



T.C.
SELÇUK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



**BAL PETEĞİNDEKİ HÜCRELERİN TESPİT
EDİLMESİ İÇİN DERİN ÖĞRENME
YAKLAŞIMLARININ KULLANILMASI**

Maliha FARAHMAND

YÜKSEK LİSANS TEZİ

Bilgisayar Mühendisliği Anabilim Dalı

Nisan-2022
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Maliha FARAHMAND tarafından hazırlanan “Bal Peteğindeki Hücrelerin Tespit Edilmesi için Derin Öğrenme Yaklaşımlarının Kullanılması” adlı tez çalışması 14/04/2022 Tarihinde aşağıdaki jüri tarafından oy birliği / ~~oy çokluğu~~ ile Selçuk Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri	İmza
Başkan Dr.Öğr.Üyesi Ersin KAYA
Danışman Doç.Dr.Mehmet Akif ŞAHMAN
Üye Dr.Öğr.Üyesi Murat KÖKLÜ

Yukarıdaki sonucu onaylarım.

Prof. Dr. Sait GEZGİN
FBE Müdürü

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

İmza

Maliha FARAHMAND

Tarih: 14.04.2022

ÖZET

YÜKSEK LİSANS TEZİ

BAL PETEĞİNDEKİ HÜCRELERİN TESPİT EDİLMESİ İÇİN DERİN ÖĞRENME YAKLAŞIMLARININ KULLANILMASI

Maliha FARAHMAND

**Selçuk Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı**

Danışman: Doç. Dr. Mehmet Akif ŞAHMAN

2022, 62

Jüri

Doç. Dr. Mehmet Akif ŞAHMAN

Dr. Öğr. Üyesi Ersin KAYA

Dr. Öğr. Üyesi Murat KÖKLÜ

Bal peteğinin doğru yöntemler kullanılarak hasat edilmesi arıcılık faaliyetleri gerçekleştirenler ve ülkelerin ekonomisi için büyük önem arz etmektedir. Bal hasatında doğru yöntemlerin kullanılmaması hem balın üretim miktarını hem de kalitesini negatif etkileyecektir. Bilinçsiz arıcılık faaliyetleri doğal yaşamın devamı için bir role sahip arıların kendi nesillerini devam ettirmelerine ket vurmaktadır. Bu tez çalışmasında temel olarak arıcılıktan, arı kovanları ve peteklerinden, bir bal peteğindeki görüntülerin elde etme yaklaşımından ve bu görüntülerin etiketlenmesinden bahsedilmiştir. Elde edilen etiketlenmiş görüntülerin sınıflandırılması için kullanılabilir metotlar ve değerlendirme teknikleri incelenmiştir. Ayrıca bu tez çalışmasında, yedi sınıftan oluşan toplam 103.451 eğitim ve 25.863 test görüntüsü içeren bal peteği veri seti oluşturulmuştur. Bu çalışmada bal peteğinin sınıflandırması için derin öğrenme yaklaşımları sunulmuştur. Derin öğrenme yaklaşımlarının başarısı probleme göre değişebilmektedir. Bu çalışmada, önceden eğitilmiş CNN algoritmalarından, VGG16, VGG19, Inception-V3, AlexNet, self-structure CNN ve iki derin sinir ağı birleşimi olan ağ (ResNet50V2+Xception) kullanılarak sınıflandırma yapılmıştır. Sınıflandırma işleminin sonucunda en yüksek başarı oranı AlexNet algoritması ile %95 başarı oranı elde edilmiştir.

Anahtar kelimeler: Bal peteği görüntüsü, Derin öğrenme, Görüntü işleme, Sınıflandırma

ABSTRACT

MS THESIS

USING DEEP LEARNING APPROACHES TO DETECT CELLS IN HONEYCOMBS

Maliha FARAHMAND

**THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF
SELÇUK UNIVERSITY
THE DEGREE OF MASTER OF SCIENCE
IN COMPUTER ENGINEERING**

**Advisor: Assoc. Prof. Dr. Mehmet Akif ŞAHMAN
2022, 62**

Jury

**Assoc. Prof. Dr. Mehmet Akif ŞAHMAN
Assist. Prof. Dr. Ersin KAYA
Assist. Prof. Dr. Murat KÖKLÜ**

Harvesting the honeycomb using the right methods is of great importance for the beekeeping activities and for the country's economy. Failure to use the right methods in honey harvest will negatively affect both the production quantity and quality of honey. Unconscious beekeeping activities prevent bees, which have a great role in the continuation of natural life, from continuing their own generation. In this thesis, it is mainly mentioned about beekeeping, beehives and honeycombs, the approach of obtaining images in a honeycomb and labeling these images. The methods and evaluation metrics that can be used for the classification of the obtained labeled images are examined. In addition, in this thesis study, a honeycomb data set containing a total of 103,451 training and 25,863 test images, consisting of 7 classes, was created as an example. Different deep learning algorithms (VGG16, VGG19, Inception-V3, AlexNet, SSCNN and ResNet50V2+Xception) were used for classification of honeycomb images. The success of deep learning approaches can vary according to the problem. Therefore, these deep learning architectures were run on the same dataset and compared according to the experimentally obtained results. As a result of the classification process, the highest success rate was obtained from AlexNet deep learning algorithm with 95%.

Keywords: Honeycomb image, Deep learning, Image processing, Classification

ÖNSÖZ

Akademik hayatım ve tez çalışmam süresince engin bilgi, birikim ve tecrübeleri ile bana yol gösterici ve destek olan değerli hocam ve tez danışmanım Sayın Doç. Dr. Mehmet Akif ŞAHMAN'a içiten teşekkürlerimi sunuyorum ve değerli değer hocalarım sayın hocam Doç. Dr. Hasan Erdiñ KOÇER'e ve Arş. Gör. Kübra UYAR'a desteklerinden dolayı teşekkürlerimi sunuyorum. Tüm hayatım boyunca bana her zaman inanan, her zaman güvenen, destek olan, bu günlere gelmemi sağlayan, kıymetli annem Momina Farahmand'a ve kardeşlerime sonsuz teşekkür ederim. Bu tez çalışmasını her konuda destekleyen ve bugünlere gelmemde büyük emekleri olan rahmetli babam Awaz Farahmand'a ve rahmeti Abilerim Ramazan Farahmand'a ve Brat Frahmand'a armağan ediyorum.

Maliha FARAHMAND

Konya 2022

İÇİNDEKİLER

ÖZET	iii
ABSTRACT.....	iv
ÖNSÖZ	v
İÇİNDEKİLER	vi
KISALTMALAR.....	viii
1. GİRİŞ.....	1
2. LİTERATÜR ARAŞTIRMASI.....	3
3. MATERYAL.....	6
4. YÖNTEM	12
4.1. MAKİNE ÖĞRENMESİ.....	12
4.1.1. Yapay sinir ağları.....	16
4.1.2. Derin öğrenme.....	16
4.1.2.1. Evrimsel sinir ağları (CNN)	17
4.1.2.1.1. Evrimsel işlemi (Convolution operation).....	18
4.1.2.1.2. Alt örnekleme işlemi (Havuzlama)	18
4.1.2.1.3. Tam bağlı katman (Fully connected layers).....	18
4.1.2.1.4. Geri yayılım (Backpropagation)	19
4.1.2.1.5. Toplu normalleştirme (Batch normalization).....	19
4.1.2.1.6. Seyreltme katmanı (Dropout).....	19
4.1.2.1.7. Aktivasyon fonksiyonları	20
4.1.2.1.8. Toplu iş boyutu (Batch size)	23
4.1.2.1.9. Epok sayısı	23
4.1.2.1.10. Öğrenme hızı (learning rate)	24
4.1.2.1.11. Kayıp fonksiyonu (Loss Function).....	24
4.1.2.1.12. AlexNet	25
4.1.2.1.13. Vgg19.....	26
4.1.2.1.14. Vgg16.....	27
4.1.2.1.15. Inception.....	28
4.1.2.1.16. ResNet.....	28

4.1.2.1.17 Xception	29
4.1.2.2. Optimizasyon algoritmaları	29
4.1.2.2.1. Dereceli azalma (Gradient descent)	29
4.1.2.2.2. Momentum gradient descent	30
4.1.2.2.3. RMSprop	30
4.1.2.2.4. Adam	30
4.1.2.3. Düzenleştirme teknikleri (Regularization techniques)	31
4.1.2.3.1. Veri artırma (Data augmentation)	31
4.1.2.3.2. Erken durdurma (Early stopping)	32
5. BAŞARI ÖLÇÜM METRİKLERİ	33
5.1. GENEL DOĞRULUK	33
5.2. DUYARLILIK (RECALL)	33
5.3. KESİNLİK (PRECISION)	34
5.4. F1-SKOR	34
5.5. AUC-ROC EĞRİSİ	34
5.6. GEOMETRİK ORTALAMA	35
6. DENEYSEL ÇALIŞMALAR	37
7. SONUÇ VE ÖNERİLER	53
KAYNAKLAR	54

KISALTMALAR

CNN	: Evriřimli sinir ađı (Convolutional Neural Network)
SSCNN	: Öz-yapılı evriřimli sinir ađı (Self-Structured CNN)
R-CNN	: Bölgesel tabanlı evriřimli sinir ađı (Region Based Convolutional Neural Networks)
MR	: Manyetik rezonans görüntüleme (Magnetic Resonance Imaging)
ML	: Makine öğrenme (Machine Learning)
AI	: Yapay zekâ (Artificial Intelligence)
YSA	: Yapay Sinir Ağları
MLP	: Çok katmanlı algılayıcı (Multi-Layer Perceptron)
RELU	: Doğrultulmuş lineer birim (Rectified Linear Units)
OA	: Genel doğruluk (Overall Accuracy)
TP	: Gerçek pozitif (True Positive)
FN	: Yanlış negatif (False Negative)
TN	: Gerçek negatif (True Negative)
FP	: Yanlış pozitif (False Positive)
AUC	: Eğri altında kalan alan (Area under the curve)
BQCV	: Kara kraliçe hücre virüsü (Black Queen Cell Virus)
SBV	: Sacbrood virüsü (Sacbrood Virus)
CBPV	: Kronik arı felci virüsü (Chronic Bee Paralysis Virus)
GD	: Dereceli azaltma (Gradient descent)
SGD	: Olasılıksal dereceli azalma (Statistics gradient descent)
GPU	: Grafik İşlemci Birimi (graphics processing unit)
Sın	: Sınıf
Kes	: Kesinlilik
Duy	: Duyarlılık
Des	: Destek

1. GİRİŞ

Arıcılık, tarihsel olarak yerel yemek kültürünün yanı sıra tozlayıcıların üretimi ile ilgili bir faaliyet olarak çeşitli yerlerde uygulanmaktadır. Örneğin Japonya, Kore, Hindistan gibi ülkelerde bal, gıda kültürünün önemli bir unsurudur (Kohsaka ve ark, 2017). İnsan beslenmesinin tahmini %35'i doğrudan veya dolaylı olarak bal arısı tozlaşmasından yararlanıyor. Bu nedenle arıcılığın ve bal üretiminin korunması, yemek kültürünün sürdürülmesine büyük katkı sağlayabilir. Bal arıları modern tarımda hayati bir rol oynamaktadır (vanengelsdorp ve ark, 2013). Bal ve diğer arı ürünlerinin hastalıklara karşı dirençli bireylerin yetişmesinde önemli faydaları vardır. İçerdikleri mineral, vitamin ve enzimlerden dolayı antimikrobiyel, antiviral, antibakteriyel ve antiparaziter fonksiyonları bulunmaktadır. Bal arıları; bal, arı zehiri, arı sütü, , balmumu polen ve propolis gibi insan beslenmesi ve ayrıca insan sağlığı için önemli çıktılar sağlamaktadır (Anonim, 2020). Ayrıca iyi bir tozlayıcı olan bal arıları, ekosistemlerimizin hayatta kalması için temel bir sürece katkı sunmaktadır. Dünyadaki yabani çiçekli bitki türlerinin yaklaşık %90'ı, dünyadaki gıda mahsullerinin %75'inden fazlası ve küresel tarım arazisinin %35'i tamamen veya en azından kısmen hayvanların tozlaşmasıyla oluşmaktadır (Anonim, 2022). Tozlayıcılar yalnızca gıda güvenliğine doğrudan katkıda bulunmakla kalmaz, aynı zamanda biyolojik çeşitliliği korumanın anahtarıdır. Dolayısıyla arıcılık (beekeeping) hem stratejik hem de ekonomik öneme sahiptir. Arılar, kelebekler, yarasalar ve sinek kuşları gibi diğer tozlayıcılar, insan faaliyetlerinden dolayı farklı tehditlerle karşı karşıyadır. Arılar ve diğer tozlayıcılar, büyük ölçüde yoğun tarım uygulamaları, tek ürün yetiştirme, tarımsal kimyasalların aşırı kullanımı ve iklim değişikliğiyle ilişkili yüksek sıcaklıklar nedeniyle dünyanın birçok yerinde giderek azalmaktadır. Arıların küresel ölümlerinin dışında, arıların yanlış yaklaşımlarından, zamanında önlem almamalarından ve bilgi eksikliklerinden dolayı da arı kolonilerinde kayıplar yaşanmaktadır. Bu ölümlerin en büyük nedeni arı kolonilerinin zayıf düşürülmesidir. Arı kolonileri zayıf düşürüldüğünde arılar virüslerden (Sacbrood Virus (SBV), Black Queen Cell Virus (BQCV), Chronic Bee Paralysis Virus (CBPV) vb.) ve zararlı türlerden (Nosemosis, Varroa mites vb.) dolayı ölmekte, yavru gelişimleri doğal bir şekilde olmamakta ve yağmacılara (yaban arısı, eşek arısı vb.) karşı koyamamaktadırlar (Usta ve Yıldırım, 2020). Duruma ekonomik açıdan bakıldığında, insanoğlu dünya genelinde bir milyon ton bal üretimi mevcut olan 50 milyon bal arısı kolonisi ile gerçekleştirmektedir. Dünyadaki en büyük koloni varlığına sahip ülke Çin'dir. FAO 2010 rakamlarına göre

koloni başına ortalama bal üretimi 20kg'dır ve dünya genelinde uluslararası bal ticareti 350.000 ton seviyelerindedir. Türkiye, dünya bal arısı kolonilerinin %13'ü ile Çin'den sonra en büyük arı kolonisi varlığına sahiptir. Dünyadaki FAO 2010 rakamlarına göre koloni başına ortalama bal üretimi 20kg Çin'de 48kg iken Türkiye'de bu miktar 15kg'dır (Burcu, 2018). Türkiye uluslararası bal ticaretinden pay alamamakta, hatta 2008 yılında bal ithal etmek zorunda kalmıştır. Türkiye'nin çok sayıda koloniye sahip olmasına rağmen bal üretiminde dünya ortalamasının altında kalma sebepleri olarak bal hasatı sırasında gerçekleşen yavru kayıpları ve mekanizasyon eksikliği gösterilebilir. Bal süzme işleminde en önemli aşama petekte balın korunmasını sağlamak için arılar tarafından oluşturulan petek sırrının alınmasıdır. Mevcut kullanılan teknolojiler Petekte bulunan yavrular gözetilmeden tüm bal süzülerek yavru ölümlerine neden olmakta veya peteklerdeki yavruları korumak için baldan feragat etmektedir. Arıcılar ise burada bir ikileme düşmektedir ya yavrulardan ya da elde edilecek bal miktarından feragat etmesi gerekmektedir. Arıcılar, yavruları göz ardı ederek daha fazla verim elde etmekte, fakat arı kolonisinin zayıflamasına, böylece dış etkenlerden (hastalık veya yağmacılık) korunamayacak hale gelmesine neden olmaktadır. Arıcıların kısa süreli kar elde etme çabaları, uzun vadeli olarak kendilerine büyük oranda zarar vermektedir. Arıcılar, üzerinde yavru bulunan peteklerin hasatını yapmadığı zaman ise elde edilecek miktarın düşmesi ile karşı karşıya kalmaktadır. Yavrulu peteklerdeki baldan feragat edilmesi durumunda yaklaşık %10 bal miktarında kayıp olmaktadır. Arıcılıktaki bal hasatında bu kayıpların önüne geçmek için, bal peteğindeki hücrelerin doğru bir şekilde belirlenmesi gerekmektedir. Bu çalışmada bal peteğindeki hücrelerin doğru tespit edilmesi problemi bir sınıflandırma problemi olarak ele alınmıştır. Yapılan araştırmalar neticesinde bal peteklerindeki hücreler için yedi adet sınıf etiketi uygun görülmüştür. Kullanılan bal peteklerindeki görüntülerinde; içi boş olan hücre (0), içinde yavru olan kapalı hücre (1), içinde larva olan açık hücre (2), içinde bal olan kapalı hücre (3), içinde bir miktar bal olan açık hücre (4), içinde polen olan hücre (5) ve kullanmayan hücreler (6) olarak etiketlenmiştir. Bu etiketlenen veriler ile örnek bir çalışma veri seti oluşturulmuş ve derin öğrenme yaklaşımı ile tespit ve sınıflandırma işlemi gerçekleştirilmiştir.

2. LİTERATÜR ARAŞTIRMASI

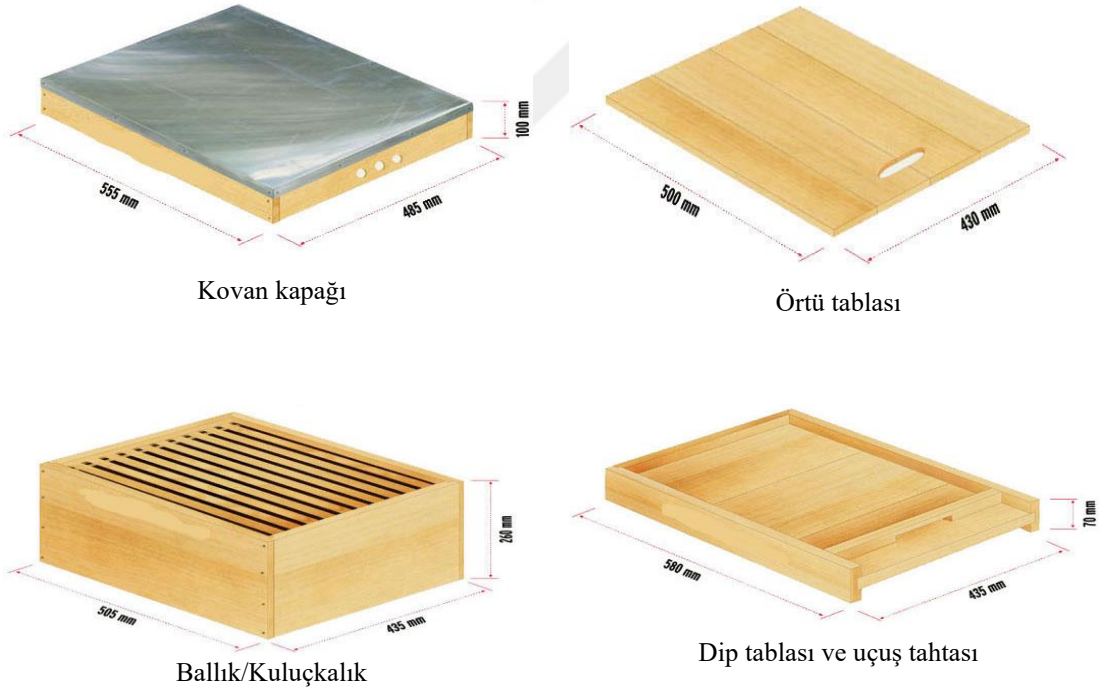
Yapılan araştırmalara göre bu konu ile ilgili görüntü işleme tekniklerinin ve derin öğrenme algoritmalarının kullanıldığı bir çalışma bulunamamıştır, fakat bu konuya benzer çalışmalar vardır. Tetila, yaptığı çalışmada soya fasulyesinin zararlı görüntülerinin sınıflandırılması için beş derin öğrenme mimarisinin değerlendirmesi yapılmıştır. Bu çalışmada; Inception-v3, Resnet-50, VGG-16, VGG-19 ve Xception mimarileri kullanarak en yüksek %93,82 doğruluk elde edilmiştir (Tetila ve ark, 2020). Barman, Utpal Choudhury yaptığı çalışmada narenciye yaprağının hastalıklarını sınıflandırmak için iki farklı Evrişimli Sinir Ağı (CNN) mimarisinin karşılaştırmasını gerçekleştirmiştir. Çalışmasında, bitkisel aşamada narenciye yaprak hastalıklarını saptamak ve sınıflandırmak için MobileNet ve Self-Structured (SSCNN) sınıflandırıcılar gibi iki tür CNN mimarisini kullanmıştır. Akıllı telefon görüntüsüne dayalı bir narenciye hastalığı veri seti hazırlanmıştır. Her iki model de aynı veri seti eğitim ve test olarak kullanılmıştır. MobileNet CNN'nin en iyi eğitim doğruluğu, 10. Epoch 'da %98 başarı ve SSCNN'nin en iyi eğitim doğruluğu, 12. Epoch 'da %98 başarı elde edilmiştir. Yapılan deneylerde; SSCNN'nin akıllı telefon görüntüsüne dayalı narenciye yaprağı hastalığının sınıflandırılmasında daha verimli olduğunu göstermektedir. Ek olarak, SSCNN algoritması MobileNet'e kıyasla daha az hesaplama süresi ve narenciye hastalığının tespiti için uygun maliyetli bir yöntem olarak kabul edilebilir (Barman ve ark, 2020). Harmandeep Singh Gill yaptığı çalışmada, meyve görüntülerini sınıflandırılmasında bütünleşmiş bir sınıflandırma geliştirmek için derin öğrenme tabanlı uygulamaları birleştirmiştir. Bu bütünleşmiş yaklaşımında CNN, RNN ve LSTM iş birliği yapmaktadır. Önerilen yaklaşımın temel amacı, meyve görüntülerinin hiyerarşik etiketlerini geliştirmektir. CNN, kaba ve ince etiketleri başlatmak için dahil olur. RNN, dinamik davranış geliştirme için kullanılır ve son olarak LSTM, öğrenme modelini eğitmek için bellek hücresi tarafından RNN ile iş birliği yapar. Önerilen yaklaşım; CNN, RNN, ANFIS ve RNN-CNN gibi diğer mevcut sınıflandırma teknikleriyle karşılaştırılmıştır. Karşılaştırma için doğruluk, F-ölçümü, duyarlılık ve özgüllük gibi çeşitli ölçüler kullanmıştır. Deneysel sonuçlardan, önerilen yaklaşımın performansının CNN, RNN, ANFIS ve RNN-CNN gibi mevcut diğer sınıflandırma tekniklerinden daha iyi olduğu gözlemlenmiştir (Gill ve Khehra, 2021). Ekrem Saralioğlu çok bantlı görüntülerin sınıflandırılması için 3d-2d CNN modeli kullanmıştır. Bu çalışmada kitle kaynak tabanlı

veri üretimini ele almıştır ve uygulama ile başarı düzeyini göstermiştir. Kullanıcılardan denetimli olarak veri sınıflandırmasının yapılması istenmiştir ve verilen puanlara göre veri etiketlemesi gerçekleştirilmiştir. Böylece, 260262 piksellik bir veri seti oluşturulmuştur. Elde edilen veri seti ile orijinal görüntüler kıyaslanmıştır ve %95'in üzerinde bir başarı elde edilmiştir (Saralioğlu, 2020). Mohammad Rahimzadeh yaptığı çalışmada, X-ışını görüntülerini üç sınıfa ayırmak için eğitim tekniklerini tanıtan birkaç derin evrişimli ağ eğitmiştir: iki açık kaynaklı veri kümesine dayalı normal zatürre ve COVID-19. Veriler, COVID-19 ile enfekte kişilere ait 180 adet X-ışını görüntüsü içermektedir ve çalışmada, mümkün olan en iyi sonuçları elde etmek için uygun yöntemler kullanılmıştır. Bu araştırmada Xception ve ResNet50V2 ağlarının bir birleşimi olan bir sinir ağı kullanılmıştır. Bu ağ, iki farklı ağ tarafından çıkarılan birden çok özelliği kullanarak en iyi doğruluğu elde etmiştir. Bu modeli, 11302 adet görüntü üzerinde test edilmiştir. COVID-19 vakalarını tespit etmek için önerilen ağın ortalama doğruluğu % 99.50 ve tüm sınıflar için genel ortalama doğruluk %91.4'tür (Rahimzadeh ve Attar, 2020). Pezhman Pasyar yaptığı çalışmada karaciğer biyopsisi yaygın karaciğer hastalıklarının saptanmasında temel standart olarak kabul edilmesine rağmen, çok sayıda yan etkisi olan bir yöntemdir. Ultrason görüntülemenin kullanıldığı yaygın karaciğer teşhisi, hekimin özneliği tarafından etkilenebilir. Bu nedenle, karaciğer hastalıklarının doğru bir şekilde sınıflandırılması kayda değer bir talep olmaya devam etmektedir. Bu çalışmada, önceden eğitilmiş derin evrişimli sinir sisteminden oluşan yeni bir derin sınıflandırıcı olan ağlar (CNN'ler) karaciğer durumunu kategorize etmek için önerilmektedir. ResNeXt, ResNet18, ResNet34, ResNet50 ve AlexNet gibi çeşitli ağlar kullanılmıştır. Transfer öğrenmeyi kullanarak çıkarılan derin özellikler, yeterli sınıflandırma bilgisi sağlayabilmektedir. Hastalıklar; normal karaciğer, karaciğer hepatiti ve sirozdur. Bu karaciğer görüntülerini ayırt etmek için iki sınıf (normal / siroz, normal / hepatit ve siroz / hepatit) ve üç sınıf (normal / siroz / hepatit) sınıflandırıcılar ile eğitilmiştir. İki sınıflı sınıflandırıcılar, üç sınıflı sınıflandırıcılar ile kıyaslanmıştır. Üç sınıfa ayrılan karaciğer görüntüleri için bir melez sınıflandırıcı ile ResNet50 kullanarak %86,4'lük bir doğruluk göstermiştir. Normal ve siroz karaciğer ile normal ve melez karaciğer arasındaki ayırmada, sonuçlar ilk grubun duyarlılık ve özgüllüğünün %90,9 ve %86,4 olduğunu ve ikinci grubun %90,9 duyarlılık ve %81,8 özgüllük olduğunu göstermiştir (Pasyar ve ark, 2021). Michal Byra yapmış olduğu çalışmada, ABD'de meme

kitle sınıflandırmasında derin öğrenmeye dayalı bir yaklaşım kullanmıştır. Evrişimli sinir ağları (CNN'ler) ile transfer öğrenimi, tıbbi görüntü analizinde nesne tanıma modelleri geliştirmek için yaygın olarak kullanılır. En yaygın olarak kullanılan ince ayar teknikleri, hedeflenen tıbbi sorunları ele almak için önceden eğitilmiş ağların ağırlıklarını değiştirmiştir. Bununla birlikte, önceden eğitilmiş ağın eğitilebilir parametrelerinin sayısı büyük olduğunda ve mevcut tıbbi veriler az olduğunda ince ayar yapmanın zor olabileceği gösterilmiştir. Bu sorunu çözmek ve ağda daha iyi bilgi akışını sağlamak için önceden eğitilmiş bir CNN'nin blokları arasına yerleştirilen derin temsil ölçekleme (DRS) katmanlarına dayanan yeni bir transfer öğrenme tekniğini önerilmiştir. Ağ eğitimi sırasında, meme kitle ABD görüntülerini işlemek ve önceden eğitilmiş CNN'yi ayarlamak için yalnızca DRS katmanlarının parametreleri güncellenmiştir. Bu durumda, DRS tabanlı yaklaşımın eğitilebilir parametre sayısını büyük ölçüde azalttığını ve standart transfer öğrenme tekniklerine göre daha iyi veya karşılaştırılabilir performans sağladığını göstermiştir (Michal, 2021).

3. MATERYAL

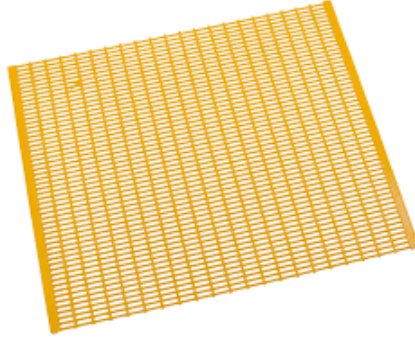
Doğal şartlarda arılar ağaç ve taş kovuklarını kendilerine yuva olarak seçmektedir. Fakat, arıcılığın tarımsal bir etkinlik olarak gelişmesiyle birlikte, arılar için farklı türde kovanlar önerilmiştir. Kovanların zaman içinde gelişimi hala devam etmektedir. Bugün dünyanın birçok yerinde hareketli çerçevelerin bulunduğu ve standart çerçeveli modern kovanlar kullanılmaktadır. Langstroth ve Dadant tipi kovanlar modern kovanlar olarak kabul görmektedir. Langstroth tipi kovan Türkiye’de en çok tercih edilen kovan tipidir. Bu nedenle, gerçekleştirilen çalışmada, kovanların Langstroth standardında olduğu kabul edilmiştir (Langstroth, 1878). Langstroth tipi kovanlar beş farklı bölümden oluşmaktadır. Bu bölümler; kovan kapağı, örtü tahtası, ballık/kuluçkalık, dip tablası ve uçuş tahtasıdır. Türkiye’de genellikle arıcılık ile uğraşanlar, mevsimlere bağlı olarak kovanlarını hava şartlarına uygun olan bölgelere taşımaktadır. Bu nedenden dolayı çıkarılabilir olması gereken dip tablası sabit olarak kullanılmaktadır. Langstroth tipi kovanların temel bölümleri Şekil 3.1’de standart ölçüleri ile verilmiştir.



Şekil 3.1 Langstroth tipi kovanların temel bölümleri

Langstroth tipi kovanlarda gövde kısmı kuluçkalık veya ballık olarak farklı iki amaç için kullanılmaktadır. Kovanlarda, kraliçe tarafından sürekli yavru üretimi yapılmaktadır ve

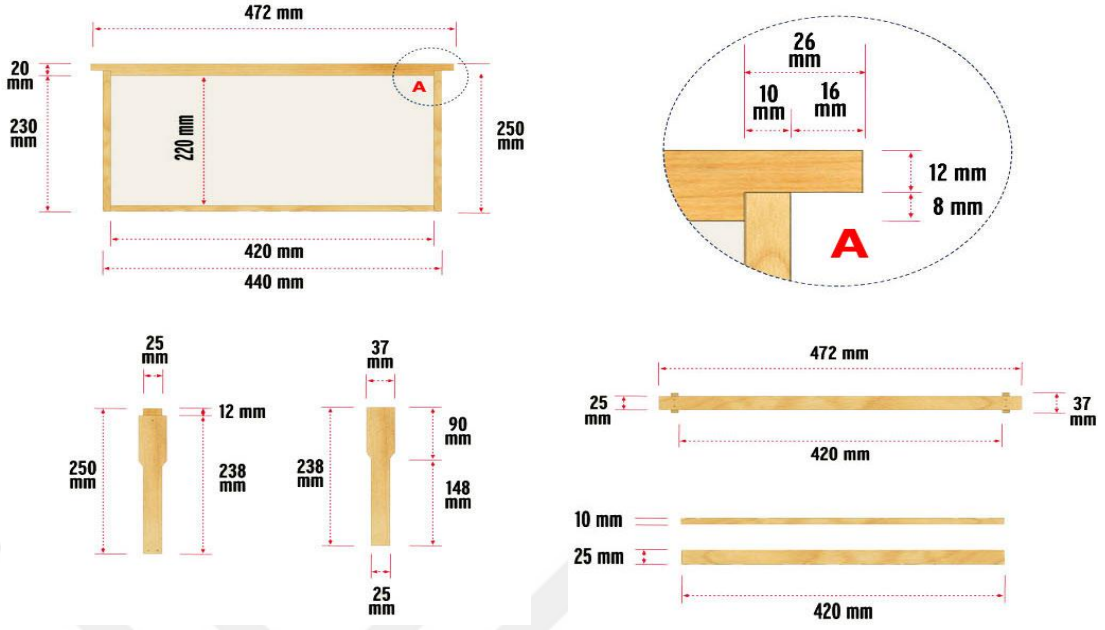
böylece kovanın popülasyonu korunmaya çalışılır. Arıcılar, yavru olan peteklerden bal alamadıkları için kovayı kuluçkalık ve ballık olarak iki kısma ayırırlar. Kovanların altında kalan kısmı kuluçkalık, üstüne koyulacak kısım ise ballık olarak belirlenir. Bu iki kısmı ayıracak bir bileşen kullanılır ve bu bölüm kraliçe dışlayıcı (queen excluder) olarak adlandırılır ve Şekil 3.2 de gösterilmiştir.



Şekil 3.2. Kraliçe dışlayıcı

Kraliçe dışlayıcının delikleri, kraliçe arının geçmesini engelleyecek kadar küçük fakat işçi arıların geçmesine imkân verecek kadar büyüktür. Böylece kraliçe arı yavrularını kuluçka bölümünde yayarlar. Böylece ballık kısmında ise sadece bal olan petekler bulunur. Kuluçka bölümü bir tane olmasına rağmen ballık bölümü ihtiyaca göre birden fazla olabilir. Kuluçka ve ballık bölümünde 10 çerçeve bulunmaktadır. Kuluçka ve ballık bölümleri aynı büyüklük ve standarttır. Genel olarak Langstroth tipi kovanın ve bu kovandaki çerçevelerin standartları Şekil 3.3'te verilmiştir.

Bu çalışmada kullanılmak üzere toplamda 19 tane Langstroth standardında farklı özellikli bal peteğinden faydalanılmıştır. Her bir peteğin iki yüzü olduğu için toplamda 38 adet bal peteği görüntüsü üzerinde çalışmalar yapılmıştır. Farklı özellikli peteklerle çalışmanın nedeni ise peteklerin farklı karakteristik özelliklere sahip olması ve çalışmanın kapsayıcılığının artırılmasıdır. Bu karakteristik özellikleri, arının cinsi, arının bulunduğu bölge, bal için topladığı nektar (çam, fi, ay çiçeği, mısır, çiçek vb.), peteğe daha önce larva konulması gibi etkenler etki etmektedir. Bu etmenler peteklerin renk dokusunu değiştirmektedir.



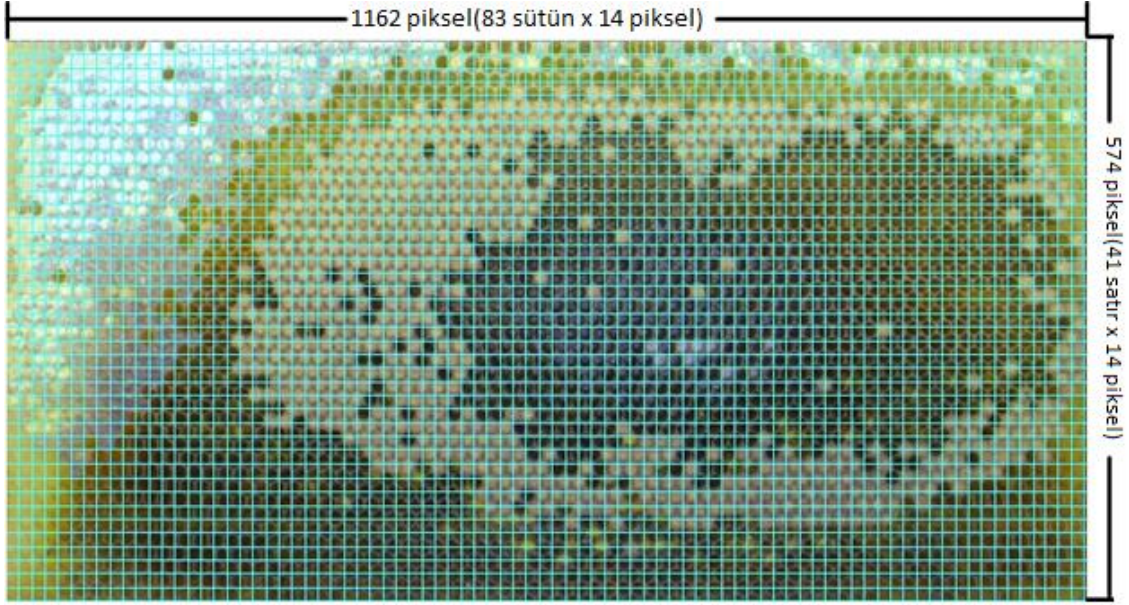
Şekil 3.3 Kovandaki çerçevelerin standartları

Bu çalışmada, görüntüler elde edilirken BASLER acA2500-14uc alan tarama kamerası kullanılmıştır. Projede kullandığımız BASLER acA2500-14uc alan tarama kamerası, maksimum 2590x1942, minimum 64x64 piksel ekran çözünürlüğünde görüntü sağlamaktadır. Ayrıca kazanım değeri 0-23,7 arasında değişebilmektedir. Görüntülerin alınırken dış ortamdaki gürültülerin engellenebileceği bir kabin hazırlanmıştır ve ışık kaynağı kullanılmıştır. Çerçeve görüntüleri 2590x1940 piksel boyutlarında çekilmiştir, fakat çalışma alanı sadece peteğin bulunduğu kısım ile sınırlandırılmıştır. Petek bölgesinin boyutları 1162x574 piksel olarak belirlenmiştir ve Şekil 3.4’ te gösterilmiştir.



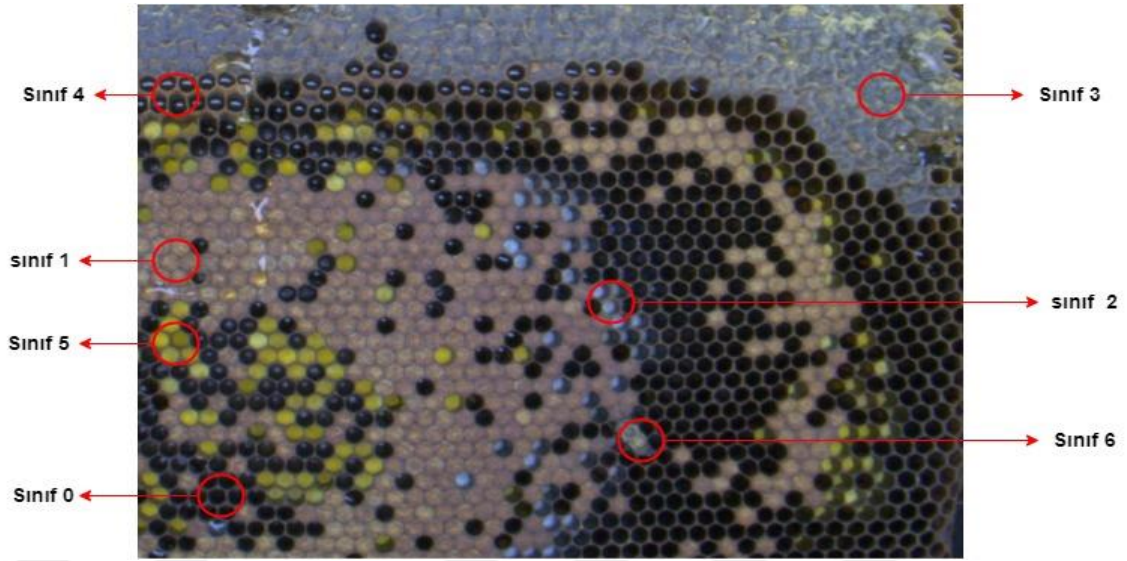
Şekil 3.4 Bal peteği görüntüleme süreci

Yapılan incelemelerde, petekteki her bir hücrenin genişliğinin 14 piksel olduğu görülmüştür. Dolayısıyla 1162x574 piksel boyutundaki petek 14x14 piksellik küçük parçalara ayrılmıştır. Böylece bir petek resminden 83 sütun ve 41 satır toplamda 3403 adet resim parçası elde edilmiştir. Görüntüler TÜBİTAK 1512 Teknogirişim Sermayesi Desteği Programı 2170060 numaralı projeden alınmıştır ve Şekil 3.5'te gösterilmektedir.



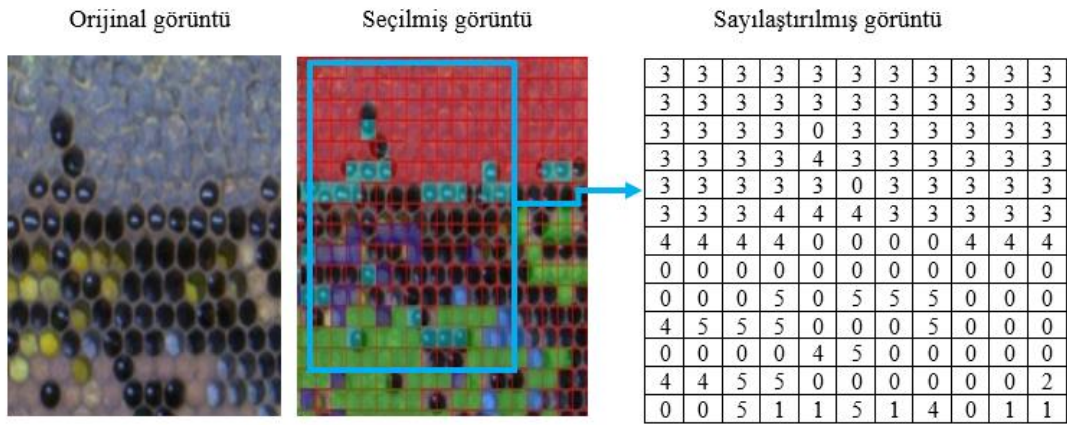
Şekil 3.5 Bal peteğinin parçalanmış hali

Bir petekteki hücreler farklı durumlara sahip olabilmektedir. Bu durumlar, daha önce belirtildiği gibi içi boş olan hücre (0), içinde yavru olan kapalı hücre (1), içinde larva olan açık hücre (2), içinde bal olan kapalı hücre (3), içinde bir miktar bal olan açık hücre (4), içinde polen olan hücre (5) ve kullanmayan hücreler (6) şeklinde belirlenmiştir ve Şekil 3.6' de göstermektedir. Bu çalışmada, mevcut durumlar yedi ana sınıfta toplanmıştır. Bunun nedenlerden birincisi arıcıların içinde çok fazla larva olan açık hücreye sahip petekleri kesinlikle bal hasatında kullanmamalarıdır. İçinde larva olan açık hücredeki yavrulu peteklerin hasat edilmesi durumunda hem gelecek nesli oluşturacak yavrular yitirilecektir hem de süzülen bala bu yavrular karışacak ve balın homojenliği bozulacaktır. Diğer neden ise yedi sınıfı tespit ederek işlenmesi gereken hücreleri işleyip, diğer hücreler için zamanı boşa harcamamış oluruz. Bu şekilde yavrular zarar görmez ve balın homojenliği bozulmaz.



Şekil 3.6 Bal peteğinin 7 farklı etiketi

Peteklerin standart bir düzeni olmadığından dolayı, resim üzerinde seçim yapmak için oluşturulan ızgaralar tam olarak hücrenin üzerine gelmemektedir. Her bir hücre içinde herhangi bir özelliğe %50'den fazla özellik varsa o hücre o sınıfın özelliğini taşımaktadır ve Şekil 3.7'de görüntünün durumları gösterilmiştir.



Şekil 3.7 Görüntünün 3 farklı durumu.

Seçilen her bir petekten 3403 işaretlenmiş 14x14 piksel büyüklüğünde resim parçası elde edilmiştir. 38 adet petek resminden toplamda $38 \times 3403 = 129314$ adet işaretlenmiş resim parçası elde edilmiştir. Bu resimlerin 68476 adeti içi boş olan hücreler (0), 26554 adeti içinde larva olan kapalı hücreler (1), 1832 adeti içinde larva olan açık hücreler (2), 22042 adeti içinde bal olan kapalı hücreler (3), 7338 adeti içinde bal olan açık hücreler (4), 2144 adeti içinde polen olan hücreler (5) ve 928 adeti kullanılmayan hücreler (6)'den meydana

gelmektedir. Ayrıca, her bir hücreye ait farklı renk uzaylarına (RGB, HSV, CIE Lab) ait bilgiler farklı görüntü işleme metotlarının farklı uygulamalarında kullanılması için kaydedilmiştir. Her sınıfın örnek sayısı ve oranları aşağıdaki çizelgede gösterilmiştir.

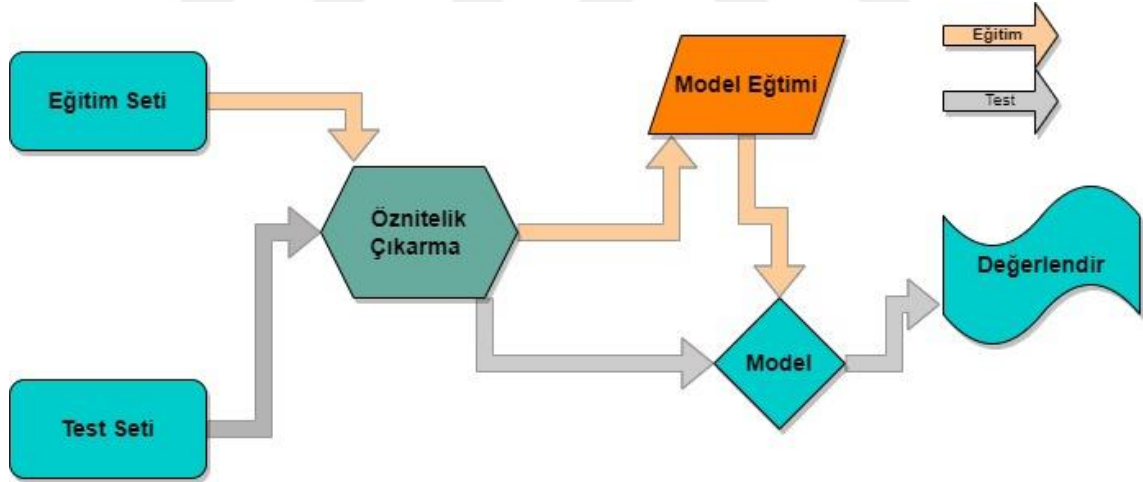
Çizelge 3.1 Sınıfların örnek sayısı ve oranları

Sınıf	Tanımı	Örnek Sayısı	Oranı
0	Boş olan hücreler	68476	52.95
1	Larva olan kapalı hücreler	26554	20.53
2	Larva olan açık	1832	1.42
3	Bal olan kapalı hücreler	22042	17.05
4	Bal olan açık hücreler	7338	5.67
5	Polen olan hücreler	2144	1.66
6	Kullanılmayan hücreler	928	0.72
	Toplam	129314	100.00

4. YÖNTEM

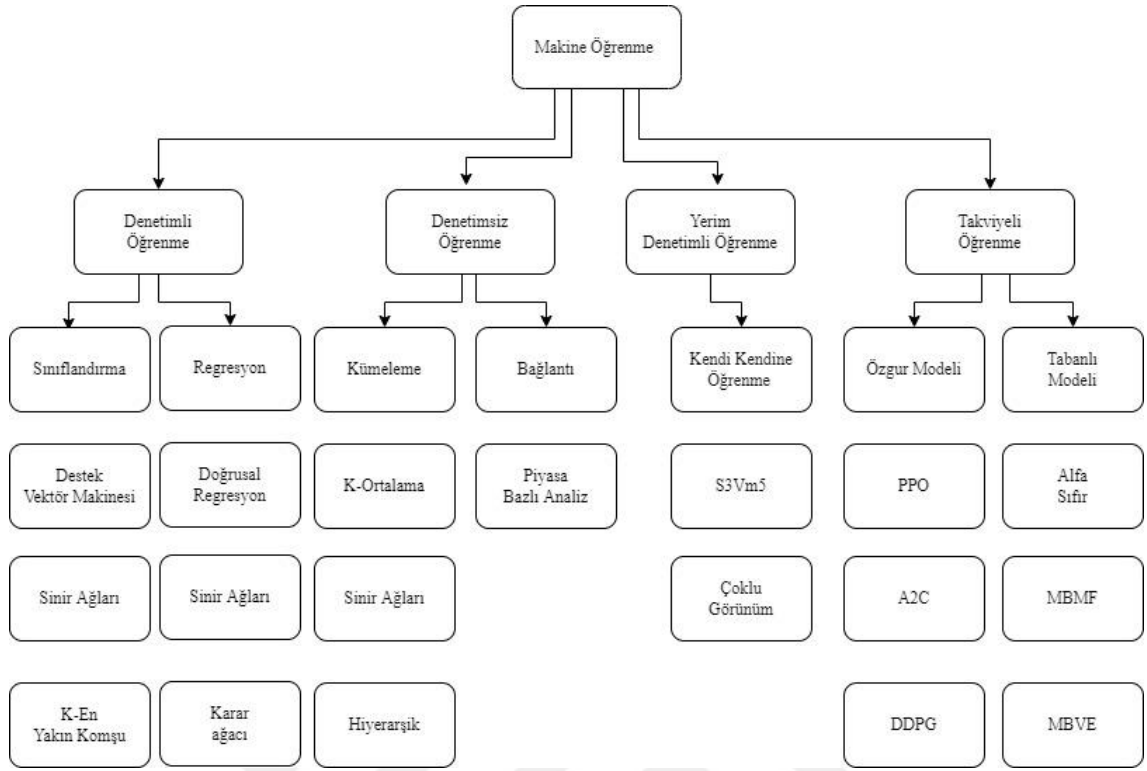
4.1. Makine Öğrenmesi

Deneyimlerden öğrenmek, insanlar veya hayvanlar için doğaldır. İnsanlar olaylardan veya gerçeklerin gözlemlerinden bilgi elde eder. Bu bilgilere ihtiyaç duyulduğunda, örneğin yeni bir olay meydana geldiğinde, toplanan bilgileri bu yeni olay için kullanılır. Makine öğrenimi (ML), bilgisayarın deneyimlerinden öğrenmesini sağlayan bir tekniktir. Makine öğrenimi, mevcut verilere dayalı tahminler yapmaya odaklanır. ML, kural tabanlı bir teknik gibi değildir. ML yöntemleri, bilgiyi doğrudan ayrıştırarak verilerden öğrenir. Bu yöntemlerin önceden belirlenmiş denklemleri veya çözümü bulan açıkça programlanmış kararları yoktur. Bu yöntemler temel olarak sadece verileri kullanarak çözümün yolunu oluşturur. ML algoritmaları, verilerdeki kalıpları veya örüntüleri bulur. Bu kalıpları kullanarak kararlar, tahminler veya tespitler yapar. Veri sayısı arttıkça performansları artar ve daha iyi sonuçlar verirler. Veri boyutu makine öğrenimi için çok önemlidir (Top, 2018). Şekil 4.1 de makine öğreniminin nasıl çalıştığını göstermektedir.



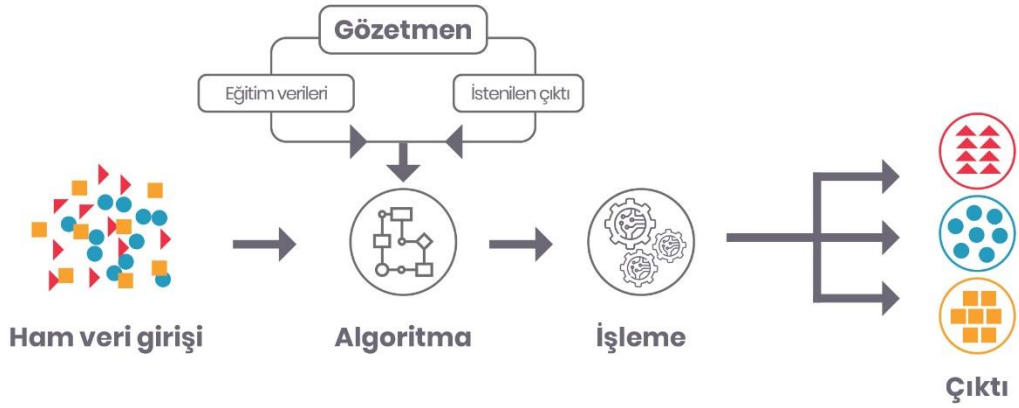
Şekil 4.1 Makine öğreniminin nasıl çalıştığını dair bir örnek

Makine öğrenmesi Yapay Zekâ (AI) 'nın bir alt dalı olarak bilinir. Dört ana alt bölüme ayrılmıştır ve bu alt bölümlerin altında, makine öğrenimi hedeflerine ulaşmak için kullanılacak çeşitli algoritmalar vardır. Aşağıdaki Şekil 4.2 makine öğrenimi sınıflandırması ve alt bölümleri göstermektedir.



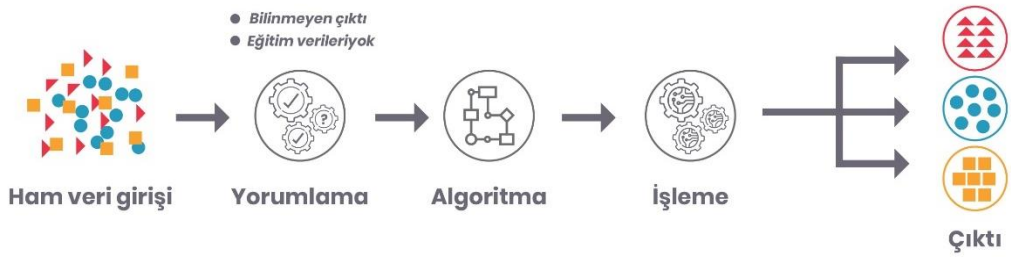
Şekil 4.2 Makine öğrenme alt bölümleri

- **Denetimli makine öğrenimi**, etiketli verilerden yararlanan bir makine öğrenimi türüdür. Etiketlenmiş bir veri kümesi üzerinde eğittikten sonra etiketlenmemiş verilerde sınıflandırma yapar. Örnek girdileri eşlemenin genelleştirilmiş bir kuralını öğrenir. Örneklerle eğitimden sonra, yeni girdileri bilinmeyen çıktıklarına eşleyebilen bir fonksiyon üretir. İstenen sonuç, yeni girdileri doğru bir şekilde eşlemesi ve görünmeyen etiketlerini bulmasıdır. Bu, bir öğrenme algoritmasının genelleme yeteneğidir. Denetimli öğrenme iki teknik kullanır: Birincisi regresyon; veriler ve etiketleri arasındaki doğrusal bir ilişki içinde sürekli değişkenlerin değerlerini tahmin etme sürecidir. İkincisi ise sınıflandırma; verileri tahmin etme veya iki veya daha fazla özel etiket halinde kategorilere ayırma sürecidir. Sınıflandırma, en çok kullanılan makine öğrenimi tekniği olarak kabul edilir. En yaygın kullanılan denetimli makine öğrenme (ML) algoritmaları destek vektör makinesi (SVM), lojistik regresyon, k-En Yakın Komşu (kNN), Naive Bayes, Yapay Sınırlı Ağları, Karar Ağaçları, Rastgele Orman'dır (Asadullayeva, 2020) ve Şekil 4.3'te denetimli makine öğrenmesini gösterilmektedir.



Şekil 4.3 Denetimli makine öğrenmesi (Anonim, 2021)

- **Denetimsiz makine öğrenimi**, etiketlenmemiş verilerdeki örüntüleri ve ilişkileri bulabilen bir öğrenme türüdür. Verilerdeki benzerlikler, farklılıklar ve temel yapı keşfedildiğinde, bunları gruplandırılabilir. Veri kümesini benzerliklere göre gruplara ayırır. Yaygın denetimsiz öğrenme algoritmaları, k-ortalama kümeleme, temel bileşen analizi (PCA), gizli markov modeli ve apriori algoritmasıdır. Tanımlayıcı öğrenmenin hedefi olarak bilinen denetimsiz öğrenme, yeni modeller veya ilişkiler bulmak veya tasarlamaktır. Önceden herhangi bir eğitim olmadan veriler üzerinde işlem yapar (Top, 2018) ve Şekil 4.4'te denetimsiz makine öğrenimi gösterilmektedir.



Şekil 4.4 Denetimsiz makine öğrenmesi (Anonim, 2021)

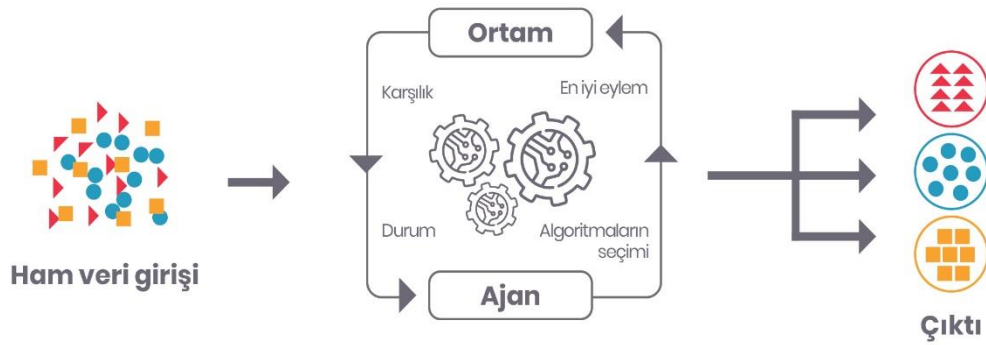
- **Yarı denetimli öğrenme**, verilerin bir kısmının kategorize edildiği ve etiketlendiği ve geri kalanının etiketlenmediği bir öğrenme türü olarak tanımlanır, denetimli öğrenme ile denetimsiz öğrenme arasında bir bağlantıdır. Burada

algoritmalar etiketli veriler ve bilgisini etiketlenmemiş veriler üzerinde işlemler yapar. Şekil 4.5'te yarı denetimli makine öğrenimi gösterilmektedir.



Şekil 4.5 Yarı denetimli makine öğrenmesi (Anonim, 2021)

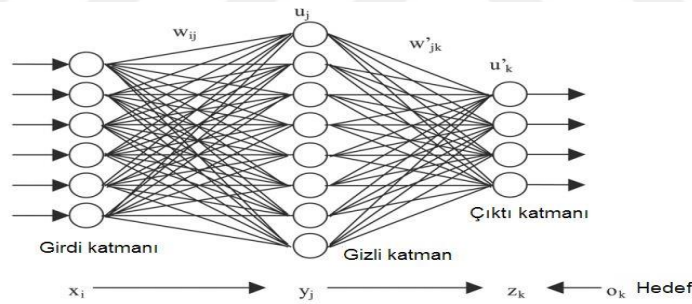
- **Takviyeli öğrenme algoritmaları**, ödül sistemiyle öğrenmenin gerçekleştiği bir türdür. Geri bildirim olarak aldığı tepkileri göre öğrenmeye gerçekleştirir. Başlangıç ve bitiş noktaları olmasına rağmen, temsilcinin amacı, hedefe ulaşmak için en kısa ve doğru yolları kullanmaktır. Temsilci doğru yollardan geçtiğinde olumlu ödüller alır. Ancak yanlış yollardan geçtiğinde olumsuz ödüller aldığı anlamına gelir. Öğrenme, hedefe giden yolda gerçekleşir (Çelik, 2018) ve şekil 4.6'de takviyeli makine öğrenimi gösterilmektedir.



Şekil 4.6 Takviyeli makine öğrenmesi (Anonim, 2021)

4.1.1. Yapay sinir ağıları

YSA, biyolojik ağlardan oluşan ve sinir sisteminin çalışma şeklini taklit eden en modern matematiksel modeldir. Bu hücrelerin birbirleri ile kurduğu sinaptik bağın dijital olarak modellenmesidir. Girdiler ve çıktı sistemleri arasındaki örüntü tanımlamasını kullanarak karmaşık ilişkileri çözmek için kullanılan bir hesaplama şemasıdır. 20. yüzyılın ortalarında ortaya çıkmıştır. Hem ileri hem de geri yayılımı hesaplamak için önemli miktarda nöron ve bu nöronlar arasındaki bağlantılardan oluşur. Havacılık, otomotiv, elektronik, imalat vb. dahil olmak üzere birçok bilim ve mühendislik alanında çeşitli uygulamaları mevcuttur (Adegbite ve ark, 2021). YSA, verileri tahmin etme, sınıflandırma ve eşleştirme konusunda olağanüstü bir yeteneğe sahiptir. Bu ağlar farklı şekillerde bölümlere ayrılmıştır, en basit ve en yaygın kullanılanı çok katmanlı algılayıcı sinir ağıdır. Bu ağlar, insan beyni ve matematiksel ilişki kalıplarını takip ederek çıktıyı küçük hatalarla doğru tahmin etmek için girdi verileri arasındaki karmaşık ilişkiyi anlar (Chen ve ark, 2021). Şekil 4.7’de çok katmanlı bir yapay sinir ağıları gösterilmektedir.



Şekil 4.7 çok katmanlı Yapay Sinir Ağı yapısı (Zaloğlu, 2015)

4.1.2. Derin öğrenme

Derin öğrenme (DL), makine öğrenimi alanında bir araştırma yönüdür. "Derin Öğrenme" ilk olarak Hinton ve arkadaşları tarafından 2006 yılında sunulmuştur. Veriler üzerinde otomatik olarak üst düzey soyutlama gerçekleştirebilen çoklu ara işlem katmanlarını içeren çoklu karmaşık doğrusal olmayan dönüşüm yapılarından oluşan bir temsili öğrenme algoritmasıdır (Mohammed, 2021). Denetimsiz kategori altında, büyük veri olarak bilinen, çok büyük miktarda yapılandırılmamış ve etiketlenmemiş veriden öğrenebildiği için, derin sinir ağı veya derin sinir öğrenmesi olarak bilinir. Gönümüzde popülerleşen "büyük veri" kavramı bazı gereksinimleri de beraberinde getirmiştir. Büyük

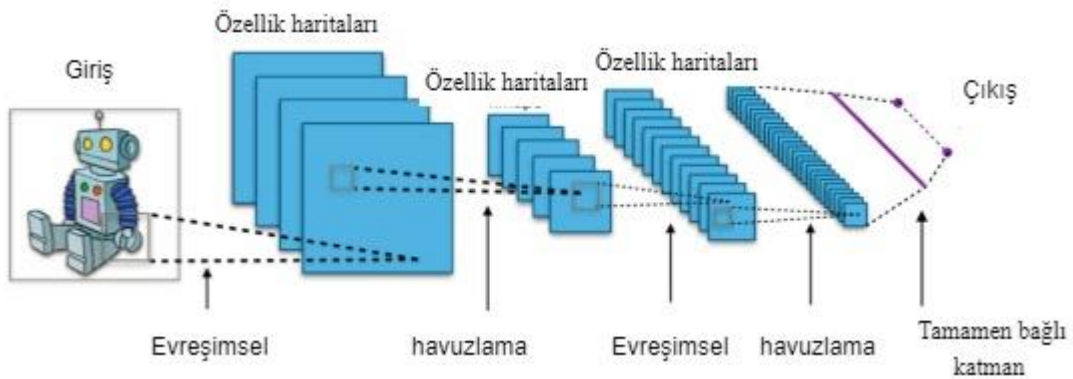
verinin gereksinimlerini karşılamak ve sorunlarını çözmek adına derin öğrenme yöntemleri zamanla eksikleri tamamlanarak geliştirilmiştir. Derin öğrenme, birden fazla basit teknolojidendir oluşur. Bu teknolojilerin iyileştirilmesi, sonunda derin öğrenmeyi geliştirmiştir. Dünyadaki alternatif kaynaklarla ilgili büyük verilerin toplanması ve işlenmesi belki de yıllarımızı alacaktır. Bu büyük miktarda yapılandırılmamış ve etiketlenmemiş veri, derin öğrenme ile öğrenilmektedir(Alafi, 2019).

4.1.2.1. Evrişimsel sinir ağları (CNN)

Birbirleriyle iletişim kuran nöronlardan oluşan bir sinir ağıdır. Her nöronun ağırlık değerleri vardır ve model bu ağırlıkları güncelleyerek, ağı eğiterek doğru sonuçlar üretmektedir(Çevik ve Zengin, 2019). Görüntüleri sınıflandırmak ve resimlerdeki nesnelere algılamak için büyük bir potansiyele sahip olduğu kanıtlanmıştır (Ba ve ark, 2021). Ana amacı ağı eğitmek ve filtre ağırlıklarını belirlemektir. CNN ileri beslemeli bir yapıdır. CNN'in öğrenmesini belirleyen kritik olabilecek bazı matematiksel kavramları vardır. Bu kavramlar aşağıda maddeler halinde verilmiştir ve alt başlıklarda detaylı olarak anlatılmıştır.

- Evrişimsel işlemi (Convolution operation)
- Havuzlama (pooling)
- Tamamen Bağlı Katman (fully connected layers)
- Geri yayılım (Backpropagation)
- Batch Normalization
- Seyreltme Katmanı (Dropout)

Genel olarak CNN mimarisi şekil 4.8'de gösterilmektedir.



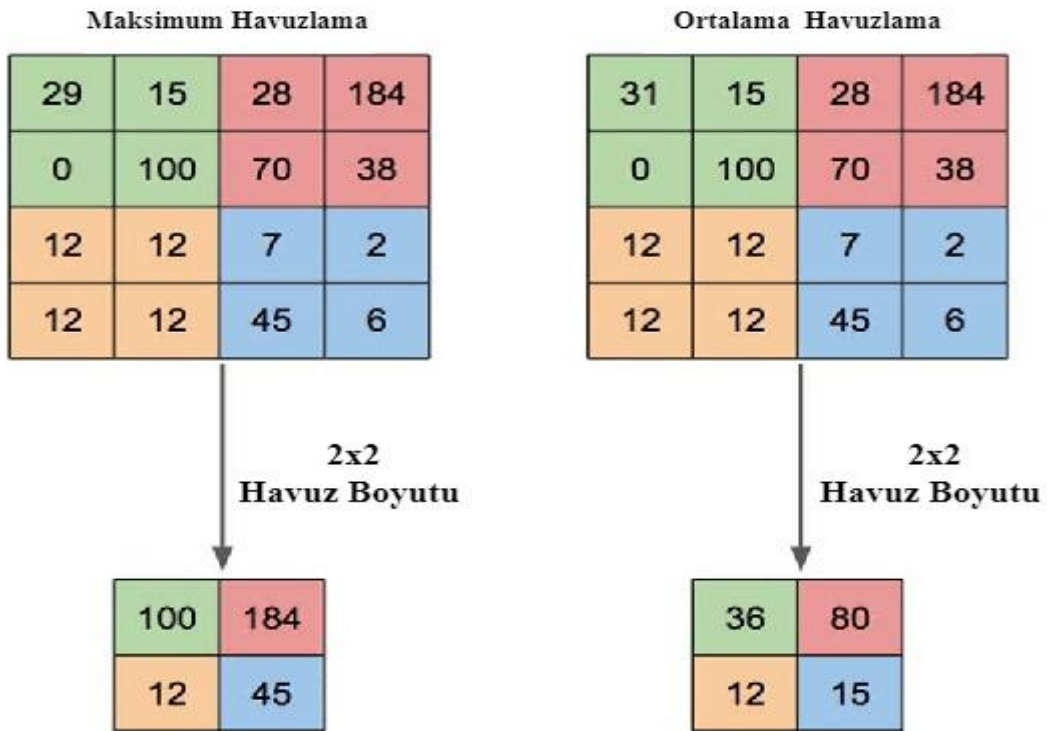
Şekil 4.8 CNN'in genel mimarisi (Natakarnkitkul, 2020)

4.1.2.1.1. Evrişimsel işlemi (Convolution operation)

CNN'de evrişim, görüntü işlemede filtreleme anlamına gelir. Ağın ilk kısmıdır. Evrişim, görüntü işlemede yaygın olarak kullanılan bir işlem olan çekirdeğin (filtre) iki boyutlu verilerin tamamı üzerinde tek tek kaydırılmasıdır.

4.1.2.1.2. Alt örnekleme işlemi (Havuzlama)

Havuzlama, özellik haritaları evrişim işlemiyle üretildikten sonra ve bir aktivasyon fonksiyonundan geçirildikten sonra, havuzlama işlemi uygulanır. Bu işlemin amacı, özellik haritalarının boyutunu ve veri kaybını minimuma indirmek ve daha az parametre ile daha hızlı bir öğrenme sürecini sağlamaktır. Bu işlem, aynı zamanda görüntü üzerinde filtre kaydırarak gerçekleştirilir (Sarigül, 2019). Maksimum havuzlama ve ortalama havuzlama olmak üzere iki tür havuzlama vardır. Şekil 4.9 ve Şekil 4.10'da gösterilmiştir.

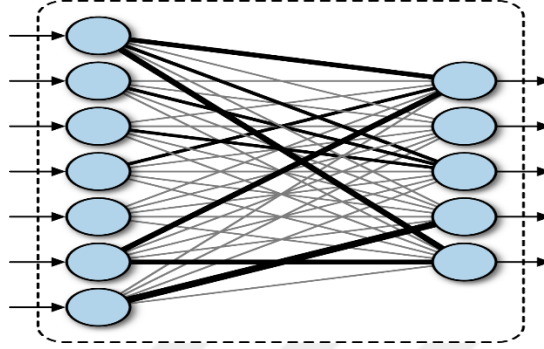


Şekil 4.9 Maksimum havuzlama (Yani ve ark, 2019) Şekil 4.10 Ortalama havuzlama(Yani ve ark, 2019)

4.1.2.1.3. Tam bağlı katman (Fully connected layers)

Öznetelik çıkarıcının son katmanındaki nöronlar, evrişim ve havuzlama katmanlar dahil olmak üzere, tek boyutlu bir vektöre dönüştürülür. Bu işlemden sonra gelen katmanlar

tam bağlantılı katmanlardır ve tüm nöronlar birbirine bağlanır, bu nedenle buna tam bağlı katman denir. Sınıflandırma çıktısı, daha önceki katmanlardan elde edilen özelliklere bağlı olarak üretilmektedir. Bu yapıyı genellikle sınıflandırma amacıyla bir Softmax çıktı katmanı izler. Şekil 4.11’de gösterilmiştir.



Şekil 4.11 Tam bağlı Katman (Ramsundar ve Zadeh, 2022)

4.1.2.1.4. Geri yayılım (Backpropagation)

Geri yayılım, YSA'lardaki ağırlıkları güncellemek adına gradyan hesaplamak için kullanılır. Geri yayılım, yaygın olarak kayıp fonksiyonun gradyanını hesaplayan dereceli azalma (gradient descent) tarafından kullanılır. Hataya, hatanın hesaplandığı çıkış nöronuna doğrudan veya dolaylı olarak bağlı olan önceki katmanlara ve nöronlara dağıtılır.

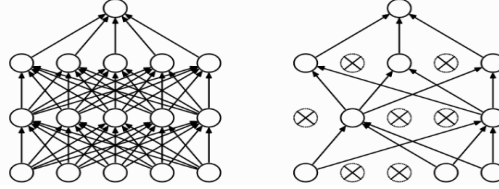
4.1.2.1.5. Toplu normalleştirme (Batch normalization)

Normalleştirme, eğitim sürecini hızlandırmak için derin öğrenme modellerinin gerekli bir katmanıdır. Normalleştirme içerisinde farklı türleri vardır; toplu normalleştirme (Batch normalization), grup normalleştirme, spektral normalleştirme ve örnek normalleştirme (instance normalization). Tartışılan normalleştirme yöntemlerin arasında yer alan Toplu Normalleştirme (Batch normalization), derin öğrenme mimarilerinin eğitimi için yaygın olarak kullanılan bir optimizasyon yöntemidir. Toplu Normalleştirme, derin öğrenme modellerinde normalize etme etkisine sahiptir (Ayanzadeh, 2020).

4.1.2.1.6. Seyreltme katmanı (Dropout)

CNN için hesaplama süresini azaltan ve eğitim sırasında aşırı öğrenmeyi (overfitting) engellemek adına bazı nöronları unutması için kullanılan katmandır. Eğitim sürecinin bir aşaması için ileri ve geri adım sırasında seçilen nöron aktiviteleri sıfırlanır. Her iterasyon

için farklı nöronlar aktif hale geldikçe, daha tutarlı ve güvenilir öğrenme sağlanır. Eğer ağırlığınız çok büyükse, çok uzun süre eğitim devam ediyorsa veya veri sayısı çok az ise aşırı öğrenme riski artabilir. Şekil 4.12’de seyreltme katmanı ile ilgili bir örnek gösterilmektedir.



Şekil 4.12 Seyreltme katmanı kullanılmadan önce ve kullandıktan sonra (Natakarnkitkul, 2020)

4.1.2.1.7 Aktivasyon fonksiyonları

Aktivasyon fonksiyonlar derin öğrenmenin esasını oluşturan sinir ağlarında en önemli role sahiptir. Derin sinir ağlarında, yapay nöronlar esasen girdi alma ve çıktı üretme özelliğine uyan doğrusal olmayan fonksiyonlardır. Gerçek dünya problemlerini tanıtmak için aktivasyon fonksiyonuna ihtiyaç vardır. Bir sinir hücresinde kullanılan aktivasyon fonksiyonu Şekil 5.11 de gösterilmiştir. Bir sinir ağındaki yapay nöron, belirli sayıda girdi alan ve buna bağlı olarak tek bir çıktı veren temel hesaplama birimidir. Aktivasyon fonksiyonununun matematiksel formülasyonu Denklem 4.1’de verilmiştir.

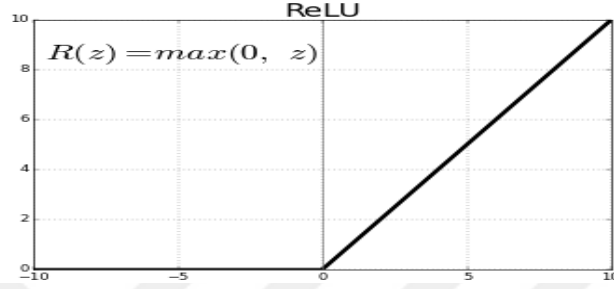
$$F(x) = \text{Aktivasyon} \left(\sum (W * x + b) \right) \quad (4.1)$$

Denklem 4.1’de verildiği gibi, x girdileri için W ağırlıkları çarpımını hesaplar ve b bias değeri ekleyerek bir $F(x)$ çıktısı üretir.

- **ReLU aktivasyon fonksiyonu:** Her evrişim katmanından sonra yaygın olarak ReLU katmanı kullanılır. ReLU, Rektifiye Doğrusal Birimler anlamına gelir. Sigmoid veya hiperbolik tanjant fonksiyonlarına benzer doğrusal olmayan aktivasyon fonksiyonu uygular. Evrişimli katmanlar sadece çarpma ve toplama gibi doğrusal işlemleri hesaplar. ReLU'nun avantajı, hesaplama açısından verimli olmasıdır (Top, 2018). Hızlı yakınsama ve sistemi geri yaymak için türev alma kolaylığı nedeniyle makine öğrenimi çalışmalarında en çok kullanılan fonksiyondur. ReLU aktivasyon fonksiyonununun matematiksel formülasyonu Denklem 4.2’de verilmiştir.

$$R(z) = \max(0, z) \quad (4.2)$$

Bu formülde R çıktıyı ve z giriş değeri temsil etmektedir. Şekil 4.13'te Relu fonksiyonuna ait grafik verilmiştir.

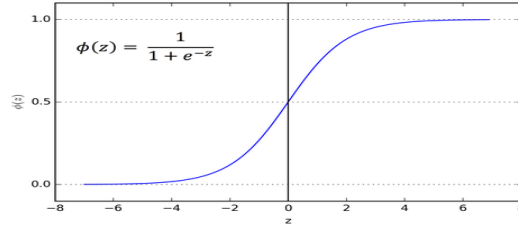


Şekil 4.13 Relu aktivasyon fonksiyonunun grafiği

- **Sigmoid aktivasyon fonksiyonu:** Sigmoid işlevi, 0 ile 1 arasındaki lineer giriş değerini sıkıştırır. Temel olarak, çoğu zaman 1 veya 0'a yakın değerler vererek, nöronun etkinleştirilip etkinleştirilmeyeceğine karar verir. Bir olayın olma olasılığını bulan modellerde kullanılır (Ada, 2019). Bildiğimiz gibi olasılık fonksiyonu 0 ile 1 arasında değer alır. Ve eğitim esnasında modelin takılıp kalmasına sebep olabilir. Bu sebepten dolayı karmaşık makine öğrenimi modellerinde kullanımı önerilmez. Ancak, daha karmaşık aktivasyon fonksiyonlarının ortaya çıkması sigmoid fonksiyonunun popülaritesini engelledi (Şekil 5.10). Sigmoid aktivasyon fonksiyonunun matematiksel formülasyonu Denklem 4.3'te verilmiştir.

$$\emptyset(z) = \frac{1}{1+e^{-z}} \quad (4.3)$$

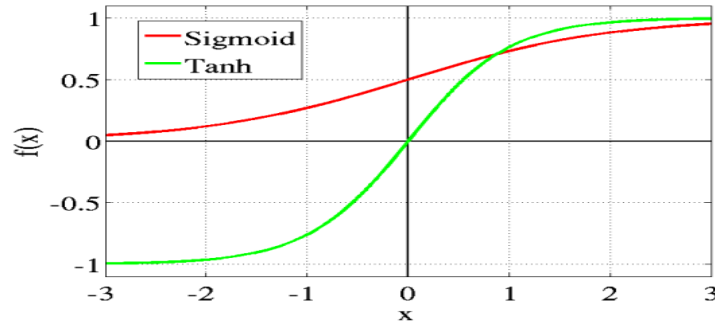
Bu formülde \emptyset çıktıyı ve z giriş değerini temsil etmektedir. Şekil 4.14 'te sigmoid fonksiyonu gösterilmektedir.



Şekil 4.15 Sigmoid aktivasyon fonksiyonunun grafiği

- **Hiperbolik tanjant aktivasyon fonksiyonu:** Sigmoid işlevine benzer şekilde tanh işlevi, girişi -1 ile 1 arasında sıkıştırır. Negatif ve pozitif çıkışlardaki varlığı geri yayılım sırasında ağ tanımlanabilirliğini artırır. Hiperbolik tanjant aktivasyon fonksiyonu çoğunlukla verileri iki sınıfa ayırmada kullanılır. Hem sigmoid hem de tanh aktivasyon fonksiyonları ileri beslemeli yapay sinir ağlarında kullanılır (Şekil 4.16). Hiperbolik tanjant aktivasyon fonksiyonunun matematiksel formülasyonu Denklem 4.4'te verilmiştir. Bu formülde $\tanh(z)$ çıktıyı ve z giriş değerini temsil etmektedir.

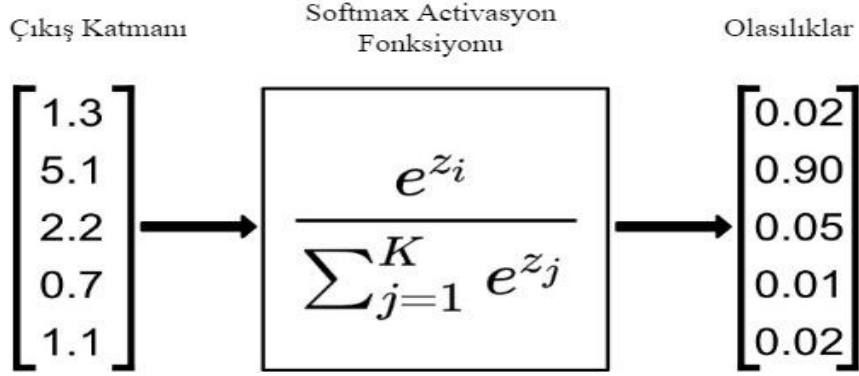
$$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (4.4)$$



Şekil 4.16 Hiperbolik tanjant fonksiyonunun grafiği

- **Softmax fonksiyonu:** girdilerin negatif veya pozitif olduğuna bakılmaz toplamda 1 değerine sahip olacak şekilde bir olasılık fonksiyonudur. Softmax fonksiyonu genellikle bir sınıflandırıcının çıkış katmanında yer almaktadır. Diğer katman türlerinden farklı olarak çıkış katmanındaki bir nöronun çıktısı katmandaki diğer tüm nöronların çıktılarına bağlıdır. Bunun nedeni, tüm çıktılarının toplamının 1'e eşit olması gerekliliğidir. Güçlü bir tahmin, vektörde 1'e yakın tek bir girişe sahipken, kalan girişler 0'a yakındır. Zayıf bir tahminde, aşağı yukarı eşit derecede

birden fazla olası etiket bulunmaktadır. Şekil 5.17’de softmax aktivasyon fonksiyonu gösterilmektedir.



Şekil 4.17 Softmax aktivasyon fonksiyonu (Radečić, 2020)

4.1.2.1.8. Toplu iş boyutu (Batch size)

Yapay sinir ağlarında veri seti miktarı öğrenme sürecinin başarısını arttırmaktadır. Aynı zamanda veri setinin miktarı eğitim için harcanan zamanı ve öğrenme sonunda elde edilen modelin boyutunu artırmaktadır. Ağın öğrenme aşamasında veri setindeki tüm verilerin aynı anda öğrenilmesi, öğrenme süresi ve hafıza açısından önemlidir. Öğrenmenin her yinelenmesinde, ağırlıkları buna göre güncellemek için ağda geri yayılım süreciyle gradyan inişi hesaplanır. Bu hesaplamadaki veri sayısı ne kadar yüksek olursa, işlemin tamamlanması o kadar uzun sürer. Bu problemi çözmek için veri seti küçük gruplara bölünür ve eğitim bu küçük gruplarla yapılır (Cura, 2919).

4.1.2.1.9. Epok sayısı

Model eğitilirken, tüm veriler aynı anda eğitime dahil edilmez. Ağ verileri parçalar halinde alınır, ilk kısım eğitilir, modelin performansı test edilir ve geri yayılım kullanılarak ağırlıklar güncellenir. Daha sonra model yeni veri seti ile yeniden eğitilir ve ağırlıklar tekrar güncellenir. Model için en uygun ağırlık değerlerini hesaplamak için bu işlem her eğitim adımında tekrarlanır. Bu eğitim adımlarının her birine bir epok denir. Epok sayısı arttıkça modelin performansı önemli ölçüde artmaktadır. Belli bir epok değerinden sonra performans çok küçük birimlerde artacağından bu noktada eğitim sonlandırılabilir (Skogly, 2019). Başarımı etkileyen faktörlerden biri de ne zaman eğitim sürecinin durdurulacağıdır. Çünkü epok sayısı arttıkça, ağlar eğitim örneklerine aşırı uyum sağlama eğilimidir. Bununla birlikte minimum epok sayısı; konunun doğasına, veri

seti arasındaki varyansa ve eğitim verileri ile test verileri arasındaki benzerliğe büyük ölçüde bağlıdır(Özgenel, 2018). Bu çalışma kapsamında tüm ağlar, değişen veri seti boyutları ile 30 epok için eğitilmiştir.

4.1.2.1.10. Öğrenme hızı (learning rate)

CNN model performansını artırmak için kullanılan yöntemlerden biri de öğrenme hızı optimizasyonudur. Öğrenme oranı, ağın öğrenme sürecinde oluşan hata miktarına bağlı olarak ağ parametrelerinin güncellenmesinde kullanılan bir katsayıdır. Öğrenme hızı çok küçük ise ağ parametreleri çok küçük adımlarla güncellenir ve işlem uzun zaman alır. Yüksek öğrenme oranı, ağın hatayı en aza indiren optimum noktayı kaçırmaya neden olabilir. Bu nedenle, öğrenme oranı optimizasyonu çok önemlidir. Öğrenme oranı, duruma göre dinamik olarak ayarlanmış bir parametredir (Sevli ve Kemaloğlu, 2020). Öğrenme hızı çoğunlukla 0,1 ile 0,000001 arasında belirlenmektedir. Beklendiği gibi, tüm problemler için uygun bir öğrenme oranı değeri tanımlamak mümkün değildir. Belirli bir problemdeki öğrenme oranı için optimal değer, farklı değerler için tahmin performansı gözlemlenerek aranmalıdır (Hariri, 2021). Bu çalışmada kullanılan model için öğrenme katsayısı 0.001 olarak belirlenmiştir.

4.1.2.1.11. Kayıp fonksiyonu (Loss Function)

Kayıp fonksiyonu, optimal yaklaşımı elde etmek için parametrelerin nasıl güncelleneceğini belirleyen bir sinir ağının en önemli bileşenidir. Sinir ağı optimizasyonunda, çoğu zaman mini toplu gradyan inişi, bir grup girdi verisi üzerinden gerçekleştirilir (Ada, 2019). Veri kaybı, her bir girdi kaybının ortalamasıdır. Bu kayıplar makine öğrenmesinde önemli bir konudur. Maliyet fonksiyonu olarak da adlandırılan kayıp fonksiyonu, sinir ağı modellerinin veri seti için ne kadar iyi olduğunu tespit etmek için eğitim sürecindeki tahmin ile gerçeği etiketi arasındaki farkı değerlendirir. Kayıp fonksiyonunun çıktısı modelin başarısı ile ters orantılıdır, yani çıktıdaki daha yüksek sayılar başarısız modelin işaretidir. Kayıp fonksiyonu, türevinin negatif yönünde ağırlıkları güncellemek için geri yayılım sırasında hesaplanan hatayı kullanmaktadır. Sınıfsal çapraz entropi en çok tercih edilen kayıp fonksiyonlarından biridir. Bu çalışmada kategorik çapraz entropi kayıp fonksiyonu tercih edilmiştir. ikili çapraz entropi kayıp fonksiyonunun genelleştirilmiş hali ikili sınıflandırma problemine uygulanmıştır.

Kategorik vektör ile vektör burada kullanılmıştır. Denklem 4.5 ve 4.6'da gösterilmektedir.

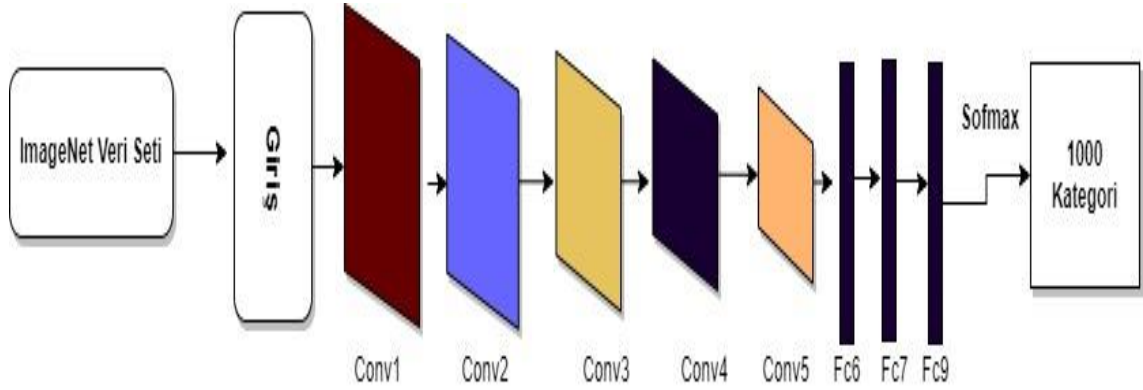
$$L(X_i, Y_i) = - \sum_{j=1}^c Y_{ij} * \log(p_{ij}) \quad (4.5)$$

$$Y_{ij} = \begin{cases} 1 & \text{Eğer } i. \text{ eleman } j. \text{ sınıftaysa} \\ 0 & \text{değilse} \end{cases} \quad (4.6)$$

Denklem (4.5) ve (4.6) deki Y_i kategorik vektör ($Y_{i1}, Y_{i2}, Y_{i3} \dots Y_{ic}$) ve $P_{ij} = f(x_{ij})$ i. elmanların j . sınıfta olma olasılığını ifade etmektedir.

4.1.2.1.12. AlexNet

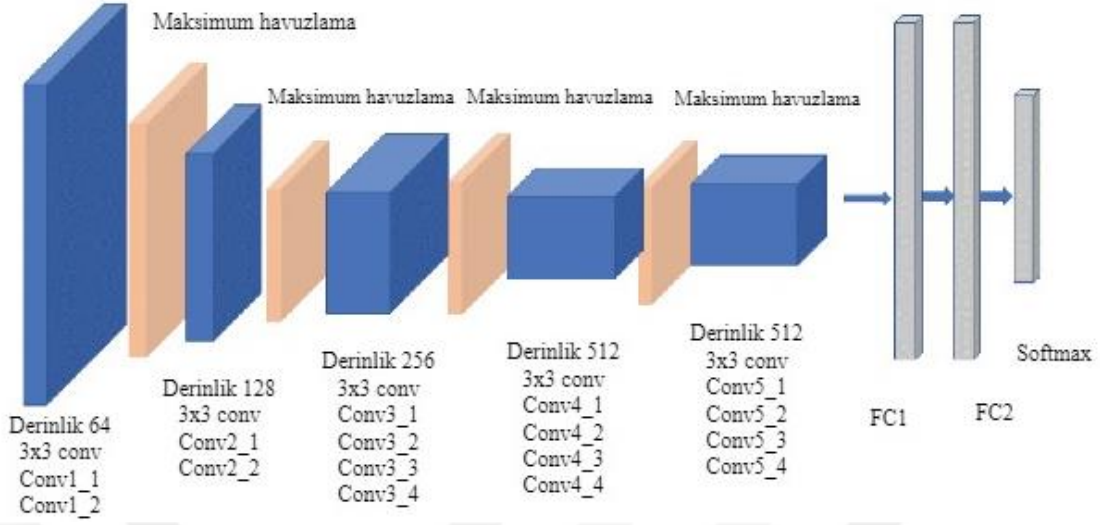
AlexNet mimarisi 2012 yılında Krizhevsky ve arkadaşları tarafından inşa edilmiştir. AlexNet; maksimum havuzlama, evrişimli ve tam bağlı katman dahil olmak üzere sekiz farklı katman kullanan CNN tabanlı önde gelen bir algoritmadır. Model ILSVRC 2012 ImageNet veri seti ile eğitilmiştir ve Büyük Ölçekli Görsel Tanıma Yarışmasında görüntü sınıflandırması için en iyi sinir ağı modeli ilan edilmiştir (Shaheen ve ark, 2021). AlexNet nesne tanıma problemleri için çok yüksek performans sergilemiştir. Daha yüksek doğruluk oranları elde etmek için daha fazla eğitim verisine ihtiyaç duyulmaktadır. AlexNet mimarisinde giriş görüntü boyutu $227 \times 227 \times 3$ olarak tanımlanmıştır. İlk konvolüsyon katmanında $11 \times 11 \times 3$ boyutlarında 96 filtre sayısı bulunmaktadır. Sonrasında normalizasyon ve havuzlama işlemi yapılır. Bu işlemlerin sonrasında $5 \times 5 \times 256$ bir konvolüsyon katmanı, $3 \times 3 \times 384$ boyutlarında iki adet daha konvolüsyon katmanı bulunan ardından üç adet tam bağlı katmanlı kullanılmıştır. Elde edilen özellikler vektöre dönüştürülerek softmax sınıflandırıcısında sınıflandırma gerçekleştirilir. Şekil 4.18'de gösterilmektedir.



Şekil 4.18 AlexNet mimarisi

4.1.2.1.13. Vgg19

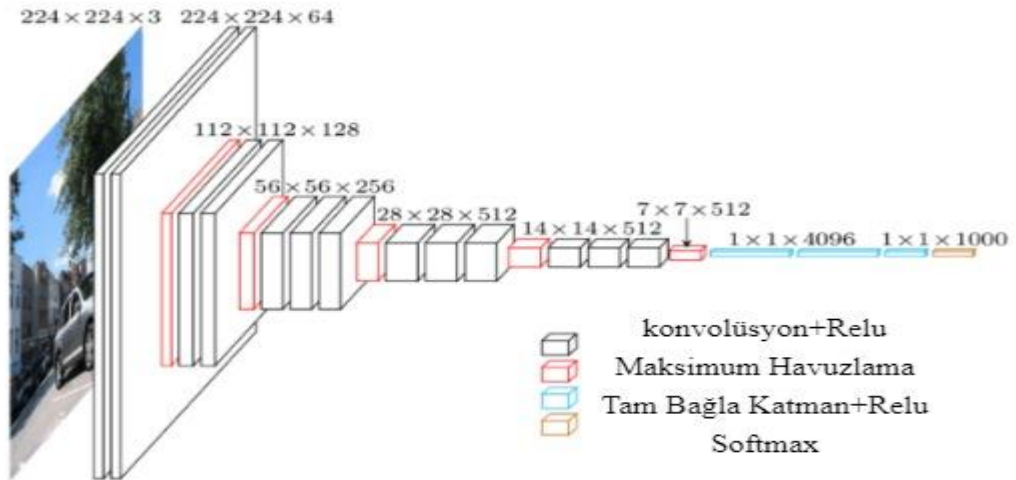
Vgg19 derin öğrenme algoritması 2014 yılında Oxford üniversitesi Görsel Geometri Grubu tarafından geliştirilmiştir. 16 adet evrişim katmanları, 3 adet tam bağlantılı katmanları, 5 adet Maxpool katmanları ve 1 adet Softmax katmandan üretilmiştir. Relu katmanı ve seyretme katmanı birlikte toplam 47 katman sahiptir. Giriş katmanı 224x224x3 boyutundadır. En son katman sınıflandırma katmanıdır. Imagenet veritabanında %88 doğruluk oranına ulaşmıştır (Doğan, 2018). Büyük Ölçekli Görsel Tanımayı Tanıma yarışması için tasarlanmıştır. Bu algoritmanın en büyük avantajı kodunun açık kaynak kodlu olması ve transfer öğrenmenin kolayca uygulanabilmesi ve ağır diğer mimariler için çalışmasını sağlayabilmesidir. Algoritma aynı zamanda tek bir büyük çekirdek yerine küçük boyutlu çekirdeği de öğrenir çünkü daha küçük boyutlu çekirdeklerin dahil edilmesi sistemin karmaşık özellikleri öğrenmesini sağlar. Bu mimari, ilk 5 puan tablosunda yer almaktadır (T ve ark, 2021). Şekil 4.19’de gösterilmiştir.



Şekil 4.19 VGG 19 mimarisi (Arishanapally, 2019)

4.1.2.1.14. Vgg16

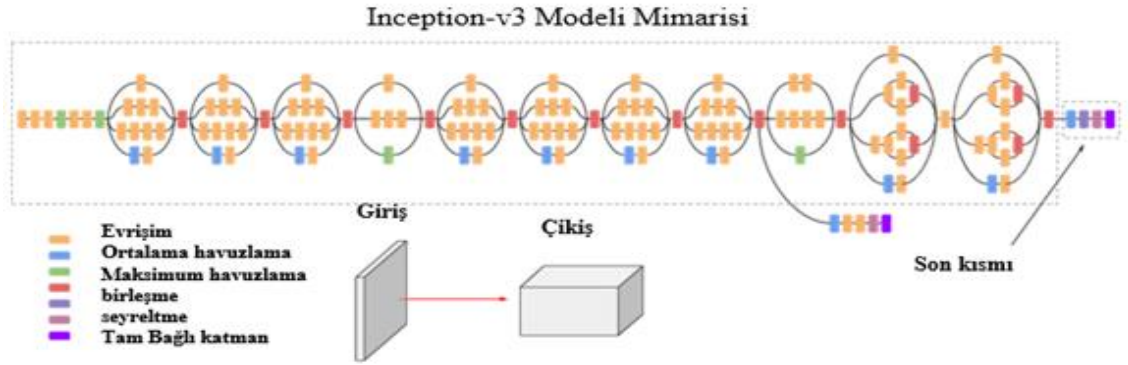
VGG 16 mimarisi Oxford üniversitesi Görsel Geometri Grubu tarafından 13 konvolüsyon ve 3 tam bağlı katmandan bir ağ geliştirmiştir. Geliştiren derin öğrenme algoritmasını ILSVRC2014 yarışmasında daha iyi sonuçlar elde edilmesi için kullanmıştır. VGG 16 mimarisi havuzlama katmanı (Maxpool), tam bağlı katmanı (Full connected Layer), Relu katmanı (Relu Layer), Seyreltme katmanı (Dropout Layer) ve softmax katmanı (Softmax Layer) katmanlarıyla birlikte toplamda 41 katman sahiptir. Giriş katmanının $224 \times 224 \times 3$ boyutundadır. En son katman sınıflandırma katmanıdır. ImageNet veri tabanında %89 doğruluk oranı yakalamıştır (Doğan, 2018). Şekil 4.20'de gösterilmiştir.



Şekil 4.20 VGG 16 mimarisi (Sugata ve Yang, 2017).

4.1.2.1.15. Inception

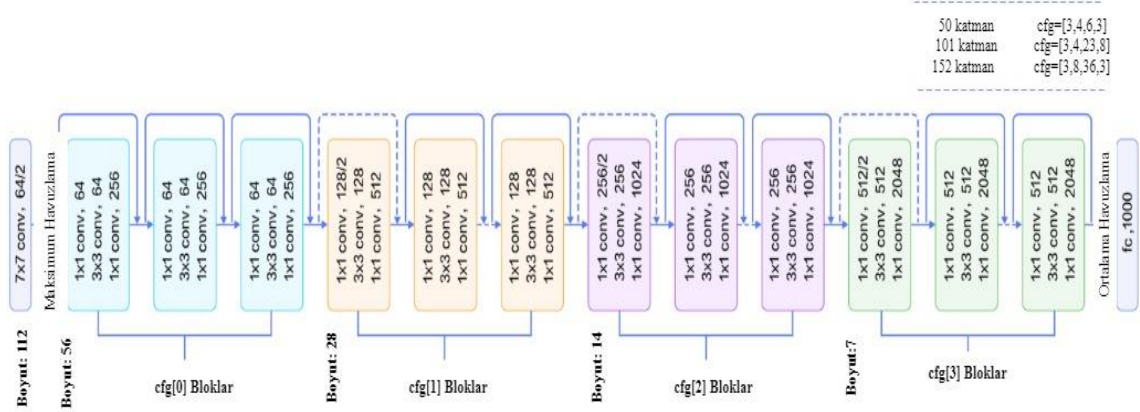
Inception mimarisi 2012 yılında Google tarafından geliştirilmiştir. Çoklu evrişim katmanlarından oluşturulmuştur. Giriş verilerini 1×1 evrişim katmanı ile işlemden geçirdikten sonra, veri evrişim katmanı kanalının her bir kanalından paralel olarak geçer. Tüm kanal çıktıları en sonda birleştirilerek derleme süreci tamamlanır. Inception mimarisinin bir başka avantajı da ağı sonundaki tam bağlı katmanın kaldırılabilmesidir. Bu durum, düşük parametre ile çok fazla katman çalıştırmaya imkân tanımaktadır. VGG mimarisine göre 5 Milyon parametre ile çok düşük parametre kullanım mimarisine sahiptir. InceptionV2'den farklı olarak Inception-V3 ağında yardımcı sınıflandırıcının yalnızca evrişim katmanlarını haricinde tam bağlı katmanı da (fully connected layer) normalize etmesi ile ortaya çıkar (Szegedy ve ark, 2016). Şekil 4.21'de gösterilmiştir.



Şekil 4.21 Inception-v3 (Szegedy ve ark, 2016)

4.1.2.1.16. ResNet

Derin Konvolüsyonel Sinir Ağı yapılarının oluşturulması, 152 katmandan oluşarak şu ana kadar ki tüm mimarilerden daha derin olarak tasarlanmıştır ve parametre sayısını azaltmak amacıyla blok yapıları kullanılmıştır. ResNet yapısında 1×1 ve 2×2 boyutlarında filtreler kullanılmıştır. Ağ sonunda 1000 adet özellik elde edilir ve Softmax sınıflandırıcısı eğitilir. ResNet-50 modelinde dropout katmanı bulunmamaktadır. Resnet50 mimarisinde 177 katmandan meydana gelen bir ağ yer almaktadır ve giriş katmanı $224 \times 224 \times 3$ boyutundadır. ResNet yapılarının farklı modelleri mevcuttur. Eğitim sürecinde daha yüksek özelliklere sahip donanımlara ihtiyaç duyar. Şekil 4.22'de ResNet-50 modelin mimarisi gösterilmiştir.



Şekil 4.22 Resnet-50 (Rastogi, 2022)

4.1.2.1.17 Xception

Xception mimarisi, 2017 yılında François Chollet tarafından tasarlanmıştır. Bu ağ Inception ağlarının üzerine inşa edilmiş bir ağ yapısıdır. Xception mimarisi her 2B boyutlu veriye ayrı ayrı bakar ve sonra 1x1 evrişimi gerçekleştirir. 2 boyutlu ve 1 boyutlu haritalamanın bu birleşimi, 3 boyutlu veri setinde eğitimi daha az hesaplama ile tamamlamaya olanak sağlar. Xception modeli, InceptionV3 ile aynı sayıda parametreye sahiptir ve “Extreme Inception (Aşırı Başlama)” olarak da adlandırılır.

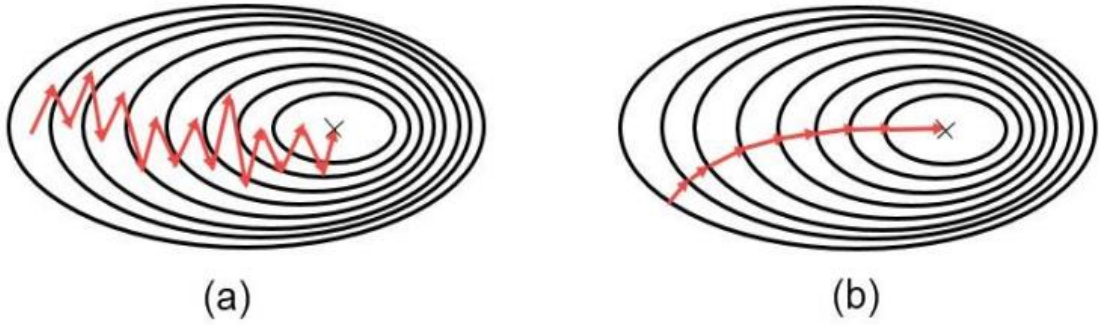
4.1.2.2. Optimizasyon algoritmaları

Optimize, derin öğrenme modellerinde odaklanılması gereken en önemli konulardan biridir. Uygun bir optimizasyon kullanılması, en iyi sonucun elde edilmesini sağlar, çünkü optimize, kayıp fonksiyonunu en aza indirmek için ağırlıkları güncellemekten sorumludur.

4.1.2.2.1. Dereceli azalma (Gradient descent)

Dereceli azalma (GD), sinir ağlarının kaybını ve doğruluğunu optimize etmek için basit bir optimizasyon prosedürüdür. Dereceli azalma fikrinin temeli parametrelerinin tekrarlı güncellenmesine dayanmaktadır. Hata fonksiyonunu en aza indirmek için veri setindeki tüm örnekleri kullanır. Bu işlem, zaman alıcı olduğu için yalnızca veri seti küçükse geçerlidir. Öte yandan, eğer veri seti büyükse, bu süreç aynı zamanda çoklu geri yayılım hesaplamalarını da içerdiğinden, hesaplama açısından pahalı ve zaman alıcı olacaktır. Bu nedenle olasılıksal dereceli azalma (SGD) adı verilen daha yeni bir yöntem

geliştirilmiştir. SGD'nin fikri, her adımda bir eğitim kümesinden bir örnek seçerek ve parametreleri yalnızca bu tek ve rastgele seçilen kayda dayalı olarak güncellemektir (Hariri, 2021). Şekil 4.23'te, GD ve SGD arasındaki fark açıklanmaktadır. GD işlemi global optimuma giden yolda daha kararlı görünürken, SGD optimizasyon için daha hızlıdır. Şekil 4.23'te gösterilmektedir.



Şekil 4.23 SGD (b) ve GD (a) arasındaki yakınsama farkı (Hariri, 2021)

4.1.2.2.2. Momentum gradient descent

Sabit bir öğrenme oranı, parametre değerleri optimal noktaya yaklaştığında salınımlara neden olacaktır. Standart Momentum Gradient Descent, minimuma doğru hızı artırmak için önceki gradyan değerlerinden yararlanır. SGD gürültülü güncellemeler yapar ancak momentum, daha az varyansla en uygun noktayı daha hızlı bulmak için geliştirilmiş bir yöntemdir.

4.1.2.2.3. RMSprop

RMSprop, gradyan tabanlı bir optimizasyon tekniğidir. Öğrenme oranını aşırı derecede düşürme sorununa çözüm sunar. Geçmiş eğimlerin karelerinden elde edilen tüm değerleri kullanmak yerine miktarı sınırlandırmıştır. RMSprop, kayıp fonksiyonundaki salınımları azaltmaya yardımcı olur.

4.1.2.2.4. Adam

Adam, stokastik yapıdaki amaç fonksiyonlarının birinci dereceden gradyan temelli iyileştirmesi için önerilmiştir. Düşük dereceli momentlerin uyarlanabilir tahminlerine

dayanmaktadır. Bu yöntem ayrıca her yinelemede öğrenme oranını değiştirir. Denklem 4.7-12'de gösterilmiştir.

$$n_t = \emptyset_1 n_{t-1} + (1 - \emptyset_1) g_t \quad (4.7)$$

$$V_t = \emptyset_2 V_{t-1} + (1 - \emptyset_2) g_t^2 \quad (4.8)$$

$$\hat{n}_t = \frac{n_t}{1 - \emptyset_1^t} \quad (4.9)$$

$$\tilde{V}_t = \frac{V_t}{1 - \emptyset_2^t} \quad (4.10)$$

$$W_{t+1} = W_t - \frac{\alpha}{\sqrt{\tilde{V}_t + \varepsilon}} \cdot \hat{n}_t \quad (4.11)$$

Burada n_t birinci anın tahminidir ve V_t gradyanların ikinci anının tahminidir. \emptyset_1 birinci momentum terimidir ve \emptyset_2 ikinci momentum terimidir ve genellikle sırasıyla 0,9 ve 0,999'a ayarlanır. \emptyset değerleri \hat{n}_t ve \tilde{V}_t hesaplamak için kullanılır.

4.1.2.3. Düzenleştirme teknikleri (Regularization techniques)

Yaygın olarak kullanılan sinir ağı düzenleme teknikleri bu bölümde ayrıntılı olarak açıklanmaktadır.

4.1.2.3.1. Veri artırma (Data augmentation)

Veri artırma, derin öğrenmede yaygın olarak kullanılan bir işleme yaklaşımıdır. Orijinal örneğe benzer ancak aynı olmayan bir görüntü dizisi oluşturmak için mevcut görüntülere rastgele dönüşümler gerçekleştirerek eğitim örneklerinin çeşitliliğini ve sayısını yapay olarak artırır. Yakınlaştırma, kırpma, döndürme, çevirme veya parlaklığı artırma veya azaltma vb. gibi bazı değişiklikler yaparak, eğitimi genişletmeye yardımcı olan tekniklerdir. Veri büyütme, aşırı uyumu önlemek için kullanılır ve özellikle eğitim veri kümesi nispeten küçük olduğunda faydalıdır. Bazı büyütme algoritmalarının sağlamlığını artırırken, alakasız dönüşümler öğrenmeyi zorlaştırabilir ve eğitim setine yeni veriler eklemek, model karmaşıklığını ve modeli oluşturmak için gereken süreyi artıracaktır (Ayanzadeh, 2020).

4.1.2.3.2. Erken durdurma (Early stopping)

Erken durdurma, eğitim verilerinin bir kısmını bir doğrulama seti olarak kullanarak aşırı uyumu (overfitting) azaltmak için bir tekniktir. Eğitim süreci bu verileri içermez. Doğrulama setinin hatası belli bir miktara ulaşırsa eğitim durdurulur. Eğitim verileri için mevcut sınır ağında bir aşırı uyum olduğu söylenebilir. Erken durdurmanın önemli bir noktası, doğrulama kümesinin seçilmesidir. Tüm verileri temsil etmelidir. Modelin eğitim verilerinin ötesinde ne kadar iyi genelleme yaptığı anlaşılabilir (Ergin, 2020).



5. BAŞARI ÖLÇÜM METRİKLERİ

Algoritmanın etkinliğini gösterebilmek için görüntülerin farklı açılardan değerlendirilmesi gereklidir. Ayrıca bazı standart değerlendirme ölçütleri, bazı durumlarda algoritmanın gerçek performansını farklı bir bakış açısıyla gösterememektedir. Bu nedenle, yöntemin değerlendirilmesi için bir metrik kullanılması, algoritmanın gerçek performansının yanlış anlaşılmasını engeller. Bu çalışmada bal peteği veri kümesinde hem transfer öğrenme hem de derin öğrenme modelini değerlendirmek için aşağıdaki farklı metrikler kullanılmıştır. Metrikleri hesaplamak için ilk önce karışıklık matrisini anlamamız gerekiyor. Karışıklık matrisi $n*n$ boyutlu bir matristir. Burada n değeri sınıf sayısını göstermektedir ve boyutlar buna bağlı olarak oluşur. Doğru tahmin edilen tüm pikseller yalnızca köşegen üzerinde bulunmaktadır. Yanlış sınıflandırılmış pikseller ise köşegen dışında bulunmaktadır. Genel Doğruluk (Overall Accuracy (OA)), F-skoru, Kesinlik (Precision), Duyarlılık (Recall) ve Cohens Kappa gibi doğruluk değerlendirme metrikleri aşağıda Şekil 5.1’de gösterilen karışıklık matrisi ile hesaplanabilmektedir.

		Tahmini Değerler	
Gerçek Değerler	Gerçek Pozitif (TP)	Yanlış Negatif (FN)	
	Yanlış pozitif (FP)	Gerçek Negatif (TN)	

Şekil 5.1 Doğruluk değerlendirme matrisi

5.1. Genel Doğruluk

Genel doğrulu, sınıflandırıcının kaç kez doğru sınıflandırdığını temsil eder. Kısaca doğru tahmin edilen miktardır. Mümkün olduğunca yüksek olması daha iyidir. Denklem 5.1’deki şekilde tanımlanabilir.

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

5.2. Duyarlılık (Recall)

Belirli bir sınıftaki bir değer kaç defa doğru tahmin ettiğini göstermektedir. Sınıflara atanan pikseller yardımıyla sınıflandırma sonucu hesaplanır. Denklem 5.2’de matematiksel olarak tanımlanmıştır.

$$\text{Duyarlılık} = \frac{TP}{TP + FN} \quad (5.2)$$

5.3. Kesinlik (Precision)

Kesinlik, doğru sınıflara ait tanımlanmış görüntülerin sayısını temsil eder. Denklem 5.3’de hesaplama formülü verilmiştir.

$$\text{Kesinlik} = \frac{TP}{TP + FP} \quad (5.3)$$

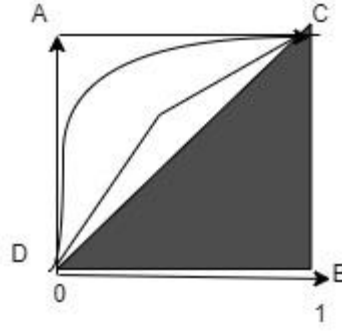
5.4. F1-Skor

F1-skor değeri kesinlik ve duyarlılığın harmonik ortalaması alınarak hesaplanmaktadır. Denklem 5.4’te gösterilmektedir.

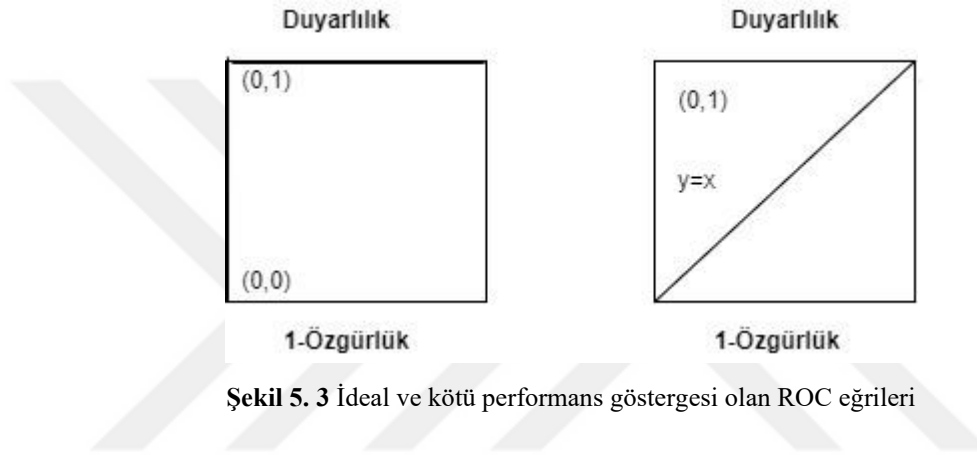
$$\text{F1 – Skor} = 2 * \frac{\text{Kesinlik} * \text{Duyarlılık}}{\text{Kesinlik} + \text{Duyarlılık}} \quad (5.4)$$

5.5. AUC-ROC Eğrisi

Tüm ölçüler TP, TN, FP, FN sabit değerlerine dayanır. Bu gibi değerler, sınıf etiketleri ve tahmin değerlerinin ikisi de ayrık (discrete) olduğunda kolayca toplanabilir. TP ve FP değerlerini birbirine ortak bağlayan ve bir 2-D eksen üzerinde çizilen, ROC (Receiver Operating Curve) grafiği Şekil 5.2’de gösterilmiştir. ROC eğrisi, modellerin doğruluk değerlendirilmesi ve kıyaslanması için en çok kullanılan yöntemdir. Bir olasılık eğrisidir ve AUC ayrılabilirliğin derecesini veya ölçüsünü gösterir. Sınıflar arasında ne kadar model ayırt edebileceğini gösterir. AUC yükseldikçe, model daha iyi tahmin etmektedir. Mükemmel bir modelde 1'e yakınlık iyi bir ayrılma ölçüsü olduğu anlamına gelir. Kötü bir modelde 0'a yakınlık AUC’de en kötü ayrılabilirlik ölçüsü anlamına gelir. AUC 0.5 olduğunda, modelin hiçbir şekilde sınıf ayırma kapasitesine sahip olmadığı anlamına gelir (İş, 2021). Şekil 5.3 ideal ve kötü performans göstergesi olan AUC eğrileri gösterilmektedir.



Şekil 5.2 AUC (ROC eğrisi altındaki alan)



Şekil 5. 3 İdeal ve kötü performans göstergesi olan ROC eğrileri

5.6. Geometrik Ortalama

Genellikle, bir sayı kümesindeki sayıların ortalaması ve bu kümedeki sayıların merkezi sayı kümesinin ortalaması ile ifade edilir. Sayı kümesinin ortalamasının ifade edilmesinde kullanılan birçok matematiksel denklem bulunmaktadır. Bunlardan birisi de geometrik ortalamadır. Geometrik ortalama n adet sayı kümesindeki değerlerin birbirleri ile çarpım sonucu yine n 'inci derecedeki kökü alınarak hesaplanır. Burada geometrik ortalamanın pozitif sayılar için geçerli olacağı unutulmamalıdır. Geometrik ortalama, aşağıdaki gibi matematiksel olarak tanımlanır; burada n , kümedeki değerlerin sayısına eşittir ve x , küme içindeki belirli bir sayıdır. Denklem 5.5'te matematiksel olarak gösterilmiştir.

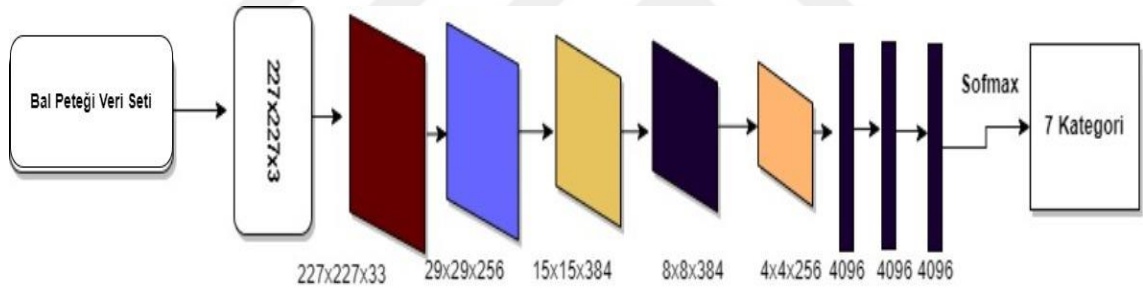
$$\left(\prod_{i=1}^n x_i \right)^{\frac{1}{n}} = \sqrt[n]{x_1 x_2 \dots x_n} \quad (5.5)$$

Denklem 5.5 teki \prod bir dizi ögenin matematiksel ürünü, x sette ayrı bir öge, i , kümedeki ayrı bir ögenin dizin numarası; burada $i = 1$, kümedeki öge 1 ile başlanıldığını gösterir ve n kümedeki toplam öge sayısı göstermektedir.

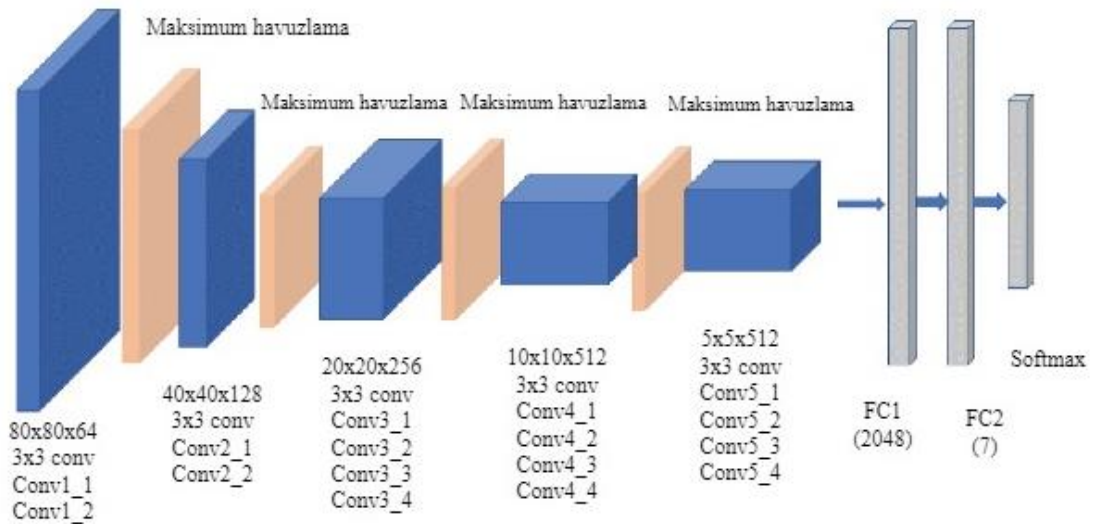


6. DENEYSSEL ÇALIŞMALAR

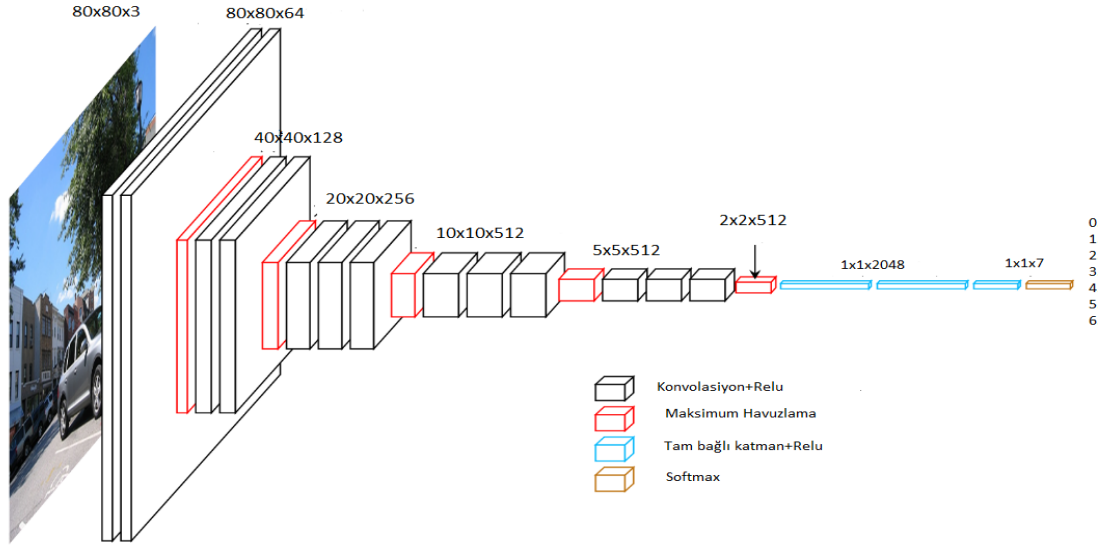
Tez çalışmasının bu kısmında derin öğrenme ile yapılan uygulamada kullanılan veri seti nitelikleri, algoritmalar, hiper parametreler ve donanımsal araçlar açıklanmıştır. Uygulamada toplam 19 adet bal peteğindeki görüntülerden 129314 adet 7 sınıftan oluşan görüntüler işaretlenmiştir. Bu görüntülerin 103455 adeti eğitim ve 25859 adeti test için kullanılmıştır. Bu görüntülerin %80 eğitim için %20 ise test için kullanılmıştır. Eğitim ve test görüntüleri rastgele olarak seçilmiştir. Bu veri setinin üzerinde yedi farklı derin öğrenme yöntemleri kullanılarak sonuçlar karşılaştırılmıştır. Kullanılan bilgisayar sisteminde; i7 2.8ghz işlemci, Gtx 960m, NVIDIA GeForce 2080 Ti 6GB GPU kartı ve 128GB RAM özelliklerine sahip donanım bulunmaktadır. Dizüstü ortamındaki uygulamalar tensorflow kütüphanesi ve python programlama dili kullanılarak yapılmıştır. Bu çalışmada AlexNet, Vgg16, Vgg19 ve ResNet50 +Xception, Inception-v3 ve SSCNN öğrenme algoritmaları kullanılmıştır. Kullanılmış mimariler şekil 6.1-6.3 'te gösterilmiştir.



Şekil 6.1 AlexNet mimarisi



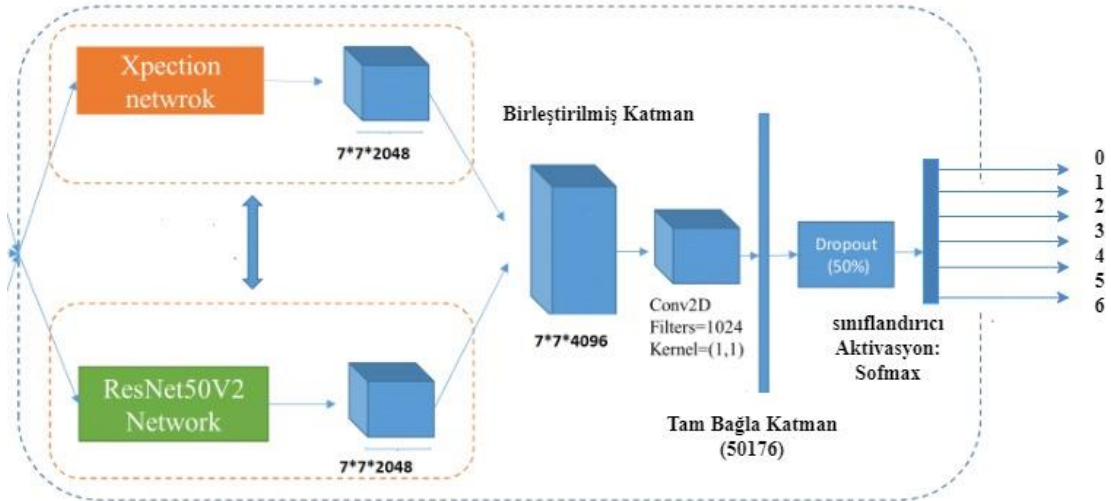
Şekil 6.2 VGG 16 mimarisi (Sugata ve Yang, 2017)



Şekil 6.3 VGG 19 mimarisi (Arishanapally, 2019)

- **Xception+Resnet50**

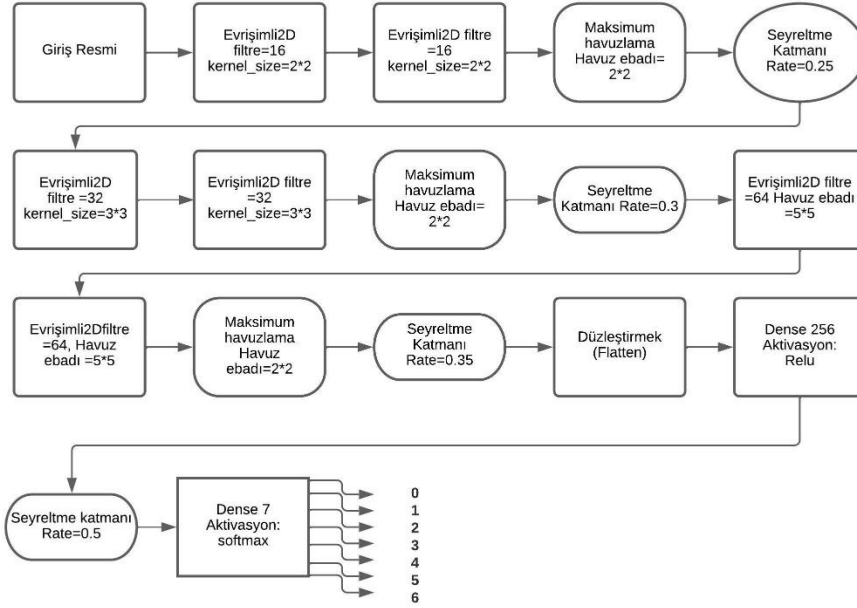
Xception ve Resnet50 mimarilerini birleştirip bal peteğin sınıflandırması için kullanmıştır ve Şekil 6.4’te gösterilmiştir.



Şekil 6.4 Xception+ Resnet50 mimarisi (Rahimzadeh ve Attar, 2020)

- **Öz-yapılı evrimsel sinir ağı (Self-Structured CNN)**

Bal Peteğin tespit ve sınıflandırması için SSCNN mimarisi kullanılmıştır. Modeldeki filtrelerin sayısı 16, 16, 32, 32 ve 64 olarak seçilmiştir. Filtre boyutların sırasıyla 2x2, 2x2, 3x3, 3x3, 5x5 olarak seçilmiştir. Şekil 6.5 ‘de SSCNN sınıflandırma modeli gösterilmektedir.



Şekil 6.5 SSCNN mimarisi

Veri seti üzerinde uygulanan derin öğrenme yöntemlerin toplam 30 epok eğitim gerçekleştirilmiş ve uygulama sonuçları analiz edilmiştir. Farklı yedi modelin en yüksek başarı oranı AlexNet modelin %95, ikinci sırada SSCNN modelin %94 başarı oranı, üçüncü sırada Inception-v3 ve VGG16 modellerin başarı oranı %89, dördüncü sırada VGG19 ve Resnet+ Xception modellerin başarı oranı %88 göstermektedir. Sın (sınıfın), Kes (kesinlik), Duyarlılık (Duy), F1 (F1-skor), Des (Destek), Doğruluk oranı, Marko ortalama ve Ağırlıklı ortalama Çizelge 6.1’de verilmiştir.

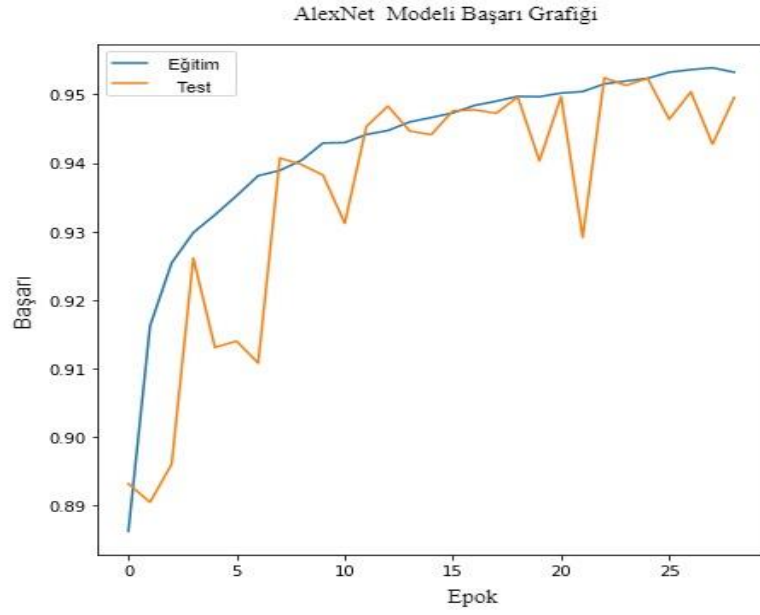
Çizelge 6.1 Modellerin karşılaştırılması

Sınıflar	Alex Net				Inception v3				VGG 16				VGG 19				Xception+ResNet-50				SSCNN			
	Kes.	Duy.	F1.	Des.	Kes.	Duy.	F1.	Des.	Kes.	Duy.	F1.	Des.	Kes.	Duy.	F1.	Des.	Kes.	Duy.	F1.	Des.	Kes.	Duy.	F1.	Des.
Sımf 0	0.96	0.96	0.96	13695	0.83	0.96	0.89	5352	0.81	0.96	0.88	5352	0.80	0.97	0.88	5352	0.79	0.97	0.87	5352	0.89	0.97	0.93	5352
Sımf 1	0.97	0.93	0.95	5310	0.93	0.94	0.94	5310	0.95	0.92	0.94	5310	0.94	0.89	0.92	5310	0.98	0.94	0.96	5310	0.97	0.97	0.97	5310
Sımf 2	0.74	0.79	0.77	366	0.78	0.43	0.56	366	0.82	0.42	0.56	366	0.80	0.38	0.51	366	0.75	0.73	0.73	366	0.87	0.69	0.77	366
Sımf 3	0.93	0.96	0.94	4408	0.94	0.90	0.92	4408	0.94	0.90	0.92	4408	0.92	0.88	0.90	4408	0.97	0.78	0.86	4408	0.97	0.94	0.96	4408
Sımf 4	0.85	0.91	0.88	1467	0.91	0.68	0.77	1467	0.89	0.80	0.84	1467	0.89	0.78	0.83	1467	0.91	0.82	0.86	1467	0.93	0.88	0.91	1467
Sımf 5	0.87	0.72	0.79	428	0.86	0.50	0.64	428	0.91	0.26	0.41	428	0.86	0.23	0.36	428	0.66	0.65	0.65	428	0.96	0.50	0.66	428
Sımf 6	0.77	0.82	0.79	180	0.93	0.66	0.77	185	0.85	0.74	0.79	185	0.92	0.65	0.77	185	0.53	0.77	0.62	185	0.84	0.78	0.81	185
Doğruluk	0.95			25859	0.89			17516	0.89			17516	0.88			17516	0.88			17516	0.94			17516
Marko	0.87	0.87	0.87	25859	0.88	0.72	0.78	17516	0.88	0.72	0.76	17516	0.88	0.68	0.74	17516	0.80	0.81	0.80	17516	0.92	0.82	0.86	25863
Ortalama																								
Ağırlıklı Ortalama	0.94	0.94	0.94	25859	0.89	0.89	0.89	17516	0.90	0.89	0.89	25863	0.88	0.88	0.87	17516	0.90	0.88	0.88	17516	0.94	0.94	0.94	17516

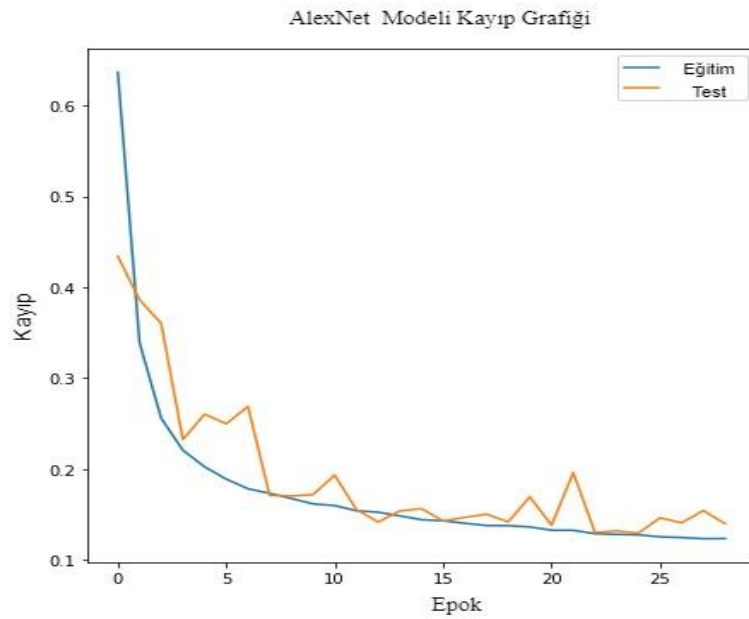
Çizelge 6.2. Farklı altı modelin kullanılan hiper parametrelere bağlı başarımları

Model	Eniyileme	Öğrenme Oranı	Epok	Batch boyutu	Eğitim Başarısı	Doğrulama Başarısı	AUC
AlexNet	Adam		30	128	%95	%94	%99
Inception-v3	Adam	0.001	30	64	%90	%89	%98
VGG19	Adam	0.001	30	64	%89	%97	%98
VGG16	Adam	0.001	30	128	%90	%88	%98
Resnet+Xception50	Adam	0.001	30	10	%93	%88	%96
SSCNN	Adam	0.001	30	128	%94	%93	%99

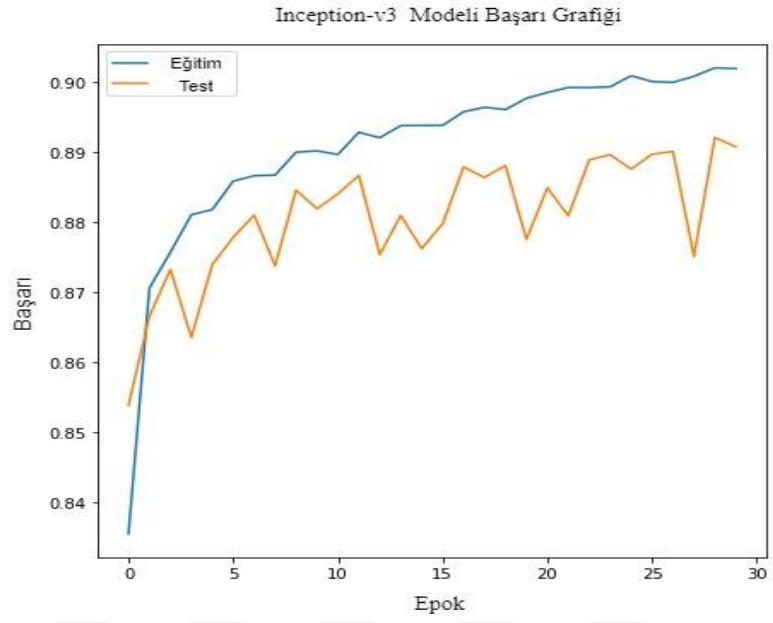
Çizelge 6.2’de farklı altı modelin eniyileme, öğrenme oranı, epok sayısı, eğitim başarısı, doğruluk başarısı ve Roc eğrisi değerleri verilir ve birbirleriyle karşılaştırılmıştır. En iyi eğitim başarısı ve doğrulama başarısını AlexNet modelin gösterilmiştir. Deneysel çalışmalarda kullanılan; AlexNet, Inception-v3, Xception+Resnet50, Vgg19, Vgg16 ve öz yapılı CNN (SSCNN) olmak üzere elde edilen sonuçların başarımları ve kayıp grafikleri Şekil 6.6 – 6.17’de verilmiştir.



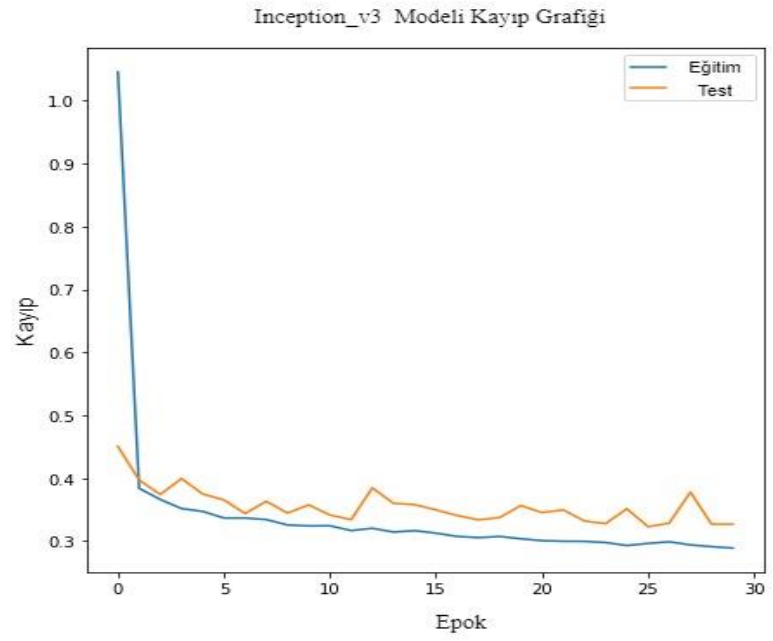
Şekil 6.6 AlexNet modeli başarı grafiği



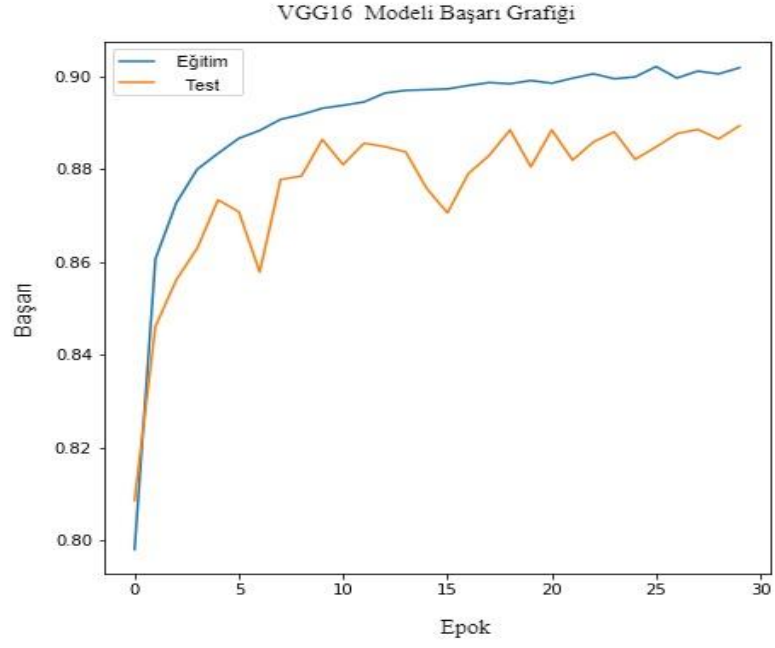
Şekil 6.7 AlexNet modeli kayıp grafiği



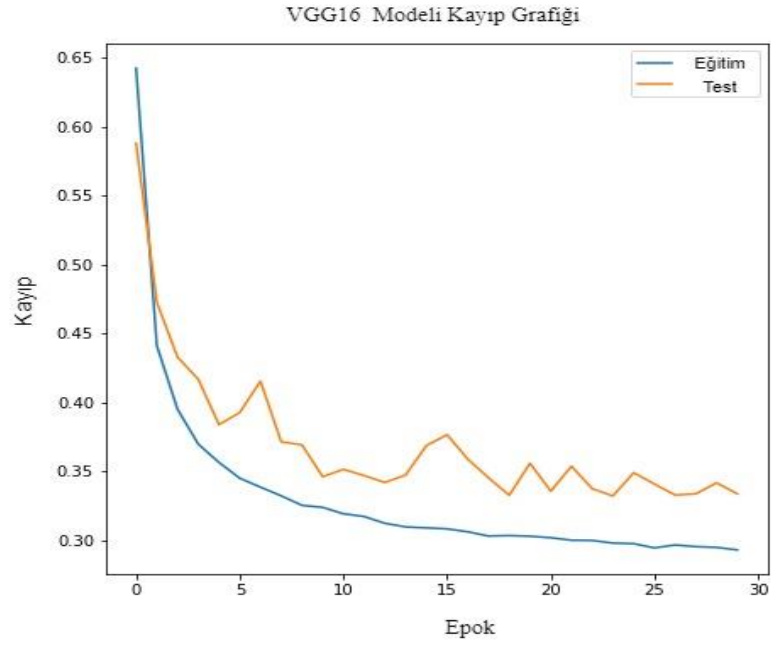
Şekil 6.8 Inception_v3 modeli başarı grafiği



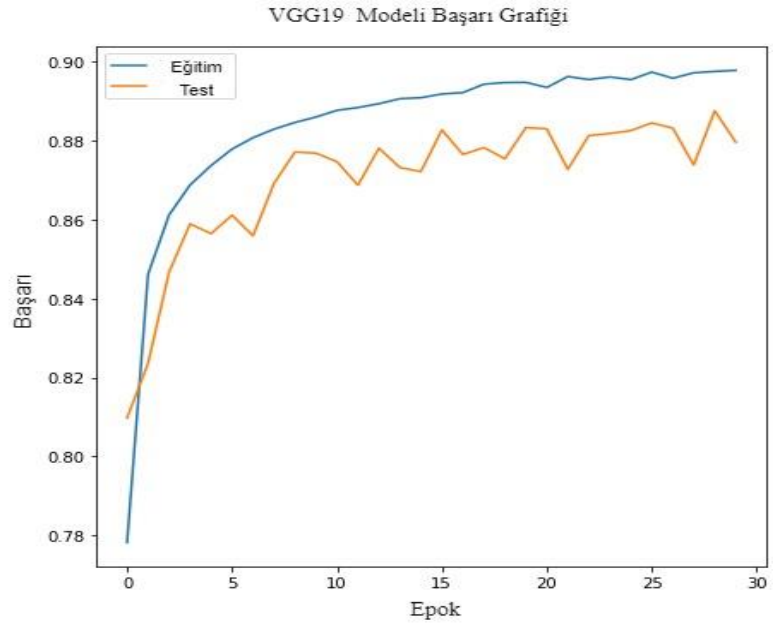
Şekil 6.9 Inception_v3 modeli kayıp grafiği



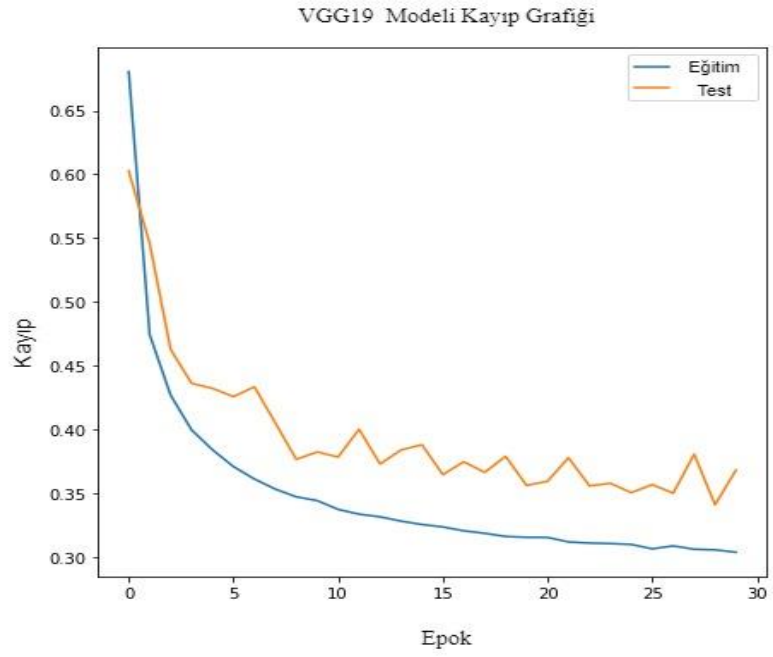
Şekil 6.10 VGG16 modeli başarı grafiği



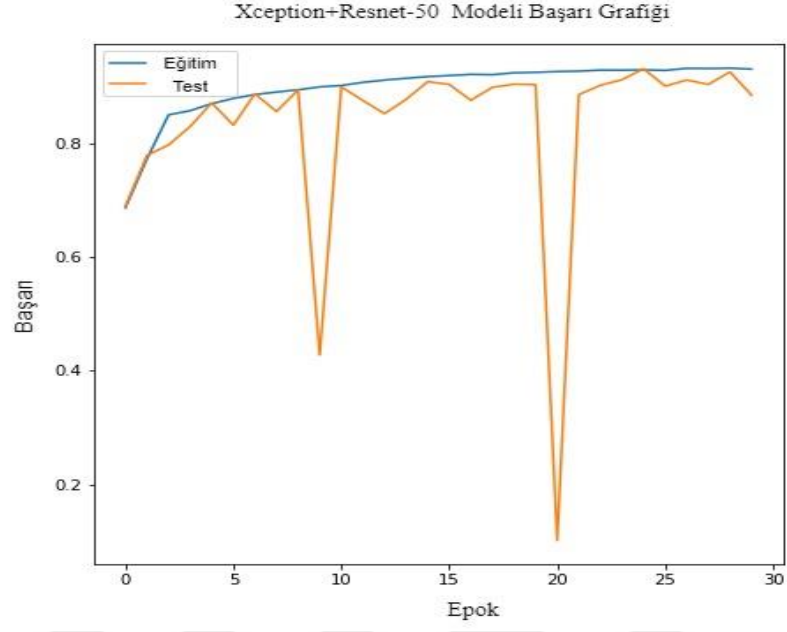
Şekil 6.11 VGG16 modeli kayıp grafiği



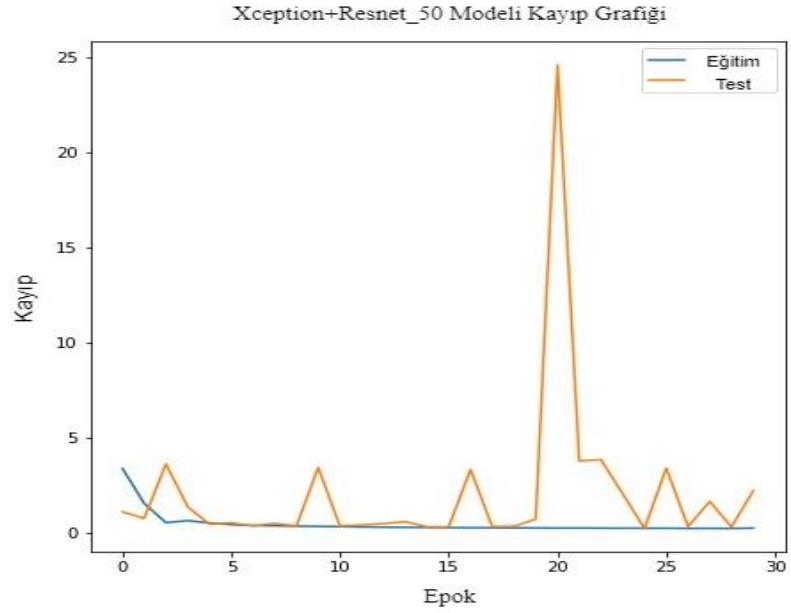
Şekil 6.12 VGG19 modeli başarı grafiği



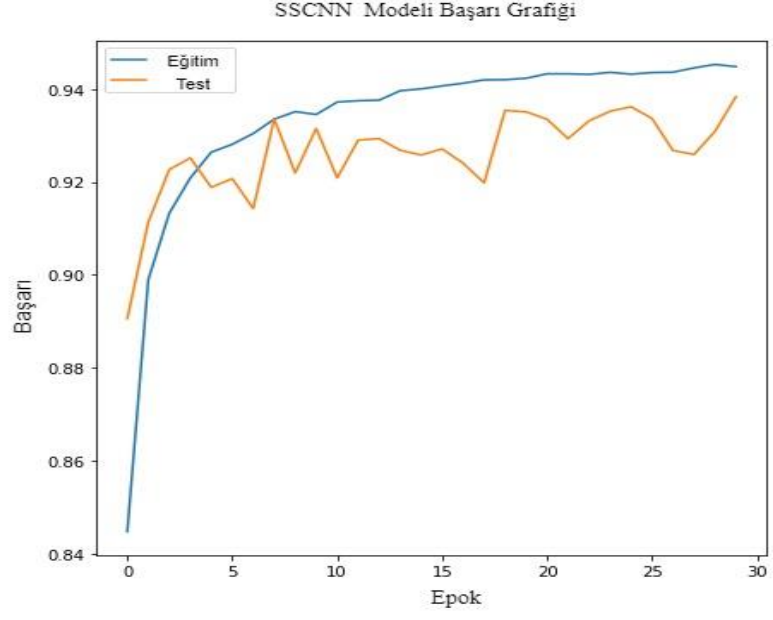
Şekil 6.13 VGG19 modeli kayıp grafiği



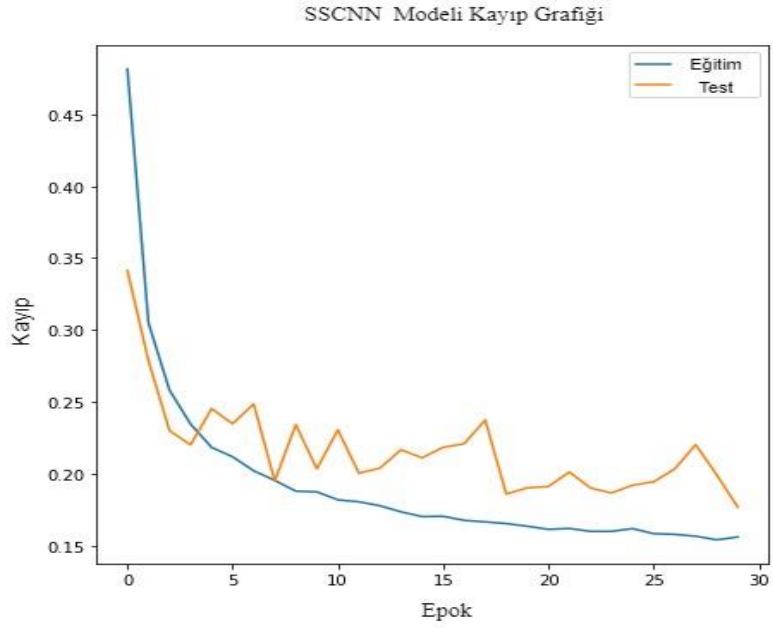
Şekil 6.14 Xception+Resnet-50 modeli başarı grafiği



Şekil 6.15 Xception+Resnet-50 modeli kayıp grafiği

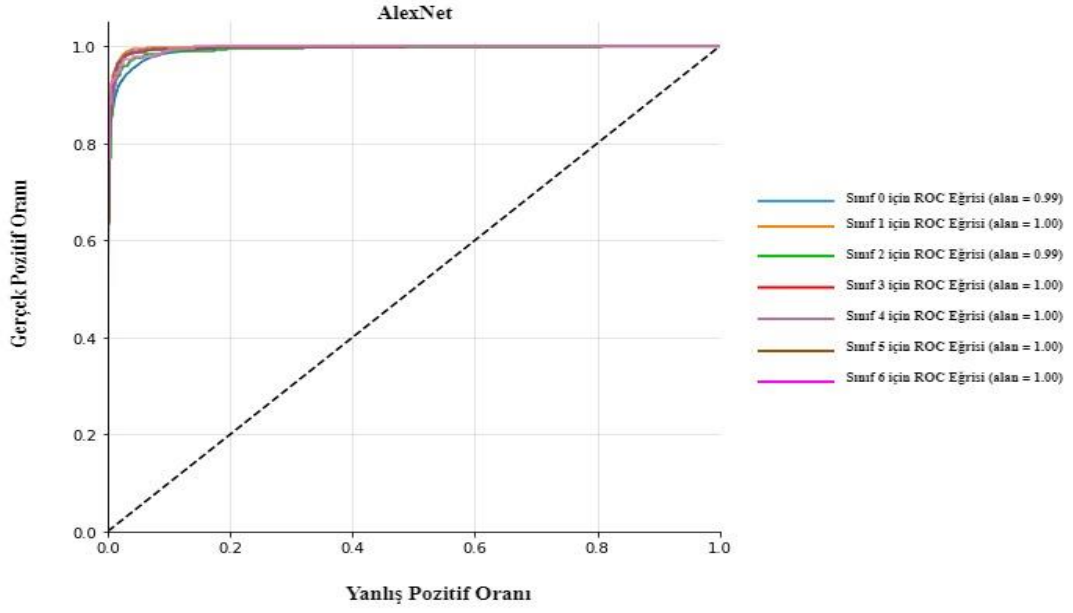


Şekil 6.16 SSCNN modeli başarı grafiği

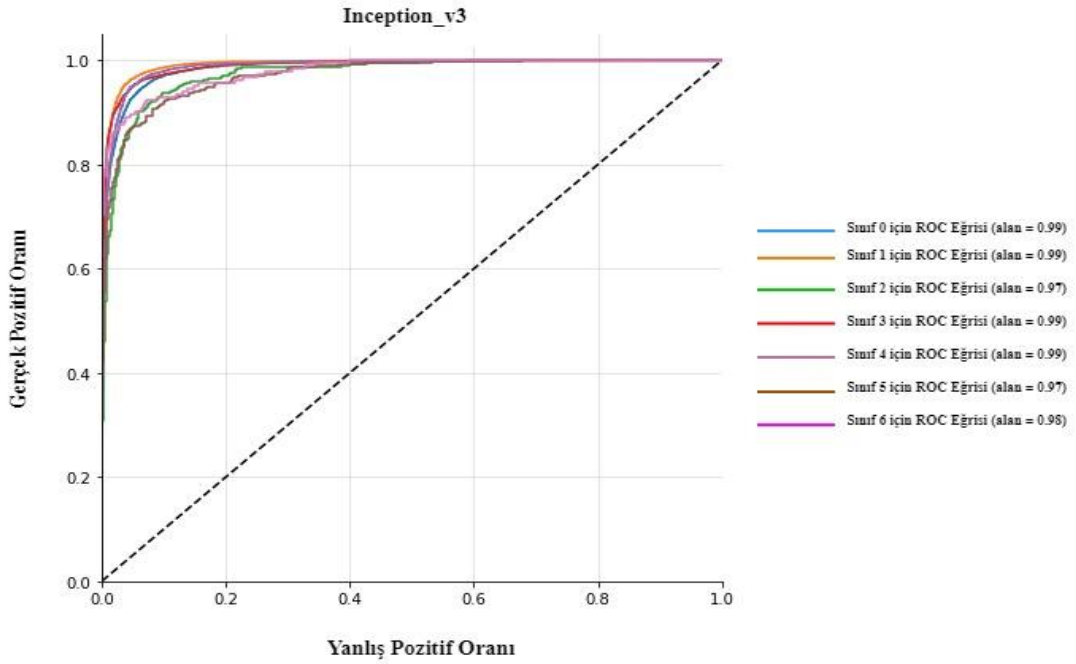


Şekil 6.17 SSCNN modeli kayıp grafiği

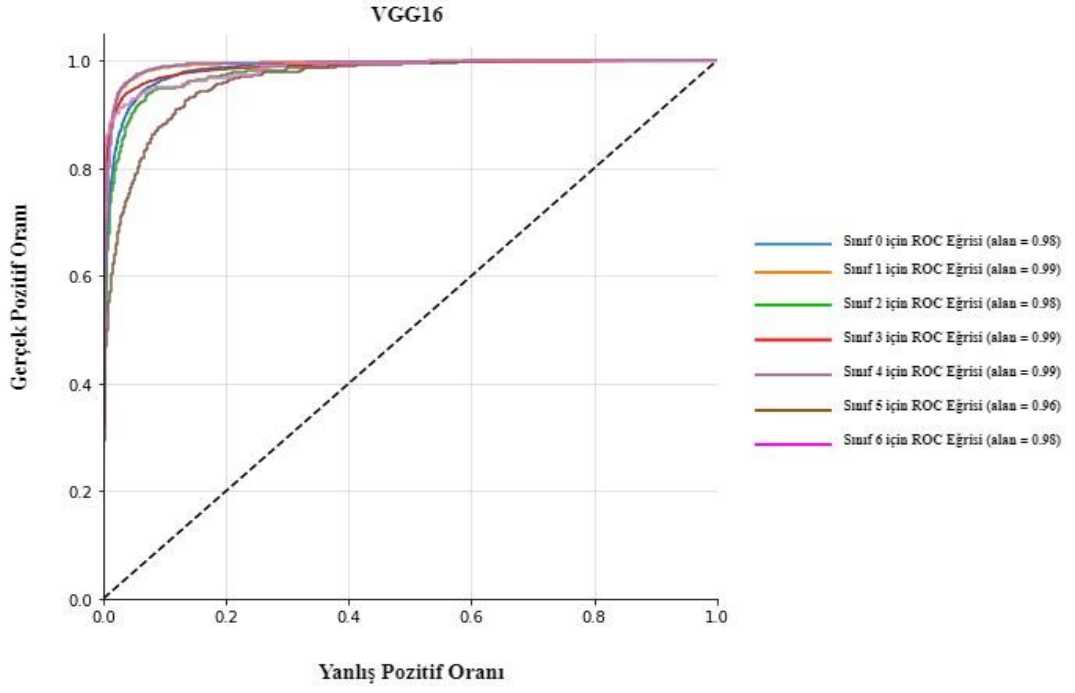
Makine öğrenmesi uygulamalarında en çok kullanılan başarı değerlendirme yöntemlerinden biri Roc eğrisidir. Grafikler modelin ne kadar başarılı olduğunu gösterir. Eğrinin altındaki alanı AUC olarak adlandırır. Bu alan ne kadar geniş ise modelin başarı oranının o kadar iyi olduğunu gösterir. Şekil 6.18-6.23’de verilmiştir.



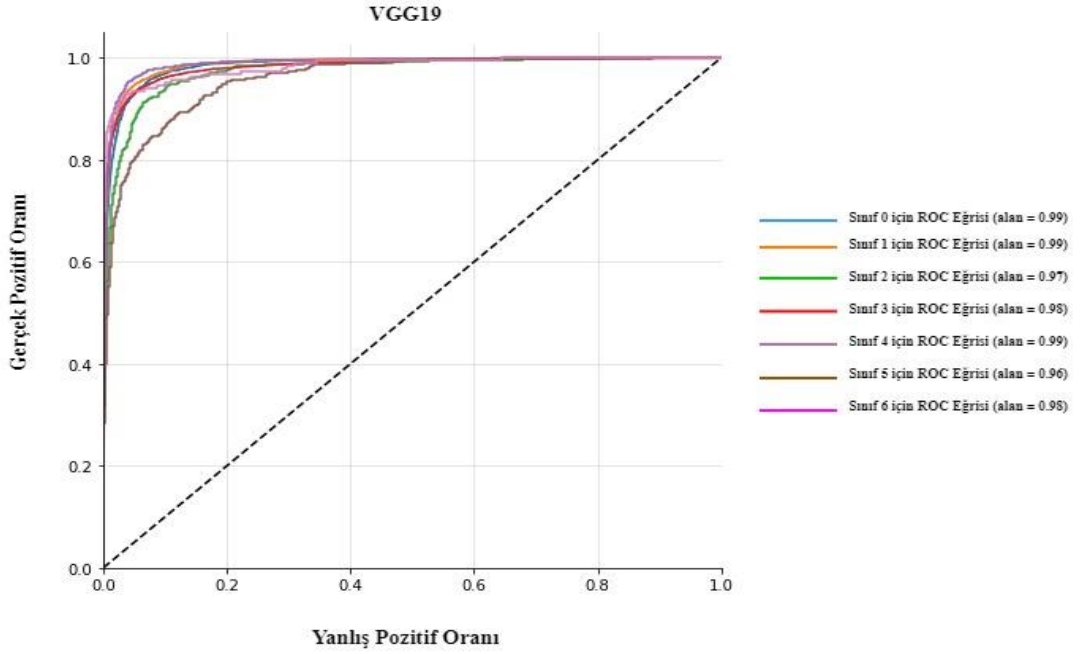
Şekil 6.18 Alexnet modeli AUC grafiği



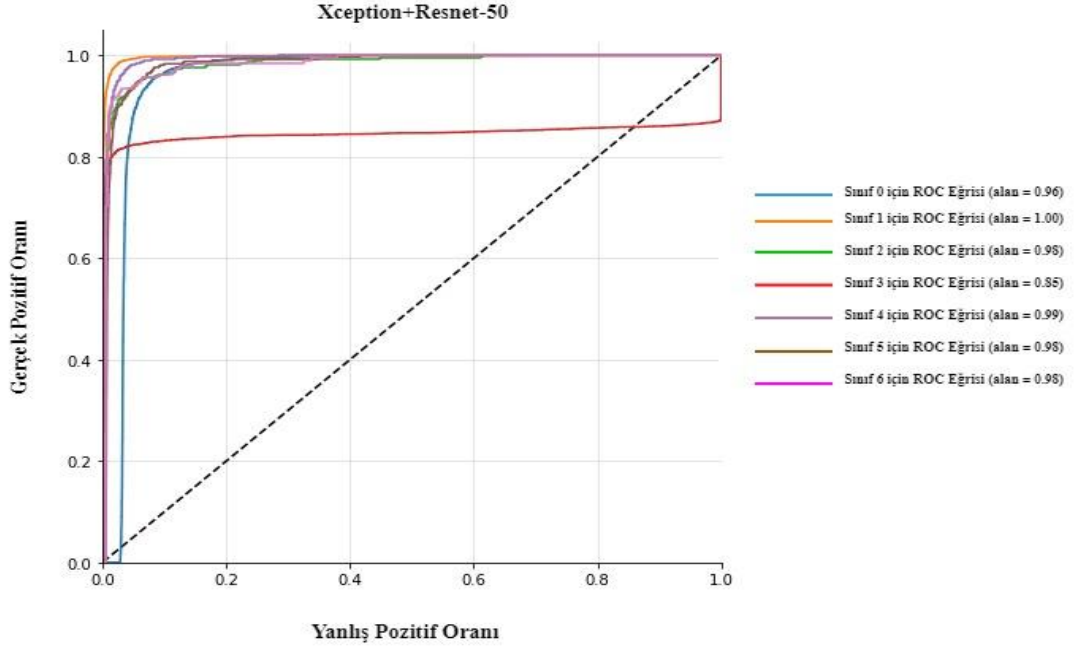
Şekil 6.19 Inception-v3 modeli AUC grafiği



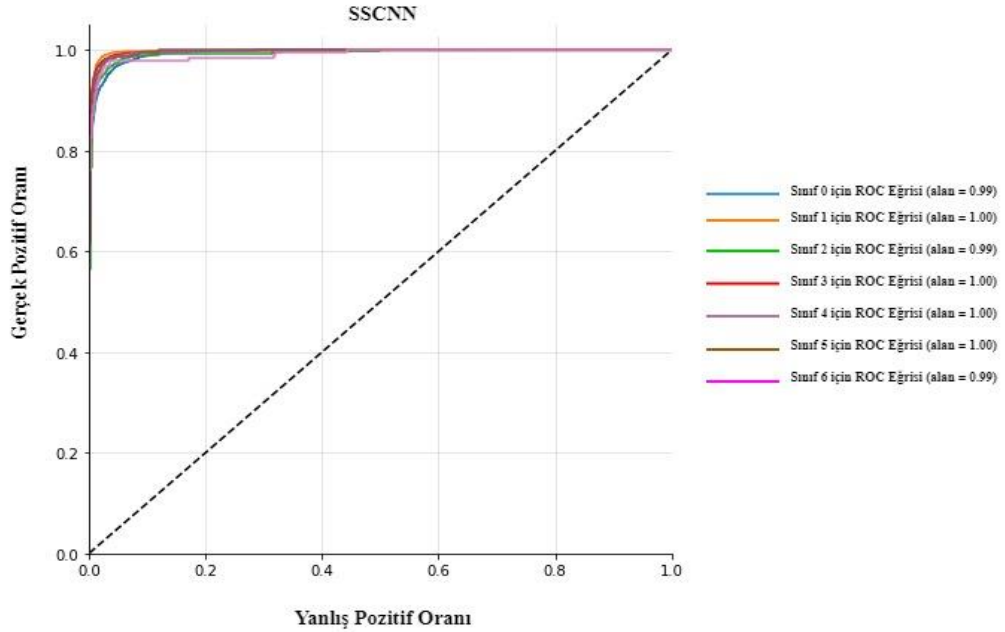
Şekil 6.20 VGG16 modeli AUC grafiği



Şekil 6.21 VGG19 modeli AUC grafiği



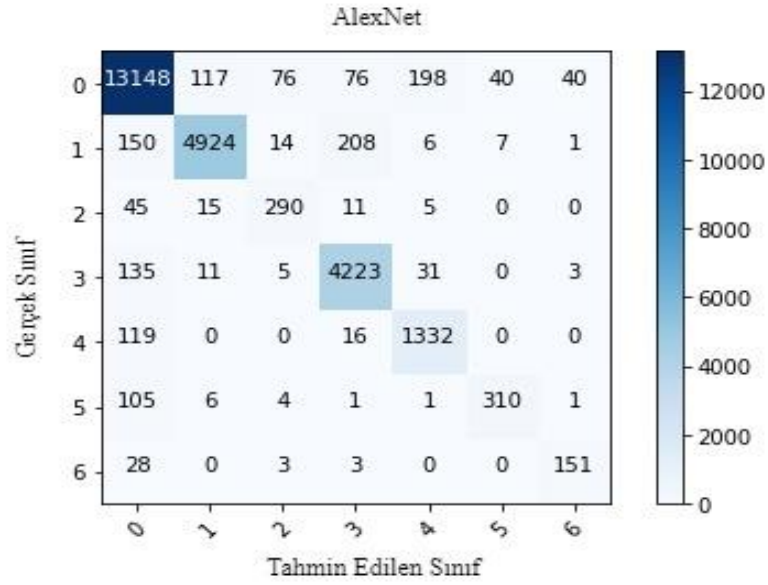
Şekil 6.22 Xception+Resnet-50 modeli AUC grafiği



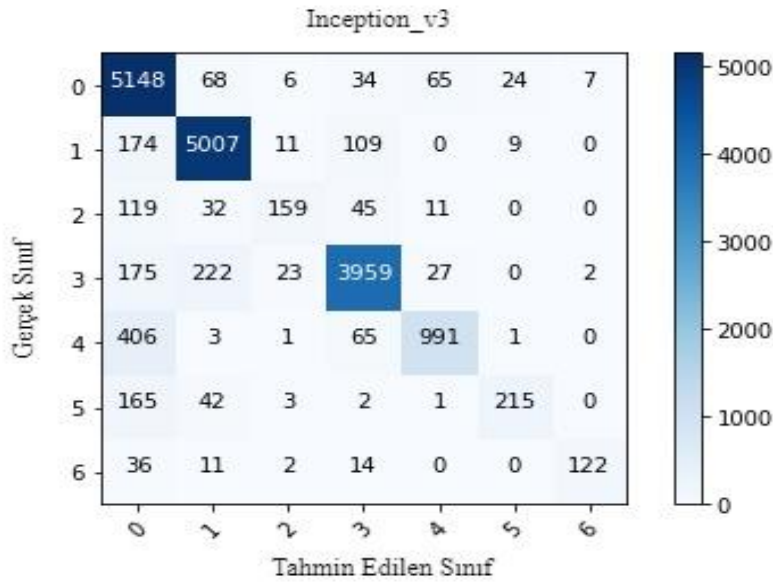
Şekil 6.23 SSCNN modeli AUC grafiği

Doğruluk oranı, modelin sınıflandırma yeteneğini değerlendirebilir ancak belirli ayrıntılar yansıtamaz. Karışıklık matrisi kullanılarak bir algoritmanın sınıflandırma başarısı görselleştirilerek sunulabilir. Karışıklık matrisi, sınıflandırma modeli tahminleri

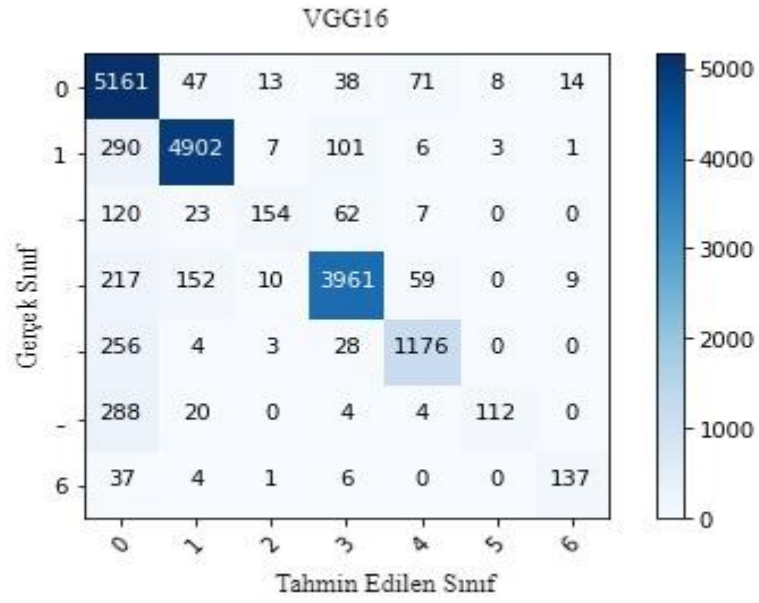
yaparken her kategorinin tahmin ayrıntılarını açıkça göstermektedir. Tahmin edilen sonuç ile gerçek değer karşılaştırılmaktadır. Altı modelin karışıklık matrisleri şekil 6.24-6.29'te gösterilmektedir.



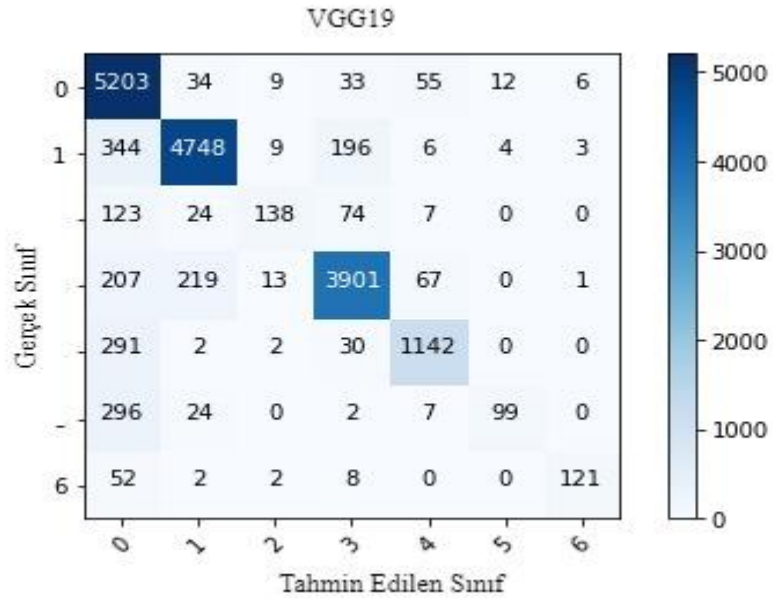
Şekil 6.24 AlexNet'in karışıklık matrisi



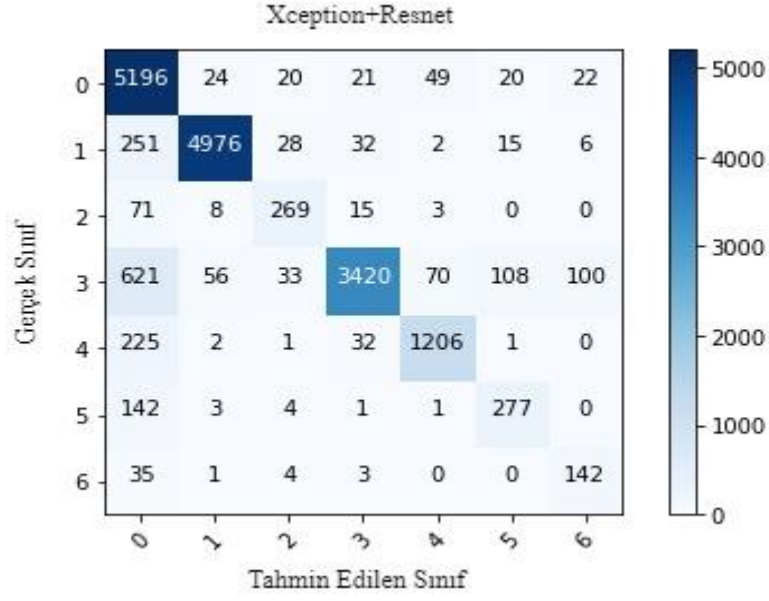
Şekil 6.25 Inception_v3'ün karışıklık matrisi



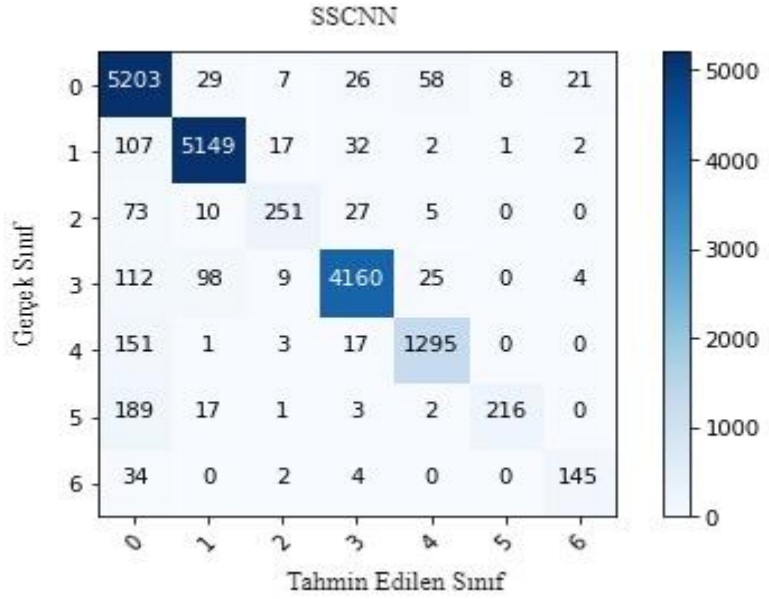
Şekil 6.26 VGG 16'nın karışıklık matrisi



Şekil 6.27 VGG 19'un karışıklık matrisi



Şekil 6.28 Exception+Resnet'in Karışıklık matrisi



Şekil 6.29 SSCNN'in karışıklık matrisi

Altı modelin karışıklık matrislerini göre sınıf sıfır, sınıf 1 ve sınıf 3 değer sınıfları göre daha iyi sonuç verilmiştir.

7. SONUÇ VE ÖNERİLER

Beslenme insanlar için en temel ihtiyaçlardan birisidir. Günümüzde nüfus artışı ile beslenme için gıda talebi de artmaktadır. Dolayısıyla gıda üretimindeki süreçler de değişmektedir. Artık, gıda üretiminde teknolojik gelişmelerden daha yoğun bir şekilde faydalanılmaktadır.

Bal, insanlar için uzun zamandır önemli bir besin kaynağı olarak kullanılmıştır. Bal üretimi de diğer gıda üretim süreçleri gibi değişmekte ve gelişmektedir. Balcılık faaliyetleri sonucundaki elde edilen ürünlerin verimi, üreticilerin bu alandaki bilgi birikimi, tecrübesi ve kullandıkları tekniklere bağlı olarak değişmektedir. Doğru tekniklerle ve bilimsel temele dayanan bilgilerle bal üretimindeki verim arttırılabilecektir. Bu bağlamda gerçekleştirilen tez çalışmasında, endüstriyel kamera ile dijital ortama aktarılmış, farklı yapısal özelliklere sahip olan bal peteklerine ait 38 adet fotoğraf kullanılmıştır. Bu fotoğraflar öncelikle hedef alan seçilecek şekilde bir temizleme işlemine tabi tutulmuştur. Elde edilen görüntüler üzerinde yedi farklı sınıf için etiketleme yapılmıştır ve böylece çalışmalar için gerekli olan veri seti oluşturulmuştur. Etiketlenen veri seti altı farklı derin öğrenme algoritması (AlexNet, VGG19, VGG16, Inception_v3, Xception+ResNet50v2, SSCNN) kullanılarak sınıflandırılmış ve belirlenen değerlendirme metrikleri ile (F1-skor, kesinlikli, duyarlılık ve Roc eğrisi) algoritmaların performansı ölçülmüştür. Gerçekleştirilen karşılaştırmalar sonucunda, AlexNet eğitim başarısı %95 ve doğrulama başarısı %94 ile diğer 5 algoritmaya göre en iyi sonucu vermiştir.

Tez çalışmasında oluşturulan veri setindeki bazı sınıflara ait örnek sayısı diğer sınıflara göre daha azdır. Örnek sayılarındaki farklılıktan dolayı dengesiz bir veri seti ile çalışılmıştır. Literatürde, dengesiz veri setlerinin dengeli hale getirilmesi ile daha yüksek başarı performansı elde edildiği görülmektedir. Araştırmacılar, mevcut dengesiz veri setini hem veri seviyesinde dengeleyerek hem de algoritmik seviyede iyileştirmeler yaparak sınıflandırma başarısını arttırabilir. Bunun yanında farklı görüntü işleme teknikleri veya gelişmiş sınıflandırma algoritmaları ile de sınıflandırma başarısının arttırılacağı düşünülmektedir.

KAYNAKLAR

- Ada S.E., 2019. Transfer learning for continuous control, Koç University, Council Of Higher Education Thesis Center.
- Adegbite J.O., Belhaj H., Bera A., 2021. Investigations on the relationship among the porosity, permeability and pore throat size of transition zone samples in carbonate reservoirs using multiple regression analysis, artificial neural network and adaptive neuro-fuzzy interface system. Petroleum Research.
- Alafi B., 2019. Artificial Intelligence and deep learning methodologies. The Journal of Cognitive Systems, 4, 2, 57-61.
- Anonim, (2020). Aracılık Raporu. Aracılık Raporu. www.ordutb.org.tr, Ordu Ticaret Borsası: 58.
- Makine Öğrenmesi, 2021. turhost.com. turhost.com, Erişim tarihi 2021/12/26. Erişim adresi, <https://www.turhost.com/blog/makine-ogrenmesi-machine-learning-nedir/#serp>.
- World Bee Day 2022. Food and Agriculture Organization of The United Nations. Erişim. Erişim adresi, <https://www.fao.org/world-bee-day/en/>.
- Building VGG19 with Keras, 2019. saicharanars.medium.com. Erişim. Erişim adresi, <https://saicharanars.medium.com/building-vgg19-with-keras-f516101c24cf>.
- Asadullayeva M., 2020. Küçük hücreli akciğer kanseri ile küçük hücreli dışı akciğer kanserinin ayırıcı tanısında kontrastli BT görüntülemeye radiomics ve makine öğrenmesinin katkısı, Kocaeli Üniversitesi, Yükseköğretim Kurulu Yayın ve Dokümantasyon Daire Başkanlığı Ulusal Tez Merkezi.
- Ayanzadeh A., 2020. Deep learning based segmentation pipeline for label-free phase-contrast microscopy images, Istanbul Technical University, Council Of Higher Education Thesis Center.
- Ba W., Wang R., Yin G., Song Z., Zou J., Zhong C., Yang J., Yu G., Yang H., Zhang L., Li C., 2021. Diagnostic assessment of deep learning for melanocytic lesions using whole-slide pathological images. Translational Oncology, 14, 9, 101161.
- Barman U., Choudhury R.D., Sahu D., Barman G.G., 2020. Comparison of convolution neural networks for smartphone image based real time classification of citrus leaf disease. Computers and Electronics in Agriculture, 177, 105661.
- Burcu V., (2018). Aracılık. Aracılık. T.C. gıda, tarım ve hayvancılık bakanlığı tarımsal ekonomi ve politika geliştirme enstitüsü
- T.C. gıda, tarım ve hayvancılık bakanlığı tarımsal ekonomi ve politika geliştirme enstitüsü.
- Çelik Ö., 2018. A research on machine learning methods and its applications. Journal of Educational Technology and Online Learning, 1, 3, 25-40.
- Çevik E., Zengin K., 2019. Classification of skin lesions in dermatoscopic images with deep convolution network. Avrupa Bilim ve Teknoloji Dergisi, 309-18.
- Chen H., Liu S., Magomedov R.M., Davidyants A.A., 2021. Optimization of inflow performance relationship curves for an oil reservoir by genetic algorithm coupled with artificial neural-intelligence networks. Energy Reports, 7, 3116-24.
- Cura A., 2019. Driver profiling Using Long Short Term Memory (LSTM) and Convolutional Neural Network(CNN), Marmara University, Council Of Higher Education Thesis Center.
- Doğan F., 2018. Derin Öğrenme Algoritmalarının Yaprak Sınıflandırma Başarımlarının Karşılaştırılması. Sakarya university journal of computer and information sciences, 21.
- Ergin F., 2020. Deep learning analysis in dermoscopy images, Galatasaray University, Council Of Higher Education Thesis Center.
- Gill H.S., Khehra B.S., 2021. An integrated approach using CNN-RNN-LSTM for classification of fruit images. Materials Today: Proceedings.
- Hariri M., 2021. Detection of calcium deficiency and physiological status in strawberry leaves using deep learning, Çukurova University, Council Of Higher Education Thesis Center.

- İş H., 2021. Sosyal Ağlarda Etkileşime Dayalı Davranış Analizi İle Karakter Tespiti, Fırat Üniversitesi, İnköğretim Kurulu Yayın ve Dokümantasyon Daire Başkanlığı Ulusal Tez Merkezi.
- Kohsaka R., Park M.S., Uchiyama Y., 2017. Beekeeping and honey production in Japan and South Korea: past and present. *Journal of Ethnic Foods*, 4, 2, 72-9.
- Langstroth L.L., 1878. *A Practical Treatise on the Hive and Honey-Bee*, Lippincott, Lippincott, p.
- Michal B., 2021. Breast mass classification with transfer learning based on scaling of deep representations. *Biomedical Signal Processing and Control*, 69, 102828.
- Mohammed S.H., 2021. Detection of cancer Area in lung images with the help of deep learning algorithm, Fırat University, Council Of Higher Education Thesis Center.
- Natakarnkitkul S. (2020). "Convolutional Neural Network from Scratch" [towardsai.net](https://towardsai.net/p/machine-learning/beginner-guides-to-convolutional-neural-network-from-scratch-kuzushiji-mnist-75f42c175b21) <https://towardsai.net/p/machine-learning/beginner-guides-to-convolutional-neural-network-from-scratch-kuzushiji-mnist-75f42c175b21>.
- Özgenel Ç.F., 2018. Crack detection with deep learning: an exemplary study of data design in architecture, Middle east technical university, Council Of Higher Education Thesis Center.
- Pasyar P., Mahmoudi T., Kouzehkanan S.-Z.M., Ahmadian A., Arabalibeik H., Soltanian N., Radmard A.R., 2021. Hybrid classification of diffuse liver diseases in ultrasound images using deep convolutional neural networks. *Informatics in Medicine Unlocked*, 22, 100496.
- Radečić D. (2020). "Softmax Activation Function Explained." *Towards Data Science* 2020.
- Rahimzadeh M., Attar A., 2020. A modified deep convolutional neural network for detecting COVID-19 and pneumonia from chest X-ray images based on the concatenation of Xception and ResNet50V2. *Informatics in Medicine Unlocked*, 19, 100360.
- Ramsundar B., Zadeh R.B. (2022). "TensorFlow For Deep Learning." *O'Reilly* <https://www.oreilly.com/library/view/tensorflow-for-deep/9781491980446/ch04.html>.
- ResNet50, 2022. [blog.devgenius.io](https://blog.devgenius.io/resnet50-6b42934db431). Erişim adresi, <https://blog.devgenius.io/resnet50-6b42934db431>.
- Saraloğlu E., 2020. Derin öğrenme için kitle kaynak tabanlı etiketli veri üretimi ve çok bantlı görüntü sınıflandırma için bir 3d-2d cnn modeli, Karadeniz Teknik Üniversitesi Yükseköğretim Kurulu Yayın ve Dokümantasyon Daire Başkanlığı Ulusal Tez Merkezi.
- Sarigül M., 2019. A new deep learning approach: differential convolutional neural network, Çukurova University, Council Of Higher Education Thesis Center.
- Sevli O., Kemalolu N., 2020. Turkish sign language digits classification with CNN using different optimizers. *International Advanced Researches and Engineering Journal*, 4, 3, 200-7.
- Shaheen M., Khan R., Biswal R., Ullah M., Khan A., Uddin M.I., Zareei M., Waheed A., 2021. Acute Myeloid Leukemia (AML) Detection Using AlexNet Model. *Complexity*, 2021.
- Skogly A., 2019. Enhancing Performance in CNN Architectures for Image Classification, Bahçeşehir university, Council Of Higher Education Thesis Center.
- Sugata T., Yang C., 2017. Leaf App: Leaf recognition with deep convolutional neural networks. *IOP Conference Series: Materials Science and Engineering*, 273, 012004.
- Szegedy C., Vanhoucke V., Ioffe S., Shlens J., Wojna Z. Rethinking the inception architecture for computer vision. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2818-26.
- T S., Khilar R., Subaja Christo M., 2021. A comparative analysis on plant pathology classification using deep learning architecture – Resnet and VGG19. *Materials Today: Proceedings*.

- Tetila E.C., Machado B.B., Astolfi G., Belete N.A.d.S., Amorim W.P., Roel A.R., Pistori H., 2020. Detection and classification of soybean pests using deep learning with UAV images. *Computers and Electronics in Agriculture*, 179, 105836.
- Top A.E., 2018. Classification of eeg signals using transfer learning on convolutional neural networks via spectrogram, Ankara Yıldırım Beyazıt University, Council Of Higher Education Thesis Center.
- Usta A., Yıldırım Y., 2020. Bal Arılarının Viral Hastalıkları Çukurova Tarım ve Gıda Bilimleri Dergisi.
- vanengelsdorp D., Tarpy D.R., Lengerich E.J., Pettis J.S., 2013. Idiopathic brood disease syndrome and queen events as precursors of colony mortality in migratory beekeeping operations in the eastern United States. *Preventive Veterinary Medicine*, 108, 2, 225-33.
- Yani M., Irawan S., Setianingsih C., 2019. Application of Transfer Learning Using Convolutional Neural Network Method for Early Detection of Terry's Nail. *Journal of Physics: Conference Series*, 1201, 012052.
- Dünyayı Değiştirmekte Olan Yapay Sinir Ağları Nedir?, 2015. BilimFili Medya Grup. Erişim. Erişim adresi, <https://bilimfili.com/dunyayi-degistirmekte-olan-yapay-sinir-aglari-nedir>.