skorlarının hesaplaması Eş. 5.1 - Eş. 5.4’de verilmiştir.

Çizelge 5.1. İkili bir karmaşıklık matrisi

		Gerçekte Olan	
		Pozitif	Negatif
Tahmin Edilen	Pozitif	Doğru Pozitif (<i>TP</i>)	Yanlış Pozitif (<i>FP</i>)
	Negatif	Yanlış Negatif (<i>FN</i>)	Doğru Negatif (<i>TN</i>)

$$\text{doğruluk} = \frac{TP+TN}{TP+TN+FP+FN} \quad (5.1)$$

$$\text{kesinlik} = \frac{TP}{TP + FP} \quad (5.2)$$

$$duyarlılık = \frac{TP}{TP + FN} \quad (5.3)$$

$$f1-skoru = 2 \times \frac{kesinlik \times duyarlılık}{kesinlik + duyarlılık} \quad (5.4)$$

5.1. Veri Artırma Deneyleri ve Bulgular

İşaret dili tanıma çalışmalarında, videoları hem kare sayılarının fazlalığı ve değişkenliği, hem de görüntünün boyutları nedeni ile bir önişlemeden geçirilmeden evrimsel sinir ağına girdi olarak vermek mümkün olmamaktadır. Bu nedenle videolardan sabit kare sayısı ve görüntü boyutlarında örneklemeler alınmaktadır. İşaret dili tanıma çalışmaları incelendiğinde derin öğrenme kullanılan çalışmalarda bir işaret dili videosundan örneklem olarak alınan kare sayısının 9 [14] ile 64 [12] arasında değiştiği görülmüştür.

Birçok işaret dili tanıma çalışmasında veri artırma sürecinin nasıl yürütüldüğüne değinilmemiştir. İşaret dili tanıma çalışmaları incelendiğinde veri artırma süreci ve parametrelerinin tanıma başarımına etkilerini inceleyen bir çalışmaya rastlanmamıştır [29]. Bu nedenle bu çalışmada öncelikle farklı kare sayıları ve boyutları ile deneyler yürütülerek işaret dili tanıma başarımına etkilerinin incelenmesi ve veri kümesine en uygun değerlerin belirlenmesi hedeflenmiştir.

Videodan alınacak örneklemdaki kare sayısının videonun önişleme öncesindeki kare sayısından az olduğu durumlarda eşit aralıklarla veya ardışık kareler alınabilir. İşaret dili videolarında videonun rastgele bir noktasından başlayarak ardışık karelerin alınması bir işaretin gösteriminin tamamını içermeyebilir. Bu nedenle videolardan eşit aralıklarla kareler alınarak videonun tümünü ifade edecek şekilde bir örneklem alma işaret dili tanıma için daha uygundur.

Özellikle son yıllardaki büyük hafızaya sahip donanımlarla birlikte yüksek kare sayılarında girdiler işaret dili tanıma sistemlerine verilebilmektedir. Bu durumda veri kümesindeki işaret dili videolarının tümü yeterli kare sayısını barındırmayabilir. Eğer bir videoda, örneklem sayısınca kare bulunmuyor ise; ilk ve/veya son kareler tekrar ettirilerek yeterli kare sayısına erişilmesi [12] veya son kareden sonra ilk kareye dönülerek videonun tekrar ettirilmesi [56] gibi yöntemlere başvurulabilir.

Videoların işaret dili tanıma sistemine girdi olarak verilmesindeki bir diğer parametre de kare boyutlarıdır. Piksel cinsinden ifade edilen kare boyutları için derin öğrenme modellerinde genellikle eşit genişlik ve yükseklik değerleri tercih edilmektedir. İlk derin öğrenme modellerinden bu yana 224 piksel genişlik ve yüksekliğinde görüntüler sıklıkla kullanılmaktadır [12, 44, 51].

Kameralar yardımı ile toplanan görüntüler, derin öğrenme modelinin girdi olarak alabileceği ölçülerden büyük olduğundan, uygulamanın amacına göre farklı şekillerde boyut küçültme işlemi yapılabilir. Çalışmalarda; görüntü belirli bir boyuta küçültüldükten sonra (ör: 256*256) bu görüntüden rastgele 224*224 boyutlarında bir alan kırılması [56] veya resmin en küçük boyutu 224 piksel olacak şekilde küçültülerek ortada kalan kare alanın kırılması [33] gibi boyutlandırma yöntemlerinin kullanıldığı görülmüştür.

Birçok derin öğrenme modelinde, kare sayısı ve boyutları arttıkça modelin işlem karmaşıklığı artmakta, eğitim süresi uzamakta ve modelin parametre sayısı artmaktadır. Bu nedenle çok sayıda karenin birlikte girdi olarak verildiği modellerde çok büyük boyutlu görüntülerin kullanılması donanım kapasitelerinin yetersizliği nedeni ile mümkün olmamıştır. Bunun sonucu olarak da derin öğrenme kullanılan ilk işaret dili tanıma çalışmalarında 64 piksel gibi küçük kare boyutları tercih edilmiştir [33].

Kare boyutları küçüldükçe görüntüdeki detaylar kaybolmakta ve özellikle işaret dili tanıma çalışmalarında parmak şekillerinin anlaşılması güçleşmektedir. Bu nedenle hem parmak işaretlerinin anlaşılabilmesi kadar detaylı, hem de derin öğrenme modelinin işleyebileceği ölçülerde girdiler belirlenmesi önemlidir. Bu çalışmada, 8, 12, 16, 20 ve 32 kare sayıları; 128, 196 ve 224 piksel boyutlarında video örneklemeleri ile deneyler yürütülerek eldeki veri kümesinde işaret dili tanıma için kullanılacak en uygun parametrelerin belirlenmesi hedeflenmiştir.

Deneyler için her bir kare sayısı parametresi için farklı görüntü boyutları ile veri artırma gerçekleştirilmiştir. Örneğin, 8 kare sayısı için 128, 196 ve 224 piksellik ölçeklendirmeler sonucunda 8@128x128, 8@196x196, 8@224x224 boyutlarında örneklerden oluşan 3 veri kümesi elde edilmiştir. 8@128x128 ifadesinde; 8 örnekte seçilen kare sayısını, 128 değerleri ise seçilen karenin piksel türünden genişlik ve yüksekliğini ifade etmektedir. Bu

doğrultuda, elde edilen veri setlerinin kare sayıları, genişlik ve yükseklikleri ile veri artırma sonrasında elde edilen örneklem sayıları Çizelge 5.2’de sunulmuştur.

Çizelge 5.2. Örneklem boyutları ve önişleme sonrasında elde edilen sayıları

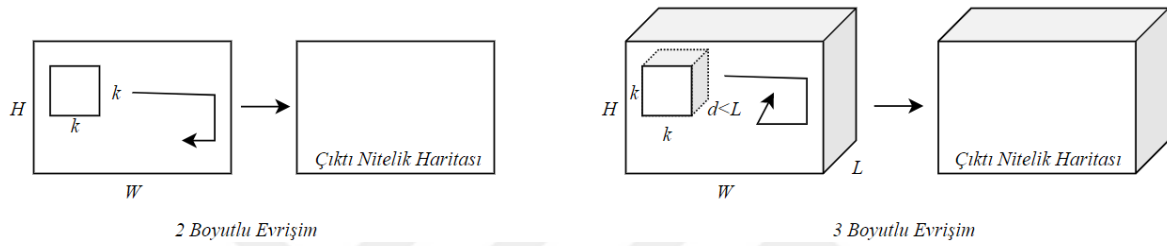
Örneklem İfadesi	Kare Sayısı	Kare Genişlik ve Yüksekliği	Örneklem Sayısı
8@128x128	8	128	31645
8@196x196	8	196	31645
8@224x224	8	224	31645
12@128x128	12	128	20610
12@196x196	12	196	20610
12@224x224	12	224	20610
16@128x128	16	128	15095
16@196x196	16	196	15095
16@224x224	16	224	15095
20@128x128	20	128	11765
20@196x196	20	196	11765
20@224x224	20	224	11765
32@128x128	32	128	6845
32@196x196	32	196	6845
32@224x224	32	224	6845

İlk grup deneylerde çok sayıda parametre ile deneyler yürütüleceğinden, verinin tamamı ile modellerin eğitimi uzun sürmektedir. Bu nedenle, bu deneylerde veri kümesinden rastgele seçilmiş 15 sınıflık bir alt küme kullanılmıştır. Bu alt kümedeki sınıflar ve veri artırımı öncesinde sınıflar bazında video örnek sayıları Çizelge 5.3’de sunulmuştur.

Çizelge 5.3. Test verisindeki sınıflar ve örnek sayıları

Sınıf Numarası	Sınıf Etiketi	Örnek Sayısı
10	Açmak	30
101	Bir hafta sonra	31
108	Bitmek	61
123	Bulmak	32
137	Çekmek	60
142	Çevirmek	61
176	Dökmek	30
204	Emekli olmak	30
215	Erken	32
248	Göndermek	63
262	Hafta	33
303	İnşallah	65
357	Kaşınmak	30
363	Kazanmak	32
51	Ay	32

Videolarda uzamsal veri kadar zamansal veri de önem arz etmektedir. Özellikle işaret dili tanıma gibi, tek bir karedeki uzamsal veriden bağlamın anlaşılamayacağı uygulamalarda, zamansal verinin önemi daha büyüktür. Ancak, statik görüntüler üzerinde yapılan 2 boyutlu evrişim ve havuzlama işlemleri ile videodaki zamansal verinin modellenmesi mümkün olmamaktadır. 2 boyutlu video kareleri üzerinde yapılan 2 boyutlu evrişim sonrasında yine 2 boyutlu bir öznitelik haritası elde edilmekte ve zamansal veri hiç hesaba katılamamaktadır. Bunu sağlayabilmek için 3 boyutlu evrişim yapan modeller önerilmiştir. 2 ve 3 boyutlu evrişim işlemleri Şekil 5.1’de verilmiştir.



Şekil 5.1. İki ve üç boyutlu evrişim işlemleri

3 boyutlu evrişim ile video karesindeki zamansal veri ile birlikte, evrişim filtresinin derinliği kadar sonraki karelerdeki veriler de birlikte değerlendirilebilmektedir. Bu sayede hem filtrenin bulunduğu piksellerdeki uzamsal ilişkiler, hem de bu piksellerin zamanla olan değişimlerindeki zamansal ilişkiler modellenebilmektedir. Bu katmanların sonrasında gelen havuzlama katmanları yardımı ile havuzlama işlemi de 3 boyutlu olarak gerçekleştirilmekte ve zamansal veri tam bağlı katmanlara kadar kaybolmadan aktarılabilmektedir [55, 79].

Bu tez çalışmasında yürütülen 1. grup deneyler için C3D [55] modelini temel alan bir modelden faydalanılmıştır. C3D modeli Facebook yapay zekâ laboratuvarı bünyesinde geliştirilmiş, 3 boyutlu evrişim ve havuzlama katmanları kullanarak hızlı ve etkili şekilde uzam-zamansal öznitelik haritaları çıkarmaya imkân tanıyan bir derin öğrenme modelidir. Çıkarılan öznitelik haritaları yardımı ile hem karelerdeki görüntülerin hem de kareler arasındaki değişimlerin başarılı bir şekilde modellenebildiği ifade edilmiştir.

C3D modelinin dezavantajlarından biri olarak, verilen girdi büyüklüğüne göre parametre sayısının artması gösterilebilir. Bu durum hem işlem karmaşıklığının artması ve dolayısı ile eğitim sürecinin uzamasına, hem eğitim sonrasında elde edilen modelin sabit diskte

kapladığı alanın artmasına neden olmaktadır. Farklı girdi ölçüleri ile gerçekleştirilen deneylerde, modelin parametre sayısının 8.170.255 (8@128x128) ile 71.215.887 (32@224x224) arasında; tek bir döngülük eğitimin 320 ile 790 saniye arasında; eğitim sonrasındaki ağırlıkların sabit diskteki boyutunun da 255 MB ile 877 MB arasında değiştiği görülmüştür.

8@128x128 ölçülerinde video örnekler alan bir C3D modelinin katmanları ve her bir katmandaki ağırlık sayıları Çizelge 5.4’de verilmiştir. Eğitimlerde AMD Ryzen 2600 işlemci, 32 GB bellek, Nvidia Titan XP GİB donanımlarına sahip bir kişisel bilgisayar kullanılmış ve eğitimler 100 döngü boyunca devam ettirilmiştir. Modelin geliştirilmesinde Python dilinden ve Keras-Tensorflow kütüphanesinden yararlanılmıştır. Eğitim sonrası, model yapısı ve ağırlıkları dondurularak daha sonra tekrar kullanılmak üzere kaydedilmiştir.

Çizelge 5.4. 1. grup deneylerde kullanılan C3D modelinin katmanları

Katman Adı (Türü)	Çıktı Boyutu	Ağırlık Sayısı
Input(Input)	(None, 12, 128, 128, 3)	0
conv3d_1 (Conv3D)	(None, 12, 128, 128, 32)	2.624
conv3d_2 (Conv3D)	(None, 12, 128, 128, 32)	27.680
max_pooling3d_1 (MaxPooling3D)	(None, 6, 64, 64, 32)	0
conv3d_3 (Conv3D)	(None, 6, 64, 64, 32)	27.680
conv3d_4 (Conv3D)	(None, 6, 64, 64, 64)	55.360
max_pooling3d_2 (MaxPooling3D)	(None, 3, 32, 32, 64)	0
conv3d_5 (Conv3D)	(None, 3, 32, 32, 64)	110.656
max_pooling3d_3 (MaxPooling3D)	(None, 1, 11, 11, 64)	0
dropout_1 (Dropout)	(None, 1, 11, 11, 64)	0
flatten_1 (Flatten)	(None, 7.744)	0
dense_1 (Dense)	(None, 1.024)	7.930.880
dropout_2 (Dropout)	(None, 1.024)	0
dense_2 (Dense)	(None, 15)	15.375
Toplam Ağırlık Sayısı:	8.170.255	

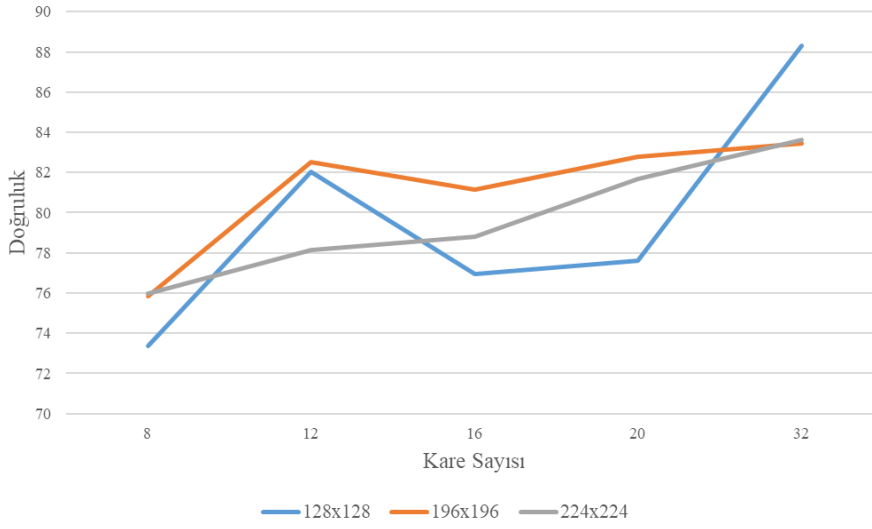
Eğitimler sonrasında her bir model, Bölüm 4.1’de açıklanan test senaryosu ile doğruluk değerlendirmesinden geçirilmiştir. Deneyler sonucunda farklı örneklem boyutları ile elde edilen doğruluk oranları Çizelge 5.5’de verilmiştir.

Çizelge 5.5. Deneyler sonucunda elde edilen doğruluk oranları

Deney Parametresi	Top 1 Doğruluk	Top 3 Doğruluk	Top 5 Doğruluk
8@128x128	73,35	92,71	96,88
8@196x196	75,86	94,61	97,83
8@224x224	75,98	93,65	97,42
12@128x128	82,03	96,25	99,16
12@196x196	82,53	95,64	98,02
12@224x224	78,14	94,50	98,36
16@128x128	76,95	94,67	97,88
16@196x196	81,14	95,47	98,35
16@224x224	78,81	94,12	97,02
20@128x128	77,62	95,47	98,65
20@196x196	82,80	96,01	98,59
20@224x224	81,67	95,36	97,73
32@128x128	88,31	98,59	99,71
32@196x196	83,45	94,67	97,28
32@224x224	83,64	94,85	96,82

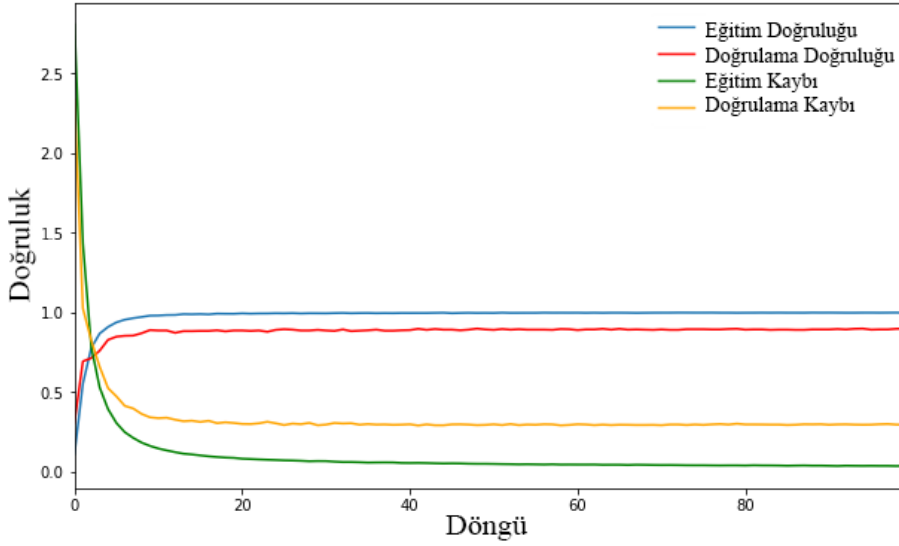
Rastgele ağırlıklar ile başlanıp, aynı öğrenme parametreleri ile yapılan eğitimler sonucunda en yüksek doğruluk oranının 32 kare sayısında 128 piksel boyutlarındaki örneklemeler ile elde edildiği görülmüştür.

Eşit kare sayılarında farklı piksel boyutları ile elde edilen sonuçlar incelendiğinde kare boyutları ile doğruluk arasında anlamlı bir bağıntı görülememiştir. Eşit kare boyutlarında ise, kare sayısı arttıkça doğruluk oranının da genelde arttığı görülmüştür. Bu durum, Şekil 5.2’de sunulmuştur.



Şekil 5.2: Kare sayısı - doğruluk grafiği

Deneylerde en yüksek doğruluk veren parametreler olan 32@128x128 ile yapılan eğitim ve doğrulama süresince modelin doğruluk ve kayıplarının değişimini gösteren grafik Şekil 5.3’de sunulmuştur.

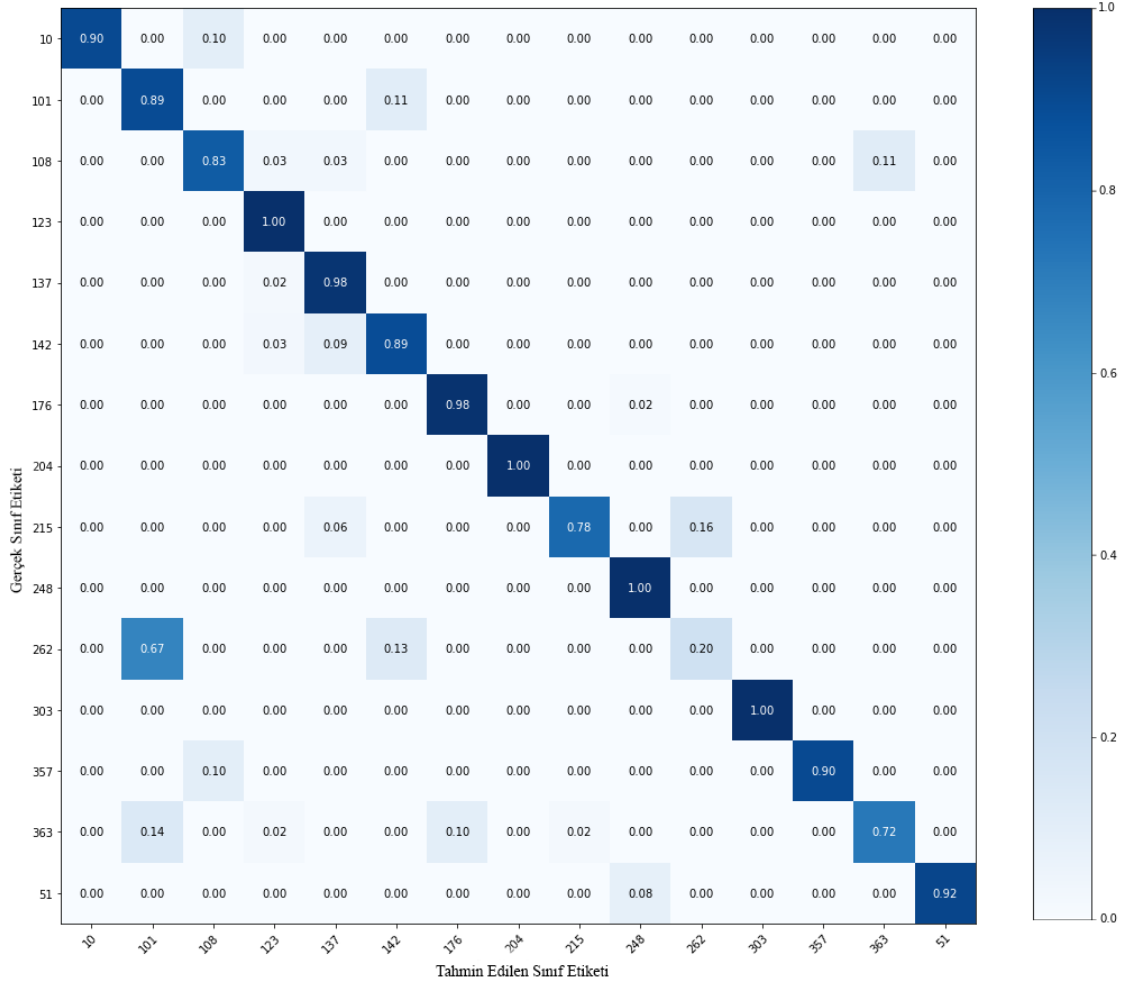


Şekil 5.3. Deneyler sonucunda en yüksek doğruluk veren parametrelerin eğitim ve doğrulama süreci

32@128x128 parametreleriyle eğitim süreci sonrasında model, 15 sınıflık test verisinde %88,31 doğruluk ile sınıflandırma gerçekleştirebilmektedir. Modelin eğitim sonrasındaki test aşamasında elde edilen kesinlik, duyarlılık oranları ve f1-skorları Çizelge 5.6’da; sınıflandırmanın karmaşıklık matrisi Şekil 5.4’de sunulmuştur.

Çizelge 5.6. En yüksek doğruluk veren modelin sınıflandırma sonuçları

Sınıf	Kesinlik	Duyarlılık	F1-Skoru
10	1	0,9	0,95
101	0,53	0,89	0,66
108	0,89	0,83	0,86
123	0,85	1	0,92
137	0,86	0,98	0,92
142	0,89	0,89	0,89
176	0,93	0,98	0,96
204	1	1	1
215	0,97	0,78	0,87
248	0,94	1	0,97
262	0,58	0,2	0,3
303	1	1	1
357	1	0,9	0,95
363	0,75	0,72	0,73
51	1	0,92	0,96
Doğruluk			0,88



Şekil 5.4. Deneyler sonucunda en yüksek doğruluk veren modelin karmaşıklık matrisi

Veri parametreleri ile ilgili olarak yapılan 1. grup deneyler sonucunda en yüksek doğruluk veren 32@128x128 parametreleri ile 154 sınıflık tüm veri kümesi önışlemeden geçirilmiştir. Deneylerde kullanılan aynı eğitim parametreleri ile 154 sınıflık veri eğitildiğinde %98 eğitim doğruluğuna erişildiğinde test doğruluğunun %66,14 olduğu görülmüştür. Bu noktada C3D modelinin eksik öğrenme gerçekleştirdiği görüldüğünden güncel hareket tanıma çalışmalarında kullanılan daha karmaşık modeller üzerinde deneyler yürütülmesi kararlaştırılmıştır.

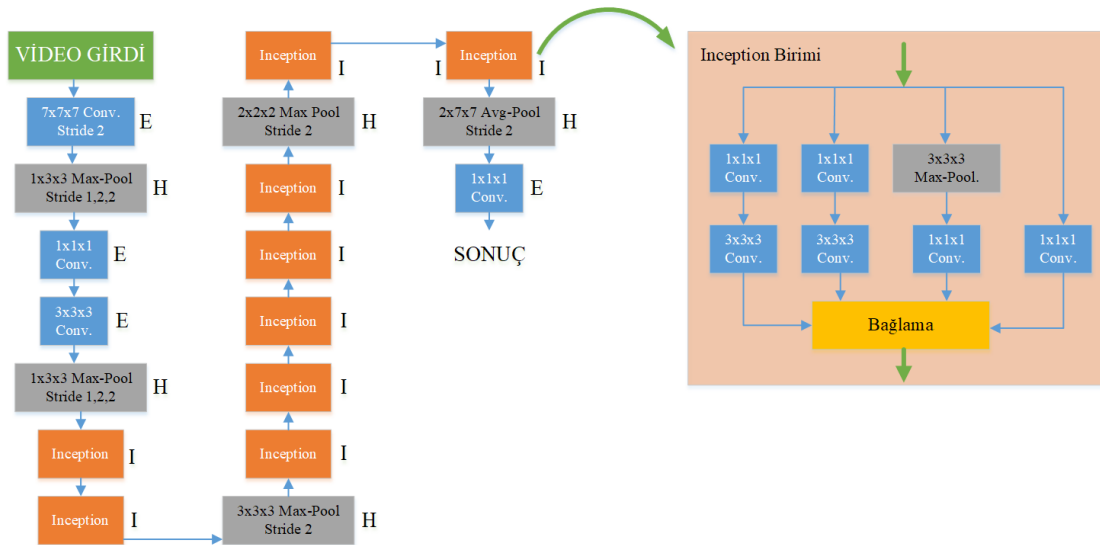
5.2. Çeşitli Derin Öğrenme Modelleri ile Gerçekleştirilen Deneyler ve Bulgular

Bu çalışmada, veri artırma hiperparametrelerinin belirlenmesi amacı ile yapılmış 1. grup deneyler sonrasında C3D derin öğrenme modeli 15 sınıflık veri kümesi yerine 154 sınıflık tüm veri kümesi ile yeniden eğitilmiştir. Bölüm 4.1’de açıklanan veri bölme senaryosu ile

yapılan eğitim sonrasında, modelin eğitim veri kümesinde %98 doğruluk sunduğu ancak test veri kümesinde %66,14 doğruluk verebildiği görülmüştür. Bu durum, modelin eksik öğrendiği ve genelleme yeteneğinin yetersizliği olarak yorumlandığı için, farklı derin öğrenme modelleri kullanılarak başarımın artırılması amacı ile 2. grup deneyler gerçekleştirilmiştir.

Yapılan 2. grup deneylerde, 1. grup deneylerde kullanılan C3D modelinin yanı sıra aksiyon tanıma için geliştirilmiş ve işaret dili tanıma çalışmalarında da yüksek başarımlar sunduğu ifade edilen [12] Inception3D (I3D) [56] ve 3DResnet [57] modelleri kullanılmıştır.

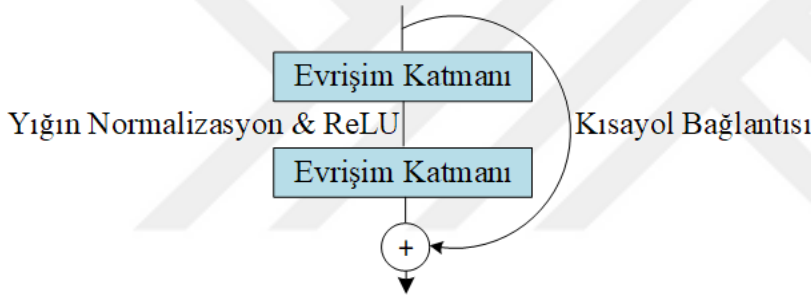
İşaret dili tanıma çalışmaları, amaç ve uygulama adımları bakımından hareket tanıma çalışmalarına benzerlik gösterir. Hareket tanıma alanında kullanılan derin öğrenme modellerinin sıklıkla işaret dili tanıma için de kullanıldığı görülmektedir. Inception [46] derin öğrenme modeli, statik görüntülerin sınıflandırılması amacı ile Google araştırmacıları tarafından geliştirilmiş ve yüksek başarı elde etmiştir. Hareket tanıma çalışmaları incelendiğinde 3 boyutlu evrişim yapan I3D modelinin karşılaştırmalarda kullanıldığı görülmüştür. I3D, resim sınıflandırma için kullanılan Inception modelindeki 2 boyutlu evrişim ve havuzlama katmanlarının filtre boyutları aynı olacak şekilde 3 boyutlu evrişim katmanları ile değiştirilmiş şeklidir. Şekil 5.5’de model mimarisi verilmiş, evrişim katmanları mavi (E), havuzlama katmanları gri (H), modele özgü Inception birimleri ise turuncu (I) renkle (harfleriyle) gösterilmiştir.



Şekil 5.5. I3D model mimarisi

Modelde yer alan Inception birimlerinde, önceki katmandan alınan girdiler farklı filtre boyutlarına sahip 4 farklı evrişim işleminden geçirilir. Her bir evrişim işleminin çıktısı bağlanarak (*concatenation*) birimin çıktısı elde edilir. Inception birimleri sayesinde aşırı öğrenmenin azaltılması ve daha az işlem karmaşıklığı hedeflenmiştir.

Hareket tanıma çalışmalarında karşılaşılan bir diğer model de Resnet [80] modelinden esinlenilerek geliştirilmiş olan 3DResnet [57] modelidir. I3D’de olduğu gibi bu modelde de 2 boyutlu evrişim ve havuzlama katmanları aynı filtre boyutlarına sahip 3 boyutlu evrişim katmanları ile değiştirilmiştir. Resnet modellerinde katman girdisini bir sonraki katmana aktaran kısayol bağlantılar (*shortcut connections*) bulunur. Şekil 5.6’da gösterilen bu kısayol bağlantılar sayesinde modelin sonraki katmanlarından ilk katmanlara gradyan akışı sağlanarak çok derin modeller eğitilebilir [57].



Şekil 5.6. Resnet modelinde kısayol bağlantıları

Resnet modelinin, içerdiği katman sayılarına göre isimlendirilmiş Resnet-18, Resnet-50 gibi çeşitleri bulunmaktadır. Tez çalışmasının 2. grup deneylerinde, daha çok tercih edilen 3DResNet-50 modeli kullanılmıştır.

2. grup deneylerde yürütülen bir diğer deney şekli de daha önce eğitilmiş modeller üzerinden transfer öğrenme yönteminin kullanılmasıdır. Transfer öğrenme, bir modelin eğitimine farklı bir veri setinde elde edilmiş ağırlıkları kullanarak başlamak ve bu sayede eski eğitimden yararlanmak olarak özetlenebilir.

Transfer öğrenme üç şekilde gerçekleştirilebilir. Bunlardan ilki (*i*), daha önce farklı bir veri üzerinde eğitilmiş model ağırlıklarını kullanarak yeni veride darboğaz öznetelik (*bottleneck feature*) çıkarmak ve sonrasında darboğaz öznetelikler üzerinden sınıflandırma gerçekleştirmektir.

Bir diğer transfer öğrenme şeklinde (ii), model ağırlıkları yüklendikten sonra öznitelik çıkarıcı katmanlardaki ağırlıklar dondurulur, sonda yer alan sınıflandırıcı katmanlar silinir ve yerine yeni veriye uygun sınıflandırıcı katmanlar eklenir. Bu sayede çıkarılmış öznitelikler korunur. Öğrenme, sadece sondaki sınıflandırıcı katmanda gerçekleştiğinden daha hızlı sonuçlandırılabilir. Ancak öznitelik çıkarıcı katmanlardaki ağırlıklar sabit kalacağından sonradan eğitilecek verinin bağlam olarak ilk veriye benzer olması beklenir. Aksi halde yeni veriye özgü ayrıntılar öğrenilemeyecektir. Bunu aşmak için öznitelik çıkarıcı katmanların ağırlıkları dondurulmadan eğitim tüm model parametreleri üzerinde gerçekleştirilebilir. Bu durumda ağırlık güncellemeleri tüm katmanlarda yapılacağından eğitim daha yavaş olmaktadır.

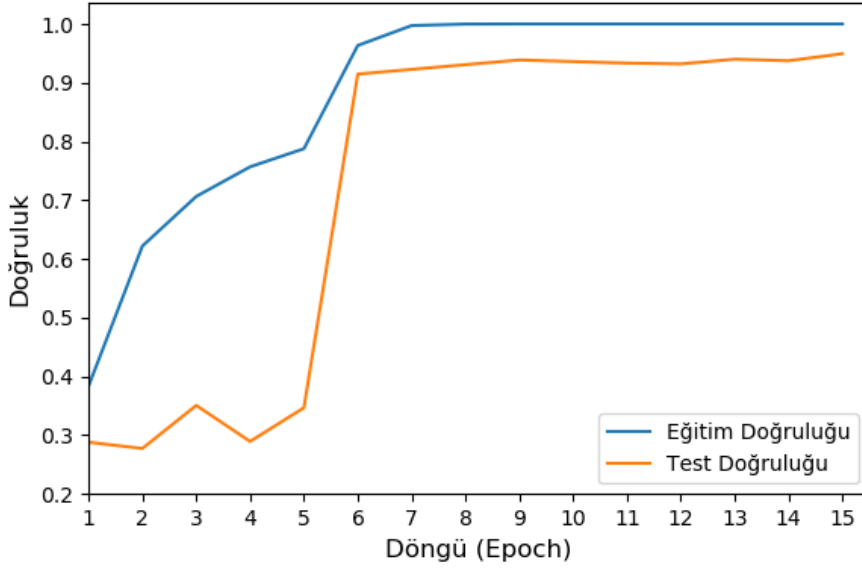
Derin öğrenmede karşılaşılan transfer öğrenme yöntemlerinin bir diğeri de (iii) aşamalı olarak katmanların eğitilmesidir. Bunun için model ağırlıkları yüklenir ve ağırlıklar dondurulur. Sondaki sınıflandırıcı katmanlar yerine yeni veriye uygun sınıflandırıcı katmanlar eklenir. Eğitim bu haliyle bir süre gerçekleştirilir. Öğrenme hızı yavaşladığında geriye doğru diğer katmanlar aşamalı olarak eğitime başlanır.

Tez çalışmasının bu aşamasındaki deneylerde, I3D modelinin Kinetics [81] veri kümesi üzerinde hareket tanıma amacı ile eğitilmiş ağırlıklarına ulaşılabilmiştir. Transfer öğrenmenin işaret dili tanımaya katkısını değerlendirmek amacı ile bu ağırlıklar kullanılarak eğitim gerçekleştirilmiştir.

2. grup deneyler 15 sınıflık veri kümesi üzerinde ve 1. grup deneylerde en yüksek doğruluğu veren 32 karelik örneklerle gerçekleştirilmiştir. Görüntü boyutu olarak da 1. grup deneylerde en yüksek doğruluk elde edilen 128x128 ölçülerinin yanı sıra çalışmalarda sıklıkla tercih edilen ve transfer öğrenmede kullanılacak modelin girdi boyutu olan 224x224 piksel ölçüleri de kullanılmıştır.

Transfer öğrenme için öncelikle I3D modelinin Kinetics veri kümesinde eğitilmiş ağırlıkları yüklenmiş, sınıflandırıcı kısmı olan son katmanları silinmiş ve 15 sınıflık veri kümesinin sınıf sayısına uygun olacak şekilde yeni bir sınıflandırıcı katman eklenmiştir. Elde edilen modelin sınıflandırıcı öncesindeki katmanlardaki ağırlıklar dondurularak 5 döngü süresince sadece yeni eklenen sınıflandırıcı katman eğitilmiştir. Ardından,

ağırlıkları dondurulmuş olan evrişim katmanları da eğitilebilir hale getirilerek 10 döngü daha eğitim devam ettirilmiştir. Transfer öğrenme kullanılarak, 32@224x224 girdi ölçülerinde yapılan eğitim süresince modelin doğruluk değişimi Şekil 5.7’de verilmiştir. Tüm model ağırlıklarının eğitime başlandığı 5. döngü sonrasında eğitim ve test doğruluğunun hızlı bir şekilde arttığı görülebilir.



Şekil 5.7. Transfer öğrenme yoluylla deney veri kümesinde yapılan eğitimin doğruluk oranı değişimi

Farklı eğitim parametreleri ve modeller kullanılarak yapılan eğitimlerde 15 sınıflık veri kümesinde elde edilen sonuçlar Çizelge 5.7’de sunulmuştur.

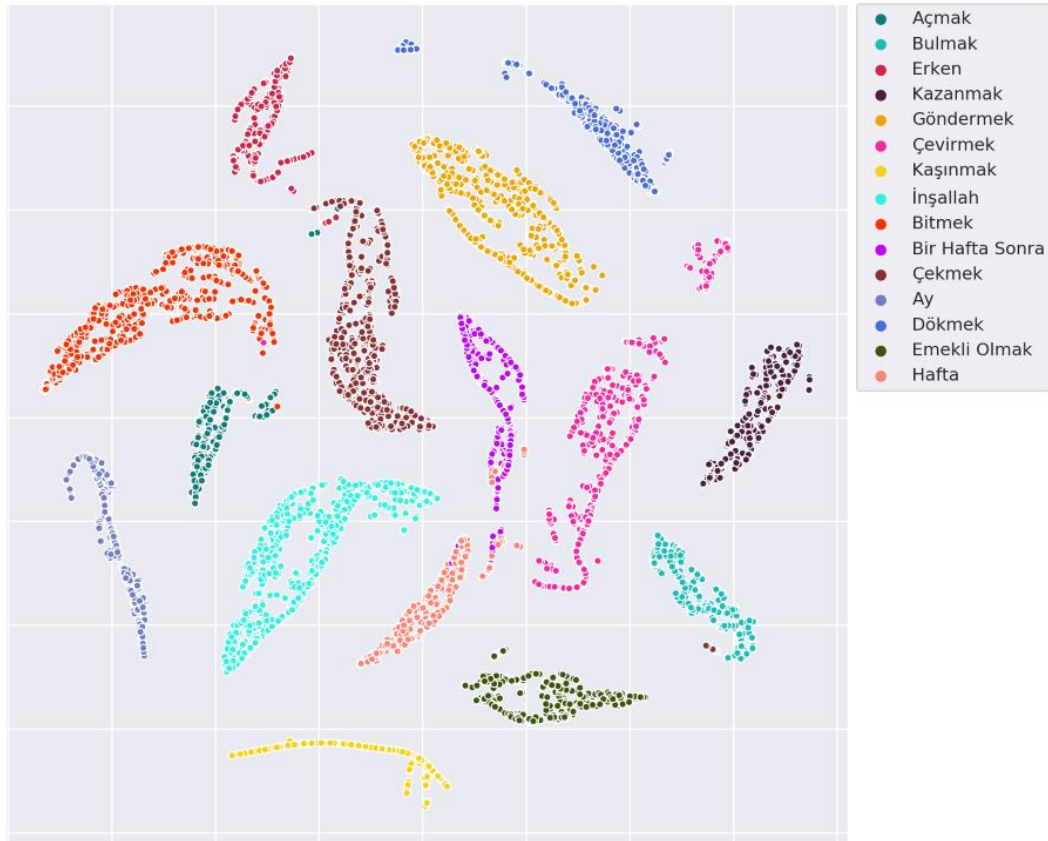
Çizelge 5.7. İkinci grup deneylerde test verisindeki sonuçlar

Model Adı	Kullanılan Parametreler	Ağırlık Sayısı	% Doğruluk
C3D	32@128x128	24.029.967	88,31
C3D	32@224x224	71.215.887	83,64
I3D	32@128x128	12.309.919	92,76
I3D	32@224x224	12.309.919	93,08
3DResnet-50	32@128x128	46.294.031	91,30
3DResnet-50	32@224x224	46.294.031	89,53
I3D Transfer	32@128x128	12.309.919	88,41
I3D Transfer	32@224x224	12.309.919	91,68

Deney sonuçlarından görüleceği üzere transfer öğrenme yönteminin eğitim süresini bir miktar kısaltmak dışında doğruluğa kayda değer bir etkisi olmamıştır.

Çeşitli derin öğrenme modelleri ile yapılan 2. grup deneyler sonucunda en yüksek doğruluğun 32@224x224 görüntü parametreleri ile rastgele ağırlıklarla başlatılarak eğitilen I3D modelinde elde edildiği görülmüştür. I3D modeli iki farklı girdi boyutunda da hem 1. deney grubunda kullanılan C3D modelinden hem de 3DResnet-50 modelinden daha yüksek başarımlar elde etmiştir. I3D modelinin avantajlarından bir diğeri de kullanılan üç model içerisinde en az ağırlığa sahip olması ve girdi boyutu arttıkça ağırlık sayısının değişmemesidir.

En yüksek doğruluğu veren I3D modelinin test verisi üzerindeki sınıflandırma sonucunu gösteren t-SNE görselleştirmesi Şekil 5.8’de verilmiştir. Şekilde görüleceği üzere “hafta” ve “bir hafta sonra” gibi gösteriminde küçük farklılıklar olan bazı sınıflar dışında sınıflandırıcının işaretler arasındaki ayrımı başarılı bir şekilde yapabilmektedir.



Şekil 5.8. 15 sınıflık test verisinde I3D modeli ile çıkarılan öz niteliklerin t-SNE görselleştirmesi

15 sınıflık veri kümesindeki deneylerin ardından, deneylerin tüm veri kümesi üzerinde devam ettirilmesi ve başarımların 154 sınıf üzerinde de benzer şekilde olup olmadığının incelenmesi için tüm veri kümesi ile eğitimler gerçekleştirilmiştir. Bu eğitimlerde I3D modeli transfer öğrenme yoluyla ve rastgele ağırlıklarla 32@128x128 ve 32@224x224 girdi boyutlarında eğitilmiştir. Sonuçlar referans olması bakımından 1. grup deneyler sonrasında tüm veri üzerinde eğitilen C3D modeli ile karşılaştırılmıştır.

1. grup deneylerde kullanılan C3D modeli tüm veri kümesi üzerinde yaklaşık 80 döngü süresince eğitilmiştir. I3D modelleri ise, rastgele ağırlıklar ile başlatıldıklarında 10 döngü süresince, transfer öğrenme yoluyla eğitildiklerinde ise 5 döngü sadece sınıflandırma katmanları, sonraki 5 döngü tüm katmanlar olacak şekilde toplamda 10 döngü eğitilmişlerdir. Eğitimler sonucunda elde edilen doğruluklar Çizelge 5.8’de verilmiştir. Sonuçlardan görüleceği üzere I3D modelinin doğrulukları C3D modelinden her eğitimde daha yüksektir.

Çizelge 5.8. Tüm veri kümesi üzerinde yapılan eğitimlerin sonuçları

Model Adı	Girdi Boyutu	% Doğruluk
C3D	12@128x128	66,14
I3D	32@128x128	82,28
I3D	32@224x224	84,34
I3D Transfer	32@128x128	80,56
I3D Transfer	32@224x224	86,75

154 sınıflık tüm veri kümesi ile yapılan eğitimlerin sonucunda en yüksek doğruluk, Kinetics veri kümesi ile ön eğitimi yapılmış I3D modelinden transfer öğrenme yöntemi ile yeniden eğitilmiş model ile elde edilmiştir. Bu eğitimde 32@224x224 boyutlarındaki girdiler kullanılmıştır. Tüm veri kümesinde 224 piksel ölçülerindeki girdi iki eğitimde de daha yüksek doğruluk sunmuştur. 154 sınıflık veri kümesinin eğitiminde modelin eğitim sürecinin grafiği Şekil 5.9’da ve test aşaması sonucunda elde edilen karmaşıklık matrisi Şekil 5.10’da verilmiştir.

5.3. Çeşitli Veri Tipleri ile Deneyler ve Bulgular

Tek başına RGB verileri kullanan işaret dili tanıma çalışmalarında derin öğrenme ile yüksek başarımlar elde edilebilmektedir. Ancak literatürdeki çalışmalarda birden çok veri kipi (*data modality*) birlikte kullanıldığında işaret dili tanıma başarımının daha da artırılabilirdiği ifade edilmiştir. Bunun için çeşitli veri kiplerinden ayrı akışlarda öznitelik çıkarımı gerçekleştirilir. Her bir akışta veri kipine uygun öznitelik çıkarım yöntemleri kullanılabilir. Farklı akışlarda çıkarılan öznitelikler, bir noktada birleştirilerek (*fusion*) öznitelik çıkarımına devam edilir veya sınıflandırıcıya aktarılır.

İşaret dili tanıma çalışmalarında veri kipleri standart RGB görüntüler ile birlikte, derinlik görüntüleri, optik akış, eklem verisi olabilir. Bunun yanı sıra, bir veri kipinde birden fazla veri akışı kullanılarak da çok akışlı çalışmalar yapılabilir. Örneğin beden RGB görüntüsü ile birlikte eller takip edilerek oluşturulan video görüntüleri ikinci bir akışta değerlendirilebilir.

İşaret dili tanıma konusunda yapılan güncel birçok çalışma çok kipli (*multimodal*) olarak yürütülmektedir. Bu nedenle bu tez çalışmasında öncelikle çeşitli veri kiplerinin tek başlarına başarımlarını değerlendiren deneyler yürütülmüş ardından veri kipleri birleştirilerek en yüksek başarımları veren veri kipleri ve akışlarını belirleme yoluna gidilmiştir.

5.3.1. Çok veri kipli işaret dili tanıma çalışmaları

İşaret dili tanıma ve hareket tanıma konusunda yakın zamandaki çalışmalarda RGB görüntülerin yanı sıra optik akış, derinlik, kızılötesi, eklem verisi gibi birden çok veri kipinin birlikte kullanıldığı görülmüştür. Bu veri kiplerinin bir kısmı tek başlarına da yüksek başarımlar sunabilirler. Ancak veri kipleri birlikte kullanıldıklarında işaret dili tanıma başarımı daha da artmaktadır [82].

Çok kipli birçok çalışmada, optik akış, RGB verilerden kolayca çıkarılabildiği ve ilave algılayıcılara ihtiyaç duyulmadığı için sıklıkla tercih edilmiştir. Optik akış için videodaki ardışık kareler incelenerek piksellerin yer değişimi üzerinden hareketin yönü ve hızı çıkarılabilir [47]. Bu sayede videodaki zamansal veri farklı bir şekilde ifade

edilebilmektedir. Optik akış verisi, piksel grupları bazında veya tekil pikseller bazında çıkarılabilir. Optik akış yönteminin dezavantajı ise hareketli nesne ve arka planın aynı renkte olması durumunda hareketin tespit edilememesidir [47].

Optik akış tespiti, Farneback [83] gibi yöntemlerle yapılabileceği gibi Deepflow [84], FlowNet [85] gibi derin öğrenme modelleri ile de yapılabilmektedir. Optik akış verisinden öznitelik çıkarımı, yapılan önışlemeye bağlı olarak evrişimsel sinir ağları ile veya UKSB'ler ile gerçekleştirilebilir.

Cui, Liu ve Zhang, RGB görüntüler ve Deepflow ile çıkardıkları optik akış verisi ile sürekli işaret dili tanıma üzerine çalışmışlardır [86]. ESA ile çıkardıkları uzamsal öznitelikleri belirli evrişim katmanları sonrasında birleştirip çift yönlü UKSB (Bi-LSTM) ile zamansal modelleme gerçekleştirmişlerdir. Yavaş birleştirme (*slow fusion*) şeklinde gerçekleştirdikleri birleştirme işinin işaretlerin temsilinde daha başarılı olduğunu ifade etmişlerdir.

İşaret dili tanımadaki sıklıkla kullanılan Kinect kamerası ile birlikte, işaret dili için en yoğun kullanılan veri kipleri RGB ve derinlik olmuştur. Derinlik görüntüleri ortamın ışıklandırması gibi durumlardan daha az etkilendiği ve arkaplan gürültüsünden az etkilendiği için işaret dili tanımadaki yararlı bir veri kipidir [24, 87]. Derin öğrenme çalışmalarında derinlik verisinden öznitelik çıkarımı için genellikle evrişimsel sinir ağları tercih edilmiştir.

Çalışmalarda sıklıkla kullanılan bir diğer veri kipi de eklem verisidir. Kinect kamerası ile tespit edilen beden eklemlerinin yanı sıra Leap motion ile tespit edilen el eklemleri de çalışmalarda kullanılmaktadır. Bu cihazlar gibi ekstra donanımların kullanılmadığı durumlarda ise eklem tahmini (*pose estimation*) yöntemi ile RGB görüntülerden eklem noktalarının konumları belirlenebilir [12, 88]. [89]'daki araştırmacılar, işaret dili tanımadaki kullanılmak üzere el ve kolları takip eden bir eklem tahmini sistemi geliştirmişlerdir. Eklem tahmininin dezavantajı, optik akış gibi yöntemlere göre işlem karmaşıklığının daha fazla olmasıdır.

Videonun her bir karesinden elde edilen eklem noktalarının konumları tek boyutlu bir diziye dönüştürülüp farklı video karelerinden elde edilen diziler bir araya getirilerek bu

konumlar bir zaman serisi haline getirilebilir. Bu durumda bu verinin sınıflandırılması UKSB'ler yoluyla gerçekleştirilebilir. Literatürdeki çalışmalarda eklem noktalarının birleştirilmesi ile oluşturulan bu uzam-zamansal veriden evrışimsel sinir ağları ile öznitelik çıkarımı yapıldığı da görülmüştür [25].

Derin öğrenme kullanılarak yapılan eklem tahmini yüksek doğruluk sunmakta ve uygulama öncesinde elle işaretlemeye gerek duymamaktadır. [90]'da işaret dili tanıma çalışmalarında işaret dili tanıma başarımını yükseltmek için kullanılabilecek eklem tahmin yöntemleri değerlendirilmiştir. Çalışmada iki farklı eklem tahmini yöntemi kullanarak sonuçları karşılaştırılmış ve sonraki çalışmalarda kullanılmak üzere bir temel model sunulmuştur.

Çok kipli çalışmalarda kullanılan veri kipleri kadar, bu verilerin veya veriden çıkarılan özniteliklerin sınıflandırıcının eğitimi esnasında ne şekilde birleştirildiği de önemlidir. Farklı veri kiplerinin birleştirilmesi, veri temelli, öznitelik temelli ve sınıflandırma sonucu temelli olarak yapılabilir.

Video verileri özelinde incelendiğinde veri bazında yapılan birleştirme, farklı donanımlar ile aynı anda elde edilmiş karelerin senkronize edilmesi gerektiği için uğraştırıcı olabilmektedir. Ancak öznitelik ve sınıflandırma sonucu temelli birleştirmenin aksine bu yöntemde birleştirme sonrasında öznitelik çıkarımı için tek bir evrışimsel sinir ağı yeterli olmaktadır. Aynı zamanda, veri kipleri birleştirilerek ağa verildiği için farklı veri kipleri arasındaki piksel temelli ilişkiler kolayca tespit edilebilmekte ve öznitelik çıkarımında kullanılacak evrışimsel modeller üzerinde çok fazla değişikliğe gidilmeden sınıflandırma gerçekleştirilebilmektedir [82].

Köpüklü ve diğerleri, RGB ve optik akış veri kiplerini birleştirmiş, bir karenin RGB kanallarına, önceki 3 kareden çıkarılmış optik akışları da kanal olarak ekleyerek 2 boyutlu evrışim gerçekleştiren bir evrışimsel sinir ağı ile sınıflandırma yapmışlardır [82]. Geliştirdikleri yöntemin sadece RGB ve optik akış kullanarak daha az veri kipi ile daha yüksek başarımlar elde ettiklerini ve birleştirme yöntemlerinin derinlik ve kızılötesi gibi veri kipleri ile de kullanılabileceğini ifade etmişlerdir.

Öznitelik temelli birleştirme, yapıldığı noktaya göre erken, yavaş ve geç birleştirme olarak sınıflandırılabilir [50]. *Erken birleştirmede*, ilk evrişim aşamalarından sonra birleştirme gerçekleştirilir ve birleştirme sonrasında evrişim katmanları ile devam edilir. Birleştirme, evrişim aşamalarının sonunda tam bağlı katman öncesinde yapılırsa *geç birleştirme*, birden fazla evrişim katmanında yapılırsa *yavaş birleştirme* olarak isimlendirilmiştir. Birleştirme işlemi toplama, ortalama veya maksimumu alma şeklinde yapılabileceği gibi bağlama şeklinde de yapılabilir.

Sınıflandırma sonucu temelinde yapılan birleştirmede, kural tabanlı yöntemler kullanılarak farklı veri kiplerinden elde edilen sonuçların ortalaması veya maksimumu birleştirmenin çıktısı olarak verilebilir. Bir diğer yöntem de farklı veri kiplerine ait çıktıları yeni bir sınıflandırıcı katmana vererek veri tabanlı bir çıktı elde etmektir [35].

Farklı bir çalışmada, RGB, optik akış ve kızılötesi görüntüleriyle birlikte tekrarlı 3D evrişimsel ağ kullanarak çok kipli bir hareket tanıma uygulaması geliştirilmiştir [87]. Çalışmada, araştırmacıların geliştirdikleri veri kümesinde tek başlarına %57,8 ile %80,3 arasında doğruluk veren veri kipleri çeşitli şekillerde birleştirildiğinde, en yüksek başarımlar (%83,8) tüm veri kiplerinin birlikte kullanıldığı birleştirmede elde etmişlerdir.

[35]'de Kinect ve LMK kullanılarak elde edilen RGB, derinlik ve eklem verilerinden yararlanan bir işaret dili tanıma sistemi önerilmiştir. Çalışmada, farklı uçtan uca ve çok adımlı birleştirme tekniklerini incelenmiş ve en yüksek başarımlar araştırmacıların geliştirdikleri birleştirme yöntemi ile elde edilmiştir. Bu birleştirme yönteminde her bir veri kipinden çıkarılan tekil özniteliklerin yanı sıra hepsinden ortak elde edilen öznitelikler de kullanılarak geç birleştirme yapılmıştır. Çalışma sonucunda çok kipli yöntemin tek veri kipine göre daha yüksek başarımlar gösterdiği ifade edilmiştir.

İşaret dili tanıma çalışmalarında, farklı veri kipi kullanımının yanı sıra, eller ve yüzü ayrı akışlarda inceleyen çalışmalar bulunmaktadır. [47], Kinect tarafından sunulan eklem verisindeki bilek ve dirsek konumlarını kullanarak ellerin konumunu tespit etmiş ve ellerin görüntüsünden de öznitelik çıkarılmıştır. [86]'da araştırmacılar, beden videolarının yanı sıra, baskın elin konumunu Faster R-CNN [91] nesne tespit modeli ile belirledikten sonra bu konumun RGB videosunu ve bu videodan çıkardıkları optik akışı iki ayrı akışta incelemişlerdir. Bu sayede işaretlerin daha başarılı temsil edilebildiğini ifade etmişlerdir.

5.3.2. Çok veri kipli deneyler için veri önışleme

İşaret dili tanıma konusundaki güncel çalışmalardan da anlaşılacağı üzere çalışmalar birden fazla veri kipi kullanılarak başarımın artırılmasına odaklanmıştır. Bu nedenle bu tez çalışmasında çok veri kipi kullanan (*çok kipli*) bir Türkçe işaret dili tanıma modeli geliştirilmiştir. Bunun için RGB video görüntülerinden mümkün olduğunca fazla veri kipi çıkarılarak işaret dili tanıma amaçlanmıştır. Literatürdeki çalışmalar incelendiğinde RGB veriden iskelet ve optik akış veri kiplerinin de çıkarıldığı görülmüştür. Bu tez çalışmasında bu iki veri kipinin yanı sıra, ham videolarda yüz ve ellerin takip edilip beden videosundan kırılarak daha yüksek çözünürlüklü el ve yüz videoları elde edilmiştir. Bu sayede işaret dilinde önemli olan iki iletişim kanalını da ayrı akışlarda incelemek mümkün olmuştur.

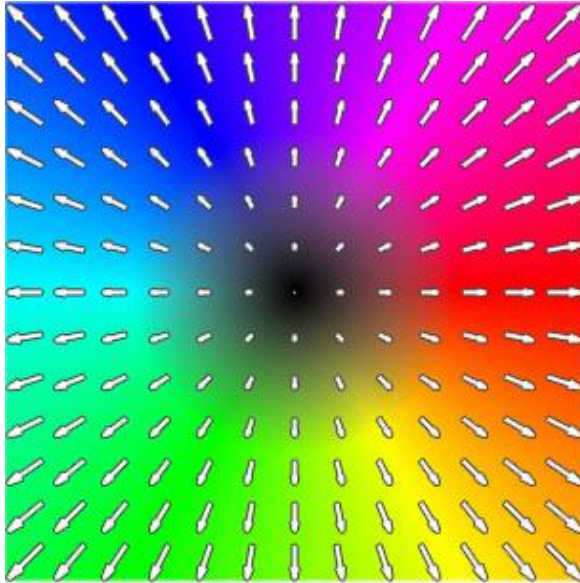
Bu tez çalışmasında, 3. grup deneylerde, ham videolardan uzamsal veri artırımına gidilmemiş, sadece zamansal veri artırımı yapılmıştır. Veri artırımını sonrası elde edilen videolardan öncelikle eklem tahmini yöntemi ile vücudun ve ellerin 2 boyutlu eklem verisi çıkarılmıştır. Bu çıkarım için Openpose [92] eklem tahmini kütüphanesi kullanılmıştır.

Openpose, evrişimsel katmanlar kullanarak beden, ayaklar, el ve yüz için 2 ve 3 boyutlu eklem tahmini yapabilen bir kütüphanedir. Genelden özele bir yaklaşım ile öncelikle beden eklemleri bulunur ve ardından beden için bulunan noktalardan yararlanılarak el parmaklarının eklemlerini ve yüzdeki belirli noktaları tespit edebilmektedir. Openpose, beden için 25, yüz için 70, her bir el için 21 eklem noktası için tahminde bulunabilmektedir [92]. Bu çalışmada, beden videosu kullanılarak çıkarılan eklem noktalarından belden üstteki 15 tanesi, eller için ise tüm eklem noktaları kullanılmıştır. Openpose her bir eklemin video üzerindeki konumunu x ve y ekseninde verebilmektedir. Sonrasında her bir eklem noktasının video üzerindeki konumu video genişlik ve yüksekliğine bağlı olarak normalleştirilmiştir.

Yüksek çözünürlük ile elde edilen işaret dili tanıma videoları, çalışmalarda modellerin girdi olarak alabileceği ölçülere küçültülmektedir. Bu esnada eller ve özellikle parmak şekillerindeki detaylar kaybolmaktadır. İşaret dilinde ellerin şekli ve yüz mimikleri de önemli olduğundan her bir elin ve yüzün ayrı akışlarda değerlendirilebilmesi için 3 farklı video daha oluşturulmuştur. Bu videoların oluşturulmasında Openpose ile elde edilen

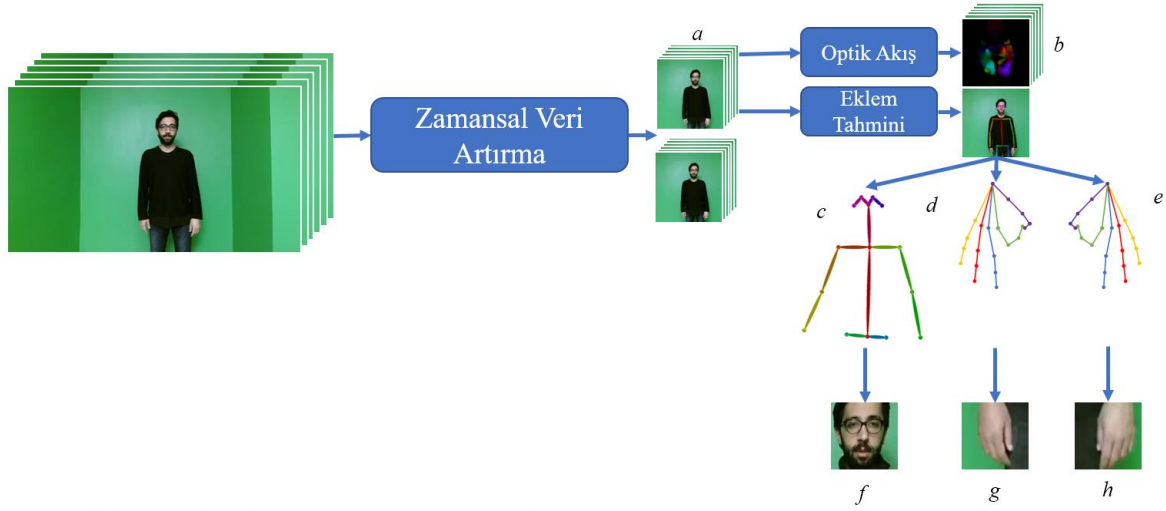
eklem noktalarından yararlanılmıştır. Yüzün videosunun çıkarılması için vücut eklem verisindeki burun noktası çevresindeki kare şekilli bir alan kırpılmıştır. Eller için ise her bir el için tespit edilmiş olan el eklem noktalarının orta noktası hesaplanmış ve bu nokta merkez olacak şekilde bir kare alan kırpılmıştır. Zamansal veri artırımında, beden videoları için önceki deneylerde en yüksek başarıımı sunan 32@224x224 parametreleri kullanılmıştır. Yüz ve eller için beden videosundaki konumlardan yararlanılarak ham videolardan çıkarılmış her biri 32 karelik videolar elde edilmiştir. Yüz videosu 80x80, eller ise 60x60 piksel boyutlarında olacak şekilde ölçeklendirilmiştir.

Optik akış, videolardaki zamansal verinin modellenmesi için hareket ve işaret dili tanıma uygulamalarında sıklıkla kullanılmaktadır. Bu tez çalışmasında, Farneback optik akış algoritmasından yararlanılarak yoğun (*dense*) optik akış çıkarılmıştır [83]. Ardından çıkarılan optik akış verisi pikselin bir sonraki karedeki yönünü ve uzaklığını ifade edecek şekilde, Şekil 5.11'deki biçimde RGB görüntüye dönüştürülmüştür.



Şekil 5.11. Optik akış renk dönüşümü

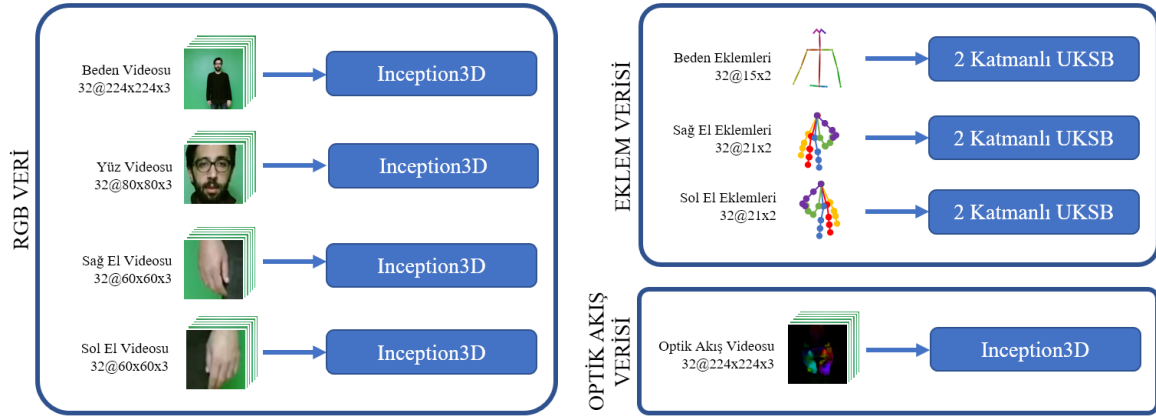
Beden videolarından çıkarılan optik akış verisi ile birlikte veri ön işleme sonrasında RGB, iskelet ve optik akış olmak üzere 3 farklı veri kipinde 8 veri akışı elde edilmiştir. Bu akışlar beden, yüz ve ellere ait RGB görüntüler, beden ve her bir el için iskelet verisi ve optik akış verisidir. Tüm akışların elde edilmesini sağlayan ön işleme aşamaları Şekil 5.12'de gösterilmiştir.



Şekil 5.12. Önişleme sonrası elde edilen veriler: a) RGB beden videosu, b) optik akış videosu, c) beden eklem verisi, d) ve e) ellerin eklem verisi, kafa ve eller için kırılmış video görüntüleri(f, g ve h)

5.3.3. Veri kiplerinin tek başına değerlendirildiği deneylerde elde edilen bulgular

Önişleme sonucunda elde edilen her bir veri kipinde öznitelik çıkarımı kendi akışında gerçekleştirilmiştir. RGB kipindeki veriler I3D ile değerlendirilmiş ve uzam-zamansal öznitelikler çıkarılmıştır. Çıkarılan eklem verisi, her bir beden karesindeki beden ve ellerin eklem verisinin karedeki konumunun x ve y türünden ifadesidir. Bu nedenle ayrıca uzamsal öznitelik çıkarımına gerek görülmemiş, zamansal değerlendirme için 2 katmanlı bir UKSB mimarisinden yararlanılmıştır. Her bir akışın girdi boyutları ve öznitelik çıkarım modelleri Şekil 5.13’de verilmiştir. Akışlar, şekilde verilen modellerle rastgele ağırlıklarla başlanarak eğitilmiştir. RGB ve optik akış verilerinin eğitiminde kullanılan I3D modeli 20 döngü süresince, eklem verilerinin eğitiminde kullanılan UKSB modeli 100 döngü süresince Adam iyileştiricisi [93] ve 10^{-3} öğrenme hızı ile eğitilmiştir.



Şekil 5.13. Her bir veri kipindeki akışlar ve öznetelik çıkarımında kullanılan derin mimariler

İşaret dillerinde kullanılan her bir iletişim kanalı, işareti anlamada farklı öneme sahiptir. Veri kiplerinin tek başlarına başarımlarının değerlendirildiği deneylerde elde edilen işaret dili tanıma sonuçlarında da bunu görmek mümkündür. Veri önışleme sonrasında elde edilen her bir akışın tek başına eğitimi sonrasında 154 sınıflık veri kümesinde gerçekleştirilen test işleminin sonuçları Çizelge 5.9’da verilmiştir.

Çizelge 5.9. Her bir akışın tek başına kullanıldığı deneylerde sınıflandırma başarımları

Veri Kipi	Akışlar	Kesinlik (%)	Duyarlılık (%)	F1-Skoru (%)	Doğruluk (%)
RGB	Beden Video	80	83	79	79,6
	Yüz	28	33	27	29,1
	Sol El	27	28	24	27,6
	Sağ El	46	51	44	45,4
Eklem	Beden İskelet	49	52	47	50,6
	Sol El İskelet	9	10	9	9,6
	Sağ El İskelet	24	28	23	23,8
Optik Akış	Beden Akış	79	81	78	78,5

Çizelge 5.9’da verilen test sonuçları incelendiğinde en yüksek doğruluğun beden RGB videosu aracılığı ile elde edildiği görülmektedir. Yüz ve sol el birbirine yakın doğruluk sunarken sağ el tek başına kullanıldığında %45,4 doğrulukla işaret dili tanıma yapılabilmektedir.

Bazı işaret dili tanıma çalışmalarının [25] aksine iskelet verisi tek başına kullanıldığında işaret dili tanıma yüksek başarımlı vermemektedir. Bunun nedeni olarak daha geniş bir veri kümesi kullanılması gösterilebilir. İşaretlerin gösteriminde yapılan hareketler arasında büyük farkların olduğu durumlarda, sadece eklemeler bakılarak tanıma gerçekleştirilebilir. Ancak işaretlerin gösteriminde yapılan hareketlerin benzer olduğu, farklılıkların sadece el hareketlerinde olduğu durumlarda sadece iskelet verisi yeterli olmayacaktır. 3. grup deneylerde veri kiplerinin tek başlarına elde ettikleri tanıma sonuçları incelendiğinde, eklem verilerinin, aynı kanaldaki video verisine göre yarı yarıya doğruluk sunabildiği görülmüştür. Bu da iskelet verisinin işaret tanıma tek başına yeterli olmadığını, farklı kanallarla birlikte kullanıldığında işaret tanıma başarımlarını artırabileceğini göstermektedir.

İşaret dili tanıma elleri inceleyen özellikle geleneksel makine öğrenmesi tabanlı birçok çalışmada, baskın elin işaret için birincil önemi olduğuna değinilmiş ve sadece baskın eli değerlendirerek işaret tanıma gerçekleştirilmiştir. Ancak 3. grup deneylerde veri kiplerinin tek başlarına elde ettikleri işaret dili tanıma sonuçları incelendiğinde baskın olmayan sol elin RGB verisi kullanılarak %27,6 doğruluk elde edildiği görülmüştür. Bu durum her ne kadar sol el işaretin gösteriminde yardımcı rolünde olsa da işaret tanıma başarımlarına olumlu etki sunabilecek veri barındırmakta olduğu yönünde yorumlanabilir.

İşaret dili tanıma çalışmalarındaki genel yönelimin gösterdiği üzere, birden çok veri kiplerinin birlikte kullanımı, işaret dili tanıma başarımlarını artırmaktadır. BosphorusSign veri kümesinin “genel” alt kümesi üzerinde yapılan deneylerin sonuçlarına göre, Türk işaret dili tanıma, tek başına en yüksek başarımlı veren veri kipleri, sırası ile; belden üzerinin kaydedildiği RGB beden videoları, beden videolarından çıkarılmış optik akış verisi ve yine bel üzeri eklem noktalarından oluşan eklem verisi akışıdır.

5.3.4. Veri kiplerinin birleşimleri ile yürütülen deneyler ve bulgular

3. grup deneylerin ilk aşamasında çeşitli veri kiplerinin tek başlarına elde ettikleri sonuçlardan sonra, bu veri kiplerini çeşitli şekillerde birleştirerek, birleşimlerin işaret dili tanıma başarımları değerlendirilmiştir. Bu deney grubunda amaç, akışları birleştirerek, tek başına kullanıldığında en yüksek doğruluğu veren RGB veri kiplerindeki beden videosu akışına göre başarımlarını artırmaktır.

Akışların birleştirilmesinde, evrişim esnasında birleştirme (*erken birleştirme - early fusion*), evrişim sonrası çıkarılan öznelilikleri birleştirme (*geç birleştirme – late fusion*) veya sınıflandırma sonucunu birleştirme (*classification based fusion*) gibi farklı yöntemler izlenebilir. Erken ve geç birleştirme mekanizmaları, uçtan uca bir model geliştiriliyorsa eğitim aşamasında gerçekleştirilir. Sınıflandırma sonucunu birleştirme ise farklı akışların eğitimi sonrasında da yapılabilir. Bunun için eğitilmiş farklı modellerin sınıflandırma sonucu alınır, çıktılar birleştirilir ve yeni bir sınıflandırıcı katmana aktararak ortak sınıflandırma sonucu elde edilir.

Çalışmanın bu noktasında, farklı akışlardan eğitilmiş model ağırlıkları bulunduğundan, sınıflandırma sonuçlarında birleştirme yöntemi kullanılmıştır. Tek başına eğitilen her bir modelden sınıflandırma çıktısı alınmış ve akışlardan darboğaz öznelilikler çıkarılmıştır. Sonrasında darboğaz öznelilikler bağlama yoluyla birleştirilerek 2 tam bağlı katmana sahip bir sınıflandırıcıya girdi olarak sunulmuştur. Akışların farklı birleşimleri sonucunda test verisinde elde edilen sınıflandırma başarımları Çizelge 5.10’da verilmiştir.

Çizelge 5.10. Akışların farklı şekillerde birleştirilmesi ile elde edilen sınıflandırma başarımları

Akışlar (Akış Sayısı)	Kesinlik (%)	Duyarlılık (%)	F1-Skoru (%)	Doğruluk (%)
Beden Video (1) - <i>Baseline Model</i>	80	83	79	79,6
Video + İskelet (2)	79	80	77	79
Video + Optik Akış (2)	85	87	84	84,9
Video + Yüz (2)	81	84	80	81,1
Video + Eller RGB (3)	79	83	78	78,5
Video + Tüm İskelet Akışları (4)	72	75	70	72,8
Video + Yüz + Eller (4)	81	86	81	81,2
Video + İskelet + Optik Akış (3)	86	88	85	85,9
Sadece Eklem Akışları (3)	41	44	40	42,1
Tüm Akışlar (8)	86	89	86	85,9
Tüm Akışlar – El Eklemleri (6)	88	91	88	88,3

Bir akışın farklı birleştirilmeleri sonucunda doğruluğa bazen olumlu bazen olumsuz etkileri görülmüştür. Eklem akışı özelinde bakılacak olursa; RGB video ve eklem akışları birleştirildiğinde, iki akış birlikte olduğunda doğruluğun düştüğü görülebilir. Eklem akışı, video ve optik akış ile birleştirildiğinde ise sadece video ve optik akışın birleştirildiği

duruma oranla doğrulukta artış olmuştur. Beden iskelet akışı, iki farklı birleştirmenin birinde doğrulukta %0,5 azalmaya neden olurken, diğerinde %1 olumlu katkı sağlamıştır.

Akışların ikili birleştirilmesinde en büyük olumlu etkiyi optik akış verisi sağlamıştır. Optik akış verisi beden videoları ile birleştirildiğinde, temel modele göre doğrulukta %4,7'lik artış elde edilmiştir.

Sadece RGB görüntülerden ibaret olan 4 video akışı değerlendirildiğinde (Beden, eller ve yüz) doğruluk temel modele göre %2'ye yakın artış göstermiş, tüm akışlar birleştirildiğinde ise %85,9'luk doğruluğa erişilmiştir.

Yapılan farklı birleştirme işlemlerinin sonucunda, ellerin eklem verisinin birleştirildiği her akışın doğruluğunu düşürdüğü görülmüştür. Bunun sebebinin, Openpose ile elde edilen iskelet verisinin kapatma (*occlusion*) nedeni ile parmaklardaki birçok eklemi tespit edememesi olduğu düşünülmektedir. Parmakların eklem verisi, birleştirildiği akışların doğruluğunu düşürdüğünden, parmak eklemleri dışındaki tüm akışlar birleştirilerek en yüksek doğruluk veren veri akışları birleşimi elde edilebilmiştir.

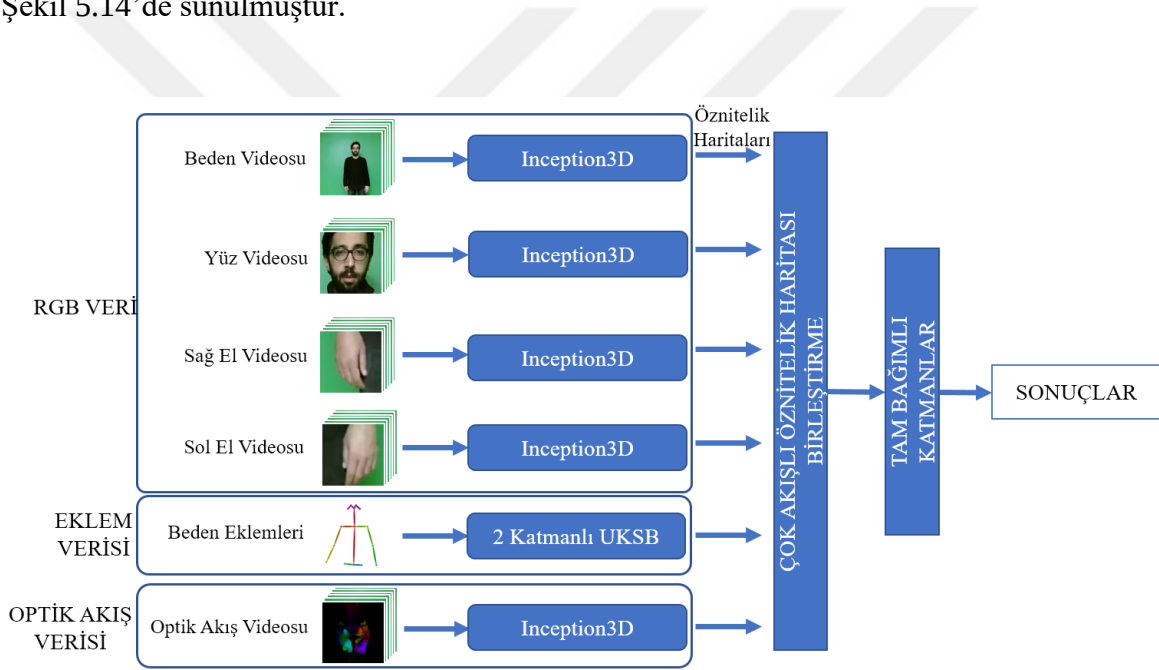
İşaret dili tanıma için kullanılabilecek veri kipleri birleştirildiğinde en yüksek doğruluğu veren akış birleşimi, beden RGB videosu, ellerin RGB videoları, yüzün RGB videosu, eklem ve optik akış verileri ile elde edilmiştir. 3 veri kipindeki (RGB, optik akış, eklem tahmini) 6 akışın kullanıldığı bu birleştirmede, doğruluk temel modele göre %8,7 artarak %88,3 olarak tespit edilmiştir.

5.4. Önerilen Çok Kipli İşaret Dili Tanıma Modeli ve Bulgular

İşaret dilinde farklı akışları birleştirerek yapılan deneylerin sonuçlarından sonra, en yüksek doğruluğu veren veri kiplerini aynı anda girdi olarak alacak bir model geliştirilmiştir. Geliştirilen modelde farklı akışlarda çıkarılan öznetelikler bir geç birleştirme mekanizması yardımı ile birleştirilerek uçtan uca işaret dili tanıma yapılmıştır. Geliştirilen işaret dili tanıma modeli, 3. grup deneylerde en yüksek doğruluğu veren 3 veri kipindeki 6 akışı kullanmaktadır.

Önerilen çok kipli model, girdi olarak 3. grup deneylerde kullanılan veri önışleme yöntemleri sonucunda elde edilen veri örneklerini almaktadır. Veri önışleme sonrasında RGB ve optik akış veri kiplerinden öznitelik çıkarımında her bir veri akışı için farklı bir I3D modeli, eklem verisinin işlenmesinde de sıralı veriyi başarılı bir şekilde modellediği bilinen UKSB'den yararlanılmıştır. Çıkarılan öznitelikler sınıflandırma öncesinde geç birleştirme yöntemi ile birleştirilmiştir. Bunun için, her bir akıştan ilgili modeller yardımı ile öznitelik haritaları çıkarılmış ve her bir akıştan çıkarılan öznitelikler bağlama yoluyla birleştirilmiştir. Birleştirilen öznitelikler tam bağı katmana aktararak sınıf tespiti gerçekleştirilmiştir. Geliştirilen çok kipli işaret dili tanıma modelinde veri kipleri, akışlar ve öznitelik çıkarma modelleri

Şekil 5.14'de sunulmuştur.

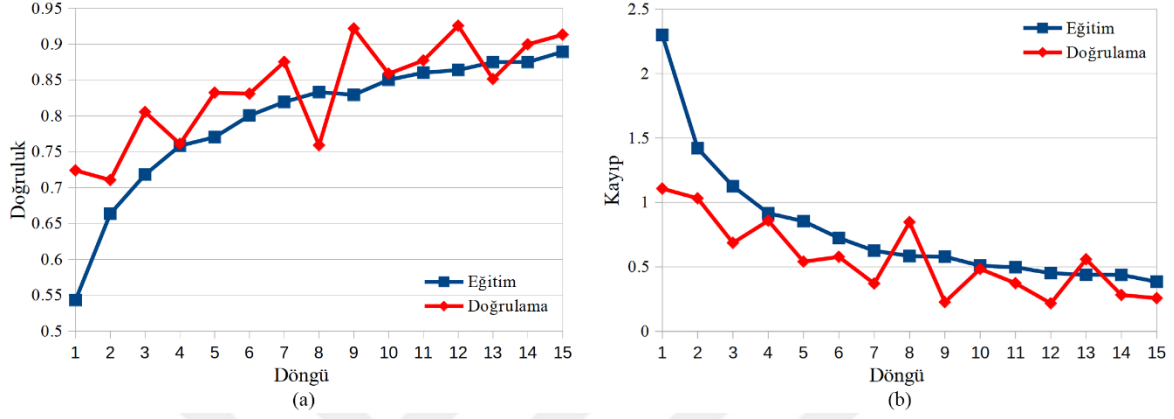


Şekil 5.14. Geliştirilen çok kipli işaret dili tanıma modeli akışları

Modelin eğitiminde mümkün olduğunca, önceki deneylerde eğitilmiş akışların ağırlıklarından transfer öğrenme yapılmıştır. Bu sayede öğrenmenin daha hızlı bir şekilde sonuçlandırılması hedeflenmiştir. Modelin eğitimi Adam iyileştirme algoritması ile 10 döngü süresince 10^{-3} , ardından 5 döngü süresince 10^{-4} öğrenme hızı parametreleri ile gerçekleştirilmiştir. Eğitim aşamasında doğruluk ve kaybın (loss) değişimini gösteren grafikler

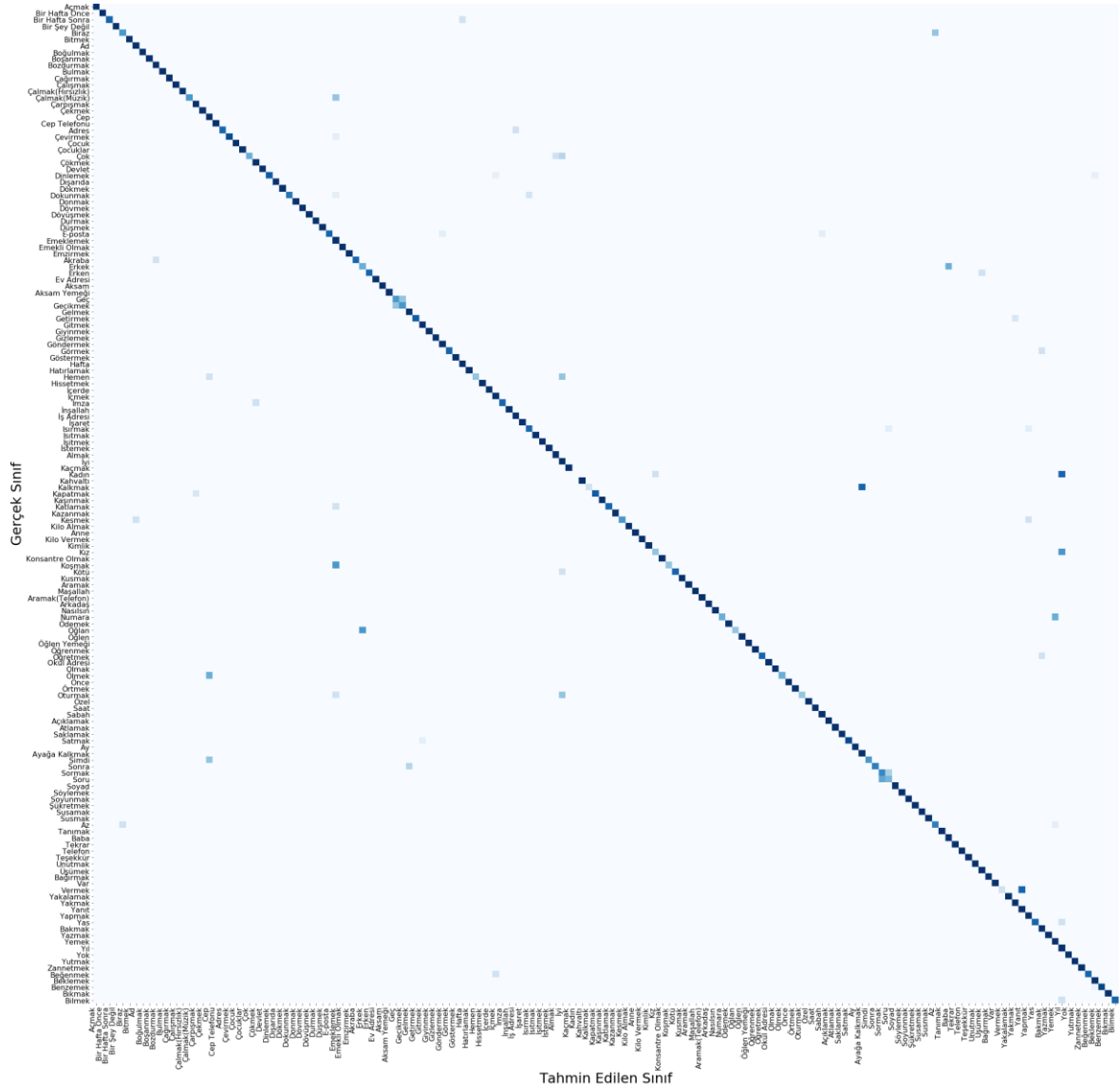
Şekil 5.15'de verilmiştir. Eğitim aşaması sonrasında 154 sınıflık test verisi üzerinde elde edilen doğruluk %89,3 olarak tespit edilmiştir. Bu sonuç, 3. grup deneylerde aynı akışların

kullanıldığı sonuca göre %1 daha yüksektir. Bu iyileştirmenin nedeni, deneylerde kullanılan sonuç temelli birleştirmenin aksine, çok verili işaret dili tanıma modelinde öznitelik temelli yapılan birleştirmedir. Test verisinde yapılan sınıflandırma sonucunda elde edilen karmaşıklık matrisi Şekil 5.16’da sunulmuştur.



Şekil 5.15. Çok akışlı modelin eğitim sürecince doğruluk (a) ve kayıp (b) değişimi

Sınıflandırma sonucu, işaretler bazında incelendiğinde modelin ayırt edemediği işaretlerin benzer veya aynı anlama gelen işaretlerden oluştuğu veya parmakların şekillerinde küçük farklılıklar içerdiği görülmüştür.

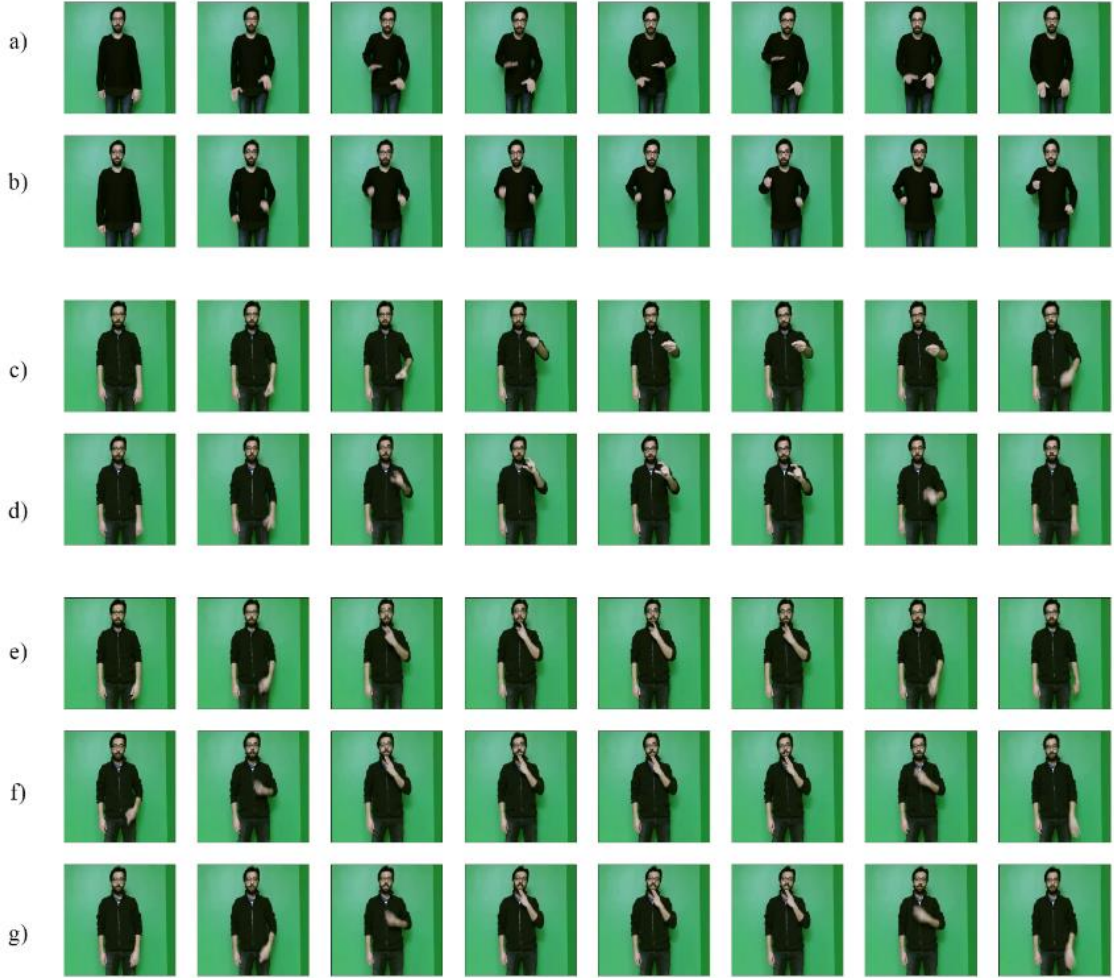


Şekil 5.16. Çok akışlı modelin karmaşıklık matrisi

İşaret tanıma sonuçlarında yapılan nitel değerlendirme sonucunda, en çok yanlış tanınan işaretlerin bazılarının Türkçe konuşma dilindeki aynı kelime kökünden gelen işaretler olduğu görülmüştür. Soru ve sormak, geç ve geç kalmak, az ve biraz işaretleri buna örnek olarak verilebilir. Bir diğer yanlış tanıma nedeni de bazı işaretlerin birden fazla anlamı olmasıdır. Örneğin erkek işareti görüşmedeki durumuna göre baba veya oğul anlamında olabilmektedir.

Yanlış tanınan bazı işaretlerde de ellerin şeklinde küçük farklılıklar olduğu da görülmüştür. Örneğin emeklemek ve koşmak işaretleri benzer hareketlere sahiptir. Ancak işaretin yapılması esnasında ellerin şekli farklılık göstermektedir. Önerilen çok kipli işaret dili

tanıma modelinin bu farklılığı tam olarak tespit edemediği görülmüştür. Benzer durum vermek ile cevap arasında, yıl ile kadın ve kız arasında da mevcuttur. Bu işaretler ile ilgili örnek görüntüler Şekil 5.17’de verilmiştir.



Şekil 5.17. Modelin yanlış sınıflandırdığı işaretler. Emeklemek (a) ve koşmak (b), Vermek (c) ve Cevap (d), Yıl (e), Kadın (f) ve Kız (g) işaretlerinin gösterimi

Önerilen çok kipli işaret dili tanıma modelinin başarımı farklı çalışmalarda elde edilen sonuçlarla karşılaştırıldığında, doğruluğun artırıldığı görülmüştür. Türkçe işaret dili için geliştirilmiş açık bir işaret dili veri kümesi bulunmamakta, çalışmaların büyük bir kısmı araştırmacılar tarafından oluşturulmuş küçük boyutlu veri setleri üzerinden gerçekleştirilmektedir. Bu nedenle önerilen çok verili işaret dili tanıma sisteminin birebir karşılaştırılmasında kullanılabilecek çok fazla çalışma bulunmamaktadır. BosphorusSign veri kümesinin farklı alt kümelerinin kullanıldığı diğer çalışmalarda elde edilen sonuçlar ile bu tez çalışmasında elde edilen sonucun karşılaştırması Çizelge 5.11’de sunulmuştur.

Çizelge 5.11. BosphorusSign veri kümesi kullanan çalışmalarda elde edilen sonuçlarla karşılaştırma

Veri Kümesi	Sınıf Sayısı	Doğruluk	Çalışma
BosphorusSign	10	%96,80	[94]
BosphorusSign “Sağlık”	33	%96,67	[95]
BosphorusSign “Genel”	154	%81,58	[96]
BosphorusSign “Genel”	154	%89,3	Bu tez çalışması

BosphorusSign veri kümesinin bu tez çalışmasında kullanılan “genel” alt kümesinde elde edilen en yüksek doğruluk oranı %81,58 olarak tespit edilmiştir [96]. Tez çalışması süresince yapılan deneylerde hem sadece RGB veriler kullanılan veri kümelerinde, hem de önerilen çok kipli işaret dili tanıma sisteminde bu doğruluk oranı artırılmıştır.

Bu tez çalışması kapsamında yapılan makale yayınında [97], önerilen çok veri kipli işaret dili tanıma modelinin farklı bir veri kümesinde de en yüksek başarıma oldukça yakın bir doğruluk sunabildiği görülmüştür. Arjantin işaret dilinde toplanmış olan LSA64 veri kümesi üzerinde yapılan eğitimlerde elde edilen sonuçların karşılaştırması Çizelge 5.12’de verilmiştir.

Çizelge 5.12. LSA64 veri kümesinde edilen sonuçların karşılaştırması

Veri Kümesi	Veri Kipleri	Doğruluk	Çalışma
LSA64	RGB, Eklem	%98,09	[88]
	RGB, Eklem	%99,91	[98]
	RGB, Eklem, Optik Akış	%99,9	Önerilen Model [97]



6. SONUÇ VE ÖNERİLER

Bu tez çalışmasında, ayrık işaret dili tanıma konusunda yapılmış çalışmalar incelenmiş ve Türkçe işaret dili tanıma amacı ile kullanılabilecek çok kipli bir işaret dili tanıma sistemi önerilmiştir. İşaret dili tanıma konusunda yapılmış geçmiş ve güncel çalışmalara yer verilmiş, çalışmalarda kullanılan yöntemler detaylandırılmıştır. Hareket ve işaret dili tanıma konusundaki güncel çalışmalarda sıklıkla kullanılan derin öğrenme yöntemleri ve veri artırma işlemleri detaylandırıldıktan sonra çeşitli parametrelerle yapılmış deneylerin sonuçları sunulmuştur. İşaret dili tanıma konusunda başarıyı yükseltmek amacı ile kullanılan çok kipli yöntemler ifade edilmiş ve sonuç olarak çok kipli bir işaret dili tanıma sistemi önerilmiştir.

Türkçe işaret dili tanıma konusunda yapılmış çalışmalar genellikle statik görüntüler üzerinde, ayrık işaret dili tanıma konusundaki çalışmalar da araştırmacılar tarafından geliştirilmiş küçük veri kümeleri üzerinde yürütülmüştür. Bu çalışmada, Türkçe işaret dili tanıma konusunda geliştirilmiş en büyük veri kümesi olan BosphorusSign veri kümesinin günlük işaretlerden oluşan “genel” isimli alt kümesi kullanılmıştır.

Derin öğrenme metotları genelleme yapabilmek için büyük miktarda veriye ihtiyaç duyarlar. Bu nedenle çalışmada öncelikle derin öğrenme yöntemlerine girdi olarak verilebilecek örneklemelerin ölçülerini ifade eden hiperparametreler ile deneyler yürütülmüştür. Literatürdeki hareket ve işaret dili tanıma çalışmalarında, örneklemelerde 9 ile 64 arasında değişen kare sayıları ve 64 ile 224 piksel arasında değişen kare boyutları kullanıldığı görülmüştür. Örneklem boyutları büyüdükçe modelin parametre sayıları artabileceğinden videodaki işareti tam olarak ifade edecek ancak işlem karmaşıklığını da çok artırmayacak bir boyut tespit edilmesi önemlidir.

Yapılan 1. grup deneylerde örneklem büyüklüğünün tespit edilmesi için çalışmalar yürütülmüştür. Bu deneylerde 8 kare sayısından başlayarak aralıklı olarak 32’ye kadar değişen aralıklarla kare sayıları denenmiştir. Kare sayıları arttıkça başarımın genellikle arttığı görülmüştür. 128, 196 ve 224 piksellik kare boyutlarında yapılan deneylerde ise kare sayısı sabitken kare büyüklüğü ile başarımlar arasında ilişki görülememiştir.

Hareket tanıma çalışmalarında veri kipi olarak RGB videolar kullanılmaktadır. Bu nedenle işaret dili tanıma çalışmalarında hareket tanıma alanında yapılmış çalışmalardan sıklıkla yararlanılmaktadır. Bu tez çalışmasında da hareket tanıma çalışmalarında kullanılan derin öğrenme modellerinden yararlanılarak farklı modellerin işaret dili tanıma başarımları incelenmiştir. 2. grup deneylerde, 1. deney grubunda kullanılan C3D modelinin yanı sıra 3DResnet ve I3D modellerinden yararlanılmıştır. Modellerin isimlerinde yer alan 3D ifadeleri bu modellerin 3 boyutlu evrişim yapabildiklerini ifade etmektedir. Bu modeller videolardaki uzamsal bilginin yanı sıra zaman boyutunun da modellenebilmesi için geliştirilmişlerdir.

Derin öğrenme çalışmalarında transfer öğrenme sıklıkla kullanılan bir yöntemdir. Transfer öğrenme ile bir veri kümesinde yapılan eğitim sonrasında elde edilen ağırlıklar başka bir çalışmada kullanılabilir. Bu sayede modellerin genelleme yeteneği artırılabilir ve eğitim daha hızlı bir şekilde tamamlanabilir. Yapılan 2. grup deneylerde C3D, 3DResnet ve I3D modelleri arasında en yüksek doğruluk, I3D modelinin transfer öğrenme yöntemi ile eğitilmesi sonucunda elde edilmiştir. Transfer öğrenmede, Kinetics hareket tanıma veri kümesinde eğitilmiş ağırlıklar kullanılmıştır. Bu model sadece RGB video görüntüleri kullanarak 154 sınıflık veri kümesinde %86,75 doğruluk vermektedir.

İşaret dillerinde işaretler farklı iletişim kanalları üzerinden aktarılabilir. Bu kanallar, el işaretlerinin dışında yüz ifadeleri ve bedenin hareketleri de olabilmektedir. Bu farklı kanallardaki verinin modellenmesi için literatürdeki çalışmalarda birden çok veri kipinin birlikte kullanıldığı işaret dili tanıma sistemleri geliştirilmiştir. Bu veri kipleri derinlik, optik akış, eklem verisi veya birden çok kameradan elde edilmiş görüntüler olabilir.

İşaret dili tanıma çalışmaları incelendiğinde optik akış kullanılarak, kişinin hareketlerinin farklı bir şekilde modellenebilmesi sağlanmıştır. Optik akış verisi, RGB videolardaki ardışık kareler incelenerek doğrudan çıkarılabilmektedir. Bu özelliği sayesinde herhangi bir ekstra donanıma gerek kalmadan kolayca hesaplanabilmektedir.

Farklı çalışmalarda, işaretlerdeki ileri-geri hareketin algılanabilmesi amacı ile derinlik bilgisi içeren görüntülerin kullanıldığı görülmüştür. Bu veri aynı zamanda işaretçinin arka plandan ayrıştırılması için de yararlıdır. Ancak bu verileri toplamak için farklı algılayıcılar içeren kameralar gereklidir.

Derinlik bilgisi sunan Kinect gibi donanımlar aynı zamanda kişinin eklem konumlarının görüntüdeki yerlerini de verebilmektedirler. Bu sayede kişinin hareketleri farklı bir veri türü ile de modellenebilmektedir. Eklem konumlarının donanımlar aracılığı ile elde edilemediği durumlarda, RGB görüntülerde eklem tahmini (*pose estimation*) yönteminden yararlanabilir. Ancak bu yöntem ile verinin çıkarılması hem hesaplama açısından maliyetlidir, hem de özellikle parmak gibi kapatma görülen eklem noktalarının tespiti konusunda yeterince başarılı değildir.

Bu tez çalışması kapsamında, RGB görüntüler üzerinden optik akış ve eklem tahmini yöntemleri kullanılarak 3 farklı veri kipini işleyecek çok kipli bir model önerilmiştir. Bu model beden, iki elin ve yüzün olmak üzere 4 RGB veriyi ve beden videolarından çıkarılan optik akış verisini I3D ile modellemektedir. Eklem verileri ise UKSB kullanılarak modellenmiştir. 6 farklı veri akışı kullanan model, 2. grup deneylerde elde edilen ağırlıklar üzerinden transfer öğrenme yöntemi ile eğitilmiştir. Eğitim sonucunda, 154 sınıflık veri kümesinde %89,3 doğruluk sunmaktadır. Bu sonuç, kullanılan veri kümesinde elde edilen en yüksek doğruluk oranıdır.

Çok akışlı modelin işaret dili tanıma sonuçları incelendiğinde modelin genellikle yanlış tespit ettiği işaretlerde üç farklı durum görülmektedir. Bu durumlar, parmak şekilleri dışında birbirine benzeyen işaretlerin olması, aynı kelime kökünden gelen işaretler ve birden fazla anlama gelen işaretler olmasıdır.

Sonuç olarak, bu tez çalışmasının Türk işaret dili tanıma alanına katkıları üç başlık altında toplanabilir. İlk olarak işaret dili tanımada kullanılacak video örneklemeleri ile ilgili en uygun hiperparametrelerin tespit edilmesi ve çeşitli derin öğrenme modellerinin başarımlarının değerlendirilmesidir. Çalışmanın ikinci katkısı, Türk işaret dilinde farklı kanalların işaret dili başarımına katkılarının belirlenmesi olmuştur. Son olarak da Türk işaret dilindeki en uygun veri akışlarını kullanan çok kipli bir işaret dili tanıma sistemi geliştirilmesidir.

Çeşitli dillerde işaret dili tanıma üzerine yapılan çalışmalar her geçen gün artmaktadır. Türk işaret dilinde de özellikle ayrık işaret dili tanıma konusunda veri kümesi hazırlama çalışmaları yürütülmektedir [99]. Bu veri kümeleri, işaret dili öğrenimine katkı sağlayacağı

gibi Türk işaret dili tanıma konusundaki çalışmalarda başarımların karşılaştırılabilmesi için de bir temel oluşturacaktır.

Sonraki çalışmalarda Türk işaret dilinde hazırlanmış olan diğer veri kümelerinde çalışmalar gerçekleştirilerek bu çalışmada elde edilen sonuçları doğrulanabilir. Aynı zamanda güncel derin öğrenme modelleri ile çalışmalar yapılarak ayrık işaret dili tanıma başarımının artırılması amaçlanabilir. Son olarak bu çalışmadaki farklı kanalların işaret dili tanıma başarımına etkileri sadece Türk işaret dili özelinde gerçekleştirilmiştir. Ancak diğer işaret dillerinde bu kanallar farklı önemlere sahip olabilirler. Bu nedenle diğer işaret dillerinde de çalışmalar yürütülerek çeşitli kanalların işaret dili tanıma başarımına etkileri değerlendirilebilir.

7. KAYNAKLAR

1. Shannon, C.E. (2001). A mathematical theory of communication. *ACM SIGMOBILE mobile computing and communications review*, 5(1), 3-55.
2. Zieren, J. and Kraiss, K.-F. (2004). *Non-intrusive sign language recognition for human-computer interaction*. Paper presented at the Proc. IFAC/IFIP/IFORS/IEA symposium on analysis, design and evaluation of human machine systems.
3. İnternet: Our Work. (2020, January). *World Federation of Deaf*, Web: <http://wfdeaf.org/our-work/> adresinden 12/1/2020 tarihinde alınmıştır.
4. Türkiye İstatistik Kurumu. (2011). *Engellilerin Sorun ve Beklentileri Araştırması 2010*. Ankara: Türkiye İstatistik Kurumu.
5. Cooper, H., Holt, B. and Bowden, R. (2011). Sign language recognition. In T. Moeslund, A. Hilton, V. Krüger, ve L. Sigal (Eds.), *Visual Analysis of Humans* (1. ed.). London: Springer, pp. 539-562.
6. Pigou, L. (2018). *Gesture and sign language recognition with deep learning*, Ghent University,
7. Ronchetti, F., Quiroga, F., Estrebou, C.A., Lanzarini, L.C. and Rosete, A. (2016). *LSA64: an Argentinian sign language dataset*. Paper presented at the XXII Congreso Argentino de Ciencias de la Computación (CACIC 2016).
8. Kang, B., Tripathi, S. and Nguyen, T.Q. (2015). *Real-time sign language fingerspelling recognition using convolutional neural networks from depth map*. Paper presented at the 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR).
9. Aloysius, N. and Geetha, M. (2020). Understanding vision-based continuous sign language recognition. *Multimedia Tools and Applications*, 1-33.
10. Eryiğit, C. (2017). *Yazılı Türkçe Dilinden Türk İşaret Diline (TİD) Makine Çevirisi Sistemi*, PhD, İstanbul Teknik Üniversitesi,
11. Zeshan, U. (2003). Aspects of Türk İşaret Dili (Turkish Sign Language). *Sign Language & Linguistics*, 6(1), 43-75.
12. Joze, H.R.V. and Koller, O. (2018). MS-ASL: A large-scale data set and benchmark for understanding american sign language. *arXiv preprint arXiv:1812.01053*.
13. Grobel, K. and Assan, M. (1997). *Isolated sign language recognition using hidden Markov models*. Paper presented at the 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation.

14. Huang, J., Zhou, W., Li, H. and Li, W. (2015). *Sign language recognition using 3D convolutional neural networks*. Paper presented at the Multimedia and Expo (ICME), 2015 IEEE International Conference on.
15. Efthimiou, E., Fontinea, S.-E., Hanke, T., Glauert, J., Bowden, R., Braffort, A.. (2010). *Dicta-sign–sign language recognition, generation and modelling: a research effort with applications in deaf communication*. Paper presented at the Proceedings of the 4th Workshop on the Representation and Processing of Sign Languages: Corpora and Sign Language Technologies.
16. Kayahan, D. and Güngör, T. (2019). *A hybrid translation system from Turkish spoken language to Turkish sign language*. Paper presented at the 2019 IEEE International Symposium on INnovations in Intelligent SysTEms and Applications (INISTA).
17. Kose, H. and Yorganci, R. (2011). *Tale of a robot: Humanoid robot assisted sign language tutoring*. Paper presented at the 2011 11th IEEE-RAS International Conference on Humanoid Robots.
18. Adamo-Villani, N. and Wilbur, R.B. (2015). *Asl-pro: American sign language animation with prosodic elements*. Paper presented at the International Conference on Universal Access in Human-Computer Interaction.
19. Camgöz, N.C., Kındıroğlu, A.A., Karabüklü, S., Kelepir, M., Özsoy, A.S. and Akarun, L. (2016). *BosphorusSign: a Turkish sign language recognition corpus in health and finance domains*. Paper presented at the Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16).
20. Li, D., Rodriguez, C., Yu, X. and Li, H. (2020). *Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison*. Paper presented at the IEEE Winter Conference on Applications of Computer Vision.
21. Büyüksaraç, B. (2015). *Sign Language Recognition By Image Analysis*, Master of Science, Middle East Technical University, Ankara.
22. Jiang, X., Satapathy, S.C., Yang, L., Wang, S.-H. and Zhang, Y.-D. (2020). A Survey on Artificial Intelligence in Chinese Sign Language Recognition. *Arabian Journal for Science and Engineering*, 1-36.
23. Ong, S.C. and Ranganath, S. (2005). Automatic sign language analysis: a survey and the future beyond lexical meaning. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 27(6), 873-91. doi:10.1109/TPAMI.2005.112

24. Kuznetsova, A., Leal-Taixé, L. and Rosenhahn, B. (2013). *Real-time sign language recognition using a consumer depth camera*. Paper presented at the Proceedings of the IEEE international conference on computer vision workshops.
25. Unutmaz, B., Karaca, A.C. and Güllü, M.K. (2019). *Turkish Sign Language Recognition Using Kinect Skeleton and Convolutional Neural Network*. Paper presented at the 2019 27th Signal Processing and Communications Applications Conference (SIU).
26. Yang, H.-D. (2014). Sign language recognition with the kinect sensor based on conditional random fields. *Sensors*, 15(1), 135-147.
27. Chuan, C.-H., Regina, E. and Guardino, C. (2014). *American sign language recognition using leap motion sensor*. Paper presented at the 2014 13th International Conference on Machine Learning and Applications.
28. Demircioğlu, B., Bülbül, G. and Köse, H. (2016). *Turkish Sign Language recognition with Leap Motion*. Paper presented at the Signal Processing and Communication Application Conference (SIU), 2016 24th.
29. Koller, O. (2020). Quantitative survey of the state of the art in sign language recognition. *arXiv preprint arXiv:2008.09918*.
30. Wang, Q., Chen, X., Zhang, L.-G., Wang, C. and Gao, W. (2007). Viewpoint invariant sign language recognition. *Computer Vision and Image Understanding*, 108(1-2), 87-97.
31. Starner, T., Weaver, J. and Pentland, A. (1998). Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12), 1371-1375.
32. Vogler, C. and Metaxas, D. (1998). *ASL recognition based on a coupling between HMMs and 3D motion analysis*. Paper presented at the Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271).
33. Pigou, L., Dieleman, S., Kindermans, P.-J. and Schrauwen, B. (2014). *Sign language recognition using convolutional neural networks*. Paper presented at the Workshop at the European Conference on Computer Vision.
34. Potter, L.E., Araullo, J. and Carter, L. (2013). *The leap motion controller: a view on sign language*. Paper presented at the Proceedings of the 25th Australian computer-human interaction conference: augmentation, application, innovation, collaboration.

35. Ferreira, P.M., Cardoso, J.S. and Rebelo, A. (2019). On the role of multimodal learning in the recognition of sign language. *Multimedia Tools and Applications*, 78(8), 10035-10056.
36. Aran, O., Ari, I., Guvensan, A., Haberdar, H., Kurt, Z., Turkmen, I.. (2007). *A database of non-manual signs in turkish sign language*. Paper presented at the 2007 IEEE 15th Signal Processing and Communications Applications.
37. Von Agris, U., Knorr, M. and Kraiss, K.-F. (2008). *The significance of facial features for automatic sign language recognition*. Paper presented at the 2008 8th IEEE International Conference on Automatic Face & Gesture Recognition.
38. Chen, F.-S., Fu, C.-M. and Huang, C.-L. (2003). Hand gesture recognition using a real-time tracking method and hidden Markov models. *Image and vision computing*, 21(8), 745-758.
39. Huang, C.-L. and Huang, W.-Y. (1998). Sign language recognition using model-based tracking and a 3D Hopfield neural network. *Machine vision and applications*, 10(5-6), 292-307.
40. Dardas, N.H. and Georganas, N.D. (2011). Real-time hand gesture detection and recognition using bag-of-features and support vector machine techniques. *IEEE Transactions on Instrumentation and measurement*, 60(11), 3592-3607.
41. Ye, J., Yao, H. and Jiang, F. (2004). *Based on HMM and SVM multilayer architecture classifier for Chinese sign language recognition with large vocabulary*. Paper presented at the Image and Graphics (ICIG'04), Third International Conference on.
42. Zaki, M.M. and Shaheen, S.I. (2011). Sign language recognition using a combination of new vision based features. *Pattern Recognition Letters*, 32(4), 572-577.
43. Hakan, H. (2005). *Real time Turkish sign language recognition system from video using hidden Markov models*, Masters, Yıldız Teknik Üniversitesi,
44. Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 1097-1105.
45. Koller, O., Ney, H. and Bowden, R. (2015). *Deep learning of mouth shapes for sign language*. Paper presented at the Proceedings of the IEEE International Conference on Computer Vision Workshops.
46. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D.. (2014). Going deeper with convolutions. arXiv 2014. *arXiv preprint arXiv:1409.4842*, 1409.

47. Zhang, S., Meng, W., Li, H. and Cui, X. (2019). Multimodal Spatiotemporal Networks for Sign Language Recognition. *IEEE Access*, 7, 180270-180280.
48. Simonyan, K. and Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199*.
49. Ji, Y., Kim, S. and Lee, K.-B. (2017). *Sign Language Learning System with Image Sampling and Convolutional Neural Network*. Paper presented at the Robotic Computing (IRC), IEEE International Conference on.
50. Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R. and Fei-Fei, L. (2014). *Large-scale video classification with convolutional neural networks*. Paper presented at the Proceedings of the IEEE conference on Computer Vision and Pattern Recognition.
51. Koller, O., Zargaran, O., Ney, H. and Bowden, R. (2016). *Deep Sign: Hybrid CNN-HMM for Continuous Sign Language Recognition*. Paper presented at the Proceedings of the British Machine Vision Conference 2016.
52. Gers, F.A., Schmidhuber, J. and Cummins, F. (1999). Learning to forget: Continual prediction with LSTM.
53. Yang, S. and Zhu, Q. (2017). *Continuous Chinese sign language recognition with CNN-LSTM*. Paper presented at the Ninth International Conference on Digital Image Processing (ICDIP 2017).
54. Liu, T., Zhou, W. and Li, H. (2016). *Sign language recognition with long short-term memory*. Paper presented at the 2016 IEEE International Conference on Image Processing (ICIP).
55. Tran, D., Bourdev, L., Fergus, R., Torresani, L. and Paluri, M. (2015). *Learning spatiotemporal features with 3d convolutional networks*. Paper presented at the Proceedings of the IEEE international conference on computer vision.
56. Carreira, J. and Zisserman, A. (2017). *Quo vadis, action recognition? a new model and the kinetics dataset*. Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.
57. Hara, K., Kataoka, H. and Satoh, Y. (2017). *Learning spatio-temporal features with 3D residual networks for action recognition*. Paper presented at the Proceedings of the IEEE International Conference on Computer Vision.
58. Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

59. Camgoz, N.C., Koller, O., Hadfield, S. and Bowden, R. (2020). *Sign language transformers: Joint end-to-end sign language recognition and translation*. Paper presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.
60. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N.. (2017). Attention is all you need. *arXiv preprint arXiv:1706.03762*.
61. Camgoz, N.C., Hadfield, S., Koller, O. and Bowden, R. (2017). *Subunets: End-to-end hand shape and continuous sign language recognition*. Paper presented at the Proceedings of the IEEE International Conference on Computer Vision.
62. Mitchell, T. (1997). Introduction to machine learning. *Machine Learning*, 7, 2-5.
63. Alpaydin, E. (2014). *Introduction to Machine Learning* (Third edition). London: The MIT Press.
64. Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep learning* (Vol. 1): MIT press Cambridge.
65. Russell, S. and Norvig, P. (2016). *Artificial intelligence: a modern approach* (3rd Edition): Pearson.
66. Öztemel, E. (2003). *Yapay sinir ağları*. Istanbul: Papatya Yayıncılık.
67. Mitchell, T.M. (1997). *Machine learning*. Burr Ridge, IL: McGraw Hill.
68. Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61, 85-117.
69. LeCun, Y., Bengio, Y. and Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
70. Zeiler, M.D. and Fergus, R. (2014). *Visualizing and understanding convolutional networks*. Paper presented at the European conference on computer vision.
71. LeCun, Y., Kavukcuoglu, K. and Farabet, C. (2010). *Convolutional networks and applications in vision*. Paper presented at the Proceedings of 2010 IEEE international symposium on circuits and systems.
72. Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*: O'Reilly Media.
73. Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.

74. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1), 1929-1958.
75. Ioffe, S. and Szegedy, C. (2015). *Batch normalization: Accelerating deep network training by reducing internal covariate shift*. Paper presented at the International conference on machine learning.
76. Zhu, X., Liu, Y., Li, J., Wan, T. and Qin, Z. (2018). *Emotion classification with data augmentation using generative adversarial networks*. Paper presented at the Pacific-Asia conference on knowledge discovery and data mining.
77. Peng, X., Sun, B., Ali, K. and Saenko, K. (2015). *Learning deep object detectors from 3d models*. Paper presented at the Proceedings of the IEEE International Conference on Computer Vision.
78. Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
79. Feichtenhofer, C., Pinz, A. and Zisserman, A. (2016). Convolutional two-stream network fusion for video action recognition.
80. He, K., Zhang, X., Ren, S. and Sun, J. (2016). *Deep residual learning for image recognition*. Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.
81. Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S.. (2017). The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.
82. Köpüklü, O., Kose, N. and Rigoll, G. (2018). *Motion Fused Frames: Data Level Fusion Strategy for Hand Gesture Recognition*. Paper presented at the CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, Utah, United States.
83. Farnebäck, G. (2003). Two-Frame Motion Estimation Based on Polynomial Expansion. *Image Analysis*, 363-370.
84. Weinzaepfel, P., Revaud, J., Harchaoui, Z. and Schmid, C. (2013). *DeepFlow: Large displacement optical flow with deep matching*. Paper presented at the Proceedings of the IEEE international conference on computer vision.
85. Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V.. (2015). *FlowNet: Learning optical flow with convolutional networks*. Paper presented at the Proceedings of the IEEE international conference on computer vision.

86. Cui, R., Liu, H. and Zhang, C. (2019). A deep neural framework for continuous sign language recognition by iterative training. *IEEE Transactions on Multimedia*, 21(7), 1880-1891.
87. Molchanov, P., Yang, X., Gupta, S., Kim, K., Tyree, S. and Kautz, J. (2016). *Online detection and classification of dynamic hand gestures with recurrent 3d convolutional neural network*. Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.
88. Konstantinidis, D., Dimitropoulos, K. and Daras, P. (2018). *Sign language recognition based on hand and body skeletal data*. Paper presented at the 2018-3DTV-Conference: The True Vision-Capture, Transmission and Display of 3D Video (3DTV-CON).
89. Charles, J., Pfister, T., Everingham, M. and Zisserman, A. (2014). Automatic and efficient human pose estimation for sign language videos. *International Journal of Computer Vision*, 110(1), 70-90.
90. Gattupalli, S., Ghaderi, A. and Athitsos, V. (2016). *Evaluation of deep learning based pose estimation for sign language recognition*. Paper presented at the Proceedings of the 9th ACM International Conference on Pervasive Technologies Related to Assistive Environments.
91. Ren, S., He, K., Girshick, R. and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*.
92. Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E. and Sheikh, Y. (2018). OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. *arXiv preprint arXiv:1812.08008*.
93. Kingma, D.P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
94. Özdemir, O., Kindiroglu, A.A. and Akarun, L. (2018). *Isolated sign language recognition with fast hand descriptors*. Paper presented at the 2018 26th Signal Processing and Communications Applications Conference (SIU).
95. Camgöz, N.C., Kindiroğlu, A.A. and Akarun, L. (2016). *Sign language recognition for assisting the deaf in hospitals*. Paper presented at the International Workshop on Human Behavior Understanding.
96. Kindiroglu, A.A., Ozdemir, O. and Akarun, L. (2019). *Temporal Accumulative Features for Sign Language Recognition*. Paper presented at the Proceedings of the IEEE International Conference on Computer Vision Workshops.

97. Gündüz, C. and POLAT, H. (2021). Turkish sign language recognition based on multistream data fusion. *Turkish Journal of Electrical Engineering & Computer Sciences*, 29(2).
98. Escobedo, E., Ramirez, L. and Camara, G. (2019). *Dynamic Sign Language Recognition Based on Convolutional Neural Networks and Texture Maps*. Paper presented at the 2019 32nd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI).
99. Sincan, O.M. and Keles, H.Y. (2020). AUTSL: A Large Scale Multi-Modal Turkish Sign Language Dataset and Baseline Methods. *IEEE Access*, 8, 181340-181355.







EK-1. Veri kümesindeki sınıflar

Sınıf Numarası	Sınıf Etiketi	Sınıf Numarası	Sınıf Etiketi
5	Açıklamak	117	Boşanmak
10	Açmak	120	Bozdurmak
11	Ad	123	Bulmak
14	Adres	130	Çağırarak
21	Akraba	131	Çalışmak
22	Aksam	132	Çalmak(Hırsızlık)
23	Aksam Yemeği	133	Çalmak(Müzik)
32	Almak	134	Çarpışmak
37	Anne	137	Çekmek
40	Aramak	138	Cep
41	Aramak(Telefon)	139	Cep Telefonu
42	Arkadaş	142	Çevirmek
50	Atlamak	145	Çocuk
51	Ay	146	Çocuklar
52	Ayağa Kalkmak	147	Çok
56	Az	149	Çökmek
57	Baba	163	Devlet
60	Bağırarak	166	Dinlemek
63	Bakmak	171	Dışarıda
79	Beğenmek	176	Dökmek
83	Beklemek	179	Dokunmak
90	Benzemek	182	Donmak
93	Bıkmak	186	Dövmek
99	Bilmek	187	Dövüşmek
100	Bir Hafta Önce	188	Durmak
101	Bir Hafta Sonra	190	Düşmek
102	Bir Şey Değil	196	E-posta
106	Biraz	202	Emeklemek
108	Bitmek	204	Emekli Olmak
113	Boğulmak	207	Emzirmek

EK-1. (devam) Veri kümesindeki sınıflar

Sınıf Numarası	Sınıf Etiketi	Sınıf Numarası	Sınıf Etiketi
214	Erkek	331	Kahvaltı
215	Erken	333	Kalkmak
219	Ev Adresi	348	Kapatmak
235	Geç	357	Kaşınmak
236	Gecikmek	360	Katlamak
238	Gelmek	363	Kazanmak
240	Getirmek	367	Kesmek
242	Gitmek	369	Kilo Almak
243	Giyinmek	370	Kilo Vermek
245	Gizlemek	371	Kimlik
248	Göndermek	375	Kız
249	Görmek	381	Konsantre Olmak
251	Göstermek	384	Koşmak
262	Hafta	385	Kötü
274	Hatırlamak	396	Kusmak
279	Hemen	409	Maşallah
285	Hissetmek	425	Nasılsın
287	İçerde	437	Numara
288	İçmek	442	Ödemek
301	İmza	443	Oğlan
303	İnşallah	444	Öğlen
306	İş Adresi	445	Öğlen Yemeği
308	İşaret	447	Öğrenmek
312	Isırmak	448	Öğretmek
313	Isıtmak	452	Okul Adresi
314	İşitmek	454	Olmak
317	İstemek	455	Ölmek
320	İyi	459	Önce
326	Kaçmak	465	Örtmek
327	Kadın	468	Oturmak

EK-1. (devam) Veri kümesindeki sınıflar

Sınıf Numarası	Sınıf Etiketi	Sınıf Numarası	Sınıf Etiketi
469	Özel	574	Telefon
497	Saat	578	Teşekkür
498	Sabah	589	Unutmak
502	Saklamak	590	Üşümek
507	Satmak	603	Var
524	Simdi	607	Vermek
543	Sonra	614	Yakalamak
545	Sormak	617	Yakmak
546	Soru	618	Yanıt
547	Soyad	619	Yapmak
548	Söylemek	625	Yas
549	Soyunmak	634	Yazmak
553	Şükretmek	635	Yemek
556	Susamak	638	Yıl
557	Susmak	642	Yok
564	Tanımak	647	Yutmak
573	Tekrar	650	Zannetmek



GAZİLİ OLMAK AYRICALIKTIR.