T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

EVRİŞİMSEL SİNİR AĞLARI İLE ORTAM TARİFLEME

ANIL ÇETİNSOY

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
BİLGİSAYAR MÜHENDİSLİĞİ PROGRAMI

DANIŞMAN
DOÇ. DR. MİNE ELİF KARSLIGİL

İSTANBUL, 2018

T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

EVRİŞİMSEL SİNİR AĞLARI İLE ORTAM TARİFLEME

Anıl ÇETİNSOY tarafından hazırlanan tez çalışması 18.12.2018 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Doç. Dr. Mine Elif KARSLIGİL

Yıldız Teknik Üniversitesi

Jüri Üyeleri

Doç. Dr. Mine Elif KARSLIGİL

Yıldız Teknik Üniversitesi _____

Doç. Dr. Songül VARLI

Yıldız Teknik Üniversitesi _____

Dr. Öğr. Üyesi Elena BATTİNİ SÖNMEZ

İstanbul Bilgi Üniversitesi _____

ÖNSÖZ

Görme engelli insanlar, insanın en temel duyularından bir tanesi olan görme duyusuna sahip olmayan veya kısıtlı sahip olan insanlardır. Görme duyusuna sahip olmadıkları için de çevrelerinde ne olup bittiğini, nerede olduklarını, etraflarında neler olduğunu bilememektedirler. Bu çalışmada, görme engelli insanların, göremedikleri için günlük hayatta karşılaştıkları zorluklara bir çözüm bulmak hedeflenmiştir. Bu kapsamda, çevresini göremeyen bir insana, çevresinde neler olduğunu ve nasıl bir dizilime sahip olduğunu anlatan bir uygulama geliştirilmiştir.

Bu çalışmanın araştırma, geliştirme ve test süreçleri dahil olmak üzere tüm aşamalarında değerli bilgi ve tecrübelerini benimle paylaşan ve her konuda yardımlarını esirgemeyen değerli danışmanım Doç. Dr. Mine Elif KARSLIGİL'e teşekkürlerimi sunarım.

Aralık, 2018

Anıl ÇETİNSOY

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ vi

KISALTMA LİSTESİ vii

ŞEKİL LİSTESİ viiii

ÖZET x

ABSTRACT xi

BÖLÜM 1

GİRİŞ 1

1.1 Literatür Özeti 1
1.2 Tezin Amacı 3
1.3 Hipotez 3

BÖLÜM 2

GENEL BİLGİLER 4

2.1 Derin Konvolüsyonel Sinir Ağları 4
2.1.1 Özellik Çıkarma Bölümü 7
2.1.1.1 Konvolüsyon Katmanı 7
2.1.1.2 Havuzlama Katmanı 8
2.1.1.3 Aktivasyon Katmanı 9
2.1.1.4 Normalizasyon Katmanı 10
2.1.2 Sınıflandırma Bölümü 10
2.1.2.1 Tam Bağlı Katmanlar 10

BÖLÜM 3

SİSTEM TASARIMI 11
3.1 Görüntüdeki Nesnelerin Anlamsal Bölütlenmesi 12
3.2 Konvolüsyonel Sinir Ağları ile Nesne Tanıma 17
3.3 Ortam Tarifinin Yapılması 23
3.3.1 Nesnelerin Tariflenme Sırasının Belirlenmesi 23
3.3.2 Doğal Dilde Cümlelerin Oluşturulması 26

BÖLÜM 4

UYGULAMA 27
4.1 Veriseti 27
4.2 Sinir Ağının Eğitimi 27
4.3 Uygulama Adımları 28
4.4 Uygulama Başarısı 29

BÖLÜM 5

SONUÇ VE ÖNERİLER 34
KAYNAKLAR 36
ÖZGEÇMİŞ 39

SİMGE LİSTESİ

$büyüklük(BB_{ij})$	iki bölütü çevreleyen kutunun piksel büyüklüğü
$büyüklük(görüntü)$	görüntünün piksel büyüklüğü
$büyüklük(r_i)$	i bölütünün piksel büyüklüğü
$büyüklük(r_j)$	j bölütünün piksel büyüklüğü
C_a	a pikselinin renk değeri
C_b	b pikselinin renk değeri
c_i^k	i bölütü renk histogramı
c_j^k	j bölütü renk histogramı
r_i	i bölütü
r_j	j bölütü
$s(r_i, r_j)$	i ve j bölütlerinin genel benzerliği
$s_{büyüklük}(r_i, r_j)$	i ve j bölütlerinin büyüklük benzerliği
$s_{desen}(r_i, r_j)$	i ve j bölütlerinin desen benzerliği
$s_{renk}(r_i, r_j)$	i ve j bölütlerinin renk benzerliği
$s_{şekil}(r_i, r_j)$	i ve j bölütlerinin şekil benzerliği
T	piksel birleştirme için eşik değeri
t_i^k	i bölütü desen histogramı
t_j^k	j bölütü desen histogramı
W_{ab}	a, b pikselleri arasındaki ağırlık değeri

KISALTMA LİSTESİ

DKSA	Derin Konvolüsyonel Sinir Ağları
ILSVRC	ImageNet Large Scale Visual Recognition Competition
KSA	Konvolüsyonel Sinir Ağları

ŞEKİL LİSTESİ

Sayfa

Şekil 1.1 Grafik tabanlı görüntü bölütleme algoritması örnek bir bölütleme 1
Şekil 1.2 Daha hızlı bölgesel konvolüsyonel sinir ağları örnek çalışma görüntüsü 2
Şekil 1.3 Görüntü altyazılama sistemi mimarisi 3
Şekil 2.1 Makine öğrenmesi ve derin öğrenme karşılaştırması 5
Şekil 2.2 Basit yapay sinir ağı ile derin öğrenme sinir ağı boyut karşılaştırması 6
Şekil 2.3 Basit yapay sinir ağı ile derin öğrenme sinir ağı derinlik karşılaştırması 7
Şekil 2.4 Konvolüsyonel sinir ağlarının özellik çıkarma ve sınıflandırma bölümleri 7
Şekil 2.5 Konvolüsyonel katmanlarda yapılan konvolüsyon işlemi 8
Şekil 2.6 Maksimum havuzlama ile 4x4 boyuttan 2x2 boyuta indirgeme işlemi 9
Şekil 2.7 ReLU aktivasyon fonksiyonu 9
Şekil 3.1 Evrişimsel sinir ağları ile ortam tarifleme sistemi blok diyagramı 11
Şekil 3.2 Grafik tabanlı bölütleme algoritması ağaç grafiği 13
Şekil 3.3 Grafik tabanlı bölütleme giriş ve çıkış görüntüleri 14
Şekil 3.4 (a) Orijinal görüntü, (b) Bölütleme sonrası görüntü 15
Şekil 3.5 (a) Hiyerarşik gruplanmış bölütler, (b) Bölütleme sonuç görüntüsü 17
Şekil 3.6 LeNet sinir ağı yapısı 18
Şekil 3.7 AlexNet sinir ağı yapısı 18
Şekil 3.8 ResNet sinir ağı yapısı 19
Şekil 3.9 Saf Inception modülü 19
Şekil 3.10 Inception v1 modülü 20
Şekil 3.11 Inception v1 mimarisi 20
Şekil 3.12 Inception v2 modülü 21
Şekil 3.13 Inception v3 sinir ağı yapısı 22
Şekil 3.14 Inception v3 ile nesne tanıma 23
Şekil 3.15 Ortam tarifinin yapılması işlem adımları 23
Şekil 3.16 Orjinal görüntü 24
Şekil 3.17 Tarifleme sırasının belirlenmesi işlem adımları 25
Şekil 3.18 Nesnelerin tariflenme sırası 26
Şekil 4.1 ADE20K verisetinden örnek bir görüntü 27
Şekil 4.2 (a) Giriş görüntüsü 1, (b) Çıkış görüntüsü 1 28
Şekil 4.3 (a) Tahmin edilen nesne adları 1, (b) Tarifleme için kurulan cümleler 1 28
Şekil 4.4 (a) Giriş görüntüsü 2, (b) Çıkış görüntüsü 2 29
Şekil 4.5 (a) Tahmin edilen nesne adları 2, (b) Tarifleme için kurulan cümleler 2 29

Şekil 4.6 (a) Giriş görüntüsü 3, (b) Çıkış görüntüsü 3 30
Şekil 4.7 (a) Tahmin edilen nesne adları 3, (b) Tarifleme için kurulan cümleler 3 30
Şekil 4.8 (a) Giriş görüntüsü 4, (b) Çıkış görüntüsü 4 31
Şekil 4.9 (a) Tahmin edilen nesne adları 4, (b) Tarifleme için kurulan cümleler 4 31
Şekil 4.10 (a) Giriş görüntüsü 5, (b) Çıkış görüntüsü 5 32
Şekil 4.11 (a) Tahmin edilen nesne adları 5, (b) Tarifleme için kurulan cümleler 5 32
Şekil 4.12 (a) Giriş görüntüsü 6, (b) Çıkış görüntüsü 6 32
Şekil 4.13 (a) Tahmin edilen nesne adları 6, (b) Tarifleme için kurulan cümleler 6 33

ÖZET

EVRİŞİMSEL SİNİR AĞLARI İLE ORTAM TARİFLEME

Anıl ÇETİNSOY

Bilgisayar Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Tez Danışmanı: Doç. Dr. Mine Elif KARSLIGİL

Bu çalışmada, içinde bulunulan ortamdaki nesneleri ve bu nesnelerin birbirlerine konumlarını tarifleyen bir sistem geliştirilmiştir. Bu sistem görme engelli kişilerin çevrelerinde neler olduğunu öğrenmesi, küçük çocukların çevrelerindeki nesneleri öğrenmesi gibi amaçlar için kullanılabilir. İsimleri bilinmeyen nesnelerin isimlerinin öğrenilmesi için kullanılabilir. Yapılan çalışmada, görüntülerde nesne bulunan bölgeler grafik tabanlı bölütleme yöntemi ile tespit edilmektedir. Tespit edilen bu bölgelerdeki aşırı bölütlenme sorunu Seçmeli Arama algoritması ile çözülmektedir. Nesne tespit edilen bölgeler, derin konvolüsyonel sinir ağı ile işlenerek, bu bölgelerde bulunan nesnelerin isimleri ve doğruluk olasılıkları tespit edilmektedir. Elde edilen nesne konumları ve isimleri cümleler kurularak kullanıcıya aktarılmaktadır. Yapılan uygulama ve testlerde görüldüğü üzere, son derece başarılı bir ortam tarifleme aracı geliştirilmiştir.

Anahtar Kelimeler: Derin öğrenme; bilgi edinimi; bilgisayarla görme.

ABSTRACT

ENVIRONMENT DESCRIPTION WITH CONVOLUTIONAL NEURAL NETWORKS

Anıl ÇETİNSOY

Department of Computer Engineering

MSc. Thesis

Adviser: Assoc. Prof. Dr. Mine Elif KARSLIGİL

In this study, a system which describes the objects present in the environment and their relative positions has been developed. This system can be used for visually impaired people to learn what is happening in their surroundings and for young children to learn objects in their surroundings. The proposed system can also be used for finding out the names of unknown objects. In the study, the regions that contain objects are detected by the graph based segmentation method in the images. The over segmentation problem of the detected regions is solved using the selective search algorithm. Detected regions are processed with deep convolutional neural network and the names and accuracy probabilities of the objects found in these regions are determined. Finally, the obtained object positions and names are transferred to the user by establishing sentences. Experimental results show that a highly successful environment description tool has been developed.

Keywords: Deep learning; information acquisition; computer vision.

BÖLÜM 1

GİRİŞ

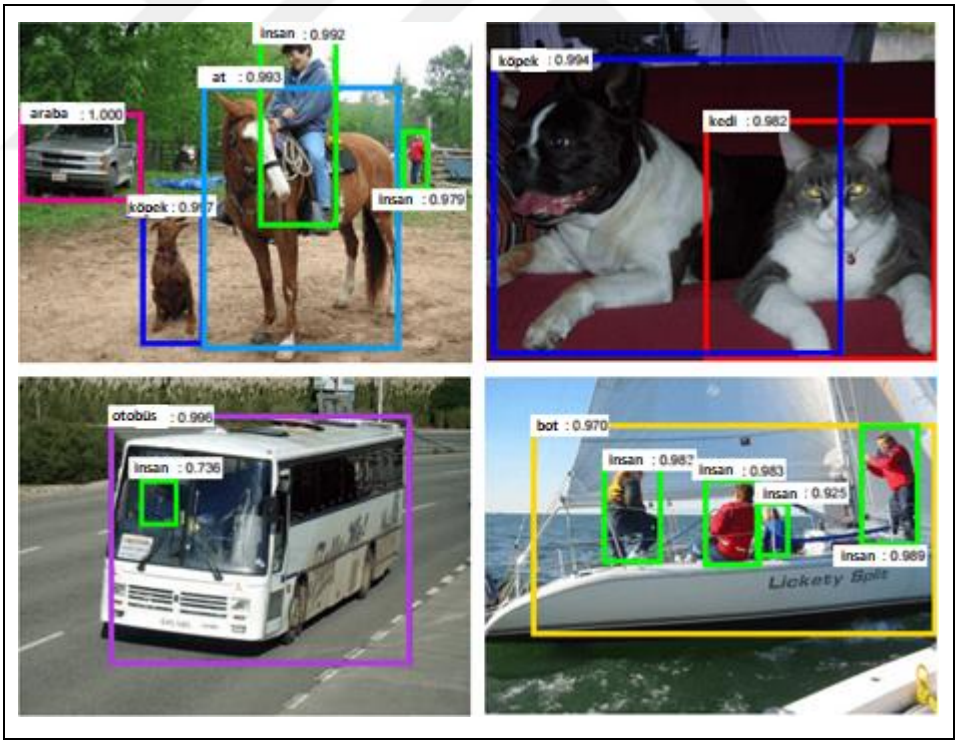
1.1 Literatür Özeti

Günümüzde, nesne tanıma sistemleri, içerisinde bir çok etiketlenmiş nesne olan eğitim setleri ile eğitilir, daha sonra bu eğitimden öğrendiği bilgiler ile çalışırlar. Bu eğitim setleri tekli nesne görüntüleri ve etiketlerden oluşurlar. Bu sistemlerin önündeki en büyük engel, gerçek hayatta bir kameradan alınan görüntüde hiçbir zaman tek bir nesne olmaması, yani bir görüntü içerisinde bir çok nesne bulunmasıdır. Bu sebeple bu sistemler nesne tanıma işlemi yapmadan önce, görüntüdeki her bir nesnenin ayrı ayrı bölütlenmesi ihtiyacı duyarlar. Felzenszwalb ve Huttenlocher tarafından geliştirilen Grafik Tabanlı Görüntü Bölütleme [1] algoritması bu konuda verimli bir çözüm sunmaktadır. Görüntü içerisinde bulunan piksellerin renk benzerliklerine göre gruplanarak, görüntü içerisinde bulunan nesnelerin konum ve büyüklük bilgilerinin elde edilmesine olanak sağlamaktadır.



Şekil 1.1 Grafik tabanlı görüntü bölütleme algoritması örnek bir bölütleme görüntüsü [1]

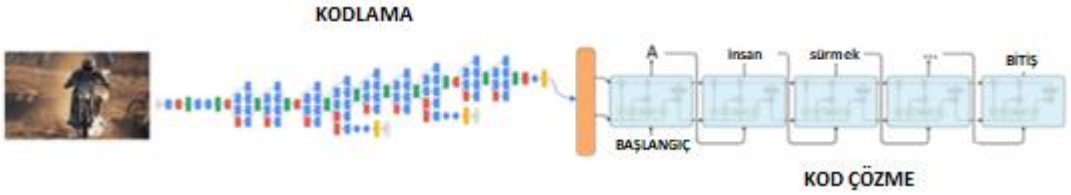
Görüntülerin anlamsal olarak bölümlere ayrılması konusunda yapılan bir diğer çalışma ise Ross Girshick tarafından geliştirilen Bölgesel Konvolüsyonel Sinir Ağları [2] modelidir. Bu ağ modelinde, görüntü üzerinde nesne bulunan bölgelerin çıkarılarak, bir sınıflandırıcıya iletilerek, nesnelerin tanınması sağlanmaktadır. Bu modelde, nesne bulunan tüm bölgeler konvolüsyonel sinir ağına iletildiği için her bir görüntüde yaklaşık 2000 adet bölge ağ tarafından işlenmektedir. Bu da işlemlerin uzun sürmesine sebep olmaktadır. Bu sorunun çözülmesi için geliştirilen Hızlı Bölgesel Sinir Ağları [3] modelinde, bir çok bölge hesaplanmayıp, tüm görüntü olduğu gibi konvolüsyonel sinir ağına iletilir ve özellik haritası çıkarılır. Daha sonra bu özellik haritası üzerinde çeşitli havuzlama işlemleri yapılarak, nesne özelikleri elde edilir. Bu modelden sonra geliştirilen Daha Hızlı Bölgesel Konvolüsyonel Sinir Ağları [4] modeli de, nesne bulunan bölgeleri bulmak için ayrı bir konvolüsyonel sinir ağı kullanarak, bölgelerin elde edilmesinde hız kazanılmasını sağlayarak, gerçek zamanlı nesne tanıma işlemlerinde kullanılabilir bir çözüm sunmuştur.



Şekil 1.2 Daha hızlı bölgesel konvolüsyonel sinir ağları örnek çalışma görüntüsü [4]

Görüntü Altyazılama alanında yapılan çalışmalarda [5], [6], [7], görüntülerde bulunan nesnelerin ve olayların doğal dilde kurulan cümleler ile ifade edilmesi sağlanmıştır. Bu

mimari, kodlama ve kod çözme olmak üzere 2 bölümden oluşmaktadır. Kodlama bölümünde giriş görüntüsü Konvolüsyonel Sinir Ağı tarafından bir özellik haritası olarak kodlanır. Bu özellik haritası kod çözme bölümünde Uzun Kısa Süreli Hafıza [8] kod çözücüsü tarafından işlenerek doğal dilde cümlelelere dönüştürülür. Günümüzde popüler hale gelmeye başlayan bu çalışmalar, görüntü üzerinde bulunan nesne ve olayları tıpkı bir insanın ifade ettiği şekilde ifade edebilmektedir.



Şekil 1.3 Görüntü altyazılama sistemi mimarisi [9]

1.2 Tezin Amacı

Bu çalışmada, görüş alanında bulunan nesneleri ve bunların birbirlerine göre konumlarını tanımlayan bir sistem geliştirilmiştir. Geliştirilen sistem özellikle görme engelli kişilerin bulundukları ortam hakkında bilgi sahibi olmalarını sağlayacağı gibi, küçük çocuklara ortamdaki nesneleri ve birbirine göre konumlarını öğretmek için de kullanılabilir.

Geliştirilen sistemde önce ortamdan alınan görüntüdeki nesneler bölütlenerek tanınmakta, daha sonra nesnelerin birbirlerine göre konumları belirlenmektedir.

1.3 Hipotez

Herhangi bir cihazın kamerasından alınan verinin, bir sistem tarafından işlenerek, görünen nesnelerin ne olduğu ve nasıl bir dizilime sahip olduklarını anlatan bir sistem görme engelli kişiler, çocuklar ve yaşlılar için büyük kolaylık sağlayacaktır. Böylece, çevredeki nesneleri ve dizilimlerini duyularıyla algılayamayan kişiler, bu uygulama vasıtasıyla bu bilgiye sahip olabilecektir.

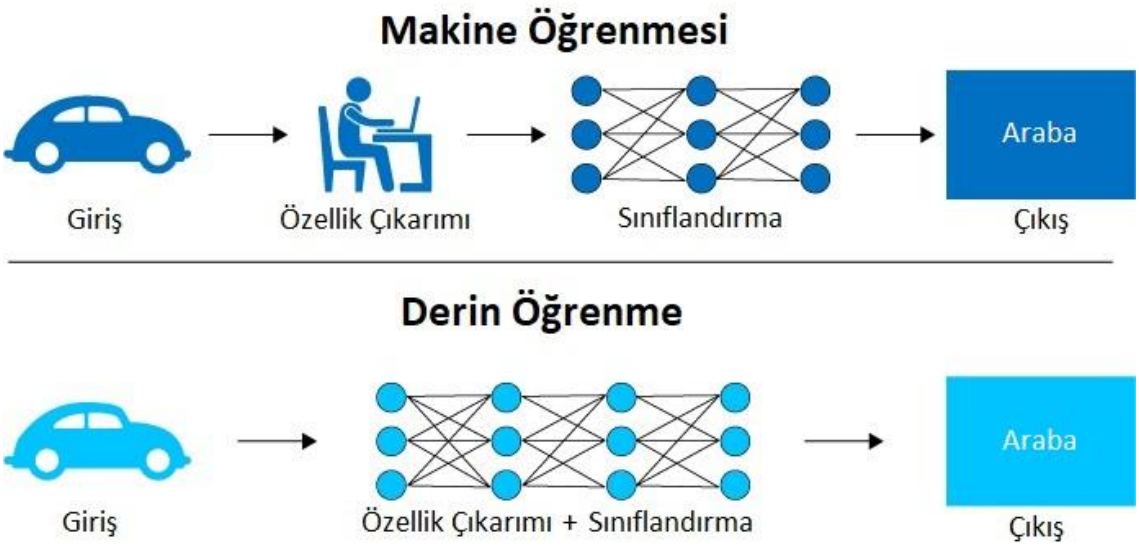
BÖLÜM 2

GENEL BİLGİLER

Bu çalışmada, bir alanda bulunan nesneleri ve bunların birbirlerine göre konumlarını tespit ederek, kullanıcıya bilgi veren bir sistem geliştirilmiştir. Bu sistem için öncelikle kameradan alınan görüntüler anlamlandırılmalıdır. Bu işlem için ilk önce kameradan alınan görüntüler bölütleme işlemine tabi tutularak, görüntü üzerindeki her bir nesne tespit edilir. Nesneler tespit edildikten sonra her bir nesne, tanıma işleminden geçirilir. Bu işlemden sonra elde edilen nesne adları ve görüntü üzerindeki yerleri, doğal dilde cümleler kurularak, kullanıcıya aktarılır.

2.1 Derin Konvolüsyonel Sinir Ağları

Yapay zeka ve makine öğrenmesinin bir alt dalı olan Derin Öğrenme, çok katmanlı yapay sinir ağlarıdır. Elle verilmiş kurallar veya insan bilgisini kullanmadan görüntü, video veya metin gibi verilere ait bilgileri otomatik olarak öğrenebilirler. Bu özellikleriyle geleneksel makine öğrenmesi tekniklerinden ayrılırlar. Mimarileri oldukça esnek olduğundan ham verilerden bilgi öğrenebilirler. Fazla veri ile eğitildiklerinde tahmin doğruluklarını artırabilirler.



Şekil 2.1 Makine öğrenmesi ve derin öğrenme karşılaştırması [10]

Derin Öğrenme, nesne tanıma, konuşma tanıma, doğal dil işleme alanlarında en iyi çözümü sunmaktadır. Görüntü ve ses tanıma, otonom araçlar, akıllı asistanlar, sosyal ağlar, güvenlik sistemleri gibi bir çok alanda kullanılmaktadır.

Derin Konvolüsyonel Sinir Ağları, görüntü tanıma, sınıflandırma ve kümeleme işleri için kullanılan derin yapay sinir ağlarıdır. İnsan yüzlerini, sokak işaretlerini, tümörleri ve diğer tüm görsel verileri tanımlayabilen algoritmalardır. Konvolüsyonel sinir ağları, metni sayısallaştırmak için optik karakter tanıma yaparak elle yazılmış belgeler üzerinde, görüntülerin sembollerin kopyalanabildiği doğal dil işlemlerini mümkün kılar. Ses verileri üzerinde kullanılarak ses tanıma, sesli yönlendirme işlerinde kullanılabilirler. Konvolüsyonel Sinir Ağları ayrıca, video etiketleme, metin analizi, konuşma tanıma, doğal dil işleme, metin sınıflandırma, tıbbi teşhislendirme ve en son teknoloji ürünü robotlar, dronlar, sanal asistanlar, otonom araçlar gibi bir çok alanda kullanılmaktadır.

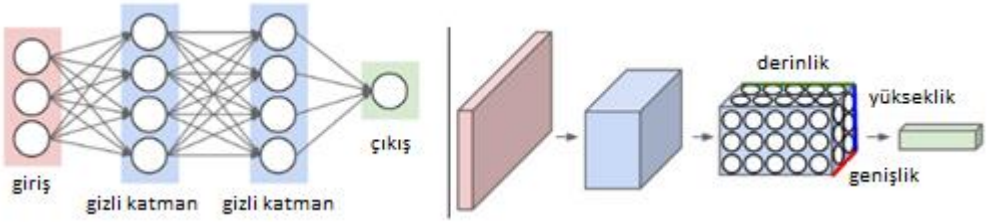
Derin Konvolüsyonel Sinir Ağları (DKSA) zor görsel tanıma işlemleri için çok başarılı sonuçlar vermektedir. Bu sonuçlar bir çok akademik çalışma ile ispatlanmıştır. DKSA ile başarılı sonuçlar veren çalışmalardan bazıları olarak AlexNet [11] ve Inception V3 [12] örnek gösterilebilir. Bu çalışmalar bir çok veri seti üzerinde çalıştırılarak başarıları ispatlanmıştır. Bu çalışmada literatürdeki çalışmalardan en yeni ve en başarılısı olan InceptionV3 algoritması kullanılmıştır. Yapılan çalışmalarda InceptionV3 algoritması ilk 5 tahminde %3.46 hata payına ulaşarak başarısını ispat etmiştir.

Konvolüsyonel Sinir Ağları, giriş bilgilerini konvolüsyonel katmanlar ile filtreleyerek, yararlı bilgiye ulaşmayı hedefleyen bir yapay sinir ağı modelidir. Konvolüsyon işlemi, giriş verisini bir filtreden geçirerek, dönüştürülmüş bir özellik haritası oluşturma işlemidir. Konvolüsyon katmanlarındaki filtreler, belirli bir görev için en yararlı bilgileri çıkarmak için öğrenilen parametrelere göre değiştirilir. Konvolüsyonel ağlar, yapılmak istenen işleme göre en iyi özelliği bulmak için otomatik olarak ayarlanır [13].

KSA'lar giriş, çıkış ve bir çok gizli katmandan oluşurlar. Gizli katmanlar genel olarak konvolüsyonel katmanlar, havuzlama katmanları, tam bağlantılı katmanlar ve normalizasyon katmanlarından oluşur. Konvolüsyonel Sinir Ağları mimarisi, bir giriş görüntüsünün 2 boyutlu yapısından tam anlamıyla yararlanabilmek için tasarlanmıştır. Konvolüsyonel Sinir Ağları daha kolay eğitilmeleri ve daha az sayıda parametreye sahip oldukları için geleneksel yapay sinir ağlarından daha avantajlıdır.

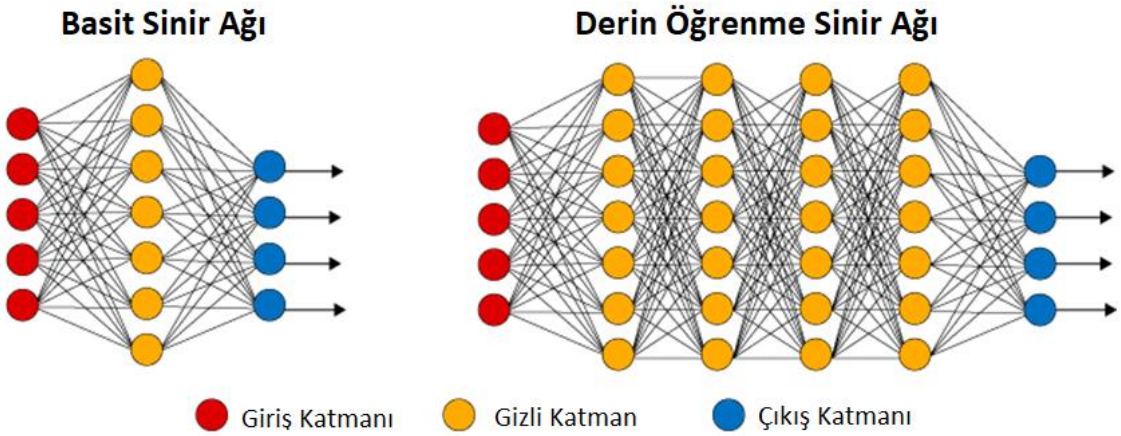
Konvolüsyonel Sinir Ağları, normal yapay sinir ağlarından farklı bir mimariye sahiptir. Normal sinir ağları, bir giriş verisini bir dizi gizli katmandan geçirerek çıkış bilgisine dönüştürür. Her katman bir dizi nörondan oluşur ve her katman kendisinden önceki katmanın tüm nöronlarına tam bağlıdır. Son olarak, tahminleri oluşturan tam bağlı çıkış katmanı bulunur.

Konvolüsyonel Sinir Ağları ise biraz farklıdır. Katmanlar genişlik, yükseklik ve derinlik olmak üzere 3 boyuttan oluşmaktadır. Ayrıca, bir katmandaki nöronlar, bir önceki katmandaki tüm nöronlara değil, sadece küçük bir bölgesine bağlanır. En sonda ise, tahminleri oluşturan bir çıkış katmanı bulunur.



Şekil 2.2 Basit yapay sinir ağı ile derin öğrenme sinir ağı boyut karşılaştırması [14]

Normal yapay sinir ağlarında genellikle bir veya iki gizli katman bulunurken, derin konvolüsyonel sinir ağlarında bir çok gizli katman bulunmaktadır.



Şekil 2.3 Basit yapay sinir ağı ile derin öğrenme sinir ağı derinlik karşılaştırması [15]

Konvolüsyonel Sinir Ağları, Özellik Çıkarma Bölümü ve Sınıflandırma Bölümü olmak üzere 2 bölümden oluşurlar.



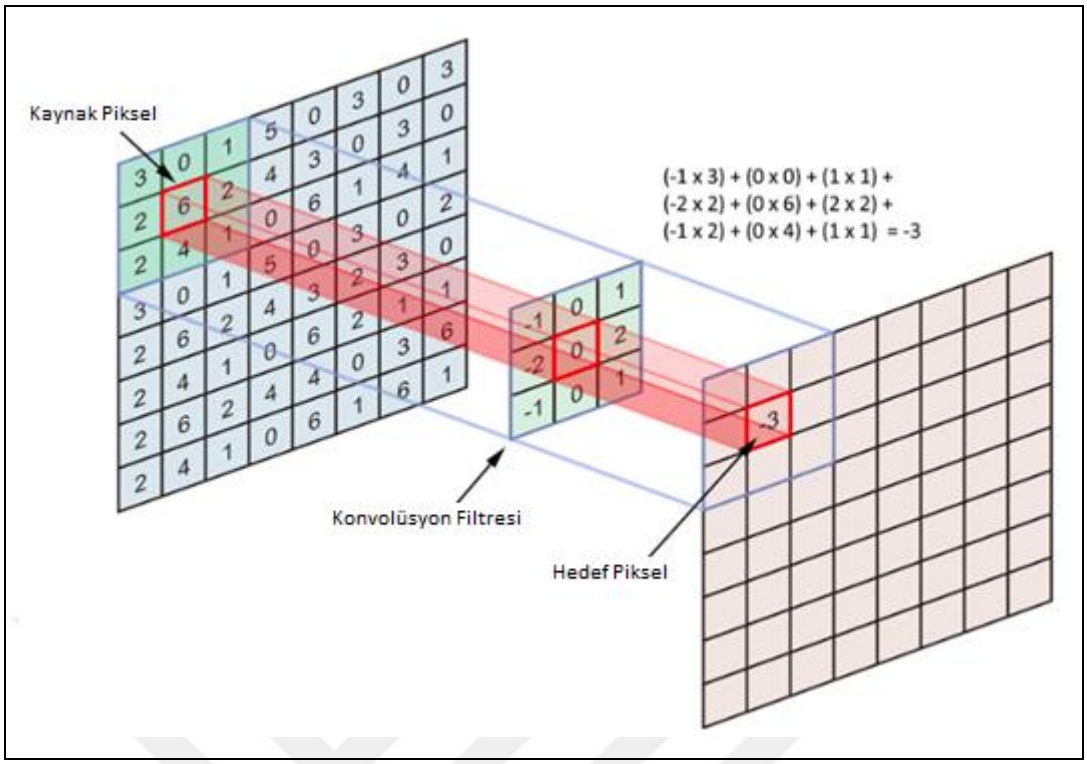
Şekil 2.4 Konvolüsyonel sinir ağlarının özellik çıkarma ve sınıflandırma bölümleri [16]

2.1.1 Özellik Çıkarma Bölümü

Bu bölümde nesnelerin özelliklerinin tespit edildiği işlemler yapılmaktadır. Özellik Çıkarma bölümünde konvolüsyon, havuzlama, aktivasyon ve normalizasyon katmanları bulunmaktadır.

2.1.1.1 Konvolüsyon Katmanı

Konvolüsyonel Sinir Ağlarının temelini konvolüsyon işlemi oluşturur. Bu sebeple, bu ağlarda bir çok konvolüsyon katmanı bulunur. Bu katmanlarda, giriş verisi bir çok farklı filtre ile konvolüsyon işlemine tabi tutularak, filtre sayısı kadar farklı özellik haritaları oluşturulur. Bu özellik haritaları bir araya gelerek, konvolüsyon katmanının çıkış verisini oluştururlar.

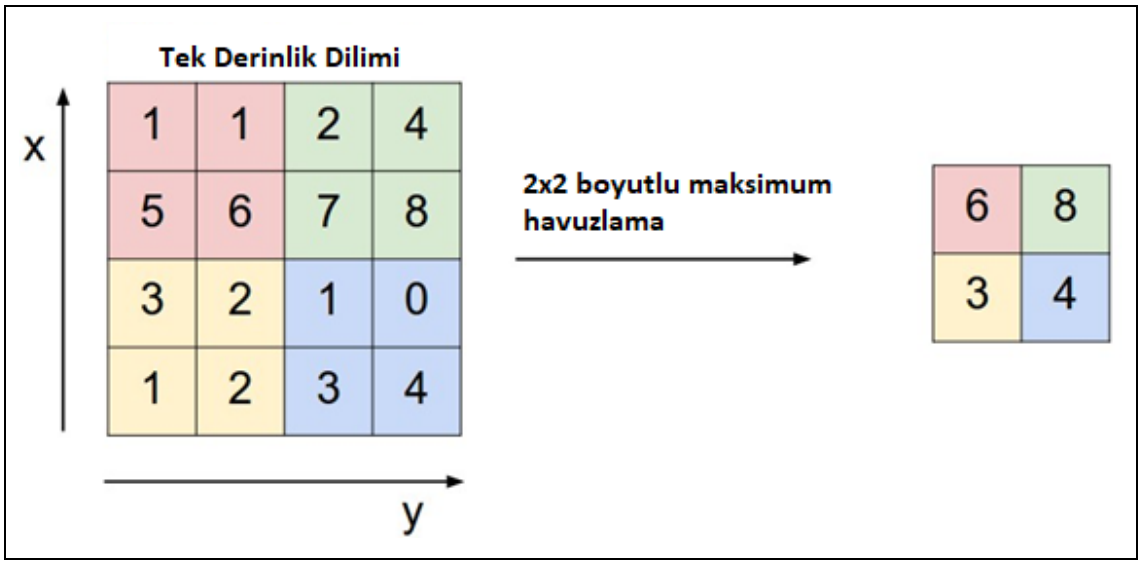


Şekil 2.5 Konvolüsyonel katmanlarda yapılan konvolüsyon işlemi [17]

2.1.1.2 Havuzlama katmanı

Konvolüsyon katmanlarından sonra, genellikle bir havuzlama katmanı eklenir. Havuzlama katmanının işlevi, boyutları azaltarak, ağdaki parametrelerin ve hesaplamaların sayısını azaltmaktır. Bu işlem eğitim süresini kısaltır ve aşırı uyumluluk (overfitting) oluşmasını engeller.

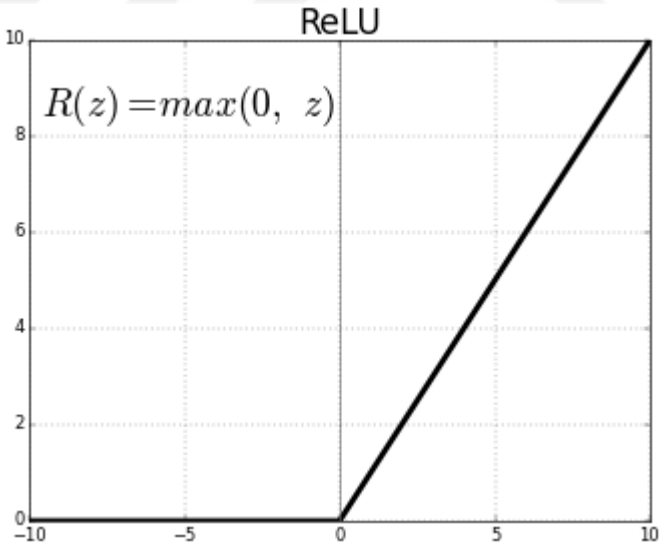
Konvolüsyonel sinir ağlarında, genellikle maksimum havuzlama kullanılır. Maksimum havuzlama işleminde, her pencerede maksimum değer alınarak boyutlar küçültülür. Bu işlem, özellik harita boyutunu azaltırken, aynı zamanda önemli bilgileri de korur.



Şekil 2.6 Maksimum havuzlama ile 4x4 boyuttan 2x2 boyuta indirgeme işlemi [14]

2.1.1.3 Aktivasyon katmanı

Konvolüsyonel sinir ağlarındaki aktivasyon katmanı, ağırlıklı giriş verileri üzerinde doğrusal olmayan bir dönüşüm uygular. KSA'larda genellikle, ReLU [18] ve SoftMax [19] fonksiyonları kullanılır. Bu işlem, doğruluğu çok düşürmeden eğitimin hızlandırılmasını sağlar.



Şekil 2.7 ReLU aktivasyon fonksiyonu [20]

2.1.1.4 Normalizasyon katmanı

Normalizasyon katmanları, giriş verisindeki tüm özellikleri aynı değer aralıklarına normalize etmek için kullanılır. Bu işlem hem ağın daha kararlı bir şekilde çalışmasını sağlar hem de eğitim işlemlerinin daha hızlı yapılmasını sağlar.

2.1.2 Sınıflandırma Bölümü

Konvolüsyon, havuzlama, aktivasyon ve normalizasyon katmanlarından sonra yer alan sınıflandırma bölümü tam bağlı katmanlardan oluşur.

2.1.2.1 Tam Bağlı Katmanlar

Tam bağlı katmandaki nöronlar, normal sinir ağlarında olduğu gibi, önceki katmandaki tüm aktivasyonlara tam olarak bağlıdır. Bu katmanın çalışması sonucunda sınıf puanları hesaplanarak, konvolüsyonel sinir ağının çıkış verisi elde edilmiş olur.

BÖLÜM 3

SİSTEM TASARIMI

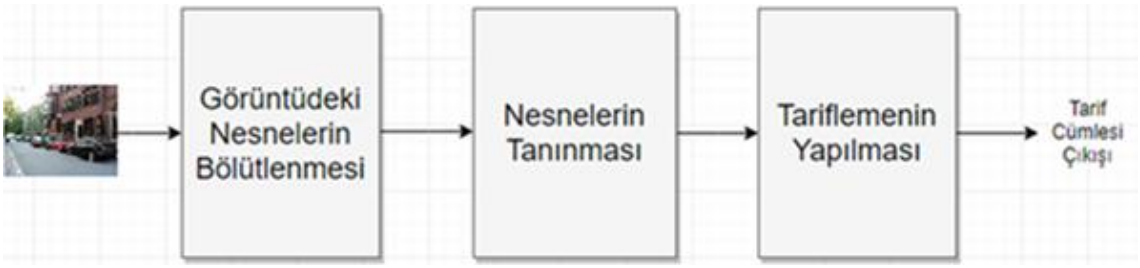
Bu çalışmada bir görüntüdeki nesneleri tanıyan ve birbirlerine göre konumlarını tarif eden bir sistem tasarlanmış ve gerçeklenmiştir.

Doğal ortamda görüntüler üzerinde birden çok nesne bulunabilmektedir. Nesne tanıma işlemi, her nesne için ayrı ayrı yapılacağı için görüntü üzerindeki tüm nesnelerin yerlerinin tespit edilerek birbirinden ayrılması gerekir. Bunun için görüntüye anlamsal bölütleme işlemi uygulanır.

Anlamsal bölütleme ile elde edilen tüm nesnelere Konvolüsyonel Sinir Ağları ile Nesne Tanıma işlemi uygulanarak her nesnenin ismi ve doğruluk olasılığı elde edilir.

Anlamsal Bölütleme ve Nesne Tanıma işlemi sonrasında nesnelerin görüntüdeki yerleri, adları ve doğruluk olasılıkları elde edilmiştir. Elde edilen bu bilgiler doğal dilde cümleler kurularak kullanıcıya aktarılır.

Sistemin ana adımları Şekil 3.1'deki blok diyagramında görülmektedir.



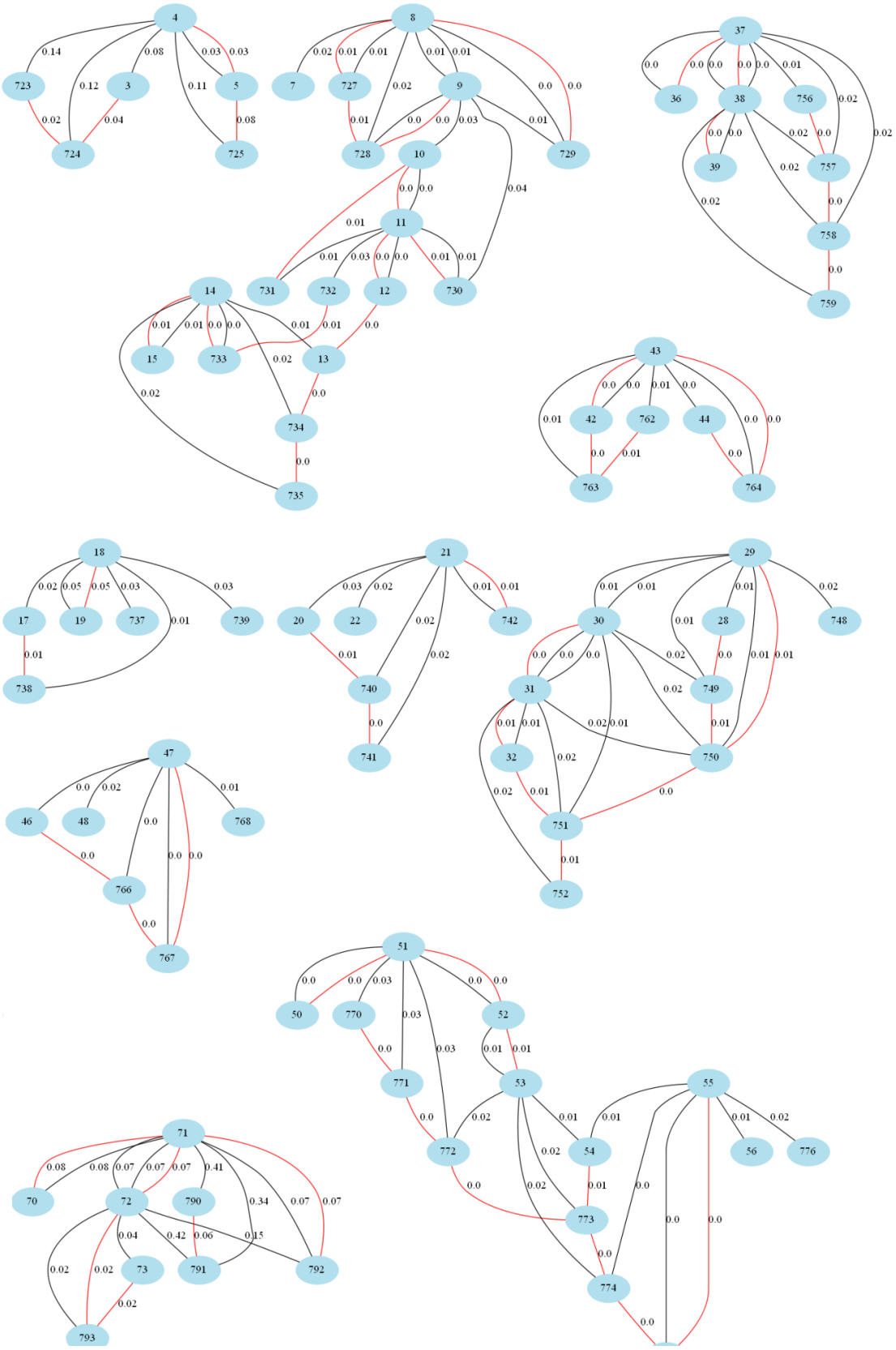
Şekil 3.1 Evrişimsel sinir ağları ile ortam tarifleme sistemi blok diyagramı

3.1 Görüntüdeki Nesnelerin Anlamsal Bölütlenmesi

Görüntü üzerinde birden fazla nesne bulunabileceğinden dolayı, nesnelerin tanınabilmesi için önce bölütlenerek arka plandan ayrılması gerekir. Bu çalışmada, anlamsal bölütleme işlemi için Grafik Tabanlı Bölütleme Algoritması [1] kullanılmıştır.

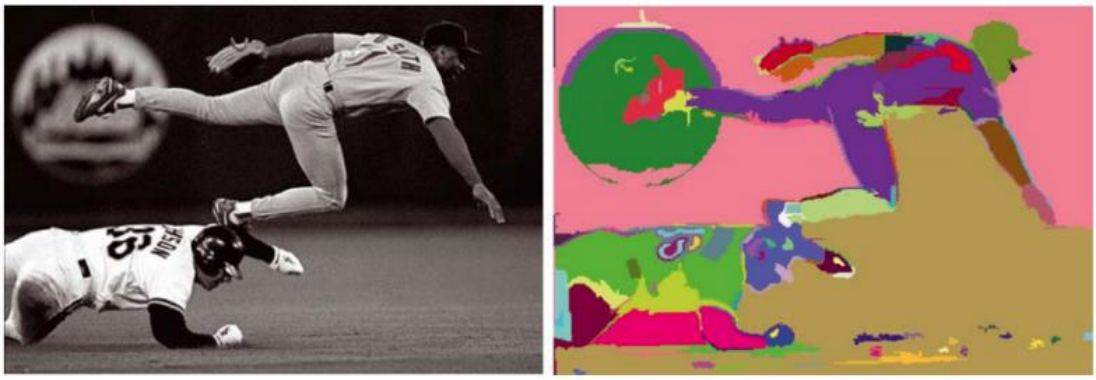
Felzenszwalb ve Huttenlocher tarafından geliştirilen Grafik Tabanlı Bölütleme Algoritması en iyi bölütleme için yüksek verimli bir çözüm yolu sunmaktadır. Bu yöntem, görüntüdeki pikselleri renk benzerliklerine göre gruplama esasına dayanır. Bu algoritmada ilk olarak görüntünün ağaç grafiği oluşturulur. Bu ağaç grafiğinde her düğüm görüntü üzerindeki bir pikseli temsil eder. Görüntü üzerindeki komşu pikselleri temsil eden düğümler kenar çizgisi ile birleştirilir. Bu iki komşu pikselin farklılık değeri hesaplanarak, düğümleri birleştiren kenar çizgilerinin üzerine yazılır. Farklılık değeri renk, yoğunluk, parlaklık gibi çeşitli parametreler kullanılarak hesaplanır. Farklılık değeri belirli bir eşik değeri üzerindeki kenar çizgileri ağaca dahil edilmez. Tüm ağaç oluşturulduktan sonra, Kruskal minimum kapsama ağacı algoritması [21] ile minimum kapsayan yol hesaplanır. Bu yol üzerindeki düğümlerin temsil ettikleri pikseller aynı bölüt olarak gruplanır. Bu algoritma oluşturulan ağaç üzerindeki düğüm ve kenar çizgisi sayısına doğru orantılı zamanda çalışarak verimli bir bölütleme yöntemi sunar.

Bu algoritmada oluşturulan ağaç grafiğine bir örnek Şekil 3.2'de verilmiştir.



Şekil 3.2 Grafik tabanlı bölütleme algoritması ağaç grafiği

Grafik tabanlı bölütleme algoritması ile bölütleme yapılan örnek bir giriş ve çıkış görüntüsü Şekil 3.3'de verilmiştir.



Şekil 3.3 Grafik tabanlı bölütleme giriş ve çıkış görüntüleri

Algoritmanın işlem adımları aşağıdaki gibidir:

1. Gauss filtresi [22] uygulanarak görüntüdeki keskin geçişler yumuşatılır.

2. Her bir piksel için, bu pikselin çevresinde yer alan 8 komşu piksel ile, 8 adet ikili piksel grubu oluşturulur. Her bir ikili piksel grubunun renk farklılıkları hesaplanır. Bu renk farklılıkları ağırlık değerini verir.

Ağırlık değeri, W_{ab} a, b pikselleri arasındaki ağırlık, C_a a pikselinin rengi, C_b b pikselinin rengi olmak üzere, eşitlik (3.1) ile ifade edilir.

$$W_{ab} = |C_a - C_b| \quad (3.1)$$

3. Tüm grupların ağırlıkları en benzerden en farklıya doğru artan şekilde sıralanır.

4. Tüm ikili piksel grupları içinde, ağırlık değeri belirli bir eşik değeri altında olan pikseller aynı bölüt olarak birleştirilir.

T, piksel birleştirme için eşik değeri olmak üzere, birleştirme koşulu, eşitsizlik (3.2) ile ifade edilir.

$$W_{ab} < T \quad (3.2)$$



(a) (b)

Şekil 3.4 (a) Orijinal görüntü, (b) Bölütleme sonrası görüntü

Şekil 3.4 (a)'daki orjinal görüntüye grafik tabanlı bölütleme uygulandığında elde edilen bölütler Şekil 3.4 (b)'de farklı renklerle gösterilmiştir. Şekil 3.4 (b)'de görüldüğü gibi nesneler birden fazla nesne parçası olarak elde edilerek aşırı bölütleme durumu oluşmuştur. Bu problemi çözmek için bölütler Seçmeli Arama Algoritması [23] ile benzerliklerine göre hiyerarşik olarak gruplanmıştır.

Seçmeli Arama Algoritmasının temel ilkesi bir görüntüdeki bölütlerin renk, desen, büyüklük ve şekil uyumluluğuna göre hiyerarşik olarak gruplanmasıdır. İşlem adımları aşağıdaki gibidir:

1. En küçük bölütten başlayarak, tüm komşu bölütlerin benzerlikleri hesaplanır. Benzerlik hesaplaması için, renk benzerliği, desen benzerliği, büyüklük benzerliği ve şekil uyumluluğu ölçütleri kullanılır. Bu ölçütlerin denklemleri aşağıda verilmiştir. Bu denklemlerde;

r_i ve r_j iki komşu bölütü ifade eder.

c_i^k ve c_j^k bu bölütlerin renk histogramlarını ifade eder. Bu renk histogramları kırmızı, yeşil, mavi renk kanallarının her biri için 25'er gruba sahip, bir boyutlu histogramlardır. Buna göre, renk benzerliği için kullanılacak n sayısı 75 olur.

t_i^k ve t_j^k bu bölütlerin desen histogramlarını ifade eder. Bu desen histogramları için kırmızı, yeşil, mavi renk kanallarının her biri için 8 yönlü Gauss türevleri alınır. Bu türevlerin, her renk kanalı ve her yön için 10 gruplu histogramı çıkarılır. Buna göre desen benzerliği için kullanılacak n sayısı 240 olur.

$büyüklük(r_i)$ ve $büyüklük(r_j)$ bölütlerin piksel büyüklüklerini,

$büyüklük(görüntü)$ görüntünün piksel büyüklüğünü,

$büyüklük(BB_{ij})$ iki bölütü çevreleyen kutunun piksel büyüklüğünü ifade eder. Buna göre;

Renk benzerliği, eşitlik (3.3) ile;

$$s_{renk}(r_i, r_j) = \sum_{k=1}^{n} min(c_i^k, c_j^k) \quad (3.3)$$

Desen benzerliği, eşitlik (3.4) ile,

$$s_{desen}(r_i, r_j) = \sum_{k=1}^{n} min(t_i^k, t_j^k) \quad (3.4)$$

Büyüklük benzerliği, eşitlik (3.5) ile,

$$s_{büyüklük}(r_i, r_j) = 1 - \frac{büyüklük(r_i) + büyüklük(r_j)}{büyüklük(görüntü)} \quad (3.5)$$

Şekil uyumluluğu, eşitlik (3.6) ile,

$$s_{şekil}(r_i, r_j) = 1 - \frac{büyüklük(BB_{ij}) - büyüklük(r_i) - büyüklük(r_j)}{büyüklük(görüntü)} \quad (3.6)$$

Benzerlik, eşitlik (3.7) ile ifade edilir.

$$s(r_i, r_j) = s_{renk}(r_i, r_j) + s_{desen}(r_i, r_j) + s_{büyüklük}(r_i, r_j) + s_{şekil}(r_i, r_j) \quad (3.7)$$

2. Benzerlikleri belirli bir eşik değerinden büyük olan bölütler birleştirilir.

3. Eşik değerinden büyük benzerliği olan bölüt kalmayana kadar işlem tekrarlanır.

Bu işlem adımları sonucunda, bölütlerin sol üst köşelerinin koordinatları, genişlikleri ve yükseklikleri elde edilir.



(a) (b)

Şekil 3.5 (a) Hiyerarşik gruplanmış bölütler, (b) Bölütleme sonuç görüntüsü

Şekil 3.4 (b)'deki aşırı bölütlenmiş görüntüye Seçmeli Arama işlemi uygulandığında elde edilen hiyerarşik gruplanmış bölütler Şekil 3.5 (a)'da farklı renkler ile gösterilmiştir. Şekil 3.5 (b)'de Seçmeli Arama algoritması sonucunda elde edilen bölütler görüntü üzerinde gösterilmiştir.

Bu işlemler tamamlandıktan sonra aşağıdaki iyileştirme adımları uygulanır:

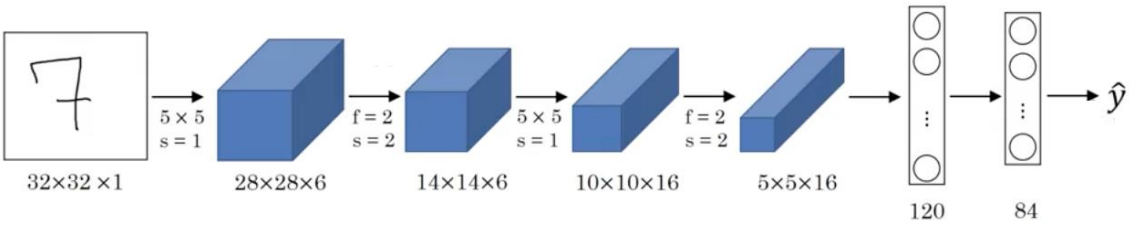
1. Görüntüde, minimum genişlik eşik değerinden daha az genişliğe sahip ve maksimum genişlik eşik değerinden daha fazla genişliğe sahip olan nesneler ile minimum yükseklik eşik değerinden daha az yüksekliğe sahip ve maksimum yükseklik eşik değerinden daha fazla yüksekliğe sahip olan nesneler elenir. Elenecek nesnelerin büyüklüğü deneysel olarak belirlenmiştir. Bu işlemin amacı, uygulamanın kapsamında olmayan çok küçük ve çok büyük nesnelerin elenmesidir.

2. En-boy veya boy-en oranı eşik değerinin üzerindeki nesneler elenir. Bu işlemin amacı, görüntüde ufuk çizgisi, yol şerit çizgileri, nesnelerin köşe çizgileri gibi renk geçişi yoğun olan fakat gerçekte nesne olmayan bölgelerin nesne olarak algılanmasının engellenmesidir.

3.2 Konvolüsyonel Sinir Ağları ile Nesne Tanıma

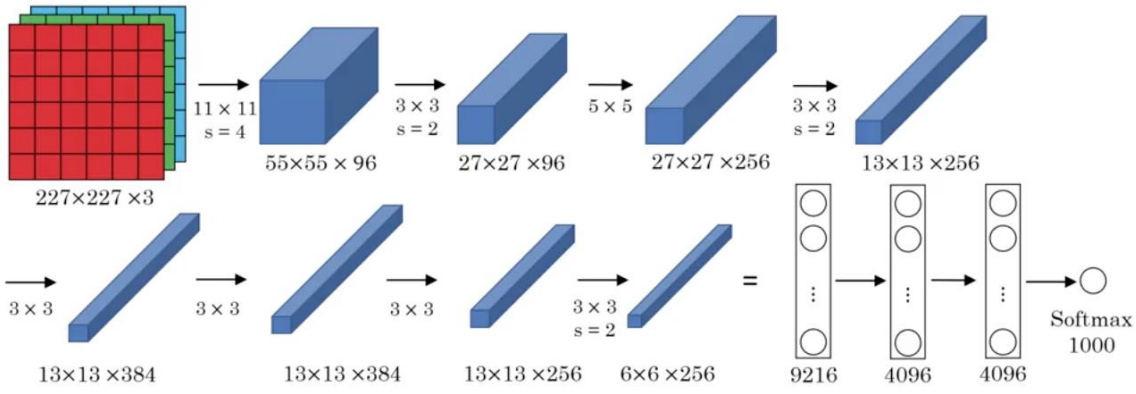
Yapay sinir ağları ile başlayıp günümüzde derin konvolüsyonel sinir ağları ile devam eden derin öğrenme yöntemlerinde şu ana kadar bir çok model ortaya çıkmıştır.

LeNet [24] sinir ağı, 1998 yılında Yann LeCun tarafından ortaya atılmıştır. Ağın geliştirilme amacı, el yazısı ile yazılan sayıların otomatik olarak okunmasıdır. 32x32 boyutlu giriş görüntüleri 5x5 boyutlu filtre ile filtrelenir. Ardından havuzlama yapılır. Ardından 120 düğümlü tam bağlı katman ve bunu takip eden 84 düğümlü tam bağlı katmandan geçirilir. Sonuç olarak 0-9 arasında rakam tahmini yapılır. 0 ile 9 arası el yazısı rakamlardan oluşan MNIST [25] verisetindeki rakamları tanıyarak %98.5 doğruluk oranını elde etmiş ve bu alandaki ilk başarılı sonuçları vermiştir. LeNet Sinir Ağı Yapısı Şekil 3.6'da verilmiştir.



Şekil 3.6 LeNet sinir ağı yapısı

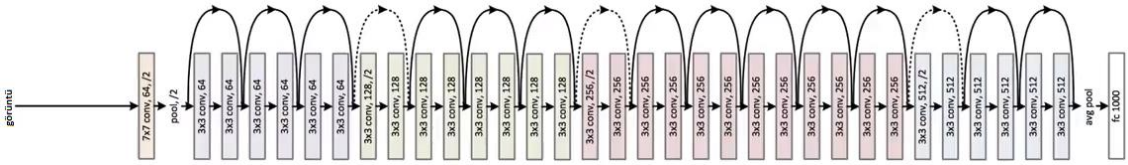
AlexNet [11] ağı, 2012 yılında Krizhevsky, Sutskever ve Hinton tarafından ortaya atılmıştır. 227x227 boyutlu giriş görüntüleri 11x11 boyutlu konvolüsyon işleminden geçirilir. Elde edilen 55x55 boyutlu veri 3x3 maksimum havuzlama işleminden geçirilir. En son tam bağlı katmandan geçerek, tahminleme yapılır. Maksimum havuzlama ve ReLU aktivasyon fonksiyonu kullandığı için LeNet modelinden ayrışmaktadır. ImageNet veriseti üzerinde %15.3 ilk 5 hata payı yakalamıştır. AlexNet Sinir Ağı Yapısı Şekil 3.7'de verilmiştir.



Şekil 3.7 AlexNet sinir ağı yapısı

ResNet [26] ağı, bir çok konvolüsyonel ve havuzlama katmanlarının bulunmasından dolayı, sinir ağlarının derin ağlar olarak adlandırılmasına sebep olmuştur. ReLU

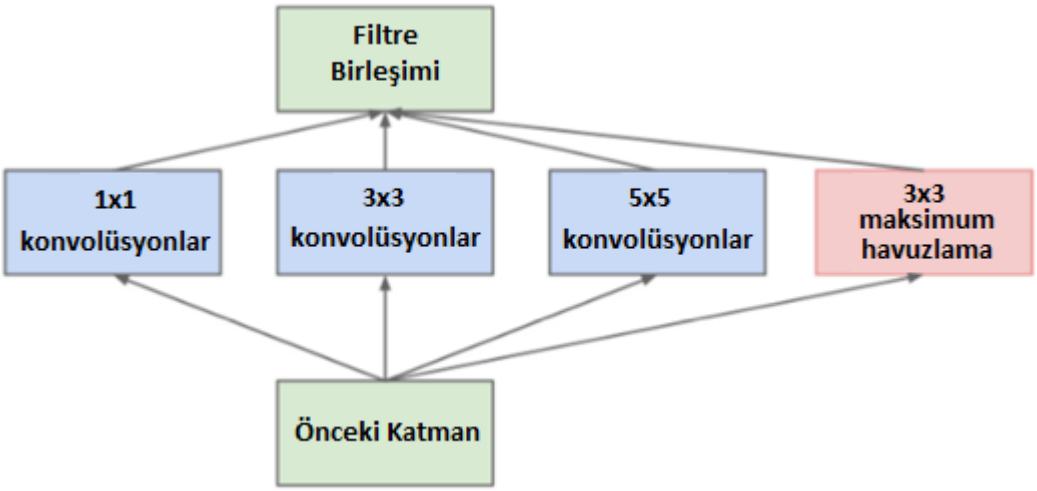
aktivasyon fonksiyonunu kullanmaktadır. ILSVRC 2015 sınıflandırma yarışmasında, ilk 5 hata oranı olarak %3.57 oran elde edip 1. olarak başarısını ispat etmiştir. ResNet Sinir Ağı Yapısı Şekil 3.8'de verilmiştir.



Şekil 3.8 ResNet sinir ağı yapısı

Inception V1 [27] ağı, Google tarafından geliştirilmiş olup, ILSVRC 2014 yarışmasında ilk 5 hata oranı olarak %6.67 oran elde edip 1. olarak başarısını göstermiştir.

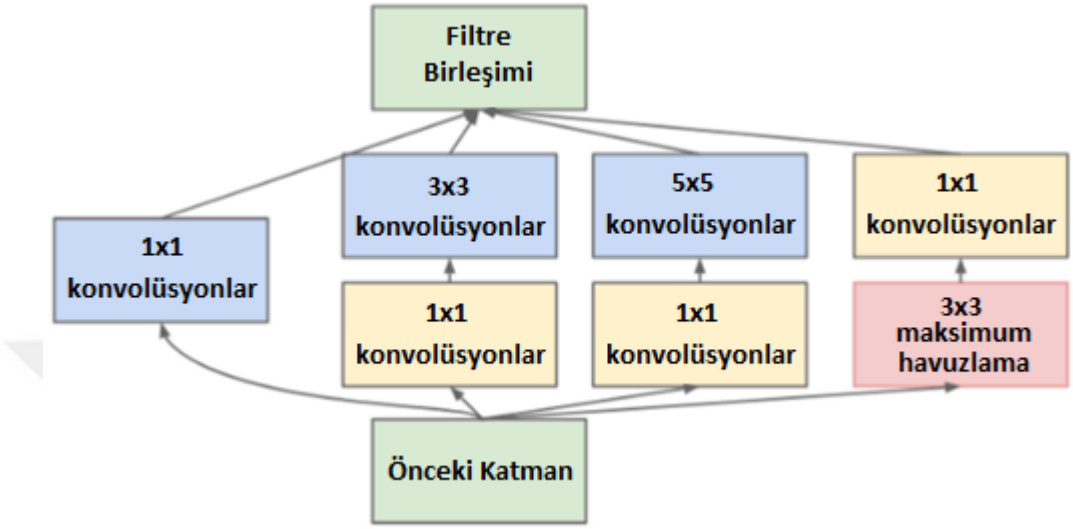
Inception V1 adı ile ortaya çıkan ilk inception modülünde, 1x1, 3x3 ve 5x5 boyutlu konvolüsyon katmanları ve buna ek olarak 3x3'lük bir maksimum havuzlama katmanı bulunmaktadır. Bu işlemler sonucunda oluşan veriler sonraki katmana iletilmek üzere birleştirilmektedir. Bu modele saf inception modülü adı verilmektedir.



Şekil 3.9 Saf Inception modülü [27]

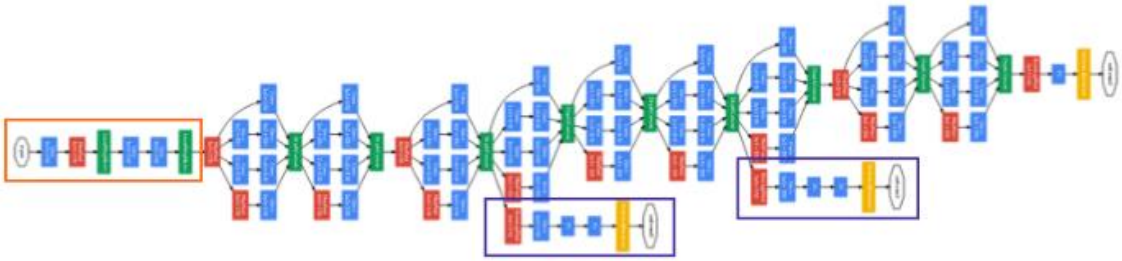
Derin sinir ağlarındaki büyük boyutlu konvolüsyon işlemleri çok hesaplama gücü gerektirdiğinden maliyetlidir. Bu sebeple saf inception modülünde bulunan 3x3 ve 5x5 boyutlu konvolüsyon işlemleri ve 3x3 boyutlu maksimum havuzlama işlemleri büyük işlem gücü maliyeti oluşturmaktadır. Bu sebeple 3x3 ve 5x5 boyutlu konvolüsyon işlemlerinin öncesine 1x1 boyutlu konvolüsyon işlemi ve 3x3 boyutlu maksimum

havuzlama işleminin sonrasına 1x1 boyutlu konvolüsyon işlemi eklenerek, boyut azaltılması, buna bağlı olarak, ağın karmaşıklığının azaltılıp, işlemlerin daha hızlı yapılabilmesi sağlanmıştır. Böylece Inception V1 modülü ortaya çıkmıştır.



Şekil 3.10 Inception v1 modülü [27]

9 adet Inception V1 modülü art arda sıralanarak Inception V1 mimarisi oluşturulmuştur. Bu mimaride ortalama havuzlama kullanılarak, boyutların düşürülmesi suretiyle, parametre sayısından büyük ölçüde tasarruf sağlanmaktadır.

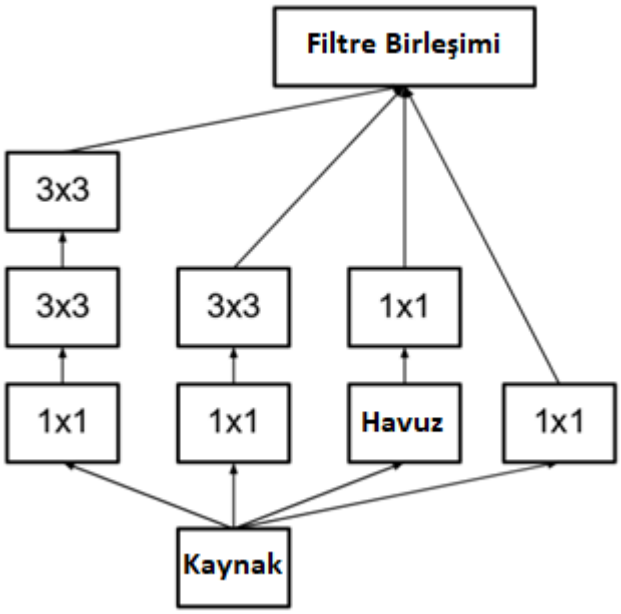


Şekil 3.11 Inception v1 mimarisi [27]

Inception V1 mimarisindeki doğruluğu yükseltmek ve hesaplama karmaşıklığını azaltmak için yapılan yeni çalışmalar ile Inception V2 [12] mimarisi oluşturulmuştur.

Inception V1 mimarisinde, hesaplama karmaşıklığını azaltmak üzere, büyük boyutlu konvolüsyon işlemlerinden önce kullanılan 1x1 boyutlu konvolüsyon işlemleri, bilgilerin kaybolmasına ve bu doğrultuda doğruluğun azalmasına sebep olmaktadır. Yeni yapıda 5x5 boyutlu büyük bir konvolüsyon işlemi yerine 2 katmanlı 3x3 boyutlu 2 konvolüsyon

işlemi kullanılmıştır. Bu işlem 5x5 boyutlu bir konvolüsyon işlemine göre %33'lük işlem gücü tasarrufu sağlamakla beraber, bilgi kaybının azaltılmasını sağlamaktadır. Eklenen unutma katmanları ile çıkış verilerinin belirli bir bölümü elenerek, karmaşıklığın azaltılması sağlanmıştır.



Şekil 3.12 Inception v2 modülü [12]

Inception V3 mimarisinde, Inception V2 mimarisine ek olarak toplu normalizasyon ve unutma katmanları eklenmiştir. Toplu normalizasyon işlemi özellik haritasındaki değerlerin normalize edilerek işlem karmaşıklığınının azaltılması ve işlem hızının artırılmasını sağlamaktadır. Unutma işlemleri ise rastgele seçilen nöronların kullanılmamasını sağlayarak, aşırı uyumluluk durumunun azaltılmasını sağlamaktadır.

Önceki Inception modüllerinde olduğu gibi inception modüllerinin art arda sıralanmasıyla Inception V3 mimarisi oluşmuştur. Bu mimari yüksek performans ile birlikte düşük işlem gücü gereksinimi ile büyük fayda sağlamaktadır. Inception V3 mimarisi, ILSVRC 2015 yarışmasında %3.46 hata payı elde ederek başarısını göstermiştir [12].



Şekil 3.13 Inception v3 sinir ağı yapısı

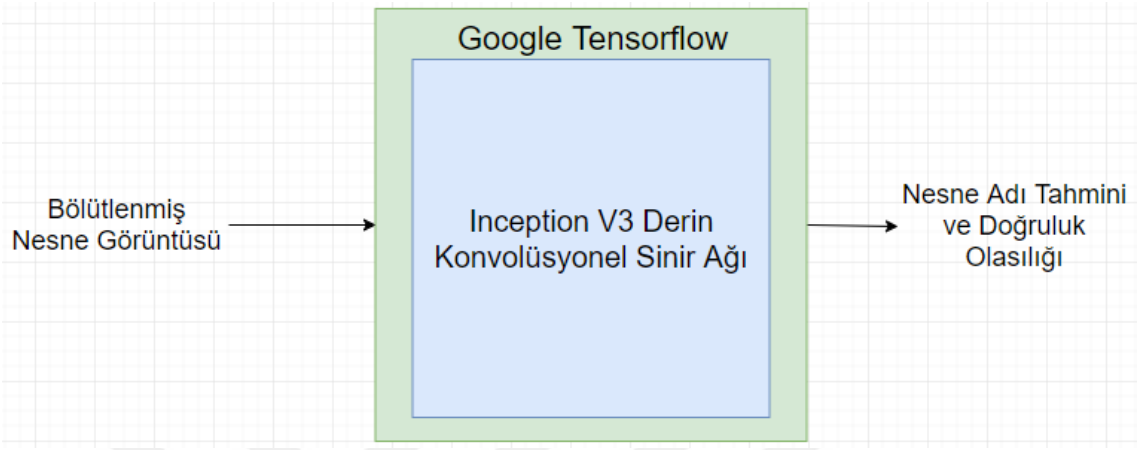
Görüntülerden çıkarılan her bir bölütte bulunan nesnelerin tanınması için her bir bölüt, bu Derin Konvolüsyonel Sinir Ağına giriş verisi olarak verilerek, nesne adı tahminleri ve doğruluk olasılıkları elde edilir.

Inception V3 algoritmasında, nesneleri sınıflandırmada kullanmak amacıyla, ImageNet 2012 veri setindeki 1000 sınıfa ait milyonlarca fotoğraf ile özellik çıkarımı işlemi yapılarak bir DKSA grafik modeli oluşturulmuştur. Özellik çıkarımı ile oluşturulan bu modelin eğitim işlemleri, işlem gücü ve işlem süresi bakımından çok maliyetlidir. Özellik çıkarımı ve eğitim işlemleri yapılmış bir model, farklı ancak ilişkili işler için tekrar özellik çıkarımı ve eğitim ihtiyacı olmadan tekrar tekrar kullanılabilmektedir. Bu yöntem ile, Inception V3 için oluşturulan Derin Konvolüsyonel Sinir Ağı modeli herhangi bir zamanda tekrardan hesaplama ihtiyacı olmadan kullanılabilmektedir.

Inception V3 Sinir Ağı modelinin sınıflandırma bölümünde tam bağlı katmanlar ve softmax vektörü bulunmaktadır. Softmax vektörü, normalize edilmiş 1000 sınıflı bir sınıflandırma özellik vektörüdür. Görüntüdeki bölütlenmiş nesneler bu sinir ağına girdi olarak verilerek, nesne adı tahminleri ve doğruluk olasılıkları elde edilir. Nesne adı tahminleri doğruluk olasılıklarına göre sıralanarak en yüksek olasılığa sahip olan tahmin ele alınır. Bu tahminin olasılık değeri, nesne tanıma olasılığı eşik değerinden yüksek ise nesne adı olarak kabul edilir, değil ise bu nesne için tahmin yapılmaz. Nesne tanıma olasılığı eşik değerinin uygulanmasının amacı, çok düşük olasılıklarla yapılan tahminlerin gözardı edilerek, sadece belirli bir olasılık değerinden yüksek olan tahminlerin gösterilmek istenmesidir. Inception V3 ağından elde edilen nesne adları İngilizcedir. Bu çalışmada nesne adlarının Türkçe karşılıkları kullanılacağından dolayı, ağda bulunan

1000 adet İngilizce nesne adı Türkçe'ye çevrilerek, bir sözlük oluşturulmuştur. Böylece ağdan elde edilen İngilizce nesne adları için bu sözlüğe bakılarak Türkçe karşılıkları bulunmuş ve kullanılmıştır.

Inception V3 ile yapılan nesne tanıma işleminin blok diyagramı Şekil 3.14'de verilmiştir.



Şekil 3.14 Inception v3 ile nesne tanıma

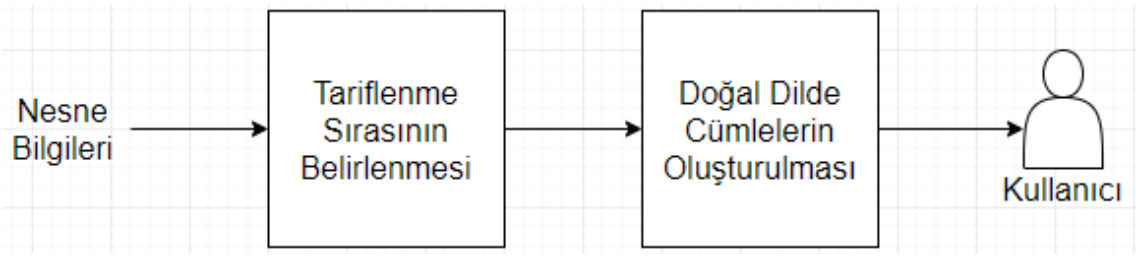
3.3 Ortam Tarifinin Yapılması

Ortam tarifinin yapılması işlemi, görüntü bölütleme ve nesne tanıma işlemleri sonucunda elde edilen bilgilerin kullanıcıya aktarılmasıdır. Bu bilgiler görüntünün üzerinde bulunan nesnelerin konum, büyüklük, isim ve doğruluk olasılığı bilgileridir.

Ortam tarifinin yapılması işlemi 2 aşamadan oluşmaktadır.

1. Nesnelerin tariflenme sırasının belirlenmesi.

2. Doğal dilde cümlelerin oluşturulması.



Şekil 3.15 Ortam tarifinin yapılması işlem adımları

3.3.1 Nesnelerin Tariflenme Sırasının Belirlenmesi

Bu çalışmada nesnelerin kullanıcıya aktarılma sırası için 2 alternatif üzerinde çalışılmıştır. Birinci alternatif görüntü üzerinde görüntü kaynağına en yakın olan nesneden başlamak, ikinci alternatif görüntü üzerindeki en büyük nesneden başlamaktır. Bu çalışmada 2 boyutlu görüntüler üzerinde çalışıldığı için derinlik bilgisi bulunmamaktadır. Bu sebeple ikinci alternatif seçilmiştir.

Algoritmanın işlem adımları aşağıdaki gibidir:

1. Görüntü üzerindeki en büyük nesne bulunur ve ilk sıraya eklenir.

2. Bu nesnenin sırayla sol, üst, sağ ve alt bölgelerinde bulunan nesneler taranarak, en büyük olan nesne sıraya eklenir.

3. Listeye son eklenen nesne için Adım 2'ye gidilerek, işlemler tekrarlanır.

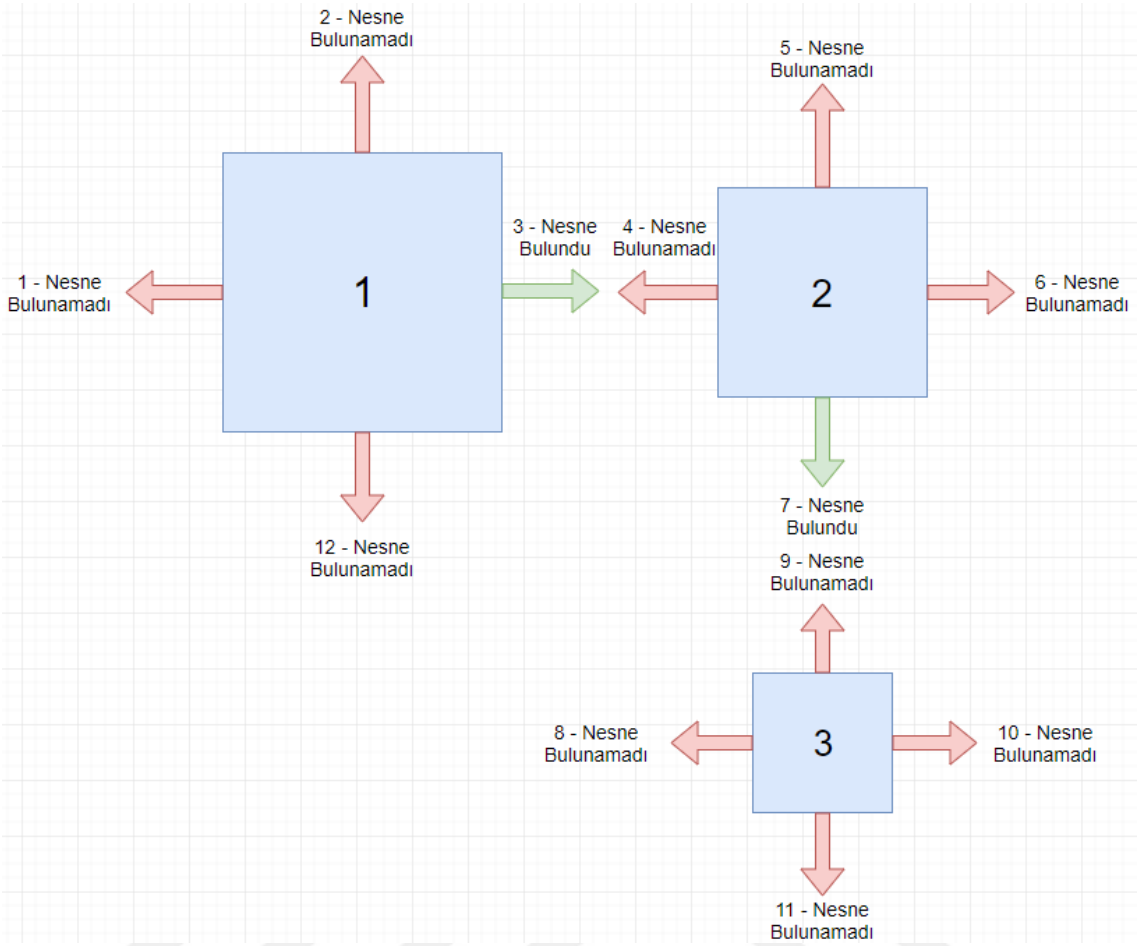
4. Bir nesne etrafında sıraya eklenmemiş nesne kalmadığında bir üst seviyeye çıkılarak, işleme devam edilir.

5. Görüntü üzerindeki tüm nesneler sıraya eklenınceye kadar işleme devam edilir.

Tariflenme sırasının belirlenmesine örnek olarak Şekil 3.16'daki orjinal görüntüye işlemler sırasıyla uygulandığında, algoritmanın çalışma adımları Şekil 3.17'de gösterilmiştir.



Şekil 3.16 Orjinal görüntü



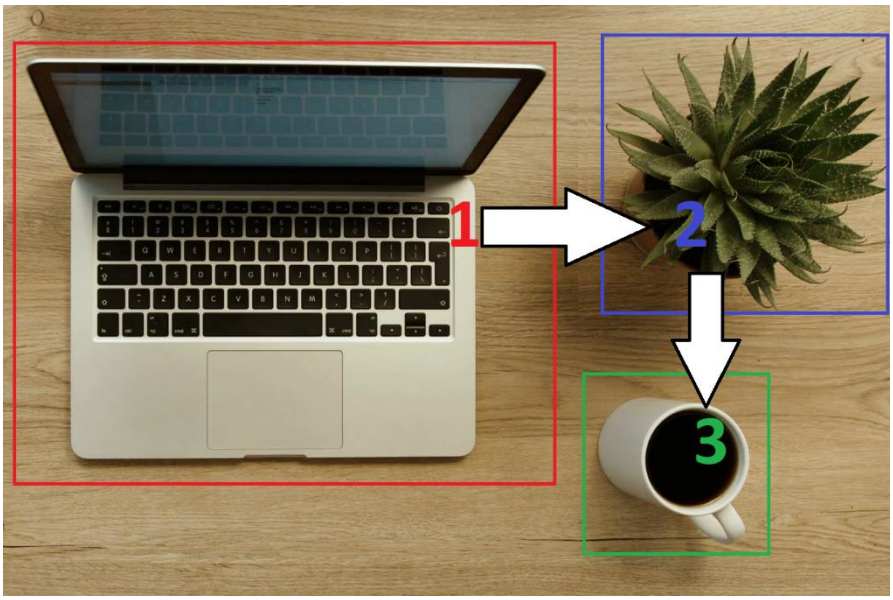
Şekil 3.17 Tarifleme sırasının belirlenmesi işlem adımları

Kırmızı oklar, o yönde sıraya eklenmemiş bir nesne bulunmadığını, yeşil oklar ise o yönde sıraya eklenmemiş bir nesne bulunduğunu ifade etmektedir. Algoritmanın işlem adımları sırası ile 1'den 12'ye kadar numaralandırılmıştır.

Bu işlem sonucunda elde edilen sıralama aşağıdaki gibidir:

1- Bilgisayar 2- Çiçek 3- Fincan

Yapılan sıralamanın görüntü üzerinde işaretlenmiş hali Şekil 3.18'de gösterilmiştir.



Şekil 3.18 Nesnelerin tariflenme sırası

3.3.2 Doğal Dilde Cümlelerin Oluşturulması

Bir nesneyi kullanıcıya tariflemek için nesnenin konumu <nesne konumu> ve nesnenin adı <nesne adı> ve nesnenin kendisinden bir üst seviyede bulunan nesne adı <bir üst seviyedeki nesne adı> olarak ifade edilmek üzere,

ilk nesne için "<nesne konumu>'da bir <nesne adı> var." şeklinde,

sonraki nesneler için "<bir üst seviyedeki nesne adı> <nesne konumu>'da bir <nesne adı> var." şeklinde cümle kurulur. Nesnelerin konumları 2 boyutlu olarak değerlendirilerek, sol, sağ, üst, alt şeklinde ifade edilir.

Buna göre Şekil 3.18'deki görüntü için tarifleme cümleleri aşağıdaki gibi oluşmuştur:

1. Solda bir bilgisayar var.

2. Bilgisayar sağında bir çiçek var.

3. Çiçek altında bir fincan var.

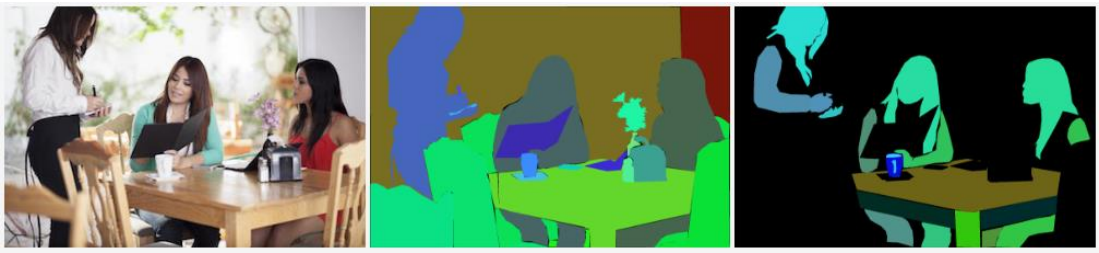
BÖLÜM 4

UYGULAMA

Tasarlanan sistem, belirlenen veriseti üzerinde çalıştırılarak, uygulamanın görüntü bölütleme, nesne tanıma ve ortam tarifleme başarıları değerlendirilmiştir.

4.1 Veriseti

Bu çalışmada uygulama testleri için ADE20K [29] veriseti kullanılmıştır. Bu veri setinde, nesneler içeren kapsamlı bir şekilde etiketlenmiş 20.210 adet eğitim, 2000 adet doğrulama görüntüsü bulunmaktadır. Gökyüzü, yol, çimen, insan, araba, yatak gibi ayrı nesneler içeren toplam 150 anlamsal sınıf vardır.



Şekil 4.1 ADE20K verisetinden örnek bir görüntü

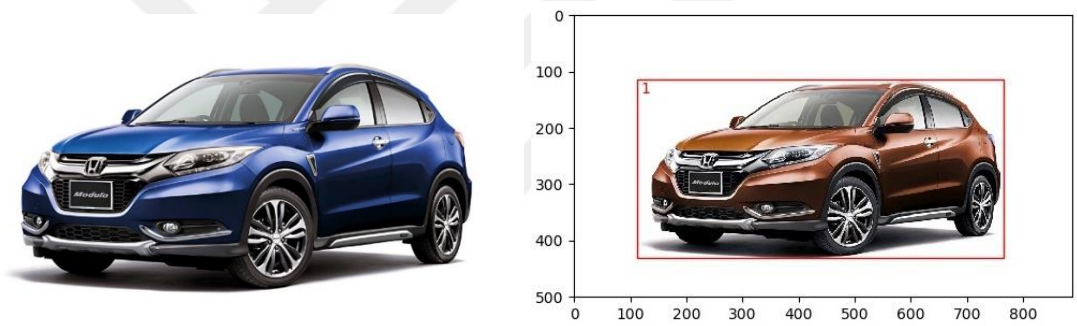
4.2 Sinir Ağının Eğitimi

Bu ağ, her birinde NVidia Kepler GPU bulunan 50 adet dağıtık makinede, 100 epoch ve 32 batch boyutu ile eğitilmiştir. Bu ağın eğitilmesi çok zaman ve çok işlem gücü gerektirdiğinden dolayı, ağ bir kere eğitildikten sonra, eğitilmiş model bilgileri kaydedilerek, ağın yeniden eğitilmesine gerek kalmadan, tekrar tekrar kullanılması sağlanmıştır [30].

4.3 Uygulama Adımları

Uygulama için ADE20K verisetinden 400 adet görüntü seçilmiştir. Görüntüler, görüntü içerisindeki nesne sayısı, nesnelerin büyüklükleri, nesnelerin arka plan ile olan renk farklılıkları göz önünde bulundurularak homojen bir şekilde seçilmiştir. İnsan gözüyle yapılan tariflemeler ile sistemin yaptığı tariflemeler karşılaştırılmıştır.

İçerisinde bir adet araba olan ve arka plandan kolay bir şekilde ayrılabilen bir görüntü üzerinde uygulama yapıldığında, araba nesnesi net bir şekilde bölütlenmiş, nesne tanıma başarıyla gerçekleşmiş ve tarifleme cümleleri doğru bir şekilde kurulmuştur. Uygulamanın nesne sayısı az ve arka plandan kolay ayrılabilme durumunda başarılı sonuç verdiği gözlemlenmiştir. Giriş görüntüsü Şekil 4.2 (a)'da, çıkış görüntüsü Şekil 4.2 (b)'de, tahmin edilen nesne adları Şekil 4.3 (a)'da ve tarifleme için kurulan cümleler Şekil 4.3 (b)'de verilmiştir.



(a) (b)

Şekil 4.2 (a) Giriş görüntüsü 1, (b) Çıkış görüntüsü 1

```
1  1. spor araba - Olasılık : 0.407714
```

(a)

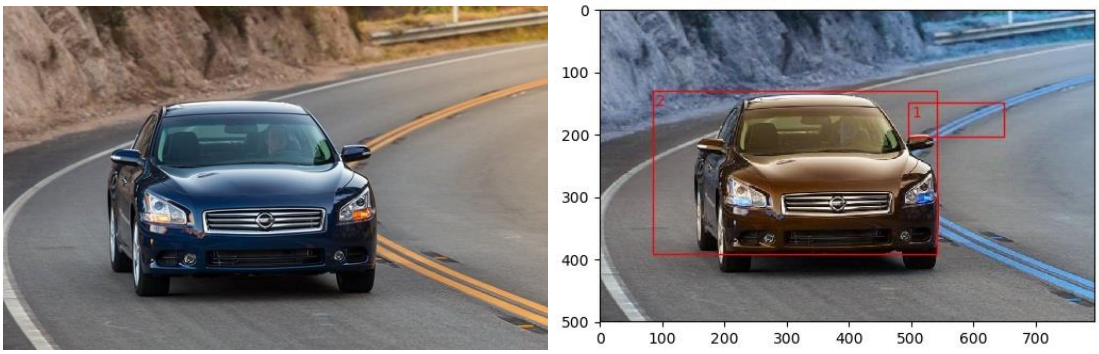
```
1  1. Ortada bir spor araba var.
```

(b)

Şekil 4.3 (a) Tahmin edilen nesne adları 1, (b) Tarifleme için kurulan cümleler 1

İçerisinde bir yol ve bu yol üzerinde bulunan bir araba olan bir görüntü üzerinde yapılan uygulamada, yol çizgisi ve araba nesneleri net bir şekilde bölütlenmiş, nesne tanıma işlemi başarıyla gerçekleşmiş ve tarifleme cümleleri doğru bir şekilde kurulmuştur. Uygulamanın nesne sayısı az ve arka plandan ayrılabilme durumu orta seviyede olduğunda başarılı sonuçlar verdiği gözlemlenmiştir. Giriş görüntüsü Şekil 4.4

(a)'da, çıkış görüntüsü Şekil 4.4 (b)'de, tahmin edilen nesne adları Şekil 4.5 (a)'da ve tarifleme için kurulan cümleler Şekil 4.5 (b)'de verilmiştir.



(a) (b)

Şekil 4.4 (a) Giriş görüntüsü 2, (b) Çıkış görüntüsü 2

```
1. yol çizgisi - Olasılık : 0.363849
2. spor araba - Olasılık : 0.520972
```

(a)

```
2. Ortada bir spor araba var.
1. Spor araba sağında bir yol çizgisi var.
```

(b)

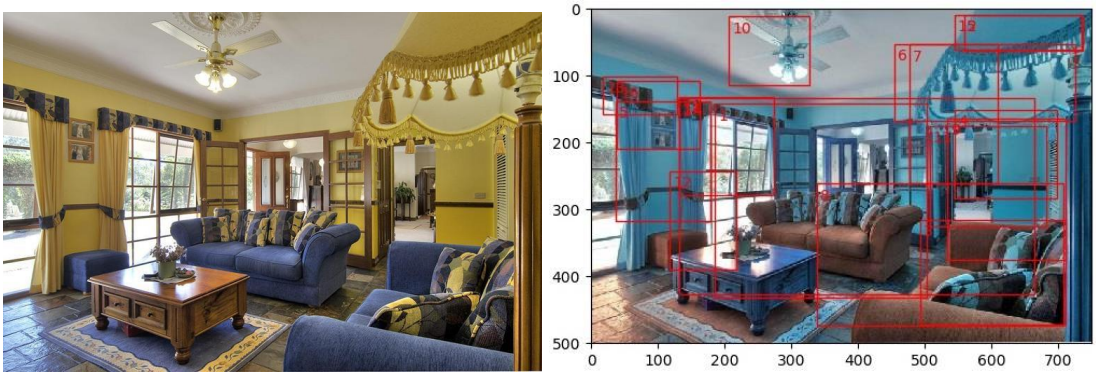
Şekil 4.5 (a) Tahmin edilen nesne adları 2, (b) Tarifleme için kurulan cümleler 2

4.4 Uygulama Başarısı

Seçilen görüntüler üzerinde çalıştırılan uygulama için başarı oranları hesaplanmıştır. Görüntülerde bölüt olarak tespit edilmesi gereken bölümlerin, bölüt olarak tespit edilme başarısı %94 olarak ölçülmüştür. Görüntülerde bölüt olarak tespit edilen bölümlerin, gerçekten bir nesne içeren doğru bir bölüt olma başarısı %92 olarak ölçülmüştür. Bölütler içerisinde bulunan nesne adlarının doğru tahmin edilme başarısı %95 olarak ölçülmüştür.

İçerisinde bir çok nesne bulunan bir evin salonunun olduğu bir görüntü üzerinde yapılan uygulamada, salon içerisinde bir çok bölüt tespit edilmiştir. Birbirine yakın renklere sahip nesnelerin aynı nesne olarak belirlendiği durumlar gözlenmiştir. Tespit edilen bazı nesnelerin isimlerinin yanlış tahmin edildiği gözlemlenmiştir. Buna göre görüntü üzerindeki nesne sayısı arttıkça, nesnelerin birbiriyle kesişim durumları arttığından dolayı bölütlemede hata yapılabildiği gözlemlenmiştir. Bölütlemede yapılan hatanın nesne tanıma başarısını etkilediği gözlemlenmiştir. Giriş görüntüsü Şekil 4.6 (a)'da, çıkış

görüntüsü Şekil 4.6 (b)'de, tahmin edilen nesne adları Şekil 4.7 (a)'da ve tarifleme için kurulan cümleler Şekil 4.7 (b)'de verilmiştir.



(a) (b)

Şekil 4.6 (a) Giriş görüntüsü 3, (b) Çıkış görüntüsü 3

```
1. kanepe - Olasılık : 0.310345
2. panço - Olasılık : 0.484797
3. kanepe - Olasılık : 0.475973
4. güneşlik - Olasılık : 0.539536
5. duş perdesi - Olasılık : 0.432362
6. abajur - Olasılık : 0.830833
7. dört direkli karyola - Olasılık : 0.431508
8. binek araba - Olasılık : 0.573372
9. güneşlik - Olasılık : 0.292562
10. vantilatör - Olasılık : 0.364834
11. dört direkli karyola - Olasılık : 0.229591
12. kapı - Olasılık : 0.329587
13. ticari araç - Olasılık : 0.208284
14. güneşlik - Olasılık : 0.20998
15. panço - Olasılık : 0.243016
16. duş perdesi - Olasılık : 0.126368
17. dört direkli karyola - Olasılık : 0.172819
```

(a)

```
1. Ortada bir kanepe var.
8. Kanepe sağında bir binek araba var.
3. Kanepe sağında bir kanepe var.
6. Kanepe sağında bir abajur var.
7. Kanepe sağında bir dört direkli karyola var.
5. Kanepe solunda bir duş perdesi var.
4. Kanepe sağında bir güneşlik var.
12. Kanepe solunda bir kapı var.
10. Kanepe üstünde bir vantilatör var.
2. Kanepe üstünde bir panço var.
```

(b)

Şekil 4.7 (a) Tahmin edilen nesne adları 3, (b) Tarifleme için kurulan cümleler 3

İçerisinde çim biçme makinesi, uçak, ağaçlar ve bulutlar bulunan bir görüntü üzerinde yapılan uygulamada, 6 adet bölüt tespit edilmiştir. 2 adet nesnenin 2'şer defa bölütlendiği görülmüştür. Bu duruma bu nesnelerin sahip olduğu gölgelerin sebep olduğu gözlemlenmiştir. Ancak bu nesnelerin isimleri doğru tespit edilmiştir. Geri kalan nesneler doğru bir şekilde bölütlemiş ve adları doğru bir şekilde tahmin edilmiştir. Konumlarına göre tarifleme cümleleri kurulmuştur. Bu örnekte nesnelerin sahip olduğu gölgelerin bölütleme başarısını etkileyebildiği ancak nesne tanıma başarısını etkilemediği görülmüştür. Giriş görüntüsü Şekil 4.8 (a)'da, çıkış görüntüsü Şekil 4.8 (b)'de, tahmin edilen nesne adları Şekil 4.9 (a)'da ve tarifleme için kurulan cümleler Şekil 4.9 (b)'de verilmiştir.



(a) (b)

Şekil 4.8 (a) Giriş görüntüsü 4, (b) Çıkış görüntüsü 4

```
1. harman makinesi - Olasılık : 0.184795
2. çim biçme makinesi - Olasılık : 0.811157
3. çim biçme makinesi - Olasılık : 0.80201
4. uçak - Olasılık : 0.67994
5. bizon - Olasılık : 0.236537
6. bulut - Olasılık : 0.471868
```

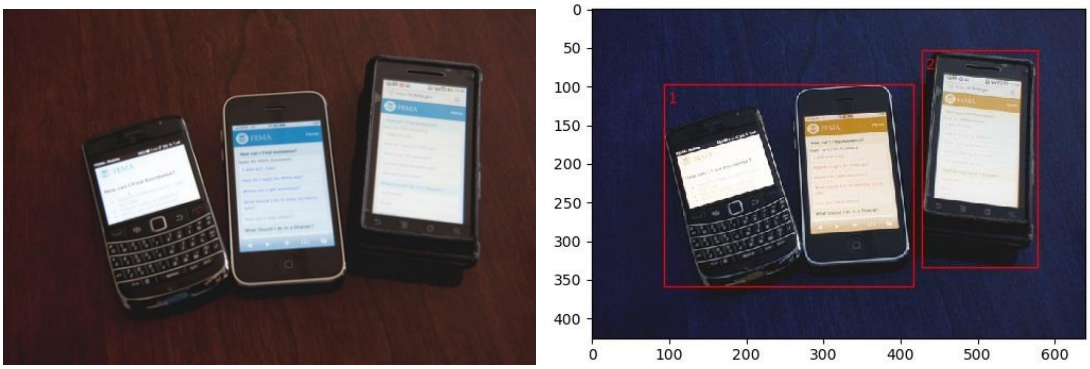
(a)

```
4. Ortada bir uçak var.
3. Uçak altında bir çim biçme makinesi var.
2. Çim biçme makinesi sağında bir çim biçme makinesi var.
6. Uçak üstünde bir bulut var.
```

(b)

Şekil 4.9 (a) Tahmin edilen nesne adları 4, (b) Tarifleme için kurulan cümleler 4

İçerisinde 3 adet cep telefonu bulunan bir görüntü üzerinde yapılan uygulamada, görüntüde 3 adet nesne bulunmasına rağmen 2 adet bölüt tespit edilmiştir. 2 adet nesnenin tek bir bölüt olarak bölütlendiği görülmüştür. Bu duruma bu nesnelerin birbirine temas etmesinin sebep olduğu gözlemlenmiştir. Ancak buna rağmen nesnelerin isimleri doğru tespit edilmiştir. Konumlarına göre tarifleme cümleleri kurulmuştur. Bu örnekte birbirine temas eden nesnelerin bölütleme başarısını etkileyebildiği ancak nesne tanıma başarısını etkilemediği görülmüştür. Giriş görüntüsü Şekil 4.10 (a)'da, çıkış görüntüsü Şekil 4.10 (b)'de, tahmin edilen nesne adları Şekil 4.11 (a)'da ve tarifleme için kurulan cümleler Şekil 4.11 (b)'de verilmiştir.



(a) (b)

Şekil 4.10 (a) Giriş görüntüsü 5, (b) Çıkış görüntüsü 5

```
1. iPod - Olasılık : 0.345789
2. el bilgisayarı - Olasılık : 0.479614
```

(a)

```
1. Ortada bir iPod var.
2. iPod sağında bir el bilgisayarı var.
```

(b)

Şekil 4.11 (a) Tahmin edilen nesne adları 5, (b) Tarifleme için kurulan cümleler 5

İçerisinde saray, yol, arabalar, trafik ışıkları ve insanlar bulunan bir görüntü üzerinde yapılan uygulamada 5 adet bölüt tespit edilmiştir. Bu 5 bölüt için yapılan nesne tanıma işleminde, 4 adet nesne için, nesne tanıma olasılığı eşik değerinden yüksek olasılıklı tahminler yapılmış, 1 adet nesne için yapılan tahmin bu eşik değerinin altında kalmıştır. Elde edilen bu 4 nesne için, konumlarına göre tarifleme cümleleri kurulmuştur. Giriş görüntüsü Şekil 4.12 (a)'da, çıkış görüntüsü Şekil 4.12 (b)'de, tahmin edilen nesne adları Şekil 4.13 (a)'da ve tarifleme için kurulan cümleler Şekil 4.13 (b)'de verilmiştir.



(a) (b)

Şekil 4.12 (a) Giriş görüntüsü 6, (b) Çıkış görüntüsü 6

```
1. kamyon - Olasılık : 0.188809
2. saray - Olasılık : 0.164259
3. trafik ışığı - Olasılık : 0.971816
4. trafik ışığı - Olasılık : 0.964157
5. limuzin - Olasılık : 0.41877
```

(a)

```
2. Ortada bir saray var.
3. Saray sağında bir trafik ışığı var.
5. Saray altında bir limuzin var.
4. Saray sağında bir trafik ışığı var.
```

(b)

Şekil 4.13 (a) Tahmin edilen nesne adları 6, (b) Tarifleme için kurulan cümleler 6

BÖLÜM 5

SONUÇ VE ÖNERİLER

Bu çalışmada bir ortamda bulunan nesneleri tanımlayan ve cümleler kurarak kullanıcıya tarifleyen bir sistem gerçeklenmiştir. Bu sistem, cep telefonu, tablet gibi çeşitli cihazların kameralarından alınan görüntülerin anlamlandırılarak, kullanıcıya aktarılmasını sağlamaktadır.

Günümüzde cep telefonu, tablet, güvenlik kameraları gibi birçok cihazdan anlık görüntü, video alınabilmektedir. Bu görüntü ve videolar bir çok bilgi içermekte fakat bu bilgilerin dijital ortamda işlenebilmesi için anlamlandırılıp, yazıya dökülme ihtiyacı bulunmaktadır. Görüntü üzerinde bulunan nesnelerin tanınıp, tariflenebilmesi için öncelikle görüntü üzerinde nesne bulunan bölgelerin tespit edilmesi gerekmektedir. Bu işlem için görüntü üzerinde grafik tabanlı bölütleme işlemi gerçekleştirilmektedir. Bu işlem sonrasında, gölge, gürültü gibi çeşitli sebeplerle farklı nesne gibi algılanan bölgelerin benzerliklerine göre birleştirilmesi amacıyla seçmeli arama algoritması uygulanarak nesnelerin bulundukları bölgeler tespit edilmektedir. Nesne görüntüleri derin konvolüsyonel sinir ağı tarafından analiz edilerek nesne tanıma işlemi gerçekleştirilmektedir. Elde edilen nesne konumları ve nesne isimleri kullanılarak, bu bilgiler kurulan cümleler vasıtasıyla kullanıcıya aktarılmaktadır.

Bu sistem, başarısının yüksek olması sebebiyle görme engelli kişilerin bulundukları ortam hakkında bilgi sahibi olmaları konusunda faydalı olacaktır. Küçük çocuklar çevrelerindeki nesnelerin isimlerini öğrenmeleri için kullanabilecektir. Günlük hayatta isimleri bilinmeyen çiçek, böcek gibi farklı nesnelerin isimlerinin öğrenilmesi amacı için de

kullanılabilir. Güvenlik kameralarından alınan görüntüler anlamlandırılarak çeşitli bilgi işleme işlemleri yapılabilir. Örneğin, güvenlik kameralarını sürekli izlemek yerine, bu kameralara belirli nesneler yakalandığında alarm veren sistemler geliştirilebilir. Cep telefonu, tablet gibi her türlü mobil cihazın kamerasından alınan görüntüler anlamlandırılarak, yaşlı kişilerin evde veya dışarıda yönünü bulmasına yardımcı olabilir. Televizyon gibi cihazlarda görüntülenen videolar üzerinde çalıştırılarak, görme engelli kişilerin bu videoları anlamasına yardımcı olabilir.

Sonraki çalışmalarda bu sisteme görüntü üzerinden derinlik bilgisi çıkarılarak ve ultrasonik sensör ile desteklenerek yakınlık bilgisi eklenecektir. Oluşturulan cümlelerin sesli bir yönlendirici ile kullanıcının kulağına iletilmesi sağlanarak, sistemin daha kullanışlı hale gelmesi sağlanacaktır.

KAYNAKLAR

[1] Felzenszwalb, P.F. ve Huttenlocher, D.P., (2004). “Efficient Graph-Based Image Segmentation”, International Journal of Computer Vision, 59(2).

[2] Girshick, R., (2014). “Rich feature hierarchies for accurate object detection and semantic segmentation”, CVPR, 24-27 Haziran 2014, Columbus.

[3] Girshick, R., (2015). “Fast R-CNN”, IEEE International Conference on Computer Vision (ICCV), 13-16 Aralık 2015, Santiago.

[4] Ren, S., He, K., Girshick, R. ve Sun, J., (2017). “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(6).

[5] Vinyals, O., Toshev, A., Bengio, S. ve Erhan, D., (2017). “Show and Tell: Lessons learned from the 2015 MSCOCO Image Captioning Challenge”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 39(4).

[6] Mullachery, V. ve Motwani, V., (2018). “Image Captioning”, arXiv preprint arXiv:1805.09137.

[7] Yang, Z., Zhang, Y., Rehman, S. ve Huang, Y., (2017). “Image Captioning with Object Detection and Localization”, The 9th International Conference on Image and Graphics, 13-15 Eylül 2017, Shangai.

[8] Hochreiter, S. ve Schmidhuber, J., (1997). “Long Short-Term Memory”, Neural Computation, 9(8).

[9] MTank, Multi Modal Methods, http://www.themtank.org, 1 Kasım 2018.

[10] Codeutsava, Deep Learning vs Machine Learning Comparison, https://codeutsava.in/blog/40, 1 Kasım 2018.

[11] Krizhevsky, A., Sutskever, I. ve Hinton, G.E., (2012). "ImageNet Classification with Deep Convolutional Neural Networks", NIPS 2012: Neural Information Processing Systems, 3-8 Aralık 2012, Lake Tahoe.

[12] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. ve Wojna, Z., (2016). "Rethinking the Inception Architecture for Computer Vision", Proceedings of IEEE

Conference on Computer Vision and Pattern Recognition, 26 Haziran-1 Temmuz 2016, Las Vegas.

[13] Nvidia Developer, Convolutional Neural Network (CNN), https://developer.nvidia.com/discover/convolutional-neural-network, 14 Ekim 2018.

[14] GitHub, Convolutional Neural Networks for Visual Recognition, http://cs231n.github.io/convolutional-networks/, 1 Kasım 2018.

[15] Mc.ai, Machine Learning : Broad Approaches, https://mc.ai/machine-learning-broad-approaches/, 24 Eylül 2018.

[16] MathWorks, Introduction to Deep Learning: What Are Convolutional Neural Networks, https://www.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html, 1 Kasım 2018.

[17] Towards Data Science, Applied Deep Learning - Part 4: Convolutional Neural Networks, https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2, 1 Kasım 2018.

[18] Glorot, X., Bordes, A. ve Bengio, Y., (2011). "Deep sparse rectifier neural networks", Proc. 14th International Conference on Artificial Intelligence and Statistics, (pp. 315–323).

[19] Wikipedia, Softmax Function, https://en.wikipedia.org/wiki/Softmax_function, 1 Kasım 2018.

[20] Towards Data Science, Activation Functions: Neural Networks, https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6, 1 Kasım 2018.

[21] Kruskal, J. B., (1956). "On the shortest spanning subtree of a graph and the traveling salesman problem", Proceedings of the American Mathematical Society, 7(1).

[22] Wikipedia, Gaussian Blur, https://en.0wikipedia.org/wiki/Gaussian_blur, 1 Kasım 2018.

[23] Uijlings, J.R., Sande, K.E., Gevers, T. ve Smeulders, A.W., (2013). "Selective search for object recognition", International Journal of Computer Vision (IJCV), 104(2).

[24] LeCun, Y., Bottou, L., Bengio, Y. ve Haffner, P., (1998). "Gradient-based learning applied to document recognition", Proceedings of the IEEE, 86(11).

[25] LeCun, Y. ve Cortes, C., (2010). MNIST handwritten digit database.

[26] He, K., Zhang, X., Ren, S. ve Sun, J., (2016). "Deep residual learning for image recognition", CVPR, 26 Haziran-1 Temmuz 2016, Las Vegas.

[27] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D. ve Rabinovich, A., (2015). "Going deeper with convolutions", CVPR, 8-10 Haziran 2015, Boston.

[28] Pratt, L. ve Jennings, B., (1996). “A survey of connectionist network reuse through transfer”, Connection Science, 8(2).

[29] Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A. ve Torralba, A., (2017). “Scene parsing through ade20k dataset”, IEEE Conference on Computer Vision and Pattern Recognition, 21-26 Temmuz 2017, Honolulu.

[30] TensorFlow, TensorFlow Image Recognition, https://www.tensorflow.org/tutorials/images/image_recognition, 15 Ekim 2018.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Anıl ÇETİNSOY

Doğum Tarihi ve Yeri : 23.06.1990 - Bakırköy

Yabancı Dili : İngilizce

E-posta : anilcetinsoy@hotmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Lisans	Bilgisayar Mühendisliği	Sakarya Üniversitesi	2012
Lise	Fen Bilimleri	Şehremini Anadolu Lisesi	2008

İŞ TECRÜBESİ

Yıl	Firma/Kurum	Görevi
2016	Finansbank - IBTech	Yazılım Mimarı
2012	Finansbank - IBTech	Yazılım Mühendisi
2011	Ford Otosan	Yazılım Mühendisi

YAYINLARI

Bildiri

1. Çetinsoy, A., (2018). “Derin Öğrenme ile Çevresel Bilgi Edinimi”, 26. IEEE Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 4 Nisan 2018, İzmir.