ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANSTEZİ

İMGE İÇERİĞİ TABANLI NESNE SINIFLANDIRMA

Yeşim KALKAN

ELEKTRİK ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA
2020

Her hakkı saklıdır

ÖZET

Yüksek Lisans Tezi

İMGE İÇERİĞİ TABANLI NESNE SINIFLANDIRMA

Yeşim KALKAN

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik Elektronik Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Ziya TELATAR

Görüntü işleme teknolojileri savunma, medikal, haberleşme ve otomotiv olmak üzere birçok alanda kullanılmaktadır. Yapay zekanın görüntü işleme teknolojilerinde kullanılması, otomotiv sektöründe otonom araç çalışmalarını hızlandırmıştır. Bu çalışmada, özgün olarak bir yapay sinir ağı tasarlanarak ağa gelen resimlerin otomatik olarak trafik işaretlerini algılaması ve tanıması amaçlanmıştır. Bu amaçla, 10 farklı sınıfa ait işaretlerin tanınması, sınıflandırılması ve görüntü üzerinde işaretlenmesi için derin öğrenme tekniği kullanılmıştır. Tez çalışması kapsamında tasarlanan özgün YZNet ağı giriş, çıkış ve gizli katmanlar dahil 16 katmandan oluşmaktadır. YZNet öncelikle sınıflandırma için Evrişimli Sinir Ağı (CNN) yöntemi tercih edilerek Alman Trafik İşareti Tanıma Veri Kümesi (The German Traffic Sign Recognotion Benchmark) kullanılarak denetimli (gözetimli) öğrenme çeşidiyle eğitilmiştir. Eğitim için kullanılan resim sayısı 26640 adettir. YZNet ağının bu veri setindeki sınıflandırma başarı oranı %93.05 olarak gözlemlenmiştir. Resimde bulunan trafik işaretlerinin tespit edilerek ve tanınarak sınırlayıcı kutu ile işaretlenebilmesi ve etiketleyebilmesi için çalışmada tasarlanan YZNet ve literatürde yer alan AlexNet ve Cifar10Net ağları, Bölgesel Evrişimli Sinir Ağı (R-CNN) yöntemi tercih edilerek tekrar eğitilmiştir. R-CNN yöntemiyle yapılan eğitim ve testler için Alman Trafik İşareti Bulma Veri Kümesi (The German Traffic Sign Detection Benchmark) kullanılmıştır. Veri setinde 600 adet resim eğitim, 300 adet resim test için kullanılmıştır. Yapılan performans değerlendirmelerinde yanlış alarm oranı en düşük AlexNet'de 0.2, Cifar10Net'de 0.25 ve YZNet'de 0.72 olarak elde edilmiştir. Doğruluk değerleri ise en yüksek AlexNet'de %98.33, Cifar10Net'de %98.33 ve YZNet'de %96.95 oranlarında elde edilmiştir.

Temmuz 2020, 74 sayfa

Anahtar Kelimeler: Görüntü İşleme, Derin Öğrenme, Tanıma, Sınıflandırma

ABSTRACT

Master Thesis

IMAGE CONTENT BASED OBJECT CLASSIFICATION

Yeşim KALKAN

Ankara University
Graduate School of Natural and Applied Sciences
Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Ziya TELATAR

Image processing technologies are used in many fields which are defense, medical, communication and automotive. The use of artificial intelligence in image processing technologies has led to speed up autonomous vehicle studies in the automotive sector. In this study, it is aimed at an artificial neural network that was originally designed to automatically detect and recognize traffic signs on the images coming to the network. For this purpose, supervised deep learning technique was used to identify, classify and mark the signs belongs to 10 different classes. YZNet network, which is designed within the scope of the thesis, consists of 16 layers including input, output and hidden layers. Firstly, YZNet was trained using the German Traffic Sign Recognition Benchmark, using the Convolution Neural Network (CNN) method with supervised learning for classification. The number of pictures used for education is 26640. In this data set, the accuracy of the YZNet network was 93.05%. YZNet network designed in the study and AlexNet and Cifar10Net networks are available in the literature were retrained using the Regional Convolution Neural Network (R-CNN) method to detect and recognize traffic signs in the image and mark them with a bounding box and put the label on them. The German Traffic Sign Detection Benchmark was used for training and testing with the R-CNN method. 600 pictures in the data set were used for training and 300 pictures for testing. In the study the lowest false alarm rates were obtained as 0.2 in AlexNet, 0.25 in Cifar10Net and 0.72 in YZNet. The highest accuracy values were obtained as 98.33% for AlexNet, 98.33% for Cifar10Net and 96.95% for YZNet.

July 2020, 74 pages

Key Words: Image processing, deep learning, detection, classification

TEŞEKKÜR

Tez çalışmam sırasında bilgi, öneri ve yardımlarını esirgemeyen danışman hocam sayın Prof. Dr. Ziya TELATAR'a (Ankara Üniversitesi Elektrik-Elektronik Mühendisliği Anabilim Dalı) teşekkürlerimi sunarım. Eğitim hayatım boyunca maddi manevi desteklerini esirgemeyen değerli anneme, babama ve ablama çok teşekkür ederim. Tezim boyunca verdiği desteklerden dolayı sevgili nişanlım Çağrı CABIOĞLU'na teşekkürler ederim.

Yeşim KALKAN
Ankara, Temmuz 2020

İÇİNDEKİLER

TEZ ONAY SAYFASI

ETİK ..... **i**

ÖZET ..... **ii**

ABSTRACT ..... **iii**

TEŞEKKÜR ..... **iv**

KISALTMALAR DİZİNİ ..... **vii**

ŞEKİLLER DİZİNİ ..... **viii**

1. GİRİŞ **1**

1.1 Yapay Zekâ **2**

1.2 Makine Öğrenmesi **4**

1.3 Derin Öğrenme **5**

1.3.1 Denetimli ve denetimsiz öğrenme **7**

2. KAYNAK ÖZETLERİ **8**

3. MATERYAL ve YÖNTEM **12**

3.1 Materyal **12**

3.1.1 Alman Trafik İşareti Tanıma Veri Kümesi (The German Traffic Sign Recognition Benchmark) **12**

3.1.2 Alman Trafik İşareti Bulma Veri Kümesi (The German Traffic Sign Detection Benchmark) **13**

3.1.3 AlexNet **14**

3.1.4 Cifar10Net **15**

3.2 Yöntem **17**

3.2.1 Yapay sinir ağları **19**

3.2.1.1 Geri yayılım algoritması **21**

3.2.2 Evrişimsel yapay sinir ağları **23**

3.2.2.1 Giriş katmanı **24**

3.2.2.2 Evrişim katmanı **25**

3.2.2.3 Havuzlama katmanı **26**

3.2.2.4 Düzleştirilmiş doğrusal birim katmanı (ReLU Katmanı) **27**

3.2.2.5 Tam bağlantı katmanı **28**

3.2.2.6 Entropi (Softmax) katmanı 29
3.2.2.7 Normalizasyon katmanı 29
3.2.2.8 Seyreltme (Droput) katmanı 30
3.2.3 Bölgesel evrişimli sinir ağı (R-CNN) 30
3.2.4 YZNet yapısı ve sınıflandırma için eğitimi 33
3.2.5 YZNet, AlexNet ve Cifar10Net'in tanıma için eğitimi 36
4. ARAŞTIRMA BULGULARI 38
5. SONUÇ 65
KAYNAKLAR 66
EKLER 70
EK 1 AlexNet ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans 71
EK 2 Cifar10Net ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans 72
EK 3 YZNet ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans 73
ÖZGEÇMİŞ 74

KISALTMALAR DİZİNİ

CNN	Convolutional Neural Network
ESA	Evrişimsel Sinir Ağı
ELM	Aşırı Öğrenme Makineleri
GPU	Graphics Processing Unit
GTSDB	German Traffic Sign Detection Benchmark
GTSRB	German Traffic Sign Recognition Benchmark
ILSRVC	ImageNet Large scale Visual Recognition Challenge
ReLu	Rectified Linear Unit
R-CNN	Regional Convolutional Neural Network
RGB	Red,Green, Blue
UoI	Intersection over Union
YSA	Yapay Sinir Ağı

ŞEKİLLER DİZİNİ

Şekil 1.1 Derin öğrenme kapsam haritası 6
Şekil 3.1 Alman Trafik İşareti Tanıma veri kümesinden bütün sınıfları içeren örnek görüntüler 12
Şekil 3.2 Alman trafik işareti bulma veri kümesinden örnek görüntüler 13
Şekil 3.3 AlexNet mimarisi.....14
Şekil 3.4 Cifar10 veri seti örneği 15
Şekil 3.5 Cifar10Net yapısı.....16
Şekil 3.6 Tez çalışmasının aşamaları 18
Şekil 3.7 Biyolojik sinir hücresi.....19
Şekil 3.8 Algılayıcı 20
Şekil 3.9 Gradyan iniş tekniğinin örnek gösterimi 23
Şekil 3.10 Görüntü üzerinde evrişim katmanın filtreleme değer hesaplanması.....26
Şekil 3.11 Maksimum havuzlama yöntemi.....27
Şekil 3.12 Düzleştirilmiş doğrusal birim katmanının çıkış verisine etkisi.....28
Şekil 3.13 Tam bağlı katmanda nöronların birbiri ile bağlantısı 29
Şekil 3.14 Seyretme (Droput) katmanı 30
Şekil 3.15 R-CNN metot yapısı 31
Şekil 3.16 Seçici arama algoritmasının çalışmasının görselleştirilmiş formu 32
Şekil 3.17 YZNet yapısı.....33
Şekil 4.1 YZNet eğitimi grafikleri 38
Şekil 4.2 YZNet ROC eğrisi 39
Şekil 4.3 YZNet ikinci katmandaki ilk evrişim katmanın çıkarttığı özellikler 40
Şekil 4.4 YZNet beşinci katmandaki ikinci evrişim katmanın çıkarttığı özellikler 41
Şekil 4.5 YZNet sekizinci katmandaki üçüncü evrişim katmanın çıkarttığı özellikler.....41
Şekil 4.6 Cifar10Net ikinci katmandaki ilk evrişim katmanın çıkarttığı özellikler 42
Şekil 4.7 Cifar10Net beşinci katmandaki ikinci evrişim katmanın çıkarttığı özellikler 43
Şekil 4.8 Cifar10Net sekizinci katmandaki üçüncü evrişim katmanın çıkarttığı özellikler.....43
Şekil 4.9 AlexNet ikinci katmandaki ilk evrişim katmanın çıkarttığı özellikler.....44
Şekil 4.10 AlexNet altıncı katmandaki ikinci evrişim katmanın çıkarttığı özellikler 44
Şekil 4.11 AlexNet onuncu katmandaki üçüncü evrişim katmanın çıkarttığı özellikler.....45
Şekil 4.12 AlexNet onikinci katmandaki dördüncü evrişim katmanın çıkarttığı özellikler.....45
Şekil 4.13 AlexNet ondördüncü katmandaki beşinci evrişim katmanın çıkarttığı özellikler.....46
Şekil 4.14 YZNet'in dur işaretini tanıyarak işaretlemesi.....47
Şekil 4.15 YZnet'in hız limiti 80 işaretini tanıyarak işaretlemesi 47
Şekil 4.16 YZNet'in hız limiti 50 işaretini tanıyarak işaretlemesi.....48

Şekil 4.17 YZet'in hız limiti 30 işaretini tanıyarak işaretlemesi 48
Şekil 4.18 YZnet'in öncelikli yol işaretini tanıyarak işaretlemesi 49
Şekil 4.19 YZet'in kamyonlar için öndeki taşıtı geçmek yasaktır işaretini tanıyarak işaretlemesi 49
Şekil 4.20 YZNet'in sağdan devam et işaretini tanıyarak işaretlemesi.....50
Şekil 4.21 YZNet'in yol ver işaretini tanıyarak işaretlemesi 50
Şekil 4.22 Cifar10Net'in sağdan devam et ve dur işaretlerini tanıyarak işaretlemesi 51
Şekil 4.23 Cifar10Net'in hız limiti 80 işaretinin tanıyarak işaretlemesi.....51
Şekil 4.24 Cifar10Net'in hız limiti 70 işaretinin tanıyarak işaretlemesi.....52
Şekil 4.25 Cifar10Net'in hız limiti 50 işaretinin tanıyarak işaretlemesi.....52
Şekil 4.26 Cifar10Net'in hız limiti 30 işaretinin tanıyarak işaretlemesi.....53
Şekil 4.27 Cifar10Net'in öncelikli yol işaretini tanıyarak işaretlemesi 53
Şekil 4.28 Cifar10Net'in kamyonlar için öndeki taşıtı geçmek yasaktır işaretinin tanıyarak işaretlemesi 54
Şekil 4.29 Cifar10Net'in sağdan devam et işaretini tanıyarak işaretlemesi.....54
Şekil 4.30 Cifar10Net'in yol ver işaretini tanıyarak işaretlemesi 55
Şekil 4.31 Cifar10Net'in tehlike işaretini tanıyarak işaretlemesi 55
Şekil 4.32 AlexNet'in dur işaretini tanıyarak işaretlemesi 56
Şekil 4.33 AlexNet'in hız limiti 80 işaretini tanıyarak işaretlemesi 56
Şekil 4.34 AlexNet'in hız limiti 50 işaretini tanıyarak işaretlemesi 57
Şekil 4.35 AlexNet'in hız limiti 50 işaretini tanıyarak işaretlemesi 57
Şekil 4.36 AlexNet'in hız limiti 30 işaretini tanıyarak işaretlemesi 58
Şekil 4.37 AlexNet'in yol ver işaretini tanıyarak işaretlemesi 58
Şekil 4.38 AlexNet'in sağdan devam et işaretini tanıyarak işaretlemesi 59
Şekil 4.39 AlexNet'in hız sınırı 30 işareti için %40 %50 ve %70 oranları için hata matrisi.....61
Şekil 4.40 AlexNet'in hız sınırı 70 ve dur işareti için %70 oranı için hata matris 61
Şekil 4.41 Cifar10Net'in dur işareti için %40 oranı için hata matrisi.....62
Şekil 4.42 YZNet'in hız sınırı 80 işareti için %40 oranı için hata matrisi.....62
Şekil 4.43 YZNet'in sağdan devam et işareti için %40 oranı için hata matrisi 63
Şekil 4.44 YZNet'in sağdan devam et işareti için %50 oranı için hata matris 63

1. GİRİŞ

Yapay zeka, insan tarafından yapıldığında zeka olarak adlandırılan davranışların (akıllı davranışların) makina tarafından da yapılmasıdır. Yapay zekanın amacı insanın zekasını bilgisayar aracılığı ile taklit etmek, bu anlamda belli bir ölçüde bilgisayarlara öğrenme yeteneği kazandırabilmektir. Günümüzde, otomotiv, medikal, savunma, robotik teknolojilerinde yapay zeka kullanılarak bu alanlarda karşılaşılan birçok problem daha kolay ve kısa zamanda çözülmeye başlanmıştır.

Otomotiv dünyasının araç tasarım temel ilkeleri arasında emniyetli sürüş, sürücü ve yolcu güvenliği ile yolcu konforu yer almaktadır. Otomotiv sektörü dijitalleşen dünyaya uyum sağlayacak şekilde, bu ilkeleri öncelik kabul ederek geliştirdiği araçlara birçok yapay zeka uygulamalarını barındıran yeni özellikler eklemektedir. Şerit takibi, öndeki aracı takip ederek adaptif hız ayarı sistemi ve kendi kendine park edebilme gibi özellikler eklenen yeni özelliklere örnek olarak verilebilmektedir (Yıldırım 2019).

Trafik akışını, yaya ve yolcu güvenliğini sağlayan temel unsurlardan biri de trafik işaretleridir. Trafikte kullanılan araç sayısı ve yolların karmaşıklığının artması trafik işleyişinin karmaşıklaştırmaktadır. Trafik işaretleri gerekli uyarıları yaparak ve yol göstererek sürücülere emniyetli bir sürüş sağlarlar. Sürücülerin bu işaretleri fark etmediği, edemediği ya da bu uyarıları dikkate almayarak kurallara uymadığı durumlarda, sürüş emniyetinin azaldığı gözlenmektedir. Bu nedenle, otomotiv firmaları güvenliği arttırıcı görsel sistemler ekleyerek erken uyarı ve otonom araçlar üzerinde çalışmalar başlatmıştır (Gürbüz 2010).

Görüntü üzerinden nesne tespiti araştırmaları her zaman ilgi çekici ve zorlayıcı bir alan olmuştur. Karşılaşılan zorluklar arasında, tespit edilmek istenen nesnenin fazla çeşitliliğe sahip olması, nesnede ışık veya hava durumu kaynaklı bozulmaların bulunması ile nesnenin başka nesneler tarafından kısmen örtülmesi yer almaktadır. Yapay sinir ağları, makine öğrenmesi ve derin öğrenme teknikleri nesne tespiti ve sınıflandırmada karşılaşılan problemlere başarılı bir şekilde çözüm bulabilmeyi başarmaktadır. Otomotivde sürücülerin fark etmediği trafik işaretlerini algılayıp

tanıyarak, sürücüye erken uyarıda bulunması istenen ek özellikler veya sürüş idarisini trafik işaretlerine göre yapılandırmasını sağlayan otonom araç üzerine tasarım çalışmaları yapılmaktadır (Yıldırım 2019). Derin öğrenme, belirtilen konularda gösterdiği yüksek problem çözme yeteneği nedeniyle tasarlanmakta olan araçlarda kullanılan bir tekniktir (LeCun vd. 1015).

Yapılan bu tez çalışmasında, özgün olarak bir yapay sinir ağı tasarlanarak ağa gelen resimlerin otomatik olarak trafik işaret veya işaretlerini tanıması amaçlanmıştır. Kullanıcı müdahalesi gerektirmeden trafik işaretinin tanınması ve sınıflandırmasının yapılması yapılmıştır. Bu amaçla, başarılı bir şekilde trafik işaretlerinin tanınması ve sınıflandırılması için derin öğrenme metodu kullanılmıştır. Ayrıca sunulan sistemde aranan trafik işaretinin bulunduğu görüntü üzerinde sınırlayıcı kutu ile başarılı bir şekilde çizdirilerek trafik işaretinin ait olduğu sınıfı etiketle gösterilmiştir.

1.1 Yapay Zekâ

Yapay zekâ, insan zekası gerektiren görevleri gerçekleştirebilen makinalar yapma bilimidir (Bunhanan 2005). Ray Kurtwell'e göre de yapay zekanın tanımı, insan beyni gibi düşünüp karar verme yetisine sahip makinalar yaratma sanatıdır (Kurtzweil 1990).

Beynin başlıca görevleri düşünme, akıl yürütme, kavrama, nesneleri algılama ve sonuç çıkarmadır. İnsan beyni matematiksel hesaplama, rakam ve harfleri tanıyabilme, hafızada bilgi saklayabilme, karar verebilme yeteneklerine sahiptir. İnsan zekasına ait bu basit fonksiyonları gerçekleştirmede bilgisayarların daha başarılı olduğu görülmektedir. Bu nedenle, insana özgü bu rasyonel niteliklerin dijitalleştirilmesi isteği bilim insanlarını bu konularda daha fazla araştırma yapmaya sevk etmiştir.

Birleşik Devletler Dartmouth'da düzenlenen bir konferansta, McCarthy ve arkadaşları bilgisayar programlarının zekâ ile donatılabilme olasılığının araştırılmasını önererek, yapay zekanın doğuşunu sağlamışlardır. Yapay zekâ terimi de ilk defa John McCarthy tarafından kullanılmıştır (Russell 2016).

Yapay zekanın kullanıldığı popüler çalışmalar tarih sırasına göre örnekler verilecek olursa, 1997 yılında Deep Blue adlı bilgisayarın Dünya Satranç Şampiyonu Kasparov'a karşı kazanması, 2011'de Apple tarafından Siri uygulamasının IOS işletim sistemine sahip akıllı telefonlarda kullanılması, 2014 yılında Google'ın sürücüsüz aracının prototipinin çıkartılması ile 2016'da DeepMind tarafından geliştirilen AlphaGo isimli yapay zekanın satrançtan daha karmaşık olan 'Go' adlı oyunda dünya şampiyonunu yenmesidir (Tan 2019).

Çağımızda Bilgisayar teknolojisinin gelişmesiyle beraber eğitim, savunma, medikal gibi alanlarda daha az iş gücü ve maliyetle en iyi performans elde edilmesi amaçlanarak, yapay zekaya sahip makinalar kullanılmaktadır. Bu tarz teknolojik sistemleri kullanmak isteyen sayısız üst düzey şirket için bu teknolojiye yatırım yapmak cazip bir hal almaktadır. Narrative Science tarafından yapılan anket çalışmasında göre şirketlerin %38'nin yapay zekâ teknolojisine sahip olduğunu bu oranın 2018 yılında %62'ye çıkmasını beklendiği açıklanmıştır. IDC'ye göre 2016 yılında 8 milyar dolar olan yapay zeka yatırımlarının 2020 yılında 47 milyon dolara çıkması beklenmektedir (Tan 2019).

Yapay zekanın kullanıldığı teknolojiler aşağıda sıralanmıştır;
Makine Öğrenmesi Platformları: Tahmin ve sınıflandırma içeren birçok kurumsal uygulamada makine öğrenmesi platformu kullanılmaktadır. Veri sağlama ve model tasarlama ve algoritmada diğer makinalara dağıtmak için veri işleme desteği sağlamaktadır.

Doğal Dil Oluşturma: Dijital veriden metin üretmek amacıyla kullanılmaktadır.
Yapay Zekâ Optimizasyonlu Donanımlar: Grafik işleme birimleri (GPU) ile yapay zekâ problemlerini etkili bir şekilde çözebilmek için tasarlanmış aygıtlardır. Bu çalışmalarda kullanılmakta olan matematiksel işlemler grafik işleme birimleri sayesinde daha hızlı hesaplanmaktadır.
Derin Öğrenme Platformları: Derin öğrenmeye dayalı yapay sinir ağları, örüntü algılama, çok büyük veri setlerinin sınıflandırılması ve nesne algılama uygulamalarında tercih edilmektedir.

Robotik Süreç Otomasyonu: İnsanların yapmakta zorlanarak vakit kaybına sebep olan uygulamalarda robotik süreç otomasyonunda kullanılmaktadır. Böylece, zaman ve iş gücü daha verimli bir şekilde kullanılmaktadır.

Konuşma Algılama: İnsan konuşmasının gerekli formatlara dönüştürmektedir. Bu teknoloji çeşitli mobil uygulamalarda kullanılmaktadır (Tan 2019).

1.2 Makine Öğrenmesi

Makine öğrenmesi, yapay zekâ ve istatistiğe dayalı veri kümelerinden bilgi üretmemizi sağlayan yönteme verilen addır (Mitchell 1997).

Günümüzde teknolojinin gelişip yaygınlaşmasıyla sebebiyle veri toplamak daha kolaylaşmıştır. Sosyal medya verileri, internet ortamından sanal alışveriş, elektronik postalar gibi teknolojik uygulamalar sayesinde bir günde gigabayt değerinde veri tutulabilmektedir. Bu verilerin anlam kazanabilmesi için analiz edilmesi gerekmektedir. Büyük verilerin analizinde makine öğrenmesinin gerçekleştirilebilmesi için çeşitli algoritmalar geliştirilmektedir (Tan 2019).

Makine öğrenmesinde iki çeşit öğrenme yöntemi kullanılmaktadır. Bunlar denetimli ve denetimsiz öğrenmedir. Bu öğrenme yöntemlerinde de geliştirilen algoritmalardan yararlanılmaktadır.

Gözetimli öğrenme, etiketlenmiş gözleme dayalı bir öğrenme çeşididir. Bu öğrenme yönteminde verileri “Sınıflandırma” veya “Regresyon” metotları uygulanmaktadır.

Gözetimsiz öğrenme ise, etiketlenmemiş gözleme dayalı bir öğrenme türüdür. Bu yöntemde, algoritmanın kendi kendine görünmeyen örüntüyü keşfederek öğrenmesi beklenmektedir. Bu öğrenme çeşidinde, “Kümeleme” ve “Boyut Azalımı” metotları uygulanmaktadır (Sertkaya 2018).

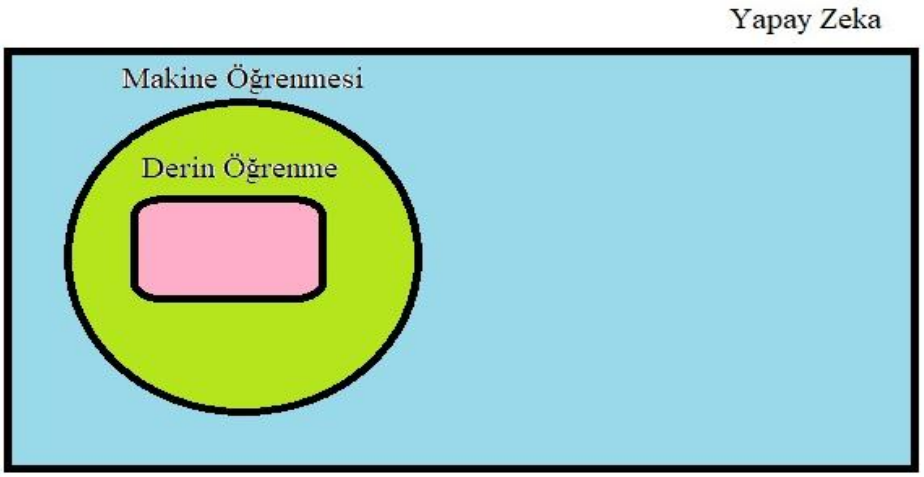
Makine öğrenmesinde kullanılan terimleri aşağıda açıklanmıştır;
Eğitim Verisi: Eğitimin sonucunda elde edilen çıktı vektörü ve giriş verilerinden oluşmaktadır (Sertkaya 2018).

Çapraz Doğrulama: Makine öğrenmesi eğitim ve doğrulama aşamalarından oluşmaktadır. Eğitim için kullanılan veriler kendi içlerinde kümelere ayrılır. Eğitim yaparken bu kümelerden biri kullanmakta, diğer kümeler ise doğrulama için kullanılmaktadır. Bu işlem, her küme için çapraz olarak tekrarlanmaktadır (Sertkaya 2018).
Gürültü: Gürültü ölçülen bir değerin rastgele bir varyans hatasıdır (Kalapanidas 2003). Ezberlemede öğretilen veriler setleriyle çok başarılı olurken, yeni bir veri setinde çok başarısız olmaktadır. Herhangi bir anlamı olmayan konuyla eğitimin ezber yapması sonucu ortaya çıkar. Makine öğrenmesinde gürültü istenmeyen bir etken olduğu için gürültü azaltma çalışmaları yapılmaktadır (Sertkaya 2018).

Aşırı Beslenme: Verinin çok büyük veya çok küçük olması durumudur. Gürültü, bu durumun ortaya çıkma sebeplerinden bir tanesidir (Sertkaya 2018).

1.3 Derin Öğrenme

Derin öğrenme, beynin işleyişi ve görevinden esinlenilerek yapay sinir ağları aracılığıyla gerçekleştirilen bir algoritmadır. Sistematik olarak incelendiğinde yapay zekanın bir alt dalında makine öğrenmesi yer almaktadır. Makine öğrenmesinin bir alt kolunda da derin öğrenme bulunmaktadır (LeCun 2015). Şekil 1.1'de derin öğrenme kapsam haritası gösterilmiştir.



Şekil 1.1 Derin öğrenme kapsam haritası (Chollet 2018)

Derin öğrenme, insan beyninin bilgiyi karmaşık bir hiyerarşiden geçirerek işlediğini ortaya çıkaran sinir bilimi keşiflerinden esinlenerek bulunmuştur (Lee vd. 2003). Bu tekniğin temelleri yapay sinir ağlarına dayanmaktadır. Derin ifadesi birbiriyle bağlantısı bulunmakta olan birçok katman olduğunu betimlemektedir. Bu katmanlardan birinin çıktısı diğerinin girdisi olmaktadır. Derin öğrenme mimarisinin katmanlarından biri olan evrişim katmanı ile verilerden farklı seviyelerde özellik haritası çıkartılmaktadır. Bu yöntemin makine öğrenmesinden farkı ise öğrenmenin çıkartılan bu özellik haritasından yararlanılarak yapılmasıdır (Lee 2003; Le 2011).

Eğitim için kullanılan derin öğrenme algoritmasında görüntü üzerindeki tüm pikseller giriş verisi olarak kullanılmaktadır. Renkli veya gri tonlu görüntüler eğitim için giriş verisi olarak kullanılabilmektedir. Gri tonlu görüntülerin tek boyutlu olması sebebiyle giriş verisi ve görüntü piksel sayısı eşittir. Renkli görüntülerin kırmızı, yeşil ve mavi renkli frekansları olması nedeniyle üç boyut bulunmaktadır. Dolayısıyla, giriş verisi görüntü piksel sayısının üç katı kadar olmaktadır (Tan 2019)

Geleneksel makine öğrenme tekniği doğal verileri işleme yeteneği açısından kısıtlıdır. Desen tanıma ve makine öğrenme sistemi dikkatli bir mühendislik bilgisi ve ham veri özniteliği çıkartabilmesi için iyi bir derecede uzmanlık gerektirmektedir. Derin öğrenme, çeşitli uygulamalarda elde ettiği yüksek başarıları sebebiyle sınıflandırma, tanıma ve tespit için kullanılmaktadır (LeCun 2015).

1.3.1 Denetimli ve denetimsiz öğrenme

Derin öğrenmede iki tür öğrenme yöntemi vardır. Bunlar gözetimli ve gözetimsiz öğrenmedir (LeCun 2015).

Bir tür öğrenme yöntemi olan gözetimli (denetimli) öğrenme, makine öğrenimi ve derin öğrenmede kullanılır (LeCun 2015). Genel olarak, x girişleri ve y çıkışları eğitim seti olarak verilir ve öğrenme algoritması x girişlerini y çıkışlarıyla ilişkilendirilir. Denetimli öğrenme eğitim setinin en önemli noktası kategorilere göre etiketlenmiş olmasıdır. Bu kategorize edilmiş çıktılar bir insan "süpervizörü" veya "otomatik" olarak sağlanır (Goodfellow 2016). Sınıflandırma uygulamalarında denetimli öğrenme tercih edilen bir yöntemdir. Makineye bir görüntü gösterilir, her kategori için birer çıkış vektörü üretilir. Temel amaç, istenen kategori etiketli puanının diğer kategorilerin puanından daha yüksek olmasıdır. Eğitimden sonra hata fonksiyonu hesaplanır. Hata fonksiyonu, çıkış skoru ile istenen model arasındaki değişikliktir. Daha iyi sonuçlar için sistem parametreleri değiştirerek geliştirilebilir (LeCun 2015).

Denetimli öğrenmenin aksine, denetimsiz öğrenmeyi bir amir veya bir hedeften ayıran objektif bir test yoktur. Girdi bilgisine göre, insan emeği gerekmeksizin yoğunluk tahmini ve dağılımına göre bir öğrenme türüdür (Goodfellow 2016).

2. KAYNAK ÖZETLERİ

Trafik işaretinin bulma ve sınıflandırılması görüntü alma, görüntüdeki nesneyi bulma ve tanıma basamaklarından oluşmaktadır. Araç içinde veya üzerinde yer alan video kameralar, sensörler ile trafik işaretinin içerisinde yer aldığı veya yer almadığı görüntüler alınmaktadır. Alınan görüntüler ilgili donanıma iletilerek, nesne bulma ve sınıflandırma amacıyla geliştirilen algoritmalar ile işlenmektedir (Aki 2019).

Fukushima (1979) tarafından önerilen "*Neocognitron*" modeliyle çok katmanlı yapay sinir ağlarının örüntü tanıma ve görüntü işleme için kullanımının önünü açılmıştır (Fukushima 1988). Böylelikle mühendislik uygulamalarında yapay sinir ağlarının kullanımı tercih edilen bir yöntem haline gelmiştir.

Kohonen (1982) kendiliğinden organize olabilen haritalar ve gözetimsiz yapay sinir ağlarıyla ilgili çalışması ile, kümelemeden sonra giren verilerin düzenli özellik eşlemelerinin yapılması sağlanmıştır. Kohonen giren verilerin anlamsal ilişkilerini, özellik haritası üzerindeki verilerin arasındaki uzaklık ile yansıtılmak olduğunu göstermiştir. İlaveten, sık giren veriler sık görülmeyen girdi verilerine göre daha büyük oranda eşlenmekte olduğunu belirtmiştir. (Chang 2002).

Rumelhart vd. (1986) yapay sinir ağları için geri yayılım (backpropagation) algoritması geliştirilmiştir (Chan vd. 1987). Hinston vd. (2006) çalışmasında çok katmanlı ileri beslemeli yapay sinir ağının, her bir katmanı Boltzman makinası ile denetimsiz olarak eğitilerek, sonrasında geri yayılım yöntemiyle ince ayar yapabileceğini göstermiştir (Şeker vd. 2017).

Aydın (2009) RGN aday görüntü gri ölçekli piksel değerleri için sinir ağı yaklaşımı, RGB aday görüntüsünün kırmızı pikseller tarafından oluşturulan binary görüntünün sinir ağı yaklaşımı, RGB piksel değerlerinin hem şablon eşleme hem de sinir ağı yaklaşımlarını kullanarak görüntüde bulunan trafik işaretinin tanıma çalışmasını yapmıştır (Aydın 2009).

Gürbüz (2010) tarafından HSV renk uzayına dönüşüm, kırmızı pikseller için eşikleme, resmi iyileştirme, resmi maskeleme, Hough Daire Dönüşümü (Hough Circle Transfer) ve şablon eşleme yöntemini yapay sinir ağlarını da kullanılarak otomatik trafik işareti tanıması çalışmasını gerçekleştirilmiştir (Gürbüz 2010).

Cireşan(2011), yaptığı çalışmada Alman Trafik İşareti Tanıma Kıyaslama (GTSRB) veri seti kullanılarak sınıflandırma yapılmıştır. Çalışmada, eğitim setinde bulunan imgelerin boyutlarını eşitlemek için bir ön işlemden geçirilmektedir. Ham pikseller üzerinde eğitilmiş bir CNN komitesinin ve standart özellik tanımlayıcıları (HOG ve HAAR) üzerinde eğitilmiş birçok katmanlı sinir ağı (MLP) kombinasyonu ile tanıma sağlanmaktadır. Hem CNN'ler hem de MLP'ler muhtemel sınıf olasılıklarına yaklaştığından, her iki modelin çıktılarının ortalaması alınarak kombinasyonu oluşturmuşlardır.

Zaklouta vd.(2012), sürüş destek sistemlerinni güvenli bir sürüş sağlaması için gerçek zamanlı olarak trafik işaretlerini sınıflandırmıştır. Sınıflandırmada eğitim ve testlerinde kullanılan veri seti Alman Trafik İşareti Tanıma (GTSRB) veri kümesidir. Yapılan çalışmada farklı boyutlu yönelimli gradyan(HOG) tanımlayıcıları ve mesafe dönüşümleri (distance transforms) kullanarak trafik işaretlerinin sınıflandırması için k-d ağaçlarının, rastsal ormanların (random forest)ve destek vektör makinelerinin (support vector machine) performansları değerlendirilmiştir. Çalışmada özellikler Fisher ölçütü ve rastsal orman ile çıkartıldığında, bellekte kullanılan alanın azaldığı ve doğruluğun arttığı görülmüştür (Zaklouta vd. 2012).

Krizhesky (2012) tarafından tam bağlantılı katman, normalizasyon ve seyreltme katmanları kullanılarak derin evrişimli sinir ağı tasarlanmıştır. 2012 yılında ILSVRC (Büyük Ölçekli Görsel Tanıma Yarışması) ImageNet yarışmasında görüntü tanıma ve sınıflandırmada derin öğrenmenin büyük başarılar elde ettiği gösterilmiştir (Krizhesky 2012). Sonuç olarak derin evrişimli sinir ağları aranan nesne veya obje resmin tümünü kaplamayıp sadece bir kısmında dahi olsa görüntü tanıma ve sınıflandırma problemlerini başarıyla çözmektedir. Başarıları sebebiyle derin öğrenmeyle ilgili yapılmakta olan çalışmalar bu yönde hız kazanmıştır (Deng vd. 2014).

Akgün(2013), elektromigrofi tabanlı protezleri başarılı bir şekilde kontrol eden elektromigrofi sinyallerini sınıflandırma çalışması yapmıştır. Elektromigrofi sinyallerinde kayıt edilirken kullanılan cihaz kaynaklı giriş ve çıkış işaretlerinde bozukluk fark edilmiştir. Bu nedenle sinyallerinin özniteliklerinin doğru olarak çıkartılabilmesi için sinyaller ön işleme tabi tutulmuştur. Ön işlemde işaretin giriş ve çıkış segmentinde bulunan bölgeler atılarak sadece ara bölgeler kullanılarak doğru işaretler elde edilmiştir. Ara segmente işaretlerin öznitelikleri yapay sinir ağı ile çıkartılmıştır. Bunun için geri yayılımlı yapay sinir ağları tercih edilmiştir. Çıkartılan bu öznitelikler ile denetimli öğrenme yöntemiyle ağ eğitilerek sınıflandırılma yeteneği kazandırılmıştır (Akgün vd.2013).

Zeng (2015), yapmış olduğu çalışmada CNN'in özellik çıkarma kabiliyeti ile aşırı öğrenme makinelerinin genelleme özelliğini birleştirmişlerdir. Bu CNN-ELM modelini Alman Trafik İşareti Tanıma Veri Seti (GTSRB) için uygulamışlardır. Bu çalışmada öncelikli olarak veri setindeki imgelerden ortalama piksel değerlerini çıkartılmıştır. Bu işlemden sonra evrişimli sinir ağı ile ileri beslemeli eğitimi yaparak, özelliklerin çıkartılması ve bu özelliklerin öğretilmesi amaçlanmıştır. ELM (aşrı öğrenme makineleri) ile ileri beslemeli öğrenme yapılmasından dolayı, daha az hesaplama gücü gerektiği tespit edilmiştir.

Li vd. (2018), obje tespitinde özgün olarak tasarladıkları DetNet kullanmışlardır. DetNet literatürde yer alan ResNet-50 tabanlıdır. Bu iki ağın birbirinden farklı havuzlama katmanında kullanılan matris boyutladır. ResNet-50 havuzlama katmanı sırasıyla kullanılan matris boyutları 4x4, 8x8, 16x16, 32x32 iken DetNet'te 4x4, 8x8,16x16 ve 16x16'dır. Sınıflandırma için evrişimli sinir ağlarından yararlanılmıştır. Objenin tanınması için Detectron adlı açık kaynak kod kullanılmıştır. Böylelikle, objeler tespit edilerek çerçeve ile çizdirilmiş ve ne olduğu etiket ile gösterilmiştir (Li vd. 2018).

Tan (2019) tarafından derin öğrenme yöntemiyle araçları marka ve modele göre sınıflandırılması yapılmıştır. Kullandığı yöntemde literatürde yer alan Resnet50, Resnet101 ve SSD Mobilnet ağlarını transfer yöntemi ile eğitmiştir. Eğitimde kullanılan veri setinde toplamda 2000 adet araç resmi bulunmaktadır. Bu veri setinde 16787 adet

resim eğitim için, 2991 adet resim test için kullanılmıştır. Yapılan çalışmada farklı netlerde sınıflandırma performansları değerlendirilmiştir (Tan 2019).

Bayram (2019), tarafından gerçekleştirilen çalışmada otomatik olarak plaka tanıma için derin öğrenme yöntemini kullanmıştır. Öncelikle veri setinde yer alan 300 adet resim manuel olarak etiketlenmiştir. Etiketlenen resimler eğitim seti ve test seti olarak ikiye ayrılmıştır. Eğitim setindeki resmiler ile açık kaynak kodlu maskeli evrişimli sinir ağı modeli ile eğitilmiştir. Eğitilen resimlerde tanıma işlemi için resmin etiketlenmesi gerekmektedir. Yapılan çalışmada tespit edilen plakanın tanınması için etiketleme işlemi Oxford Ünversitesi tarafından geliştirilen VIA görüntü etiketleme uygulaması ile yapılmıştır. Böylelikle, araç plakaları otonom olarak başarılı bir şekilde tanınması sağlanmıştır (Bayram 2019).

Sardoğan vd. (2020) çalışmasında elma ağacının yaprakları üzerindeki hastalık belirtilerini Fast R-CNN yöntemiyle Inception v2 ağ mimarisi kullanarak tespit etmiştir. Inception v2 literatürde yer alan CNN yöntemile daha önceden eğitilmiş bir ağdır. Bu ağ ile yapılan eğitimler ile yaprakların hastalıklı ve sağlıklı olarak sınıflandırma öğretilmiştir. Sınıflandırmanın ardından ağlar tanıma işlemini yapabilmesi için Faster R-CNN metoduyla tekrar eğitilmiştir. Bu eğitimin sonucunda eğitilen net yapraktaki hastalık izlerini tanımayı öğrenmiştir. Hasta olan yapraklardaki hastalık belirti izleri tanınarak ve dikdörtgen bir kutu ile çerçevelenerek, hasta etiketiyle gösterilerek sınıflandırılabilmiştir (Sardoğan 2020).

3. MATERYAL ve YÖNTEM

3.1 Materyal

3.1.1 Alman Trafik İşareti Tanıma Veri Kümesi (The German Traffic Sign Recognition Benchmark)

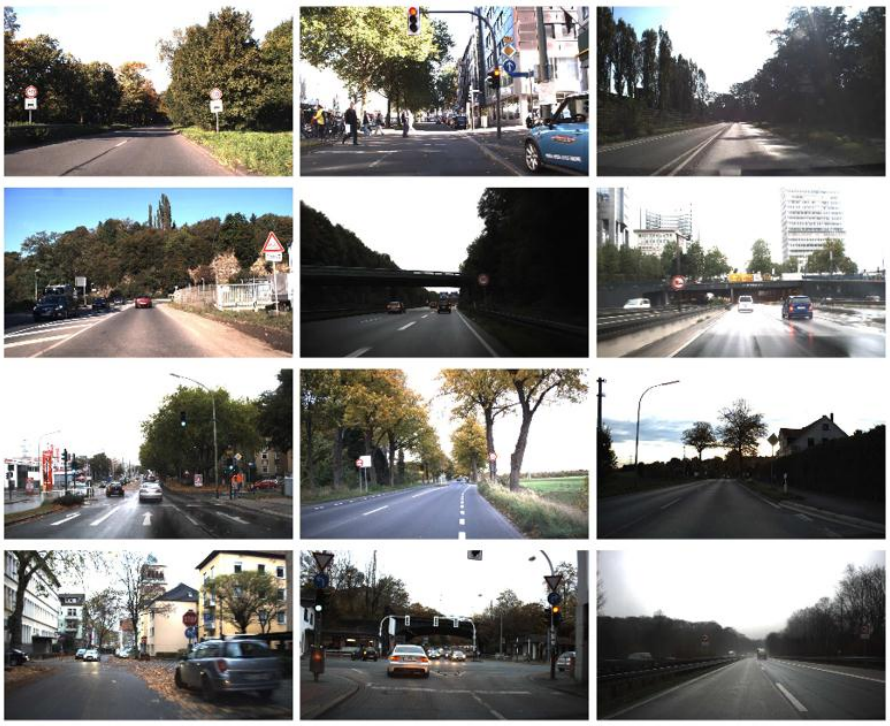
Alman Trafik İşareti Kıyaslaması, 2011 yılında düzenlenen Uluslararası Yapay Sinir Ağları Konferansında (IJCNN) çok sınıflı, tek görüntülü bir sınıflandırma mücadelesi için hazırlanmıştır. Bu veri kümesi, 43 farklı trafik işaretinin gerçeğe yakın yaklaşık 50.000 renkli görüntüsünü içermektedir. Her bir görüntü sadece bir trafik işaretini barındıracak şekilde düzenlenmiştir. Bu görüntülerin boyutları 15 x 15 ile 250 x 250 piksel arasında değişkenlik göstermektedir. Veri kümesi, 26640 adet eğitim seti ve 12569 adet test seti olacak şekilde ikiye ayrılmıştır. Veri kümesinde olan örnek görüntüler Şekil 3.1'de gösterilmiştir. Tez çalışmasında bu veri kümesi YZNet eğitimi için kullanılmıştır.



Şekil 3.1 Alman Trafik İşareti Tanıma veri kümesinden bütün sınıfları içeren örnek görüntüler

3.1.2 Alman Trafik İşareti Bulma Veri Kümesi (The German Traffic Sign Detection Benchmark)

Alman Trafik İşareti Bulma veri kümesi, 2013 yılında düzenlenen Uluslararası Yapay Sinir Ağları Konferansında (IJCNN) bilgisayarla görme, örüntü tanıma ve görüntü tabanlı sürücü yardımı alanlarına çalışan araştırmacılar için hazırlanmıştır. Görüntüler farklı hava, ışık ve sürüş koşullarını temsil edecek şekilde toplanmıştır. Bu veri kümesi, 43 farklı trafik işaretinden oluşan 900 renkli görüntü içermektedir. Her bir görüntüde bir veya birden fazla trafik işaretini barındıracak şekilde düzenlenmiştir. Barındırdığı trafik işaretleri aynı ya da farklı işaretler olabilmektedir. Bu görüntülerin boyutları 1360 x 800 pikseldir ve PPM formatındadırlar. Veri kümesine ait bütün görüntülerin trafik işaretlerini belirten gerçek referans değerleri mevcuttur. Veri kümesi, 600 adet eğitim seti ve 300 adet test seti olacak şekilde ikiye ayrılmıştır. Veri kümesinde olan örnek görüntüler Şekil 3.2'de gösterilmiştir. Tez çalışmasında bu veri kümesi önceden eğitilmiş olan YZNet, AlexNet, Cifar10Net'lerinde trafik işareti tanıma ve bulunması eğitimi ve testleri için kullanılmıştır.

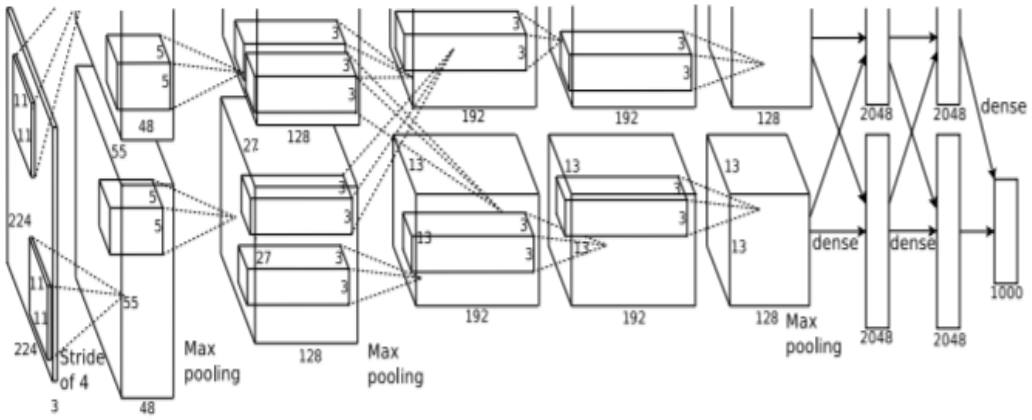


Şekil 3.2 Alman trafik işareti bulma veri kümesinden örnek görüntüler

3.1.3 AlexNet

2012 yılında yayınlanan “ImageNet Classification with Deep Convolutional Neural Networks” bilgisayarlı görme alanında yeni bir bakış açısı ortaya koymuştur. AlexNet modelinin ve tasarımının anlatıldığı bu çalışma, ImageNet Büyük Ölçekli Görsel Tanıma Yarışmasını (ILSVRC) kazanmıştır. Bu yarışma ile evrişimsel sinir ağlarının bilgisayarlı görü alanında başarısı kanıtlanmıştır. Bu çalışma, yüksek çözünürlüklü görüntülerin eğitimiyle donanımların paralelleştirilmesi alanında çalışan araştırmacılara yol göstermiştir (Krizhevsky 2012).

AlexNet’in eğitimi için 22.000 sınıfa ait internetten toplanan yaklaşık 15 milyon yüksek çözünürlüklü görüntüye sahip ImageNet veri kümesi kullanılmıştır. AlexNet’in birincilik ödülü kazandığı yarışmada, bu veri setinin bir alt kümesi kullanılmıştır. Bu alt küme, 1000 sınıfa ait yaklaşık 1,3 milyon görüntüden oluşmaktadır. Toplam görüntülerin 1,2 milyon âdeti eğitim, 50.000 âdeti doğrulama ve 150.000 âdeti ise test için ayrılmıştır. AlexNet’in eğitimi için kullanılacak görüntülere ön işlem uygulanmıştır. Ön işlem olarak, kullanılan veri kümesi, değişken çözünürlüklere sahip olduğu için eğitimde kullanılmak üzere sabit bir çözünürlük olan 256 x 256'ya çevrilmiştir. Ardından giriş katmanı için 224x224x3 boyutuna kırpılma işlemi uygulanarak yeniden ölçeklendirilmiştir (Krizhevsky 2012). Şekil 3.3’de AlexNet mimarisi gösterilmiştir.



Şekil 3.3 AlexNet mimarisi

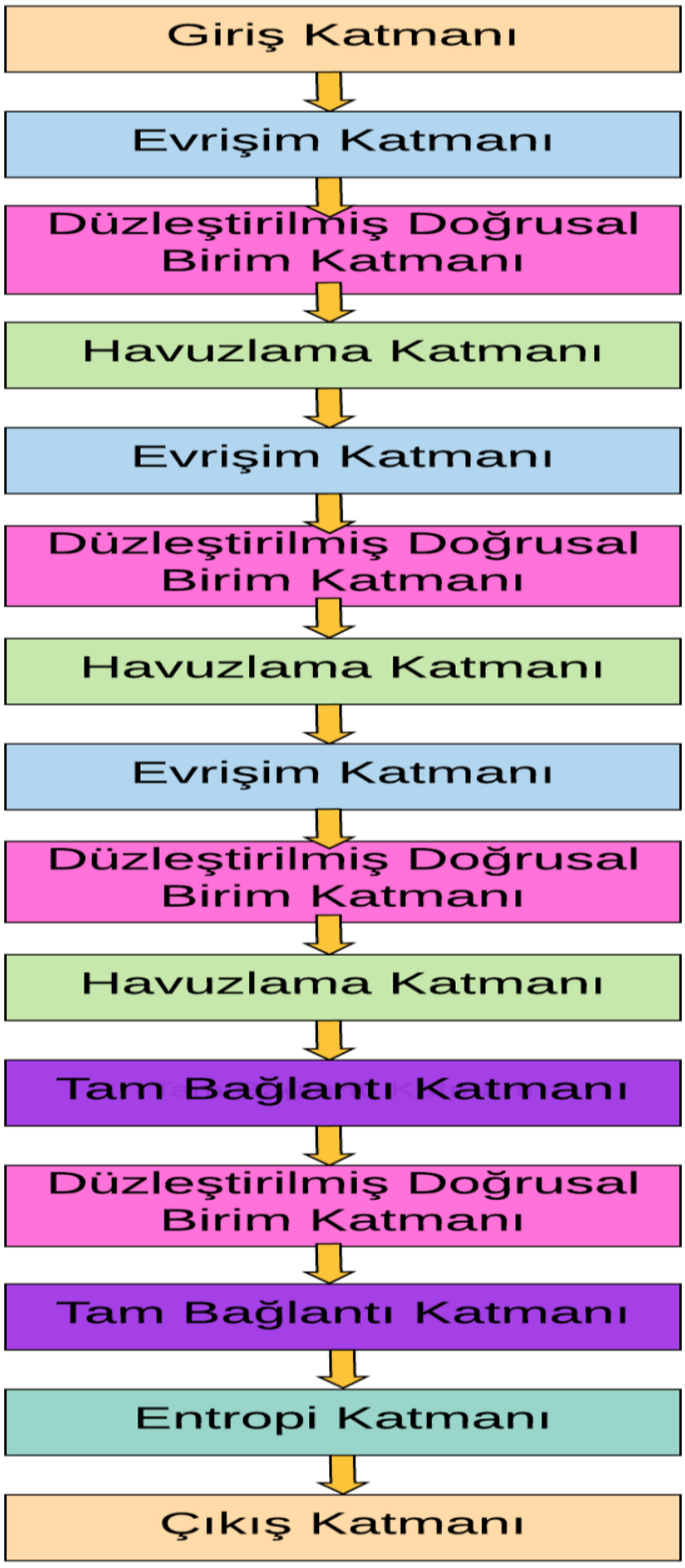
AlexNet'i günümüzde popüler yapan mimarisi değildir. Çalışmanın temel katkısı, daha önce derin öğrenme ağlarında kullanılmamış bazı özellikleri kullanmasıdır. AlexNet tanh işlevi yerine ReLU kullanmıştır. ReLU'nun standart tanh fonksiyonundan altı kat daha hızlı olduğunu bu çalışmada ortaya konulmuştur. Ayrıca, AlexNet'in eğitimi sırasında birden fazla GPU kullanılmıştır. Modelde, nöronların yarısı bir GPU'da işlenirken, diğer yarısı diğer GPU'da işlenmiştir. Bu katkı büyük boyutlara sahip ağların daha kısa zamanda eğitilmesine ön ayak olmuştur. Bir diğer katkısı ise, örtüşen havuzlama yöntemidir. Çalışmada, örtüşen havuzlama yöntemi kullanımının hata oranını yaklaşık %0.5 azalttığını tespit edilmiştir (Krizhevsky 2012).

3.1.4 Cifar10Net

Cifar10Net, Kanada İleri Araştırmalar Enstitüsü (Canadian Institute for Advanced Research) tarafından geliştirilmiştir. Netin temeli evrişimli sinir ağına dayanmaktadır ve altmış bin resim ve on sınıf için renkli resimler ile eğitilmiştir (Angelov vd. 2016). On beş katmanlı bu derin ağda, üç adet evrişim, üç adet havuzlama, iki adet tam bağlı, bir entropi katman ile giriş ve çıkış katmanları bulunmaktadır. Şekil 3.4'de Cifar10 veri setinden bir örnek gösterilmiş, Şekil 3.5'de ise Cifar10Net yapısı verilmiştir.



Şekil 3.4 Cifar10 veri seti örneği



Şekil 3.5 Cifar10Net yapısı

3.2 Yöntem

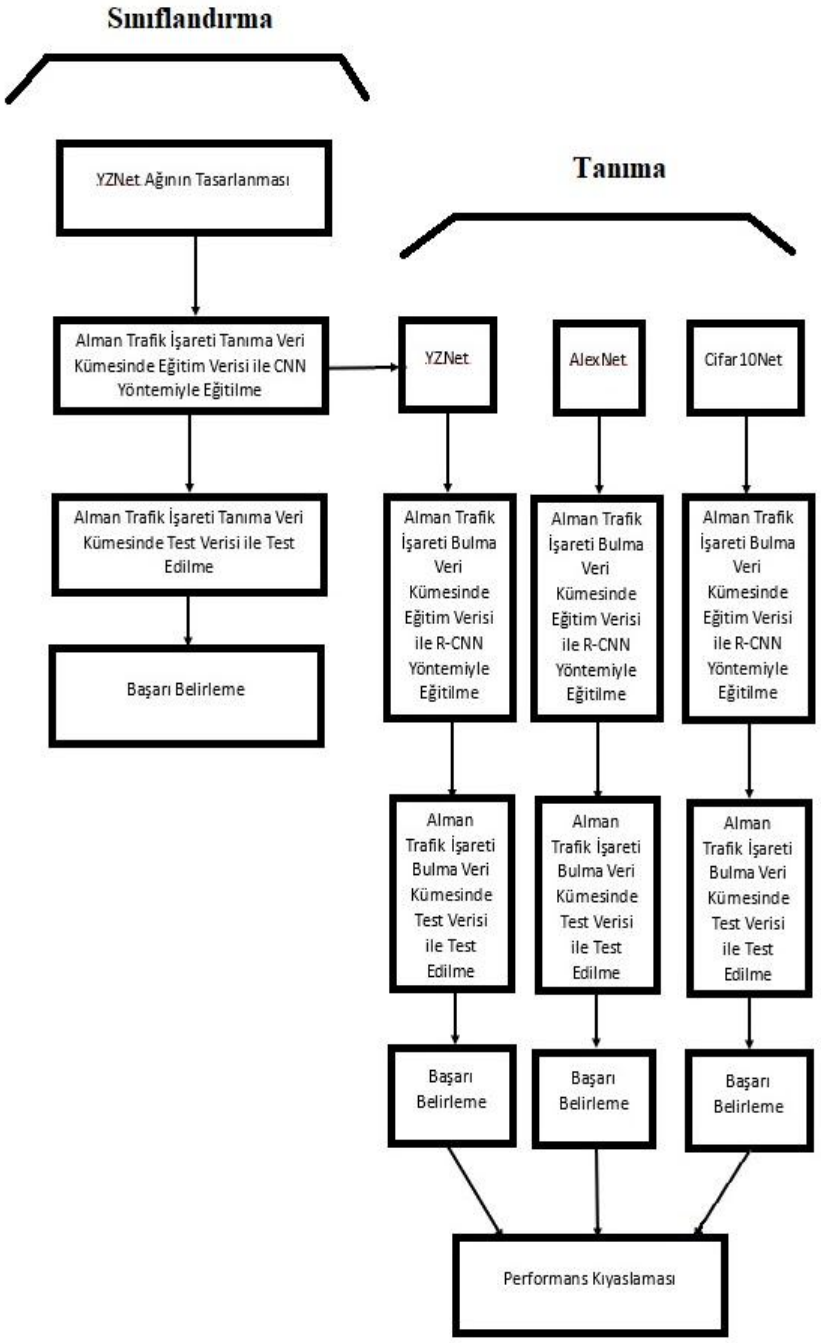
Resimde bulunan trafik işaret veya işaretlerinin tanımasını sağlayan özgün bir net tasarlanması amaçlanan çalışmamızda, hem sınıflandırma hem de tanıma için çözümler sunulmuştur.

Tanıma işlemi için öncelikle resimde tanınması istenen trafik işaretlerinin, resmin hangi bölgesinde olduğu tespit edilir ve bu bölgedeki ögelerin sınıflandırılması ile işaretin çeşidi belirlenir. Tez çalışmasının ilk aşamasında, trafik işaretlerinin sınıflandırılması amacıyla özgün bir YZNet tasarımı yapılmıştır. YZNet evrişimsel yapay bir sinir ağıdır. Tasarlanan YZNet, resmin hangi bölgesindeki yerlere bakması gerektiği problemi R-CNN yöntemi ile çözmüştür. Tanıma işlemi sonucunda YZNet, gösterilen resimlerde trafik işaretlerini algılayarak işareti sınırlayıcı kutu ile çizmiş ve trafik işaretinin adını etiket ile göstermiştir. Özgün olarak tasarlanan YZNet'in başarısını kıyaslamak için sınıflandırma işleminde başarılı olan ve literatürde yer alan AlexNet ve Cifar10Net kullanılmıştır.

AlexNet ve Cifar10Net sınıflandırma için daha önceden eğitilmiş hazır ağlardır. Doğru bir kıyaslama yapabilmek ve sınıflandırmayı öğretmek için özgün olarak tasarladığımız YZNet sinir ağı modeli evrişimsel sinir ağı kullanarak Alman Trafik İşaretleri Tanıma Veri Kümesinde yer alan 26640 adet eğitim resmi ile eğitilmiştir. Eğitilirken derin öğrenmenin bir çeşidi olan denetimli (gözetimli) öğrenme tercih edilmiştir. Eğitilen YZNet Alman Trafik İşareti Tanıma Veri Kümesinde test için ayrılan resimler ile test edilmiştir. Testin sonucunda tez çalışmasında tasarlanan YZNet başarısı saptanmıştır.

Tez çalışmasında tanıma için kullanılan resimler Alman Trafik İşareti Bulma Veri Kümesine ait resimlerdir. Bu data setinde yer alan resimler günlük hayatta çekilmiş cadde ve sokak resimleridir. Bu nedenle, resimlerde trafik işareti haricinde trafik ışıkları, arabalar, binalar, ağaçlar ve farklı tabelalar bulunmaktadır. Trafik işaret veya işaretlerinin sınıflandırılmasının yapılabilmesi için öncelikli olarak resimde aranan işaretin nerede olduğu problemi R-CNN ile çözülmüştür.

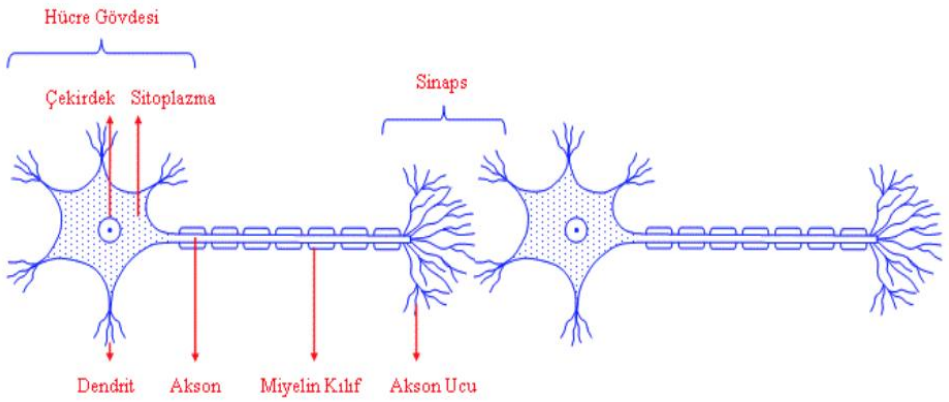
Tanıma işlemi için, özgün YZNet, AlexNet ve Cifar10Net'leri de R-CNN ile tekrar eğitilmiştir. Bu işlemin sonucunda üç tane farklı ağ ile resimlerde yer alan trafik işaret veya işaretleri sınırlayıcı kutu ile çerçevelenerek işaretlenmiş ve etiket ile ait olduğu sınıf belirlenmiştir. Sonuç olarak, üç ağın performans sonuçları kıyaslanmıştır. Şekil 3.6'da tez çalışmasının aşamaları gösterilmiştir.



Şekil 3.6 Tez çalışmasının aşamaları

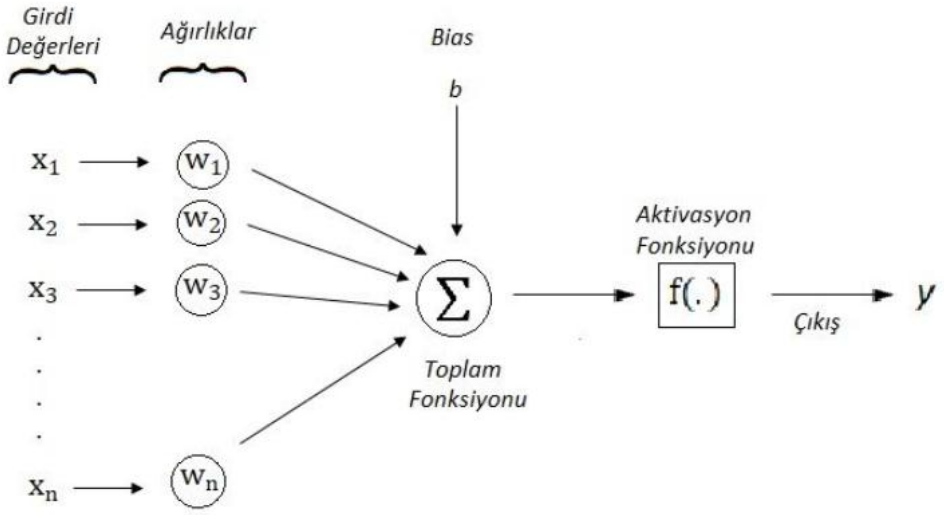
3.2.1 Yapay sinir ağları

Yapay sinir ağları (YSA), birçok karmaşık gerçek dünya sorununu çözmede yaygın olarak kullanılan nispeten yeni hesaplama araçlarıdır. YSA'ların çekiciliği, esas olarak doğrusal olmama, yüksek paralellik, hata ve gürültü toleransı ve öğrenme ve genelleme yetenekleri ile ilgili dikkate değer bilgi işleme özelliklerinden kaynaklanmaktadır. Birçok bilimsel disiplinden araştırmacılar, örüntü tanıma, tahmin, optimizasyon, ilişkilendirilebilir bellek ve kontroldeki çeşitli problemleri çözmek için YSA'lardan faydalanmaktadır. YSA, beyindeki bilgi işleme sürecini sağlayan biyolojik sinir sisteminden ilham alan bir bilgi işleme paradigmasıdır. Belirli bir problemi çözmek için birlikte çalışan çok sayıda birbiriyle bağlantılı işleme elemanlarından (nöronlar) oluşmaktadır (Arı 2017). Şekil 3.7'de biyolojik bir sinir hücresi yapısı gösterilmiştir.



Şekil 3.7 Biyolojik sinir hücresi

Biyolojik Nöronlar (sinir hücreleri), bilgiyi almak, işlemek ve iletmek ile sorumlu hücrelerdir. Sinir hücreleri hücre gövdesi ve aksonlardan oluşmaktadır. Hücre gövdesinde hücre çekirdeği, stoplazma ve dendritler bulunmaktadır. Dendrit, sinir hücresine gelen sinyali çekirdeğe ileten yapıdır. Sinir hücreleri, dendritleri sayesinde bilgiyi dış dünyadan ya da başka nöronların sinapslarından alarak çekirdeğe ulaştırabilmektedir. Bu gelen sinyallerin toplandığı yapıya soma adı verilmektedir. Soma sinir hücresinin gövdesidir. Soma topladığı bu bilgileri akson ile diğer hücrelere dağıtmaktadır. Alınan bilgi soma bölümünde zamanla işlenir ve işlenen bu bilgi çıktıya dönüştürülerek akson ve sinapslar aracılığıyla diğer nöronlara aktarılır (Hopfield 1988).



Şekil 3.8 Algılayıcı (Perceptron)

Biyolojik bir nörondan esinlenerek modellenen yapay sinir ağlarında dendritleri toplama fonksiyonu, somayı aktivasyon fonksiyonu, aksonu çıkış, sinapsı ağırlık değerleri temsil etmektedir. YSA modellerinde biyolojik nöronların yapısına benzer bir yapı kullanılmaktadır ve buna algılayıcı denmektedir (Şekil 3.8). Tek bir algılayıcı için, x_0, x_1, x_2, x_3 ... x_n diğer algılayıcılardan gelen çeşitli girişleri (bağımsız değişkenleri) temsil etmektedir. Bu girişlerin her biri bir bağlantı ağırlığı yani sinaps ile çarpılmaktadır. Ağırlıklar, w_0, w_1, w_2, w_3 ... w_n olarak temsil edilmektedir. Bias değeri b ile temsil edilmektedir. Bias değeri, aktivasyon fonksiyonunu yukarı veya aşağı kaydırmamızı sağlamaktadır. En temelde, bu ürünler toplanır (Denklem 3.1), bir çıktı verebilmek için bir aktivasyon fonksiyonu beslenir (Denklem 3.2) ve bu çıktı (y) diğer algılayıcılar için girdi olarak verilir.

$$NET = \sum_{i=1}^{n} x_i w_i + b = x_1 w_1 + x_2 w_2 + x_3 w_3 + \ldots\ldots\ldots + x_n w_n + b \qquad (3.1)$$

Aktivasyon fonksiyonunun temel amacı sisteme doğrusal olmayan karmaşıklıklar katmak ve üretilecek değerin daha anlamlı olmasını sağlamaktadır. Günümüzde öğrettiğimiz ve modellediğimiz görüntüleri, videoları, sesleri vb. gibi verileri işleneme

olanağı verir. Tercih edilen aktivasyon fonksiyonları öğrenme ve başarı performansı açısından farklılık göstermektedir. Aktivasyon fonksiyonunun birden fazla çeşidi mevcuttur. Bunlar; Lineer aktivasyon fonksiyonu, Eşik aktivasyon fonksiyonu, Sigmoid aktivasyon fonksiyonu, Tanh aktivasyon fonksiyonu ve ReLU aktivasyon fonksiyonudur. YSA'da kullanılacak aktivasyon fonksiyonu seçilirken modellenmek istenen problem ve gereksinimler göz önüne alınmalıdır (Abraham 2005).

3.2.1.1 Geri yayılım algoritması

Yapay sinir ağlarında geri yayılım algoritmasında nörona giren değerlere bir yanıt olarak neyin üretilmesi gerektiğini tanımlamak için ağın ürettiği çıktı ile olması gereken çıktı arasındaki farkın belirlenmesi gerekmektedir.Geri yayılım algoritmasının amacı, yanıt olarak alınan çıkış verisi ile beklenen çıkış verisi arasındaki hatayı minimuma indirmektir (Yıldırım 2019). Bu farkı azaltmak, ağın öğrenme performansını artırmak demektir. Maliyet fonksiyonları gerçek değer ile öngörülen değer arasındaki farkı hesaplamak için kullanılmaktadır. En çok kullanılan maliyet fonksiyonları ortalama kare hatası (mean squared error) ve çapraz entropi (cross entropy) fonksiyonlarıdır (Tan 2019).

$$C(w,b) = \frac{1}{2N}\sum_{i=1}^{N}(y(x) - a)^2 \tag{3.3}$$

Denklem 3.3 ortalama kara hatası maliyet fonksiyonunu ifade etmektedir. N eğitim setindeki eleman sayısını, a girdi indisine bağlı olarak ağın çıktısını, y ise gerçek değeri temsil etmektedir (Tan 2019).

$$C(w,b) = -\frac{1}{N}\sum_{i=1}^{N}(y \ln a + (1-y)\ln(1-\mathrm{a})) \tag{3.4}$$

Denklem 3.4 çapraz entropi maliyet fonksiyonunu ifade etmektedir. Tez çalışmasında çapraz entropi ile maliyet fonksiyonu hesaplanmıştır. Denklem 3.4.'de N eğitim setindeki eleman sayısını, a girdi indisine bağlı olarak ağın çıktısını, y ise gerçek değeri temsil etmektedir (Tan 2019).

YSA'da bulunan her katman için maliyet fonksiyonu hesaplanır ve bir sonraki girdinin eşik ve ağırlıklarını ayarlamak için kullanılır. Bu işlemdeki temel amaç, maliyet fonksiyonunu sıfıra yakınsamak veya eşitlemektir. Denklem 3.3 ve Denklem 4.4'ten de anlaşılacağı üzere, maliyet fonksiyonu ne kadar sıfır değerine yakın olursa, gerçek değer ile tahmin edilen değer birbiri ile o kadar örtüşüyor demektir. Gerçek değer ile öngörülen değer arasında bir eşitsizlik olduğu sürece, ağırlıklar ayarlanmadır. Maliyet değeri mümkün olduğunca küçük bir seviyeye indirinceye kadar bu işlem tekrarlanmalıdır (Tan 2019).

Gradyan inişi, maliyet fonksiyonu ile tespit edilen hata oranını en aza indirmek için kullanılan bir optimizasyon algoritmasıdır. Makine öğrenme algoritmalarında, modellerin maliyet fonksiyonlarının çıktısını en aza indiren parametreleri bulmak için kullanılır. Yapay sinir ağlarında bu parametreler ağırlık ve bias değerlerini temsil etmektedir (Bengio 2017, Patterson 2017,Yıldırım 2019).

Gradyan, bir fonksiyonun en büyük artışına işaret eden bir vektörü ifade etmektedir. Negatif gradyan ise bir fonksiyonun en büyük azalmasını gösteren bir vektörü temsil etmektedir. Bu nedenle, bir fonksiyonu negatif gradyan yönünde tekrarlayarak, o yönde hareket ettirmek bir fonksiyonun alabileceği en küçük değeri bulabilmektedir. Gradyan iniş maliyet fonksiyonunun türevi ile fonksiyon üzerindeki herhangi bir noktanın eğimini vermektedir. Bu mantığını kullanarak modelde kullanılan maliyet fonksiyonun en az olduğu parametreler bulunmaktadır (Yıldırım 2019).

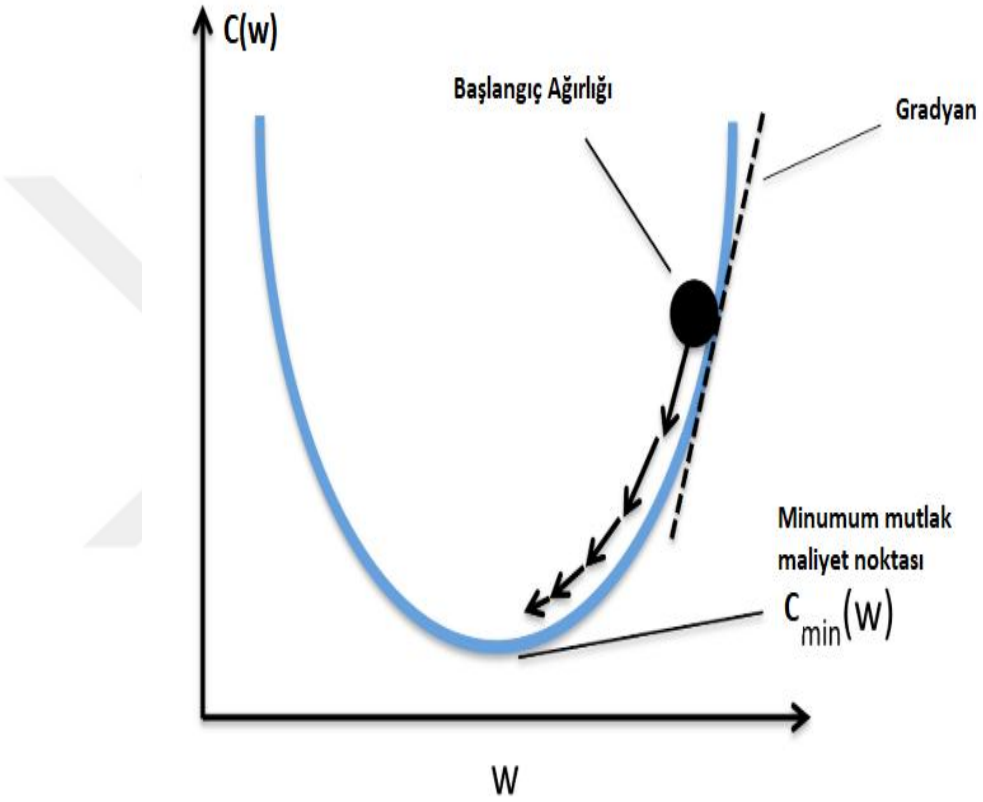
Gradyan İniş optimizasyon algoritması kullanılarak, ağırlıklar her tekrardan sonra aşamalı olarak güncellenir. Maliyet fonksiyonumuzda kontrol edebileceğimiz iki parametre vardır. Bunlar w (ağırlık) ve b (bias)'tır. Ağırlık güncellemesinin büyüklüğü ve yönü, maliyet gradyanının ters yönünde bir adım atılarak hesaplanır. Denklem 3.5'te maliyet fonksiyonun $C(w,b)$ ile, öğrenme katsayısı η ile ifade edilmiştir (Bengio 2017, Patterson 2017). Öğrenme katsayısı gradyan iniş algoritmasının her bir adımı için ne kadar gradyan iniş yapacağı belirlemektedir.

$$\Delta w_i = -\eta \frac{\partial C}{\partial w_i} \tag{3.5}$$

Δw_i ağırlıkların güncelleneceği yönü ve büyüklüğünü temsil etmektedir. Ağırlıkları güncellenme işlemi denklem 3.6'da gösterilmiştir (Bengio 2017, Patterson 2017).

$$\widehat{w}_i = w_i + \Delta w_i \tag{3.6}$$

Yalnızca tek bir ağırlık katsayısına sahip bir maliyet fonksiyonu göz önüne alındığında, bu kavram Şekil 3.9'daki gibi gösterilebilmektedir.



Şekil 3.9 Gradyan iniş tekniğinin örnek gösterimi (Patterson ve Gibson 2017)

3.2.2 Evrişimsel yapay sinir ağları

Çok katmanlı yapay sinir ağları birçok katmandan oluşmaktadır Bu katmanlar giriş ve çıkış katmanları arasında yer almaktadır. Çok katmanlı yapay sinir ağları, yüksek seviyeli hesaplamaları yapabilme yeteneği sayesinde daha karmaşık problemlerin çözümü için tercih edilmektedir (Yıldırım 2019)

Evrişimsel sinir ağları da çok katmanlı yapay sinir ağlarının çeşididir. Bu ağların

temelini evrişim işlem oluşturmaktadır. Bu yöntem yüksek hesaplama yeteneği nedeniyle derin öğrenmede kullanılmakta olan popülerleşmiş bir yöntemdir (Sertkaya 2018).

Huwel ve Wiesel 1960'lı yıllarda hayvanlar üzerine yapılan çalışmalarda ışığın tespitinde alıcı alan ismi verilen yapılarda bulunan görsel korteks hücrelerinin sorumlu olduğunu keşfetmişlerdir. Araştırmacıların maymunların görsel sinir sistemi üzerine yaptıkları çalışmalarda, maymunlara belirli desene sahip ışık demeti gösterilmiş ve beyin ön korteksinden gelen cevapları ölçmüşlerdir. Bu ölçümlerde görsel sinir sistemine ait bazı nöronlarda güçlü tepkilerin gerçekleştiği gözlemlenmiştir (Gerrits vd. 1966). 1970'li yıllarının sonunda Fukushima "*Neocognitron*" mimarisini önermiştir. Bu mimari sadece biyolojk olmayan sonuçların kullanılabilirliğine önem veren bir yaklaşım içermektedir (Fukushima 1988, Keskenler vd. 2017). Maymunlar ve kediler üzerinde yapılan deneyde nöronlardan alınan ölçümler ile memeli beyni görseli belirli bir hiyerarşik sıraya göre algılamakta olduğu keşfedildi. Öncelikli olarak beyin resimden daha basit seviyede olan çizgiler, kenarlar ve bunların birbirilerine göre olan açı özelliklerini çıkarttığı görülmüştür. Beyin bu çıkardığı basit özellikleri işlemeye kare üçgen dikdörtgen gibi belirli şekilleri çıkartarak devam etmektedir. Ardından resimde daha üst seviyeye ait karmaşık özelliklerle çıkartılarak daha önceden çıkartılan özelliklerle birleştirilerek görselin tamamını algıladığı görülmüştür. Bu çalışmalar sayesinde yapay sinir ağlarında evrişim katmanı kullanılmaya başlanmıştır. Evrişimsel yapay sinir ağları da evrişim katmanı ile öncelikle ağa giren görselin daha basit seviyede özellikler çıkartılmaktadır. Derinleşen katmanlarda bulunan evrişim katmanları kullanılarak daha üst seviyeye ait karmaşık özellikler çıkartılarak öğrenme sağlanmaktadır (Goodfellow vd. 2016).
Görüntü işleme çalışmalarında evrişimsel sinir ağları sınıflandırma için tercih edilmektedir (Tan 2019).

3.2.2.1 Giriş katmanı

Giriş katmanı, evrişimsel sinir ağının ilk katmanını oluşturmaktadır. Verilen verinin boyutu bellek kullanımı ve modelin başarısı için önemli bir kriterdir. Boyutun çok

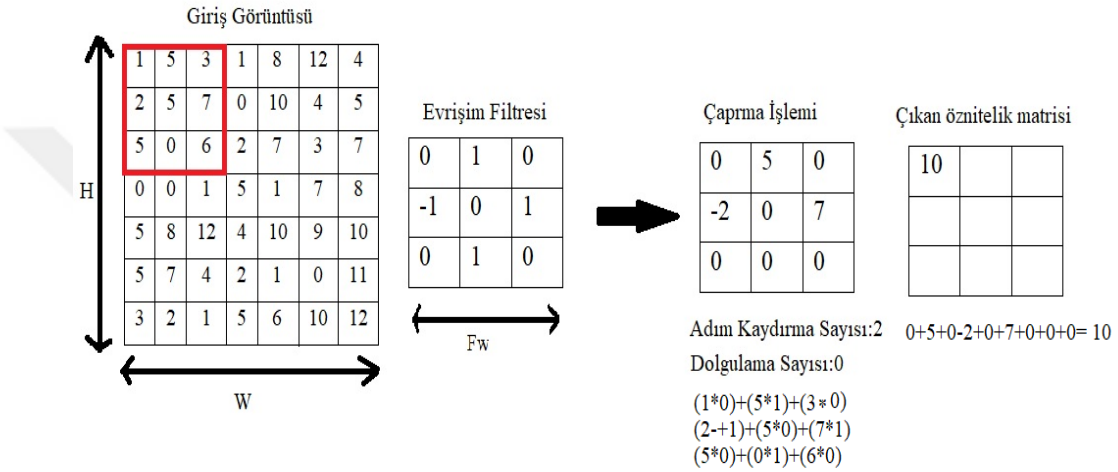
yüksek seçilmesinin, eğitim ve test süreçlerinde süre ve bellek ihtiyacının artması gibi olumsuz etkisi olmaktadır. Her ne kadar boyutun çok küçük seçilmesi, eğitim ve test süresi ile bellek ihtiyacının azalması açısından olumlu bir etki olarak gözükmese de, modelin performans düşmesi nedeniyle tercih edilmemektedir (İnik vd. 2017; Tan 2019)

Tez çalışması kapsamında tasarlanan özgün YZNet giriş katmanına girecek resimlerin büyüklükleri 15x15 ve 250x250 piksel arasında değişiklik göstermektedir. Yapılan deneysel işlem sonucunda 15x15 büyüklüğüne sahip giriş katman verisi çok küçük olmasından dolayı ağ resimlerden verimli özellik çıkartamadığı için öğrenme işlemini olumsuz yönde etkilediği tespit edilmiştir. 250x250 piksel giriş verisi için, küçük boyuta sahip resimler büyütülmesi resmin özelliklerini bozmuş ve öğrenmeyi olumsuz etkilemiştir. Bu nedenle deneme yanılma yöntemiyle tespit edilen 32x32 boyutunda giriş katmanı verisi hem eğitim süresi hem de öğrenme performansı açısında tasarladığımız YZNet için ideal olduğu gözlenmiştir. YZNet giriş katmanında kullanılan 15x15 ve 250x250 piksel boyutudaki tüm resimlerin boyutu yeniden boyutlandırma ile 32x32 yapılarak verilmiştir.

3.2.2.2 Evrişim katmanı

Evrişimsel sinir ağlarının en önemli katmanlarından biri evrişim katmanıdır. Bu katmanın kullanım amacı, girişi verisinin belirgin özelliklerini yansıtan daha üst seviyedeki gösterimleri öğrenmesini sağlamaktır. Özgün YZNet çok katmanlı yapay sinir ağında üç adet evrişim katmanı bulunmaktadır. Çok katmanlı yapay sinir ağında evrişim katmanı kullanılarak erken katmanlarda köşe, kenarlar, çizgi gibi özellikleri çıkartılır. Bu özellikler seviye ve hiyerarşik sıraya göre birleştirilerek, verilen görüntünün özellikleri çıkartılmaktadır (Bengio vd. 2017). Özellik çıkartımı deneysel çalışma sonucunda belirlenen 5x5 boyutundaki filtrelerin tüm görüntünün üzerinde dolaştırılması suretiyle evrişim işlemi uygulayarak yapılmaktadır (Liu vd. 2015). Görüntüye uygulanan evrişim işlemi şu şekilde gerçekleşmektedir. Filtre görüntünün en üst sol tarafından başlanarak adım kaydırma sayısı kadar kaydırılarak görüntünün değerleriyle filtre değerleri çarpılarak toplanır. Görüntünün üzerinde filtre kaydırılırken görüntünün dışına taşmamalıdır. Bu işlem resmin en üst sağ ucuna ulaşana kadar devam

etmektedir. Ardından filtre tekrar en soldan adım kaydırma sayısı kadar aşağı tarafa kaydırıldıktan sonra yine aynı şekilde filtre sola doğru kaydırılarak katlama işlemi yapılır. Görüntünün en alt sağ ucuna kadar bu işlem aynı şekilde tekrarlanmaktadır. Tez çalışmasında kullanılan resimler renkli olması sebebiyle filtre tüm renk kanalları için uygulanmıştır. Yapılan filtreleme işlemi sonucunda görüntünün özellik haritası çıkmaktadır (Ovtcharov 2015). Şekil 3.10'da görüntü üzerinde evrişim katmanın filtreleme değer hesaplanması örneği verilmiştir.



Şekil 3.10 Görüntü üzerinde evrişim katmanın filtreleme değer hesaplanması

Filtreleme sonucunda çıkan görüntünün boyutu aşağıdaki verilen denklemlere (Denklem 3.7 ve Denklem 3.8) göre hesaplanmaktadır.

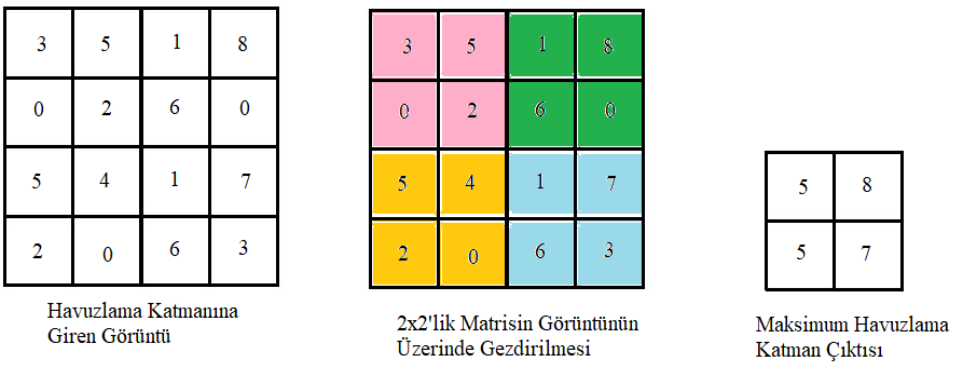
$$Çıktı\ genişliği\ = \frac{W - F_w + 2P}{S_w} + 1 \tag{3.7}$$

$$Çıktı\ yüksekliği\ = \frac{H - F_h + 2P}{S_h} + 1 \tag{3.8}$$

3.2.2.3 Havuzlama katmanı

Evrişimsel sinir ağlarında havuzlama katmanı veri boyutunu küçülterek hesaplamayı hızlandırmakta ve özelliklerin daha iyi tespit edilmesini sağlamaktadır (Boureau vd. 2010).

Havuzlama işlemi, giren veriler 2x2,3x3…NxN boyutunda belirlenen matrisin görüntü üzerinde gezdirilmesiyle yapılır. Bu yöntem maksimum havuzlama ve ortalama havuzlama olmak üzere iki çeşittir. Maksimum havuzlama yönteminde, en yüksek değerler hesaba katıldığı için o bölgeye ait dominant özellikler daha iyi tespit edildiğinden yüksek başarıya sahip bir yöntemdir (Şahin 2017). Bu neden ile tasarladığımız YZNet ağında maksimum havuzlama tercih edilmiştir. Bu katmana gelen verinin üzerinde 2x2 boyutunda matris gezdirilerek tespit edilen maksimum değerler alınarak daha baskın özellikler ortaya çıkartılmıştır. Şekil 3.11'de maksimum havuzlama yöntemi gösterilmiştir.



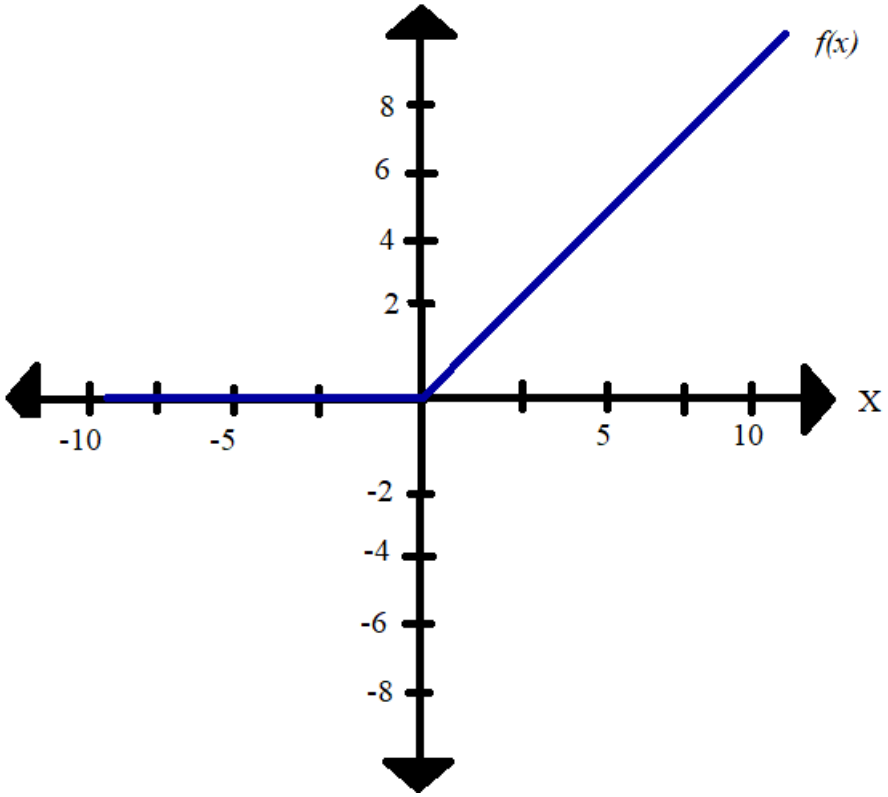
Şekil 3.11 Maksimum havuzlama yöntemi

3.2.2.4 Düzleştirilmiş doğrusal birim katmanı (ReLU Katmanı)

ReLU aktivasyon fonksiyonun gerçekleştirildiği katmana verilen isimdir (Yang vd. 2015). Negatif değerleri sıfır Kabul etmesinden dolayı ağın öğrenimi daha hızlı gerçekleşmektedir. Düzleştirilmiş doğrusal birim katmanı, derin sinir ağlarında sıklıkla kullanılan aktivasyon katmanlarından biridir (Maas vd. 2013). Özgün YZNet'inde iki adet ReLU katmanı bulunmaktadır.

Şekilde doğrusal olmayan birim katmanı formülü (Denklem 3.9) ve fonksiyon grafiği (Şekil 3.12) verişmiştir.

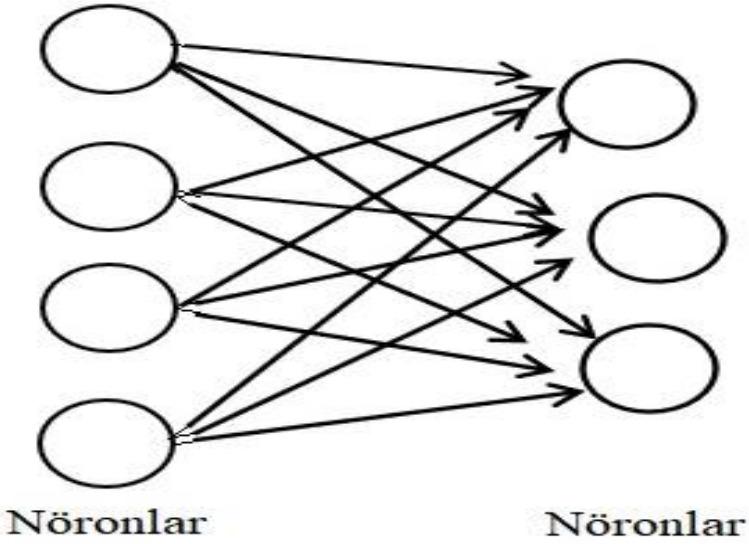
$$f(x) = \begin{cases} 0 \; eğer \; 0 > x \\ x \; eğer \; x \geq 0 \end{cases} \tag{3.9}$$



Şekil 3.12 Düzleştirilmiş doğrusal birim katmanının çıkış verisine etkisi (He 2015)

3.2.2.5 Tam bağlantı katmanı

Tam bağlantı katmanı sınıflandırmanın yapılmaya başlandığı katmandır ve sinir ağlarının son katmanlarında yer almaktadır (Ciresan 2011). Özgün YZNet ağında 2 adet tam bağlantı katmanı bulunmaktadır. Bu katmanın kendinden bir önceki katmanda bulunan nöronlara birebir bağlantısı bulunmaktadır (Adler vd. 2016). Tam bağlantı katmana girdi olarak alınan veriler, sınıflandırma için puanları hesaplanarak tek boyutlu matris elde edilmektedir. Bu tek boyutlu matris, yüksek seviyeli özellikleri taşıyan bir vektördür (Lin vd. 2013). Şekil 3.13'te tam bağlantı katmanı ve nöron bağlantısı gösterilmiştir.



Şekil 3.13 Tam bağlı katmanda nöronların birbiri ile bağlantısı

3.2.2.6 Entropi (Softmax) katmanı

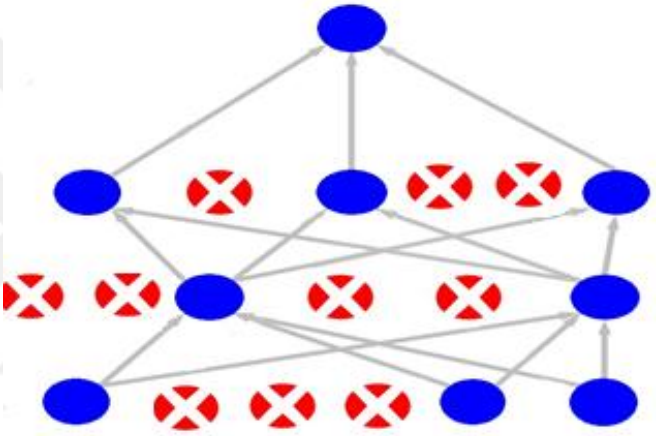
Entropi katmanında, derin öğrenme ağında sınıflandırma yapılması için değerler üretilmektedir. Bu katmanda girdi olarak gelen değerler probalistik hesaplama yapılarak her sınıf için olasılıksal değeri belirlenmektedir. Olasılıksal hesaplamada, bir önceki katmandan elde edilen verilere çapraz (cross) entropi uygulanmaktadır. Belirlenen bu değerler, elde edilen verilen hangi sınıfa daha yakın olduğunu göstermektedir. Bu sayede sınıflandırma entropi katmanında yapılmaktadır (Tang 2013).

3.2.2.7 Normalizasyon katmanı

Derin yapay sinir ağları katmana girdi olarak çok büyük veya çok küçük değer alabilmektedir. Bu değerlerin normalize edilerek belirli bir aralıkta temsil edilmesi gereklidir. Normalizasyon katmanı kullanılarak verilerin belirli bir aralıkta ölçeklendirilmesi ile girdi verileri ve ağ performansının belirli bir aralıkta temsil edilmesi sağlanmaktadır (Ioffe vd. 2015).

3.2.2.8 Seyreltme (Droput) katmanı

Ağın bazı durumlarda ezberlemeye gitmesi durumlarında, bu durumu engellemek amacıyla seyreltme (droput) katmanı kullanılmaktadır. Bu katman öğrenmede kullanılan bazı bağlantıların atılmasını sağlar. Böylelikle ağ ezber yapamaz ve ezberden dolayı oluşacak performans düşüklüğü ortadan kalkarak ağın performansı arttırılır (Xiao 2016). YZNet tasarımı esnasında deneme yanılma yöntemiyle yapılan deneylerde %50 oranında seyreltme ile ağ ezberlemesinin önüne geçilmiştir. Şekil 3.14'te seyreltme katmanı gösterilmiştir.

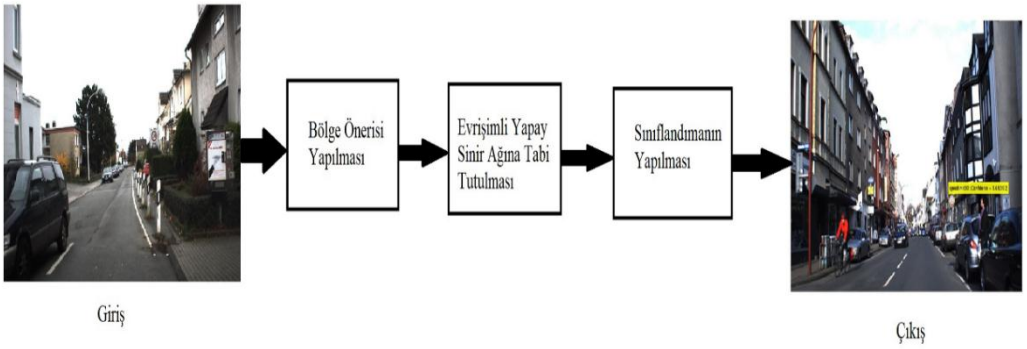


Şekil 3.14 Seyretme (Droput) katmanı

3.2.3 Bölgesel evrişimli sinir ağı (R-CNN)

Algılama yapılırken sınıflandırmadan farklı olarak nesnenin yerinin tespit edilmesi gerekmektedir. R-CNN nesne algılama için kullanılan evrişimsel sinir sinir ağı metodudur. Tez çalışması kapsamında tanıma işlemi için bölgesel evrişimli sinir ağı metodu kullanılmıştır.

Bölgesel evrişimli sinir ağı metodunda giren resime bölge önerileri yapılmaktadır. Ardından bölge önerileri yapılanmış resimlere CNN ile sınıflandırma yapılmaktadır. Sınıflandırılan resim, üzerinde sınırlayıcı kutu ile işaretlenmiş ve etiketlenmiş olarak çıkartılmaktadır. Şekil 3.15'te R-CNN metot yapısı görselleştirilmiştir.



Şekil 3.15 R-CNN metot yapısı

R-CNN yönteminde 200 tane bölge önerisi yapılmaktadır. Bölge önerisi yapılırken yapay sinir ağlarıyla yapılmayıp seçici arama algoritması kullanılarak yapılmaktadır.

Seçici arama algoritmasının başarısı üç faktöre dayanmaktadır. Bu faktörler de tüm ölçekleri yakalama, çeşitlendirme ve hızlı hesaplama faktörlerine bağlıdır. Bu faktörler aşağıda açıklanmıştır.

Tüm Ölçekleri Yakalama: Nesneler görüntüde herhangi bir boyutta olabilmektedir. Bazı durumlarda nesnelerin diğer nesnelere göre daha az sınırları bulunmaktadır. Bu nedenle, tüm nesnelerin ölçeklerini dikkate alan bir algoritma kullanılması gerekmektedir (Uijlings 2013).

Çeşitlendirme: Bölgeler bir araya getirildiğinde tek bir tane optimal strateji bulunmamaktadır. Bölgeler renklerine, dokularına veya parçalı kapalı oluşlarına göre oluşturulduğu için bunların her biri nesne adayını temsil etmektedir. Bunlara ek olarak, aydınlatmada kullanılan ışığın rengi ve gölgeler gibi aydınlatma ile ilgili koşullar bir nesnenin sınır bölgesinin nasıl oluştuğunu etkilemektedir. Bu nedenle tek bir stratejiye göre değil tüm zorluklarla başa çıkabilecek stratejiye sahip bir algoritma kullanılması gerekmektedir (Uijlings 2013).

Hızlı Hesaplama: Seçici aramanın amacı, pratik bir şekilde nesne tespiti amacıyla kullanılacak bir dizi olası nesne konumunu vermektir. Bu setin oluşturulmasının makul

derecede hızlı olması gerekmektedir (Uijlings 2013).

Seçici arama algoritmasında geleneksel görüntü işleme yöntemi olan segmentasyon yöntemi kullanılmaktadır. Segmentasyon, bir görüntüyü aynı özelliklere sahip piksellerle parçalar halinde bir araya toplanarak görüntüyü belirli bölge gruplarına ayırmaktır. Segmentasyon ile resim anlamlı bölge gruplarına ayrılması sağlanır. Öncelikle resim tüm boyutlar için renk benzerliği, büyüklük benzerliği ve doku benzerliğine göre bölütleme (segmentasyon) yapılarak bölgeler oluşturulur. Bu bölgelerin gruplandırılması tek bir bölge haline gelene kadar devam etmektedir. Bu nedenle, öncelikle tüm komşu bölgeler üzerinde benzerlik aranmaktadır. En benzer iki bölge birlikte gruplandırılacaktır. Ortaya çıkan yeni bölgeler benzerlik oranları hesaplanarak gruplandırılmaya devam edecektir. Bu işlem, hiyerarşik olması sebebiyle tek bir bölge halini alana kadar devam etmektedir. Böylelikle, seçici arama algoritması ile nesnenin aranacağı bölgeler belirlenmiş olur. Şekil 3.16'da segmentasyon yapılarak gruplandırılması ve bölge önerileri gösterilmiştir



Şekil 3.16 Seçici arama algoritmasının çalışmasının görselleştirilmiş formu

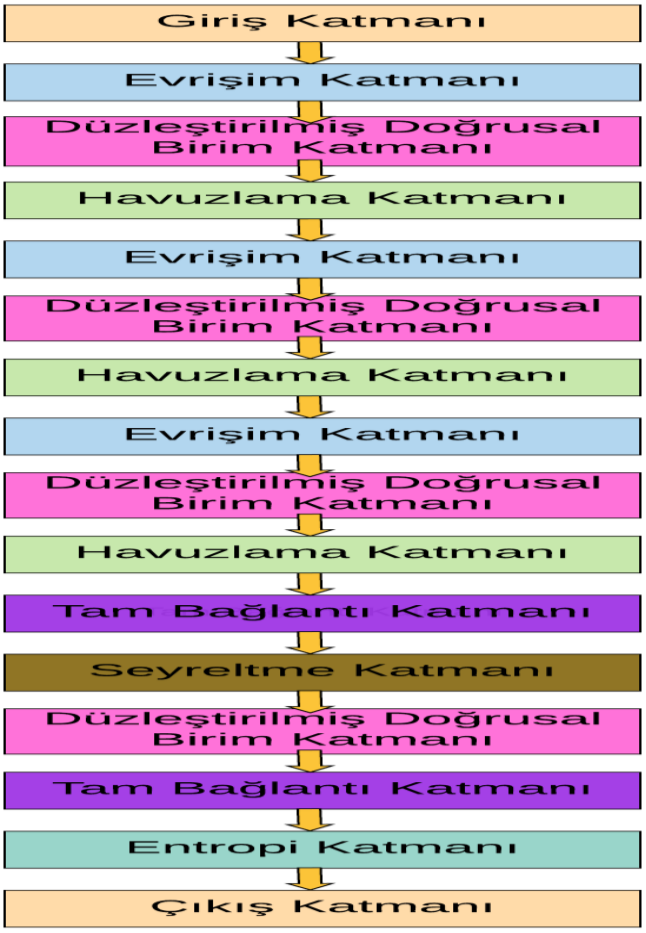
Ardından belirlenen bu bölgeler evrişimsel sinir ağı uygulanarak bu bölgelerin özellikleri çıkartılarak sınıflandırılması yapılmaktadır. R-CN, CNN yöntemi ile bölge önerisinin birbiriyle kombinlenmesidir. Önerilen bölgeler ile nesnenin resimde

olabileceği bölgeler çıkartılmakta ve evrişimsel sinir ağı ile de aranan nesnenin ne olduğuna karar verilmektedir.

3.2.4 YZNet yapısı ve sınıflandırma için eğitimi

Tez çalışmasında sınıflandırma problemi derin öğrenme yaklaşımı kullanarak çözülmüştür. Bu kapsamda görüntü işlemeye yönelik sınıflandırmayı başarıyla yapan evrişimli sinir ağı yapısında deneysel olarak YZNet adını verdiğimiz ağ özgün olarak tasarlanmıştır.

YZNet giriş ve çıkış dahil 16 katmandan oluşmaktadır. Bu katmanlar bir adet girişi, üç adet evrişim, üç adet havuzlama, iki adet tam bağlı, bir adet çıkış katmanından meydana gelmektedir. Şekil 3.17'de YZNet ağ yapısı gösterilmiştir.



Şekil 3.17 YZNet yapısı

Alman Trafik İşareti Tanıma Veri Kümesi (The German Traffic Sign Recognition Benchmark) veri setinde eğitim için 26640 adet resim bulunmaktadır. Bu resimler YZNet'in eğitimi için kullanılmıştır. Boyutları 15x15 ve 250x250 piksel değerinde değişen renkli resmilerin boyutu 32x32 boyutuna getirilerek YZNet giriş katmanına verilmiştir.

Evrişim katmanında özellik haritası çıkarılması sebebiyle, bu katman evrişimsel sinir ağlarının temelini oluşturur. Bu katmanda resimlere ait özellik haritası, filtrenin resim üzerinde katlama yapılarak gezdirilmesiyle elde edilir. YZNet ağında 5x5 boyutunda filtreye sahip 3 adet evrişim katmanı bulunmaktadır. İlk evrişim katmanında daha basit seviyede özellikler çıkarılırken, ilerleyen katmanlardaki evrişim katmanlarıyla giderek daha karmaşık özellikler çıkarılmıştır. Her evrişim katmandaki katlamanın sonucunda giderek matrisin boyutu küçülmektedir. Eğitim sırasında giderek küçülen matrisler özelliklerin kaybolmasına yol açtığından dolayı istenen bir sonuç değildir. Bu istenilmeyen durum, havuzlama katmanında uygulanan sıfır dolgulama (zero-padding) yöntemi ile çözülmüştür. Sıfır dolgulama (zero-padding) yönteminde matrisin etrafına görüntü özelliklerine etki etmemek için sıfırlar yerleştirilir. Elde edilen matrise maksimum havuzlama uygulamasıyla, görüntüde baskın özellikler daha çok vurgulanarak ortaya çıkar. Havuzlama katmanı sonucunda matris küçülmesine rağmen sınıflandırma için kullanılacak olan özellikler daha net ortaya konulmaktadır. Aşağıda orta katman ve özellikleri sırasıyla verilmiştir.

Orta Katman;

- Evrişimli (Convolution) 32 tane filtre, filtre boyutu 5x5 filtre adım kayma [1 1] ve dolgulama [2 2 2 2]
- Düzgünleştirilmiş Doğrusal Birim (ReLu) Katmanı
- Havuzlama (Max Pooling) Katmanı 3x3 boyutunda havuzlama matrisi adımlama [2 2] ve dolgulama [0 0 0 0]
- Evrişimli (Convolution) Katmanı 32 tane filtre 5x5 filtre boyutu, adımlama [1 1] ve dolgulama [2 2 2 2]
- Düzgünleştirilmiş Doğrusal Birim (ReLu) Katmanı

- Havuzlama (Max Pooling) Katmanı 3x3 boyutunda havuzlama matrisi adımlama [2 2] ve dolgulama [0 0 0 0]
- Evrişimli (Convolution) Katmanı 64 tane filtre 5x5 filtre boyutu, adımlama [1 1] ve dolgulama [2 2 2 2]
- Düzgünleştirilmiş Doğrusal Birim (ReLu) Katmanı
- Havuzlama (Max Pooling) Katmanı 3x3 boyutunda havuzlama matrisi adımlama [2 2] ve dolgulama [0 0 0 0]

Evrişimsel sinir ağının eğitiminde ezberleme (over-fitting) istenmeyen bir sonuçtur ve performans düşmesine sebep olmaktadır. Son katmanlarda ağın ezber yapmaması için %50 oranında seyreltme yapılmıştır. Böylelikle rastgele olarak seçilen nöronların yarısı atılarak eğitim gerçekleştirilmiştir. Sınıflandırma için eğitilen evrişimli sinir ağının en sonuncu tam bağlantı katmanında yer alan nöron sayısı, yapılacak sınıflandırma sayısı kadar olması gerektiğinden ve çalışmamızda 43 adet trafik işareti için sınıflandırılma öğretildiğinden, 43 tanedir. Tam bağlantılı katmandan entropi katmanına gelen değerler olasılıksal hesaplama ile hangi sınıfa ait olduğu belirlenmiştir. YZNet ağına ait son katmanlar ve özellikleri aşağıda sırasıyla verilmiştir.

Son Katmanlar;

- Tam Bağlı (Fully Connected) Katmanı 64 adet nöron
- Seyreltme (Dropout) Katmanı %50 oranında seyreltme yapılmıştır.
- Düzgünleştirilmiş Doğrusal Birim (ReLu) Katmanı
- Tam Bağlı (Fully Connected) Katmanı 43 adet nöron bulunmaktadır.
- Entropi (Softmax) Katmanı
- Sınıflandırma Çıkış Katmanı

Eğitilen YZNet Alman Trafik İşareti Tanıma Veri Kümesi'nde yer alan renkli 12569 adet resim ile test edilerek başarısı ölçülmüştür. Test setindeki resimler ile eğitim setindeki resimler kopyanın önüne geçilmesi amacıyla birbirinden farklı resimlerdir.

3.2.5 YZNet, AlexNet ve Cifar10Net'in tanıma için eğitimi

Cifar10Net ve AlezNet literatürde yer alan, YZNet ise tez çalışması kapsamında özgün olarak tasarlanmış, sınıflandırma için eğitilmiş ağlardır.

Tez çalışmasında tanıma için yapılan eğitime Alman Trafik İşareti Bulma Veri Kümesi (The German Traffic Sign Detection Benchmark) veri seti kullanılmıştır. Bu veri setinde 600 tane renkli resim eğitim için, 300 tane renkli resim test için kullanılmıştır. Eğitim seti ve test setinde yer alan resimler ağın kopya çekmesini önlemek amacıyla birbirlerinden tamamen farklıdır. Eğitim setinde 43 tane sınıfa ait trafik işareti her görüntüde bir adet veya birden fazladır.

Trafik İşareti Veri Kümesi resmilerinde sadece trafik işareti olmayıp, çeşitli farklı ögeler de bulunmaktadır. Trafik işaretinin resmin bütünü taranarak bulmak yerine Bölgesel Evrişim Sinir Ağı (R-CNN) tekniği ile bölge önerileri oluşturulmuştur. Oluşturulan bu bölgelere YZNet katman yapıları aynen kullanılarak Evrişimsel Sinir Ağı ile eğitim yapılmıştır. Literatürde yer alan ve sınıflandırma için eğitilmiş olan AlexNet sinir ağının son tam bağlantı katmanın nöron sayısı 1000, Cifar10Net'in ise 10'dur. AlexNet'de nöron sayısının 1000 olmasının sebebi, 1000 tane farklı sınıf için eğitilmiş olmasıdır. Cifar10Net'in de 10 tane nörona sahip olmasının sebebi yine 10 tane farklı sınıf için eğitilmiş olmasıdır. Tez çalışmasında 43 tane farklı sınıf için eğitim yaptığımızdan dolayı bu ağların diğer katmanlarına ve parametrelerini değiştirmeden sadece son katmanda bulunan tam bağlantı katmanındaki nöron sayıları 43 yapılarak Evrişimsel Sinir Ağı ile eğitime tabi tutulmuştur. Eğitim sonucunda önerilen bölgelerde trafik işareti varlığı ile bu işaretlerin hangi sınıfa ait olduğu öğrenilerek tanıma (algılama) yapılmıştır.

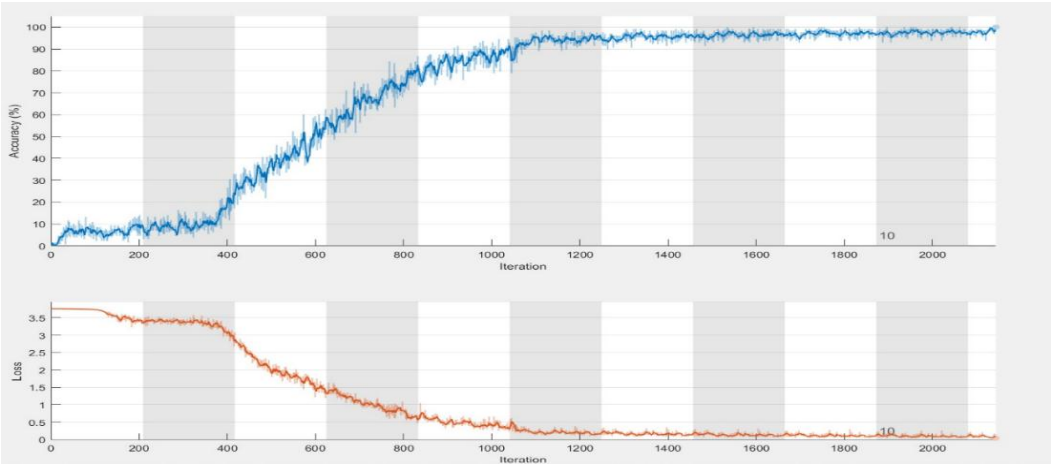
Eğitimin yapılmasının iki tane amacı vardır. Birincisi, AlexNet ve Cifar10Net'in, veri setinde yer alan 43 tane sınıfa ait trafik işaretleri için eğitilmemeleridir. Eğitim ile bu işaretler, sınıflandırma yapabilmesi amacıyla netlere öğretilmiştir. İkinci amaç, resimlerde trafik işareti haricinde ağaç, bina vb. ögelerin bulunmasından dolayı AlexNet, Cifar10Net ve YZNet'in R-CNN ile nasıl tanıma yapması gerektiği

öğretilmesidir. Eğitilmiş olan bu ağlar eğitim setinde yer alan resimlerden farklı olan test veri setinde bulanan 300 tane renkli resim ile test edilerek performans sonuçlarına bakılmıştır.

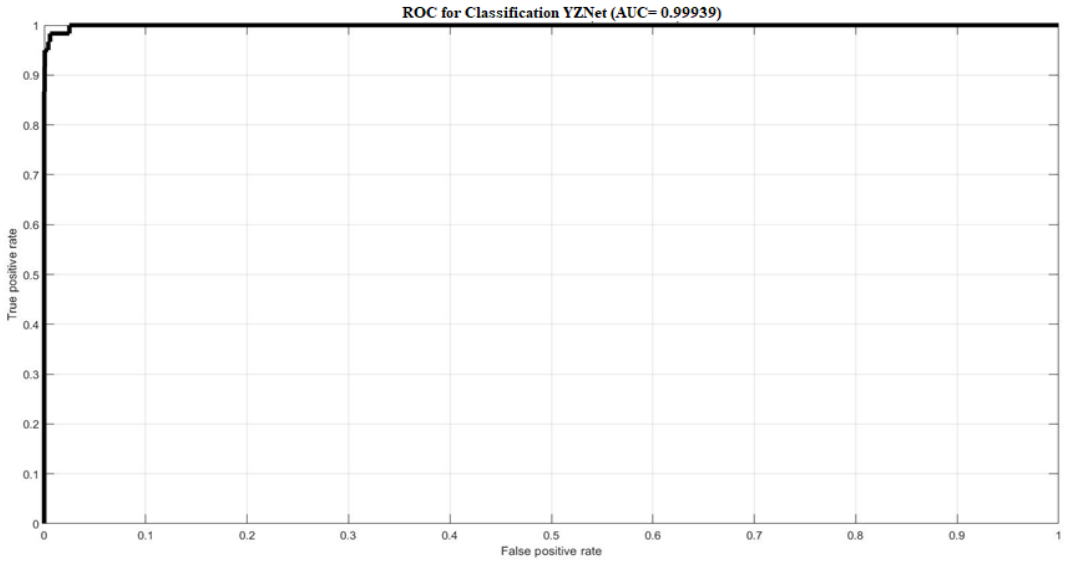
Bu işlemlerin sonucunda, tezin amaçları arasında yer alan trafik işaret veya işaretlerinin nerede olduğu ve hangi işaret olduğu resim üzerinde sınırlayıcı kutu ve etiket ile gösterilmiştir.

4. ARAŞTIRMA BULGULARI

Trafik işaretlerinin tanınması amacıyla yapılan çalışma, iki kısımdan oluşmaktadır. Birinci kısımda sınıflandırılma yapması için özgün YZNet tasarlanmıştır. Tasarlanan netin yapısı evrişimsel yapay sinir ağıdır. YZNet ağı eğitilirken Alman Trafik İşareti Tanıma Veri Kümesinde yer alan resimler veri seti olarak kullanılmıştır. Bu sette yer alan resimlerden 26640 adeti eğitim setinde, 12569 adet resim ise test setinde yer almaktadır. Eğitim ve test setlerinde 43 tane sınıfa ait trafik işareti bulunmaktadır. Deneysel olarak tasarlanan özgün YZNet 26640 adet resim kullanılarak sınıflandırma için CNN ile denetimli olarak eğitilmiştir. Eğitim sırasında YZNet'in giriş katmanına resimler boyutu 15x15 yapılarak verildiğinde resimden çıkartılan özelliklerin başarılı bir öğrenme için yeterli olmadığı ve yeterli özelliklerin çıkarılamadığı gözlemlenmiştir. Giriş katmanına verilen resimlerin boyutu 250x250 olarak verildiğinde 15x15 boyutundaki küçük resimlerin yapısı bozulduğu için çıkartılan özellikler başarılı bir öğrenme için yeterli olmadığına karar verilmiştir. Bu nedenle resimler 32x32 olarak boyutlandırılarak giriş katmanına verilmiştir. Eğitilen YZNet'in 12569 adet test verisi ile başarısı ölçülmüştür. YZNet ağı ile eğitim sonucunda %93 başarı elde edilmiştir. Eğitimdeki başarı ile yitim fonksiyonları arasında ters orantı olduğu görülmüştür. Şekilde 4.1'de YZNet eğitimi sırasında çıkan doğruluk ve yitim fonksiyonlarının grafiği gösterilmiştir. Şekil 4.2'de ise YZNet'in sınıflandırma için kullandığı ROC eğrisi gösterilmiştir.



Şekil 4.1 YZNet eğitimi grafikleri



Şekil 4.2 YZNet ROC eğrisi

YZNet ağı, evrişimsel sinir ağlarının evrişim katmanlarından çıkartılan özelliklerin öğrenilmesiyle sınıflandırma yapmıştır.

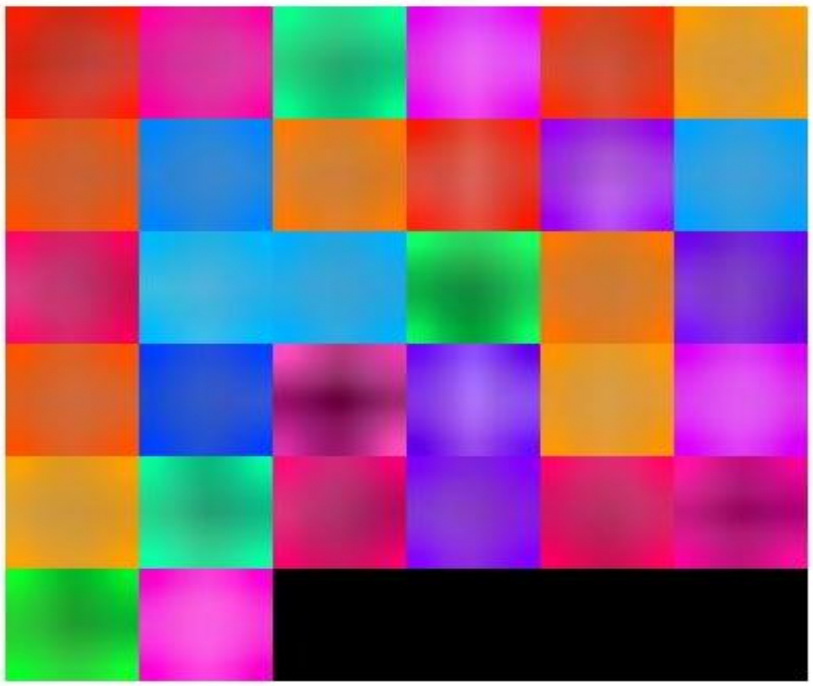
Cifar10Net, AlexNet daha önceden sınıflandırma için eğitilmiş ağlardır. YZNet ağıda tez kapsamında tasarlanarak eğitilmiştir. Böylelikle elimizde olan bütün ağlar sınıflandırma için önceden eğitilmiş ağlardır.

Tezde tanıma işlemi içerisinde sadece trafik işareti olan resimlerden değil çeşitli başka ögeleri de barındıran resimlerden yapılması planlanmıştır. Bu nedenle Alman Trafik İşareti Bulma Veri Kümesi veri setinden yararlanılmıştır. Bu resimler gündelik hayatta çekilen sokak, cadde ve otoban resimlerinden oluşmaktadır. Bu resimlerde ağaç, araba, trafik ışıkları vb. gibi harici ögelerde bulunmaktadır. Alman Trafik İşareti Bulma Veri Kümesi veri setinden 600 adet renkli resim eğitim setinde, 300 adet renkli resim test setinde yer almaktadır. Hem eğitim hem de test setinde 43 tane farklı sınıf içere trafik işareti bulunmaktadır.

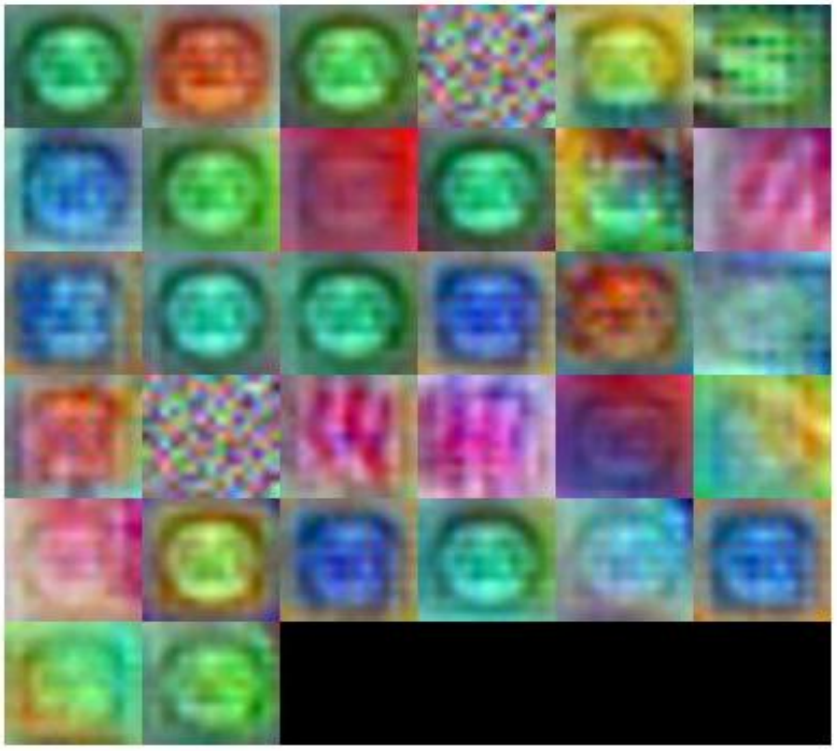
YZNet'e verilen Alman Trafik İşareti Bulma Veri Kümesi resimlerinde tanımanın yapılabilmesi için öncelikle resmin hangi bölge veya bölgelerinde trafik işaret veya işaretlerinin olacağı belirlenmesi problemi çözülmesi gerekmektedir. Bu alanlar

belirlenmeyerek resim bütünü her boyut için kayan pencere metodu ile tarayarak sınıflandırmak hem zaman kaybına hem de bellek kullanımına olumsuz etkisi olacağı için tercih edilmemiştir. Bu problem R-CNN tekniği ile çözülmüştür.

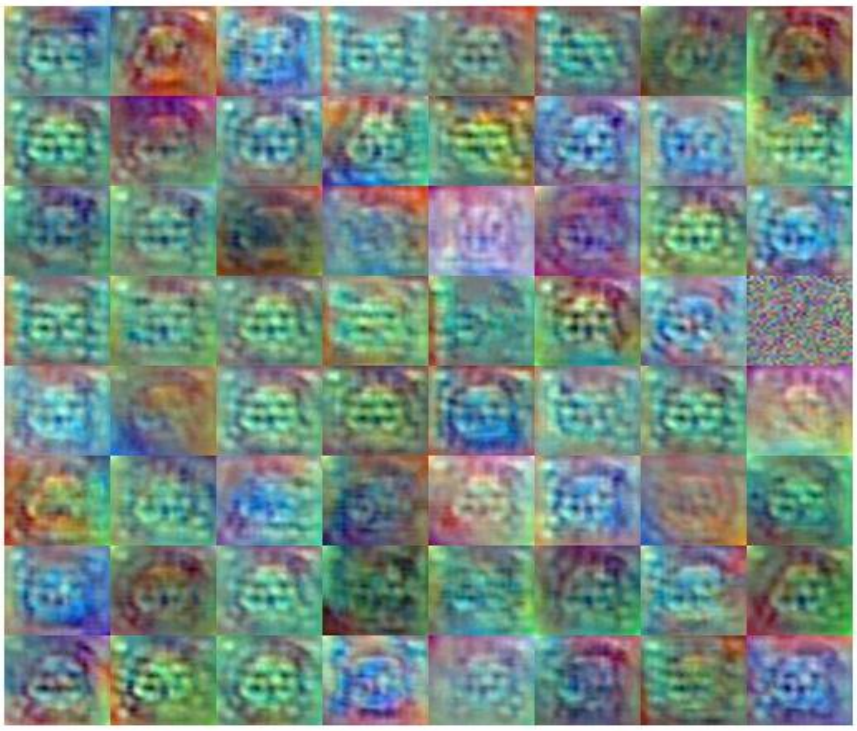
YZNet'e R-CNN ile verilen resimlerde hangi bölgelere bakarak sınıflandırma yapması gerektiği öğretilmiştir. R-CNN yöntemi CNN ile bölge önerilerinin kombinasyonu olmasından dolayı YZNet ağında bulunan evrişim katmanı ile özellikleri çıkartmaktadır. Şekil 4.3-4.5'de YZNet ağının evrişim katmanlarına göre çıkarttığı özellikler gösterilmiştir.



Şekil 4.3 YZNet ikinci katmandaki ilk evrişim katmanın çıkarttığı özellikler



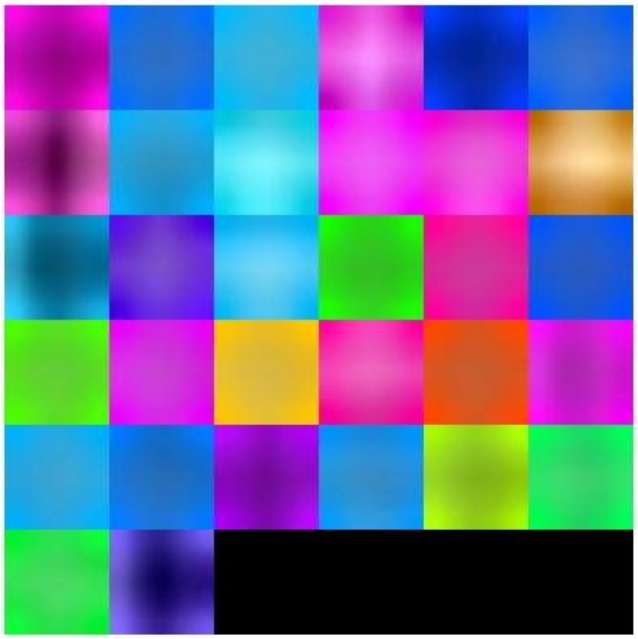
Şekil 4.4 YZNet beşinci katmandaki ikinci evrişim katmanın çıkarttığı özellikler



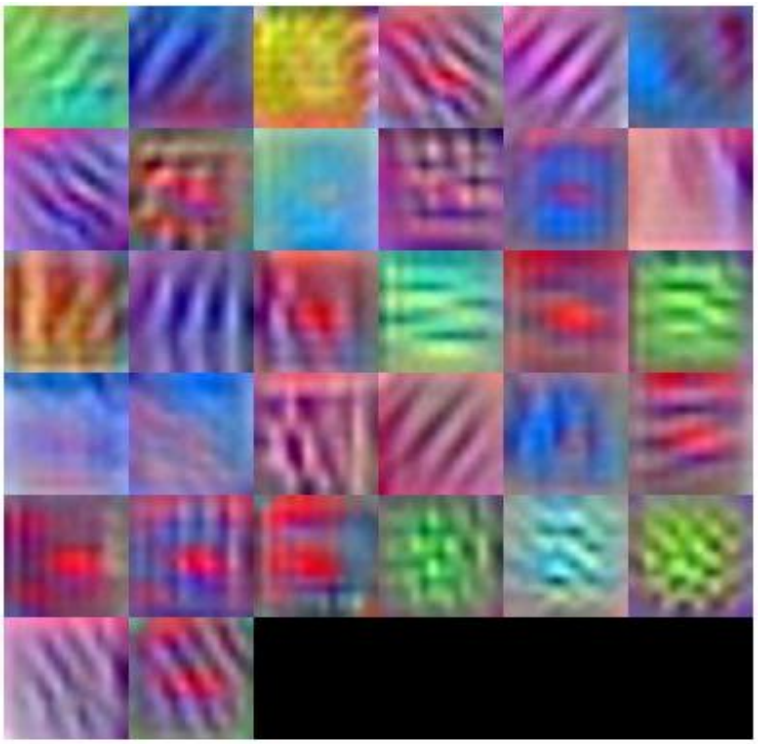
Şekil 4.5 YZNet sekizinci katmandaki üçüncü evrişim katmanın çıkarttığı özellikler

YZNet'in ilk evrişim katmanlarının sadece dikey ve yatay yönde özellikleri çıkardığı görülmektedir. İkinci ve üçüncü evrişim katmanında daha üst seviyede özellikler çıkartılmaktadır. Çalışma sonuçları çıkartılan özellikler arasında hız sınırı 50 ve hız sınırı 70 trafik işaretine ait özellikler seçilebilmekte olduğu gözlemlenmiştir. Evrişim katmanlarının artmasıyla çıkartılan özelliklerin daha yüksek seviyelere ait olduğu görülmüştür.

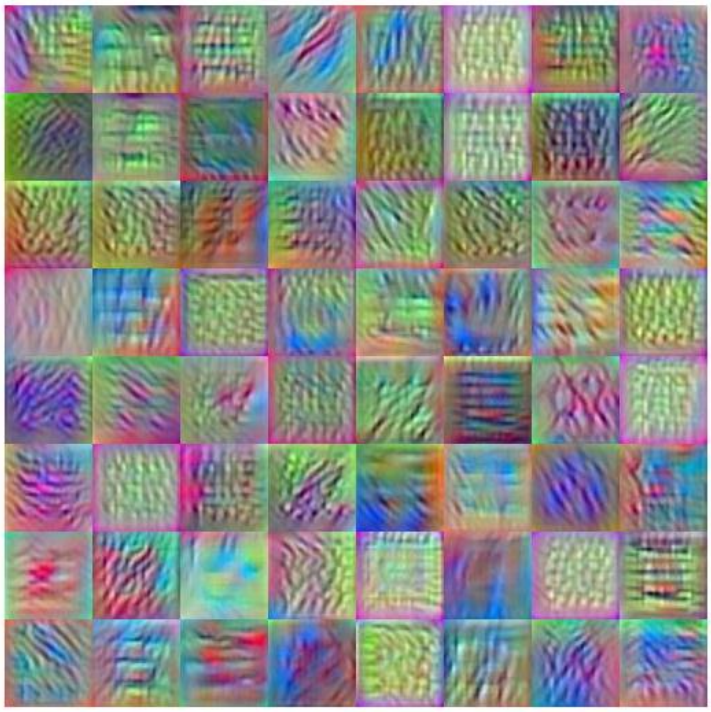
YZNet literatürde sınıflandırmada başarılı bulunan AlexNet ve Cifar10Net ile kıyaslanmıştır. AlexNet ve Cifar10Net ağları da R-CNN ile eğitime tabi tutulmuştur. R-CNN eğitimi ile hem Alman Trafik İşareti Bulma Veri Kümesinde yer alan trafik işaretleri hem de resimlerin hangi bölgesinde bu trafik işaretlerini araması gerektiği ağlara öğretilmiştir. Trafik işaretlerinin sınıflandırılması, bu ağların sahip olduğu evrişim katmanlarının çıkardığı özellikler sayesinde öğrenilmektedir. Şekil 4.6, 4.7 ve 4.8'de ise Cifar10Net ile elde edilen özellikler gösterilmiştir.



Şekil 4.6 Cifar10Net ikinci katmandaki ilk evrişim katmanın çıkarttığı özellikler

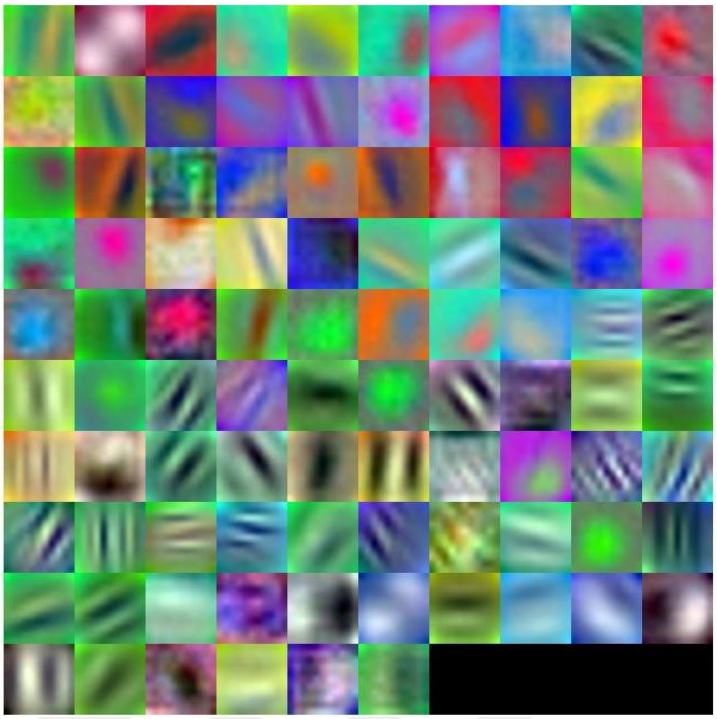


Şekil 4.7 Cifar10Net beşinci katmandaki ikinci evrişim katmanın çıkarttığı özellikler

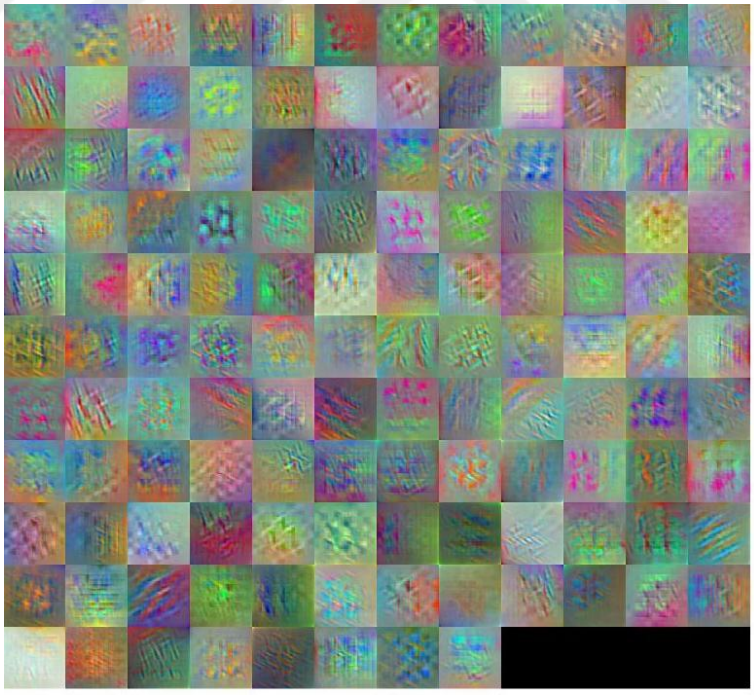


Şekil 4.8 Cifar10Net sekizinci katmandaki üçüncü evrişim katmanın çıkarttığı özellikler

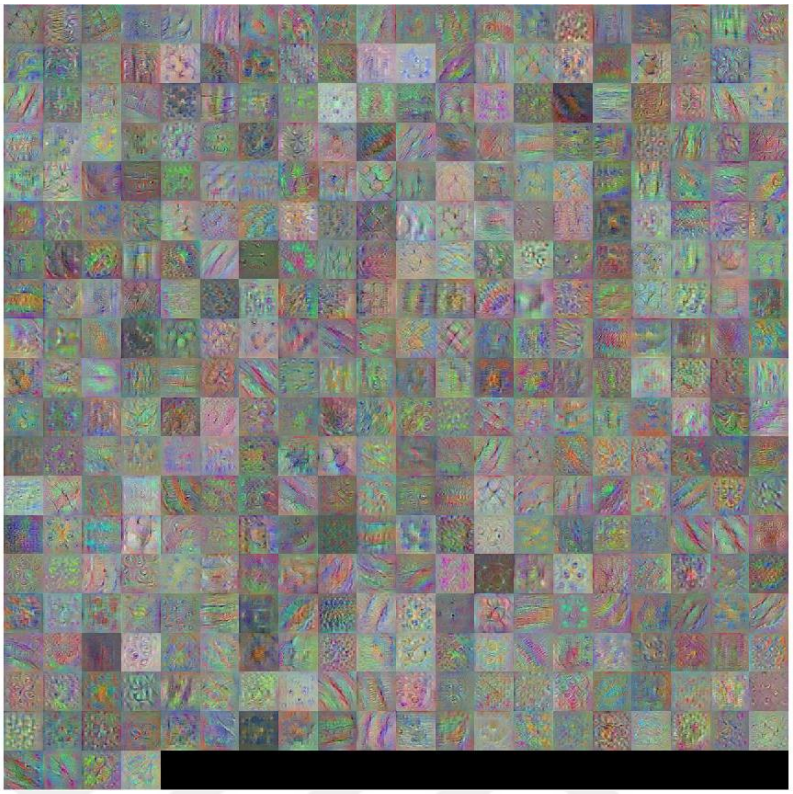
Şekil 4.9 - 4.13'de ise AlexNet ile elde edilen özellikler gösterilmiştir.



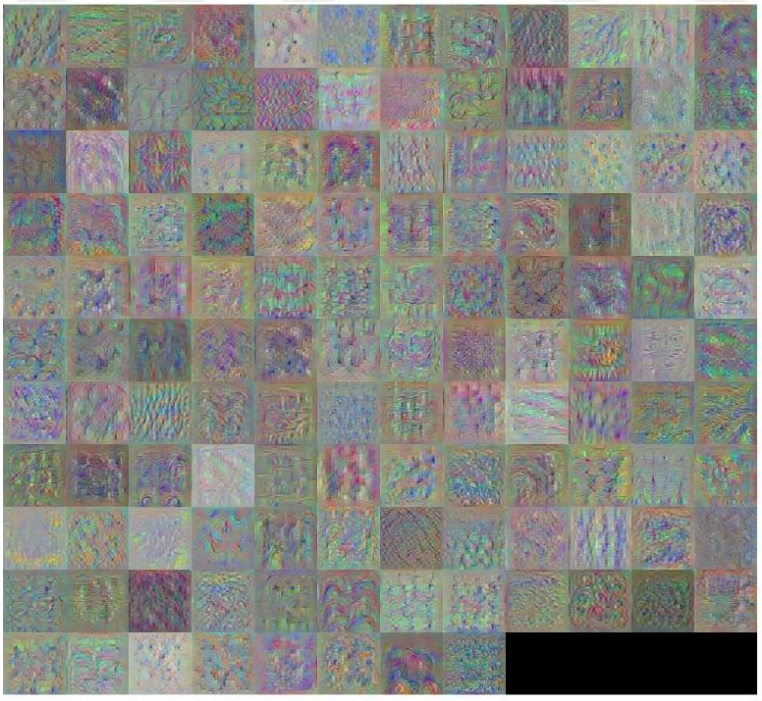
Şekil 4.9 AlexNet ikinci katmandaki ilk evrişim katmanın çıkarttığı özellikler



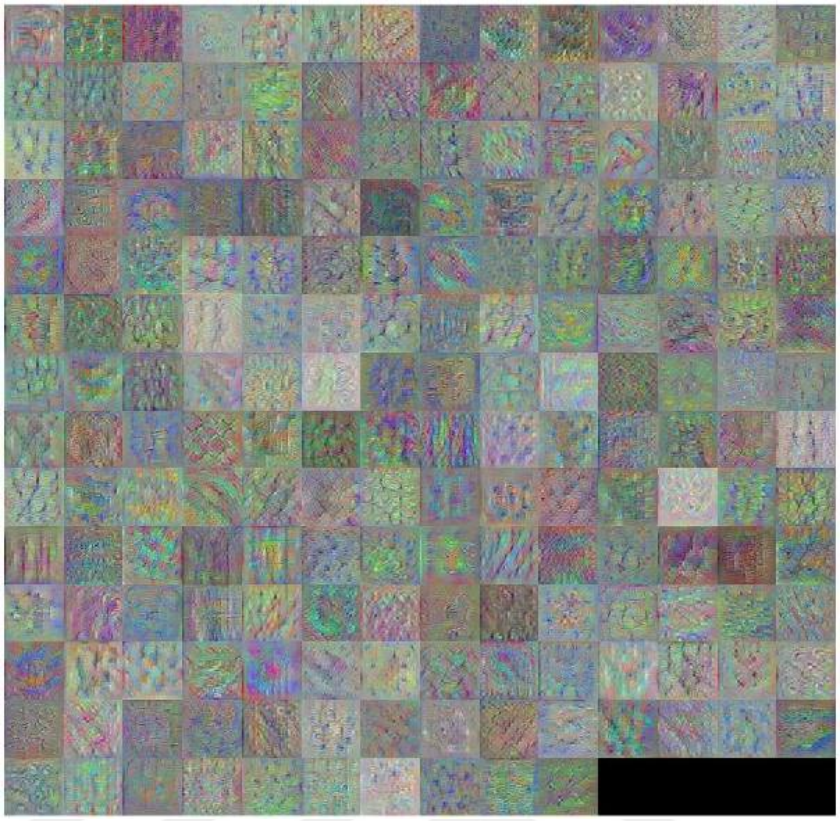
Şekil 4.10 AlexNet altıncı katmandaki ikinci evrişim katmanın çıkarttığı özellikler



Şekil 4.11 AlexNet onuncu katmandaki üçüncü evrişim katmanın çıkarttığı özellikler



Şekil 4.12 AlexNet onikinci katmandaki dördüncü evrişim katmanın çıkarttığı özellikler



Şekil 4.13 AlexNet ondördüncü katmandaki beşinci evrişim katmanın çıkarttığı özellikler

AlexNet'e ait ilk evrişim katmanında yatay ve dikey yönde birbirine doksan derecelik kesitlerin özelliklerinin yanı sıra, her açıya ait kesit özelliklerinin de çıkartıldığı görülmektedir. AlexNet ve CifarNet'in derinleşen evrişim katmanlarıyla çıkartılan özelliklerin karmaşılığı artmıştır. YZnet'in üçüncü ve en son evrişim katmanında çıkan seviyedeki özelliklerin, AlexNet'in daha ikinci evrişim katmanında elde edildiği görülmektedir. AlexNet'in tüm evrişim katmanları incelendiğinde, bu ağa ait elde edilen özelliklerin daha kompleks ve üst seviyelere ait olduğu gözlemlenmiştir.

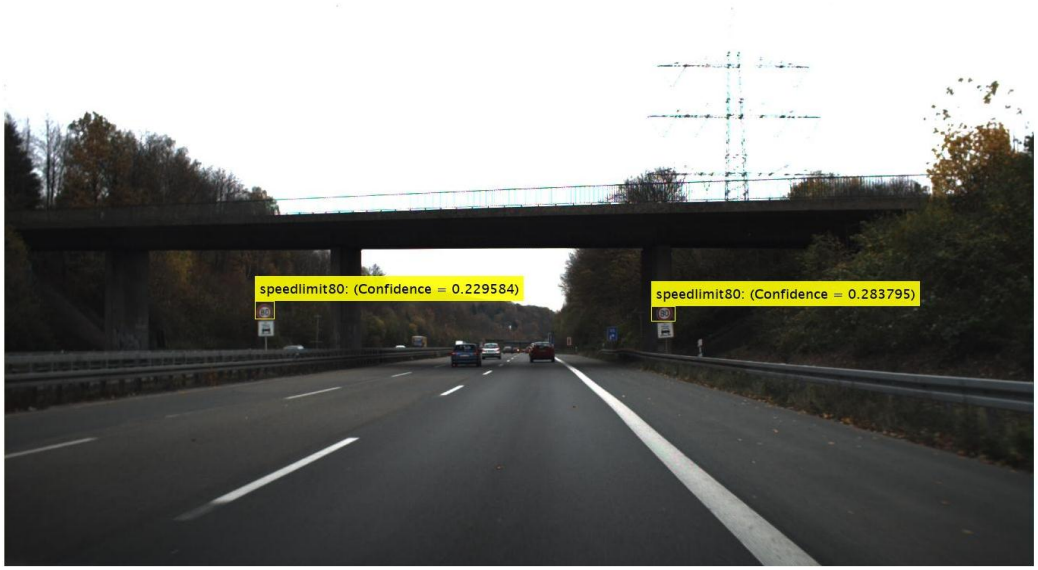
AlexNet'in, YZNet ve Cifar10Net'e göre daha karmaşık özellikler çıkartmasının nedeni, kullanılan evrişim katman sayısı ve eğitim için kullanılan veri sayısıdır. AlexNet'in eğitim için 1.2 milyon resim kullanılmıştır. Cifar10Net'in eğitimi için 50.000 adet resim kullanılmıştır. YZNet eğitimi için ise 26.640 adet resim kullanılmıştır. Ağlar çok miktarda veri ile daha fazla özellik öğrenmektedir. Böylelikle, evrişim katmanları sayesinde daha çok özellik çıkartmakta ve evrişim katman sayısı

arttıkça çıkartılan özniteliklerin seviyeleri yükselmektedir. Bu nedenle daha karmaşık özellikler görülmektedir.

Sınırlayıcı kutu ile resimlerde yer alan trafik işaretleri işaretlenmiştir. Şekil 4.14 – 4.38 arasında, YZNet, Cifar10Net ve AlexNet ağları ile test resimleri üzerinde trafik işaretlerinin sınırlayıcı kutu ile işaretlenerek ait olduğu sınıf etiketleriyle (test resimlerine verdiği cevap ve tahminleri) gösterilmiştir.



Şekil 4.14 YZNet'in dur işaretini tanıyarak işaretlemesi



Şekil 4.15 YZnet'in hız limiti 80 işaretini tanıyarak işaretlemesi



Şekil 4.16 YZNet'in hız limiti 50 işaretini tanıyarak işaretlemesi



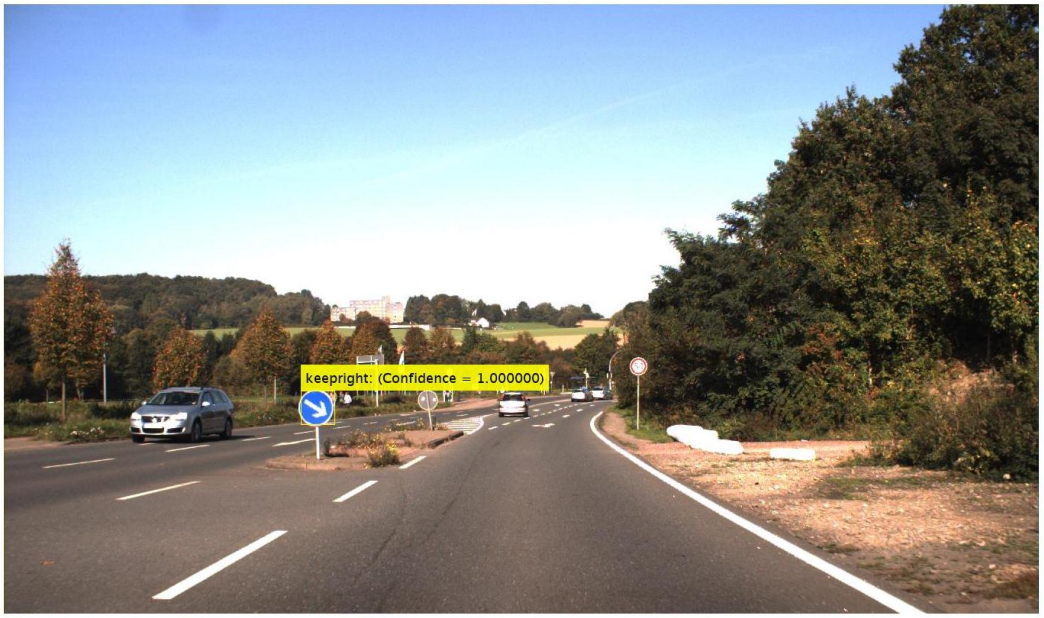
Şekil 4.17 YZet'in hız limiti 30 işaretini tanıyarak işaretlemesi



Şekil 4.18 YZnet'in öncelikli yol işaretini tanıyarak işaretlemesi



Şekil 4.19 YZet'in kamyonlar için öndeki taşıtı geçmek yasaktır işaretini tanıyarak işaretlemesi



Şekil 4.20 YZNet'in sağdan devam et işaretini tanıyarak işaretlemesi



Şekil 4.21 YZNet'in yol ver işaretini tanıyarak işaretlemesi



Şekil 4.22 Cifar10Net'in sağdan devam et ve dur işaretlerini tanıyarak işaretlemesi



Şekil 4.23 Cifar10Net'in hız limiti 80 işaretinin tanıyarak işaretlemesi



Şekil 4.24 Cifar10Net'in hız limiti 70 işaretinin tanıyarak işaretlemesi



Şekil 4.25 Cifar10Net'in hız limiti 50 işaretinin tanıyarak işaretlemesi



Şekil 4.26 Cifar10Net'in hız limiti 30 işaretinin tanıyarak işaretlemesi



Şekil 4.27 Cifar10Net'in öncelikli yol işaretini tanıyarak işaretlemesi



Şekil 4.28 Cifar10Net'in kamyonlar için öndeki taşıtı geçmek yasaktır işaretinin tanıyarak işaretlemesi



Şekil 4.29 Cifar10Net'in sağdan devam et işaretini tanıyarak işaretlemesi



Şekil 4.30 Cifar10Net'in yol ver işaretini tanıyarak işaretlemesi



Şekil 4.31 Cifar10Net'in tehlike işaretini tanıyarak işaretlemesi



Şekil 4.32 AlexNet'in dur işaretini tanıyarak işaretlemesi



Şekil 4.33 AlexNet'in hız limiti 80 işaretini tanıyarak işaretlemesi



Şekil 4.34 AlexNet'in hız limiti 50 işaretini tanıyarak işaretlemesi



Şekil 4.35 AlexNet'in hız limiti 50 işaretini tanıyarak işaretlemesi



Şekil 4.36 AlexNet'in hız limiti 30 işaretini tanıyarak işaretlemesi

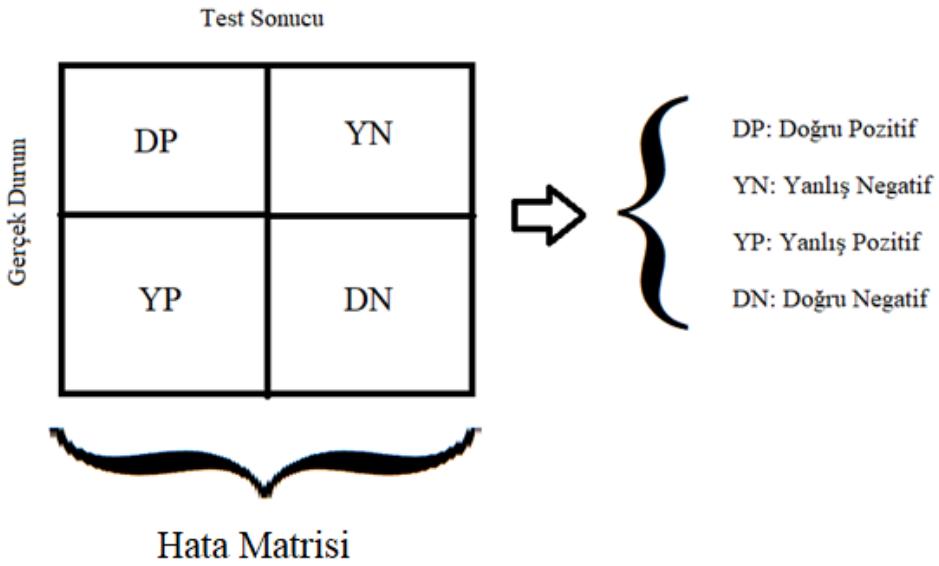


Şekil 4.37 AlexNet'in yol ver işaretini tanıyarak işaretlemesi



Şekil 4.38 AlexNet'in sağdan devam et işaretini tanıyarak işaretlemesi

Ağların eğitim performans değerlendirmeleri, 0.4, 0.5, 0.6 ve 0.7 olan üst üste binen bölge oranlarına göre rastgele olarak 10 tane belirlenen sınıf için aşağıda verilen parametreler aracılığıyla yapılmıştır.



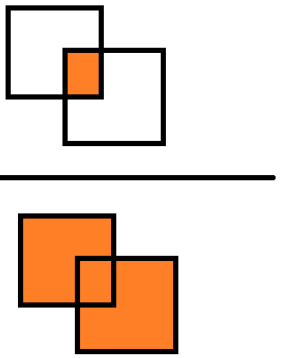
$$Duyarlılık = \frac{DP}{DP + YN}$$

$$Keskinlik = \frac{DP}{DP + YP}$$

$$Doğruluk = \frac{DP + DN}{DP + DN + YP + YN}$$

$$Yanlış\ Alarm\ Oranı = \frac{YP}{DP + YP}$$

Kesisme Oranı (IoU) = Üst Üste Binen Bölge / Tüm Bölge =

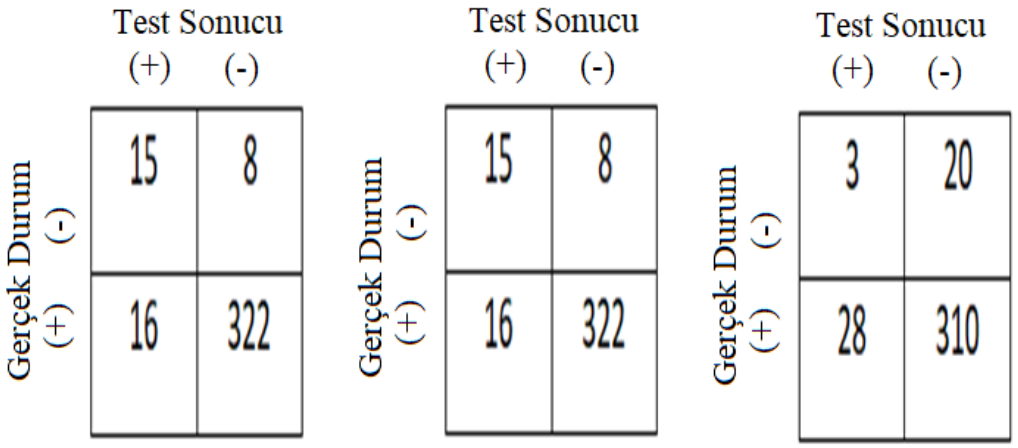


Değerlendirmenin yapıldığı 10 tane sınıf aşağıda verilmiştir.

- Hız sınırı 30 (speedlimit30)
- Hız sınırı 50 (speedlimit50)
- Hız sınırı 70 (speedlimit70)
- Hız sınırı 80 (speedlimit80)
- Yol ver (giveway)
- Sağdan devam et (keepright)
- Öncelikli yol (priorityroad)
- Kamyonlar için öndeki aracı geçmek yasaktır (noovertaking trucks)
- Dur (stop)
- Tehlike (danger)

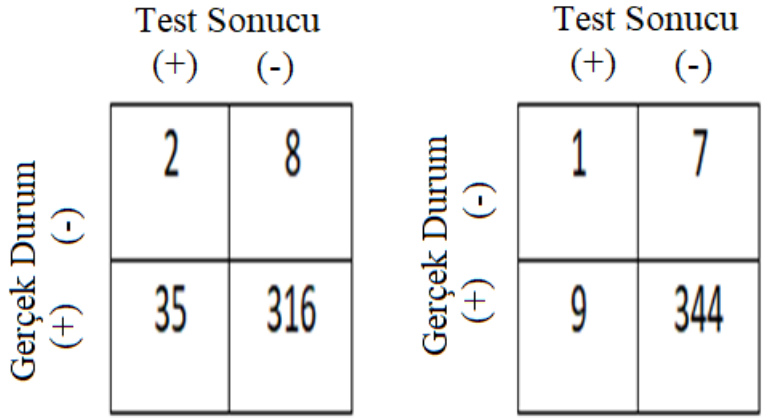
Ek 1'de AlexNet ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans gösterilmiştir. Ek 1'de de görüldüğü üzere, keskinlik açısından AlexNet ağında en yüksek değer hız sınırı 30 sınıfında üst üste kesişme oranı %40 iken 0.65 olarak

hesaplanmıştır. En düşük değer üst üste kesişme oranı %70 için hız sınırı 30 ve dur sınıflarında 0.13 olarak hesaplanmıştır. Duyarlılık açısından AlexNet ağında en yüksek değer hız sınırı 30 sınıfında üst üste kesişme oranı %40 ve %50 iken 0.48 olarak hesaplanmıştır. En düşük değer hız sınırı 70 sınıfında üst üste kesişme oranı %70 için ise 0.05 olarak hesaplanmıştır. Şekil 4.39'da AlexNet için hata matrisleri hız sınırı 30 da %40,%50 ve %70 için sırasıyla gösterilmiştir.



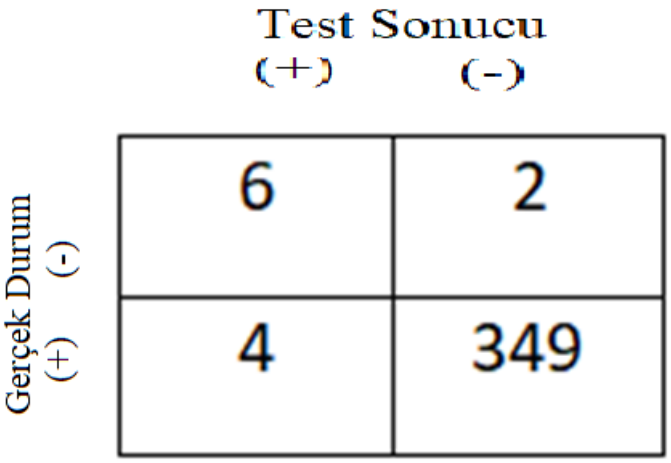
Şekil 4.39 AlexNet'in hız sınırı 30 işareti için %40 %50 ve %70 oranları için hata matrisi

Şekil 4.40'da AlexNet için hata matrisleri hız sınırı 70 ve dur işareti %70 için sırasıyla gösterilmiştir.



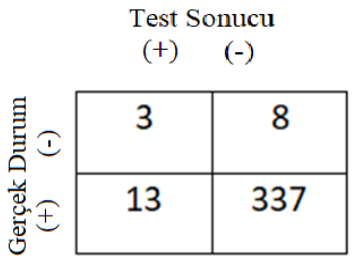
Şekil 4.40 AlexNet'in hız sınırı 70 ve dur işareti için %70 oranı için hata matris

EK 2.'de Cifar10Net ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans gösterilmiştir. Ek 2'de de görüldüğü üzere, keskinlik yönünden Cifar10Net ağında en yüksek değer dur sınıfında üst üste kesişme oranı %40 iken 0.75 olarak hesaplanmıştır. Duyarlılık açısından Cifat10Net ağında en yüksek değer dur sınıfında üst üste kesişme oranı %40 iken 0.60 olarak hesaplanmıştır. Şekil 4.41'da Cifar10Net için hata matrisi dur işareti %40 için gösterilmiştir.



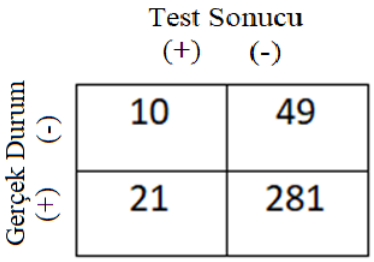
Şekil 4.41 Cifar10Net'in dur işareti için %40 oranı için hata matrisi

Ek 3'de YZNet ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performansı gösterilmiştir. Ek 3'de de görüldüğü üzere, keskinlik açısından YZNet ağında en yüksek değer hız sınırı 80 sınıfında üst üste kesişme oranı %40 iken 0.27 olarak hesaplanmıştır. Duyarlılık açısından YZNet ağında en yüksek değer sağdan devam et sınıfında üst üste kesişme oranı %40 ve %50 iken 0.32 olarak hesaplanmıştır. Şekil 4.42'de YZNet için hata matrisleri hız sınırı 80 işaretinde %40 için gösterilmiştir.



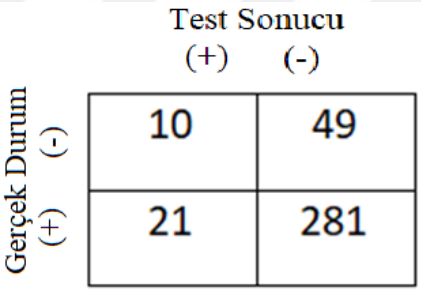
Şekil 4.42 YZNet'in hız sınırı 80 işareti için %40 oranı için hata matrisi

Şekil 4.43'de YZNet için hata matrisleri sağdan devam et işaretinde %40 için gösterilmiştir.



Şekil 4.43 YZNet'in sağdan devam et işareti için %40 oranı için hata matrisi

Şekil 4.44'de YZNet için hata matrisleri sağdan devam et işaretinde %50 için gösterilmiştir.



Şekil 4.44 YZNet'in sağdan devam et işareti için %50 oranı için hata matris

Çalışmamızda kullanılan YZNet ile, 43 sınıfa ait trafik işaretleri için eğitilmiş ve 10 tane sınıf için test sonuçları alınmıştır. Bu sınıflar hız sınırı 30, hız sınırı 50, hız sınırı 70, hız sınırı 80, yol ver, sağdan devam et, öncelikli yol, kamyonlar için öndeki aracı geçmek yasaktır, dur işaretlerine aittir. YZNet ağı R-CNN yönteminde, seçici arama algoritmasının bulduğu aday sınırlayıcı kutuların sınıflandırılmasında kullanılmıştır. Bu yöntem ile yol ver ve dur trafik işaretlerinde sırası ile elde edilen başarı oranları %75 ve %82,27 olarak bulunmuştur. Benzer olarak, Gürbüz (2010) tarafından yapılan çalışmada trafik işaretlerini kare, yuvarlak ve üçgen olarak sınıflara ayrılmıştır. Bu işaretleri resmin bütününden ayırt etmek için HSV renk uzayı, kırmızı eşikleme ve Hough Daire Dönüşümü (Hough Circle Transform) kullanarak seçilen resimlerdeki işaretler belirginleştirilmiştir. Belirginleştirilen işaretler yapay sinir ağı kullanarak sınıflandırmıştır. Yapılan testler sonucunda yol ver işaretinde %39, dur işaretinde ise

%52 oranında başarı elde edilmiştir. Çalışmamız sonuçları göstermiştir ki, R-CNN ile görüntülerden trafik işareti bulma işlemi herhangi bir ön çalışma gerektirmeden aynı kategoride bulunan işaretler için daha yüksek başarı sağlamıştır.

Çalışmamızda kullanılan özgün YZNet'in 3 tane evrişim katmanı bulunmaktadır. YZNet ağı tez çalışması kapsamında hazırlanan ağ Alman Trafik İşareti Tanıma Veri Kümesi (GTSRB) veri seti kullanılarak eğitilmiştir. YZNet ağının sınıflandırma başarı oranı %93.07 olarak bulunmuştur. Yıldırım (2019) yapmış olduğu derin öğrenme yöntemleri ile trafik işaretlerini tanıma çalışmasında aynı veri setini kullanılarak sınıflandırma yapmıştır. Sınıflandırma için kullanılan modelde 7 adet evrişimsel katman bulunmaktadır. Elde edilen sınıflandırma başarısı ise %99.1 olarak hesaplanmıştır. Çalışmamız sonuçları göstermiştir ki, evrişimsel katman sayısının artması ile başarı oranı doğru orantılı olarak artış göstermektedir.

5. SONUÇ

Bu çalışmada özgün olarak tasarlanan YZNet ağı ile, ağın giriş katmanına giren resimlerde trafik işaretlerini algılaması ve tanıması üzerine çalışılmıştır. Bu amaçla, 43 farklı sınıfa ait trafik işaretlerinin tanınması ve sınıflandırılması için derin öğrenme altyapısı kullanılmıştır. Performans değerlendirmesi yapılırken 43 tane sınıftan 10 tane sınıf performans değerlendirmesinin efektif olması açısından rastgele seçilmiştir. Çalışma kapsamında tasarlanan ve eğitilen YZNet ile daha önceden eğitilmiş olan ve literatürde yer alan AlexNet ve Cifar10Net yapılarıyla, aynı veri tabanları kullanılarak deneyler gerçekleştirilmiştir. Eğitilen ağ yapılarının sınıflandırma ve tanıma yapabilmesi için Evrişimsel Sinir Ağı (CNN) ve Bölgesel Evrişimli Sinir Ağı (R-CNN) yöntemleri kullanılmıştır. Eğitim setinde bulunmayan trafik işaretlerinden oluşan veri setiyle testler yapılarak, bu üç ağ yapısı için başarı sonuçları kıyaslanmıştır. Derin öğrenme yönteminin sınıflandırma ve işaret tespitinde kullanılabileceği belirlenmiştir.

AlexNet ve Cifar10Net eğitimlerinde daha fazla miktarda veri setinin eğitim için kullanılmasından dolayı bu ağlar daha çeşitli nitelikler öğrenmiştir. Böylelikle, bu ağların entropi katmanında hesaplanan hangi sınıflara ait olduğu olasılıksal değerler daha keskindir. Bunun sonucunda birbirine yakın değerlere sahip özellikler doğru sınıf olarak etiketlenir. Eğer entropi katmanında belirlenen olasılıksal değerler birbirinden uzak ise özelliğini en yakın gördüğü değere sınıf olarak etiketlemektedir. Eğer çıkartılan özellikler çok detaylı ve üst seviyede değilse bu hesaplama yeterince ayırt edici olarak yapılmamaktadır. Bu durum, genelleme yeteneğini düşürmektedir.

Sonuç olarak, tezde kullanılan tüm ağ yapıları için tespit edilen genel özellikler aşağıda sıralanmıştır;

- Veri setinde bulunan resimlerin miktarının çok olmasının öğrenmeyi olumlu etkilediği,
- Evrişim katmanının sayısının artması ile eğitimi için kullanılan veri setindeki resimlerden daha üst seviyeye ait karmaşık özelliklerin çıkartıldığı ve genelleme yeteneğini arttırdığı
- Üst üste kesişme oranı arttıkça ağların performanslarının düştüğü tespit edilmiştir.

KAYNAKLAR

Abraham, A 2005. Artificial neural networks. Handbook of measuring system design.

Adler, A., Elad, M., & Zibulevsky, M. 2016. Compressed learning: A deep neural network approach. arXiv preprint arXiv:1610.09615.

Akgün, G., Demetgül, M., Kaplanoğlu, E., & ABD, M. 2013. EMG Sinyallerinin Öznitelik Çıkarımı ve Geri Yayılımlı Yapay Sinir Ağı, Algoritması İle Sınıflandırılması. Otomatik Kontrol Ulusal Toplantısı, TOK2013, 26-28.

Aki, K. 2019. Derin öğrenme tabanlı sürücüsüz araç sistemleri. Yüksek Lisans Tezi. Bursa Uludağ Üniversitesi, Fen Bilimleri Enstitüsü, Bursa.

Angelov, P., Gegov, A., Jayne, C., & Shen, Q. 2016. Advances in Computational Intelligence Systems Intelligence.

Anonymous. 2013. Web Sitesi: http://benchmark.ini.rub.de/ Erişim Tarihi:09.01.2020.

Anonymous 2019. Web Sitesi: https://supervise.ly/explore/plugins/precision-and-recall-75278/overview Erişim Tarihi:09.01.2020.

Arı, A., Berberler, M. E. 2017. Yapay Sinir Ağları ile Tahmin ve Sınıflandırma Problemlerinin Çözümü İçin Arayüz Tasarımı. Acta INFOLOGICA, 1(2), 55-73.

Aydın, U. A., 2009, Traffic Sign Detection. Master Thesis. Middle East Techical University Natural and Applied Science, Ankara.

Bayram, F. 2020. Derin öğrenme tabanlı otomatik plaka tanıma. Politeknik Dergisi.

Bengio, Y., Goodfellow, I., & Courville, A. 2017. Deep learning (Vol. 1). MIT press.

Boureau, Y. L., Ponce, J., & LeCun, Y. 2010. A theoretical analysis of feature pooling in visual recognition. In Proceedings of the 27th international conference on machine learning (ICML-10),111-118.

Buchanan, B. G. 2005. A (very) brief history of artificial intelligence. Ai Magazine, 26(4), 53-53.

Chan, L. W., & Fallside, F. 1987. An adaptive training algorithm for back propagation networks. Computer speech & language, 2(3-4), 205-218.

Chang, H. C., Kopaska-Merkel, D. C., Chen, H. C. 2002. Identification of lithofacies using Kohonen self-organizing maps. Computers & Geosciences, 28(2), 223-229.

Ciresan, D. C., Meier, U., Masci, J., Gambardella, L. M., Schmidhuber, J. 2011. Flexible, high performance convolutional neural networks for image classification. In Twenty-Second International Joint Conference on Artificial Intelligence.

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., Fei-Fei, L. 2009. Imagenet: A large scale hierarchical image database. 2009 IEEE conference on computer vision and pattern recognition, 248-255.

Deng, L., & Yu, D. 2014. Deep learning: methods and applications. Foundations and

Trends® in Signal Processing, 7(3–4), 197-387.

Dollár, P., Zitnick, C. L. 2013. Structured forests for fast edge detection. In Proceedings of the IEEE international conference on computer vision, 1841-1848.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., Zisserman, A. 2010. The pascal visual object classes (voc) challenge. International journal of computer vision, 88(2), 303-338.

Francois, C. 2017. Deep learning with Python.

Fukushima, K. 1988. Neocognitron: A hierarchical neural network capable of visual pattern recognition. Neural networks, 1(2), 119-130.

Gerrits, H. J. M., De Haan, B., Vendrik, A. J. H. 1966. Experiments with retinal stabilized images. Relations between the observations and neural data. Vision research, 6(7-8), 427-440.

Girshick, R., Donahue, J., Darrell, T., Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of theIEEE conference on computer vision and pattern recognition,580 587.

Goodfellow, I., Bengio, Y., & Courville, A. 2016. Deep learning. MIT press.

Gürbüz, A. 2010. Otomatik trafik işareti tanıma. Yüksek Lisans Tezi. Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Eskişehir.

He, K., Zhang, X., Ren, S., Sun, J. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE International conference on computer vision, 1026-1034.

Hopfield, J. J. 1988. Artificial neural networks. IEEE Circuits and Devices Magazine, 4(5), 3-10.

Ioffe, S., Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprintarXiv:1502.03167.

Li, Z., Peng, C., Yu, G., Zhang, X., Deng, Y., & Sun, J. 2018. Detnet: Design backbone for object detection. In Proceedings of the European conference on computer vision (ECCV), 334-350.

Kalapanidas, E., Avouris, N., Craciun, M., & Neagu, D. 2003. Machine learning algorithms: a study on noise sensitivity. In Proc. 1st Balcan Conference in Informatics, 356-365.

Keskenler, M. F., Keskenler, E. F. 2017. Geçmişten Günümüze Yapay Sinir Ağları ve Tarihçesi. Takvim-i Vekayi, 5(2), 8-18.

Kızrak, A. 2018. Derin bir karşılaştırma: INCEPTION & RES-NET Versiyonları. https://medium.com/@ayyucekizrak/deri%CC%87n-bi%CC%87r kar%C5%9Fila%C5%9Ftirma-inception-res-net-versiyonlar%C4%B1-f5cfb83df131. Erişim tarihi: 21.07.2020

Krizhevsky, A., Sutskever, I., Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, 1097-1105.

Kurzweil, R., Richter, R., Kurzweil, R., & Schneider, M. L. 1990. The age of intelligent machines (Vol. 579). Cambridge, MA: MIT press.

Le, Q. V., Ngiam, J., Coates, A., Lahiri, A., Prochnow, B., & Ng, A. Y. 2011. On optimization methods for deep learning. In Proceedings of the 28th International Conference on International Conference on Machine Learning, 65 272.

LeCun, Y., Bengio, Y., Hinton, G. 2015. Deep learning. Nature, 521(7553), 436.

Lee, T. S., & Mumford, D. 2003. Hierarchical Bayesian inference in the visual cortex. JOSA A, 20(7), 1434-1448.

Lin, M., Chen, Q., Yan, S. 2013. Network in network. arXiv preprintarXiv:1312.4400.

Liu, L., Shen, C., Van den Hengel, A. 2015. The treasure beneath convolutional layers: Cross-convolutional-layer pooling for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4749-4757.

Maas, A. L., Hannun, A. Y., & Ng, A. Y. 2013. Rectifier nonlinearities improve neural network acoustic models. In Proc. icml (Vol. 30, No. 1, p. 3).

Mitchell, T. M. 1997. Does machine learning really work. AI magazine, 18(3), 11-11.

Ovtcharov, K., Ruwase, O., Kim, J. Y., Fowers, J., Strauss, K., Chung, E. S. 2015. Accelerating deep convolutional neural networks using specialized hardware. Microsoft Research Whitepaper, 2(11), 1-4.

Özkan, İ. N., İ. K., Ülker, E. 2017. Derin Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri. Gaziosmanpaşa Bilimsel Araştırma Dergisi, 6(3), 85-104.

Patterson, J., Gibson, A. 2017. Deep learning: A practitioner's approach. " O'Reilly Media, Inc.".

Russell, S. J., Norvig, P. 2016. Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited,.

Sardoğan, M., Yunus, Ö. Z. E. N., & Tuncer, A. 2020. Detection of Apple Leaf Diseases using Faster R-CNN. Düzce Üniversitesi Bilim ve Teknoloji Dergisi, 8(1), 1110-1117.

Sertkaya, M. E. 2018. Derin Öğrenme Tekniklerinin Biyomedikal İmgeler Üzerine Uygulamaları, Yüksek Lisans Tezi. Fırat Üniversitesi Fen Bilimleri Enstitüsü, Elazığ.

Stallkamp, J., Schlipsing, M., Salmen, J., & Igel, C. 2012. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. Neural Networks, 32, 323-332.

Sudowe, P., Leibe, B. 2011. Efficient use of geometric constraints for sliding window object detection in video. In International Conference on Computer Vision Systems, Springer yayınevi, Berlin, Heidelberg, 11-20.

Şahin, Ö. 2017. iOs platformunda görme engelliler için TL Tanıma Uygulaması, Doktora Tezi. Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya.

Şeker, A., Diri, B., & Balık, H. H. 2017. Derin Öğrenme Yöntemleri ve Uygulamaları Hakkında Bir İnceleme. Gazi Mühendislik Bilimleri Dergisi (GMBD), 3(3), 47- 64.

Tan, Z. 2019. Derin öğrenme yardımıyla araç sınıflandırma. Yüksek Lisans Tezi. Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ.

Tang, Y. 2013. Deep learning using linear support vector machinesarXiv:1306.0239.

Uijlings, J. R., Van De Sande, K. E., Gevers, T., Smeulders, A. W. 2013. Selective search for object recognition. International journal of computer vision, 104(2), 154-171.

Viola, P., Jones, M. J. 2004. Robust real-time face detection. International journal of computer vision, 57(2), 137-154.

Xiao, T., Li, H., Ouyang, W., Wang, X. 2016. Learning deep feature representations with domain guided dropout for person re-identification. In Proceedings of the IEEE conference on computer vision and pattern recognition, 1249-1258.

Yang, J., Nguyen, M. N., San, P. P., Li, X. L., Krishnaswamy, S. 2015. Deep convolutional neural networks on multichannel time series for human activity recognition. In Twenty-Fourth International Joint Conference on Artificial Intelligence.

Yıldırım, O. 2019. Derin öğrenme yöntemleri ile trafik işareti tanıma. Yüksek Lisans Tezi. Marmara Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

Zaklouta, F., & Stanciulescu, B. 2012. Real-time traffic-sign recognition using tree classifiers. IEEE Transactions on Intelligent Transportation Systems, 13(4), 1507-1514.

Zeng, Y., Xu, X., Fang, Y., & Zhao, K. 2015. Traffic sign recognition using deep convolutional networks and extreme learning machine. In International Conference on Intelligent Science and Big Data Engineering,Springer yayınevi, 272-280.

Zhang, Q., Xu, J., Xu, L., & Guo, H. 2016. Deep convolutional neural networks for forest fire detection. 2016 International Forum on Management, Education and Information Technology Application. Atlantis Press.

Zhao, Z. Q., Zheng, P., Xu, S. T., Wu, X. 2019. Object detection with deep learning: A review. IEEE transactions on neural networks and learning systems.

Zitnick, C. L., Dollár, P. 2014. Edge boxes: Locating object proposals from edges. In European conference on computer vision, 391-405.

EKLER

EK 1 AlexNet ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans

EK 2 Cifar10Net ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans

EK 3 YZNet ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans

EK 1 AlexNet ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans

Trafik işareti sınıfı	Üst üste kesişen bölge oranı	YP	YN	DP	DN	Keskinlik	Duyarlılık	Yanlış alarm oranı	Doğruluk
Hız sınırı 30	40%	8	16	15	308	0,65	0,48	0,347826087	0,933518
	50%	8	16	15	307	0,65	0,48	0,347826087	0,933518
	60%	16	24	7	306	0,3	0,23	0,695652174	0,8891967
	70%	20	28	3	305	0,13	0,1	0,869565217	0,867036
Hız sınırı 50	40%	3	12	10	333	0,77	0,45	0,230769231	0,9584488
	50%	5	14	8	333	0,62	0,36	0,384615385	0,9473684
	60%	8	17	5	333	0,38	0,23	0,615384615	0,9307479
	70%	10	19	3	333	0,23	0,14	0,769230769	0,9196676
Hız sınırı 70	40%	3	30	7	324	0,7	0,19	0,3	0,9085873
	50%	4	31	6	324	0,6	0,16	0,4	0,9030471
	60%	4	31	6	324	0,6	0,16	0,4	0,9030471
	70%	8	35	2	324	0,2	0,05	0,8	0,8808864
Hız sınırı 80	40%	2	11	5	337	0,71	0,31	0,285714286	0,9639889
	50%	3	12	4	337	0,57	0,25	0,428571429	0,9584488
	60%	5	14	2	337	0,29	0,13	0,714285714	0,9473684
	70%	6	15	1	336	0,14	0,06	0,857142857	0,9418283
Yol ver	40%	6	19	12	268	0,67	0,39	0,333333333	0,9307479
	50%	10	23	8	265	0,44	0,26	0,555555556	0,9085873
	60%	15	28	3	260	0,17	0,1	0,833333333	0,8808864
	70%	15	28	3	259	0,17	0,1	0,833333333	0,8808864
Sağdan devam et	40%	4	19	12	281	0,75	0,39	0,25	0,9362881
	50%	4	19	12	281	0,75	0,39	0,25	0,9362881
	60%	8	23	8	279	0,5	0,26	0,5	0,9141274
	70%	12	27	4	273	0,25	0,13	0,75	0,8919668
Öncelikli yol	40%	3	20	11	299	0,79	0,35	0,214285714	0,9362881
	50%	5	22	9	299	0,64	0,29	0,357142857	0,9252078
	60%	6	23	8	299	0,57	0,26	0,428571429	0,9196676
	70%	11	28	3	298	0,21	0,1	0,785714286	0,8919668
Kamyonların öndeki aracı geçmesi yasak	40%	2	11	6	303	0,75	0,35	0,25	0,9639889
	50%	2	11	6	303	0,75	0,35	0,25	0,9639889
	60%	3	12	5	301	0,63	0,29	0,375	0,9584488
	70%	5	14	3	300	0,38	0,18	0,625	0,9473684
Dur	40%	2	4	6	295	0,75	0,6	0,25	0,9833795
	50%	2	4	6	295	0,75	0,6	0,25	0,9833795
	60%	3	5	5	294	0,63	0,5	0,375	0,9778393
	70%	7	9	1	293	0,13	0,1	0,875	0,9556787
Tehlike	40%	1	7	4	350	0,8	0,36	0,2	0,9778393
	50%	1	7	4	350	0,8	0,36	0,2	0,9778393
	60%	2	8	3	350	0,6	0,27	0,4	0,9722992
	70%	3	9	2	350	0,4	0,18	0,6	0,966759

EK 2 Cifar10Net ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans

Trafik işareti sınıfı	Üst üste kesişen bölge oranı	YP	YN	DP	DN	Keskinlik	Duyarlılık	Yanlış alarm oranı	Doğruluk
Hız sınırı 30	40%	9	19	12	321	0,57	0,39	0,428571429	0,92243767
	50%	9	19	12	321	0,57	0,39	0,428571429	0,92243767
	60%	14	24	7	316	0,33	0,23	0,666666667	0,89473684
	70%	17	27	4	313	0,19	0,13	0,80952381	0,87811634
Hız sınırı 50	40%	7	15	7	332	0,5	0,32	0,5	0,93905817
	50%	9	17	5	330	0,36	0,23	0,642857143	0,92797784
	60%	11	19	3	328	0,21	0,14	0,785714286	0,91689751
	70%	13	21	1	326	0,07	0,05	0,928571429	0,90581717
Hız sınırı 70	40%	4	34	3	320	0,43	0,08	0,571428571	0,89473684
	50%	4	34	3	320	0,43	0,08	0,571428571	0,89473684
	60%	4	34	3	320	0,43	0,08	0,571428571	0,89473684
	70%	6	36	1	318	0,14	0,03	0,857142857	0,88365651
Hız sınırı 80	40%	5	11	5	340	0,5	0,31	0,5	0,95567867
	50%	5	11	5	340	0,5	0,31	0,5	0,95567867
	60%	6	12	4	339	0,4	0,25	0,6	0,9501385
	70%	7	13	3	338	0,3	0,19	0,7	0,94459834
Yol ver	40%	9	19	12	321	0,57	0,39	0,428571429	0,92243767
	50%	13	23	8	317	0,38	0,26	0,619047619	0,90027701
	60%	19	29	2	311	0,1	0,06	0,904761905	0,86703601
	70%	21	31	0	309	0	0	1	0,85595568
Sağdan devam et	40%	7	22	9	323	0,56	0,29	0,4375	0,91966759
	50%	7	22	9	323	0,56	0,29	0,4375	0,91966759
	60%	9	24	7	321	0,44	0,23	0,5625	0,90858726
	70%	14	29	2	316	0,13	0,06	0,875	0,88088643
Öncelikli yol	40%	13	23	8	317	0,38	0,26	0,619047619	0,90027701
	50%	14	24	7	316	0,33	0,23	0,666666667	0,89473684
	60%	16	26	5	314	0,24	0,16	0,761904762	0,88365651
	70%	20	30	1	310	0,05	0,03	0,952380952	0,86149584
Kamyonların öndeki aracı geçmesi yasak	40%	4	11	6	340	0,6	0,35	0,4	0,95844875
	50%	4	11	6	340	0,6	0,35	0,4	0,95844875
	60%	6	13	4	338	0,4	0,24	0,6	0,94736842
	70%	7	14	3	337	0,3	0,18	0,7	0,94182825
Dur	40%	2	4	6	349	0,75	0,6	0,25	0,9833795
	50%	4	6	4	347	0,5	0,4	0,5	0,97229917
	60%	7	9	1	344	0,13	0,1	0,875	0,95567867
	70%	7	9	1	344	0,13	0,1	0,875	0,95567867
Tehlike	40%	0	10	1	350	1	0,09	0	0,97229917
	50%	1	11	0	349	0	0	1	0,966759
	60%	1	11	0	349	0	0	1	0,966759
	70%	1	11	0	349	0	0	1	0,966759

EK 3 YZNet ağında trafik işaretlerinin tespiti ve işaretlenmesine yönelik performans

Trafik işareti sınıfı	Üst üste kesişen bölge oranı	YP	YN	DP	DN	Keskinlik	Duyarlılık	Yanlış alarm oranı	Doğruluk
Hız sınırı 30	40%	22	26	5	308	0,19	0,16	0,814814815	0,867036
	50%	23	27	4	307	0,15	0,13	0,851851852	0,861496
	60%	24	28	3	306	0,11	0,1	0,888888889	0,855956
	70%	25	29	2	305	0,07	0,06	0,925925926	0,850416
Hız sınırı 50	40%	6	21	1	333	0,14	0,05	0,857142857	0,925208
	50%	6	21	1	333	0,14	0,05	0,857142857	0,925208
	60%	6	21	1	333	0,14	0,05	0,857142857	0,925208
	70%	6	21	1	333	0,14	0,05	0,857142857	0,925208
Hız sınırı 70	40%	0	37	0	324	NaN*	0	NaN*	0,897507
	50%	0	37	0	324	NaN*	0	NaN*	0,897507
	60%	0	37	0	324	NaN*	0	NaN*	0,897507
	70%	0	37	0	324	NaN*	0	NaN*	0,897507
Hız sınırı 80	40%	8	13	3	337	0,27	0,19	0,727272727	0,941828
	50%	8	13	3	337	0,27	0,19	0,727272727	0,941828
	60%	8	13	3	337	0,27	0,19	0,727272727	0,941828
	70%	9	14	2	336	0,18	0,13	0,818181818	0,936288
Yol ver	40%	62	21	10	268	0,14	0,32	0,861111111	0,770083
	50%	65	24	7	265	0,1	0,23	0,902777778	0,753463
	60%	70	29	2	260	0,03	0,06	0,972222222	0,725762
	70%	71	30	1	259	0,01	0,03	0,986111111	0,720222
Sağdan devam et	40%	49	21	10	281	0,17	0,32	0,830508475	0,806094
	50%	49	21	10	281	0,17	0,32	0,830508475	0,806094
	60%	51	23	8	279	0,14	0,26	0,86440678	0,795014
	70%	57	29	2	273	0,03	0,06	0,966101695	0,761773
Öncelikli yol	40%	31	29	2	299	0,06	0,06	0,939393939	0,833795
	50%	31	29	2	299	0,06	0,06	0,939393939	0,833795
	60%	31	29	2	299	0,06	0,06	0,939393939	0,833795
	70%	32	30	1	298	0,03	0,03	0,96969697	0,828255
Kamyonların öndeki aracı geçmesi yasak	40%	41	13	4	303	0,09	0,24	0,911111111	0,850416
	50%	41	13	4	303	0,09	0,24	0,911111111	0,850416
	60%	43	15	2	301	0,04	0,12	0,955555556	0,839335
	70%	44	16	1	300	0,02	0,06	0,977777778	0,833795
Dur	40%	56	8	2	295	0,03	0,2	0,965517241	0,822715
	50%	56	8	2	295	0,03	0,2	0,965517241	0,822715
	60%	57	9	1	294	0,02	0,1	0,982758621	0,817175
	70%	58	10	0	293	0	0	1	0,811634
Tehlike	40%	0	11	0	350	NaN*	0	NaN*	0,969529
	50%	0	11	0	350	NaN*	0	NaN*	0,969529
	60%	0	11	0	350	NaN*	0	NaN*	0,969529
	70%	0	11	0	350	NaN*	0	NaN*	0,969529

ÖZGEÇMİŞ

Adı Soyadı : Yeşim KALKAN

Doğum Yeri : Ankara

Doğum Tarihi : 26.08.1988

Medeni Durumu : Bekar

Yabancı Dili : İngilizce

Eğitim Durumu (Kurum ve Yıl)

Lise : İncesu Anadolu Lisesi (09.2002 – 06.2006)

Lisans : Karedeniz Teknik Üniversitesi Elektrik Elektronik Mühendisliği Bölümü (09.2007 – 06.2013)

Yüksek Lisans : Ankara Üniversitesi Fen Bilimleri Enstitüsü Elektrik Elektronik Mühendisliği Anabilim Dalı (Ağustos 2020)

Çalıştığı Kurum/Kurumlar, Görev ve Yıl

Gate Elektronik A.Ş, Donanım Tasarım Mühendisi, 04.2014 – 04.2016

SDT Uzay ve Savunma Teknolojileri A.Ş, Pcb Tasarım Mühendisi, 08.2017 –04.2019

Havelsan A.Ş., Sistem Mühendisi, 04.2020 – Halen Devam Etmektedir