

KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ODAK TABANLI KONVOLÜSYONEL SİNİR AĞLARI İLE GAUSS
GÜRÜLTÜSÜNÜN GİDERİLMESİ

DOKTORA TEZİ

Ahmet ULU

OCAK 2024

TRABZON



KARADENİZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

**ODAK TABANLI KONVOLÜSYONEL SİNİR AĞLARI İLE GAUSS
GÜRÜLTÜSÜNÜN GİDERİLMESİ**

Ahmet ULU

Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsünde
"DOKTOR (BİLGİSAYAR MÜHENDİSLİĞİ)"
Unvanı Verilmesi İçin Kabul Edilen Tezdir.

Tezin Enstitüye Verildiği Tarih : 28 / 12 / 2023

Tezin Savunma Tarihi : 24 / 01 / 2024

Tez Danışmanı : Prof. Dr. Bekir DİZDAROĞLU

Trabzon 2024

ÖNSÖZ

“Odak Tabanlı Konvolüsyonel Sinir Ağları İle Gauss Gürültüsünün Giderilmesi” isimli bu tez Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı, Doktora Programı’nda hazırlanmıştır.

Tez çalışmam boyunca hiçbir desteğini esirgemeyen ve bilim insanı olma yolundaki ilk adımlarımı atarken kıymetli bilgileriyle ve tecrübesiyle yolumu aydınlatan danışman hocam Sayın Prof. Dr. Bekir DİZDAROĞLU’na teşekkürü bir borç bilirim. Ayrıca, yapıcı eleştirileri ve önerileri ile tezime büyük katkıları bulunan saygıdeğer hocalarım Sayın Doç. Dr. Önder AYDEMİR’e ve Sayın Dr. Öğr. Üyesi Murat AYKUT’a, eğitim öğretim hayatımda katkısı olan tüm hocalarıma ve ortak olarak çalışmalar yürüttüğümüz ve tez çalışması süresinde de hiçbir yardımdan kaçınmayan arkadaşlarıma teşekkür ederim.

Tez çalışması sürecinde bana desteklerini esirgemeyen Karadeniz Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü’ndeki tüm değerli hocalarıma ve asistan arkadaşlarıma teşekkürlerimi sunarım.

Son olarak, doğduğum günden aldığım tüm kararlarda bilgi ve tecrübelerinden faydalandığım, maddi ve manevi desteklerini hiçbir zaman esirgemeyen ve bugünlere gelmemde en büyük paya sahip olan annem Ayşe ULU ve babam Emin ULU’ya, ve doktora hayatımın en zor günlerinde yanımda olan değerli ve sevgili hayat arkadaşım Sayın Esra ULU’ya teşekkür eder, minnettarlığımı sunarım.

Bu tez çalışmasının bundan sonraki çalışmalara katkı sağlamasını temenni ederim.

Ahmet ULU

Trabzon 2024

TEZ ETİK BEYANNAMESİ

Doktora Tezi olarak sunduđum “Odak Tabanlı Konvolüsyonel Sinir Ağları ile Gauss Gürültüsünün Giderilmesi” başlıklı bu çalışmayı baştan sona kadar danışmanım Sayın Prof. Dr. Bekir DİZDAROĞLU’nun sorumluluğunda tamamladığımı, verileri/örnekleri kendim topladığımı, deneyleri/analizleri ilgili laboratuvarlarda yaptığımı/yaptırdığımı, başka kaynaklardan aldığım bilgileri metinde ve kaynakçada eksiksiz olarak gösterdiğimi, çalışma sürecinde bilimsel araştırma ve etik kurallara uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul ettiğimi beyan ederim. 24/01/2024

Ahmet ULU

İÇİNDEKİLER

	<u>Sayfa No</u>
ÖNSÖZ.....	III
TEZ ETİK BEYANNAMESİ.....	IV
İÇİNDEKİLER.....	V
ÖZET	VII
SUMMARY	VIII
ŞEKİLLER DİZİNİ	IX
TABLolar DİZİNİ.....	XII
SEMBOLLER VE KISALTMALAR DİZİNİ	XIII
1. GENEL BİLGİLER	1
1.1. Giriş.....	1
1.2. Sayısal Görüntü	2
1.2.1. Gri seviye Görüntü.....	2
1.2.2. Renkli (RBG) Görüntü.....	2
1.3. Gürültü.....	4
1.3.1. Gürültü Çeşitleri.....	4
1.4. Yerel İkili Örüntüler	6
1.5. Değerlendirme Metrikleri.....	9
1.5.1. Tepe İşareti Gürültü Oranı – PSNR	9
1.5.2. Yapısal Benzerlik İndeksi – SSIM.....	10
1.6. Genel Yapısı ile Derin Öğrenme	12
1.6.1. Konvolüsyonel Sinir Ağları	12
1.6.2. Konvolüsyon Katmanı	14
1.6.3. Aktivasyon Fonksiyonu	17
1.6.4. Yığın Normalizasyonu	21
1.6.5. Havuzlama Katmanı.....	23
1.6.6. Atlamalı Bağlantı	24
1.6.7. Kayıp Fonksiyonu	24
1.6.8. Adam Optimizasyon Algoritması	26
1.6.9. Genel CNN Mimarileri	27
1.6.10. Öznitelik Odak Mekanizması.....	31

2. LİTERATÜR TARAMASI	33
2.1. Derin Öğrenme ile Gürültü Azaltma	35
3. YAPILAN ÇALIŞMA	47
3.1. Motivasyon	47
3.2. Tezin Kapsamı	49
3.3. Yapılan Çalışma	51
3.3.1. Doku Çıkarım Ağı	51
3.3.2. Gauss Gürültüsü Azaltma Ağı	57
3.3.3. Deneysel Karşılaştırmalar	68
3.3.4. Ablasyon Çalışması	92
4. ÖNERİLER VE TARTIŞMA	107
5. KAYNAKLAR	110
ÖZGEÇMİŞ	

Doktora Tezi

ÖZET

ODAK TABANLI KONVOLÜSYONEL SİNİR AĞLARI İLE GAUSS GÜRÜLTÜSÜNÜN GİDERİLMESİ

Ahmet ULU

Karadeniz Teknik Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Prof. Dr. Bekir DİZDAROĞLU
2024, 115 Sayfa

Gürültü azaltma sayısal imgelerde meydana gelen bozulmaların imgenin orijinal yapısının mümkün olduğunca korunarak ortadan kaldırılması işlemi olup bir imgenin görsel kalitesini iyileştirmesinden dolayı görüntü işleme alanında yapılan birçok farklı çalışma için bir ön işleme adımı olarak büyük önem arz etmektedir. Gauss gürültüsü en çok karşılaşılan gürültülerden bir tanesidir. Tez çalışması kapsamında gri ve renkli seviyeli görüntülerden farklı seviyelerde gauss gürültüsünün giderilmesi için konvolüsyonel sinir ağları (CNN) tabanlı yeni bir ağ önerilmiştir. İlk olarak gürültü azaltma işlemi sırasında imgenin orijinal yapısının korunması için asıl gürültü azaltma işlemine kılavuz olması amacıyla gürültülü imgeden Yerel İkili Örüntü (YİÖ) tabanlı doku imgesi çıkarılmış ve bu amaçla iki dallı bir yapıya sahip TENet isimli bir CNN ağı önerilmiştir. Daha sonra gürültü azaltma işlemi için gürültülü imge ve gürültülü imgeden elde edilen YİÖ imgesini girdi olarak alan CNN tabanlı MLFAN ağı önerilmiştir. MLFAN ağı tasarımı olarak çok seviyeli öznitelik çıkarımı için farklı çekirdek boyutlarına sahip konvolüsyon katmanlarından ve bu özniteliklerin ağırlıklandırılması için kanal tabanlı odak mekanizmalarından oluşmaktadır. Buna ek olarak MLFAN ağının tasarımının değerlendirilmesi için kapsamlı bir ablasyon çalışması sunulmuş sayısal ve görsel sonuçlarla birlikte değerlendirilmesi yapılmıştır.

Anahtar Kelimeler: Gürültü azaltma, Derin öğrenme, Konvolüsyonel sinir ağları, Öznitelik odağı, Yerel ikili örüntüler, Çok seviyeli öznitelik çıkarımı

PhD. Thesis

SUMMARY

GAUSSIAN NOISE REMOVAL WITH ATTENTION-BASED CONVOLUTIONAL NEURAL NETWORKS

Ahmet ULU

Karadeniz Technical University
The Graduate School of Natural and Applied Sciences
Computer Engineering Graduate Program
Supervisor: Prof. Dr. Bekir DİZDAROĞLU
2024, 115 Pages

The process of noise reduction in digital images involves mitigating distortions that occur, aiming to eliminate them while preserving the original structure of the image as much as possible. This operation holds significant importance as a preprocessing step in various image-processing studies due to its ability to enhance the visual quality of an image. Gaussian noise is one of the most encountered types of noise. In the scope of this thesis, a new network based on Convolutional Neural Networks (CNN) is proposed for the removal of Gaussian noise at different levels in grayscale and color images. Initially, a Local Binary Pattern (LBP) based texture image is extracted from the noisy image to serve as a guide for the main noise reduction process, aiming to preserve the original structure of the image. For this purpose, a CNN named TENet, with a two-branch structure, is proposed. Subsequently, for the noise reduction process, a new CNN-based MLFAN network is proposed, taking both the noisy image and the LBP-based texture image obtained from the noisy image as inputs. The MLFAN network is designed with convolutional layers having different kernel sizes for multi-level feature extraction and channel-based attention mechanisms for weighting these features. Additionally, a comprehensive ablation study is presented to evaluate the design of the MLFAN network, accompanied by numerical and visual results.

Key Words: Noise reduction, Deep learning, Convolutional neural networks, Feature Attention, Local binary patterns, Multi-level feature extraction.

ŞEKİLLER DİZİNİ

	<u>Sayfa No</u>
Şekil 1.1 Gri düzeyli Barbara imgesi	2
Şekil 1.2. RGB renk uzayı	3
Şekil 1.3. Renkli Barbara imgesi (a), RGB kanalına ait histogram (b)	3
Şekil 1.4. Gauss gürültüsü eklenmiş renkli Barbara imgesi	6
Şekil 1.5. Basit YİÖ operatörü	7
Şekil 1.6. YİÖ operatörü	8
Şekil 1.7. YİÖ örnek imge, Barbara imgesi (a), YİÖ Barbara imgesi (b)	9
Şekil 1.8. Çeşitli bozulmalara uğramış gri seviye Barbara görüntüsü (Ulu, 2018)	11
Şekil 1.9. Konvolüsyon işlemine bir örnek	15
Şekil 1.10. Genişlemiş konvolüsyon	17
Şekil 1.11. Sigmoid fonksiyonu	18
Şekil 1.12. ReLU aktivasyon fonksiyonu	19
Şekil 1.13. Leaky ReLU aktivasyon fonksiyonu	21
Şekil 1.14. Ortalama ve maksimum havuzlama işlemi	24
Şekil 1.15. LeNet mimarisi [16]	27
Şekil 1.16. AlexNet mimarisi [17]	28
Şekil 1.17. Inception modülü	29
Şekil 1.18. Artık atlamalı bağlantı	29
Şekil 1.19. 34 katmanlı ResNet mimarisi	30
Şekil 1.20. Odak mekanizması, (a) sert odak, (b) yumuşak odak	32
Şekil 2.1. Benzer karakteristiğe sahip imge parçaları içeren Barbara görüntüsü	34
Şekil 2.2. DnCNN modeli [41]	35
Şekil 2.3. RED30 ağı [42]	36
Şekil 2.4. FFDNet ağı [43]	37
Şekil 2.5. MWCNN ağı [44]	37
Şekil 2.6. McCNN ağı için önerilen çok seviyeli öznetelik çıkarım bloğu	38
Şekil 2.7. IRCNN ağı [46]	39
Şekil 2.8. ADNet ağı [48]	40
Şekil 2.9. BRDNet ağı	40
Şekil 2.10. RIDNet ağı [50]	41
Şekil 2.11. PRIDNet ağı	45
Şekil 2.12. Çekirdek seçim modülü (ÇSM)	45

Şekil 3.1. Gürültülü ve gürültüsüz Barbara imgesi ve piksel değerleri ile gösterilen imge parçaları	48
Şekil 3.2. Gri seviye Barbara imgesi ve farklı gürültü seviyelerinde YİÖ imgeleri	51
Şekil 3.3. İkili konvolüsyonel blok (DCB).....	52
Şekil 3.4. Doku çıkarma ağı – TENet.....	54
Şekil 3.5. Barbara imgesi ve gürültü imgeler üzerine elde edilen YİÖ sonuçları	56
Şekil 3.6. Çok seviyeli öznetelik çıkarım bloğu (MLF).....	58
Şekil 3.7. Barbara imgesi için farklı çekirdek boyutlarına sahip konvolüsyon katmanlarında elde edilmiş öznetelik haritaları.....	59
Şekil 3.8. Odak mekanizması	61
Şekil 3.9. Odak mekanizması öncesi ve sonrası öznetelik haritaları	62
Şekil 3.10. MLFAN ağı.....	64
Şekil 3.11. MLFAN ve TENet ağı.....	67
Şekil 3.12. Set12 veri seti Monar imgesinin $\sigma=25$ için gürültü azaltma sonuçları.....	72
Şekil 3.13. Set68 veri seti test011 imgesinin $\sigma=15$ için gürültü azaltma sonuçları.....	74
Şekil 3.14. Set12 veri seti Parrot imgesinin $\sigma=50$ için gürültü azaltma sonuçları	75
Şekil 3.15. Set12 veri seti Monar imgesinin $\sigma=15$ için gürültü azaltma sonuçları.....	76
Şekil 3.16. Set12 veri seti Monar imgesinin $\sigma=50$ için gürültü azaltma sonuçları.....	77
Şekil 3.17. Set12 veri seti Parrot imgesinin $\sigma=15$ için gürültü azaltma sonuçları	78
Şekil 3.18. Set12 veri seti Parrot imgesinin $\sigma=25$ için gürültü azaltma sonuçları	79
Şekil 3.19. Set68 veri seti test011 imgesinin $\sigma=25$ için gürültü azaltma sonuçları.....	80
Şekil 3.20. Set68 veri seti test011 imgesinin $\sigma=50$ için gürültü azaltma sonuçları.....	81
Şekil 3.21. Kodak24 veri seti kodim06 imgesinin $\sigma=50$ için gürültü azaltma sonuçları.....	82
Şekil 3.22. McMaster veri seti kadim06 imgesinin $\sigma=15$ için gürültü azaltma sonuçları.....	83
Şekil 3.23. CBSD68 veri seti test011 imgesinin $\sigma=25$ için gürültü azaltma sonuçları	84
Şekil 3.24. Kodak24 veri seti kodim06 imgesinin $\sigma=15$ için gürültü azaltma sonuçları.....	85
Şekil 3.25. Kodak24 veri seti kodim06 imgesinin $\sigma=25$ için gürültü azaltma sonuçları.....	86
Şekil 3.26. McMaster veri seti kadim06 imgesinin $\sigma=25$ için gürültü azaltma sonuçları.....	87
Şekil 3.27. McMaster veri seti kadim06 imgesinin $\sigma=50$ için gürültü azaltma sonuçları.....	88

Şekil 3.28. CBSD68 veri seti test011 imgesinin $\sigma=15$ için gürültü azaltma sonuçları	89
Şekil 3.29. CBSD68 veri seti test011 imgesinin $\sigma=50$ için gürültü azaltma sonuçları	90
Şekil 3.30. CBAM odak mekanizması, (a) kanal odak mekanizması, (b) uzamsal odak mekanizması	95
Şekil 3.31. Yeni MLF bloğu	96
Şekil 3.32. SMLFAN Ağı	99



TABLULAR DİZİNİ

	<u>Sayfa No</u>
Tablo 3.1. Set68 veri seti için farklı metotlar ile elde edilen ortalama PSNR sonuçları.....	68
Tablo 3.2. Set12 veri seti için farklı metotlar ile elde edilen PSNR değerleri	69
Tablo 3.3. Set12 ve Set68 veri setinde farklı metotlar ile elde edilen ortalama SSIM değerleri	71
Tablo 3.4. CBSD68, Kodak24 ve McMaster veri setinde farklı metotlar ile elde edilen ortalama PSNR ve SSIM değerleri	91
Tablo 3.5. PSNR ve SSIM metrikleri altında renkli imgeler için ablasyon sonuçları	105
Tablo 3.6. PSNR ve SSIM metrikleri altında gri seviye imgeler için ablasyon sonuçları.....	106

SEMBOLLER VE KISALTMALAR DİZİNİ

BN	: Yığın Normalizasyon (Batch Normalization)
CBAM	: Konvolüsyonel Blok Odak Modülü (Convolutional Block Attention Module)
CBMLFAN	: Konvolüsyonel Blok MLFAN (Convolutional Block MLFAN)
CNN	: Konvolüsyonel Sinir Ağları (Convolutional Neural Network)
DCB	: İkili Konvolüsyonel Blok (Dual Convolutional Block)
FAN	: Öznitelik Odak Ağı (Feature Attention Network)
FC	: Tam Bağlantılı Katmanı (Fully Connected Layer)
GAP	: Global Ortalama Havuzlama (Global Average Pooling)
GPU	: Grafik İşlemci Birimi (Graphics Processing Unit)
MLF	: Çok Seviyeli Öznitelik Çıkarımı (Multilevel Feature Extraction)
MLFAN	: Çok Seviyeli Öznitelik Odak Ağı (Multilevel Feature Attention Network)
MLMLFAN	: Ortalama İkili MLFAN (Mean Local MLFAN)
MSE	: Ortalama Kare Hata (Mean Square Error)
NLM	: Yerel Olmayan Ortalamalar (Non-Local Means)
NMLFAN	: Gürültülü MLFAN (Noisy MLFAN)
ReLU	: Doğrultulmuş Doğrusal Birim (Rectified Linear Unit)
RGB	: Kırmızı Yeşil Mavi (Red Green Blue)
SENet	: Sıkıştırma ve Uyarma Ağları (Squeeze and Excitation Networks)
SGD	: Olasılıksal Gradyan İnişi (Stochastic Gradient Descent)
SMLFAN	: Atlamalı Bağlantı MLFAN (Skip Connection MLFAN)
TENet	: Doku Çıkarım Ağı (Texture Extraction Network)
TİGO	: Tepe İşareti Gürültü Oranı (Peak to Signal Noise Ratio – PSNR)
YBI	: Yapısal Benzerlik İndisi (Structural Similarity – SSIM)
YİÖ	: Yerel İkili Örüntüler
YOİÖ	: Yerel Ortalama İkili Örüntü

1. GENEL BİLGİLER

1.1. Giriş

Günümüzde teknolojilerin gelişimi ile sayısal görüntülerin kazanımı yaygınlaşmış ve kullanımı artmıştır. Bu da görüntü işleme alanını günümüzün en popüler konularından biri haline getirmiştir. Görüntü işleme genel olarak bir görüntüden faydalı bilgilerin çıkarılması ve bu faydalı bilgiler üzerinden örüntü tanıma ya da anlam çıkarma işlemi yapılması olarak düşünülebilir. Buna nesne tanıma, hastalık teşhisi, kimlik tespiti gibi örnekler verilebilir. Ancak bir görüntünün kazanımı, iletimi sırasında görüntülerden sıklıkla gürültü oluşabilmektedir, bu da görüntüden çıkarılacak olan faydalı bilgiler kalitesini düşürmekte ve görüntü işleme yönteminin başarısını dolaylı olarak düşürmektedir. Bu nedenle gürültü azaltma işlemi, görüntü işleme alanında bir ön işleme adımı olarak düşünülmekte ve literatürde sıklıkla çalışılmaktadır.

Gürültü azaltma genel olarak, gürültü bir görüntü üzerinde gürültünün kaldırılarak özgün görüntünün elde edilmesi işlemi olarak düşünülebilir. Ancak burada dikkat edilmesi gereken gürültüyü en iyi şekilde gidermek ama bu sırada gürültüye ait detay (yapı ve doku) bilgilerin kaybolmasının önüne geçerek, imgenin özgün yapısının korunmasını sağlamaktır.

Bu çalışmada gri seviye ve renkli görüntülerde en çok karşılaşılan Gauss gürültüsünün giderimi için son yılların en popüler konularından olan konvolüsyonel sinir ağı(CNN) kullanılarak bir ağ tasarımı yapılmış ve gürültü azaltma işlemi gerçekleştirilmiştir.

Yapılan çalışmada ilk olarak görüntünün özgün yapısını korumaya destek olması ve gürültü azaltma işleminde detay bilgilerin korunması için bir rehber olması için gürültü görüntüden doku bilgisi çıkarılmıştır. Bu amaçla gürültü görüntüden doku bilgisi çıkarmak için hiyerarşik yapıda bir ağ tasarımı yapılmış (TENet) ve doku bilgisi içeren Yerel İkili Örüntü (YİÖ) imgesi elde edilmiştir. Çıkarılan bu görüntü, gürültülü görüntüyle birlikte asıl gürültü azaltma işlemini yapacak olan ağa girdi olarak verilmiştir. Gürültü azaltma ağı (MLFAN) tasarımında çok seviyeli özneteliklerin çıkarımı için bir blok tasarımı yapılmış ve bu özneteliklerin ağırlıklandırılması için kanal tabanlı odak mekanizması kullanılmıştır.

1.2. Sayısal Görüntü

1.2.1. Gri seviye Görüntü

İki boyutlu bir görüntünün sayısal ortamda değerlendirilebilmesi amacıyla ayrık olarak ifade edilmesine sayısal görüntü(imge) denilmektedir. Bu sayısal görüntüler, sayısal değerleri [0-255] aralığından değişen ve satır sütun bilgileri ile ifade edilen “piksel” denilen elemanlardan oluşmaktadır (Gonzales, 2009). Sayısal bir görüntüye örnek Şekil 1.1'de verilmiştir.

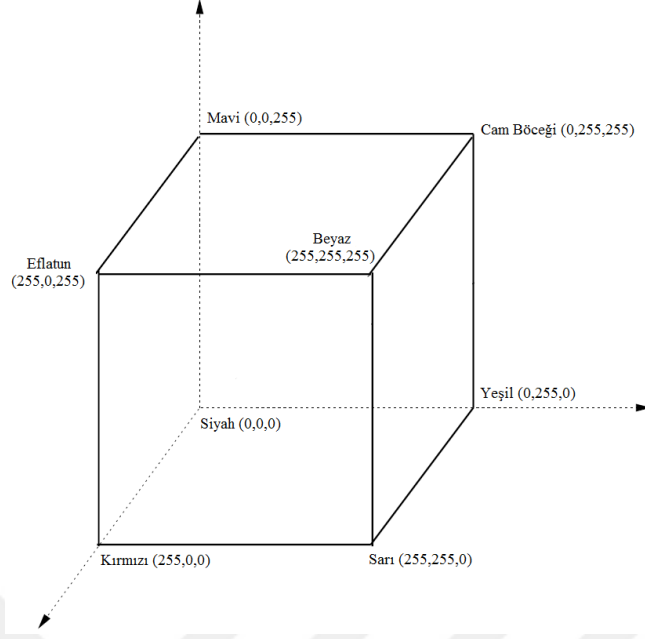


Şekil 1.1 Gri düzeyli Barbara imgesi

1.2.2. Renkli (RGB) Görüntü

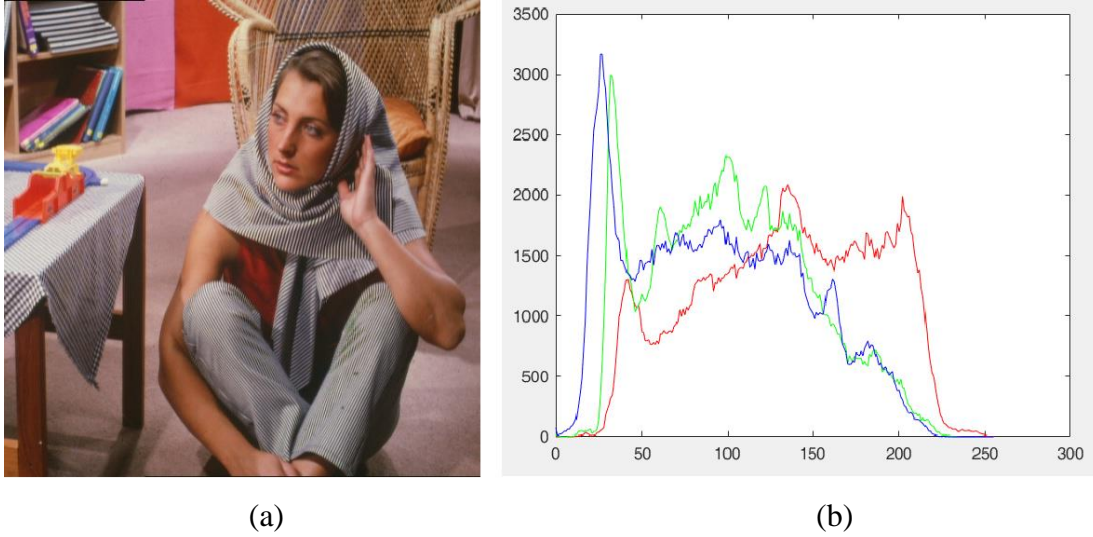
Renk uzayları siyah beyaz görüntünün dışında renkleri de göstermek için 3 veya 4 boyutlu koordinat sistemiyle ifade edilen matematiksel modellerdir. Renk uzayları arasında dönüşüm işlemleri doğrusal ve doğrusal olmayan metotlar ile yapılabilmektedir. Çok sayıda renk uzayı bulunmaktadır ve bunlardan renkli görüntünün ifade edilmesi için de en çok kullanılanı RGB renk uzayıdır.

RGB uzayı görüntü işlemede de en çok kullanılan renk uzayıdır. İlgili çalışmada da RGB renk uzayı tercih edilmiştir. RGB renk uzayı sırasıyla Kırmızı (Red), Yeşil (Green) ve Mavi (Blue) dalga boyları ve bu dalga boylarının belirli oranda birleşimini ifade etmektedir (Plataniotis ve Venetsanopoulos, 2000). RGB uzayının kartezyen koordinat sisteminde tanımlanması Şekil 1.2'deki gibidir.



Şekil 1.2. RGB renk uzayı

RGB uzayında tanımlanan bir imge renkli imgeyi ifade etmektedir (URL-1 2023). Renkli bir imgeye örnek olarak Şekil 1.3'te renkli Barbara imgesi ve tek bir satır üzerinde elde edilen R,G,B kanallarına ait renk bilgilerinin grafiksel dağılımı verilmiştir (Ulu, 2018).



Şekil 1.3. Renkli Barbara imgesi (a), RGB kanalına ait histogram (b)

Şekil 1.3 (b)'de yer alan Barbara imgesine ait olan histogramda, RGB uzayında 3 kanala ait olan histogram sırasıyla kırmızı, yeşil ve mavi renkli çizgilerle gösterilmiştir.

1.3. Gürültü

Sayısal görüntülerde (imgelerde) görülen, pikselin yapısını kısmen ya da tamamıyla bozan rastgele yapıya sahip olan işaretlere gürültü adı verilir. Gürültü, sayısal imgenin iletimi, kopyalanması, kaydedilmesi, sıkıştırılması ve elde edinimi sırasında ortaya çıkabileceği gibi çevresel faktörlere bağlı olarak da oluşabilmektedir (Bovik, 2010).

1.3.1. Gürültü Çeşitleri

Gürültü, sayısal görüntülerde pikselin sayısal değerini azaltan ya da arttıran ve mevcut piksel değerini ölçekleme yoluyla çarpımsal olarak etkileyebildiği gibi toplamsal olarak da etkileyebilen bir bozulma türüdür.

Toplamsal gürültü, imge bir kanalda iletilirken imgeden bağımsız işaretlerden etkilenmesi sonucu ortaya çıkar. Bu tür gürültü, pikselin mevcut sayısal değerinden bağımsız olarak pozitif ya da negatif bir değişim olarak etkileyebilir. Toplamsal gürültünün matematiksel gösterimi Eşitlik (1.1)'de bulunmaktadır:

$$f(x, y) = I(x, y) + n(x, y) \quad (1.1)$$

burada $I(x, y)$ gürültüsüz görüntüyü, $n(x, y)$ gürültüyü ve $f(x, y)$ ise gürültülü görüntüyü ifade etmektedir. Eşitlik (1.1)'den de anlaşılacağı üzere gürültü ve görüntü birbirinden bağımsız bir yapıdadır.

Bazı durumlarda, gürültünün şiddeti, ilgili görüntünün piksel değerinin büyüklüğüne bağlı olabilir. Gürültünün, görüntünün piksel değerinden belirgin biçimde daha yüksek olması durumuna çarpımsal gürültü adı verilir. Yeni bir görüntünün piksel değeri, orijinal görüntünün piksel değeri ile gürültünün çarpımı olarak hesaplanır (Yıldırım 2007). Çarpımsal gürültünün matematiksel ifadesi Eşitlik (1.2)'de sunulmuştur.

$$f = I(x, y) + n(x, y) \times I(x, y) = I(x, y) \times (1 + n(x, y)) \quad (1.2)$$

Televizyon iletim hatlarında meydana gelen hataların sebep olduğu, televizyon görüntülerindeki tesadüfi deformasyonları çarpımsal gürültü örneği olarak gösterebiliriz.

Sayısal görüntülerde gözlemlenen ve piksel değerlerini etkileyen gürültü türlerine Gauss gürültüsü, darbe gürültüsü, leke gürültüsü ve nicemeleme gürültüsü gibi örnekler

verilebilir. Bu gürültüler arasında, en yaygın olarak rastlanana Gauss gürültüsüdür. Bu çalışmada, özellikle Gauss gürültüsüne odaklanılmıştır.

A. Gauss Gürültüsü:

Beyaz gürültü, uzamsal olarak sabit şiddete sahip ve frekansa bağlı olmayan bir gürültü türünü temsil eder. Gauss gürültüsü ise beyaz gürültünün bir alt tipidir (Baştürk, 2006). Gauss gürültüsü, genellikle elektronik görüntüleme sistemlerinde bulunan iletkenlerdeki termal ısının etkisiyle oluşur ve görüntü üzerindeki tüm piksel değerlerini belirli bir oranda değiştirir, bu da toplamsal bir gürültü türüdür. Gürültünün pikseller üzerindeki etkileri, olasılık yoğunluk fonksiyonunun Gauss eğrisine benzerliği nedeniyle bu ismi alır. Gürültünün miktarı görüntüde belirlenirken, olasılık yoğunluk fonksiyonunun ortalaması ve standart sapması değerlerine bakılır (McAndrew, 2004). Gauss gürültüsüne ait olasılık yoğunluk fonksiyonu Eşitlik (1.3)'de verilmiştir.

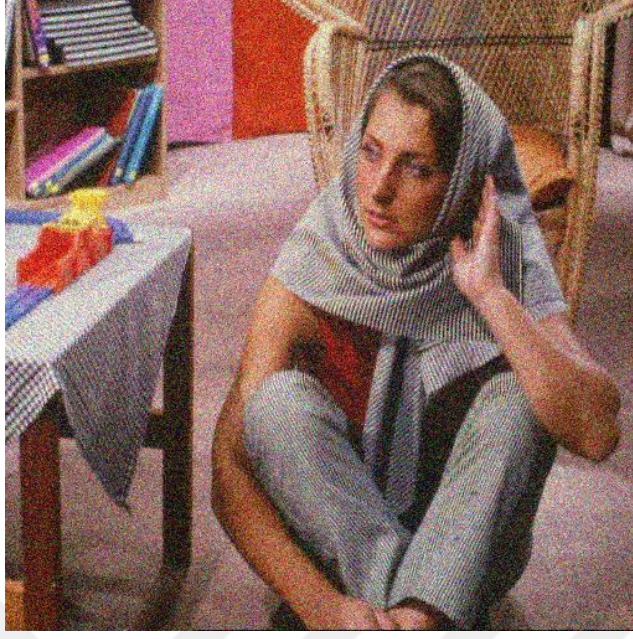
$$P(n) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(n-\mu)^2}{2\sigma^2}} \quad (1.3)$$

Eşitlik (1.3)'deki μ terimi Gauss gürültüsünün ortalamasını, σ ise standart sapmasını ifade etmektedir. Standart sapmanın artmasıyla, piksel değerleri pozitif ve negatif yönde daha fazla değişebilirken, gürültünün ortalaması arttıkça tüm piksel değerlerinin aynı oranda pozitif yönde artmasına neden olur. Gauss gürültüsünün tüm piksellere yayılan bir gürültü olduğundan, görüntünün tamamında bozulmalar meydana gelmektedir.

Eşitlik 1.3'te gri seviye bir görüntüye ait Gauss gürültüsünün dağılımı gösterilmektedir. Renkli görüntü için Gauss gürültüsünün dağılımı Eşitlik (1.4)'deki gibidir.

$$P(\mathbf{n}) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^3 4\pi \|\mathbf{n}\|_{L^2}^2 e^{-\frac{\|\mathbf{n}\|_{L^2}^2}{2\sigma^2}} \quad (1.4)$$

Yukarıdaki eşitlikte \mathbf{n} renkli görüntü için gürültüyü ifade etmektedir. Renkli imgelelerde Gauss gürültüsüne örnek olarak $\mu = 0$ ve $\sigma = 30$ alınmak üzere Barbara imgesi Şekil 1.4'te gösterilmiştir.



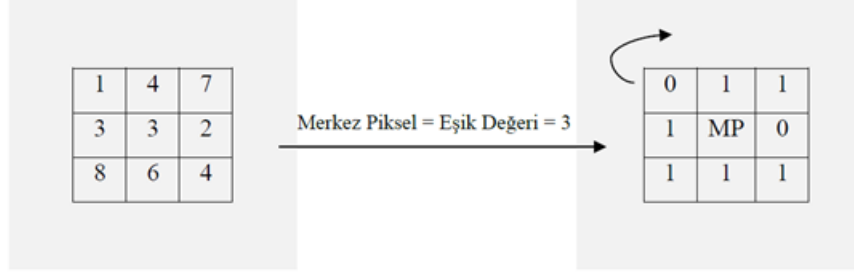
Şekil 1.4. Gauss gürültüsü eklenmiş renkli Barbara imgesi

Şekil 1.4'ten de anlaşılacağı üzere Gauss gürültüsü bütün piksellere bulaşan bir gürültü türüdür.

1.4. Yerel İkili Örüntüler

Yerel ikili örüntüler (YİÖ – Local Binary Patterns (LBP)), ilk olarak Ojala ve arkadaşları tarafından tanıtılmış ve doku yapısının çıkarımı için kullanılmıştır (Ojala vd., 1996). YİÖ basit ancak etkili bir doku tanımlama operatörü olarak düşünülebilir. Hesaplanması basit olması, ayırt ediciliği yönünden güçlü bir yapı sunması ve özellikle gri seviye görüntüler üzerindeki aydınlanma farklılıklarından çok fazla etkilenmemesi YİÖ yönteminin çokça tercih edilen bir doku çıkarma yöntemi olmasını sağlamıştır. Bu nedenle, gerçek ortamlardan elde edilen görüntüler üzerinde kullanılmasının ideal olması ve hesaplama basitliği, görüntülerin eş zamanlı olarak işlenmesini de kolaylaştırmaktadır. Sağladığı bu faydalardan dolayı YİÖ operatörü, doku segmentasyonu (Ojala ve Pietikäinen, 1996), yüz tanıma (Ahonen vd., 2004) ve görüntü yeniden yapılandırma (Waller vd., 2013) gibi birçok alanda kullanılmaktadır. Temel olarak YİÖ doku çıkarma işleminde, 3x3'lük bir pencere kullanılır ancak isteğe bağlı olarak pencere boyutu genişletilebilir. Daha sonra ortadaki pikselin sayısal değeri ile değerlendirme kapsamına alınan komşu piksellerin

sayısal değerleri karşılaştırılarak bir eşikleme işlemine tabi tutulur. Eğer karşılaştırılan piksel büyük ve eşitse 1 küçükse 0 yazılır. Bütün pikseler karşılandıktan sonra tüm pikseler sol üstten başlayarak saat yönünde alınır. Elde edilen ikili sayı ondalık değere dönüştürülerek nihai sayısal değer elde edilir. YİÖ operatöründe eşikleme işlemine bir örnek Şekil 1.5'te gösterilmiştir.



Şekil 1.5. Basit YİÖ operatörü

Şekil 1.5'te eşikleme işlemi sonucunda oluşan ikili sayı, sol üst köşedeki piksel değerinden başlamak üzere saat yönünde ilerleyerek oluşturulmuştur. İkilik tabanda elde edilen bu sayılar; Yerel İkili Örüntü kodları olarak adlandırılır. Daha sonra bu sayı Eşitlik (1.5)'de örneği verildiği gibi ondalık tabana çevrilir.

$$(01101111)_2 = 103 \quad (1.5)$$

YİÖ yönteminin matematiksel gösterimi Eşitlik (1.6) ve Eşitlik (1.7)'deki gibidir:

$$YİÖ(x) = \sum_{i=0}^n f(x_i - x) 2^{i-1} \quad (1.6)$$

$$f(y) = \begin{cases} 0, & y < 0 \\ 1, & y \geq 0 \end{cases} \quad (1.7)$$

burada, n komşu sayısını, x_i i. sıradaki komşu piksel değeri, x merkez piksel değerini ifade etmektedir. Örnek bir YİÖ işleminin yapılışı Şekil 1.6'da verilmiştir.

i_6	5	i_7	7	i_8	3
i_5	2		5	i_1	4
i_4	8	i_3	1	i_2	9

Merkez Piksel ve 3x3 Komşuluğu

i_6	1	i_7	1	i_8	0
i_5	0			i_1	0
i_4	1	i_3	0	i_2	1

Eşikleme İşlemi Sonrası

i_6	32	i_7	64	i_8	128
i_5	16			i_1	0
i_4	8	i_3	4	i_2	2

İkili Üssel Ağırlıklar

i_6	32	i_7	64	i_8	0
i_5	0			i_1	0
i_4	8	i_3	0	i_2	2

 $YİÖ = 2+8+32+64 = 106$

Şekil 1.6. YİÖ operatörü

Buna ek olarak örnek bir görüntü üzerine YİÖ operatörünün uygulanmasına ait bir sonuç Şekil 1.7’de gösterilmiştir.



Şekil 1.7. YİÖ örnek imge, Barbara imgesi (a), YİÖ Barbara imgesi (b)

Şekil 1.7- (a)'da Barbara imgesi ve (b)'de YİÖ operatörü uygulanmış doku bilgisi içeren imge yer almaktadır.

1.5. Değerlendirme Metrikleri

1.5.1. Tepe İşareti Gürültü Oranı – PSNR

Tepe İşaret Gürültü Oranı (Peak Signal-to-Noise Ratio - PSNR), imge düzenleme işlemlerinde, özellikle gürültü azaltma ve imge sıkıştırma gibi süreçlerde elde edilen sonuç görüntüsüne ait kalitenin ölçülmesi için sıkça kullanılır. Temelde, bu ifade ortalama kare hata (Mean Square Error - MSE) kavramına dayanmaktadır. İki imge arasındaki ortalama kare hata (OKH), aşağıdaki Eşitlik (1.8) ile hesaplanır:

$$OKH = \frac{1}{n \times m} \left(\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (I(x,y) - u(x,y))^2 \right) \quad (1.8)$$

buradaki I ve u sırasıyla gürültü içermeyen orijinal görüntü ve gürültü azaltma işlemi sonucu elde edilen görüntüyü ifade etmektedir. Eşitlik (1.8)'deki m ve n ise görüntüdeki yatay ve dikey piksellerin uzunluklarını göstermektedir. Buna göre Tepe İşareti Gürültü Oranı (TİGO) Eşitlik (1.9)'da gösterildiği gibi hesaplanmaktadır:

$$T\dot{I}GO = 10 \times \log_{10} \left(\frac{enb^2}{OKH} \right) = 20 \times \log_{10} \left(\frac{enb}{\sqrt{OKH}} \right) \quad (1.9)$$

buradaki “enb” ifadesi görüntüde bulunan pikselin en büyük sayısal değerini ifade etmektedir. Gri seviye ya da RGB uzayına ait bir görüntünün kullanılması durumunda enb değeri 255 sayısına karşılık gelmektedir. Burada gürültüsüz sayısal görüntü I ile, gürültü azaltma işlemi sonucu elde edilen u görüntüsünün birbirine yakınlığı ölçülmüştür. Bu iki görüntü yapısal olarak birbirine ne kadar yakın olursa TİGO değeri o kadar yüksek çıkmaktadır.

Burada değerlendirme işlemini yaparken I gürültü eklenmemiş orijinal imge, u ise yumuşatılmış imge olarak alınmıştır ve böylece yumuşatılmış imgenin orijinal imgeye yakınlığı ölçülmüştür. İki imge birbirine ne kadar yakınsa yani gürültü azaltma işlemi ne kadar başarılı ise PSNR değeri o kadar büyük çıkmaktadır (URL-2, 2023).

1.5.2. Yapısal Benzerlik İndeksi – SSIM

Yapısal Benzerlik İndisi (YBI – Structural Similarity (SSIM)) iki sayısal görüntü arasında benzerliğin ölçülmesi için kullanılan bir başka metriktir. TİGO metriğine oranla, bu indisin görsel olarak algılanan benzerliği daha iyi ölçtüğü ifade edilmektedir (Wang vd., 2004). Buna örnek olarak Şekil 1.8’de farklı görsel bozulmalara tabi tutulmuş Barbara imgesi verilmiştir.

Buna göre Şekil 1.8’de (a) orijinal Barbara görüntüsünü ifade etmektedir. Diğer görüntüler ise Barbara görüntüsünün çeşitli bozulmalara maruz kalmış halleridir. Buna göre sırasıyla (b) kontrast germe, (c) JPEG sıkıştırma, (d) bulanıklaştırma, (e) gauss gürültüsü ekleme ve (f) ise literatürde çokça kullanılan bir başka gürültü türü olan tuz-biber gürültüsü eklenmiş Barbara imgesidir. Eğer bu görüntüler TİGO metriği altında sonuçlarına bakacak olursak hepsi yaklaşık olarak 21.7 değerini vermektedir. Fakat aynı PSNR değerine sahip olan ve farklı bozulmalara uğramış bu görüntüleri görsel olarak değerlendirecek olursak, rahatlıkla en büyük bozulmaların (e) veya (d) görüntüsünde meydana geldiğini söyleyebiliriz. Buradan hareketle TİGO metriğinin tek başına iki farklı görüntü arasındaki benzerliği ölçmeye yeterli olmadığı söylenebilir. Bu nedenle literatürde benzerlik ölçümünün yapılması için çok farklı metrikler önerilmiştir ve YBI metriği de bunlardan en çok tercih edilenlerindedir. YBI metriği altında bakacak olursak (a) görüntüsü için 0.8591,

(b) görüntüsü için 0.7069, (c) görüntüsü için 0.5754, (d) görüntüsü için 0.3882 ve son olarak tuz-biber görüntüsü ile bozulmuş (e) görüntüsü için 0.5631 değerleri hesaplanmıştır.



Şekil 1.8. Çeşitli bozulmalara uğramış gri seviye Barbara görüntüsü (Ulu, 2018)

TİGO metriğine kıyasla YBI metriği altında yapılan değerlendirmenin görüntüleri ayırıştırma ya da başka bir deyişle benzerlik ölçümünde kullanımının faydası burada açıkça görülmektedir. Bu nedenle bu çalışmada TİGO metriğinin yanı sıra YBI metriği de tercih edilmiştir. Yapısal benzerlik indisi (YBI) Eşitlik (1.10)'daki gibi hesaplanmaktadır:

$$YBI = \left(\frac{(2\mu_I\mu_u + C_1)(2\sigma_{Iu} + C_2)}{(\mu_I^2 + \mu_u^2 + C_1)(\sigma_I^2 + \sigma_u^2 + C_2)} \right) \quad (1.10)$$

burada μ_I ve μ_u girdi imgesi ve düzenlenmiş imge için yerel ortalama değerler, σ_I^2 ve σ_u^2 yerel değişimler, σ_{Iu} yerel ortak değişinti, C_1 ve C_2 sabit değerlerdir. Eşitlik (1.10)'dan anlaşılacağı üzere YBI metriği parlaklık ve zıtlığı tek bir yerel metrikte birleştirerek iki görüntü için benzerliğini ölçmektedir. YBI metriği [0-1] aralığında bir değer almakla birlikte

yüksek YBI değerleri iki imge için benzerliğin daha yüksek olduğunu ve gürültü azaltma probleminde elde edilen sonucun daha iyi olduğunu ifade etmektedir.

1.6. Genel Yapısı ile Derin Öğrenme

1.6.1. Konvolüsyonel Sinir Ağları

Derin öğrenme çok sayıda farklı katmandan oluşan, biyolojik olarak insan beyni yapısından ilham alınarak tasarlanmış bir öğrenme tekniğidir. İnsan beyni içerisinde görsel korteksteki farklı nöronların bir görüntü içerisinde yer alan farklı yapılara cevap verdiği bilinmektedir. Örneğin, belirli bir salınımaya sahip çizgilerin olduğu bir görüntüde yalnızca belirli bir grup nöronunda aktivasyon gözlenmektedir (Wani vd., 2020). Belirli bir nöron grubunun belirli bir yapıya cevap vermesi fikri konvolüsyonel (evrişimli) sinir ağlarının temelini oluşturmaktadır.

Derin kelimesi, birden fazla gizli katmana sahip mimariyi ifade etmektedir. Geleneksel makine öğrenme algoritmalarında, yapay sinir ağları (YSA) dahil, veriler/girdiler modele verilmeden önce öznitelik çıkarma (feature extraction) algoritmaları kullanılarak öznitelik vektörleri çıkarılır. Bu kavram derin sinir ağlarının ortaya çıkması ile rafa kaldırılmıştır. Derin öğrenme algoritmaları ham verileri kendi bünyesinde bulundurduğu birden çok gizli katman içerisinde çeşitli işlemlere tabi tutarak geleneksel makine öğrenmesi ve görüntü işleme yöntemlerinin aksine öğrenme işlemi sırasında yonteme uygun olarak kendi özniteliklerini (öznitelik vektörlerini) çıkarmaktadır. Konvolüsyonel sinir ağları (Convolutional Neural Network - CNN) derin öğrenme yöntemlerinden sadece biridir.

İlk konvolüsyonel sinir ağı modeli Yann LeCun tarafından 1988 yılında önerilen ve 1998 yılına kadar iyileştirilmeye devam edilen LeNet modeldir (LeCun vd., 1998). Çoğu YSA modelinde olduğu gibi eğitim sırasında büyük hesaplama maliyeti ve bellek gereksinimine sahip olan LeNet modeli bu gereksinimlerden dolayı ilk yıllara popülerite kazanamamış ve derin öğrenme çalışmalarında yavaşlama olmuştur. 2012 yılında yapılan nesne tanıma alanında büyük bir yarışma olan ImageNet'te Krizhevsky ve arkadaşlarının derin öğrenme yöntemlerinden biri olan CNN ağlarını kullanarak elde ettikleri büyük başarı bilim dünyasında büyük bir etkili yaratmış ve derin öğrenmenin popüleritesinin artmasını sağlamıştır (Krizhevsky vd., 2017).

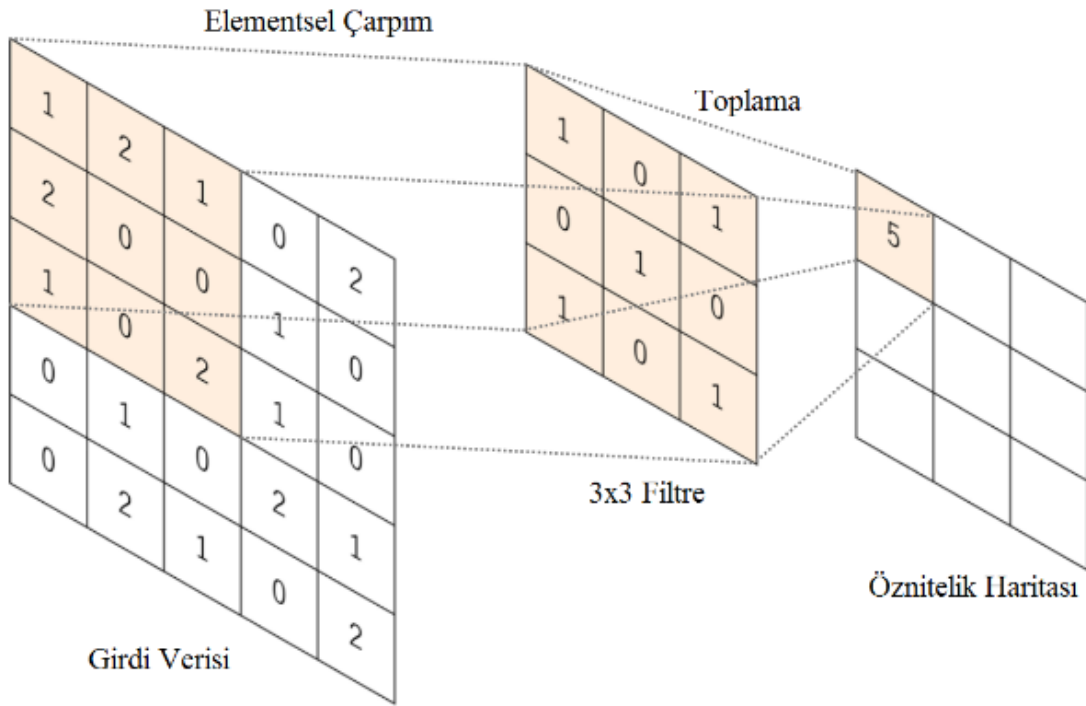
CNN ağları görüntü sınıflandırma, nesne algılama, doğal dil işleme ve tıbbi görüntü analizi gibi birçok alanda kullanılmakta olup, başarılı sonuçlar vermektedir. CNN ağlarında genel amaç görüntü vb. karmaşık yapıdaki girdi verilerinden yerel öznitelikler elde etmek ve bu öznitelikleri bir araya getirerek daha karmaşık gösterimler oluşturmaktır. Buna ek olarak, çok katmanlı bir mimari yapıya sahip olduğu için yüksek bir hesaplama gücüne ihtiyaç duymakta olup, büyük veri setleri için tasarlanan ağ yapılarında eğitim aşaması standart işlemciler için haftalar hatta aylar sürebilmektedir. Bu yüzden, CNN gibi ağ yapılarının eğitimi genellikle grafik işlem birimleri (GPU) kullanılarak gerçekleştirilmektedir.

Genel yapısı itibariyle CNN ağlarında, normalize edilmiş görüntülerin alındığı bir giriş katmanı, bir dizi konvolüsyon ve alt-örnekleme işlemlerinin gerçekleştirildiği ara katmanlar ve tam-bağlı yapıdaki ağdan oluşan bir çıkış katmanı bulunmaktadır. CNN ağının en dikkat çekici özelliği ise bir katmanda yer alan her bir düğümün, bir önceki katmanda bulunan sınırlı bir uzamsal bölgeden girdi verisi almasıdır (Gu vd., 2018). Bu işlem ise lineer bir operatör olan konvolüsyon işlemi ile yapılmaktadır. Bu mimaride yer alan görsel sistem, tüm görüntünün küçük bir alt kümesine duyarlı olan, algılayıcı alan (receptive field) olarak tanımlanan, karmaşık hücre kümelerinden oluşmaktadır. Konvolüsyon işlemi sırasında görüntünün tamamıyla değerlendirilebilmesi için bu hücre kümelerine karşılık gelen konvolüsyon çekirdekleri (filtreler) görüntü üzerinde adım adım kaydırılmaktadır. Bu filtreler aracılığıyla yapılan konvolüsyon işlemi neticesinde bir görüntüden yönlü kenar bilgileri, köşe bilgileri ve uç noktalar gibi kullanışlı öznitelikler çıkarabilmektedir. Daha sonra, bu temel öznitelikler, üst seviye özniteliklerin elde edilmesi amacıyla sonraki katmanlarda birleştirilmektedir (O'Shea & Nash, 2015). Her bir filtre kendi içinde bir dizi ağırlık ile tanımlanmakta olup bu ağırlıklar görüntüde filtrenin konumlandığı alan üzerinde yapılan çarpma işleminde çarpan olarak kullanılmakta, çarpımların toplamı ileri beslemeli ağda sonraki katmanlara iletilmektedir. Bu çarpma işlemi aynı ağırlıklar kullanılarak görüntünün tamamında tekrarlanmaktadır. Bir görüntünün üzerine bir filtre ile konvolüsyon işlemi gerçekleştirildiğinde çıktı, filtrenin her bir uzamsal bölgedeki çarpımından elde edilen yeni bir görüntü olmaktadır. Bir öznitelik haritası olarak kabul edilen bu görüntüde, giriş görüntüsünün kenar, dikey/yatay çizgi veya köşe gibi öznitelikleri açığa çıkarılmaktadır.

1.6.2. Konvolüsyon Katmanı

Veriler ham olarak CNN ağının ilk katmanı olan konvolüsyon katmanına verilir. Giriş görüntülerinin boyutunun büyük veya küçük olması tasarlanacak modelin başarımında önemlidir. Büyük boyuta sahip giriş görüntülerinin kullanımı ağın başarısını artırabilirken, kullanılacak parametrelerin sayısının çok olması da yüksek bellek ihtiyacından dolayı eğitim süresini artırabilmektedir. Buna ek olarak test aşamasında görüntü başına test süresini de arttırmaktadır. Küçük boyuta sahip görüntülerin kullanılması durumunda ağın derinliği azalacağından eğitim süresi kısalmış ve bellek ihtiyacı azalır. Ancak buna ilaveten tasarlanan ağın performansı da düşebilir. Giriş görüntü boyutu; ağ derinliği, donanımsal hesaplama maliyeti ve ağ başarısı için uygun seçilmelidir (Özkan & Ülker, 2017).

Konvolüsyon katmanı ham girdi verilerinden öznitelik(özellik) çıkaran ilk katman olarak düşünülebilir. Konvolüsyon işlemi, görüntü ile bir filtre/çekirdek gibi iki matrisin matematiksel bir işlemidir. Bu işlem içerisinde, belirli bir boyuttaki filtreyi görüntünün sol üst köşesinden başlatarak sağ alt köşeye kadar tüm görüntü üzerinde adım adım kaydırılır. Filtre içerisinde özdeşleşen değerlerin bire bir çarpılması ve tüm değerlerin toplanması sonucunda elde edilen sonuç çıkış matrisinin ilgili elemanına kaydedilir. 3x3, 5x5, 7x7 gibi farklı boyutlarda olabilen bu filtreler ağırlık olarak isimlendirilen bir dizi sayıdan oluşur. Her konvolüsyon işlemi sonucunda 2 boyutlu bir aktivasyon haritası veya öznitelik haritası (feature map) elde edilmiş olur. Oluşturulan filtrelerin her biri giriş verilere ait farklı desenleri ve özellikleri öğrenir. CNN ağının eğitimi sırasında bu filtrelerin katsayıları, eğitim kümesindeki her öğrenme aşamasında güncellenir. Bu sayede CNN ağı, özelliklerin belirlenmesi için, verinin hangi bölgelerinin önem taşıdığını belirler. Konvolüsyon işlemine örnek gösterim Şekil 1.9'da verilmiştir.



Şekil 1.9. Konvolüsyon işlemine bir örnek

Konvolüsyonel sinir ağı içerisinde tasarlanan modelin davranışını kontrol etmek amacıyla bazı hiper parametreler kullanılmaktadır. Bu hiper parametrelerin bazıları çıkışın boyutunu kontrol ederken, bazıları modelin çalışma süresini ve bellek maliyetini ayarlamak için kullanılır. CNN mimarisinde konvolüsyon katmanı için kullanılan dört önemli hiper parametre aşağıda verilmektedir:

1. Filtre boyutu: konvolüsyon katmanlarında kullanılan filtre (konvolüsyon çekirdeği) boyutları giriş boyutundan daha küçük herhangi bir boyutta olabilmekte birlikte, CNN mimarilerinde genellikle 3x3 ile 11x11 arasında seçilmektedir. Filtre boyutu giriş boyutundan bağımsızdır.
2. Filtre sayısı: konvolüsyon katmanının içerisinde daha fazla sayıda filtre kullanılması, giriş görüntüsü üzerinde daha fazla sayıda öznelik tanımaya olanak sağlamaktadır. Ancak bu durum kullanılan ağı parametresini arttırmakla birlikte, ağı eğitimi zorlaştırmaktadır. Bu nedenle filtre sayısı belirlenirken optimum düzeyde seçilmesine dikkat edilmelidir.
3. Konvolüsyon adımı: konvolüsyon sırasında filtrenin giriş görüntüsü üzerindeki uzamsal olarak ne kadar kayma işlemi yapacağını ifade etmektedir. Konvolüsyon adımı filtre boyutuna uygun olarak seçilmelidir. Konvolüsyon işlemi sırasında

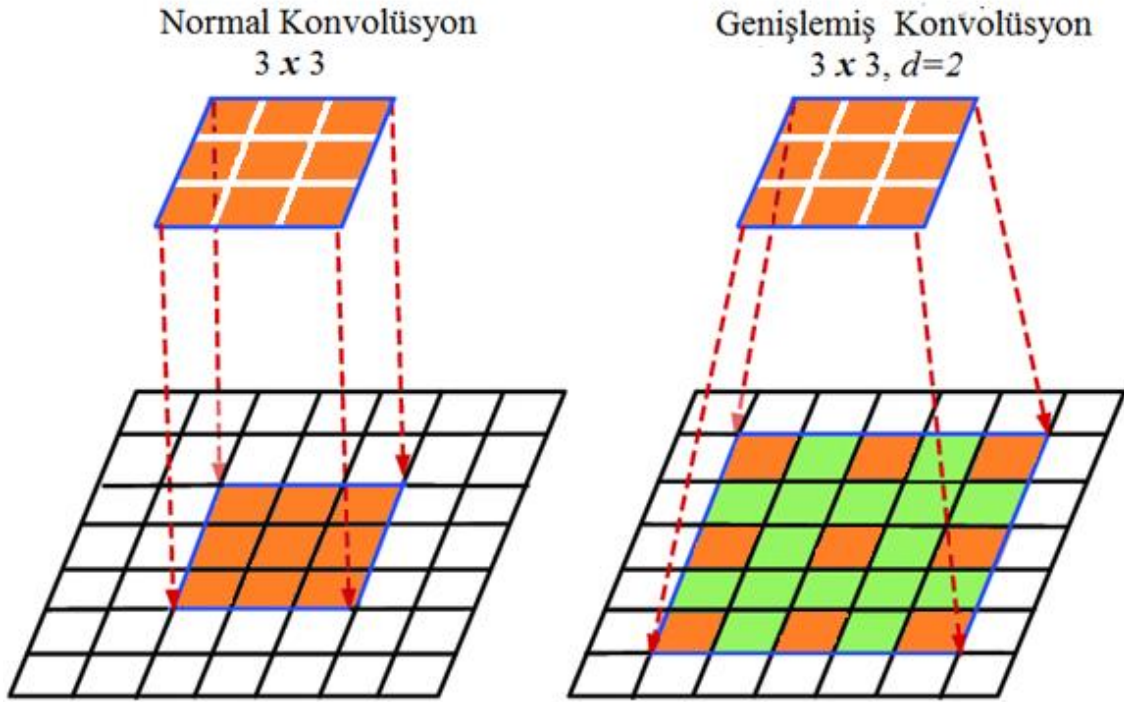
oluşacak olan çıktının da boyutunu belirlemekte olup, hesaplama maliyetinden kaçınmak için altörnekleme işlemi içinde kullanılmaktadır.

Doldurma: Bu hiper parametre girdi görüntüsü üzerinde sınır bölgelerde konvolüsyon işleminin nasıl yapılacağını ifade etmektedir. Sınır bölgesinde filtrenin merkez pikseli yerleştirildiğinde dışarıda kalan filtre ağırlıkları için yapılacak işlemi belirtmektedir. Doldurma işlemi, genelde sıfır ile doldurma ya da aynalama işlemi olarak iki farklı şekilde yapılabilmektedir. İlkinde sınır bölgelerin dışında kalan kısımlar 0 ile doldurulurken diğerinde sınır içerisindeki piksellerin simetriği alınmaktadır. Doldurma işlemi tasarıma bağlı olarak yapılmayabilir de ancak bunun girdi görüntüsünün boyutunu azalttığı unutulmamalıdır.

A. Genişlemiş Konvolüsyon

Bilindiği üzere bir CNN ağı için daha fazla öznitelik çıkarımı görüntü işleme açısından ağın başarımını arttırmaktadır. CNN ağının alıcı alanını (receptive field) arttırmak gürültü azaltma için daha fazla öznitelik çıkarmak açısından çok etkili bir yol olabilir. Bir CNN ağı için alıcı alanını arttırmanın temelde iki yolu vardır. Bunlardan ilki CNN ağının genişliğini arttırmak yani filtre boyutunu genişletmek, diğeri ise ağın derinliğini arttırmaktır yani konvolüsyon katmanları eklemektir. Ancak ilk yöntemde ağın genişliğini arttırmak daha fazla parametre üretimine neden olacaktır ve bu aşırı uydurma (overfitting) sorununa neden olabilir. Buna ek olarak hesaplama maliyetini de arttıracaktır. İkinci yöntem ise grandyanların kaybolması problemine neden olabilir.

Bu yöntemlere ek olarak alıcı alanını arttırmak için genişlemiş konvolüsyon (dilated convolution) yöntemi önerilmiştir (Yu & Koltun, 2015). Genişlemiş konvolüsyon, geleneksel konvolüsyon işleminde komşu pikseller arasında ilişkiye bakarken, komşu piksellerin ardışık elemanları arasına boşluklar bırakarak çekirdeğin boyutunu arttırma işlemi olarak düşünülebilir. Daha basit bir ifadeyle, konvolüsyon işlemi ile aynıdır, ancak girdinin daha geniş bir alanını kapsayacak şekilde piksel atlama içermektedir ve bu sayede hesaplama işlemine katılacak piksel sayısı değişmeden alıcı alan arttırılabilmektedir. Kısacası bu yöntem elde edilen bilgiyi arttırmak için f genişleme faktörüne bağlı olarak genişlemiş filtre kullanılmaktadır. Bu filtre $(2f + 1) \times (2f + 1)$ boyutunda bir filtre olarak ifade edilebilir. Genişlemiş konvolüsyon işlemine örnek gösterim Şekil 1.10'da verilmiştir.



Şekil 1.10. Genişlemiş konvölüsyon

CNN ağlarının eğitim sırasında, etiketli girdi verileri kullanılarak ağın en yüksek başarıyı verecek parametrelerin kestirilmesi yani hesaplanması hedeflenmektedir. Doğru model parametreleri ile ağın yeni örnekler üzerinde sınıf etiketlerinin doğru bir şekilde bulunması amaçlanmaktadır.

Eğitim sırasında parametrelerinin optimizasyonu için bir kayıp fonksiyonu (loss function) belirlenmekte ve CNN ağının bu kayıp fonksiyonunu minimize edecek şekilde parametre güncellemesi yapması beklenmektedir. Kullanılacak olan kayıp fonksiyonu ele alınan probleme ve modele bağlı olarak değişmektedir. Minimize etme işlemi için CNN ağlarında çeşitli şekillerde optimizasyon işlemi yapılmakta ve çeşitli optimizasyon algoritmaları kullanılmaktadır.

1.6.3. Aktivasyon Fonksiyonu

Derin öğrenme algoritmalarında kullanılan aktivasyon fonksiyonları, sinir ağlarının öğrenme kapasitesini artırmak ve daha kompleks işlevler öğrenmelerine yardımcı olmak için kullanılır. Aktivasyon fonksiyonları, sinir ağlarının doğrusal olmayan işlevler öğrenmesine olanak tanır. Eğer doğrusal bir aktivasyon fonksiyonu kullanılırsa, çok katmanlı yapılar bir

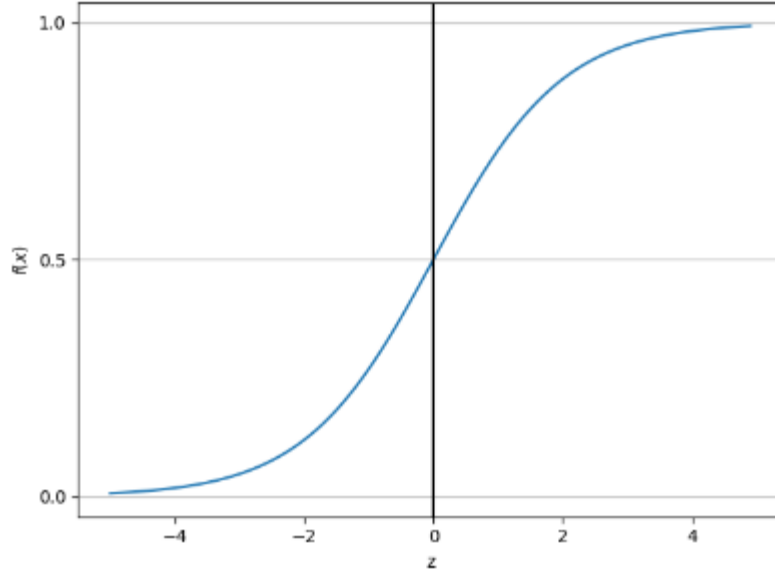
araya geldiğinde bile sonuç hala doğrusal olacaktır. Bu, sinir ağlarının çok karmaşık ve esnek olmalarını sınırlar.

A. Sigmoid Fonksiyonu

Lineer olmayan aktivasyon fonksiyonlarından biri olan Sigmoid aktivasyon fonksiyonu girdi verilerini alarak bu değerleri [0-1] aralığına sıkıştırarak bir çıktı üretmektedir. Genel olarak sigmoid aktivasyon fonksiyonu için girdi ne kadar büyük olursa çıktı o kadar 1 sayısına yakınsamakta ve aynı şekilde girdi ne kadar küçükse (negatif yönde) çıktı o kadar 0 sayısına yakınsamaktadır. Sigmoid fonksiyonun matematiksel gösterimi Eşitlik (1.11)'deki gibi verilebilir.

$$f(x) = \frac{1}{1+e^{-x}} \quad (1.11)$$

Olasılığı bir çıktı olarak tahmin etmemiz gereken modeller için çıkış katmanında sıklıkla kullanılmaktadır. Çünkü bir olasılık değeri [0-1] aralığında değiştiği için Sigmoid kullanımı doğru seçim olacaktır. Ayrıca türevlenebilir ve yumuşak gradyan geçişleri sağladığı için üretilecek olası çıktılarda sert değişimlerin önüne geçmektedir. Sigmoid fonksiyonuna ait grafiksel gösterim Şekil 1.11'deki gibidir.



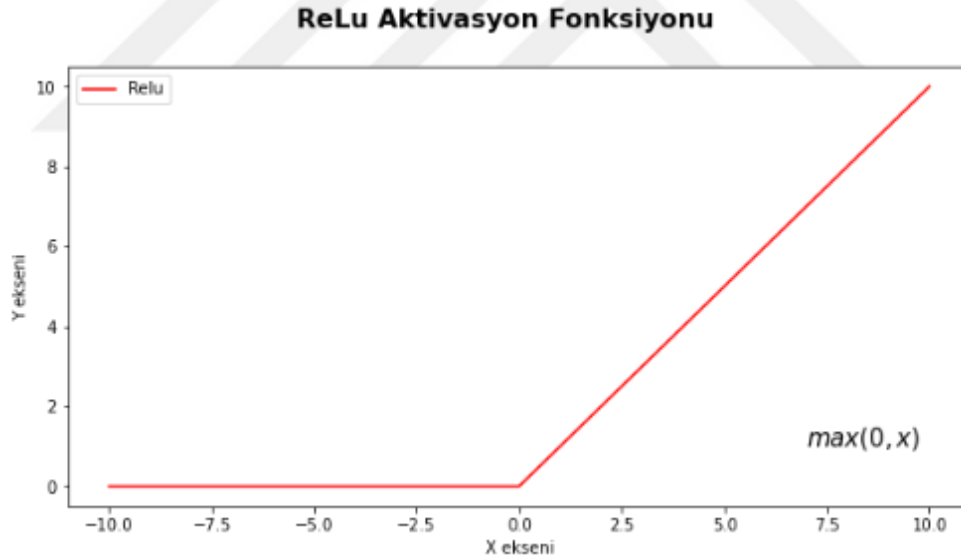
Şekil 1.11. Sigmoid fonksiyonu

B. Düzleştirilmiş Doğrusal Birim (ReLU)

Hesaplanma yöntemi $f(x) = \max(0, x)$ şeklinde olan ve genellikle konvolüsyon katmanına bağlanan düzleştirilmiş doğrusal birim (Rectified Linear Unit - ReLU) fonksiyonu son zamanlarda yaygın şekilde kullanılan bir aktivasyon fonksiyonudur (Nair & Hinton, 2010). Giriş verilerinden negatif değerleri sıfır yaparken pozitif değerleri olduğu gibi bırakan ReLU aktivasyon fonksiyonu, giriş verilerindeki gürültü gibi küçük değişikliklere karşı dirençli bir yapıya sahiptir ve CNN ağlarından doğrusal olmayan bir yapı kazandırmakla birlikte (nonlinearity) ağı daha hızlı öğrenmesini sağlar. Buna göre ReLU aktivasyon fonksiyonunun hesaplanması Eşitlik (1.12)'deki gibidir.

$$f(x) = \begin{cases} x_i & \text{eğer } x_i \geq 0 \\ 0 & \text{eğer } x_i < 0 \end{cases} \quad (1.12)$$

ReLU aktivasyon fonksiyonu için grafiksel gösterim Şekil 1.12'de sunulmuştur.



Şekil 1.12. ReLU aktivasyon fonksiyonu

Bazı aktivasyon fonksiyonları daha fazla bilgi taşıma kapasitesine sahiptir. Örneğin, sigmoid ve tanjant fonksiyonları, girişlerin sınırlı bir aralığa sıkıştırılmasını sağlar, bu da daha fazla bilgi taşıma kapasitesine sahip olmalarını engeller. ReLU gibi fonksiyonlar ise bu sınırlama yapmazlar ve bu nedenle de çokça tercih edilmektedirler. Ancak Şekil 1.12'de de görüleceği üzere ReLU aktivasyon fonksiyonu bazı bölgelerde sürekli değildir (sıfırdan

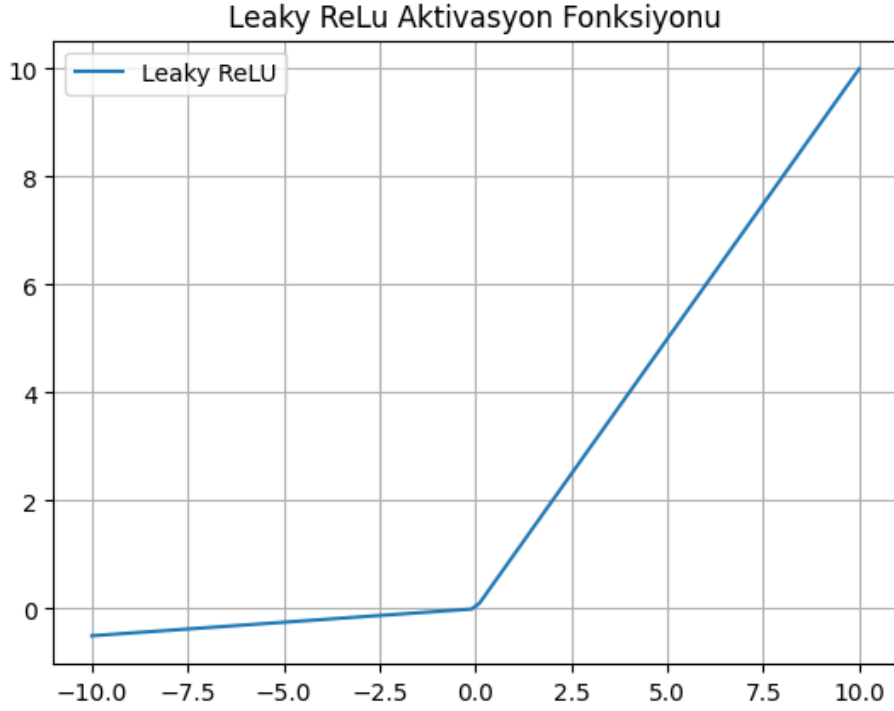
küçük olan değerler için). Bu nedenle bu bölgelerde gradyanlar sıfır olarak hesaplanarak ölü nöronların oluşmasına neden olmaktadır ve bu da CNN ağının eğitimini zorlaştırabilmektedir. Bunu aşmak için literatürde ReLU aktivasyon fonksiyonunun türevleri önerilmiştir.

C. Leaky ReLU (LReLU)

ReLU aktivasyon fonksiyonu sıfır ve negatif değerler için sıfır sayısını üretmekte ve bu da o sayıyı içeren nöron için aktivasyon işleminin olmaması ve nöronun ölü kalmasına neden olmaktadır. Bu sorunu aşabilmek için literatürde ReLU aktivasyon fonksiyonunun bir türevi olan Leaky ReLU (LReLU) önerilmiştir (Maas vd., 2013). Buna göre sıfırdan küçük değerler için aktivasyon işleminde direkt olarak sıfır atamak yerine sıfıra çok yakın bir değer alınmaktadır. Buna göre LReLU Eşitlik (1.13)'deki gibidir:

$$f(x) = \begin{cases} x_i & \text{eğer } x_i \geq 0 \\ \frac{x_i}{a} & \text{eğer } x_i < 0 \end{cases} \quad (1.13)$$

burada a değeri $(1, +\infty)$ arasında bir değer almakla birlikte Maas'a (2013) göre 100'den daha büyük bir sayı olarak alınması önerilmektedir. Şekil 1.13'te LReLU aktivasyon fonksiyonunun gösterimi yer almaktadır. LReLU aktivasyon fonksiyonunun kullanımı noktasında bir dezavantaj olarak a parametresinin seçiminin probleme özgü olarak ne seçileceğinin bilinmemesi söylenebilir.



Şekil 1.13. Leaky ReLU aktivasyon fonksiyonu

D. Parametrik ReLU (PReLU)

LReLU aktivasyon fonksiyonunda yani Eşitlik (1.13)'de yer alan a seçiminin ilgili problem için seçimi bir hiper parametre sorunu olmakla birlikte sürekli olarak farklı sayılar denenerek karar verilmesi sorununa yol açmaktadır. Bu parametrenin seçiminin optimal olmasının CNN ağının kendisi tarafından belirlenmesi için parametrik ReLU (PReLU) fonksiyonu önerilmiştir (He vd., 2015). Buna göre a sayısı CNN ağı için kestirimi yapılacak yeni bir parametre değerini almakta ve geri yayınımla eğitim sürecinde optimal değer belirlenmektedir. Bir dezavantaj olarak a parametresinin kestirimi CNN ağı üzerinde ekstra bir hesaplama ve eğitim maliyeti getirmesi söylenebilir.

1.6.4. Yiğın Normalizasyonu

Yapay sinir ağlarının eğitimi karmaşık bir işlemdir. Bunun en büyük nedenlerinden bir tanesi verilerin sayısal değerlerinin dağılımının değişmesidir. Geleneksel makine öğrenmesi ve yapay sinir ağları eğitimi sırasında veriler girdi olarak verilirken bir normalizasyon işlemine tabi tutulur. Bu sayede farklı ölçeklerdeki girdi verileri aynı aralığa normalize edilmiş olur. Aynı zamanda bir sinir ağı içerisinde geleneksel olarak kullanılan Sigmoid ve

Tanjant aktivasyon fonksiyonları aracılığıyla sıkıştırma yapılarak (sigmoid [0-1] ve ya tanjant [-1,+1]) ağ içerisindeki kovaryans kaymasının da (internal covariate shift) önüne geçilmektedir. Ancak CNN ağlarında katmanlar arasında çokça kullanılan ReLU aktivasyon fonksiyonu yapısı gereği doğrusal sonuçlar üretmekte ve bu da sayısal verilerin dağılımını çokça değiştirmektedir. Bu ise sinir ağı içerisinde kovaryans kaymasına ve standartlaşmadan uzaklaşma sorununa neden olmaktadır. Ağ içerisindeki kovaryans kaymasını bir örnek ile anlatacak olursak; örneğin ayakkabı tanımak için bir CNN ağı tasarlayalım ve veri setimiz sadece beyaz spor ayakkabılardan oluşsun. Eğer tanınması için CNN ağına siyah renkte bir ayakkabıyı girdi olarak verirsek güzel sonuçlar elde edemeyebiliriz. Bunun en büyük nedeni CNN ağının katmanları arasında elde edilen sayısal sonuçların kovaryansının çok olması ve ağın genelleştirme hatasının (generalization error) artmasıdır.

Bu sorunu aşmak için Ioffe ve Szegedy yığın normalizasyonunu (Batch Normalization – BN) önerdiler (Ioffe & Szegedy, 2015). Genel mantık yığın halinde eğitilen bir CNN ağı için katmanlar arasında üretilen sayısal sonuçların bir sonraki katmana iletilmeden önce ortalama ve standart sapmaya bağlı olarak normalize edilmesi işlemidir. Ioffe ve Szegedy yaptıkları çalışmalarında yığın normalizasyonunun ağın hızlı yakınsamasına katkı sağladığını ve genelleştirme hatasının azaltarak (generalization error reducing) başarımı arttırdığını göstermişlerdir. Yığın normalizasyonunun formülü Eşitlik (1.14) ve (1.15)'deki gibidir:

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 - \epsilon}} \quad (1.14)$$

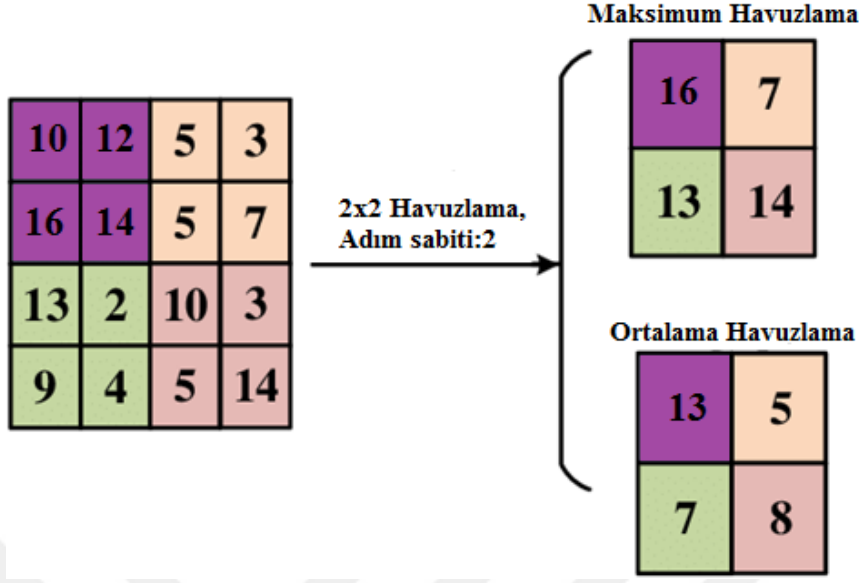
$$y_i = \gamma \hat{x}_i + \beta \quad (1.15)$$

burada μ_B yığının ortalamasını, σ_B yığının standart sapmasını, γ ve β ise sırasıyla ölçeklendirme ve kaydırma parametrelerini ifade etmektedir.

1.6.5. Havuzlama Katmanı

Konvolüsyon katmanları girdi görüntüsü üzerinde belirli bir pencere boyutu içerisinde konvolüsyon işlemi uygulayarak öznitelik çıkarımı yapmaktadır ve çok başarılı sonuçlar ürettiği literatürde bilinmektedir. Ancak konvolüsyon katmanlarının girdi görüntüsü üzerinde özniteliklerin kesin konumunu kaydetmesi bir dezavantaj oluşturmaktadır. Bunun sonucu olarak girdi öznitelik haritasındaki küçük kaymalar farklı özniteliklerin oluşumuna neden olabilmektedir. Bu durum ayrıca girdi görüntüsü üzerindeki rotasyon, kaydırma gibi küçük değişikliklerde olabilir. Bunun önüne geçebilmek için literatürde alt-örnekleme yapılmakta ve girdi öznitelik haritasının önemli bilgiler ve yapılar içeren düşük çözünürlüklü bir versiyonu girdi verisi olarak kullanılmaktadır. CNN ağlarında ise bu işlem adım sayısını değiştirerek ya da havuzlama katmanları kullanılarak yapılmaktadır (Gholamalinezhad & Khosravi, 2020).

Havuzlama katmanı (pooling layer) konvolüsyon katmanları arasında sıklıkla eklenen bir katmandır ve maksimum ya da ortalama havuzlama olmak üzere iki farklı şekilde kullanılmaktadır. Kısaca havuzlama katmanı, konvolüsyon katmanlarına benzer olarak belirli bir pencere boyutu içerisinde imge üzerinde gezinerek, o pencere içerisinde kalan piksellerin ortalaması veya maksimum olanın alınmasıdır. Eğer seçilen pencere boyutu girdi görüntüsüne eşit olarak seçilirse global havuzlama katmanı olarak isimlendirilir. En büyük dezavantajları ise veri boyutunu azaltırken önemli bilgi kayıplarına neden olma ihtimalleridir. Havuzlama katmanına örnek Şekil 1.14'te verilmiştir.



Şekil 1.14. Ortalama ve maksimum havuzlama işlemi

1.6.6. Atlamalı Bağlantı

Atlamalı bağlantı (skip connections) bir CNN ağı içerisinde bir katmanın çıkışının kendisine bitişik olmayan başka bir katmanın girişine bağlayan bir kısayol olarak düşünülebilir. Bir ağ içerisinde sığ katmanlarda yer alan özniteliklerin daha derin katmanlara aktarılmasını sağlayan atlamalı bağlantılar artık bağlantı ya da yoğun bağlantı gibi farklı şekillerde kullanılabilir. Birçok katmandan geçerken kaybolabilecek veya seyrelebilecek bilgilerin korunmasına yardımcı olabileceği gibi, farklı seviye ya da çözünürlükteki özniteliklerin birleşmesine yardımcı olarak ağın temsil gücünü artırabilir ve gradyan yok olması probleminin önüne geçebilir ve böylece CNN ağlarında başarıyı artırabilir (Drozdzal vd., 2016). Bunun yanı sıra dezavantajlarından bahsedecek olursak karmaşıklığı ve bellek ihtiyacını arttırması söylenebilir. Farklı şekillerde atlamalı bağlantı türlerinin olduğu bilinmeli ve probleme ve tasarımı yapılan CNN ağına uygun olarak kullanılmalıdır.

1.6.7. Kayıp Fonksiyonu

Bir modelin eğitim aşamasında güncellenmesi işlemi hesaplanan bir hata oranına bağlı olarak yapılmaktadır. Aslında bu hata oranı aynı zamanda modelin başarısını da bize göstermektedir. Hata oranını ve dolayısıyla bir modelin başarımını ölçmek için kullanılan

fonksiyonlara kayıp fonksiyonları denmektedir. Belli bir sayıda örnek ağdan geçirildikten sonra modelin kestirimi ne kadar hata ile yaptığı hesaplanarak ağırlıkların güncellenmesi işlemi yapılır. Yapay sinir ağının amacı bu hata oranını minimize etmektir. Bu sebeple işleme maliyet fonksiyonu ya da amaç fonksiyonu da denilmektedir. Hesaplanan hata bize modelin ağırlıklarının nihai istenilen modelden ne kadar uzakta olduğunu göstermektedir. Hata ne kadar büyükse mevcut model nihai modelden o kadar uzak anlamına gelmektedir. Hata değerinin az olması modelin başarımının yüksek olduğunu ifade etmektedir. Nihai model hata oranı “0” olan modeldir ve ulaşılmak istenen hedef budur.

Yapay sinir ağlarının normalize edilmesi için kullanılan bu kayıp fonksiyonlarının birçok türü bulunmaktadır. Bunlara örnek olarak ortalama kare hatası (OKH), ortalama mutlak hata (OMH) veya entropi verilebilir. Bu kayıp fonksiyonları kullanılarak klasik bir yapay sinir ağı içerisinde yer alan genellikle “W” ile gösterilen ağı ağırlıklarının ve “b” ile gösterilen bias parametrelerinin normalizasyon işlemi gerçekleştirilmektedir. Buna göre gürültü azaltma işlemi için OKH altında L isimli CNN ağı için kayıp fonksiyonu Eşitlik (1.16)’daki gibi verilebilir.

$$L(\theta) = \frac{1}{2N} \sum_{i=1}^N \|f_L(I_C - I_E)\|^2 \quad (1.16)$$

Eşitlik (1.16)’da θ kestirimi yapılacak ağ parametrelerini, I_C ve I_E sırasıyla orijinal ve ağı ürettiği gürültüsüz azaltılmış imgeyi ifade etmektedir. CNN ağı boyunca gradyanların hesaplanarak konvolüsyon çekirdeklerinin ağırlıklarının güncellenmesi yani optimizasyon işlemi 3 farklı şekilde yapılmaktadır. Bu 3 farklı gradyan güncellenmesi stokastik, yığın ve küçük yığın şeklinde isimlendirilebilir.

Stokastik optimizasyon eğitim veri setinin sadece bir örneğini kullanarak her bir eğitim döneminde ağı güncelleyen optimizasyon işlemidir. Her bir örnek için gradyan hesabı ve güncelleme yapılması ek hesaplama maliyeti getireceği gibi tek bir örnek için ağı güncellemesi CNN ağının yerel minimumlara takılması ihtimalini arttıracaktır. Yığın optimizasyon işlemi ise eğitim veri setinin tamamını kullanarak her bir eğitim döneminde ağı güncelleyen bir optimizasyon algoritmasıdır. Bütün veri setinin aynı anda ağa verilerek eğitimi GPU belleğine yük getirebilmektedir ve tüm veri seti için gradyan hesaplamalarının aynı anda yapılması tahmin hatalarının birikmesi nedeniyle ek karmaşıklık getirebilir. Derin öğrenme algoritmalarında en çok tercih edilen küçük-yığın optimizasyonu ile öğrenme işlemi ise büyük veri setlerinde daha hızlı ve verimli bir şekilde eğitim yapmayı sağlayan ve

veri setini küçük gruplara (mini-batch) bölünmesi ve her bir küçük yığın için CNN ağının ağırlıklarının güncellenmesi işlemidir. Stokastik optimizasyona göre yerel minimumlara takılma ihtimalinin azaltılması ve yığın optimizasyona göre hesaplama maliyetinin düşürülmesinin yanı sıra GPU belleği üzerindeki yükü azaltması küçük-yığın optimizasyonunu en çok tercih edilen optimizasyon türü yapmaktadır. Tercih edilen küçük-yığın boyutlarının seçilmesi bir hiper parametre seçimi olmakla beraber, uygun boyutun seçilmesi CNN ağının eğitiminde başarıyı etkilemektedir. Literatürde genellikle küçük-yığın boyutu 2 sayısının kuvvetleri olarak seçilmektedir.

Sinir ağının eğitimi sırasında bütün veri setinin ağ üzerinde ileri yönde bir kere yayılımına devir (epoch-epok), her ağırlık güncellenmesi işlemine ise (küçük-yığın veya stokastik) iterasyon denilmektedir.

1.6.8. Adam Optimizasyon Algoritması

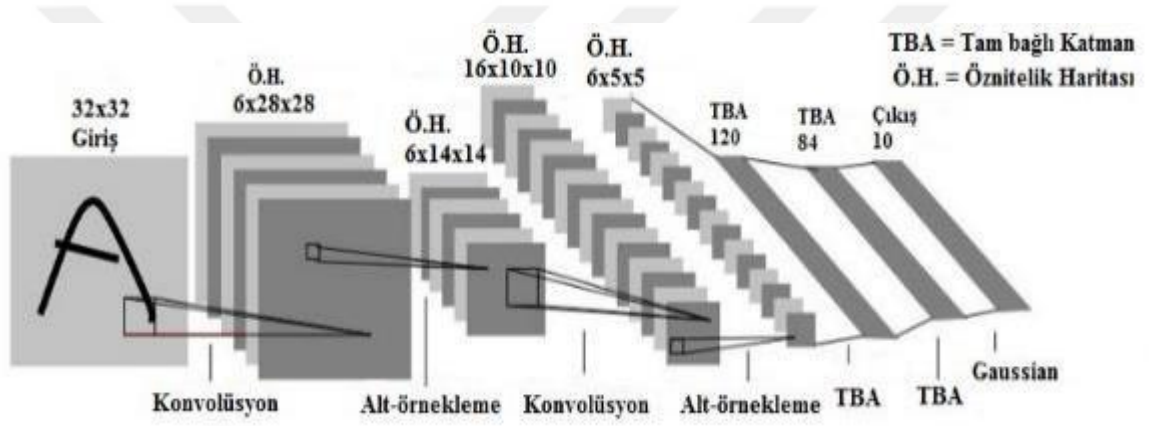
Yapay sinir ağlarının yanı sıra konvolüsyonel sinir ağlarında da hata fonksiyonun minimize edilmesi için ağırlıkların güncellenmesi ve bu sayede ağın öğrenmesi işlemi gradyanların hesaplanması yoluyla gerçekleşmektedir. Türev alma işlemi ile gradyanlar hesaplanarak değişimin gözlemlenmesi ve türevin karşıt yönünde küçük adımlar atarak azaltmaya gradyan iniş yöntemi denilmektedir. Tüm veri seti yerine, bu veri seti içerisinde seçilen küçük yığınlar üzerinde gradyanın azaltılması işlemi ise Olasılıksal Gradyan İniş (SGD) ile yapılmaktadır (Saad, 1998). SGD algoritmasına gradyanın hesabının yanı sıra daha etkili minimuma yakınsama için momentum değerleri de eklenmiştir. Momentum mevcut gradyan değerlerinin yanı sıra geçmişe ait gradyan değerlerinin dikkate alınmasını ifade etmektedir (Polyak, 1964). Bunların yanı sıra öğrenme sırasında ağırlıkların güncellenmesi yapılarak kullanılacak olan öğrenme oranı da ayarlanması gereken bir hiper parametredir. Çünkü doğru bir minimuma yakınsama için her bir ağırlığın güncellenmesini farklı oranda yapmak etkili bir yöntem olarak düşünülebilir. Bu nedenle literatürde öğrenme oranlarını uyarlayan optimizasyon algoritmaları da önerilmiştir. Adam optimizasyon algoritması da bunlardan bir tanesidir (Kingma & Ba, 2015).

Olasılıksal Gradyan İniş (SGD) algoritmasının bir uzantısı olan Adam optimizasyon algoritması derin öğrenme ağlarında çok sıklıkla kullanılan bir optimizasyon algoritmasıdır. Momentum kullanımının yanı sıra, parametreler için öğrenme oranının adaptif olarak belirlenmesi Adam algoritmasını gürültü azaltma alanında çokça kullanılan bir yöntem haline

getirmiştir (Tian vd., 2019). Bu nedenle bu çalışmada da Adam optimizasyon algoritması kullanılmıştır.

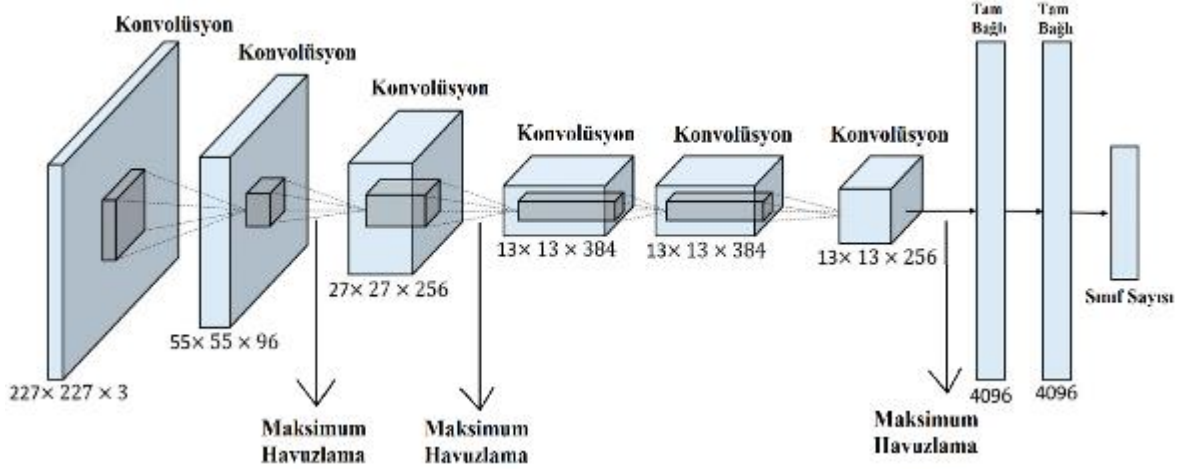
1.6.9. Genel CNN Mimarileri

Konvolüsyonel sinir ağlarının gelişimi ilk olarak Yann LeCun tarafından el yazımı rakamların tanınması için MNIST veri seti üzerin eğitilmiş LeNet modelidir (LeCun vd., 1998). Ağ içerisinde konvolüsyon katmanlarının yanı sıra ortalama havuzlama kullanılmış ve aktivasyon fonksiyonu olarak ise sigmoid ve hiperbolik tanjant fonksiyonlarından faydalanılmıştır. Şekil 1.15'te LeNet mimarisi gösterilmiştir.



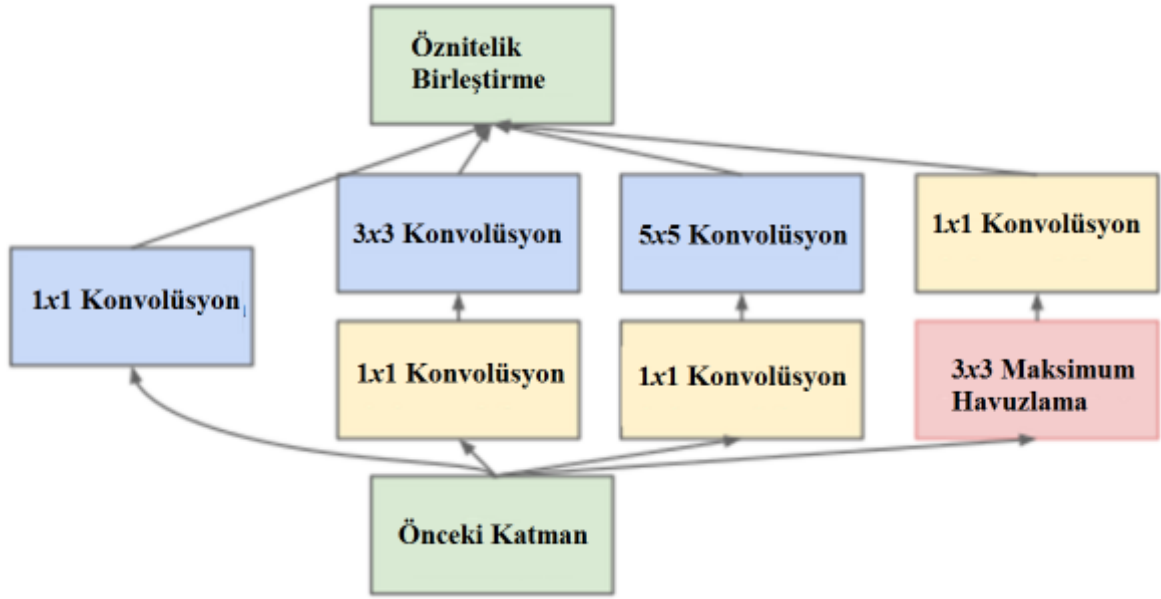
Şekil 1.15. LeNet mimarisi [16]

İlk CNN modeli 1998 yılında önerilmesine rağmen yüksek hesaplama maliyetinden ve GPU mimarisinin tam olarak gelişmemiş olması, derin öğrenme kavramının popülerlik kazanmasının 2012 yılında AlexNet mimarisinin doğuşuna kadar ertelemiştir (Krizhevsky vd., 2017). 9 katmanlı AlexNet modelinde farklı olarak ReLU aktivasyon fonksiyonu ve maksimum havuzlama kullanılmıştır. AlexNet mimarisi Şekil 1.16'da gösterilmiştir.



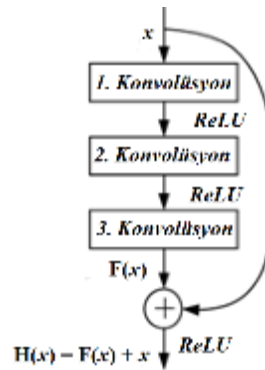
Şekil 1.16. AlexNet mimarisi [17]

Szegedy ve arkadaşları 2015 yılında InceptionNet isimli yeni bir model önerdiler (Szegedy vd., 2015). Bu model içerisinde CNN ağlarında hesaplama maliyetini ve ağın derinliğini azaltmak için tek bir katmanda birden fazla konvolüsyon işlemi yapılarak bunların birleştirilmesi kavramını tanıttılar. Ayrıca parametre sayısını ve boyutsallığı azaltmak için 1×1 boyutunda konvolüsyon çekirdeklerinin kullandılar. Bu sayede parametre sayısını %90 oranında azaltmayı başarmışlardır. Mimari 9 adet Inception olarak isimlendirilen modülden oluşmaktadır. Inception modülüne ait örnek gösterim Şekil 1.17'de verilmiştir.



Şekil 1.17. Inception modülü

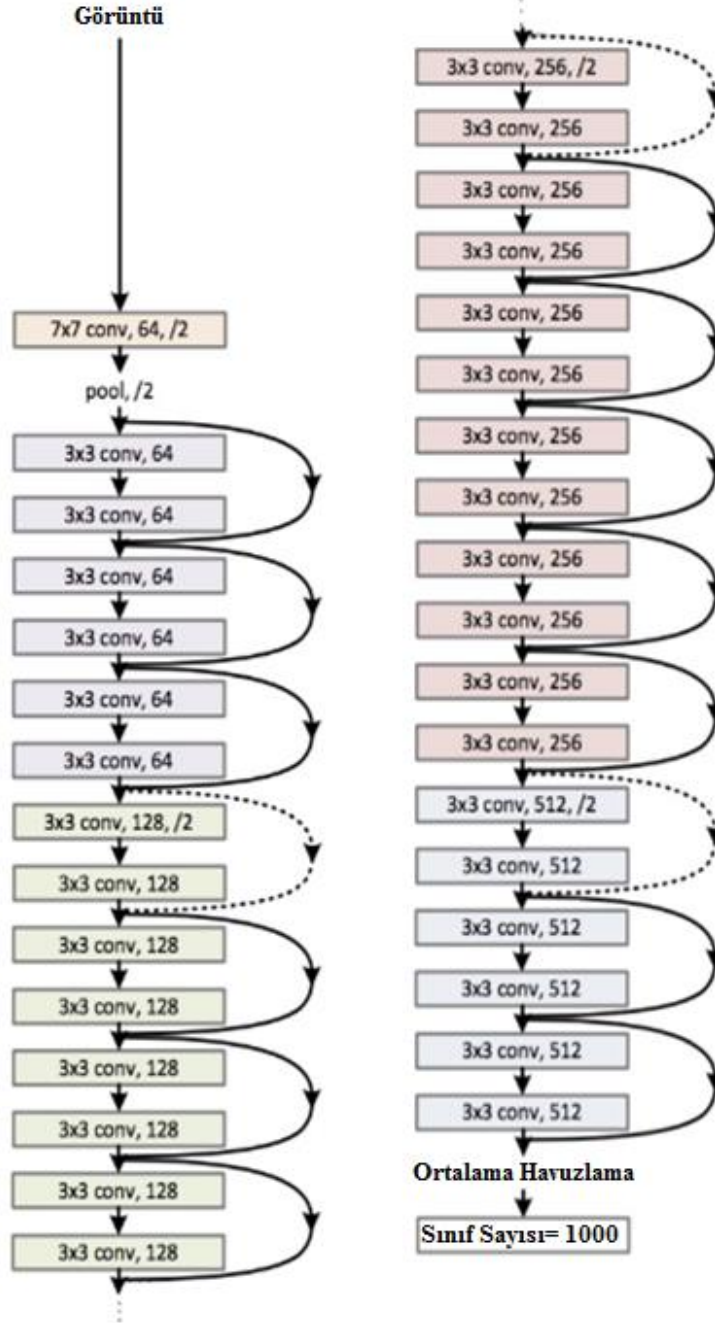
He ve arkadaşları tarafından ResNet isimli yeni bir model önerildi (He vd., 2016). Bu modelin en büyük özelliği atlamalı bağlantılar (skip connections) kullanıyor olmasıydı. Bu sayede sığ katmanlardaki öznitelikler ağına daha derin katmanlarına toplama yoluyla bağlanarak derin CNN ağlarında çokça karşılaşılan gradyan kaybolması probleminin önüne geçilmiştir. ResNet mimarisinde atlamalı bağlantı artık bağlantı şeklinde gerçekleştirilmiştir. ResNet mimarisinde kullanılan artık atlamalı bağlantı Şekil 1.18’de verilmiştir.



Şekil 1.18. Artık atlamalı bağlantı

152 katmandan oluşan ResNet mimarisinin gösterimi örnek olarak 34 katman için Şekil 1.19’da verilmiştir.

34 katmanlı Mimari



Şekil 1.19. 34 katmanlı ResNet mimarisi

1.6.10. Öznitelik Odak Mekanizması

Odak mekanizması insanların görsel algılama sistemi üzerine ilham alınmış olan ve insanların bir görüntü üzerinde önemli olan kısımlara odaklanıp, gereksiz bilgi içeren kısımları göz ardı etmesi mantığı üzerine kurulmuş bir mekanizmadır (Guo vd., 2022). Yapay zekâ alanında öznitelik odak mekanizması aktif olarak özniteliklerin ağırlıklandırılarak önemlerin belirlenmesi işlemini yapan dinamik bir seçim yöntemi olarak düşünülebilir. Odak mekanizması kendi içerisinde sert ve yumuşak olmak üzere iki kategoride sınıflandırılabilir.

Yumuşak odak mekanizması, öznitelik haritası üzerinde özniteliklere (0-1) arasında bir değer ataması yaparak bütün öznitelikleri dikkate alır. Buna göre yumuşak odak mekanizması Eşitlik (1.17)'deki gibi tanımlanabilir:

$$y_i = \sum_i^n a_i h_i \quad (1.17)$$

burada a_i odak skorunu, h_i i. özniteliği ve y_i ağırlıklandırılmış i. özniteliği ifade etmektedir. Global odak mekanizması olarak da bilinmektedir çünkü girdi özniteliklerinin tamamı bir sonraki katmana iletilir.

Buna ek olarak sert odak mekanizması girdi özniteliklerin 0 ya da 1 değerini atar yani hangi özniteliğe bakılmalı ya da bakılmamalı karar verir. Genel olarak yerel odak mekanizması olarak da bilinmektedir çünkü girdi özniteliklerinin sadece belirli bir kısmı bir sonraki katmana iletilir. Buna göre sert odak mekanizması Eşitlik (1.18) ve Eşitlik (1.19)'daki gibi hesaplanmaktadır:

$$y_i = \sum_i^n s_i h_i \quad (1.18)$$

$$s_i \sim Multi(a_i) \quad (1.19)$$

burada $Multi(.)$ ifadesi multinoulli dağılımını ve s_i ise bu dağılım sonucundan elde edilen değişkeni ifade etmektedir. Multinoulli dağılımı yapısı gereği 0 ya da 1 değerini üretmekte ve bu sayede sert odak mekanizmasında bir bölgenin seçilip seçilmeyeceğine karar verilmektedir (Xu vd., 2015). Buna göre yumuşak ve sert odak mekanizmasına örnek Şekil 1.20'de verilmiştir.



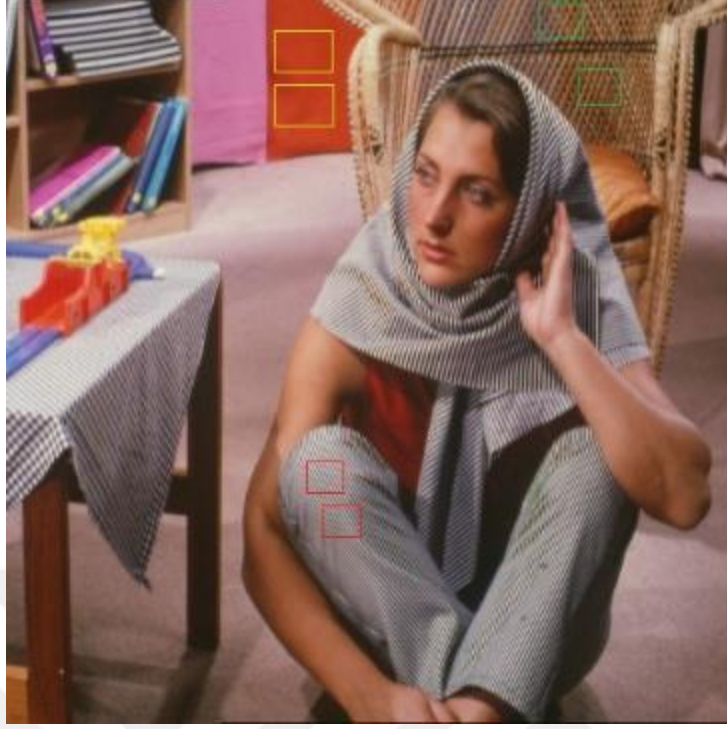
Şekil 1.20. Odak mekanizması, (a) sert odak, (b) yumuşak odak

Şekil 1.20'den de anlaşılacağı üzere sert odak mekanizması bir görüntüde sadece belirli kısımları dikkate alır ve gerisini elemine eder. Bu nedenle elemine edilen bölgelerde hesaplama yapılmayacağı için, hesaplama maliyeti açısından avantajlıdır. Ancak bu bölgelerin az da olsa öğrenme katkısı olabileceği ve atıldığı için bu bölgelerde gradyan yayılımı zorlaşacağı için eğitim zor hale gelebilir ve doğru bölgelerin seçiminin yapılamaması performans düşmesine neden olabilir. Yumuşak odak mekanizmasında ise bir görüntü üzerindeki bütün pikselleri (aynı oranda olmasa da) dikkate alır. Bu nedenle daha esnek ve sürekli bir dikkat dağılımı sağlar. Yani, her özelliğin belirli bir derecede önemi vardır. Ancak hesaplama açısından daha yoğun bir işlemdir, çünkü her öznelik için bir ağırlık öğrenilmesi gerekmektedir.

2. LİTERATÜR TARAMASI

Görüntü iyileştirme, genel olarak bir görüntünün anlaşılabilirliğini, okunabilirliğini bozan etkilerin görüntü üzerinden elemine edilmesi işlemi olarak düşünülebilir. Görüntü iyileştirme işlemlerinden bir tanesi ise gürültü azaltma işlemidir. Buna ek olarak JPEG sıkıştırma yöntemi sonucu oluşan bozulmaların kaldırılması, düşük çözünürlükteki imgenin yüksek çözünürlüğe artırılması (deblurring) işlemleri de görüntü iyileştirme teknikleri içerisinde yer almaktadır. Bu işlemlerin hepsi görüntü kalitesini arttırmakla beraber imgenin anlaşılabilirliğini arttırdığı için nesne sınıflandırma, nesne tanıma, hareket tespiti gibi işlemlerde de başarımın dolaylı olarak artmasını sağlamaktadır. Bu nedenle birçok yöntem için görüntü iyileştirme, önemli bir ön işlemi adımı olarak literatürde yerini almaktadır. Gürültü azaltma işlemi ise literatüre bakıldığında en çok üzerine çalışılan yöntem olarak görülebilir. Gürültü azaltma yöntemleri kendi içerisinde model tabanlı ve öğrenme tabanlı olmak üzere iki gruba ayrıştırılabilir.

Model tabanlı yöntemler daha çok uzamsal filtre tabanlı yöntemler (Wiener filtresi), dönüşüm tabanlı yöntemler ve seyreklik tabanlı yöntemler olarak düşünülebilir (Shao vd., 2013). Bu algoritmalar genellikle orijinal görüntünün belirli karakteristikler içerdiği varsayımına dayanmaktadır. Örnek verecek olursak yerel olmayan ortalamalar (Non-Local Means – NLM) yöntemi bir imge içerisinde benzer yapıların olduğu varsayımı üzerine imge içerisinde belli bir pencere boyutu içerisinde küçük parçalar arayarak orijinal piksel kestirimi için benzer imge parçalarının tamamı kullanılmakta ve orijinal piksel değerine daha iyi yakınsanmaktadır (Buades vd., 2005). NLM algoritmasının işleyişine örnek Şekil 2.1’de verilmiştir. BM3D algoritmasında ise NLM algoritmasına benzer şekilde ilk olarak 2 boyutlu benzer imge parçaları çıkarılmaktadır (Dabov vd., 2007). Daha sonra dönüşüm katmanında seyrekliği azaltmak için bu imge parçaları 3 boyutlu bir veriye dönüştürülmektedir. Sonrasında ise dalgacık dönüşüm gibi bir dönüşüme tabi tutularak sert eşikleme (hard thresholding) yapılarak 0’dan küçük değer atılarak büyük değerlere dokunulmadan ters dönüşüm ile 2 boyutlu imge elde edilir. Bu işlem 2 defa yapılmakta olup, ilkinden farklı olarak sert eşikleme yerine Wiener filtresi kullanılmaktadır. En son aşama olarak ise elde edilen sonuçlar birleştirilmekte ve nihai imge elde edilmektedir.



Şekil 2.1. Benzer karakteristiğe sahip imge parçaları içeren Barbara görüntüsü

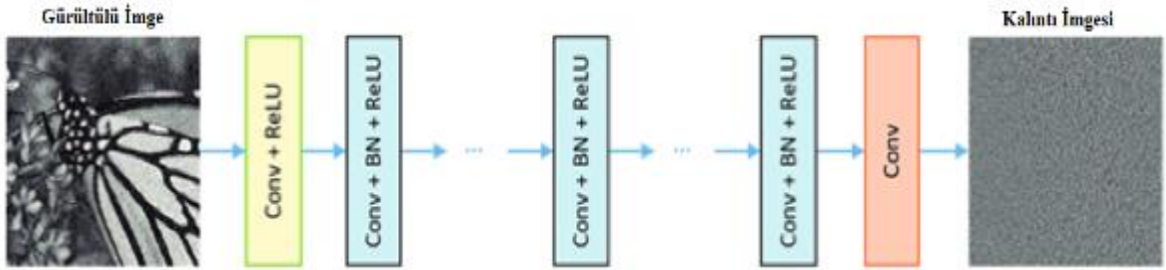
Başka örnek yöntemler olarak ise seyrek kodlamadaki gösterimi merkezileştiren bir yöntem olan NCSR (Dong vd., 2012) yöntemi ve düşük dereceli temsil (low-rank representation) ve nükleer norm minimizasyonu kullanan WNNM (Gu vd., 2019) algoritmaları verilebilir. Bu yöntemler gürültü azaltma işlemi için genel olarak başarılı sonuçlar verse de 3 tane büyük dezavantajları bulunmaktadır. İlk olarak iteratif yöntemler oldukları için büyük bir hesaplama maliyetleri vardır. İkinci olarak manuel olarak parametre ayarı gerektirmektedirler. Örneğin Gauss gürültüsünün giderimi için kullanılan ortalama filtresinde, filtre boyutunun ne olacağına belirlenmesi ya da NLM yönteminde benzer küçük imge parçaları aranırken tercih edilecek pencere boyutunun belirlenmesi örnek olarak düşünülebilir. Son olarak ise gürültü azaltmanın yanı sıra, JPEG sonucu oluşan dezenformasyonların giderilmesi, düşük çözünürlükteki (bulanık yapıdaki) görüntülerin keskinleştirme ile iyileştirilmesi gibi yöntemleri aynı anda yapacak şekilde bir beceriye sahip değildirler. Kısacası birden fazla farklı görüntü iyileştirme yöntemlerini aynı anda yerine getiremezler.

Diğer taraftan öğrenme tabanlı yöntemler ise daha çok gürültülü ve gürültüsüz imge arasında bir haritalama fonksiyonunun (mapping function) kestirimi ile gürültüyü

görüntüden elemine eden yöntemlerdir. Özellikle yapay sinir ağları ile gürültü azaltma için başarılı bir ağ yapısı ilk olarak Chen ve arkadaşları tarafından TNRD ağı önerilmiştir (Chen & Pock, 2016). Buna göre Chen ve arkadaşları parça tabanlı bir algoritma ve çok katmanlı perceptron ağı kullanarak büyük sinir ağlarının büyük sayıda veri kümesi kullanılarak model tabanlı yöntemlerin çok daha üstünde başarılı bir sonuç vereceğini göstermişlerdir. Literatürde yapay sinir ağları kullanarak gürültü azaltma popülerite kazanmıştır ve yüksek öğrenme becerileri nedeniyle özellikle konvolüsyonel sinir ağları başta olmak üzere derin öğrenme yöntemleri de çokça tercih edilmektedir.

2.1. Derin Öğrenme ile Gürültü Azaltma

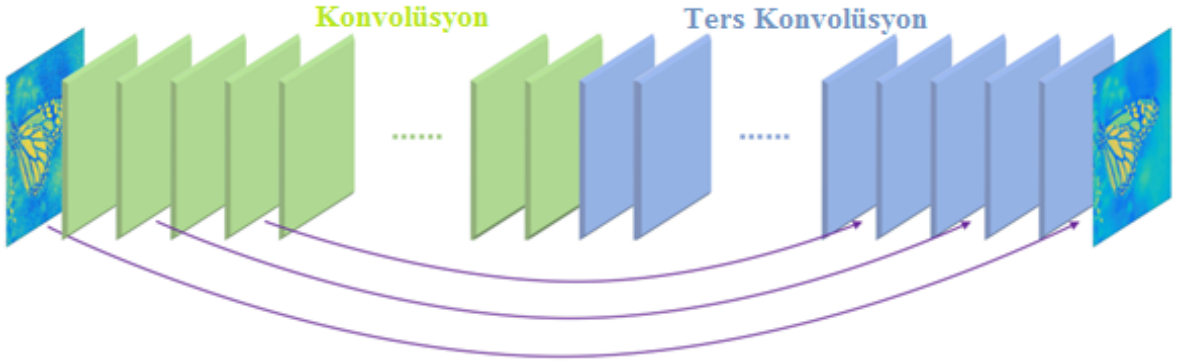
Zhang ve arkadaşlarının (Zhang vd., 2017a) çalışmasında Gauss gürültüsünü elemine etmek için önerdikleri 17 katmanlı ağ (DnCNN) literatürde önerilmiş diğer yöntemlerden çok daha başarılı sonuçlar elde etmesi ile derin öğrenme ile gürültü azaltma üzerine olan ilgiyi arttırmış ve bu alandaki çalışmalar hız kazanmıştır. Bu çalışmada Zhang ve arkadaşları gürültülü imgeden gürültüsüz imgeyi öğrenmek yerine, gürültülü imgede yer alan kalıntı imgesini yani gürültüyü öğrenmeyi hedeflemişlerdir. Son aşamada ise gürültülü imgeden bu gürültünün çıkarılması işlemini gerçekleştirerek gürültüsüz görüntüyü elde etmişlerdir. Her bir gürültü seviyesi için DnCNN ağı ayrı ayrı eğitilmiştir. Ağ yapısı içerisinde genel olarak konvolüsyon (Conv), yığın normalizasyonu (BN) ve ReLU aktivasyon fonksiyonu işlemleri yer almaktadır. DnCNN modeli Şekil 2.2’de gösterilmiştir.



Şekil 2.2. DnCNN modeli

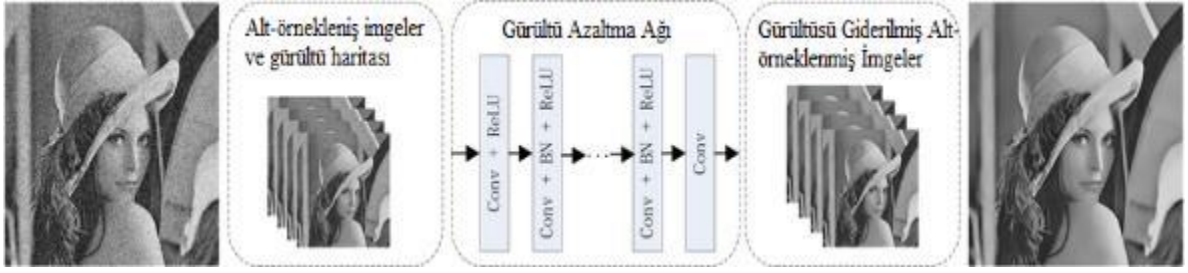
Mao ve arkadaşları çalışmalarında RED30 isimli imge düzenleme için çok derin bir kodlayıcı-çözücü (encoder-decoder) ağ yapısı önermişlerdir (Mao vd., 2016). Önerilen ağ yapısı içerisinde bir dize konvolüsyon katmanı ve bir dize ters konvolüsyon katmanı

kullanılmıştır. Ağ yapısının derinliğinin fazla olması ve eğitim aşamasında yakınsamanın uzun sürmesinden dolayı bunun önüne geçmek için atlama bağlantıyı simetrik olarak (symmetric skip connection) kullanılmıştır ve bu sayede eğitim aşamasında hızlı ve daha iyi bir yakınsama elde edilmiştir. Ayrıca atlama bağlantı kullanmanın ve bunun simetrik olarak tercih edilmesi üzerine ablasyon çalışmaları da eklemiştir. Şekil 2.3'te RED30 ağı gösterilmiştir.



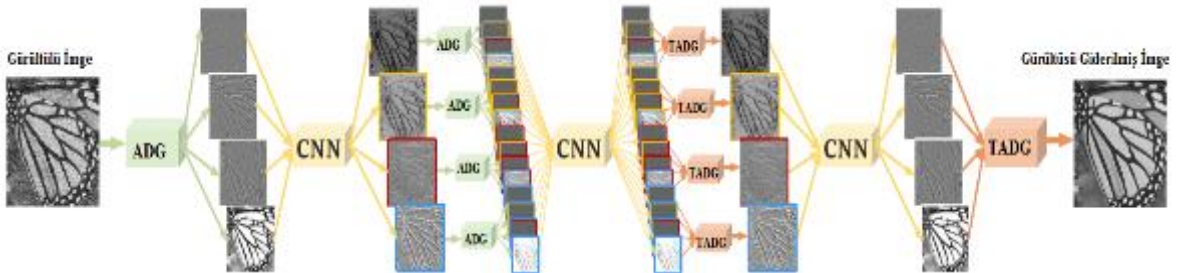
Şekil 2.3. RED30 ağı

Zhang ve arkadaşları gürültü azaltma işlemi için hızlı ve esnek bir ağ yapısı önermişler (Zhang vd., 2018). Ağ yapısı içerisinde ilk olarak gürültülü görüntü alt-örnekleme ile 4 alt imgeye dönüştürülmüştür. Daha sonra gürültü seviye haritası ile birleştirilerek (noise level map) 15 katmanlı CNN ağına girdi olarak verilmiş ve ağın en son aşamasında ise üst örnekleme ile orijinal görüntü elde edilmiştir. Alt-örnekleme işlemi CNN ağının hızını büyük oranda artırırken gürültü seviye haritası önerilen FFDNet isimli ağa büyük esneklik kazandırmıştır. Önerilen çalışma hızlı ve esnek bir yapıya sahip olmasının yanı sıra tek bir defa eğitilmiş CNN ağı ile farklı seviyelerdeki gürültüleri elemine edilebilmiştir ve tek düze olmayan gürültülerde yani uzamsal olarak değişen gürültülerde de eliminasyon işlemi yapabilmekte ve de başarılı sonuçlar üretmektedir. FFDNet mimarisinin gösterimi Şekil 2.4'te yer almaktadır.



Şekil 2.4. FFDNet ağı

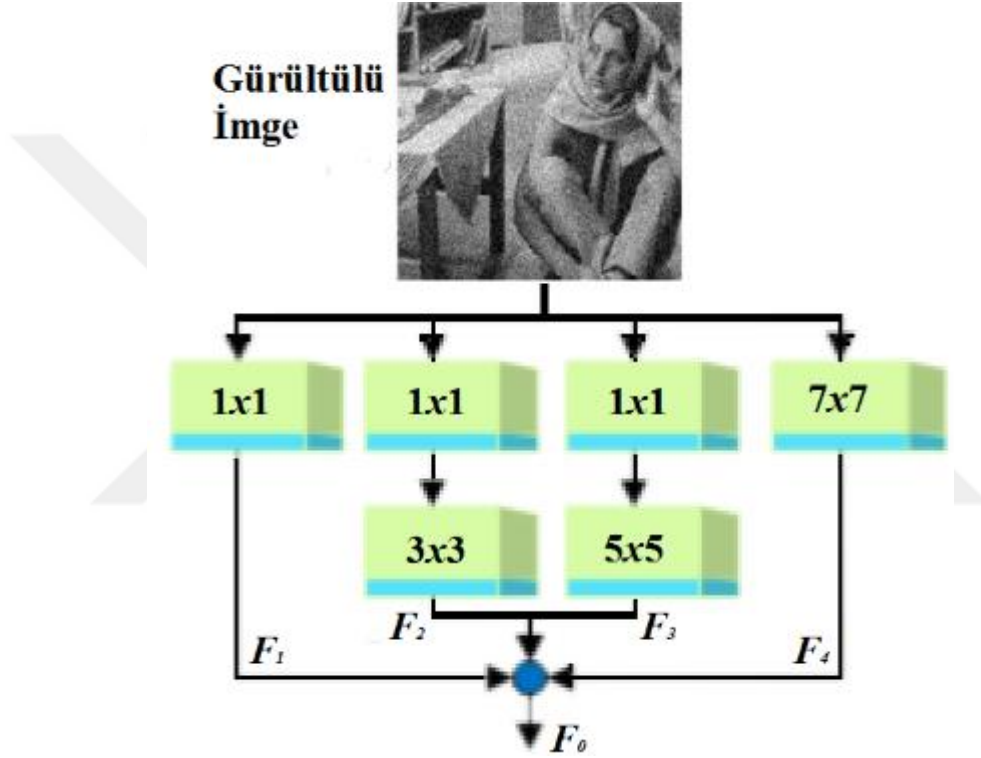
Liu ve arkadaşları çalışmalarında gürültü azaltma için önerdikleri CNN ağ yapısının içerisinde konvolüsyon katmanının yanı sıra dalgacık dönüşümü de kullanmışlardır (Liu vd., 2018). İlk olarak gürültülü görüntü ayrık dalgacık dönüşümü (ADG) kullanılarak 4 alt bant imge elde edilerek bu çıktı bir dizi konvolüsyon katmanına tabi tutulmaktadır. Daha sonra elde edilen 4 imge çıktısı tekrar ADG işleme tabi tutularak 16 alt imge elde edilmiş ve yine bu imgeler bir dizi konvolüsyon katmanına tabi tutularak işlem aynı şekilde devam ettirilmiştir. Önerilen ağın yarısına gelindiğinde ise ADG işlemi yerine ters ADG(TADG) kullanılarak alt bant olan imgeler birleştirilerek en son katmanda sonuç imgesi elde edilmiştir. Önerilen MWCNN isimli ağ yapısının diğer çalışmalardan temelde farklı olması nedeniyle literatüre katkısı büyüktür. MWCNN ağı Şekil 2.5'te gösterilmiştir.



Şekil 2.5. MWCNN ağı

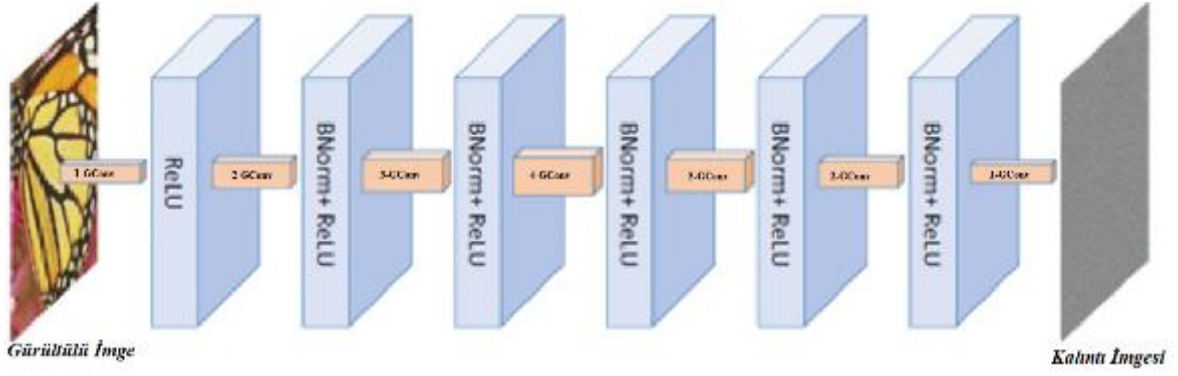
Guo ve arkadaşları çalışmalarında önerdikleri 24 katmanlı ağ yapısı üzerinde öncül özelliklerin çıkarımı için imge üzerinden farklı ölçeklerde özellik çıkarımı yöntemi kullanmışlardır (Guo vd., 2020). Bunun için ilk olarak girdi imgesi üzerinde 4 farklı seviyede konvolüsyon filtresi ile özellik çıkarılmış ve bunlar sıralı şekilde birleştirilerek ağa girdi olarak verilmiştir. Bu işlem sayesinde girdi imgesinde ilk aşaması yerel ve global özellikler

elde edilmiş ve ağ girdi çıktısı arasından gürültü haritasının çıkarılması için farklı ölçeklerdeki özneliklere sahip olunmuştur. Girdi özneliklerinin elde edilmesi için kullanılan konvolüsyon bloğu Şekil 2.6'da gösterilmiştir. Daha sonra farklı ölçeklerdeki bu öznelikler 24 katmanlı CNN ağına girdi olarak verilmiş ve ayrıca ağ içerisinde atlamalı bağlantılar kullanılarak ilk katmanlarda kaybolması muhtemel özneliklerin ileriki katmanlara aktarılması sağlanmıştır. DnCNN ağına benzer şekilde, önerilen McCNN ağına da kalıntı imgesinin öğrenimi yapılarak gürültü azaltma işlemi gerçekleştirilmiştir.



Şekil 2.6. McCNN ağı için önerilen çok seviyeli öznelik çıkarım bloğu

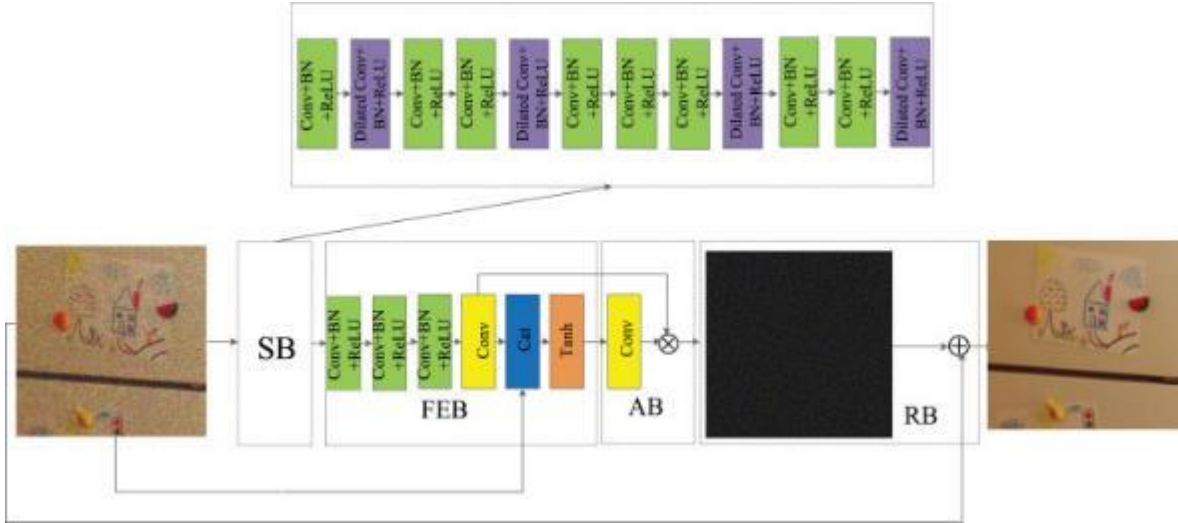
Zhang ve arkadaşları model tabanlı ve öğrenme tabanlı yöntemleri yarım kuadratik bölme (HQS) ve CNN kullanarak birleştiren görüntü restorasyonu için yeni bir ağ (IRCNN) önermişlerdir (Zhang vd., 2017b). Çalışmalarında CNN ağının kendi içinde gürültüye tolere yapısından faydalanmış ve bu sayede etkili bir gürültü giderme yöntemi önerilmiştir. IRCNN ağı genel olarak genişletilmiş konvolüsyon (Gconv) ve yığın normalizasyonu (BNorm) ve ReLU aktivasyon fonksiyonundan oluşmakta olup, Şekil 2.7'de ağın genel yapısı verilmiştir.



Şekil 2.7. IRCNN ağı

Tian ve arkadaşları çalışmalarında gürültü azaltma işlemi için ikili bir CNN (DudeNet) ağı önermişler (Tian vd., 2021). İkili olarak tasarlanan 17 katmanlı ağ yapısı içerisinde yığın normalizasyonu, konvolüsyon ve parametre sayısını arttırmadan alıcı alanı arttırmaya yarayan genişletilmiş konvolüsyon kullanılmıştır. İlk dalda 17 katmanlı yapı içerisinde konvolüsyon, genişletilmiş konvolüsyon, yığın normalizasyon ve ReLU kullanılarak bir yapı tasarlanarak gürültülü imgeden yerel ve global öznitelikler çıkarılmış, diğer dalda ise konvolüsyon ve ReLU kullanılarak daha çok yerel özniteliklere odaklanılmıştır. Bu iki daldan elde edilen öznitelikler art arda ekleme, yığın normalizasyon ve ReLU uygulanarak birleştirilmiştir. Ayrıca sıkıştırma bloğu kullanılarak, yani 1×1 konvolüsyon, elde edilen öznitelikler içerisinde bulunması muhtemel olan gürültüye ait özniteliklerin bastırılması işlemi yapılmıştır.

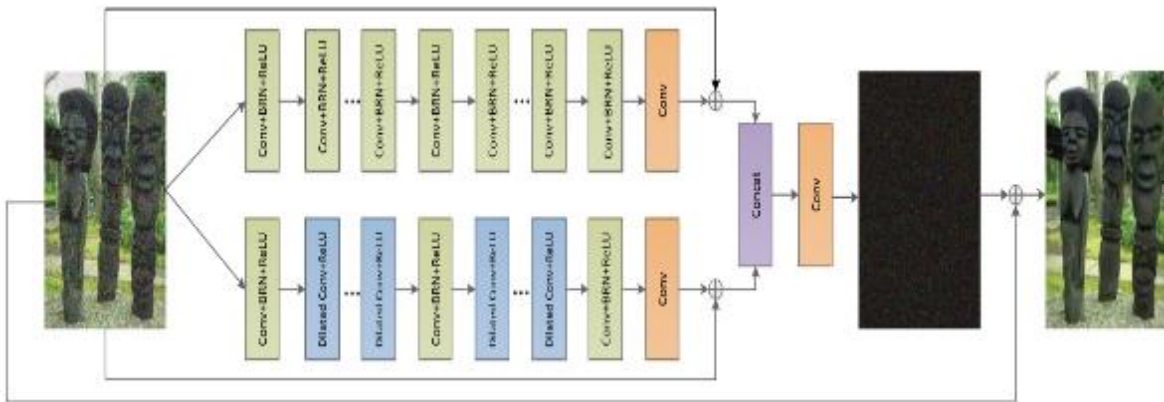
Tian ve arkadaşları çalışmalarında kompleks arka planlarda saklı gürültü bilgisini çıkarmak için basit bir odak mekanizması eklenmiş bir konvolüsyonel ağ tasarlayarak ADNet ağını tanıttılar (Tian vd., 2020a). Ayrıca alıcı alanı arttırmak ve geniş çekirdekli konvolüsyonlardan oluşan hesaplama maliyetini de düşürmek amacıyla genişletilmiş konvolüsyon (Dilated Conv) kullanmışlardır. ADNet ağı Şekil 2.8’de gösterilmiştir.



Şekil 2.8. ADNet ağı

Şekil 2.8’de “Cat” art arda ekleyerek birleştirme işlemini, “Tanh” tanjant aktivasyon fonksiyonunu, “x” işareti elementsel çarpım işlemini ve “+” işareti de elementsel toplama işlemini ifade etmektedir.

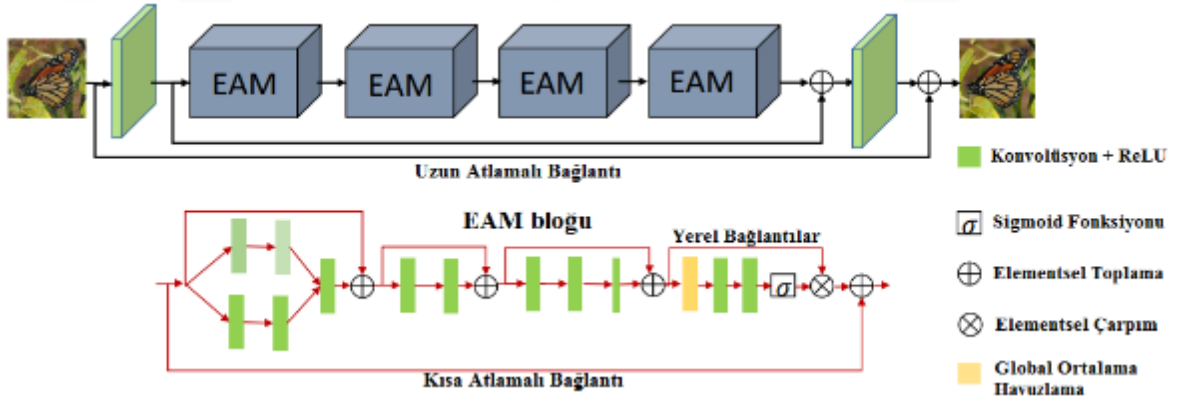
Tian ve arkadaşları çalışmalarında iki alt konvolüsyonel ağdan oluşan yani iki dallı bir CNN ağı olan BRDNet ağını önerdiler (Tian vd., 2020b). Önceki çalışmalardan farklı olarak küçük yığın probleminin üstesinden gelmek için geleneksel yığın normalizasyonu yerine yığın yeniden normalizasyonunu (Batch renormalization) yöntemini kullanmaları olmuştur. Gürültü azaltma alanında önceki çalışmaları ADNet ve DudeNet ağına kıyasla gri seviye ve renkli imgeler için daha iyi sonuçlar vermektedir. Şekil 2.9’da BRDNet ağı gösterilmiştir.



Şekil 2.9. BRDNet ağı

Şekil 2.9’da yer alan BRN, yığın yeniden normalizasyon işlemi ve “concat” ise art arda ekleyerek birleştirme işlemi ifade etmektedir.

Anwar ve Barnes gürültü azaltma alanında öznitelik odak mekanizmasını kullanan ilk CNN ağı (RIDNet) modelini önerdiler (Anwar & Barnes, 2019). Daha derin bir sinir ağı inşa edebilmek için uzun ve kısa atlamalı bağlantılar kullanarak kalıntı yapısı üzerine kalıntı öğrenme tabanlı bir tasarım kullandılar. Odak mekanizması olarak ise sıkıştırma ve uyarma (squeeze-and-excitation networks) ağı ismiyle önerilen ve kanal tabanlı bir odak mekanizması olan SENet (Hu vd., 2018) öznitelik odak mekanizmasından faydalandılar. RIDNet ağının Şekil 2.10’da gösterilmiştir.



Şekil 2.10. RIDNet ağı

Tian ve arkadaşları yaptıkları çalışmada karmaşıklığı az olan odak mekanizması ve CNN tabanlı yeni bir ağ önermişlerdir (Tian vd. 2020). Diğer çalışmalardan farklı olarak kanal tabanlı odak mekanizmasını kanallar arasındaki genel bilgiyi yakalarken piksel tabanlı odak mekanizması da kullanarak uzamsal tabanlı bilgileri elde etmeyi amaçlamışlardır. Bu sayede iki farklı odak mekanizmasını birleştirerek etkili ve yeni bir öznitelik odak mekanizması elde etmişlerdir. Bu odak mekanizmasını CNN ağının içerisinde kullanmak yerine atlamalı bağlantı aşamasında ileri katmanlara iletilecek olan öznitelikler üzerine uygulamışlardır. Bu sayede sık katmanlardaki öznitelikler derin katmanlara taşınırken odak mekanizması aracılığıyla gürültülü bilgileri elemeyi amaçlamışlardır. Küçük bir farklılık olarak ise ReLU aktivasyon fonksiyonu yerine parametrik ReLU aktivasyon fonksiyonunu tercih etmişlerdir.

Chupraphawan and Ratanamahatana kenar bilgisi ile CNN ağını birleştirerek gürültü azaltma için yapılan ilk çalışmayı önerdiler (Chupraphawan & Ratanamahatana, 2019). Burada kenar çıkarma işlemi için basit bir kenar tespit yöntemi olan Canny algoritmasından faydalandılar. İlk olarak gürültülü görüntü üzerine Canny uygulanarak kenar bilgisi çıkarılmakta ve bir öncül (rehber) bilgi olarak gürültülü görüntüyle beraber gürültü azaltma ağına girdi olarak verilmektedir. Çok etkili bir yöntem olmasa da ilk çalışma olması açısından literatüre katkısı büyüktür.

Fang ve arkadaşları ise gürültülü imgeden elde edilen kenar öncül bilgisini kullanan yeni bir CNN ağı tanıttılar (Fang vd., 2020). Diğer çalışmalardan farklı olarak gürültülü imgelerden kenar bilgisini çıkarmada çok da etkili olmayan kenar tespit yöntemleri yerine, aynı işlemi yapması için ayrıca bir CNN ağı tasarladılar. Buna ek olarak kenar çıkarma ağı ve gürültü azaltma ağı aynı anda eğitilmiştir. Her iki ağ için ayrı bir kayıp fonksiyonu tanımlanarak, eğitim aşamasında geri yayınımlı sağlamak adına bu kayıp fonksiyonları ağırlıklandırılmış şekilde birleştirilerek toplam kayıp fonksiyonu elde edilmiş ve eğitim işlemi bunun üzerine gerçekleştirilmiştir.

Liu ve arkadaşları kenar bilgisinin dışında gürültü azaltma işlemi sırasında bir öncül bilgi olması ve bu sayede gürültüsü indirgenmiş imge üzerinde kenar ve doku bilgilerini koruyabilmek için gradyan bilgisini ek bir girdi olarak kullanan bir ağ önermişlerdir (Liu vd., 2020). İlk olarak literatürde ilk çalışmalardan olarak görülen DnCNN ağı kullanılarak gürültü imge üzerinden kısmi gürültüsü giderilmiş görüntü elde edilmiş ve Sobel maskesi kullanılarak gradyan bilgileri elde edilmiştir. Farklı olarak ise sadece bu gradyan imgesi gürültü azaltma ağına sadece girdi olarak verilmemiş, atlamalı bağlantılar kullanılarak CNN ağının farklı katmanlarına da bu gradyan bilgisi ek bir girdi olarak eklenmiştir. Buna gerekçe olarak ise sığ katmanlarda yer alan filtrelerin düşük seviye görsel uyarana karşı duyarlı olduğu belirtilmiştir. Gürültüsü azaltılmış imgede gradyan bilgisini korumak için gürültüsü azaltılmış imge ile orijinal imge arasındaki gradyan farkı L_1 norm kayıp fonksiyonu kullanılarak hesaplanmış ve bu kayıp değeri L_2 norm kayıp fonksiyonu ile hesaplanan gürültüsü azaltılmış imge ile orijinal imge arasındaki fark ile birleştirilerek toplam bir kayıp fonksiyonu tanımlanmış ve eğitim işlemi buna göre yapılmıştır.

CNN ağlarında hesaplama maliyeti ve hız açısından daha hafif ağların kullanılması çokça düşünülmektedir ancak bu yaklaşımın en büyük dezavantajı derin ve genişlik açısından sınırlı aralığa sahip olduklarından performanslarının istenilen seviye ulaşmamasıdır. Bunu gidermek için Tian ve arkadaşları çalışmalarında dinamik

konvolüsyon (Chen vd., 2020) tabanlı yeni bir CNN ağı önerdiler (Tian vd., 2023). Bu ağ içerisinde konvolüsyon işleminin yanı sıra bir sinyal işleme yöntemi olan dalgacık dönüşümünü iki aşamalı kullanarak ağlarına entegre etmişler ve yoğun kalıntı tabanlı bir CNN ağı tasarımı önermişlerdir.

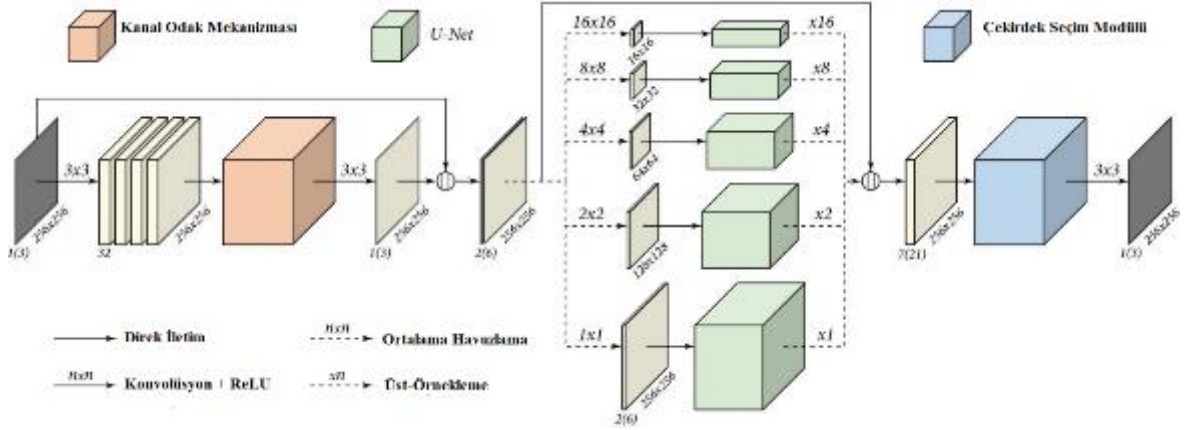
Zhang ve arkadaşları çalışmalarında tek bir girdi verisine değil birden fazla girdi verisine sahip bir ağ yapısı önermişlerdir (Zhang vd., 2022). Burada tek bir gürültülü imgeyi giriş olarak vermek yerine konvolüsyon katmanları ile çıkarılacak olan özniteliklerin zenginliğini arttırmak için gürültü imgeye ek olarak aynı imgenin 90° ve 180° derece rotasyon hallerini de CNN ağına girdi olarak vermişlerdir. Bu 3 farklı imge (gürültü imgenin $0^\circ(x)$, $90^\circ(x_1)$ ve $180^\circ(x_2)$ derece rotasyon hali) tasarımı aynı olan yoğun mimariye sahip bir dizi konvolüsyon katmanına tabi tutularak 3 farklı seviyede öznitelik çıkarımı yapılmaktadır. Daha sonra bu 3 rotasyonlu görüntüden çıkarılan özniteliklerin birleşimi için çok yollu yeni bir odak mekanizması (MPA) önerilmişlerdir. Kanal ya da piksel seviyesinde öznitelik ilişkisine bakan geleneksel odak mekanizmalarından farklı olarak görüntü seviyesinde özniteliklere odaklanılmış ve bunun için bu 3 farklı öznitelikler farklı şekillerde ($x_1 * x_2$, $x * x_1$ şeklinde) çarpma ve konvolüsyon işlemleriyle birleştirilmiştir. MPA genel olarak içerisinde örnek normalizasyon (instance normalization) ve 1×1 konvolüsyon işlemleri içermektedir. Daha sonra x için elde edilen öznitelikler 128 adet 1×1 konvolüsyon işlemine tabi tutularak MPA çıktısı ile aynı boyuta getirilmiş ve bir çıkarma işlemine tabi tutularak gürültü haritaları elde edilmiştir. Bu gürültü haritalarında kanallar arasında ilişkinin çıkarılması için literatürde çokça kullanılan ECANet (Wang vd., 2020) kanal odak mekanizmasına kullanılmış ve çıkarılan bu ilişkilerin ileri seviye işlenmesi için yoğun atlamalı bağlantıya sahip 4 adet konvolüsyon işlemine tabi tutularak nihai gürültü haritası çıkarılmıştır. Son olarak ise gürültüsüz imgenin elde edilmesi için girdi imgesi ile bir çıkarma işlemine tabi tutulmuştur.

Wu ve arkadaşları çalışmalarında gürültü seviyesinin bilinmediği kör gürültü azaltma durumu için bir CNN ağı önermişlerdir (Wu vd., 2023). İkili ağ yapısı ile odak mekanizmasını birleştiren ilk yapı olması ile ön plana çıkmaktadır. İlgili çalışmada 3 yapıllı bir yöntem önerilmiştir. Bunlardan ilki gürültülü imgeyi girdi olarak alan ve gürültü seviyesinin kestirimini yapan 7 katmanlı bir alt CNN ağıdır. Burdan elde edilen gürültülü seviyesi bilgisi ve gürültülü imge ikili yapıda tasarlanan gürültü azaltma ağına verilmektedir. Bu ağda ilk dalda U-Net yapısına benzer olarak üst ve alt örnekleme işlemleri yer alan bir ağ varken, diğer dalda yoğun atlamalı bağlantı içeren sıralı konvolüsyon katmanlarından

oluşan bir ağ yer almaktadır. Odak mekanizması olarak ise kanal ve uzamsal odak mekanizmaları paralel olarak uygulanmakta ve elde edilen çıktı matrisi elementsel olarak toplanmaktadır.

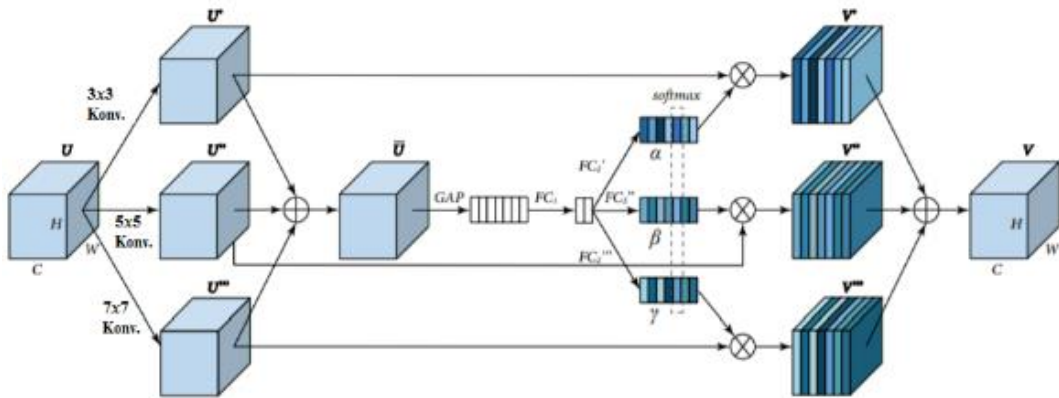
Thakur ve Maji çalışmalarında çok ölçekli öznitelik çıkarımının yanı sıra çok ölçekli odak mekanizmasını da kullanan yeni bir CNN ağı önermişlerdir (Thakur & Maji, 2023). Bu çalışmada ayrıca gürültülü imgenin yanı sıra gürültü imgeden elde edilen negatif imge de girdi verisi olarak kullanılmıştır. Ağ tasarımı ikili yapıda sunulurken, bir dalda negatif imgeyi girdi olarak kullanan çok ölçekli odak ağı, diğer dalda ise gürültülü imgeyi girdi olarak kullanan çok seviyeli öznitelik çıkarımı ağı yer almaktadır. Hesaplama maliyetinin azaltması için genişlemiş konvolüsyonlar kullanılmıştır.

Zhao ve arkadaşları çalışmalarında gerçek gürültünün azaltılması için piramit yapılı bir CNN ağı (PRIDNet) önermişlerdir (Zhao vd., 2019). Diğer gürültü azaltma tabanlı CNN ağlarından farklı olarak parça tabanlı bir öğrenme yerine gürültülü imge 256x256 boyutuna indirgenerek CNN ağına girdi olarak verilmiştir. Bu gürültülü imge piramit yapılı modüle girdi olarak verilmeden önce etkili bir öncül öznitelik çıkarımı için sırasıyla 4 adet 32x32x3 şekilde konvolüsyon işlemine tabi tutulmuş ve sonrasında kanal odak mekanizması uygulanmıştır. Daha sonra öznitelik haritası piramit yapısına her seviyede ayrı ayrı girdi olarak verilmiştir. Piramit yapısının oluşması için her seviyede sırasıyla 1x1, 2x2, 4x4, 8x8, 16x16 boyutunda ortalama havuzlama işlemine tabi tutularak yine sırasıyla 256x256, 128x128, 64x64, 32x32, 16x16 boyutunda imgeler elde edilmiş ve ayrı ayrı U-Net ağına verilmiştir. (U-Net ağı içinde bütün imgeler tekrar 256x256 boyutuna getirilmiştir). Son olarak ise çıktı imgeleri öncül öznitelikler ile de birleştirilerek çekirdek seçim modülü (ÇSM) olarak isimlendirilen yeni bir ağa verilmiştir. En son olarak ise 3x3 bir konvolüsyon işlemi kullanılarak girdi görüntüsüne ait gürültü haritası elde edilmiştir. Önerilen PRIDNet ağı Şekil 2.11'de gösterilmiştir.



Şekil 2.11. PRIDNet ağı

PRIDNet ağı içerisinde kullanılan ÇSM modülü Şekil 2.12’de gösterilmiştir. ÇSM modülü tasarımındaki amaç birleştirilen özneliklerin hangi seviyede birleştirileceğini karar vermek için ağırlıklandırma yapmaktır.



Şekil 2.12. Çekirdek seçim modülü (ÇSM)

Burada “GAP” global ortalama havuzlama, “FC” tam bağlantılı katmanı ifade etmektedir.

Odak mekanizması ve geleneksel CNN mimarilerinin yanı sıra son yıllarda transformatör(transformer) tabanlı derin öğrenme yöntemleri de gürültü azaltma alanında büyük popülerlik kazanmıştır. Bunun en büyük nedeni büyük model kapasiteleri ve uzunlu menzilli bağımlılıkları yakalayabilme becerileri sayesinde yüksek başarıma sahip sonuçlar vermeleridir. Kişisel odak (self-attention) bu ağların tasarımında kullanılan temel

unsurlardan bir tanesidir. En büyük dezavantajları ise konvolüsyonel sinir ağlarına kıyasla yüksek hesaplama maliyetlerine sahip olmalarıdır. Liang ve arkadaşları bu sorunu aşmak için Swin transformatörleri (Liu vd., 2021) kullanarak bir ağ önermişlerdir (Liang vd., 2021). Swin transformatörler genel olarak imgenin tamamı üzerinde odak mekanizması kullanmak yerine 8×8 boyutunda kaydırmalı küçük pencereler kullanarak odak mekanizmasını kullanmakta ve hesaplama maliyetini büyük oranda azaltmaktadır.

Zamir ve arkadaşları, görüntü restorasyonu için etkili bir transformatör modeli önerdiler (Zamir vd., 2022). Geleneksel dönüştürücü bloğunu, yani çok başlı odak ve ileri besleme ağı bloğunu, tasarlarken çeşitli değişiklikler yaptılar. İlk olarak, çok başlı dikkat mekanizmasını (multi-head attention) doğrusal karmaşıklığa sahip çoklu Dconv aktarılmış odak (MDTA) ile değiştirdiler. MDTA, 1×1 konvolüsyon kullanarak yerel bağlam karışımı yapmaktadır. Bu, uzamsal olarak yerel bağlamın vurgulanması, konvolüsyonun gücü ve pikseller ile model arasındaki bağlamsallaştırılmış global ilişkiler gibi avantajlar sağlamaktadır. Ayrıca tamamen bağlantılı iki katmandan oluşan ve aralarında doğrusal olmayan bir durum bulunan temel ileri beslemeli ağı (FN) değiştirdiler. FN'nin ilk doğrusal katmanında bilgi akışını iyileştirmek için bir geçit mekanizması kullandılar (Dauphin vd., 2017). Buna ek olarak ise eğitim stratejisini de değiştirmişlerdir. Ağlarını ilk epoklarda küçük parça (patch) ve büyük yığınlar, daha sonraki epoklarda ise büyük görüntü parçaları ve küçük yığınlar üzerinde eğitmişlerdir. Bu değişim kademeli olarak gerçekleştirilmiş ve buna aşamalı öğrenme adını vermişlerdir.

Literatüre bakıldığında derin öğrenme ile gürültü azaltma işlemi son yılların en popüler konusu haline geldiği görülmektedir. Buna ek olarak görsel odak mekanizmalarının ve özellikle görsel dönüştürücülerin eklenmesi ile derin öğrenme tabanlı yöntemlerle geleneksel yöntemlere oranla çok daha başarılı sonuçlar elde edilmiştir. Derin öğrenme ağları ile ilgili en büyük sorun eğitim aşamalarının uzun sürmesi olarak düşünülebilir. Ancak grafik kart tasarımlarının her yıl gelişmesi ve eğitim aşamasının sadece bir kez gerçekleştirilmesi (eğitilen ağın bir görüntü için gürültüyü azaltması çok hızlıdır) bu sorunları önemsiz kılmaktadır. Buna ek olarak derin öğrenme ile eğitim yüksek sayıda veriye ihtiyaç duymaktadır. Bu sorun da veri arttırma ve gürültü azaltma yöntemlerinde görüntünün kendisi yerine parça görüntülerin çıkarılarak eğitim yapılması ile aşılabilmektedir. Ayrıca günümüzde görüntünün kullanımının çok artması da veri setlerinin oluşturulmasını bir hayli kolaylaştırmaktadır.

3. YAPILAN ÇALIŞMA

3.1. Motivasyon

Literatüre bakıldığında gürültü azaltma performansını arttırmak için gradyan ve kenar bilgisini öncü alan çalışmalar bulunmasına rağmen doku bilgisini öncül bilgi olarak kullanan bir çalışmaya rastlanılmamıştır. Bu çalışmada öncü bilgisi olarak doku bilgisi kullanılmış ve doku çıkarma işlemi için hesaplama maliyetinin düşük olması ve kolay bir yöntem olmasından dolayı Yerel İkili Örüntüler (YİÖ) seçilmiştir (Ojala vd., 1996). Bu amaçla gürültülü imgelerden YİÖ doku bilgisinin çıkarılması hedeflenmiş ve bunun için iki dallı bir yapı kullanılarak TENet isimli yeni bir CNN ağı tasarlanmıştır.

Buna ek olarak literatürde çoklu seviye öznitelik çıkarma bloğu kullanan yöntemler bulunmasına rağmen, bu öznitelik çıkarma işlemi çoğunlukla sadece girdi imgesi üzerine uygulanmış olup, ağın derin yapılarında kullanımı göz ardı edilmiştir. Aynı zamanda yerel ve global öznitelikler birleştirilip çıkarılan öznitelikler ağın ileriki katmanlarına iletilirken, bütün özniteliklere aynı önem değeri verilmiştir ancak bu yaklaşım doğru değildir. Çünkü her özniteliğin çıkışa etkisi farklı olacaktır. Buna bir örnek olması açısından ve motivasyon kaynağı olarak Barbara görüntüsünün gürültüsüz ve $\sigma = 10$ ve $m = 0$ olmak üzere Gauss gürültüsü bulaşmış hali ve bu imgelere ait mavi kutucuk ile gösterilen bölgelerden elde edilen imge parçaları ve piksel değerleri Şekil 3.1’de gösterilmiştir.



(a) Barbara imgesi



(b) Gauss gürültülü Barbara imgesi

72	120	132	141	155	157	159	111	37
87	115	135	151	150	155	149	91	29
100	89	139	159	152	156	152	96	30
117	97	149	154	152	159	150	95	32
123	108	151	149	154	153	146	105	33
129	121	151	151	155	151	152	120	42
112	135	150	150	158	152	155	121	42

(c) Gürültüsüz imge parçası

86	124	138	136	162	157	147	107	43
68	115	134	144	170	145	159	99	26
112	103	138	153	153	152	153	100	22
108	94	149	148	160	154	136	98	35
119	112	152	150	148	159	132	99	37
135	118	150	145	154	155	103	100	40
112	133	149	144	159	147	146	114	62

(d) Gürültülü imge parçası

Şekil 3.1. Gürültülü ve gürültüsüz Barbara imgesi ve piksel değerleri ile gösterilen imge parçaları

Gauss gürültüsünü elemine etmenin en basit yolu bilindiği üzere ortalama filtre kullanmaktır. Burada Şekil 3.1- (d)'de sarı kutucuk ile gösterilen pikselin gürültüsüz asıl değerini, yani Şekil 3.1- (c)'de yer alan pembe kutucuk ile gösterilen 152 değerini, komşu piksellerin yardımıyla bulmaya çalışalım. En yerel komşu pikseller olan yeşil kutucukların ortalamasını alırsak 157,57 değerini, biraz daha global değerlere gidip mavi kutucukları da ortalamaya dahil edersek 152,4 değerini ve son olarak turuncu kutucukları dahil edersek de 151,89 değerini bulmuş oluruz. Sonuçlardan da anlaşılacağı üzere yerelden globale gidilerek ortalama değerler alındıkça asıl değer olan 152 piksel değerine daha çok yaklaşılmaktadır. Ancak turuncu kutucuklar ile aynı seviyede globalliğe sahip olan kırmızı kutucukları ortalamaya dahil ettiğimizde ise ortalama değer 140 olmakta ve 152 değerinden başlangıca göre çokça uzaklaşmaktadır. Buradan anlaşılacağı üzere yerel özellikler kadar global özellikler de kestirime katkı sağlamaktadır ancak küresel özelliklerin tamamının aynı oranda sonuca etkisi sonucu bir yerden sonra elde edilen sonuç kötüleşmektedir. Bu motivasyon

ile gürültü azaltma işlemi için tasarlanan çok seviyeli öznitelik çıkarım bloğu ile çıkarılan öznitelikler ileriki katmanlara dahil edilirken kanal tabanlı bir odak mekanizması kullanılarak ağırlıklandırma yapılmıştır. Kısaca yerel ve global öznitelikler arasında bir ağırlıklandırma ya da başka bir deyişle seçim işlemi yapılmıştır. Özetle, gürültülü imgeden gürültünün elemine edilmesi için çok seviyeli öznitelik bloğu (Multilevel Feature Block-MLF) ve odak bloğundan (Attention Block - AB) oluşan MLFAN isimli yeni bir CNN ağı önerilmiştir. MLFAN ağına girdi verisi olarak gürültülü imgenin yanı sıra öncül bilgi olması ve gürültü azaltma işlemi sırasında görüntünün orijinal yapısından uzaklaşmasının önüne geçmek için TENet ağı kullanılarak elde edilen YİÖ tabanlı doku imgesi verilmiştir. Önerilen yöntem gri seviye ve renkli görüntülerde Gauss gürültüsünün elemine edilmesi işlemi için literatürde önerilen yöntemlerle kıyaslandığında hem niteliksel hem niceliksel açıdan daha iyi sonuç vermektedir.

3.2. Tezin Kapsamı

Bu çalışmada gri ve renkli görüntüler için CNN tabanlı yeni bir yöntem önerilmiştir. Bu kapsamda ilk olarak öncül bilgi olması için gürültülü imgelerden YİÖ tabanlı doku imgesini çıkaran iki dallı bir yapıda tasarlanan TENet ağı önerilmiştir. İkinci olarak ise gürültülü imgeyi ve YİÖ imgesini girdi olarak alan gürültü azaltma işlemi gerçekleştiren MLFAN isimli yeni bir CNN ağı önerilmiştir. MLFAN ağı genel olarak çok seviyeli öznitelik çıkarımı ve bu özniteliklerin ağırlıklandırılması için kanal tabanlı odak mekanizmasından oluşmaktadır. Ayrıca önerilen çalışmada sunulan yenilikçi fikirlerin başarımını ölçmek için kapsamlı bir ablasyon çalışması da sunulmuştur. Buna göre tezin kapsamı ve yapılan çalışmanın literatüre katkısı şu şekilde özetlenebilir:

- Gürültü azaltma sırasında detay bilgilerinin korunması ve imgenin özgün yapısından uzaklaşmasının önüne geçmek için gürültülü imgeden çıkarılan doku imgesi gürültü azaltma işlemine dahil edilmiştir.
- Doku imgesi olarak basit ve hesaplaması kolay olmasından dolayı YİÖ yöntemi seçilmiş ve gürültülü imgeden etkili bir doku imgesi çıkarımı için iki dallı yapı kullanılarak TENet isimli yeni bir CNN ağı önerilmiştir.
- Farklı seviyelerde (ölçeklerde) etkili bir öznitelik çıkarımı için yeni çok seviyeli öznitelik bloğu (MLF) önerilmiş ve yerel ve global özniteliklerin etkili şekilde birleştirilmesi(işlenmesi) için kanal tabanlı bir odak mekanizması entegre

edilmiştir. Ayrıca literatürdeki çalışmalarından birçoğundan farklı olarak MLF sadece girdi imgesine değil ağı derin katmanlarında da kullanılmıştır.

- Etkili bir gürültülü azaltma için odak mekanizması ve MLF bloklarından oluşan ve girdi olarak gürültülü imge ve YİÖ doku imgesini girdi olarak alan MLFAN isimli yeni bir CNN ağı önerilmiştir. PSNR ve SSIM metrikleri altında gri seviye ve renkli görüntüler üzerine yapılan deneyler önerilen ağın, literatürdeki benzer çalışmalara kıyasla başarılı sonuçlar verdiğini göstermektedir.
- Önerilen yöntemdeki yeniliklerin, CNN ağına katkısını değerlendirmek için kapsamlı bir ablasyon çalışması sunulmuştur. Bu kapsamda 5 farklı durum senaryosu çalışılmıştır. Bunlar:
 - o İlk olarak, kanal tabanlı odak mekanizmasının başarımını değerlendirmek için uzamsal odak mekanizmasını da kullanan yeni bir odak mekanizması tercih edilmiş mevcut odak mekanizmasının yerine entegre edilmiştir.
 - o İkinci olarak ise doku imgesinin etkisi incelenmiştir. Bu amaçla doku imgesi çıkarılmış ve MLFAN ağına girdi olarak sadece gürültülü imge verilmiştir.
 - o Üçüncü olarak MLF bloğunun etkisi incelenmiş ve bunun için MLF bloğu içerisinde farklı ölçeklerdeki konvolüsyon katmanları çıkarılarak tek bir seviyede konvolüsyon katmanı eklenmiş ve sonuçlar değerlendirilmiştir.
 - o Dördüncü olarak ise TENet ağının doku imgesinin çıkarımına ve gürültü azaltma işlemine katkısı değerlendirilmiş ve bu amaçla TENet ağı çıkarılmıştır. Gürültülü imgeden YİÖ imgesinin çıkarımı için gürültüye tolere bir YİÖ tabanlı doku çıkarımı yöntemi kullanılmış ve MLFAN ağına gürültülü imgenin yanı sıra YİÖ tabanlı doku imgesi olarak bu yöntemle çıkarılan imge verilmiştir.
 - o Son olarak ise atlamalı bağlantı kullanımını üzerine bir değerlendirme yapabilmek için MLFAN ağında kullanılan atlamalı bağlantı sayısı arttırılmıştır.

MLFAN ağı diğer parametreler sabit tutularak bu yeni odak mekanizması altında test edilmiştir.

3.3. Yapılan Çalışma

3.3.1. Doku Çıkarım Ağı

Yerel ikili örüntü yöntemi doku çıkarma işlemi için basit ve etkili bir yöntem olmasına rağmen piksel komşulukları arasındaki ilişkiyi baz alan bir yöntem olduğundan dolayı komşuluk ilişkilerinin bozulduğu özellikle gürültü seviyesi yüksek olan imgeler için etkili bir sonuç üretememektedir. Buna bir örnek Şekil 3.2’de sunulmuştur.



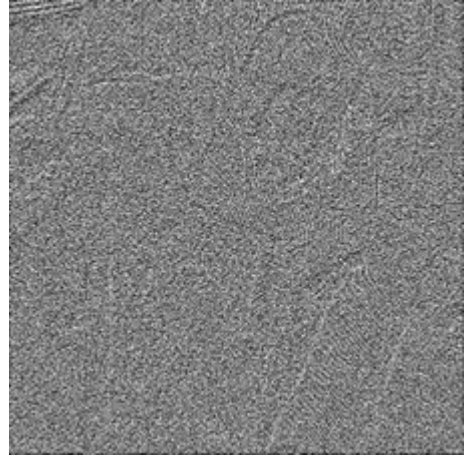
(a) Barbara imgesi



(b) Barbara için YİÖ imgesi (gürültüsüz)



(c) $\sigma=15$ için YİÖ imgesi

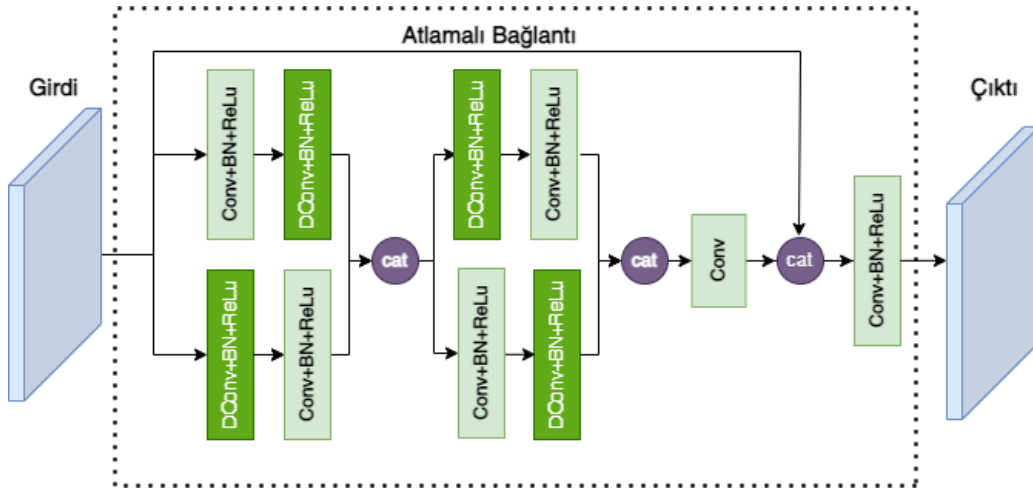


(d) $\sigma=50$ için YİÖ imgesi

Şekil 3.2. Gri seviye Barbara imgesi ve farklı gürültü seviyelerinde YİÖ imgeleri

Şekil 3.2’de Barbara imgesinin orijinal hali ile birlikte, gürültüsüz imgeden elde edilen YİÖ imgesi Şekil 3.2-(b)’de, $\sigma=15$ olmak üzere Gauss gürültüsü eklenmiş Barbara imgesi için YİÖ imgesi Şekil 3.2-(c)’de ve $\sigma=50$ olmak üzere Gauss gürültüsü eklenmiş Barbara imgesi için YİÖ imgesi Şekil 3.2-(d)’de verilmiştir. Şekilden de anlaşılacağı üzere Gauss gürültüsü eklendikçe çıkarılacak olan doku bilgisi bozulmakta ve özellikle $\sigma=50$ gibi yüksek seviye bir gürültü için ise doku bilgisi tamamıyla elemine olmaktadır. Bu nedenle (Fang vd., 2020) çalışmasından ilham alınarak gürültülü imgelerden YİÖ tabanlı doku imgesinin çıkarımı için CNN tabanlı Doku Çıkarım Ağı (Texture Extraction Network – TENet) tasarlanmıştır.

Farklı konvolüsyonel işlemlerin peş peşe yapılması farklı özniteliklerin elde edilmesine imkân sağlamaktadır. Bu nedenle doku çıkarımı için tasarlanan CNN ağı Şekil 3.3’te gösterildiği gibi iki dallı konvolüsyonel bloklardan (Dual Convolutional Block – DCB) oluşmaktadır. Büyük çekirdek boyutuna sahip konvolüsyon işlemleri global ilişkileri yakalamakta daha iyi oldukları için CNN ağları tasarımında sıklıkla tercih edilmektedir. Bu nedenle DCB bloklarının tasarımını yaparken 5×5 çekirdek boyutuna sahip konvolüsyonlardan faydalanılmıştır. Ancak hesaplama maliyetinin arttırdığı için klasik konvolüsyon işlemi yerine genişleme oranı 2 olan genişlemiş konvolüsyon tercih edilmiştir. Farklı çekirdek boyutuna sahip konvolüsyon işlemlerinin uygulanması ile farklı seviyelerdeki öznitelikler elde edilebilmiştir.



Şekil 3.3. İkili konvolüsyonel blok (DCB)

Şekil 3.3’ye daha detaylı bakacak olursak “DConv” genişlemiş konvolüsyon işlemini, “Conv” klasik konvolüsyon işlemini, “BN” yığın normalizasyonunu, “ReLU” ReLU aktivasyon fonksiyonunu ve son olarak “cat” ise art arda ekleme şeklinde yapılan birleştirme işlemini ifade etmektedir.

İki dallı yapıda farklı ölçeklerdeki konvolüsyon işlemi 2 defa uygulanarak farklı öznetelikler çıkarıldıktan sonra bu öznetelikler birleştirilmekte ve tekrar bir ikili konvolüsyon işlemi uygulanmıştır. Elde edilen nihai öznetelikler birleştirildikten sonra ortak özneteliklerin elde edilmesi için bir tane 3×3 çekirdek boyutuna sahip konvolüsyon işlemi uygulanmaktadır. Atlamalı bağlantı kullanılarak birçok imge detayı içeren önceki katmanlarda elde edilen öznetelikler çıkarılan bu ortak öznetelikler ile birleştirilmektedir. Tekrar ortak özneteliklerin çıkarılması için konvolüsyon, yığın normalizasyon ve ReLU aktivasyon fonksiyonu uygulanmıştır. İkili konvolüsyonel bloğun matematiksel gösterimi Eşitlik (3.1), (3.2) ve (3.3)’deki gibidir:

$$f_{out1} = C_{at} \left(C_{br} (D_{br} (f_{in})), D_{br} (C_{br} (f_{in})) \right) \quad (3.1)$$

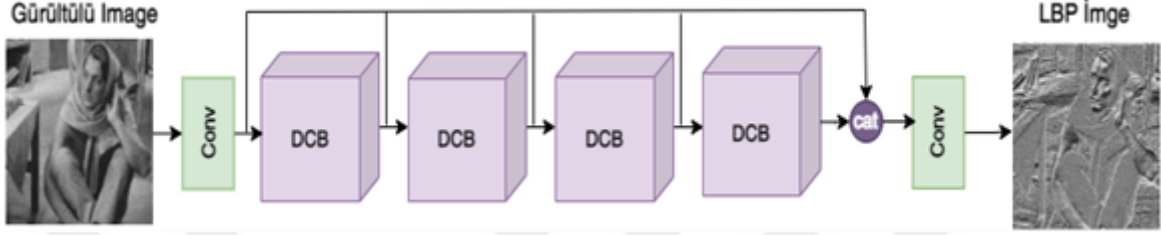
$$f_{out2} = C_3 \left(C_{at} \left(C_{br} (D_{br} (f_{out1})), D_{br} (C_{br} (f_{out1})) \right) \right) \quad (3.2)$$

$$f_{DCB} = C_{br} (C_{at} (f_{out2}, f_{in})) \quad (3.3)$$

burada C_{br} konvolüsyon, yığın normalizasyon ve ReLU işlemini, D_{br} genişlemiş konvolüsyon olan yapıyı, C_3 3×3 konvolüsyon işlemini, C_{at} art arda eklenerek yapılan birleştirme işlemini, f_{in} girdi verisini, f_{out1} ve f_{out2} ara çıktı olarak hesaplanan öznetelikleri ve son olarak f_{DCB} DCB bloğu uygulanarak elde edilen çıktı özneteliklerini ifade etmektedir. DCB bloğu altında uygulanan konvolüsyon işlemlerinin tamamında tercih edilen filtrelerin sayısı 64’tür.

Doku çıkarma ağının tasarımı için 4 adet DCB bloğu kullanılmıştır. İlk özneteliklerin çıkarımı için girdi imgesine 3×3 boyutunda ve 64 filtre sayısına sahip tek bir konvolüsyon işlemi uygulanmış ve bu öznetelikler DCB bloğuna girdi olarak verilmiştir. Yoğun atlamalı bağlantı yaklaşımından ilham alınarak DCB blokları yoğun atlamalı bağlantılar kullanılarak birleştirilmiştir (Huang vd., 2017). Yani bir DCB bloğunun çıktısı kendisinden sonra gelen bütün DCB bloklarına art arda ekleme işlemi ile birleştirilerek girdi olarak verilmiştir.

Sonuncu DCB bloktan sonra önceki bütün DCB bloklardan elde edilen öznetelikler ve çıkarılan ilk öznetelikler birleştirilmiş ve nihai YİÖ doku imgesinin elde edilmesi için 1 adet 3×3 boyutunda bir konvolüsyon işlemine uygulanmıştır. Doku çıkarma ağına ait gösterim Şekil 3.4’te yer almaktadır.



Şekil 3.4. Doku çıkarma ağı – TENet

Şekil 3.4’te yer alan DCB, ikili konvolüsyonel bloğu, “cat” birleştirme işlemi ve “conv” ise konvolüsyon işlemi ifade etmektedir.

TENet ağının eğitilmesi için gürültü azaltma ağı dışında farklı bir veri seti kullanılmıştır. Bu amaçla eğitim için 500 imgeden oluşan BSD500 (Martin vd., 2001) veri seti kullanılırken validasyon için 100 imgeden oluşan Urban100 veri seti (Huang vd., 2015) kullanılmıştır. Doku çıkarım ağının eğitilmesi için gürültü azaltma yöntemine benzer olarak BSD500 imgesinden biküçük interpolasyon ve veri arttırma yöntemleri kullanılarak küçük parça imgeler elde edilmiştir. Biküçük interpolasyon parametreleri olarak [1.0, 0.9, 0.8, 0.7] tercih edilirken veri arttırma yöntemleri olarak rotasyon ve aynalama yöntemleri tercih edilmiştir. Parça imge boyutu olarak 50×50 ve piksel bazında kaydırma işlemi içinse kaydırma adımı olarak 20 tercih edilmiştir. Bu işlemler altında 321×481 boyutunda 500 imgeden 50×50 boyutunda 252000 tane imge parçası elde edilmiştir.

TENet ağı eğitilirken varsayılan parametreleri ile Adam optimizasyon algoritması kullanılmıştır. Başlangıç öğrenme katsayı olarak 0.001 alınmış TENet ağı 50 epok boyunca eğitilmiştir. Öğrenme katsayısının değişimi için ise her 10 epok sonrası mevcut öğrenme katsayısı 0.5 ile çarpılarak kademeli olarak azaltılmıştır. Eğitim aşamasında kayıp fonksiyonu olarak MSE kayıp fonksiyonu tercih edilmiştir. TENet eğitimi için kullanılan kayıp fonksiyonu Eşitlik (3.4)’te verilmiştir:

$$l(\theta) = \frac{1}{2N} \sum_{i=1}^N \|f_{TENet}(I_N^i) - I_{YiÖ}\|^2 \quad (3.4)$$

burada θ TENet ağının parametrelerini, $I_{YiÖ}$ gerçek YİÖ imgesini, I_N ise gürültülü imgeyi ifade etmektedir. Gauss gürültüsü altında renkli ve gri seviye görüntüler için $\sigma=\{15,25,50\}$ olmak üzere 3 farklı gürültü seviyesinde eğitim yapılmıştır. Bütün deneyler Windows 11 ve Python 3.8 ortamında gerçekleşmiş ve AMD Ryzen 5700x CPU, 16 GB RAM ve Nvidia GeForce RTX 3090Ti ekran kartına sahip bir bilgisayar kullanılarak koşulmuştur. Programlama arayüzü olarak ise Keras kütüphanesinden faydalanılmıştır (Chollet, 2021). TENet ağı kullanılarak gürültülü imgeler üzerine elde edilen örnek sonuç imgesi Şekil 3.5'te verilmiştir.

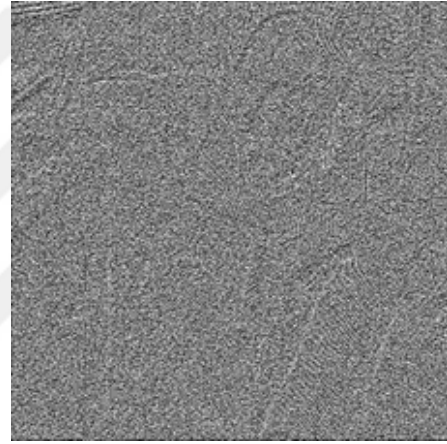
Şekil 3.5'te gri seviye Barbara imgesi için TENet ağı kullanılarak yapılan bir doku çıkarma işlemine örnek sunulmuştur. Bu amaçla Şekil 3.5-(a)'da gürültüsüz Barbara imgesi ve Şekil 3.5-(c)'de $\sigma=50$ olmak üzere Gauss gürültülü Barbara imgesi yer alırken, Şekil 3.5-(b)'de gürültü eklenmemiş Barbara imgesi için geleneksel yöntem uygulanarak elde edilen YİÖ çıkarma işleminin sonucu sunulmuştur. Şekil 3.5-(d)'de $\sigma=50$ Gauss gürültüsü eklenerek Barbara imgesine geleneksel yöntemler uygulanarak YİÖ imgesi çıkarılmıştır. Şekilden de anlaşılacağı üzere doku imgesinde gürültünün etkisi ile Barbara imgesine ait YİÖ imgesi büyük oranda bozulmaya uğramıştır. Son olarak ise Şekil 3.5-(e)'de TENet ağı kullanılarak gürültülü imgeden YİÖ imgesinin çıkarımı yapılmıştır. Şekilden de anlaşılacağı üzere elde edilen sonuç gürültünün varlığını ortadan kaldırmış ve gürültü imgeden doku bilgisinin etkili bir şekilde çıkarımına imkân sağlamıştır.



(a) Gri seviye Barbara imgesi



(b) Barbara imgesine ait YİÖ imgesi

(c) $\sigma=50$ için gürültülü Barbara imgesi(d) $\sigma=50$ için Barbara YİÖ imgesi(e) $\sigma=50$ için TENet ile elde edilen YİÖ imgesi

Şekil 3.5. Barbara imgesi ve gürültü imgeler üzerine elde edilen YİÖ sonuçları

3.3.2. Gauss Gürültüsü Azaltma Ağı

A. Çok Seviyeli Öznitelik Çıkarım Bloğu (MLF)

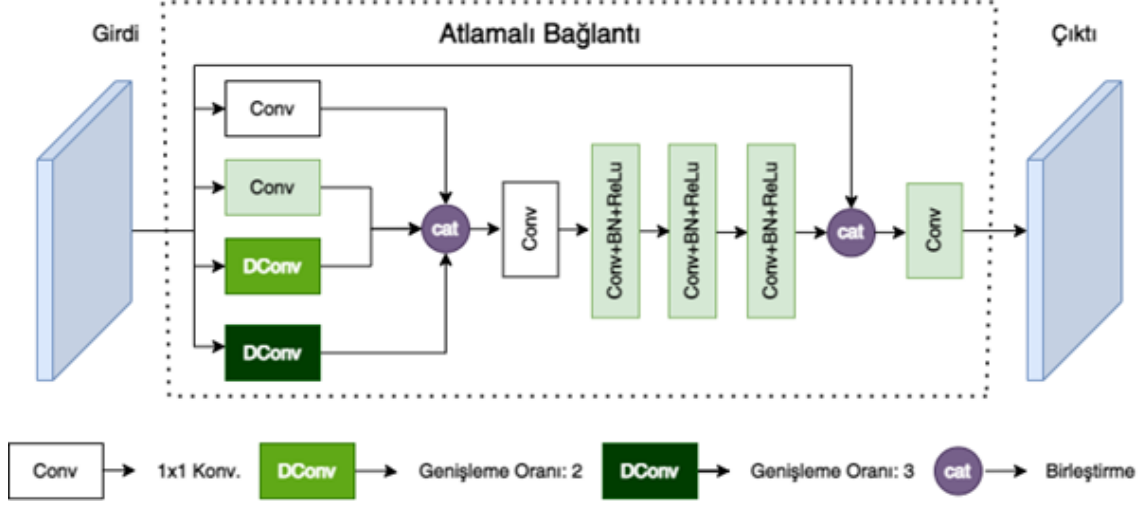
Derin öğrenme ağlarında farklı seviye öznitelik çıkarımı için genel olarak farklı boyutlarda konvolüsyon çekirdekleri tercih edilmektedir. Bu nedenle bu çalışmada, farklı seviyede özniteliklerin çıkarımı için 1x1, 3x3, 5x5 ve 7x7 boyutlarında konvolüsyon çekirdeklerinden oluşan etkili çok seviyeli öznitelik (Multilevel Feature Extraction – MLF) bloğu tasarlanmıştır. Ancak bilindiği üzere çekirdek boyutları arttıkça yapılacak olan çarpma işlemi üssel olarak arttığı için hesaplama maliyeti çokça artmaktadır ve ağın eğitim süresini uzatmakla birlikte ağın yakınsamasını zorlaştırmaktadır. Bu nedenle klasik konvolüsyon işlemini kullanmak yerine bu çalışmada genişlemiş konvolüsyon kullanımı tercih edilmiştir. Buna göre sırasıyla 5x5 ve 7x7 konvolüsyon kullanmak yerine genişleme oranı 2 ve 3 olan genişlemiş konvolüsyonlar tercih edilmiştir.

Burada farklı seviyelerde elde edilen öznitelikler art arda birleştirilmiş ve hem doğrusal olmamayı arttırmak hem de karmaşıklığı azaltmak için 1x1 konvolüsyon işlemi ve ReLU aktivasyon fonksiyonu uygulanmıştır. Daha sonra ortak özniteliklerin çıkarımı için 3 tane art arda konvolüsyon, yığın normalizasyonu ve ReLU aktivasyon fonksiyonu kullanılmıştır. Öncü elde edilen özniteliklerin ileri katmanlara iletilmesi için atlamalı bağlantı kullanılmış ve öznitelikler art arda eklenerek birleştirildikten sonra 64 adet 3x3 boyutunda konvolüsyon çekirdeği kullanarak çıktı öznitelikleri elde edilmiştir. Tasarlanan MLF bloğunun matematiksel gösterimi Eşitlik (3.5) ve (3.6)'deki gibidir:

$$f_{o1} = C_1 \left(C_{at} (D_3(f_i), D_2(f_i), C_3(f_i), C_1(f_i)) \right) \quad (3.5)$$

$$f_{o2} = C_3(C_{at}(C_{br3}(f_{o1}), f_i)) \quad (3.6)$$

burada C_{br} konvolüsyon, yığın normalizasyon ve ReLU işlemini, C_{br3} art arda uygulanan 3 tane C_{br} işlemini, C_3 3x3 konvolüsyon, C_1 1x1 konvolüsyon, D_2 ve D_3 sırasıyla genişleme oranı 2 ve 3 olan genişlemiş konvolüsyonu, C_{at} art arda eklenerek yapılan birleştirme işlemini, f_i girdi verisini, f_{o1} ara değer olarak hesaplanan öznitelikleri ve son olarak f_{o2} çıktı özniteliklerini ifade etmektedir. Bütün konvolüsyon işlemlerinde çekirdek sayısı 64 olarak seçilmiştir. MLF bloğuna ait gösterim Şekil 3.6'da yer almaktadır.



Şekil 3.6. Çok seviyeli öznetelik çıkarım bloğu (MLF)

Şekil 3.6’da yer alan “DConv” genişlemiş konvolüsyon işlemi, “Conv” işlemi klasik konvolüsyon işlemi ifade ederken, “Conv+BN+ReLU” ise konvolüsyon, yığın normalizasyon ve ReLU işlemi göstermektedir. MLF bloğunda farklı seviyede çıkarılan özneteliklerin farklılığını göstermek amacıyla Barbara imgesinin $\sigma=25$ olmak üzere eğitimi tamamlanmış ağ üzerinde öznetelik haritalarının gösterimi Şekil 3.7’de verilmiştir.



(a) 1x1 konvolüsyon katmanı



(b) 3x3 konvolüsyon katmanı



(c) 5x5 konvolüsyon katmanı



(d) 7x7 konvolüsyon katmanı

Şekil 3.7. Barbara imgesi için farklı çekirdek boyutlarına sahip konvolüsyon katmanlarında elde edilmiş öznelilik haritaları.

Şekil 3.7’de Barbara imgesi için farklı konvolüsyonlar için öznitelik haritaları sunulmuştur. Bu amaçla MLF bloğu içerisinde kullanılan konvolüsyon katmanlarından faydalanılmış ve sırasıyla 1×1 , 3×3 , 5×5 , 7×7 boyutlarındaki konvolüsyon katmanlarının çıktısı sunulmuştur. Bu amaçla eğitilmiş MLFAN ağının ilk MLF bloğu kullanılmış ve her bir konvolüsyon katmanından ilk 3 kanal tercih edilmiştir. Şekilden de anlaşılacağı üzere konvolüsyon katmanlarında çekirdek boyutunun değişimi çıkarımı yapılan özniteliklerin değişmesini sağlamıştır.

B. Öznitelik Odak Mekanizması

Farklı seviyede çıkarılan özniteliklerinin sonuca farklı etkilerinin olduğundan motivasyon kısmında bahsetmiştik. Farklı seviyelerdeki özniteliklerin ağırlıklandırılması için literatürde de sıkça kullanılan odak mekanizmalarından faydalanılmıştır. Bu amaçla özniteliklerin kanal tabanlı ağırlıklandırılması için kullanılan bir odak mekanizması tercih edilmiş ve SENet odak mekanizması kullanılmıştır (Hu vd., 2018). SENet kanal tabanlı odak mekanizması birkaç aşamadan oluşmaktadır. Evrişimli katmanlar yalnızca yerel bilgileri kullandığından ve diğer bir ifadeyle küresel bağlamsal bilgileri kullanmadığından, tüm görüntüyü ifade eden istatistikleri elde etmek için önce global ortalama havuzlama kullanılmaktadır. Buna göre $H \times W$ boyutunda C adet kanala sahip olan öznitelik haritası için global ortalama havuzlama, öznitelik haritasının boyutunu $H \times W \times C$ 'den $1 \times 1 \times C$ boyutuna indirgemektedir. Buna göre global havuzlama Eşitlik (3.7)'deki gibi ifade edilebilir:

$$g_p = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W f_c(i, j) \quad (3.7)$$

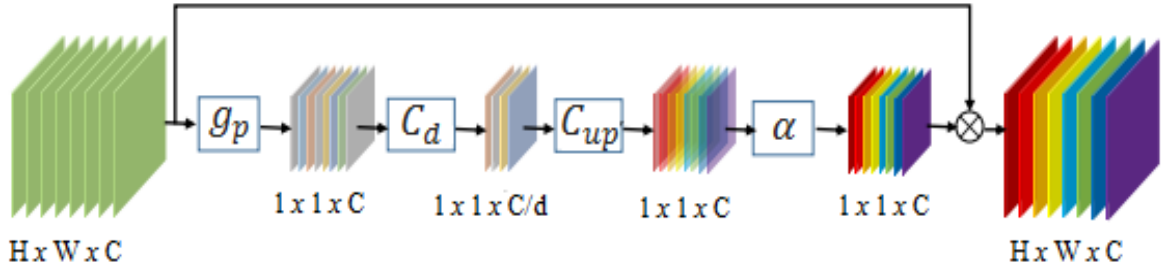
burada g_p küresel ortalama havuzu işleminin sonucunu, f_c ise girdi öznitelik haritasını ifade etmektedir. Daha sonra (Roth, 2009) makalesine göre kendinden geçmeli kapı mekanizması (self-gating mechanism -SGM) kullanılmıştır. Bu sayede kanallar arası lineer olmayan sinerjiler ve özel ilişkiler çıkarılmaktadır. Bunun için ilk olarak yumuşak daraltma (soft-shrinkage) ve sigmoid fonksiyonu kullanılmıştır. Buna göre SGM denklemi Eşitlik (3.8)'de ifade edilmiştir:

$$r_c = \alpha \left(c_{up} \left(\delta \left(c_d(g_p) \right) \right) \right) \quad (3.8)$$

burada C_d ve C_{up} sırasıyla kanal boyut azaltma ve kanal üst örnekleme operatörleridir. Boyut azaltma işlemi d adet konvolüsyon katmanı ve ReLU aktivasyon fonksiyonu ile yapılırken üst örnekleme işlemi C adet (girdi kanal sayısı) konvolüsyon katmanı ile yapılır, ağırlıklandırma işlemi için Sigmoid aktivasyon fonksiyonu kullanılmıştır. Boyut azaltma işlemi $1 < d < C$ olmak üzere d katsayısına bağlı olarak C/d oranında istenilen şekilde yapılabilmektedir. SENet makalesinde yazarlar $1/4$ oranını tavsiye etmişlerdir ve ilgili çalışmada da aynı şekilde kullanılmış yani $d=16$ olarak alınmıştır. Son olarak ise farklı seviyelerde elde edilen f_c öznetelikleri r_c ağırlıkları ile kanal boyunca çarpılarak ağırlıklandırma işlemi gerçekleştirilmektedir. Ağırlıklandırma işlemi Eşitlik (3.9)'daki gibi ifade edilebilir:

$$f_c^* = f_c * r_c \quad (3.9)$$

burada f_c^* kanal bazında ağırlıklandırılmış öznetelik haritasını ifade etmektedir. Buna göre kanal tabanlı odak mekanizması Şekil 3.8'deki gibi gösterilebilir

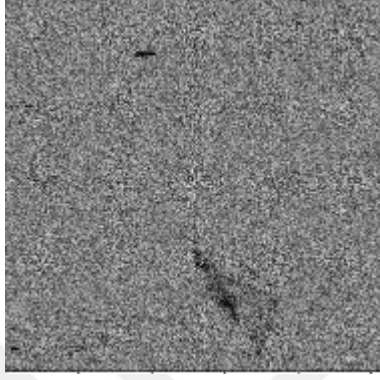


Şekil 3.8. Odak mekanizması

Öznetelik odak mekanizmasının etkisini göstermek için MLF bloğunun odak mekanizmasına verilmeden önceki ve sonraki öznetelik haritalarından örnekler Şekil 3.9'da sunulmuştur. Bunun için Barbara imgesine $\sigma=25$ olmak üzere Gauss gürültüsü eklenmiş ve eğitimi tamamlanan ağırlık öznetelik haritaları incelenmiştir.

Şekil 3.9-(a)'da Barbara imgesinin odak mekanizması öncesi bir öznetelik haritası verilmiş ve Şekil 3.9-(b)'de odak mekanizması uygulandıktan sonraki hali verilmiştir. Şekil 3.9-(c), Şekil 3.9-(d), Şekil 3.9-(e) ve Şekil 3.9-(f)'de sunulan imgeler de aynı şekilde ifade

edilebilir. Kanal odak mekanizmasının ağırlıklandırılma yoluyla gürültü içerisindeki Barbara imgesine ait faydaları bilgileri ön plana çıkarması Şekil 3.9 aracılığıyla kolaylıkla görülebilmektedir. Şekil 3.9'daki gösterim için MLFAN ağındaki ilk MLF bloğu ve odak mekanizmasının çıktısı dikkate alınmıştır.



(a) Odak mekanizması öncesi



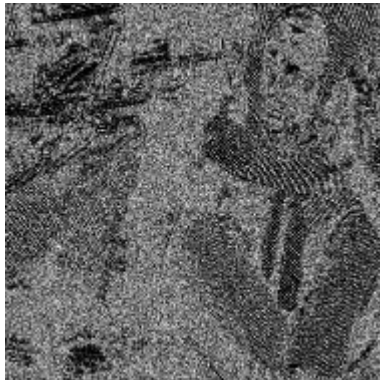
(b) Odak Mekanizması sonrası



(c) Odak mekanizması öncesi



(d) Odak mekanizması sonrası



(e) Odak mekanizması öncesi



(f) Odak mekanizması sonrası

Şekil 3.9. Odak mekanizması öncesi ve sonrası öznelilik haritaları

C. Çok Seviyeli Öznitelik Odak Ağı (MLFAN)

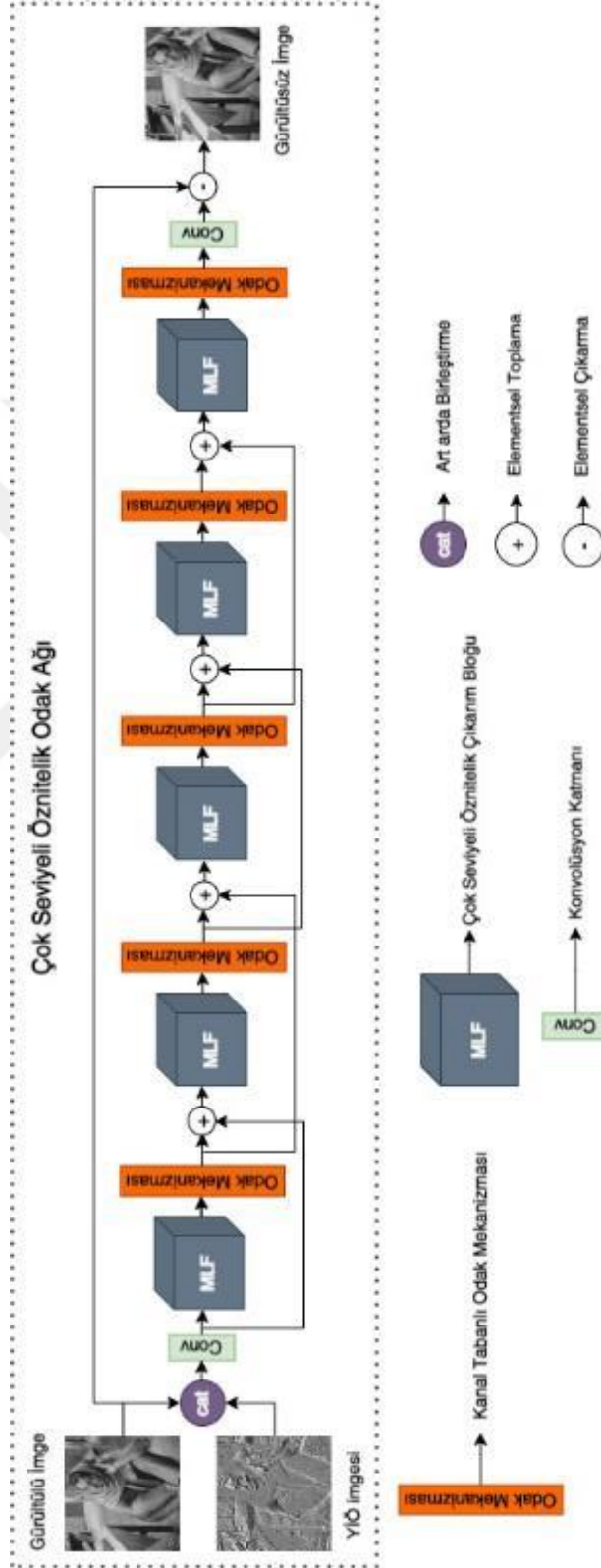
Gürültü azaltma işlemi için tasarlanan Çok Seviyeli Öznitelik Odak Ağı (MLFAN) yapı olarak 5 adet MLF bloğu ve 5 adet kanal tabanlı odak mekanizmasından oluşmaktadır. Buna göre ilk olarak gürültülü imge ve TENet ağı ile gürültülü görüntüden elde edilen YİÖ imgesi art arda ekleme yoluyla birleştirilmekte ve MLFAN ağına girdi olarak verilmektedir. Burada amaç MLFAN ağı ile gürültü azaltma işlemi gerçekleştirilirken YİÖ imgesi aracılığı ile bu işleme bir rehberlik yapmak ve imgenin detay bilgilerinin korunarak özgün yapısından uzaklaşmasının önüne geçmektir. MLFAN ağı için girdi verisi olarak gürültülü imge ve doku imgesi olan YİÖ imgesi birleştirildikten sonra öncül özniteliklerin elde edilmesi için 64 kanallı 3×3 çekirdek boyutuna sahip bir konvolüsyon işlemine tabi tutulmaktadır. Daha sonra MLF bloğuna girdi olarak verilmekte ve MLF bloğunun çıktı öznitelik haritasına, çoklu seviyede elde edilen bu özniteliklerin ağırlıklandırılması amacıyla kanal tabanlı odak mekanizması uygulanmaktadır. Elde edilen ağırlıklandırılmış bu öznitelikler yeni bir MLF ağına girdi olarak verilerek işlem 5 kez tekrarlanmakta ve son çıktı öznitelik haritasında çıktı görüntüsünü yani gürültü haritasının (kalıntı imgesinin) elde edilmesi için tek kanallı (renkli imgeler için 3 kanal) 1×1 bir konvolüsyon işlemi uygulanmaktadır. Gürültüsü giderilmiş imgenin elde ediniimi için ise girdi verisi olarak kullanılan gürültülü imgeden kalıntı imgesi çıkarılmaktadır. MLFAN ağı içerisinde ağın yakınsamasına katkı sunması ve sığ katmanlarda çıkarılan faydalı öznitelik bilgilerinin ileriki katmanları iletimi için atlamalı bağlantılar kullanılmıştır. Bunun için her bir MLF ve odak mekanizması işleminin çıktısı bir sonraki MLF ve odak mekanizması bloğunun çıktısı ile elementsel toplama yoluyla birleştirilmektedir. Bu bilgiler ışığında MLFAN ağı için matematiksel gösterim Eşitlik (3.10) ve Eşitlik (3.11)'deki gibi verilebilir:

$$f_1 = C_3(C_{at}(I_n, I_{YİÖ})) \quad (3.10)$$

$$I_d = I_n - C_1(M_{FA5}(f_1)) \quad (3.11)$$

burada I_n ve $I_{YİÖ}$ sırasıyla gürültülü imge ve YİÖ imgesini, f_1 öncül öznitelikleri, C_{at} art arda eklenerek yapılan birleştirme işlemi, C_3 64 filtre sayılı ve 3×3 çekirdek boyutunda konvolüsyon işlemi, M_{FA5} aralarında atlamalı bağlantı bulunan 5 adet MLF bloğu ve odak mekanizmasını, C_1 kalıntı imgesinin elde edilmesi için uygulanan 1 filtre sayılı (renkli

imgeler için 3) ve 1×1 çekirdek boyutundaki konvolüsyon işlemini ve I_d gürültüsü azaltılmış imgeyi ifade etmektedir. MLFAN ağına ait gösterim Şekil 3.10'da verilmiştir.



Şekil 3.10. MLFAN ağı

D. MLFAN Ağının Eğitilmesi

MLFAN ağının eğitilmesi için doku çıkarma ağından farklı bir veri seti kullanılmıştır. Bu amaçla 4744 imgeden oluşan Waterloo Keşif veri seti tercih edilmiştir (Ma vd., 2016). Derin öğrenme yöntemlerinde kullanılan görüntünün kendisi yerine parça çıkarma yapılarak eğitilmesi işlemi bu çalışmada da kullanılmıştır. Bu amaçla bikübik interpolasyon ve veri artırma yöntemleri kullanılarak 4744 adet çeşitli boyutlardaki yüksek çözünürlüğe sahip imgelerden küçük parça imgeler elde edilmiştir. Bikübik interpolasyon parametreleri olarak [1.0, 0.9, 0.8, 0.7] tercih edilirken veri artırma yöntemleri olarak rotasyon ve aynalama yöntemleri tercih edilmiştir. Parça imge boyutu olarak 44x44 ve piksel bazında kaydırma işlemi içinse kaydırma adımı olarak 40 tercih edilmiştir. Bu işlemler altında çeşitli boyutlardaki yüksek çözünürlüklü 4744 imgeden 44x44 boyutunda yaklaşık 1.1 milyon tane imge parçası elde edilmiştir. Validasyon aşamasında ise ImageNet validasyon seti tercih edilmiştir (Deng vd., 2014).

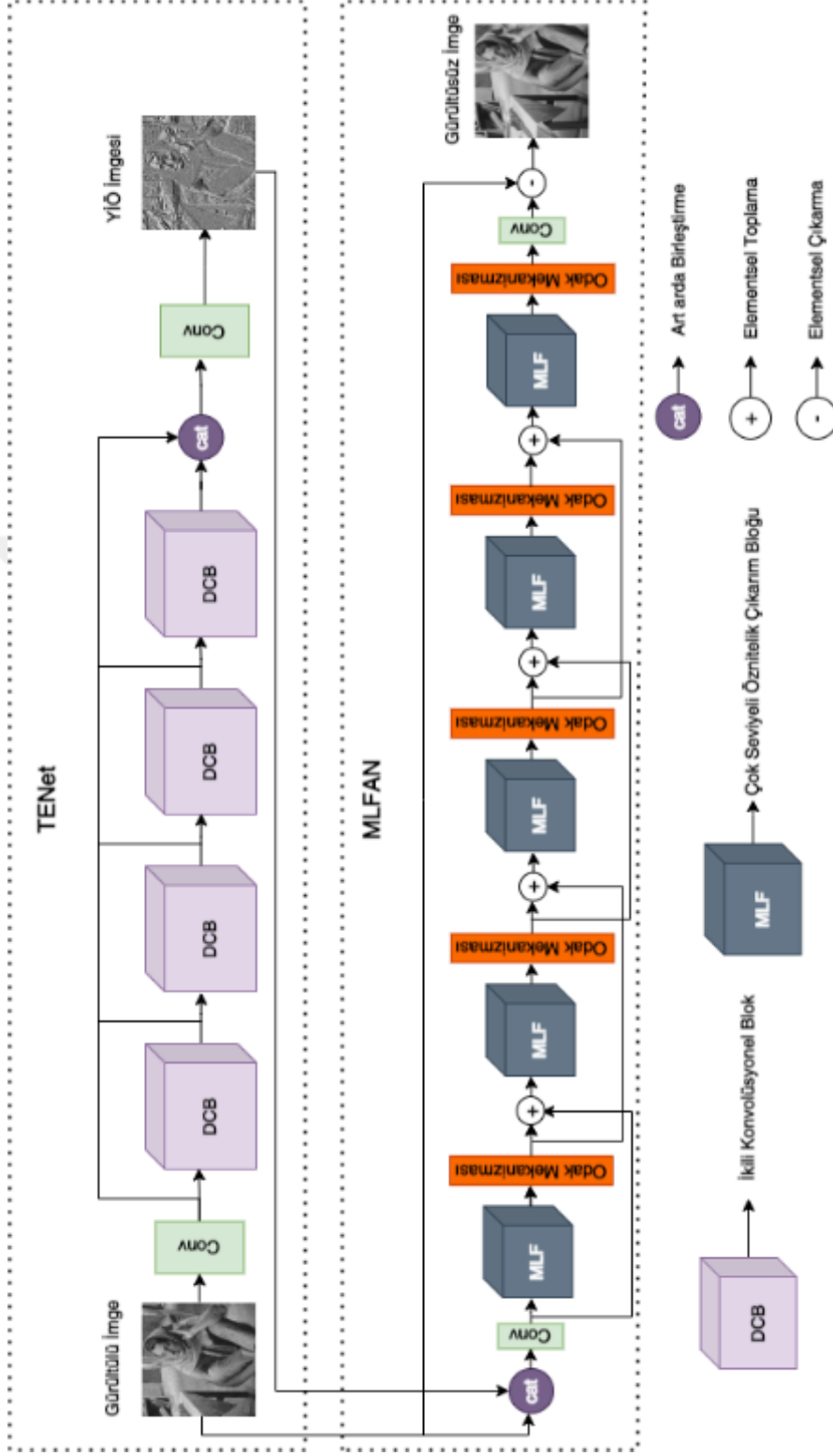
MLFAN ağı eğitilirken varsayılan parametreleri ile Adam optimizasyon algoritması kullanılmıştır. Başlangıç öğrenme katsayı olarak 0.001 alınmış ve MLFAN ağı 80 epok boyunca eğitilmiştir. Öğrenme katsayısının değişimi için ise 40. epok bitiminde öğrenme katsayı 1/10 ile çarpılmış ve 20 epok daha eğitildikten sonra 1/10 ile tekrar çarpılarak 0.00001'e kadar azaltılmıştır. Gauss gürültüsü altında renkli ve gri seviye görüntüler için $\sigma=\{15,25,50\}$ olmak üzere 3 farklı gürültü seviyesinde eğitim yapılmıştır. Eğitim aşamasında kayıp fonksiyonu olarak MSE kayıp fonksiyonu tercih edilmiştir. MLFAN ağının eğitimi için kullanılan kayıp fonksiyonu Eşitlik (3.12)'de verilmiştir:

$$l(\theta) = \frac{1}{2N} \sum_{i=1}^N \left\| \left(I_N^i - f_{MLFAN}(I_N^i, I_{YIÖ}^i) \right) - I_C \right\|^2 \quad (3.12)$$

burada θ MLFAN ağının parametrelerini, $I_{YIÖ}$ TENet ile elde edilen YİÖ imgesini, I_N gürültülü imgeyi ve I_C gürültüsüz imgeyi ifade etmektedir. Bütün deneyler Windows 11 ve Python 3.8 ortamında gerçekleşmiş ve AMD Ryzen 5700x CPU, 16 GB RAM ve Nvidia GeForce RTX 3090Ti ekran kartına sahip bir bilgisayar kullanılarak koşulmuştur. Programlama arayüzü olarak ise Keras kütüphanesinden faydalanılmıştır (Chollet vd., 2021).

MLFAN ađı eđitim ařamasında TENet ađından elde edilen YİÖ imgesini öncül bilgi olarak girdi olarak almaktadır. Buna göre gürültü azaltma sürecinin tamamını ifade eden TENet ve MLFAN ađının birleřtirilmiř hali Őekil 3.11’de verilmiřtir.





Şekil 3.1.1. MLFAN ve TENet ağı

3.3.3. Deneysel Karşılaştırmalar

Bu kısımda, yapılan çalışma ile elde edilen sonuçlar sunulmuştur. MLFAN ve TENet ağı eğitildikten sonra gri seviye ve renkli imgeler için test edilmiştir. Test aşamasında gri seviyeler için veri seti olarak Set12 ve Set68 (Roth & Black, 2005) kullanılırken renkli imgeler için ise Kodak24 (Franzen, 1999), McMaster (Zhang vd., 2011) ve Set68 imgesinin renkli versiyonu olan CBSD68 veri seti tercih edilmiştir. Önerilen yöntem model tabanlı yöntemlerden BM3D (Dabov vd., 2007) algoritması ve öğrenme tabanlı yöntemlerden TNRD (Chen & Pock, 2016), DnCNN (Zhang vd., 2017), FFDNet (Zhang vd., 2018), IRCNN (Zhang vd., 2017b) yöntemleri ile karşılaştırılmıştır. Bunlara ek olarak çok seviyeli öznetelik çıkarma yöntemlerinden olan McCNN (Guo vd., 2020) ve yakın zamanda yapılmış ve karşılaştırma için çokça kullanılan odak tabanlı ADNet (Tian vd., 2020) ve ikili sinir ağı yapısına sahip olan DudeNet (Tian vd., 2021) yöntemleri ile de karşılaştırılmıştır. Ek olarak ise gri seviyeler için düşük sayıda veri seti için koşulan DnCNN ağı, adil bir karşılaştırma yapılabilmesi için aynı hiper parametreler altında Waterloo Keşif veri seti ile koşulmuş ve DnCNN+ olarak isimlendirilerek eklenmiştir.

Set68 veri seti için MLFAN ağı ve karşılaştırma için kullanılan yöntemler ile elde edilen sonuçlar Tablo 3.1’de sunulmuştur. Burada veri seti için ortalama PSNR değerleri verilmiştir. En iyi sonucu veren yöntem kırmızı renk ile işaretlenirken en iyi ikinci sonucu veren ise mavi renk ile belirtilmiştir. Buna göre MLFAN ağı diğer yöntemlerden PSNR metriği altında daha iyi sonuç vermiştir. Set68 veri seti için ortalama PSNR değerleri altında en iyi ikinci sonucu çok seviyeli öznetelik çıkarımı kullanan McCNN yöntemi vermiştir. Bu da çok seviyeli öznetelik çıkarımının etkisini göstermektedir.

Tablo 3.1. Set68 veri seti için farklı metotlar ile elde edilen ortalama PSNR sonuçları

Yöntemler	BM3D	TNRD	DnCNN	DnCNN+	FFDNet	IRCNN	McCNN	ADNet	DudeNet	MLFAN
$\sigma = 15$	31.07	31.42	31.72	31.71	31.63	31.63	31.78	31.74	31.78	31.80
$\sigma = 25$	28.57	28.92	29.23	29.21	29.19	29.15	29.30	29.25	29.29	29.34
$\sigma = 50$	25.62	25.97	26.23	26.25	26.29	26.19	26.33	26.29	26.31	26.38

Set12 veri seti için MLFAN ağı ve karşılaştırma için kullanılan yöntemler ile elde edilen sonuçlar Tablo 3.2’de sunulmuştur. Burada veri seti için ayrı ayrı imgelerin PSNR

değerlerinin yanı sıra ortalama PSNR değerleri de verilmiştir. En iyi sonucu veren yöntem kırmızı renk ile işaretlenirken en iyi ikinci sonucu veren ise mavi renk ile belirtilmiştir.

Tablo 3.2. Set12 veri seti için farklı metotlar ile elde edilen PSNR değerleri

İmgeler	C.Man	House	Peppers	Starfish	Monar.	Airpl.	Parrot	Lena	Barbara	Boat	Man	Couple	Ortalama
$\sigma = 15$													
BM3D	31.91	34.93	32.69	31.14	31.85	31.07	31.37	34.26	33.10	32.13	31.92	32.10	32.37
TNRD	32.19	34.53	33.04	31.75	32.56	31.46	31.63	34.24	32.13	32.14	32.23	32.11	32.50
DnCNN	32.61	34.97	33.30	32.20	33.09	31.70	31.83	34.62	32.64	32.42	32.46	32.47	32.86
FFDNet	32.43	35.07	33.25	31.99	32.66	31.57	31.81	34.62	32.54	32.38	32.41	32.46	32.77
IRCNN	32.55	34.89	33.31	32.02	32.82	31.70	31.84	34.53	32.43	32.34	32.40	32.40	32.77
McCNN	32.65	35.20	33.35	32.29	33.30	31.78	31.98	34.65	32.69	32.47	32.44	32.53	32.94
DudeNet	32.71	35.13	33.38	32.29	33.28	31.78	31.93	34.66	32.73	32.46	32.46	32.49	32.94
ADNet	32.81	35.22	33.49	32.17	33.17	31.86	31.96	34.71	32.80	32.57	32.47	32.58	32.98
DnCNN+	32.74	35.15	33.44	32.21	33.18	31.79	31.95	34.57	32.72	32.48	32.45	32.53	32.93
MLFAN	32.88	35.39	33.59	32.36	33.33	31.92	32.05	34.78	32.86	32.61	32.53	32.68	33.08
$\sigma = 25$													
BM3D	29.45	32.85	30.16	28.56	29.25	28.42	28.93	32.07	30.71	29.90	29.61	29.71	29.97
TNRD	29.72	32.53	30.57	29.02	29.85	28.88	29.18	32.00	29.41	29.91	29.87	29.71	30.06
DnCNN	30.18	33.06	30.87	29.41	30.28	29.13	29.43	32.44	30.00	30.21	30.10	30.12	30.43
FFDNet	30.10	33.28	30.93	29.32	30.08	29.04	29.44	32.57	30.01	30.25	30.11	30.20	30.44
IRCNN	30.08	33.06	30.88	29.27	30.09	29.12	29.47	32.43	29.92	30.17	30.04	30.08	30.38
McCNN	30.30	33.39	30.93	29.48	30.47	29.18	29.43	32.55	30.20	30.27	30.14	30.19	30.54
DudeNet	30.23	33.24	30.98	29.53	30.44	29.14	29.48	32.52	30.15	30.24	30.08	30.15	30.52
ADNet	30.34	33.41	31.14	29.41	30.39	29.17	29.49	32.61	30.25	30.37	30.08	30.24	30.58
DnCNN+	30.36	33.36	30.97	29.40	30.42	29.17	29.54	32.52	30.16	30.26	30.08	30.20	30.54
MLFAN	30.53	33.79	31.18	29.67	30.61	29.34	29.66	32.75	30.41	30.42	30.18	30.41	30.75
$\sigma = 50$													
BM3D	26.13	29.69	26.68	25.04	25.82	25.10	25.90	29.05	27.22	26.78	26.81	26.46	26.72
TNRD	26.62	29.48	27.10	25.42	26.31	25.59	26.16	28.93	25.70	26.94	26.98	26.50	26.81
DnCNN	27.03	30.00	27.32	25.70	26.78	25.87	26.48	29.39	26.22	27.20	27.24	26.90	27.18
FFDNet	27.05	30.37	27.54	25.75	26.81	25.89	26.57	29.66	26.45	27.33	27.29	27.08	27.32
IRCNN	26.88	29.96	27.33	25.57	26.61	25.89	26.55	29.40	26.24	27.17	27.17	26.88	27.14
McCNN	27.13	30.33	27.46	25.89	26.91	25.94	26.59	29.62	26.53	27.29	27.31	27.13	27.34
DudeNet	27.22	30.27	27.51	25.88	26.93	25.88	26.50	29.45	26.49	27.26	27.19	26.97	27.30
ADNet	27.31	30.59	27.69	25.70	26.90	25.88	26.56	29.59	26.64	27.35	27.17	27.07	27.37
DnCNN+	27.37	30.38	27.48	25.64	26.89	25.96	26.58	29.52	26.45	27.21	27.22	27.10	27.32
MLFAN	27.53	30.90	27.69	25.97	27.06	26.09	26.64	29.75	26.96	27.44	27.36	27.36	27.56

Tablo 3.2’de $\sigma = 15$ için bakıldığında hem ortalama değer olarak hem de imge bazında bakıldığında en iyi sonucu önerilen MLFAN ağı vermiştir ve ortalama PSNR değeri açısından ikinci en iyi sonucu veren ADNet yönteminden 10db daha iyi sonuç vermiştir. Ancak “Barbara” imgesi göz önüne alındığında model tabanlı öğrenme yöntemi olan BM3D algoritması öğrenme tabanlı diğer yöntemleri de geçerek en iyi sonucu vermiştir. Bunun sebebi Barbara imgesinin benzer imge parçalarını çok fazla içermesi ve BM3D yönteminin gürültü azaltma işlemi sırasında benzer imge parçalarından faydalanması sonucu gürültü azaltma işleminde başarılı olmasıdır. Buna rağmen MLFAN ağı diğer öğrenme tabanlı yöntemleri aşarak en iyi ikinci sonucu vermiştir ve bu da önerilen yöntemin üstünlüğünü göstermektedir. $\sigma = 25$ ve $\sigma = 50$ için bakacak olursak ortalama PSNR değerleri için önerilen

yöntem yine en iyi sonuçları vermiştir. Buna ek olarak kendisine en yakın sonucu veren ADNet algoritmasına karşı PSNR metriği altında sırasıyla $\sigma = 25$ ve $\sigma = 50$ için 17 dB ve 19db daha üstün bir sonuç üretmektedir. Bu da önerilen yöntem olan MLFAN ağının Set12 veri seti için yüksek seviye gürültülerde elde edilen PSNR sonuçlarının doku bilgisinin öncül bilgi olarak alınması ve öznel odak mekanizması ile çok seviyeli öznel odakların ağırlıklandırılması sayesinde başarımının daha da arttığını göstermektedir. İmgelere bireysel olarak baktığımızda bütün gürültü seviyeleri için en iyi sonucu “Barbara” imgesi hariç MLFAN ağı vermiştir. İkinci en iyi sonuçları ise çoğunlukla odak mekanizması kullanan ADNet ve çoklu seviye öznel odak çıkarımı kullanan McCNN ağları vermiştir. Bu da odak mekanizmasının ve çoklu seviye öznel odak çıkarımının derin öğrenme yöntemleri ile gürültü azaltma işlemi için etkili bir yöntem olduğunu gözler önüne sermektedir. Ayrıca DnCNN+ sonuçlarına bakıldığında bazı imgeler için en iyi ikinci sonucu verdiğini görmekteyiz. Bu da büyük veri setinin önerilen yöntem katkısını gözler önüne sermektedir. Ayrıca ortalama PSNR metriği için de DnCNN+, küçük veri seti ile eğitilen DnCNN yönteminden bütün gürültü seviyelerinde daha iyi sonuç vermiştir.

PSNR metriğinin yanı sıra değerlendirme metriği olarak SSIM metriği de kullanılmıştır. Bu amaçla Set12 ve Set68 için ortalama SSIM değerleri Tablo 3.3’te sunulmuştur. McCNN ve DnCNN yöntemleri yazarlar tarafından çalışmada sunulmadığı için bu tabloya eklenememiştir. Tablo 3.3’te sunulan sonuçlardan da anlaşılacağı üzere MLFAN ağı, SSIM metriği altında da karşılaştırılan diğer yöntemlerden daha iyi sonuç vermektedir. Kendisine en yakın sonucu ADNet yöntemi vermiştir. Burada dikkate değer bir konu da FFDNet yönteminin PSNR metriği altında DnCNN yönteminin gerisinde kalırken SSIM metriği altında daha üstün bir sayısal sonuç göstermesi olmuştur. Bu da PSNR metriğinin tek başına değerlendirme işlemi için kullanımının yeterli olmayacağını gözler önüne sermektedir.

Tablo 3.3. Set12 ve Set68 veri setinde farklı metotlar ile elde edilen ortalama SSIM değerleri

Veri seti	Yöntem	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$
Set12	BM3D	0.8946	0.8491	0.7653
	DnCNN	0.9026	0.8617	0.7825
	DnCNN+	0.9056	0.8645	0.7835
	FFDNet	0.9031	0.8632	0.7869
	IRCNN	0.9008	0.8601	0.7804
	ADNet	0.9075	0.8669	0.7885
	MLFAN	0.9087	0.8702	0.7939
Set68	BM3D	0.8722	0.8017	0.6869
	DnCNN	0.8906	0.8278	0.7190
	DnCNN+	0.8948	0.8322	0.7204
	FFDNet	0.8905	0.8282	0.7213
	IRCNN	0.8881	0.8249	0.7171
	ADNet	0.8965	0.8344	0.7236
	MLFAN	0.8973	0.8371	0.7271

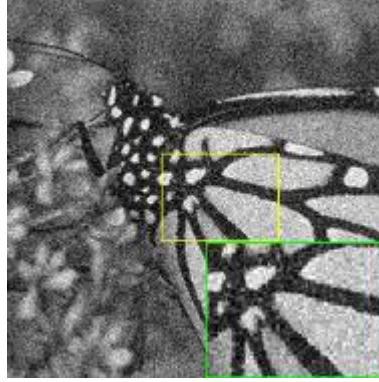
Sayısal sonuçların yanı sıra önerilen yöntemin çeşitli gauss gürültü seviyeleri altında gürültü azaltma işlemi sonrasında elde edilen görsel sonuçları da verilmiştir. Gri seviye ve renkli seviye olmak üzere 3 farklı gürültü seviyesi için alınan görsel sonuçlar SSIM ve PSNR metrikleri altında sayısal sonuçlarla beraber sunulmuştur.

Bu amaçla ilk olarak Set12 veri seti içerisinde yer alan “Monar” imgesinin $\sigma = 25$ Gauss gürültüsü altında MLFAN ağı ve literatürdeki diğer çalışmalar ile elde edilen gürültü azaltma işlemi sonucu Şekil 3.12’de sunulmuştur. Buna göre imge içerisindeki benzer parçaları baz alarak gürültü azaltma yapan ve model tabanlı yöntemlerden biri olan BM3D yöntemi PSNR ve SSIM metriği altında sırasıyla 29.25 ve 0.8984 sayısal sonuçlarını vermiştir. TNRD yöntemi ise çok katmanlı sinir ağı ve eğitim için kullanılan büyük eğitim seti sayesinde PSNR ve SSIM metriği altında sırasıyla 29.85 ve 0.9035 sonuçları ile BM3D yöntemini geçmiştir. Ek olarak IRCNN yöntemi de kabul edilebilir sonuçlar göstermiştir, bu da matematiksel metotların CNN ağlarına entegrasyonunun başarısını gözler önüne sermektedir. DnCNN ile FFDNet SSIM metriği altında hemen hemen aynı sonuçları verirken PSNR metriği altında DnCNN 0.2db yüksek sonuç vererek daha başarılı olmuştur.

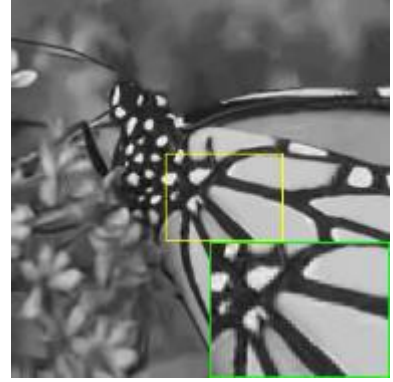
Bunun nedeni FFDNet ağının gürültü seviyesi haritasını kullanarak esneklik kazanırken, DnCNN ağının her gürültü seviyesi için ayrı ayrı eğitilmesi olarak da düşünülebilir.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 20.25 dB - 0.4364



(c) BM3D / 29.25 dB - 0.8984



(d) TNRD / 29.85 dB - 0.9035



(e) DnCNN / 30.28 dB - 0.9204



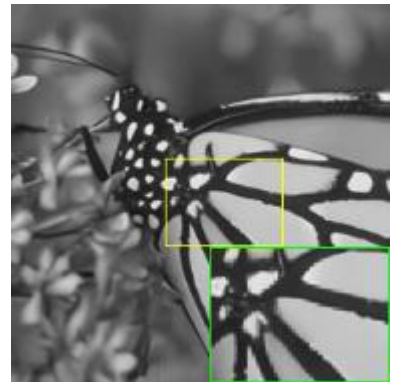
(f) FFDNet / 30.08 dB - 0.9205



(g) IRCNN / 30.09 dB - 0.9171



(h) ADNet / 30.39 dB - 0.9221



(i) MLFAN / 30.61 dB - 0.9252

Şekil 3.12. Set12 veri seti Monar imgesinin $\sigma=25$ için gürültü azaltma sonuçları

Ayrıca bu karşılaştırmadan tek bir metrik yerine birden fazla metrik kullanımının isabetli bir karar olduğu anlaşılmaktadır. ADNet ağı ise çıkarılan özneliklerin ağırlıklandırmak için odak mekanizması kullanılması sayesinde PSNR ve SSIM metriği altında diğerlerinden daha üstün bir sonuç vermektedir. Önerilen MLFAN ağı ise hem PSNR hem SSIM metriği altında kendisine en yakın sonucu veren ADNet ağına kıyasla 0.22 dB ve 0.0031 daha üstün sonuç vererek ve görsel olarak da incelendiğinde imge detaylarını daha iyi koruyarak diğer yöntemler arasında en başarılı sonucu göstermiştir.

Set68 veri seti içerisinde yer alan test011 imgesi için $\sigma=15$ gürültü seviyesi altında gürültü azaltma işleminin sonuçları Şekil 3.13'te sunulmuştur. Model tabanlı bir yöntem olan BM3D ile öğrenme tabanlı bir yöntem olan TNRD yöntemi SSIM metriği altında 0.9152 ile aynı sonuçları verirken PSNR metriği altında 31.33db ve 31.38db ile birbirine çok yakın sonuçlar vermiştir. Bunun ana nedeni test011 imgesinin çok fazla benzer küçük yapılar içermesi ve BM3D yönteminin bu benzerlikleri kullanarak gürültü azaltma işlemi yapmasıyla etkili sonuçlar üretebilmesidir. IRCNN ağı bir sonraki en iyi sonucu verirken, FFDNet ve DnCNN ağı SSIM metriği altında sırasıyla 0.9256 ve 0.9259 sonuçları ile birbirine yakın sonuçlar vermiştir. Buna ek olarak PSNR metriği altında FFDNet ağına kıyasla DnCNN ağı 0.18db daha iyi sonuç göstermiş ancak görsel olarak incelendiğinde DnCNN ağı çok fazla detay kaybına neden olurken FFDNet ağı bunu koruyabilmiştir. Bunun nedeninin FFDNet ağının imgenin özgün yapısından uzaklaşmayı önlemek için gürültülü görüntüye ek olarak ekstra bir girdi olarak gürültü seviyesi haritasını kullanması olduğu söylenebilir. ADNet ağı ise PSNR için 31.99db ve SSIM metriği için 0.9295 sonucunu verirken, önerilen yöntem MLFAN ağı ise sırasıyla 32.13db ve 0.9323 sonucunu vermiştir. Görsel olarak da incelendiğinde diğer yöntemlere kıyasla imge özgün yapısını daha iyi koruduğu görülmektedir.



Şekil 3.13. Set68 veri seti test011 imgesinin $\sigma=15$ için gürültü azaltma sonuçları

Gri seviyelerde görsel sonuçlar için son olarak ise $\sigma=50$ seviyesinde Gauss gürültüsünün giderimi incelenmiştir. Bunun için Set12 veri seti içerisinde Parrot imgesi tercih edilmiştir. Önerilen yöntem MLFAN ağı ve karşılaştırma işlemi için kullanılan model tabanlı ve öğrenme tabanlı yöntemler için elde edilen sonuçlar Şekil 3.14’te verilmiştir. Buna göre Parrot imgesi için yüksek seviyeli gauss gürültüsü altında incelendiğinde PSNR ve SSIM metrikleri altında MLFAN ağı en iyi sonucu verirken en iyi ikinci sonucu veren çalışma ise her iki metrik altında da FFDNet ağı olmuştur. MLFAN ağı FFDNet ağından PSNR metriği altında 0.07db daha iyi sonuç verirken SSIM metriği altında MLFAN ağı 0.8039 sonucu vermiş FFDNet ağı ise 0.8004 sonucunu vermiştir. MLFAN ağının her iki

metrik altında da en iyi sonucu vermesi önerilen yöntemin gri seviye görüntüler için yüksek seviyeli gürültülerdeki başarımını ortaya koymaktadır.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 15.04 dB - 0.2459



(c) BM3D / 25.90 dB - 0.7868



(d) TNRD / 26.16 dB - 0.7847



(e) DnCNN / 26.48 dB - 0.7981



(f) FFDNet / 26.57 dB - 0.8004



(g) IRCNN / 26.55 dB - 0.7953



(h) ADNet / 26.56 dB - 0.7999



(i) MLFAN / 26.64 dB - 0.8039

Şekil 3.14. Set12 veri seti Parrot imgesinin $\sigma=50$ için gürültü azaltma sonuçları

Gri seviye görüntüler için Monar imgesinin diğer gürültü seviyelerinde ($\sigma=15$ ve $\sigma=50$ için) elde edilen görsel sonuçları Şekil 3.15 ve Şekil 3.16'da yer almaktadır.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 24.61 dB - 0.6399



(c) BM3D / 31.86 dB - 0.9353



(d) TNRD / 32.56 dB - 0.9417



(e) DnCNN / 33.09 dB - 0.9510



(f) FFDNet / 32.66 dB - 0.9491



(g) IRCNN / 32.82 dB - 0.9477



(h) ADNet / 33.17 dB - 0.9508



(i) MLFAN / 33.33 dB - 0.9523

Şekil 3.15. Set12 veri seti Monar imgesinin $\sigma=15$ için gürültü azaltma sonuçları



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 14.73 dB - 0.3007



(c) BM3D / 25.82 dB - 0.8197



(d) TNRD / 26.31 dB - 0.8264



(e) DnCNN / 26.78 dB - 0.8492



(f) FFDNet / 26.81 dB - 0.8511



(g) IRCNN / 26.61 dB - 0.8454



(h) ADNet / 26.90 dB - 0.8541



(i) MLFAN / 27.06 dB - 0.8550

Şekil 3.16. Set12 veri seti Monar imgesinin $\sigma=50$ için gürültü azaltma sonuçları

Gri seviye görüntüler için Parrot imgesinin diğer gürültü seviyelerinde ($\sigma=15$ ve $\sigma=25$ için) elde edilen görsel sonuçları Şekil 3.17 ve Şekil 3.18'de yer almaktadır.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 24.86 dB - 0.5639



(c) BM3D / 31.37 dB - 0.8972



(d) TNRD / 31.63 dB - 0.9043



(e) DnCNN / 31.83 dB - 0.9093



(f) FFDNet / 31.81 dB - 0.9045



(g) IRCNN / 31.84 dB - 0.9039



(h) ADNet / 31.96 dB - 0.9111



(i) MLFAN / 32.05 dB - 0.9124

Şekil 3.17. Set12 veri seti Parrot imgesinin $\sigma=15$ için gürültü azaltma sonuçları



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 20.63 dB - 0.4117



(c) BM3D / 28.93 dB - 0.8533



(d) TNRD / 29.18 dB - 0.8573



(e) DnCNN / 29.43 dB - 0.8667



(f) FFDNet / 29.44 dB - 0.8624



(g) IRCNN / 29.47 dB - 0.8608



(h) ADNet / 29.49 dB - 0.8682



(i) MLFAN / 29.66 dB - 0.8756

Şekil 3.18. Set12 veri seti Parrot imgesinin $\sigma=25$ için gürültü azaltma sonuçları

Gri seviye görüntüler için Set68 veri setinde yer alan test011 imgesinin diğer gürültü seviyelerinde ($\sigma=25$ ve $\sigma=50$ için) elde edilen görsel sonuçları Şekil 3.19 ve Şekil 3.20'de yer almaktadır.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 20.71 dB – 0.5360



(c) BM3D / 28.56 dB – 0.8608



(d) TNRD / 28.60 dB – 0.8620



(e) DnCNN / 29.17 dB – 0.8781



(f) FFDNet / 29.15 dB – 0.8796



(g) IRCNN / 29.11 dB – 0.8753



(h) ADNet / 29.22 dB – 0.8830



(i) MLFAN / 29.45 dB – 0.8881

Şekil 3.19. Set68 veri seti test011 imgesinin $\sigma=25$ için gürültü azaltma sonuçları



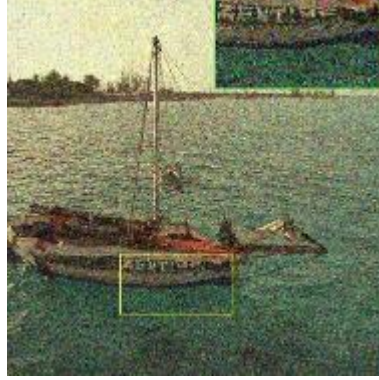
Şekil 3.20. Set68 veri seti test011 imgesinin $\sigma=50$ için gürültü azaltma sonuçları

Renkli imgeler altındaki sonuçları görmek için ilk olarak Kodak24 veri setinde yer alan kodim06 imgesi için $\sigma=50$ olmak üzere görsel gürültü azaltma sonuçları Şekil 3.21’de yer almaktadır. BM3D yerine renkli imgeler için olan versiyonu CBM3D kullanılmıştır. CBM3D yöntemi en kötü sonucu vermiştir ki sırasıyla PSNR ve SSIM için 27.22dB ve 0.7532 sonuçları elde edilmiştir. IRCNN, FFDNet ve DnCNN ağları karşılaştırıldığında PSNR metriği için FFDNet daha iyi sonuç verirken SSIM metriği için IRCNN daha iyi sonuç vermiştir. Görsel olarak incelendiğinde ise FFDNet ve DnCNN özellikle deniz yüzeyinde yumuşak yüzeylere neden olurken IRCNN kısmen bunu koruyabilmektedir. ADNet ağı bakıldığında ise PSNR ve SSIM metriği altında 27.78 dB ve 0.7644 ile başarılı sonuçlar

verirken, MLFAN ağı PSNR için 27.78dB ve SSIM için 0.7698 ile en iyi sonuçları vermesinin yanı sıra görsel olarak detay bilgilerini daha iyi koruduğu söylenebilir.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 20.45 dB – 0.2776



(c) CBM3D / 27.22 dB - 0.7532



(d) DnCNN / 27.71 dB - 0.7576



(e) FFDNet / 27.73 dB – 0.7596



(f) IRCNN / 27.64 dB – 0.7618



(g) ADNet / 27.78 dB - 0.7644



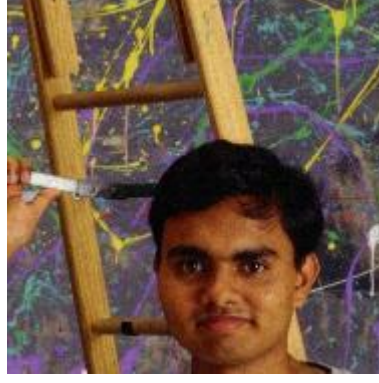
(h) MLFAN / 27.87 dB – 0.7698

Şekil 3.21. Kodak24 veri seti kodim06 imgesinin $\sigma=50$ için gürültü azaltma sonuçları

Renkli imgelere başka bir örnek olarak McMaster veri setinde bulunan kadim06 için $\sigma=15$ olmak üzere görsel sonuçları PSNR ve SSIM metrikleriyle Şekil 3.22’de verilmiştir.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 24.79 dB - 0.5502



(c) CBM3D / 34.27 dB - 0.9161



(d) DnCNN / 35.09 dB - 0.9264



(e) FFDNet / 34.81 dB - 0.9202



(f) IRCNN / 34.60 dB - 0.9165



(g) ADNet / 35.05 dB - 0.9262



(h) MLFAN / 35.28 dB - 0.9292

Şekil 3.22. McMaster veri seti kadim06 imgesinin $\sigma=15$ için gürültü azaltma sonuçları

McMaster veri setinde yer alan kadim06 imgesi $\sigma=15$ için gürültü azaltma sonuçları incelendiğinde PSNR ve SSIM metrikleri altında en iyi sonucu MLFAN ağı vermiştir. İkinci

en iyi sonuç olarak ise her iki metrik için odak mekanizması kullanan ADNet ağının verdiği görülmektedir.

Renkli imgeler için son bir örnek olarak CBSD68 veri seti içerisinde yer alan bir imge seçilmiştir. Gürültü seviyesi olarak $\sigma=25$ olarak belirlenmiş ve PSNR ve SSIM metrik sonuçları ile Şekil 3.23'te sunulmuştur. MLFAN ağı PSNR ve SSIM metrikleri altında daha iyi sonuç vermiştir.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 20.69 dB - 0.5341



(c) CBM3D / 30.84 dB - 0.9113



(d) DnCNN / 31.33 dB - 0.9213



(e) FFDNet / 31.22 dB - 0.9168



(f) IRCNN / 31.22 dB - 0.9168



(g) ADNet / 31.49 dB - 0.9227



(h) MLFAN / 31.60 dB - 0.9258

Şekil 3.23. CBSD68 veri seti test011 imgesinin $\sigma=25$ için gürültü azaltma sonuçları

Renkli görüntüler için Kodak24 kodim06 imgesinin diğer gürültü seviyelerinde ($\sigma=15$ ve $\sigma=25$ için) elde edilen görsel sonuçları Şekil 3.24 ve Şekil 3.25'te yer almaktadır.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 26.68 dB – 0.5543



(c) CBM3D / 33.30 dB – 0.9113



(d) DnCNN / 33.66 dB - 0.9208



(e) FFDNet / 33.62 dB – 0.9187



(f) IRCNN / 33.78 dB – 0.9116



(g) ADNet / 33.70 dB - 0.9219

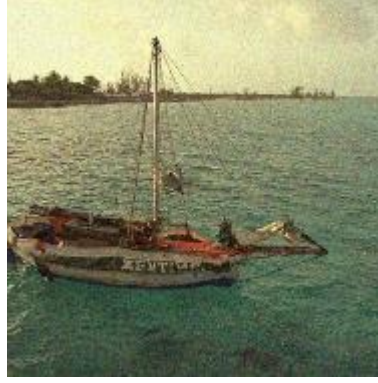


(h) MLFAN / 33.76 dB – 0.9222

Şekil 3.24. Kodak24 veri seti kodim06 imgesinin $\sigma=15$ için gürültü azaltma sonuçları



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 24.61 dB – 0.3810



(c) CBM3D / 30.51 dB – 0.8560



(d) DnCNN / 30.93 dB - 0.8682



(e) FFDNet / 30.92 dB – 0.8653



(f) IRCNN / 30.89 dB – 0.8661



(g) ADNet / 30.98 dB - 0.8706



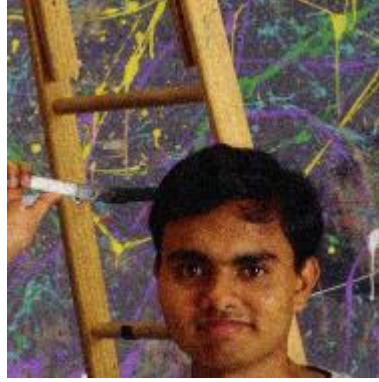
(h) MLFAN / 31.05 dB – 0.8723

Şekil 3.25. Kodak24 veri seti kodim06 imgesinin $\sigma=25$ için gürültü azaltma sonuçları

Renkli görüntüler için McMaster kodim06 imgesinin diğer gürültü seviyelerinde ($\sigma=25$ ve $\sigma=50$ için) elde edilen görsel sonuçları Şekil 3.26 ve Şekil 3.27’de yer almaktadır.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 20.46 dB - 0.3578



(c) CBM3D / 31.81 dB - 0.8722



(d) DnCNN / 32.51 dB - 0.8850



(e) FFDNet / 32.33 dB - 0.8806



(f) IRCNN / 32.06 dB - 0.8739



(g) ADNet / 32.52 dB - 0.8867

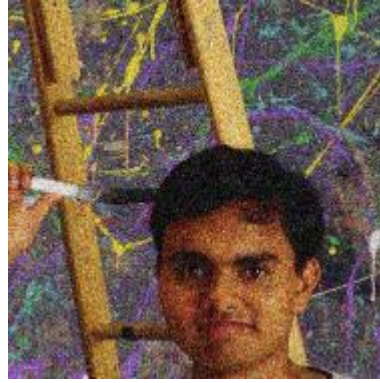


(h) MLFAN / 32.72 dB - 0.8898

Şekil 3.26. McMaster veri seti kadim06 imgesinin $\sigma=25$ için gürültü azaltma sonuçları



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 14.87 dB - 0.1622



(c) CBM3D / 28.40 dB - 0.7803



(d) DnCNN / 28.97 dB - 0.7938



(e) FFDNet / 28.82 dB - 0.7963



(f) IRCNN / 28.53 dB - 0.7854



(g) ADNet / 29.00 dB - 0.8000



(h) MLFAN / 29.16 dB - 0.8003

Şekil 3.27. McMaster veri seti kadim06 imgesinin $\sigma=50$ için gürültü azaltma sonuçları

Renkli görüntüler için CBSD68 test011 imgesinin diğer gürültü seviyelerinde ($\sigma=15$ ve $\sigma=50$ için) elde edilen görsel sonuçları Şekil 3.28 ve Şekil 3.29'da yer almaktadır.



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 24.91 dB - 0.6916



(c) CBM3D / 33.76 dB - 0.9477



(d) DnCNN / 34.25 dB - 0.9535



(e) FFDNet / 34.07 dB - 0.9512



(f) IRCNN / 34.07 dB - 0.9502



(g) ADNet / 34.30 dB - 0.9538



(h) MLFAN / 34.38 dB - 0.9548

Şekil 3.28. CBSD68 veri seti test011 imgesinin $\sigma=15$ için gürültü azaltma sonuçları



(a) Gerçek İmge / PSNR- SSIM



(b) Gürültülü / 15.23 dB - 0.3177



(c) CBM3D / 27.07 dB - 0.8240



(d) DnCNN / 27.80 dB - 0.8483



(e) FFDNet / 27.73 dB - 0.8453



(f) IRCNN / 27.54 dB - 0.8393



(g) ADNet / 27.85 dB - 0.8503



(h) MLFAN / 28.00 dB - 0.8568

Şekil 3.29. CBSD68 veri seti test011 imgesinin $\sigma=50$ için gürültü azaltma sonuçları

Renkli imgelerin görsel sonuçlarının yanı sıra sayısal sonuçlarına ait ortalama PSNR ve SSIM sonuçları Kodak24, McMaster, CBSD68 veri setlerinde Tablo 3.4'te verilmiştir. Burada karşılaştırma işlemi için CBM3D, DnCNN, FFDNet, IRCNN, ADNet ve DudeNet ağına ait sayısal sonuçlar kullanılmıştır. DudeNet ağının SSIM sonuçları yazarlar tarafından paylaşılmadığı için konulmamıştır. MLFAN ağı her iki metrik altında da en iyi sonuçları verdiği Tablo 3.4'te açıkça görülmektedir. İkinci en iyi sonucu CBSD68 ve Kodak24 veri setleri için DudeNet verirken, McMaster veri seti için ADNet ağı vermiştir.

Tablo 3.4. CBSD68, Kodak24 ve McMaster veri setinde farklı metotlar ile elde edilen ortalama PSNR ve SSIM değerleri

Veri seti	Yöntem	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$
CBSD68	CBM3D	33.52 / 0.9248	30.71 / 0.8716	27.38 / 0.7669
	DnCNN	33.98 / 0.9317	31.31 / 0.8863	28.01 / 0.7915
	FFDNet	33.80 / 0.9318	31.18 / 0.8857	27.96 / 0.7916
	IRCNN	33.86 / 0.9285	31.16 / 0.8824	27.86 / 0.7898
	ADNet	33.99 / 0.9334	31.31 / 0.8889	28.04 / 0.7977
	DudeNet	34.01 / Na	31.34 / Na	28.09 / Na
	MLFAN	34.06 / 0.9336	31.37 / 0.8894	28.11 / 0.7982
Kodak24	CBM3D	34.28 / 0.9164	31.68 / 0.8682	28.46 / 0.7751
	DnCNN	34.73 / 0.9212	32.23 / 0.8774	29.02 / 0.7895
	FFDNet	34.55 / 0.9230	32.11 / 0.8790	28.99 / 0.7929
	IRCNN	34.56 / 0.9198	32.03 / 0.8766	28.81 / 0.7929
	ADNet	34.76 / 0.9248	32.26 / 0.8830	29.10 / 0.7997
	DudeNet	34.81 / Na	32.26 / Na	29.10 / Na
	MLFAN	34.85 / 0.9257	32.33 / 0.8835	29.15 / 0.8000
McMaster	CBM3D	34.06 / 0.9150	31.66 / 0.8739	28.51 / 0.7934
	DnCNN	34.80 / 0.9070	32.47 / 0.8724	29.21 / 0.7985
	FFDNet	34.47 / 0.9247	32.25 / 0.8891	29.14 / 0.8157
	IRCNN	34.58 / 0.9195	32.18 / 0.8818	28.91 / 0.8069
	ADNet	34.93 / 0.9287	32.56 / 0.8944	29.36 / 0.8246
	DudeNet	34.83 / Na	32.40 / Na	29.12 / Na
	MLFAN	35.08 / 0.9288	32.68 / 0.8956	29.47 / 0.8263

3.3.4. Ablasyon Çalışması

Bu çalışmada gri seviye ve renkli imgelerde gürültü azaltma işlemi için yeni bir yöntem önerilmiştir. Bu amaçla ilk olarak öncül bilgi olması ve gürültü azaltma işlemi sırasında imgenin detay bilgilerini kaybederek özgün yapısından uzaklaşmasının önüne geçmek için YİÖ tabanlı doku imgesinin çıkarımı gerçekleştirilmiştir. Bu amaçla TENet isimli bir ağ önerilerek $\sigma = \{15,25,50\}$ olmak üzere 3 farklı seviyede gürültülü imgeden YİÖ imgesi elde edilmiş ve gürültü azaltma işlemi yapan MLFAN ağına girdi olarak verilmiştir. Gürültülü imgenin yanı sıra ilgili imgeye ait YİÖ tabanlı doku imgesini de girdi verisi olarak alan MLFAN ağı genel olarak 2 farklı modülden oluşacak şekilde tasarlanmıştır. Bu iki modül sırası ile farklı seviyelerde öznetelik çıkarımı için kullanılan MLF bloğu ve SENet yöntemi dikkate alınarak gerçekleştirilen kanal tabanlı odak mekanizmasıdır. Gürültü azaltma işlemi sürecine katkı sağlaması için önerilen bu yeniliklerin etkilerini ölçmek için kapsamlı bir ablasyon çalışması yürütülmüştür. Bu amaçla 5 farklı senaryo incelenmiş ve yeni 5 farklı tasarım içeren ağ yapısı önerilmiştir. Bunlar sırasıyla NMLFAN, FAN, CBMLFAN ve MLMLFAN, SMLFAN ağlarıdır. Bu ağların detayları sırasıyla şöyledir;

- CBMLFAN: Kullanılan kanal tabanlı odak mekanizmasının etkisini ölçmek için SENet yöntemi dışında uzamsal odak mekanizması ile kanal tabanlı odak mekanizmasını birleştiren CBAM odak mekanizması kullanılmıştır (Woo vd., 2018). MLFAN ağında sadece odak mekanizması değiştirilerek aynı parametreler altında eğitilmiş ve sonuçlar değerlendirilmiştir.
- NMLFAN: Gürültü azaltma işlemi sırasında öncül bilgisi olması için doku bilgisinin kullanıldığından bahsedilmiştir. Doku bilgisinin kullanımının MLFAN ağının gürültü azaltma performansına etkisini değerlendirmek için MLFAN ağında hiçbir değişiklik yapılmadan girdi verisi değiştirilmiş ve sadece gürültülü imgeyi girdi verisi olarak alacak şekilde aynı hiper parametreler altında tekrar eğitilmiştir. Bu yeni ağ NMLFAN olarak adlandırılmıştır.
- FAN: MLF bloğunun etkisi incelenmiş ve bunun için MLF bloğu içerisinde farklı ölçeklerdeki konvolüsyon katmanları çıkarılarak tek bir seviyede konvolüsyon katmanı eklenmiş ve sonuçlar değerlendirilmiştir.
- MLMLFAN: TENet ağının katkısını ölçebilmek için gürültü imgelerden YİÖ imgesini çıkarmak adına geleneksel YİÖ yönteminin gürültülü imgelerde etkili bir versiyonu olan Yerel Ortalama İkili Örüntü (YOİÖ) yöntemi kullanılmıştır

(Goyani vd., 2017). Kısaca, ENet ağı tamamıyla çıkarılarak YOİÖ yöntemi ile elde edilen doku imgesi MLFAN ağına girdi olarak verilmiştir.

- SMLFAN: Atlamalı bağlantı yapısını değerlendirebilmek için MLFAN ağı içerisinde kullanılan atlamalı bağlantı sayısı arttırılmış ve yeni bir tane eklenmiştir.

A. CBMLFAN Ağı

Bu çalışmada çok seviyeli elde edilen öznitelikler üzerine odak mekanizması kullanılarak farklı seviyelerdeki özniteliklerin ağırlıklandırma yoluyla ayırt edilerek başarımın arttırılması hedeflenmiştir. Bu etkinin başarısını ölçmek için ablasyon çalışması kapsamında, MLFAN ağına kullanılan ve kanal tabanlı bir öznitelik odak mekanizması olan SENet mekanizmasının başarımının test edilmesi istenmiştir. Bu amaçla literatürde önerilmiş olan ve çeşitli görüntü işleme problemleri için çokça tercih edilen CBAM(Convolutional Block Attention Module) odak mekanizması tercih edilmiştir (Woo vd., 2018). Şekil 3.10'da verilen MLFAN modelinde yer alan öznitelik odak mekanizmasında SENet çıkarılmış ve CBAM eklenmiştir. CBAM kanal bazlı özniteliklerin ağırlıklandırılmasının yanı sıra uzamsal olarak özniteliklerin ağırlıklandırılması işlemi de dikkate alan ikili bir öznitelik odak mekanizmasıdır. Bu amaçla CBAM odak mekanizması girdi öznitelikleri üzerine ilk olarak kanal tabanlı odak mekanizması uygulanır ve elde edilen çıktı öznitelikleri üzerine uzamsal tabanlı odak mekanizması uygulanarak ağırlıklandırılmış öznitelikler elde edilir. CBAM modülü için tasarlanan ve girdi ağırlıklarına ilk olarak uygulanan kanal odak mekanizmasının matematiksel gösterimi Eşitlik (3.13) ve Eşitlik (3.14)'de verilmiştir:

$$r_c = \alpha \left(d_{up} \left(\delta \left(d_{down}(f_{gap}(x)) \right) \right) + d_{up} \left(\delta \left(d_{down}(f_{gmp}(x)) \right) \right) \right) \quad (3.13)$$

$$f_c = x * r_c(x) \quad (3.14)$$

burada x girdi öznitelikleri, f_{gap} ve f_{gmp} sırasıyla küresel ortalama havuzlama ve küresel maksimum havuzlama işlemi, d_{down} ve d_{up} ise sırasıyla kanal sayısını azaltan yoğun katmanı ve arttıran yoğun katmanı ifade etmektedir. Buradaki oran yazarların makalede tavsiyeleri doğrultusunda 1/8 oranında seçilmiştir. δ ve α sırasıyla ReLu ve sigmoid aktivasyon fonksiyonunun ifade ederken r_c kanal bazlı odak mekanizması ile edilmiş

ağırlıkları ifade etmektedir. Son olarak ağırlıklandırılmış öznitelikleri elde etmek için Eşitlik (3.14)'de gösterildiği üzere girdi öznitelikleri x ile r_c elementsel olarak çarpılmaktadır. CBAM modülü için tasarlanan uzamsal odak mekanizmasının matematiksel gösterimi ise Eşitlik (3.15) ve (3.16)'daki gibidir:

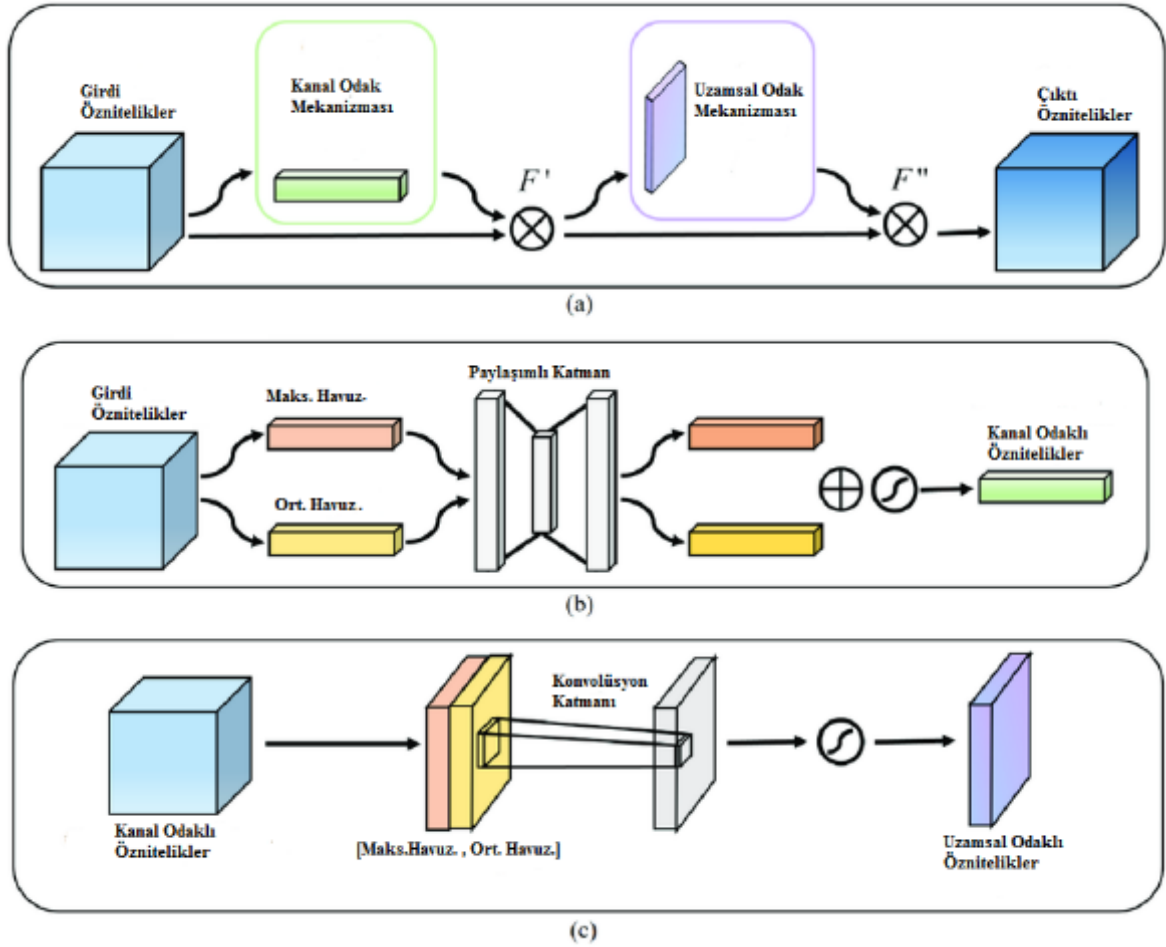
$$r_s = \alpha \left(C_7 \left(C_{at} (f_{cap}(x), f_{cmp}(x)) \right) \right) \quad (3.15)$$

$$f_s = x * r_s(x) \quad (3.16)$$

burada r_s uzamsal olarak uygulanan odak mekanizması ile elde edilen ağırlıkları, f_{cap} ve f_{cmp} sırasıyla kanal bazlı uygulanan ortalama ve maksimum pooling işlemini, C_{at} ard arda ekleme işlemi şeklinde gerçekleştirilen toplama işlemini, C_7 çekirdek boyutu 7×7 olan 1 adet konvolüsyon işlemini ve son olarak α ise sigmoid aktivasyon fonksiyonunun ifade etmektedir. Eşitlik (3.16)'da gösterildiği üzere uzamsal ağırlıkları ifade eden r_s ve x girdi öznitelikleri elementsel olarak çarpılmaktadır. Buna göre CBAM modülünün nihai matematiksel gösterimi Eşitlik (3.17)'deki gibidir.

$$CBAM(x) = f_s(f_c(x)) \quad (3.17)$$

Matematiksel gösterime ek olarak CBAM modülünün gösterimi Şekil 3.30'da verilmiştir. Buradan hareketle MLFAN ağı içerisinde yer alan kanal odak mekanizması olan SENet modülü çıkarılmış ve yerine CBAM modülü eklenmiştir. Odak mekanizması değiştirilerek tasarlanan yeni ağ CBMLFAN olarak isimlendirilmiştir. Bu yeni ağ MLFAN ağı ile aynı şekilde eğitilerek gri ve renkli seviye imgeler için $\sigma = \{15, 25, 50\}$ için sonuçlar alınmıştır. Eğitimi tamamlanan CBMLFAN ağı gri seviye için Set12 ve Set68 veri setleri ve renkli imgeler için ise Kodak24, McMaster ve CBSD68 için test edilmiştir. Değerlendirme metriği olarak ise yine PSNR ve SSIM metrikleri kullanılmıştır.



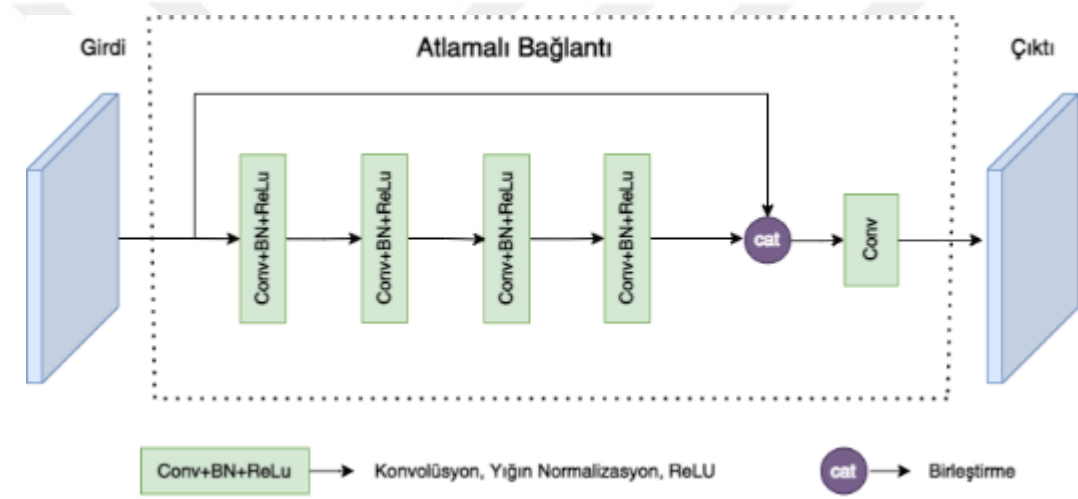
Şekil 3.30. CBAM odak mekanizması, (a) kanal odak mekanizması, (b) uzamsal odak mekanizması

B. NMLFAN Ağı

Bu çalışmada yapılan en önemli katkılardan bir tanesi gürültü azaltma işleminde performans artışı sağlaması için gürültülü görüntünün yanı sıra detay bilgileri içeren doku bilgisinin kullanılmasıdır. Burada amaç gürültü azaltma işlemi sırasında MLFAN ağının YİÖ doku bilgisi içeren imgeyi referans olarak kullanması ve görüntünün detaylı bilgilerinin korunarak özgün yapısında meydana gelebilecek kaybolmaların önüne geçmektir. Bu katkının ilgili çalışmaya katkısını ölçmek için gürültü azaltma işlemi sırasında TENet ağı kullanılarak elde edilen YİÖ imgesi MLFAN ağına girdi olarak verilmeyerek, eğitim işlemi sadece gürültü imge baz alınarak gerçekleştirilmiş ve bu yeni ağ NMLFAN olarak isimlendirilmiştir. Eğitimi tamamlanan NMLFAN ağı gri seviye için Set12 ve Set68 veri setleri ve renkli imgeler için ise Kodak24, McMaster ve CBSD68 için test edilmiştir. Değerlendirme metriği olarak ise yine PSNR ve SSIM metrikleri kullanılmıştır.

C. FAN Ağı

MLFAN ağı tasarımını yapılırken farklı seviyelerdeki özniteliklerin çıkarımı için farklı çekirdek boyutuna sahip konvolüsyonlar kullanılmış ve yerelden globale farklı öznitelikler elde edilmiştir. Burada amaç global bilgilerden faydalanmak ve MLFAN ağı başarımını arttırmaktır. Bu tasarımın gürültü azaltma işlemine katkısını ölçmek için MLFAN ağı içerisinde çok seviyeli öznitelik çıkarım (MLF) bloğunun tasarımında bir değişiklik yapılmış ve farklı çekirdek boyutuna sahip genişletilmiş konvolüsyonlar çıkarılmış ve yerine bir adet 64 kanallı 3x3 boyutunda konvolüsyon, yığın normalizasyonu ve ReLU aktivasyon fonksiyonu eklenmiştir. Buna göre MLF bloğunun yeni tasarımı Şekil 3.31'deki gibidir.



Şekil 3.31. Yeni MLF bloğu

MLF bloğunun tasarımını değiştirdikten ve çok seviyeli özniteliklerin çıkarım işlemi yerine tek seviye konvolüsyon işlemi eklendikten sonra MLFAN ağı diğer tasarım aşamaları ve eğitim aşamasında tercih edilen diğer bütün parametreler sabit tutularak gri ve renkli imgeler için $\sigma = \{15,25,50\}$ gürültü seviyeleri altında eğitilmiş ve bu yeni ağ FAN olarak isimlendirilmiştir. Eğitimi tamamlanan FAN ağı gri seviye için Set12 ve Set68 veri setleri ve renkli imgeler için ise Kodak24, McMaster ve CBSD68 için test edilmiştir. Değerlendirme metriği olarak ise yine PSNR ve SSIM metrikleri kullanılmıştır.

D. MLMLFAN

MLFAN ağı ile gürültü azaltma işlemi yapılırken gürültülü görüntünün özgün yapısını korumak, detay bilgilerini muhafaza ederek gürültü azaltma işleminin performansını arttırmak için TENet kullanılarak YİÖ tabanlı doku çıkarılmış ve gürültülü imgeye ek olarak girdi verisi olarak sürece eklenmişti. TENet ağının başarısını değerlendirmek ve gürültülü imgeden TENet kullanılarak elde edilen gürültülü imgeden elde edilen YİÖ imgesinin başarımını ortaya koymak bir MLMLFAN ağı önerilmiştir. Bu ağda gürültülü imgeye ek olarak YİÖ imgesi de girdi olarak alınmıştır. Fakat, MLFAN ağından farklı olarak gürültülü imgeden YİÖ imgesinin çıkarımı için TENet gibi bir CNN ağı kullanmak yerine geleneksel YİÖ çıkarma işleminin gürültüye dayanaklı bir versiyonu olan Yerel Ortalama İkili Örüntü (YOİÖ) yöntemi kullanılmıştır (Goyani vd., 2017). Bu sayede CNN ağı ile yapılan YİÖ doku imgesinin çıkarımının başarısı test edilmiştir.

Yerel Ortalama İkili Örüntü (YOİÖ) yöntemi Goyani ve Patel tarafından önerilmiş yeni bir YİÖ yöntemidir (Goyani vd., 2017). Goyani ve Patel bu çalışmalarında geleneksel YİÖ yönteminin gürültüye ve aydınlatma bozulmalarına karşı tolere olan bir türünü önermişlerdir. Hatırlanacağı üzere geleneksel YİÖ yönteminde amaç merkez pikselin belirli bir komşuluktaki pikseller ile sayısal değerlerinin karşılaştırılarak büyük ve eşit olan piksellere 1 atanması diğerlerine 0 atanması ve saat yönünde bu sayıların ikili tabandan onluk tabana dönüşümünün yapılması şeklindedir. YOİÖ yönteminde ise komşu pikseller ile kıyaslama işlemi için merkez piksel yerine merkez piksel ve komşu piksellerin ortalamasını alınmaktadır. Buna göre YOİÖ yöntemi Eşitlik (3.18), Eşitlik (3.19) ve Eşitlik (3.20)'deki gibi verilebilir:

$$YOİÖ(x_c, y_c) = \sum_{m=0}^7 2^m * f(i_m - i_\mu) \quad (3.18)$$

$$f(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (3.19)$$

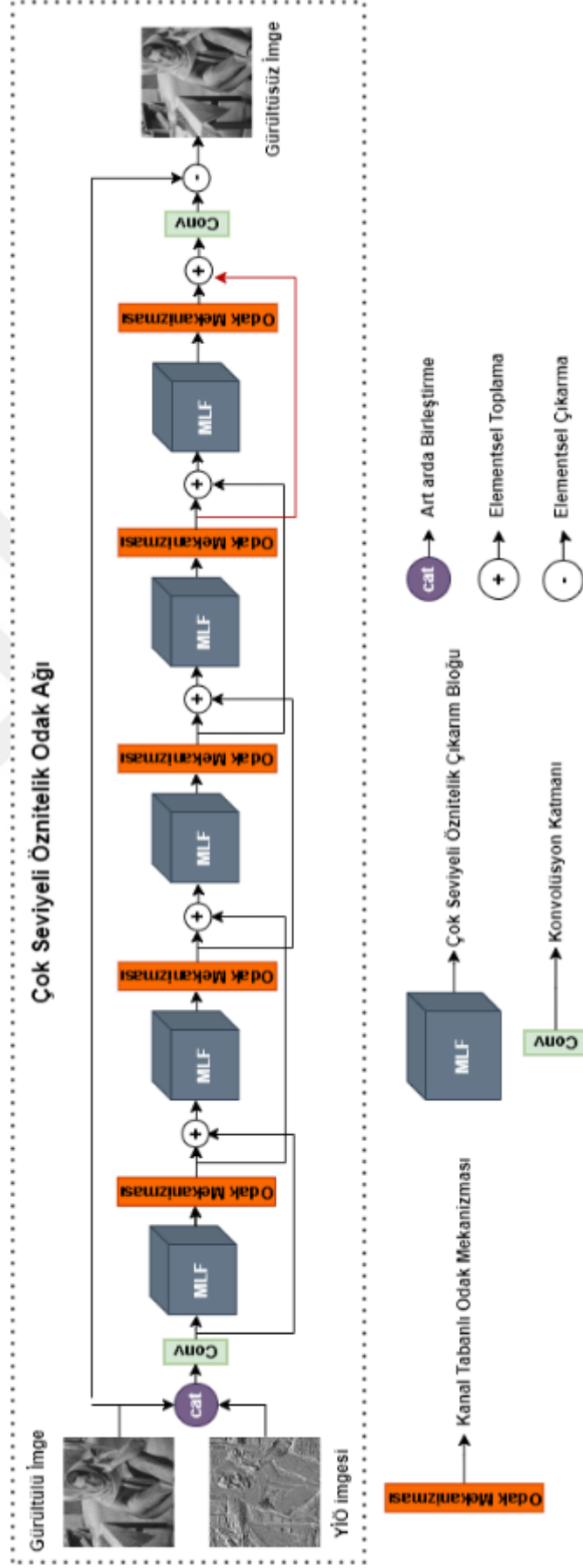
$$i_\mu = \frac{1}{9} \sum_{i=1}^3 \sum_{j=1}^3 I_{i,j} \quad (3.20)$$

burada $f(x)$ komşu pikseller ile kıyaslama işlemi ifade eden karşılaştırma fonksiyonunu, i_μ komşu piksel ve merkez pikselin ortalamasını, i_m m. komşu pikseli ifade etmektedir. YOİÖ yöntemi (Goyani vd., 2017) çalışmasında Gauss gürültüsünün yanı sıra farklı gürültü

türleri için de etkili bir yöntem olduğu göstermiştir ve çeşitli veri setleri üzerinde test etmişlerdir.

E. SMLFAN

Atlamalı bağlantı yapıları CNN ağlarında, ilk katmanlarda elde edilen ve faydalı olabilecek özniteliklerin ileriki katmanları iletilmesi amacıyla literatürde sıklıkla kullanılmaktadır. Bu sayede CNN ağlarında derin katmanlarda farklı seviyede elde edilen özniteliklerin birleşimine fayda sağlamakta olan atlamalı bağlantı yapıları probleme ve tasarıma özgü olarak elde edilen sayısal sonuçları büyük oranda etkilemektedir. Bu amaçla, MLFAN ağında kullanılan atlamalı bağlantı sayısı değiştirilmiş ve bu amaçla son MLF bloğundan sonra yeni bir tane eklenmiştir. Tasarımda yapılan değişiklik Şekil 3.32'de kırmızı ok ile gösterilmiştir. Bu yeni ağ SMLFAN olarak isimlendirilmiş ve diğer bütün bütün parametreler sabit tutularak aynı şekilde eğitilmiştir.



F. Ablasyon Sonuçları

Ablasyon çalışmaları kapsamında CBMLFAN, NMLFAN, FAN ve MLMLFAN isimli 4 farklı ağın tasarımı ve eğitim aşaması tamamlanmış olup gri ve renkli görüntülü imgeler için test edilmiştir. Elde edilen sonuçlar, ilgili çalışmada önerilen MLFAN ağı ile karşılaştırılarak, ağın tasarımında yapılan değişimlerin gürültü azaltma işleminin sonucuna etkisi değerlendirilmiş ve MLFAN ağının tasarımında sunulan yeniliklerin etkileri tartışılmıştır.

İlk olarak MLFAN ağında yer alan SENet odak mekanizması çıkarılarak yerine yeni bir odak mekanizması olan CBAM modülü kullanılarak eğitimi yapılan CBMLFAN ağının sonuçlarını inceleyelim. Buna göre CBMLFAN ağı Set12 veri seti için $\sigma = \{15,25,50\}$ gürültü seviyeleri altında PSNR metriğinde sırasıyla $\{33.06, 30.65, 27.39\}$ dB sonuçlarını verirken SSIM metriği altında da sırasıyla $\{0.9081, 0.8691, 0.7870\}$ sonuçlarını vermiştir. Aynı şekilde Set68 veri seti için bakıldığında ise $\sigma = \{15,25,50\}$ için PSNR metriği altında sırasıyla $\{31.77, 29.25, 26.29\}$ dB sonuçlarını ve SSIM metriğinde $\{0.8568, 0.8368, 0.7240\}$ dB sonuçlarını vermiştir. Buna göre önerilen yöntem MLFAN ağı ile kıyaslandığında Set12 için $\sigma = \{15,25,50\}$ seviyelerinde sırasıyla $\{0.02, 0.10, 0.17\}$ dB bir düşüş görülürken Set68 için $\{0.03, 0.09, 0.09\}$ dB bir düşüş görülmektedir. Aynı şekilde SSIM metriği için Set12 veri setinde $\sigma = \{15,25,50\}$ seviyelerinde sırasıyla $\{0.0006, 0.0011, 0.0069\}$ bir düşüş görülürken Set68 için $\{0.0005, 0.0003, 0.0031\}$ bir düşüş görülmektedir. Renkli imgeler için ise PSNR metriğinde CBSD68, Kodak24, McMaster veri setlerinde $\sigma = \{15,25,50\}$ için sırasıyla $\{33.97, 31.32, 27.95\}$, $\{34.76, 32.26, 29.00\}$ ve $\{34.91, 32.63, 29.28\}$ dB sonuçlarını verirken, SSIM metriğinde $\{0.9323, 0.8881, 0.7908\}$, $\{0.9242, 0.8815, 0.7935\}$ ve $\{0.9261, 0.8932, 0.8153\}$ sonuçlarını vermiştir. Buna göre PSNR metriği CBSD68, Kodak24, McMaster veri setleri ve $\sigma = \{15,25,50\}$ gürültü seviyelerinde PSNR metriğinde $\{0.09, 0.05, 0.16\}$ ve $\{0.09, 0.07, 0.15\}$, $\{0.17, 0.05, 0.19\}$ dB bir düşüş görülürken SSIM için $\{0.0013, 0.0013, 0.0074\}$, $\{0.0015, 0.0020, 0.0065\}$ ve $\{0.0027, 0.0024, 0.0110\}$ bir düşüş görülmüştür. Anlaşılacağı üzere, CBAM modülü SENet modülü ile kıyaslandığında özellikle yüksek seviye gürültüler için daha düşük sonuçlar vermektedir. Bunun nedeni olarak ise gürültünün özellikle uzamsal bilgileri bozması olduğu düşünülmektedir. Bilindiği üzere gürültüler çevre pikseller arasındaki ilişkileri bozmaktadır. Çünkü Şekil 3.30'da da gösterildiği üzere CBAM modülü girdi öznitelikleri üzerine ilk olarak kanal bazlı odak mekanizması uygulamakta ve elde edilen ağırlıklandırılmış bu öznitelikler daha sonra uzamsal odak mekanizmasına tabi tutulmaktadır. Uzamsal olarak özniteliklerin

ağırlıklandırılması gerçekleşirken gürültünün varlığının bu ağırlıklandırma işlemi etkilediği düşünülmektedir ve bu nedenle CBAM modülünün SENet modülünde daha başarısız sonuçlar vermiştir.

İkinci olarak ise doku bilgisinin gürültü azaltma işlemi sırasında görüntünün özgün yapısının korunması üzerindeki etkisini değerlendirmek için MLFAN ağı sadece gürültü imgenin girdi verisi olarak alındığı bir şekilde eğitilmiştir ve bu ağ NMLFAN olarak isimlendirilmiştir. Buna göre NMLFAN ağı Set12 için $\sigma = \{15,25,50\}$ gürültü seviyeleri için PSNR metriği altında sırasıyla $\{32.97, 30.62, 27.41\}$ dB sonuçlarını verirken SSIM metriği altında da sırasıyla $\{0.9061, 0.8661, 0.7868\}$ sonuçlarını vermiştir. Aynı şekilde Set68 veri seti için bakıldığında ise $\sigma = \{15,25,50\}$ için sırasıyla PSNR metriği altında $\{31.73, 29.24, 26.27\}$ dB ve SSIM metriği altında $\{0.8947, 0.8320, 0.7203\}$ sonuçlarını elde edilmiştir. Buradan elde edilen sonuçlar altında PSNR metriği için MLFAN ağı ile karşılaştırıldığında $\sigma = \{15,25,50\}$ gürültü seviyelerinde Set12'de $\{0.11, 0.13, 0.15\}$ dB ve Set68 için $\{0.07, 0.10, 0.11\}$ dB değerinde bir performans düşüşü görülmüştür. Aynı şekilde SSIM metriği için Set12'de $\sigma = \{15,25,50\}$ seviyelerinde sırasıyla $\{0.0026, 0.0041, 0.0071\}$ bir düşüş görülürken Set68 için $\{0.0026, 0.0051, 0.0068\}$ bir düşüş görülmektedir. Renkli imgeler için konuşacak olursak, PSNR metriği altında CBSD68, Kodak24, McMaster veri setleri için $\sigma = \{15,25,50\}$ seviyelerinde $\{33.95, 31.22, 27.74\}$, $\{34.69, 32.15, 28.72\}$ ve $\{34.95, 32.53, 29.12\}$ dB sonuçlarını verirken, SSIM metriğinde $\{0.9314, 0.8842, 0.7901\}$, $\{0.9214, 0.8774, 0.7907\}$ ve $\{0.9252, 0.8914, 0.8042\}$ sonuçlarını vermiştir. Buna göre PSNR metriği CBSD68, Kodak24, McMaster veri setleri ve $\sigma = \{15,25,50\}$ gürültü seviyelerinde PSNR metriği için $\{0.11, 0.15, 0.37\}$ ve $\{0.16, 0.18, 0.43\}$, $\{0.13, 0.15, 0.35\}$ dB bir düşüş görülürken SSIM için $\{0.0022, 0.0052, 0.0081\}$, $\{0.0043, 0.0061, 0.0093\}$ ve $\{0.0036, 0.0042, 0.0221\}$ bir düşüş görülmüştür. Anlaşılacağı üzere, doku bilgisinin gürültü görüntüye ek olarak bir girdi olarak alınması gürültü azaltma işleminin sonucunu büyük oranda etkilemektedir. MLFAN ağı ile karşılaştırıldığında hem gri hem renkli görüntüler için yüksek seviye gürültülerde, doku bilgisinin başarıma etkisi çok daha fazla olmuş ve bu bilginin bir rehber olarak kullanılmasıyla PSNR ve SSIM metrikleri altında elde edilen sonuçlar daha da iyileşmiştir.

Üçüncü olarak ise çok seviyeli öznetelik çıkarımının sonuca etkisini değerlendirmek için MLFAN ağının tasarımında yer alan farklı çekirdek boyutundaki konvolüsyon katmanları çıkarılarak tek bir boyutta konvolüsyon katmanı eklenerek gürültü azaltma işlemi yapılmıştır ve bu yeni ağ FAN olarak isimlendirilerek, MLFAN ağı ile aynı şekilde

eğitilmişti. Buna göre FAN ağı Set12 için $\sigma = \{15,25,50\}$ gürültü seviyeleri için PSNR metriği altında sırasıyla $\{33.02, 30.68, 27.49\}$ dB sonuçlarını verirken SSIM metriği altında da sırasıyla $\{0.9077, 0.8689, 0.7909\}$ sonuçlarını vermiştir. Aynı şekilde Set68 veri seti için bakıldığında ise $\sigma = \{15,25,50\}$ için sırasıyla PSNR metriği altında $\{31.75, 29.30, 26.34\}$ dB ve SSIM metriği altında $\{0.8962, 0.8362, 0.7246\}$ sonuçlarını vermiştir. Buradan elde edilen sonuçlar altında PSNR için MLFAN ağı ile karşılaştırıldığında $\sigma = \{15,25,50\}$ gürültü seviyelerinde Set12 için sırasıyla $\{0.06, 0.07, 0.07\}$ dB ve Set68 için $\{0.05, 0.04, 0.04\}$ dB değerinde bir performans düşüşü görülmüştür. Aynı şekilde SSIM metriği için Set12 veri setinde $\sigma = \{15,25,50\}$ seviyelerinde sırasıyla $\{0.0010, 0.0013, 0.0030\}$ bir düşüş görülürken Set68 için $\{0.0011, 0.0009, 0.0025\}$ bir düşüş görülmektedir. Renkli imgeler için konuşacak olursak, PSNR metriği altında CBSD68, Kodak24, McMaster veri setleri için $\sigma = \{15,25,50\}$ seviyelerinde $\{33.74, 31.19, 28.01\}$, $\{34.76, 32.20, 29.05\}$ ve $\{34.95, 32.41, 29.25\}$ dB sonuçlarını, SSIM metriğinde ise $\{0.9255, 0.8846, 0.7971\}$, $\{0.9240, 0.8778, 0.7996\}$ ve $\{0.9266, 0.8888, 0.8176\}$ sonuçlarını vermiştir. Buna göre PSNR metriği CBSD68, Kodak24, McMaster veri setleri ve $\sigma = \{15,25,50\}$ gürültü seviyelerinde PSNR metriği için sırasıyla $\{0.32, 0.18, 0.10\}$ $\{0.09, 0.13, 0.10\}$ ve $\{0.13, 0.27, 0.22\}$ dB bir düşüş görülürken SSIM metriği altında bakıldığında $\{0.0081, 0.0048, 0.0009\}$, $\{0.0017, 0.0057, 0.0004\}$ ve $\{0.0022, 0.0068, 0.0087\}$ bir düşüş görülmüştür. Anlaşılacağı üzere, çok seviyeli öznitelik çıkarımının çeşitli gürültü seviyelerinde MLFAN ağına büyük katkı sağlamıştır. Özellikle renkli imgeler için PSNR ve SSIM metriği altındaki en büyük kayıp düşük seviye gürültülü veri setlerinde görülmüştür. Fakat gri seviye veri setlerinde SSIM metriği altında en büyük kayıp ise yüksek gürültü seviyelerinden görülmüştür. Bu da çok seviyeli öznitelik çıkarımının gürültü azaltma işlemine katkı sağlamasının yanı sıra, gürültü seviyesi ve imgenin türüne bağlı olarak global çıkarılacak özniteliklerin seçiminin odak mekanizması ile ağırlıklandırma işlemine tabi tutulması fikrini desteklemektedir.

Dördüncü olarak ise TENet ağının başarısını değerlendirmek için YİÖ yönteminin gürültüye tolere bir versiyonu olarak YOİÖ yöntemi ile gürültülü imgeden doku bilgisi çıkarılarak MLFAN ağına girdi olarak verilmiş ve bu sayede TENet ağı tamamıyla kapsam dışı bırakılmıştır. Bu yeni ağ MLMLFAN olarak isimlendirilerek, MLFAN ağı ile aynı şekilde eğitilmişti. Buna göre FAN ağı Set12 için $\sigma = \{15,25,50\}$ gürültü seviyeleri için PSNR metriği altında sırasıyla $\{32.92, 30.57, 27.23\}$ dB sonuçlarını verirken SSIM metriği altında da sırasıyla $\{0.9053, 0.8643, 0.7791\}$ sonuçlarını vermiştir. Aynı şekilde Set68 veri seti için bakıldığında ise $\sigma = \{15,25,50\}$ için sırasıyla PSNR metriği altında $\{31.69, 29.21,$

26.16} dB ve SSIM metriği altında {0.8945, 0.8298, 0.7150} sonuçlarını vermiştir. Buradan elde edilen sonuçlar altında PSNR için MLFAN ağı ile karşılaştırıldığında $\sigma = \{15,25,50\}$ gürültü seviyelerinde Set12 için sırasıyla {0.16, 0.18, 0.33} dB ve Set68 için {0.11, 0.13, 0.22} dB değerinde bir performans düşüşü görülmüştür. Aynı şekilde SSIM metriği altında Set12'de $\sigma = \{15,25,50\}$ seviyelerinde sırasıyla {0.0034, 0.0059, 0.0148} bir düşüş görülürken Set68 için {0.0028, 0.0073, 0.0121} bir düşüş görülmektedir. Renkli imgeler için konuşacak olursak, PSNR metriği altında CBSD68, Kodak24, McMaster veri setleri için $\sigma = \{15,25,50\}$ seviyelerinde {33.91, 31.12, 27.60}, {34.64, 32.02, 28.60} ve {34.90, 32.38, 28.99} dB sonuçlarını, SSIM metriğinde sırasıyla {0.9301, 0.8810, 0.7630}, {0.9198, 0.8742, 0.7646} ve {0.9242, 0.8887, 0.7995} sonuçlarını vermiştir. Buna göre PSNR metriği CBSD68, Kodak24, McMaster veri setleri ve $\sigma = \{15,25,50\}$ gürültü seviyelerinde PSNR metriği için sırasıyla {0.15, 0.25, 0.51}, {0.21, 0.31, 0.55} ve {0.18, 0.30, 0.48} dB bir düşüş görülürken SSIM metriği altında bakıldığında {0.0035, 0.0084, 0.0352}, {0.0059, 0.0093, 0.0354} ve {0.0046, 0.0069, 0.0268} bir düşüş görülmüştür. Bu sonuçlardan da anlaşılacağı üzere TENet ağı dışında YİÖ yönteminin gürültüye tolere bir versiyonu tercih edildiğinde özellikle yüksek seviyeli gürültülerde büyük düşüşler meydana gelmiştir. Öyle ki, NMLFAN ağı ile kıyaslandığında elde edilen sonuçlar daha kötüleşmiştir. Bu da etkisiz bir doku çıkarım yönteminin gürültü azaltma işlemi sırasında elde edilecek sonuçları daha da aşağı çektiğini, CNN ağının doku bilgisini dikkate aldığını ve TENet yöntemi ile doku çıkarım işleminin başarımını göstermiştir.

Son olarak ise MLFAN ağı içerisinde kullanılan atlamalı bağlantı yapısının katkısını incelemek için ağ tasarımında Şekil 3.32'de gösterildiği üzere bir değişiklik yapılarak son MLF bloğuna bir tane daha atlamalı bağlantı eklenmiştir. Bu yeni ağ SMLFAN olarak isimlendirilerek, MLFAN ağı ile aynı şekilde eğitilmiştir. Buna göre FAN ağı Set12 için $\sigma = \{15,25,50\}$ gürültü seviyeleri için PSNR metriği altında sırasıyla {33.10, 30.71, 27.59} dB sonuçlarını verirken SSIM metriği altında da sırasıyla {0.9086, 0.8687, 0.7949} sonuçlarını vermiştir. Aynı şekilde Set68 veri seti için bakıldığında ise $\sigma = \{15,25,50\}$ için sırasıyla PSNR metriği altında {31.80, 29.32, 26.39} dB ve SSIM için {0.8970, 0.8356, 0.7281} sonuçlarını vermiştir. Buradan elde edilen sonuçlar altında PSNR için MLFAN ağı ile karşılaştırıldığında $\sigma = \{15,25,50\}$ gürültü seviyelerinde Set12 veri seti için sırasıyla {+0.02, -0.04, +0.03} dB ve Set68 için {0.00, -0.02, +0.01} dB değerinde bir performans değişimi görülmüştür. Aynı şekilde SSIM metriği altında Set12'de $\sigma = \{15,25,50\}$ seviyelerinde sırasıyla {-0.0001, -0.0005, +0.0010} bir değişim görülürken Set68 veri

setinde $\{-0.0003, -0.0004, +0.0010\}$ bir deęişim görölmüştür. Renkli imgeler için konuşacak olursak, PSNR metrięi altında CBSD68, Kodak24, McMaster veri setleri için $\sigma = \{15,25,50\}$ seviyelerinde $\{34.03, 31.34, 28.13\}$, $\{34.85, 32.31, 29.23\}$ ve $\{35.10, 32.65, 29.54\}$ dB sonuçlarını, SSIM metrięinde sırasıyla $\{0.9334, 0.8884, 0.7985\}$, $\{0.9251, 0.8819, 0.8012\}$ ve $\{0.9280, 0.8940, 0.8298\}$ sonuçlarını vermiştir. Buna göre PSNR metrięi CBSD68, Kodak24, McMaster veri setleri ve $\sigma = \{15,25,50\}$ gürültü seviyelerinde PSNR metrięi için sırasıyla $\{-0.03, -0.03, +0.02\}$, $\{0.00, -0.02, +0.08\}$ ve $\{+0.02, -0.03, +0.07\}$ dB bir deęişim görölürken SSIM metrięinde $\{-0.0002, -0.0010, +0.0003\}$, $\{-0.0006, -0.0016, +0.0012\}$ ve $\{-0.0008, -0.0016, +0.0035\}$ bir deęişim görölmüştür. Bu sonuçlardan da anlaşılacağı üzere SMLFAN aęında yeni bir atlamalı baęlantı kullanımı düşük seviye görüntülerde gözle görölür bir deęişime neden olmazken özellikle $\sigma = 50$ gibi yüksek bir gürültü seviyesinde gri ve renkli imgelerdeki bütün veri setlerinde PSNR ve SSIM metrikleri altında performans artışı sağlamıştır. Bu da yüksek gürültülerde önceki katmanlarda seyrelme ihtimali olan faydalı özniteliklerin ileriki katmanlara aktarımının CNN aęlarında öğrenmeye katkı sağladığı görüşünü desteklemektedir.

Renkli görüntü veri setleri CBSD68, Kodak24, McMaster için $\sigma = \{15,25,50\}$ gürültü seviyeleri altında CBMLFAN, NMLFAN, FAN, MLMLFAN, MLFAN aęları için elde edilen ortalama SSIM ve PSNR sonuçları Tablo 3.5'te verilmiştir.

Tablo 3.5. PSNR ve SSIM metrikleri altında renkli imgeler için ablasyon sonuçları

Veri seti	Yöntem	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$
CBSD68	CBMLFAN	33.97 / 0.9323	31.32 / 0.8881	27.95 / 0.7908
	NMLFAN	33.95 / 0.9314	31.22 / 0.8842	27.74 / 0.7901
	FAN	33.74 / 0.9255	31.19 / 0.8846	28.01 / 0.7971
	MLMLFAN	33.91 / 0.9301	31.12 / 0.8810	27.60 / 0.7630
	SMLFAN	34.03 / 0.9334	31.34 / 0.8884	28.13 / 0.7985
	MLFAN	34.06 / 0.9336	31.37 / 0.8894	28.11 / 0.7982
Kodak24	CBMLFAN	34.76 / 0.9242	32.26 / 0.8815	29.00 / 0.7935
	NMLFAN	34.69 / 0.9214	32.15 / 0.8774	28.72 / 0.7907
	FAN	34.76 / 0.9240	32.20 / 0.8778	29.05 / 0.7996
	MLMLFAN	34.64 / 0.9198	32.02 / 0.8742	28.60 / 0.7646
	SMLFAN	34.85 / 0.9251	32.31 / 0.8819	29.23 / 0.8012
	MLFAN	34.85 / 0.9257	32.33 / 0.8835	29.15 / 0.8000
McMaster	CBMLFAN	34.91 / 0.9261	32.63 / 0.8932	29.28 / 0.8153
	NMLFAN	34.95 / 0.9252	32.53 / 0.8914	29.12 / 0.8042
	FAN	34.95 / 0.9266	32.41 / 0.8888	29.25 / 0.8176
	MLMLFAN	34.90 / 0.9242	32.38 / 0.8887	28.99 / 0.7995
	SMLFAN	35.10 / 0.9280	32.65 / 0.8940	29.54 / 0.8298
	MLFAN	35.08 / 0.9288	32.68 / 0.8956	29.47 / 0.8263

Gri seviye veri setleri Set12 ve Set68 için $\sigma = \{15,25,50\}$ gürültü seviyeleri altında CBMLFAN, NMLFAN, FAN, MLMLFAN, MLFAN ağları için elde edilen ortalama SSIM ve PSNR sonuçları Tablo 3.6’da verilmiştir.

Tablo 3.6. PSNR ve SSIM metrikleri altında gri seviye imgeler için ablasyon sonuçları

Veri seti	Yöntem	$\sigma = 15$	$\sigma = 25$	$\sigma = 50$
Set12	CBMLFAN	33.06 / 0.9081	30.65 / 0.8691	27.39 / 0.7870
	NMLFAN	32.97 / 0.9061	30.62 / 0.8661	27.41 / 0.7868
	FAN	33.02 / 0.9077	30.68 / 0.8689	27.49 / 0.7909
	MLMLFAN	32.92 / 0.9053	30.57 / 0.8643	27.23 / 0.7791
	SMLFAN	33.10 / 0.9086	30.71 / 0.8687	27.59 / 0.7949
	MLFAN	33.08 / 0.9087	30.75 / 0.8702	27.56 / 0.7939
Set68	CBMLFAN	31.77 / 0.8968	29.25 / 0.8368	26.29 / 0.7240
	NMLFAN	31.73 / 0.8947	29.24 / 0.8320	26.27 / 0.7203
	FAN	31.75 / 0.8962	29.30 / 0.8362	26.34 / 0.7246
	MLMLFAN	31.69 / 0.8945	29.21 / 0.8298	26.16 / 0.7150
	SMLFAN	31.80 / 0.8970	29.32 / 0.8356	26.39 / 0.7281
	MLFAN	31.80 / 0.8973	29.34 / 0.8371	26.38 / 0.7271

4. ÖNERİLER VE TARTIŞMA

Tez çalışmasında, odak mekanizması, çok seviyeli öznitelik çıkarımı ve doku bilgisinin kullanımı ile tasarlanmış bir CNN ağı ile gri seviye ve renkli imgelerde gauss gürültüsünün giderimi amaçlanmıştır. Elde edilen sonuçlar PSNR ve SSIM metriği altında sayısal olarak incelendiğinde literatürde karşılaştırılan yöntemlere kıyasla daha iyi sonuçlar vermiş ve görsel örnekler sunularak da bu başarı desteklenmiştir. Buna ek olarak, tasarlanan bu yeni CNN ağında öne sürülen yenilikçi fikirlerin, yönteme katkısını değerlendirmek için geniş kapsamlı ablasyon çalışmaları yürütülmüştür.

Bu kapsamda kullanılan odak mekanizmasının, çok seviyeli öznitelik çıkarımının ve doku bilgisinin katkıları ayrı ayrı değerlendirilmiştir. Bu bağlamda elde edilen sonuçlar gri seviye ve renkli imgeler için yine PSNR ve SSIM metrikleri altında incelendiğinde yenilikçi fikirlerin önerilen MLFAN ağına katkısı görülmüştür. Bunlar arasında özellikle doku bilgisinin, gürültü azaltma işlemi sırasında MLFAN ağına bir rehber bilgi sunduğu ve gürültü azaltma sırasında imgenin özgün yapısını korumaya katkı sağladığı anlaşılmıştır. En çok katkıyı ise yüksek bir gürültü seviyesi olan $\sigma = 50$ için vermiştir. Daha sonra ise en büyük katkı çok seviyeli öznitelik çıkarımında görülmüştür. Özellikle düşük seviyeli görüntülerde global bilgilerin çıkarımının CNN ağına katkısının çok fazla olduğu görülmüştür. Ancak yüksek seviyede gürültüde bu katkının azaldığı, doku bilgisinin katkısının arttığı görülmüştür. Bunun nedeni yüksek seviye gürültü altında global bilgilerin de bozulmaya uğraması olduğu muhtemeldir.

Doku bilgisinin çıkarımı işlemi sırasında katkının büyük olması, gürültülü imgelerden doku çıkarımı işleminin çeşitlendirilmesinin gürültü azaltma işlemine katkı sağlayacağını fikrini güçlendirmektedir. Bu nedenle basit ve hesaplaması kolay olan YİÖ tabanlı doku çıkarma işleminin yanı sıra farklı doku bilgisi çıkarma yöntemlerinin dikkate alınması ileriki çalışmalar için düşünülmektedir. Bunun yanı sıra CNN ağı modelinin tasarımının iyileştirilmesi, farklı kayıp fonksiyonları ve çeşitli hiper parametre ayarlarının yapılması ile önerilen TENet ağının güçlendirilmesi de ikincil bir çalışma alanı olarak ele alınabilir. Doku çıkarımı alanında son olarak ise yapılması planlanan bir çalışma gürültülü imgeden doku bilgisi çıkarımını farklı bir bakış açısı yani görüntü-görüntü dönüştürümü şeklinde ele alıp CNN ağı yerine çekişmeli-üretici ağlar (GAN) kullanılmasıdır. GAN yapısı gereği çok güçlü bir ağ yapısı olmasından dolayı literatüre bakıldığında görüntü dönüşümü için çokça tercih edilmektedir (Creswell vd., 2018).

Tez çalışması kapsamında farklı çekirdek boyutunda konvolüsyon katmanları ile tasarlanan MLF bloğunun kullanımı ile farklı öznitelik çıkarımların elde edildiği gösterilmiştir ve gürültü azaltma işlemine katkı sağladığı yapılan ablasyon çalışmaları ile de ispatlanmıştır. Ancak farklı çekirdek boyutunda konvolüsyon katmanları farklı öznitelik çıkarabildiği gibi, farklı MLF blok tasarımları da aynı sonucu vermektedir. Bu nedenle tek bir MLF bloğu tabanlı CNN ağı tasarımından ziyade çok farklı MLF blokları içeren CNN ağları da tasarlanabilir. Buna ek olarak ikili yapı da bir CNN ağı tasarımı yapılarak farklı dallarda farklı MLF bloklarının etkisinin değerlendirilmesi çarpıcı bir fikir olarak ele alınabilir. Ayrıca iki farklı daldaki farklı MLF blok tasarımlarına sahip CNN ağlarının etkileşimini ölçmek için daha başka MLF blokları tasarlanarak kombinasyonel olarak birleşimleri ele alınıp kapsamlı bir ablasyon çalışması altında da incelenebileceği düşünülmektedir.

Odak mekanizmaları son yılların en popüler konularından biri olmakla birlikte, CNN ağlarına entegrasyonu ile ciddi performans artışı sağlamaları onları vazgeçilmez yapmaktadır. Bunun yanı sıra literatüre bakıldığında yeni CNN ağlarının geliştirilmesinin yanı sıra yeni odak mekanizmalarının geliştirilmesi üzerine de birçok çalışma olduğu ve yeni odak mekanizmaları ile mevcut CNN ağlarının performanslarının daha da arttığı görülmektedir. Ayrıca tek bir boyut üzerinde ağırlıklandırma yapan geleneksel odak mekanizmalarının yanı sıra, bu boyutlar arasındaki etkileşimlerinde dikkate alan yenilikçi odak mekanizmalarının kullanımının gürültü azaltma işlemi sırasında öznitelikler arasındaki ilişkiyi daha iyi ortaya çıkaracağı için, ciddi performans artışları sağlayabileceği öngörülmektedir. Farklı odak mekanizmalarının performans etkilerinin araştırılmasının yeni bir çalışma olabileceği gibi etkili sonuçlar vereceği de düşünülmelidir.

Son olarak ise konvolüsyon katmanlarının yerini alacağı düşünülen transformatör yapılarına değinmekte fayda görülmüştür. Gürültü azaltma işleminin yanı sıra farklı görüntü işleme alanlarındaki etkileri göz önüne alındığında transformatörler, önümüzdeki yılların en popüler alanı olacağı düşünülmektedir. Geleneksel konvolüsyon katmanları yerel bilgileri yakalarken, transformatörler global bağımlıkları yakalamasıyla yeni ağ tasarımlarında sıklıkla tercih edilmekte ve birçok alanda (gürültü azaltma dahil) çok ciddi performans artışına imkân sağlamaktadır. Bu nedenle geleneksel CNN mimarilerinin yanı sıra, transformatöre tabanlı mimarilerin de ileriki çalışmalar için elzem olmaktadır. Buna ek olarak önerilen mimaride iki farklı CNN ağının kullanılarak gürültü azaltma işlemi yapılması, parametre sayısının ciddi oranda arttırılmış olup eğitim aşamasının yanı sıra tek bir

görüntü için gürültü azaltma işlemi için geçen süreyi de arttırmıştır. Bu nedenle ileriki bir çalışma olarak son yılların GAN yapıları ile gürültü azaltmada sıklıkla kullanılan difüzyon modelleri de düşünülebilir (Xiao vd., 2021). Çünkü difüzyon GAN yapılarının parametre sayısını azaltmasının yanı sıra, güçlü yapısı sayesinde kör gürültü azaltma ve yapay gürültü imgeler dışında gerçek gürültülü görüntüler için de iyi sonuçlar vermesi literatürdeki önemini arttırmıştır (Kawar vd., 2022).



5. KAYNAKLAR

- Ahonen, T., Hadid A., & Pietikäinen M. (2004). Face recognition with local binary patterns. *European conference on computer vision*, s. 469–481. Prague, Czech Republic.
- Anwar, S., & Barnes, N. (2019). Real image denoising with feature attention. In *Proceedings of the IEEE/CVF international conference on computer vision*, s. 3155-3164. Seoul, Korea.
- Baştürk, A. (2006). *Esnek Hesaplamaya Dayalı Yöntemler ile Sayısal İmgelerde Gürültü Giderimi ve İmge İyileştirme*. Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, Kayseri.
- Bovik, A. C. (2010). *Handbook of image and video processing*. Academic press.
- Buades, A., Coll, B., & Morel, J. M. (2005). A review of image denoising algorithms, with a new one. *Multiscale modeling & simulation*, 4(2), 490-530.
- Chollet, F. (2021). *Deep learning with Python*. Simon and Schuster.
- Chen, Y., & Pock, T. (2016). Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 39(6), 1256-1272.
- Chen, Y., Dai, X., Liu, M., Chen, D., Yuan, L., & Liu, Z. (2020). Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, s. 11030-11039. Seattle, USA
- Chupraphawan, S., & Ratanamahatana, C. A. (2019). Deep convolutional neural network with edge feature for image denoising. In *International Conference on Computing and Information Technology*, s. 169-179. Thailand.
- Creswell, A., White, T., Dumoulin, V., Arulkumaran, K., Sengupta, B., & Bharath, A. A. (2018). Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1), 53-65.
- Dabov, K., Foi, A., Katkovnik, V., & Egiazarian, K. (2007). Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8), 2080-2095.
- Dauphin, Y. N., Fan, A., Auli, M., & Grangier, D. (2017). Language modeling with gated convolutional networks. In *International conference on machine learning*, s. 933-941. PMLR.
- Deng, J., Russakovsky, O., Krause, J., Bernstein, M. S., Berg, A., & Fei-Fei, L. (2014, April). Scalable multi-label annotation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, s. 3099-3102. Toronto, Canada.

- Dong, W., Zhang, L., Shi, G., & Li, X. (2012). Nonlocally centralized sparse representation for image restoration. *IEEE transactions on Image Processing*, 22(4), 1620-1630
- Drozdal, M., Vorontsov, E., Chartrand, G., Kadoury, S., & Pal, C. (2016). The importance of skip connections in biomedical image segmentation. In *International Workshop on Deep Learning in Medical Image Analysis, International Workshop on Large-Scale Annotation of Biomedical Data and Expert Label Synthesis*, s. 179-187. Springer, Cham.
- Fang, F., Li, J., Yuan, Y., Zeng, T., & Zhang, G. (2020). Multilevel edge features guided network for image denoising. *IEEE Transactions on Neural Networks and Learning Systems*, 32(9), 3956-3970.
- Franzen, R. (1999). Kodak lossless true color image suite. *source: <http://r0k.us/graphics/kodak>*, 4(2), 9.
- Gholamalinezhad, H., & Khosravi, H. (2020). Pooling methods in deep neural networks, a review. *arXiv preprint arXiv:2009.07485*.
- Gonzalez, R. C. (2009). *Digital image processing*. Pearson education india.
- Goyani, M. M., & Patel, N. (2017). Recognition of facial expressions using local mean binary pattern. *ELCVIA: electronic letters on computer vision and image analysis*, 16(1), 54-67.
- Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., ... & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern recognition*, 77, 354-377.
- Gu, S., Zhang, L., Zuo, W., & Feng, X. (2014). Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2862-2869).
- Guo, M. H., Xu, T. X., Liu, J. J., Liu, Z. N., Jiang, P. T., Mu, T. J., ... & Hu, S. M. (2022). Attention mechanisms in computer vision: A survey. *Computational visual media*, 8(3), 331-368.
- Guo, Y., Jia, X., Zhao, B., Chai, H., & Huang, Y. (2020). Multifeature extracting CNN with concatenation for image denoising. *Signal Processing: Image Communication*, 81, 115690.
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, s. 1026-1034, Chile.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, s. 770-778. Las Vegas, USA.
- Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, s. 7132-7141.

- Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, s. 4700-4708. Honolulu, USA.
- Huang, J. B., Singh, A., & Ahuja, N. (2015). Single image super-resolution from transformed self-exemplars. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, s. 5197-5206. Boston, USA.
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *In International conference on machine learning*, s. 448-456, pmlr.
- Kawar, B., Elad, M., Ermon, S., & Song, J. (2022). Denoising diffusion restoration models. *Advances in Neural Information Processing Systems*, 35, 23593-23606.
- Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. *International Conference on Learning Representations*, s. 1637-1645. San Diego
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84-90.
- LeCun Y., Bottou L., Bengio Y. & Haffner P. (1998). Gradient-based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., & Timofte, R. (2021). Swinir: Image restoration using swin transformer. *In Proceedings of the IEEE/CVF international conference on computer vision*, s. 1833-1844. Nashville, USA.
- Liu, P., Zhang, H., Zhang, K., Lin, L., & Zuo, W. (2018). Multi-level wavelet-CNN for image restoration. *In Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. s.773-782. Salt Lake City, USA.
- Liu, Y., Anwar, S., Zheng, L., & Tian, Q. (2020). Gradnet image denoising. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, s. 508-509. Seattle, USA.
- Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., ... & Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. *In Proceedings of the IEEE/CVF international conference on computer vision*, s. 10012-10022. Nashville, USA.
- Ma, K., Duanmu, Z., Wu, Q., Wang, Z., Yong, H., Li, H., & Zhang, L. (2016). Waterloo exploration database: New challenges for image quality assessment models. *IEEE Transactions on Image Processing*, 26(2), 1004-1016.
- Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. *In Proc. icml*, 30(1), 3-9.
- Mao, X., Shen, C., & Yang, Y. B. (2016). Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. *Advances in neural information processing systems*, 29.

- Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *In Proceedings Eighth IEEE International Conference on Computer Vision*, s. 416-423). Vancouver, Canada.
- McAndrew, A. (2004). *An introduction to digital image processing with MATLAB*. Course Technology Press.
- Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. *In Proceedings of the 27th international conference on machine learning (ICML-10)* s. 807-814. Haifa, Israel.
- Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1), 51–59.
- Ojala, T., Pietikäinen M. (1999). Unsupervised texture segmentation using feature distributions. *Pattern recognition*, 32(3), 477–486.
- O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458.
- Özkan, İ. N. İ. K., & Ülker, E. (2017). Derin öğrenme ve görüntü analizinde kullanılan derin öğrenme modelleri. *Gaziosmanpaşa Bilimsel Araştırma Dergisi*, 6(3), 85-104.
- Plataniotis, K., & Venetsanopoulos, A. N. (2000). *Color image processing and applications*. Springer Science & Business Media.
- Polyak, B. T. (1964). Some methods of speeding up the convergence of iteration methods. *Ussr computational mathematics and mathematical physics*, 4(5), 1-17.
- Roth, S., & Black, M. J. (2005). Fields of experts: A framework for learning image priors. *In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, s. 860-867. San Diego, USA.
- Roth, S., & Black, M. J. (2009). Fields of experts. *International Journal of Computer Vision*, 82, 205-229.
- Saad, D. (1998). Online algorithms and stochastic approximations. *Online Learning*, 5(3), 6.
- Shao, L., Yan, R., Li, X., & Liu, Y. (2013). From heuristic optimization to dictionary learning: A review and comprehensive comparison of image denoising algorithms. *IEEE transactions on cybernetics*, 44(7), 1001-1013.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, s. 1-9. Boston, USA.
- Thakur, R. K., & Maji, S. K. (2023). Multi scale pixel attention and feature extraction based neural network for image denoising. *Pattern Recognition*, 141, 109603.

- Tian, C., Xu, Y., Fei, L., & Yan, K. (2019). Deep learning for image denoising: A survey. *In Genetic and Evolutionary Computing: Proceedings of the Twelfth International Conference on Genetic and Evolutionary Computing*, s. 563-572. Jiangsu, China.
- Tian, C., Xu, Y., Li, Z., Zuo, W., Fei, L., & Liu, H. (2020). Attention-guided CNN for image denoising. *Neural Networks*, 124, 117-129.
- Tian, C., Xu, Y., & Zuo, W. (2020b). Image denoising using deep CNN with batch renormalization. *Neural Networks*, 121, 461-473.
- Tian, C., Zheng, M., Zuo, W., Zhang, B., Zhang, Y., & Zhang, D. (2023). Multi-stage image denoising with the wavelet transform. *Pattern Recognition*, 134, 109050.
- Tian, Y., Wang, Y., Yang, L., & Qi, Z. (2020). CANet: Concatenated attention neural network for image restoration. *IEEE Signal Processing Letters*, 27, 1615-1619.
- Tian, C., Xu, Y., Zuo, W., Du, B., Lin, C. W., & Zhang, D. (2021). Designing and training of a dual CNN for image denoising. *Knowledge-Based Systems*, 226, 106949.
- URL-1. (2023, Aralık 27). Renk uzayları: <https://mesutpiskin.com/blog/renk-uzaylari.html> adresinden alınmıştır.
- URL-2. (2023, Aralık 27). PSNR: https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio adresinden alınmıştır.
- Ulu, A. (2018). *Renkli İmgelerde Gauss Gürültüsü Azaltma*. Karadeniz Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi, Trabzon.
- Waller, B. M., Nixon, M. S., & Carter J. N. (2013). Image reconstruction from local binary patterns. *International Conference on Signal-Image Technology & Internet-Based Systems*, s.118–123. Kyoto, Japan.
- Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., & Hu, Q. (2020). ECA-Net: Efficient channel attention for deep convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, s. 11534-11542. Seattle, USA.
- Wang, Z., Bovik A. C., Sheikh, H. R. & Simoncelli, E. P. (2004). Image Quality Assessment: From Error Measurement to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4) 600-612.
- Wani, M. A., Bhat, F. A., Afzal, S., & Khan, A. I. (2020). *Advances in deep learning*. Springer, Germany.
- Woo, S., Park, J., Lee, J. Y., & Kweon, I. S. (2018). Cbam: Convolutional block attention module. In *Proceedings of the European conference on computer vision (ECCV)*, s. 3-19. Munich, Germany.
- Wu, W., Lv, G., Duan, Y., Liang, P., Zhang, Y., & Xia, Y. (2023). DCANet: Dual Convolutional Neural Network with Attention for Image Blind Denoising. *arXiv preprint arXiv:2304.01498*.

- Xiao, Z., Kreis, K., & Vahdat, A. (2021). Tackling the generative learning trilemma with denoising diffusion gans. *arXiv preprint arXiv:2112.07804*.
- Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., ... & Bengio, Y. (2015, June). Show, attend and tell: Neural image caption generation with visual attention. *In International conference on machine learning*, s. 2048-2057, PMLR.
- Yıldırım, M. T. (2007). *2.Tip Bulanık Sistemler ile Sayısal Görüntülerden Dürtü Gürültüsünün Giderilmesi*. Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Doktora Tezi, Kayseri.
- Yu, F., & Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.
- Zamir, S. W., Arora, A., Khan, S., Hayat, M., Khan, F. S., & Yang, M. H. (2022). Restormer: Efficient transformer for high-resolution image restoration. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, s. 5728-5739. New Orleans, USA.
- Zhang, J., Qu, M., Wang, Y., & Cao, L. (2022). A Multi-Head Convolutional Neural Network with Multi-Path Attention Improves Image Denoising. *In Pacific Rim International Conference on Artificial Intelligence*, s. 338-351. Switzerland.
- Zhang, K., Zuo, W., Chen, Y., Meng, D., & Zhang, L. (2017). Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26(7), 3142-3155.
- Zhang, K., Zuo, W., Gu, S., & Zhang, L. (2017b). Learning deep CNN denoiser prior for image restoration. *In Proceedings of the IEEE conference on computer vision and pattern recognition*. s. 3929-3938. Honolulu, USA.
- Zhang, K., Zuo, W., & Zhang, L. (2018). FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Transactions on Image Processing*, 27(9), 4608-4622.
- Zhang, L., Wu, X., Buades, A., & Li, X. (2011). Color demosaicking by local directional interpolation and nonlocal adaptive thresholding. *Journal of Electronic imaging*, 20(2), 023016-023016.
- Zhao, Y., Jiang, Z., Men, A., & Ju, G. (2019). Pyramid real image denoising network. *In 2019 IEEE Visual Communications and Image Processing (VCIP)*, s. 1-4. IEEE.

ÖZGEÇMİŞ

Ahmet ULU, ilk ve orta öğrenimini Yeşil Giresun İlköğretim Okulu, lise öğrenimini Giresun Lisesi'nde tamamladı. 2009 yılında 19 Mayıs Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü'nde lisans programına başladı ve 2013 yılında bu bölümden mezun oldu. 2015 yılı itibari ile Karadeniz Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda yüksek lisans öğrenimine başladı. 2016 yılı bahar döneminden itibaren aynı üniversitede Bilgisayar Mühendisliği bölümünde araştırma görevlisi olarak işe başlamıştır. 2018 yılında yüksek lisans eğitimi tamamlayarak aynı yıl KTÜ Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda doktora eğitimine başladı. 2023 yılından itibaren Artvin Çoruh Üniversitesi Bilgisayar Mühendisliği bölümünde araştırma görevlisi olarak çalışmaktadır. Yabancı dil olarak iyi seviyede İngilizce bilmektedir. Tezden üretilen akademik çalışmaları ve yayınları aşağıdadır.

SCI, SSCI, AHCI İndekslerine Giren Dergilerde Yayınlanan Makaleler

Ulu, A., Yıldız, G., & Dizdaroğlu, B. (2023). MLFAN: Multilevel Feature Attention Network With Texture Prior for Image Denoising. *IEEE Access*.

Diğer Dergilerde Yayınlanan Makaleler

Ulu, A., Özen, Y., Öztürk, E., & Dizdaroğlu, B., (2019) Renkli İmgelerde Çoklu Gürültü Giderme. *International Journal of 3D Printing Technologies and Digital Industry*, 2(3), 22-29.

Hakemli Kongre / Sempozyumların Bildiri Kitaplarında Yer Alan Yayınlar

ULU, A., Dizdaroğlu, B. (2023). Image Denoising via Multiscale Feature Extraction and Triplet Attention based CNN. 14th International Conference on Electrical and Electronics Engineering. ELECO 2023.