



T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ



**YENİ DOĞAN BEBEKLERİN TERMAL
GÖRÜNTÜLERİNİN DERİN ÖĞRENME
YÖNTEMLERİ İLE SINIFLANDIRILMASI**

Ahmet Haydar ÖRNEK

YÜKSEK LİSANS TEZİ

Elektrik - Elektronik Mühendisliği Anabilim Dalı

Temmuz - 2019
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Ahmet Haydar ÖRNEK tarafından hazırlanan “Yeni Doğan Bebeklerin Termal Görüntülerinin Derin Öğrenme Yöntemleri ile Sınıflandırılması” adlı tez çalışması 08.07.19 tarihinde aşağıdaki jüri tarafından oy birliği / oy çokluğu ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Elektrik - Elektronik Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

Başkan

Prof. Dr. Seral ÖZŞEN

Danışman

Doç. Dr. Murat CEYLAN

Üye

Dr. Öğr. Üyesi Hüseyin Oktay ALTUN

İmza


.....


.....


.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Hakan KARABÖRK
Enstitü Müdürü

Bu tez çalışması TÜBİTAK tarafından 215E019 numaralı proje ile desteklenmiştir.

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.



Ahmet Haydar ÖRNEK

Ahmet Haydar ÖRNEK

Tarih: 08.07.19

ÖZET

YÜKSEK LİSANS TEZİ

YENİ DOĞAN BEBEKLERİN TERMAL GÖRÜNTÜLERİNİN DERİN ÖĞRENME YÖNTEMLERİ İLE SINIFLANDIRILMASI

Ahmet Haydar ÖRNEK

**Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Elektrik - Elektronik Mühendisliği Anabilim Dalı**

Danışman: Doç. Dr. Murat CEYLAN

2019, 101 Sayfa

**Jüri
Prof. Dr. Seral ÖZŞEN
Doç. Dr. Murat CEYLAN
Dr. Öğr. Üyesi Hüseyin Oktay ALTUN**

Yeni doğan bebeklerin hastalıklarının erken teşhis edilmesi ve tedavi süreçlerinin izlenmesi sağlık durumlarının korunması bakımından büyük önem taşımaktadır. Termal görüntüleme iyonize olmayan, zararlı radyasyon içermeyen ve temassız bir görüntüleme tekniği olarak yeni doğan bebeklerin görüntülenmesinde on yıllardır kullanılmaktadır. Vücudun termal simetri dengesi temel alınarak gerçekleştirilen bu çalışmalar incelendiğinde klasik görüntü işleme tekniklerinin uygulandığı görülmektedir. Bu teknikler görüntü ön işleme (yeniden boyutlandırma, gri seviyeye dönüşüm), görüntü iyileştirme (gürültü giderme, histogram eşitleme), manuel ilgi alanı (İA) seçimi ve İA'da meydana gelen değişimlerin izlenmesidir. Probleme dayalı olarak geliştirilen bir algoritma bu teknikler kullanılarak başarılı sonuçlar elde edebilir fakat görüntüleme esnasında meydana gelebilecek her türlü durumun algoritmada tanımlanması gerekmektedir. Örneğin bebekler kuvöz içerisinde hareket ettiğinden dolayı oluşturulan İA'nın bu hareketlere bağlı olarak şekil değiştirmesi, yeniden eşleşmesi ve izlemeye devam etmesi gerekmektedir. Bütün olasılıklar hesaba katıldığında işlem yükünün artacağı buna bağlı olarak da gerçek zamanlı uygulamalardan uzaklaşılacağı görülmektedir. Çok katmanlı algılayıcılar, evrişimli sinir ağları ve çekişmeli üretici ağlar gibi derin öğrenme yöntemlerinin gelişmesi ile birlikte görüntü sınıflandırma ve görüntü üretme gibi işlemler görüntüden öğrenme felsefesi esas alınarak gerçekleştirilmektedir. Böylece her duruma karşılık gelen bir kod dizini oluşturmak yerine her durumu ifade eden görüntüler bulunması yeterli olmaktadır. Bu tez kapsamında Selçuk Üniversitesi, Tıp Fakültesi, Yeni Doğan Yoğun Bakım Ünitesinde tedavi gören 40 bebeğe ait termal görüntüler hem makine öğrenimi ve derin öğrenme yöntemleri kullanılarak sınıflandırılmış hem de termal görüntülerin yeniden üretilmesi gerçekleştirilmiştir. Burada derin öğrenme yöntemlerinin eğitilmesi için binlerce görüntüye ihtiyaç duyulmaktadır ancak medikal alanda böyle bir görüntü setinin oluşturulması çok uzun süreler alacağından görüntülerin artırılması için veri artırımı yöntemleri kullanılmıştır. Sonuçlar 10 – katlamalı çapraz doğrulama tekniğine göre elde edilmiş ve çeşitli değerlendirme kriterleri (karmaşıklık matrisi, özgüllük, doğruluk, hassasiyet, alıcı işlem karakteristiği, eğri altında kalan alan, yapısal benzerlik indeksi ve tepe sinyal gürültü oranı) ile değerlendirilmiştir. Çalışmalar arasında en iyi sınıflama sonucu

evriřimli sinir ađları ve veri artırımı kullanılarak % 99.85 hassasiyet, % 99.82 özgüllük ve % 99.84 doğruluk deđeri ile elde edilmiřtir. Bu sonuçlar derin öğrenme yöntemlerinin termal görüntülerin sınıflandırılmasında oldukça başarılı olduğunu göstermektedir.

Anahtar Kelimeler: Çekiřmeli Üretici Ağlar, Derin Öğrenme, Evriřimli Sinir Ağları, Hastalık Tespiti, Makine Öğrenimi, Sınıflandırma, Termografi, Transfer Öğrenme, Veri Artırma, Yeni Dođan



ABSTRACT

MS THESIS

CLASSIFICATION OF THERMAL IMAGES OF NEONATES WITH DEEP LEARNING METHODS

Ahmet Haydar ÖRNEK

**Konya Technical University
Institute of Graduate Studies
Department of Electrical and Electronics Engineering**

Advisor: Assoc. Prof. Dr. Murat CEYLAN

2019, 101 Pages

**Jury
Prof. Dr. Seral ÖZŞEN
Assoc. Prof. Dr. Murat CEYLAN
Asst. Prof. Dr. Hüseyin Oktay ALTUN**

Early detection of diseases and monitoring of treatment process of neonates are critically important in terms of maintaining their health status. Thermal imaging, which is a non – ionized, no harmful radiation and non – contact method, has been used in medical applications for decades. When conventional studies based on thermal symmetry balance of the body are examined, it is seen that classical image processing techniques are applied. These techniques include image preprocessing (resizing, gray level transform), image enhancement (noise reduction, histogram equalization), manual region of interest (ROI) selection and monitoring of changes in ROI. A problem-based algorithm can achieve successful results using these techniques, but any situation that may occur during imaging should be defined in the algorithm. For example, because the infants move inside the incubator, the ROI must change shape, re-match and continue to monitor. When all the possibilities are taken into account, it will be seen that the cost of process will increase and therefore the real time applications will be removed. With the development of deep learning methods such as multi-layer perceptions, convolutional neural nets and generative adversarial nets, processes such as image classification and image generation are carried out on the basis of philosophy of learning from image. Thus, instead of creating a code index corresponding to each situation, it is sufficient to have images representing each situation. Within the scope of this thesis, thermal images were obtained belonging to 40 neonates who were treated in Selcuk University, Faculty of Medicine, Neonatal Intensive Care Unit and thermal images both were classified by using machine learning and deep learning methods and the thermal images were regenerated. Thousands of images are needed to train deep learning methods, since such a set of images would take a very long time in the medical field, data augmentation methods have been used to augment images. The results were obtained according to the 10 – fold cross validation technique and evaluated with various evaluation criteria (confusion matrix, specificity, accuracy, sensitivity, receiver operating characteristic, area under curve, structural similarity index and peak signal noise ratio). The best classification results were obtained by using convolutional neural nets and data augmentation with 99.85 % sensitivity, 99.82 % specificity and 99.84 % accuracy. These results show that deep learning methods are quite successful in the classification of thermal images.

Keywords: Classification, Convolutional Neural Nets, Data Augmentation, Deep Learning, Disease Detection, Generative Adversarial Nets, Machine Learning, Neonate, Thermography, Transfer Learning.



ÖNSÖZ

Yüksek lisans tez çalışmalarım boyunca değerli katkılarıyla beni yönlendiren danışmanım Konya Teknik Üniversitesi Mühendislik ve Doğa Bilimleri Fakültesi Elektrik-Elektronik Mühendisliği Öğretim Üyesi Sayın Doç. Dr. Murat Ceylan'a, bölümümüz öğretim elemanlarına, her türlü maddi ve manevi desteğini esirgemeyen anneme, babama, kardeşlerime ve öğrenci arkadaşlarıma teşekkür ederim. Ayrıca yeni doğan bebeklere ait görüntülerin kaydedilmesi aşamalarında destek veren Prof. Dr. Hanifi Soylu'ya ve Selçuk Üniversitesi Tıp Fakültesi personeline teşekkür ederim.

Ahmet Haydar ÖRNEK
KONYA-2019

İÇİNDEKİLER

ÖZET	iv
ABSTRACT	vi
ÖNSÖZ	viii
İÇİNDEKİLER	ix
SİMGELER VE KISALTMALAR	xi
1 GİRİŞ	1
2 KAYNAK ARAŞTIRMASI	3
3 MATERYAL VE YÖNTEM	6
3.1 Yeni Doğan Bebeklere ait Termogramların Elde Edilmesi	6
3.2 Termogram Yaklaşımları	9
3.2.1 Sıcaklık haritaları	9
3.2.2 RGB görüntüler.....	9
3.2.3 Sıcaklık haritaları ile RGB görüntülerin karşılaştırılması	10
3.3 Termal Görüntülerin Bölütlenmesi	12
3.4 Termal Görüntülerden Özellik Çıkarma	15
3.4.1 Yerel ikili desen (YİD)	16
3.4.2 Ayrık dalgacık dönüşümü (ADD).....	17
3.5 Hızlı Korelasyon Tabanlı Filtre (HKTF) ile Özellik Seçme	18
3.6 Makine Öğrenimi	19
3.6.1 Yapay sinir ağları (YSA)	20
3.6.2 Lojistik regresyon (LR)	21
3.6.3 Karar ağaçları (KA) ve rastgele orman (RO).....	22
3.6.4 Destek vektör makineleri (DVM)	23
3.7 Derin Öğrenme	24
3.7.1 Evrişimli sinir ağları (ESA)	25
3.7.2 Transfer öğrenme	34
3.7.3 Çekişmeli üretici ağlar (ÇÜA)	36
3.8 Termal Görüntüler için Veri Artırımı	37
3.9 Sınıflandırma Sonuçlarının Değerlendirilmesinde Kullanılan Kriterler	38
3.9.1 Tepe sinyal gürültü oranı (TSGO)	38
3.9.2 Yapısal benzerlik indeksi (YBİ)	39
3.9.3 Karmaşıklık matrisi, doğruluk, özgüllük, hassasiyet, alıcı işlem karakteristiği (AİK) ve eğri altında kalan alan (EAA)	39
3.10 Sınıflayıcıların Genelleştirme Kabiliyetlerinin Belirlenmesi	41
3.10.1 K-katlamalı çapraz doğrulama	41
4 ARAŞTIRMA SONUÇLARI VE TARTIŞMA	43
4.1 Termogramların Değerlendirilmesinde Doğru Yaklaşımların Belirlenmesi....	43

4.2	Sıcaklık Haritaları ve Makine Öğrenimi Kullanılarak Sağlıklı ve Hasta Bebeklerin Tespiti	46
4.3	Evrişimli Sinir Ağları Kullanılarak Yeni Doğan Bebeklerin Sağlık Durumlarının Tespit Edilmesi	52
4.3.1	Geleneksel veri artırma yöntemlerinin sınıflama üzerine etkilerinin karşılaştırılması	57
4.4	Transfer Öğrenme'nin Medikal Termografik Görüntülerin Sınıflandırılması Üzerindeki Etkileri	62
4.5	Çekişmeli Üretici Ağlar ile Yeni Termal Görüntülerin Üretilmesi	66
5	SONUÇLAR VE ÖNERİLER	72
5.1	Sonuçlar	72
5.2	Öneriler	74
	KAYNAKLAR	75
	EKLER	79
	ÖZGEÇMİŞ	100

SİMGELER VE KISALTMALAR

Simgeler

f	:f Görüntüsü
$\frac{df}{dx}$:f Görüntüsünün Türevi
$f(x, y)$:f Görüntüsünün x Satır ve y Sütündeki Piksel Değeri
$g(x, y)$:f (x, y)'nin İkili Hale Geldikten Sonraki Değeri
$Y(x, y)$:RGB Görüntünün Parlaklık Değeri
R	:Renkli Görüntünün Kırmızı Bandı
G	:Renkli Görüntünün Yeşil Bandı
B	:Renkli Görüntünün Mavi Bandı
$SB(A, C)$:A ile C arasındaki Simetrik Belirsizlik
$BK(A, C)$:A ile C arasındaki Bilgi Kazancı
$H(A)$:A'nın Marjinal Entropi Değeri
$H(A C)$:A ile C arasındaki Durumsal Entropi Değeri
$P(A_i)$:A _i 'nin Olasılık Değeri
$P(A_i, C_i)$:A = A _i ve C = C _i iken Olasılık Değeri
X	:Doğrusal Regresyon Giriş Değeri
β_0	:X = 0'iken Bulunan Değer
β_1	:Regresyon Katsayısı
ε	:Hata
$do(X, Y)$:Doğrusal Kernel Fonksiyonu
$po(X, Y)$:Polinomik Kernel Fonksiyonu
$rad(X, Y)$:Radyal Kernel Fonksiyonu
$X \bullet Y$:X ile Y Arasındaki Noktasal Çarpım
γ	:Kernel Fonksiyonunun Ayarlanabilir Parametresi
C	:Polinom Kaydırma Parametresi
d	:Polinom Derecesi
$\ X - Y\ ^2$:X ile Y arasındaki Öklit Mesafesi
h	:Konvolüsyon Kerneli

$*$:Konvolüsyon İşlemi
n_{ζ}	:Yeni Oluşan Özellik Haritası Boyutu
$n_{giriş}$:Giriş Görüntüsü Boyutu
e_k	:Görüntüye Eklenen Sıfır Ekleme Sayısı
f_i	:h Filtresinin Boyutu
k	:h Filtresinin Kaydırma Parametresi
s	:Sigmoid Aktivasyon Fonksiyonu
t	:Tanjant Hiperbolik Aktivasyon Fonksiyonu
r	:ReLU Aktivasyon Fonksiyonu
lr	:Sızıntı ReLU Aktivasyon Fonksiyonu
Ha_{toplam}	:Toplam Hata
ω	:Ağırlık Değeri
η	:Öğrenme Oranı
$Imax$:Görüntüde Bulunan En Yüksek Piksel Değeri
$pa(x, y)$:İki Görüntü Arasındaki Parlaklık
$ko(x, y)$:İki Görüntü Arasındaki Kontrast
$ya(x, y)$:İki Görüntü Arasındaki Yapısal Özellik
μ	:Ortalama Değer
σ	:Standart Sapma
σ_{xy}	:İki Görüntü Arasındaki Kovaryans
$C1$:Imax'ın 0.01 Katı Değeri
$C2$:Imax'ın 0.03 Katı Değeri
$C3$:C2'nin 0.5 Katı Değeri

Kısaltmalar

ADD	:Ayrık Dalgacık Dönüşümü
AIK	:Alıcı İşlem Karakteristiği
AORTC	:AORT Koarktasyon hastalığı
BK	:Bilgi Kazancı
°C	:Santigrat
DAB	:Diyabetik Anne Bebeği hastalığı
DH	:Diyafraam Fıtığı Hastalığı
DN	:Doğru Negatif
DP	:Doğru Pozitif
DVM	:Destek Vektör Makineleri
EAA	:Eğri Altında Kalan Alan
ESA	:Evrışimli Sinir Ağları
HKTF	:Hızlı Korelasyon Tabanlı Filtre
HLH	:Hipoplastik Sol Kalp Hastalığı
IKK	:İntrakranial Kanama Hastalığı
IO	:Bağırsak Tıkanıklığı Hastalığı
IUGR	:Intrauterin Büyüme Kısıtlaması Hastalığı
İA	:İlgi Alanı
K	:Kelvin
KA	:Karar Ağaçları
LR	:Lojistik Regresyon
M.Ö	:Milattan önce
MAS	:Meconium Aspirasyon Sendromu aHastalığı
NEC	:Nekrotizan Enterokolit Hastalığı
OA	:Ozofagus Atrezi Hastalığı
PAB	:Pulmoner Arter Bantlaması Hastalığı
PDA	:Patent Ductus Arteriosus Hastalığı
RDS	:Solunum Güçlüğü Sendromu Hastalığı
RO	:Rastgele Orman
SB	:Simetrik Belirsizlik
TTN	:Geçici Taşiton Hastalığı
YİD	:Yapısal Benzerlik İndeksi

YN :Yanlıř Negatif
YP :Yanlıř Pozitif
YSA :Yapay Sinir Ađı



1 GİRİŞ

Görüntüleme teknolojilerinin geliştirilmesi ile birlikte, gözün göremediği dalga boylarındaki (0-380 nm ve 780 – 25000 nm) ışınımalar görüntüleme araçları vasıtasıyla görülebilir hale gelmiştir. Bir cismin 780 – 25000 nm dalga boyları arasında (kızılötesi) ışınım yapabilmesi için sıcaklık değerinin mutlak sıfır noktası (0 K, -273 °C) üzerinde olması gerekmektedir (Govindarajan, 2007). Termal kameralar bu ışınımaları yakalayıp elektronik sinyallere dönüştürürler (Hildebrandt ve ark., 2012). Bu elektronik sinyaller bir işlemci ünitesi tarafından termogramlara dönüştürülür.

Kızılötesi termal görüntüleme yöntemi *termografi* ve bu yöntemden elde edilen görüntüler *termogram* olarak adlandırılmaktadır. Termogramlar kullanılarak endüstri (Usamentiaga ve ark., 2014), tarım ve gıda (Vadivambal ve Jayas, 2011), spor (Lopez ve ark., 2017), askeri (Kruse, 2001) ve medikal alanlarda çalışmalar gerçekleştirilmektedir. Medikal alanda termoregülasyon (Kammersgaard ve ark., 2013), meme kanseri tespiti (Arora ve ark., 2008; Kontos ve ark., 2011), yeni doğan görüntüleme (Clark ve Stothers, 1980), üroloji (Ng ve ark., 2009) ve dolaşım rahatsızlıkları (Bagavathiappan ve ark., 2008; Bagavathiappan ve ark., 2009) üzerine yapılmış termografi analizi çalışmaları bulunmaktadır.

Yeni doğan bebeklerin hastalıklarının erken teşhis edilmesi ve tedavi süreçlerinin izlenmesi, sağlık durumlarının korunması bakımından büyük önem taşımaktadır (Nur, 2014). Bilgisayarlı tomografi ve manyetik rezonans ile gerçekleştirilen teşhis işlemi hem uzun sürmesi hem de bebeğin bulunduğu termonötral ortamının bozulması bakımından elverişli yöntemler değildirler (Goetz ve ark., 2005). Tedavinin geç başlaması ölüm oranının artmasına neden olabilmektedir. Dünya Bankası verilerine bakıldığında yeni doğan bebeklerin ölüm oranı 2017 yılında ‰ 18 olarak görülmektedir (Group, 2017).

Yeni doğan bebeklerin sıcaklık bilgisinin elde edilmesi ve kalp atış hızı-soluk alıp verme hızı gibi hayati parametrelerin belirlenmesi için vücuda yapıştırılmış elektrotlar veya sensörler kullanılmaktadır. Ancak vücuda yapıştırılan bu cisimler strese, acıya ve bebeklerin hassas vücutlarında zarara neden olabilmektedir (Villarroel ve ark., 2014). Termal görüntüleme teknolojisinin geliştirilmesi ile birlikte hem sıcaklık bilgisinin hem de görüntülerin temassız bir şekilde elde edilmesi sağlanmıştır.

Hastalıkların olduğu bölgelerde, kılcal damarlardaki kan akışının değiştiği ve bu değişimlerin de termal asimetrilere neden olduğu bilinmektedir (Nur, 2014). Bu alanda yapılan çalışmalar termografi kullanılarak yeni doğan bebeklerin vücut sıcaklıklarının takibi, hastalıklarının tespiti ve tedavi süreçlerinin izlenmesi gibi önemli çalışmaların gerçekleştirilebilir olduğunu göstermektedir.

Yeni doğan bebeklerin görüntülenmesi üzerine yapılan çalışmalar incelendiğinde klasik yöntemler kullanılarak İA'nın seçilmesi ve bu alanlar üzerinde sıcaklık analizlerinin gerçekleştirildiği görülmüştür (Abbas ve Leonhardt, 2014; Nur, 2014). Bu yöntemler görüntü ön işleme (yeniden boyutlandırma, gri seviyeye dönüşüm), görüntü iyileştirme (gürültü giderme, histogram eşitleme), İA seçimi ve İA'da meydana gelen değişimlerin izlenmesidir. Probleme dayalı olarak geliştirilen bir algoritma bu teknikler kullanılarak başarılı sonuçlar elde edebilir fakat görüntüleme esnasında meydana gelebilecek bebeğe ait bütün hareketlerin (sırt üstü yatarken tüm açılarda sağa-sola ve yatay eksen boyunca dönmesi vs.) algoritmada tanımlanması gerekmektedir.

Çok katmanlı algılayıcılar, evrişimli sinir ağları, otomatik kodlayıcılar ve çekişmeli üretici ağlar gibi derin öğrenme yöntemlerinin (Deng ve Yu, 2014) gelişmesi ile birlikte görüntü bölütleme ve sınıflandırma, gürültü giderme ve görüntü üretme gibi işlemler görüntüden öğrenme felsefesi esas alınarak gerçekleştirilmektedir. Böylece her duruma karşılık gelecek kod dizini oluşturmak yerine her durumu temsil eden görüntüler bulmak önemli hale gelmiştir.

Bu tez kapsamında ilk defa yeni doğan bebeklerden alınan termal görüntülerin derin öğrenme yöntemleri ile sınıflandırılması gerçekleştirilmiştir. Böylece herhangi bir manuel İA belirleme, özellik çıkarma ve seçme yöntemlerine gerek kalmadan bütün sınıflandırma sürecinin otomatik olarak işlediği bir model oluşturulmuştur. Yeni doğan bebeklere ait termal görüntüler Aralık 2017 – Aralık 2018 tarihleri arasında Variocam HD kızılötesi termal kamera kullanılarak Selçuk Üniversitesi, Tıp Fakültesi, Yeni Doğan Yoğun Bakım Ünitesinde yatan 20 hasta ve 20 sağlıklı bebekten elde edilmiştir.

2 KAYNAK ARAŞTIRMASI

Yeni doğan bebekler ile ilk termografi çalışması 1980 yılında Clark ve Stothers tarafından yapılmıştır (Clark ve Stothers, 1980). Yeni doğan bebeklerin sıcaklık değerlerinin hem termokupl hem de termal kamera ile kayıt edildiği çalışmada iki yöntemden elde edilen değerlerin ortalama karesel hatası karşılaştırılmış ve 0.107 olarak bulunmuştur. Bu sonuç direkt ölçülen sıcaklık değeri ile uzaktan algılanan sıcaklık değeri arasındaki farkın yaklaşık 0.1 °C olduğunu göstermektedir.

Christidis ve arkadaşları 2003 yılında doğumdan itibaren ilk bir saatin termal analizini gerçekleştirmişlerdir (Christidis ve ark., 2003). Bu çalışmaya göre, doğumdan kısa bir süre sonra gövde içerisindeki sıcaklık korunurken periferik bölgelerin daha soğuk hale geldiği görülmüştür. Bu durumu önlemek için bebeğin ılık su ile yıkanması veya anne ile cilt temasında bulunması gerektiği belirtilmiştir.

2010 yılında Rice ve arkadaşları tarafından çok düşük doğum ağırlığına sahip bebeklerin termografi ile analizleri gerçekleştirilmiştir (Rice ve ark., 2010). Bu çalışma kapsamında termografinin çok düşük ağırlıklı bebeklerin görüntülenmesi için uygunluğu, bebeklerin abdomen ve göğüs sıcaklık değerlerinin karşılaştırılması ve abdomen sıcaklığı ile nekrotizan enterokolit (NEC) hastalığı arasındaki ilişkinin ortaya çıkarılması üzerinde durulmuştur. Elde edilen sonuçlar NEC hastalığına sahip bebeklerin abdomen sıcaklığının sağlıklı bebeklere göre daha az olduğunu göstermektedir.

2012 yılında Abbas ve arkadaşları, yeni doğan bebeklerin farklı senaryolar altında (kapalı kuvöz, anne kucağı bakımı ve açık kuvöz) meydana gelebilecek hatalı termal ölçüm durumlarına karşı kompanzasyon teknikleri önermişlerdir (Abbas ve ark., 2012). Görüntüleme esnasında harici ısıtıcı kaynakları, homojen olmayan arka plan sıcaklığı ve hava akışı gibi dış etkenlerin doğru sıcaklık ölçümlerini engellediği belirtilen bu çalışmada matematiksel yaklaşımlar ile sorunların giderilmesine çalışılmıştır. Ancak bu yaklaşımların uygulanabilmesi için ek olarak ortam sıcaklığının termometre ile ölçülerek sisteme eklenmesi gerekmektedir.

2013 yılında Heimann ve arkadaşları tarafından erken doğan bebeklerde termal dengenin ve dağılımın kayıtlanması üzerine bir çalışma gerçekleştirilmiştir (Heimann ve ark., 2013). 10 erken doğan bebeğin bacak, kol, sırt, baş ve üst abdomen sıcaklıkları farklı zaman aralıklarında kayıt edilerek karşılaştırılmıştır. Elde edilen sonuçlar termografinin, vücutta meydana gelecek sıcaklık dalgalanmalarını yakaladığını, belirli

zaman aralıklarında bebeklerin sıcaklık değerlerinin düştüğünü ve bebeklerin günlük bakımlarında bu zaman aralıklarının hesaba katılması gerektiğini göstermektedir.

Yeni doğan bebeklerin abdomen termogramları analiz edilerek termal anomalilerin belirlenmesi çalışması 2014 yılında Ruqia tarafından yüksek lisans tezi olarak gerçekleştirilmiştir (Nur, 2014). Teşhislerin radyolojik bulgularla sınırlı kaldığı belirtilen bu çalışmada 20 normal ve 9 NEC hastalığına sahip bebeğin termogramları analiz edilerek termografi yöntemiyle de teşhislerin gerçekleştirilebileceği gösterilmiştir. Geliştirilen sistem (1) görüntü ön işleme (rotasyon değiştirme ve normalizasyon), (2) görüntü iyileştirme (gürültü giderme, arka plan kaldırma ve karşıtlık iyileştirme), (3) İA bölütleme (manuel abdomen İA seçimi ve İA'yı ikiye ve dörde bölme) ve (4) görüntü analizi (birinci ve ikinci dereceden istatistiksel özelliklerin çıkarılması, bu özelliklerin farklılığının hesaplanması ve ortalamalarının alınması) adımlarını içermektedir. Elde edilen sonuçlar NEC hastalığı taşıyan bebekler ile sağlıklı bebeklerin termal asimetri derecelerinin birbirlerinden farklı olduğunu göstermektedir.

2014 yılında Abbas ve Leonhardt tarafından termal sensörler ile yeni doğan izlemesinde termonötral çevresel ortamın sürdürülmesi için akıllı bir sistem tasarımı önerilmiştir (Abbas ve Leonhardt, 2014). Termonötral çevresel ortam, yeni doğan bebeklere minimum enerji – oksijen gereksinimi sağlayan ve vücut sıcaklığını 36,5 °C'de koruyan ortamdır. Çalışmada en uygun termografi düzeltmeleri için yayılım varyansları dikkate alınmıştır ve sıcaklık değerlerinin analiz edilmesi için vektör tabanlı bir şablon eşleştirme metodu geliştirilmiştir. Bebeklerin geometrisine göre İA izleme algoritmaları uygulanmıştır ancak vücudun her durumu için uygun bir İA bulunamadığından hatalar oluştuğu belirtilmiştir.

Yine 2014 yılında Abbas ve Leonhart yeni doğan bebeklerin termal görüntüleme analizlerini açıklamışlardır (Abbas ve Leonhardt). Tümör, iltihap ve enfeksiyon gibi anomalilerin lokal sıcaklık artışlarına veya asimetric modeller oluşmasına neden olduğu belirtilen bu çalışmada yüksek ısı yayma potansiyeline sahip olduğu için baş görüntüleri üzerinden analizler gerçekleştirilmiştir. Termogramlara RGB görüntü yaklaşımı uygulanarak özellik katsayılarını çıkarmak için dalgacık dönüşümü ve görüntülerin boyutunu azaltmak için temel bileşenler analizi kullanmışlardır.

2017 yılında Knobel ve arkadaşları tarafından bebeklerin doğumundan itibaren ilk bir haftadaki vücut sıcaklıklarının analizi gerçekleştirilmiştir (Knobel-Dail ve ark., 2017). Bu çalışma kapsamında bölgesel olarak sıcaklık farklılıkları termografi ve termistör ile ölçülerek termografi başarımının gösterilmesi amaçlanmıştır. Çalışmanın

sonucunda erken doğan bebeklerin termal anomaliler gösterdiği ve ilk 12 saat içerisinde ayak sıcaklık değerlerinin abdomen sıcaklık değerlerinden daha yüksek olduğu hem termografi hem de termistör kullanılarak gösterilmiştir.

Savaşçı ve Ceylan, 2018 yılında 32 yeni doğan üzerinden, termal asimetrisinin veya zamana bağlı olarak ortaya çıkan termal farklılıkların izlenmesinin önemini belirtmişlerdir (Savaşçı ve Ceylan, 2018). Bu değişimlerin değerlendirilmesi; acil müdahale ihtiyacı, yeni tedavi planlaması veya mevcut durumun korunması gibi hayati kararların verilmesini sağlayacaktır. RGB görüntüleri gri seviyeye çevirme, görüntü ön – işleme, görüntü geliştirme ve İA seçme yöntemleri ile analiz etmişlerdir. Elde edilen sonuçlar soldan-sağa ortalama, standart sapma, varyans ve çarpıklık değerlerinin hasta ve sağlıklı yeni doğanlarda farklı olduğunu göstererek bu değerlerin hasta yeni doğanlarda daha geniş bir dağılıma ve daha yüksek değerlere sahip olduğunu göstermişlerdir. Böylece, hasta yeni doğanlardaki termal asimetri derecesinin sağlıklı yeni doğanlardan daha yüksek olduğu gösterilmiştir.

Örnek ve arkadaşları, termogramlar değerlendirilirken doğru yaklaşımların belirlenmesi üzerine bir çalışma gerçekleştirmişlerdir (Örnek ve ark., 2018). Bu çalışmada 5'i hasta ve 5'i sağlıklı olmak üzere 10 yeni doğan bebek üzerinden elde edilen termogramlar RGB uzayına ve sıcaklık haritalarına dönüştürülmüş ve dalgacık dönüşümü ile görüntü katsayıları elde edilmiştir. Ters dalgacık dönüşümü ile yeniden birleştirilen görüntülerde meydana gelen bozulmalar tepe sinyal gürültü oranı, ortalama karesel hata ve yapısal benzerlik indeksi esas alınarak gösterilmiştir. Elde edilen sonuçlar sıcaklık haritalarındaki bozulma miktarının RGB görüntülerdeki bozulma miktarından daha düşük olduğunu ortaya koymuştur. Bu çalışma klasik yöntemler kullanıldığında RGB görüntüler yerine sıcaklık haritaları kullanımının daha etkili olduğunu göstermektedir.

3 MATERYAL VE YÖNTEM

Bu bölümde termal görüntülerin elde edilmesi, termogram yaklaşımlarının incelenmesi, termal görüntülerin artırılması, bölütlenmesi, termal görüntülerden özellik çıkarılması, önemli özelliklerin seçilmesi, makine öğrenimi-derin öğrenme yöntemleri ile sınıflandırılması uygulamalarının ve doğrulama-değerlendirme kriterlerinin detayları verilmiştir. Bu bölümdeki işlemler MATLAB, PYTHON ve WEKA kullanılarak Nvidia Quadro K2200 4 GB 128 Bit 640 Cuda GPU üzerinde koşturulmuştur.

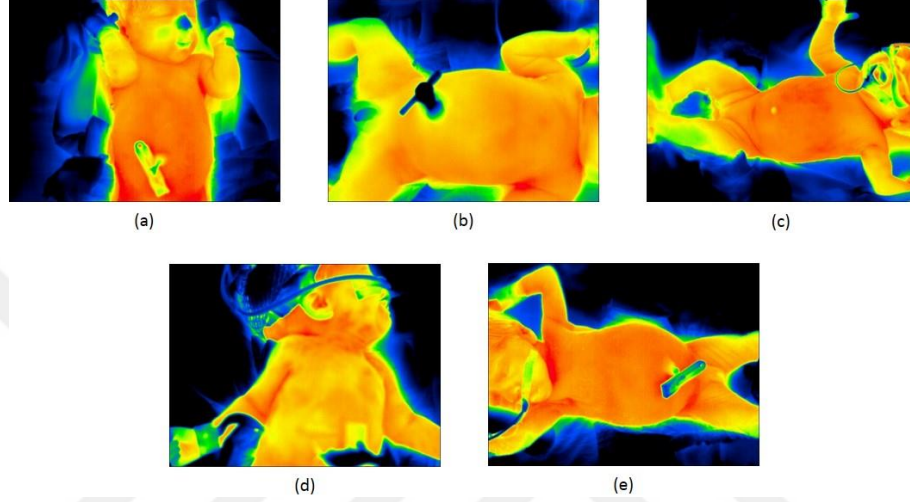
3.1 Yeni Doğan Bebeklere ait Termogramların Elde Edilmesi

Bu tez çalışmasında, Infratec© firmasının ürünü olan Variocam HD kızılötesi kamera kullanarak Selçuk Üniversitesi, Tıp Fakültesi, Yeni Doğan Yoğun Bakım Ünitesinde bulunan 20 hasta ve 20 sağlıklı bebekten bir yıllık periyot içerisinde (Aralık 2017 – Aralık 2018) 4000 termogram elde edilmiştir. Etik kurul raporu EK-1’de verilmiştir. Kullanılan termal kameranın çözünürlüğü 640 x 480 piksel ve termal duyarlılığı yaklaşık 0.01 °C’olup bir dakika içerisinde art arda 100 termogram elde etmektedir. Görüntü almak için oluşturulan sistem Şekil 3.1’de gösterilmiştir.



Şekil 3.1. Görüntü almak için oluşturulan sistem (a) kızılötesi termal kamera (b) kuvöz (c) yeni doğan bebek (d) taşınabilir bilgisayar

Bu sisteme göre görüntü alınmadan önce bebeğin üstü çıkartılmalı ve kuvözden yaklaşık 60-100 cm arası mesafede görüntü alınmalıdır. Bebeğin konforunun bozulmaması için tüm işlemler olabildiğince hızlı olmalıdır. Uzun süren çekimler bebeğin termal dengesini bozarak hipotermiye (Danzl ve Pozos, 1994) girmesine neden olabilmektedir. Oluşturulan sistem kullanılarak elde edilen görüntülerin bir kısmı Şekil 3.2’de gösterilmiştir.



Şekil 3.2. Elde edilen görüntüler arasından rastgele seçilen beş görüntü

Çizelge 3.1’den görüldüğü gibi, yeni doğan sağlıklı bebeklerin doğum haftaları 24 ile 34 hafta, doğum ağırlıkları 720 ile 2800 g arasında değişmektedir.

Çizelge 3.1. Sağlıklı bebeklere ait doğum ağırlıkları ve doğum haftaları

Bebek	Doğum Ağırlığı (g)	Doğum Haftası + Günü
Sağlıklı_1	720	24
Sağlıklı_2	1825	34
Sağlıklı_3	1300	28
Sağlıklı_4	1100	28
Sağlıklı_5	1375	28 + 2
Sağlıklı_6	2200	32
Sağlıklı_7	1580	29
Sağlıklı_8	1690	30 + 5
Sağlıklı_9	955	27
Sağlıklı_10	1175	28 + 4
Sağlıklı_11	1870	31 + 6
Sağlıklı_12	1900	32 + 5
Sağlıklı_13	2300	33 + 5
Sağlıklı_14	1195	29 + 3
Sağlıklı_15	950	28 + 5
Sağlıklı_16	2800	35
Sağlıklı_17	1605	33 + 6
Sağlıklı_18	1885	32 + 1
Sağlıklı_19	1660	32 + 1
Sağlıklı_20	1435	29

20 sağlıklı bebekten 10'una bir hastalık için teşhis konulurken diğer 10 bebeğe iki hastalık için teşhis konulmuştur. Çizelge 3.2'de gösterildiği gibi hastalık teşhisi konulmuş yeni doğan bebeklerin doğum haftaları 28 ile 39 hafta, doğum ağırlıkları ise 565 ile 3300 g arasında değişmektedir.

Çizelge 3.2. Hasta bebeklere ait doğum ağırlıkları, doğum haftaları ve teşhis konulan hastalıkları

Bebek	Doğum Ağırlığı (g)	Doğum Haftası + Günü	Hastalık 1	Hastalık 2
Sağlıksız_1	2015	34	IKK	Hypothermia
Sağlıksız_2	1100	28	RDS	—
Sağlıksız_3	3300	33 + 1	RDS	—
Sağlıksız_4	3000	36	NEC	PDA
Sağlıksız_5	865	28	RDS	Moaning
Sağlıksız_6	1890	33 + 2	RDS	DAB
Sağlıksız_7	2055	34	PAB	—
Sağlıksız_8	1985	32	NEC	—
Sağlıksız_9	2305	34 + 1	IO	—
Sağlıksız_10	2280	37	HLH	—
Sağlıksız_11	2200	35	OA	—
Sağlıksız_12	1590	30	RDS	Neonatal syphilis
Sağlıksız_13	1790	31	NEC	Sepsis
Sağlıksız_14	2680	38	DH	Adeno
Sağlıksız_15	2700	35	TTN	—
Sağlıksız_16	565	28	IUGR	—
Sağlıksız_17	3079	37	OA	—
Sağlıksız_18	3300	38	Hydrocephalus	Myelomeningocele
Sağlıksız_19	1100	33	AORTC	Down syndrome
Sağlıksız_20	3200	39	NEC	MAS

Çizelge 3.3'te sağlıklı ve hasta bebeklere ait doğum ağırlığı ve doğum haftası özelliklerinin istatistiksel dağılımları verilmiştir.

Çizelge 3.3. Yeni doğan bebeklere ait özelliklerinin istatistiksel dağılımları

Bebekler		Doğum Ağırlığı (g)			Doğum Haftası		
Sağlık Durumu	Sayı	Ortalama	Medyan	Standart Sapma	Ortalama	Medyan	Standart Sapma
Sağlıklı	20	1576	1592,5	498,0351	30,1	29,5	2,6814
Hasta	20	2149,95	2127,5	802,9194	33,65	34	3,3056

3.2 Termogram Yaklaşımları

Termogramlar elde edildikten sonra sıcaklık haritaları ve RGB görüntüler olarak kaydedilip kullanılabilir. Bu bölümde sıcaklık haritaları ve RGB görüntü yaklaşımları açıklanarak karşılaştırmaları yapılmıştır.

3.2.1 Sıcaklık haritaları

Sıcaklık haritası, direkt olarak sıcaklık değerlerinin bulunduğu iki boyutlu matrise verilen isimdir. Bu sıcaklık değerleri minimum sıcaklık 0, maksimum sıcaklık 255 değerine eşit olacak şekilde normalize edilip görselleştirildiğinde Şekil 3.3'te görüldüğü gibi gri seviyeli bir görüntü elde edilmektedir.

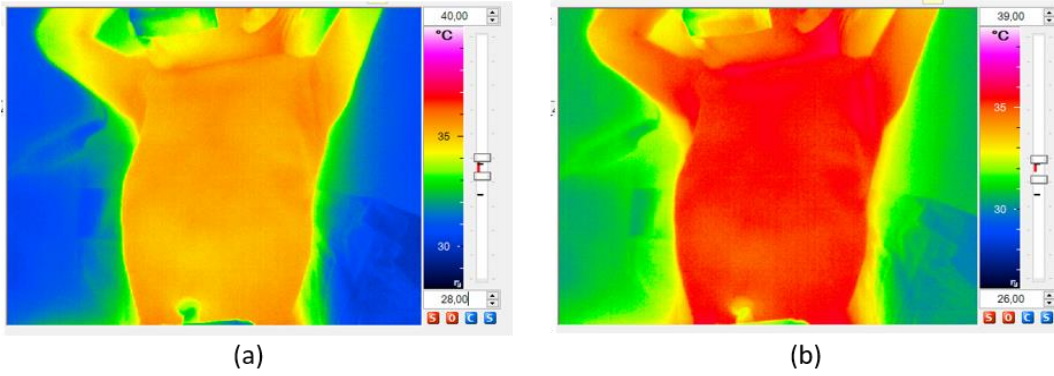


Şekil 3.3. Sıcaklık haritasının görselleştirilmiş hali ve ilgili bölgenin sıcaklık değerleri

Şekil 3.3'te yer verilen görüntüde düşük sıcaklık değerleri siyah ve yüksek sıcaklık değerleri beyaz bölgelere karşılık gelmektedir.

3.2.2 RGB görüntüler

Termogramlar elde edildikten sonra sıcaklık değerlerine göre alt ve üst eşik değerleri belirlenerek RGB görüntüler elde edilmektedir. Şekil 3.4'te farklı eşik değerleri arasında renklendirilmiş bir termogramın RGB görüntüleri gösterilmiştir.



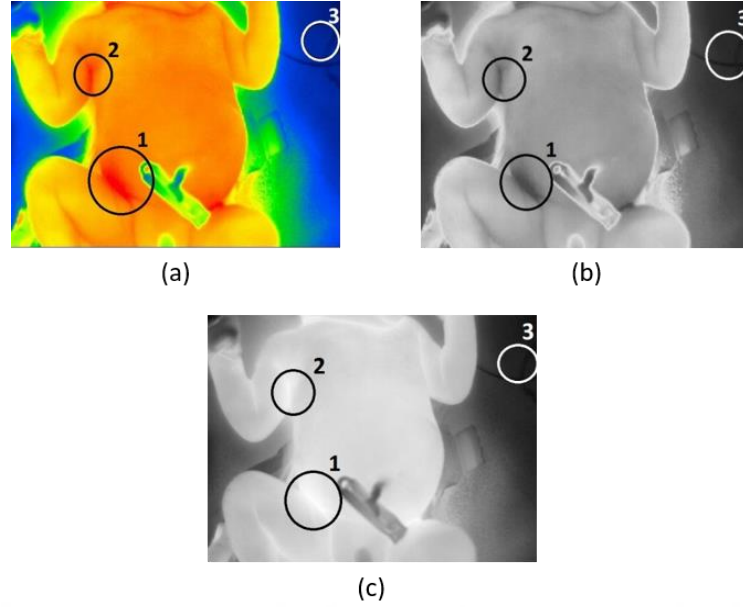
Şekil 3.4. Farklı eşik değerleri arasında renklendirilmiş görüntüler (a) 28°C-40°C aralığında renklendirilmiş görüntü (b) 26°C-39°C aralığında renklendirilmiş görüntü

Şekil 3.4 (a)'da yer alan görüntü 28 – 40 °C, Şekil 3.4 (b)'de yer alan görüntü ise 26 – 39 °C değerleri arasında minimum seçilen değer siyah – koyu mavi, maksimum seçilen değer açık pembe – beyaz olacak şekilde renklendirilmiştir. Her bebek kendine özgü sıcaklık değerleri içerdiğinden, her bebek için farklı eşik değerleri belirlenmesi gerekmektedir. Bu eşik değerleri tamamen kullanıcının gözlem yeteneğine ve deneyimlerine bağlı olarak seçilmektedir.

3.2.3 Sıcaklık haritaları ile RGB görüntülerin karşılaştırılması

Termal görüntülerde medikal analizler yapılırken termal asimetri esas alındığından termogramı oluşturan tüm sıcaklık değerlerinin saklanması gerekmektedir. Bu nedenle RGB görüntü oluştururken seçilen eşik değerleri hiçbir bilgi kaybı yaşanmayacak şekilde (minimum eşik değeri minimum sıcaklık değerinden küçük, maksimum eşik değeri maksimum sıcaklık değerinden büyük) seçilmelidir. Aksi takdirde eşik değerleri dışarısında kalan alanlarda siyah bölgeler meydana gelecek yani RGB görüntüde bilgi kaybı yaşanmış olacaktır. Bu işlemlerin gerçekleştirilmesi, işlem yükünün artması ve zamandan kaybetmek anlamlarına da gelmektedir.

Diğer taraftan RGB görüntüler direkt olarak gri seviyeli görüntülere dönüştürüldüklerinde bu gri seviyeli görüntüler üzerinde hatalı bilgiler oluşmaktadır. Bu durum Şekil 3.5'te gösterilmiştir.



Şekil 3.5. RGB görüntüler ve sıcaklık haritalarının karşılaştırılmasına ait görüntüler (a) RGB görüntü (b) gri seviyeye dönüştürülmüş RGB görüntü (c) sıcaklık haritası

Şekil 3.5'te 1. ve 2. bölge sıcak bölgelere 3. bölge ise soğuk bölgeye karşılık gelmektedir. Şekil 3.5 (a)'da sıcak bölgeler kırmızı ve soğuk bölgeler mavi olarak gösterilmesine rağmen Şekil 3.5 (b)'de görüldüğü gibi gri seviyeli hale çevrildiğinde üç bölge de siyah olarak görülmektedir. Bu durum RGB'den gri seviyeye geçişin yanlış bir yaklaşım olduğunu ispatlamaktadır. RGB'den gri seviyeye geçiş yerine direkt olarak sıcaklık haritası görselleştirildiğinde ise sıcak alanlar beyaza yakın soğuk alanlar ise siyaha yakın tonlarda görülmektedir. RGB görüntü ve sıcaklık haritası yaklaşımları özet bir şekilde Çizelge 3.4'te gösterilmiştir.

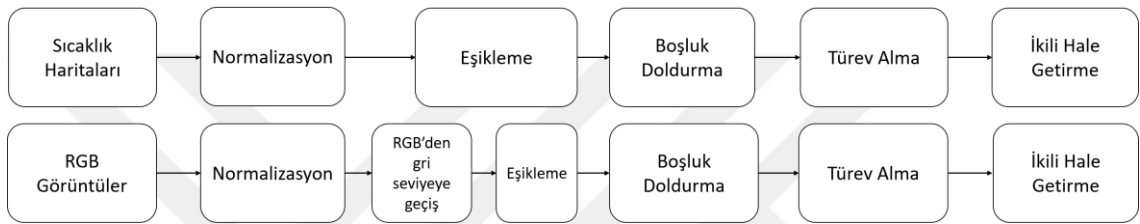
Çizelge 3.4. RGB görüntüler ve sıcaklık haritalarının karşılaştırılması

RGB GÖRÜNTÜ	SICAKLIK HARİTASI
Oluşturmak için alt ve üst eşik değerleri belirlenmesi gerekir.	Oluşturmak için herhangi bir eşik değeri seçimi gerektirmez.
Kullanıcının gözlem yeteneğine bağlıdır.	Kullanıcıdan bağımsızdır.
Zaman maliyeti vardır.	Zaman maliyeti yoktur.
Gri seviyeye çevrildiğinde hatalı bilgiler ortaya çıkar.	Termogramı içeren tüm değerler korunur.

Bu bölümde sıcaklık haritaları yerine RGB görüntüler ile çalışılacağı zaman (1) bilgi kaybı yaşanmayacak şekilde eşik değerleri seçilmesi (2) analiz yapılırken görüntülerin RGB'den gri seviyeye dönüştürülmemesi gerektiği açıkça belirtilmektedir.

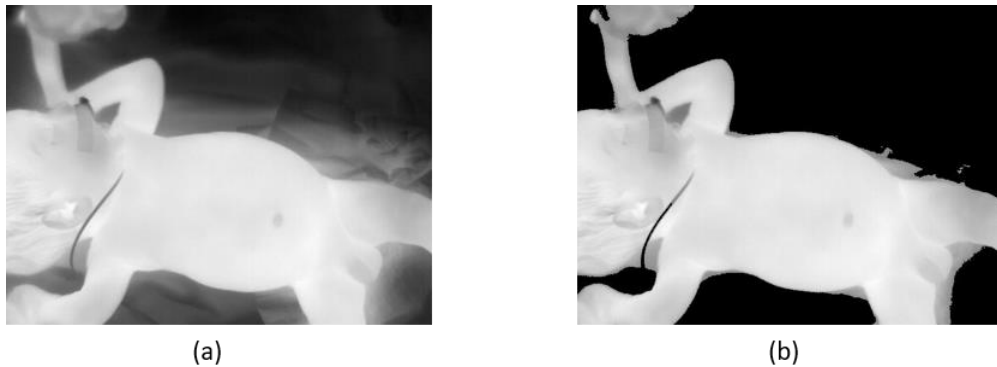
3.3 Termal Görüntülerin Bölütlenmesi

Termal görüntüler nesnelerin sıcaklık değerleri esas alınarak oluşturulduğundan canlı bir nesne ile cansız bir nesnenin birbirinden ayrılması diğer yöntemlere nazaran daha kolay olabilmektedir. Çünkü canlı nesneler yüksek sıcaklık değerine, cansız nesneler ise düşük sıcaklık değerine sahip olduğundan belirlenecek bir eşik değeri ile ayırım kolaylıkla yapılabilecektir ancak görünür görüntülemeye yakın bir arka plan olması durumunda nesne ile canlılığın ayrımı kolaylıkla yapılamayacaktır. Bölütleme işlemlerini gösteren blok diyagramı Şekil 3.6'da gösterilmiştir.



Şekil 3.6. Sıcaklık haritaları ve RGB görüntüler için oluşturulan bölütleme adımlarının blok diyagramı

Kuvöz içerisinde bulunan bebeklerin zemin görüntüsünden ayrılması için Otsu eşikleme metodu (Otsu, 1979) kullanılmıştır. Otsu metodu görüntü içerisindeki tüm değerler için sınıf içi varyans değerlerini hesaplar ve en düşük varyansa sahip değer eşik değeri olarak seçilir. Kuvöz içerisinde bu değerler 33.3 °C, 33.7 °C ve 34.1 °C gibi değerlere karşılık gelmektedir. Eşik değerinden yüksek olan değerler beyaz diğer değerler ise siyah olarak görünür. Sıcaklık haritalarına eşikleme uygulamadan önceki görüntü Şekil 3.7 (a)'da ve uygulandıktan sonra oluşan görüntü Şekil 3.7 (b)'de gösterilmiştir.



Şekil 3.7. (a) Sıcaklık haritası (b) eşik değeri uygulandıktan sonra oluşan görüntü

Bazı yeni doğan bebekler Şekil 3.8 (a)'da görüldüğü göbek bağlarına tutturulmuş mandallar taşımaktadırlar ve bu cisimler vücuda göre daha soğuk olduklarından dolayı eşikleme ile beraber görüntüden atılırlar. Vücut yüzeyinde meydana gelecek bu durumun engellenmesi için *boşluk doldurma* yöntemi (Soille, 2013) kullanılmıştır ve oluşan görüntü Şekil 3.8 (b)'de gösterilmiştir. Boşluk doldurma ikili görüntüler (beyaz bölge = 1, siyah bölge = 0) için kullanılan ve beyaz bölgeler (1 değerleri) arasında kalan siyah bölgeleri (0 değerleri) tespit ederek buralara 1 değeri atayan algoritmadır.



Şekil 3.8. (a) Göbeğinde mandal taşıyan bir yeni doğana ait sıcaklık haritası (b) boşluk doldurma uygulandıktan sonra oluşan görüntü

Boşluk doldurma işlemi tamamlandıktan sonra vücut yüzeyinde meydana gelen sıcaklık değişiminin etkisinin gözlemlenmesi için görüntünün türevi alınmıştır. Eşitlik 3.1'de verilen türev işlemi her bir satırdaki piksel değerinin $f(x, y)$ bir önceki piksel değerinden $f(x, y-1)$ farkını ifade etmektedir.

$$\frac{df}{dx} = f(x, y) - f(x, y-1) \quad (3.1)$$

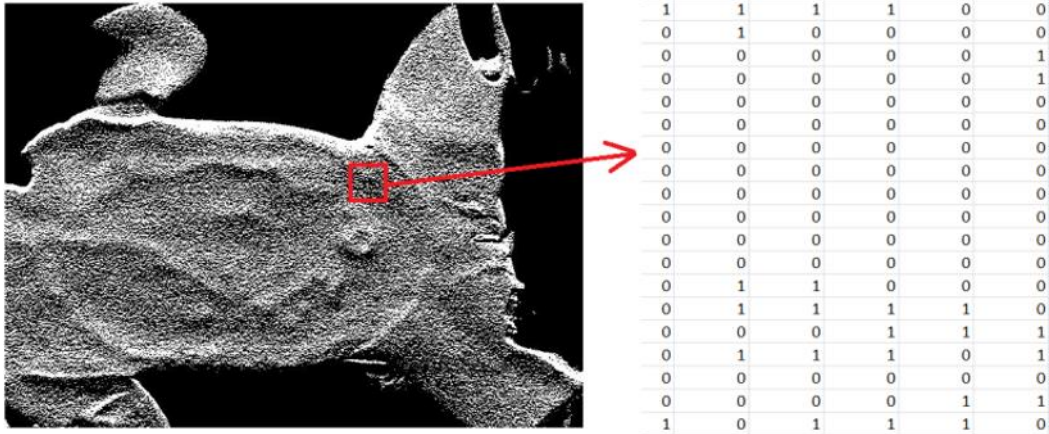
Böylece yalnızca sıcaklık farklılıklarının bulunduğu bir ifade elde edilmiştir. Şekil 3.9'da görüldüğü gibi bu ifade hem pozitif hem de negatif değerler içermektedir. Bu nedenle 0'dan küçük değerler 0'a, 0'dan büyük değerler ise 1'e eşitlenerek (Eşitlik 3.2) ifade ikili hale getirilmiştir. İkili hale getirilmiş görüntü Şekil 3.10'da gösterilmiştir.

$$g(x, y) = \begin{cases} 1, f(x, y) > 0 \\ 0, f(x, y) \leq 0 \end{cases} \quad (3.2)$$

Burada $f(x, y)$ mevcut piksel değerini $g(x, y)$ ise çıkış değerini ifade etmektedir.

0.09	0.08	0.02	0.02	-0.02	-0.06
-0.01	0.03	-0.05	-0.07	-0.03	-0.03
-0.21	-0.22	-0.17	-0.13	-0.08	0.04
-0.24	-0.27	-0.21	-0.09	-0.04	0.05
-0.27	-0.19	-0.12	-0.09	-0.06	-0.09
-0.15	-0.14	-0.1	-0.11	-0.12	-0.15
-0.05	-0.08	-0.06	-0.02	-0.01	-0.16
-0.03	0	-0.06	-0.12	-0.15	-0.11
-0.14	-0.14	-0.15	-0.16	-0.21	-0.15
-0.24	-0.34	-0.34	-0.36	-0.35	-0.26
-0.28	-0.24	-0.31	-0.36	-0.36	-0.4
-0.03	-0.16	-0.17	-0.2	-0.2	-0.21
-0.03	0.05	0.02	-0.04	-0.07	-0.1
-0.06	0.02	0.04	0.05	0.03	0
-0.04	-0.04	-0.02	0.02	0.05	0.05

Şekil 3.9. Sıcaklık Değişimleri Matrisi

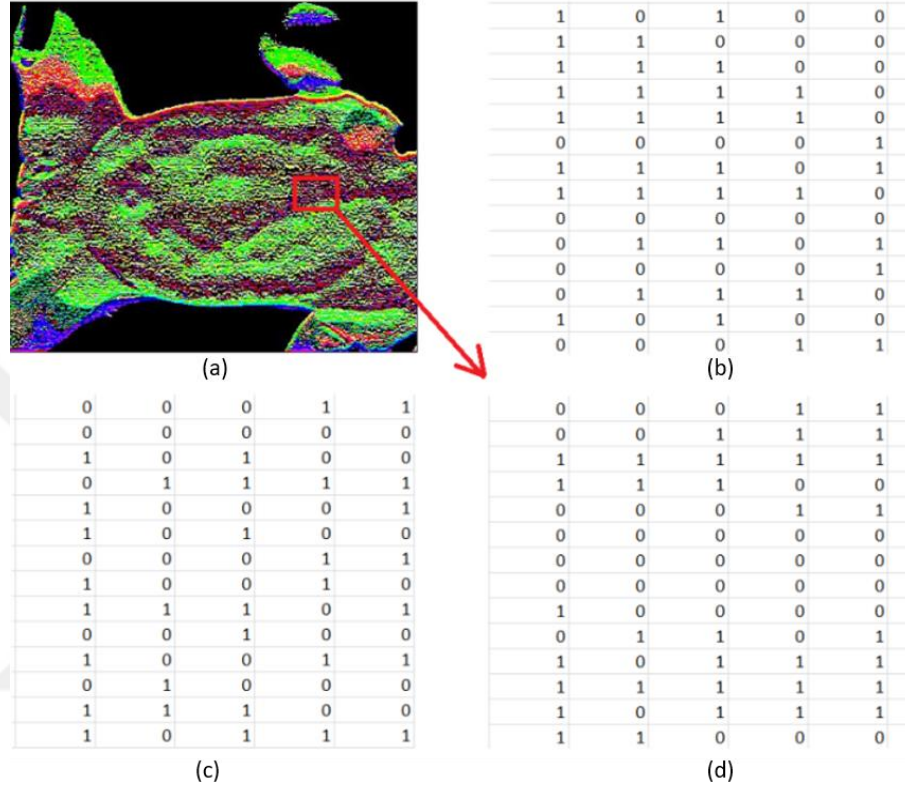


Şekil 3.10. İkili hale getirilmiş görüntü ve seçilen bölgenin piksel değerleri

Buraya kadar olan kısımda sıcaklık haritalarının nasıl bölümlendiği açıklanmıştır. Sıcaklık haritaları tek boyutlu matrisler iken RGB görüntüler üç boyutlu matrislerdir. RGB görüntüleri tek boyuta indirmek amacıyla öncelikle renkli görüntüden gri seviyeli görüntüye geçiş yapılır. Bu işlem renkli görüntünün parlaklık bilgisinin çıkarılması anlamına gelmektedir. Renkli bir görüntünün parlaklık bilgisi (Y) Eşitlik 3.3'te verildiği gibi hesaplanmaktadır (Gunjal ve Mali, 2011):

$$Y(x, y) = 0.299R(x, y) + 0.587G(x, y) + 0.114B(x, y) \quad (3.3)$$

Burada $R(x, y)$, $G(x, y)$ ve $B(x, y)$ renkli bir görüntüye ait kırmızı, yeşil ve mavi renk bantlarındaki değerlere karşılık gelmektedir. RGB görüntünün parlaklık bilgisi kullanılarak sıcaklık haritalarında uygulanan bölütleme adımları tekrar eder. Bölütlenmiş RGB görüntü Şekil 3.11’de gösterilmiştir.



Şekil 3.11. İkili hale getirilmiş RGB görüntü ve renk uzaylarındaki piksel değerleri (a) ikili hale getirilmiş RGB görüntü (b) kırmızı renk uzayındaki piksel değerleri (c) yeşil renk uzayındaki piksel değerleri (d) mavi renk uzayındaki piksel değerleri

3.4 Termal Görüntülerden Özellik Çıkarma

Makine öğrenimi algoritmaları ile sınıflandırma gerçekleştirileceği zaman direkt olarak görüntüyü kullanmak yerine görüntüyü temsil eden özelliklerin çıkarılması gerekmektedir (Sebastiani, 2002). Bu işlem sınıflandırma yapılırken gerçekleştirilen en önemli aşamalardan bir tanesidir. Dalgacık dönüşümü, ripplet – ridgelet dönüşümü (Öztürk ve ark., 2015) ve yerel ikili desen (Ojala ve ark., 2002) gibi geliştirilmiş pek çok özellik çıkartma algoritması bulunmaktadır. Bu algoritmalar kullanılarak görüntülerden kenar, köşe, yönelim, hacim, çözünürlük ve histogram gibi özelliklerin çıkartılması sağlanmaktadır.

3.4.1 Yerel ikili desen (YİD)

Yerel ikili desen algoritması doku sınıflandırma uygulamalarında yüksek performans gösteren bir özellik çıkarma algoritmasıdır (Qian ve ark., 2011). Görüntü 3x3, 5x5 ve 16x16 gibi deneysel olarak belirlenen hücelere bölünerek hücre merkezindeki değer ile komşu değerlerin karşılaştırılması yapılır.

3x3 hücre örneği için Eşitlik 3.4'teki gibi bir hücre tanımlansın:

$$\begin{bmatrix} g1 & g2 & g3 \\ g8 & gm & g4 \\ g7 & g6 & g5 \end{bmatrix} \quad (3.4)$$

Burada gm merkez değeridir ve çevresindeki elemanlar sol üst köşeden başlayarak Eşitlik 3.5'teki gibi bir matrise yazdırılır.

$$[g1, g2, g3, g4, g5, g6, g7, g8] \quad (3.5)$$

Bir sonraki aşamada Eşitlik 3.6'da verildiği gibi matris içerisindeki tüm değerlerden merkez değeri çıkarılır.

$$[g1 - gm, g2 - gm, g3 - gm, g4 - gm, g5 - gm, g6 - gm, g7 - gm, g8 - gm] \quad (3.6)$$

Eşitlik 3.6'dan elde edilen matris Eşitlik 3.2'de açıklandığı gibi ikili hale getirilir. Bu sekiz basamaklı ikili tabandaki sayı onluk tabana çevrilerek kaydedilir. Şekil 3.12'de 3x3 hücreye ait örneğin basit kod şekli verilmiştir

```

Giriş: Görüntü // veri seti
Çıkış: Histogram // özellikler

başla

for i=0 to görüntünün_yüksekliği:

    for j=0 to görüntünün_genişliği:

        merkez = görüntü(i,j)
        bir = görüntü (i-1,j-1) // merkezin sol üst köşesi
        iki = görüntü (i,j-1) // merkezin üstü
        üç = görüntü (i+1,j-1) // merkezin sağ üst köşesi
        dört = görüntü (i+1,j) // merkezin sağı
        beş = görüntü (i+1,j+1) // merkezin sağ alt köşesi
        altı = görüntü (i,j+1) // merkezin altı
        yedi = görüntü (i-1,j+1) // merkezin sol alt köşesi
        sekiz = görüntü (i-1,j) // merkezin solu

        matris = [bir, iki, üç, dört, beş, altı, yedi, sekiz]

        for t=0 to matris_genişliği:
            matris (t) = matris (t) - merkez
            if matris (t) > 0:
                matris (t) = 1
            else:
                matris (t) = 0
        end
    end

    desimal_değer = ikiliden_desimal_hesapla (matris)

    yeni_veri (i,j) = desimal_değer;
end
end

özellikler = histogram_hesapla (yeni_veri)

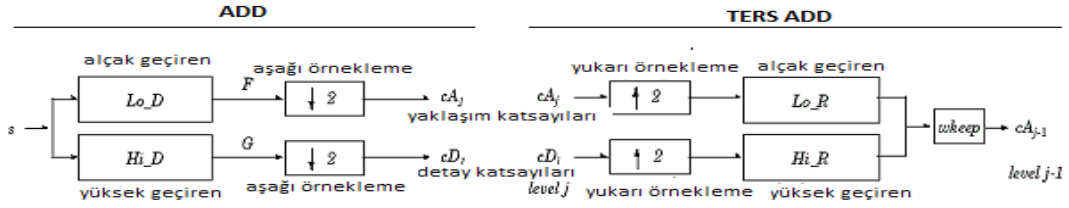
bitir

```

Şekil 3.12. 3x3 hücreye ait YİD algoritması

3.4.2 Ayrık dalgacık dönüşümü (ADD)

Dönüşüm, bir verinin özniteliklerinin korunarak farklı bir forma getirilmesi işlemidir. Fourier dönüşümü gibi yöntemlerde görüntüdeki frekans bileşenlerinin, görüntünün tümünde bulunduğu kabul edilir (Yaşar, 2015). Bu nedenle bir frekans bileşeninin görüntünün neresinde olduğu bilinmemektedir. ADD, görüntüleri alçak ve yüksek frekans bileşenlerine ayırarak alçak frekanslarda yaklaşım, yüksek frekanslarda detay katsayılarının elde edilmesini sağlar (Öztürk, 2015). Veri özellikleri yaklaşım kısmında korunur ve bu dönüşüm işlemi kademeli olarak yaklaşım kısmı üzerinden devam ettirilir. Gerçek sinyal, elde edilen frekans bileşenlerinin yukarı örnekleme ile yeniden elde edilmektedir. Şekil 3.13'te ADD ve ters ADD gösterilmiştir.



Şekil 3.13. ADD ve Ters ADD Yapısı

3.5 Hızlı Korelasyon Tabanlı Filtre (HKTF) ile Özellik Seçme

Bir veriye ait ne kadar çok özellik var ise sınıflandırıcı işlem yoğunluğu da o kadar artmaktadır. Özellik seçme kullanarak gereksiz özelliklerin kaldırılması, özellik boyutunun azalması ve dolayısı ile işlem süresinin azalması sağlanmaktadır (Dash ve Liu, 1997).

HKTF algoritması ile çıkartılan özelliklerden önemli olanlarının seçilmesi gerçekleştirilmektedir (Yu ve Liu, 2003). HKTF iki aşamadan meydana gelir:

- Her bir özelliğin sınıfla ilgili olup olmadığı incelenir ve sınıfla ilgili olmayan özellikler kaldırılır.
- Kalan özellikler kendi aralarında ele alındığında gerekli olup olmadığına karar verilir.

Bu işlemlerin gerçekleştirilmesi için Eşitlik 3.7’de verilen simetrik belirsizlik değerinin hesaplanması gerekmektedir. Simetrik belirsizliğin hesaplanması için bilgi kazancı (BK) (Eşitlik 3.8), marjinal entropi (Eşitlik 3.9) ve durumsal entropi (Eşitlik 3.10) değerleri elde edilir.

$$SB(A, C) = 2 \frac{BK(A, C)}{H(A) + H(C)} \quad (3.7)$$

$$BK(A, C) = H(A) - H(A | C) = H(C) - H(C | A) \quad (3.8)$$

$$H(A) = - \sum_i p(A_i) \cdot \log_2 p(A_i) \quad (3.9)$$

$$H(A | C) = - \sum_i p(A_i, C_i) \log \frac{p(A_i, C_i)}{p(C_i)} \quad (3.10)$$

BK (A, C), A ile C arasındaki bilgi kazancını ifade etmektedir. Bilgi kazancının hesaplanması için Eşitlik 3.8’de verilen marjinal entropi ($H(A)$) ve Eşitlik 3.9’de verilen durumsal entropi ($H(A|C)$) değerleri hesaplanır. Burada $p(A_i)$ A_i elemanının bulunduğu sınıf içerisindeki eleman sayısının tüm sınıflarda bulunan eleman sayısına oranını, $p(A_i, C_j)$ ise $A = A_i$ ve $C = C_j$ iken buldukları sınıf içerisindeki eleman sayısının tüm sınıflarda bulunan eleman sayısına oranını ifade eder.

Algoritma işleyişi şu şekildedir:

- Eşik değeri deneysel olarak belirlenir.
- Tüm özellikler ile sınıf arasında SB değeri hesaplanır.
- Eşik değeri altında kalan SB değerlerine sahip özellikler kaldırılır.
- Kalan özelliklerin kendi aralarında SB değerleri hesaplanır.
- Elde edilen SB değerleri büyükten küçüğe doğru sıralanır.
- İstenilen sayıda özellik seçilir.

3.6 Makine Öğrenimi

Bilgisayar programı yazma, akıl yürütme, matematiksel işlemler gerçekleştirme ve araba sürme gibi birçok insan aktivitesi zeka kavramı ile ilişkilendirilmektedir. Bu işlemleri insan gücü ve düşüncesi olmaksızın bir sistem ile gerçekleştirme çalışmaları *Yapay Zeka* olarak adlandırılmaktadır (Nilsson, 2014). Alan Turing 1950 yılında “Makineler Düşünebilir mi?” diyerek Turing Testini tanıtmıştır (Turing, 1950). Bu teste göre bir işleyişin arkasında bir makinenin olup olmadığı ayırt edilemiyorsa yapay zeka ile karşı karşıya olduğu anlamına gelmektedir. Bazı yapay zeka uygulamaları:

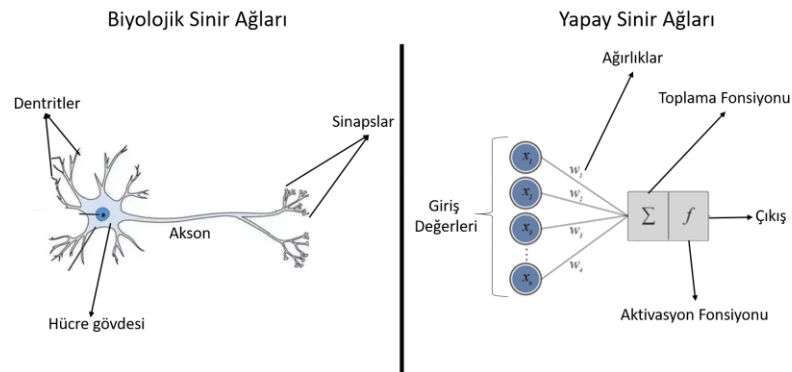
- Doğal dil işleme
- Uzman danışman sistemler
- Teorem kanıtlama
- Robotik
- Otomatik programlama
- Otomatik rota oluşturma
- Çok oyunculu oyunlar

- Otonom araçlar
- Bilgisayar görmesi
- Makine öğrenimi

Makine öğrenimi ise örnek verilere veya geçmiş deneyimlere dayanarak bir performans kriterini optimize etmek için bilgisayarın kullanımınıdır (Alpaydin, 2009). Bu algoritmalar kullanılarak sınıflandırma, tahmin etme ve bölütleme gibi işlemler gerçekleştirilmektedir. Bu tez çalışmasında, makine öğrenimi algoritmalarından yapay sinir ağları (Cowan ve Sharp, 1988), destek vektör makineleri (Cortes ve Vapnik, 1995), karar ağaçları (Safavian ve Landgrebe, 1991), rastgele orman (Breiman, 1999) ve lojistik regresyon (Pampel, 2000) yöntemleri kullanılmıştır.

3.6.1 Yapay sinir ağları (YSA)

İlk olarak 1943 yılında McCulloch ve Pitts (McCulloch ve Pitts, 1943) tarafından ortaya atılan YSA biyolojik sinir sisteminden esinlenilerek oluşturulmuş bir makine öğrenimi algoritmasıdır. Şekil 3.14 ve Çizelge 3.5'te biyolojik sinir sistemi ile YSA'nın karşılaştırılması gösterilmiştir.

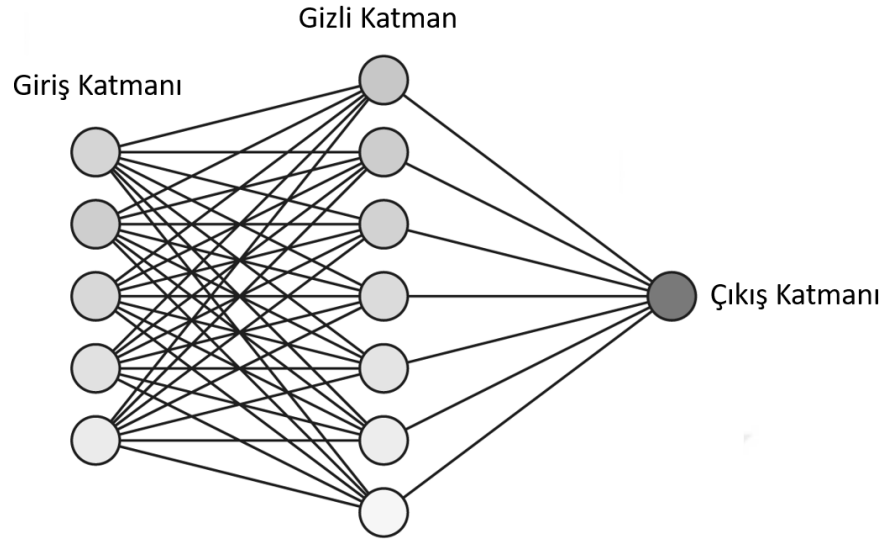


Şekil 3.14. Biyolojik sinir ağları ile YSA'nın karşılaştırılması

Çizelge 3.5. Biyolojik sinir ağları ile YSA'nın karşılaştırılması

Biyolojik Sinir Sistemi	Yapay Sinir Sistemi
Nöron	İşlemci eleman
Dentrit	Toplama fonksiyonu
Hücre gövdesi	Transfer fonksiyonu
Aksonlar	Yapay nöron çıkışı
Sinapslar	Ağırlıklar

YSA giriş katmanı, gizli katman ve çıkış katmanı olmak üzere üç ana bölümden meydana gelir. Genel bir YSA yapısı Şekil 3.15'te gösterilmiştir.



Şekil 3.15. Genel bir YSA yapısı

Sınıflandırılması istenen veri öncelikle giriş katmanına verilir. Burada sınıflandırılacak verinin özellik sayısı ile giriş katmanında bulunan nöron sayısının eşit olması hususuna dikkat edilmelidir. Giriş katmanının ardından özellikler gizli katmana geçerek ağırlıklar oluşmaya başlar. Gizli katmandan geçen özellikler en son çıkış katmanına aktarılır. Burada istenilen sınıfın sayısal değeri ile elde edilen değer arasındaki fark hesaplanır ve ağırlıklar geri yayılım ile geriye doğru güncellenir (Ceylan, 2004). Geriye yayılım algoritması Bölüm 3.7.1.5.1'de açıklanmıştır. Geri yayılım işlemi istenilen sonuca ulaşılan kadar devam eder.

3.6.2 Lojistik regresyon (LR)

Regresyon, veri kümesi içerisindeki rastgele gibi gözükken bir ilişkiyi matematiksel bir model aracılığı ile tanımlamaya çalışan bir makine öğrenimi modelidir (Myers ve Myers, 1990). Oluşturulan matematiksel ifade tek bağımsız değişkenli veya çok bağımsız değişkenli şekilde olabilmektedir. Tek bağımsız değişkenli regresyon basit doğrusal regresyon olarak adlandırılır ve aşağıdaki gibi tanımlanır:

$$RÇ = \beta_0 + \beta_1 X + \varepsilon \quad (3.11)$$

Burada X giriş değeri, $RÇ$ çıkış değeri, β_0 $X = 0$ iken bulunan sabit değer, β_1 regresyon katsayısını ve ε hata değerini ifade etmektedir.

Doğrusal regresyonda bağımlı ve bağımsız değişkenler nicem değerlere sahip olmalı ve normal dağılım göstermelidirler. Ancak lojistik regresyonda bu durumlar aranmamaktadır (Kleinbaum ve ark., 2002). Lojistik regresyonda amaç veri kümesini ayırabilecek en sade ifadenin elde edilmesidir. Bunun gerçekleşmesi için *OLASILIK* (Eşitlik 3.12) ve *LOGIT* (Eşitlik 3.13) değerlerinin hesaplanması gerekir.

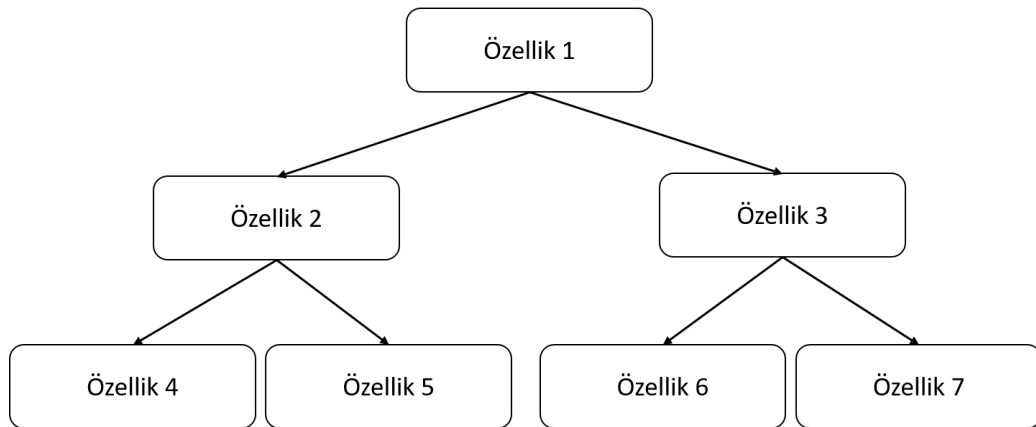
$$OLASILIK = \frac{p}{1-p} \quad (3.12)$$

$$LOGIT = \ln\left(\frac{p}{1-p}\right) \quad (3.13)$$

Burada p değeri bir özelliğin görülme olasılığını ifade etmektedir. Yani *OLASILIK* değeri bir özelliğin görülme olasılığının görülmeme olasılığına oranı olarak ifade edilebilir. *LOGIT* değeri ise olasılık değerinin doğal logaritmasıdır ve doğrusal regresyon analizindeki β_1 katsayısına karşılık gelmektedir.

3.6.3 Karar ağaçları (KA) ve rastgele orman (RO)

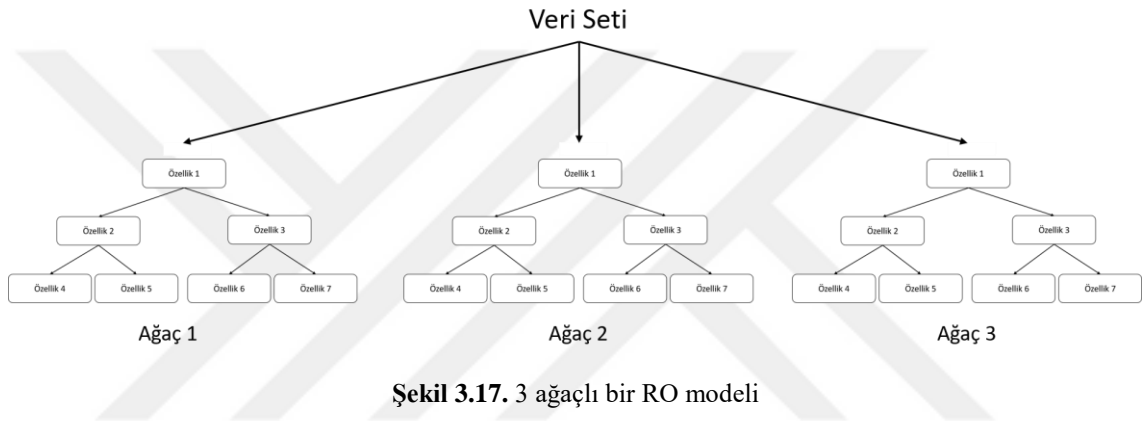
Karar ağaçları, kural tabanlı bir sınıflandırma modeli olarak diğer modellere nazaran karar verme sürecinin net olarak ortaya koyulduğu bir makine öğrenimi modelidir (Dietterich, 2000). Genel bir karar ağacı yapısı Şekil 3.16'da gösterilmektedir.



Şekil 3.16. Genel bir karar ağacı yapısı

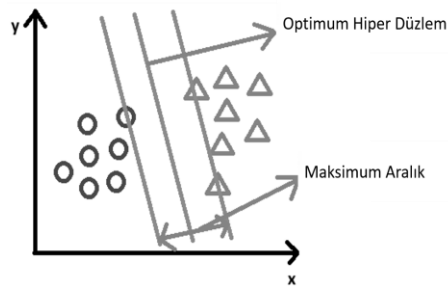
Ağacın dallanmaya başlamasında hangi özelliğin kullanılmaya başlanacağı bilgi kazancı hesaplanarak elde edilmektedir. Tüm özelliklerin bilgi kazançları hesaplandıktan sonra özellikler büyükten küçüğe doğru sıralanırlar ve en yüksek değerli özellik ile karar ağacının dallanması başlar.

Rastgele orman algoritması birden fazla karar ağacı kullanılarak sınıflandırma performansının yükselmesini amaçlayan makine öğrenimi modelidir (Breiman, 1999). Bu modele göre veri seti rastgele olarak K alt parçaya bölünür ve her bir parça için ayrı bir karar ağacı modeli oluşturulur. İşlem sonucunda K adet sonuç toplanarak sınıflandırma tamamlanır. Şekil 3.17’de 3 karar ağacı ile oluşturulmuş bir RO modeli gösterilmiştir.



3.6.4 Destek vektör makineleri (DVM)

Destek vektör makineleri sınıflandırma işlemi için en sık kullanılan makine öğrenimi algoritmalarından birisidir. Şekil 3.18’de görüldüğü gibi DVM bir optimum hiper düzlem oluşturarak sınıflar arasındaki maksimum aralığı elde etmeye çalışır (Cortes ve Vapnik, 1995). Hata oranı düşük bir sınıflandırma gerçekleştirmek için optimum hiper düzlemin şeklini belirleyen doğrusal (Eşitlik 3.14), polinomik (Eşitlik 3.15) ve radyal (Eşitlik 3.16) gibi fonksiyonlar kullanılmaktadır.

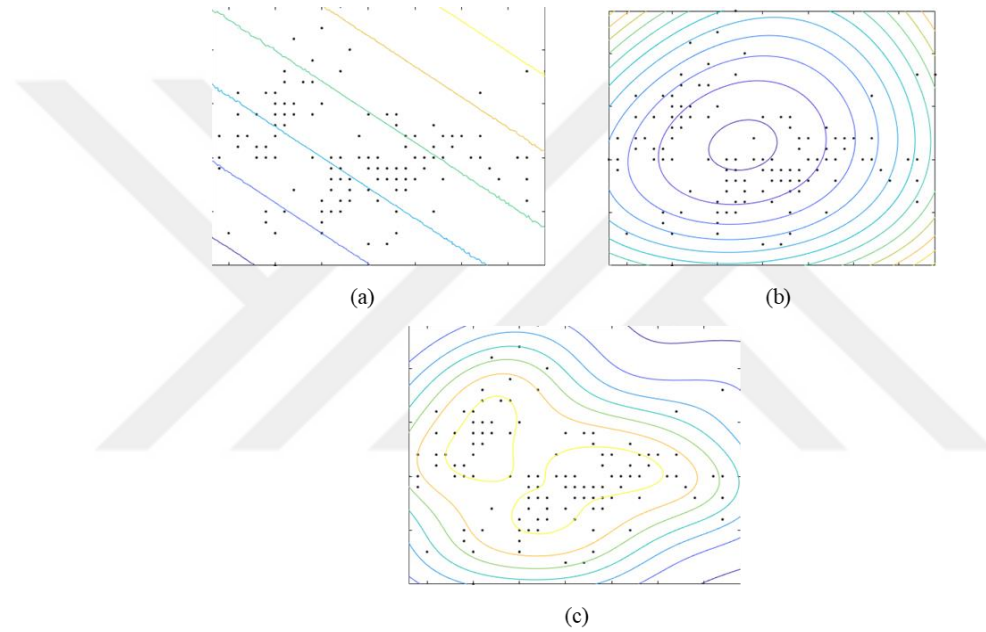


$$do(X,Y) = (X \bullet Y) \quad (3.14)$$

$$po(X,Y) = (\gamma^* (X \bullet Y) + C)^d \quad (3.15)$$

$$rad(X,Y) = -\gamma^* \|X - Y\|^d \quad (3.16)$$

Burada $(X \bullet Y)$ X ve Y arasındaki nokta çarpım, γ fonksiyonun ayarlanabilir parametresi, d polinomun derecesini, C kaydırma parametresini ve $\|X - Y\|$ X ile Y arasındaki mesafeyi ifade etmektedir. Aynı verinin doğrusal, polinomik ve radyal fonksiyonlar ile sınıflandırılmasına ait görüntüler Şekil 3.19'da verilmiştir.



Şekil 3.19. Aynı verinin farklı DVM fonksiyonları ile ayrılması temsili (a) doğrusal fonksiyon kullanıldığında (b) polinomik fonksiyon kullanıldığında (c) radyal fonksiyon kullanıldığında

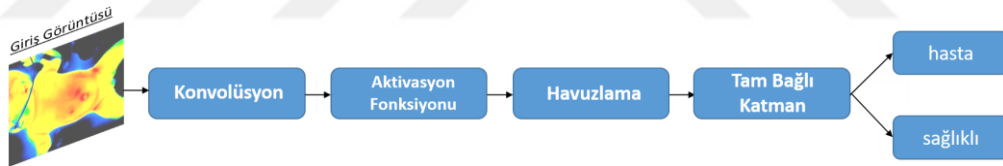
3.7 Derin Öğrenme

Derin öğrenme, öğrenme işleminin gerçekleşmesi için çoklu katmanlardan meydana gelen modellerin oluşturulmasıdır (Krizhevsky ve ark., 2012). Örneğin YSA'nın daha fazla sayıda katman ve daha fazla sayıda nöron içeren hali bir derin öğrenme modelidir ve çok katmanlı algılayıcı olarak tanımlanmaktadır. Çok katmanlı algılayıcılar, evrişimli sinir ağları, otomatik kodlayıcılar ve çekişmeli üretici ağlar gibi derin öğrenme modelleri bulunmaktadır. Bu yöntemler görüntü sınıflandırma, konuşma tanıma, görsel veri tanıma, nesne algılama ve ilaç keşfi gibi birçok alanda başarı performansını önemli ölçüde geliştirmiştir.

3.7.1 Evrişimli sinir ağıları (ESA)

Görüntü sınıflandırma, giriş olarak alınan bir görüntünün hangi sınıfa ait olacağını olasılıksal olarak belirleme işlemidir (Deshpande, 2016). İnsan, doğduğu andan itibaren çevresini izleme, nesnelere öğrenme, farklı ortamlarda analizler yapma ve önceki gördükleri ile karşılaştırarak yorum yapma kapasitesine sahiptir. İnsanlar için anlık olarak gerçekleşen bu işlemler makineler tarafından uzun süren eğitim işlemleri ve yüksek güç gerektiren işlemciler tarafından gerçekleştirilir (Strigl ve ark., 2010).

ESA konvolüsyon, aktivasyon fonksiyonu, havuzlama ve tam bağı katman gibi katmanlardan meydana gelen bir derin öğrenme modelidir. Art arda gelen konvolüsyon katmanları görüntüde bulunan kenar, köşe ve eğrilik gibi düşük seviyeli özelliklerden doku gibi yüksek seviyeli özelliklere doğru özelliklerin elde edilmesini sağlar. Bir resme bakıldığında uzakta duran bir cismin iki kolu ve iki bacağı fark edilirse yüksek ihtimalle bir insan olduğu düşünülür. ESA'da da benzer biçimde konvolüsyon ile görüntü içerisinde belirli dokuların varlığı aranır ve sınıflandırma işlemi gerçekleştirilir. Genel bir ESA yapısı Şekil 3.20'de verilmiştir.



Şekil 3.20. Basit Bir ESA Yapısı

3.7.1.1 Konvolüsyon

Giriş görüntüsü ardından gelen ilk ESA katmanı konvolüsyon katmanıdır. Konvolüsyon katmanında giriş görüntüsü üzerinde çeşitli filtreler gezdirilerek özellik haritalarının oluşturulması sağlanır (Krizhevsky ve ark., 2012). Bu filtreler genellikle 3x3, 5x5 ve 7x7 boyutlarında olmaktadır. Şekil 3.21'de çeşitli filtreler gösterilmiştir.

-1	-2	3
1	-4	5
-1	-1	2

(a)

-3	1	2	1	-4
-1	2	-3	-1	-1
-2	3	-2	1	-2
-4	5	-1	2	-3
-1	2	4	1	-1

(b)

1	-2	2	1	-1	-2	3
2	-3	1	2	1	-4	5
1	-1	2	-3	-1	-1	2
-1	-2	3	-2	1	-2	2
1	-4	5	-1	2	-3	1
-1	-1	2	4	1	-1	2
-1	2	2	-3	1	-1	2

(c)

Şekil 3.21. Çeşitli boyutlarda filtreler (a) 3x3 boyutunda filtre (b) 5x5 boyutunda filtre (c) 7x7 boyutunda filtre

Konvolüsyon işlemi matematiksel olarak aşağıdaki gibidir:

$$f(x, y) * h(x, y) = \sum_{n1=-\infty}^{\infty} \sum_{n2=-\infty}^{\infty} f(n1, n2) h(x - n1, y - n2) \quad (3.17)$$

Eşitlik 3.17'nin görüntülere uygulanması ise görüntünün sol üst köşesine bir filtre yerleştirilerek başlar ve filtrenin eleman değerleri ile görüntü üzerinde denk gelen eleman değerleri çarpılarak sonuçlar toplanır. Böylece elde edilen değer, özellik haritasının ilk elemanı olarak yerine yazılır. Şekil 3.22'de 3x3'lük bir filtre için bu durum gösterilmiştir. İkinci aşamada Şekil 3.23'de gösterildiği gibi filtre bir yan sütuna kaydırılır ve denk gelen elemanların çarpılarak toplama işlemi yeniden gerçekleştirilir. Bu işlem, tüm görüntü taranincaya kadar devam eder. Burada en son elde edilen matris *aktivasyon haritası* veya *özellik haritası* olarak adlandırılır. Ayrıca filtrenin görüntü üzerinde bulunduğu alan *algılayıcı alan* olarak tanımlanır.

123	224	190	123	224	190	123
120	145	191	120	145	191	120
110	107	189	110	107	189	110
204	166	183	204	166	183	204
154	240	185	154	240	185	154
142	121	187	142	121	187	142
140	120	181	140	120	181	140

655				

Şekil 3.22. Konvolüsyon işleminde birinci elemanın elde edilmesi

123	224 ⁻¹	190 ⁻²	123 ³	224	190	123
120	145 ¹	191 ⁻⁴	120 ⁵	145	191	120
110	107 ⁻¹	189 ⁻¹	110 ²	107	189	110
204	166	183	204	166	183	204
154	240	185	154	240	185	154
142	121	187	142	121	187	142
140	120	181	140	120	181	140

655	-330			

Şekil 3.23. Konvolüsyon işleminde ikinci elemanın elde edilmesi

Çıkış olarak elde edilen özellik haritasının boyutu Eşitlik 3.18'teki gibi hesaplanır:

$$n_c = \frac{n_{giriş} + 2e_k - fi}{k} + 1 \quad (3.18)$$

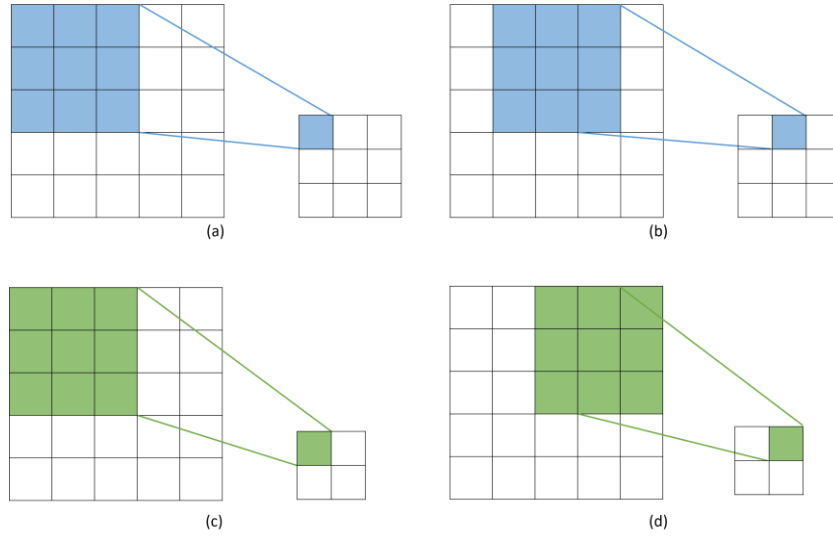
Burada $n_{giriş}$ giriş görüntüsünün boyutunu, e_k resmin çevresine eklenecek sıfır katmanının sayısını, fi filtrenin boyutunu, k filtrenin her bir seferde kaç adım ilerlediği ve $n_{çıkış}$ ise çıkış boyutunu ifade etmektedir. Kaydırma ve sıfır ekleme işlemleri alt bölümde açıklanmıştır. Şekil 3.22'de verilen örnek için $n_{giriş} = 7$, $e_k = 0$, $f = 3$ ve $k = 1$ 'dir. Buna göre çıkış boyutu 5 olarak elde edilir.

Aslında her bir filtre bir özellik tanımlayıcısı olarak görev almaktadır. Örneğin bir filtre kenar, bir filtre eğri ve bir filtre renk arayıcısı olarak çalışabilir. Filtre katsayıları ile algılayıcı alan katsayıları aynı olduğu durumlarda konvolüsyon işleminin sonucu yüksek değerler üreterek aranan özelliğin bulunduğunu işaret eder.

ESA'ya eklenecek ikinci konvolüsyon katmanı artık giriş görüntüsüne değil elde edilen özellik haritasına uygulanacaktır. Burada konvolüsyon katmanları arasına yerleştirilmiş aktivasyon fonksiyonu ve havuzlama gibi katmanlar bulunmaktadır. Bu katmanlar modeli doğrusal olmaktan kurtararak aşırı öğrenme durumuna karşı bir önlem oluşturmaktadırlar.

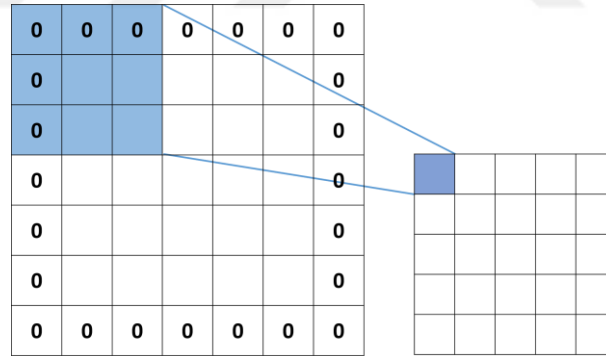
3.7.1.1.1 Kaydırma ve sıfır ekleme

Kaydırma parametresi (k) konvolüsyon filtresinin her bir aşamada kaç birim kaydırılacağını ifade eden parametredir. Şekil 3.24'te k değerinin 1 ve 2 olduğu durumlar için oluşan özellik matrisi temsilleri gösterilmiştir.



Şekil 3.24. k değeri 1 ve 2 iken oluşan aktivasyon haritaları temsili (a) k değeri 1 iken oluşan ilk eleman temsili (b) k değeri 1 iken oluşan ikinci eleman temsili (c) k değeri 2 iken oluşan ilk eleman temsili (d) k değeri 2 iken oluşan ikinci eleman temsili.

Şekil 3.24'ten görüldüğü gibi konvolüsyon işleminin ardından oluşan özellik haritasının boyutu Eşitlik 3.15'e göre orijinal görüntünün boyutundan daha düşük olmaktadır. Boyut azalmasının istenmediği durumlarda orijinal görüntünün etrafına Şekil 3.25'da görüldüğü gibi *sıfır ekleme* gerçekleştirilmektedir.



Şekil 3.25. Sıfır ekleme uygulandıktan sonra özellik haritasının oluşumu

3.7.1.2 Aktivasyon fonksiyonları

Aktivasyon fonksiyonları gelen giriş değerlerini belirli bir aralığa getiren veya giriş değerlerinin bir kısmının alınıp bir kısmının kesilmesini sağlayan fonksiyonlardır. Literatürde en sık karşılaşılan aktivasyon fonksiyonları sigmoid (Eşitlik 3.19), tanh (Eşitlik 3.20) ve ReLU (Eşitlik 3.21) aktivasyon fonksiyonlarıdır.

$$s = \frac{1}{1 + e^{-z}} \quad (3.19)$$

$$t = \frac{1 - e^{-z}}{1 + e^{-z}} \quad (3.20)$$

$$r = \max(0, z) \quad (3.21)$$

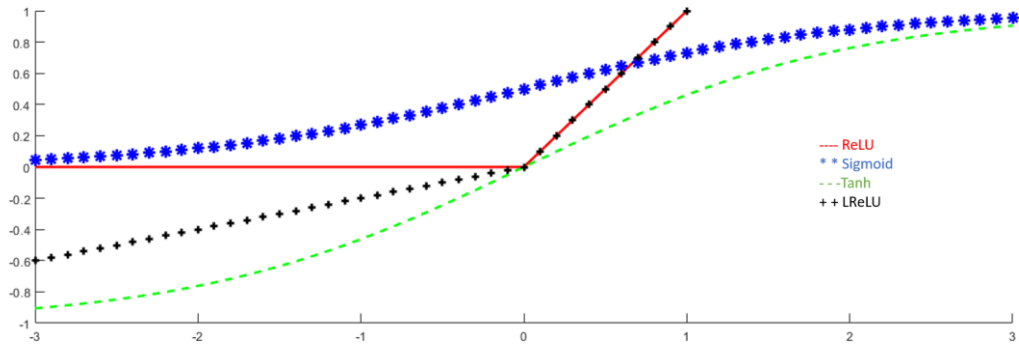
Burada z giriş değeri ve sırası ile s , t ve r sigmoid, tanh ve relu aktivasyon fonksiyonlarının çıkış değeridir.

Şekil 3.26'da görüldüğü gibi sigmoid aktivasyon fonksiyonu giriş değerlerini 0 ile 1 arasına; tanh aktivasyon fonksiyonu ise -1 ile 1 arasına getirir. Negatif değerleri elimine ederek yalnızca pozitif değerlerin geçmesini sağlayan aktivasyon fonksiyonu ise ReLU aktivasyon fonksiyonudur.

ReLU aktivasyon fonksiyonunun bir farklı versiyonu LReLU (Eşitlik 3.22) aktivasyon fonksiyonudur (Lee, 2017). LReLU giriş değeri ile giriş değerinin bir katsayı ile çarpımlarını karşılaştırmakta ve büyük olan değeri geçirmektedir. Böylece pozitif değerler kendi değerleri ile fonksiyondan geçiş yaparken negatif değerlerin geçişine katsayı ile çarpıldıktan sonra karar verilir. Bu katsayı *sızıntı miktarı* olarak adlandırılmaktadır.

$$LReLU = \max(z, z * st) \quad (3.22)$$

Burada z giriş değerini ve st sızıntı miktarını ifade etmektedir.



Şekil 3.26. Aktivasyon fonksiyonları çıkış değerleri. Şekilde LReLU için sızıntı miktarı 0.2 seçilmiştir.

3.7.1.3 Havuzlama katmanı

Havuzlama katmanı aşağı örnekleme olarak da adlandırılan ve özellik haritalarının boyutunun azaltılmasını sağlayan (Bölüm 3.5'te açıklanan özellik seçme gibi de düşünülebilir) ESA katmanıdır (Krizhevsky ve ark., 2012). Temel olarak 2x2, 3x3 ve 5x5 gibi bir filtre özellik haritası üzerinde dolaştırılmakta ve her seferinde filtre ile denk gelen elemanların ortalama veya maksimum değeri alınmaktadır. Havuzlama boyutu büyüdükçe, meydana gelecek özellik haritasının boyutu küçüleceğinden havuzlama boyutunun dikkatli bir şekilde seçilmesi gerekmektedir. Literatürde genellikle 2x2 havuzlama ile karşılaşmaktadır. Şekil 3.27'de 2x2 havuzlama işlemi gösterilmiştir.

189	110	107	189
185	204	166	186
185	154	240	185
187	142	124	187

(a)

204	189
187	240

(b)

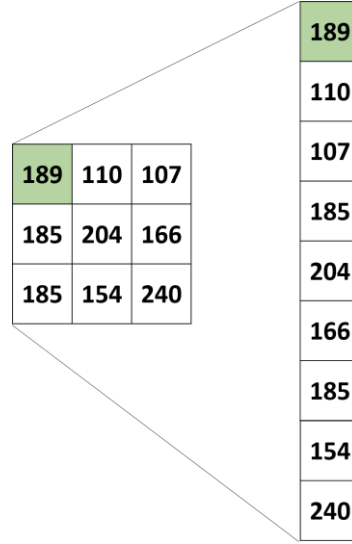
172	162
167	184

(c)

Şekil 3.27. Maksimum ve ortalama havuzlama gösterimi (a) orijinal görüntü (b) maksimum havuzlama (c) ortalama havuzlama

3.7.1.4 Düzleştirme katmanı

Konvolüsyon ve havuzlama işlemlerinin ardından elde edilen özellik haritaları matris şeklindedirler. Sınıflandırma işleminin gerçekleştirilmesi için iki boyutlu olan bu matrislerin tek boyutlu vektöre çevrilmesi gerekmektedir (Krizhevsky ve ark., 2012). Matris formundan vektör formuna çevirme işlemi *düzleştirme* olarak adlandırılır (Şekil 3.28).



Şekil 3.28. Düzleştirme işlemi

3.7.1.5 Tam bağlı katman

Tam bağlı katmana kadar olan, konvolüsyon, aktivasyon ve havuzlama işlemleri görüntüden özellik çıkarma, modeli doğrusal olmayan hale getirme ve özellik azaltma kısımlarını oluşturmaktadır. Tam bağlı katman (Krizhevsky ve ark., 2012) elde edilen özellik haritalarının sınıflandırılmasını gerçekleştiren katmandır ve çalışma yapısı YSA ile aynıdır.

3.7.1.5.1 Geri yayılım algoritması

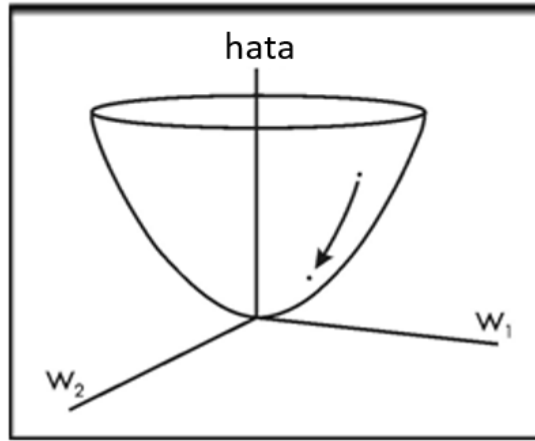
Model eğitime başlandığı anda hata değeri yüksek seviyede olmaktadır. Bunun nedeni modelde bulunan ağırlık değerlerinin ilk aşamada rastgele olarak atanmasıdır. Geri yayılım, hata değerini en aza indirmek için gerçekleştirilen ve dört aşamadan meydana gelen işlemidir (Deshpande, 2016). Bu işlemler *ileri gidiş*, *hata hesaplama*, *geri gidiş* ve *ağırlık güncelleme* işlemleridir.

İleri gidiş modelin başından sonuna kadar bir kere tamamlanması işlemidir. Yani giriş görüntüsü gelir, özellikler çıkartılır, sınıflandırma gerçekleştirilir ve bir sonuç elde edilir.

Hata hesaplamada elde edilen sonuç değeri ile hedeflenen sonuç değeri arasındaki hata miktarı hesaplanır. Bu hata değerini hesaplayan fonksiyonlara *hata fonksiyonu* denir. En sık kullanılan hata fonksiyonu ortalama karesel hata fonksiyonudur ve aşağıdaki şekilde hesaplanır.

$$Ha_{toplam} = \frac{1}{n} \sum (hedef - \text{çikiş})^2 \quad (3.23)$$

Burada Ha hata değerini ve n eleman sayısını ifade etmektedir. Hata değeri hesaplandıktan sonra hataya neden olan ağırlık değerlerinin güncellenmesi gerekmektedir. Şekil 3.29'da iki ağırlık değerine sahip bir problemin üç boyutlu hata gösterimi verilmiştir.



Şekil 3.29. İki ağırlık değerine sahip bir problemin üç boyutlu hata gösterimi (Deshpande, 2016)

Şekil 3.29'da görüldüğü gibi hata değeri yüksek bir noktadan sıfır noktasına doğru azaltılmaya çalışılmaktadır. Bunun gerçekleştirilmesi için hatanın ağırlığa göre türevinin alınması gerekmektedir. *Geri gidiş* boyunca hataya neden olan ağırlıklar ile hata değeri arasında türev hesabı yapılarak *ağırlık güncelleme* (Eşitlik 3.24) işlemi gerçekleştirilmektedir.

$$\omega = \omega_i - \eta x \frac{dHa}{dW} \quad (3.24)$$

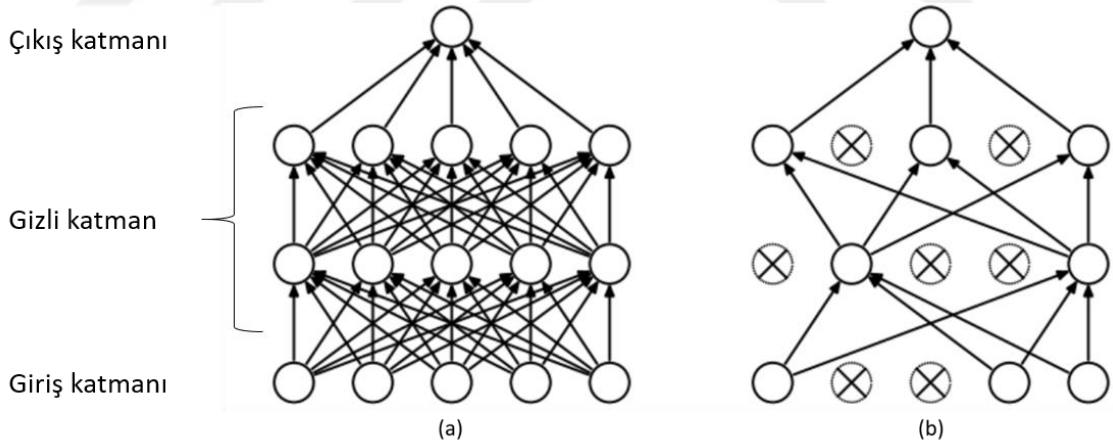
Burada ω_i mevcut ağırlık, η öğrenme oranı, H hata ve ω yeni ağırlık değerini ifade etmektedir. Öğrenme oranı deneysel olarak seçilen bir parametredir ve her bir ağırlık güncelleme işleminde gerçekleşecek olan adım büyüklüğünü ifade eder. Öğrenme oranı yüksek seçildiği durumda bulunulan noktadan istenilen noktaya giderken ulaşılmama problemi ile karşılaşılabilir. Örneğin 0.5 noktasından 1 noktasına giderken öğrenme oranı 1 olarak seçilir ise bir sonraki adımda 1.5 noktasına

gidilecektir. Öğrenme oranı çok düşük seçildiği durumda istenilen noktaya varmak uzun süre alacaktır. Örneğin öğrenme oranı 0.001 seçilirse 0.5 noktasından 1 noktasına 0.501, 0.502 ... 0.999 şeklinde 500 adımda gidilecektir. Bu örnek için öğrenme oranı 0.1 seçildiğinde istenilen noktaya ulaşılması daha az zaman ve işlem yükü ile mümkün olacaktır.

İleri gidiş, hata hesaplama, geri dönüş ve ağırlık güncelleme işlemlerinin bir kere tekrarlanması eğitim işleminin bir iterasyon gerçekleştirdiğini ifade etmektedir. Eğitim işlemi ya belirli bir hata değerinin altına düşene kadar ya da belirlenecek bir iterasyon sayısına kadar devam ettirilir.

3.7.1.6 Unutturma katmanı

Unutturma katmanı sinir ağları için kullanılan özel bir katmandır. Özellikle aşırı öğrenme probleminin üstesinden gelmek için kullanılmaktadır (Srivastava ve ark., 2014). Eğitim işlemi sırasında rastgele olarak nöronlar arasındaki ağırlık değerlerinin sıfıra eşitlenmesi ile gerçekleştirilir. Böylece tüm ağırlık değerleri sürekli olarak en uygun değere gelmeye çalışır. Unutturma işlemi Şekil 3.30'da gösterilmiştir.



Şekil 3.30. Unutturma işlemi (a) genel bir sinir ağı yapısı (b) unutturma işleminin ardından sinir ağı yapısı (Srivastava ve ark., 2014)

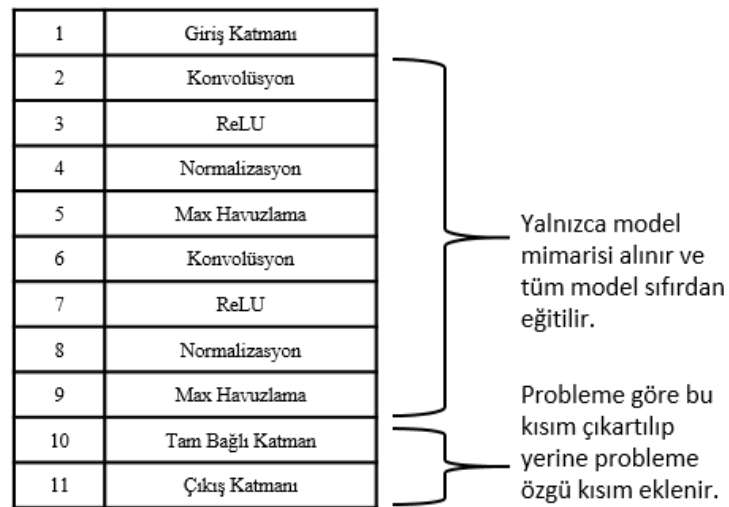
Şekil 3.30 (a)'da görüldüğü gibi, bir sinir ağında tüm nöronlar ağırlıklar ile birbirlerine bağlı halde bulunmaktadır. Unutturma işlemi, Şekil 3.31 (b)'de görüldüğü gibi bazı nöronlara ağırlık girişi ve ağırlık çıkışı yapılmayarak gerçekleştirilir.

3.7.2 Transfer öğrenme

Sıfırdan etkili bir model oluşturabilmek için fazla sayıda görüntüye ve yüksek işlemci gücüne sahip makinelere ihtiyaç duyulmaktadır. Transfer öğrenme daha önceden milyonlarca resim ile eğitilmiş modellerin alınarak elde bulunan az sayıda görüntünün sınıflandırılması için kullanımıdır (Pan ve Yang, 2010). AlexNet, VGGNet ve GoogleNet gibi yüzlerce farklı sınıfa ait binlerce görüntü ile eğitilmiş hazır modeller bulunmaktadır.

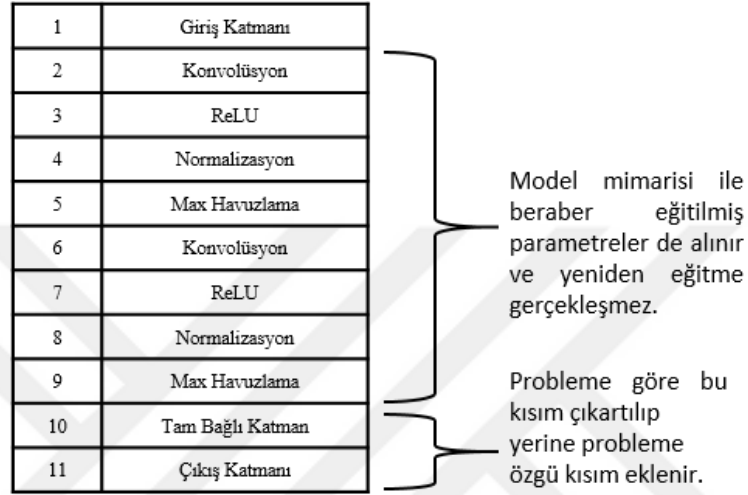
Eğitilmiş modeller incelendiği zaman ilk konvolüsyon katmanlarında kenar, köşe ve eğri gibi düşük seviyeli özelliklerin bulunduğu görülmektedir. Sıfırdan bir model oluşturulduğunda ise ilk ağırlıklar rastgele olarak atanmakta ve eğitim işlemi süresince bu ağırlıklar güncellenmektedir. Sıfırdan ağırlık güncellemek yerine önceden eğitilmiş modellerin ilk katmanlarında bulunan ağırlıkları direkt olarak almak özellik çıkarma işleminin yapılmasına olanak sağlamaktadır. Ardından elde bulunan veriye özgü yeni konvolüsyon ve sınıflandırma katmanları eklenerek eğitim işlemi gerçekleştirilmiş olur.

Transfer öğrenme metotları *skreç*, *ön-eğitilmiş* ve *ince-ayarlı* olmak üzere üç farklı yöntem şeklinde uygulanabilmektedir. Skreç yönteminde herhangi bir eğitilmiş parametre kullanılmadan yalnızca hazır modellerin mimari kısımları alınmaktadır. Eğitim işlemi sıfırdan rastgele atanmış parametreler ile başlar. Şekil 3.31’de skreç transfer öğrenme işlemi gösterilmiştir.



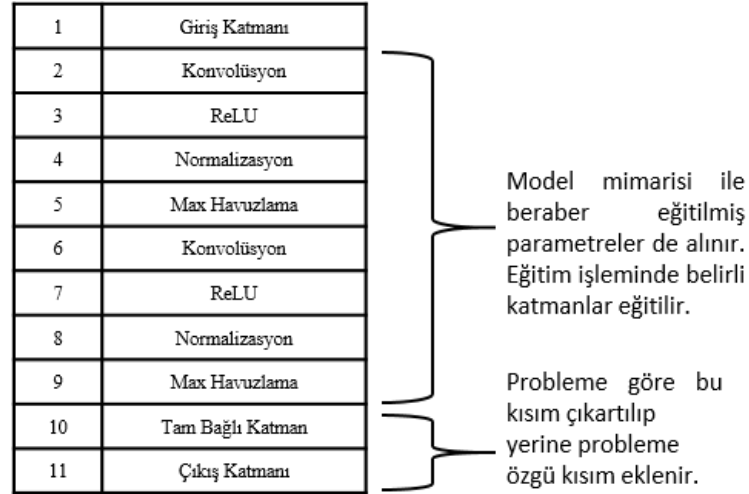
Şekil 3.31. Skreç transfer öğrenme

Ön-egitimli yönteminde hazır modelin özellik çıkarma işlemlerinin yapıldığı bölümlerdeki (konvolüsyon, aktivasyon ve havuzlama) eğitilmiş parametreler alınır ve eğitilmiş tam bağlı katman çıkartılarak kendi problemine özgü tam bağlı katman kısmı eklenir. Yani özellik çıkarma işlemi hazır modele yaptırılır, yalnızca sınıflandırıcı yapı kullanıcı tarafından eğitilir. Şekil 3.32’de ön-egitimli transfer öğrenme işlemi gösterilmiştir.



Şekil 3.32. Ön-egitimli transfer öğrenme

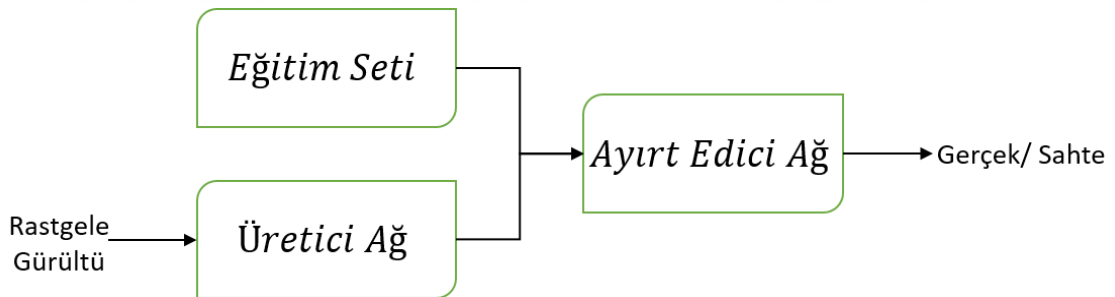
İnce-ayarlı yönteminde ön-egitimli yönteminde olduğu gibi özellik çıkarma bölümleri alınır ve eğitilmiş tam bağlı katman çıkartılarak kendi problemine özgü tam bağlı katman kısmı eklenir. Buradaki fark eğitim işlemi boyunca sınıflayıcı yapı ile beraber istenilen özellik katmanlarının da eğitim işlemine katılmasıdır. Yani eğitim işlemi tüm modele uygulanır ancak bazı katmanların eğitilme özellikleri durdurulur ve buralardaki parametreler değişmez, yalnızca eğitilebilir durumda olan özellikler ile sınıflayıcı yapı eğitilir. Şekil 3.33’te ince-ayarlı transfer öğrenme işlemi gösterilmiştir.



Şekil 3.33. İnce-ayarlı transfer öğrenme

3.7.3 Çekişmeli üretici ağlar (ÇÜA)

Çekişmeli üretici ağlar 2014 yılında Goodfellow ve arkadaşları tarafından ortaya atılan görüntü artırma, gürültü giderme, süper çözünürlüklü görüntü elde etme ve yazıdan görüntü üretme gibi işlemleri yerine getiren bir derin öğrenme modelidir (Goodfellow ve ark., 2014). Şekil 3.34'te bir ÇÜA modeli gösterilmiştir.

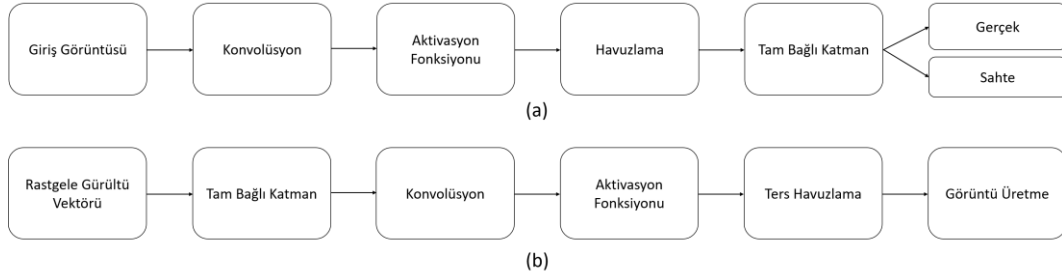


Şekil 3.34. Genel bir ÇÜA yapısı

ÇÜA modeli temel olarak bir *üretici ağ* ve bir *ayırt edici ağ* olmak üzere, aynı anda eğitilen iki ağın birleşiminden meydana gelir. Üretici ağın görevi, elde bulunan görüntülerin dağılımlarını öğrenerek bu dağılımlara göre yeni görüntüler üretmek; ayırt edici ağın görevi ise kendisine giriş olarak gelen görüntülerin gerçek görüntüler mi yoksa üretilmiş (sahte) görüntüler mi olduğunun olasılıksal değerini tayin etmektir.

Şekil 3.35 (a)'da görüldüğü gibi, ayırt edici ağ iki sınıflı (gerçek (1) - sahte (0)) bir ESA modelidir. Üretici ağ ise Şekil 3.35 (b)'de görüldüğü gibi bir ters ESA modelidir. Üretici ağın bir görüntü üretebilmesi için tam bağlı katmanın giriş kısmında

bulunan nöron sayısı boyutunda giriş vektörü alması gerekmektedir. Bu giriş vektörü rastgele sayılardan meydana gelir ve *rastgele gürültü vektörü* olarak adlandırılır. Tam bağlı katmandan geçen değerler, konvolüsyon katmanlarında bir görüntü meydana getirir.



Şekil 3.35. Üretici ve ayırt edici ağlara ait temel mimariler (a) ayırt edici ağ mimarisi (b) ayırt edici ağ mimarisi

ÇÜA'nın çalışması:

- Üretici ağ görüntü üretmek için rastgele gürültü vektörü alır.
- Üretici ağ bir görüntü üretir.
- Veri setinden orijinal bir görüntü alınır.
- Üretilen görüntünün sınıfı "0" orijinal görüntünün etiketi "1" olarak ayırt edici ağa verilir ve bu sınıf değerlerine göre geri yayılım gerçekleştirilerek **ayırt edici ağın eğitimi** yapılır.
- Üretici ağ rastgele bir gürültü vektörü daha alır ve bir görüntü daha üretir.
- Üretilen bu görüntünün sınıfı bu kez "1" olarak ayırt edici ağa verilir ve bu sınıf değerine göre geri yayılım yapılarak **üretici ağın eğitimi** yapılır.

Yukarıdaki işlemler ayırt edici ağın üretilen görüntüleri 1'e çok yakın sınıflandırdığı durumda yani üretilen görüntüleri gerçek olarak algılamaya başladığında sona erer.

3.8 Termal Görüntüler için Veri Artırımı

Derin öğrenme görüntü sınıflandırmada iyi sonuçlar üretmesine karşın eğitim işlemi için fazla sayıda görüntüye ihtiyaç duymaktadır. Medikal alanda gerek hastane

şartları gerek görüntülenen durumu her zaman elverişli olmadığından binlerce farklı canlı içeren veri setleri elde edilememektedir. Elde bulunan verilerin makineler vasıtasıyla artırılması *veri artımı* olarak adlandırılır (Mikołajczyk ve Grochowski, 2018).

Veri artırma yöntemleri geleneksel (Rotasyon değiştirme, aynalama, yakınlaştırma, 3B döndürme, histogram eşitleme, keskinleştirme, bulanıklaştırma, aydınlık ve karışıklık değiştirme) ve gelişmiş (doku ve stil transferi yapma) yöntemler olmak üzere iki gruba ayrılabilir. Geleneksel yöntemler kolay uygulanabilir, işlem ve zaman maliyetleri düşük yöntemlerken; doku ve stil transferi yapma gibi gelişmiş yöntemler işlem ve zaman maliyetleri yüksek yöntemlerdir.

3.9 Sınıflandırma Sonuçlarının Değerlendirilmesinde Kullanılan Kriterler

Tez kapsamında yapılan çalışmalardan elde edilen sonuçların değerlendirilmesi için tepe sinyal gürültü oranı, yapısal benzerlik indeksi, karmaşıklık matrisi, özgüllük, doğruluk, hassasiyet, alıcı işlem karakteristiği ve eğri altında kalan alan değerleri hesaplanmıştır (Zhu ve ark., 2010).

3.9.1 Tepe sinyal gürültü oranı (TSGO)

TSGO iki görüntü arasından meydana gelen bilgi bozulmalarının desibel cinsinden ölçüldüğü değerlendirme kriteridir (Huynh-Thu ve Ghanbari, 2008) ve aşağıdaki gibi hesaplanır:

$$TSGO = 10 \log_{10} \left(\frac{I_{max}}{OKH} \right) \quad (3.25)$$

Burada I_{max} görüntü içerisinde bulunan en yüksek değerli piksel seviyesini ifade etmektedir ve genellikle 255 olarak kullanılmaktadır. OKH, ortalama karesel hata değerini ifade eder. Tek boyutta OKH değeri Eşitlik 3.26'teki gibi hesaplanırken görüntüler için yani iki boyutta OKH değeri aşağıdaki şekilde hesaplanır.

$$OKH = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (Y_{i,j} - S_{i,j})^2 \quad (3.26)$$

Burada M ve N görüntülerin satır ve sütun sayılarını ifade ederken Y ve S karşılaştırılan görüntüleri ifade etmektedir.

3.9.2 Yapısal benzerlik indeksi (YBİ)

Yapısal benzerlik indeksi iki görüntü arasındaki kalite değerlendirme kriteri olarak parlaklık (pa), kontrast (ko) ve yapısal (ya) değerlerin hesaplanmasına dayanmaktadır (Wang ve ark., 2003). Bu üç değer çarpılarak YBİ (3.27) elde edilir.

$$YBI = pa(x, y)^\alpha xko(x, y)^\beta xy a(x, y)^\gamma \quad (3.27)$$

$$pa(x, y) = \frac{2\mu_x\mu_y + C1}{\mu_x^2 + \mu_y^2 + C1} \quad (3.28)$$

$$ko(x, y) = \frac{2\sigma_x\sigma_y + C2}{\sigma_x^2 + \sigma_y^2 + C2} \quad (3.29)$$

$$ya(x, y) = \frac{\sigma_{xy} + C3}{\sigma_x\sigma_y + C3} \quad (3.30)$$

$$C1 = (0.01x \text{Im } ax)^2 \quad (3.31)$$

$$C2 = (0.03x \text{Im } ax)^2 \quad (3.32)$$

$$C3 = \frac{C2}{2} \quad (3.33)$$

Burada μ_x ve μ_y ortalama σ_x ve σ_y standart sapma ve σ_{xy} görüntüler arasındaki çapraz kovaryans değerlerini ifade etmektedir.

3.9.3 Karmaşıklık matrisi, doğruluk, özgüllük, hassasiyet, alıcı işlem karakteristiği (AİK) ve eğri altında kalan alan (EAA)

Medikal verilerin sınıflandırma sonuçları değerlendirilirken genellikle aşağıda verilen istatistiki ölçümler kullanılmaktadır (Yaşar, 2015).

Doğru Pozitif (DP): Gerçekte hasta olarak etiketlenen bir verinin sınıflandırıcı tarafından hasta olarak belirlenmesidir.

Doğru Negatif (DN): Gerçekte sağlıklı olarak etiketlenen bir verinin sınıflandırıcı tarafından sağlıklı olarak belirlenmesidir.

Yanlış Pozitif (YP): Gerçekte sağlıklı olarak etiketlenen bir verinin sınıflandırıcı tarafından hasta olarak belirlenmesidir.

Yanlış Negatif (YN): Gerçekte hasta olarak etiketlenen bir verinin sınıflandırıcı tarafından sağlıklı olarak belirlenmesidir.

Bu dört gruba ait (DP, DN, YP, YN) verilerin sayısı hesaplanarak Şekil 3.36'da görüldüğü gibi karmaşıklık matrisi oluşturulmaktadır.

		Tahmin Edilen Sınıf	
		Hasta	Sağlıklı
Gerçek Sınıf	Hasta	DP	YN
	Sağlıklı	YP	DN

Şekil 3.36. Karmaşıklık matrisi gösterimi

Karmaşıklık matrisi incelenerek sınıflandırma performansı hakkında yorum yapmak mümkün hale gelmektedir. Asal köşegen üzerinde bulunan DP ve DN değerleri doğru olarak sınıflandırılan verileri gösterirken diğer YP ve YN değerleri yanlış olarak sınıflandırılan verilerin sayısını göstermektedir. Doğruluk, özgüllük ve hassasiyet değerleri bu ifadeler kullanılarak hesaplanmaktadır (Zhu ve ark., 2010).

Doğru olarak sınıflandırılan verilerin tüm verilere oranı *doğruluk* (Eşitlik 3.34) olarak tanımlanır.

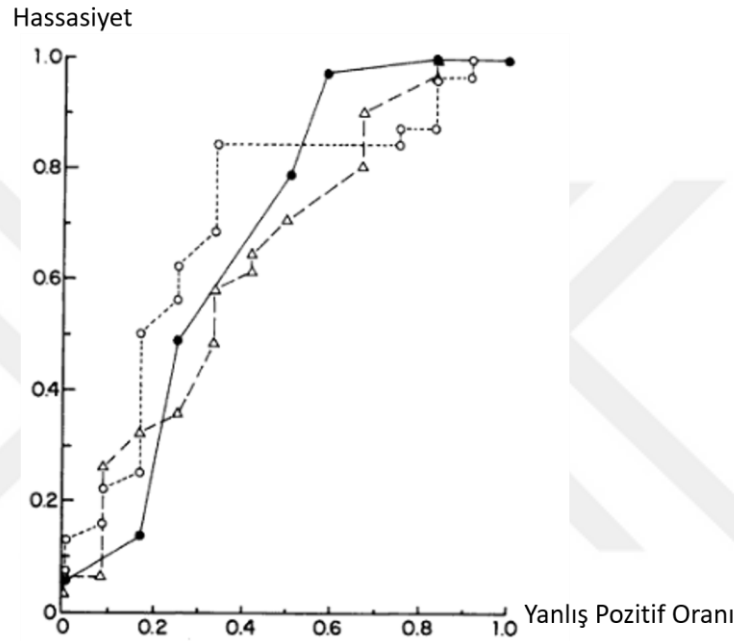
$$Doğruluk = \frac{DP + DN}{DP + YN + YP + DN} \quad (3.34)$$

Sağlıklı olarak sınıflandırılan verilerin tüm sağlıklı verilere oranı *özgüllük* (Eşitlik 3.35) ve hasta olarak sınıflandırılan verilerin tüm hasta verilere oranı ise *hassasiyet* (Eşitlik 3.36) olarak tanımlanır.

$$Ozgulluk = \frac{DN}{YP + DN} \quad (3.35)$$

$$Hassasiyet = \frac{DP}{YN + DP} \quad (3.36)$$

Doğruluk, özgülük ve hassasiyet değerleri ile birlikte alıcı işlem karakteristiği eğrisinin kullanılması da bir performans değerlendirme kriteridir (Yaşar, 2015). Bu eğri farklı değerlendirme kriterlerinin birbirlerine göre değişimlerini yansıtmaktadır. Alıcı işlem karakteristiği eğrisi altında kalan alan değerine bakılarak farklı sınıflandırmalar arasında karşılaştırma yapılabilmektedir. Eğri altında kalan alan değeri 1'e yaklaştıkça sınıflandırmanın hatasız olduğu 0'a yaklaştıkça ise hata değerinin arttığı söylenmektedir. Örnek bir alıcı işlem karakteristiği eğrisi Şekil 3.37'de gösterilmiştir.



Şekil 3.37. 3 farklı sınıflama içeren örnek bir alıcı işlem karakteristiği eğrisi (DeLong ve ark., 1988)

3.10 Sınıflayıcıların Genelleştirme Kabiliyetlerinin Belirlemesi

Değişkenler arasındaki ilişkileri ölçen sayısal sonuçların verileri açıklayıp açıklamadığına karar verme işlemi doğrulama olarak adlandırılır (Drakos, 2018). Diğer bir ifade ile elde bulunan veriler kullanılarak bir sınıflandırıcı yapı oluşturulmak istendiğinde elde edilen değerlendirme kriterlerinin güvenilirliğinin incelenmesi gerekmektedir.

3.10.1 K-katlamalı çapraz doğrulama

Veri kümesinin yalnızca bir bölümünün eğitim ve yalnızca bir bölümünün test olarak ayrılması sonuçların değerlendirilmesi bakımından yeterli değildir. Örneğin 100

adet görselin ilk % 70'lik kısmı yerine son % 70'lik kısmının eğitim için kullanılması test sonuçlarının değişmesine neden olmaktadır.

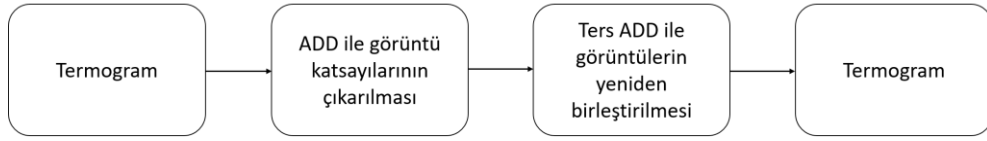
Bu belirsizliğin önüne geçebilmek için k-katlamalı çapraz doğrulama yöntemi geliştirilmiştir (Kohavi, 1995). K-katlamalı çapraz doğrulama yöntemine göre veri K adet bölüme ayrılır ve her bir eğitim işleminde $K - 1$ adet bölüm eğitim kümesi olarak kalan 1 adet bölüm ise test işleminde kullanılır. Gerçekleştirilen çalışmalarda K değeri 10 olarak kabul edilmiştir. Böylece veri kümesi 10 parçaya ayrılarak 9 parçası eğitim 1 parçası test olarak kullanılmıştır. Bu işlem her bölüm test işleminde kullanılana kadar yani K kere tekrar eder. Her bir test işleminin sonucunda karmaşıklık matrisi hesaplanır ve en son K adet karmaşıklık matrisi toplanarak nihai karmaşıklık matrisi elde edilir. Böylece her bir veri hem eğitim hem de test işleminde kullanılmış olur.



4 ARAŞTIRMA SONUÇLARI VE TARTIŞMA

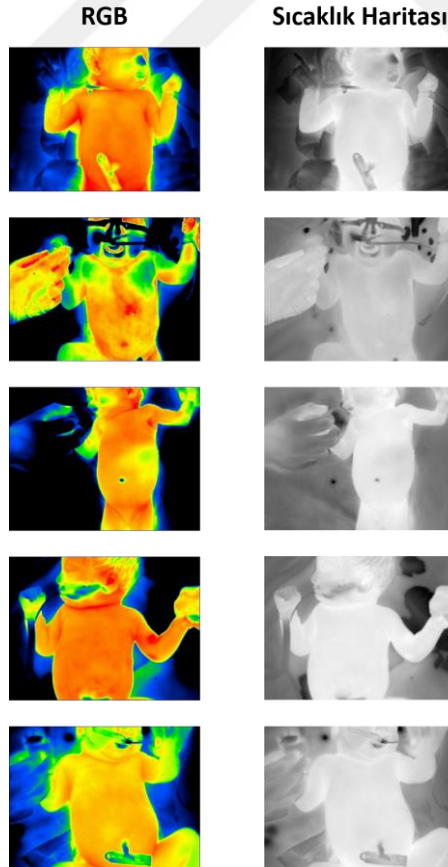
4.1 Termogramların Değerlendirilmesinde Doğru Yaklaşımların Belirlenmesi

Bu çalışmada yeni doğan bebeklere termogramlar hem sıcaklık haritalarına hem de RGB görüntülere dönüştürülerek karşılaştırmaları yapılmıştır. Bu çalışma için gerçekleştirilen işlemlerin blok diyagramı Şekil 4.1’de verilmiştir.

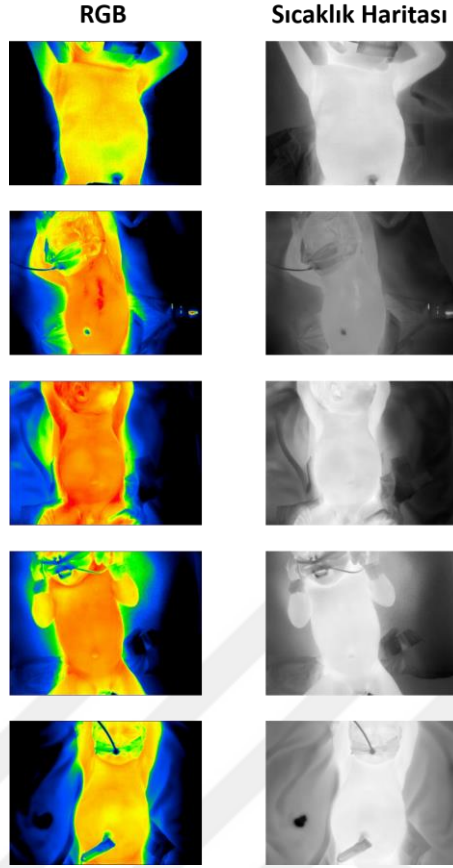


Şekil 4.1. Termogramların değerlendirilmesi için kullanılan yöntemin blok diyagramı

Çalışmada kullanılan beş hasta bebeğe ait RGB görüntüler ve sıcaklık haritaları Şekil 4.2’de, beş sağlıklı bebeğe ait RGB görüntüler ve sıcaklık haritaları Şekil 4.3’te gösterilmiştir.

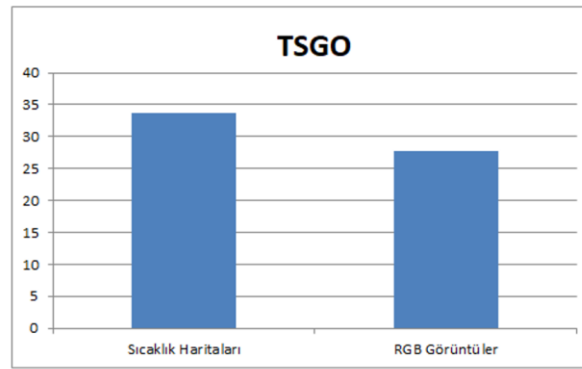


Şekil 4.2. Hasta bebeklere ait RGB görüntüler ve sıcaklık haritaları

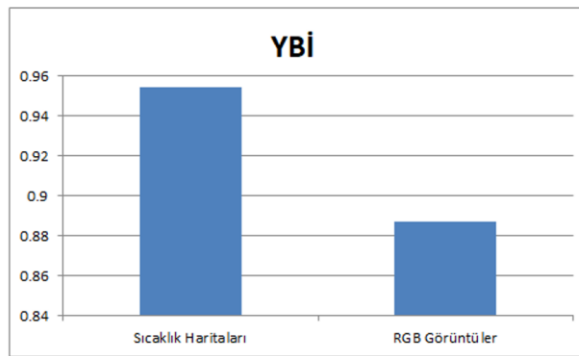


Şekil 4.3. Sağlıklı bebeklere ait RGB görüntüler ve sıcaklık haritaları

Ayrık dalgacık dönüşümü, görüntüleri yaklaşım ve detay olmak üzere alt frekans bileşenlerine ayırarak gürültü giderme, çözünürlük değiştirme, boyut azaltma ve özellik çıkarma gibi işlemlerde kullanılmaktadır (Yaşar, 2015). Haar, daubechies ve morlet gibi dalgacık dönüşümü aileleri bulunmaktadır. Bu çalışmada deneysel olarak *daubechies2* ailesi seçilmiştir. 362x482 boyutunda olan tüm RGB görüntülere ve sıcaklık haritalarına bir kere ADD uygulanarak 181x241 boyutunda yaklaşım ve detay katsayıları elde edilmiştir. Elde edilen bu yaklaşım ve detay katsayıları ters ADD ile geri çatılarak görüntülerin yeniden oluşturulması sağlanmıştır. Böylece, gürültü giderme gibi bir özel işlem yapılmadığında bile ne kadar bilgi kaybının yaşandığı ölçülmüştür. Orijinal görüntüler ile yeniden oluşturulan görüntüler TSGO ve YBİ kriterleri kullanılarak değerlendirilmiştir ve sonuçlar Şekil 4.4'te gösterilmiştir.



(a)



(b)

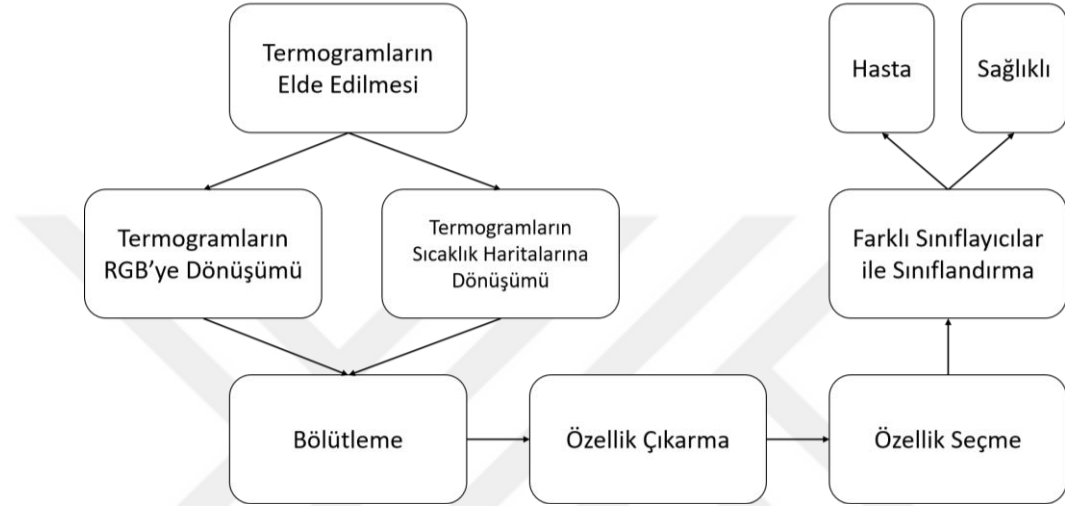
Şekil 4.4. Termogram yaklaşımlarının değerlendirilmesi (a) TSGO değerleri (b) YBİ değerleri

Şekil 4.4 (a)'da gösterilen TSGO değeri sıcaklık haritalarında 36.625 dB; RGB görüntülerde 27.695 dB ve Şekil 4.4 (b)'de gösterilen YBİ değeri sıcaklık haritalarında 0.954; RGB görüntülerde 0.887 olarak elde edilmiştir. Her iki kritere göre de sıcaklık haritalarının değerlerinin daha yüksek olduğu görülmektedir.

Elde edilen sonuçlar termogramlar üzerinde ADD ile işlem yapılacağı zaman sıcaklık haritalarının kullanımının daha uygun olacağını göstermektedir. Özellikle medikal alanda termal görüntüler analiz edilirken direkt sıcaklık haritalarının kullanılması herhangi bir sıcaklık bilgisi kaybı olmadan işlem yapabilmeyi sağlayacaktır.

4.2 Sıcaklık Haritaları ve Makine Öğrenimi Kullanılarak Sağlıklı ve Hasta Bebeklerin Tespiti

Bu çalışmada yirmi sağlıklı ve yirmi hastalık teşhisi konmuş yeni doğan bebeğe ait kırk termogram kullanılarak hasta - sağlıklı sınıflandırma işlemi gerçekleştirilmiştir. Bu işlem için önerilen sistemin blok gösterimi Şekil 4.5'te verilmiştir.

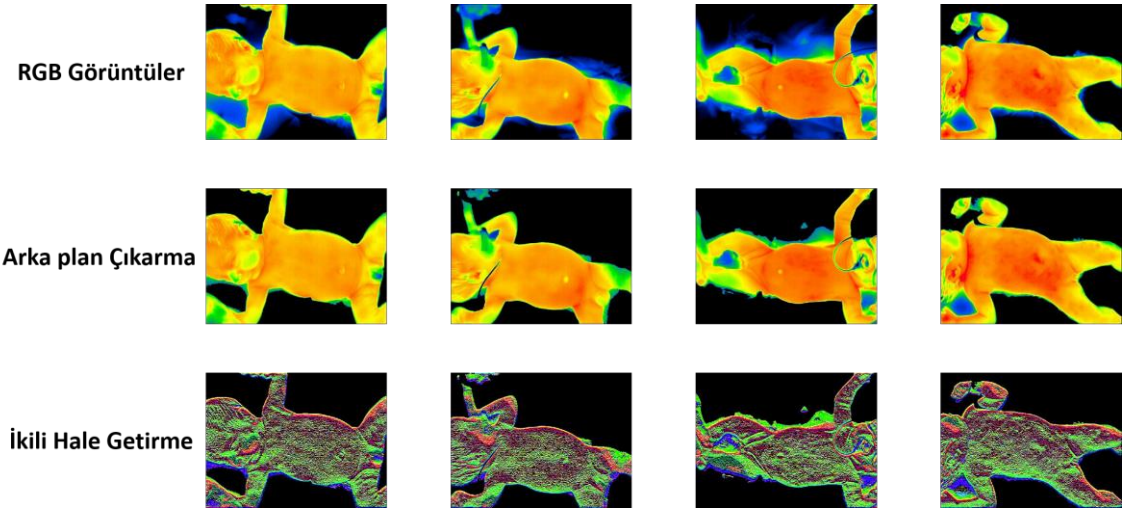


Şekil 4.5. Makine öğrenimi ile sınıflama gerçekleştirmek için önerilen sistemin blok diyagramı

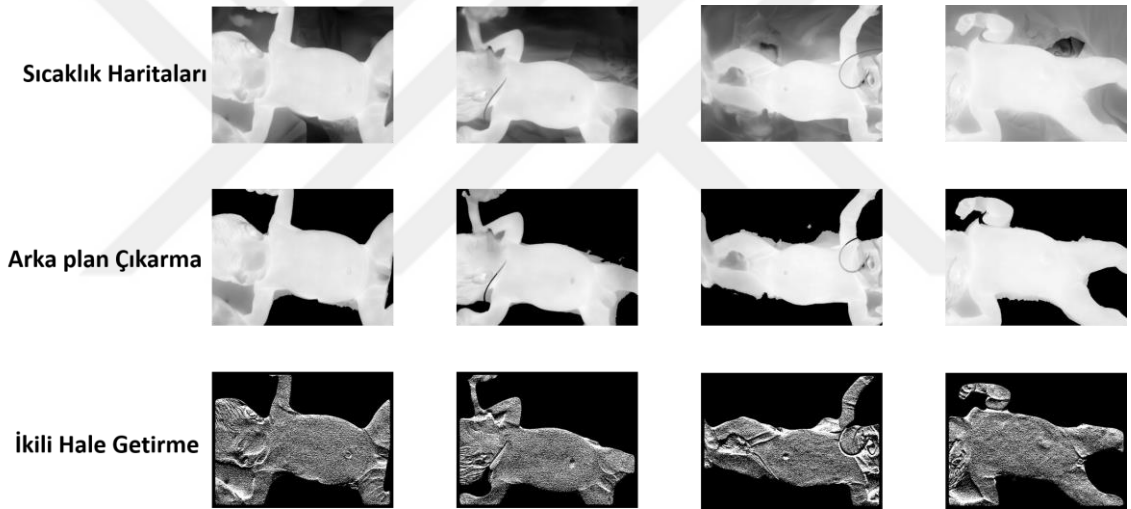
Termogramlar öncelikle RGB görüntülere ve sıcaklık haritalarına çevrilmiştir ve Bölüm 3.3'te açıklanan bölütleme algoritmasına göre:

- Otsu eşikleme yöntemi kullanılarak arka plan çıkarma,
- Türev alıp, 0'dan küçük değerler 0'a, 0'dan büyük değerler ise 1'e eşitlenerek ikili hale getirme işlemleri gerçekleştirilmiştir.

Termogramlara ait RGB ve bölütlenmiş görüntüler Şekil 4.6'da, sıcaklık haritaları ve bölütlenmiş görüntüler Şekil 4.7'de gösterilmiştir.

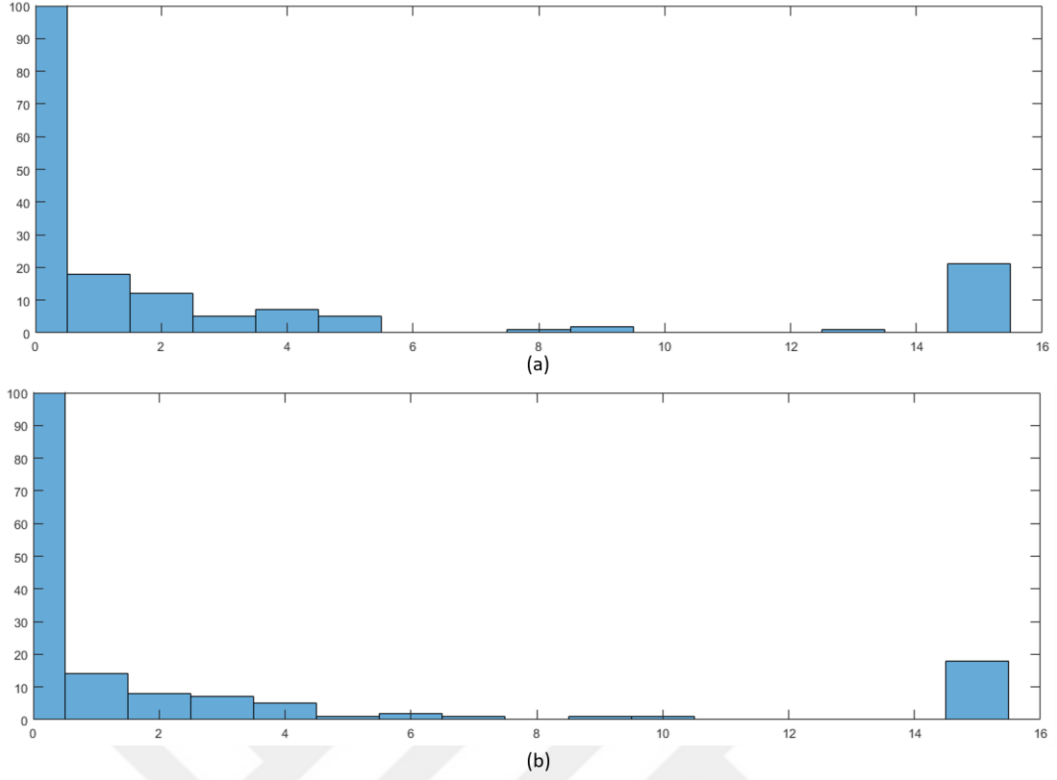


Şekil 4.6. Görüntü veri tabanında yer alan bazı termogramlara ait RGB görüntüler, arka plan çıkarma ve ikili hale getirme ile bölütlenmiş halleri



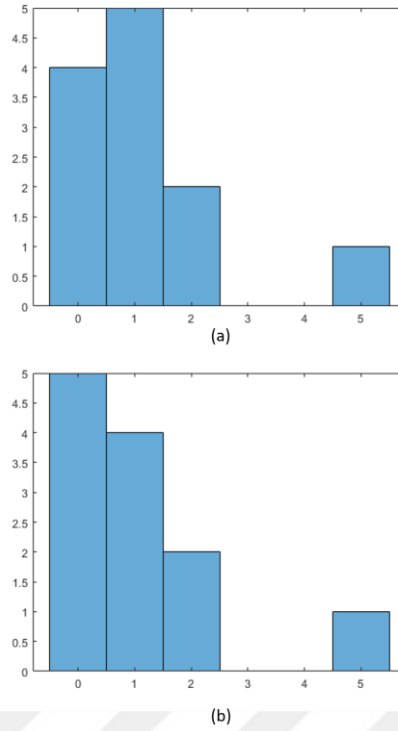
Şekil 4.7. Görüntü veri tabanında yer alan bazı termogramlara ait RGB sıcaklık haritaları, arka plan çıkarma ve ikili hale getirme ile bölütlenmiş halleri

362 x 482 boyutundaki görüntülere *yerel ikili desen* özellik çıkarma algoritması uygulanarak, 757 x 1 boyutunda görüntüyü temsil eden özellik vektörleri elde edilmiştir. Bu özellik vektörlerinden, 1 numaralı hasta bebeğe ait özellik vektörü histogramı Şekil 4.8 (a)'da ve 1 numaralı sağlıklı bebeğe ait özellik vektörü histogramı Şekil 4.8 (b)'de gösterilmiştir.



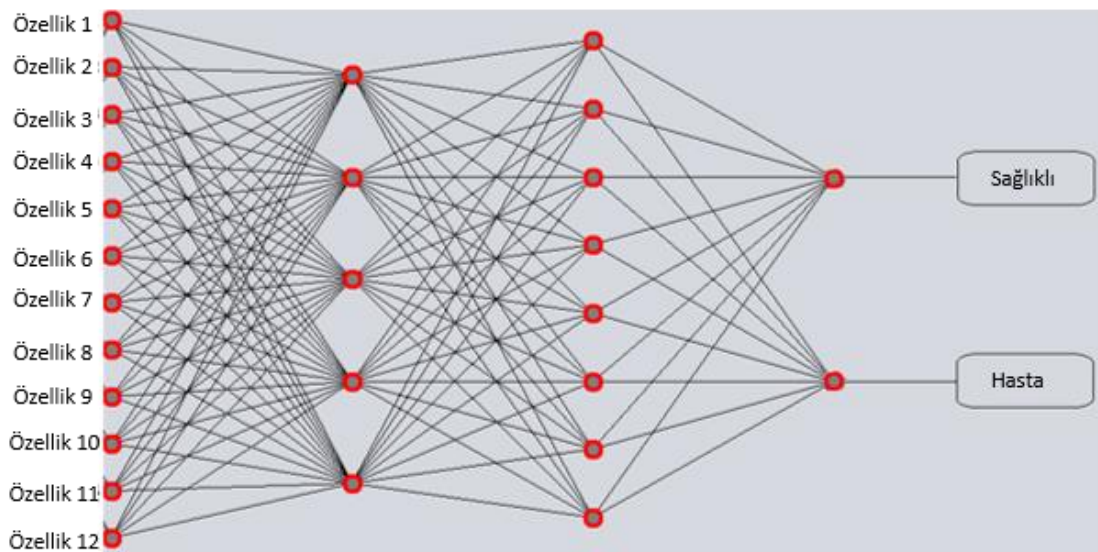
Şekil 4.8. İki bebeğe ait özellik vektörlerinin histogramları (a) bir numaralı hasta bebeğe ait özellik vektörü histogramı (b) bir numaralı sağlıklı bebeğe ait özellik vektörü histogramı

757 x 1 boyutundaki özellik vektörlerine *hızlı korelasyon tabanlı filtre* özellik seçme algoritması uygulanarak vektör boyutları 12 x 1 hale getirilmiştir. Böylece aynı görüntüyü temsil eden önemli özellikler ortaya çıkarılmıştır. 1 numaralı hasta ve 1 numaralı sağlıklı bebeğe ait özellik 12 x 1 vektör histogramları sırasıyla Şekil 4.9 (a) ve Şekil 4.9 (b)'de verilmiştir.



Şekil 4.9. Özellik seçimi uygulandıktan sonra elde edilen vektörlerin histogramları (a) bir numaralı hasta bebeğe ait 12 x 1 özellik vektörü histogramı (b) bir numaralı sağlıklı bebeğe ait 12 x 1 özellik vektörü histogramı

Özellik seçme işleminin ardından elde edilen özellikler yapay sinir ağları, lojistik regresyon, karar ağaçları ve rastgele orman algoritmaları ile sınıflandırılmıştır. Görüntüleri temsil eden 12 x 1 boyutlu özellik vektörlerinin sınıflanması için oluşturulan YSA modeli Şekil 4.10'da gösterilmiştir.



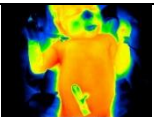

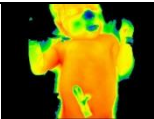

Şekil 4.10. Sınıflandırma için oluşturulmuş 2 gizli katmanlı ve 2 çıkış katmanı bir YSA modeli. Giriş katmanında bulunan 12 özellik HKTF kullanılarak elde edilmiştir.

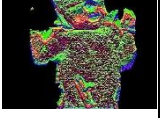

Burada model parametreleri deneysel olarak aşağıdaki şekilde seçilmiştir.

- Maksimum hata: 1e-20
- İlk gizli katman: 5 nöron
- İkinci gizli katman: 8 nöron
- Öğrenme oranı: 0.3
- Maksimum iterasyon: 500

Özellik seçimi yapılmadan 757 x 1 boyutlu vektörler ile gerçekleştirilen sınıflama için yalnızca YSA'nın giriş katmanında bulunan nöronların sayısı değişmektedir (12'den 757'ye yükseliyor). Karar ağaçları, rastgele orman ve lojistik regresyon ile gerçekleştirilen sınıflandırmalarda WEKA programının varsayılan parametreleri kullanılmıştır (WEKA, 2019b; 2019c; 2019a). 10 – katlamalı çapraz doğrulama kullanılarak elde edilen sonuçlar Çizelge 4.1'de gösterilmiştir.

Çizelge 4.1. Makine öğrenimi ile gerçekleştirilen sınıflamaların sonuçları

Termogram Yaklaşımları	Görüntü Tipleri			Örnek Görüntüler	Sınıflandırma Algoritmaları (% doğruluk) Özellik seçimi yapmadan, Özellik seçimi yaparak				
	Orijinal	Bölütlenmiş			YSA	LR	KA	RO	
		Arka plan çıkarma	İkili hale getirme						
1	RGB	<input checked="" type="checkbox"/>				55.00, 72.50	55.00, 67.50	42.50, 55.00	52.50, 72.50
2	Sıcaklık Haritaları	<input checked="" type="checkbox"/>				47.50, 62.50	40.00, 72.50	47.50, 62.50	50.00, 67.50
3	RGB		<input checked="" type="checkbox"/>			50.00, 62.50	52.50, 60.00	37.50, 55.00	52.50, 77.50
4	Sıcaklık Haritaları		<input checked="" type="checkbox"/>			47.50, 65.00	47.50, 75.00	40.00, 70.00	52.50, 70.00

5	RGB		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		60.00, 67.50	50.00, 72.50	40.00, 72.50	57.50, 80.00
6	Sıcaklık Haritaları		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		62.50, 92.50	52.50, 87.50	45.00, 82.50	65.00, 90.00

Çizelge 4.1’de bölütleme ve özellik seçiminin etkisi görülmektedir. Bölütleme ve özellik seçimi uygulanmadan RGB görüntülerin sınıflanması en düşük % 42.5 (RGB+özellik çıkarma+KA) ve en yüksek % 55 (RGB+özellik çıkarma+LR) doğruluk ile gerçekleşmiştir. Yine bölütleme ve özellik seçimi uygulanmadan sıcaklık haritalarının sınıflanması en düşük % 40 (sıcaklık haritası+özellik çıkarma+LR) en yüksek % 50 (sıcaklık haritası+özellik çıkarma+RO) doğruluk ile gerçekleşmiştir. Buradan görüldüğü gibi görüntüler orijinal halleriyle özellik çıkarma yapılarak sınıflandırıldığında en yüksek % 55 (RGB+özellik çıkarma+LR) doğruluk elde edilmektedir.

Orijinal görüntülere özellik çıkarma uygulandıktan sonra özellik seçme uygulandığında ise doğruluk değeri RGB görüntülerde % 12.5 ile % 20; sıcaklık haritalarında % 15 ile % 32.5 arasında artış göstermiştir. Böylece orijinal görüntülerin sınıflandırılmasında en iyi doğruluk değeri % 72.5 (RGB+özellik çıkarma+özellik seçme+RO) olarak elde edilmiştir.

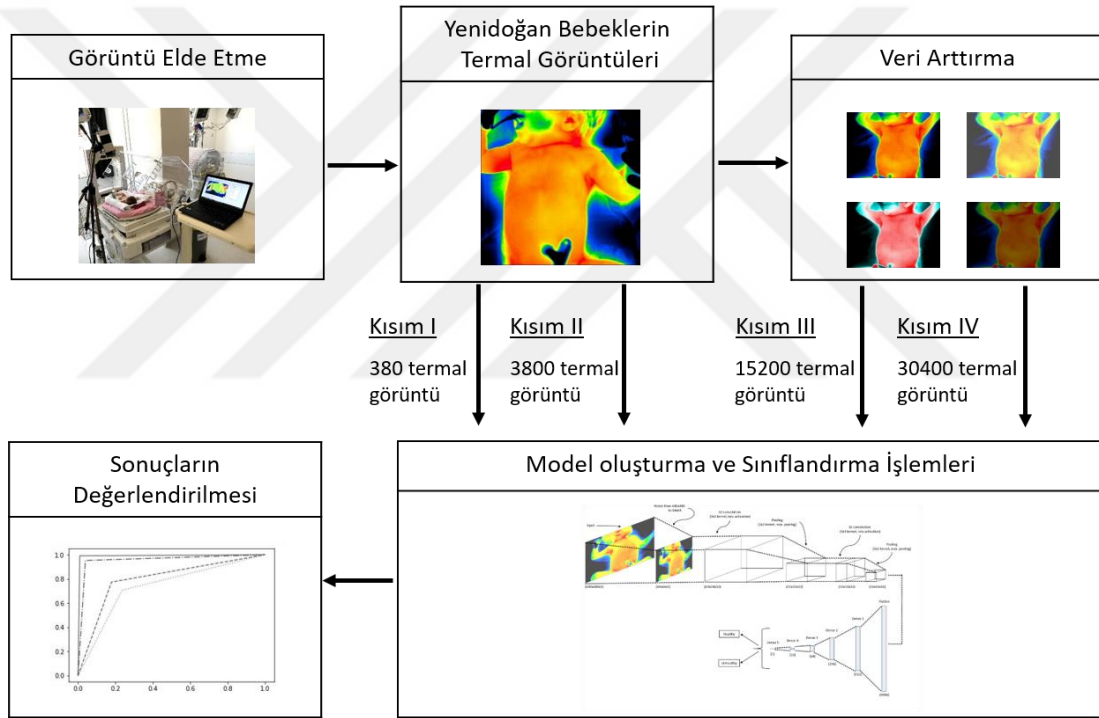
Orijinal görüntülere, OTSU eşikleme metodu ile arka plan çıkarma işlemi uygulandığında ve özellik seçimi yapılmadığında RGB görüntülerde ve sıcaklık haritalarında en yüksek % 52.5 (RGB+arka plan çıkarma+özellik çıkarma+LR, sıcaklık haritaları+arka plan çıkarma+özellik çıkarma+RO) doğruluk elde edilmiştir. Özellik seçimi de uygulandığında ise en yüksek doğruluk değeri % 77.5 (RGB+arka plan çıkarma+özellik çıkarma+özellik seçme+RO) olarak elde edilmiştir.

Arka planı çıkarılmış görüntüler ikili hale getirildikten sonra, özellik seçimi uygulanmadan sınıflandırdıklarında RGB görüntülerde en yüksek % 60 (RGB+arka plan çıkarma+ikili hale getirme+özellik çıkarma+YSA); sıcaklık haritalarında en yüksek % 65 (sıcaklık haritası+arka plan çıkarma+ikili hale getirme+özellik çıkarma+RO) doğruluk değeri ile elde edildiği görülmektedir. Özellik seçimi uygulandığında ise RGB görüntülerde en yüksek % 80 (RGB+arka plan çıkarma+ikili hale getirme+özellik çıkarma+özellik seçme+RO); sıcaklık haritalarında en yüksek % 92.5 (sıcaklık haritası+arka plan çıkarma+ikili hale getirme+özellik çıkarma+özellik seçme+YSA)

doğruluk değeri elde edilmiştir. Buradan; sıcaklık haritaları, arka plan çıkarma, ikili hale getirme, özellik çıkarma ve özellik seçme bir arada kullanıldığında elde edilen sınıflandırma sonuçlarının diğer tüm sonuçlardan yüksek olduğu görülmektedir.

4.3 Evrişimli Sinir Ağları Kullanılarak Yeni Doğan Bebeklerin Sağlık Durumlarının Tespit Edilmesi

Bu çalışmada yeni doğan bebeklerin sağlık durumlarının belirlenmesi için ESA ve veri artırımı yöntemleri kullanılmıştır. Çalışma toplamda dört kısımdan meydana gelmektedir ve önerilen sisteme ait blok diyagram Şekil 4.11’de verilmiştir.



Şekil 4.11. Evrişimli sinir ağları ile sınıflama gerçekleştirmek için önerilen sistemin blok diyagramı

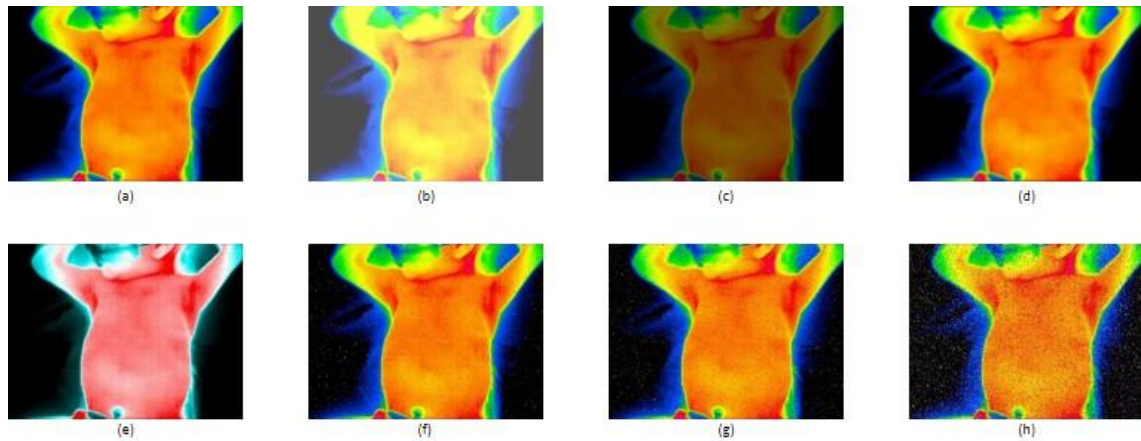
Yeni doğan yoğun bakım ünitesinde görüntü alınırken bir bebekten bir dakika içerisinde yüz görüntü elde edilmektedir. Çizelge 4.2’den görüldüğü gibi, çalışmanın birinci kısmında bu yüz görüntü içerisinde ilk on görüntü seçilerek 19 hasta ve 19 sağlıklı bebeğe ait 380 görüntü ile sınıflandırma gerçekleştirilmiştir. İkinci kısımda ise bebeklere ait tüm görüntüler kullanılmış ve 3800 görüntü ile sınıflandırma gerçekleştirilmiştir.

Çizelge 4.2. Dört kısım boyunca kullanılan görüntü sayıları

Kısım	Bir bebeğe ait original görüntü sayısı	Bir bebeğe ait seçilen orijinal görüntü sayısı	Bir bebeğe ait artırılmış görüntü sayısı	Toplam bebek sayısı	Toplam görüntü sayısı
I	100	10	-	38	380
II	100	100	-	38	3800
III	100	100	300	38	15200
IV	100	100	700	38	30400

Üçüncü ve dördüncü kısımda veri artırımının sınıflama performansına etkisini gözlemek amacıyla orijinal verilere ek olarak artırılmış görüntüler de kullanılmıştır. Üçüncü kısımda 100'er görüntüye parlaklık artırma, kontrast değiştirme ve tuz – biber gürültüsü ekleme yöntemleri uygulanarak 400'er görüntü (100 orijinal, 300 artırılmış) elde edilmiştir ve böylece 15200 görüntü ile sınıflandırma gerçekleştirilmiştir. Dördüncü kısımda ise çözünürlük değiştirme, renk dönüştürme ve iki farklı yoğunlukta tuz – biber gürültüsü ekleme yöntemleri de dahil edilerek her bir bebekten 800'er görüntü (100 orijinal, 700 artırılmış) elde edilmesi sağlanmıştır ve böylece 30400 görüntü ile sınıflandırma gerçekleştirilmiştir.

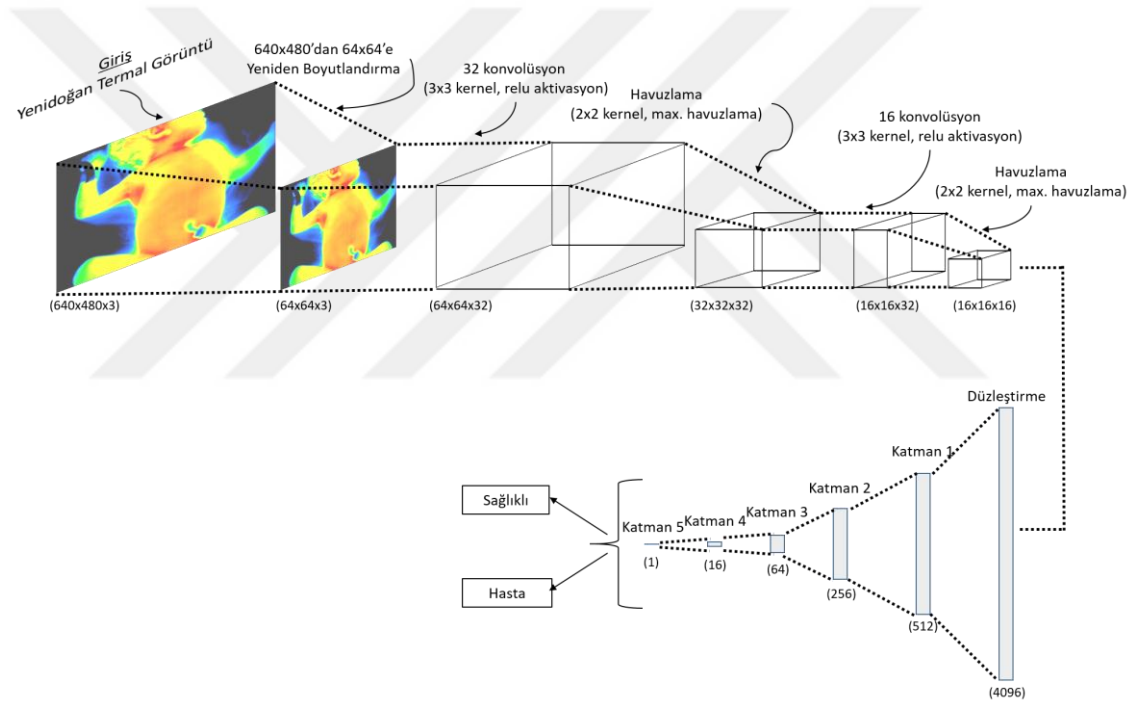
Artırılmış görüntüler Şekil 4.12'de gösterilirken sınıflandırma için oluşturulan ESA modeli Şekil 4.13'te verilmiştir.



Şekil 4.12. Artırma ile oluşturulan termal görüntüler (a) orijinal (b) parlaklık artırma (c) kontrast değiştirme (d) çözünürlük değiştirme (e) renk dönüşümü (f-h) sırasıyla 0.01, 0.02 ve 0.08 tuz ve biber gürültüsü ekleme.

Şekil 4.12 (a)'da görülen orijinal görüntünün piksel değerleri 0 ile 1 arasında değişmektedir. Görüntünün parlaklığını artırmak için tüm değerlere 0.3 eklenmiştir. Bu işlem sonucunda 1'den büyük olan değerler 1'e eşitlenerek yeni piksel değerleri 0.3 ile

1 arasına getirilmiştir. Elde edilen görüntü Şekil 4.12 (b)'de görülmektedir. Görüntünün kontrastını değiştirmek için tüm piksel değerleri 0.5 ile çarpılmıştır. Böylece Şekil 4.12 (c)'de görüldüğü gibi daha karanlık bir görüntü elde edilmiştir. Renk dönüştürme işlemi için RGB görüntüler kırmızı, yeşil ve mavi uzaylarına ayrılmıştır ve ardından kırmızı ve yeşil uzaylar ile RGG olarak yeniden birleştirilmiştir. Böylece görüntüdeki mavi bant silinerek Şekil 4.12 (e)'deki gibi bir sonuç elde edilmiştir. Oluşturulan sistemin görüntü alma esnasında yaşanabilecek sıcaklık kaymalarına karşı güçlü olması gerekmektedir. Bu durumu modellemek için görüntülere 0.01, 0.02 ve 0.08 yoğunlukta olmak üzere tuz – biber gürültüsü eklenmiştir. Elde edilen görüntüler Şekil 4.12 (f-h) arasında sırası ile görülmektedir.



Şekil 4.13. Kısım 1-2-3-4'teki görüntüleri sınıflandırmak için oluşturulmuş ESA modeli

Şekil 4.13'den görüldüğü gibi oluşturulan ESA modeli 2 konvolüsyon, 2 havuzlama, 1 düzleştirme ve 5 nöron katmanına sahiptir. İlk konvolüsyon katmanı 3x3 boyutunda 32 farklı filtreden (aktivasyon = ReLU) ikinci konvolüsyon katmanı ise 3x3 boyutunda 16 farklı filtreden (aktivasyon = ReLU) meydana gelir. Eklenen iki havuzlama işleminde de 2x2 boyutunda filtre kullanılmış ve maksimum havuzlama gerçekleştirilmiştir. Konvolüsyon ve havuzlama işlemleri tamamlandıktan sonra 16x16 boyutunda 16 adet özellik haritası elde edilmiştir. Bu özellik haritaları düzleştirme katmanı ile 4096 x 1 (16x16x16) olarak vektörel hale getirilmiştir. Nöron katmanının ilk

dört katmanı sırası ile 512, 256, 64 ve 16 nöron içerirken (aktivasyon = ReLU) çıkış katmanı olan beşinci katman 1 nöron (aktivasyon = sigmoid) içermektedir. Bölüm 3.7.1.2’de açıklandığı gibi sigmoid fonksiyonu 0 ile 1 değerleri arasında çıkış üretmektedir. Burada eğer çıkış değeri 0.5’ten büyük ise hasta, 0.5’e eşit veya daha düşük ise sağlıklı olarak sınıflandırma gerçekleşmektedir.

Dört uygulama için de 10 – katlamalı çapraz doğrulama yöntemi kullanılmıştır. Buna göre eğitim ve test için kullanılan veri sayıları Çizelge 4.3’te verilmiştir.

Çizelge 4.3. 10 – katlamalı çapraz doğrulama yöntemine göre eğitim ve test için kullanılan veri sayıları

Kısım	Toplam	Eğitim	Test
I	380	342	38
II	3800	3420	380
III	15200	13680	1520
IV	30400	27360	3040

Elde edilen sonuçlar hassasiyet, özgüllük, doğruluk, alıcı işlem karakteristiği ve eğri altında kalan alan (EAA) kullanılarak değerlendirilmiştir.

Birinci kısımda Çizelge 4.4’ten görüldüğü gibi 190 hasta sınıfına ait görüntünün 156’sı hasta ve 190 sağlıklı sınıfına ait görüntünün 147’si sağlıklı olarak sınıflandırılmıştır. Buna göre % 82.10 hassasiyet, % 77.36 özgüllük, % 79.73 doğruluk ve EAA değeri 0.797 olarak elde edilmiştir.

Çizelge 4.4. Kısım 1 Sonuçları

Karmaşıklık Matrisi		Hassasiyet	Özgüllük	Doğruluk	EAA
156	34	% 82.10	% 77.36	% 79.73	0.797
43	147				

Kısım 2’de 3800 görüntü ile sınıflandırma yapıldığında Çizelge 4.5’ten görüldüğü gibi sınıflandırma performansı düşmüştür. Örneğin doğruluk değeri % 79.73’ten % 70.73’e düşmüştür.

Çizelge 4.5. Kısım 2 Sonuçları

Karmaşıklık Matrisi		Hassasiyet	Özgüllük	Doğruluk	EAA
1451	449	% 76.36	% 70.73	% 73.55	0.736
556	1344				

Kısım 3'te parlaklık artırma, kontrast değiştirme ve tuz – biber gürültüsü ekleme ile artırılmış 15200 (3800 orijinal, 11400 artırılmış) görüntü sınıflandırıldığında Çizelge 4.6'dan görüleceği gibi hassasiyet, özgüllük ve doğruluk değerleri % 99'un üzerine çıkmıştır ve EAA değeri 1'e çok yaklaşarak 0.991 olmuştur.

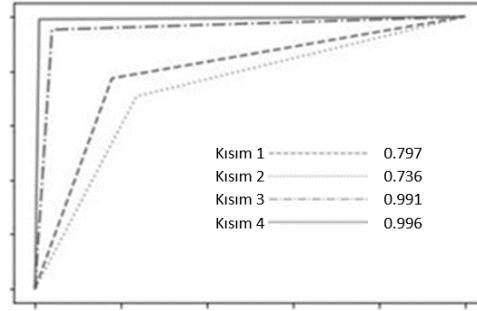
Çizelge 4.6. Kısım 3 Sonuçları

Karmaşıklık Matrisi		Hassaslık	Özgüllük	Doğruluk	EAA
7529	71	% 99.06	% 99.07	% 99.07	0.991
70	7530				

Dördüncü kısımda ESA'nın eğitilmesi ve test edilmesi için 30400 (3800 orijinal, 26600 artırılmış) görüntü kullanılmıştır ve en iyi sonuçlar burada elde edilmiştir. Çizelge 4.7'den görüleceği gibi 15200 hasta sınıfına ait görüntünün 15159'u hasta ve 15200 sağlıklı sınıfına ait görüntünün 15114'ü sağlıklı olarak sınıflandırılmıştır. Böylece % 99.58 doğruluk, % 99.73 özgüllük, % 99.43 hassasiyet ve 0,996 EAA değeri elde edilmiştir. Ayrıca 4 kısımdan elde edilen AİK eğrisi Şekil 4.14'de gösterilmiştir.

Çizelge 4.7. Kısım 4 Sonuçları

Karmaşıklık Matrisi		Hassaslık	Özgüllük	Doğruluk	EAA
15159	41	% 99.73	% 99.43	% 99.58	0.996
86	15114				



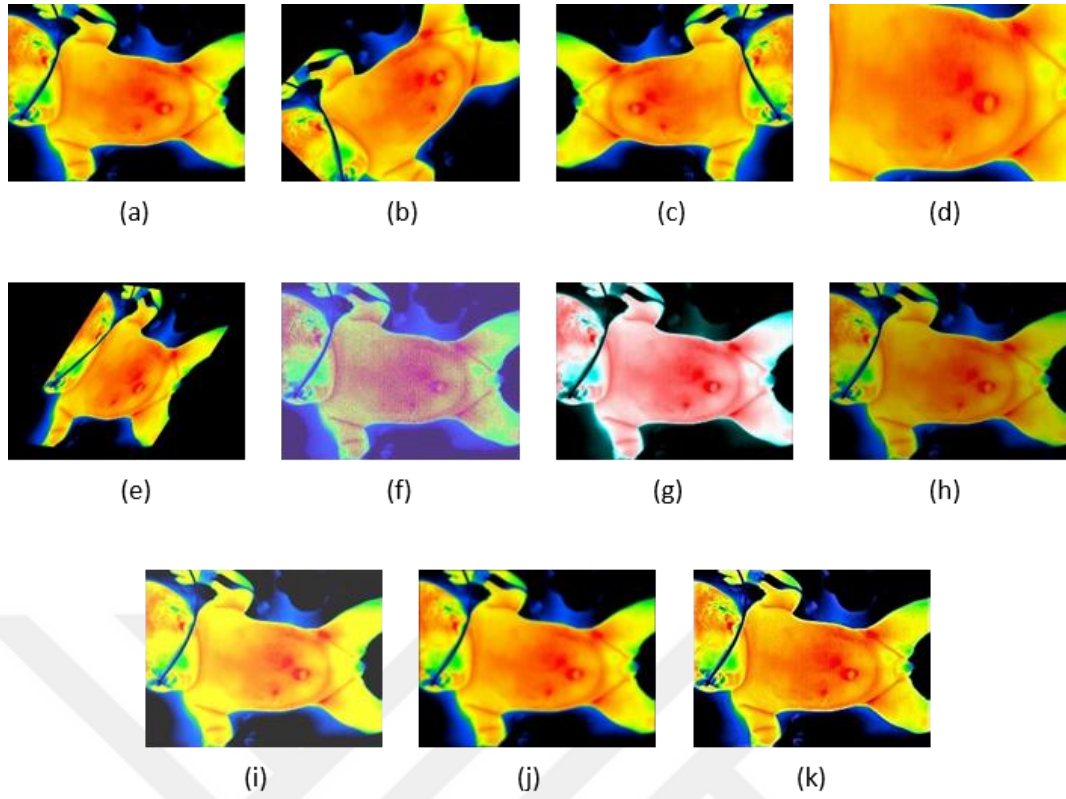
Şekil 4.14. Kısım 1-2-3-4'e ait sınıflama sonuçlarının alıcı işlem karakteristiği eğrileri ve EAA değerleri

4.3.1 Geleneksel veri artırma yöntemlerinin sınıflama üzerine etkilerinin karşılaştırılması

Bölüm 4.3'te, veri artırımının sınıflama performansını artırdığı gözlenmiştir. Bölüm 4.3.1'de farklı veri artırma teknikleri (rotasyon değiştirme, aynalama, yakınlaştırma, 3B döndürme, histogram eşitleme, keskinleştirme, bulanıklaştırma, renk değiştirme, aydınlık artırma ve kontrast değiştirme) üçerli olarak bir araya getirilmiş ve sınıflama performansına olan etkileri incelenmiştir. Bölüm 4.3.1 için kullanılan veri artırımı yöntemleri aşağıda açıklanmıştır:

- Rotasyon değiştirme: Görüntünün rotasyonunun değiştirilmesi için merkez nokta etrafında saat yönünün tersi yönde 45^0 çevirme işlemi gerçekleştirilmiştir. Görüntünün boyutlarında değişim olmaması için 640×480 çerçeve dışında kalan kısım Şekil 4.15 (b)'de gösterildiği gibi kırılmıştır.
- Aynalama: Görüntüdeki her bir piksel değerinin y eksenine göre simetriği alınarak aynalama işlemi gerçekleştirilmiştir. Bu işleme göre pikselin mevcut konumu (x, y) iken $(640-x, y)$ haline getirilmiştir. Elde edilen görüntüler Şekil 4.15 (c)'de gösterilmiştir.
- Yakınlaştırma: Görüntüyü yakınlaştırmak için öncelikle 640×480 boyutlu görüntünün 140. sütundan 570. sütuna kadar ve 140. satırdan 400. satıra kadar olan kısmı kesilmiştir. Böylece görüntü merkezi etrafında bir alan meydana getirilmiştir. Şekil 4.15 (d)'de görüldüğü gibi kesilen kısım yeniden boyutlandırma ile 640×480 hale getirilerek yakınlaştırma işlemi gerçekleştirilmiştir.

- 3 boyutlu döndürme: Rotasyonu değiştirilen görüntüler, Şekil 4.15 (e)'de görüldüğü gibi z eksenini boyunca da çevrilerek üç boyutlu döndürme gerçekleştirilmiştir.
- Histogram eşitleme: Gri seviyeli görüntülerde siyah – beyaz dengesi oluşturmak için kullanılan yöntem histogram eşitleme denir. Histogram eşitlemeyi görüntülere uygulamak için öncelikle görüntüler kırmızı, yeşil ve mavi bantlarına ayrılmıştır ve ardından her bir banda histogram eşitleme uygulanarak görüntü yeniden birleştirilmiştir. Elde edilen görüntüler Şekil 4.15 (f)'de gösterilmiştir.
- Renk değiştirme: Görüntünün rengini değiştirmek için görüntü yeniden kırmızı, yeşil ve mavi bantlara bölünmüştür. Görüntü yeniden birleştirilirken mavi bant elimine edilerek kırmızı, yeşil ve yeniden yeşil bant kullanılmıştır. Böylece Şekil 4.15 (g)'de görüldüğü gibi farklı renklerde görüntüler elde edilmiştir.
- Karşıtlık ve aydınlık değiştirme: Tüm piksel değerleri 0,75 değeri ile çarpılarak görüntülerin karşıtlığı ve tüm piksel değerlerine 0,2 değeri eklenerek görüntülerin aydınlığı değiştirilmiştir. Oluşan görüntüler Şekil 4.15 (h, i)'da gösterilmiştir.
- Bulanıklaştırma ve keskinleştirme: Görüntülerin bulanıklaştırılması için standart sapması 2 olan iki boyutlu alçak geçiren gauss filtresi kullanılmıştır. Görüntülerin keskinleştirilmesi için ise görüntülerden bulanık hallerinin çıkartılması gerçekleştirilir. Burada görüntülerin bulanıklaştırılması için standart sapması 1,5 olan gauss filtresi kullanılmıştır. Elde edilen görüntüler Şekil 4.15 (j, k)'da gösterilmiştir.



Şekil 4.15. Orijinal termal görüntü ve artırılmış termal görüntüler (a) orijinal (b) rotasyon (c) aynalama (d) yakınlaştırma (e) 3B döndürme (f) histogram eşitleme (g) renk değiştirme (h) kontrast değiştirme (i) aydınlık değiştirme (j) bulanıklaştırma (k) keskinleştirme

Toplamda on farklı karşılaştırma içeren bu çalışmada her bir karşılaştırmada üç farklı veri artırım metodu uygulanmıştır. Böylece 15200 (3800 orijinal, 11400 artırılmış) görüntü kullanılarak hasta/ sağlıklı sınıflandırılması gerçekleştirilmiştir. Sınıflandırıcı olarak kullanılan ESA modeli bir önceki çalışmada (Bölüm 4.3.) açıklanmıştır. 10 – katlamalı çapraz doğrulama ile elde edilen sonuçlar özgüllük, hassasiyet ve doğruluk kriterleri ile değerlendirilmiştir. Gerçekleştirilen tüm karşılaştırmalar Çizelge 4.8’de ve Çizelge 4.9’da görülmektedir. Karmaşıklık matrisleri ise Şekil 4.16 ve 4.17’de gösterilmiştir.

Çizelge 4.8. Veri artırımı için bir araya getirilen yöntemler

Karşılaştırma	Kullanılan Metotlar		
	Metot 1	Metot 2	Metot 3
A	Rotasyon Değiştirme	Aynalama	Yakınlaştırma
B	Histogram Eşitleme	Renk Değiştirme	Rotasyon Değiştirme
C	Karşıtlık Değiştirme	Aydınlık Arttırma	Keskinleştirme
D	Karşıtlık Değiştirme	Keskinleştirme	Bulanıklaştırma
E	3B Döndürme	Bulanıklaştırma	Aynalama
F	Yakınlaştırma	Renk Değiştirme	Aydınlık Arttırma
G	Rotasyon Değiştirme	3B Döndürme	Yakınlaştırma
H	Keskinleştirme	3B Döndürme	Renk Değiştirme
I	Rotasyon Değiştirme	Bulanıklaştırma	Aydınlık Arttırma
J	Histogram Eşitleme	Yakınlaştırma	Aydınlık Arttırma

	Karşılaştırma A		Karşılaştırma B		Karşılaştırma C		Karşılaştırma D		Karşılaştırma E	
Parça 1	419	341	760	0	759	1	760	0	760	0
	506	254	2	758	0	760	2	758	760	0
Parça 2	625	135	731	29	657	103	756	4	752	8
	163	597	0	760	0	760	0	760	11	749
Parça 3	760	0	674	86	760	0	760	0	756	4
	760	0	29	731	1	759	9	751	9	751
Parça 4	671	89	760	0	760	0	760	0	716	44
	496	264	135	625	4	756	0	760	3	757
Parça 5	505	255	749	11	760	0	760	0	396	364
	248	512	178	588	1	759	0	760	364	396
Parça 6	280	480	760	0	760	0	760	0	760	0
	178	582	9	751	0	760	2	758	0	760
Parça 7	281	479	501	259	383	377	760	0	662	98
	79	681	0	760	372	388	0	760	334	426
Parça 8	577	183	689	71	692	68	757	3	552	208
	22	738	1	759	0	760	0	760	165	595
Parça 9	667	93	737	23	712	48	756	4	273	487
	330	430	2	758	23	737	0	760	17	743
Parça 10	518	242	755	5	760	0	760	0	584	176
	412	348	8	752	5	755	0	760	176	584
TOPLAM	5303	2297	7116	484	7003	597	7589	11	6211	1389
	3194	4466	358	7242	406	7194	13	7587	1839	5761

Şekil 4.16. Karşılaştırma A-B-C-D-E'ye ait 10 – katlamalı çapraz doğrulama kullanılarak elde edilen matrisleri

	Karşılaştırma F		Karşılaştırma G		Karşılaştırma H		Karşılaştırma I		Karşılaştırma J	
Parça 1	760	0	560	200	712	48	760	0	759	1
	0	760	382	378	0	760	43	717	1	759
Parça 2	733	27	705	55	757	3	760	0	751	9
	0	760	135	625	3	757	0	760	319	441
Parça 3	744	16	297	463	759	1	760	0	757	3
	0	760	90	670	0	760	1	759	8	752
Parça 4	758	2	386	374	759	1	756	4	755	5
	19	741	315	445	22	738	0	760	230	530
Parça 5	677	83	446	314	684	76	760	0	713	47
	0	760	87	673	31	729	5	755	4	756
Parça 6	752	8	543	217	754	6	760	0	760	0
	2	758	137	623	243	517	411	349	288	472
Parça 7	726	34	251	509	719	41	756	4	748	12
	0	760	0	760	193	567	552	208	344	416
Parça 8	682	78	533	227	635	125	648	112	624	136
	4	756	55	705	79	681	103	657	192	568
Parça 9	660	100	741	19	238	522	226	534	480	280
	293	467	260	500	273	487	0	760	0	760
Parça 10	540	220	534	226	456	304	390	370	672	88
	325	435	360	400	10	750	10	750	257	503
TOPLAM	7032	568	4996	2604	6473	1127	6576	1024	7019	581
	643	6957	1821	5779	857	6746	1125	6475	1643	5957

Şekil 4.17. Karşılaştırma F-G-H-I-J'ye ait 10 – katlamalı çapraz doğrulama kullanılarak elde edilen matrisleri

10 – katlamalı çapraz doğrulama tekniğine göre elde edilen hassasiyet, özgüllük ve doğruluk değerleri Çizelge 4.9’da verilmiştir.

Çizelge 4.9. Veri artırımı için kullanılan yöntemlerin sınıflama sonuçları

Karşılaştırma	Kullanılan Metotlar			Hassasiyet (%)	Özgüllük (%)	Doğruluk (%)
	Veri Artırımı Olmadan			76,36	70,73	73,55
A	Rotasyon Değişirme	Aynalama	Yakınlaştırma	69,77	58,76	64,26
B	Histogram Eşitleme	Renk Değişirme	Rotasyon Değişirme	93,63	95,28	94,46
C	Karşıtlık Değişirme	Aydınlık Arttırma	Keskinleştirme	92,14	94,65	93,40
D	Karşıtlık Değişirme	Keskinleştirme	Bulanıklaştırma	99,85	99,82	99,84
E	3B Döndürme	Bulanıklaştırma	Aynalama	81,72	75,80	78,76
F	Yakınlaştırma	Renk Değişirme	Aydınlık Arttırma	92,52	91,53	92,03
G	Rotasyon Değişirme	3B Döndürme	Yakınlaştırma	65,73	76,03	70,88
H	Keskinleştirme	3B Döndürme	Renk Değişirme	85,17	88,76	86,96
I	Rotasyon Değişirme	Bulanıklaştırma	Aydınlık Arttırma	86,52	85,21	85,86
J	Histogram Eşitleme	Yakınlaştırma	Aydınlık Arttırma	92,35	78,38	85,36

Çizelge 4.9’den görüldüğü gibi hiçbir veri artırımı tekniği uygulanmadan 3800 görüntü ile sınıflama gerçekleştirildiğinde % 76,36 hassasiyet, % 70,73 özgüllük ve % 73,55 doğruluk değeri elde edilmektedir. Karşılaştırma A’daki gibi (rotasyon değişirme, aynalama, yakınlaştırma) yöntemler bir araya getirildiğinde değerlendirme kriterleri % 69.77 hassasiyet, % 58.76 özgüllük ve % 64.26 doğruluk değerlerine düşerken; karşılaştırma B (histogram eşitleme, renk değişirme, rotasyon değişirme), karşılaştırma C (karşıtlık değişirme, aydınlık artırma, keskinleştirme), karşılaştırma D (karşıtlık değişirme, keskinleştirme, bulanıklaştırma) ve karşılaştırma F’deki (yakınlaştırma, renk değişirme, aydınlık artırma) yöntemler bir araya getirildiğinde tüm sınıflama değerleri % 90’ın üzerinde elde edilmiştir.

Burada en iyi sınıflandırma, karşılaştırma D’de karşıtlık değişirme, keskinleştirme ve bulanıklaştırma yöntemlerinin bir araya getirilmesi ile % 99,85 hassasiyet, % 99,82 özgüllük ve % 99,84 doğruluk ile gerçekleşmiştir.

Bu problem için, karşılaştırma C ve D'den görüldüğü gibi karşıtlık değiştirme ve keskinleştirme yöntemlerinin bir arada kullanılması sınıflama doğruluğunun sırası ile % 93.40 ve % 99.84 değerlerine yükselmesini sağlamıştır.

Karşılaştırma A'daki gibi aynalama + yakınlaştırma; E'deki gibi 3B döndürme + aynalama; G'deki gibi 3B döndürme + yakınlaştırma yöntemleri bir arada kullanıldığında ise sırası ile % 64.26, %78.76 ve % 70.88 doğruluk değerleri elde edilmiştir. Buradan aynalama, 3B döndürme ve yakınlaştırma yöntemlerinin bir arada kullanılmasının sınıflama performansını doğruluk olarak A'da % 9.29; G'de % 2.67 düşürdüğü ve E'de yalnızca % 5.21 artırdığı görülmektedir. Termal görüntülerin sınıflandırılmasında bu üç yöntemin bir arada kullanılması yerine karşıtlık değiştirme ve keskinleştirme yöntemleri ile bir arada kullanılmaları daha uygun olacaktır.

4.4 Transfer Öğrenme'nin Medikal Termografik Görüntülerin Sınıflandırılması Üzerindeki Etkileri

Bu çalışmada 19 hasta ve 19 sağlıklı bebeğe ait 3800 görüntü ön-eğitilmiş transfer öğrenme yöntemi kullanılarak sınıflandırılmıştır. Kolay ulaşılabilir ve yaygın kullanımlarından dolayı, eğitilmiş modellerden AlexNet, VGG16 ve VGG19 modelleri seçilmiştir. Bu modellerin mimarileri sırası ile Çizelge 4.10, 4.11 ve 4.12'de verilmiştir.

Çizelge 4.10. AlexNet mimarisi

1	Giriş Katmanı
2	Konvolüsyon
3	ReLU
4	Normalizasyon
5	Max Havuzlama
6	Konvolüsyon
7	ReLU
8	Normalizasyon
9	Max Havuzlama
10	Konvolüsyon
11	ReLU

12	Konvolüsyon
13	ReLU
14	Konvolüsyon
15	ReLU
16	Maksimum havuzlama
17	Tam bağı katman
18	ReLU
19	Unutturma
20	Tam bağı katman
21	ReLU
22	Unutturma
23	Tam bağı katman
24	Olasılık katmanı
25	Çıkış katmanı

Çizelge 4.11. VGG16 mimarisi

1	Giriş Katmanı	22	ReLU
2	Konvolüsyon	23	Konvolüsyon
3	ReLU	24	ReLU
4	Konvolüsyon	25	Max havuzlama
5	ReLU	26	Konvolüsyon
6	Max havuzlama	27	ReLU
7	Konvolüsyon	28	Konvolüsyon
8	ReLU	29	ReLU
9	Konvolüsyon	30	Konvolüsyon
10	ReLU	31	ReLU
11	Max havuzlama	32	Max havuzlama
12	Konvolüsyon	33	Tam bağı katman
13	ReLU	34	ReLU
14	Konvolüsyon	35	Unutturma
15	ReLU	36	Tam bağı katman

16	Konvolüsyon	37	ReLu
17	ReLU	38	Unutturma
18	Max havuzlama	39	Tam bağı katman
19	Konvolüsyon	40	Olasılık katmanı
20	ReLU	41	Çıkış katmanı
21	Konvolüsyon		

Çizelge 4.12. VGG19 mimarisi

1	Giriş Katmanı	25	Konvolüsyon
2	Konvolüsyon	26	ReLU
3	ReLU	27	Konvolüsyon
4	Konvolüsyon	28	ReLU
5	ReLU	29	Max havuzlama
6	Max havuzlama	30	Konvolüsyon
7	Konvolüsyon	31	ReLU
8	ReLU	32	Konvolüsyon
9	Konvolüsyon	33	ReLU
10	ReLU	34	Konvolüsyon
11	Max havuzlama	35	ReLU
12	Konvolüsyon	36	Konvolüsyon
13	ReLU	37	ReLU
14	Konvolüsyon	38	Max havuzlama
15	ReLU	39	Tam bağı katman
16	Konvolüsyon	40	ReLu
17	ReLU	41	Unutturma
18	Konvolüsyon	42	Tam bağı katman
19	ReLU	43	ReLU
20	Max havuzlama	44	Unutturma
21	Konvolüsyon	45	Tam bağı katman
22	ReLU	46	Olasılık katmanı
23	Konvolüsyon	47	Çıkış katmanı

24	ReLU
----	------

Bölüm 3.7.2’te açıklandığı gibi ön-eğitilmiş transfer öğrenme yönteminde özellik çıkarma işlemi önceden eğitilmiş modelin özellik çıkarma bölümlerindeki (konvolüsyon, aktivasyon, havuzlama) parametrelerin alınması ile yapılmakta ve bu özelliklerin sınıflandırılması için probleme özgü sınıflandırıcı yapı oluşturulmaktadır. Bu çalışmada sınıflandırıcı yapı olarak hem tam bağlı katman hem de DVM kullanılmıştır.

Kullanılan tam bağlı katmanın ilk katmanında 4096 nöron (aktivasyon = ReLU, unutturma = %50), ikinci katmanında hasta - sağlıklı sınıflandırılması için 2 nöron kullanılmıştır. Burada veri birinci nöronun çıkış değeri daha yüksek ise hasta sınıfına, ikinci nöronun çıkış değeri daha yüksekse sağlıklı sınıfına atanmıştır. DVM’de sınıflandırıcı kernel olarak doğrusal kernel kullanılmıştır.

10 – katlamalı çapraz doğrulama kullanılarak elde edilen karmaşıklık matrisleri hassasiyet, özgülük ve doğruluk kriterleri kullanılarak değerlendirilmiştir. Elde edilen karmaşıklık matrisleri Şekil 4.18’de ve değerlendirme sonuçları Çizelge 4.13’te verilmiştir.

1324	576	1697	203
859	1041	302	1598
(a)		(b)	
936	964	1531	369
1042	858	363	1537
(c)		(d)	
1140	760	1447	68
1140	760	49	141
(e)		(f)	

Şekil 4.18. Ön-eğitilmiş transfer öğrenme tekniği kullanılarak gerçekleştirilen sınıflamalardan elde edilen karmaşıklık matrisleri (a) AlexNet + tam bağlı katman (b) AlexNet + DVM (c) VGG16 + tam bağlı katman (d) VGG16 + DVM (e) VGG19 + tam bağlı katman (f) VGG19 + DVM

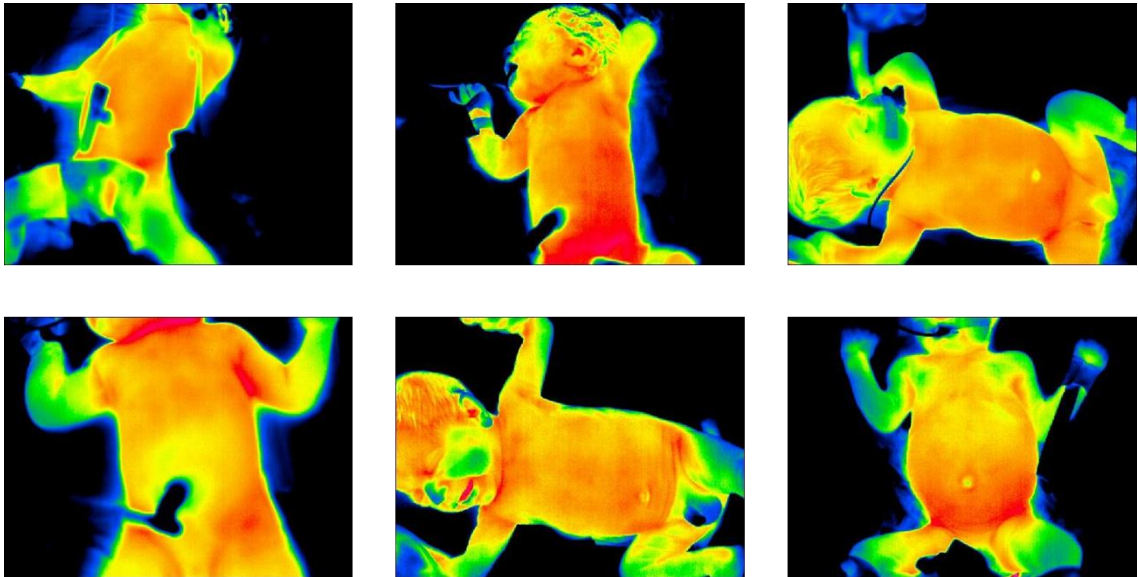
Çizelge 4.13. Transfer öğrenme kullanılarak elde edilen sonuçlar

Modeller	Sınıflandırıcılar					
	Tam Bağlı Katman			DVM		
	Hassasiyet (%)	Özgüllük (%)	Doğruluk (%)	Hassasiyet (%)	Özgüllük (%)	Doğruluk (%)
Alexnet	69.68	54.79	62.24	89.32	84.11	86.71
VGG16	49.26	45.16	47.21	80.58	80.89	80.74
VGG19	60.00	40.00	50.00	76.16	73.00	74.58

Elde edilen sonuçlar sınıflandırıcı yapı olarak DVM'nin tam bağlı katmandan daha iyi sınıflandırdığını göstermektedir. Çizelge 4.12'den görüldüğü gibi en iyi sınıflandırma % 89.32 hassasiyet, % 84.11 özgüllük ve % 86.71 doğruluk ile AlexNet + DVM kullanılarak elde edilmiştir.

4.5 Çekişmeli Üretici Ağlar ile Yeni Termal Görüntülerin Üretilmesi

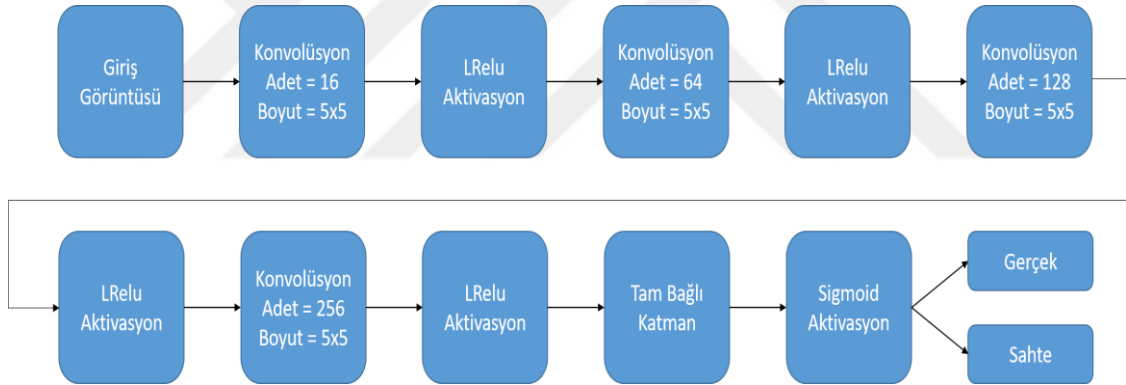
Bu çalışmada yeni doğan bebeklere ait 95 termal görüntü kullanılarak ÇÜA ile yeni termal görüntülerin üretilmesi gerçekleştirilmiştir. Veri seti içerisinde rastgele seçilen altı görüntü Şekil 4.19'da gösterilmiştir.

**Şekil 4.19.** Eğitim verisi içerisinde rastgele seçilmiş 6 görüntü

ÇÜA modelinin eğitim işleminde karşılaşılan en sık problem aşırı öğrenmedir. ÇÜA eğitim algoritmasında öncelikle ayırt edici ağın kendini bir kere eğittiği ve

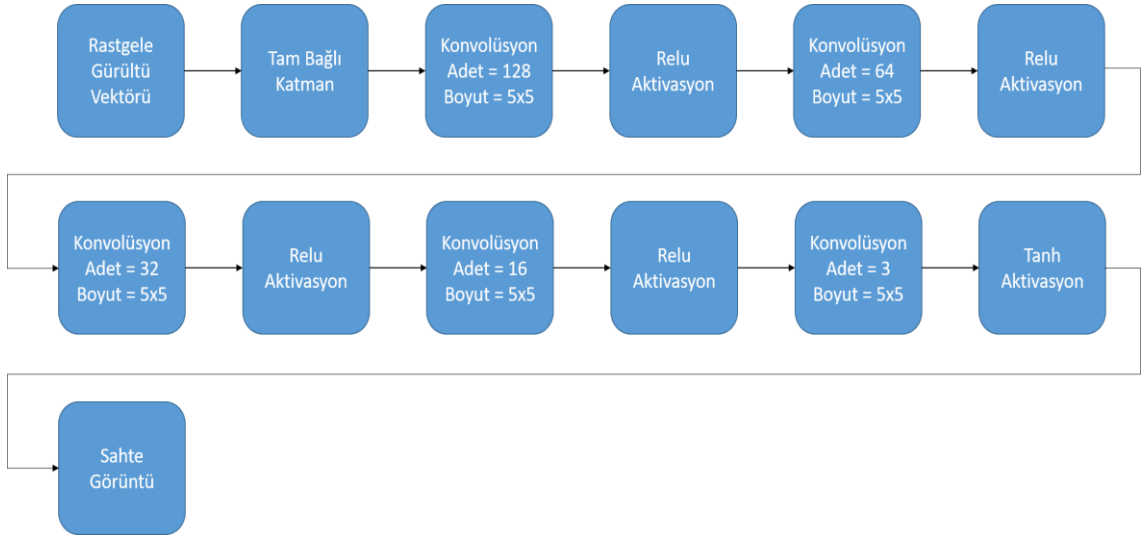
ardından üretici ağı kendini bir kere eğittiği Bölüm 3.7.3'te açıklanmıştı. Ancak uygulama esnasında birer kere eğitim gerçekleştirmek, ayırt edici ağı başlangıçta gerçek görüntüler ile sahte görüntüleri birbirinden ayırt etmesinin önüne geçmektedir. Bu nedenle ayırt edici ağı kendini üretici ağdan daha fazla eğitmesi gerekmektedir. Ancak ayırt edici ağ kendini üretici ağa göre çok fazla eğitirse bu kez de üretici ağ tarafından oluşturulan görüntüler direkt sahte olarak sınıflanmaya başlar ve bu kez de üretici ağı eğitileceği geri yayılım gerçekleşmez. Öyle bir denge yakalanmalıdır ki hem ayırt edici ağ kendini eğitirken üretici ağ devre dışı kalmamalı hem de üretici ağ ayırt edici ağı yanıltabilecek nitelikte görüntüler üretebilmelidir.

Ağların kendilerini eğitme sayılarının yanı sıra ağları oluşturan modellerin mimarileri de önemli rol oynamaktadır. Örneğin ESA mimarisi ile ağlar oluşturulduğunda konvolüsyon boyutu, aktivasyon tipi, havuzlama sayısı, filtre sayısı ve nöron sayısı gibi pek çok parametrenin seçimi de gerekmektedir. Bu çalışma için kullanılan ayırt edici ağ mimarisi Şekil 4.20'de gösterilmiştir.



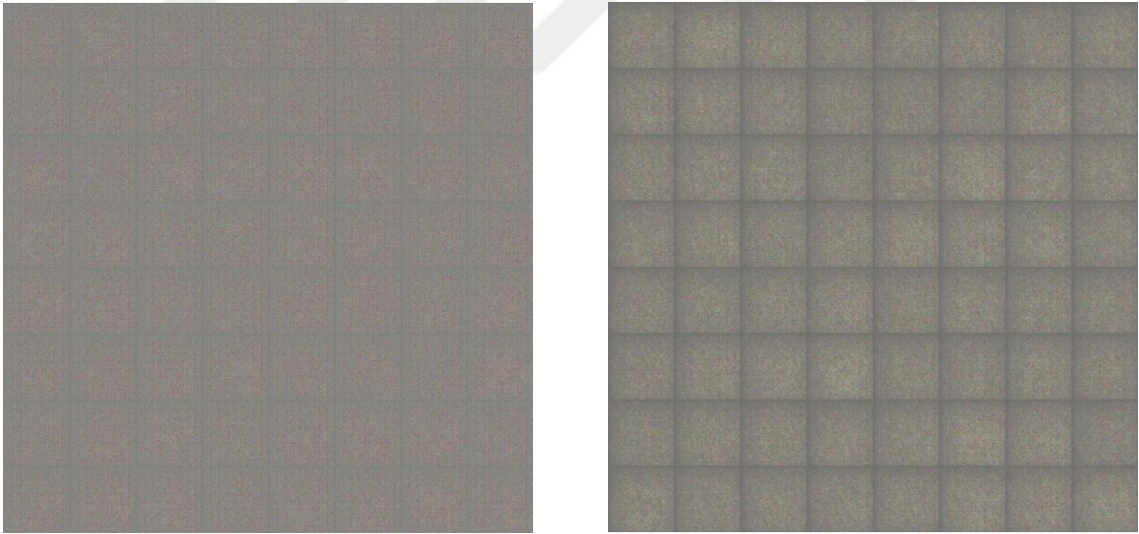
Şekil 4.20. Ayırt edici ağ mimarisi

ReLU aktivasyon fonksiyonu Bölüm 3.7.1.2'de açıklandığı gibi giriş değerlerinin yalnızca 0'dan büyük olanlarını geçiriyordu ($\text{relu} = \max(0, \text{giriş})$). LReLU ise giriş değeri ile giriş değerinin bir sızıntı miktarı ile çarpımlarını karşılaştırmakta ve büyük olan değeri geçirmektedir ($\text{lrelu} = \max(\text{giriş}, \text{giriş} * \text{sızıntı})$). Burada sızıntı miktarı deneysel olarak 0.2'dir. Kullanılan üretici mimarisi Şekil 4.21'de gösterilmiştir.



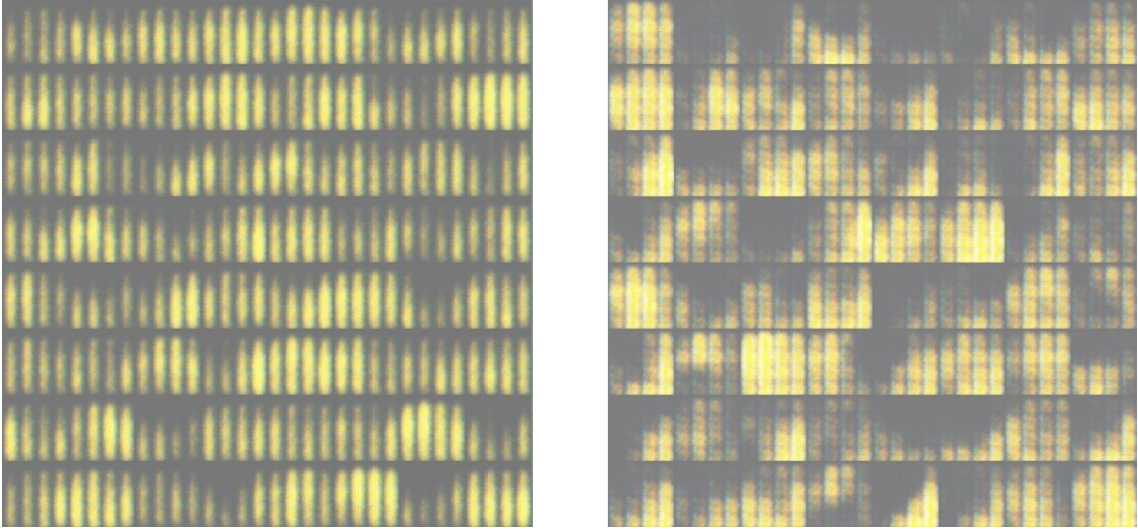
Şekil 4.21. Üretici ağ mimarisi

Ayırt edici ağın kendini beş kere üretici ağına ise kendini bir kere eğittiği durumda başlangıçtan itibaren oluşan görüntüler aşağıda gösterilmiştir. Şekil 4.22’de ilk on iterasyon boyunca oluşan görüntüler gösterilmiştir. Görüldüğü gibi ilk on iterasyonda yalnızca rastgele gürültülerden meydana gelen görüntüler oluşmuştur.



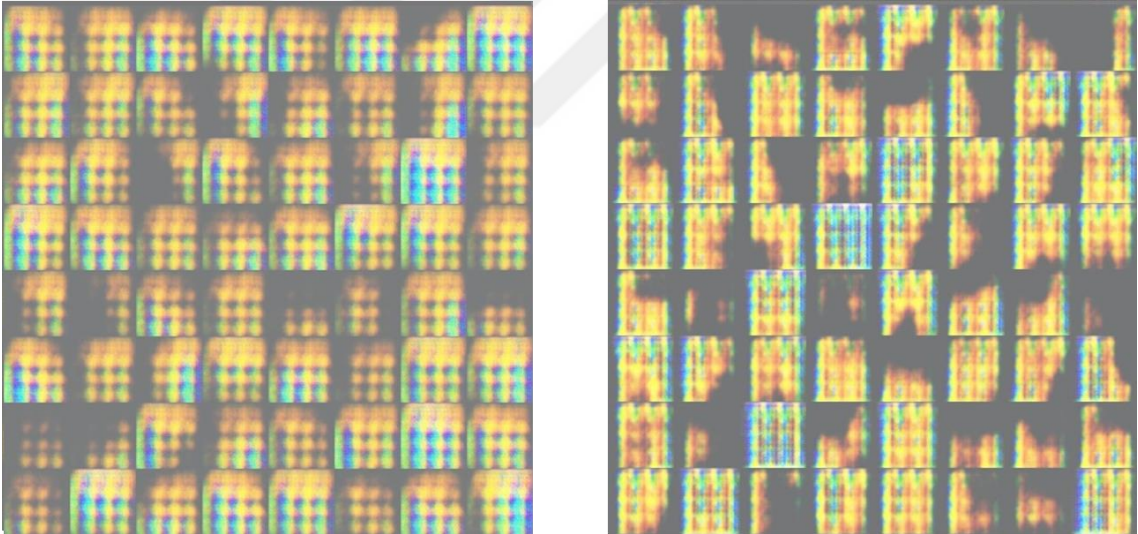
Şekil 4.22. İlk 10 iterasyon boyunca üretilen 128 görüntü (her kare 8x8 boyutunda 64 görüntü içermektedir)

İlk yüz iterasyona gelindiğinde rastgele gürültüler yerine Şekil 4.23’te görüldüğü gibi sarı ve yeşil tonların baskın olduğu fakat bir örüntü ifade etmeyen görüntüler meydana gelmiştir.



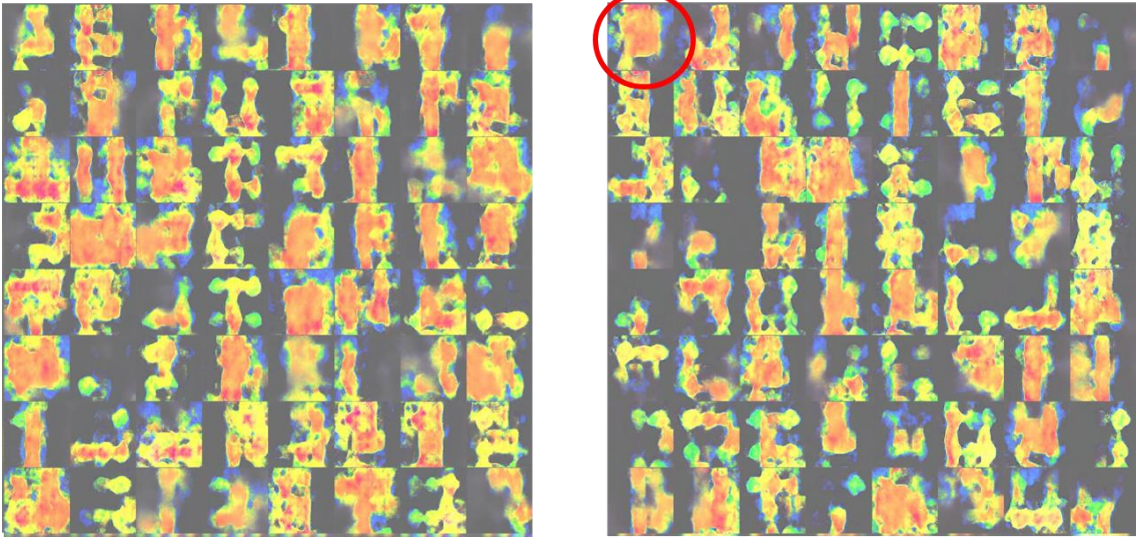
Şekil 4.23. 100 iterasyon boyunca üretilen 128 görüntü (her kare 8x8 boyutunda 64 görüntü içermektedir)

Bin iterasyona gelindiğinde ise ÇÜA artık termal görüntülerin renklerini öğrenmeye başlamış durumdadır. Şekil 4.24'ten görüldüğü gibi mavi, sarı ve yeşil tonları oluşmuştur.



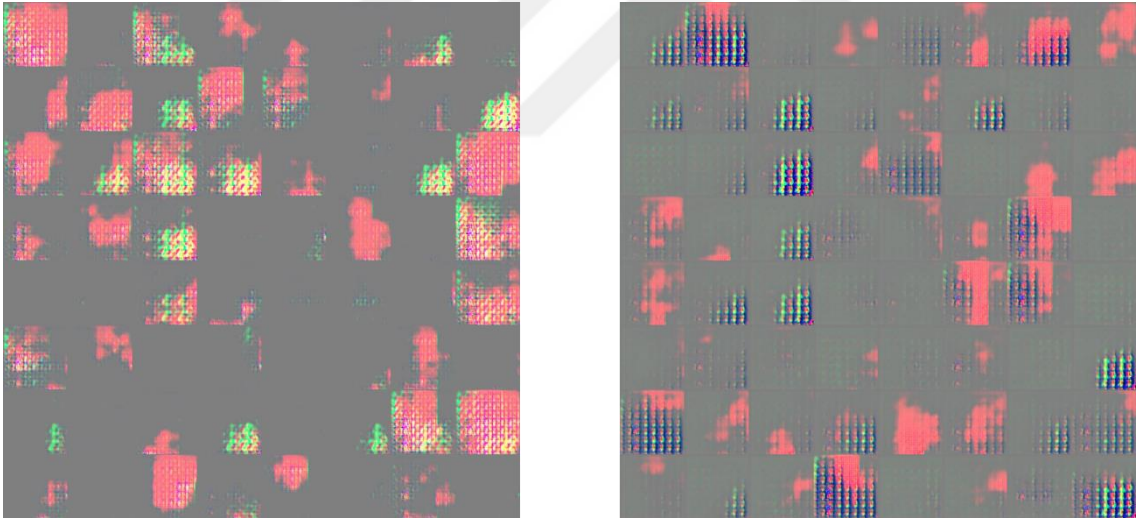
Şekil 4.24. 1000 iterasyon boyunca üretilen 128 görüntü (her kare 8x8 boyutunda 64 görüntü içermektedir)

On bin iterasyona gelindiğinde artık renkler iyice öğrenilmiş ve örüntüler oluşmaya başlamış durumdadır. Şekil 4.25'te kuvözde yatan yeni doğan bir bebeğe oldukça benzeyen bir görüntü işaretlenmiştir.

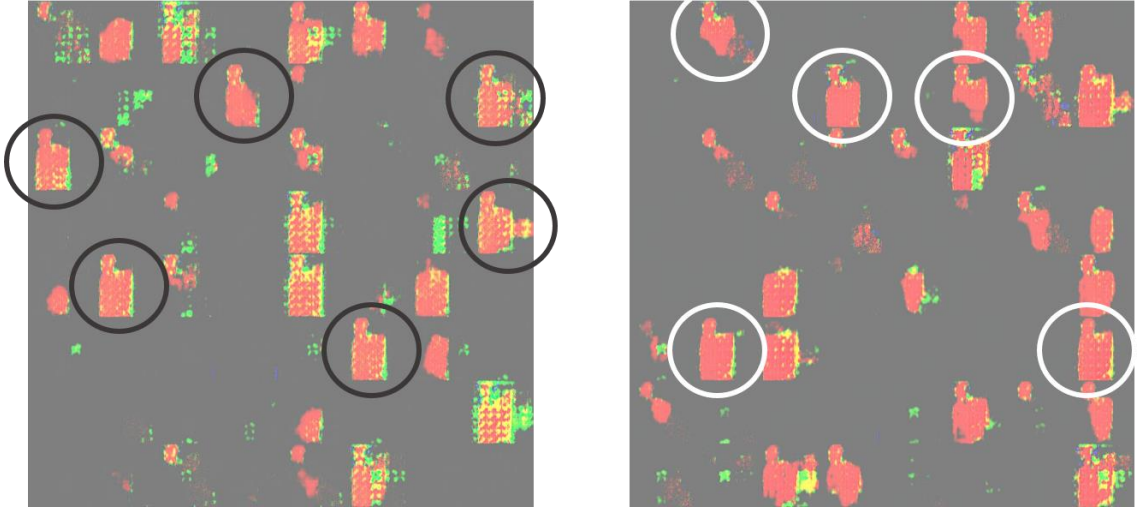


Şekil 4.25. 10000 iterasyon boyunca üretilen 128 görüntü (her kare 8x8 boyutunda 64 görüntü içermektedir)

ÇÜA mimarisi sabit tutularak ağların kendi eğitim sayıları değiştirildiğinde Şekil 4.26 ve 4.27'deki gibi görüntüler meydana gelmiştir.



Şekil 4.26. Ayırt edici ağın 7, üretici ağın 2 kere eğitilmesi durumunda 2200 iterasyon boyunca üretilen görüntüler (her kare 8x8 boyutunda 64 görüntü içermektedir)



Şekil 4.27. Ayırt edici ağı 9, üretici ağı 3 kere eğitilmesi durumunda 4700 iterasyon boyunca üretilen görüntüler (her kare 8x8 boyutunda 64 görüntü içermektedir)

Şekil 4.26 ve Şekil 4.27'den görüldüğü gibi ayırt edici ağı ve üretici ağı eğitim sayıları değiştirildiği durumlarda aşırı öğrenme meydana gelmiştir. Şekillerde işaretlenen yerlere bakıldığında hep aynı örüntünün üretilmeye çalışıldığı görülecektir.

Bu çalışma ile beraber, ÇÜA kullanılarak yeni doğan bebeklere ait termogramları üretilebileceği gösterilmiştir. Üretilen görüntüler kullanılarak, görüntülerin sınıflama performansı ve hastalık tespiti üzerine etkisi araştırmaya açık konulardır.

5 SONUÇLAR VE ÖNERİLER

5.1 Sonuçlar

Bu tez çalışması ile birlikte termal görüntülerin makine öğrenimi ve derin öğrenme yöntemleri ile sınıflandırılması gerçekleştirilmiştir.

Sıcaklık haritaları ve RGB görüntüler kullanılarak makine öğrenimi algoritmaları (yapay sinir ağları, lojistik regresyon, karar ağaçları ve rastgele orman) ile sınıflama gerçekleştirilmeden önce geliştirilen bölütleme algoritması ile görüntülerdeki termal farklılıklarının baskın hale getirilmesi, yerel ikili desen ile görüntülerden özellik çıkarılması ve hızlı korelasyon tabanlı filtre ile özellik seçimi işlemleri gerçekleştirilmiştir. Elde edilen sonuçlar bölütleme ve özellik seçiminin sınıflama üzerindeki önemini göstermiştir. Bölütleme ve özellik seçimi kullanmadan doğruluk değeri maksimum % 55 (RGB+yapay sinir ağı, RGB+lojistik regresyon) elde edilirken, bölütleme uygulamadan özellik seçimi kullandığında doğruluk değeri % 72.5'e (RGB+yapay sinir ağı, RGB+rastgele orman, Sıcaklık Haritaları+lojistik regresyon) yükselmiştir. En iyi sınıflama sıcaklık haritaları, bölütleme, özellik seçimi ve yapay sinir ağı kullanılarak % 92.5 doğruluk değeri ile elde edilmiştir.

Derin öğrenme ile sınıflama gerçekleştirileceği zaman çok sayıda görüntüye ihtiyaç duyulmaktadır. Bu nedenle elde bulunan görüntülere parlaklık artırma, kontrast değiştirme, çözünürlük değiştirme, renk dönüştürme ve üç farklı yoğunlukta tuz – biber gürültüsü ekleme gibi veri artırımı yöntemleri uygulanarak toplamda 3800 görüntüden 15200 ve 30400 görüntü elde edilmiştir. 3800 görüntü kullanarak ESA ile sınıflama gerçekleştirildiğinde % 70.73 doğruluk elde edilmiştir. Parlaklık artırma, kontrast değiştirme ve tuz – biber gürültüsü ekleme ile görüntü sayısı 15200'e çıkarıldığında hassaslık, özgüllük ve doğruluk kriterlerinin % 99'un üzerine çıktığı görülmüştür (hassaslık=% 99.06, özgüllük % 99.01 ve doğruluk % 99.07). Çözünürlük değiştirme ve iki farklı yoğunlukta tuz – biber gürültüsü ekleme yöntemleri de eklenerek 30400 görüntü ESA ile sınıflandırıldığında 15200 sağlıklı sınıfına ait görüntünün 15114'u sağlıklı ve 15200 hasta sınıfına ait görüntünün 15159'ü hasta olarak sınıflandırılmıştır. Böylece %99.58 doğruluk, %99.73 özgüllük ve %99.43 hassasiyet elde edilmiştir.

Geleneksel veri artırma yöntemlerinin ESA ile sınıflama üzerine etkilerinin incelenmesi için renk değiştirme, aynalama, rotasyon değiştirme, 3B döndürme, yakınlaştırma, histogram eşitleme, keskinleştirme, aydınlık artırma, bulanıklaştırma ve

kontrast deęiřtirme gibi geleneksel veri artırma yöntemleri kullanılmıřtır. Bu on yöntem üçerli halde birleřtirilerek on farklı 15200 görüntü seti elde edilmiř ve tüm veriler ESA ile sınıflandırılmıřtır. Elde edilen sonuçlar; aynalama, 3B döndürme ve yakınlařtırma tekniklerinin bir arada kullanımının sınıflama performansını azalttıęını; karřıtlık deęiřtirme ve keskinleřtirme tekniklerinin bir arada kullanımının ise sınıflandırma performansını yükselttięi göstermiřtir. Örneęin rotasyon deęiřtirme, yakınlařtırma ve aynalama kullanıldıęında % 69.77 hassasiyet, % 58.76 özgülük ve % 64.26 doęruluk deęerleri elde edilirken; bulanıklařtırma, keskinleřtirme ve karřıtlık deęiřtirme kullanılarak % 99,85 hassasiyet, % 99,82 özgülük ve % 99,84 doęruluk elde edilmiřtir.

Veri artırımı kullanmadan da derin öğrenme ile sınıflama gerçekleřtirmek için ön-eęitimli transfer öğrenme yöntemi kullanılmıřtır. Böylece Alexnet, VGG16 ve VGG19 gibi hazır modellerin eęitilmiř konvolüsyonel aęırlıkları alınarak özellik çıkarma iřlemi gerçekleřtirilmiřtir. Çıkartılan özelliklerin sınıflandırılmasında hem tam-baęlı katman hem de DVM kullanılmıřtır. Elde edilen sonuçlar DVM'nin tam baęlı katmandan daha iyi sınıflandırdıęını göstermektedir. En iyi sınıflandırma % 86.71 doęruluk deęeri ile AlexNet + DVM kullanılarak elde edilmiřtir.

5.2 Öneriler

Tez çalışması kapsamında, makine öğrenimi ve derin öğrenme yöntemleri kullanılarak yeni doğan bebeklere ait termal görüntülerin sınıflandırılması ve derin öğrenme yöntemlerinden çekişmeli üretici ağlar kullanılarak termal görüntülerin artırılması gerçekleştirilmiştir. Tüm bu çalışmalar, bebeklerde meydana gelebilecek hastalıkların erken teşhis edilmesi ve sağlık durumlarının takip edilmesi üzerine oluşturulacak gerçek zamanlı ve kararlı bir sistem için büyük önem arz etmektedir.

Görüntü alma işlemi boyunca kullanılan termal kamera bir dakika içerisinde yüz görüntü elde ediyordu. Sınıflandırma aşamasında, görüntü sayısını artırmak ve görüntü çeşitliliğini sağlamak için, sabit bir düzenek ile 20-30 görüntü aldıktan sonra bebek serbest bırakılarak ve kamera hareket ettirilerek farklı açılardan ve farklı mesafelerden görüntü alınmalıdır. Böylece hem mevcut makine öğrenimi ve derin öğrenme yöntemleri ile analiz hem de ileride geliştirilecek yeni yöntemler ile analiz için hazır veri setleri oluşmuş olacaktır.

Makine öğrenimi ile görüntü sınıflama gerçekleştirildiğinde, görüntülerden özellik çıkarmak için yerel ikili desen algoritması yerine ripple ve ridgelet gibi çoklu çözünürlük analizi metotları da kullanılmalıdır. Böylece özellik çıkarmanın yanı sıra boyut azaltma ve gürültü giderme gibi uygulamalar da gerçekleştirilebilir.

Özellik seçimi, hem sınıflama karmaşıklığını azaltması hem de sınıflama performansını yükseltmesi bakımından önemli bir işlemdir. Hızlı korelasyon tabanlı filtre yerine, gömülü ve sarmal özellik seçme yöntemlerinin etkileri de incelenmelidir.

Derin öğrenme yöntemlerinden evrişimli sinir ağları ile sınıflama gerçekleştirileceğinde ise konvolüsyon, aktivasyon fonksiyonu, havuzlama ve tam bağlı katman dikkatlice seçilmelidir. Konvolüsyon ve havuzlama katmanlarından her biri eklenmeden önce, oluşacak özellik haritasının boyutu hesaplanmalıdır. Aksi takdirde farkında olmadan, sınıflanacak görüntüye ait tüm bilgiler tam bağlı katmana gelene kadar kaybolabilir. Konvolüsyon ve havuzlama hesaplandıktan sonra tam bağlı katman için nöronlar yerleştirilmelidir. Nöronlar arasına unutturma katmanı eklemek aşırı öğrenmenin önüne geçmek için iyi bir çözümdür. Tüm bunlar yapıldıktan sonra sigmoid, tanh ve relu gibi aktivasyon fonksiyonları katmanlar arasına yerleştirilerek iyi bir sonuç elde edilene kadar program çalıştırılmalıdır.

KAYNAKLAR

- Abbas, A. K. ve Leonhardt, S., Neonatal IR-Thermography Pattern Clustering based on ICA Algorithm.
- Abbas, A. K., Heimann, K., Blazek, V., Orlikowsky, T. ve Leonhardt, S., 2012, Neonatal infrared thermography imaging: analysis of heat flux during different clinical scenarios, *Infrared Physics & Technology*, 55 (6), 538-548.
- Abbas, A. K. ve Leonhardt, S., 2014, Intelligent neonatal monitoring based on a virtual thermal sensor, *BMC medical imaging*, 14 (1), 9.
- Alpaydin, E., 2009, Introduction to machine learning, MIT press, p.
- Arora, N., Martins, D., Ruggerio, D., Tousimis, E., Swistel, A. J., Osborne, M. P. ve Simmons, R. M., 2008, Effectiveness of a noninvasive digital infrared thermal imaging system in the detection of breast cancer, *The American Journal of Surgery*, 196 (4), 523-526.
- Bagavathiappan, S., Saravanan, T., Philip, J., Jayakumar, T., Raj, B., Karunanithi, R., Panicker, T. M., Korath, P. ve Jagadeesan, K., 2008, Investigation of peripheral vascular disorders using thermal imaging, *The British Journal of Diabetes & Vascular Disease*, 8 (2), 102-104.
- Bagavathiappan, S., Saravanan, T., Philip, J., Jayakumar, T., Raj, B., Karunanithi, R., Panicker, T., Korath, M. P. ve Jagadeesan, K., 2009, Infrared thermal imaging for detection of peripheral vascular disorders, *Journal of medical physics/Association of Medical Physicists of India*, 34 (1), 43.
- Breiman, L., 1999, Random forests, *UC Berkeley TR567*.
- Ceylan, M., 2004, Kompleks değerli yapay sinir ağı ile algoritma geliştirilmesi ve uygulanması, Yüksek Lisans, *Selçuk Üniversitesi, Fen bilimleri Enstitüsü, Konya*.
- Christidis, I., Zotter, H., Rosegger, H., Engele, H., Kurz, R. ve Kerbl, R., 2003, Infrared thermography in newborns: the first hour after birth, *Gynäkologisch-geburtshilfliche Rundschau*, 43 (1), 31-35.
- Clark, R. ve Stothers, J., 1980, Neonatal skin temperature distribution using infra-red colour thermography, *The Journal of physiology*, 302 (1), 323-333.
- Cortes, C. ve Vapnik, V., 1995, Support-vector networks, *Machine learning*, 20 (3), 273-297.
- Cowan, J. D. ve Sharp, D. H., 1988, Neural nets and artificial intelligence, *Daedalus*, 85-121.
- Danzl, D. F. ve Pozos, R. S., 1994, Accidental hypothermia, *New England Journal of Medicine*, 331 (26), 1756-1760.
- Dash, M. ve Liu, H., 1997, Feature selection for classification, *Intelligent data analysis*, 1 (1-4), 131-156.
- DeLong, E. R., DeLong, D. M. ve Clarke-Pearson, D. L., 1988, Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach, *Biometrics*, 44 (3), 837-845.
- Deng, L. ve Yu, D., 2014, Deep learning: methods and applications, *Foundations and Trends® in Signal Processing*, 7 (3-4), 197-387.
- Deshpande, A., 2016, A Beginner's Guide To Understanding Convolutional Neural Networks, <https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/>: [28.04.2019].

- Dietterich, T. G., 2000, An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization, *Machine learning*, 40 (2), 139-157.
- Drakos, G., 2018, Cross-Validation, <https://towardsdatascience.com/cross-validation-70289113a072>: [28.04.2019].
- Goetz, C., Foertsch, D., Schoenberger, J. ve Uhl, E., 2005, Thermography - a valuable tool to test hydrocephalus shunt patency, 11, p.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. ve Bengio, Y., 2014, Generative adversarial nets, *Advances in neural information processing systems*, 2672-2680.
- Govindarajan, J., 2007, A Case for Joint Development of IR Cameras in India, *Journal on Intelligent Electronic Systems*, 1 (1).
- Group, W. B., 2017, Mortality rate, neonatal (per 1,000 live births), <https://data.worldbank.org/indicator/SH.DYN.NMRT?locations=IN>: [November 12, 2018].
- Gunjal, B. L. ve Mali, S. N., 2011, Comparative performance analysis of DWT-SVD based color image watermarking technique in YUV, RGB and YIQ color spaces, *International Journal of Computer Theory and Engineering*, 3 (6), 714.
- Heimann, K., Jergus, K., Abbas, A. K., Heussen, N., Leonhardt, S. ve Orlikowsky, T., 2013, Infrared thermography for detailed registration of thermoregulation in premature infants, p.
- Hildebrandt, C., Zeilberger, K., Ring, E. F. J. ve Raschner, C., 2012, The application of medical infrared thermography in sports medicine, In: An international perspective on topics in sports medicine and sports injury, Eds: IntechOpen, p.
- Huynh-Thu, Q. ve Ghanbari, M., 2008, Scope of validity of PSNR in image/video quality assessment, *Electronics letters*, 44 (13), 800-801.
- Kammersgaard, T. S., Malmkvist, J. ve Pedersen, L. J., 2013, Infrared thermography—a non-invasive tool to evaluate thermal status of neonatal pigs based on surface temperature, *Animal*, 7 (12), 2026-2034.
- Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M. ve Klein, M., 2002, Logistic regression, Springer, p.
- Knobel-Dail, R. B., Holditch-Davis, D., Sloane, R., Guenther, B. ve Katz, L. M., 2017, Body temperature in premature infants during the first week of life: Exploration using infrared thermal imaging, *Journal of thermal biology*, 69, 118-123.
- Kohavi, R., 1995, A study of cross-validation and bootstrap for accuracy estimation and model selection, *Ijcai*, 1137-1145.
- Kontos, M., Wilson, R. ve Fentiman, I., 2011, Digital infrared thermal imaging (DITI) of breast lesions: sensitivity and specificity of detection of primary breast cancers, *Clinical radiology*, 66 (6), 536-539.
- Krizhevsky, A., Sutskever, I. ve Hinton, G. E., 2012, Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, 1097-1105.
- Kruse, P. W., 2001, Uncooled thermal imaging: arrays, systems, and applications, SPIE press Bellingham, WA, p.
- Lee, D.-S., 2017, Improved Activation Functions of Deep Convolutional Neural Networks for Image Classification.
- Lopez, M. B., del-Blanco, C. R. ve Garcia, N., 2017, Detecting exercise-induced fatigue using thermal imaging and deep learning, *2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA)*, 1-6.

- McCulloch, W. S. ve Pitts, W., 1943, A logical calculus of the ideas immanent in nervous activity, *The bulletin of mathematical biophysics*, 5 (4), 115-133.
- Mikołajczyk, A. ve Grochowski, M., 2018, Data augmentation for improving deep learning in image classification problem, *2018 international interdisciplinary PhD workshop (IIPhDW)*, 117-122.
- Myers, R. H. ve Myers, R. H., 1990, Classical and modern regression with applications, Duxbury press Belmont, CA, p.
- Ng, W. K., Ng, Y. K. ve Tan, Y. K., 2009, Qualitative study of sexual functioning in couples with erectile dysfunction: prospective evaluation of the thermography diagnostic system, *The Journal of reproductive medicine*, 54 (11-12), 698-705.
- Nilsson, N. J., 2014, Principles of artificial intelligence, Morgan Kaufmann, p.
- Nur, R., 2014, Identification of thermal abnormalities by analysis of abdominal infrared thermal images of neonatal patients, *Carleton University*.
- Ojala, T., Pietikäinen, M. ve Mäenpää, T., 2002, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on Pattern Analysis & Machine Intelligence* (7), 971-987.
- Otsu, N., 1979, A threshold selection method from gray-level histograms, *IEEE transactions on systems, man, and cybernetics*, 9 (1), 62-66.
- Örnek, A. H., Savaşçı, D., Ceylan, M., Ervural, S. ve Soylu, H., 2018, Termografların Değerlendirilmesinde Doğru Yaklaşımların Belirlenmesi. URSTÜRKİYE'2018 IX. Bilimsel Kongresi, 6-8 Eylül 2018. KTO Karatay Üniversitesi, Konya.
- Öztürk, A. E., 2015, Ripplet, Tetrolet ve Ridgelet dönüşümleri kullanılarak karaciğer fokal lezyonlarının belirlenmesi, Yüksek Lisans, *Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya*.
- Öztürk, A. E., Ceylan, M. ve Kıvrak, A. S., 2015, A new approach for liver classification using ridgelet/ripplet-II transforms, feature groups and ANN, *6th European Conference of the International Federation for Medical and Biological Engineering*, 130-133.
- Pampel, F. C., 2000, Logistic regression: A primer, Sage, p.
- Pan, S. J. ve Yang, Q., 2010, A survey on transfer learning, *IEEE Transactions on knowledge and data engineering*, 22 (10), 1345-1359.
- Qian, X., Hua, X.-S., Chen, P. ve Ke, L., 2011, PLBP: An effective local binary patterns texture descriptor with pyramid representation, *Pattern Recognition*, 44 (10-11), 2502-2515.
- Rice, H. E., Hollingsworth, C. L., Bradsher, E., Danko, M. E., Crosby, S., Goldberg, R. N., Tanaka, D. T. ve Knobel, R. B., 2010, Infrared thermal imaging (thermography) of the abdomen in extremely low birthweight infants, *J Surg Radiol*, 1 (2), 61-122.
- Safavian, S. R. ve Landgrebe, D., 1991, A survey of decision tree classifier methodology, *IEEE transactions on systems, man, and cybernetics*, 21 (3), 660-674.
- Savaşçı, D. ve Ceylan, M., 2018, Thermal image analysis for neonatal intensive care units (First evaluation results), *2018 26th Signal Processing and Communications Applications Conference (SIU)*, 1-4.
- Sebastiani, F., 2002, Machine learning in automated text categorization, *ACM computing surveys (CSUR)*, 34 (1), 1-47.
- Soille, P., 2013, Morphological image analysis: principles and applications, Springer Science & Business Media, p.

- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. ve Salakhutdinov, R., 2014, Dropout: a simple way to prevent neural networks from overfitting, *The Journal of Machine Learning Research*, 15 (1), 1929-1958.
- Strigl, D., Kofler, K. ve Podlipnig, S., 2010, Performance and scalability of GPU-based convolutional neural networks, *2010 18th Euromicro Conference on Parallel, Distributed and Network-based Processing*, 317-324.
- Turing, A. M., 1950, Can a machine think, *Mind*, 59 (236), 433-460.
- Usamentiaga, R., García, D. F., Molleda, J., Bulnes, F. G. ve Orgeira, V. G., 2014, Temperature tracking system for sinter material in a rotatory cooler based on infrared thermography, *IEEE Transactions on Industry Applications*, 50 (5), 3095-3102.
- Vadivambal, R. ve Jayas, D. S., 2011, Applications of thermal imaging in agriculture and food industry—a review, *Food and Bioprocess Technology*, 4 (2), 186-199.
- Villarroel, M., Guazzi, A., Jorge, J., Davis, S., Watkinson, P., Green, G., Shenvi, A., McCormick, K. ve Tarassenko, L., 2014, Continuous non-contact vital sign monitoring in neonatal intensive care unit, *Healthcare technology letters*, 1 (3), 87-91.
- Wang, Z., Simoncelli, E. P. ve Bovik, A. C., 2003, Multiscale structural similarity for image quality assessment, *The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, 1398-1402.
- WEKA, 2019a, Logistic Regression, <http://weka.sourceforge.net/doc.stable/weka/classifiers/functions/Logistic.html>: [June 9, 2019].
- WEKA, 2019b, J48, <http://weka.sourceforge.net/doc.stable/weka/classifiers/trees/J48.html>: [June 9, 2019].
- WEKA, 2019c, Random Forest, <http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/RandomForest.html>: [June 9, 2019].
- Yaşar, H., 2015, Medikal görüntülerin çoklu çözünürlük metotları ile analizi Yüksek Lisans, *Selçuk Üniversitesi Fen Bilimleri Enstitüsü, Konya*.
- Yu, L. ve Liu, H., 2003, Feature selection for high-dimensional data: A fast correlation-based filter solution, *Proceedings of the 20th international conference on machine learning (ICML-03)*, 856-863.
- Zhu, W., Zeng, N. ve Wang, N., 2010, Sensitivity, specificity, accuracy, associated confidence interval and ROC analysis with practical SAS implementations, *NESUG proceedings: health care and life sciences, Baltimore, Maryland*, 19, 67.

EKLER**EK-1 Tez Çalışmasında Kullanılan Termal Görüntülerin Alınması için Etik Kurul Kararı**

T.C.
SELÇUK ÜNİVERSİTESİ
TIP FAKÜLTESİ DEKANLIĞI

GİRİŞİMSEL OLMAYAN KLİNİK ARAŞTIRMALAR ETİK KURULU KARARLARI

Toplantı Sayısı: 2015/1

Toplantı Tarihi : 06.01.2015

Karar Sayısı 2015/16 S.Ü. Mühendislik Fakültesi Elektrik-Elektronik Mühendisliği Anabilim Dalı Öğretim Üyesi Yrd.Doç.Dr. Murat CEYLAN'ın, "Termal Görüntüleme Tabanlı Yeni Doğan Destek Ünitesi" başlıklı araştırmasının değerlendirilme talebi ile ilgili 25.12.2014 tarihli dilekçesi ve ekleri görüşüldü.

Yapılan inceleme ve görüşmelerden sonra; Yrd.Doç.Dr. Murat CEYLAN'ın, "Termal Görüntüleme Tabanlı Yeni Doğan Destek Ünitesi" adlı araştırmanın kabulüne, TÜBİTAK desteği alındıktan sonra protokolün dosyaya ilave edilmek üzere Etik Kurul sekreteryasına teslim edilmesine oy birliği ile karar verildi.

ASLI GİBİDİR

06/01/2015


KESKİ
Sekreteryası

EK-2 RGB Görüntüler ve Sıcaklık Haritalarının ADD Kullanılarak Detay ve Yaklaşım Katsayılarının Elde Edilmesi ve Yeniden Çatılması (Bölüm 4.1) MATLAB Kodları

```

clear;
clc;
close all;

disp('db2, db3, db4, haar, sym2, sym3, sym4..')
disp('coif2, coif3, coif4, bior1.1.. ')
disp('bior1.3, bior2.2, bior3.3, bior3.5..')
disp('_____')
wname=input('Dalgacık Türünü Giriniz = ', 's');

%termal katsayıları içeren görüntülerin okunması ve
%0-1 arasında normalize edilmesi
load ('haydar_veriler_double.mat')
h1t=mat2gray(hastaakbalikgovde180217001);
h2t=mat2gray(hastaaasisikgovde18021730);
h3t=mat2gray(hastakahramangovde180303001);
h4t=mat2gray(hastakergegovde180224001);
h5t=mat2gray(hastateke2govde180303001);

p1t=mat2gray(prmcandangovde1803032);
p2t=mat2gray(prmelhuseyin1govde18030310);
p3t=mat2gray(prmsoylemezgovde180217001);
p4t=mat2gray(prmugurgovde4001);
p5t=mat2gray(prmustundaggovde180217001);
%%
%RGB görüntülerin okunması, gray formatına dönüştürülmesi
%ve 0-1 arasında normalize edilmesi
h1n=mat2gray(rgb2gray(imread('hasta_akbalik_govde_180217_001.jpg')));
h2n=mat2gray(rgb2gray(imread('hasta_aras_isik_govde_180217_30.jpg')));
h3n=mat2gray(rgb2gray(imread('hasta_kahraman_govde_180303_001.jpg')));
h4n=mat2gray(rgb2gray(imread('hasta_kerge_govde_180224_001.jpg')));
h5n=mat2gray(rgb2gray(imread('hasta_teke2_govde_180303_001.jpg')));
p1n=mat2gray(rgb2gray(imread('prm_candan_govde_180303_2.jpg')));
p2n=mat2gray(rgb2gray(imread('prm_el_huseyin1_govde_180303_10.jpg')));
p3n=mat2gray(rgb2gray(imread('prm_soylemez_govde_180217_001.jpg')));
p4n=mat2gray(rgb2gray(imread('prm_ugur_govde_4_001.jpg')));
p5n=mat2gray(rgb2gray(imread('prm_ustundag_govde_180217_001.jpg')));
%%
%tüm görüntülerin aynı boyuta getirilmesi (imresize)
h1t=imresize(h1t,[362 482]);
h2t=imresize(h2t,[362 482]);
h3t=imresize(h3t,[362 482]);
h4t=imresize(h4t,[362 482]);
h5t=imresize(h5t,[362 482]);
p1t=imresize(p1t,[362 482]);
p2t=imresize(p2t,[362 482]);
p3t=imresize(p3t,[362 482]);
p4t=imresize(p4t,[362 482]);
p5t=imresize(p5t,[362 482]);
%%
%dalgaçık dönüşümü
%CA1 - CA5 arası hasta bebekler
%CA6 - CA10 arası sağlıklı bebekler
%CA1n - CA10n RGB görüntüler
%CA1t - CA10t TERMAL içerikli görüntüler
[CA1n,CH1n,CV1n,CD1n] = dwt2(h1n,wname,'mode','per');
[CA2n,CH2n,CV2n,CD2n] = dwt2(h2n,wname,'mode','per');

```

```

[CA3n,CH3n,CV3n,CD3n] = dwt2(h3n,wname,'mode','per');
[CA4n,CH4n,CV4n,CD4n] = dwt2(h4n,wname,'mode','per');
[CA5n,CH5n,CV5n,CD5n] = dwt2(h5n,wname,'mode','per');

[CA6n,CH6n,CV6n,CD6n] = dwt2(p1n,wname,'mode','per');
[CA7n,CH7n,CV7n,CD7n] = dwt2(p2n,wname,'mode','per');
[CA8n,CH8n,CV8n,CD8n] = dwt2(p3n,wname,'mode','per');
[CA9n,CH9n,CV9n,CD9n] = dwt2(p4n,wname,'mode','per');
[CA10n,CH10n,CV10n,CD10n] = dwt2(p5n,wname,'mode','per');

[CA1t,CH1t,CV1t,CD1t] = dwt2(h1t,wname,'mode','per');
[CA2t,CH2t,CV2t,CD2t] = dwt2(h2t,wname,'mode','per');
[CA3t,CH3t,CV3t,CD3t] = dwt2(h3t,wname,'mode','per');
[CA4t,CH4t,CV4t,CD4t] = dwt2(h4t,wname,'mode','per');
[CA5t,CH5t,CV5t,CD5t] = dwt2(h5t,wname,'mode','per');

[CA6t,CH6t,CV6t,CD6t] = dwt2(p1t,wname,'mode','per');
[CA7t,CH7t,CV7t,CD7t] = dwt2(p2t,wname,'mode','per');
[CA8t,CH8t,CV8t,CD8t] = dwt2(p3t,wname,'mode','per');
[CA9t,CH9t,CV9t,CD9t] = dwt2(p4t,wname,'mode','per');
[CA10t,CH10t,CV10t,CD10t] = dwt2(p5t,wname,'mode','per');

%geri çatma (RGB görüntüler)
Z1n=(idwt2(CA1n,CH1n,CV1n,CD1n,'db2'));
Z2n=(idwt2(CA2n,CH2n,CV2n,CD2n,'db2'));
Z3n=(idwt2(CA3n,CH3n,CV3n,CD3n,'db2'));
Z4n=(idwt2(CA4n,CH4n,CV4n,CD4n,'db2'));
Z5n=(idwt2(CA5n,CH5n,CV5n,CD5n,'db2'));
Z6n=(idwt2(CA6n,CH6n,CV6n,CD6n,'db2'));
Z7n=(idwt2(CA7n,CH7n,CV7n,CD7n,'db2'));
Z8n=(idwt2(CA8n,CH8n,CV8n,CD8n,'db2'));
Z9n=(idwt2(CA9n,CH9n,CV9n,CD9n,'db2'));
Z10n=(idwt2(CA10n,CH10n,CV10n,CD10n,'db2'));
%geri çatma (termal görüntüler)
Z1t=(idwt2(CA1t,CH1t,CV1t,CD1t,'db2'));
Z2t=(idwt2(CA2t,CH2t,CV2t,CD2t,'db2'));
Z3t=(idwt2(CA3t,CH3t,CV3t,CD3t,'db2'));
Z4t=(idwt2(CA4t,CH4t,CV4t,CD4t,'db2'));
Z5t=(idwt2(CA5t,CH5t,CV5t,CD5t,'db2'));
Z6t=(idwt2(CA6t,CH6t,CV6t,CD6t,'db2'));
Z7t=(idwt2(CA7t,CH7t,CV7t,CD7t,'db2'));
Z8t=(idwt2(CA8t,CH8t,CV8t,CD8t,'db2'));
Z9t=(idwt2(CA9t,CH9t,CV9t,CD9t,'db2'));
Z10t=(idwt2(CA10t,CH10t,CV10t,CD10t,'db2'));
%% değerlendirme
mm=360; nn=480;
Z1n=Z1n(1:mm,1:nn);
h1n=h1n(1:mm,1:nn);
Z6n=Z6n(1:mm,1:nn);
p1n=p1n(1:mm,1:nn);
Z6t=Z6t(1:mm,1:nn);
p1t=p1t(1:mm,1:nn);
Z1t=Z1t(1:mm,1:nn);
h1t=h1t(1:mm,1:nn);
%% PSNR
p1 = psnr(Z1n,h1n);
p2 = psnr(Z6n,p1n);
p3 = psnr(Z1t,h1t);
p4 = psnr(Z6t,p1t);
peaksnr=(p1+p2+p3+p4)/4;
%% SSIM

```

```
[mssim1, ssim_map] = ssim(Z1n ,h1n); mssim1;
[mssim2, ssim_map] = ssim(Z6n,p1n); mssim2;
[mssim3, ssim_map] = ssim(Z1t,h1t); mssim3;
[mssim4, ssim_map] = ssim(Z6t,p1t); mssim4;
%% göster
disp('PSNR1 (HASTA) (RGB) ')
disp(p1)
disp('PSNR2 (SAGLIKLI) (RGB) ')
disp(p2)
disp('PSNR3 (HASTA) (Therm) ')
disp(p3)
disp('PSNR4 (SAGLIKLI) (Therm) ')
disp(p4)
disp('SSIM (HASTA) (RGB) ')
disp(mssim1)
disp('SSIM (SAGLIKLI) (RGB) ')
disp(mssim2)
disp('SSIM (HASTA) (Therm) ')
disp(mssim3)
disp('SSIM (SAGLIKLI) (Therm) ')
disp(mssim4)
```


EK-3 Hasta Bebeklere ait Termal Görüntülerin Yedi Farklı Veri Artırımı Yöntemi ile Artırılıp Kayıt Edilmesi MATLAB Kodları

```

zindan = cell(1,1900); %orijinal verilerin okunup yazilacağı hücre
%artirilmis goruntulerin yazilacagi hücreler
aydinlikzindan = cell(1,1900);
kontrastzindan = cell(1,1900);
gurultu1 = cell(1,1900);
gurultu2 = cell(1,1900);
gurultu3 = cell(1,1900);
renkdegis = cell(1,1900);
cozunurluk = cell(1,1900);
%hasta bebeklere ait yuzer goruntu okunuyor
%bebek isimleri xxx olarak gizlendi
for i = 1:100
zindan{i} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+100} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+200} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+300} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+400} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+500} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+600} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+700} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+800} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+900} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+1000} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+1100} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+1200} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+1300} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+1400} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+1500} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+1600} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+1700} = im2double(imread(sprintf('xxx%03d.jpg',i)));
zindan{i+1800} = im2double(imread(sprintf('xxx%03d.jpg',i)));
end
%gurultuler ekleniyor
for i = 1:1900
aydinlikzindan{i} = zindan{i} + 0.3;
kontrastzindan{i} = zindan{i} * 0.5;
gurultu1{i} = imnoise(zindan{i}, 'salt & pepper', 0.08);
gurultu2{i} = imnoise(zindan{i}, 'salt & pepper', 0.02);
gurultu3{i} = imnoise(zindan{i}, 'salt & pepper', 0.01);
renkdegis{i} =
cat(3, (zindan{i}(:, :, 1)), (zindan{i}(:, :, 2)), (zindan{i}(:, :, 2)));
cozunurluk{i} = imresize(imresize(zindan{i}, [128, 128]), [512, 512]);
end
mkdir('augment_hasta')
cd augment_hasta

%artitilmis goruntuler kayit ediliyor
for j = 1:1900
imwrite(zindan{j}, sprintf('hasta%d.jpg',j));
imwrite(aydinlikzindan{j}, sprintf('aydinlikhasta%d.jpg',j));
imwrite(kontrastzindan{j}, sprintf('kontrasthasta%d.jpg',j));
imwrite(gurultu1{j}, sprintf('noisyekilhasta%d.jpg',j));
imwrite(gurultu2{j}, sprintf('noisyeki2hasta%d.jpg',j));
imwrite(gurultu3{j}, sprintf('noisyeki3hasta%d.jpg',j));
imwrite(renkdegis{j}, sprintf('renkdegishasta%d.jpg',j));
imwrite(cozunurluk{j}, sprintf('cozunurlukhasta%d.jpg',j));
end

```

EK-4 Hasta Bebeklere Ait Artırılmış Görüntülerin 10 – Katlamalı Çapraz Doğrulama Tekniğine Göre Dosyalanması MATLAB Kodları

%islem bos bir klasorde basliyor ve egitim, train, hasta klasorleri
%olusturup iclerine girerek goruntuleri kayıt ediyor

```

mkdir('egitim_1')
cd egitim_1
mkdir('train_1')
cd train_1
mkdir('hasta')
cd hasta

for i = 3041:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..

mkdir('test_1')
cd test_1
mkdir('hasta')
cd hasta

for i = 1:3040
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..
cd ..
%%
mkdir('egitim_2')
cd egitim_2
mkdir('train_2')
cd train_2
mkdir('hasta')
cd hasta

for i = 1:3040
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
for i = 6081:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..

mkdir('test_2')
cd test_2
mkdir('hasta')
cd hasta

for i = 3041:6080
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..

```

```
cd ..
cd ..
%%
mkdir('egitim_3')
cd egitim_3
mkdir('train_3')
cd train_3
mkdir('hasta')
cd hasta

for i = 1:6080
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
for i = 9121:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..

mkdir('test_3')
cd test_3
mkdir('hasta')
cd hasta

for i = 6081:9120
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..
cd ..
%%
mkdir('egitim_4')
cd egitim_4
mkdir('train_4')
cd train_4
mkdir('hasta')
cd hasta

for i = 1:9120
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
for i = 12161:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..

mkdir('test_4')
cd test_4
mkdir('hasta')
cd hasta

for i = 9121:12160
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
```

```
cd ..
cd ..
cd ..
%%
tic
mkdir('egitim_5')
cd egitim_5
mkdir('train_5')
cd train_5
mkdir('hasta')
cd hasta

for i = 1:12160
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
for i = 15201:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..

mkdir('test_5')
cd test_5
mkdir('hasta')
cd hasta

for i = 12161:15200
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..
cd ..
toc
%%
mkdir('egitim_6')
cd egitim_6
mkdir('train_6')
cd train_6
mkdir('hasta')
cd hasta

for i = 1:15200
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
for i = 18241:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..

mkdir('test_6')
cd test_6
mkdir('hasta')
cd hasta

for i = 15201:18240
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
```

```
end

cd ..
cd ..
cd ..
%%
mkdir('egitim_7')
cd egitim_7
mkdir('train_7')
cd train_7
mkdir('hasta')
cd hasta

for i = 1:18240
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
for i = 21281:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..

mkdir('test_7')
cd test_7
mkdir('hasta')
cd hasta

for i = 18241:21280
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..
cd ..
%%
mkdir('egitim_8')
cd egitim_8
mkdir('train_8')
cd train_8
mkdir('hasta')
cd hasta

for i = 1:21280
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
for i = 24321:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..

mkdir('test_8')
cd test_8
mkdir('hasta')
cd hasta

for i = 21281:24320
```

```
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..
cd ..
%%
mkdir('egitim_9')
cd egitim_9
mkdir('train_9')
cd train_9
mkdir('hasta')
cd hasta

for i = 1:24320
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
for i = 27361:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end

cd ..
cd ..

mkdir('test_9')
cd test_9
mkdir('hasta')
cd hasta

for i = 24321:27360
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
cd ..
cd ..
cd ..
%%
mkdir('egitim_10')
cd egitim_10
mkdir('train_10')
cd train_10
mkdir('hasta')
cd hasta
for i = 1:27360
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
cd ..
cd ..
mkdir('test_2')
cd test_2
mkdir('hasta')
cd hasta
for i = 27361:30400
    imwrite(zindan{i}, sprintf('hasta%d.jpg',i));
end
cd ..
cd ..
cd ..
```

EK-5 VGG19 Modelini Pre – Trained Şekilde Kullanarak Termal Görüntülerin İlk Katlamasının Tam Bağlı Katman İle Sınıflandırılması MATLAB Kodları

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%EK-5'teki kodlar aşağıdaki bağlantıdan alınarak çalışmaya göre
%%düzenlenmiştir.
%%https://www.mathworks.com/help/deeplearning/examples/transfer-
%%learning-using-alexnet.html;jsessionid=d362b2f7626e76979677140e896a
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all
tic;
cd D:\transfer_learning\svm_matlab
%analyzeNetwork(net)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% 1
x=6;
y=5;
y1=10000;
g = gpuDevice;
net = vgg19;clear egitim_1 test_1
imageSize = net.Layers(1).InputSize;
egitim_1 =
imageDatastore(fullfile('D:\transfer_learning\data\egitim_1\train_1',
{'hasta', 'saglikli'}), ...
'LabelSource', 'foldernames');egitim_1.ReadFcn =
@(filename) readAndPreprocessImage(filename, imageSize);
test_1 =
imageDatastore(fullfile('D:\transfer_learning\data\egitim_1\test_1',
{'hasta', 'saglikli'}), ...
'LabelSource', 'foldernames');test_1.ReadFcn =
@(filename) readAndPreprocessImage(filename, imageSize);
inputSize = net.Layers(1).InputSize; %input size 227x227x3
layersTransfer = net.Layers(1:end-x);
numClasses = numel(categories(egitim_1.Labels)); %veriseti kac sinifli
layers = [
layersTransfer
'daha hizli ogrenmesi icin
'WeightLearnRateFactor',20,'BiasLearnRateFactor',20
fullyConnectedLayer(numClasses,'WeightLearnRateFactor',20,'BiasLearnRa
teFactor',20)
softmaxLayer
classificationLayer];
%resimleri otomatik 227x227x3 boyutuna getirecek
augimdsTrain = augmentedImageDatastore(inputSize(1:2),egitim_1);
augimdsValidation = augmentedImageDatastore(inputSize(1:2),test_1);
options = trainingOptions('sgdm', ...
'MiniBatchSize',y, ...
'MaxEpochs',10, ...
'InitialLearnRate',1e-5, ...
'Shuffle','every-epoch', ...
'ValidationData',augimdsValidation, ...
'ValidationFrequency',y1, ...
'Verbose',false);
%'Plots','training-progress');
%egitim
netTransfer = trainNetwork(augimdsTrain,layers,options);
[YPred,scores] = classify(netTransfer,augimdsValidation);
C1 = confusionmat(test_1.Labels, YPred);
save('bir.mat', 'C1')
reset(g);

```

EK-6 VGG19 Modelini Pre – Trained Şekilde Kullanarak Termal Görüntülerin İlk Katlamasının DVM İle Sınıflandırılması MATLAB Kodları

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%EK-6'daki kodlar aşağıdaki bağlantıdan alınarak çalışmaya göre
%%düzenlenmiştir.
%%https://www.mathworks.com/matlabcentral/fileexchange/57280-cnn-
%%transfer-learning-example
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear, close all
cd D:\transfer_learning\svm_matlab_download
tic
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 1
%analyzeNetwork(net)
convnet = resnet50;
imageSize = convnet.Layers(1).InputSize;
egitim_1 =
imageDatastore(fullfile('D:\transfer_learning\data\egitim_1\train_1',
{'hasta', 'saglikli'}), ...
'LabelSource', 'foldernames'); egitim_1.ReadFcn =
@(filename) readAndPreprocessImage(filename, imageSize);
test_1 =
imageDatastore(fullfile('D:\transfer_learning\data\egitim_1\test_1',
{'hasta', 'saglikli'}), ...
'LabelSource', 'foldernames'); test_1.ReadFcn =
@(filename) readAndPreprocessImage(filename, imageSize);
featureLayer = 'fc7';
trainingFeatures = activations(convnet, egitim_1, featureLayer,
'MiniBatchSize', 1, 'OutputAs', 'rows');
classifier = fitcsvm(trainingFeatures, egitim_1.Labels);
testFeatures = activations(convnet, test_1, featureLayer,
'MiniBatchSize', 1, 'OutputAs', 'rows');
predictedLabels = predict(classifier, testFeatures);
C1 = confusionmat(test_1.Labels, predictedLabels);

```



```

class_mode = 'binary')

classifier.fit_generator(training_set,
                        samples_per_epoch = 2000,
                        nb_epoch = 3
                        )

test_set.reset()
pred=classifier.predict_generator(test_set,verbose=1)

pred[pred > .5] = 1
pred[pred <= .5] = 0

test_labels = []

for i in range(0,int(len(test_set))):
    test_labels.extend(np.array(test_set[i][1]))

dosyaisimleri = test_set.file_names

sonuc = pd.DataFrame()
sonuc['dosyaisimleri']= dosyaisimleri
sonuc['tahminler'] = pred
sonuc['test'] = test_labels

from sklearn.metrics import confusion_matrix

cm = confusion_matrix(test_labels, pred)
print("Thermal Karmaşıklik Matrisi")
print (cm)

from sklearn.metrics import confusion_matrix, accuracy_score, precision_score,
recall_score, roc_auc_score

cm = confusion_matrix(test_labels, pred)
accuracy = accuracy_score(test_labels, pred)
recall = recall_score(test_labels, pred)
precision = precision_score(test_labels, pred)
roc = roc_auc_score(test_labels, pred)

print("Thermal Karmaşıklik Matrisi")
print (cm)
print("Accuracy")
print(accuracy)
print("Precision")
print(precision)
print("Recall")
print(recall)
print("Roc")
print(roc)

```

EK-8 ÇÜA Kullanılarak Yeni Termal Görüntülerin Üretilmesi PYTHON Kodları

```

# -*- coding: utf-8 -*-
"""
EK-8'deki kodlar aşağıdaki bağlantıdan alınarak çalışmaya göre düzenlenmiştir.
https://github.com/llSourceCell/Pokemon_GAN
"""

import os
import tensorflow as tf
import numpy as np
import cv2
import random
import scipy.misc
from utils import *

slim = tf.contrib.slim

HEIGHT, WIDTH, CHANNEL = 128, 128, 3
BATCH_SIZE = 64
EPOCH = 100000
version = 'z_neonatal_1'
newPoke_path = './'+ version

def lrelu(x, n, leak=0.2):
    return tf.maximum(x, leak * x, name=n)

def process_data():
    current_dir = os.getcwd()
    # parent = os.path.dirname(current_dir)
    pokemon_dir = os.path.join(current_dir, 'veriler')
    images = []
    for each in os.listdir(pokemon_dir):
        images.append(os.path.join(pokemon_dir, each))
    # print images
    all_images = tf.convert_to_tensor(images, dtype = tf.string)

    images_queue = tf.train.slice_input_producer(
        [all_images])

    content = tf.read_file(images_queue[0])
    image = tf.image.decode_jpeg(content, channels = CHANNEL)
    # sess1 = tf.Session()
    # print sess1.run(image)
    image = tf.image.random_flip_left_right(image)
    image = tf.image.random_brightness(image, max_delta = 0.1)
    image = tf.image.random_contrast(image, lower = 0.9, upper = 1.1)
    # noise = tf.Variable(tf.truncated_normal(shape
    [HEIGHT, WIDTH, CHANNEL], dtype = tf.float32, stddev = 1e-3, name = 'noise'))
    # print image.get_shape()

```

```

size = [HEIGHT, WIDTH]
image = tf.image.resize_images(image, size)
image.set_shape([HEIGHT,WIDTH,CHANNEL])
# image = image + noise
# image = tf.transpose(image, perm=[2, 0, 1])
# print image.get_shape()

image = tf.cast(image, tf.float32)
image = image // 255.0

iamges_batch = tf.train.shuffle_batch(
    [image], batch_size = BATCH_SIZE,
    num_threads = 4, capacity = 200 + 3* BATCH_SIZE,
    min_after_dequeue = 200)
num_images = len(images)

return iamges_batch, num_images

def generator(input, random_dim, is_train, reuse=False):
    c4, c8, c16, c32, c64 = 256, 128, 64, 32, 16 # channel num
    s4 = 4
    output_dim = CHANNEL # RGB image
    with tf.variable_scope('gen') as scope:
        if reuse:
            scope.reuse_variables()
        w1 = tf.get_variable('w1', shape=[random_dim, s4 * s4 * c4],
dtype=tf.float32,
            initializer=tf.truncated_normal_initializer(stddev=0.02))
        b1 = tf.get_variable('b1', shape=[c4 * s4 * s4], dtype=tf.float32,
            initializer=tf.constant_initializer(0.0))
        flat_conv1 = tf.add(tf.matmul(input, w1), b1, name='flat_conv1')
        #Convolution, bias, activation, repeat!
        conv1 = tf.reshape(flat_conv1, shape=[-1, s4, s4, c4], name='conv1')
        bn1 = tf.contrib.layers.batch_norm(conv1, is_training=is_train, epsilon=1e-
5, decay = 0.9, updates_collections=None, scope='bn1')
        act1 = tf.nn.relu(bn1, name='act1')
        # 8*8*256
        #Convolution, bias, activation, repeat!
        conv2 = tf.layers.conv2d_transpose(act1, c8, kernel_size=[5, 5], strides=[2,
2], padding="SAME",

kernel_initializer=tf.truncated_normal_initializer(stddev=0.02),
            name='conv2')
        bn2 = tf.contrib.layers.batch_norm(conv2, is_training=is_train, epsilon=1e-
5, decay = 0.9, updates_collections=None, scope='bn2')
        act2 = tf.nn.relu(bn2, name='act2')
        # 16*16*128
        conv3 = tf.layers.conv2d_transpose(act2, c16, kernel_size=[5, 5],
strides=[2, 2], padding="SAME",

```

```

kernel_initializer=tf.truncated_normal_initializer(stddev=0.02),
                name='conv3')
    bn3 = tf.contrib.layers.batch_norm(conv3, is_training=is_train, epsilon=1e-
5, decay = 0.9, updates_collections=None, scope='bn3')
    act3 = tf.nn.relu(bn3, name='act3')
    # 32*32*64
    conv4 = tf.layers.conv2d_transpose(act3, c32, kernel_size=[5, 5],
strides=[2, 2], padding="SAME",

kernel_initializer=tf.truncated_normal_initializer(stddev=0.02),
                name='conv4')
    bn4 = tf.contrib.layers.batch_norm(conv4, is_training=is_train, epsilon=1e-
5, decay = 0.9, updates_collections=None, scope='bn4')
    act4 = tf.nn.relu(bn4, name='act4')
    # 64*64*32
    conv5 = tf.layers.conv2d_transpose(act4, c64, kernel_size=[5, 5],
strides=[2, 2], padding="SAME",

kernel_initializer=tf.truncated_normal_initializer(stddev=0.02),
                name='conv5')
    bn5 = tf.contrib.layers.batch_norm(conv5, is_training=is_train, epsilon=1e-
5, decay = 0.9, updates_collections=None, scope='bn5')
    act5 = tf.nn.relu(bn5, name='act5')

    #128*128*3
    conv6 = tf.layers.conv2d_transpose(act5, output_dim, kernel_size=[5, 5],
strides=[2, 2], padding="SAME",

kernel_initializer=tf.truncated_normal_initializer(stddev=0.02),
                name='conv6')
    # bn6 = tf.contrib.layers.batch_norm(conv6, is_training=is_train,
epsilon=1e-5, decay = 0.9, updates_collections=None, scope='bn6')
    act6 = tf.nn.tanh(conv6, name='act6')
    return act6

def discriminator(input, is_train, reuse=False):
    c2, c4, c8, c16 = 16, 64, 128, 256 # channel num: 64, 128, 256, 512
    with tf.variable_scope('dis') as scope:
        if reuse:
            scope.reuse_variables()

        #Convolution, activation, bias, repeat!
        conv1 = tf.layers.conv2d(input, c2, kernel_size=[5, 5], strides=[2, 2],
padding="SAME",

kernel_initializer=tf.truncated_normal_initializer(stddev=0.02),
                name='conv1')

```

```

        bn1 = tf.contrib.layers.batch_norm(conv1, is_training = is_train,
epsilon=1e-5, decay = 0.9, updates_collections=None, scope = 'bn1')
        act1 = lrelu(conv1, n='act1')
        #Convolution, activation, bias, repeat!
        conv2 = tf.layers.conv2d(act1, c4, kernel_size=[5, 5], strides=[2, 2],
padding="SAME",

kernel_initializer=tf.truncated_normal_initializer(stddev=0.02),
                name='conv2')
        bn2 = tf.contrib.layers.batch_norm(conv2, is_training=is_train, epsilon=1e-
5, decay = 0.9, updates_collections=None, scope='bn2')
        act2 = lrelu(bn2, n='act2')
        #Convolution, activation, bias, repeat!
        conv3 = tf.layers.conv2d(act2, c8, kernel_size=[5, 5], strides=[2, 2],
padding="SAME",

kernel_initializer=tf.truncated_normal_initializer(stddev=0.02),
                name='conv3')
        bn3 = tf.contrib.layers.batch_norm(conv3, is_training=is_train, epsilon=1e-
5, decay = 0.9, updates_collections=None, scope='bn3')
        act3 = lrelu(bn3, n='act3')
        #Convolution, activation, bias, repeat!
        conv4 = tf.layers.conv2d(act3, c16, kernel_size=[5, 5], strides=[2, 2],
padding="SAME",

kernel_initializer=tf.truncated_normal_initializer(stddev=0.02),
                name='conv4')
        bn4 = tf.contrib.layers.batch_norm(conv4, is_training=is_train, epsilon=1e-
5, decay = 0.9, updates_collections=None, scope='bn4')
        act4 = lrelu(bn4, n='act4')

        # start from act4
        dim = int(np.prod(act4.get_shape()[1:]))
        fc1 = tf.reshape(act4, shape=[-1, dim], name='fc1')

        w2 = tf.get_variable('w2', shape=[fc1.shape[-1], 1], dtype=tf.float32,
                initializer=tf.truncated_normal_initializer(stddev=0.02))
        b2 = tf.get_variable('b2', shape=[1], dtype=tf.float32,
                initializer=tf.constant_initializer(0.0))

        # wgan just get rid of the sigmoid
        logits = tf.add(tf.matmul(fc1, w2), b2, name='logits')
        # dcgan
        acted_out = tf.nn.sigmoid(logits)
        return logits #, acted_out

def train():
    random_dim = 100

```

```

with tf.variable_scope('input'):
    #real and fake image placholders
    real_image = tf.placeholder(tf.float32, shape = [None, HEIGHT, WIDTH,
CHANNEL], name='real_image')
    random_input = tf.placeholder(tf.float32, shape=[None, random_dim],
name='rand_input')
    is_train = tf.placeholder(tf.bool, name='is_train')

    # wgan
    fake_image = generator(random_input, random_dim, is_train)

    real_result = discriminator(real_image, is_train)
    fake_result = discriminator(fake_image, is_train, reuse=True)

    d_loss = tf.reduce_mean(fake_result) - tf.reduce_mean(real_result) # This
optimizes the discriminator.
    g_loss = -tf.reduce_mean(fake_result) # This optimizes the generator.

    t_vars = tf.trainable_variables()
    d_vars = [var for var in t_vars if 'dis'in var.name]
    g_vars = [var for var in t_vars if 'gen'in var.name]
    trainer_d = tf.train.RMSPropOptimizer(learning_rate=2e-4).minimize(d_loss,
var_list=d_vars)
    trainer_g = tf.train.RMSPropOptimizer(learning_rate=2e-4).minimize(g_loss,
var_list=g_vars)
    # clip discriminator weights
    d_clip = [v.assign(tf.clip_by_value(v, -0.01, 0.01)) for v in d_vars]

    batch_size = BATCH_SIZE
    image_batch, samples_num = process_data()

    batch_num = int(samples_num // batch_size)
    total_batch = 0
    sess = tf.Session()
    saver = tf.train.Saver()
    sess.run(tf.global_variables_initializer())
    sess.run(tf.local_variables_initializer())
    # continue training
    save_path = saver.save(sess, "/tmp/model.ckpt")
    ckpt = tf.train.latest_checkpoint('./model/'+ version)
    saver.restore(sess, save_path)
    coord = tf.train.Coordinator()
    threads = tf.train.start_queue_runners(sess=sess, coord=coord)

    print('total training sample num:%d'% samples_num)
    #print('image_batch:%d'% image_batch)

```

```

        print('batch size: %d, batch num per epoch: %d, epoch num: %d'%
(batch_size, batch_num, EPOCH))
    print('start training...')
    for i in range(EPOCH):
        print("Running epoch {}/{}..." .format(i+1, EPOCH))
        for j in range(batch_num):
            ##print('batch_num dongusu: %d'% j)
            d_iters = 5
            g_iters = 1

            train_noise = np.random.uniform(-1.0, 1.0, size=[batch_size,
random_dim]).astype(np.dtype(float).type)
            for k in range(d_iters):
                ## print('d_iters: %d'%(k+1))
                #print(k)
                train_image = sess.run(image_batch)
                #print("train_image gecti")
                #wgan clip weights
                sess.run(d_clip)
                #print("sess.run gecti")
                # Update the discriminator
                _, dLoss = sess.run([trainer_d, d_loss],
                                feed_dict={random_input: train_noise, real_image:
train_image, is_train: True})
                #print(", dLoss gecti")

                # Update the generator
                for y in range(g_iters):
                    # train_noise = np.random.uniform(-1.0, 1.0, size=[batch_size,
random_dim]).astype(np.float32)
                    ##print('g_iters: %d'%(y+1))
                    _, gLoss = sess.run([trainer_g, g_loss],
                                        feed_dict={random_input: train_noise, is_train: True})

                # print 'train:[%d/%d],d_loss:%f,g_loss:%f'% (i, j, dLoss, gLoss)

# save check point every 500 epoch
if i%500 == 0:
    if not os.path.exists('./model/'+ version):
        os.makedirs('./model/'+ version)
    saver.save(sess, './model/'+version + '/' + str(i))
if i%50 == 0:
    # save images
    if not os.path.exists(newPoke_path):
        os.makedirs(newPoke_path)
    sample_noise = np.random.uniform(-1.0, 1.0, size=[batch_size,
random_dim]).astype(np.dtype(float).type)
    imgtest = sess.run(fake_image, feed_dict={random_input: sample_noise,
is_train: False})
    # imgtest = imgtest * 255.0

```



```
# imgtest.astype(np.uint8)
save_images(imgtest, [8,8], newPoke_path + '/epoch'+ str(i) + '.jpg')

print('train:[%d],d_loss:%f,g_loss:%f'% (i, dLoss, gLoss))
coord.request_stop()
coord.join(threads)

if __name__ == "__main__":
train()
```



ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Ahmet Haydar ÖRNEK
Uyruğu : Türkiye Cumhuriyeti
Doğum Yeri ve Tarihi : İstanbul / Bakırköy – 24.03.1995
Telefon : +90 541 3384541
E-Posta : ahmethaydarornek@gmail.com

EĞİTİM

Derece	Adı	İlçe	İl	Bitirme Yılı
Lise	: Şehremini Anadolu Lisesi,	Fatih,	İstanbul	2013
Üniversite	: Selçuk Üniversitesi,	Selçuklu,	Konya	2017
Yüksek Lisans	: Konya Teknik Üniversitesi,	Selçuklu,	Konya	Devam Ediyor

İŞ DENEYİMLERİ

Yıl	Kurum	Görevi
2015	HİDROKON KONYA HİDROLİK MAKİNA SANAYİ ve TİCARET LTD. ŞTİ.	Stajyer
2016	ELEKTRONİK SİSTEMLER MÜDÜRLÜĞÜ – İSTANBUL BÜYÜKŞEHİR BELEDİYESİ	Stajyer
2017	MEKATROSİS	Elektrik – Elektronik Mühendisi
2018 - ~	TÜBİTAK 1001 PROJESİ BURSİYERLİĞİ (Proje No: 215E019)	Elektrik – Elektronik Mühendisi – Derin Öğrenme Geliştirici

UZMANLIK ALANLARI

Lisans döneminde: Gömülü sistem tasarımı, programlanması ve gerçekleşmesi.

Yüksek lisans döneminde: Termal görüntüleme, görüntü işleme ve yapay zeka yazılımlarının geliştirilmesi ve uygulanması.

YABANCI DİLLER

İngilizce (YÖKDİL = 81.25)

İLGİ ALANLARI

Seyahat etmek, müzik dinlemek, saz – gitar - keman çalmak, spor yapmak, yüzmek, şiir okumak.

YAYINLAR

- Ahmet Haydar Ornek, Saim Ervural, Murat Ceylan, Hanifi Soylu, Duygu Savasci, Termogramların Değerlendirilmesinde Doğru Yaklaşımların Belirlenmesi, URSI-TÜRKİYE'2018 IX. Bilimsel Kongresi, 6-8 Eylül 2018, KTO Karatay Üniversitesi, Konya
- Duygu Savasci, Ahmet Haydar Ornek, Saim Ervural, Murat Ceylan, Murat Konak, Hanifi Soylu, Classification Of Unhealthy And Healthy Neonates In Neonatal Intensive Care Units Using Medical Thermography Processing And Artificial Neural Network, Classification in Clinical Applications, Elsevier (Kabul edildi, basım aşamasında)
- Ahmet Haydar Ornek, Murat Ceylan, Termal Görüntülerin Evrişimli Sinir Ağları ile Sınıflandırılmasında Data Artırımının Etkileri, 27. Sinyal İşleme ve İletişim Uygulamaları Kurultayı, 24-26 Nisan
- Ahmet Haydar Ornek, Murat Ceylan, Comparison of Traditional Transformations for Data Augmentation in Deep Learning of Medical Thermography, International Conference on Telecommunications and Signal Processing, 1 – 3 Temmuz