

**T.C.  
İNÖNÜ ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**ÜRETKEN AĞLAR ve UYGULAMALARI**



**DOKTORA TEZİ**

**Gaffari ÇELİK**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Tez Danışmanı: Doç. Dr. Muhammed Fatih TALU**

**HAZİRAN 2021**

**T.C  
İNÖNÜ ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**ÜRETKEN AĞLAR ve UYGULAMALARI**

**DOKTORA TEZİ**

**Gaffari ÇELİK  
(23616190013)**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Tez Danışmanı: Doç. Dr. Muhammed Fatih TALU**

**HAZİRAN 2021**

## TEŐEKKÜR VE ÖNSÖZ

Bu tez alıőmasının her aőamasında yardım, öneri, bilgi, tecrübe ve desteklerini esirgemedен beni her konuda yönlendiren danışman hocam Sayın Do. Dr. Muhammed Fatih TALU hocama,

Tez alıőmam süresince deęerli görüşleriyle tezime katkı saęlayan deęerli hocalarım Prof. Dr. Ali KARCI, Do. Dr. Barıő Baykant ALAGÖZ hocalarıma ve ayrıca bana destek olan Bilgisayar Mühendislięi Bölümünde görev yapan tüm dięer deęerli hocalarıma;

Görev yaptığım üniversitede bu süreçteki desteklerinden dolayı deęerli Rektörüm, Müdürüm ve mesai arkadaşlarıma;

Ayrıca her konuda olduęu gibi bu süreçte de desteklerini esirgemeyen deęerli eşime ve beni sabırla bekleyen çocuklarıma;

Sonsuz teşekkürlerimi sunarım.

## ONUR SÖZÜ

Doktora tezi olarak sunduđum “Üretken Ağlar ve Uygulamaları” başlıklı bu çalışmanın bilimsel ahlak ve geleneklere aykırı düşecek bir yardıma başvurmaksızın tarafımdan yazıldığına ve yararlandığım bütün kaynakların hem metin içinde hem de kaynakçada yöntemine uygun biçimde gösterilenlerden oluştuđunu belirtir, bunu onurumla doğrularım.

Gaffari ÇELİK



## İÇİNDEKİLER

TEŞEKKÜR VE ÖNSÖZ .....	i
ONUR SÖZÜ .....	ii
İÇİNDEKİLER.....	iii
ÇİZELGELER DİZİNİ.....	v
ŞEKİLLER DİZİNİ.....	vi
SEMBOLLER VE KISALTMALAR .....	vii
ÖZET.....	x
ABSTRACT .....	xi
<b>1. GİRİŞ.....</b>	<b>1</b>
1.1 GAN Mimarisi ve Eğitimi .....	3
1.2 Tezin Amaç ve Hedefleri.....	5
1.3 Tezin Organizasyonu ve Yapılan Bilimsel Katkılar .....	6
<b>2. SENTETİK GÖRÜNTÜ ÜRETİMİ .....</b>	<b>7</b>
2.1 Klasik GAN mimarileri .....	7
2.1.1 cGAN.....	7
2.1.2 DCGAN .....	8
2.1.3 ACGAN.....	8
2.1.4 InfoGAN .....	9
2.1.5 SGAN .....	9
2.1.6 WGAN-GP .....	10
2.1.7 LSGAN .....	11
2.2 Önerilen GAN mimarisi .....	11
2.3 Deneysel Çalışmalar ve Sonuçları .....	12
<b>3. HİSTOPATOLOJİK GÖRÜNTÜLERİN ONARIMI VE TEMİZLENMESİ</b>	<b>18</b>
3.1 Çözünürlük Arttırmada Kullanılan Klasik Yaklaşımlar .....	18
3.1.1 Bilinear interpolasyon .....	18
3.1.2 Bicubic interpolasyon .....	19
3.2 Süper Çözünürlüklü Çekişmeli Üretken Ağ (SRGAN).....	19
3.3 Gürültü Temizleme Yöntemleri.....	20
3.3.1 Ortanca filtre.....	20
3.3.2 Ortalama filtre .....	20
3.3.3 Adaptif filtre .....	21
3.4 Veri Kümesinin Hazırlanması .....	21
3.5 Deneysel Çalışmalar ve Sonuçları .....	22
<b>4. EEG SİNYALLERİNDEN GÖRÜNTÜ ÜRETİMİ .....</b>	<b>28</b>
4.1 ACGAN Mimarisi .....	29
4.2 Kodlama Teknikleri .....	30
4.2.1 Otomatik kodlayıcı .....	30
4.2.2 Değişken otomatik kodlayıcı .....	31
4.2.3 Fourier dönüşümü.....	32
4.3 Önerilen EEG-GAN Mimarisi .....	32
4.4 Veri Kümesi.....	34
4.5 Deneysel Çalışmalar ve Sonuçları .....	35
<b>5. 3D BEYİN MRI TARAMALARININ BÖLÜTLENMESİ.....</b>	<b>41</b>
5.1 Veri Kümeleri .....	44
5.2 Ön İşlemler .....	45
5.2.1 Beyin bölgesinin çıkarılması .....	45
5.2.2 Etiketlerin düzenlenmesi .....	46
5.2.3 Örtüştürme .....	46

5.2.4 Kırpma ve örnekleme .....	47
5.3 Önerilen Bölütleme Mimarisi .....	47
5.4 Deneysel Çalışmalar ve Sonuçları .....	51
<b>6. SONUÇLAR.....</b>	<b>58</b>
<b>KAYNAKLAR.....</b>	<b>61</b>
<b>ÖZGEÇMİŞ .....</b>	<b>72</b>



## ÇİZELGELER DİZİNİ

Çizelge 2.1 : Üretilen görüntülerin FID metriğine göre benzerlik sonuçları.....	14
Çizelge 2.2 : GAN mimarilerinin CNN yöntemine göre başarıları. ....	14
Çizelge 3.1 : Görüntü büyütme sonuçları. ....	26
Çizelge 3.2 : Gürültü temizleme sonuçları.....	27
Çizelge 4.1 : MindWave ve DB veri kümelerinin güvenirlilik test sonuçları.....	35
Çizelge 4.2 : EEG-Görsel atama işleminden sonra bir kaydın boyutları.....	36
Çizelge 4.3 : EEG-GAN mimarisi. ....	37
Çizelge 4.4 : EEG sinyallerinden üretilen örnek görüntüler.....	38
Çizelge 4.5 : EEG sinyallerinden üretilen görüntülerin CNN başarıları.....	40
Çizelge 5.1 : Veri kümeleri ve özellikleri.....	44
Çizelge 5.2 : Örtüştürme işlemi öncesi ve sonrasında MRI boyutları. ....	47
Çizelge 5.3 : Geliştirilen dört farklı mimarinin üç parça bölütleme sonuçları.....	52
Çizelge 5.4 : Önerilen bölütleme mimarisinin mevcut yöntemler içerisindeki konumu.....	53
Çizelge 5.5 : Sekiz parçalı bölütleme sonuçları.....	54
Çizelge 5.6 : Mimamarilerin MRI bölütleme süreleri.....	56

## ŞEKİLLER DİZİNİ

Şekil 1.1 : GAN'ın literatürdeki farklı kullanım alanları.....	3
Şekil 1.2 : Klasik GAN mimarisi [34,35]. .....	4
Şekil 1.3 : Logaritmik maliyet fonksiyonu .....	5
Şekil 2.1 : Klasik GAN mimarileri. ....	8
Şekil 2.2 : SGAN mimarisi [54].....	10
Şekil 2.3 : cDCGAN mimarisi [40]. ....	12
Şekil 2.4 : GAN mimarileri ile üretilen örnek MNIST görüntüleri. ....	13
Şekil 2.5 : Üretilen örnek MNIST görüntüleri.....	16
Şekil 2.6 : Üretilen örnek Fashion-MNIST görüntüleri.....	16
Şekil 2.7 : Modellerin FID benzerlik metriğine göre zamansal olarak toplu görüntü üretme performansları. ....	17
Şekil 3.1 : SRGAN Mimarisi. ....	19
Şekil 3.2 : WSI görüntülerin piramit yapısı. ....	22
Şekil 3.3 : Görüntü büyütme örnekleri.....	23
Şekil 3.4 : Gürültü temizleme örnekleri.....	24
Şekil 4.1 : ACGAN mimarisi.....	30
Şekil 4.2 : Önerilen EEG-GAN modeli. ....	33
Şekil 5.1 : Beyin çıkarma işlemi. ....	45
Şekil 5.2 : Önerilen bölütleme mimarisi (Vol2SegGAN).....	49
Şekil 5.3 : PAM modülü [136].....	50
Şekil 5.4 : Üretici mimarisi. ....	50
Şekil 5.5 : Ayrıcı mimarisi.....	51
Şekil 5.6 : Örnek bir MRI üzerinde yöntemlerin bölütleme sonuçları.....	54
Şekil 5.7 : Örnek bir MRI üzerindeki bölütleme sonuçları (yukarıdan aşağıya doğru 20, 25, 30 ve 35 nolu görüntüler).....	56



## SEMBOLLER VE KISALTMALAR

$\mathbf{z}^{(i)}$	: Gürültü vektörü
$\mathbf{x}^{(i)}$	: Gerçek görüntüler
$G(\mathbf{z}^i)$	: Üretilen (sahte) görüntü
$\mathbf{c}$	: Sınıf etiket bilgisi
$X_{sahte}$	: Üretilen (sahte) görüntü
$L_S$	: Görüntüler üzerindeki log-olabilirlik dağılımını
$L_C$	: Sınıflar üzerindeki log-olabilirlik dağılımını
$\mu_g$	: Üretilen görüntülerin ortalama değeri
$C_g$	: Üretilen görüntülerin kovaryans değeri
$\mu_x$	: Gerçek görüntülerin ortalama değeri
$C_x$	: Gerçek görüntülerin kovaryans değeri
$\rho$	: Çürüme oranı
$lr$	: Öğrenme katsayısı
$u(x, y)$	: Piksel renk değeri
$\nabla_{\theta_{\text{çekişme}}}$	: Çekişme maliyeti
$\nabla_{\theta_{\text{içerik}}}$	: İçerik maliyeti
$\beta_1$	: Üstel bozulma oranı
$hr$	: Gerçek görüntü
$sr$	: Yeniden boyutlandırılan görüntü
$\mu_{hr}$	: Gerçek görüntülerin renk ortalaması
$\mu_{sr}$	: Yeniden boyutlandırılan görüntülerin renk ortalaması
$\sigma_{hr}$	: Gerçek görüntülerin standart sapması
$\sigma_{sr}$	: Yeniden boyutlandırılan görüntülerin standart sapması
$\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$	: Sıfıra bölme hatasını engelleyen küçük skaler değerler
$L$	: Maksimum renk yoğunluk değeri
$fft$	: Uzaysal alandaki sinyalin frekans alanına dönüştürülmesi
$f_{kodlayıcı}$	: Kodlayıcı (Encoder)
$f_{çözücü}$	: Kod çözücü (Decoder)
$\mathcal{L}_{KDE}$	: Düzgünleştirme hatası
$\mathcal{L}_{MSE}$	: Oturma hatası
$X_{FD}(f)$	: Frekans alanı
$x(t)$	: Zaman alanı
$randn$	: Normal dağılım fonksiyonu

<b>TP</b>	: Doğru sınıflandırılan pozitif örnek sayısı
<b>TN</b>	: Doğru sınıflandırılan negatif örnek sayısı
<b>FP</b>	: Yanlış sınıflandırılan pozitif örnek sayısı
<b>FN</b>	: Yanlış sınıflandırılan negatif örnek sayısını
<b>Mask(x)</b>	: Görüntü maske oluşturma
<b>Brain(x)</b>	: Beyin bölgesini elde etme
<b>GT<sup>new</sup><sub>MR13</sub>(x)</b>	: MR13 veri kümesindeki gerçek değerlerin (GroundTrue) etiketlenmesi
<b>GT<sup>new</sup><sub>MR18</sub>(x)</b>	: MR18 veri kümesindeki gerçek değerlerin (GroundTrue) etiketlenmesi
<b>L<sub>2</sub></b>	: Karesel hata
<b>ACFP</b>	: Atrus evrişimli özellik piramidi (Atrous convolution feature pyramid)
<b>ACGAN</b>	: Yardımcı sınıflandırıcı GAN
<b>AE</b>	: Otomatik kodlayıcı (Auto encoder)
<b>BCI</b>	: Beyin/Bilgisayar arayüzleri
<b>cDCGAN</b>	: Koşullu derin konvolüsyonel GAN
<b>cGAN</b>	: Koşullu (conditional) GAN
<b>CNN</b>	: Evrişimsel sinir ağları (Convolutional neural networks)
<b>Conv2D</b>	: 2B Evrişim Katmanı
<b>CT</b>	: Bilgisayarlı tomografi
<b>D</b>	: Ayırıcı (discriminator)
<b>DB</b>	: EEG veri kümesi
<b>DCGAN</b>	: Derin konvolüsyonel GAN
<b>Dense</b>	: Tam Bağlı Katman
<b>Dropout</b>	: Düğüm Düşürme Fonksiyonu
<b>EEG</b>	: Elektroensefalografi
<b>EM</b>	: Earth-Mover
<b>Faster-RCNN</b>	: Daha Hızlı Bölge Tabanlı Evrişimli Sinir Ağı (Faster Region-Based Convolutional Neural Network)
<b>FD</b>	: Fourier dönüşümü
<b>FFT_MSE</b>	: Frekansın ortalama karesel hatası
<b>FID</b>	: Fréchet başlangıç mesafesi
<b>Flatten</b>	: Vektöre Dönüştürme
<b>G</b>	: Üretici (generative)
<b>GAN</b>	: Çekişmeli üretken ağlar (Generative adversarial networks)
<b>GT</b>	: Gerçek değer (GroundTrue)
<b>HD</b>	: Hausdorff mesafesi (Hausdorff distance)

<b>HR</b>	: Yüksek çözünürlükteki görüntü
<b>InfoGAN</b>	: Bilgi GAN
<b>LR</b>	: Düşük çözünürlükteki görüntü
<b>LSGAN</b>	: En az kareler GAN
<b>LSTM</b>	: Uzun Kısa Süreli Bellek (Long Short Term Memory)
<b>MR13</b>	: MRBRAINS13
<b>MR18</b>	: MRBRAINS18
<b>MRI</b>	: Manyetik rezonans görüntüleme
<b>Multiply</b>	: Çarpma
<b>PAM</b>	: Konum dikkat modülü (position attention module)
<b>PSNR</b>	: Sinyal-gürültü oranı (Peak signal-to-noise ratio)
<b>R-CNN</b>	: Bölge Tabanlı Evrişimli Sinir Ağı (Region-Based Convolutional Neural Network)
<b>RNN</b>	: Tekrarlayan Sinir Ağları (Recurrent Neural Networks)
<b>SCE</b>	: Sigmoid Cross Entropy
<b>SGAN</b>	: Yarı eğiticili GAN
<b>SR</b>	: Süper çözünürlüklü görüntü
<b>SRGAN</b>	: Süper çözünürlüklü çekişmeli üretken ağ
<b>SSIM</b>	: Yapısal benzerlik metriği (Structural similarity index measure)
<b>VAE</b>	: Değişken otomatik kodlayıcı (Variational auto encoder)
<b>VS</b>	: Hacimsel benzerlik (Volumetric Similarity)
<b>WGAN-GP</b>	: Wasserstein GAN'ların geliştirilmiş eğitimi
<b>ZeroPadding</b>	: Sıfırla Doldurma

# ÖZET

Doktora Tezi

ÜRETKEN AĞLAR ve UYGULAMALARI

GAFFARİ ÇELİK

İnönü Üniversitesi  
Fen Bilimleri Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalı

73+XI sayfa

2021

Danışman: Doç. Dr. Muhammed Fatih TALU

Bu tez çalışmasında gürültüden görsel üretme, çözünürlük artırma, sinyalden görüntü üretme ve bölütleme gibi alanlardaki en güncel yaklaşımlar incelenmiş ve Çekişmeli Üretken Ağlar (GAN) temelli yeni mimariler geliştirilerek literatüre katkılar sunulmuştur. Tez kapsamında yapılan ilk deneysel çalışmada, klasik GAN mimarileri (cGAN, DCGAN, InfoGAN, SGAN, ACGAN, WGAN-GP ve LSGAN) kullanılarak MNIST ve Fashion-MNIST veri kümelerindeki görseller üretilmektedir. Ayrıca bu yaklaşımlara alternatif olacak yeni bir hibrit GAN mimarisi (cDCGAN) önerilmektedir. Yapılan ikinci deneysel çalışma, SRGAN mimarisinin Camelyon17 veri kümesine uygulanmasını içermektedir. Buna göre SRGAN mimarisinin görsel çözünürlüğünü iyileştirilme ve gürültüyü giderilme performansı klasik yaklaşımlarla kıyaslanmaktadır. Tez kapsamında yapılan üçüncü deneysel çalışmada, GAN mimarilerinin sinyalden görsel üretme performansı araştırılmıştır. Bunun nedeni EEG sinyal verilerinden planlanan aktivitenin tahmin edilmesi veya engelliler için tekerlekli sandalyenin hareketlendirilmesi gibi ihtiyaçların varlığıdır. Ancak mevcut çalışmalar incelendiğinde EEG sinyalinin sadece sınıflandırılma ile sınırlı kaldığı görülmektedir. GAN ve Otomatik Kodlayıcı (Auto encoder-AE) tekniklerinin kullanıldığı az sayıda çalışma incelenmiş ve yeni mimarilerle sinyal-görsel üretim performansı iyileştirilmiştir. Tez kapsamındaki yapılan son deneysel çalışma, 3D MRI verilerinin bölütlenmesi hakkındadır. Yapılan araştırma sonucunda güncel bölütleme mimarileri belirlenmiş ve üç farklı veri kümesi (IBSR18, MRBRAINS13 ve MRBRAINS18) kullanılarak bölütleme sonuçları elde edilmiştir. Mevcut yaklaşımlara alternatif olacak Vol2SegGAN adlı yeni bir bölütleme yaklaşımı önerilmiştir. Bu mimarinin üretici ağında ACFP ve PAM modülleri yer alırken, ayırıcı ağı gerçek/sahte ayrımını gerçekleştirmektedir. T1 modaliteye sahip 3D-MRI taramalarının kullanıldığı verilerde üç farklı bölge (GM, WM, CSF) ve sekiz farklı bölge (CGM, BG, WM, WMH, CF, VE, CE ve BS)'nin bölütlenme çalışmaları yapılmıştır.

Yapılan deneysel faaliyetler sonucunda, GAN temelli yaklaşımların birçok problem çözümüne olumlu katkılar sağladığı görülmüştür.

**Anahtar Kelimeler:** Derin Öğrenme, Çekişmeli Üretken Ağlar, Sentetik Görüntü Üretimi, GAN Uygulamaları, Görüntü Boyutlandırma, SRGAN, Gürültü Temizleme, EEG-GAN, EEG Sinyalleri, 3D MRI Taramaları, Bölütleme, Örtüşürme, Vol2SegGAN, PAM, ACFP

# ABSTRACT

Phd. Thesis

GENERATIVE NETWORKS and THEIR APPLICATIONS

Gaffari ÇELİK

Inonu University  
Graduate School of Natural and Applied Sciences  
Department of Computer Science

73+XI page

2021

Supervisor: Doç. Dr. Muhammed Fatih TALU

In this thesis, the most up to date approaches in areas such as image generation from noise, resolution enhancement, image generation from signal and segmentation have been examined, and new architectures based on Generative Adversarial Networks (GAN) have been developed and contributed to the literature. In the first experimental study conducted within the scope of the thesis, images in MNIST and Fashion-MNIST datasets are produced using classical GAN architectures (cGAN, DCGAN, InfoGAN, SGAN, ACGAN, WGAN-GP and LSGAN). In addition, a new hybrid GAN architecture (cDCGAN) is proposed as an alternative to these approaches. The second experimental study includes the application of the SRGAN architecture to the Camelyon17 dataset. Accordingly, the visual resolution improvement and noise removal performance of the SRGAN architecture is compared with classical approaches. In the third experimental study conducted within the scope of the thesis, the visual generation performance of the GAN architectures from the signal was investigated. The reason for this is the existence of needs such as estimating the planned activity from the EEG signal data or mobilizing the wheelchair for the disabled. However, when the existing studies are examined, it is seen that the EEG signal is limited to classification only. Few studies using GAN and Auto encoder (AE) techniques have been examined and signal-visual production performance has been improved with new architectures. The last experimental work within the scope of the thesis is about segmentation of 3D MRI data. As a result of the research, current segmentation architectures were determined and segmentation results were obtained using three different datasets (IBSR18, MRBRAIN13 and MRBRAIN18). A new segmentation approach called Vol2SegGAN has been proposed as an alternative to existing approaches. While ACFP and PAM modules are included in the generative network of this architecture, the discriminator network realizes the real/fake distinction. In the data using 3D-MRI scans with T1 modality, segmentation studies of three different regions (GM, WM, CSF) and eight different regions (CGM, BG, WM, WMH, CF, VE, CE and BS) were performed.

As a result of the experimental activities, it has been seen that GAN-based approaches make positive contributions to many problem solutions.

**Keywords:** Deep Learning, Generative Adversarial Networks, Synthetic Image Generation, GAN Applications, Image Resizing, SRGAN, Noise Removal, EEG-GAN, EEG Signals, 3D MRI Scans, Segmentation, Registration, Vol2SegGAN, PAM, ACFP

## 1. GİRİŞ

Yapay Sinir Ağları yöntemi, insan beynindeki bir sinir hücresinin çalışma modelini matematiksel olarak taklit etme fikrine dayanmaktadır. 1996'lı yıllarda özellikle veri sınıflandırma ve tahminleme alanında oldukça popüler olan bu yaklaşımın aşağıda listelenilen belirli dezavantajları görülmüştür [1-4]:

- 1) Bir kara kutu gibi olması ve olası nedensel ilişkileri belirleme konusunda sınırlı yeteneklere sahip olması
- 2) Tek boyutlu vektörel veri (öznitelik vektörü-feature vector) ile çalışabilmesi
- 3) Öznitelik çıkarma algoritmalarına ihtiyaç duyması
- 4) Katman sayısı parametresiyle hesaplama maliyetinin eksponansiyel ilişkisi

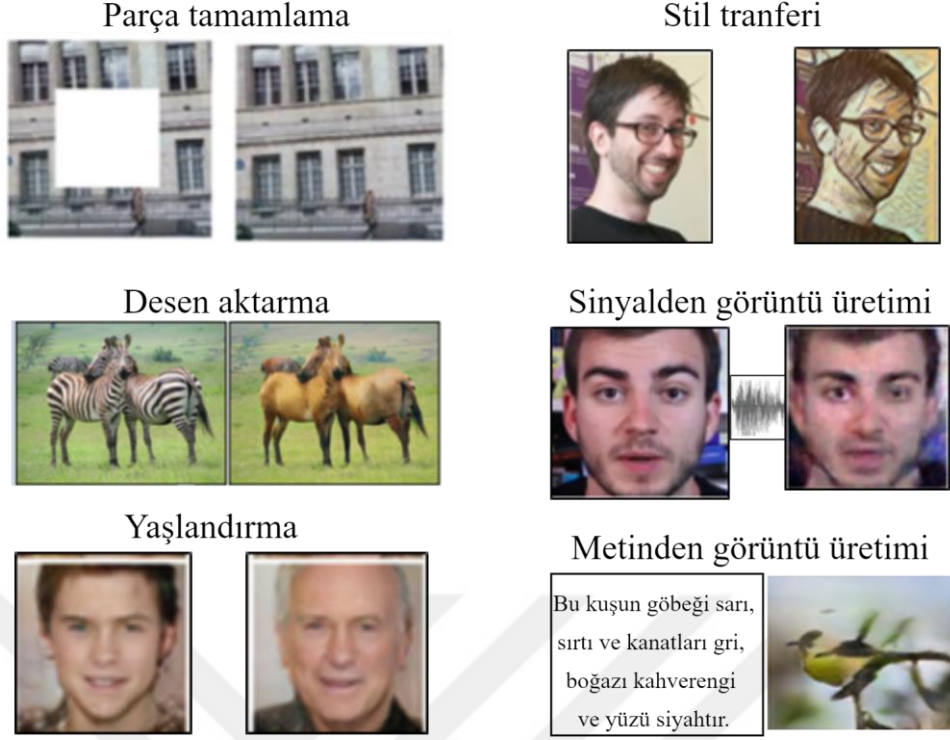
Bahsedilen bu dezavantajların ortadan kaldırılabilmek için 1998 yılında LeNet adlı derin öğrenme mimarisinin görüntü sınıflandırma problemine çözüm olarak önerilen ilk evrimsel sinir ağı (Convolution Neural Network-CNN) olarak görülmektedir [5]. Ancak derin öğrenme yaklaşımının yüksek işlem kapasiteli donanıma ihtiyaç duyulması ve o tarihlerde bu ihtiyacın yeterince karşılanamaması nedeniyle akademik camiada yaygınlaşmadığı görülmektedir. Son dönemlerde, bulut bilişim, GPU hızlandırma ve çok çekirdekli işlemcilerin paralelleştirilmesi gibi donanımsal gelişmeler sayesinde bu sınırlamalar kısmen aşılmış ve derin öğrenme yapay öğrenme alanında önemli bir atılım olarak tanınmasını sağlamıştır [6].

Krizhevsky ve arkadaşları [7] tarafından önerilen AlexNet, 2012 yılında ILSVRC yarışmasında ImageNet büyük ölçekli görüntü sınıflandırılması problemini çözmek için bir GPU destekli program kullanarak yarışmayı kazanması ile CNN mimarileri yeniden popüler hale gelmiş araştırmacıların ilgisini çekmiştir [8]. 2014 yılında Google tarafından önerilen Inception (GoogleNet) adlı mimari, farklı boyutlu konvolüsyonel maske çıktılarının birlikte değerlendirilmesiyle sınıflandırma başarımının bir derece daha iyileştirdiği görülmüştür [9]. 2015 yılında Microsoft tarafından önerilen ResNet [10] mimarisiyle sınıflandırma başarımının arttırıldığı görülmektedir. Bu yaklaşım “artık (residual)” olarak isimlendirilen güncel katman çıktısının sonraki katmanlarda kullanılması fikrine dayanmaktadır.

Aşağıda listelenen çalışma alanlarında farklı derin öğrenme teknikleri ortaya çıkmıştır:

- 1) Doğal dil işleme: Bu alanda, arama metinlerindeki maksadın en iyi şekilde anlaşılması, metin özetlerinin ve sözcük etiketlerinin çıkarılması gibi çalışmalar yapılmaktadır [11]. Bunun için RNN [12] ve LSTM [13] gibi tek yönlü, BERT [14] gibi çift yönlü derin öğrenme mimarilerinin kullanıldığı görülmektedir.
- 2) Veri sıkıştırma ve açma: Derin öğrenmenin diğer bir uygulama alanı olan veri sıkıştırma ve çözme problemlerinde veriyi en iyi ifade eden daha küçük bir boyutta tanımlamak ve boyutu indirgenmiş veriyi çözmek için kullanılmaktadır. Bunun için kodlayıcı ve çözücü ağlarından oluşan otomatik kodlayıcı (Auto Encoder, AE) mimariler aktif olarak kullanıldığı görülmektedir. Kodlayıcı ağ, giriş verisini daha küçük boyuta indirgeyerek en önemli özelliklerini öğrenmek amacıyla veriyi sıkıştırmaktadır. Çözücü ağ, sıkıştırılan veriyi minimum bir kayıpla çözme yöntemi ile giriş verisine benzetmeyi amaçlamaktadır [15].
- 3) Sınıflandırma ve nesne tanıma: CNN mimarileri 2014 yılında ImageNet yarışmasında binlerce görüntü ve sınıftan oluşan görüntülerin sınıflandırılması ve tanınmasında büyük bir başarı sergilediği görülmüştür. İlerleyen dönemlerde bu mimariler Akciğer [16], alzheimer [17] ve kanser [18] gibi farklı hastalıkların sınıflandırılmasında ve farklı sınıflandırma problemlerinde etkin bir şekilde kullanıldığı görülmektedir [3]. Nesne tanıma, sınıflandırma probleminden farklı olarak görüntülerdeki nesnelere bulmaya yönelik çalışmaları içermektedir. Bu tür çalışmalarda R-CNN [19] ve Faster-RCNN [20] mimarileri yaygın bir şekilde kullanılmaktadır.
- 4) Görüntü üretme: Gerçek görüntülere benzer sentetik görüntülerin üretiminde Çekişmeli Üretken Ağlar (Generative Adversible Networks-GAN) oldukça başarılı bir performans sergilediği görülmektedir [21]. Bu mimarilerin görüntü üretimindeki başarısının yanı sıra görüntüdeki eksik parçaların tamamlanması [22,23], çözünürlük artırma [24,25], nesneye desen aktarma [26-28], yüz yaşlandırma [29], ressam stil transferi [30], ses sinyalinin ve metin bilgisinden görüntü üretimi [31-33] gibi birçok farklı alanda çalışmalar (bak. Şekil 1.1) yapılmaktadır.

Bu teknikler içerisinde GAN mimarisinin oldukça geniş bir kullanım alanına sahip olması ve iki farklı mimarinin çekişmeli olarak çalışabilmesinin oldukça özgün bir kullanım olması bizleri bu alana sevk etmiştir.



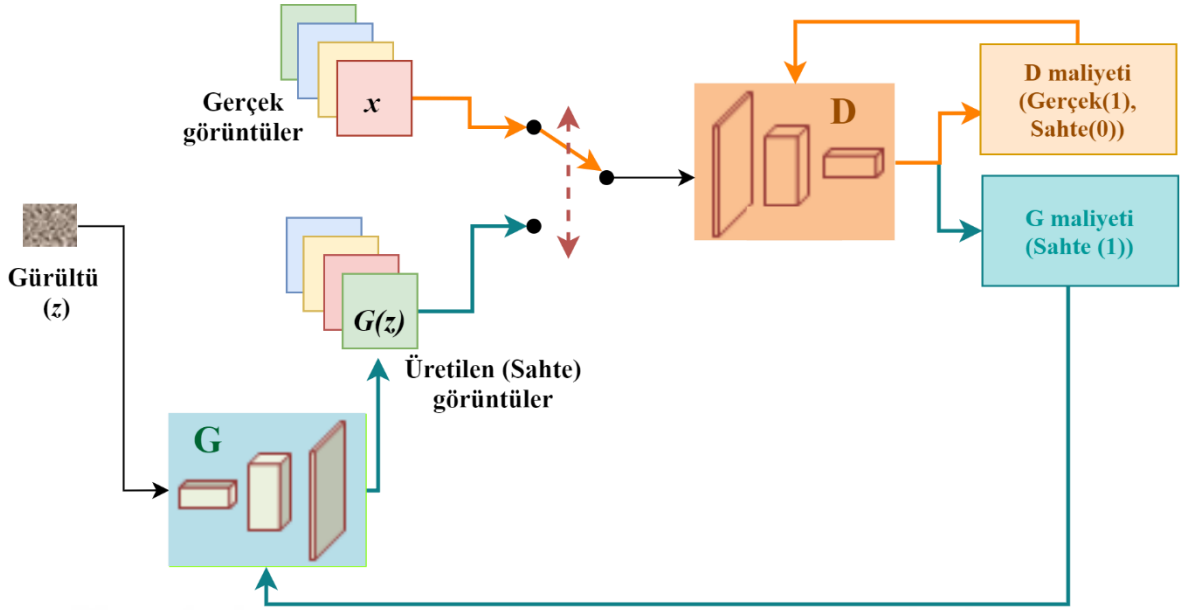
Şekil 1.1 : GAN'ın literatürdeki farklı kullanım alanları

## 1.1 GAN Mimarisi ve Eğitimi

Çekişmeli üretken ağlar ilk olarak Goodfellow ve arkadaşları [21] tarafından önerilmiştir. Bu ağların amacı, veri kümesindeki örnek görsellere benzer sentetik görüntüler üretmektir. Klasik derin ağ mimarilerinde farkı, üretici ve ayırıcı olmak üzere iki derin ağ mimarisinin eğitimini çekişmeli ve eşzamanlı bir şekilde gerçekleştirebilmesidir.

Klasik GAN mimarisi Şekil 1.2'de gösterildiği gibi bir üretici (generative,  $G$ ) ve bir ayırıcı (discriminator,  $D$ ) olmak üzere iki farklı derin ağ mimarisine sahiptir. Bu yapısı ile bilinen derin ağ mimarilerinden farklılık gösterir ve öğrenme işlemi bu iki ağın rekabetçi bir şekilde çalışmasıyla gerçekleşir.





Şekil 1.2 : Klasik GAN mimarisi [34,35].

$G$  derin ağ, giriş olarak aldığı bir gürültü vektöründen gerçek görüntülere benzer görüntüler üretmeye çalışırken;  $D$  derin ağ, üretici tarafından üretilen sentetik görüntüler (sahte (0)) ile gerçek(1) görüntüleri ayırt etmeye çalışır. Bu şekilde her iki ağ da paralel, çekişmeli bir şekilde eğitilir ve bu eğitim aşamasında  $G$  kademeli olarak gerçek görüntüler ile ayırt edilemeyecek yeni görüntüler üretir. Klasik GAN mimarisin sözde kodu aşağıdaki şekilde verilmiştir [21,36]:

**Giriş:**  $z^{(i)}$ -> gürültü vektörü,  $x^{(i)}$ -> gerçek görüntü örneği,

**Çıkış:**  $G(z^i)$ -> sahte görüntü örneği

**For her bir epok do**

**For her bir iterasyon do**

$m$  adet sahte görüntü üret  $\{G(z^{(1)}), \dots, G(z^{(m)})\}$

$m$  adet gerçek görüntü al  $\{x^{(1)}, \dots, x^{(m)}\}$

$D$ 'yi güncelle:  $\min_{\theta^D} \left( \frac{1}{m} \sum_{i=1}^m \left[ -y^{(i)} \log D(x^{(i)}) - (1 - y^{(i)}) \log (1 - D(G(z^i))) \right] \right)$

$m$  adet sahte görüntü üret  $\{G(z^{(1)}), \dots, G(z^{(m)})\}$

$G$ 'yi güncelle:  $\max_{\theta^G} \left( \frac{1}{m} \sum_{i=1}^m - \log (1 - D(G(z^i))) \right)$

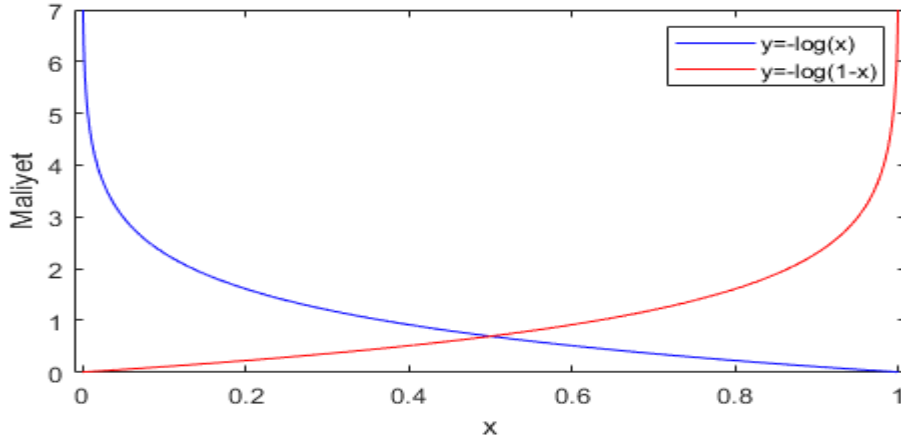
**end for**

**end for**

Sözde kodunda,  $z^{(i)}$  gürültü sinyalini,  $G(z^i)$  nütasyonuyla üretilen sahte görüntüleri temsil etmektedir.  $x^{(i)}$  veri kümesindeki gerçek görüntüleri ifade etmektedir.  $epok$ ,  $G$  ve  $D$  ağlarının

kaçar defa eğitileceğini göstermektedir. Ağların eğitimi, tüm veri kümesi üzerinde gerçekleşmesi bellek ve zaman açısından maliyetli bir işlem olduğu için, veri kümeleri gruplar (mini paket) halinde alınarak eğitim işlemi gerçekleşir [37]. Bu sayede eğitim, veri kümesindeki tüm görüntüler aynı anda kullanılmaz ve her pakette  $GrupGörüntüSayısı$  kadar görüntü alınarak öğrenme işlemi gerçekleşir. *her bir iterasyon, ToplamGörüntüSayısı/GrupGörüntüSayısı* ile hesaplanır.

$D$  derin ağ, “logistic regression” maliyet fonksiyonu kullanarak, gerçek ve sahte görüntüleri birbirinden ayıran ikili sınıflandırıcıdır (binary classifier).  $D$ , giriş olarak gerçek  $(x^{(i)})(1)$  ve sahte  $(G(z^i))(0)$  görüntülerini alır. Çıkış olarak ikili sınıfa (gerçek (1), sahte (0)) göre sınıflandırır ve elde edilen maliyet değerine göre geri yayılım yöntemi ile eğitilir.  $G$ , giriş olarak aldığı gürültü  $(z^{(i)})$  sinyalinden sentetik  $(G(z^i))$  görüntüler üretir, bu görüntüler gerçekmiş gibi  $D$  mimarisine verilir ve bu şekilde  $D$  mimarisi kandırılmaya çalışılırken elde edilen yüksek maliyet değeriyle  $G$  eğitilir [35]. Şekil 1.2’de logistic regression” maliyet grafiği verilmiştir.



Şekil 1.3 : Logaritmik maliyet fonksiyonu

## 1.2 Tezin Amaç ve Hedefleri

Bu tez çalışmasının amacı, sinyalden görüntü üretme, çözünürlük artırma ve bölütleme gibi farklı çalışma alanlarında klasik yaklaşımlara kıyasla performans artışı sağlayan özgün GAN mimarilerinin geliştirilmesidir. Bunun için ilk aşamada bahsedilen alanlardaki mevcut yaklaşımlar detaylıca incelenmekte, daha sonra performans artışı sağlayan özgün mimariler geliştirilerek literatüre katkılar sunulmaktadır.

### 1.3 Tezin Organizasyonu ve Yapılan Bilimsel Katkılar

Tez çalışmasının ikinci bölümünde, bir gürültü sinyalinden görüntü üretebilen farklı GAN mimarileri incelenmekte ve yeni bir hibrit GAN mimarisi önerilmektedir. Klasik ve önerilen mimarilerin performans karşılaştırmaları MNIST [38] ve Fashion MNIST [39] veri kümeleri üzerinde gerçekleştirilmektedir. Yapılan bilimsel çalışma, “Çekişmeli üretken ağ modellerinin görüntü üretme performanslarının incelenmesi” başlığı altında bir makaleye dönüştürülerek Balıkesir Üniversitesi Fen Bilimleri Enstitüsü dergisinde yayınlanmıştır [40].

Üçüncü bölümde, Camelyon17 [41] veri kümesindeki histopatolojik görüntülerin çözünürlük artışı ve gürültü eliminasyonu amacıyla SRGAN mimarisinin kullanılmasına dayalı bir çalışma gerçekleştirilmiştir. Geleneksel yaklaşımlara kıyasen ümit verici sonuçların elde edildiği bu çalışma, “Resizing and cleaning of histopathological images using generative adversarial networks” başlıklı bir makede şekline dönüştürülerek “Physica A: Statistical Mechanics and its Applications” adlı bilimsel dergide yayınlanmıştır [42].

Dördüncü bölümde, EEG sinyallerinden [43] görüntü üretimi hakkında GAN temelli özgün bir çalışma önerilmiştir. Klasik GAN mimarileri girişine gürültü sinyalini alırken, önerilen mimari EEG sinyal ortalaması ve standart sapmasına dayalı bir girdiyi kullanmaktadır. Görsel üretim performansının önerilen mimariyle %5-8 oranında arttığı görülmüştür. Bu bölümde yapılan çalışma “EEG sinyallerinden bakılan görselin üretilmesi” başlığı ile Pamukkale Üniversitesi Mühendislik Bilimleri dergisinde yayınlanmıştır [44].

Beşinci bölümde, 3D- Beyin MRI verilerinin bölütlenmesi için yeni bir GAN tabanlı mimari (Vol2SegGAN) önerilmiştir. Klasik mimarilerle kıyaslandığında daha az parametreye sahip olan bu mimari, doğruluğu düşürmeden hızlı bir bölütleme sağladığı görülmektedir. Bu çalışmada yaygın olarak bilinen IBSR18 [45], MRBRAINS13 [46] ve MRBRAINS18 [47] veri kümeleri kullanılmıştır. Deneysel çalışmalarla elde edilen çıktılar bir makede formatına dönüştürülmüş ve yayınlanması için Biomedical Signal Processing and Control dergisine gönderilmiştir. Çalışma sürecinde geliştirilen algoritmalara ait kodlar Github (<https://github.com/GaffariCelik/Vol2SegGAN>) paylaşım platformuyla akademik paydaşların kullanımına açılmıştır.

Altıncı sonuç bölümünde, tez sürecinde yapılan çalışmalar genel olarak değerlendirilmiş ve sonraki çalışmalara ışık tutacak fikirler ele alınmıştır.

## 2. SENTETİK GÖRÜNTÜ ÜRETİMİ

Bu bölümde, literatürdeki mevcut GAN mimarilerinin anlaşılması, görüntü üretim performanslarının gözlemlenmesi ve birbirlerine göre üstünlük/eksikliklerin ortaya çıkarılması için yapılan deneysel çalışmalar ifade edilmektedir. İlk olarak klasik GAN mimarilerinden bahsedilmektedir. Daha sonra önerilen yeni hibrit bir GAN mimarisi tanıtılmaktadır. Sentetik görüntülerin üretilmesinde iki farklı gerçek veri kümesi (MNIST ve Fashion-MNIST) kullanılmıştır. Mimari performanslarının değerlendirilmesinde FID ve CNN tabanlı benzerlik metrikleri kullanılmıştır.

### 2.1 Klasik GAN mimarileri

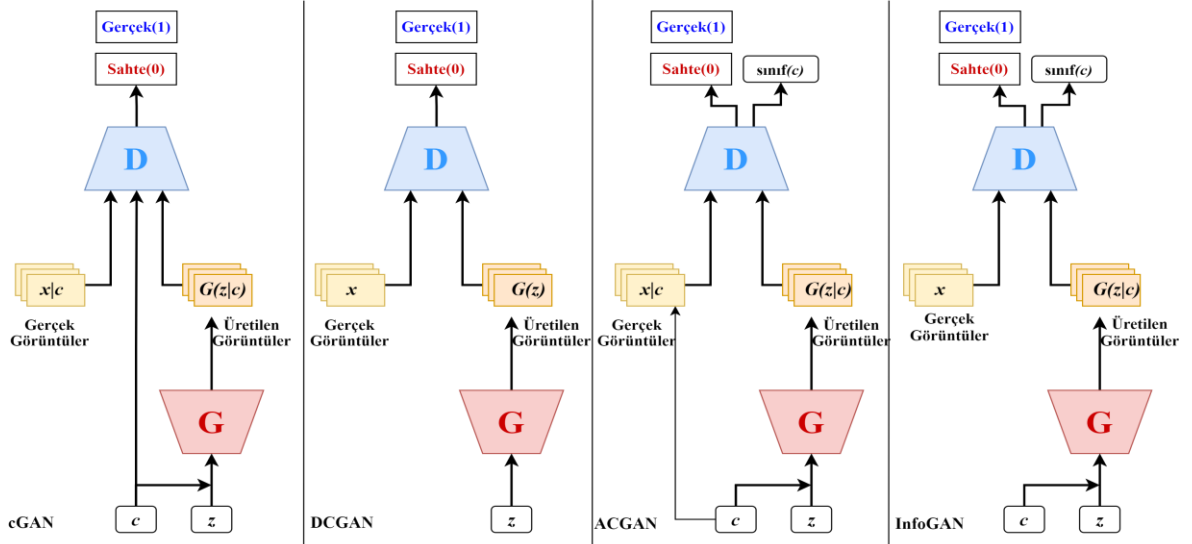
Bu bölümde cGAN, DCGAN, InfoGAN, SGAN, ACGAN, WGAN-GP ve LSGAN mimarileri hakkında özet bilgiler yer almaktadır.

#### 2.1.1 cGAN

Koşullu GAN (Conditional GAN-cGAN) mimarisi Şekil 2.1’de gösterilmektedir. Bu mimarinin farkı,  $G$  ve  $D$  ağlarına sınıf etiket bilgisini ( $c$ ) giriş olarak almasıdır. Bu şekilde  $D$ , gerçek veya sahte görüntülerin hangi sınıfa ait olduğunu öğrenmektedir. Aynı şekilde  $G$  ağı gürültü vektörü ile birlikte sınıf bilgisine göre yeni görüntüler üretmektedir [48]. Bu şekilde eğitilen mimarinin amaç fonksiyonu aşağıdaki gibi verilmiştir.

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}} [\log D(x|c)] + \mathbb{E}_{z \sim p_z} [\log(1 - D(G(z|c)))] \quad (2.1)$$

$D(x|c)$  ve  $G(z|c)$  sırasıyla koşullu ayırıcı ve üretici fonksiyonlardır.



Şekil 2.1 : Klasik GAN mimarileri.

### 2.1.2 DCGAN

Derin konvolüsyonel GAN (Deep Convolutional GAN – DCGAN) [49], Radford ve arkadaşları tarafından önerilmiştir. Şekil 2.1’de mimarisi gösterilen DCGAN yaklaşımında yapılan güncellemeler aşağıda listelenmektedir [40]:

- $D$  ve  $G$  mimarilerinde havuzlama işlemi (pooling) kullanılmamaktadır.  $D$ ’de görüntü daraltma işlemi konvolüsyon işlemiyle yapılmıştır.
- $D$  ve  $G$  mimarilerinde toplu normalleştirme işlemi kullanılmıştır.
- Daha derin mimariler için birbirine bağlı gizli katmanlar kullanılmamaktadır.
- RELU aktivasyon fonksiyonu,  $G$ ’nin çıkış katmanı haricinde diğer katmaların hepsinde kullanılmıştır.
- LeakyReLU aktivasyon fonksiyonu,  $D$ ’nin tüm katmanlarında kullanılmıştır.

[50] numaralı çalışmada yapılan güncellemelerle DCGAN mimarisinin daha kaliteli görüntüler üretebildiği ifade edilmektedir.

### 2.1.3 ACGAN

Yardımcı sınıflandırıcı GAN (Auxiliary classifier GAN – ACGAN) mimarisinde farklı olarak hem  $D$  hem de  $G$  ağlarında log-olabilirlik dağılımı kullanılmıştır.  $G$ , giriş olarak  $z$  gürültü vektörü ile birlikte  $c$  sınıf etiket bilgisi olarak sahte görüntü ( $X_{sahte} = G(z, c)$ ) üretir. ACGAN mimarisi Şekil 2.1 (c)’de verilmiştir.  $G$  ve  $D$ ’nin eğitimi için aşağıdaki maliyet fonksiyonu kullanılır.

$$L_S = E[\log P(S = ger\ccek | X_{ger\ccek})] + E[\log P(S = sahte | X_{sahte})] \quad (2.2)$$

$$L_C = E[\log P(C = c | X_{ger\ccek})] + E[\log P(C = c | X_{sahte})] \quad (2.3)$$

$L_S$ , görüntüler üzerindeki log-olabilirlik dağılımını,  $L_C$  ise sınıflar üzerindeki log-olabilirlik dağılımını ifade etmektedir.  $G$ ,  $L_C + L_S$ 'den,  $D$  ise  $L_C - L_S$ 'den en yüksek değer elde etmek için eğitilir [51].

#### 2.1.4 InfoGAN

InfoGAN (Bilgi GAN) mimarisi Şekil 2.1'de gösterildiği gibi,  $G$ 'nin girişine verilen gürültü vektörüne ek olarak gizli kod veya ek bilgi ( $c$ ) eklenir. Bu şekilde  $G$ , gürültü sinyali ( $z$ ) ve  $c$  bilgisine bağlı olarak örnekler üretir.  $G$ 'de,  $G(z)$  fonksiyonu yerine  $G(z, c)$  fonksiyonu kullanılır. Amaç fonksiyonu  $c$  ek bilgisine bağlı olarak, üretilen görüntü arasında en yüksek karşılıklı bilgiyi ( $I$ ) elde etmeye yönelik bir yöntem kullanılmaktadır [52]:

$$\min_G \max_D V_{infoGAN} = V_{GAN}(D, G) - \lambda I(c; G(z, c)) \quad (2.4)$$

$$V_{GAN}(D, G) \equiv \mathbb{E}_{x \sim p_{data}} [\log D(x|c)] + \mathbb{E}_{x \sim p_z} [\log(1 - D(G(z|c)))] \quad (2.5)$$

Geleneksel GAN modelinde  $G$ , ek bilgi ( $c$ )'ye bağlı olarak istenilen bilgiye bağlı görüntüyü üretemezler. Böyle bir problemin çözümü için,  $G$ 'nin üreteceği görüntü ( $G(z, c)$ ) ile  $c$  arasında karşılıklı bilgi ( $I$ )'ye ihtiyaç vardır. Bundan dolayı,  $I(c; G(z, c))$  değerinin yüksek çıkması beklenir.  $I$ ,  $entropi(H)$  kullanarak bulunur [52].

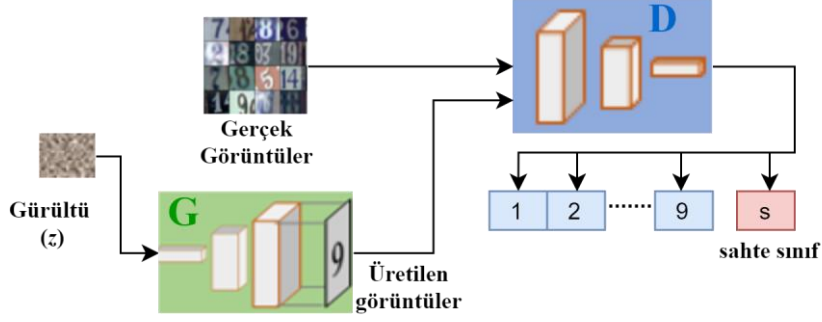
$$I(c; G(z, c)) = H(c) - H(c|G(z, c)) \quad (2.6)$$

Burada,  $I(c; x)$ ,  $c$  biliniyorsa,  $x$ 'i ne kadar bildiğimiz bilgisini ölçer.  $I(c; x)$  0 çıkması,  $x$  ile tahmin edilen  $c$ 'nin tamamen alakasız olduğunu gösterir.  $D$ ,  $c$ 'yi doğru tahmin etmesi,  $I$  değerinin yüksek çıkmasına, bu da maliyeti düşmesini sağlayacaktır. Basit gizli kodlar için Hiper-parametresi ( $\lambda$ )'ne 1 değeri verilmelidir.

#### 2.1.5 SGAN

Klasik GAN mimarilerinde  $D$ , tek bir sigmoid fonksiyonu ile biten yani giriş olarak aldığı görüntüden bir olasılık değeri üreten bir ağıdır. Ancak SGAN (Yarı eğiticili GAN)

mimarisinin çıkış katmanı, gerçek ve sahte görüntüler için sınıf bilgisi üreten bir softmax katmanı kullanır [53]. SGAN mimarisi Şekil 2.2’de verilmiştir [40].



Şekil 2.2 : SGAN mimarisi [54].

### 2.1.6 WGAN-GP

Arjovsky ve arkadaşları [55] tarafından önerilen WGAN-GP (Wasserstein GAN’ların geliştirilmiş eğitimi) mimarisi  $D$  ve  $G$  derin ağlarında maliyet fonksiyonları aşağıdaki gibi güncellenmiştir.  $D$ , olasılık mesafesini hesaplamak için Earth-Mover (EM) metriği kullanılmaktadır. Aynı zamanda sigmoid katmanı yerine ağırlık kırpma yönteminin kullanılmakta ve maliyet fonksiyonunda log olabilirlik dağılımı kullanılmaması dikkat çekici güncellemeler olarak karşımıza çıkmaktadır [55]:

$$L_D^{WGAN} = E[D(x)] - E[D(G(z))] \quad (2.7)$$

$$L_G^{WGAN} = E[D(G(z))] \quad (2.8)$$

WGAN ile bazen kötü örnekler üretildiği görülmüştür [56]. Bu durumun mimaride kullanılan ağırlık kırpma fonksiyonunun neden olduğu görülmüştür. Bundan dolayı ağırlık kırpma yöntemine karşılık olarak yeni bir mimari WGAN-GP geliştirilmiştir. Buna göre maliyet fonksiyonu aşağıdaki şekilde güncellenmiştir:

$$L_D = \lambda E[|\nabla D(ax - (1 - aG(z)))| - 1]^2 \quad (2.9)$$

$$L_D^{WGAN-GP} = L_D^{WGAN} + L_D \quad (2.10)$$

$$L_G^{WGAN-GP} = L_G^{WGAN} \quad (2.11)$$

Burada,  $a$  ve  $\lambda$  hiper parametrelerin varsayılan değerleri olarak  $a = 0.0001$ ,  $\lambda=10$  verilmelidir [40].

### 2.1.7 LSGAN

$D$  mimarisini bir sınıflandırıcı olarak gören GAN mimarilerinde hata fonksiyonu olarak Sigmoid Cross Entropy (SCE) fonksiyonu kullanılmaktadır. Bu fonksiyonla  $G$  güncellenme esnasında bazı örneklerde gradyan bilgisinin yok olduğu görülmüştür. Böyle bir soruna çözüm olarak SCE'ye alternatif olarak en az kareler yaklaşımı önerilmiştir [57]. Buna göre LSGAN (En az kareler GAN) maliyet fonksiyonları aşağıdaki gibi güncellenmektedir.

$$\min_D V_{LSGAN}(D) = \frac{1}{2} \mathbb{E}_{x \sim p_{data}(x)} [(D(x) - b)^2] + \frac{1}{2} \mathbb{E}_{z \sim p_z(z)} [(D(G(z)) - a)^2] \quad (2.12)$$

$$\min_G V_{LSGAN}(G) = \frac{1}{2} \mathbb{E}_{z \sim p_z(z)} [(D(G(z)) - c)^2] \quad (2.13)$$

Burada  $b$  ve  $a$ , gerçek ve sahte veriler için kullanılan etiketler,  $c$  ise, üretici ağı ayırıcı ağı sahte verilerle kandırmak için gerekli değeri göstermektedir [57].

### 2.2 Önerilen GAN mimarisi

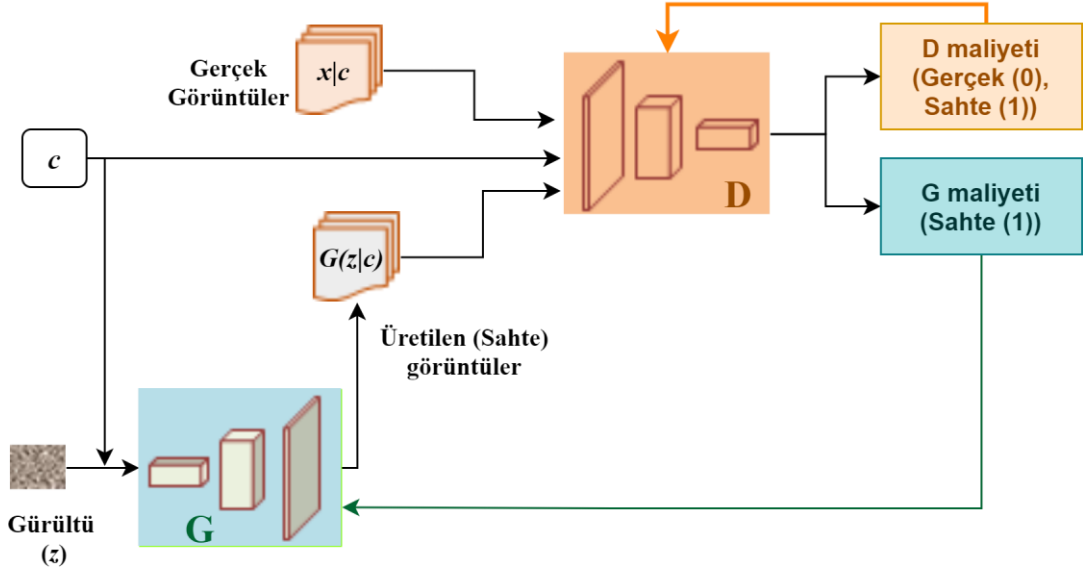
cGAN ve DCGAN mimarilerinin avantajlarını birleştiren hibrit bir GAN mimarisi önerilmiştir [40]. Bu mimari Koşullu derin konvolüsyonel GAN (cDCGAN) olarak isimlendirilmiştir. cGAN ile üretilen görüntüler sınıf bilgisine göre sınıflandırılabilen ancak üretilen görüntülerin düşük çözünürlükte olduğu görülmüştür. DCGAN mimarisinde ise cGAN'nın aksine daha kaliteli görsellerin üretildiği, bu karşılık sınıf bilgisinin göz ardı edildiği görülmektedir. Her iki mimarinin avantaj ve dezavantajları göz önünde bulundurularak yeni bir mimari geliştirilmiştir. Önerilen bu yöntemle sınıf bilgisiyle birlikte yüksek çözünürlükte görsellerin üretilmesi amaçlanmıştır. Önerilen mimari Şekil 2.3'te gösterilmektedir. Bu mimarinin maliyet fonksiyonu aşağıdaki gibidir:

$$\max_D V_{cDCGAN}(D) = \mathbb{E}_{x \sim p_{data}} [\log D(x|c)] + \mathbb{E}_{x \sim p_z} [\log(1 - D(G(z|c)))] \quad (2.14)$$

$$\min_G V_{cDCGAN}(G) = \mathbb{E}_{x \sim p_z} [\log(1 - D(G(z|c)))] \quad (2.15)$$

Burada  $c$ , sınıf etiketi (ek bilgi) bilgisini ifade etmektedir.  $D(x|c)$ ,  $c$  etiketi ile verilen görüntüyü ayırt edilmesini,  $G(z|c)$  ise, bu etikette görüntü üretilmesi için gerekli fonksiyonu göstermektedir. Bu mimariye aynı zamanda DCGAN modelinde yapılan kısıtlamalar eklenmiştir [40].





Şekil 2.3 : cDCGAN mimarisi [40].

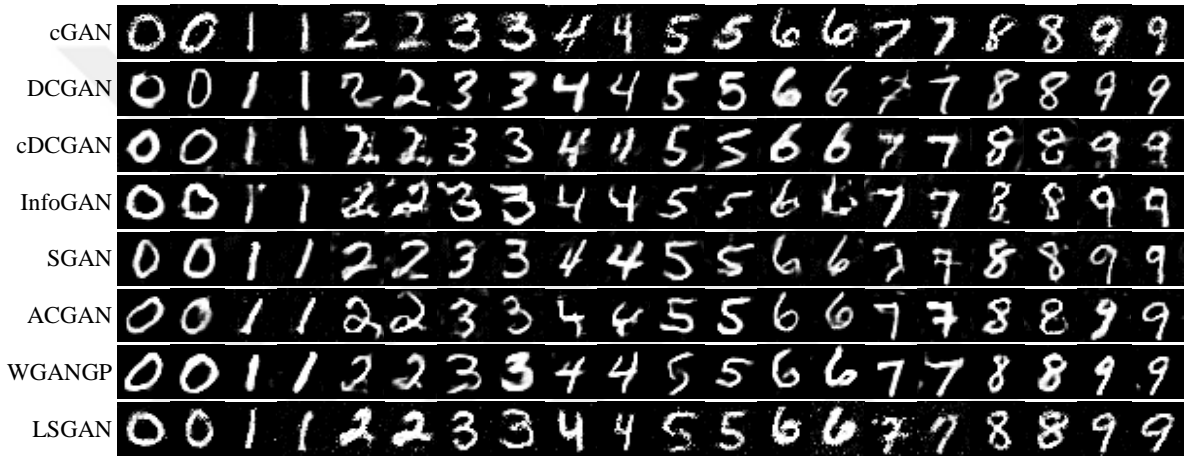
### 2.3 Deneysel Çalışmalar ve Sonuçları

Bu bölümde GAN modellerinin görüntü üretme performanslarını incelemek için MNIST ve Fashion-MNIST veri kümeleri kullanılmıştır. MNIST veri kümesi 0-9 arasındaki rakamları içeren 60.000 el yazması görseli içermektedir. Fashion-MNIST veri kümesi ise 10 farklı giyim eşyasından oluşan 60.000 görseli içermektedir. Veri kümelerindeki her sınıfta yaklaşık 6.000 görüntü bulunmaktadır. Modellerin performanslarını analiz etmek için FID benzerlik metriği ve CNN tabanlı bir sınıflandırma yöntemi kullanılmıştır [40]. FID benzerlik metriği aşağıda ifade edilmektedir:

$$FID(x, g) = \left\| \mu_x - \mu_g \right\|_2^2 + Tr \left( C_x + C_g - 2(C_x C_g)^{\frac{1}{2}} \right) \quad (2.16)$$

Burada \$(\mu\_g, C\_g)\$ ve \$(\mu\_x, C\_x)\$ sırayla üretilen (sentetik) görüntüler ile gerçek görüntülerin ortalama ve kovaryans değerlerini ifade etmektedir [58]. FID metriği 0'a yakın çıkması üretilen görüntünün gerçek görüntüye çok benzediğini, 1'e yakın çıkması ise üretilen görüntünün gerçek görüntüye benzemediği anlamına gelmektedir. CNN yaklaşımında, MNIST ve Fashion-MNIST veri kümelerinin her biri için ayrı bir CNN ağı eğitilmiştir (doğruluk \$\cong\$ %99,1) gerçekleştirilmiştir. Buna göre GAN mimarileriyle üretilen bir görüntünün eğitilmiş CNN ağındaki sınıflandırma doğruluğu üretim kalitesini belirlemektedir. Sınıflandırma doğruluğunun yüksek olması görselin kaliteli olduğu anlamına gelmektedir [40].

GAN modellerin performansları, kullanılan veri kümeleri üzerinde sınıf bazlı ve toplu (tüm sınıflar) bir şekilde farklı iki uygulama ile analiz edilmiştir. Her iki uygulamada optimizasyon yöntemi olarak adadelta ( $\rho=0.95$ ,  $\text{lr}=1.0$ ) kullanılmıştır. Rastgele dereceli inişin (stokastik gradient descent) farklı olarak minimum hesaplama maliyetine sahip olan bu optimizasyon yöntemi, zamanla sadece birinci dereceden bilgiyi kullanır [59]. Modellerin eğitimi için veri kümeleri 128 örnekli (batch) alınarak yapılmıştır. Yapılan ilk uygulamada, GAN modellerinin sadece MNIST görüntülerini üretebilme başarıları incelenmiştir. Bu uygulamada eğitim süresi sabit tutulmuş (12 dk.) ve her sınıf için GAN modelleri ayrı ayrı eğitilmiştir. Eğitim sonrası her modelin her bir sınıf için 100 görüntü üretilmesi sağlanmıştır. Üretilen görsellerden bazıları Şekil 2.4'te gösterilmektedir.



Şekil 2.4 : GAN mimarileri ile üretilen örnek MNIST görüntüleri.

Üretilen 100 adet görüntüyle, eğitim kümesindeki 100 adet gerçek görüntü arasındaki FID benzerlik oranları Çizelge 2.1 (sol)'da sunulmuştur. Elde edilen sonuçlarının daha iyi anlaşılabilmesi için tablonun son satırında gerçek görüntülerin (100 adet) FID benzerlik değerleri verilmiştir. Bu satırdaki FID benzerlik oranının yüksek çıkması veri kümesindeki veri çeşitliliğini göstermektedir. Normalde FID oranının düşük çıkması, görüntüler arasındaki benzerliğin yüksek olduğunu gösterir. Ancak gerçek görüntülerin bile kendi aralarında bir benzerlik hatası verdiği görülmektedir. Bu hata değeri (ofset) genel FID sonuçlarından çıkarılarak elde edilen sonuçlar Çizege 2.1 (sağ)'da verilmiştir. Bulunan sonuçlarda en yüksek değerler modelin gerçek görüntüye en yakın görüntüyü ürettiği yargısına varılmıştır. Elde edilen sonuçlara göre, LSGAN 1, 4, 7, 9 sınıflarında, cGAN 2, 8 sınıflarında; cGAN ve LSGAN 3, 5 sınıflarında, InfoGAN ve cGAN 0 sınıfında en benzer görüntüleri ürettiği görülmektedir.

**Çizelge 2.1 :** Üretilen görüntülerin FID metriğine göre benzerlik sonuçları.

	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9
cGAN	4.60	4.11	4.00	3.00	3.28	3.13	3.54	3.89	3.43	3.28	<b>0,53</b>	0,29	<b>0,9</b>	<b>0,36</b>	0,64	<b>0,27</b>	0,04	0,36	<b>0,36</b>	0,26
DCGAN	3.95	3.91	2.93	2.85	2.75	2.84	2.84	3.55	2.98	2.87	0,12	0,09	0,17	0,21	0,11	0,02	<b>0,74</b>	0,02	0,09	0,15
cDCGAN	3.92	3.70	3.11	2.62	3.52	2.72	3.41	4.23	2.84	3.05	0,15	0,12	0,01	0,02	0,88	0,14	0,17	0,7	0,23	0,03
InfoGAN	3.54	3.46	2.94	2.48	2.69	2.61	3.17	3.63	2.86	2.61	<b>0,53</b>	0,36	0,16	0,16	0,05	0,25	0,41	0,1	0,21	0,41
SGAN	3.65	3.67	2.82	2.60	2.65	2.72	3.27	3.58	2.76	2.75	0,42	0,15	0,28	0,04	0,01	0,14	0,31	0,05	0,31	0,27
ACGAN	4.11	3.87	2.91	2.49	2.95	2.72	3.39	3.37	3.00	2.88	0,04	0,05	0,19	0,15	0,31	0,14	0,19	0,16	0,07	0,14
WGAN-GP	3.90	4.09	2.92	2.81	3.09	2.92	3.31	3.63	2.92	3.18	0,17	0,27	0,18	0,17	0,45	0,06	0,27	0,1	0,15	0,16
LSGAN	4.09	4.48	3.20	3.00	3.58	3.13	3.63	4.33	3.26	3.56	0,02	<b>0,66</b>	0,1	<b>0,36</b>	<b>0,94</b>	<b>0,27</b>	0,05	<b>0,8</b>	0,19	<b>0,54</b>
MNIST	4.07	3.82	3.10	2.64	2.64	2.86	3.58	3.53	3.07	3.02	0	0	0	0	0	0	0	0	0	0

FID benzerlik metriği ile bulunan sonuçlarının doğruluğunu test etmek için CNN tabanlı bir yöntem kullanılmıştır. Üretilen görüntüler CNN ile sınıflandırılmış ve modellerin sınıflandırma performansları Çizelge 2.2’de sunulmuştur. Başarı oranı Denklem 2.17’ye göre hesaplanmıştır:

$$Başarı(GAN_i) = \left(1 - \frac{Hata(GAN_i)}{\sum_j Hata(GAN_j)}\right) * 100 \quad (2.17)$$

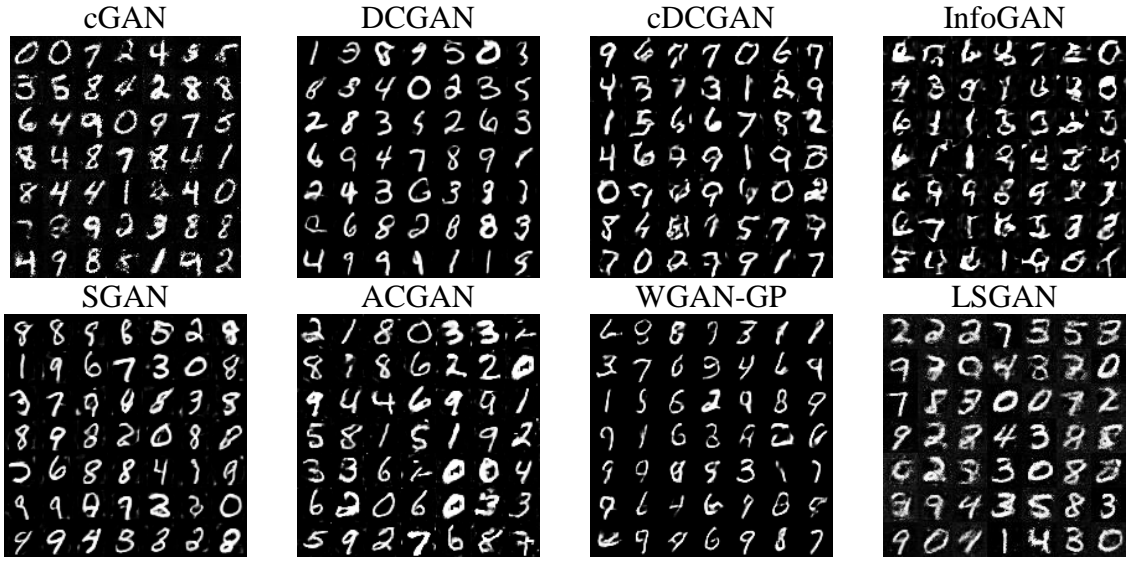
Çizelge 2.2’de 0-9 sütunlarındaki rakamlar üretilen görüntülerin sınıflandırma işleminde başarısız bir şekilde sınıflandırılan görüntülerin sayısını göstermektedir. Çizelge 2.2’deki sonuçlara göre, LSGAN modeli ile üretilen görüntülerde 8 sınıfı hatasız (%100) sınıflandırılmıştır. cDCGAN modeli ile 5 sınıfta üretilen görüntüler hatasız bir şekilde sınıflandırılmış ve bazı sınıflarda (8 gibi) ise diğer modellerden daha iyi performans sergilediği görülmüştür.

**Çizelge 2.2 :** GAN mimarilerinin CNN yöntemine göre başarıları.

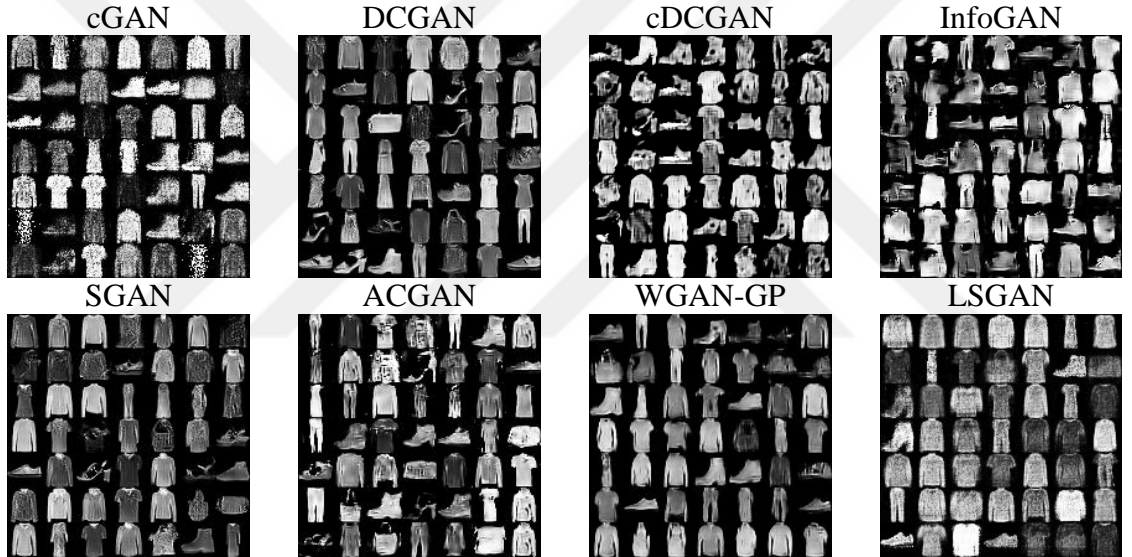
	0	1	2	3	4	5	6	7	8	9	Hata	Başarı
cGAN	<b>0</b>	<b>0</b>	3	2	2	<b>0</b>	2	<b>0</b>	3	2	14	90,60
DCGAN	<b>0</b>	<b>0</b>	<b>0</b>	3	<b>0</b>	2	<b>0</b>	1	2	1	9	93,96
cDCGAN	<b>0</b>	<b>0</b>	2	4	<b>0</b>	3	<b>0</b>	5	<b>0</b>	15	29	80,54
InfoGAN	<b>0</b>	<b>0</b>	4	1	3	2	2	1	15	12	40	73,15
SGAN	<b>0</b>	1	1	2	4	<b>0</b>	1	3	2	6	20	86,58
ACGAN	1	<b>0</b>	1	2	2	1	<b>0</b>	1	<b>0</b>	4	12	91,95
WGAN-GP	1	<b>0</b>	2	1	4	2	1	3	2	5	21	85,91
LSGAN	<b>0</b>	<b>0</b>	2	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	2	<b>0</b>	4	<b>97,32</b>
Toplam Hata											149	

Bu bölümde yapılan ikinci uygulamada modellerin başarıları, veri kümeleri üzerinde toplu eğitim yapılarak performansları incelenmiştir. Önceki uygulamada modellerin eğitimi her

bir sınıf için ayrı yapılırken, bu uygulamada veri kümeleri sınıflara ayrılmadan tüm sınıflar toplu olarak eğitilmiştir. Eğitim işlemi, hem MNIST hem de Fashion-MNIST veri kümeleri üzerinde ayrı ayrı iki farklı şekilde gerçekleştirilmiştir. Modeller her veri kümesi için toplamda 90 dk. eğitilmişlerdir. Eğitim sonrası üretilen örnek görüntüler Şekil 2.5 ve Şekil 2.6'da verilmiştir. Şekil 2.5'deki örnek görüntülere bakıldığında, cGAN, LSGAN ve InfoGAN'ın gürültülü görüntüler ürettiği, DCGAN, WGAN-GP ve SGAN modellerin ürettiği görüntülerin daha kaliteli olduğu görülmektedir. Ancak üretilen görüntülerin FID benzerlikleri ve CNN tabanlı sınıflandırma sonuçları incelendiğinde LSGAN modelinin diğer modellere kıyasla daha başarılı sonuçlar verdiği görülmektedir. Şekil 2.6'da görüldüğü gibi, modellerin MNIST görüntülerini üretme performansları Fashion-MNIST sonuçlarıyla paralellik göstermektedir. DCGAN ve WGAN-GP modelleri tarafından üretilen görüntüler daha kaliteli iken, diğer GAN modellerinde daha gürültülü ve net olmayan görüntüler üretilmiştir.



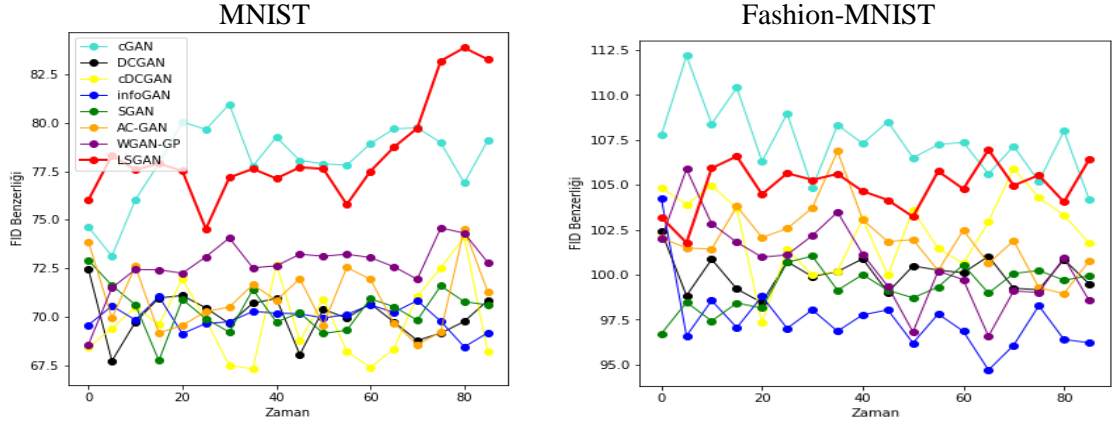
Şekil 2.5 : Üretilen örnek MNIST görüntüleri.



Şekil 2.6 : Üretilen örnek Fashion-MNIST görüntüleri.

Eğitim süresinin model performansına etkisini incelemek için modeller 90 dk. eğitilmiştir. Eğitim süresi boyunca 5 dk.'lık periyotlarda toplam 18 zaman dilimi için FID benzerlik oranları bulunmuştur. Her bir zaman diliminde kaydedilen ağırlıklar kullanılarak görüntüler üretilmiştir. Belirtilen periyotlarda üretilen imgelerin zaman ve FID benzerlik değerleri Şekil 2.7'de sunulmuştur. Şekil 2.7 (sol)'de MNIST veri kümesi, Şekil 2.7 (sağ)'de Fashion-MNIST veri kümesi kullanılarak gerçekleştirilen eğitim sonucunda üretilen görüntülerin FID benzerlik sonuçlarını göstermektedir. Şekil 2.7 (sol)'deki sonuçlar incelendiğinde LSGAN mimarisi ile üretilen görüntülerin FID benzerlik değerleri zamanla daha dengeli bir şekilde arttığı, 90 dk. sonunda en yüksek benzerlik değerini elde ettiği görülmektedir.

Benzer şekilde Şekil 2.7 (sağ)'deki grafiğin sonuçları incelediğinde LSGAN modelinin 18 zaman dilim sonunda en yüksek benzerlik oranı elde ettiği görülmüştür [40].



Şekil 2.7 : Modellerin FID benzerlik metriğine göre zamansal olarak toplu görüntü üretme performansları.

### 3. HİSTOPATOLOJİK GÖRÜNTÜLERİN ONARIMI VE TEMİZLENMESİ

Görüntü ve video işleme alanında düşük çözünürlükte (LR) veya gürültülü görüntüler üzerinde işlem yapmak önemli bir problemdir [60]. Bu durum nesne algılama, bölütleme, modelleme, takip gibi algoritmaların performansını olumsuz yönde etkilemektedir. Medikal uygulamalarda ise hastalık teşhisinde ve doğru tanı koyulabilmesinde düşük çözünürlüklü görüntüler uzmanları olumsuz yönde etkilemektedir.

Yüksek maliyetli donanımsal cihazlar kullanılarak yüksek çözünürlükte görüntüler elde edilebilmektedir. Bunun yanında düşük maliyetli görüntüleme cihazlarından elde edilen düşük çözünürlükteki imgeler belirli algoritmalar yardımıyla büyütülebilir. Böylece donanım maliyetinden önemli ölçüde tasarruf edilebilir [42].

Görüntü çözünürlüğünün artırılması alanında birçok yöntem bulunmaktadır [61]. Piksel çoğullama, en yakın komşu, Lanczos [62], Cubic [63], bilinear ve bicubic interpolasyon [64,65] teknikleri bunlardan birkaçıdır. Son yıllarda derin öğrenme yaklaşımlarının gelişmesiyle çözünürlüğün artırılması alanında güncel yaklaşımlar önerilmiştir [66].

#### 3.1 Çözünürlük Arttırmada Kullanılan Klasik Yaklaşımlar

##### 3.1.1 Bilinear interpolasyon

Bilinear interpolasyon tekniği yardımıyla imge boyutları istenildiği kadar büyütülebilir. Küçük bir imgenin büyütülmesi işleminde aşağıdaki doğrusal denklem kullanılır.

$$u(x, y) = ax + by + cxy + d \quad (3.1)$$

$u(x, y)$  piksel renk değerini,  $x$  ve  $y$  piksel konumu bilgisini ifade eder. Toplamda dört farklı piksel için dört denklem elde edilerek denklem sisteminin cebirsel çözümü yapılır ve  $(a, b, c, d)$  katsayıları hesaplanır. Böylece herhangi bir  $x$  ve  $y$  konumundaki yeni renk değeri hesaplanabilir.

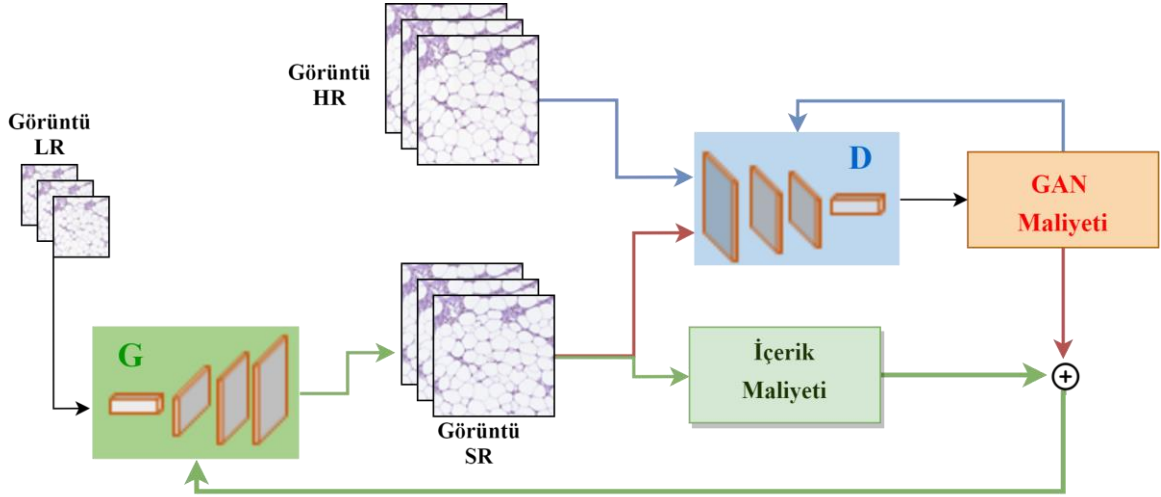
### 3.1.2 Bicubic interpolasyon

Bicubic interpolasyon yönteminde ilk olarak görüntünün yatay, dikey ve çapraz türevleri elde edilir. Türevlerle birlikte toplam dört farklı görüntü ortaya çıkar. Sonraki aşamada, dört farklı piksel noktasının her bir görüntüdeki renk değerleri kullanılarak 16 bilinmeyenli 16 denklem üretilir [64]. Denklem sisteminin cebirsel çözümü sonrasında bilinmeyen 16 katsayı ( $a_{i,j}$ ) hesaplanır. Son olarak istenilen  $(x, y)$  koordinatındaki renk değeri aşağıdaki denklem kullanılarak hesaplanır [65].

$$P(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 a_{ij} x^i y^j \quad (3.2)$$

### 3.2 Süper Çözünürlüklü Çekişmeli Üretken Ağ (SRGAN)

SRGAN [24], yüksek çözünürlüklü görüntüler üretmek için GAN mimarisini baz alır. Eğitim işlemine başlamadan önce yüksek çözünürlükteki görüntülerin (HR) düşük çözünürlükteki (LR) karşılıkları elde edilir.  $G$ , giriş olarak LR görüntülerini alır ve süper çözünürlüklü görüntüler (SR) üretir.  $D$ , HR ve SR görüntülerini ayırt etmek için kullanılır.  $G$  ve  $D$ 'yi eğitmek için GAN maliyeti geriye yayılır. SRGAN mimarisi Şekil 3.1'de gösterilmektedir.



Şekil 3.1 : SRGAN Mimarisi.

SRGAN mimarisini klasik GAN mimarilerinden ayıran özellik  $G$  ağındaki ağırlıkların güncelleme mantığıdır. Klasik GAN mimarilerinde  $G$ 'nin  $D$ 'yi kandırabilmesi için  $G$  ağındaki ağırlıkların güncellenmesinde sadece çekişme maliyeti ( $\nabla_{\theta_{\text{çekişme}}}$ ) kullanılırken, SRGAN algoritmasında çekişme maliyetine ilaveten içerik maliyeti ( $\nabla_{\theta_{\text{içerik}}}$ ) eklenmektedir.



İçerik maliyeti,  $D$  ağına giriş olarak verilen  $I^{HR}$  ve  $I^{SR}$  görüntü çiftleri arasındaki algısal benzerliği ifade etmektedir. Görüntülerin algısal benzerlik değerlerinin hesaplanabilmesi için eğitilmiş VGG19 [67] evrimsel ağı kullanılmaktadır. Buna göre her iki görüntü VGG19 ağından süzülerek elde edilen yeni görüntüler arasındaki Öklid uzaklığı hesaplanır. SRGAN mimarisinin sözde kodu aşağıda verilmiştir.

**For iterasyon\_sayısı do**

*Düşük boyutlu  $I^{LR}$  den yüksek boyutlu  $I^{SR}$  elde et:  $I^{SR} = G(I^{LR})$*

*Yüksek boyutlu gerçek ( $I^{HR}$ ) ve sahte ( $I^{SR}$ ) görüntüyü  $D$  'ye ver.*

*$D$  'yi aşağıdaki gibi eğit (ağırlıkları güncelle):*

$$\nabla_{\theta_d} = \frac{1}{m} \sum_{i=1}^m [\log D(I^{HR}) + \log(1 - D(I^{SR}))]$$

*$G$  'yi aşağıdaki gibi eğit (ağırlıkları güncelle):*

$$\nabla_{\theta_g} = 10^{-3} * \nabla_{\theta_{\text{çekişme}}} + \nabla_{\theta_{\text{içerik}}}$$

$$\nabla_{\theta_{\text{çekişme}}} = \frac{1}{m} \sum_{i=1}^m -\log(D(I^{SR}))$$

$$\nabla_{\theta_{\text{içerik}}} = \|VGG19(I^{HR}) - VGG19(I^{SR})\|_2$$

**endFor**

### 3.3 Gürültü Temizleme Yöntemleri

#### 3.3.1 Ortanca filtre

Bir maskeyi görüntü üzerinde dolaştırırken, maskeye denk gelen pikselleri büyükten küçüğe doğru sıralanarak ortanca pikseli, maskeye denk gelen merkez piksele atama işlemi ile gerçekleşir [68]. Verilen bir  $I$  imgesi için medyan filtresi [69]:

$$I_{mf}(i,j) = \text{median}\{I(i+r, j+s), (r,s) \in W\} \quad (3.3)$$

Burada  $(i,j) \in (1,2, \dots, H) \times (1,2, \dots, L)$ ,  $H$  ve  $L$  verilen imgenin genişliği ve yüksekliğini verir.  $W$ , maskenin koordinatlarını gösterir.

#### 3.3.2 Ortalama filtre

Ortalama filtrede, görüntüdeki bir pikselin orijinal yoğunluk değeri, komşu piksellerin ortalamasıyla değiştirilir. Bu işlem gürültülü piksellerin yumuşamasına neden olur. Bunun

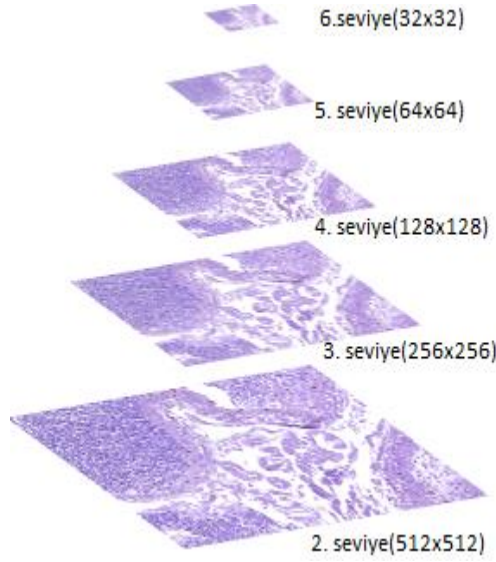
için 3x3, 5x5 gibi tek sayıdan oluşan maskelerin görüntüyle konvolüsyonu gerçekleştirilir. Maske boyutu yükseldikçe yumuşama etkisi artacağından bulanık bir görüntü ortaya çıkar [70].

### **3.3.3 Adaptif filtre**

Adaptif filtrenin ortalama ve medyan filtreden farkı görüntüdeki lokal içeriğe göre filtre katsayılarını otomatik ayarlamasıdır [68]. Buna göre görüntüde kenar yoğunluğunun fazla olduğu bir bölgeye geldiğinde filtre katsayılarını düşürür, kenar bilgisinin olmadığı düz bir bölgeye geldiğinde filtre katsayısını yükseltir. Böylece hem kenarları korumuş olur hem de ortalama ve ortanca filtreden daha güçlü gürültü temizleme işlemini yapar.

### **3.4 Veri Kümesinin Hazırlanması**

Bu bölümde Camelyon17 [41] veri kümesine ait görüntüler kullanılmıştır. Camelyon17, meme kanseriyle ilgili yaklaşık 3TB'lik histopatolojik görüntü içermektedir. Bu görüntüler Hollanda merkezli beş tıp merkezi (RUMC, UMCU, RST, CWZ ve LPON) tarafından dijital patoloji tekniği kullanılarak elde edilmiştir. Dijital patolojik görüntüleme işleminde bir yandan mikroskop yüksek çözünürlükte (piksel başına 160nm) görüntüler elde edilirken diğer yandan mikroskoba yerleştirilen preparat otomatik hareket ettirilmektedir. Böylece preparat üzerindeki tüm yüzeyin fotoğraflanması sağlanmış olur. Ardışık elde edilen görüntüler birleştirilerek devasa büyüklükteki (200.000x100.000) tek bir görüntü (whole slide image-WSI) elde edilir. Kaydedilen yüksek çözünürlükteki bu görüntü piramitin 0. seviye olarak düşünülür ve her bir üst seviye görüntü alt seviyenin iki katı küçüklüğünde olacak şekilde 6. seviyeye kadar görüntüler kaydedilmiştir. Herhangi bir seviyeden istenilen boyutta görüntü kopyalamak mümkündür (bak Şekil 3.2). Bu tür görüntüler ASAP, OpenSlide gibi özel amaçlı programlar yardımıyla görüntülenebilir [41,71]. Bu çalışmada Python programlama dili kullanılarak WSI görüntülerinden toplamda 5682 görüntü kopyalanarak küçük bir veri kümesi inşa edilmiştir.



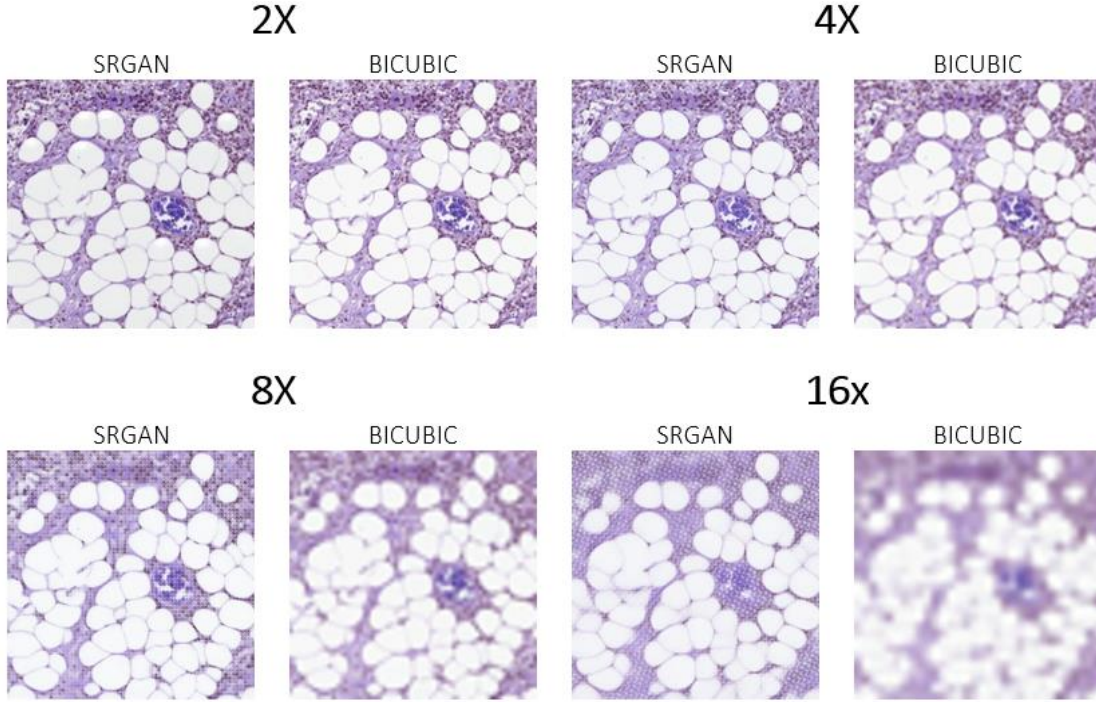
Şekil 3.2 : WSI görüntülerin piramit yapısı.

### 3.5 Deneysel Çalışmalar ve Sonuçları

Deneysel çalışmalar 20 çekirdekli 2 işlemciden oluşan bir CPU, NVIDIA QUARD 4000 GPU ve 64 GB ram bileşenlerine sahip bir sunucu bilgisayarda gerçekleştirilmiştir. Uygulamalar için Python ve Matlab programlama dilleri kullanılmıştır.

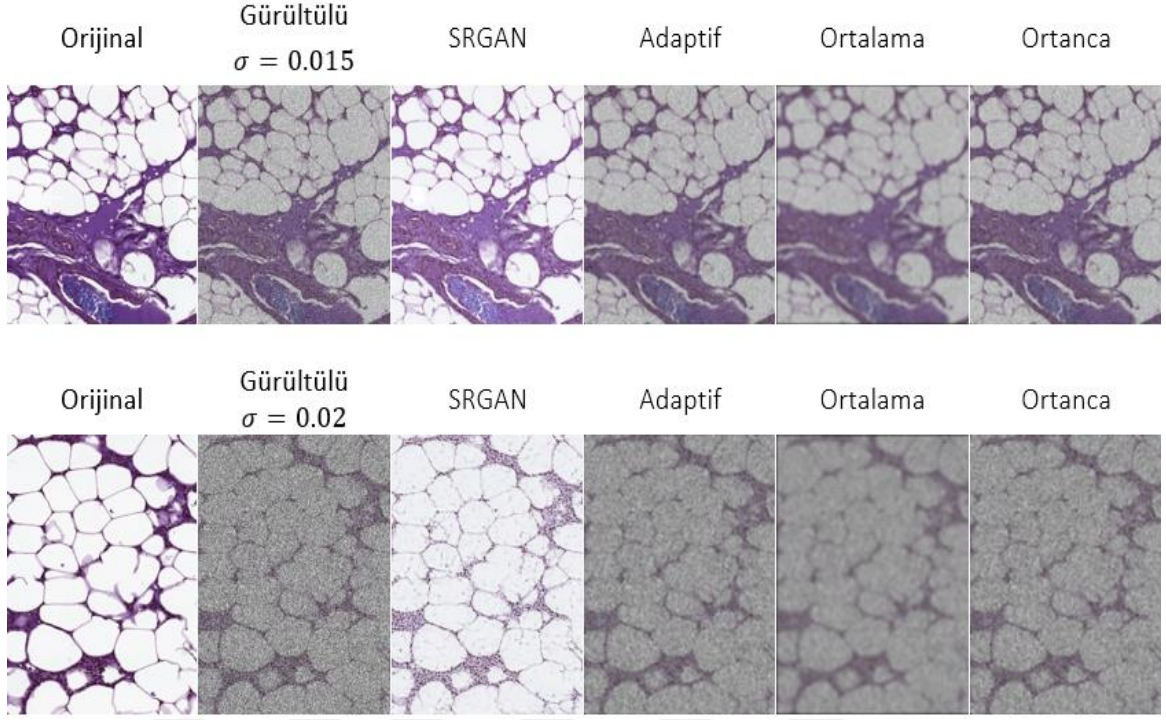
Birinci deneysel çalışmada görüntü büyütme alanındaki klasik yöntem (Bicubic) ile güncel yöntem (SRGAN)'in performansları gözlemlenmiştir. Bunun için ilk aşamada WSI görüntülerinin 2. seviyesinden 512x512 boyutlarında 5682 adet yüksek çözünürlüklü (HR) görüntü elde edilmiştir. İkinci aşamada her bir HR görüntü sırayla 2 kat (256x256), 4 kat (128x128), 8 kat (64x64), 16 kat (32x32) oranında küçültülerek düşük çözünürlükteki (LR) görüntüler elde edilmiştir. Üçüncü aşamada LR görüntüleri büyütme oranına göre algoritmalar yardımıyla büyütülüp orijinal görüntüye benzerlikleri hesaplanmıştır. SRGAN mimarisi her büyütme için 30.000 iterasyon eğitilmiştir. HR ve LR imgeleri  $[-1,1]$  aralığına ölçeklenerek kullanılmıştır. Optimizasyon algoritması için Adam ( $lr = 0.0002$ ,  $\beta_1=0.5$ ) [72] optimizasyon algoritması kullanılmıştır.  $G$  mimarisi için 16 adet özdeş residual blok [73], her blokta 64 özellik haritası,  $3 \times 3$ 'lük maske ve sıçrama adımı (stride) 1 olan evrişim katmanı, batch-normalizasyon ( $momentum = 0.8$ ) [74] katmanı ve RELU aktivasyon fonksiyonu kullanılmıştır. Aynı zamanda büyütme oranına bağlı olarak UpSampling katmanı ve bu katmandan sonra 256 özellik haritası,  $3 \times 3$  maske, sıçrama adımı 1 olan evrişim katmanı ve RELU aktivasyon fonksiyonu kullanılmıştır.  $D$  mimarisi için ise 8 adet evrişim katmanı, her evrişim katmanından sonra LeakyReLU ( $\alpha = 0.2$ ) aktivasyon fonksiyonu, vektörleştirme işlemi için denselayer ve son olarak sigmoid aktivasyon

fonksiyonu kullanılmıştır. Derin ağlarda, ekstra bir katman eklemek yüksek eğitim hatasına ve validation hatasının daha da artmasına neden olabilir. Residual blocks, konvolüsyon katmanların çıkışı ile toplanan kısayol bağlantıları eklenerek böyle problemleri çözer. Elde edilen sonuçlar Şekil 3.3'te gösterilmektedir. Sonuçlar incelendiğinde, büyütme oranının artmasıyla SRGAN sonucunun Bicubic'e olan üstünlüğü daha net anlaşılmaktadır. SRGAN büyütme sonucu daha net ve anlaşılır olmaktadır, Bicubic sonucu gittikçe daha bulanık bir şekle dönüşmektedir.



**Şekil 3.3 :** Görüntü büyütme örnekleri.

İkinci deneysel çalışmada SRGAN yönteminin gürültü eleme işleminde klasik yaklaşımlarla karşılaştırılması yapılmıştır. Ağ mimarisi olarak bir önceki deneysel çalışmanın büyütme katmanı hariç aynı mimari kullanılmıştır. Mimari 30.000 iterasyon eğitilmiştir. Bu deneyde WSI imgelerin 3. seviyesinden 256x256 boyutlarında görüntüler alınmıştır. Bu görüntülere varyansı  $\sigma = 0.015$  ve  $\sigma = 0.02$  olan gauss gürültüsü eklenerek klasik gürültü temizleme yaklaşımlarıyla SRGAN temizleme sonuçları elde edilmiştir (bak. Şekil 3.4). SRGAN'ın gürültü temizlemedeki performansının klasik yöntemlerden daha iyi olduğu gözlemlenmiştir.



**Şekil 3.4 :** Gürültü temizleme örnekleri.

Büyütme ve gürültü eleme sonuçlarının orijinal görüntüyle karşılaştırılabilmesi için üç farklı benzerlik metriği kullanılmıştır. Bunlar yapısal benzerlik metriği SSIM [75], sinyal-gürültü oranı PSNR [76] ve frekansın ortalama karesel hatası FFT\_MSE'dir. SSIM benzerlik katsayısı Denklem 3.4-3.7 arasında ifade edilmektedir. Gerçek görüntü  $hr$ , SRGAN sonucunda elde edilen görüntüye  $sr$  olarak ifade edilmiştir.

$$SSIM(hr, sr) = [I(hr, sr)]^\alpha \cdot [C(hr, sr)]^\beta \cdot [S(hr, sr)]^\gamma \quad (3.4)$$

$$I(hr, sr) = \frac{2\mu_{hr}\mu_{sr} + c_1}{\mu_{hr}^2 + \mu_{sr}^2 + c_1} \quad (3.5)$$

$$C(hr, sr) = \frac{2\sigma_{hr}\sigma_{sr} + c_1}{\sigma_{hr}^2 + \sigma_{sr}^2 + c_1} \quad (3.6)$$

$$S(hr, sr) = \frac{\sigma_{hrsr} + c_3}{\sigma_{hr}\sigma_{sr} + c_3} \quad (3.7)$$

$hr$  ve  $sr$  imgeleri için,  $\mu_{hr}, \mu_{sr}$  renk ortalamalarını,  $\sigma_{hr}, \sigma_{sr}$  standart sapmayı ve  $\sigma_{hrsr}$  kovaryansı ifade etmektedir.  $c_1, c_2, c_3$  katsayıları sıfıra bölme hatasını engellemek için küçük skaler değerlerdir. Her terimi ağırlıklandırmak  $\alpha, \beta, \gamma$  katsayıları kullanılır. PSNR [76,77] sinyal gürültü oranının hesaplanabilmesi için Denklem 3.8-3.9 kullanılmıştır.

$$MSE(hr, sr) = \frac{1}{N} \sum_{i=1}^N (hr_i - sr_i)^2 \quad (3.8)$$

$$PSNR(hr, sr) = 10 \log_{10} \frac{L^2}{MSE} \quad (3.9)$$

Burada  $N$  toplam piksel sayısını,  $L$  maksimum renk yoğunluk değeridir (örneğin  $L$  değeri 8bitlik gri tonlama görüntülerde  $L = 2^8 - 1 = 255$  dir).

FFT\_MSE [78] değerinin hesaplanması için uzaysal alandaki  $hr$  ve  $sr$  görüntüleri  $fft(hr)$  ve  $fft(sr)$  komutu yardımıyla frekans alanına dönüştürülmüştür. Frekans alanındaki görüntülerin ortalama karesel farkları hesaplanmaktadır.

Klasik ve güncel görüntü büyütme algoritmalarının performanslarının daha net anlaşılabilmesi için Camelya17 den seçilen yedi farklı görüntü seçilmiş ve 2, 4, 8, 16 kat küçültmeye tabi tutulmuştur. Daha sonra büyütme algoritmaları tarafından seviyeli olarak büyütme işlemi gerçekleştirilmiştir. Elde edilen büyütme sonuçlarının orijinal görüntüye benzerlikleri üç farklı benzerlik metriği (SSIM, PSNR ve FFT\_MSE) kullanılarak hesaplanmıştır. Sonuçlar Çizelge 3.1’de verilmiştir. SSIM değerinin 1’e yakın olması ve PSNR değerlerinin yüksek olması benzerliğin yüksek olduğunu gösterirken FFT\_MSE değerinin düşük çıkması daha iyi benzerlik anlamına gelmektedir. Renklendirilmiş değerler en iyi sonuçları ifade etmektedir.

**Çizelge 3.1 : Görüntü büyütme sonuçları.**

	2X						4X					
	SRGAN			BICUBIC			SRGAN			BICUBIC		
	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE
Img1	<b>0,91</b>	27,24	16,95	0,87	<b>27,88</b>	<b>16,83</b>	<b>0,78</b>	<b>24,88</b>	<b>16,99</b>	0,71	24,09	17,79
Img2	<b>0,90</b>	27,51	16,92	0,89	<b>29,52</b>	<b>16,44</b>	<b>0,82</b>	<b>27,01</b>	<b>16,54</b>	0,77	25,75	17,40
Img3	<b>0,91</b>	27,92	16,86	<b>0,91</b>	<b>31,32</b>	<b>15,99</b>	<b>0,84</b>	<b>28,20</b>	<b>16,19</b>	0,82	27,28	17,00
Img4	<b>0,94</b>	26,82	17,08	0,92	<b>28,47</b>	<b>16,62</b>	<b>0,84</b>	<b>24,64</b>	<b>17,04</b>	0,80	23,53	17,87
Img5	0,92	26,42	17,30	<b>0,93</b>	<b>32,29</b>	<b>15,69</b>	<b>0,90</b>	<b>28,98</b>	<b>16,23</b>	0,86	27,09	17,01
Img6	<b>0,94</b>	27,39	17,09	0,92	<b>30,54</b>	<b>16,16</b>	<b>0,88</b>	<b>28,24</b>	<b>16,32</b>	0,82	26,02	17,29
Img7	<b>0,95</b>	26,41	17,35	0,94	<b>32,10</b>	<b>15,73</b>	<b>0,88</b>	<b>27,57</b>	<b>16,46</b>	0,86	26,73	17,09

İmge	8X						16X					
	SRGAN			BICUBIC			SRGAN			BICUBIC		
	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE
Img1	<b>0,61</b>	20,83	<b>17,84</b>	0,59	<b>21,63</b>	18,41	0,50	19,46	<b>18,38</b>	<b>0,55</b>	<b>20,18</b>	18,77
Img2	<b>0,69</b>	22,92	<b>17,34</b>	0,67	<b>23,32</b>	18,01	0,62	21,75	<b>17,79</b>	<b>0,64</b>	<b>21,98</b>	18,36
Img3	<b>0,72</b>	23,65	<b>17,24</b>	0,72	<b>24,41</b>	17,74	0,65	21,94	<b>17,59</b>	<b>0,68</b>	<b>22,70</b>	18,17
Img4	<b>0,70</b>	20,18	<b>18,20</b>	0,65	<b>20,38</b>	18,69	<b>0,59</b>	18,31	<b>18,57</b>	0,58	<b>18,69</b>	19,12
Img5	<b>0,82</b>	<b>24,61</b>	<b>17,05</b>	0,74	23,19	18,00	<b>0,72</b>	<b>22,02</b>	<b>17,66</b>	0,67	20,70	18,62
Img6	<b>0,80</b>	<b>24,15</b>	<b>17,16</b>	0,72	23,07	18,05	<b>0,71</b>	<b>22,06</b>	<b>17,65</b>	0,67	21,25	18,50
Img7	<b>0,79</b>	<b>23,18</b>	<b>17,39</b>	0,74	22,84	18,07	<b>0,68</b>	<b>20,87</b>	<b>17,90</b>	0,66	20,51	18,64

Klasik ve güncel gürültü temizleme yaklaşımların performanslarını daha iyi anlayabilmek için test veri kümesinden rastgele 10 örnek alınmış ve farklı oranlarda gürültü eklenmiştir. Daha sonra dört farklı gürültü temizleme algoritması kullanılarak temizleme işlemi gerçekleştirilmiştir. Elde edilen sonuçlar Çizelge 3.2’de sunulmuştur. Sonuçlar incelendiğinde, SRGAN modelinin 10 görüntüde de farklı metriklerde en iyi performansı sergilediği gözlemlenmiştir [42].

**Çizelge 3.2 : Gürültü temizleme sonuçları.**

$\sigma = 0.015$												
İmge	SRGAN			Adaptif			Ortalama			Ortanca		
	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE
İmg1	<b>0,86</b>	<b>21,33</b>	<b>16,94</b>	0,62	14,01	18,89	0,54	13,25	19,07	0,62	14,01	18,89
İmg2	<b>0,86</b>	<b>18,44</b>	<b>17,71</b>	0,62	13,77	18,95	0,58	13,09	19,11	0,62	13,77	18,95
İmg3	<b>0,87</b>	<b>20,74</b>	<b>17,12</b>	0,59	14,08	18,87	0,51	13,18	19,08	0,59	14,08	18,87
İmg4	<b>0,87</b>	<b>21,97</b>	<b>16,79</b>	0,60	13,16	19,10	0,56	12,65	19,22	0,60	13,16	19,10
İmg5	<b>0,90</b>	<b>25,50</b>	<b>15,84</b>	0,59	11,63	19,47	0,57	11,13	19,58	0,59	11,63	19,47
İmg6	<b>0,86</b>	<b>19,86</b>	<b>17,41</b>	0,62	14,21	18,85	0,54	13,41	19,03	0,62	14,21	18,85
İmg7	<b>0,87</b>	<b>21,08</b>	<b>17,00</b>	0,62	13,86	18,93	0,53	13,06	19,12	0,62	13,86	18,93
İmg8	<b>0,89</b>	<b>28,02</b>	<b>15,03</b>	0,55	11,81	19,42	0,54	11,40	19,52	0,55	11,81	19,42
İmg9	<b>0,84</b>	<b>18,55</b>	<b>17,73</b>	0,62	14,31	18,82	0,55	13,60	18,98	0,62	14,31	18,82
İmg10	<b>0,87</b>	<b>25,26</b>	<b>15,95</b>	0,58	12,74	19,20	0,54	12,17	19,34	0,58	12,74	19,20

$\sigma = 0.02$												
İmge	SRGAN			Adaptif			Ortalama			Ortanca		
	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE	SSIM	PSNR	FFT_MSE
İmg1	<b>0,54</b>	<b>14,67</b>	<b>18,52</b>	0,34	10,50	19,70	0,35	10,30	19,76	0,34	10,50	19,70
İmg2	<b>0,59</b>	<b>13,59</b>	<b>18,88</b>	0,35	10,27	19,76	0,38	10,08	19,82	0,35	10,27	19,76
İmg3	<b>0,58</b>	<b>13,20</b>	<b>18,95</b>	0,33	10,73	19,64	0,35	10,52	19,71	0,33	10,73	19,64
İmg4	<b>0,58</b>	<b>15,33</b>	<b>18,44</b>	0,34	10,34	19,74	0,37	10,19	19,79	0,34	10,34	19,74
İmg5	<b>0,64</b>	<b>17,16</b>	<b>17,97</b>	0,34	9,27	20,00	0,42	9,10	20,05	0,34	9,27	20,00
İmg6	<b>0,58</b>	<b>14,42</b>	<b>18,63</b>	0,35	10,83	19,62	0,37	10,61	19,69	0,35	10,83	19,62
İmg7	<b>0,54</b>	<b>14,66</b>	<b>18,52</b>	0,33	10,09	19,80	0,33	9,90	19,86	0,33	10,09	19,80
İmg8	<b>0,48</b>	<b>19,64</b>	<b>17,06</b>	0,31	7,67	20,38	0,39	7,57	20,41	0,31	7,67	20,38
İmg9	<b>0,55</b>	<b>13,78</b>	<b>18,80</b>	0,35	10,93	19,59	0,36	10,74	19,65	0,35	10,93	19,59
İmg10	<b>0,51</b>	<b>18,70</b>	<b>17,32</b>	0,31	8,16	20,26	0,39	8,04	20,30	0,31	8,16	20,26



#### 4. EEG SİNYALLERİNDEN GÖRÜNTÜ ÜRETİMİ

Beyin/Bilgisayar arayüzleri (BCI), insanlar ile elektronik cihazlar arasında iletişimi sağlayan temel çalışmalardan biri EEG sinyallerinden elde edilen verinin anlaşılmasını sağlamıştır. Temel olarak BCI'ler, engelli bireylerden alınan EEG sinyalinin sınıflandırılması ile makinelerin entegre edilmesini amaçlamaktadır [79]. Aynı şekilde BCI, insan beyni ile harici aygıtlar arasında alternatif bir iletişim şekli sunmaktadır [80-82]. EEG, beyin etkinliklerini kaydetmesini sağlayan, kolay ve ucuz bir yöntemdir. EEG sinyali, kafatasının çeşitli bölgelerinde saniyede yüzlerce defa elektrik potansiyeli ölçülerek elde edilir. Sinyaller, kafatası derisinin belirli bölgelerine yerleştirilen çoklu elektrotların kullanılmasıyla kaydedilir [44].

EEG sinyalleri, manyetik rezonans görüntüleme (magnetic resonance imaging-MRI) veya bilgisayarlı tomografi (computed tomography-CT) gibi en güncel görüntüleme tekniklerinde dahi mümkün olmayan, milisaniye aralıklarla elde edilen bir çözünürlüğe sahiptir. Bu özellik, beyin fonksiyonlarını, bozukluklarını ya da aktivitelerini daha iyi anlaşılabilmesini sağlaması, EEG sinyalini daha da önemli bir hale getirmiştir [83].

EEG sinyalleri kullanılarak makine öğrenme teknikleriyle sınıflandırmasını sağlayan birçok bilimsel araştırma yapılmıştır [43,79,83,84]. Ancak son dönemler yapılan yeni çalışmalar EEG verileri, sadece sınıflandırma problemlerinde değil aynı zamanda yeni çalışma alanlarında kullanılmaya başlanmıştır. GAN mimarileri temel alınarak geliştirilen bu yeni metotlarla (EEG-GAN), EEG sinyallerinden bakılan görüntülerin üretilmesi sağlanmıştır [31]. GAN, CNN ve AE gibi güncel makine öğrenme tekniklerini kullanılarak inşa edilen bu mimariler, basit görsellerde üretim kalitesinin yüksek olduğu, ancak karmaşık görsellerde üretilen görüntülerde iyi sonuç vermediği görülmektedir.

EEG verileri kullanılarak EEG-GAN'lar ile, EEG/EOG sinyallerinden sürücülerin uyku durumu tespiti [85], EEG sinyallerinden yeni EEG sinyallerinin üretilmesi [86,87], EEG sinyallerinin sınıflandırılması [88] ve bozuk sinyallerin onarımı [89] gibi yeni çalışmalar yapılmıştır. Belirli özelliklere sahip ve/veya belli bir sınıftan sinyaller üretilme ihtimali, beyin sinyalleri ile ilgili yeni araştırmalara yol açılabilir [90]. Üzerinde çok fazla çalışmanın devam ettiği bu alanda, adli vakaların çözülmesine yönelik önemli ve dikkat çekici yeni

araştırmaların önünü açılmasını sağlayacaktır. Ayrıca tıbben bir kişinin kalp fonksiyonları dursa bile beyin aktivitelerinin yaklaşık altı dakikaya kadar devam ettiği görülmüştür [91]. Bu da çok kısa süre önce ölen bir kişinin beyin EEG sinyalleri üzerinde görselleştirme işlemlerinin yapılabileceğini göstermektedir [44].

Bu bölümde, ACGAN mimarisinin temel alındığı yeni bir EEG-GAN mimarisinin geliştirilmesi ve EEG sinyallerinden daha kaliteli görüntülerin üretilmesi amaçlanmıştır. Aynı zamanda üç farklı kodlama yönteminin (FD, AE ve VAE) görüntü üretme performansları incelenmiştir. Bu çalışmanın birinci bölümünde ACGAN mimarisi detaylı bir şekilde sunulmuştur. İkinci bölümde, önerilen kodlama yaklaşımları verilmiştir. Üçüncü bölümde önerilen EEG-GAN mimarisinden detaylı bir şekilde bahsedilmektedir. Dördüncü bölümde bu çalışma için kullanılan veri kümesinden bahsedilmiştir. Beşinci bölümde yapılan deneysel çalışmalar ve elde edilen sonuçlar sunulmaktadır.

#### 4.1 ACGAN Mimarisi

ACGAN'ın detaylı mimarisi Şekil 4.1'de sunulmuştur. Bir  $G$  ve bir de  $D$  içeren bu mimari,  $G$ , giriş olarak gürültü vektörüne ( $z$ ) ilaveten sınıf bilgisini ( $c$ ) alarak sahte görüntüler  $X_{sahte} = G(z, c)$  üretir.  $D$ , giriş olarak aldığı gerçek veya sahte görüntüler için çıkış bilgisi olarak  $[0-1]$  aralığında bir olasılık değeri üreterek, giriş verisinin sahte(0)/gerçek(1) ayırımını yapabilme yeteneğini arttırmaya çalışır. Aynı zamanda giriş verisinin sınıf bilgisini de doğru tahmin etmeyi amaçlar. Bu iki tahmin sonucunda bir maliyet değeri elde edilir.

$D$ , elde edilen bu maliyet değerini geriye yayılım algoritması ile ağırlıkları güncellenir. Aynı şekilde  $D$ 'nin çıkışında elde edilen maliyet değeri,  $G$ 'nin eğitimi için de kullanılır. Ayrıca  $G$ 'nin ürettiği görüntüler gerçekmiş gibi  $D$ 'ye verilerek,  $D$ 'nin yüksek hata değeri üretmesi amaçlanır. Elde edilen bu yüksek hata değeri  $G$ 'ye yansıtılarak, gerçek görüntüler ile ayırt edilemeyecek yeni görüntüler üretmeye zorlanır [44]. ACGAN mimarisinde  $L_S$  ve  $L_C$  olmak üzere iki farklı maliyet fonksiyonu bulunur:

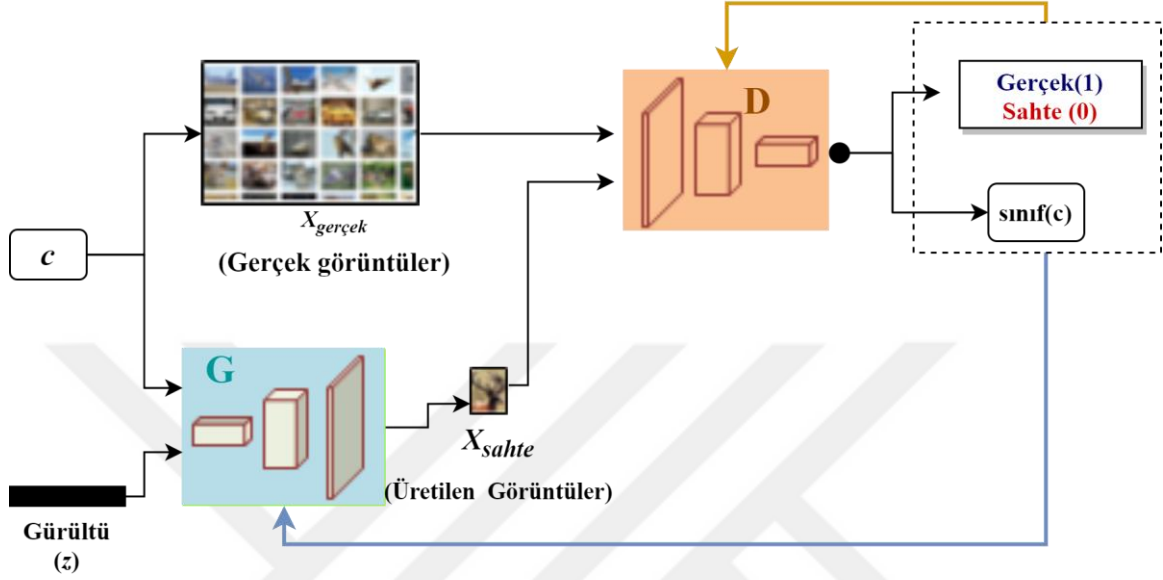
$$L_S = E[\log P(S = \text{gerçek} | X_{\text{gerçek}})] + E[\log P(S = \text{sahte} | X_{\text{sahte}})] \quad (4.1)$$

$$L_C = E[\log P(C = c | X_{\text{gerçek}})] + E[\log P(C = c | X_{\text{sahte}})] \quad (4.2)$$

Burada  $X_{\text{gerçek}}$ , gerçek veri kümesindeki görüntüleri,  $X_{\text{sahte}}$  üretici ağ tarafından üretilen sahte görüntüleri göstermektedir.  $c$  gerçek görüntülerin sınıf bilgisini ifade etmektedir.  $L_C$ ,

girdi olarak verilen görüntünün bulunduğu sınıfla ilgili maliyet bilgisini gösterirken;  $L_S$  ise, bu görüntülerin gerçek mi yoksa sahte mi hakkındaki maliyet bilgisini ifade etmektedir.

$G$ ,  $L_C - L_S$ 'yi maksimum değeri elde etmek için eğitilirken,  $D$  ise  $L_S + L_C$ 'yi en yüksek değer elde etmek için eğitilir. Ağlar, eğitim sürecince ağırlıkları sürekli güncellenir [51].



Şekil 4.1 : ACGAN mimarisi.

## 4.2 Kodlama Teknikleri

Kodlama teknikleri, yüksek çözünürlükte ve tek boyutlu EEG sinyallerinin temel karakteristiği kaybedilmeden daha küçük boyuta taşınma işlemidir [92]. Bu işlem ile orijinal veri boyutu düşürülerek daha küçük boyutta verinin temel yapısının keşfedilmesi amaçlanır. Veri boyutunun düşürülmesi, hesaplama karmaşıklığının azaltılmasını ve öğrenme performansını olumlu yönde artmasını sağlar. Bilindiği gibi özniteliklerin kalitesi sınıflandırıcı mimarilerin performansını büyük ölçüde etkiler [93]. Regresyon, sınıflandırma, bilgi görselleştirme gibi birçok farklı alanda kodlama yöntemi sıklıkla kullanılmaktadır [94].

### 4.2.1 Otomatik kodlayıcı

Kodlayıcı (encoder) ve çözücü (decoder) ağlarından oluşan otomatik kodlayıcı (Auto encoder-AE), eğitimsiz bir öğrenme yöntemidir. Giriş verisi (x) önce kodlayıcı işleminden daha sonra kod çözücü ağından geçirilerek giriş verisine benzer yeni bir çıkış (z) elde etmek istenir. Kodlayıcı ağ,  $m$  boyutta giriş vektörü olan  $x$ 'i,  $m'$  boyutta kodlanmış vektöre  $h$ 'ye dönüştürür. Çözücü ağ ise giriş vektörü  $x$  ile çıkış vektörü  $z$  arasındaki farkı minimize ederek

bu vektörlerin birbirine daha çok benzetmeye çalışır. Elde edilen yeniden yapılanma kaybının istenilen oranda düşük çıkması durumunda, kodlanmış  $h$  vektörünün giriş vektörünü tam olarak ifade ettiği düşünülür [15,94]. Kodlayıcı mimari aşağıda verildiği gibi tanımlanabilir:

$$h = f_{kodlayıcı}(W_e x + b_e) \quad (4.3)$$

Burada,  $x$  giriş verisini,  $b_e$ ,  $m'$  boyutunda bias vektörünü ifade etmektedir.  $W_e$ ,  $m' \times m$  boyutunda ağırlık matrisini,  $f_{kodlayıcı}$  aktivasyon fonksiyonunu (*sigmoid*, *tanh* v.b) ifade etmektedir.

Çözücü modül aşağıdaki gibi ifade edilebilir:

$$z = f_{çözücü}(W_d h + b_d) \quad (4.4)$$

$h$ ,  $x$  giriş verisinin kodlanmış bilgisini temsil ederken,  $W_d$ ,  $m \times m'$  boyutunda ağırlık matrisini,  $b_d$ ,  $m$  boyutunda bias vektörü ifade etmektedir.  $f_{çözücü}$  ise aktivasyon fonksiyonunu göstermektedir. AE'nin amaç fonksiyonu ( $J$ ), giriş ile çıkış verisi arasındaki maliyetin minimizasyonuna dayanır [95].

$$J(z, x) = \min \frac{1}{N} \sum_{i=1}^N ||z_i - x_i|| \quad (4.5)$$

#### 4.2.2 Değişken otomatik kodlayıcı

Giriş görüntülerinin yeniden üretilmesinde değişken otomatik kodlayıcı (Variational auto encoder – VAE) [95] tekniği önemli bir performans göstermiştir. VAE, AE mimarisi ile benzer bir yapıya sahip olsa da çok daha büyük hedeflere hizmet etmektedir. Bu kodlama tekniğinin, AE tekniğinde farkı, mimarinin gizli katmanlardaki nöronlarda normal dağılım parametrelerini içermesidir. Bu şekilde giriş görüntüleri, yönden bağımsız (izotropik) normal dağılımlarla tanımlanmaktadır [96]. Veri karmaşıklığı artması ile normal dağılımla modellemenin eksik kalacağı görülmektedir. VAE maliyet fonksiyonu, düzgünleştirme hatasını ( $\mathcal{L}_{KLD}(\mu, \sigma)$ ) ve dağılımın girdiye oturma hatasını ( $\mathcal{L}_{MSE}$ ) içermektedir [97].

$$\mathcal{L}_{VAE} = \mathcal{L}_{MSE}(x, G_{\theta}(z)) + \lambda \mathcal{L}_{KLD}(\mu, \sigma) \quad (4.6)$$

$$\mathcal{L}_{MSE}(x, G_{\theta}(z)) = \|x - G_{\theta}(z)\|^2 \quad (4.7)$$

$$\mathcal{L}_{KLD}(\mu, \sigma) = \frac{1}{2}(1 + \log(\sigma^2) - \mu^2 - \sigma^2) \quad (4.8)$$

Burada,  $\mu$  giriş ( $x$ ) verisinin ortalamasını,  $\sigma$  ise standart sapma değerini göstermektedir.  $\mathcal{L}_{MSE}$ , giriş verisi ( $x$ ) ile bunların yeniden yapılanmaları ( $G_{\theta}(z)$ ) arasındaki ortalama karesel hatayı ifade etmektedir [44].

### 4.2.3 Fourier dönüşümü

Fourier dönüşümü (FD), zaman gösterimli bir sinyali frekans uzayına taşınmasını sağlar. FD, sinyali harmonikler toplamı şeklinde tanımlar. Kayıpsız bir şekilde gerçekleştirilen bu dönüşümün tersi alınabilir [98,99]. Bilhassa tek boyutlu sinyal analizi uygulamaları için en temel dönüşümlerinden biridir. FD aşağıdaki gibi tanımlanabilir.

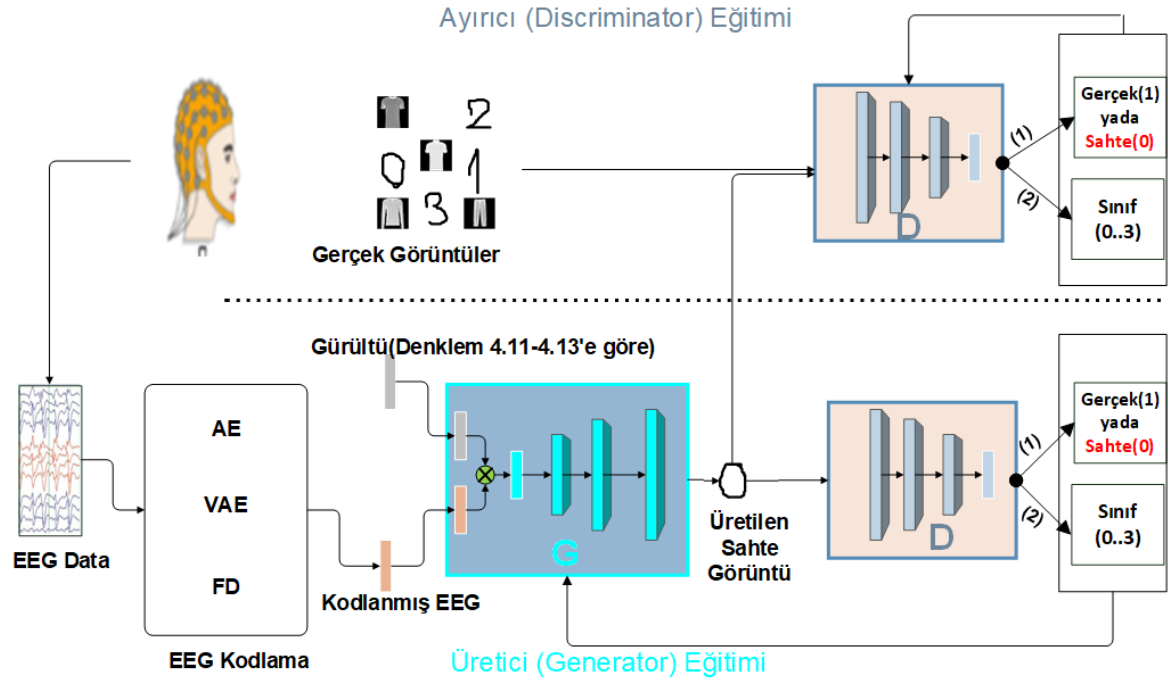
$$X_{FD}(f) = \int_{-\infty}^{+\infty} x(t)e^{-i2\pi ft} dt \quad (4.9)$$

$$x(t) = \int_{-\infty}^{+\infty} X_{FD}(f)e^{i2\pi ft} df \quad (4.10)$$

$X_{FD}(f)$ , frekans alanı ifade ederken,  $x(t)$ , EEG sinyalinin zaman alanını göstermektedir.

### 4.3 Önerilen EEG-GAN Mimarisi

Önerilen EEG-GAN mimarisi Şekil 4.2’de detaylı bir şekilde verilmiştir. Buna göre, denklere gösterilen görsellerden elde edilen EEG sinyalleri kaydedilmektedir. Elde edilen EEG sinyalleri farklı kodlama yöntemleri (FD, AE ve VAE) ile kodlanarak EEG-GAN mimarisine giriş olarak verilmektedir. Bu şekilde EEG ile görsel arasında eşlenmesini sağlayan bir ağın eğitilmesi amaçlanmaktadır [44].



Şekil 4.2 : Önerilen EEG-GAN modeli.

Önerilen mimari, ACGAN mimarisini temel alarak bu mimarinin güncellenmiş bir versiyonudur. Görüntü üretiminde ACGAN mimarisi, giriş olarak gürültü vektörünü ( $z$ ) ve sınıf bilgisi ( $c$ ) alırken, önerilen mimaride ise, girdi olarak kodlanmış EEG sinyali ile Denklem 4.11-4.13 göre üretilen gürültü vektörü kullanılır. Her iki GAN mimarisinde aynı maliyet fonksiyonu kullanılmaktadır [44].

ACGAN mimarisinin klasik kullanımda EEG sinyallerinden kaba görüntülerin üretildiği görülmektedir. Bunu çözenin bir yolu olarak girdi verisi sınıfların ortalama ve standart sapma değerleri ve rasgele giriş vektörünün (Denklem 4.11) rasgelelik düzeyinin indirgenmesi (Denklem 4.12-4.13)) ile birlikte kullanılması üretilen görsellerin kalitesi üzerinde olumlu etki yapacağı öngörülmüştür. Bahsi geçen rasgelelik düzeyi Denklem 4.12 ve Denklem 4.13 kullanılarak indirgenmektedir.

$$z_1 = randn(1, sizeEEG) \quad (4.11)$$

$$z_2 = \mu_{class} + \sigma_{class} * randn(1,1) \quad (4.12)$$

$$z_3 = \mu_{class} + \sigma_{class} * randn(1, sizeEEG) \quad (4.13)$$

Burada  $\mu_{class}$  ve  $\sigma_{class}$ , kodlanmış EEG sınıflarının sırasıyla ortalamasını ve standart sapmasını ifade etmektedir.  $size_{EEG}$ , kodlanmış EEG sinyalin vektör uzunluğunu,  $randn$  normal dağılım fonksiyonunu göstermektedir.

ACGAN mimarisi üzerinde çeşitli modifikasyonların yapılması EEG sinyalinden daha kaliteli görsellerin üretilmesi sağlamaktadır [44].

#### 4.4 Veri Kümesi

EEG sinyallerinden bakılan görselin üretilmesi hedeflendiği bu çalışmada, bazı deneklere belirli görseller gösterilerek elde edilen EEG sinyali içeren veri kümeleri incelenmiştir. Bu gereklilik gözetilerek yapılan literatür çalışmasında MindWave [100] ve DB [43] veri kümelerine ulaşılmıştır.

MindWave [100] veri kümesi, EEG sinyalleri deneklere rakam (0...9) içeren görseller iki saniye boyunca gösterilerek elde edilmiştir. Her bir EEG sinyalinin boyutu 141-1017 arasında değişkenlik gösteren bu veri kümesinde, toplamda 54.934 adet EEG sinyali içermektedir.

DB [43] veri kümesi, dört farklı karakterden oluşan görseller, deneklere üç buçuk saniye gösterilerek elde edilen EEG sinyallerinden oluşmaktadır. Toplam 2670 adet EEG sinyali içeren bu veri kümesinde, her bir sinyalin uzunluğu 1344'tür. Bu veri kümesindeki EEG sinyalleri zaman alanından frekans alanına dönüşüm işlemi yapılmıştır.

İlk olarak incelenen veri kümeleri Çizelge 4.1(a)'da verilen CNN mimarisi kullanılarak bu veri kümelerinin güvenilirliği test edilmiştir. CNN mimarisi her bir veri kümesi üzerinde 100 iterasyon eğitilmiştir. Veri kümelerinin yaklaşık %25'i test , %75'i eğitim için kullanılmıştır.

CNN mimarisinin MindWave veri kümesi üzerindeki eğitimi sonucunda EEG sinyallerini sınıflandırılma başarımının %9.6 gibi düşük bir değerin çıktığı görülmüştür. Başarım oranının düşük çıkması, EEG sinyallerini içeren bu veri kümesinin sağlıklı bir şekilde oluşturulmadığını göstermektedir (bk. Çizelge 4.1(b)). Bu sebeple MindWave veri kümesi kullanılmamıştır. Benzer şekilde, DB [43] veri kümesindeki EEG sinyalleri üzerinde veri doğrulama işlemi gerçekleştirilmiştir ve %91 gibi yüksek bir başarımla sinyallerin sınıflandırıldığı görülmüştür. Yapılan güvenlik testi sonuçları Çizelge 4.1(b)'de sunulmuştur. DB veri kümesi ile yüksek başarımın elde edilmesi bu veri kümesinin kullanımına karar verilmiştir.

**Çizelge 4.1 :** MindWave ve DB veri kümelerinin güveninirlik test sonuçları.

(a) CNN Mimarisi	(b) Test Sonuçları	
Dense(512)		
Activation(relu)	Veri Kümeleri	Doğruluk (%)
Dropout(0.2)	MindWave	9.6
Dense(512)	DB	91
Activation(relu)		
Dropout(0.2)		
Dense(sınıf sayısı)		
Activation(softmax)		

#### 4.5 Deneysel Çalışmalar ve Sonuçları

Bu bölümde, EEG sinyalleri üzerinde önerilen yaklaşımlar ile klasik EEG-GAN [31] yaklaşımının görüntü üretme başarıları analiz edilmiştir. Çalışmalar, 20 çekirdekli 2 işlemciye sahip bir CPU, 64 GB ram ve Nvidia Quard M4000 GPU donanım birimlerinden oluşan bir sunucu ortamında, Matlab ve Python programlama dilleri kullanılarak gerçekleştirilmiştir.

DB veri kümesinde EEG sinyallerinin frekans karşılıkları bulunmaktadır. AE ve VAE yöntemleri frekans sinyalleri frekans→zaman dönüşümü gerçekleştirilmiştir. Frekans alanından zaman alanında dönüşümden kodlama işlemi gerçekleştirilmiştir. AE ve VAE mimarilerinin eğitimi için, EEG sinyalleri giriş ve çıkış katmanına verilerek 1000 iterasyon eğitilmiştir. Eğitim işleminden sonra 2670x1344 (kayıt sayısı x EEG sinyal boyutu) boyutunda olan veri kümesi AE yönteminin kodlayıcı ağı ile kodlanma işlemi sonrası 2670x100 boyutuna indirgenmiştir.

İkinci aşamada, EEG sinyalinden görüntü üretimi gerçekleştirmek için, DB [43] veri kümesinde bulunan EEG sinyallerinin sınıf (etiket) değerlerine karşılık gelen uygun görsel atama işlemi gerçekleştirilmiştir. Görsel atama işlemi için MNIST [38] ve Fashion-MNIST [39] veri kümelerindeki görseller kullanılarak atama işlemi gerçekleştirilmiştir. MNIST veri kümesi 0-9 arasında el yazısıyla yazılmış rakamları içerirken, Fashion MNIST veri kümesi 0-9 arasında 10 farklı sınıf içeren (0=tişört, 1=pantolon, 2=kazak, 3=elbise, 4=ceket, 5=sandalet, 6=gömlek, 7=spor ayakkabı, 8=çanta, 9=kısa bot) görüntülerden oluşur. DB veri kümesi dört sınıflı EEG sinyallerinden oluştuğundan dolayı, EEG sinyallerine görsel atama işlemi için MNIST ve Fashion-MNIST veri kümelerinin yalnızca ilk dört sınıfın (0-3) görüntüleri kullanılmıştır. EEG sinyallerin etiket (sınıf) numarası ile aynı olan veri



kümelerinden gelişi güzel bir görsel seçilerek EEG sinyalinin yanına konulmuştur. Bu şekilde her bir kodlama yöntemi (Orijinal, FD, AE, VAE) için dört farklı veri kümesi oluşturulmuştur. Bu veri kümeleri, her bir kaydın etiket bilgisini, kodlanmış EEG sinyalini ve görsel bilgisini içerecek şekilde toplamda 2670 kayıttan oluşmaktadır. Sinyal-görsel atama işlemi ile oluşan bir kaydın boyutları Çizelge 4.2’de gösterilmiştir.

**Çizelge 4.2 : EEG-Görsel atama işleminden sonra bir kaydın boyutları.**

Kodlayıcılar	Etiket	Kodlanmış_EEG	Görsel	Toplam
AE	1	100	784	885
Orijinal	1	1344	784	2129
VAE	1	1344	784	2129
FD	1	1344	784	2129

Üçüncü aşamada, önerilen EEG-GAN mimarisi hazırlanarak farklı yöntemler ile eğitimi yapılmıştır. Eğitim işlemi, dört farklı kodlama yöntemi (Orijinal, FD, AE, VAE) ve Denklem 4.11-4.13’ e göre üç farklı gürültü oluşturma yöntemi ile ayrı ayrı 12 farklı şekilde 100 iterasyon gerçekleştirilmiştir. Çizelge 4.2’deki bilgiler göz önünde bulundurularak hazırlanan dört farklı veri kümesinin %25’i test , %75’i eğitim için kullanılmıştır. Önerilen model Şekil 4.2’de sunulmuş, aynı zamanda mimarisi Çizelge 4.3’te verilmiştir.

**Çizelge 4.3 : EEG-GAN mimarisi.**

Ağ	Katman	Özellik Haritası	Çıkış Boyutu	Filtre Boyutu	Adım	Aktivasyon
Üretici	Giriş görüntüsü	-	100	-	-	-
	Giriş_gürültüsü (Denklemler 4.11-4.13 göre)	-	100	-	-	-
	Multipliy (Giriş_EEG x Girişgürültü)	-	100	-	-	-
	Dense	-	6272	-	-	-
	Reshape	-	7x7x128	-	-	-
	BatchNormalizasyon	-	7x7x128	-	-	-
	UpSampling2D	-	14x14x128	-	-	-
	Conv2D	128	14x14x128	3x3	-	relu
	BatchNormalizasyon	-	14x14x128	-	-	-
	UpSampling2D	-	28x28x128	-	-	-
	Conv2D	64	28x28x64	3x3	-	-
	BatchNormalizasyon	-	28x28x64	-	-	relu
	Conv2D	1	28x28x1	3x3	-	tanh
	Ayrııcı	Giriş	-	28x28x1	-	-
Conv2D		16	14x14x16	3x3	2	-
LeakyReLU (alpha=0.2)		-	14x14x16	-	-	-
Dropout (0.25)		-	14x14x16	-	-	-
Conv2D		32	7x7x32	3x3	2	-
ZeroPadding2D		-	8x8x32	-	-	-
LeakyReLU (alpha=0.2)		-	8x8x32	-	-	-
Dropout (0.25)		-	8x8x32	-	-	-
BatchNormalizasyon		-	8x8x32	-	-	-
Conv2D		64	4x4x64	3x3	2	-
LeakyReLU (alpha=0.2)		-	4x4x64	-	-	-
Dropout (0.25)		-	4x4x64	-	-	-
BatchNormalizasyon		-	4x4x64	-	-	-
Conv2D		128	4x4x128	3x3	1	-
LeakyReLU (alpha=0.2)		-	4x4x128	-	-	-
Dropout (0.25)		-	4x4x128	-	-	-
Flatten		-	2048	-	-	-
Ek olarak ayrııcı ağ için aşağıdaki katmanlar, giriş verisinin gerçek(1) / sahte(0) durumu için (a), etiket bilgisi için (b) seçilir.						
	(a) Dense		1			sigmoid
	(b) Dense		Sınıf sayısı			softmax

Dördüncü aşamada, farklı yöntemler eğitim aşaması tamamlandıktan sonra, test veri kümesindeki kodlanmış EEG sinyalleri kullanılarak bu farklı yöntemler ile görsel üretme

gerçekleştirilmiştir. Görsel üretme için önerilen mimarinin sadece  $G$  ağı kullanılmıştır. Test veri kümelerindeki her sınıftan alınan 100 adet EEG sinyali,  $G$  ağına giriş olarak verilerek görüntülerin üretilmesi sağlanmıştır. Bu şekilde 400 EEG sinyalinden her bir yöntem ile 400 görüntü üretilmiştir. Çizelge 4.4'te üretilen örnek görüntüler verilmiştir. Üretilen görseller incelendiğinde, FD kodlama yöntemiyle daha kaliteli ve kendi sınıfından görüntüler ürettildiği görülmektedir.

**Çizelge 4.4 :** EEG sinyallerinden üretilen örnek görüntüler.

Kodlayıcı	Gürültü (z)	MNIST görselleri			
		0	1	2	3
[31] AE	$z_1$				
	$z_2$				
	$z_3$				
Orijinal Orijinal	$z_1$				
	$z_2$				
	$z_3$				
Önerilen VAE	$z_1$				
	$z_2$				
	$z_3$				
Önerilen FD	$z_1$				
	$z_2$				
	$z_3$				

Kodlayıcı	Gürültü (z)	Fashion-MNIST görselleri			
		0 (Tişört)	1 (Pantolon)	2 (Kazak)	3 (Elbise)
[31] AE	$z_1$				
	$z_2$				
	$z_3$				
Orijinal Orijinal	$z_1$				
	$z_2$				
	$z_3$				
Önerilen VAE	$z_1$				
	$z_2$				
	$z_3$				
Önerilen FD	$z_1$				
	$z_2$				
	$z_3$				

Son aşamada, farklı yöntemlerle üretilen görsellerin kalitesinin ve performansını değerlendirmek için MNIST ve Fashion-MNIST veri kümeleri ile ayrı ayrı eğitilmiş CNN yöntemi kullanılmıştır. Üretilen görseller CNN yöntemi ile sınıflandırılması sağlanmıştır. Bu aşamada ilk olarak CNN mimarisinin eğitimi için MNIST ve Fashion-MNIST veri kümelerindeki ilk dört sınıfın (0-3) verileri kullanılarak ayrı ayrı iki eğitim veri kümesi hazırlanmıştır. Hazırlanan iki farklı veri kümesinde birinde MNIST veri kümesindeki 0 ile 3 arasındaki rakamları içeren görüntülerden oluşurken, diğer veri kümesi ise Fashion-MNIST veri kümesindeki 0 ile 3 arasında etiketlenmiş (0=tişört, 1=pantolon, 2=kazak ve 3=elbise) görsellerden oluşmaktadır. Bu veri kümeleri CNN mimarisi ile ayrı ayrı 100 adım (iterasyon) (MNIST için başarı oranı=%99.9, Fashion-MNIST için başarı oranı=%99.8) eğitilmiştir. Eğitim işleminden sonra EEG-GAN mimarisi kullanılarak farklı yöntemler ile üretilen görüntüler CNN ağına giriş olarak verilerek, uygun sınıfa atanıp atanmadığına bakılmıştır. Çizelge 4.7’de üretilen görsellerin CNN doğrulukları sunulmuştur.

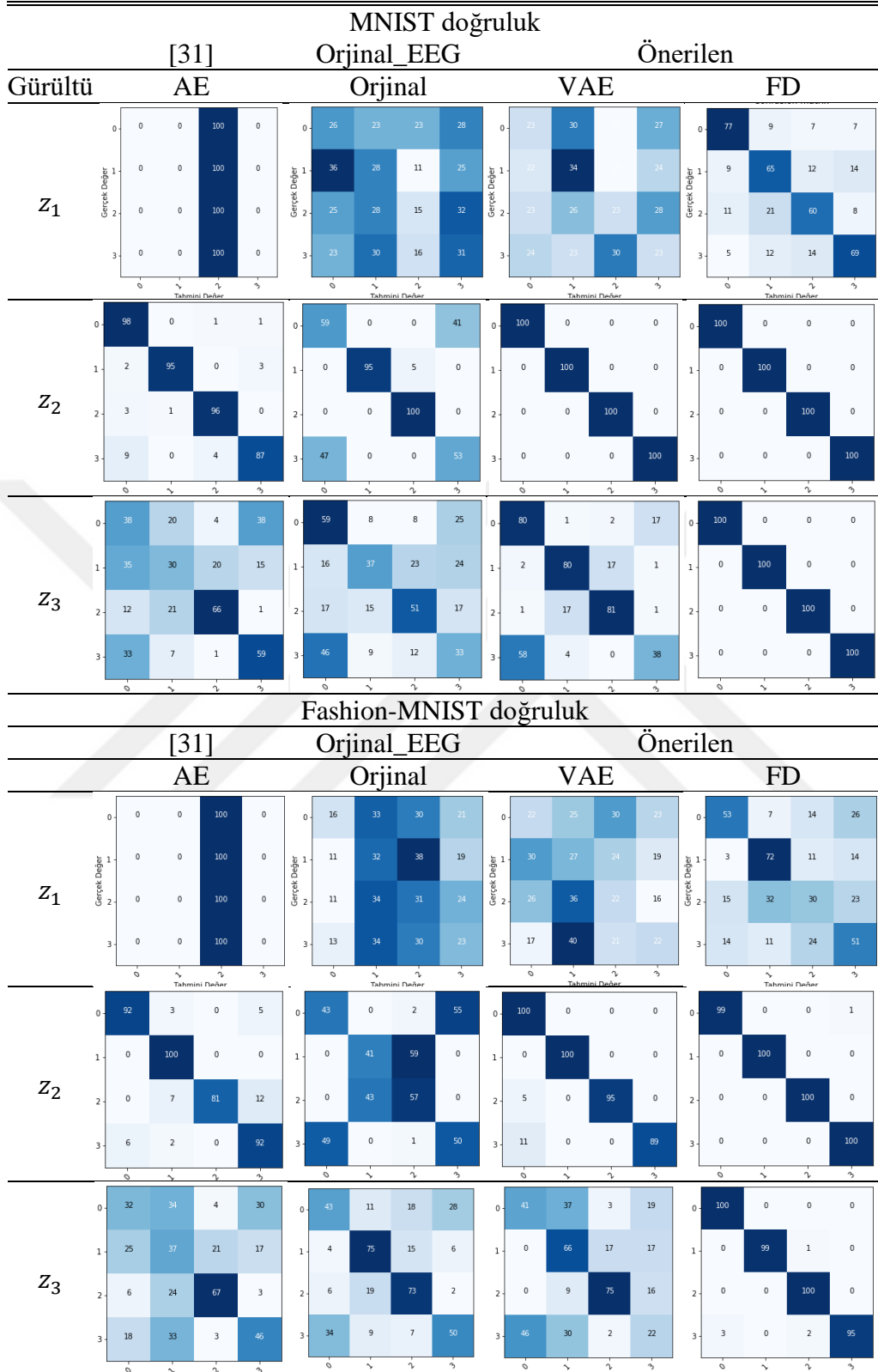
CNN sınıflandırma doğrulukları için karışıklık matrisi (confusion matrix) yöntemi kullanılmıştır. Bu yöntem sınıflandırma başarımını ölçmek için kullanılan bir tekniktir. Karışıklık matrisi, Denklem 4.14’te verilen formül ile sınıflandırma başarımı hesaplanmaktadır.

$$Doğruluk = \frac{TP + TN}{TP + TN + FP + FN} \quad (4.14)$$

Burada,  $TP$  doğru sınıflandırılan pozitif örnek sayısı,  $TN$  doğru sınıflandırılan negatif örnek sayısı,  $FP$  yanlış sınıflandırılan pozitif örnek sayısı ve  $FN$  yanlış sınıflandırılan negatif örnek sayısını göstermektedir [101].

Çizelge 4.5’teki sonuçlara göre, önerilen EEG-GAN yöntemi ile ve önerilen FD ve VAE kodlama tekniklerinin kullanılmasıyla EGG sinyallerinden üretilen görüntülerin eğitilmiş CNN tekniği ile sınıflandırma performansının oldukça yüksek çıktığı görülmektedir. Bu sonuçlar, önerilen GAN mimarisinin ve kodlama teknikleri ile EEG sinyallerinin görüntüye dönüştürme yeteneklerinin klasik yaklaşımlara göre daha iyi olduğunu göstermektedir. Çizelge 4.4 ve Çizelge 4.5’teki sonuçlar detaylı bir şekilde incelendiğinde, özellikle FD kodlama tekniği ile Denlem 4.12 ( $z_2$ )’deki gürültü vektörünün kullanılmasıyla daha kaliteli görüntüler ve daha yüksek performans sergilediği görülmektedir. FD kodlama tekniğinden sonra sırasıyla VAE ve AE kodlama teknikleri ile uygun sonuçlar elde edildiği görülmektedir [44].

**Çizelge 4.5 : EEG sinyallerinden üretilen görüntülerin CNN başarıları.**



## 5. 3D BEYİN MRI TARAMALARININ BÖLÜTLENMESİ

Beyin MRI verilerinin işlenmesi, nöroloji, psikoloji ve bilgisayar bilimleri (görüntü işleme, bilgisayarlı görme ve yapay öğrenme) alanlarının kesişiminde yer almaktadır. Beyin anatomisinin keşfi, hasarlı bölge tespiti, tümör tespiti ve beyin bölgelerinin bölütlenmesi gibi birçok faaliyette MRI verileri yaygın bir şekilde kullanılmaktadır [102,103]. Son yıllarda beyin görüntüleme teknolojisindeki gelişimler, yüksek çözünürlükte (yaklaşık 0.2mm hassasiyette) ve farklı modalitelerde (T1, T2, Flair) MRI verilerinin elde edilebilmesi sağlamıştır. Bu durum, daha iyi görülebilen anatomik yapıların bölütlenmesi ve analizi hakkındaki çalışmalara olumlu katkı sağlayarak araştırmacıları MRI verilerini kullanmaya sevk etmiştir [104].

MRI verileri kullanılarak beyin bölgelerinin bölütlenmesi, psikiyatrik bozuklukların gelişimi ve nörolojik hastalıkların teşhisi gibi klinik ve/veya klinik olmayan analizlerde temel bir süreçtir. Aynı zamanda bölütleme işlemi, ön işleme tekniklerinde tıbbi görüntüleme için önemli rol oynar. Bu işlem doktorların tanıyı doğrulamasına ve tedavi süreçlerinde doğru aksiyon almasına yardımcı olur [105-107]. Manuel olarak tüm beyin bölgelerinin bölütlenmesi yüksek başarıma sahip olsada oldukça maliyetli ve zaman alıcı bir işlemdir [108,109]. Bu maliyetlerin ortadan kaldırılması ve subjektif sonuçlar yerine objektif çıktılar elde edilme ihtiyacı otonom bölütleme çalışmalarına ivme kazandırmıştır.

Bölütleme alanı oldukça doygun bir literatüre sahiptir. Uzun yıllar kullanılan Grabcut ve Lazy Snapping algoritmaları, çizge kesim yaklaşımına dayanmaktadır. Bu yaklaşımda MRI görüntüleri çizgelere dönüştürülmektedir. Bu dönüşümde, görüntü pikselleri çizgenin düğümlerini ve pikseller arasındaki yoğunluk farkları ise çizgenin aygıt ağırlıklarını temsil eder. Daha sonra enerji fonksiyonlarının minimizasyonu ile çizge önplan/arkaplan olmak üzere ikiye bölünerek bölütleme gerçekleştirilir. “Max-flow min-cut” teoremi [110] uzun yıllar bölütleme algoritmalarında tercih edilen bir enerji fonksiyonu olmuştur. Yüksek bölütleme doğruluğuna sahip bu algoritmaların 2D görüntülerde çalışması, görüntü-çizge dönüşüm işleminin maliyetli olması ve ön/arka plana ait önsel bilgiye ihtiyaç duyulması gibi dezavantajlarından dolayı araştırmacılar farklı bölütleme yaklaşımlarına yönelmişlerdir.

Literatürde 3D MRI bölütlemesi yapabilen birçok güncel yazılım bulunmaktadır. FSL [111], SPM [112], FreeSurfer [113] ve VolBrain [114] bunlardan birkaçıdır. FSL ve SPM yazılımları, bölütleme yapmak için Gauss Karışım Modeli ve Markov Rastgele Alanı tekniklerini kullanarak, beyin gray matter (GM), white matter (WM), cerebrospinal fluid (CSF) dokuları için farklı olasılık yoğunluk haritaları çıkarmaya çalışır [115]. Ancak beyin bölgeleri verileri (özellikle farklı yaş grup verileri) incelendiğinde karışım modelleriyle modellenemeyecek derecede karmaşık bir dağılıma sahip olduğu bilinmektedir. Bu nedenle bu yazılımların bölütleme doğrulukları tatmin edici düzeyde değildir. FreeSurfer, MRI verisi üzerinde birçok işlemin (skull stripping, bias field correction, reconstruction, segmentation, labeling of regions, nonlinear registration) yapılabildiği önemli bir araçtır. Bu araç kullanılarak tüm beyin bölgelerini bölütleyen bir yaklaşım [116] çalışmasında görülmektedir. Farklı tarayıcı verilerinin kullanıldığı bu çalışmada, anatomik yapıların şekillerini ayırt eden modeller geliştirilmiş ve sadece WM bölgesinin bölütlenmesine odaklanılmıştır. VolBrain, beyin bölümlerinin bölütlenmesi ve bölgelere ait hacimsel değerlerin çıkarılması için geliştirilmiş web tabanlı bir yazılımdır. Sisteme yüklenen işlenmemiş MRI verilerini otomatik olarak bölütlemekte ve her bir bölgenin hacim değerlerini içeren grafiksel bir rapor üretmektedir. Sisteme erişimin ücretsiz ve online olması birçok araştırmacı ve doktor tarafından aktif kullanılmasına neden olmuştur [117,118]. VolBrain, bölütleme işlemini yapabilmek için beyin bölgelerine ait önceden tanımlanmış görüntü parçalarının sınıflandırılmasına dayanmaktadır. Ancak, günlük sınırlı sayıda bölütlemeye izin vermesi ve her bir bölütleme raporunu 10-15 dakikada üretmesi, grupsal veri analizi işlemlerini oldukça zorlaştırmakta ve manuel yapılan işlemler için hata riskini arttırmaktadır.

Evrişimsel derin öğrenme mimarilerinin 2D/3D verilerde çalışabilmesi, yüksek genelleme performansı sergilemesi, otomatik öznitelik çıkarma kabiliyetine sahip olması ve paralel çalışmaya elverişli olması nedeniyle son yılların popüler uygulama alanlarından biri haline gelmesine neden olmuştur. Derin öğrenme tabanlı bölütleme yaklaşımlarını U-Net [119] mimarisini ve GAN [21] tekniği kullananlar şeklinde ikiye ayırmak mümkündür. Çiçek ve arkadaşları tarafından geliştirilen U-Net mimarisinin modifiye edilerek birçok MRI bölütleme ağında kullanıldığı görülmektedir. Kodlayıcı ve çözücü şeklinde iki katmandan oluşan U-Net mimarisinin en temel özelliği, kodlayıcı katman çıktılarının çözücü katmanda yeniden kullanılıyor olmasıdır. NeuroNet [120] ve QuickNAT [121], U-Net temelli bölütleme mimarilerine örnek olarak verilebilir. NeuroNet, 3D bölütlemeye izin verirken,

QuickNAT, MRI'ın 2D görüntülerinde (coronal, axial ve sagittal) bölütleme yapmakta ve sonuçları birleştirmektedir.

Luna ve Park tarafından geliştirilen 3D P-U-Net [122] mimarisi, 2018 yılından düzenlenen MRBRAIN18 (MR18) [47] beyin MRI bölütleme yarışmasında birinci olmuştur. Bu mimari incelendiğinde, geleneksel U-Net mimarisindeki kodlama ve çözme katmanları arasına yeni bir geçiş katmanı eklendiği görülmektedir. Mimarinin oldukça uzun bir eğitim sürecinden (500 bin iterasyon) geçmesi yüksek doğruluk sağlamasına neden olmuştur. Eğitim sürecindeki bu farklılıklar mimariler arasında adil bir karşılaştırma yapılmasını önlemektedir. Bu çalışmada, mimari eğitim süreleri eşitlenerek adil bir karşılaştırmanın yapılması hedeflenmiştir. Bu durumda 3D P-U-Net yönteminin yeterli başarıyı gösteremediği gözlemlenmiştir. Bununla birlikte, sadece CPU'nun kullanıldığı bir konfigürasyonda yüksek zaman maliyetine sahip olması bir eksiklik olarak değerlendirilmiştir.

Vox2Vox [123], tümörlü bölgenin tespiti için geliştirilen GAN tabanlı bir bölütleme yöntemidir. Üretici ve ayırıcı şeklinde iki farklı mimarinin eğitildiği bu yöntemde, üretici ağ kodlama, dar boğaz ve çözücü katmanlarından oluşmaktadır. Üretici groundTrue'ya benzer sınıf etiketleri üretmek için eğitilirken, ayırıcı, MRI verileriyle birlikte aldığı sınıf etiket bilgisini gerçek veya sahte olarak sınıflamaktadır. Tümör bölütleme probleminde yüksek başarımlı sağlayan bu mimarinin beyin MRI bölütlenme probleminde tatmin edici sonuçlar vermediği görülmektedir.

Mimarilerin katmansal yapılarının haricinde bölütleme başarısına etki eden etkenlerden biride girişte kullanılan veri boyutudur. VoxResNet [124], 3D MRI verisini volume şeklinde bütün olarak sınıf etiket değerlerine eşlemektedir. Yüksek boyutlu MRI verileri, bellek sınırlamaları nedeniyle işlenememektedir. Bu nedenle kırpma veya yeniden boyutlandırma işlemleri gerçekleştirilerek hafıza problemlerinin üstesinden gelinmektedir. Ancak yapılan boyutlandırma işlemleri verinin orijinalliğini bozduğu için bölütleme performansını olumsuz etkilemektedir. Bununla birlikte, Dolz ve ark.[125] çalışmasında bellek gereksinimlerini azaltmak, havuzlama işleminden kaçınmak, hacimsel çözünürlüğünü korumak ve eğitim örnek sayısını artırmak için giriş verisini bütün bir volume olarak değil volume içerisinden örneklenen parçalar şeklinde organize etmektedir. HyperDense-Net isimli modelin önerildiği bu çalışma incelendiğinde patch kullanımıyla bölütleme performansının arttığı, ancak volume kullanan mimarilere göre işlem süresinde uzadığı görülmüştür.



Bu çalışmada, klasik U-Net ve GAN mimarilerinden farklı olarak, pozisyon dikkat mekanizmasının ve lokal/global bağlam veri füzyonunun bölütleme başarısına etkisini gözlemleyebilmek için GAN temelli yeni bir mimari (Vol2SegGAN) önerilmektedir. Bu mimarinin literatürdeki konumunun net anlaşılabilmesi için ortak veri kümeleri üzerinde eğitim ve doğrulama faaliyetleri gerçekleştirilmiştir. Github üzerinden araştırmacılarla paylaşılan bu yeni mimari kullanılarak grupsal analiz işlemlerinin objektif yapılacağı düşünülmektedir.

## 5.1 Veri Kümeleri

Bu çalışmada IBSR18 [45], MRBRAINS13 (MR13) [46] ve MRBRAINS18 (MR18) [47] veri kümeleri kullanılmıştır. IBSR18 veri kümesi, toplam 18 hastanın T1 modaliteye sahip MRI verilerini ve bunların bölütleme sonuçlarını (groundTrue) içermektedir. Bölütleme işlemi üç doku (WM, GM ve CSF) üzerinde gerçekleştirilmiştir. Hastaların MRI verilerinde kafatası sıyırma işlemi gerçekleştirilmiştir.

MR13 veri kümesi, toplam 5 hastadan elde edilmiş MRI verileri ve etiketlenmiş bölütleme sonuçları bulunmaktadır. Her bir MRI 3 farklı modaliteye (T1, T1-IR ve FLAIR) sahiptir. Veriler 9 farklı kategoride (Background, Cortical GM (CGM), Basal ganglia (BG), WM, WM lesions(WMH), Cerebrospinal fluid in the extracerebral space (CF), Ventricles (VE), Cerebellum (CE) ve Brainstem (BS)) etiketlenmiştir.

MR18 veri kümesi, toplamda 7 hastadan alınmış MRI verilerini içermektedir. Veri kümesi, 3 farklı modaliteye (T1, T1-IR, T2\_FLAIR) ve etiketlenmiş bölütleme sonuçlarını barındırmaktadır. Veriler 11 farklı etiketle (Background, CGM, BS, WM, WMH, CF, VE, CE, BS, Infarction ve Other) bölütlenmiştir. MR13 ve MR18'deki MRI verileri herhangi bir ön işlemten geçirilmediği için kafatasıyla birlikte tüm kemiksi yapıları içermektedir. Bu üç veri kümesine ait detaylı bilgiler Çizelge 5.1'de özetlenmektedir.

**Çizelge 5.1 : Veri kümeleri ve özellikleri.**

Veri Kümesi	Volume (pixels)	Voxel (mm)	Modalite	Sınıf sayısı	MRI sayısı	Kullanılan modalite
IBSR	256x256x128	0.93x0.93x1.5	T1	3	18	T1
MR13	240x240x48	0.95x0.95x3	T1, T1-IR ve FLAIR	8	5	T1
MR18	240x240x48	0.95x0.95x3	T1, T1-IR ve FLAIR	11	7	T1

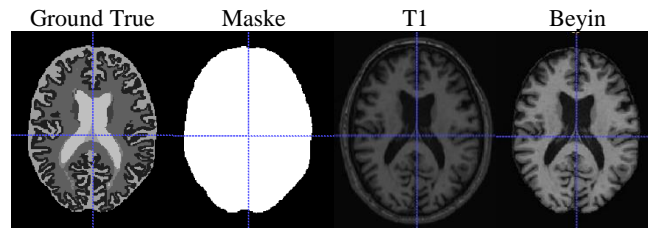
Bu veri kümelerindeki sadece T1 modalitesine sahip veriler kullanılmaktadır. Bunun nedenleri, klasik MRI çekimlerinde T1 modalitesinin yaygın olarak tercih edilmesi, T1'deki WM/GM doku zıtlık değerinin yüksek olması [115,126] ve çoklu modalite eğitimlerinde hafıza problemlerinin yaşanmasıdır. Böylece toplamda 30 adet T1 modalitesine sahip MRI verisi içeren bir veri kümesi hazırlanmıştır.

## 5.2 Ön İşlemler

Bu bölümde üç farklı veri kümesinden toparlanan 30 adet T1 modaliteye sahip MRI verisinin standart bir forma dönüştürülebilmesi için gerekli olan ön işlem aşamalarından bahsedilmektedir. Veriler, kafatasının sıyrılmış olup olmaması, çözünürlük değerleri ve sınıf etiket sayıları noktasında farklılıklar içermektedir. Ön işlemler sayesinde verilerdeki bu farklılıklar giderilerek bölütleme mimarilerinin eğitim/test faaliyetlerine hazır hale getirilmektedir.

### 5.2.1 Beyin bölgesinin çıkarılması

Bilindiği gibi MRI verileri beyin bölgesiyle birlikte kafatası ve birçok kemiksel yapıları (diş, çene vb.) içermektedir. Burada beyin bölgelerinin bölütlenmesine odaklanıldığı için kafatası ve diğer kemiksi yapıların silinmesi gerekir. Bu işlem için BET [127], BSE [128], ROBEX [129] ve BEaST [130] gibi araçların kullanıldığı bilinmektedir. Ancak bu araçların kafatası sıyırma başarımlarının giriş veri kalitesine oldukça bağımlı olduğu görülmüştür. Bununla birlikte toparlanan veriler GroundTrue (GT) sınıf etiket verisini barındırdığı için Şekil 5.1'de gösterilen aşamalar gerçekleştirilerek beyin bölgesi çıkarılmaktadır.



Şekil 5.1 : Beyin çıkarma işlemi.

Buna göre, ilk olarak GT'deki etiketlerin birleşimiyle iki seviyeli 3D-Maske volumeleri elde edilmiştir (Denklem 5.1). Daha sonra Maske ve ham MRI verisinin noktasal çarpımıyla beyin bölge çıkarımı gerçekleştirilmiştir (Denklem 5.2).

$$Mask(x) = \begin{cases} 1 & \text{if } GT(x) > 0 \\ 0 & \text{else} \end{cases} \quad (5.1)$$

$$Brain(x) = MRI(x) * Mask(x) \quad (5.2)$$

$x$ , volume içerisindeki her bir voxeli göstermektedir. IBSR verilerinde beyin bölgesi sıyrıldığı için, burada bahsedilen işlem sadece MR13 ve MR18 verilerinde gerçekleştirilmiştir.

### 5.2.2 Etiketlerin düzenlenmesi

Kafatası sıyırma işleminden sonra yapılan ikinci işlem sınıf etiket verilerindeki farklılıkların giderilmesidir. Hatırlanacağı gibi bu çalışmada üç farklı sınıfın (GM, WM ve CSF) bölütlenmesi amaçlanmaktadır. IBSR verileri istenilen sınıf etiketlerine sahip olduğu için  $GT$  verileri üzerinde herhangi bir değişiklik yapılmamıştır. Ancak MR13 ve MR18 veri kümelerinde sırayla 8 ve 11 sınıf etiketi bulunduğundan  $GT$  verisi Denklem 5.3'te gösterildiği gibi güncellenmiştir.

$$GT_{MR13}^{new}(x) = \begin{cases} 0 & GT_{MR13}(x) \in [7,8] \\ 1 & GT_{MR13}(x) \in [5,6] \\ 2 & GT_{MR13}(x) \in [3,4] \\ 3 & GT_{MR13}(x) \in [1,2] \end{cases}, \quad GT_{MR18}^{new}(x) = \begin{cases} 0 & GT_{MR18}(x) \in [7-11] \\ 1 & GT_{MR18}(x) \in [5,6] \\ 2 & GT_{MR18}(x) \in [3,4] \\ 3 & GT_{MR18}(x) \in [1,2] \end{cases} \quad (5.3)$$

### 5.2.3 Örtüştürme

MRI verilerinin farklı çözünürlük ve slice kalınlık değerlerinde elde edilmesi, ayrıca her hastanın kafa anatomisinin farklı bir yapıda olması önemli bir sorundur. Uzaysal alandaki bu farklılıkların giderilmesi ve veri boyutlarında standardizasyonun sağlanması için örtüştürme (registration) işlemi uygulanmaktadır [131]. Beyin MRI verilerinin örtüştürmesinde MNI152 (the Montreal Neurological Institute) template'i olarak bilinen standart bir beyin atlası kullanılmaktadır [132]. 152 farklı hastanın MRI verisinden türetilen MNI152 template'i beyin için standart bir koordinat sistemi olarak benimsenmiştir. Bu işlemi gerçekleştirmek için FSL-FLIRT yazılımı kullanılmıştır [111]. Örtüştürme işleminde, her hastanın MRI verisindeki beyin bölümü MNI152 template'i üzerine gelecek şekilde dönüşüm parametreleri hesaplanır. Dönüşüm parametreleri hesaplandıktan sonra veri kümesindeki MRI ve  $GT$  verilerine birlikte aynı dönüşüm uygulanmaktadır. Böylece

etiketlenmiş verinin tutarlılığı korunmuş olur. Çizelge 5.2’de örtüştürme işlemi öncesi ve sonrası veri boyutlarındaki değişim gösterilmektedir.

**Çizelge 5.2 :** Örtüştürme işlemi öncesi ve sonrasında MRI boyutları.

Veri kümesi	Önce		Sonra	
	Volume (pixel)	Voxel (mm)	Volume (pixel)	Voxel (mm)
IBSR	256x256x128	0.93x0.93x1.5	182x182x218	1x1x1
MR13	240x240x48	0.95x0.95x3	182x182x218	1x1x1
MR18	240x240x48	0.95x0.95x3	182x182x218	1x1x1

### 5.2.4 Kırpma ve örnekleme

Örtüştürme işlemi sonrası 182x182x218 boyutlarına sahip olan MRI verileri direk olarak mimarilerin eğitiminde kullanıldığında bellek yetersizliği problemleriyle karşılaşmaktadır. Bu problemin çözümü için kırpma veya örnekleme işlemleri gerçekleştirilmektedir. Kırpma işleminde beyin bölgesi içeride kalacak şekilde MRI boyutları 176x160x192 olarak güncellenmektedir. Deneysel çalışmalar sonucunda, bu boyutlarda mimarilerin eğitilebildiği ve hafıza problemlerinin yaşanmadığı görülmektedir.

Bununla birlikte yüksek boyutlu volume’lerin bütün bir şekilde eğitilmesine alternatif olacak ikinci bir yaklaşım (örnekleme) test edilmektedir. Örnekleme yaklaşımında, kırılan MRI ve GT verileri içerisinde 24x24x24 boyutlu örnekler alınmakta ve mimarilerin eğitimleri gerçekleştirilmektedir. Bu çalışmada, kırpma ve örnekleme işlemlerinin ayrı ayrı bölütleme başarımlarını üzerindeki etkileri detaylı bir şekilde incelenmektedir.

### 5.3 Önerilen Bölütleme Mimarisi

Bu çalışmada 3D MRI bölütlemesi için GAN temelli bir bölütleme mimarisi önerilmektedir (bak Şekil 5.2). Vol2SegGAN olarak adlandırılan bu mimari, üretici ağ (G) ve ayırıcı ağ (D) olmak üzere iki bölümden oluşmaktadır.  $D$ ’nin amacı, gerçek GT verisiyle ile GAN tarafından üretilen sahte etiket verisini birbirinden ayırmaktır. İkili sınıflandırıcı olarak çalışan ayırıcının maliyet fonksiyonu Denklem 5.4’te ifade edilmektedir.

$$L_D = L_2[D(MRI, GT), 1] + L_2[D(MRI, G(MRI)), 0] \quad (5.4)$$

Burada  $MRI$ , ön işlemler sonrası elde edilen standart veridir.  $D(MRI, x)$ , ayırıcı mimarinin  $MRI$  ve  $x$  sınıf etiket verilerine karşılık ürettiği maliyet değerini ifade etmektedir.  $x$  sınıf etiket verisi,  $GT$  veya  $G(MRI)$  çıktısı olmaktadır.  $L_2$  maliyet fonksiyonu, gerçek ile tahmin

edilen değer arasındaki karesel hataların toplamı olan hatayı en aza indirmek için kullanılır. Bu şekilde  $D$ , zamanla gerçek ve sahte sınıf etiket verilerini birbirinden ayırma kabiliyetini kazanmaktadır.

$$L_2 = \sum_{i=0}^n (y_{true}^i - y_{predicted}^i)^2 \quad (5.5)$$

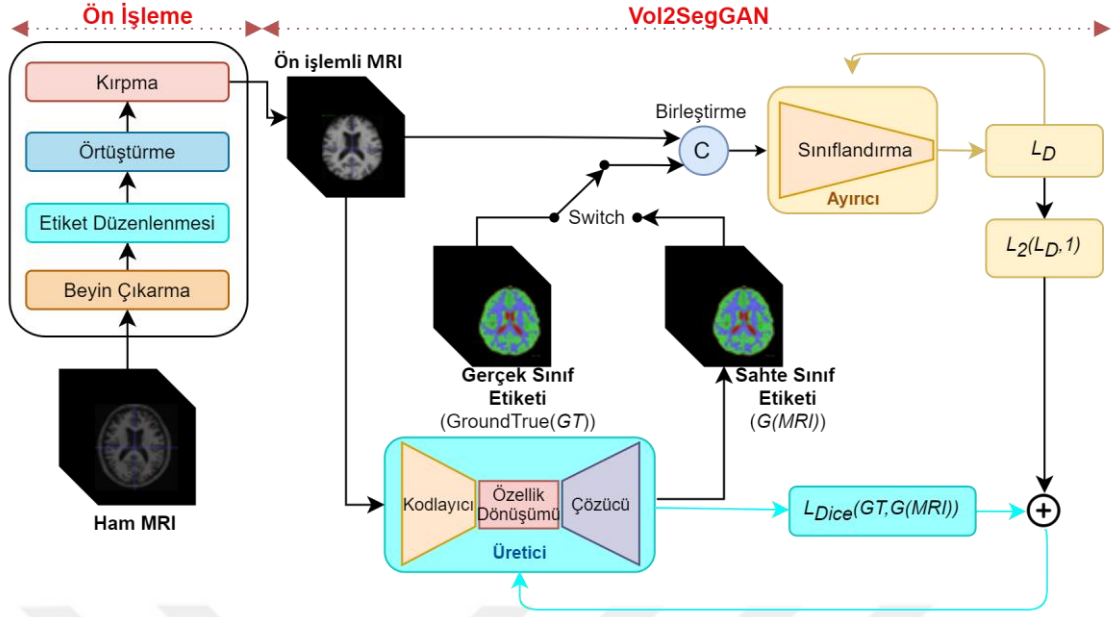
Üretici mimarinin ( $G$ ) hedefi, girdi olarak aldığı  $MRI$  verisinden  $GT$  benzeri sınıf etiket verilerini üretebilmektir. Üreticinin maliyet fonksiyonu Denklem 5.6'da ifade edilmektedir.

$$L_G = \underbrace{L_2[D(MRI, G(MRI)), 1]}_{karesel\ hata} + \alpha \underbrace{L_{Dice}[GT, G(MRI)]}_{benzerlik} \quad (5.6)$$

Buna göre, üreticinin maliyeti iki maliyetin (karesel hata ve benzerlik) toplamına eşittir. Dikkat edilirse,  $L_2$  karesel hata hesaplanırken, sahte üretici çıktısı gerçekmiş gibi  $D$ 'ye verilmektedir. Bu durumda profesyonelleşmiş bir  $D$  yüksek bir maliyet değeri üretecektir. Elde edilen bu yüksek maliyet değeri,  $G$  mimarinin önemli ölçüde ağırlıklarının güncellenmesini tetikler. Böylece üretici mimari zamanla daha yüksek doğruluğa sahip sınıf etiket verilerini üretme yönünde zorlanır.  $L_{Dice}$  maliyet fonksiyonu (bak Denklem 5.7), gerçek ve sahte sınıf etiket verileri arasındaki benzerliği hesaplamaktadır [133].  $\alpha$  parametresiyle benzerlik maliyetinin etkisi ayarlanabilmektedir [123].

$$L_{Dice} = \frac{2|X \cap Y|}{|X| + |Y|} \quad (5.7)$$

Burada  $X$  ve  $Y$ , gerçek ( $GT$ ) ve sahte ( $G(MRI)$ ) sınıf etiket verilerini ifade etmektedir.

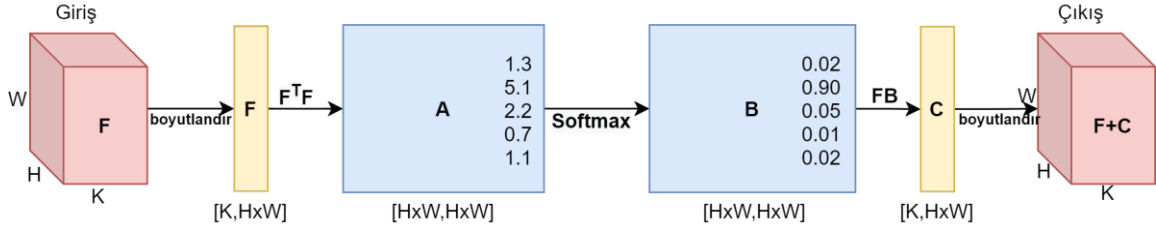


Şekil 5.2 : Önerilen bölütleme mimarisi (Vol2SegGAN).

$G$  mimarisi Şekil 5.4'te detaylı bir şekilde gösterilmektedir. Bu mimari kodlayıcı, özellik aktarıcı ve çözücü aşamalarından oluşmaktadır. Kodlayıcı, ön işlem sonucunda üretilen  $176 \times 160 \times 192$  boyutlu MRI verilerini giriş olarak almaktadır. Yüksek boyutlu giriş verisinin kodlanması için konvolüsyon, normalizasyon, aktivasyon ve çürütme operatörleri kullanılmaktadır. Veri boyutu konvolüsyon operatörün stride parametresi kullanılarak küçültülmektedir. Başlangıç filtre boyutu, kademeli olarak her seviyede iki kat arttırılmaktadır. Böylece veri boyutu küçüldükçe özellik uzayı genişleyen bir yaklaşım benimsendiği görülmektedir.

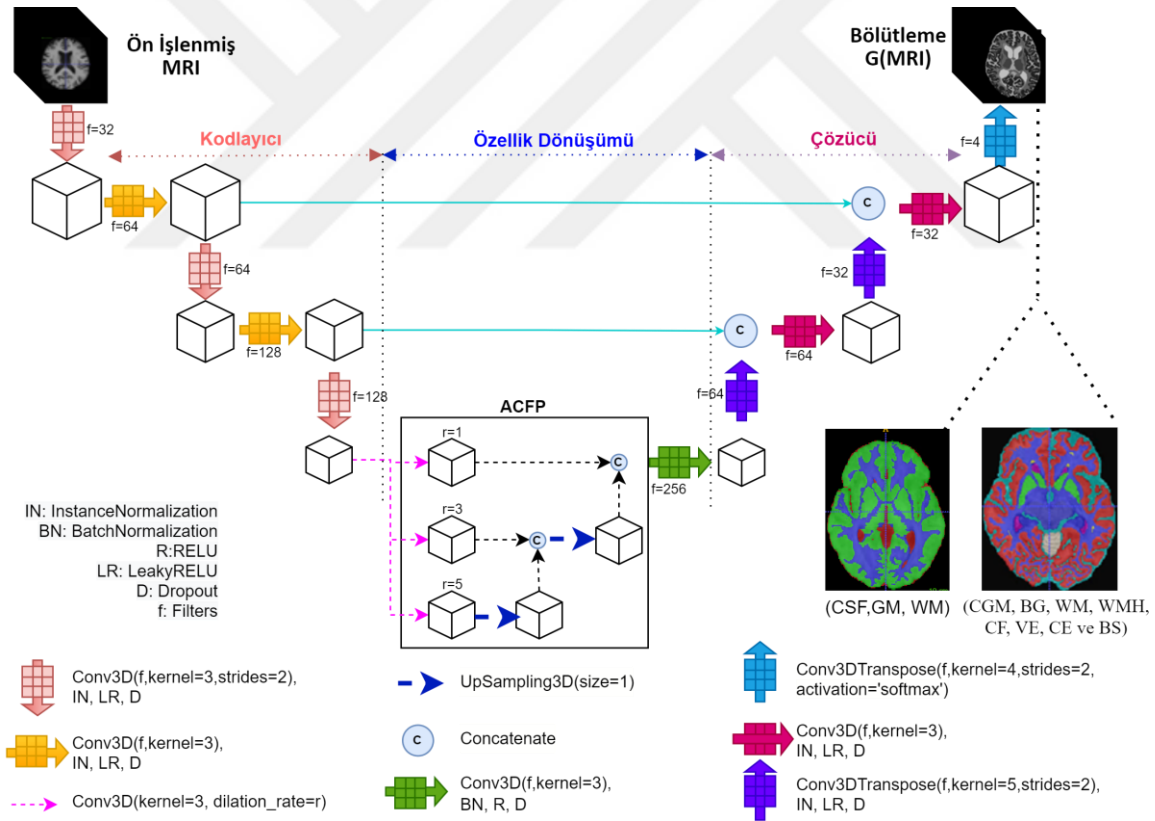
Özellik aktarma bölümü, mimarinin dar boğazı olarak isimlendirilmektedir. Bu bölümde klasik, Atrus evrişimli özellik piramidi (ACFP) [134,135] ve PAM [136] gibi üç farklı tekniğin değişik kombinasyonel kullanımı gerçekleştirilmektedir. Klasik yaklaşımda, kodlanmış öznelikler küçültülmeden 3D filtreden süzülerek çözücüye verilmektedir. ACFP tekniğinde, lokal ve global bilgilerin birlikte değerlendirilmesi sağlanmaktadır. Bu teknik, kodlanmış özneliklerin çok ölçekli bir şekilde incelenmesine imkan verir. Ölçek parametresi ( $r$ ), 1, 3 ve 5 seçilmekte ve kodlanmış veriden üç farklı seviyede öznelikler elde edilmektedir. Farklı seviyedeki öznelikler upsampling operatörüyle birleştirilmektedir. Bu aşamada kullanılan son teknik konum dikkat modülü (PAM)'dür. Bu teknik, öznelikleri pozisyonlarına göre keskinleştirmektedir. Şekil 5.3'te aşamaları gösterilen PAM'de, ilk olarak giriş özneliklerin ( $F$ ) kanal boyutu ( $K$ ) kaldırılmakta ve  $F^T F$  işlemiyle özellikler pozisyon uzayına aktarılmaktadır. Belirli pozisyonlarda sık tekrar eden

öznitelikler vurgulanmakta, nadir görülenler sönümlenmektedir. Daha sonra softmax işlemiyle vurgu ve sönümlenme değerlerinin 0-1 aralığına dönüşümü gerçekleştirilmekte ve sonuçta özniteliklere eklenmektedir. Böylece belirli bir pozisyonda sık tekrar eden öznitelikler güçlendirilmektedir.



Şekil 5.3 : PAM modülü [136].

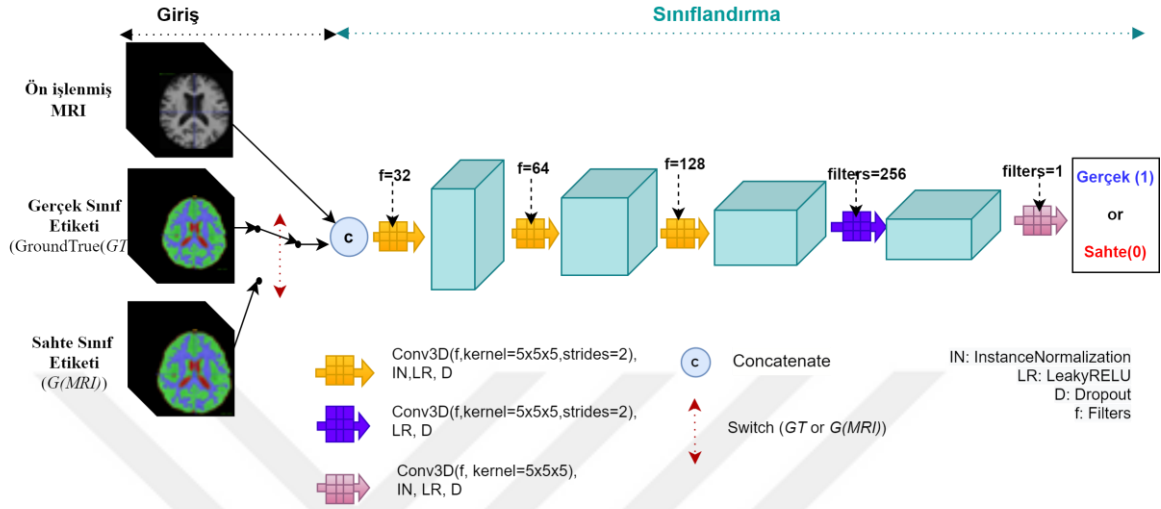
$G$  mimarisinin çözücü aşamasında, transpoze konvolüsyon kullanılarak özniteliklerin genişletilmesi sağlanmaktadır. Kodlama aşamasındaki benzer olarak konvolüsyon (transpose), normalizasyon, aktivasyon ve çürütme operatörleri kullanılmaktadır.



Şekil 5.4 : Üretici mimarisi.

Şekil 5.5'te gösterilen  $D$  mimarisi, giriş ve sınıflandırma aşamalarından oluşmaktadır. Bu mimari MRI verisiyle birlikte sınıf etiket verisini giriş olarak almaktadır. Sınıf etiket verisi gerçek ise mimari çıkışı 1, sahte ise 0 olarak belirlenir. Sahte etiket verisi üretici mimariden gelmektedir. Sınıflandırma aşamasında dört defa tekrar eden konvolüsyon, normalizasyon,

aktivasyon ve çürüme operatörleri kullanılmaktadır. Dördüncü adımda normalizasyon kullanılmamıştır. Üretici ağdakine benzer şekilde konvolüsyon filtre sayısı aşamalı olarak iki kat arttırılmaktadır. Mimarinin son katmanında öznelikler iki sınıftan birine aktarılmaktadır.



Şekil 5.5 : Ayırıcı mimarisi.

#### 5.4 Deneysel Çalışmalar ve Sonuçları

Bu bölümde beyin doku bölgelerinin bölütlenmesiyle ilgili gerçekleştirilen üç farklı deneysel çalışmanın sonuçları paylaşılmaktadır. Uygulamalar, Google Colaboratory platformunda ve Tensorflow/Keras kütüphaneleri kullanılarak gerçekleştirilmiştir. Bölütleme sonuçlarının değerlendirilmesinde Dice, VS [137] ve HD [138] metrikleri kullanılmıştır. Dice ve VS'nin yüksek değerleri, HD'nin ise düşük değerleri iyi bir bölütlemenin yapıldığını göstermektedir.

$$VS = 1 - \frac{||X|| - ||Y||}{|X| + |Y|} \quad (5.8)$$

$$HD(X, Y) = \max(d(X, Y), d(Y, X)) \quad (5.9)$$

$$d(X, Y) = \max(\min(|X - Y|)) \quad (5.10)$$

İlk deneysel çalışmada, önerilen bölütleme mimarisine ACFP ve PAM tekniklerini entegre etmenin bölütleme başarımı üzerindeki etkisi araştırılmaktadır. Üç parça bölütlemenin gerçekleştirildiği bu deneyde, ilgili teknikler üretici mimarinin özellik aktarma kısmına entegre edilmektedir. Entegrasyon işleminde farklı kombinasyonlar göz önüne alınarak dört



farklı mimari inşa edilmiş ve eğitim/test faaliyetleri gerçekleştirilmiştir. İnşa edilen dört farklı mimarinin bölütleme başarımları Çizelge 5.3'te gösterilmektedir. Her bir mimari 8 saat eğitilmiş ve doğruluk değerleri tabloya aktarılmıştır. Eğitim işleminde optimizasyon tekniği olarak Adam (learning\_rate=0.0001, beta\_1=0.5) [72] kullanılmıştır.

**Çizelge 5.3 :** Geliştirilen dört farklı mimarinin üç parça bölütleme sonuçları.

	Dice			VS			HD		
	CSF	GM	WM	CSF	GM	WM	CSF	GM	WM
Vol2SegGAN	0,728	0,868	0,776	0,968	0,983	0,974	2,165	2,216	<b>1,926</b>
Vol2SegGAN+PAM	0,719	0,852	0,754	0,962	0,966	0,950	2,699	3,192	2,937
Vol2SegGAN+ACFP+PAM	0,725	0,863	0,774	0,968	<b>0,988</b>	<b>0,981</b>	2,532	2,429	2,103
Vol2SegGAN+ACFP	<b>0,739</b>	<b>0,875</b>	<b>0,793</b>	<b>0,970</b>	0,983	0,973	<b>2,155</b>	<b>2,206</b>	1,983

Çizelge 5.3'deki bölütleme sonuçları incelendiğinde, Vol2SegGAN+ACFP mimarisi üç metriğe göre CSF bölgesinin, iki metriğe göre GM bölgesinin bölütlenmesinde en yüksek başarımlara sahiptir. Lokal ve global bağlam bilgilerini birleştiren ACFP tekniğinin kullanımı, bölütleme başarımlarını olumlu yönde etkilemiştir. Enteresant sonuçlardan biri, PAM tekniği kullanımının olumsuz bir etki oluşturmasıdır. Dikkat edilirse PAM kullanıldığında elde edilen sonuçlar, klasik teknikle elde edilen sonuçlardan daha düşüktür. Buna ön işlem aşamasında kullanılan örtüştürme işleminin neden olduğu düşünülmektedir. Çünkü örtüştürme işlemiyle beyin bölgelerindeki yapılar benzer pozisyonlara yakınsayarak pozisyon farklılıkları elimine edilmektedir. Bu durum pozisyona göre dikkat mekanizması sağlayan PAM modülünün başarımlarını olumsuz etkilemiştir. Bu aşamada en iyi performansı sergileyen Vol2SegGAN+ACFP mimarisi, önerilen mimari olarak seçilmiş ve sonraki deneylerde önerilen mimari derken bu mimari kastedilmektedir.

İkinci deneysel çalışmada, önerilen mimarinin bölütleme performansının mevcut yaklaşımlar (FSL-FAST[111], VolBrain [114], 3D P-U-Net [122], 3D U-Net [119] ve Vox2Vox [123]) içerisindeki konumu araştırılmaktadır. FSL-FAST ve VolBrain yöntemleri için herhangi bir kodlama gerçekleştirilmeyerek hazır araçlar kullanılmıştır. Derin ağ tabanlı diğer bölütleme mimarileri yaklaşık 8 saat eğitilerek test sonuçları elde edilmiştir. 3D P-U-Net [122] yönteminde, giriş MRI verileri orijinal makalede belirtilen 8x24x24 boyutlarında parçalara bölünmüş ve eğitim gerçekleştirilmiştir. MRI verileri direk olarak 3D U-Net [119] mimarisine verildiğinde hafıza problemleri ortaya çıkmıştır. Bu nedenle veriler 24x24x24 boyutlu parçalara bölündükten sonra eğitim gerçekleştirilmiştir. Vox2Vox [123] ve önerilen

mimari için MRI verileri üzerinde herhangi bir parçalama işlemi gerçekleştirilmemektedir. Üç parça bölütlemesinin gerçekleştirildiği bu deneyin sonuçları Çizelge 5.4'te sunulmuştur.

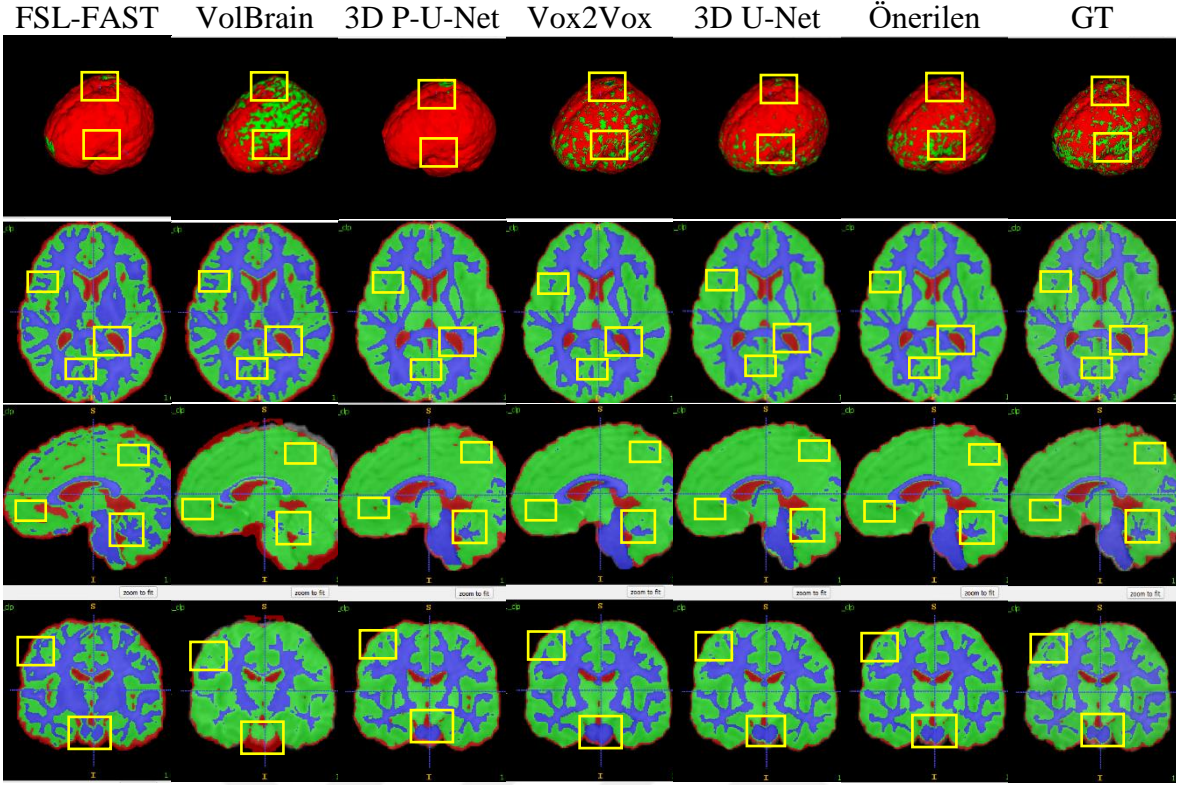
**Çizelge 5.4 :** Önerilen bölütleme mimarisinin mevcut yöntemler içerisindeki konumu.

Models	Dice			VS			HD		
	CSF	GM	WM	CSF	GM	WM	CSF	GM	WM
FSL-FAST	0,448	0,640	0,586	0,613	0,902	0,871	6,293	5,044	5,962
VolBrain	0,594	0,789	<b>0,814</b>	0,931	0,929	0,932	4,552	8,216	8,721
3D P-U-Net	0,696	0,856	0,761	0,790	0,969	0,969	2,452	2,442	2,041
Vox2Vox	0,648	0,829	0,717	0,952	0,977	0,929	3,388	3,760	3,958
3D U-Net	0,735	0,873	0,785	<b>0,976</b>	<b>0,983</b>	<b>0,980</b>	2,234	2,994	<b>1,966</b>
Önerilen	<b>0,739</b>	<b>0,875</b>	0,793	0,970	<b>0,983</b>	0,973	<b>2,155</b>	<b>2,206</b>	1,983

Sonuçlar incelendiğinde, önerilen yöntem, tüm metriklere göre GM bölütlemesini en iyi yapmaktadır. Dice ve HD ye göre CSF'yi de en iyi bir şekilde bölütlemektedir. WM dokusunun bölütlemesinde Volbrain ve 3D U-Net mimarilerinin daha iyi sonuçlar sağladığı görülmektedir. Elde edilen bu sonuçlar, önerilen mimariyi literatürde oldukça saygın bir konuma taşımaktadır.

Bölütleme yöntemlerinin örnek bir MRI üzerindeki sonuçları Şekil 5.6'da görsel olarak gösterilmektedir. GT sınıf etiket verisi son sütunda gösterilmektedir. Elde edilen sonuçlarda CSF bölgesi kırmızı renkle, GM bölgesi yeşil renkle ve WM bölgesi kraliyet mavisi rengiyle ifade edilmektedir. Yöntem sonuçlarında GT'ye en yüksek benzerliğin görüldüğü bölgeler sarı renkli dikkörgen içerisinde vurgulanmaktadır. İlk satırda, elde edilen bölütleme sonuçları 3D-volume şeklinde gösterilmiştir. Volume içerisinde seçilen bir noktanın axial, sagittal ve coronal görüntüleri için elde edilen bölütleme sonuçları sırayla ikinci, üçüncü ve dördüncü satırda gösterilmektedir.

Beyni kapsayan en dış bölgede CSF dokusu bulunduğu için ilk satırda gösterilen 3D-volume sonuçları çoğunlukla kırmızı renkle boyanmıştır. GM bölgesini ifade eden yeşil renk nadiren görülmektedir. Çizelge 5.4'teki sonuçlardan hatırlanacağı üzere 0,739 benzerlik oranıyla en yüksek CSF doğruluğu önerilen yönteme aittir. 3D-volume sonuçları bunu doğrulamaktadır. Yöntemlerin WM bölgesini yakalama doğruluğu ikinci satırda gösterilen sonuçlardan rahatlıkla algılanabilir.



Şekil 5.6 : Örnek bir MRI üzerinde yöntemlerin bölütleme sonuçları.

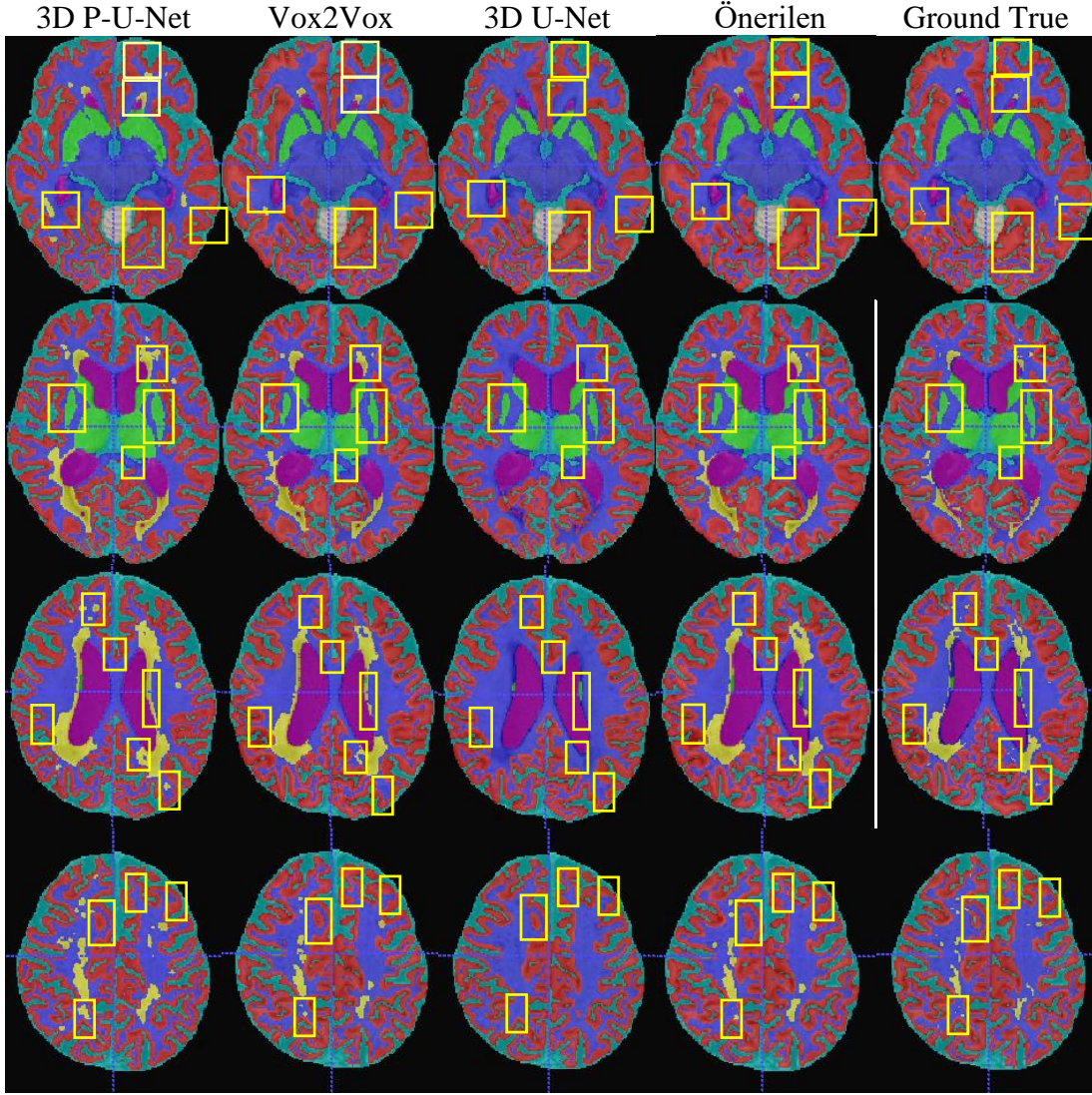
Üçüncü deneysel çalışmada, mimarilerin T1 MRI taramaların üzerinde sekiz farklı dokuya göre bölütleme başarıları incelenmiştir. Bu deneysel çalışmada, ön işleme adımları olarak sadece beyin çıkarma işlemi kullanılmıştır. Aynı zamanda MR18 veri kümesinde infarction ve other dokuları göz ardı edilmiştir [47]. Daha önceki uygulamalarda olduğu gibi mimariler 8 saat eğitilmiş ve test sonuçları Çizelge 5.5’te sunulmuştur. VolBrain modeli üç doku bölgesini ve FSL-FAST modeli ise en fazla 6 doku bölgesini bölütleyebildiği için bu modeller bu deneysel çalışmaya dahil edilmemiştir.

Çizelge 5.5 : Sekiz parçalı bölütleme sonuçları.

Doku	Dice				VS				HD			
	3D P-U-Net	Vox2Vox	3D U-Net	Önerilen	3D P-U-Net	Vox2Vox	3D U-Net	Önerilen	3D P-U-Net	Vox2Vox	3D U-Net	Önerilen
CGM	0,867	<b>0,902</b>	0,899	0,901	0,975	0,971	0,974	<b>0,978</b>	<b>1,410</b>	1,986	1,542	1,597
BG	0,860	0,814	<b>0,920</b>	0,864	0,882	0,970	<b>0,981</b>	0,975	3,226	4,293	<b>2,164</b>	2,738
WM	0,861	<b>0,888</b>	0,885	<b>0,888</b>	0,967	0,976	0,971	<b>0,992</b>	1,937	2,164	2,033	<b>1,821</b>
WMH	<b>0,678</b>	0,377	0	0,460	<b>0,794</b>	0,406	0	0,488	<b>3,761</b>	16,304	None	13,937
CF	0,892	0,916	<b>0,924</b>	0,914	0,966	0,990	0,982	<b>0,994</b>	1,410	1,638	<b>1,353</b>	<b>1,353</b>
VE	0,935	0,929	<b>0,950</b>	0,942	0,948	<b>0,982</b>	0,967	0,963	5,784	4,535	3,895	<b>2,885</b>
CE	0,948	0,933	<b>0,956</b>	0,954	0,976	0,968	0,983	<b>0,991</b>	3,376	4,837	<b>2,845</b>	3,314
BS	0,913	0,912	0,933	<b>0,939</b>	0,918	<b>0,985</b>	0,955	0,955	2,849	2,753	2,638	<b>2,323</b>

Üçüncü deneysel çalışmanın sonuçları incelendiğinde, önerilen mimarinin bölütleme performansı VS ve HD'a göre en iyi, Dice'a göre 3D U-Net'den sonra en iyi ikinci seviyede olduğu görülmektedir. Dice metriğine göre 3D U-Net mimarisi dört beyin dokusunda (BG, CF, VE ve CE) en iyi performansı sergilemiştir. Önerilen mimari ise WM ve BS dokularında en yüksek bölütleme performansı sağlamış ve diğer dokularda genellikle en iyi ikinci performansı göstermiştir. VS ve HD metrikleri incelendiğinde genel olarak önerilen mimarinin yüksek performans sergilediği görülmüştür.

Mimarilerin örnek bir MRI verisinin 20, 25, 30 ve 35 numaralı görüntülerindeki bölütleme sonuçları Şekil 5.7'de gösterilmektedir. Sonuçların daha net kıyaslanabilmesi için açık sarı renkli dikdörtgenel bölgeye odaklanılabilir. 3D U-Net yöntemi, WMH dokusunu (sarı bölge) bölütleyemediği görülmektedir. Önerilen mimarinin GT'ye oldukça benzer sonuçlar elde ettiği görülmektedir.



Şekil 5.7 : Örnek bir MRI üzerindeki bölütleme sonuçları (yukarıdan aşağıya doğru 20, 25, 30 ve 35 nolu görüntüler).

Çizelge 5.6’da bir MRI verisinin bölütlenme yöntemleri kullanılarak etiketlerine dönüştürülme süreleri ve mimarilerin toplam parametre sayıları ifade edilmektedir. Çalışma zamanı CPU ve GPU olarak iki farklı şekilde hesaplanmıştır. Derin öğrenme yaklaşımını kullanan yöntemlerin parametre sayıları verilmektedir.

Çizelge 5.6 : Mimamarilerin MRI bölütleme süreleri.

Modeller	CPU	GPU	Parametre (mil.)
FSL-FAST	4-5 dk.	-	-
VolBrain	13-14 dk.	-	-
3D P-U-Net	35-36 dk.	14-15 sn.	12,065
Vox2Vox	29-30 sn.	0-1 sn.	293,939
3D U-Net	13-14 dk.	8-9 sn.	16,325
Önerilen	<b>11-12 sn.</b>	<b>0-1 sn.</b>	<b>6,883</b>

Sonuçlar incelendiğinde, önerilen bölütleme mimarisinin hem CPU hem de GPU kullanımında en yüksek bölütleme hızına sahip olduğu göstermektedir. Önerilen mimarinin GAN temelli bölütleme mimarisinin oldukça etkileyici bir performans sergilediği düşünülmektedir. Elde edilen sonuçlar hızlı bölütleme ihtiyacının olduğu uygulamalarda önerilen yaklaşımın tercih edilebileceği ortaya koymaktadır. Ayrıca akranlarıyla kıyaslandığında önerilen mimarinin en düşük parametreye sahip olduğu ve hesaplama maliyetlerinin düşük çıkmasında bununda bir etken olduğu görülmektedir.



## 6. SONUÇLAR

Gerçek görüntülere oldukça benzeyen sentetik görüntülerin otomatik üretilmesi problemine çözüm olarak sunulan Çekişmeli Üretken Ağlar (GAN), derin öğrenme alanında önemli buluşlarından biri olarak görülmektedir. Bu tez çalışmasında, mevcut üretici ağ mimarileri detaylı bir şekilde incelenmiş, üstün/eksik yönleri belirlenerek güncel problemlerin çözümüne ışık tutacak üretken ağlar temelli yeni mimariler önerilmiş ve literatüre katkılar sunulmuştur.

Tez çalışması kapsamında sentetik görüntü üretimi, çözünürlük iyileştirme, sinyal-görüntü dönüşümü ve bölütleme gibi dört farklı güncel problemin çözümüne odaklanılmış ve her problem için üretken ağ temelli özgün çözüm yaklaşımları sunulmuştur. Problemlerin çözümleri için literatürde bulunan mevcut çözüm yaklaşımları ile önerilen yaklaşım doğruluk ve performans gibi kriterlerle karşılaştırılmış ve elde edilen sonuçlar bilimsel dergilerde makale şeklinde yayınlanmıştır. Her bir problem özelinde yapılan akademik çalışma tezin farklı bir bölümünde sunulmaktadır.

Buna göre yapılan ilk çalışmada, GAN mimarilerin sentetik görüntü üretim kabiliyetleri incelenmiş ve hibrit bir GAN mimarisi önerilmiştir. Yedi farklı üretken ağ modeli ile birlikte önerilen hibrit mimari (cDCGAN) performanslarının incelenebilmesi için MNIST ve Fashion-MNIST veri kümeleri kullanılmıştır. Önerilen hibrit mimari, DCGAN ve cGAN modellerinin avantajlarını birleştirmektedir. Kıyaslama metrikleri olarak FID benzerliği ve eğitilmiş CNN çıktısı kullanılmıştır. Elde edilen sonuçlara göre, FID benzerlik metriğine LSGAN mimarisini veri kümelerinin sınıfları hem ayrı ayrı hem de toplu sonuçlarda en iyi performansı sergilediği görülmüştür. CNN tabanlı eğitilmiş mimari çıktıları değerlendirildiğinde ise LSGAN modeliyle üretilen görsellerin %97.32 oranında sınıflama doğruluğu sağladığı görülürken, cDCGAN mimarisi 5 sınıfta en iyi başarıyı sağlamıştır. Sonuçlar genel bir şekilde ele alındığında, LSGAN gürültü seviyesi yüksek, ancak sınıflandırma doğruluğu yüksek görseller üretirken, cDCGAN gürültü seviyesi düşük, ancak sınıflandırma doğruluğu orta seviyeli görseller ürettiği görülmektedir.

Yapılan ikinci çalışmada, GAN tabanlı çözünürlük iyileştirme yönteminin (SRGAN) klasik yaklaşımlarla kıyaslamasını içeren bir çalışma yapılmıştır. Klasik çözünürlük iyileştirme

(Bilinear ve Bicubic) ve gürültü eleme yöntemlerinin (Ortanca, Ortalama ve Adaptif) birlikte kullanıldığı bu çalışmada, farklı büyütme altında SRGAN mimarisinin mevcut yaklaşımlara olan üstünlüğü gözlemlenmiştir. Uygulama çalışmasında Camelyon17 veri kümesindeki meme kanseri görüntüleri ve SSIM, PSNR ve FFT\_MSE benzerlik metrikleri kullanılmıştır. Elde edilen sonuçlar incelendiğinde, 2 kat büyültmede Bicubic, 4, 8 veya 16 kat büyültmede SRGAN yönteminin en doğru büyültme sağladığı gözlemlenmiştir. Benzer şekilde gürültü eleme sonuçları incelendiğinde SRGAN yönteminin klasik yöntemlere üstünlüğü açıkça görülmektedir.

Üçüncü çalışmada tek boyutlu sinyallerden iki boyutlu görsel üretim problemi ele alınmıştır. Bu kapsamda EEG sinyallerini girdi alan ve bakılan görseli çıktı olarak üreten bir çalışma yapılmıştır. Mevcut GAN çözümlerinde bulunduğu bu alana yeni bir mimari (EEG-GAN) kazandırılmıştır. Önerilen mimari, Fourier ve VAE kodlayıcı tekniğinin birlikte kullanılmasına dayanmaktadır. Böylece, üretici ağı girdi olarak gürültü vektörü yerine EEG sinyallerinin Fourier dönüşümü sonrası ortalama ve standart sapmalarına dayalı kodlanmış bir hali verilmektedir. Gerçekleştirilen deneysel çalışmalarda, yöntemlerin üç farklı gürültü altında görüntü üretim performansları incelenmiştir. MNIST ve Fashion-MNIST veri kümelerinin kullanıldığı bu çalışmada benzerlik metriği olarak önceden eğitilen bir CNN kullanılmıştır. Elde edilen sonuçlar incelendiğinde, klasik GAN mimarileriyle %93 benzerliğe sahip görüntüler üretilirken, önerilen mimariyle bu oran %95-100 aralığına yükseltilmiştir.

Dördüncü çalışmada, güncelliğini sürekli muhafaza eden bölütleme problemi ele alınmıştır. Özellikle medikal alanda hassas bölütleme yaklaşımlarına halen büyük ihtiyaç duyulmaktadır. Bu kapsamda T1 ağırlığına sahip 3D-MRI verileri kullanılarak beyin bölümlerinin üç seviye (CSF, GM ve WM) ve sekiz seviye (CGM, BG, WM, WMH, CF, VE, CE ve BS) olarak bölütlenme işlemleri yapılmıştır. Klasik bölütleme yaklaşımlarına ek olarak ACFP ve PAM modüllerinin kullanıldığı özgün bir bölütleme mimarisi önerilmektedir. Yöntemlerin eğitim ve test faaliyetleri için IBSR18, MR13 ve MR18 veri setleri kullanılırken, başarımlar değerlendirilmede Dice, VS ve HD benzerlik metrikleri kullanılmıştır. Eğitim setinin inşasında parça ve volüm olmak üzere iki farklı kullanımın doğruluk üzerine etkisi incelenmiştir. Elde edilen sonuçlar ışığında, volüm kullanımının parça kullanımından hızlı çalıştığı, üç seviyeli bölütlemeye önerilen mimarinin Dice ve HD metriklerine göre CSF ve GM dokularında en iyi bölütleme sağladığı, WM dokusunda ise en yüksek ikinci bölütleme başarısı sağladığı görülmüştür. Ayrıca VS metriğine göre 3D-U-



Net mimarisinden sonra en yüksek bölütleme performansı sergilediği gözlemlenmiştir. Sekiz seviyeli bölütleme sonuçları incelendiğinde, önerilen mimarinin VS ve HD metriklerine göre en iyi, Dice metriğine göre ise 3D U-Net mimarisinden sonra en yüksek bölütleme performansını sergilediği görülmüştür.

Bununla birlikte önerilen mimarinin parametre sayısının (6.8 milyon) düşük olması bölütleme süresinin düşmesini sağlamaktadır. Örnek bir 3D-MRI verisinin bölütlenme süresi, CPU kullanıldığında 11-12 saniye olurken, GPU kullanıldığında 0-1 saniye aralığına düşmektedir. Elde edilen sonuçlar değerlendirildiğinde, önerilen bölütleme mimarisiyle kaydadeğer bir bölütleme sonucunun hızlı bir şekilde elde edilebildiği görülmektedir.



## KAYNAKLAR

- [1] **Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A. E., & Arshad, H.** (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11), e00938. <https://doi.org/10.1016/j.heliyon.2018.e00938>
- [2] **Doğan, F., & Türkoğlu, İ.** (2019). Derin Öğrenme Modelleri ve Uygulama Alanlarına İlişkin Bir Derleme. *DÜMF Mühendislik Dergisi*, 10(2), 409–445. <https://doi.org/10.24012/dumf.411130>
- [3] **Şeker, A., Diri, B., & Balık, H.** (2017). Derin Öğrenme Yöntemleri ve Uygulamaları Hakkında Bir İnceleme. *Gazi Journal of Engineering Sciences*, 3(3), 47–64.
- [4] **Tu, J. V.** (1996). Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of Clinical Epidemiology*, 49(11), 1225–1231. [https://doi.org/10.1016/S0895-4356\(96\)00002-9](https://doi.org/10.1016/S0895-4356(96)00002-9)
- [5] **LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P.** (1998). Gradient-based learning applied to document recognition, *Proceedings of the IEEE*, 86(11), 2278–2323. <https://doi.org/10.1109/5.726791>
- [6] **Ravi, D., Wong, C., Deligianni, F., Berthelot, M., Andreu-Perez, J., Lo, B., & Yang, G. Z.** (2017). Deep Learning for Health Informatics, *IEEE Journal of Biomedical and Health Informatics*, 21(1), 4–21. <https://doi.org/10.1109/JBHI.2016.2636665>
- [7] **Alex, K., Ilya, S., & E., H. G.** (2012). ImageNet Classification with Deep Convolutional Neural Networks, *Proceedings of the Advances in Neural Information Processing Systems*, 1097–1105. <https://doi.org/10.1145/3065386>
- [8] **Dong, S., Wang, P., & Abbas, K.** (2021). A survey on deep learning and its applications, *Computer Science Review*, 40, 100379. <https://doi.org/10.1016/j.cosrev.2021.100379>
- [9] **Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A.** (2015). Going deeper with convolutions, *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 91(8), (pp.1–9). Boston, MA, USA, June 7-12. <https://doi.org/10.1109/CVPR.2015.7298594>
- [10] **Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L.** (2015). ImageNet Large Scale Visual Recognition Challenge, *International Journal of Computer Vision*, 115(3), (pp.211–252). <https://doi.org/10.1007/s11263-015-0816-y>

- [11] **Küçük, D., & Arıcı, N.** (2018). Doğal Dil İşlemede Derin Öğrenme Uygulamaları Üzerine Bir Literatür Çalışması, *Uluslararası Yönetim Bilişim Sistemleri ve Bilgisayar Bilimleri Dergisi*, 2(2), 76–86
- [12] **Mikolov, T., Karafiát, M., Burget, L., Černocký, J., & Khudanpur, S.** (2010). Recurrent Neural Network Based Language Model, *INTERSPEECH 2010*, (pp.1045–1048). Makuhari, Chiba, Japan, September 26-30. [https://www.isca-speech.org/archive/interspeech\\_2010/i10\\_1045.html](https://www.isca-speech.org/archive/interspeech_2010/i10_1045.html)
- [13] **Hochreiter, S., & Schmidhuber, J.** (1997). Long Short-Term Memory, *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- [14] **Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K.** (2019). BERT: Pre-training of deep bidirectional transformers for language understanding, *Proceedings of the 2019 Conference of the North, 1*, (pp.4171–4186). Minneapolis, Minnesotai, June. <https://doi.org/10.18653/v1/N19-1423>
- [15] **Zhao, X., Jia, M., & Lin, M.** (2020). Deep Laplacian Auto-encoder and its application into imbalanced fault diagnosis of rotating machinery, *Measurement*, 152, 107320. <https://doi.org/10.1016/j.measurement.2019.107320>
- [16] **Li, Q., Cai, W., Wang, X., Zhou, Y., Feng, D. D., & Chen, M.** (2014). Medical image classification with convolutional neural network, *2014 13th International Conference on Control Automation Robotics and Vision, ICARCV 2014*. (pp.844–848). Singapore, December 10-12. <https://doi.org/10.1109/ICARCV.2014.7064414>
- [17] **Li, F., Tran, L., Thung, K. H., Ji, S., Shen, D., & Li, J.** (2015). A Robust Deep Model for Improved Classification of AD/MCI Patients, *IEEE Journal of Biomedical and Health Informatics*, 19(5), 1610–1616. <https://doi.org/10.1109/JBHI.2015.2429556>
- [18] **Sirinukunwattana, K., Raza, S. E. A., Tsang, Y. W., Snead, D. R. J., Cree, I. A., & Rajpoot, N. M.** (2016). Locality Sensitive Deep Learning for Detection and Classification of Nuclei in Routine Colon Cancer Histology Images, *IEEE Transactions on Medical Imaging*, 35(5), 1196–1206. <https://doi.org/10.1109/TMI.2016.2525803>
- [19] **Girshick, R., Donahue, J., Darrell, T., & Malik, J.** (2014). Rich feature hierarchies for accurate object detection and semantic segmentation, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (pp.580–587). Columbus, OH, USA, June 23-28. <https://doi.org/10.1109/CVPR.2014.81>
- [20] **Ren, S., He, K., Girshick, R., & Sun, J.** (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>
- [21] **Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y.** (2014). Generative Adversarial Networks, *In Proceedings of the 27th International Conference on Neural Information Processing Systems - (NIPS'14)*, 2, (pp.2672–2680). Montreal, Canada, December.

- [22] **Xu, L., Zeng, X., Li, W., & Huang, Z.** (2020). Multi-granularity generative adversarial nets with reconstructive sampling for image inpainting, *Neurocomputing*, 402, 220–234. <https://doi.org/10.1016/j.neucom.2020.04.011>
- [23] **Hedjazi, M. A., & Genc, Y.** (2021). Efficient texture-aware multi-GAN for image inpainting, *Knowledge-Based Systems*, 217, 106789. <https://doi.org/10.1016/j.knosys.2021.106789>
- [24] **Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., & Shi, W.** (2017). Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp.105–114). IEEE, July 21-26. <https://doi.org/10.1109/CVPR.2017.19>
- [25] **Gupta, R., Sharma, A., & Kumar, A.** (2020). Super-Resolution using GANs for Medical Imaging, *Procedia Computer Science*, 173(2019), 28–35. <https://doi.org/10.1016/j.procs.2020.06.005>
- [26] **Zhu, J. Y., Park, T., Isola, P., & Efros, A. A.** (2017). Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks, *Proceedings of the IEEE International Conference on Computer Vision*, (pp.2242–2251). IEEE, October 22-29. <https://doi.org/10.1109/ICCV.2017.244>
- [27] **Yi, Z., Zhang, H., Tan, P., & Gong, M.** (2017). DualGAN: Unsupervised Dual Learning for Image-to-Image Translation, *Proceedings of the IEEE International Conference on Computer Vision*, (pp.2868–2876). IEEE, October 22-29. <https://doi.org/10.1109/ICCV.2017.310>
- [28] **Mao, Xiaofeng, Wang, S., Zheng, L., & Huang, Q.** (2018). Semantic invariant cross-domain image generation with generative adversarial networks, *Neurocomputing*, 293, 55–63. <https://doi.org/10.1016/j.neucom.2018.02.092>
- [29] **Antipov, G., Baccouche, M., & Dugelay, J. L.** (2018). Face aging with conditional generative adversarial networks, *Proceedings - International Conference on Image Processing, ICIP*, (pp.2089–2093). Beijing, China, September 17-20. <https://doi.org/10.1109/ICIP.2017.8296650>
- [30] **Li, C., & Wand, M.** (2016). Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks, In *ECCV 2016: Computer Vision – ECCV 2016: Vol. 9907 LNCS* (pp.702–716). [https://doi.org/10.1007/978-3-319-46487-9\\_43](https://doi.org/10.1007/978-3-319-46487-9_43)
- [31] **Tirupattur, P., Spampinato, C., Rawat, Y. S., & Shah, M.** (2018). ThoughtViz: Visualizing human thoughts using generative adversarial network, *MM 2018 - Proceedings of the 2018 ACM Multimedia Conference*, (pp.950–958). Seoul, Republic of Korea, October. <https://doi.org/10.1145/3240508.3240641>
- [32] **Duarte, A., Roldan, F., Tubau, M., Escur, J., Pascual, S., Salvador, A., Mohedano, E., McGuinness, K., Torres, J., & Giro-I-Nieto, X.** (2019). Wav2Pix: Speech-conditioned Face Generation Using Generative Adversarial Networks, *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, (pp.8633–8637). IEEE, May 12-17. <https://doi.org/10.1109/ICASSP.2019.8682970>

- [33] **Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., & Metaxas, D.** (2017). StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks, *Proceedings of the IEEE International Conference on Computer Vision*, (pp.5908–5916). IEEE, October 22-29. <https://doi.org/10.1109/ICCV.2017.629>
- [34] **A Beginner’s Guide to Generative Adversarial Networks (GANs).** (n.d.). Eriřim: 28 Ekim 2018. <https://wiki.pathmind.com/generative-adversarial-network-gan>
- [35] **Langr, J., & Bok, V.** (2018). *GANs in Action*. MEAP. Manning Publications.
- [36] **Creswell, A., & Bharath, A. A.** (2016). Adversarial Training for Sketch Retrieval, *ECCV 2016: Computer Vision – ECCV 2016 Workshops*, (pp. 798–809). [https://doi.org/10.1007/978-3-319-46604-0\\_55](https://doi.org/10.1007/978-3-319-46604-0_55)
- [37] **Derin Öğrenme Uygulamalarında En Sık Kullanılan Hiper-Parametreler.** (n.d.). Eriřim: 15 Eylül 2018. <http://shorturl.at/mprV3>
- [38] **THE MNIST DATABASE of handwritten digits.** (n.d.). Eriřim: 22 Kasım 2019, <http://yann.lecun.com/exdb/mnist/>
- [39] **Xiao, H., Rasul, K., & Vollgraf, R.** (2017). Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms, *ArXiv*, 1–6. <https://arxiv.org/abs/1708.07747>
- [40] **Çelik, G., & Talu, M. F.** (2020). Çekiřmeli Üretken Ağ Modellerinin Görüntü Üretme Performanslarının İncelenmesi, *Balıkesir Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 22(1), 181–192. <https://doi.org/10.25092/baunfbed.679608>
- [41] **Litjens, G., Banid, P., Bejnordi, B. E., Geessink, O., Balkenhol, M., Bult, P., Halilovic, A., Hermsen, M., Loo, R. van de, Vogels, R., Manson, Q. F., Stathonikos, N., Baidoshvili, A., Diest, P. van, Wauters, C., Dijk, M. van, & Laak, J. van der.** (2018). 1399 H&E-stained sentinel lymph node sections of breast cancer patients: the CAMELYON dataset, *GigaScience*, 7(6). <https://doi.org/https://doi.org/10.1093/gigascience/giy065>
- [42] **Çelik, G., & Talu, M. F.** (2020). Resizing and cleaning of histopathological images using generative adversarial networks, *Physica A: Statistical Mechanics and Its Applications*, 554, 122652. <https://doi.org/10.1016/j.physa.2019.122652>
- [43] **Bashivan, P., Rish, I., Yeasin, M., & Codella, N.** (2016). Learning representations from EEG with deep recurrent-convolutional neural networks, *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, (pp.1–15). San Juan, Puerto Rico, May 2-4.
- [44] **Çelik, G., & Talu, M. F.** (2021). EEG sinyallerinden bakılan görselin üretilmesi, *Pamukkale University Journal of Engineering Sciences*, 27(2), 129–138. <https://doi.org/10.5505/pajes.2020.76399>
- [45] **IBSR.** (n.d.). Eriřim: 10 Haziran 2020, <https://www.nitrc.org/projects/ibsr>
- [46] **Mendrik, A. M., Vincken, K. L., Kuijf, H. J., Breeuwer, M., Bouvy, W. H., De Bresser, J., Alansary, A., De Bruijne, M., Carass, A., El-Baz, A., Jog, A., Katyal, R., Khan, A. R., Van Der Lijn, F., Mahmood, Q., Mukherjee, R., Van Opbroek, A., Paneri, S., Pereira, S., ... Viergever, M. A.** (2015). MRBrainS Challenge: Online Evaluation Framework for Brain Image Segmentation in 3T MRI Scans, *Computational Intelligence and Neuroscience*, 2015. <https://doi.org/10.1155/2015/813696>

- [47] **MRBrainS18, Grand Challenge on MR Brain Segmentation at MICCAI 2018.** (n.d.). Erişim: 6 Ağustos 2020, <https://mrbrains18.isi.uu.n>.
- [48] **Mirza, M., & Osindero, S.** (2014). Conditional Generative Adversarial Nets. *ArXiv*, 1–7. <https://arxiv.org/abs/1411.1784>
- [49] **Radford, A., Metz, L., & Chintala, S.** (2015). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, *ArXiv*, 1–16. <https://arxiv.org/abs/1511.06434>
- [50] **Wu, X., Xu, K., & Hall, P.** (2017). A survey of image synthesis and editing with generative adversarial networks, *Tsinghua Science and Technology*, 22(6), 660–674. <https://doi.org/10.23919/TST.2017.8195348>
- [51] **Odena, A., Olah, C., & Shlens, J.** (2016). Conditional Image Synthesis With Auxiliary Classifier GANs, *ICML'17 Proceedings of the 34<sup>th</sup> International Conference on Machine Learning*, (pp.2642–2651). JMLR.org, August 06-11.
- [52] **Chen, X., Duan, Y., Houthoofd, R., Schulman, J., Sutskever, I., & Abbeel, P.** (2016). InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets, *30th Conference on Neural Information Processing Systems (NIPS 2016)*, (pp.2180–2188). Barcelona, Spain, December 5-10.
- [53] **Odena, A.** (2016). Semi-Supervised Learning with Generative Adversarial Networks, *ArXiv*, (pp.1–3). <https://doi.org/arXiv:1504.01391>.
- [54] **Semi-supervised learning with Generative Adversarial Networks (GANs).** (n.d.). Erişim: 5 Aralık 2018. <http://shorturl.at/fFH13>
- [55] **Arjovsky, M., Chintala, S., & Bottou, L.** (2017). Wasserstein Generative Adversarial Networks, *Proceedings of the 34th International Conference on Machine Learning, ICML'17*, (pp. 214–223).
- [56] **Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A.** (2017). Improved Training of Wasserstein GANs, *Proceedings of the 31st International Conference on Neural Information Processing Systems*, (pp. 5769–5779).
- [57] **Mao, X., Li, Q., Xie, H., Lau, R. Y. K., Wang, Z., & Smolley, S. P.** (2017). Least Squares Generative Adversarial Networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, (pp.2813–2821). IEEE, October 22-29. <https://doi.org/10.1109/ICCV.2017.304>
- [58] **Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S.** (2017). GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, (pp.6629–6640). <https://dl.acm.org/doi/10.5555/3295222.3295408>
- [59] **Zeiler, M. D.** (2012). ADADELTA: An Adaptive Learning Rate Method, *ArXiv*. <http://arxiv.org/abs/1212.5701>
- [60] **Zhao, L., Bai, H., Liang, J., Zeng, B., Wang, A., & Zhao, Y.** (2019). Simultaneous color-depth super-resolution with conditional generative adversarial networks. *Pattern Recognition*, 88, 356–369. <https://doi.org/10.1016/j.patcog.2018.11.028>

- [61] **Wei, S., Zhou, X., Wu, W., Pu, Q., Wang, Q., & Yang, X.** (2018). Medical image super-resolution by using multi-dictionary and random forest, *Sustainable Cities and Society*, 37, 358–370. <https://doi.org/10.1016/j.scs.2017.11.012>.
- [62] **Duchon, C. E.** (1979). Lanczos filtering in one and two dimensions, *Journal of Applied Meteorology*, 18(8), 1016–1022.
- [63] **Keys, R. G.** (1981). Cubic Convolution Interpolation for Digital Image Processing, in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(6), 1153-1160. doi: 10.1109/TASSP.1981.1163711
- [64] **Gavader, A. B., & Rajpurohit, V. S.** (2015). Single Frame Image Resolution Enhancement Using Super Resolution, *International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)*, 14(2), 293–298.
- [65] **Rajapollu, P. R., & Mankar, V. R.** (2017). Bicubic Interpolation Algorithm Implementation for Image Appearance Enhancement, *Ijcst*, 8(2), 23–26.
- [66] **Mazurowski, M. A., Buda, M., Saha, A., & Bashir, M. R.** (2018). Deep learning in radiology: An overview of the concepts and a survey of the state of the art with focus on MRI, *Journal of Magnetic Resonance Imaging*, 153, 1–27. <https://doi.org/10.1002/jmri.26534>
- [67] **Simonyan, K., & Zisserman, A.** (2015). Very deep convolutional networks for large-scale image recognition, *CoRR*, abs/1409.1556.
- [68] **Görüntü Restorasyonu** (n.d). Erişim: 20 Aralık 2019. <http://shorturl.at/jzPX4>
- [69] **Gao, H., Hu, M., Gao, T., & Cheng, R.** (2019). Robust detection of median filtering based on combined features of difference image, *Signal Processing: Image Communication*, 72(January), 126–133. <https://doi.org/10.1016/j.image.2018.12.014>
- [70] **Kumar, N.R., & Kumar, J.U.** (2015). A Spatial Mean and Median Filter For Noise Removal in Digital Images, *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 4(1), 246–253.
- [71] **Camelyon17: Breast cancer metastases in lymph nodes and the TNM classification system.** (n.d.). Erişim: 20 Ocak 2019. <http://shorturl.at/wzL07>
- [72] **Kingma, D. P., & Ba, J.** (2015). Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. San Diego, CA, USA, May 7-9.
- [73] **Training and investigating Residual Nets** (n.d). Erişim : 28 Şubat 2019. <http://torch.ch/blog/2016/02/04/resnets.html>
- [74] **Ioffe, S., & Szegedy, C.** (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, *ArXiv*. <https://arxiv.org/abs/1502.03167>
- [75] **Daoud, A. O., Tsehayae, A. A., & Fayek, A. R.** (2017). A guided evaluation of the impact of research and development partnerships on university, industry, and government, *Canadian Journal of Civil Engineering*, 44(4), 253–263. <https://doi.org/10.1139/cjce-2016-0381>
- [76] **Wang, Z., & Bovik, A. C.** (2009). Mean squared error: Lot it or leave it? A new look at signal fidelity measures, *IEEE Signal Processing Magazine*, 26(1), 98–117. <https://doi.org/10.1109/MSP.2008.930649>

- [77] **Ishak, A. B.** (2017). Choosing parameters for Rényi and Tsallis entropies within a two-dimensional multilevel image segmentation framework, *Physica A: Statistical Mechanics and Its Applications*, 466, 521–536. <https://doi.org/10.1016/j.physa.2016.09.053>
- [78] **Sarbishei, O., & Radecka, K.** (2011). Analysis of Mean-Square-Error (MSE) for fixed-point FFT units, *Proceedings - IEEE International Symposium on Circuits and Systems*, (pp.1732–1735). IEEE, May 15-18. <https://doi.org/10.1109/ISCAS.2011.5937917>
- [79] **Spampinato, C., Palazzo, S., Kavasidis, I., Giordano, D., Souly, N., & Shah, M.** (2017). Deep learning human mind for automated visual classification, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (pp.4503–4511). IEEE, July 21-26. <https://doi.org/10.1109/CVPR.2017.479>.
- [80] **Reading EEGs: A Practical Approach.** (n.d). Erişim: 10 Ocak 2020. <http://shorturl.at/izGIR>
- [81] **Pfurtscheller, G., & Lopes Da Silva, F. H.** (1999). Event-related EEG/MEG synchronization and desynchronization: Basic principles, *Clinical Neurophysiology*, 110(11), 1842–1857. [https://doi.org/10.1016/S1388-2457\(99\)00141-8](https://doi.org/10.1016/S1388-2457(99)00141-8)
- [82] **Grosse-Wentrup, M., & Buss, M.** (2008). Multiclass Common Spatial Patterns and Information Theoretic Feature Extraction, *IEEE Transactions on Biomedical Engineering*, 55(8), 1991–2000. <https://doi.org/10.1109/TBME.2008.921154>.
- [83] **Amin, S. U., Alsulaiman, M., Muhammad, G., Mekhtiche, M. A., & Shamim Hossain, M.** (2019). Deep Learning for EEG motor imagery classification based on multi-layer CNNs feature fusion, *Future Generation Computer Systems*, 101, 542–554. <https://doi.org/10.1016/j.future.2019.06.027>
- [84] **Chen, He, Song, Y., & Li, X.** (2019). A deep learning framework for identifying children with ADHD using an EEG-based brain network, *Neurocomputing*, 356, 83–96. <https://doi.org/10.1016/j.neucom.2019.04.058>
- [85] **Jiao, Y., Deng, Y., Luo, Y., & Lu, B.** (2020). Driver Sleepiness Detection from EEG and EOG signals Using GAN and LSTM Networks, *Neurocomputing*. <https://doi.org/10.1016/j.neucom.2019.05.108>
- [86] **Luo, Y., & Lu, B.-L.** (2018). EEG Data Augmentation for Emotion Recognition Using a Conditional Wasserstein GAN, *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, (pp.2535–2538). IEEE, July 18-21. <https://doi.org/10.1109/EMBC.2018.8512865>
- [87] **Fahimi, F., Zhang, Z., Goh, W. B., Ang, K. K., & Guan, C.** (2019). Towards EEG Generation Using GANs for BCI Applications, *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, (pp.1-4). IEEE, May 19-22. <https://doi.org/10.1109/BHI.2019.8834503>
- [88] **Hwang, S., Hong, K., Son, G., & Byun, H.** (2019). EZSL-GAN: EEG-based Zero-Shot Learning approach using a Generative Adversarial Network, *2019 7th International Winter Conference on Brain-Computer Interface (BCI)*, (pp.1-4). IEEE, February 18-20. <https://doi.org/10.1109/IWW-BCI.2019.8737322>



- [89] **Corley, I. A., & Huang, Y.** (2018). Deep EEG super-resolution: Upsampling EEG spatial resolution with Generative Adversarial Networks, *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, (pp.100–103). IEEE, March 4-7. <https://doi.org/10.1109/BHI.2018.8333379>
- [90] **Hartmann, K. G., Schirrmeister, R. T., & Ball, T.** (2018). EEG-GAN: Generative adversarial networks for electroencephalographic (EEG) brain signals, *ArXiv*. <http://arxiv.org/abs/1806.01875>
- [91] **How Brain Death Works.** (n.d.). Erişim: 13 Ocak 2019. <http://shorturl.at/fzAE0>
- [92] **Meziani, A., Djouani, K., Medkour, T., & Chibani, A.** (2019). A Lasso quantile periodogram based feature extraction for EEG-based motor imagery, *Journal of Neuroscience Methods*, 328, 108434. <https://doi.org/10.1016/j.jneumeth.2019.108434>
- [93] **Gao, J., & Li, L.** (2019). A robust geometric mean-based subspace discriminant analysis feature extraction approach for image set classification, *Optik*, 199. <https://doi.org/10.1016/j.ijleo.2019.163368>
- [94] **Luo, X., Li, X., Wang, Z., & Liang, J.** (2019). Discriminant autoencoder for feature extraction in fault diagnosis, *Chemometrics and Intelligent Laboratory Systems*, 192, 103814. <https://doi.org/10.1016/j.chemolab.2019.103814>
- [95] **Kingma, D. P., & Welling, M.** (2014). Auto-encoding variational bayes. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings, ML*. Banff, AB, Canada, April 14-16.
- [96] **Fu, X., Wei, Y., Xu, F., Wang, T., Lu, Y., Li, J., & Huang, J. Z.** (2019). Semi-supervised Aspect-level Sentiment Classification Model based on Variational Autoencoder, *Knowledge-Based Systems*, 171, 81–92. <https://doi.org/10.1016/j.knosys.2019.02.008>
- [97] **Wang, X., Du, Y., Lin, S., Cui, P., Shen, Y., & Yang, Y.** (2019). adVAE: A self-adversarial variational autoencoder with Gaussian anomaly prior knowledge for anomaly detection. *Knowledge-Based Systems*, 190, 105187. <https://doi.org/10.1016/j.knosys.2019.105187>
- [98] **Merry, R. J. E.** (2005). Wavelet theory and applications : a literature study, *Technische Universiteit Eindhoven, DCT rapporten, 2005.053*.
- [99] **Öztürk, H.** (2016). Eeg Sinyallerinde Farklı Boyut İndirgeme Ve Sınıflandırma Yöntemlerinin Karşılaştırılması (Yüksek Lisans Tezi). Adnan Menderes Üniversitesi, Sağlık Bilimleri Enstitüsü, Aydın.
- [100] **MindBigData The “MNIST” of Brain digits** (n.d.). Erişim: 15 Ağustos 2019. <http://www.mindbigdata.com/opendb/index.html>
- [101] **Acı, Ç. İ., & Çırak, A.** (2019). Türkçe Haber Metinlerinin Konvolüsyonel Sinir Ağları ve Word2Vec Kullanılarak Sınıflandırılması, *Bilişim Teknolojileri Dergisi*, 12(3), 219–228. <https://doi.org/10.17671/gazibtd.457917>.
- [102] **Singh, C., & Bala, A.** (2021). An unsupervised orthogonal rotation invariant moment based fuzzy C-means approach for the segmentation of brain magnetic resonance images, *Expert Systems with Applications*, 164(September 2020), 113989. <https://doi.org/10.1016/j.eswa.2020.113989>

- [103] **Despotović, I., Goossens, B., & Philips, W.** (2015). MRI segmentation of the human brain: Challenges, methods, and applications, *Computational and Mathematical Methods in Medicine*, 2015, Article ID 450341., <https://doi.org/10.1155/2015/450341>
- [104] **Marras, I., Nikolaidis, N., & Pitas, I.** (2014). 3D geometric split-merge segmentation of brain MRI datasets, *Computers in Biology and Medicine*, 48(1), 119–132. <https://doi.org/10.1016/j.compbiomed.2014.02.013>
- [105] **Bontempi, D., Benini, S., Signoroni, A., Svanera, M., & Muckli, L.** (2020). CEREBRUM: a fast and fully-volumetric Convolutional Encoder-decoder for weakly-supervised sEGmentation of BRain strUctures from out-of-the-scanner MRI, *Medical Image Analysis*, 62, 101688. <https://doi.org/10.1016/j.media.2020.101688>
- [106] **Kahali, S., Adhikari, S. K., & Sing, J. K.** (2017). A two-stage fuzzy multi-objective framework for segmentation of 3D MRI brain image data, *Applied Soft Computing Journal*, 60, 312–327. <https://doi.org/10.1016/j.asoc.2017.07.001>
- [107] **Aranguren, I., Valdivia, A., Morales-Castañeda, B., Oliva, D., Abd Elaziz, M., & Perez-Cisneros, M.** (2021). Improving the segmentation of magnetic resonance brain images using the LSHADE optimization algorithm, *Biomedical Signal Processing and Control*, 64, 102259. <https://doi.org/10.1016/j.bspc.2020.102259>
- [108] **Coupé, P., Mansencal, B., Clément, M., Giraud, R., Denis de Senneville, B., Ta, V. T., Lepetit, V., & Manjon, J. V.** (2020). AssemblyNet: A large ensemble of CNNs for 3D whole brain MRI segmentation, *NeuroImage*, 219, 117026. <https://doi.org/10.1016/j.neuroimage.2020.117026>
- [109] **Zhan, M., Goebel, R., & de Gelder, B.** (2018). Ventral and dorsal pathways relate differently to visual awareness of body postures under continuous flash suppression, *ENeuro*, 5(1), 1–18. <https://doi.org/10.1523/ENEURO.0285-17.2017>
- [110] **Dantzig, G., & Fulkerson, D. R.** (1956). On the Max-Flow MinCut Theorem of Networks, in *Linear Inequalities and Related Systems*, 38, 215–221.
- [111] **Jenkinson, M., Beckmann, C. F., Behrens, T. E. J., Woolrich, M. W., & Smith, S. M.** (2012). Fsl, *NeuroImage*, 62(2), 782–790. <https://doi.org/10.1016/j.neuroimage.2011.09.015>
- [112] **Ashburner, J., & Friston, K. J.** (2000). Voxel-based morphometry - The methods, *NeuroImage*, 11(6), 805–821. <https://doi.org/10.1006/nimg.2000.0582>
- [113] **Fischl, B.** (2012). FreeSurfer, *NeuroImage*, 62(2), 774–781. <https://doi.org/10.1016/j.neuroimage.2012.01.021>
- [114] **Manjón, J. V., & Coupé, P.** (2016). Volbrain: An online MRI brain volumetry system, *Frontiers in Neuroinformatics*, 10, 1–14. <https://doi.org/10.3389/fninf.2016.00030>
- [115] **Dorent, R., Booth, T., Li, W., Sudre, C. H., Kafiabadi, S., Cardoso, J., Ourselin, S., & Vercauteren, T.** (2021). Learning joint segmentation of tissues and brain lesions from task-specific hetero-modal domain-shifted datasets, *Medical Image Analysis*, 67, 101862. <https://doi.org/10.1016/j.media.2020.101862>

- [116] Cerri, S., Puonti, O., Meier, D. S., Wuerfel, J., Mühlau, M., Siebner, H. R., & Van Leemput, K. (2021). A contrast-adaptive method for simultaneous whole-brain and lesion segmentation in multiple sclerosis, *NeuroImage*, 225, 117471. <https://doi.org/10.1016/j.neuroimage.2020.117471>
- [117] Wang, Y., Zuo, C., Wang, W., Xu, Q., & Hao, L. (2021). Reduction in hippocampal volumes subsequent to heavy cannabis use: a 3-year longitudinal study, *Psychiatry Research*, 295, 113588. <https://doi.org/10.1016/j.psychres.2020.113588>
- [118] Xu, Q., Zuo, C., Liao, S., Long, Y., & Wang, Y. (2020). Abnormal development pattern of the amygdala and hippocampus from childhood to adulthood with autism, *Journal of Clinical Neuroscience*, 78, 327–332. <https://doi.org/10.1016/j.jocn.2020.03.049>
- [119] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016). 3D U-net: Learning dense volumetric segmentation from sparse annotation, *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9901 LNCS, 424–432. [https://doi.org/10.1007/978-3-319-46723-8\\_49](https://doi.org/10.1007/978-3-319-46723-8_49)
- [120] Rajchl, M., Pawlowski, N., Rueckert, D., Matthews, P. M., & Glocker, B. (2018). NeuroNet: Fast and robust reproduction of multiple brain image segmentation pipelines, *ArXiv, Midl*, 1–9. <https://arxiv.org/abs/1806.04224>
- [121] Guha Roy, A., Conjeti, S., Navab, N., & Wachinger, C. (2019). QuickNAT: A fully convolutional network for quick and accurate segmentation of neuroanatomy, *NeuroImage*, 186, 713–727. <https://doi.org/10.1016/j.neuroimage.2018.11.042>
- [122] Luna, M., & Park, S. H. (2019). 3D patchwise U-net with transition layers for MR brain segmentation, *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11383 LNCS 394–403. [https://doi.org/10.1007/978-3-030-11723-8\\_40](https://doi.org/10.1007/978-3-030-11723-8_40)
- [123] Cirillo, M. D., Abramian, D., & Eklund, A. (2020). Vox2Vox: 3D-GAN for Brain Tumour Segmentation, *ArXiv*, 1–10. <https://arxiv.org/abs/2003.13653>
- [124] Chen, Hao, Dou, Q., Yu, L., Qin, J., & Heng, P. A. (2018). VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images, *NeuroImage*, 170, 446–455. <https://doi.org/10.1016/j.neuroimage.2017.04.041>
- [125] Dolz, J., Gopinath, K., Yuan, J., Lombaert, H., Desrosiers, C., & Ben Ayed, I. (2019). HyperDense-Net: A Hyper-Densely Connected CNN for Multi-Modal Image Segmentation, *IEEE Transactions on Medical Imaging*, 38(5), 1116–1126. <https://doi.org/10.1109/TMI.2018.2878669>
- [126] Bitar, R., Leung, G., Perng, R., Tadros, S., Moody, A. R., Sarrazin, J., McGregor, C., Christakis, M., Symons, S., Nelson, A., & Roberts, T. P. (2006). MR pulse sequences: What every radiologist wants to know but is afraid to ask, *Radiographics*, 26(2), 513–537. <https://doi.org/10.1148/rg.262055063>
- [127] Smith, S. M. (2002). Fast robust automated brain extraction, *Human Brain Mapping*, 17(3), 143–155. <https://doi.org/10.1002/hbm.10062>

- [128] **Shattuck, D. W., & Leahy, R. M.** (2000). Brainsuite: An automated cortical surface identification tool, *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1935, 50–61. [https://doi.org/10.1007/978-3-540-40899-4\\_6](https://doi.org/10.1007/978-3-540-40899-4_6)
- [129] **Iglesias, J. E., Cheng-Yi Liu, Thompson, P. M., & Zhuowen Tu.** (2011). Robust Brain Extraction Across Datasets and Comparison With Publicly Available Methods, *IEEE Transactions on Medical Imaging*, 30(9), 1617–1634. <https://doi.org/10.1109/TMI.2011.2138152>
- [130] **Eskildsen, S. F., Coupé, P., Fonov, V., Manjón, J. V., Leung, K. K., Guizard, N., Wassef, S. N., Østergaard, L. R., & Collins, D. L.** (2012). BEaST: Brain extraction based on nonlocal segmentation technique, *NeuroImage*, 59(3), 2362–2373. <https://doi.org/10.1016/j.neuroimage.2011.09.012>
- [131] **Punjabi, A., Martersteck, A., Wang, Y., Parrish, T. B., & Katsaggelos, A. K.** (2019). Neuroimaging modality fusion in Alzheimer’s classification using convolutional neural networks, *PLoS ONE*, 14(12), 1–14. <https://doi.org/10.1371/journal.pone.0225759>
- [132] **Fonov, V., Evans, A. C., Botteron, K., Almli, C. R., McKinstry, R. C., & Collins, D. L.** (2011). Unbiased average age-appropriate atlases for pediatric studies, *NeuroImage*, 54(1), 313–327. <https://doi.org/10.1016/j.neuroimage.2010.07.033>
- [133] **Milletari, F., Navab, N., & Ahmadi, S. A.** (2016). V-Net: Fully convolutional neural networks for volumetric medical image segmentation, *Proceedings - 2016 4th International Conference on 3D Vision, 3DV 2016*, (pp.565–571). IEEE, October 25-28. <https://doi.org/10.1109/3DV.2016.79>
- [134] **Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L.** (2018). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4), 834–848. <https://doi.org/10.1109/TPAMI.2017.2699184>
- [135] **Zhou, Z., He, Z., & Jia, Y.** (2020). AFPNet: A 3D fully convolutional neural network with atrous-convolution feature pyramid for brain tumor segmentation via MRI images, *Neurocomputing*, 402, 235–244. <https://doi.org/10.1016/j.neucom.2020.03.097>
- [136] **Fu, J., Liu, J., Tian, H., Li, Y., Bao, Y., Fang, Z., & Lu, H.** (2019). Dual attention network for scene segmentation, *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, (pp.3141–3149).IEEE, June 15-20. <https://doi.org/10.1109/CVPR.2019.00326>
- [137] **Cárdenes, R., de Luis-García, R., & Bach-Cuadra, M.** (2009). A multidimensional segmentation evaluation for medical image data, *Computer Methods and Programs in Biomedicine*, 96(2), 108–124. <https://doi.org/10.1016/j.cmpb.2009.04.009>
- [138] **Taha, A. A., & Hanbury, A.** (2015). Metrics for evaluating 3D medical image segmentation: Analysis, selection, and tool, *BMC Medical Imaging*, 15(1). <https://doi.org/10.1186/s12880-015-0068-x>

## ÖZGEÇMİŞ

**Ad-Soyad** : Gaffari ÇELİK

### ÖĞRENİM DURUMU:

- **Lisans** : 2005, Mersin Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği
- **Yüksek Lisans** : 2013, Atatürk Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı

### MESLEKİ DENEYİM:

- 2019-Devam Ağrı İbrahim Çeçen Üniversitesi / Meslek Yüksekokulu / Bilgisayar Teknolojileri Bölümü / Bilgisayar Programcılığı
- 2006-2009 DMSSoft Yazılım Şirketi / Mersin

### TEZ SÜRECİNCE TÜRETİLEN BİLİMSEL YAYINLAR

#### SCI

1. **Çelik, G., & Talu, M.F. (2020)**. Resizing and cleaning of histopathological images using generative adversarial networks. *Physica A: Statistical Mechanics and its Applications*, 2020.
2. **Çelik, G., & Talu, M.F. (2021)**. A New 3D MRI Segmentation Method Based on Generative Adversarial Network and Atrous Convolution. *Biomedical Signal Processing and Control (Review aşamasında)*(Gönderimtarihi: 09/03/2021, Reviewtarihi: 29/05/2021).

#### TRDİZİN

1. **Çelik, G., & Talu, M.F. (2020)**. Çekişmeli üretken ağ modellerinin görüntü üretme performanslarının incelenmesi. *Balıkesir Üniversitesi Fen Bilim. Enstitüsü Dergisi*, 2020.
2. **Fırıldak, K., Çelik, G., & Talu, M.F. (2020)**. Cilt kanseri sınıflandırmasında klasik veri çoğallama ve DCGAN modeli ile veri artırmanın CNN performansına etkisi. *Fırat Üniversitesi Mühendislik Bilimleri Dergisi*, 2020.

3. **Çelik, G., & Talu, M.F. (2021).** EEG sinyallerinden bakılan görselin üretilmesi. Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 2021.

#### **ULUSAL DERGİ**

1. **Çelik, G., & Talu, M.F. (2018).** “Sentetik Büyük Veri“ İnşasında kullanılan desen yayma yaklaşımlarının incelenmesi. Anatolian Science - Bilgisayar Bilimleri Dergisi, 2018.

#### **ULUSLARARASI KONFERANS (Bildiriler)**

1. **Çelik, G., & Talu, M.F. (2017, Eylül).** Darbeli Yapay Sinir Ağları Uygulamaları. International Artificial Intelligence and Data Processing Symposium (IDAP 2017) (pp. 1-5). IEEE.
2. **Çelik, G., & Talu, M.F. (2018, May).** Synthetic Image Production For Large Data. International Conference on Advances in Natural and Applied Sciences (ICANAS 2018) (pp. 364).
3. **Yazan, E., Çelik, G., Talu, M.F., & Yeroğlu, C. (2018, Eylül).** Girdap Optimizasyon Algoritması Destekli Kumaş Hatası Tespiti. International Artificial Intelligence and Data Processing Symposium (IDAP 2018) (pp. 1-6). IEEE.