GAZİOSMANPAŞA BİLİMSEL ARAŞTIRMA DERGİSİ (GBAD)
Gaziosmanpasa Journal of Scientific Research
ISSN: 2146-8168
http://dergipark.gov.tr/gbad
Araştırma/Derleme Makalesi (Research/Review Article)

Cilt/Volume : 6
Sayı/Number: 3
Yıl/Year: 2017
Sayı/Pages: 85-104

***Alınış tarihi (Received)*:** 24.07.2017
***Kabul tarihi (Accepted)*:** 11.12.2017

Baş editor/*Editors-in-Chief*: ***Ebubekir ALTUNTAŞ***
Alan editörü/*Area Editor*: ***Bülent TURAN***

Derin Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri

Özkan İNİK[a,*] **Erkan ÜLKER**[b]

[a] *Bilgisayar Mühendisliği Bölümü, Gaziosmanpaşa Üniversitesi, Tokat, Türkiye*
[b] *Bilgisayar Mühendisliği Bölümü, Selçuk Üniversitesi, Konya, Türkiye*
[*]*: Sorumlu yazar, e-posta: ozkan.inik@gop.edu.tr*

ÖZET: Klasik Makine öğrenme teknikleri ile bir model tanımlama veya makine öğrenimi sistemi kurmak için öncelikle özellik vektörünün çıkarılması gerekmektedir. Özellik vektörünün çıkarılması için alanında uzman kişilere ihtiyaç duyulmaktadır. Bu işlemler hem çok zaman almakta hem de uzmanı çok meşgul etmektedir. Bu sebeple bu teknikler, ham bir veriyi ön işlem yapmadan ve uzman yardımı olmadan işleyemezler. Derin Öğrenme makine öğrenimi alanında çalışanların uzun yıllar boyunca uğraştığı bu sorunu ortadan kaldırarak büyük ilerleme sağlamıştır. Çünkü derin ağlar geleneksel makine öğrenmesi ve görüntü işleme tekniklerinin aksine öğrenme işlemini ham veri üzerinde yapmaktadır. Ham veriyi işlerken gerekli bilgiyi farklı katmanlarda oluşturmuş olduğu temsillerle elde etmektedir. Derin Öğrenme ilk defa 2012 yılında nesne sınıflandırma için yapılan, büyük ölçekli görsel tanıma (ImageNet) yarışmasında elde ettiği başarı ile dikkatleri üzerine çekmiştir. Derin Öğrenmenin temelleri geçmişe dayansa da özellikle son yıllarda popüler olmasının en önemli sebeplerinden ilki eğitim için yeteri kadar verinin olması ve ikinci olarak bu veriyi işleyecek donanımsal alt yapının olmasıdır. Bu çalışmada Derin Öğrenme hakkında detaylı bilgi verilmiştir. Evrişimsel Sinir Ağı(ESA) mimarisinin katmanları olan Konvolüsyon, Havuzlama, ReLu, DropOut, Tam bağlantılı ve Sınıflandırma katmanı hakkında açıklamalar yapılmıştır. Ayrıca Derin Öğrenmede temel mimariler olarak kabul edilebilecek AlexNet, ZFNet, GoogLeNet, Microsoft RestNet ve R-CNN mimarileri anlatılmıştır.

Anahtar Kelimeler *— Derin Öğrenme, ESA, Konvolüsyon, Havuzlama, AlexNet, ZFNet, GoogLeNet, Microsoft RestNet, R-CNN*

Deep Learning and Deep Learning Models Used in Image Analysis

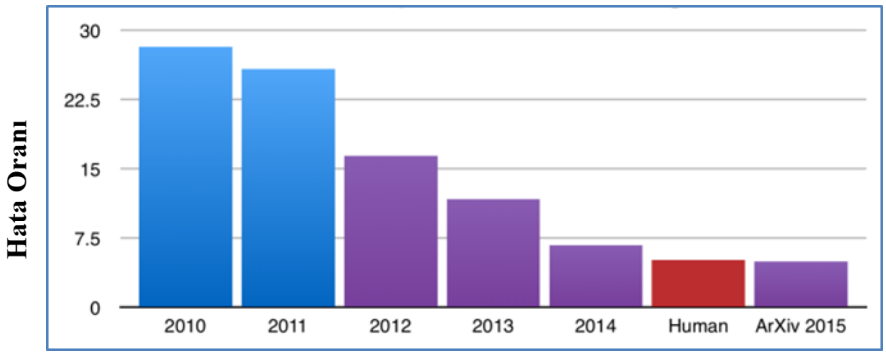
ABSTRACT: In order to establish a machine learning system or model definition with classical machine learning techniques, it is necessary to first extract the feature vector. In order to extract the feature vector, Experts are needed. These transactions both take a lot of time and make the expert very busy. For this reason, these techniques can not process a raw data without pre-processing and without Expert help. Deep learning has made tremendous progress by eliminating this problem, which has been a challenge for many years in the field of machine learning. Because deep nets do the learning process on raw data, unlike traditional machine learning and image processing techniques. It obtains the necessary information from the representations that it formed in different layers. Deep learning first attracted attention with its success in the Large Scale Visual Recognition (ImageNet) competition for object classification in 2012. Although, the foundations of Deep Learning depend on the past, it has become popular in recent years mainly due to two reasons. The first is the existence of as much data as training. The second is the hardware infrastructure that will process this data. In this study, information about deep learning was given and detailed information about layers of Convolution, Pooling, ReLu, Fully Connected and Classification layers, which are layers of Convolution Neural Network (CNN) architecture. It also describes AlexNet, ZFNet, GoogLeNet, Microsoft RestNet and Region with Convolution Neural Network (R-CNN) architectures, which can be considered as basic architects for Deep learning.

Keywords *— Deep Learning, CNN, Convolution, Pooling, AlexNet, ZFNet, GoogLeNet, Microsoft RestNet, R-CNN*

1. Giriş

Yapay zekâ ve alt dalları ile ilgilenen araştırmacılar her zaman daha zeki sistemlerin tasarlanmasını hedeflemektedirler. İnsan düşünce yapısını ve karar verme yetisini modellemek, kuşkusuz bu hedeflerin en önemlisidir. Bu amaçla ilk defa McCulloch-Pitts tarafından insan sinir sisteminden esinlenerek beyin fonksiyonlarının işleyişinin mantıksal olarak hesaplayan bir model ortaya konulmuştur(McCulloch and Pitts 1943). Bu aynı zamanda insan sinir sisteminin bir taklidi olan Yapay Sinir Ağlarının(YSA) temelini oluşturmuştur. Daha sonraki süreçlerde Perceptron (Rosenblatt 1958, Rosenblatt 1962), Adaptive linear element (ADALINE)(Widrow and Hoff 1960) gibi modeller ortaya konuldu. Oluşturulan bu doğrusal modellerin en büyük dezavantajı, XOR gibi doğrusal olmayan problemleri çözememeleridir(Minsky 1969). Bu yüzden o yıllarda yapay sinir ağı temelli yöntemlere ilgi azaldı. 1980'lerde, sinir ağı araştırmaları yeniden paralel dağıtık işlem(Rumelhart, McClelland et al. 1986, McClelland, Rumelhart et al. 1995) olarak ortaya çıktı ve bugünün Derin Öğrenme (LeCun, Bengio et al. 2015) temeli de o yıllarda ortaya atılmış oldu. O yıllarda yapay sinir ağlarını eğitmek için geri yayılım algoritması(Rumelhart, Hinton et al. 1986, LeCun 1987) başarıyla kullanılmış ve bu kullanım yaygınlaştırılmıştır. 2006 yılında Geoffrey Hinton, derin sinir ağlarının açgözlü katmanlı ön eğitim yöntemi ile etkili bir şekilde eğitilebileceğini göstermiştir(Hinton, Osindero et al. 2006). Diğer araştırma grupları, aynı stratejiyi birçok başka derin ağları eğitmek için kullanmıştır(Bengio, Lamblin et al. 2007, Ranzato, Poultney et al. 2007). Daha iyi performans sergileyen sinir ağlarını tasarlamanın yolu daha derin ağların kurulması gerektiği ve derinliklerin teorik önemine dikkat çekilmesi için "Derin Öğrenme" teriminin kullanılması yaygınlaştırılmıştır(Bengio and LeCun 2007, Delalleau and Bengio 2011, Montufar 2014, Pascanu, Gülçehre et al. 2014).

Derin Öğrenme ilk defa 2012 yılında bilim dünyasında büyük etki oluşturmuştur. Nesne tanımlama alanında en büyük yarışma olan Büyük Ölçekli Görsel Tanıma Yarışması(ImageNet)(Competition 2012) o yıl derin öğrenmede temel mimari kabul edilen Evrişimsel Sinir Ağı(ESA)(Krizhevsky, Sutskever et al. 2012) ile kazanıldı. Bu Derin Öğrenmenin inanılmaz bir yükselişi olmuştur. Çünkü yarışmada %26,1 olan Top-5 hata oranı %15,3(Şekil 1) gibi bir orana düşürüldü(Krizhevsky, Sutskever et al. 2012). Derin Öğrenmedeki ilerlemeler Top-5 hata oranını %3,6'ya kadar düşürmüştür (Şekil 1).



Şekil 1. Yıllara göre imageNet yarışmasının Top-5 hata oranları(Julia 2016).
***Figure 1**. Top-5 error rates for the imageNet competition over the years (Julia 2016).*

Derin öğrenme ile ilgili ilk çalışmalar çok geçmişe dayanmasına rağmen son yıllarda başarılı bir şekilde kullanılmasının başlıca sebeplerinden biri yeteri kadar verinin olmasıdır. Günümüzde karmaşık görevlerde kullanılan Derin Öğrenme modelleri eğiten algoritmalar, 1980'lerde oyuncak problemlerini çözmek için kullanılan öğrenme algoritmalarıyla hemen

hemen aynıdır, ancak bu algoritmalarla hazırladığımız modeller çok derin mimarilerin eğitimini basitleştiren değişiklikler yapmıştır. Bunun yanında bir diğer önemli yeni gelişme günümüzde bu algoritmalara başarılı olmak için ihtiyaç duydukları kaynakların sağlanmasıdır. Bu kaynakların ilkini oluşturan veri, toplumun dijitalleşmesinin gittikçe artması ile sağlanmaktadır. Bilgisayarlarda gerçekleştirilen faaliyetlerin artmasıyla yapılan işlemler daha çok kaydedilmektedir. Bilgisayarlar daha fazla ağa bağlandığından, bu kayıtları merkezileştirmek ve bunları makine öğrenmesi uygulamaları için uygun bir veri kümesi haline getirmek daha kolay hale gelmiştir. Artan bu veri yapısı son yıllarda **"Büyük Veri (Big Data)"** adı altında yeni bir alan oluşturmuştur. Büyük veri ile makine öğrenimi çok daha kolay hale gelmiştir. Derin öğrenmenin daha çok popüler olmasının bir diğer nedeni ise, günümüzde daha büyük modelleri çalıştırmak için hesaplama kaynaklarının var olmasıdır. Yapay sinir ağlarında(YSA) gizli katmanların sunulmasıyla kullanılan bellek hafızası ve hesaplama için işlemci kapasitesi artmıştır. Gizli katman sayısının arttırılmasıyla derinleştirilen ağ, daha büyük belleklere sahip daha hızlı bilgisayar ihtiyacını meydana getirir. Örneğin derin bir ağın giriş görüntüsü 220x220x3(renkli bir görüntü) boyutta ve bu görüntülerden 96500 adet varsa, eğitim verisinin oluşturulması için ilk aşamada 220*220*3*96500 baytlık bellek hafızasına ihtiyaç duyulmaktadır. Bazı veri kümelerinin milyonlar seviyesinde(örneğin İmageNet yarışması için kullanılan veri seti 1.2milyon) olduğu düşünülürse kullanılacak bellek hafizasının önemi ortaya çıkmaktadır. Aynı şekilde birden fazla gizli katman sahip bir ağın eğitilmesi esnasında geriye yayılım algoritmasının kullanılması için yapılan hesaplamalar paralel işlemciler ile daha hızlı gerçekleştirilebilir. Bu sebeple derin ağların eğitimi için Central Processing Unit(CPU) yerine genel amaçlı kullanılmak üzere ortaya çıkan Graphic Processing Unit(GPU) kullanılmaktadır.

Büyük veri ve GPU'ların geliştirilmesiyle farklı Derin Öğrenme modelleri tasarlanmasına olanak sağlanmıştır. Tasarlanan bu modeller giriş verisinden kullanıcı tarafından belirlenen özellikler olmadan öğrenme işlemini kendisi yapmaktadır(Bengio, Courville et al. 2013). Bu öğrenme işlemini farklı katmanlarda veriye ait farklı özellikler keşfetmekle elde etmektedir. Bu mimarilerin temel modeli ESA olarak kabul edilir. ESA'lar görüntü sınıflandırma, nesne tanımlama, görüntü segmentasyon v.b problemlerde başarılı bir şekilde uygulanmaktadır. ESA'lar YSA'ların geliştirilmiş halidir. YSA'lardaki gizli katman sayılarının daha da arttırılması sonucu derinleşen ağ ESA olarak tanımlanabilir. ESA'daki bu derinlik 2 boyutlu filtrelerin kullanılmasıyla gerçekleştirilmiştir. Derinlikteki bu farklılığa ek olarak ESA'lar hiyerarşik bir yapıda öğrenme işlemini gerçekleştirir. Şekil 2'de gösterildiği gibi ESA'da bir nesnenin tanımlanması her bir katmanda o nesneye ait bir alt özelliğin keşfedilmesiyle mümkün olmaktadır. Son olarak, ESA'ların YSA'lardan ayıran temel fark, ESA'ların derinleşen ağ yapısının eğitilmesi aşamasında ezberlemeyi önlemek için kullanmış olduğu DropOut(Srivastava, Hinton et al. 2014) yöntemidir. Bu yöntem, eğitim aşamasında her iterasyonda ağa ait bazı düğümleri gelişi güzel kaldırarak ezberlemeyi önlemeyi amaçlamaktadır. DropOut ile ilgili bilgi Bölüm 2'de verilmiştir.

1.2. Derin Öğrenme ile İlgili Yapılan Çalışmalar

Derin öğrenmenin ortaya çıkışıyla görüntü analizi, ses analizi, robotik, otonom araçlar, gen analizleri, kanser teşhisleri ve sanal gerçeklik v.b. birçok alanda kullanılmaya başlandı. Bu derece çok yaygın bir alanda kullanılması en büyük sebebi problemlerin çözümünde elde ettiği yüksek doğruluktur. Hatta ses tanımlama, görüntü tanımlama gibi bazı problemlerde insan performansının üzerine çıkılmıştır. Yapılan bazı çalışmalara bakılırsa;

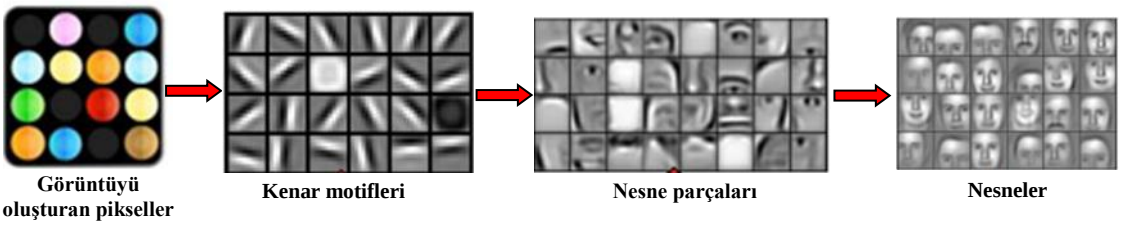
Görüntülerin renklendirilmesinde Derin Öğrenme kullanılmıştır(Hwang and Zhou , Cheng, Yang et al. 2015, Larsson, Maire et al. 2016, Zhang, Isola et al. 2016). Bu çalışmalardaki amaç siyah beyaz bir resmin renklendirilmesidir. Siyah beyaz resimlerin renklendirilmesine benzer olarak geçmişte renksiz olarak çekilen filmlerin renklendirilmesi de Derin Öğrenme ile yapılmıştır. Sessiz bir videoda insanların dudak hareketinden konuşma tahmininden yola çıkılarak Derin Öğrenme ile sessiz videonun seslendirilmesi yapılmıştır(Assael, Shillingford et al. 2016). Dudak hareketini okuyarak tahminde bulanan insanların başarısı %52 iken derin öğrenme ile yapılan bu çalışmada başarı oranı %93'tür. Bir resim taslağında resmin elde edilmesi, bir çizimden nesnenin şeklinin elde edilmesi veya harita taslağından gerçek haritanın elde edilmesi Derin Öğrenme ile mümkün hale gelmiştir(Isola, Zhu et al. 2016). Yapılan farklı bir çalışmada bir resimdeki insanlara ait bakışın değiştirilmesi Derin Öğrenme ile yapılmıştır(Ganin, Kononenko et al. 2016). Facebook ve Google gibi büyük şirketlerinde kullanmış olduğu görüntülerin etiketlenmesi, obje tanımlanması ve görüntülerin sınıflandırılması Derin Öğrenme modelleri ile gerçekleştirilmiştir(Kiros, Salakhutdinov et al. 2014, Mao, Xu et al. 2014, Donahue, Anne Hendricks et al. 2015, He, Zhang et al. 2015, Long, Shelhamer et al. 2015, Ren, He et al. 2015, Venugopalan, Rohrbach et al. 2015, Redmon, Divvala et al. 2016). İnsanların anlık hareketi ve duruşunun tahmini için Derin Öğrenmede çalışmalar yapılmıştır(Cao, Simon et al. 2016). Yapılan çalışmada insan iskelet yapısındaki hareketler 2 boyutlu olarak tahmin edilmiştir. Normal bir yazının el yazısına dönüştürülmesi Derin Öğrenme ile gerçekleştirilmiştir(Graves 2013). İnsanların eski konuşmalarından sentezleme yapılarak hiç konuşmadığı bir metnin canlı bir şekilde okuyacak şekilde seslendirilmesi için Derin Öğrenme yapısı kullanılmıştır. Örnek olarak Barack Obama'nın sentezlenmesi (Suwajanakorn, Seitz et al. 2017) yapmış olduğu çalışmada verilmiştir. Derin Öğrenme ile insan gibi oyun oynayan modeller tasarlanmıştır(Mnih, Kavukcuoglu et al. 2013, Lillicrap, Hunt et al. 2015, Mnih, Kavukcuoglu et al. 2015, Silver, Huang et al. 2016). Yapılan bu çalışmalar ek olarak ses tanımlamada(Hinton, Deng et al. 2012, Graves, Mohamed et al. 2013, Amodei, Ananthanarayanan et al. 2016, Bahdanau, Chorowski et al. 2016), doğal dil işlemede(Hermann, Kocisky et al. 2015, Luong, Pham et al. 2015, Jozefowicz, Vinyals et al. 2016, Lample, Ballesteros et al. 2016), robotik(Lenz, Lee et al. 2015, Levine, Pastor et al. 2016) ve medikal(Lanchantin, Singh et al. 2016, Chaudhary, Poirion et al. 2017, Esteva, Kuprel et al. 2017, Liu 2017, Qin and Feng 2017, Shrikumar, Greenside et al. 2017) alanlarında Derin Öğrenme modelleri kullanılmıştır.

Bu çalışmanın devamında, Bölüm 2'de ESA mimarisi ve mimarideki katmanlar anlatılmıştır. Bölüm 3'te bazı Derin Öğrenme modelleri sunulmuştur. Bölüm 4'te yapılan çalışma ile ilgili tartışma yapılmıştır. Son olarak Sonuç bölümü aktarılmıştır.

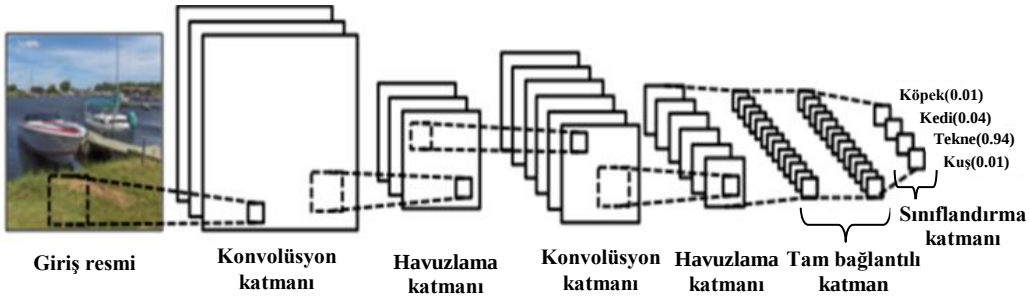
2. ESA Mimarisi

Derin Öğrenme kavramına ait temel mimari ESA mimarisi kabul edilir. Şekil 3'te ESA mimarisi verilmiştir. Bu mimariye göre ilk birkaç aşama Konvolüsyon (Convulation) ve Havuzlama (Pooling) katmanlarından oluşur. Son aşama ise Tam Bağlı katmandan oluşur ve akabinde Sınıflandırma katmanı mevcuttur. Özetle ESA'lar ard arda yerleştirilmiş birden fazla eğitilebilir bölümlerden oluşur. Devamında eğitici bir sınıflandırıcı ile devam edilir. ESA'da giriş verilerini aldıktan sonra katman katman işlemler yapılarak eğitim süreci gerçekleştirilir. En sonunda doğru sonuç ile karşılaştırma yapmak için bir final çıktısı verir. Üretilen sonuç ile istenen sonucun farkı kadar bir hata oluşur. Bu hatanın bütün ağırlıklara aktarılması için geriye yayılım algoritması kullanılır. Her bir iterasyonla ağırlıkların

güncellenmesi yapılarak hatanın azaltılması sağlanır. ESA'lar giriş verisi görüntü, ses ve video gibi herhangi bir sinyal olabilir. Temel olarak görüntü sınıflandırma üzerinde yoğunlaşan ESA'lar son yıllarda yaygın bir şekilde diğer alanlarda da kullanılmaktadır. ESA'larda görüntü sınıflandırma işlemlerinde, Şekil 2'de görüldüğü gibi pikseller, kenar kombinasyonundan oluşan motifleri, bu motifler birleşerek nesne parçalarını ve nesne parçaları birleşerek nesneleri oluşturur (LeCun, Bengio et al. 2015).



Şekil 2. ESA'ların farklı katmanlarda nesne ile ilgili oluşturduğu farklı temsiller(Lee, Grosse et al. 2009)
Figure 2*. Different representations of object parts in different layers of ESA. (Lee, Grosse et al., 2009)*



Şekil 3. Evrişimsel sinir ağının genel mimarisi(WILDML 2016)
Figure 3. *General architecture of ESA (WILDML 2016)*

2.1. ESA'yı Oluşturan Katmanlar

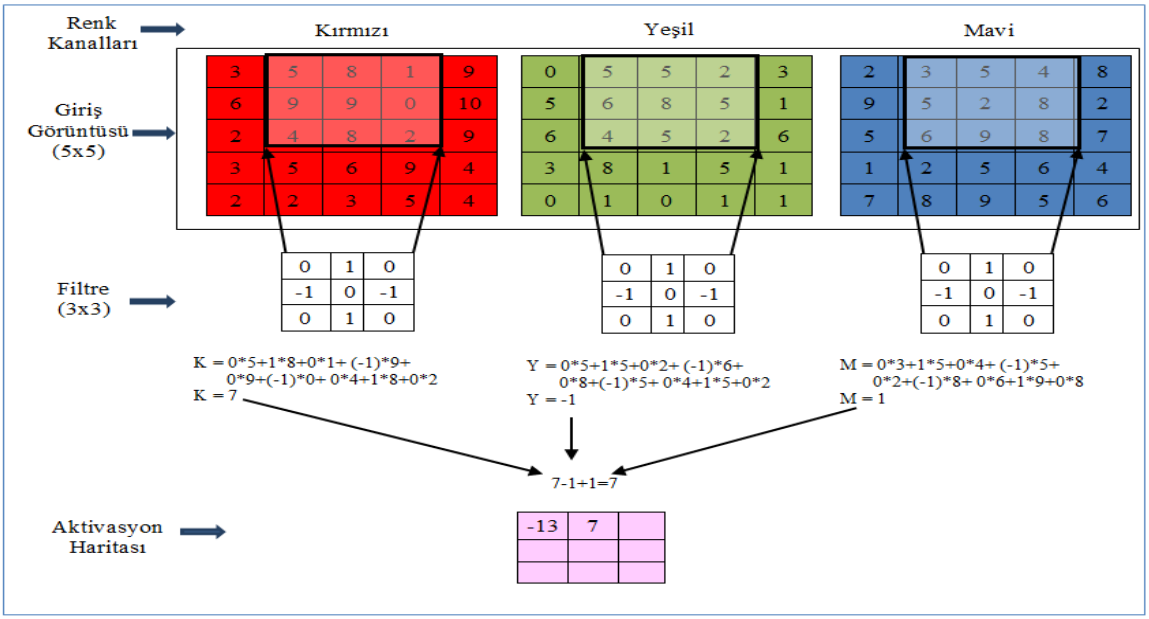
a. Giriş Katmanı (Input Layer)

Bu katman isminden de anlaşılacağı üzere ESA'ın ilk katmanını oluşturmaktadır. Bu katmanda veri ham olarak ağa verilmektedir. Tasarlanacak modelin başarımı için bu katmandaki verinin boyutu önem kazanmaktadır. Giriş görüntü boyutunun yüksek seçilmesi hem yüksek bellek ihtiyacını hem eğitim süresini hem de görüntü başına düşen test süresini uzatabilir. Bunun yanında ağ başarısını arttırabilir. Giriş görüntü boyutunun düşük seçilmesi bellek ihtiyacını azaltır ve eğitim süresini kısaltır. Fakat kurulacak ağın derinliği azalır ve performansı düşük olabilir. Görüntü analizinde hem ağ derinliği hem donanımsal hesaplama maliyeti hem de ağ başarısı için uygun bir giriş görüntü boyutu seçilmelidir. Örnek olması açısından Bölüm 3'te verilen ağ modellerinin giriş görüntü boyutu ile ilgili bilgilere bakılabilir.

b. Konvolüsyon Katmanı (Convulation Layer)

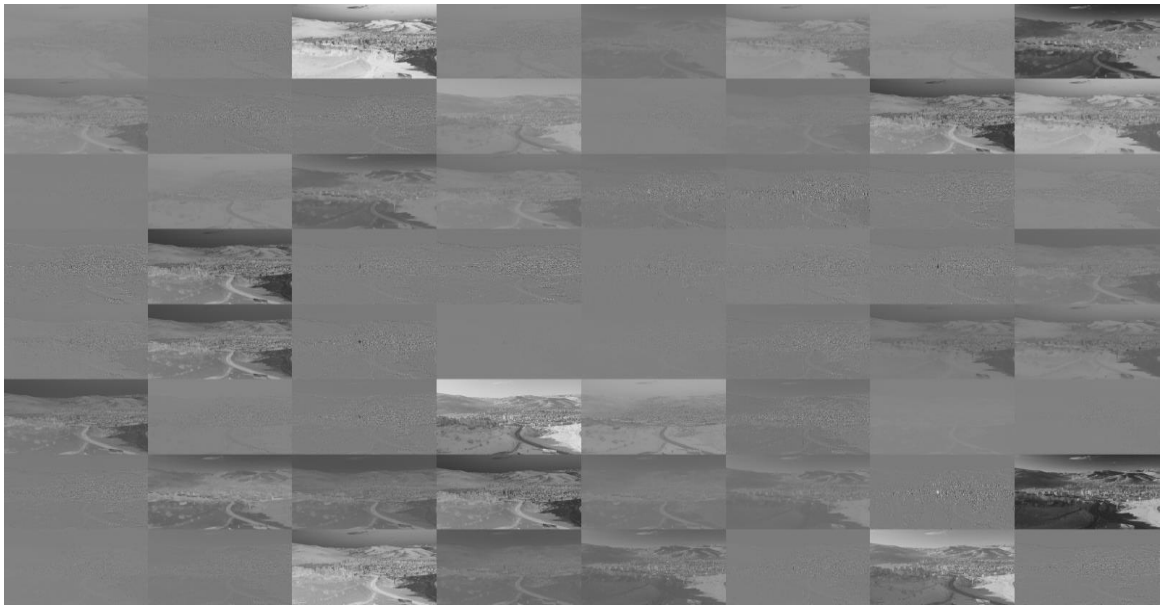
ESA'nın temelini oluşturan bu katman dönüşüm katmanı olarakta bilinir. Bu dönüşüm işlemi belirli bir filtrenin tüm görüntü üzerinde dolaştırılması işlemine dayanmaktadır. Bu

sebeple filtreler katmanlı mimarinin ayrılmaz bir bileşenidir. Filtreler 2x2, 3x3, 5x5 gibi farklı boyutlarda olabilir. Filtreler, bir önceki katmandan gelen görüntülere konvolüsyon işlemini uygulayarak çıkış verisini oluştururlar. Bu konvolüsyon işlemi sonucu aktivasyon haritası (Özellik haritası) oluşur. Aktivasyon haritası, her bir filtreye özgü özelliklerin keşfedildiği bölgelerdir. ESA'ların eğitimi esnasında bu filtrelerin katsayıları, eğitim kümesindeki her öğrenme yinelemesiyle değişir. Böylelikle ağ, özelliklerin belirlenmesi için, verinin hangi bölgelerinin önem taşıdığını belirler. Bir filtrenin görüntü üzerine uygulanması Şekil 4'te gösterilmiştir. Şekil 4'te görüldüğü gibi giriş görüntüsü renkli (RGB) ve 5x5 bir matris olduğu düşünülürse giriş veri boyutu 5x5x3 olacaktır. Konvolüsyon işlemi için 3x3'lük bir filtre ile giriş görüntüsü üzerinle sağa veya sola doğru belirli Adım(Stride) kaydırılarak dolaşma yapılır. Bu dolaşma esnasında matris sınırına gelindiğinde ise bir basamak aşağı kayıp tekrar devam edilir. Bu dolaşma işlemi görüntü matrisinin tümü üzerinde yapılır. Filtre katsayıları her bir renk kanalındaki değerlerle çarpılıp bunların toplamı alınır. Her üç kanal üzerinde bu işlem yapıldıktan sonra üçünün toplamı Aktivasyon haritasını oluşturur. Her bir renk kanalı matrisine uygulanan filtre katsayıları farklı olabilir. Bu filtre katsayılarındaki değişimler tasarımcılar tarafından modellerine uygun olacak şekilde yapılır. Aktivasyon haritasındaki değerler, giriş ve çıkış boyutu arasında aynı yoğunluk aralığını sağlamak için normalize edilir. Bu normalize için her bir renk kanalı için hesaplanan değerler filtre katsayılarının toplamına bölünür. Şekil 4'te gösterilen örnekte filtre katsayılarının toplamı sıfır olduğu için bu işlem yapılmamıştır. Konvolüsyon katmanının ESA üzerinde gösterimi Şekil 5-6'da verilmiştir. Ağın birinci konvolüsyon katmanı için 3x3'lük boyutta ve toplam 64 filtreden oluşan bir ESA tasarlanmıştır. Her bir filtrenin Şekil 5'teki giriş görüntüsüne uygulanması sonucu Şekil 6'daki görüntüler elde edilmiştir. Şekil 6'ya bakıldığında her bir filtrenin giriş görüntüsüne etkisi farklı olduğu görülmektedir.



Şekil 4. 5x5x3 boyutta bir giriş görüntüsüne 3x3'lük filtrenin uygulandığı konvolüsyon işlemi
Figure 4. *Convolution process with a 3x3 filter applied to an input image of 5x5x3 size.*

Şekil 5. ESA için giriş görüntüsü
Figure 5. *Input image for ESA*

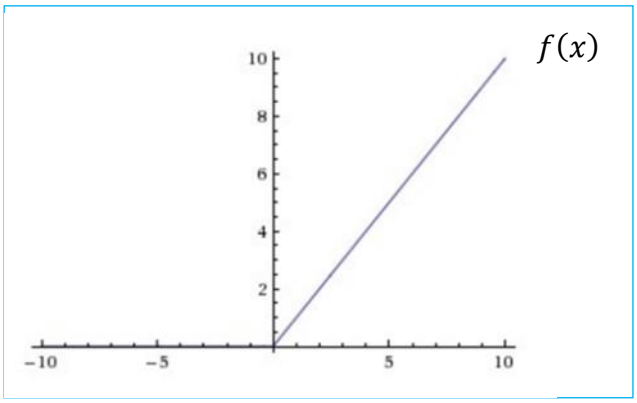


Şekil 6. ESA'nın birinci konvolüsyon katmanından sonra oluşan görüntüler.
Figure 6. *Activation maps formed after the first convolution layer of the ESA.*

c. Düzleştirilmiş Doğrusal Birim Katmanı(Rectified Linear UnitsLayer(ReLu))

Bu katman konvolüsyon katmanlarından sonra gelir ve ESA nöronlarının çıktıları için en yaygın şekilde devreye sokulan doğrultucu birim olarak bilinir. Matematiksel olarak Eşitlik 1'deki gibi tanımlanır ve Şekil 7'de gösterilmiştir. Bu katman aynı zamanda aktifleştirme katmanı olarakta bilinir. Giriş verisine yapmış olduğu etki negatif değerleri sıfıra çekmesidir. Bu katmandan önce kullanılan konvolüsyon katmanında belirli matematiksel işlemler yapıldığı için ağ doğrusal bir yapıdadır. Bu derin ağı doğrusal olmayan bir yapıya sokmak için bu katman uygulanır. Bu katmanın kullanılması ile ağ daha hızlı öğrenir.

$$f(x) = \begin{cases} 0 & eğer\ x < 0 \\ x & eğer\ x \geq 0 \end{cases} \quad (1)$$



Şekil 7. Düzleştirilmiş doğrusal birim katmanı çıkış verisine etkisi.
Figure 7. *Effect of ReLu layer to the output d*ata.

ReLu katmanının giriş görüntüsü üzerine yaptığı etki Şekil 8'de gösterilmiştir. Şekil 8'de soldaki görüntü giriş görüntüsü, ortadaki konvolüsyon katmanında 3x3'lük bir filtre uygulanması sonucu oluşan görüntü ve sağdaki görüntü ise ReLu katmanından çıkan görüntüyü göstermektedir.

Şekil 8. Bir ESA modelinde konvolüsyon katmanı ve ReLu katmanının giriş görüntüsüne yapmış olduğu etki
Figure 8. *The effect of the convolution and ReLu layer on the input image in ESA model.*

d. Havuzlama Katmanı (Pooling Layer)

Havuzlama genellikle ReLu katmanından sonra yerleştirilir. Temel amacı, sonraki konvolüsyon katmanı için giriş boyutunu (Genişlik x Yükseklik) azaltmaktır. Bu işlem veride derinlik boyutunu etkilemez. Bu katmanda gerçekleştirilen işlem, "Aşağı Örnekleme" olarak da adlandırılır. Bu katman sonucu boyuttaki azalma bilgi kaybına yol açar. Böyle bir kayıp ağ için iki nedenden dolayı faydalıdır. Birincisi, bir sonraki ağ katmanları için daha az hesaplama yükü oluşturur. İkincisi ise sistemin ezberlemesini önler. İlk basamakta gerçekleştirilen konvolüsyon işlemi gibi, havuzlama katmanında da belli filtreler tanımlanır. Bu filtreler görüntü üzerinde belli bir adım atma değerine göre gezdirilerek görüntüdeki piksellerin maksimum değerlerini (maksimum havuzlama) veya değerlerin ortalamasını (ortalama havuzlama) alarak işlem yapılır. Genellikle maksimum havuzlama, daha iyi performans gösterdiği için tercih edilir. Havuzlama işlemi, konvolüsyon katmanı sonucu oluşan fitre adedince görüntülerin hepsi için gerçekleştirilir. ESA'larda Havuzlama katmanı isteğe bağlıdır ve bazı mimariler bu işlemi gerçekleştirmez. Havuzlama işleminin yapılışı ile ilgili örnek Şekil 9'da verilmiştir. Şekil 9'da giriş görüntü boyutu 4x4 ve filtre boyutu 2x2. Bir adım kaydırılacak şekilde oluşan görüntünün boyutu 3x3 olur. İki adım kaydırılacak şekilde oluşan görüntünün boyutu ise 2x2 olur. Havuzlama

işlemi sonucunda oluşan görüntünün boyutu Eşitlik 2'e göre hesaplanır(cs.stanford.edu 2016).

$$Üretilen\ Görüntünün\ Boyutu = G_2 x Y_2 x D_2 \quad (2)$$

$$G_2=(G_1-F)/A+1 \quad (3)$$

$$Y_2=(H_1-F)/A+1 \quad (4)$$

$$D_2=D_1 \quad (5)$$

G1= Giriş görüntü boyutunun genişlik değeri

Y1= Giriş görüntü boyutunun yükseklik değeri

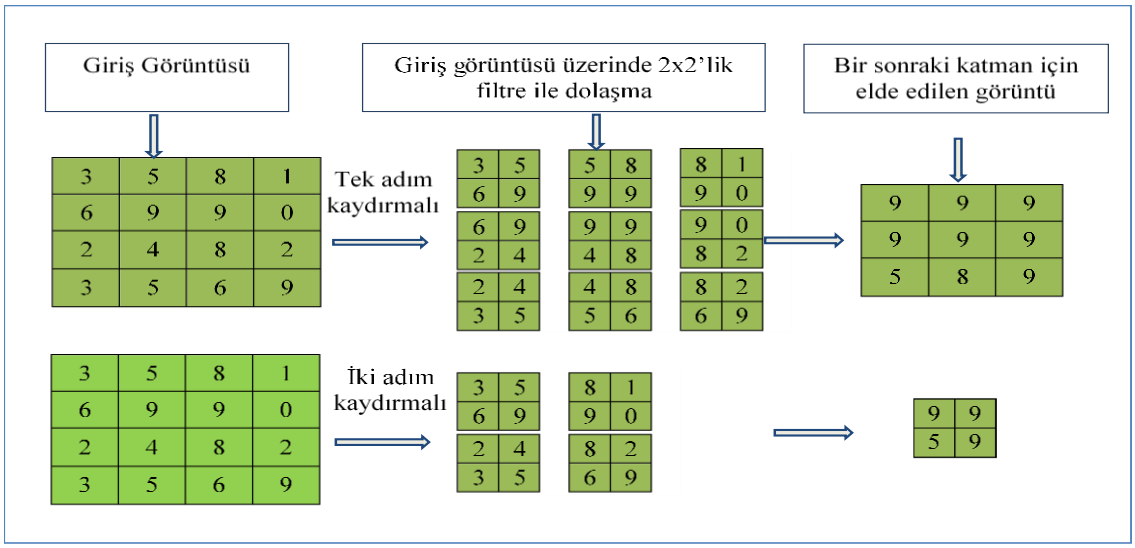
D1= Giriş görüntü boyutunun derinlik değeri

F=Filtre boyutu

A=Adım sayısı

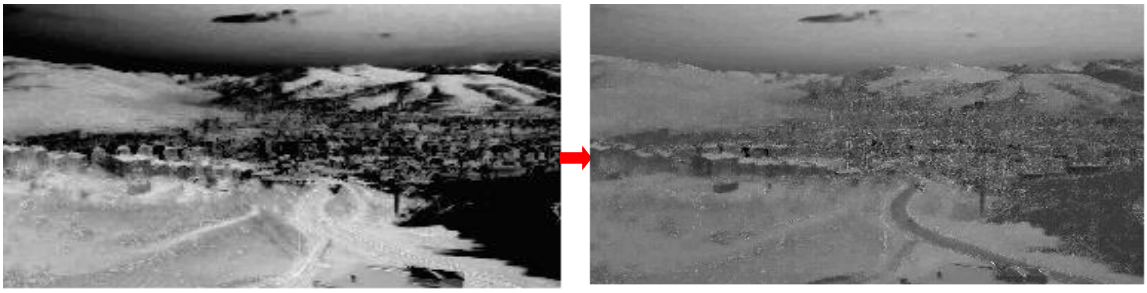
Havuzlama işleminde çoğunlukla F=2 ve A=2 olarak seçilir.

Havuzlama katmanının gerçek bir görüntüye uygulanması Şekil 10'da verilmiştir. Şekil 10'da soldaki görüntü ReLu katmanı sonucunda elde edilen görüntüyü, sağdaki görüntü ise havuzlama katmanı sonucunda oluşan görüntüyü temsil eder.



Şekil 9. 5x5'lik giriş görüntüsüne 2x2 filtre ile bir ve iki adım kaymalı maksimum havuzlama işleminin uygulanması

Figure 9. *Implementation of max-pooling with 2x2 filter in 5x5 input image.*



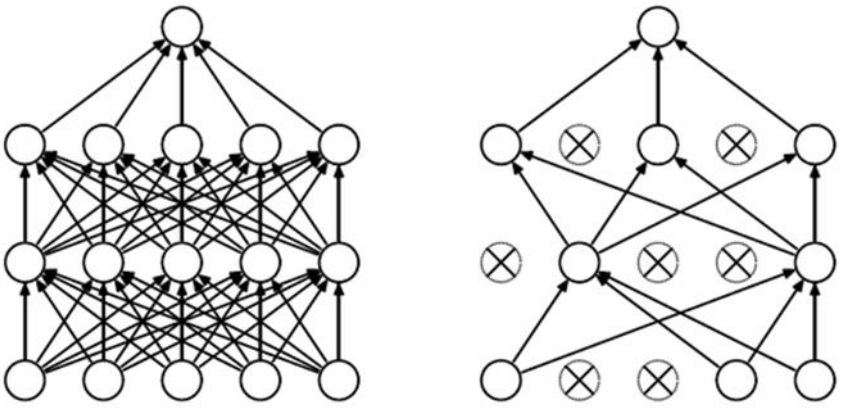
Şekil 10. ESA modelinde havuzlama katmanın bir önceki katmandan gelen görüntüye uygulanması sonucu oluşan görüntü
Figure 10. *Effect of the pooling layer on the image in the ESA model.*

e. Tam Bağlantılı Katman (Fully Connected Layer)

ESA mimarisinde ard arda gelen konvolüsyon, ReLu ve havuzlama katmanından sonra tam bağlantılı katman gelir. Bu katman kendinden önceki katmanın tüm alanlarına bağlıdır. Farklı mimarilerde bu katmanın sayısı değişebilir. ESA mimarisinde en son katmanın üretmiş olduğu matris boyutu 25x25x256=160000x1 ve tam bağlantılı katmandaki matris boyutu 4096x1 olarak seçilirse. Toplamda 160000x4096 ağırlık matrisi oluşur. Yani her bir 160000 nöron 4096 nöron ile bağlanmaktadır. Bu sebepten dolayı bu katmana tam bağlantılı katman denilmektedir.

f. DropOut Katmanı

ESA'da büyük veriler ile eğitim işlemi yapıldığı için bazen ağ ezberleme yapar. Ağın ezberlemesinin önüne geçmek için bu katman kullanılır(Srivastava, Hinton et al. 2014). Bu katmanda uygulanan temel mantık ağın bazı düğümlerinin kaldırılmasıdır. Şekil 11 (a)'da ağın orijinal yapısı gösterilirken, (b)'de DropOut katmanından sonraki hali gösterilmiştir.



Şekil 11. Standart bir ESA ağına DropOut katmanın uygulanması
Figure 11. *Implementation of a DropOut layer on a standard ESA network*

g. Sınıflandırma Katmanı(Classification Layer)

Bu katman tam bağlantılı katmandan sonra gelir. Derin öğrenme mimarilerinin bu katmanında sınıflandırma işlemi yapılmaktadır. Bu katmanın çıkış değeri, sınıflandırması yapılacak nesne sayısına eşittir. Örneğin 15 farklı nesnenin sınıflandırılması yapılacaksa, sınıflandırma katmanı çıkış değeri 15 olmalıdır. Tam bağlantılı katmanda çıkış değeri 4096 olarak seçilirse, bu çıkış değerine göre sınıflandırma katmanı için 4096x15 ağırlık matrisi elde edilir. Bu katmanda farklı sınıflandırıcılar kullanılmaktadır. Çoğunlukla başarısından dolayı softmax sınıflandırıcı tercih edilir. Sınıflandırmada 15 farklı nesne 0-1 aralığında belli bir değerde çıkış üretir. 1'e yakın sonucu üreten çıkış, ağın tahmin ettiği nesne olduğu anlaşılır.

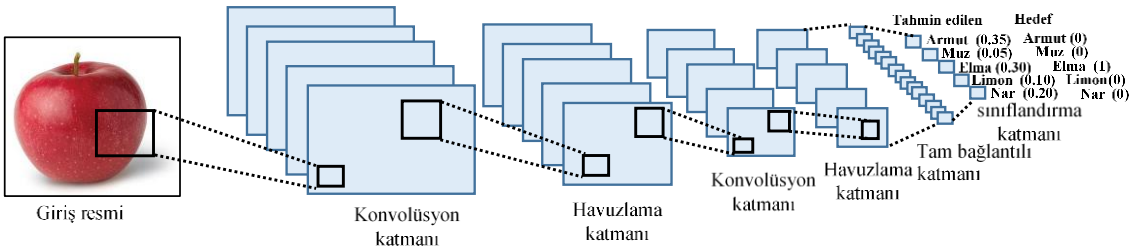
2.2.Evrişimsel Sinir Ağının Eğitilmesi

ESA'nın eğitilmesi adımlar halinde aşağıda belirtilmiştir.

Adım 1. Öncelikle bir ESA modeli oluşturulur. Bu modelde konvolüsyon katman sayısı, havuzlama katman sayısı, tam bağlantılı katman sayısı ve sınıflandırma katmanı belirlenir. Bu katmanların sıralanması ve adetleri tasarımcıya özgüdür.

Adım 2. Model oluşturulduktan sonra başlangıç değişkenleri tanımlanır. Bu değişkenler temel olarak filtre boyutları, filtre sayısı ve adım kayma miktarı olarak sıralanabilir. Ayrıca her bir filtre için 0-1 aralığında gelişigüzel değerler atanır.

Adım 3. Oluşturulan model giriş verisi olarak eğitim setinden bir görüntü verilir. Bu görüntü ağdaki katmanlardan geçirilerek bir sonuç değeri elde edilir. Bu aşamaya ileri besleme denir. İleri beslemede her katmanda her bir filtrenin ağırlıkları ile görüntüdeki piksel değerleri çarpılıp bunların toplamı alınarak bir sonraki katmana aktarılır. Şekil 12'de örnek olarak verilen ESA modelinde 5 farklı nesnenin sınıflandırılması yapılmaktadır. Ağa Elma görüntüsü verildiğinde, [Armut, Muz, Elma, Limon, Nar] için tahmin ettiği değerler [0.35, 0.05, 0.30, 0.10, 0.20] olduğu görülmektedir.



Şekil 12. Örnek bir ESA modelinin ileri besleme anında nesne için tahmin ettiği sonuçlar
Figure 12. *Estimated results of the ESA model for the object during forward feed.*

Adım 4. Eşitlik 6'ya göre, ağın üretmiş olduğu sonuçlar ile hedef sonuçların farkı alınarak toplam hata değeri hesaplanır.

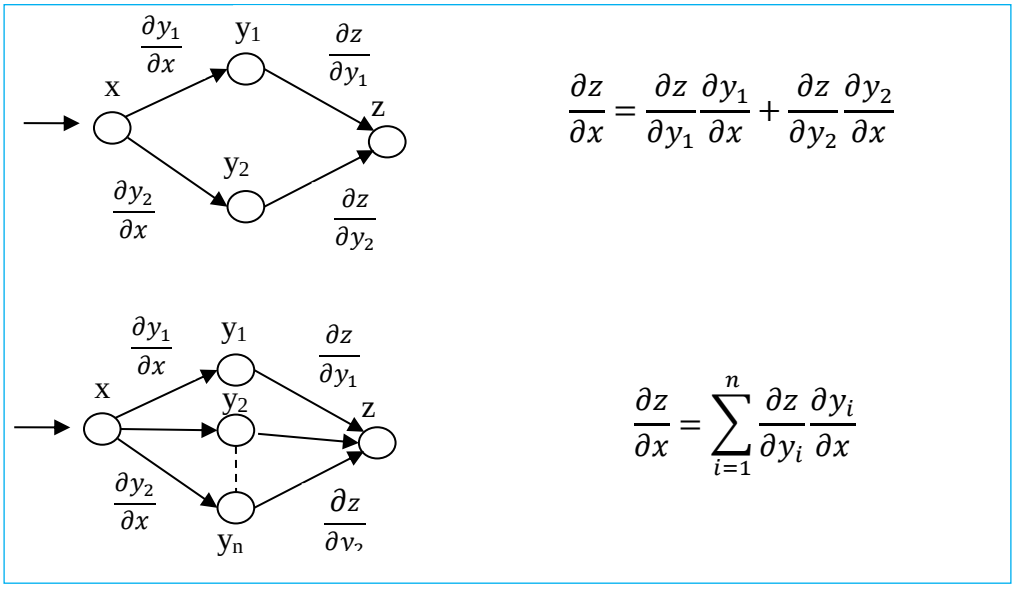
$$Toplam\ Hata = \sum_{i=1}^{n} \frac{1}{2}(Hedef_i - Tahmin\ Edilen_i)^2 \qquad (6)$$

Adım 5. Elde edilen hata değerinin ağdaki bütün ağırlıklara dağıtılması gerekmektedir. Bu işlem için Geriye Yayılım Algoritması(GYA) kullanılır. GYA'da her bir ağırlığın toplam hataya olan etkisinin hesaplanması için Stokastik Gradyan İniş(SGİ) optimizasyon algoritması kullanılır. Buradaki ağırlıkların güncellenmesiyle ağın çıkışındaki toplam hata

değeri düşürülmeye çalışılır. Dolayısıyla sınıflandırma başarısının arttırılması amaçlanmaktadır. Bu adımdan sonra ağa tekrar Elmaya ait başka bir görüntü verildiğinde bu sefer [Armut, Muz, Elma, Limon, Nar] için tahmin edilen değerler [0.1, 0.01, 0.8, 0.07, 0.02] olarak elde edilir. Dikkat edilirse ağırlıklar ilk görüntüden sonra güncellendiğinden Elma için yapılan tahmin 1'e yaklaşmıştır.

ESA'larda ağırlıkların güncellenmesi için farklı gradyan tabanlı optimizasyon algoritmaları kullanılmaktadır. Bu algoritmaların birbirlerine göre avantaj ve dezavantajları(Ruder 2016) çalışmada detaylı bir şekilde anlatılmıştır. GYA'da her bir ağırlığın değeri Gradyan iniş yöntemine göre hesaplanırken kısmi türev kullanılır ve Zincir kuralına göre yapılır. Şekil 13'te verilen örnek bir ağ üzerinde ağırlıkların güncellenmesi için öncelikle z'nin y_1'e göre ve y_1'in x'e göre türevi ile z'nin y_2'ye göre ve y_2'in x'e göre türevi alınarak ikisinin toplamı alınır. Bu iki değerin toplamı, z'nin x'e göre kısmi türevini verir.

Adım 6. Eğitim kümesindeki bütün görüntüler için Adım 3-5 tekrar edilir.

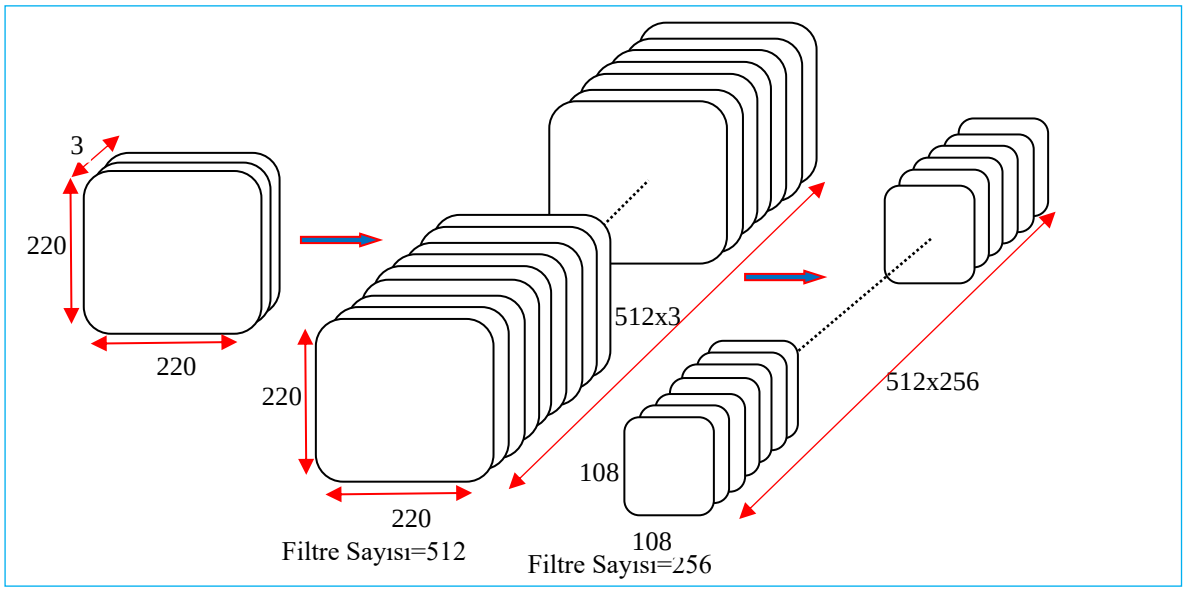


Şekil 13. GYA'da ağırlıkların güncellenmesi için kısmi türevin kullanılması
Figure 13*. Using a partial derivative to update weights in the back propagation algorithm.*

2.3.ESA'da Derinlik ve Genişlik Kavramı

ESA'daki derinlik, toplam katman sayısı ile ilişki bir kavramdır. Genişlik ise konvolüsyon katmanlarındaki filtre sayıları ile ilişkilidir. Öncelikle filtre sayılarının genişlik kavramına olan etkisi Şekil 14'te gösterilmiştir. Her bir filtre ile görüntüdeki bir özellik öğrenilir. Bu yüzden ne kadar fazla filtre varsa o kadar fazla özellik keşfedilir ve dolayısıyla ağın genişliği artar. Şekil 14'te örnek olarak bir ESA ağının ilk 3 katmanı verilmiştir. Giriş katmanı renkli görüntü boyutu 220x220x3, İkinci katmanda filtre sayısı 512 ve bir sonraki katmandaki filtre sayısı 256 olarak belirtilmiştir. Filtrelerin uygulanmasıyla, ikinci katmanda 512*3 görüntü üçüncü katmanda ise 512*256 adet görüntü oluşur. Bu işlem konvolüsyon katmanlarında kullanılan filtre sayısıyla orantılı olarak artarar ağın genişliğini arttırır. Şekil 14'te 3 katman olduğu için ağın derinliği 3'tür. Eğer tasarlanacak ağda havuzlama katmanı kullanılacaksa, ağ derinliği giriş katmanındaki görüntü boyutu ile ilişkili hale gelir. Çünkü daha önce belirtildiği gibi havuzlama katmanı boyut azaltma etkisi yapar. Dolayısıyla giriş görüntüsünün boyutu yüksek olması ile ağ katmanının sayısı

arttırılabilir. Eğer giriş görüntü boyutu düşük olursa, ağdaki havuzlama katmanının yapmış olduğu boyut azaltma işleminden dolayı ancak sığ bir ağ tasarlanabilir.



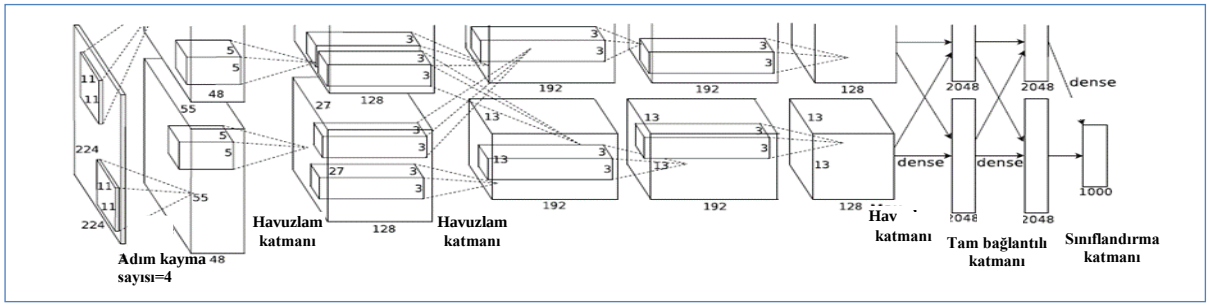
Şekil 14. ESA'da filtre sayısının oluşturmuş olduğu genişlik ve katman sayısına göre derinlik.
Figure 14. *The width and depth of the CNN.*

3. Derin Öğrenme Modelleri

Giriş bölümünde derin öğrenmenin 2012 yılında yapılan İmageNet yarışmasıyla popüler olduğu belirtildi. Bu yarışmanın devam eden yıllarında katılımcıların çoğu modellerini derin öğrenme mimarisine göre tasarladı. Bu sebeple İmageNet yarışmasını 2012-2015 yıllarda kazanan modellerin açıklaması bu bölümde yapıldı. Bu modellerin her biri derin öğrenmede temel taşları olarak kabul edilmektedir. ImageNet yarışmasını kazanan modellere ilaveten, görüntü içerisinde nesne tanımlama için geliştirilen modeller hakkında bilgiler bu bölümde verilmiştir.

a. Alex Net

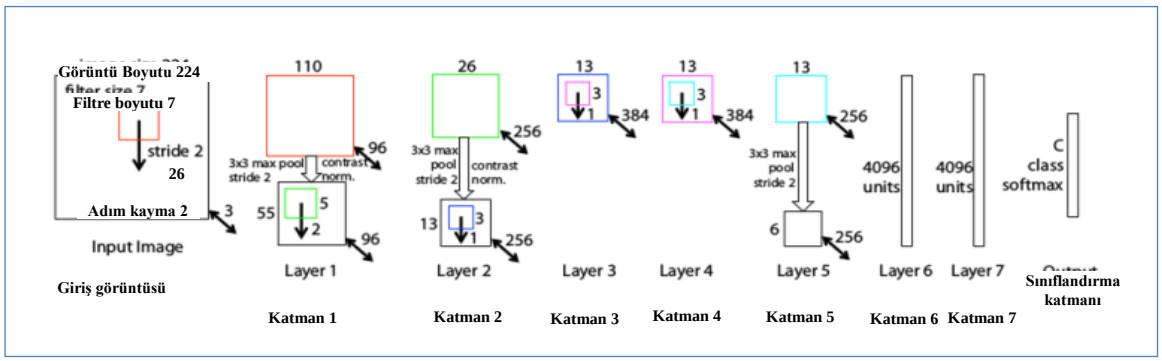
Her ne kadar derin öğrenmenin ilk olarak 1998 yılında Yann LeCun'nun yayınlamış olduğu makale(Lecun, Bottou et al. 1998) ile ilk uygulama yapıldığı söylense de, dünya çapında duyulması 2012 yılında olmuştur. O yıl gerçekleştirilen İmageNet yarışmasını, derin öğrenme mimarisi ile tasarlanan AlexNet modeli kazanmıştır. Yapılan çalışma "ImageNet Classification with Deep Convolutional Networks"(Krizhevsky, Sutskever et al. 2012) isimli makale ile yayınlanmış ve Ekim 2017 tarihi itibari ile 16227 alıntı yapılmıştır. Bu mimari ile bilgisayarlı nesne tanımlama hata oranı %26,2'den %15,4'de düşürülmüştür. Şekil 15'te verilen mimari 5 konvolüsyon katmanı, havuzlama katmanı ve 3 tam bağlantılı katmandan oluşmaktadır. Mimari 1000 nesneyi sınıflandıracak şekilde tasarlanmıştır. Filtreler 11x11 boyutunda ve adım kayma sayısı 4 olarak belirlenmiştir.



Şekil 15. AlexNet mimarisi(Krizhevsky, Sutskever et al. 2012)
Figure 15. *AlexNet architecture (Krizhevsky, Sutskever et al., 2012)*

b. ZF Net

2012 yılında AlexNet'in ImageNet yarışmasını kazanmasının akabinde yapılan yarışmalarda derin öğrenme modelleri kullanılmaya başlandı. Matthew Zeiler ve Rob Fergus tasarlamış olduğu ZFNet(Zeiler and Fergus 2014) 2013 yılında ImageNet yarışmasının kazananı olmuştur. Bu model ilenesne tanımada hata oranı %11,2'ye indirilmiştir. Bu mimari AlexNet mimarisinin geliştirilmiş halidir ve Şekil 16'da verilmiştir.



Şekil 16. ZF Net mimarisi(Zeiler and Fergus 2014)
Figure 16. *ZF Net architecture (Zeiler and Fergus 2014)*

ZF Net modelinde, ilk katmanda AlexNet'in uyguladığı 11x11 boyutlu filtreler kullanmak yerine, 7x7 boyutundaki filtreleri ve havuzlama katmanında 2 adım kayma miktarı kullanılmıştır. Bu değişikliğin arkasındaki mantık, birinci konvolüsyon katmanındaki daha küçük bir filtre boyutunun, giriş boyutundaki birçok orijinal piksel bilgisinin korunmasına yardımcı olmasıdır. Aktivasyon fonksiyonu için ReLu, hata fonksiyonu için Cross-Entropy Loss ve eğitim için Stochastic Gradient Descent kullanılmıştır. Ekran kartı olarak bir GTX 580 GPU üzerinde on iki gün boyunca eğitimi sürmüştür.

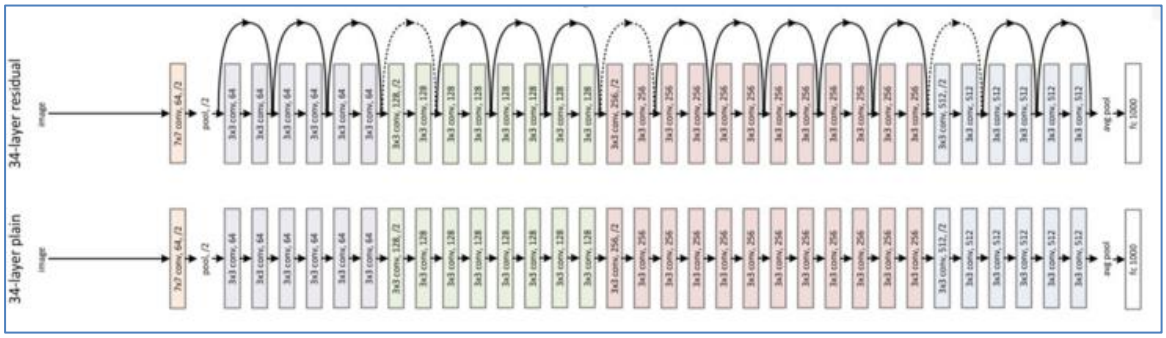
c. GoogLeNet

GoogLeNet(Szegedy, Liu et al. 2015) yapısındaki Inception modüllerinden dolayı karmaşık bir mimaridir. GoogLeNet 22 katmanlı ve %5,7 hata oranı ile ImageNet 2014 yarışmasının kazananı olmuştur. Bu mimari genel olarak, ardışık bir yapıda konvolüsyon ve havuzlama katmanlarını üst üste istiflemekten uzaklaşan ilk CNN mimarilerinden birisidir. Ayrıca bu yeni model bellek ve güç kullanımı üzerinde önemli bir yere sahiptir. Çünkü katmanların hepsini yığınlamak ve çok sayıda filtre eklemek, bir hesaplama ve bellek maliyeti getirir ve ezberleme olasılığını artırır. GoogLeNet bu durumun üstesinden gelmek için paralel olarak birbirine bağlı modüller kullanılmıştır. Şekil 17'de GoogLeNet ağ mimarisi verilmiştir.

Şekil 17. GoogLeNet ağ mimarisi(Szegedy, Liu et al. 2015)
Figure 17. *GoogLeNet network architecture (Szegedy, Liu et al., 2015)*

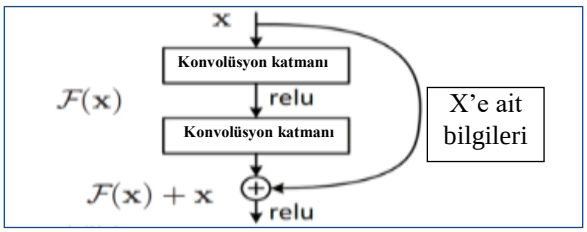
d. Microsoft RestNet

ResNet(He, Zhang et al. 2016) şuana kadarki tüm mimarilerden daha derin olarak tasarlanan bir mimaridir. 152 katmandan oluşmaktadır. Aynı zamanda %3,6 (Becerileri ve uzmanlıklarına bağlı olarak, insanlar genelde% 5-10 hata oranına sahipler) hata oranı ile ImageNet 2015 yarışmasının kazananı olmuştur. Microsoft RestNet ilk 34 katmanlı ağ mimarisi Şekil 18'de gösterilmiştir.



Şekil 18. Microsoft RestNet ilk 34 katmanının ağ mimarisi(He, Zhang et al. 2016)
Figure 18. *The network architecture of the first 34 layers of Microsoft RestNet (He, Zhang et al., 2016)*

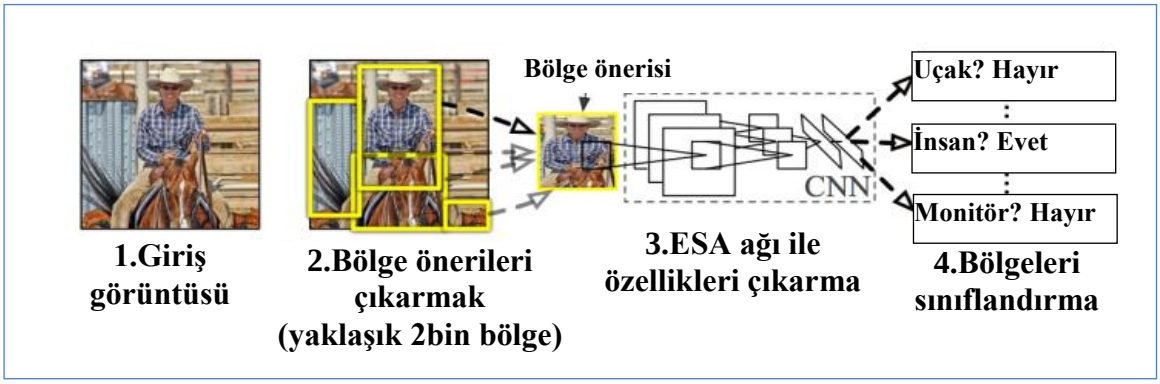
Bu mimari Residual bloklardan oluşmaktadır. Residual blokta, x girişinin konvolüsyon-ReLu-konvolüsyon serisinden sonra bir F (x) sonucu vermektedir. Bu sonuç daha sonra orijinal x girişine eklenir ve H (x) = F (x) + x olarak ifade edilir (Şekil 19).



Şekil 19. Residual blok(He, Zhang et al. 2016)
Figure 19. *Residual block (He, Zhang et al., 2016)*

e. R-CNN, Fast R-CNN ve Faster R-CNN Modelleri

Görüntü sınıflandırma işlemi, genellikle bir görüntüdeki nesnenin tahmini üzerine çalışmaktadır. Nesnenin görüntünün neresinde olduğu ve kapladığı sınırların tespiti ise nesne tanımlama işlemidir. Nesne tanımlama için derin öğrenmede R-CNN(Girshick, Donahue et al. 2014) modeli tasarlanmış olup mimarisi Şekil 20'de verilmiştir.



Şekil 20. R-CNN mimarisi(Girshick, Donahue et al. 2014)
Figure 20. *R-CNN architecture (Girshick, Donahue et al., 2014*)

Bu mimari Şekil 20'de gösterildiği gibi başlıca 4 bölümden oluşmaktadır. Birinci bölümde görüntüler alınır. İkinci bölümde ise Seçici Arama(SA) ile bölge önerileri yapılır. SA, bir nesneyi ihtimali yüksek olan 2000 farklı bölge üretme işlevi gerçekleştirir. Üçüncü bölümde her bir bölge önerisi AlexNet'te benzer şekilde tasarlanmış ESA mimarisine verilir. Son olarak ESA çıktısı eğer bir nesne ise SA ile belirlenen bölge üzerinde bir düzenleme yapılarak nihai sonuç üretilir. R-CNN bazı dezavantajları vardır. Öncelikle her bir görüntüdeki her bölge önerisi için ESA'den ileri geçiş gerektirmekte (Görüntü başına 2000 ileri geçiş). Bu test aşamasında zaman kaybına sebep olmaktadır. İkinci olarak mimari ayrı ayrı üç farklı modeli eğitmek zorundadır. Bunlar, görüntü özelliklerini oluşturmak için ESA, nesnenin sınıfını tahmin eden sınıflandırıcı ve sınırlayıcı çerçeveleri sıkıştırmak için regresyon modeli. Bu işlem boru hattını eğitmek son derece zordur ve ortalama görüntü başına 50 saniye gibi bir zaman almaktadır.

Bu sorunların üstesinden gelmek için Fast R-CNN(Girshick 2015) modeli geliştirilmiştir. Bu mimaride öncelikle her bir görüntü için yaklaşık 2000 kez ESA çalıştırmak yerine tek bir kez ESA çalıştırıp 2000 öneri arasında bu hesaplamanın paylaşımı yapılmaktadır. Böylelikle eğitim ve test süresi kısaltılmıştır.

Hem R-CNN hem de Fast R-CNN'nin sergilediği karmaşık eğitim hattını iyileştirmek için Faster R-CNN(Ren, He et al. 2017) tasarlanmıştır. Bu işlemi son konvolüsyon katmanından sonra bir bölge öneri ağı (Region Proposal Network (RPN)) ekleyerek gerçekleştirmiştir. Bu 3 model PASCAL VOC(Everingham, Van Gool et al. 2007) veri seti üzerinde denenmiş ve sonuçları Tablo 1'de verilmiştir.

Çizelge 1. Nesne tanımlama için kullanılan modellerin karşılaştırılması
Table 1. *Comparison of models used for object detection*

	R-CNN	Fast R-CNN	Faster R-CNN
Görüntü başına test süreleri	50sn	2sn	0.2sn
Hızlanma	1x	25x	250x
Ortalama tahmin(%)	66	66.9	66.9

4. Tartışma

Derin Öğrenme bazı araştırmacılar tarafından bir yöntem veya algoritma olarak algılanmaktadır. Bu algının aksine derin öğrenme aslında bir kavramdır. Bu kavramın temelinde ise YSA'ların daha da derinleştirilip ESA'ların elde edilme süreci yatar. Giriş

bölümünde ESA'ların YSA'lardan farkı ve benzerlikleri anlatılmıştır. Bilgisayar bilimlerinde kavramların anlaşılması açısından şöyle bir açıklama yapılabilir; makine öğrenmesi yapay zekânın bir alt dalıdır. YSA'lar ise makine öğrenmesinin içerisinde bir sınıf olarak kabul edilebilir. ESA'lar ise YSA'ların zamanla geliştirilmiş halidir ve aynı zamanda Derin Öğrenme kavramı altında kullanılmaktadır. Derin öğrenme ayrıca temelinde bulunan YSA ve ESA yapılarıyla beraber yapay zekânın bir alanı olarak kabul edilebilir ve makine öğrenmesi alanından ayrılabilir.

ESA'ların temeline bakıldığında konvolüsyon katmanındaki filtrelerin varlığı bu modellerin başarısının temelini oluşturmaktadır. Buna paralel olarak daha derin ağarın eğitilmesi için geliştirilen Gradyan tabanlı optimizasyon algoritmaları ve eğitim esnasında ezberlemenin önlenmesi için sunulan DropOut yöntemi önemli bir yere sahiptir. Derin ağların belkide en üstün tarafı, bir problemin çözümünde sonsuz sayıda modellerin tasarlanmasına olanak sunmasıdır. Bu yüzden kısır bir yönü yoktur ve gelişime her zaman açıktır. Bu yapıyı bünyesinde bir den fazla parametrenin ayarlanması zorunluluğundan dolayı elde etmektedir. Bu parametrelerin bazıları, katman sayısı, konvolüsyon katmanı filtre boyutu ve sayısı, havuzlama katmanı filtre boyutu ve adım kayma sayısı, ağın öğrenme oranı örnek olarak verilebilir. Her ne kadar bu parametrelerin optimizasyonu ile ilgili çalışma(Bergstra and Bengio 2012) yapılmış olsa da hala bu noktada eksikler mevcuttur.

Derin öğrenme ağlarının uygulandığı alanlara bakıldığında birçok yönden başarılar elde edildiği görülmektedir. Bunun yanında yaygın bir şekilde kullanılmasının altında yatan en büyük bir diğer sebep ise, açık kaynak kodlu yazılım kütüphanelerin oluşturulması ve veri setlerin çokluğu olarak açıklanabilir.

Derin ağların belki de en büyük dezavantajları modellerinin eğitilmesinde ve test aşamasında donanımsal kaynakların getirmiş olduğu sınırlamalardır. Bu problemin ortadan kaldırılması için 2 farklı çözüm sunulabilir. Birincisi, derin ağların bellek kullanımı ve gerçek zamanlı çalıştıklarında test süreleri göz önünde bulundurularak tasarlanmasıdır. Bununla ilgili örnek bir çalışma yapılmıştır(Iandola, Han et al. 2016). İkincisi, derin ağların daha düşük hesaplama maliyetiyle çalıştırılması üzerine donanımsal ve yazılımsal çalışmaların yapılmasıdır. Gelecekte bu iki dezavantajın giderilmesi üzerine çalışmaların yoğunlaşacağı tahmin edilmektedir.

5. Sonuç

Yapılan çalışmada öncelikle Derin Öğrenmenin tarihsel olarak geçirmiş olduğu aşamalar ve günümüzde ortaya çıkmasının sebepleri açıklanmıştır. Derin öğrenmede temel mimari kabul edilen ESA mimarisinin yapısı ve kullanılan katmanlar hakkında bilgiler verilmiştir. Kullanılan her bir katmanın arka planında gerçekleşen işlemler anlatılarak konvolüsyon, havuzlama ve ReLu katmanıyla ilgili yapılan uygulamaların çıktı görüntüleri sunulmuştur. Her bir katmanın modele olan etkisi anlatıldıktan sonra, ESA modelinde yapılan eğitim işlemi adımlar halinde belirtilmiştir. Derin öğrenme kavramındaki genişlik ve derinlik ifadeleri şekilsel olarak örnek bir ESA mimarisi üzerinde açıklanmıştır. Derin öğrenmenin ilk popüler olmasını sağlayan AlexNet modeli ve akabinde temel kabul edilen bazı ESA modellerinin yapıları anlatılmıştır. Derin Öğrenme ile ilgili yapılan çalışmalar hakkında bilgiler sunulmuştur. Sonuç olarak, derin öğrenme alanında çalışma yapmak isteyenler için temel bilgiler ve kavramlar açıklanmış ve bu kişilerin istifadesine sunulmuştur.

Kaynaklar

Amodei, D., S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng and G. Chen (2016). Deep speech 2: End-to-end speech recognition in english and mandarin. International Conference on Machine Learning.

Assael, Y. M., B. Shillingford, S. Whiteson and N. de Freitas (2016). "LipNet: End-to-End Sentence-level Lipreading."

Bahdanau, D., J. Chorowski, D. Serdyuk, P. Brakel and Y. Bengio (2016). End-to-end attention-based large vocabulary speech recognition. Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on, IEEE.

Bengio, Y., A. Courville and P. Vincent (2013). "Representation Learning: A Review and New Perspectives." Ieee Transactions on Pattern Analysis and Machine Intelligence **35**(8): 1798-1828.

Bengio, Y., P. Lamblin, D. Popovici and H. Larochelle (2007). "Greedy layer-wise training of deep networks." In NIPS'2006 . 14, 19, 200, 323, 324, 530, 532.

Bengio, Y. and Y. LeCun (2007). "Scaling learning algorithms towards AI." In Large Scale Kernel Machines . 19.

Bergstra, J. and Y. Bengio (2012). "Random search for hyper-parameter optimization." Journal of Machine Learning Research **13**(Feb): 281-305.

Cao, Z., T. Simon, S.-E. Wei and Y. Sheikh (2016). "Realtime multi-person 2d pose estimation using part affinity fields." arXiv preprint arXiv:1611.08050.

Chaudhary, K., O. B. Poirion, L. Lu and L. Garmire (2017). "Deep Learning based multi-omics integration robustly predicts survival in liver cancer." bioRxiv: 114892.

Cheng, Z., Q. Yang and B. Sheng (2015). Deep colorization. Proceedings of the IEEE International Conference on Computer Vision.

Competition, I. L. S. V. R. (2012). "Available online: http://www. image-net. org/challenges." LSVRC/(accessed on 27 December 2016).

cs.stanford.edu. (2016). "CS231n Convolutional Neural Networks for Visual Recognition." from http://cs231n.github.io/convolutional-networks/.

Delalleau, O. and Y. Bengio (2011). "Shallow vs. deep sum-product networks." In NIPS. 19, 556.

Donahue, J., L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko and T. Darrell (2015). Long-term recurrent convolutional networks for visual recognition and description. Proceedings of the IEEE conference on computer vision and pattern recognition.

Esteva, A., B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau and S. Thrun (2017). "Dermatologist-level classification of skin cancer with deep neural networks." Nature **542**(7639): 115-118.

Everingham, M., L. Van Gool, C. Williams, J. Winn and A. Zisserman (2007). The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results http://www. pascal-network. org/challenges. VOC/voc2007/workshop/index. html.

Ganin, Y., D. Kononenko, D. Sungatullina and V. Lempitsky (2016). DeepWarp: Photorealistic image resynthesis for gaze manipulation. European Conference on Computer Vision, Springer.

Girshick, R. (2015). Fast r-cnn. Proceedings of the IEEE international conference on computer vision.

Girshick, R., J. Donahue, T. Darrell and J. Malik (2014). "Rich feature hierarchies for accurate object detection and semantic segmentation." 2014 Ieee Conference on Computer Vision and Pattern Recognition (Cvpr): 580-587.

Graves, A. (2013). "Generating sequences with recurrent neural networks." arXiv preprint arXiv:1308.0850.

Graves, A., A.-r. Mohamed and G. Hinton (2013). Speech recognition with deep recurrent neural networks. Acoustics, speech and signal processing (icassp), 2013 ieee international conference on, IEEE.

He, K., X. Zhang, S. Ren and J. Sun (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. Proceedings of the IEEE international conference on computer vision.

He, K. M., X. Y. Zhang, S. Q. Ren and J. Sun (2016). "Deep Residual Learning for Image Recognition." 2016 Ieee Conference on Computer Vision and Pattern Recognition (Cpvr): 770-778.

Hermann, K. M., T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman and P. Blunsom (2015). Teaching machines to read and comprehend. Advances in Neural Information Processing Systems.

Hinton, G., L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen and T. N. Sainath (2012). "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups." IEEE Signal Processing Magazine **29**(6): 82-97.

Hinton, G. E., S. Osindero and Y.-W. Teh (2006). "A fast learning algorithm for deep belief nets." Neural computation **18**(7): 1527–1554.

Hwang, J. and Y. Zhou "Image Colorization with Deep Convolutional Neural Networks."

Iandola, F. N., S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally and K. Keutzer (2016). "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size." arXiv preprint arXiv:1602.07360.

Isola, P., J.-Y. Zhu, T. Zhou and A. A. Efros (2016). "Image-to-image translation with conditional adversarial networks." arXiv preprint arXiv:1611.07004.

Jozefowicz, R., O. Vinyals, M. Schuster, N. Shazeer and Y. Wu (2016). "Exploring the limits of language modeling." arXiv preprint arXiv:1602.02410.

Julia, D. L. f. (2016). "devblogs.nvidia.com." from https://devblogs.nvidia.com/parallelforall/mocha-jl-deep-learning-julia/.

Kiros, R., R. Salakhutdinov and R. S. Zemel (2014). "Unifying visual-semantic embeddings with multimodal neural language models." arXiv preprint arXiv:1411.2539.

Krizhevsky, A., I. Sutskever and G. Hinton (2012). "ImageNet classification with deep convolutional neural networks." In NIPS'2012 . 23, 24, 27, 100, 200, 371, 456, 460.

Lample, G., M. Ballesteros, S. Subramanian, K. Kawakami and C. Dyer (2016). "Neural architectures for named entity recognition." arXiv preprint arXiv:1603.01360.

Lanchantin, J., R. Singh, B. Wang and Y. Qi (2016). "Deep Motif Dashboard: Visualizing and Understanding Genomic Sequences Using Deep Neural Networks." arXiv preprint arXiv:1608.03644.

Larsson, G., M. Maire and G. Shakhnarovich (2016). Learning representations for automatic colorization. European Conference on Computer Vision, Springer.

LeCun, Y. (1987). Modèles connexionistes de l'apprentissage, Université de Paris VI. 18, 504, 517.

LeCun, Y., Y. Bengio and G. Hinton (2015). "Deep learning." Nature **521**(7553): 436-444.

Lecun, Y., L. Bottou, Y. Bengio and P. Haffner (1998). "Gradient-based learning applied to document recognition." Proceedings of the IEEE **86**(11): 2278–2324.

Lee, H., R. Grosse, R. Ranganath and A. Y. Ng (2009). Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. Proceedings of the 26th annual international conference on machine learning, ACM.

Lenz, I., H. Lee and A. Saxena (2015). "Deep learning for detecting robotic grasps." The International Journal of Robotics Research **34**(4-5): 705-724.

Levine, S., P. Pastor, A. Krizhevsky, J. Ibarz and D. Quillen (2016). "Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection." The International Journal of Robotics Research: 0278364917710318.

Lillicrap, T. P., J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver and D. Wierstra (2015). "Continuous control with deep reinforcement learning." arXiv preprint arXiv:1509.02971.

Liu, X. (2017). "Deep Recurrent Neural Network for Protein Function Prediction from Sequence." arXiv preprint arXiv:1701.08318.

Long, J., E. Shelhamer and T. Darrell (2015). Fully convolutional networks for semantic segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Luong, M.-T., H. Pham and C. D. Manning (2015). "Effective approaches to attention-based neural machine translation." arXiv preprint arXiv:1508.04025.

Mao, J., W. Xu, Y. Yang, J. Wang and A. L. Yuille (2014). "Explain images with multimodal recurrent neural networks." arXiv preprint arXiv:1410.1090.

McClelland, J., D. Rumelhart and G. Hinton (1995). The appeal of parallel distributed processing . In Computation & intelligence, American Association for Artificial Intelligence. 17.

McCulloch, W. S. and W. Pitts (1943). "A logical calculus of the ideas immanent in nervous activity." The Bulletin of Mathematical Biophysics **5**(4): 115–133.

Minsky, M. L. a. P., S. A. (1969). "Perceptrons." MIT Press, Cambridge. 15.

Mnih, V., K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra and M. Riedmiller (2013). "Playing atari with deep reinforcement learning." arXiv preprint arXiv:1312.5602.

Mnih, V., K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland and G. Ostrovski (2015). "Human-level control through deep reinforcement learning." Nature **518**(7540): 529-533.

Montufar, G. F. (2014). "Universal approximation depth and errors of narrow belief networks with discrete units." Neural computation **26**(7): 1386–1407.

Pascanu, R., Ç. Gülçehre, K. Cho and Y. Bengio (2014). "How to construct deep recurrent neural networks." In ICLR'2014 . 19, 199, 265, 398, 399, 400, 412, 462.

Qin, Q. and J. Feng (2017). "Imputation for transcription factor binding predictions based on deep learning." PLoS computational biology **13**(2): e1005403.

Ranzato, M., C. Poultney, S. Chopra and Y. LeCun (2007). "Efficient learning of sparse representations with an energy-based model." In NIPS'2006 . 14, 19, 509, 530, 532.

Redmon, J., S. Divvala, R. Girshick and A. Farhadi (2016). You only look once: Unified, real-time object detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Ren, S., K. He, R. Girshick and J. Sun (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in neural information processing systems.

Ren, S. Q., K. M. He, R. Girshick and J. Sun (2017). "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." Ieee Transactions on Pattern Analysis and Machine Intelligence **39**(6): 1137-1149.

Rosenblatt, F. (1958). "The perceptron: A probabilistic model for information storage and organization in the brain." Psychological review **65**(6): 386–408.

Rosenblatt, F. (1962). Principles of Neurodynamics. Spartan, New York. 15, 27.

Ruder, S. (2016). "An overview of gradient descent optimization algorithms." arXiv preprint arXiv:1609.04747.

Rumelhart, D. E., G. E. Hinton and R. J. Williams (1986). "Learning representations by back-propagating errors." Nature **323**(6088): 533–536.

Rumelhart, D. E., J. L. McClelland and T. P. R. Group (1986). Parallel Distributed Processing: Explorations in the Microstructure of Cognition, MIT Press, Cambridge. 17.

Shrikumar, A., P. Greenside and A. Kundaje (2017). "Reverse-complement parameter sharing improves deep learning models for genomics." bioRxiv: 103663.

Silver, D., A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam and M. Lanctot (2016). "Mastering the game of Go with deep neural networks and tree search." Nature **529**(7587): 484-489.

Srivastava, N., G. E. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov (2014). "Dropout: a simple way to prevent neural networks from overfitting." Journal of machine learning research **15**(1): 1929-1958.

Suwajanakorn, S., S. M. Seitz and I. Kemelmacher-Shlizerman (2017). "Synthesizing obama: learning lip sync from audio." ACM Transactions on Graphics (TOG) **36**(4): 95.

Szegedy, C., W. Liu, Y. Q. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich (2015). "Going Deeper with Convolutions." 2015 Ieee Conference on Computer Vision and Pattern Recognition (Cvpr): 1-9.

Venugopalan, S., M. Rohrbach, J. Donahue, R. Mooney, T. Darrell and K. Saenko (2015). Sequence to sequence-video to text. Proceedings of the IEEE international conference on computer vision.

Widrow, B. and M. E. Hoff (1960). "Adaptive switching circuits." Adaptive switching circuits. In 1960 IRE WESCON Convention Record, volume 4, pages 96–104. IRE, New York. 15, 21, 24, 27.

WILDML. (2016). "UNDERSTANDING CONVOLUTIONAL NEURAL NETWORKS FOR NLP." from http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/.

Zeiler, M. D. and R. Fergus (2014). "Visualizing and Understanding Convolutional Networks." Computer Vision - Eccv 2014, Pt I **8689**: 818-833.

Zhang, R., P. Isola and A. A. Efros (2016). Colorful image colorization. European Conference on Computer Vision, Springer.