

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

**ÇEKİŞMELİ ÜRETİCİ AĞLAR İLE DENİM KUMAŞINDAN OTOMATİK  
BIYIK DESEN ÜRETİMİ**

**YÜKSEK LİSANS TEZİ**

**Emrullah ŞAHİN**

**Bilgisayar Mühendisliği Anabilim Dalı**

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

**MART 2022**

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

**ÇEKİŞMELİ ÜRETİCİ AĞLAR İLE DENİM KUMAŞINDAN OTOMATİK  
BIYIK DESEN ÜRETİMİ**

**YÜKSEK LİSANS TEZİ**

**Emrullah ŞAHİN  
(36193619026)**

**Bilgisayar Mühendisliği Anabilim Dalı**

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

**MART 2022**

## TEŐEKKÜR VE ÖNSÖZ

Bu tez alıőmasının her aőamasında yardım, öneri, bilgi, tecrübe ve desteklerini esirgemedi beni her konuda yönlendiren danışman hocam Sayın Prof. Dr. Muhammed Fatih TALU'ya,

alıőmalarında ayrıca tüm hayatım boyunca olduđu gibi bu alıőmalarım süresince benden her türlü desteklerini aileme,

Tezin uygulama aőamasında "FKP-2021-2144" nolu proje numarasıyla ÜSİ kapsamında vermiş oldukları maddi ve manevi destekten dolayı, Baykan Denim Konfeksiyon A.Ő. ve İnönü Üniversitesi BAP birimine

teőekkür ederim.



## ONUR SÖZÜ

Yüksek lisans tezi olarak sunduđum “Çekiřmeli Üretici Ağlar ile Denim Kumařından Otomatik Bıyık Desen Üretimi” bařlıklı bu çalıřmanın bilimsel ahlak ve geleneklere aykırı düřecek bir yardıma bařvurmaksızın tarafımdan yazıldıđına ve yararlandıđım bütün kaynakların hem metin içinde hem de kaynakçada yöntemine uygun biçimde gösterilenlerden oluřtuđunu belirtir, bunu onurumla dođrularım.

Emrullah řAHİN



## İÇİNDEKİLER

TEŞEKKÜR VE ÖNSÖZ .....	i
ONUR SÖZÜ .....	ii
İÇİNDEKİLER.....	iii
ÇİZELGELER DİZİNİ.....	v
ŞEKİLLER DİZİNİ.....	vi
SEMBOLLER VE KISALTMALAR .....	vii
ÖZET.....	ix
ABSTRACT .....	x
<b>1. GİRİŞ. .</b> .....	<b>1</b>
1.1 Bıyık Desen Üretimi .....	4
1.2 Tezin Önemi ve Hedefleri .....	5
1.3 Tezin Organizasyon Yapısı.....	5
<b>2. DENİM KUMAŞINDAN OTOMATİK BIYIK DESEN ÜRETİMİ</b> .....	<b>6</b>
2.1 Pix2Pix Mimarisi ile Bıyık Deseni Üretimi.....	6
2.2 Güncellenmiş Pix2Pix Mimarisi ile Bıyık Deseni Üretimi .....	8
2.3 Denim2Bıyık Veri Kümesinin Hazırlanması.....	9
2.4 Test Sonuçlarının Değerlendirilmesinde Kullanılan Benzerlik Metrikleri.....	9
2.4.1 SSIM.....	9
2.4.2 Kesinlik skoru.....	10
2.4.3 MAE .....	10
2.5 Uygulama Sonuçları .....	10
<b>3. BIYIK DESENİ ÜRETİMİNDE ÇÜA MİMARİLERİNİN PERFORMANS İNCELENMESİ</b> .....	<b>13</b>
3.1 CycleGAN Mimarisi.....	13
3.2 AttentionGAN Mimarisi .....	15
3.3 Denim2Bıyık Veri Kümesinin İyileştirilmesi.....	16
3.4 Test Sonuçlarının Değerlendirilmesinde Kullanılan Benzerlik Metrikleri.....	17
3.4.1 MSE.....	17
3.4.2 PSNR .....	18
3.5 Uygulama Sonuçları .....	18
<b>4. ALGISAL BENZERLİK MALİYETİ TABANLI ÇÜA MİMARİSİ İLE GÖRÜNTÜ ÜRETİMİ</b> .....	<b>21</b>
4.1 ESA Tabanlı Görüntü Karşılaştırma Ölçütleri .....	21
4.1.1 LPIPS.....	21
4.1.2 DISTS .....	22
4.1.3 CONTENT .....	22
4.1.4 Ölçütlerin $L_I$ ile kullanımı .....	23
4.2 Klasik Görüntü Karşılaştırma Ölçütleri .....	23
4.3 Güncel Görüntü Karşılaştırma Ölçütleri.....	24
4.4 Uygulama Sonuçları .....	24
<b>5. ÇÜA İLE GÖRÜNTÜ ÜRETİMİNDE YENİ BİR MİMARİNİN GELİŞTİRİLMESİ</b> .....	<b>27</b>
5.1 Ölçüm Metrikleri .....	27
5.1.1 Hinge gömülü maliyeti .....	27
5.2 DiscoGAN Mimarisi ile Görüntü Üretimi .....	28
5.2.1 Üretici ağ .....	28
5.2.2 Ayırıcı ağ.....	29
5.2.3 Maliyet fonksiyonu.....	29
5.2.3.1 Üretici ağ maliyeti .....	30

5.2.3.2 Ayırıcı ağ maliyeti .....	30
5.3 Y-Net: Önerilen Mimari .....	31
5.3.1 Üretici ağın iyileştirilmesi .....	31
5.3.2 Ayırıcı ağın iyileştirilmesi .....	32
5.3.3 Maliyet fonksiyonu .....	33
5.3.3.1 Üretici ağ maliyeti .....	33
5.3.3.2 Ayırıcı ağ maliyeti .....	34
5.4 Yeni Mimarinin Analizinde Kullanılan Veri Kümeleri .....	34
5.4.1 Cityscapes .....	34
5.4.2 Maps .....	35
5.5 Uygulama Sonuçları .....	35
<b>6. YÜKSEK ÇÖZÜNÜRLÜKTE BİYİK DESENİ ÜRETİMİ .....</b>	<b>41</b>
6.1 Pix2PixHD Mimarisi .....	41
6.1.1 Üretici ağ .....	41
6.1.2 Ayırıcı ağ .....	42
6.1.3 Maliyet fonksiyonu .....	43
6.1.3.1 Üretici ağ maliyeti .....	43
6.1.3.2 Ayırıcı ağ maliyeti .....	44
6.2 Denim2BıyıkHD: Yüksek Çözünürlüklü Denim2Bıyık Veri Kümesi .....	44
6.3 Uygulama Sonuçları .....	45
<b>7. SONUÇ VE ÖNERİLER .....</b>	<b>49</b>
<b>KAYNAKLAR .....</b>	<b>51</b>
<b>ÖZGEÇMİŞ .....</b>	<b>55</b>

## ÇİZELGELER DİZİNİ

Çizelge 2.1 : Orijinal ve güncellenmiş Pix2Pix mimarisinin başarıml sonuçları. ....	11
Çizelge 3.1 : Mimarilerin Denim2Bıyk test veri kümesindeki başarıml sonuçları. ....	19
Çizelge 4.1 : Mimarilerin görüntü sentezleme doğrulukları. ....	25
Çizelge 5.1 : Mimarilerin nicel özellik karşılaştırması. ....	36
Çizelge 5.2 : Mimarilerin performans karşılaştırması. ....	37
Çizelge 6.1 : Pix2PixHD mimarisinin başarıml sonuçları. ....	46



## ŞEKİLLER DİZİNİ

Şekil 1.1 : ÇÜA mimarisinin çalışma mekanizması..	1
Şekil 1.2 : Lazer kazıma sistemi.	4
Şekil 2.1 : Pix2Pix mimarisinin bıyık deseni çıkarımında kullanımı.....	7
Şekil 2.2 : Güncellenmiş Pix2Pix mimarisi.	8
Şekil 2.3 : Denim2Bıyık veri kümesinden örnek görüntüler.	9
Şekil 2.4 : Denim2Bıyık test veri kümesinden birkaç örneğin üretim sonuçları....	12
Şekil 3.1 : AttentionGAN mimarisinin çalışma mekanizması.....	15
Şekil 3.2 : Denim2bıyık veri kümesinin ön işleme şeması.	17
Şekil 3.3 : Mimarilerin Denim2Bıyık test kümesindeki görüntü üretim sonuçları....	20
Şekil 4.1 : LPIPS fonksiyonu.	22
Şekil 4.2 : DISTs fonksiyonu.	22
Şekil 4.3 : CONTENT fonksiyonu.....	23
Şekil 4.4 : Denim2Bıyık veri kümesi sentezleme çıktıları.....	26
Şekil 5.1 : DiscoGAN üretici ağların şeması.	28
Şekil 5.2 : DiscoGAN ayırıcı ağların şeması.	29
Şekil 5.3 : Y-Net mimarisinin üretici ağ şeması.	32
Şekil 5.4 : Y-Net mimarisinin ayırıcı ağ şeması.	33
Şekil 5.5 : Cityscapes veri kümesinden örnek görüntüler.	35
Şekil 5.6 : Maps veri kümesinden örnek görüntüler..	35
Şekil 5.7 : Cityscapes veri kümesinden örnek test sonuçları.	38
Şekil 5.8 : Maps veri kümesinden örnek test sonuçları.	39
Şekil 5.9 : Denim2Bıyık veri kümesinden örnek test sonuçları.....	40
Şekil 6.1 : Pix2PixHD mimarisinin üretici ağ şeması.....	42
Şekil 6.2 : Pix2PixHD mimarisinin ayırıcı ağ şeması.....	42
Şekil 6.3 : Denim2BıyıkHD veri kümesinden örnek görüntüler.	45
Şekil 6.4 : Denim2BıyıkHD veri kümesinin eğitim setindeki üretim sonuçları.	47
Şekil 6.5 : Denim2BıyıkHD veri kümesinin test setindeki üretim sonuçları.....	48



## SEMBOLLER VE KISALTMALAR

<b>ÇÜA</b>	: Çekişmeli üretici ağlar
<b>ESA</b>	: Evrişimsel sinir ağları
<b>G</b>	: Üretici ağ
<b>D</b>	: Ayırıcı ağ
<b>G<sub>X2Y</sub></b>	: X alanından Y alanına dönüşüm yapan üretici ağ
<b>G<sub>Y2X</sub></b>	: Y alanından X alanına dönüşüm yapan üretici ağ
<b>D<sub>X</sub></b>	: X alanındaki görüntülerin özelliklerini öğrenmeye çalışan ayırıcı ağ
<b>D<sub>Y</sub></b>	: Y alanındaki görüntülerin özelliklerini öğrenmeye çalışan ayırıcı ağ
<b>A<sub>X</sub></b>	: X alanındaki görüntülerin dikkat ağı
<b>A<sub>Y</sub></b>	: Y alanındaki görüntülerin dikkat ağı
<b>Conv2d</b>	: İki boyutlu evrişim işlemi
<b>Decode</b>	: Kod çözücü
<b>Encode</b>	: Kodlayıcı
<b>Deconv</b>	: Ters evrişim işlemi
<b><math>\mathcal{L}_{cGAN}</math></b>	: Koşullu çekişmeli üretici ağların temel maliyet fonksiyonu
<b><math>L_1</math></b>	: Ortalama mutlak hata ile uzaklık ölçme metriği
<b><math>L_2</math></b>	: Ortalama karesel hata ile uzaklık ölçme metriği
<b>MAE</b>	: Ortalama mutlak hata
<b><math>\mathcal{L}_{similarity_x}</math></b>	: X görüntü alanı için benzerlik maliyeti
<b>SSIM</b>	: Yapısal benzerlik metriği
<b>FID</b>	: Fréchet başlangıç mesafesi
<b>KID</b>	: Kernel başlangıç mesafesi
<b>FSIM</b>	: Özellik tabanlı benzerlik indeksi
<b>HaarPSI</b>	: Haar dalgacık tabanlı algısal benzerlik indeksi
<b>MS-SSIM</b>	: Çok ölçekli yapısal benzerlik metriği
<b>PSNR</b>	: Pik Sinyal Gürültü Oranı
<b>VIF</b>	: Görsel bilgi doğruluğu
<b>VSI</b>	: Görsel belirginliğe bağlı indeks
<b>DISTS</b>	: Derin görüntü yapısı ve doku benzerliği
<b>LPIPS</b>	: Öğrenilmiş algısal görüntü yama benzerliği
<b>CONTENT</b>	: İçerik ağı
<b>Acc</b>	: Kesinlik skoru
<b>MSE</b>	: Ortalama karesel hata
<b><math>\mathcal{L}_{cyc}</math></b>	: Döğü tutarlılık kaybı

$\mathcal{L}_{adv}$	: Tartışmalı kayıp
$\mathcal{L}_{identity}$	: Özdeşlik kaybı
$\mathcal{L}_{feature}$	: Özellik kaybı
$\mathcal{L}_{FM}$	: Özellik eşleştirme kaybı
$\mathcal{L}_{perceptual}$	: Algısal kayıp
$\mathcal{L}_{const}$	: Yeniden oluşturma kaybı
$L_{HE}$	: Hinge gömülü maliyeti
$Real\_X, x$	: X alanındaki gerçek görüntü
$Real\_Y, y$	: Y alanındaki gerçek görüntü
$Fake\_X, \hat{x}$	: Y alanındaki gerçek görüntünün X alanına dönüşmüş imajı
$Fake\_Y, \hat{y}$	: X alanındaki gerçek görüntünün Y alanına dönüşmüş imajı
$Recons\_X, \tilde{x}$	: X alanındaki gerçek görüntünün yeniden oluşturulmuş imajı
$Recons\_Y, \tilde{y}$	: Y alanındaki gerçek görüntünün yeniden oluşturulmuş imajı
$z$	: Rastgele gürültü
$x_a$	: X alanındaki görüntünün dikkat haritası
$x_f$	: X alanındaki görüntünün önplan görüntüsü
$x_b$	: X alanındaki görüntünün arkaplan görüntüsü
$\mu$	: Ortalama
$\sigma$	: Standart sapma
$L$	: Piksel aralığı
$n$	: Toplam piksel sayısı
$w$	: Bir görüntünün vektörel formu

## ÖZET

Yüksek Lisans Tezi

### ÇEKİŞMELİ ÜRETİCİ AĞLAR İLE DENİM KUMAŞINDAN OTOMATİK BİYİK DESEN ÜRETİMİ

Emrullah ŞAHİN

İnönü Üniversitesi  
Fen Bilimleri Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalı

55+X sayfa

2022

Danışman: Prof. Dr. Muhammed Fatih TALU

Bu tez çalışması, denim kumaşlarında bulunan “bıyık” desenlerinin otomatik olarak algılanması ve üretilmesi hakkındaki güncel yaklaşımları ve önerilen yeni mimarileri kapsamaktadır. Tezin odağındaki problem denim görsellerindeki bıyık desenlerinin otomatik ve kaliteli üretilmesidir. Bu çalışma Malatya 2. Organize Sanayi Bölgesinde faaliyet gösteren Baykan Denim firmasıyla yapılan görüşmeler neticesinde belirlenmiştir. Firma müşterileri, örnek bir bıyık desenli denim kumaşını getirerek benzer bıyık desenli kumaşlar talep etmektedir. Klasik bıyık desen üretimi firmada bulunan Photoshop uzmanlarının örnek nümune üzerinde yaklaşık 2-3 saatlik çalışmasıyla yapılmaktadır. Yüksek personel maliyeti, işlemin uzun sürmesi ve özneliği (uzmanların aynı görselden farklı bıyık deseni üretebilmesi) sahada karşılaşılan olumsuzluklardır. Bahsedilen olumsuzlukların önüne geçmek ve denim-bıyık dönüşüm problemine uygun çözümler üretebilmek için yüksek çözünürlükte denim-bıyık görsellerinden oluşan yeni bir veri kümesi inşa edilmiştir. Bu veri kümesi ile üretici ağ tabanlı çeviri sağlayan mimariler karşılaştırmalı bir şekilde incelenmiştir. İncelenen mimarilerden ilham alınarak ve dezavantajları göz önünde bulundurularak literatüre özgün yaklaşımlar kazandırılmıştır.

**Anahtar Kelimeler:** Denim Kumaşı, Bıyık Deseni, Çekişmeli Üretici Ağlar

# ABSTRACT

Master Thesis

## AUTOMATIC MUSTACHE PATTERN PRODUCTION ON DENIM FABRIC WITH GENERATIVE ADVERSARIAL NETWORKS

Emrullah ŞAHİN

Inonu University  
Graduate School of Nature and Applied Sciences  
Department of Computer Engineering

55+X sayfa

2022

Supervisor: Prof. Dr. Muhammed Fatih TALU

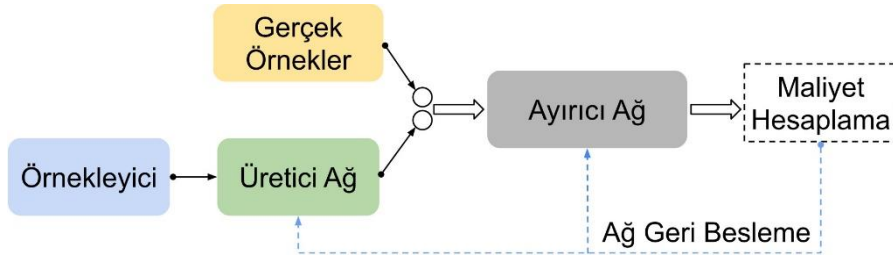
This thesis covers current approaches and proposed new architectures on the automatic detection and production of "mustache" patterns in denim fabrics. The problem in the focus of the thesis is the automatic and high quality production of mustache patterns in denim images. This study was determined as a result of interviews with Baykan Denim company operating in Malatya 2nd Organized Industrial Zone. Company customers demand similar mustache-patterned fabrics by bringing a sample mustache-pattern denim fabric. Classic mustache pattern production is done by Photoshop experts working on the sample for approximately 2-3 hours. High personnel cost, long duration of the process and subjectivity (experts can produce different mustache patterns from the same image) are the disadvantages encountered in the field. A new dataset consisting of high-resolution denim-mustache images has been constructed in order to avoid the aforementioned negativities and to produce suitable solutions for the denim-mustache translation problem. With this dataset, architectures that provide generative network-based translation are comparatively examined. Inspired by the examined architectures and considering their disadvantages, original approaches have been brought to the literature.

**Keywords:** Denim Fabric, Mustache Pattern, Generative Adversarial Networks

## 1. GİRİŞ

Derin öğrenme, bir veriyi en iyi ifade eden özniteliklerin otonom öğrenilebildiği, yapay öğrenmenin bir alt alanıdır. Bu alanın kullanım alanları genişlemekte ve bununla birlikte dallandığı alt alanlarda çoğalmaktadır. Son zamanlarda derin öğrenme tabanlı uygulamalar gittikçe artmaktadır. Örneğin dil çevirisi, sohbet botları, yüz tanımlama, sesli imza, hastalık teşhisi, veri artırma, otonom araçlar ve anomali analizi gibi birçok alanda kullanılmaktadır. Bu alan birçok öğrenme yöntemi barındırmasına rağmen günümüzde en popüler olanlardan biriside 2014 yılında Goodfellow ve ekibi tarafından geliştirilen min-max algoritması tabanlı Çekişmeli üretici ağlardır (ÇÜA) [1].

ÇÜA bir veri kümesinde bulunan örnekler arasındaki ilişkiyi modelleyebilmek için geliştirilmiş özel bir sinir ağı türüdür. Ağ örnekler arasındaki ilişkiyi en iyi ifade eden öznitelikleri öğrenmeye çalışır. Bu ağların temel yapısı, ünlü matematikçi John Nash'in oyun teorisini baz alır ve birbiriyle çekişmeli bir şekilde çalışarak öğrenen, iki sinirsel modülden oluşur [1-3]. Bu modüllerden biri üretici (generator) ağı, diğeri ise ayırıcı (discriminator) ağı olarak adlandırılır. Üretici ağı, kendisine verilen rastgele ya da özel bir veri bloğunu (örnekleyici) istenilen veri örneğine benzetmeye çalışır. Ayırıcı ağı, üretici ağın sentezlediği yapay örnek ile gerçek veri örneğini birbirinden ayırt etmeye çalışan bir sınıflandırıcıdır [2,3]. Eğitim aşamasında üretici ve ayırıcı ağın parametreleri gerçek ve yapay çıktılarının benzerlik maliyeti (loss) ile beslenir. Bu modellerin çalışma mekanizması **Şekil 1.1**'de gösterilmektedir.



**Şekil 1.1** : ÇÜA mimarisinin çalışma mekanizması.

ÇÜA mimarileri; sentetik görüntü üretimi, görüntü manipülasyonu, video dönüşümleri, veri artırma ve nesne algılama gibi birçok çalışmada kullanılmaktadır. Günümüzde çeşitli ÇÜA mimarileri mevcuttur.

Bir veri kümesindeki örneklere ait özniteliklerin sayısal dağılımını öğrenerek benzer görüntü sentezleyebilen DCGAN mimarisi, 2015 yılında Radford ve diğerleri tarafından geliştirilmiştir [4].

Düşük çözünürlüklü görüntüleri yüksek çözünürlükte sentezleyebilmek için algısal benzerlik maliyeti tabanlı SRGAN mimarisi, 2016 yılında Ledig ve diğerleri tarafından geliştirilmiştir [5].

Denetimli veri kümelerindeki eşleşmiş görüntü çiftleri arasında dönüşüm işlemi yapan Pix2Pix mimarisi, 2017 yılında Isola ve ekibi tarafından geliştirilmiştir [6]. Mimari geleneksel ÇÜA maliyetine ek olarak  $L_1$  metriği içermektedir.

Denetimsiz veri kümelerinde alanlar arası görüntüden görüntüye dönüşüm işlemi yapabilmek için çift taraflı doğrulama işlemi yaparak öğrenme gerçekleştirebilen CycleGAN mimarisi, 2017 yılında Zhu ve diğerleri tarafından geliştirilmiştir [7].

Yüksek çözünürlükte üretilen görüntülerdeki bozulmaları azaltmak için aşamalı öğrenim gerçekleştiren Progressive GAN mimarisi, 2017 yılında Karras ve diğerleri tarafından geliştirilmiştir [8]. Mimari eğitim aşamasında 4x4'lük görüntü ölçeğiyle öğrenim gerçekleştirdikten sonra adım adım ölçeği artırıp 1024x1024'lük görüntü sentezleyebilene kadar devam etmektedir.

Bir içerik görüntüsüne herhangi bir görüntüdeki stili aktarmak için içerik ve stil görüntüsü ile beslenebilen StyleGAN mimarisi, 2018 yılında Karras ve diğerleri tarafından geliştirilmiştir [9].

ÇÜA mimarilerindeki üretici ve ayırıcı ağıın eğitim aşamasında dengesiz öğrenme gerçekleştirmesinden dolayı öğrenme eğrisindeki istikrarsızlığı azaltmak için konumsal dikkat ve spektral normalizasyon tabanlı SAGAN mimarisi, 2018 yılında Zhang ve diğerleri tarafından geliştirilmiştir [10].

Yüksek çözünürlükte sentetik görüntü üretimi için ImageNet veri kümesi üzerinde sınıfsal olarak üretim yapabilen BigGAN mimarisi, 2018 yılında Brock ve diğerleri tarafından geliştirilmiştir [11]. Bu mimari kaliteli ve çeşitli görüntü üretimi için ikili filtreler arasındaki normları kısıtlamadan kosinüs benzerliğini en aza indirgemeye çalışır. Filtrelerin parametrelerine eklenen bu düzenleme terimi farklı sınıflara ait yüksek kalitede görüntü üretilmesini sağlar.

Yüksek çözünürlükteki anlamsal görüntü haritalarından gerçekçi görüntüler üretebilen Pix2PixHD mimarisi, 2018 yılında Wang ve diğerleri tarafından geliştirilmiştir [12].

Denetimsiz anlamsal görüntü haritalarından gerçekçi doğa fotoğrafları üretebilmek için mekansal uyarlanabilir normalleştirme özelliğine sahip SPADE mimarisi, 2019 yılında Nvidia'da araştırmacılar tarafından geliştirilmiştir [13].

StyleGAN mimarisindeki adaptif örnek normalleştirme ve ağ kaynaklı görüntü üzerindeki bozulmaları minimize etmek için modülasyon ve ters modülasyon tabanlı StyleGAN2 mimarisi, 2019 yılında Karras ve diğerleri tarafından geliştirilmiştir [14].

Kaliteli ve gerçekçi görüntüler üretebilen BigBiGAN mimarisi, 2019 yılında Donahue ve diğerleri tarafından geliştirilmiştir [15]. Bu mimari gerçek görüntüleri rastgele bir değişkene dönüştürebilen bir kodlayıcı ve ikili maliyet fonksiyonuyla çalışan bir ayırıcı ağa sahiptir. Sistem daha kaliteli öğrenme için kodlayıcı ağdan gelen özellikler ile gizli alandan gelen değerleri aynı temsile getirmeye çalışır.

Çeşitli içerik üretimi ve derinlemesine görüntü değişimi için MixNMatch mimarisi, 2020 yılında Li ve diğerleri tarafında geliştirilmiştir [16]. MixNMatch mimarisi; içerik, şekil, poz ve arkaplan görüntülerini alıp bunları birleştiren bir yapıya sahiptir.

Anlamsal harita tabanlı görüntü sentezlemedeki kaliteyi artırmak için panoptik harita bazlı görüntü üretimi yapan PIS mimarisi, 2020 yılında Dundar ve diğerleri tarafından geliştirilmiştir [17].

İnsan yüzüne ait eski veya kalitesiz görsellerin yüksek kalitede gürültüsüz ve renkli olarak sentetik bir şekilde üretilebilmesi için uzamsal özellik dönüşümlü GFP-GAN mimarisi, 2021 yılında Wang ve diğerleri tarafından geliştirilmiştir [18].

StyleGAN mimarilerindeki öteleme ve döndürmeden kaynaklı gizli alandan gelen sinyal probleminin çözümü için bu mimariye Fourier tabanlı StyleGAN3 mimarisi, 2021 yılında Karras ve diğerleri tarafından geliştirilmiştir [19].

Bir görüntüdeki düşük çözünürlük, bulanıklık, sıkıştırma ve gürültü gibi efektleri azaltıp, kaliteli görüntü üretimi için aynı anda birden fazla problem alanına odaklanabilen Real-ESRGAN mimarisi, 2021 yılında Wang ve diğerleri tarafından geliştirilmiştir [20].

Görüntüler üzerindeki istenmeyen nesnelere veya bölgelerin görüntünün ana yapısını bozmadan kaldırılarak sentezlenebilmesi için Fourier konvolüsyon tabanlı LaMa mimarisi, 2021 yılında Suvorov ve diğerleri tarafından geliştirilmiştir [21].

Görüntü dönüşümü alanındaki bu zorlukların temel görüntü işleme algoritmalarıyla gerçekleştirilmesi güçtür. Yüksek dereceli bu problemlerin çözümünde ÇÜA mimarilerinin yaygın bir şekilde kullanıldığı bilinmektedir. Bu tezde denim kumaşı görselinin otomatik büyük desen motifine dönüştürebilmek için ÇÜA mimarileri kullanılmıştır.

## 1.1 Bıyık Desen Üretimi

Denim kumaşlarının hayatın her alanında rahatlıkla kullanılabilir olması talebin gün geçtikçe artmasına neden olmakta ve kumaş üreticilerini artan talebi karşılama noktasında yeni aksiyonlar almaya itmektedir. Teknolojinin gelişmesine bağlı olarak denim kumaş üretiminde personel giderleri ve üretim maliyetlerin düşürülmesi, yüksek konfor ve kaliteye sahip ürünlerin hızlı bir şekilde üretilmesi gibi konular işletmeler için gün geçtikçe artan bir öneme sahip olmaktadır [22-24].

Moda dünyasındaki gelişmelerin hazır giyim sektörüne yansımalarından biride denim kumaşları üzerine çizilen bıyık desenleridir. Bıyık desenleri, kumaşın sade görünümüne farklı bir tarz kazandırmaktadır. Üretilen sade denim kumaşlarının üzerine bıyık desenlerini çizebilmek için **Şekil 1.2**'nin sol sütununda gösterilen lazer kazıma makineleri kullanılmaktadır. Bu makineler, yaklaşık 2m<sup>2</sup>'lik bir alan içerisine koyulan denim kumaşı yüzeyi üzerine lazer ışığı ( $\lambda = 10640\text{nm}$ ) göndererek 0.01mm gibi oldukça hassas büyüklüklerde yakma işlemi gerçekleştirebilmektedir.



**Şekil 1.2** : Lazer kazıma sistemi: (sol) Makine, (sağ) Yazılım [25].

Yakma prosesindeki konfigürasyon parametreleri **Şekil 1.2**'nin sağ sütununda gösterilen yazılım ile gerçekleştirilmektedir. Bu yazılım, kumaş üzerine çizilecek bıyık desenine ait görseli giriş olarak almakta ve belirlenen konfigürasyonda kumaş üzerine aktarabilmektedir.

Lazer kazıma makinesine girdi olarak verilecek bıyık desen görüntüleri için ya yeni bir tasarım yapılır veya hazır bir numuneden (bıyıklı denim kumaşı) alınan görüntüden bıyık



efekti çıkarılmaktadır. Bu nedenle işletmede Photoshop gibi görsel düzenleme programlarında uzmanlaşmış personellere ihtiyaç duyulmaktadır. Uzmanların bir bıyık desen görselini inşa edebilmesi için yaklaşık 2-3 saat süren bir çalışmaya ihtiyaç duyulur. Yeni bıyık tasarımlarının yapılması, mevcut bıyık modelinin taklit edilmesine nispeten kolay bir işlemdir. Çünkü sade bir denim kumaş görseli üzerinde istenilen efektler rahatlıkla oluşturulabilir. Ancak müşteriden temin edilen numuneler fotoğraflandıktan sonra elde edilen gerçek görseller üzerinde çalışma yapmak ve bıyık efektlerinin çıkarılması işlemi, birçok görüntü işleme operasyonunun uygulanmasını gerektirdiğinden dolayı oldukça karmaşık bir işlemdir. Ayrıca uzmanın tecrübesine göre elde edilen çıktı (bıyık desen görseli) değişim göstermektedir. Üretilen bıyık desen görselindeki küçük bir hata veya istenmeyen bir efekt ürüne yansıtacağından dolayı oldukça hassas bir işlem olduğu açıktır.

## **1.2 Tezin Önemi ve Hedefleri**

Bu tez çalışmasının amacı, denim kumaşından otomatik yüksek kaliteli bıyık desen üretimini sağlamaktır. Böylece uzman personellere ihtiyaç duyulmadan bıyık desenlerinin hızlı ve nesnel olarak üretilebilmesi, bu alandaki firmalara yüksek performans sağlayacaktır. Kaliteli bıyık deseni üretimi için ÇÜA mimarileri kullanılmıştır. İlk aşamada görüntü sentezleme alanındaki mevcut ÇÜA yaklaşımları ile bıyık deseni üretimi sağlanmakta, daha sonra performans artışı sağlayan özgün mimariler geliştirilerek literatüre katkılar sunulmaktadır.

## **1.3 Tezin Organizasyon Yapısı**

Bölüm 2’de Denim2Bıyık veri kümesinin hazırlanması ve ÇÜA tabanlı olarak otomatik bıyık desen üretiminin gerçekleşmesini içermektedir. Bölüm 3’te otomatik bıyık deseni üretiminde ÇÜA mimarilerinin performans incelemesini içermektedir. Bölüm 4’te ÇÜA mimarisine ESA tabanlı maliyet fonksiyonları dahil edilerek, görüntü üretimindeki performans karşılaştırmasını içermektedir. Bölüm 5’te kaliteli ve hızlı görüntü üretimi için geliştirilen ÇÜA tabanlı yeni bir mimarinin detayları ve performans sonuçları anlatılmaktadır. Bölüm 6’da yüksek çözünürlüklü Denim2bıyık veri kümesinin hazırlanmasını ve üretilmesini içermektedir. Bölüm 7’de tezde yapılan çalışmalara ait sonuçlarının birleştirilerek özetlenmesini içermektedir.

## 2. DENİM KUMAŞINDAN OTOMATİK BIYIK DESEN ÜRETİMİ

Denim kumaşları üzerine çizilen bıyık desenleri lazer ışın cihazıyla oluşturulmaktadır. Bu cihazın istenilen bıyık desenini çizebilmesi için desen görselinin hazırlanması gerekir. Müşteriden alınan numune kotlardaki bıyık desenlerinin görsel aktarılabilmesi için Photoshop programında uzmanlaşmış bir personelin ortalama 2-3 saat süren bir çalışma yapması gerekir. Bu durum üretim hızının yavaşlamasına ve insana bağlı hataların ortaya çıkmasına neden olmaktadır. Tezin bu bölümünde, müşteriden alınacak örnek kot numunelerindeki bıyık desenlerini otomatik algılayarak desen görselini üreten yeni bir yaklaşım önerilmektedir. Bu yaklaşımda, bıyık desen görüntülerinin üretilebilmesi için çekişmeli üretici ağlar içerisinde yer alan Pix2Pix mimarisinin güncellenmiş bir versiyonu kullanılmaktadır.

### 2.1 Pix2Pix Mimarisi ile Bıyık Deseni Üretimi

Pix2Pix, görüntüden görüntüye dönüşüm yapan koşullu ÇÜA mimarilerinin özel bir versiyonudur. Bu mimari DCGAN örnek alınarak geliştirilmiştir. Mimari bir üretici ( $G$ ) ve bir ayırıcı ( $D$ ) ağ bulundurur. Üretici ağda U-Net [26] mimarisi, ayırıcı ağda PatchGAN [6] kullanılır. Pix2Pix ile üretilen görsellerde bulanıklık seviyesinin yüksek olduğu gözlemlenmiştir. Isola ve diğerleri [6], bulanıklığı bir derece giderebilmek için üretici mimarinin maliyet fonksiyonuna ek bir  $L_1$  benzerlik terimi eklemiştir.

Geleneksel ÇÜA mimarilerindeki maliyet değeri Denklem 2.1 gibi ifade edilmektedir. Denklemdaki  $x$  girdisi  $y$  çıktısına bağlıdır. Ayırıcı ağ gerçek  $x$  ve  $y$  girdilerin öznitelik çıktısı ile gerçek  $x$  ve  $z$  özniteliğinin üretici ağa verilerek elde edilen  $G(x, z)$  çıktısı arasındaki maliyeti hesaplar.

$$\mathcal{L}_{CGAN}(G, D) = \mathbb{E}_{x,y}[\log D(x, y)] + \mathbb{E}_{x,z}[\log D(x, G(x, z))] \quad (2.1)$$

Buna göre üretici  $G$  ağı bu amaç fonksiyonunu minimize etmek isterken, ayırıcı  $D$  ağı maksimize etmek ister. Böylece üreticinin optimal maliyeti Denklem 2.2 gibi tanımlanır:

$$\arg \min_G \max_D \mathcal{L}_{CGAN}(G, D) \quad (2.2)$$

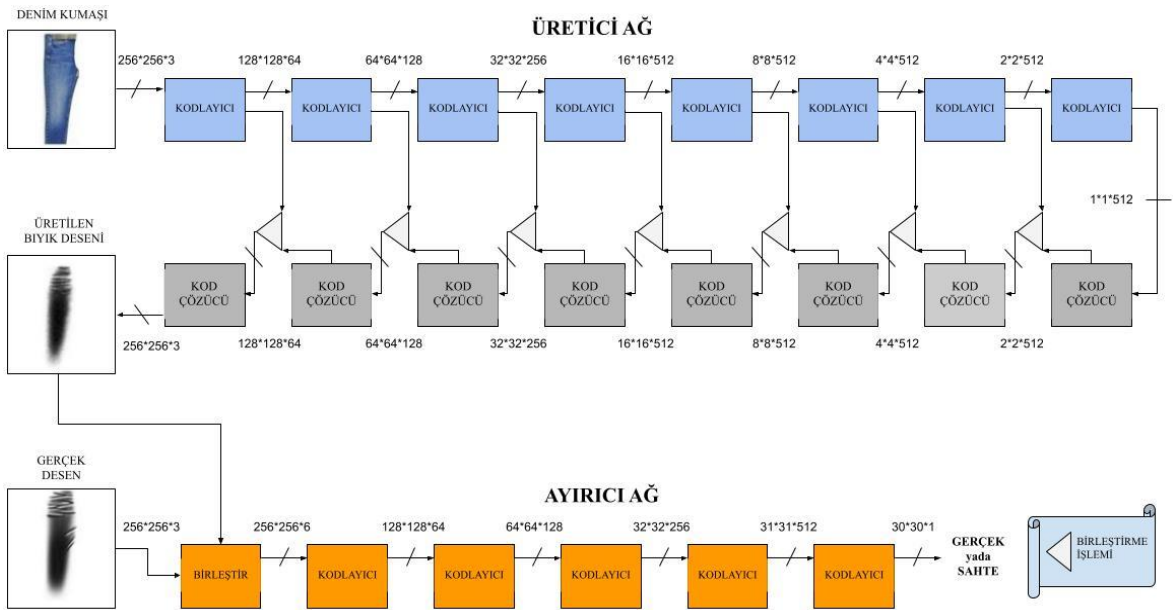
Denklem 2.1’de ifade edilen maliyet fonksiyonuna birde gerçek görüntü ( $x$ ) ile üretilen sahte görüntü ( $\hat{x}$ ) arasındaki benzerlik maliyeti eklenerek Denklem 2.3 gibi ifade edilir. Bu maliyet değeri  $L_1$  uzaklık metriği olarak adlandırılan ortalama mutlak hatadır (MAE [27]).

$$\mathcal{L}_{similarity_x} = L_1(x, \hat{x}) \quad (2.3)$$

$L_1$  değerinin eklenmesi sonucunda Pix2Pix mimarisinin maliyet fonksiyonu Denklem 2.4 gibi tanımlanır. Denklem klasik ÇÜA maliyet sonucu ve  $L_1$  değerinin  $\lambda = 100$  değeri ile çarpılmasıyla hesaplanır.

$$\mathcal{L}_{G,D} = \arg \min_G \max_D \mathcal{L}_{CGAN}(G,D) + \lambda * \mathcal{L}_{similarity_x} \quad (2.4)$$

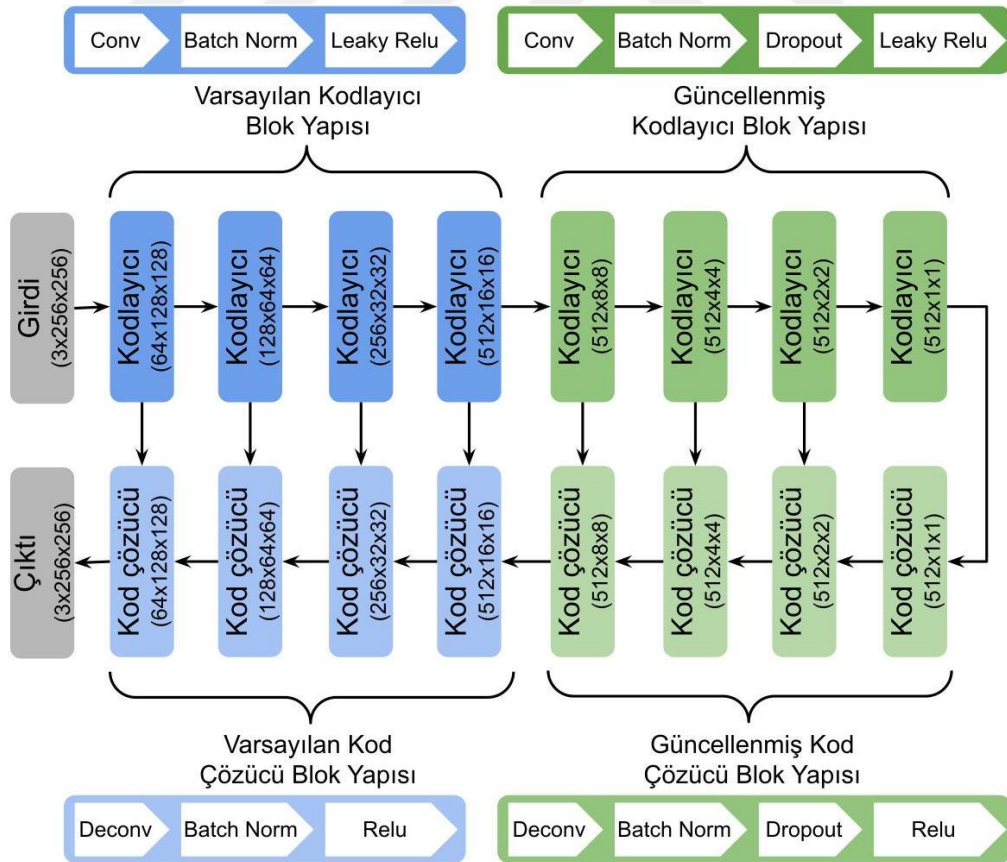
Pix2Pix üretici ağı  $256 \times 256 \times 3$  ebatlarında renkli görüntü almaktadır. Giriş görüntüleri kodlayıcı (Encode) bloklardan geçerek  $1 \times 1 \times 512$  ebatlarındaki öznitelik haritasına dönüşür. Kodlayıcı bloklar evrişim , Leaky Relu ve toplu normalleştirme (Batch Norm) katmanlarını içermektedir. Elde edilen öznitelik haritası ve her kodlayıcı bloğun çıktısı kendi karşısındaki kod çözücü (Decode) blokların çıktısı ile birleştirilerek üst kod çözücü bloğa giriş olarak verilir. Bu işlemle dönüşmüş görüntü elde edilir. Kod çözücü bloklar kodlayıcı bloklardan farklı olarak aktivasyon fonksiyonunda Relu kullanmaktadır. Ayırıcı ağ gerçek ve üretilen görüntüleri kodlayıcı bloklarda işleyerek  $30 \times 30 \times 1$  ebatlarına dönüştürür. Elde edilen bu öznitelik haritası gerçek veya sahte olarak sınıflandırılmaya çalışır. Pix2Pix mimarisi ile denim kumaşından büyük desen üretim işleminin şeması **Şekil 2.1**’de gösterilmiştir.



**Şekil 2.1** : Pix2Pix mimarisinin büyük deseni çıkarımında kullanımı.

## 2.2 Güncellenmiş Pix2Pix Mimarisi ile Bıyık Deseni Üretimi

Çekişmeli üretici ağlar, eğitim aşamasında bir görüntü alanından diğer görüntü alanına dönüşüm yaparken eğitim veri kümesindeki görüntülerin bilgilerini ezberleyebilir. Bu problem literatürde aşırı öğrenme olarak ifade edilmektedir. Eğitim aşamasında aşırı öğrenmeden dolayı test aşamasında istenilen kalitede görüntü üretimi sağlanamaz. Bu problemi engellemek için ağı bazı katmanlarına veya maliyet fonksiyonuna düzenleme işlemi uygulanır. Görüntü üretiminde bu problemi engellemek için Pix2Pix mimarisindeki üretici ağın temel yapısı üzerinde değişiklik yapılmıştır. Varsayılan olarak üretici ağın kodlayıcı bölümünde evrişim (Conv), toplu normalleştirme (Batch Norm) ve Leaky Relu katmanları varken kod çözücüde ters evrişim (Deconv), toplu normalleştirme ve Relu katmanları vardır. Güncellenmiş sürümde kodlayıcı bölümün son 4 bloğuna ve kod çözücü bölümün ilk 4 bloğuna Dropout katmanı ilave edilerek ağı aşırı öğrenmeye karşı daha dayanıklı olması sağlanmıştır. Böylece test aşamasında da kaliteli ve gürültüsüz görüntü üretmesi sağlanmıştır. Önerilen metot Pix2Pix v2 olarak adlandırılmıştır. Bu mimarinin şeması **Şekil 2.2**'de gösterilmiştir.



**Şekil 2.2 :** Güncellenmiş Pix2Pix mimarisi.

### 2.3 Denim2bıyık Veri Kümesinin Hazırlanması

Bu çalışma, Baykan Denim A.Ş.'nin [28] Malatya'daki üretim fabrikasında gerçekleşmiştir. Veri kümesinin inşasında, firmanın şimdiye kadar elde ettiği denim kumaş numune görselleri ve bunlara bakılarak manuel çizilen bıyık görüntüleri kullanılmıştır. Toplamda 976 adet denim-bıyık görüntü ikilisi elde edilmiştir. Veri kümesi, pantolon, etek ve şort olmak üzere üç farklı denim kumaş türünün ön ve arka yüz ve paça gibi farklı bölgelerine ait görseller içermektedir. Firma çalışanlarının cep telefonlarıyla elde edilen bu görüntülerinin standart bir forma sahip olmadığı görülmüştür. Daha açık bir ifadeyle, kumaş görsellerinin arkaplan görünümü, ışık dağılımları, boyutları ve perspektiflerinin farklı olduğu görülmüştür. Verinin standart bir forma dönüştürülebilmesi için her bir denim kumaş görseli üzerinde manuel bir temizleme çalışması yapılmıştır. Bu çalışmada ön plan nesnesi (kumaş görüntüsü) arka plandan ayıklanmış ve görüntü merkezinde olacak şekilde beyaz bir zemin üzerine yerleştirilmiştir. Benzer işlem bıyık desenlerinde de yapılarak girdi ve çıktı boyutlarının benzer olması (256x256x3) sağlanmıştır. Ayrıca görseller üzerinde normalizasyon yapılarak ışık dağılımından kaynaklanan farklılıkların elimine edilmesi sağlanmıştır. Ön işlemler sonucu veri kümesinde yer alan birkaç denim kumaş görseli ve bunlara karşılık elle hazırlanan bıyık desen görüntüleri **Şekil 2.3**'te gösterilmektedir.



Şekil 2.3 : Denim2Bıyık veri kümesinden örnek görüntüler.

### 2.4 Test Sonuçlarının Değerlendirilmesinde Kullanılan Benzerlik Metrikleri

#### 2.4.1 SSIM

Yapısal benzerlik metriği (SSIM) parlaklık, maskeleye ve kontrast gibi önemli algısal özelliklerle birlikte görüntüde algılanan değişikliği de analiz eden bir metriktir. Bu metrik görüntüdeki pikseller arasında bulunan konum farklılığını göz ardı ederek, görüntü çiftleri arasındaki benzerliğin hesaplanmasında ortalama ( $\mu$ ) ve standart sapma ( $\sigma$ ) parametreleri üzerine inşa edilen istatistiksel bir ölçümdür [29]. İstatistiksel olarak gerçek ( $x$ ) ve sahte ( $\hat{x}$ ) görüntüsü arasındaki yapısal benzerlik hesaplaması Denklem 2.5'te gösterilmektedir. Bu

denklemdaki gerçek ve sahte görüntünün  $\mu_x$  ve  $\mu_{\hat{x}}$  piksel ortalamasını,  $\sigma_x^2$  ve  $\sigma_{\hat{x}}^2$  varyansını belirtirken  $\sigma_{x\hat{x}}$  gerçek ve sahte görüntü arasındaki kovaryansı belirtmektedir. Ek olarak  $c_1 = (k_1L)^2$  ve  $c_2 = (k_2L)^2$  sabit değerleri, piksel aralığını belirten  $L$  değeri 255. alınarak hesaplanır.

$$SSIM(x, \hat{x}) = \frac{(2\mu_x\mu_{\hat{x}} + c_1)(2\sigma_{x\hat{x}} + c_2)}{(\mu_x^2 + \mu_{\hat{x}}^2 + c_1)(\sigma_x^2 + \sigma_{\hat{x}}^2 + c_2)} \quad (2.5)$$

#### 2.4.2 Kesinlik skoru

İki görüntü örneği arasındaki benzerlik maliyetini hesaplamak için gerçek ( $x$ ) ve sahte ( $\hat{x}$ ) görüntüdeki piksel değerinin eşitliği kontrol edilir [30]. Değerler eşitse 1 döndürülerek toplanır, değilse 0 döner. Toplam sonuç piksel sayısına ( $n$ ) bölünür. Metriğin ana formülü Denklem 2.6 gibi tanımlanır.

$$Acc(x, \hat{x}) = \frac{1}{n} \sum_{i=1}^n 1(x_i = \hat{x}_i) \quad (2.6)$$

#### 2.4.3 MAE

İki görüntü örneği arasındaki benzerlik maliyeti noktasal hatanın mutlak değeri alınarak hesaplanır [27]. Bu metriğin ana formülü Denklem 2.7 gibi tanımlanır. Denklem de gerçek ( $x$ ) ve sahte ( $\hat{x}$ ) görüntü arasındaki herbir piksel farkının mutlak değeri alınıp toplanır. Toplam sonuç piksel sayısına ( $n$ ) bölünür. Ölçüm yapılırken hatanın yönü dikkate alınmaz, mutlak farka odaklanır.

$$L_1 = MAE(x, \hat{x}) = \frac{1}{n} \sum_{i=1}^n |x_i - \hat{x}_i| \quad (2.7)$$

### 2.5 Uygulama Sonuçları

Denim-bıyık görsellerine ait veri kümesinin eğitim işlemi GTX 1660 Titan ekran kartına sahip bir bilgisayarda gerçekleştirilmiştir. 300 iterasyon gerçekleşen eğitim işlemi yaklaşık olarak 15 saat sürmüştür. Eğitim ve test işlemleri için Python programlama dili ve Keras kütüphanesi kullanılmıştır. Veri kümesi %90 eğitim ve %10 test olarak bölümlenmiştir.

Pix2Pix mimarisinin ilk eğitimi tamamlandıktan sonra kayıp fonksiyonları incelenmiş ve eğitim kümesinin kaybı çok düşük olmasına rağmen test kümesinin yüksek bir kayba sahip olduğu görülmüştür. Bu sonuçlar ağın orijinal halinin aşırı öğrenme gerçekleştirdiğini göstermektedir. Aşırı öğrenmenin engellenmesi için çözücü ağı son 4 katmanında Dropout operasyonu gerçekleştirilmiştir. **Çizelge 2.1**'de orijinal ve güncellenmiş Pix2Pix mimarilerinin doğruluk oranlarının sonuçları verilmektedir. Orijinal Pix2Pix mimari sonuçları v1 ile gösterilirken güncellenmiş mimari v2 olarak ifade edilmektedir.

**Çizelge 2.1** : Orijinal ve güncellenmiş Pix2Pix mimarisinin eğitim ve test sonuçları.

Model Adı	Aşama	Metrik Adı (Benzerlik)		
		SSIM ↑	Kesinlik skoru ↑	MAE ↓
Pix2Pix v1	Test	0.86	0.58	72.02
Pix2Pix v1	Eğitim	0.94	0.66	55.54
Pix2Pix v2	Test	<b>0.89</b>	<b>0.63</b>	<b>61.53</b>
Pix2Pix v2	Eğitim	<b>0.92</b>	<b>0.70</b>	<b>40.96</b>

Eğitim tamamlandıktan sonra her iki mimarinin de test sonuçları elde edilmiştir. **Şekil 2.4**'te birkaç denim kumaş görseline ait test sonuçları gösterilmektedir. Buna göre Pix2Pix v1 mimarisiyle elde edilen görsellerin yüksek oranda gürültü içerdiği görülmüştür. Güncellenmiş Pix2pix v2 mimarisiyle önemli ölçüde gürültülerin elimine edildiği gözlemlenmiştir. Ancak hedeflenen gerçek bıyık desen görüntüsünün henüz istenilen benzerlikte üretilmediği görülmektedir. Bunun en büyük nedeni, veri kümesindeki görsellerin belirli bir standartta çekilmemiş olmasıdır. Tezin ilerleyen bölümlerinde bu konuyla ilgili çalışmalar yapılmıştır.

Denim Kumaşı Gerçek Bıyık Deseni Pix2Pix v1 Sonucu Pix2Pix v2 Sonucu



Şekil 2.4 : Denim2Bıyık test veri kümesinden birkaç örneğin üretim sonuçları.



### 3. BIYIK DESENİ ÜRETİMİNDE ÇÜA MİMARİLERİNİN PERFORMANS İNCELENMESİ

Tezin bu bölümünde görüntüden görüntüye dönüşüm yapan çekişmeli üretici ağ mimarilerinin performans incelemesi yapıp, sentetik görüntü üretimindeki başarımları değerlendirilmiştir. Bu modellerin kaliteli başarımlarını değerlendirmesi için standartlaştırılmış veri kümeleri yerine iş alanından toplanılan Denim2Bıyık veri kümesi kullanılmıştır. Otomatik bıyık üretim işleminin gerçekleşmesi ve manuel üretimdeki hataların elimine edilmesi yapılan çalışmanın akademik özgün değerini yükseltmektedir. Çalışmada kullanılan ÇÜA mimarileri Pix2Pix, CycleGAN, DiscoGAN [31] ve AttentionGAN'dır [32]. Her bir mimarinin denim2bıyık veri kümesindeki eğitim ve test verileri ile bıyık deseni üretim başarımlarını değerlendirilmesi ve maliyet analizi yapılmıştır. Pix2Pix ve DiscoGAN mimarisinin detaylı açıklamaları sırasıyla bölüm 2.1 ve 5.1'de verilmiştir.

#### 3.1 CycleGAN Mimarisi

CycleGAN mimarisi eşleştirilmiş görüntü çiftlerine ihtiyaç duymadan bir görüntünün X alanından Y alanına dönüşüm yapmasını sağlar [7]. Bu mimari Pix2Pix üzerinde yapılan güncellemeler sonucunda 2017 yılında geliştirilmiştir. Pix2Pix eşleştirilmiş görüntü çiftlerine ihtiyaç duyarken, CycleGAN ile bu gereksinim ortadan kaldırılmıştır. Başka bir deyişle, kaynak ve hedef alan arasında bire bir eşleştirme olmadan bir alandan diğerine dönüşüm yapabilir. CycleGAN mimarisi bir görüntü kümesinden diğer görüntü kümesine dönüşüm işlemi için döngü tutarlılık kaybı kullanır [7]. Genel yapısı Pix2Pix'e benzerdir ve üretici ağı aynı mimariyi kullanmaktadır. Pix2Pix bir üretici ve bir ayırıcı ağına sahipken, CycleGAN iki üretici ve iki ayırıcı ağına sahiptir [6,7].

Mimarinin eğitiminde kullanılan iki üretici ( $G_{X2Y}$  ve  $G_{Y2X}$ ) ve iki ayırıcı ( $D_X$  ve  $D_Y$ ) ağların çalışma mekanizması:

- Üretici  $G_{X2Y}$  ağı, X alanındaki ( $x$ ) görüntüyü Y alanındaki ( $y$ ), görüntüye dönüştürmeyi öğrenir. ( $G_{X2Y}: x \rightarrow y$ )
- Üretici  $G_{Y2X}$  ağı, Y alanındaki görüntüyü X alanındaki görüntüye dönüştürmeyi öğrenir. ( $G_{Y2X}: y \rightarrow x$ )
- Ayırıcı  $D_X$  ağı, X alanındaki ( $x$ ) görüntü ile  $G_{X2Y}$  ağının sonucunda üretilen  $\hat{x}$  görüntüsünü gerçek ya da sahte olarak sınıflandırmaya çalışır.

- Ayırıcı  $D_Y$  ağı,  $Y$  alanındaki ( $y$ ) görüntü ile  $G_{Y2X}$  ağının sonucunda üretilen  $\hat{y}$  görüntüsünü gerçek ya da sahte olarak sınıflandırmaya çalışır.

CycleGAN mimarisinde gerçek görüntü ile yeniden inşa edilen (reconstructed) görüntü arasındaki benzerlik maliyeti için döngü tutarlılık kaybı ( $\mathcal{L}_{cyc}$ = cycle consistency loss) kullanılır. Bu maliyet işlemi Denklem 3.1 gibi ifade edilir. Denklem de gerçek görüntüler ile yeniden inşa edilen imajları arasında ortalama karesel hata (MSE [33]) metriği ile benzerlik maliyeti hesaplanır.

$$\mathcal{L}_{cyc} = L_2(G_{Y2X}(\hat{y}) - x) + L_2(G_{X2Y}(\hat{x}) - y) \quad (3.1)$$

Üretici ağlar birinci alandan ikinci alana dönüşüm sağlarken, ağa ikinci alandan bir görüntü gelirse görüntünün yapısı değişmemelidir. Mimarinin eğitim aşamasında bu şartı sağlayabilmesi için özdeşlik (identity) kaybı kullanılır. Bu kaybın formülü Denklem 3.2 gibi ifade edilir.

$$\mathcal{L}_{identity} = L_2(G_{Y2X}(x) - x) + L_2(G_{X2Y}(y) - y) \quad (3.2)$$

Mimarideki iki üretici ve iki ayırıcı ağın maliyetleri Denklem 2.1'deki geleneksel ÇÜA yöntemi ( $\mathcal{L}_{CGAN}$ ) ile hesaplanır. Buna ek olarak döngü tutarlılık ve özdeşlik kayıpları olduğu için toplam maliyet üç ayrı kayıptan oluşur. Genel maliyet Denklem 3.3 gibi tanımlanır. Bu denklemdeki  $\lambda$  değeri 10 alınarak hesaplama yapılır.

$$\begin{aligned} & \mathcal{L}(G_{X2Y}, G_{Y2X}, D_X, D_Y) \\ &= \mathcal{L}_{CGAN}^x(G_{X2Y}, D_Y) + \mathcal{L}_{CGAN}^y(G_{Y2X}, D_X) + \lambda * (\mathcal{L}_{cyc} + \mathcal{L}_{identity}) \end{aligned} \quad (3.3)$$

Optimizasyonun bir parçası olarak, Denklem 3.4'teki gibi kayıp işlevi optimize edilir.

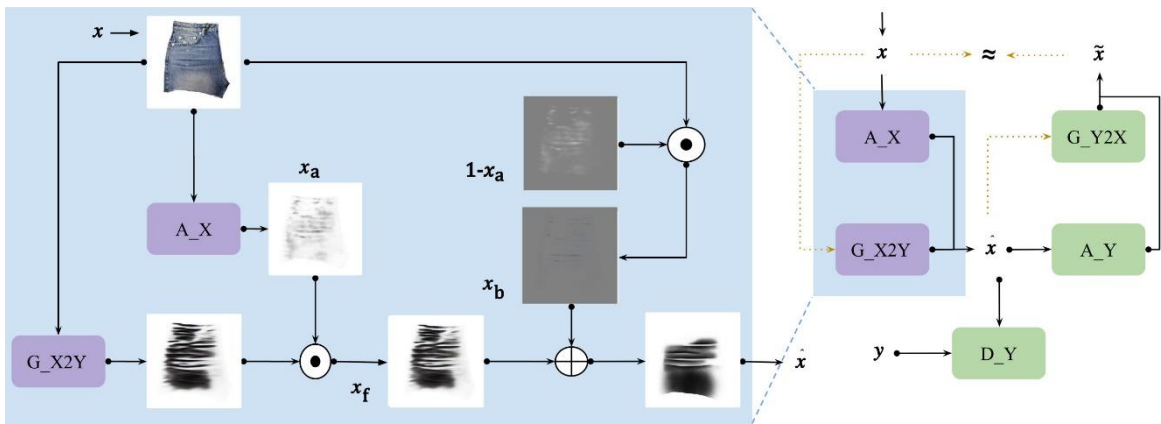
$$\mathcal{L}_{G,D} = \arg \min_{G_{X2Y}, G_{Y2X}} \max_{D_X, D_Y} \mathcal{L}(G_{X2Y}, G_{Y2X}, D_X, D_Y) \quad (3.4)$$

Üretici ( $G_{X2Y}$ ,  $G_{Y2X}$ ) ağların kayıp değeri minimize edilmeye çalışılırken, ayırıcı ( $D_X$ ,  $D_Y$ ) ağların kayıp değeri maksimize edilmeye çalışılır. Üretici ağların maliyet değerinin düşmesi ağın daha gerçekçi görüntüler üretmesini sağlar ve ayırıcı ağlardaki maliyetin artması üretilen sentetik görüntü ile gerçek görüntü arasındaki ayırım yapmasını zorlaştırır. Bu maliyetin artması ayırıcı ağın gerçek ve sentetik görüntüyü sınıflandırmaya çalışırken daha dikkatli olmasını sağlar. Böylece üretici ve ayırıcı ağlar birbirini besleyerek giriş görüntülerine en uygun dönüşüm görüntüsünü elde etmeye çalışır.

### 3.2 AttentionGAN Mimarisi

AttentionGAN, denetimsiz veri kümelerinde görüntüden görüntüye dönüşüm sağlayan dikkat güdümlü bir mimaridir [32]. Başlangıç noktası olarak CycleGAN mimarisi referans alınarak, sıfır ile bir aralığında değerleriyle dikkat haritaları oluşturan, dikkat ağı'nın mimariye entegrasyonu ile geliştirilmiştir [7,32]. Klasik mimarilerde üretici ve ayırıcı ağları sadece kodlayıcı ve kod çözücü yapısını barındırmasından dolayı görüntü üzerinde belirli bir alana dikkat yoğunlaştırmaları zordur. Bu yüzden görüntünün odaklanması gereken bölümüne yeterli önem verilememektedir. AttentionGAN görüntü içeriğini üretici ağ yapısıyla iletirken aynı zamanda görüntüde odaklanması gereken kısımların dikkat haritasını da çıkartmaktadır [32]. Böylece ağ değişimin olduğu yerlere odaklanıp arka plan veya ön plan içeriğine gereğinden fazla odaklanmasını engeller.

AttentionGAN mimarisinin akış diyagramı Şekil 3.1'de gösterilmiştir. Mimari;  $A_X$ ,  $A_Y$  dikkat,  $G_{X2Y}$ ,  $G_{Y2X}$  üretici ve  $D_X$ ,  $D_Y$  ayırıcı ağlarından oluşmaktadır. Giriş görüntüsü ( $x$ ), bu görüntünün dönüştürülmek istendiği görüntü ( $y$ ) olarak ifade edilir. Giriş görüntüsü üretici  $G_{X2Y}$  ve dikkat  $A_X$  ağına verilerek elde edilen dikkat haritası ( $x_a$ ) ve içerik görüntüsü 3 kanalda (RGB) noktasal olarak çarpılır. Elde edilen çıktı  $x_f$  ön plan (foreground) görüntüsü olarak adlandırılır. Arka plan (background) içeriği içinde  $x$  giriş görüntüsü ve  $x_a$  dikkat haritasının tersi alınarak noktasal olarak çarpılır. Ön plan ve arka plan görüntüleri toplanarak giriş görüntüsünün dönüşmüş hali  $\hat{x}$  elde edilir. Çift taraflı doğrulama işlemi için  $\hat{x}$  görüntüsü  $G_{Y2X}$  ve  $A_Y$  ağına verilerek aynı işlem tekrar edilir. Elde edilen  $\tilde{x}$  görüntüsü giriş görüntüsünün yeniden inşa edilmiş imajıdır. Son olarak  $x$  ve  $\tilde{x}$  görüntüsü arasında döngü tutarlılık maliyeti hesaplanır [32].



Şekil 3.1 : AttentionGAN mimarisinin çalışma mekanizması.

AttentionGAN, CycleGAN mimarisi gibi gerçek görüntü ile yeniden inşa edilen görüntü arasındaki benzerlik maliyeti için döngü tutarlılık kaybı kullanılır [7,32]. Bu maliyet fonksiyonu Denklem 3.5 gibi tanımlanır. Denklemdaki maliyet  $x$  ve  $y$  giriş görüntüleri ile  $\tilde{x}$  ve  $\tilde{y}$  yeniden inşa edilmiş imajlar arasında hesaplanır.

$$\mathcal{L}_{cyc} = L_2(x, \tilde{x}) + L_2(y, \tilde{y}) \quad (3.5)$$

Kaynak ve hedef alan için toplam maliyet fonksiyonu Denklem 3.6'da  $\lambda = 10$  alınarak hesaplanır.

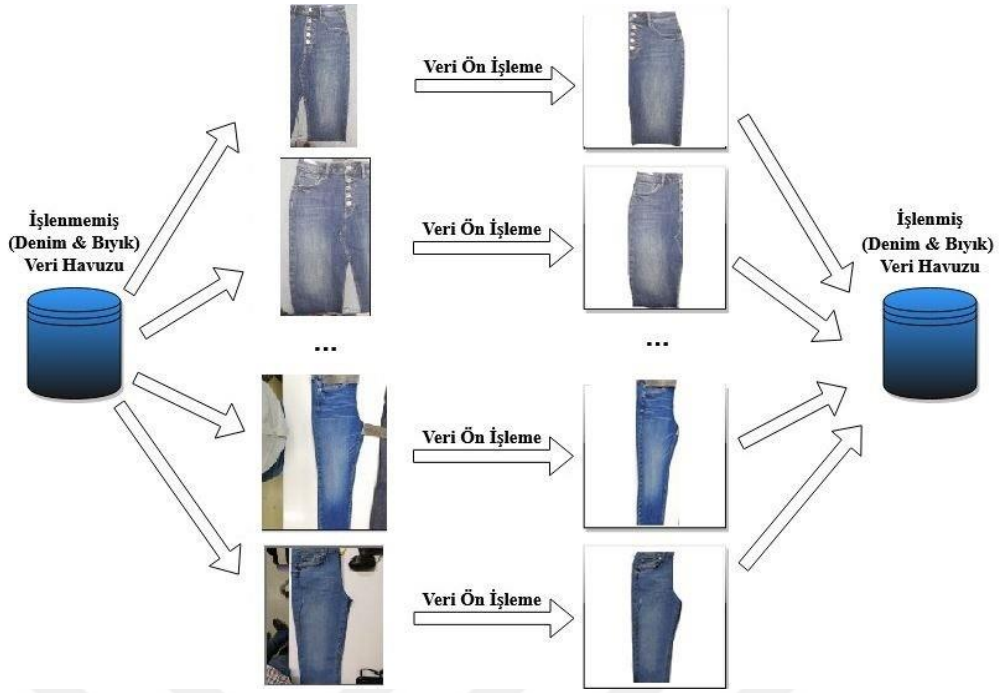
$$\begin{aligned} & \mathcal{L}(G_{X2Y}, G_{Y2X}, A_X, A_Y, D_X, D_Y) \\ &= \mathcal{L}_{cGAN}^x(G_{X2Y}, A_X, D_Y) + \mathcal{L}_{cGAN}^y(G_{Y2X}, A_Y, D_X) + \lambda * \mathcal{L}_{cyc} \end{aligned} \quad (3.6)$$

Optimizasyonun bir parçası olarak Denklem 3.7'de maliyet değeri optimize edilmiş olur.

$$\begin{aligned} & \mathcal{L}_{G,A,D} \\ &= \arg \min_{G_{X2Y}, G_{Y2X}, A_X, A_Y} \max_{D_X, D_Y} \mathcal{L}(G_{X2Y}, G_{Y2X}, A_X, A_Y, D_X, D_Y) \end{aligned} \quad (3.7)$$

### 3.3 Denim2Bıyık Veri Kümesinin İyileştirilmesi

Denim ve bıyık desen çiftlerinden oluşan Denim2Bıyık veri kümesine ait ilk çalışma bölüm 2.3'te detaylı açıklanmıştır. Veri kümesinin yeni olması ve manuel olarak temizlenmesi görüntü çiftlerinde görüntü, bozulma, eşleşme ve benzer problemler içermesine sebep olmuştur. Bu problemlerin indirgenmesi için ilk önce tüm görüntüler PhotoShop programı ile temizlenmiştir. Ardından bu görüntülerdeki döndürme, eşleşim gibi bozuklukları ortadan kaldırmak için Python ortamında OpenCV kütüphanesine ait algoritmalar kullanılarak gerekli düzenlemeler yapılmıştır. Veri kümesinde yüksek derecede bozukluk içeren görüntüler silinmiştir. Bu veri kümesinin nihai halinde 950 adet görüntü çifti kalmıştır. Bu görüntü çiftlerin 900 tanesi eğitim, 50 tanesi test için bölümlenmiştir. Yapılan veri ön işleme şeması **Şekil 3.2**'de gösterilmiştir.



Şekil 3.2 : Denim2bıyık veri kümesinin ön işleme şeması.

### 3.4 Test Sonuçlarının Değerlendirilmesinde Kullanılan Benzerlik Metrikleri

Bu bölümde gerçek (Ground True) ve üretilen görüntülerin karşılaştırmasında kullanılan ölçüm metrikleri (SSIM, MSE, PSNR [34]) anlatılmaktadır. SSIM metriği ile ilgili detaylar bölüm 2.4.1’de açıklanmıştır.

#### 3.4.1 MSE

İki veri örneği arasındaki benzerlik maliyetini hesaplamak için noktasal farkın karesel ortalamasının alındığı bir karşılaştırma metriğidir [33]. Bu metriğin ana formülü Denklem 3.8’de gösterilmiştir. Denklem de gerçek ( $x$ ) ve yapay ( $\hat{x}$ ) görüntüsü arasındaki her bir piksel farkının karesi alınıp toplanır. Toplam sonuç piksel sayısına ( $n$ ) bölünür.

$$L_2 = MSE(x, \hat{x}) = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2 \quad (3.8)$$

Ölçüm değerinin ne kadar düşük olduğu yapay görüntü ve gerçek arasındaki benzerliğin o kadar yüksek olduğunu gösterir.

### 3.4.2 PSNR

Bir sinyalin olası maksimum gücü ile temsilinin doğruluğunu etkileyen bozucu gürültünün gücü arasındaki oran için logaritmik tabanlı kullanılan bir metriktir [34]. Sinyal gerçek görüntü veya veri olarak kabul edilir, gürültü ise verideki sıkıştırma veya bozulmadan kaynaklanan hatadır. İki görüntü arasındaki bu oran desibel olarak hesaplanır. Bu metriğin ana formülü Denklem 3.9'da gösterilmektedir. Denklemdaki gerçek ( $x$ ) ve yapay ( $\hat{x}$ ) görüntüsü arasındaki ölçüm için  $MSE$  değeri ve gerçek görüntünün en büyük piksel değeri ( $MAX_x^2$ ) hesaplamaya dahil edilir.

$$PSNR(x, \hat{x}) = 10 * \log_{10} \left( \frac{MAX_x^2}{MSE(x, \hat{x})} \right) \quad (3.9)$$

Ölçüm değerinin ne kadar yüksek olduğu sinyal kalitesinin o kadar iyi olduğunu gösterir, yani gerçek ile yapay veri arasındaki maliyetin düşük olduğunu gösterir.

### 3.5 Uygulama Sonuçları

Bu çalışmada denim2büyük veri kümesi ile 4 farklı ÇÜA mimarisinin performans karşılaştırması yapılmıştır. Mimarilerin eğitim ve test işlemleri PyTorch derin öğrenme kütüphanesiyle RTX 2080 ekran kartı üzerinde çalıştırılmıştır. Eğitim işlemi 150 iterasyon (epoch) boyunca part boyutu (batch size) 1 alınarak gerçekleştirilmiştir. Mimarilerin temel yazılımsal kodları github: “<https://github.com/hee9joon/Image-to-Image-Translation>” adresli kaynaktan alınmıştır. Bu kaynaktan görüntüden görüntüye dönüşüm yapan temel mimarilerin kodları bulunmaktadır. Temel kodların üzerinde hiperparametre, iterasyon ve görüntü boyutları gibi ayarlamalar ve düzenlemeler yapılmıştır. Böylece bu modellerin uygun koşullar altında başarımlarını performans karşılaştırması için gerekli ortam hazırlanmıştır. Başarım performans kriteri olarak görüntü üretim kalitesi ve bir adet görüntünün üretimi için gereken zamansal maliyet hesaplaması göz önünde bulundurulmuştur. Büyük deseni üretim kalitesi, test kümesindeki görüntü çıktılarının 3 farklı metrik ile karşılaştırılmasıyla elde edilmiştir. Gerçek ve üretilen görüntü arasındaki benzerliği tam değerlendirebilmek için SSIM metriği ile görüntülerin yapısal benzerliği, PSNR yöntemi ile görüntü piksellerindeki maksimum sinyal gücü ve MSE yöntemi ile iki görüntü arasındaki piksel farklılık maliyeti hesaplanmıştır. Mimarilerin test veri kümesinde bulunan 50 görüntü çifti üzerindeki başarımlarını Çizelge 3.1’de gösterilmektedir.












































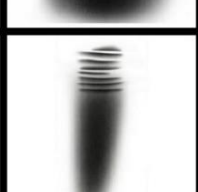
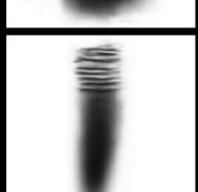



Sonuçlar incelendiğinde Pix2Pix mimarisinin sentetik görüntü üretim aşamasında yapısal (SSIM) olarak yüzde 86 oranında başarımlı göstererek en iyi sonuç verdiği gözlemlenmektedir. DiscoGAN mimarisi eğitim gerçekleştirirken çift taraflı dönüşüm ve doğrulama işlemi kullandığından Pix2Pix mimarisine yakın kalitede görüntüler üretmektedir. Pix2Pix mimarisi aynı zamanda zamansal maliyet karşılaştırmasında da daha iyi sonuç verdiği görülmektedir. Bu mimarinin ağ yapısının denetimli ve daha basit olmasından dolayı kısa sürede kaliteli görüntü ürettiği gözlemlenmiştir.

**Çizelge 3.1** : Mimarilerin Denim2Bıyık test veri kümesindeki başarımlı sonuçları.

Model Adı	Metrik Adı (Benzerlik)			Zaman (Saniye)
	SSIM ↑	PSNR ↑	MSE ↓	Bir görüntü üretim süresi
AttentionGAN	0.757	32.72	2293.70	0,865
CycleGAN	0.754	32.55	2465.01	0,024
Pix2Pix	<b>0.868</b>	<b>33.32</b>	<b>808.37</b>	<b>0,022</b>
DiscoGAN	0.834	32.90	1628.74	0,848

Eğitim aşamasından sonra veri kümesindeki test görüntüleri her bir mimariye verilerek çıktıları elde edilmiştir. **Şekil 3.3**'te test kümesinden birkaç denim kumaş görseline ait sonuçlar gösterilmektedir. Buna göre Pix2Pix ve DiscoGAN mimarisiyle elde edilen görsellerin yüksek oranda gerçek sonuçlara benzediği görülmüştür. CycleGAN ve AttentionGAN mimarilerinin ise bıyık desen üretiminde beklenen seviyede kaliteli görüntüler üretilmediği görülmektedir.



Denim Kumaş	Gerçek Bıyık Deseni	Pix2Pix	DiscoGAN	CycleGAN	AttentionGAN
					
					
					
					
					
					
					
					

Şekil 3.3 : Mimarilerin Denim2Bıyık test kümesindeki görüntü üretim sonuçları.



## 4. ALGISAL BENZERLİK MALİYETİ TABANLI ÇÜA MİMARİSİ İLE GÖRÜNTÜ ÜRETİMİ

Görüntü sentezleme, farklı yapıya sahip görseller arasında yapılan bir dönüşümdür. Evrişimsel Sinir Ağlar (ESA [35]) ve ÇÜA mimarileri görüntü sentezlemede yaygın olarak kullanılmaktadır. Bu çalışmanın amacı ÇÜA mimarileri için en uygun maliyet fonksiyonunu bulabilmek ve daha iyi görüntü sentezleyebilmektir. Bunun için ÇÜA mimarilerinden olan Pix2Pix yöntemi üzerinde maliyet fonksiyonları değiştirilerek deneysel sonuçlar elde edilmiştir. Pix2Pix yönteminde kullanılan mevcut maliyet fonksiyonu ortalama mutlak hata (MAE) yani  $L_1$  metriğidir. Pix2Pix yönteminde kullanılan maliyet fonksiyonları olarak ESA tabanlı görüntü karşılaştırma ölçütleri CONTENT [36], LPIPS [37] ve DISTS'dir [38]. Aynı zamanda Pix2Pix'te bulunan  $L_1$  maliyetine ESA tabanlı görüntü karşılaştırma maliyet metrikleri eklenerek  $L_1\_CONTENT$ ,  $L_1\_LPIPS$  ve  $L_1\_DISTS$  sonuçlarına yer verilmiştir. Test çalışması Denim2Bıyık veri kümesi üzerinde yapılmıştır. Sonuçların analizinde klasik görüntü karşılaştırma ölçütleri (FSIM [39], HaarPSI [40], MS\_SSIM [41], PSNR, SSIM, VIFp [42] ve VSI [43]) ve güncel görüntü metrikleri olarak FID [44] ve KID [45] kullanılmıştır. Pix2Pix yönteminin detaylı açıklaması bölüm 2.1'de detaylı olarak verildiği için tezin bu bölümünde açıklama gereği duyulmamıştır.

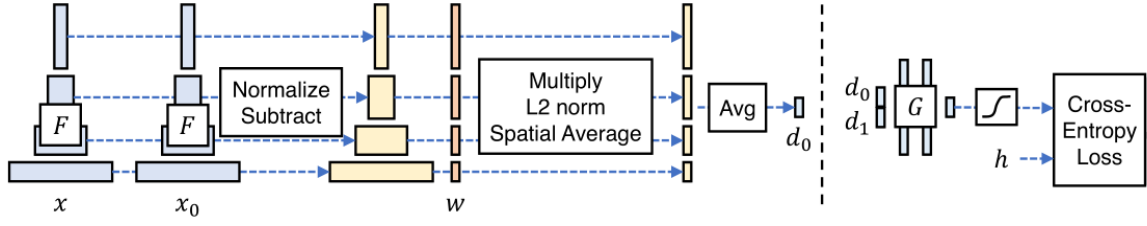
### 4.1 ESA Tabanlı Görüntü Karşılaştırma Ölçütleri

Literatürde görüntü çiftleri arasında benzerlik skorunu üretmek için eğitilmiş ESA mimarilerini kullanan üç farklı ölçüt olduğu görülmektedir: LPISP, DISTS ve CONTENT. ESA tabanlı modellerden LPIPS ve DISTS fonksiyonları birbirine benzemektedir ancak yüksek hesaplama ve yorumlanabilirlik eksikliği, modellerin uygulanabilirliğini azaltabilmektedir [46].

#### 4.1.1 LPIPS

LPIPS (Learned Perceptual Image Patch Similarity), görüntü çiftleri arasındaki benzerliği hesaplamak için kullanılan en güncel ölçütlerden biridir [37]. **Şekil 4.1**'de LPIPS fonksiyonunun nasıl hesaplandığı gösterilmektedir. Buna göre  $x$  ve  $x_0$  görüntülerin  $F$  eğitilmiş ESA ağındaki (VGG [47]) aynı seviyeli katman çıktıları (aktivasyonlar) normleştirilir ve birbirinden çıkarılır. Elde edilen fark verileri ölçeklenir (scaling) ve

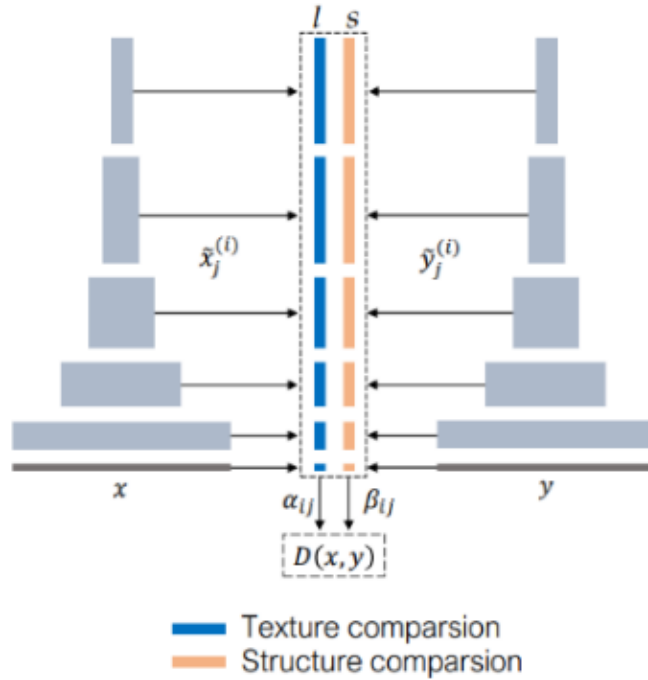
çıktılar tek bir vektörel forma ( $w$ ) dönüştürülüp vektör normu ( $L_2$ ) hesaplanarak  $d$  uzaklığı elde edilir [37].



**Şekil 4.1 :** LPIPS fonksiyonu. (Sol) İki görüntü parçası arasındaki uzaklık, (Sağ) İki uzaklık değerlerden  $h$  algısal kararını tahmin etmek için yapılan küçük bir ağ eğitimi [37].

#### 4.1.2 DISTS

DISTS (Deep Image Structure and Texture Similarity) fonksiyonu, VGG ağını kullanmaktadır [38]. DISTS uzaklığı  $x$  ve  $y$  giriş görselleri arasında **Şekil 4.2**'deki gibi hesaplanır. DISTS fonksiyonu, yapı ve doku benzerliğini birleştirir, hafif geometrik bozulmalara karşı dayanıklıdır ve dokusal görüntülerde iyi performans gösterir.

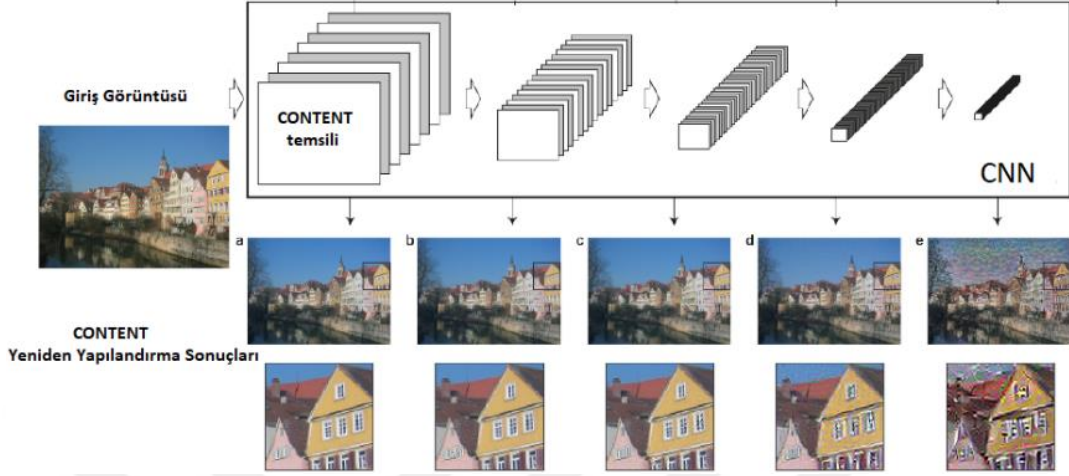


**Şekil 4.2 :** DISTS fonksiyonu [38].

#### 4.1.3 CONTENT

CONTENT fonksiyonu, orijinal görüntüyü VGG ağı ile evrişimli katmanların her birinden geçirerek girdi görüntüsünü yeniden oluşturur [36]. CONTENT fonksiyonu, diğer ESA

tabanlı yöntemlerden farklı olarak görüntü içeriği ve stilini birbirinden ayırmak için kullanılır. Şekil 4.3'te CONTENT yönteminin nasıl çalıştığına dair bir örnek gösterilmektedir.



Şekil 4.3 : CONTENT fonksiyonu [36].

#### 4.1.4 Ölçütlerin $L_1$ ile kullanımı

Düzgünleştirme terimi hesaplanırken %50 oranda  $\{LPIPS, DISTS, CONTENT\}$  ile %50 oranda  $L_1$  kullanımı da incelenmiştir. Buna göre Pix2Pix'in üretici ağ maliyeti herbir ESA yöntemi ( $LPIPS, DISTS, CONTENT$ ) için bireysel olarak Denklem 4.1'deki gibi güncellenmektedir. ESA yöntemleri ile  $L_1$  maliyetinin ortak olarak kullanılmasında Denklem 4.2 gibi ifade edilmektedir.

$$\mathcal{L}_{G,D} = \arg \min_G \max_D \mathcal{L}_{CGAN}(G, D) + \lambda * \mathcal{L}_{\{LPIPS, DISTS, CONTENT\}}(x, \hat{x}) \quad (4.1)$$

$$\begin{aligned} \mathcal{L}_{G,D} = \arg \min_G \max_D \mathcal{L}_{CGAN}(G, D) + \lambda \\ * \{0.5 * L_1(x, \hat{x}) + 0.5 * \mathcal{L}_{\{LPIPS, DISTS, CONTENT\}}(x, \hat{x})\} \end{aligned} \quad (4.2)$$

#### 4.2 Klasik Görüntü Karşılaştırma Ölçütleri

Özellik Tabanlı Benzerlik İndeksi Ölçümü (Feature Similarity Index Measurement, FSIM) [39], görüntü çiftlerinin faz uyumu ve gradyan büyüklüğü özelliklerini karşılaştırır.

Haar dalgacık tabanlı algısal benzerlik indeksi (Haar Wavelet-Based Perceptual Similarity Index, HaarPSI) [40], görüntü parçalarından çıkarılan yerel dalgacık katsayıların karşılaştırılmasına dayanır.

Yapısal benzerlik ölçütünde (Structural Smilarity Index Metric, SSIM), görüntü çiftlerinin benzerlik skorunu elde etmek için ortalama ( $\mu$ ) ve standart sapma ( $\sigma$ ) gibi birkaç istatistiksel moment bilgisi kullanılır. Çok Ölçekli SSIM (Multi Structural Smilarity Index Metric, MS-SSIM) [41] ölçütünde farklı çözünürlük seviyelerinde hesaplanan SSIM sonuçlarının birleştirilmesi söz konusudur.

Pik Sinyal Gürültü Oranı (Peak Signal Noise Ratio, PSNR), en yaygın kullanılan objektif görüntü kalitesi metriğidir. Bununla birlikte, insan görsel sisteminin karmaşık, oldukça doğrusal olmayan davranışı nedeniyle PSNR değerleri algılanan görüntü kalitesiyle iyi bir şekilde ilişkili değildir.

Görsel bilgi doğruluğu (Visual Information Fidelity, VIF) [42], sentetik ve gerçek görüntüler arasında paylaşılan bilgileri ölçmek için bozulma (kanal) modeliyle birlikte doğal sahne istatistik modelleri (NSS) kullanır. Bu yöntem izleyicinin dikkatini çeken bir alanda meydana gelen bir bozulmanın diğer herhangi bir alana göre daha rahatsız edici olduğu varsayımına dayanır ve yerel bir belirginlik haritası ile yerel bozulmaları ağırlıklandırmaya çalışır.

Görsel Belirginliğe Bağlı İndeks (Visual Saliency-Induced Index, VSI) [43], izleyicinin dikkatini çeken bir alanda meydana gelen bir bozulmanın diğer herhangi bir alana göre daha rahatsız edici olduğu varsayımına dayanır ve yerel belirginlik haritası ile yerel bozulmaları ağırlıklandırmaya çalışır.

### 4.3 Güncel Görüntü Karşılaştırma Ölçütleri

Fréchet başlangıç mesafesi (Fréchet Inception Distance, FID)[44], ÇÜA'lardan üretilen görüntülerin kalitesini değerlendirmek için kullanılan performans ölçütüdür. FID, üretilen görüntülerin dağılımını gerçek görüntülerin dağılımıyla karşılaştırır.

Kernel başlangıç mesafesi (Kernel-Inception distance, KID) [45], FID'e benzer metrik olan ve görüntüler arasında karesel Maksimum Ortalama Tutarsızlık (MMD) kullanan metriktir. Aktivasyonların dağılımında parametrik form almadığından ve ReLU aktivasyon fonksiyonu barındırdığı için FID metriğine göre avantajlıdır.

### 4.4 Uygulama ve Sonuçları

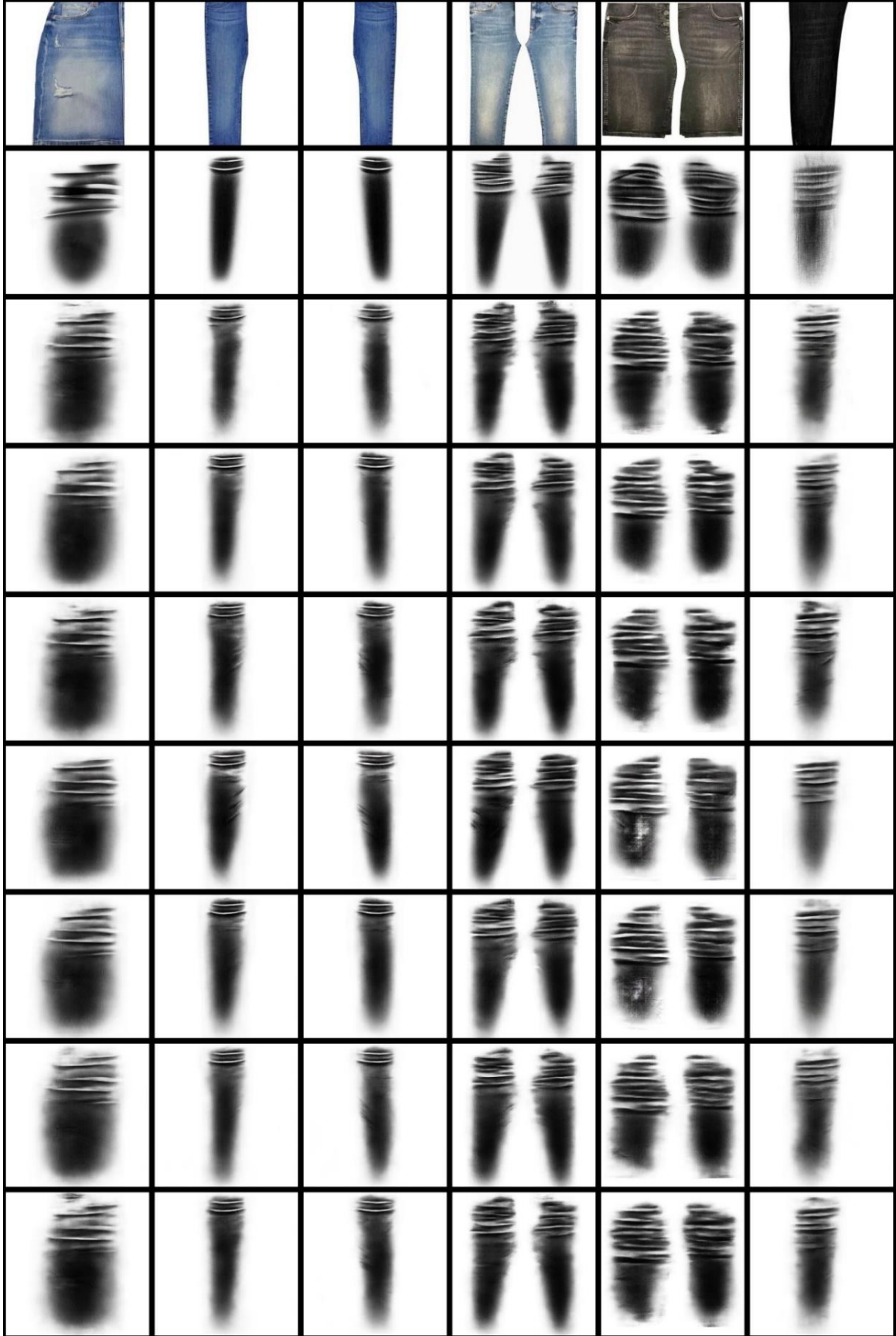
Pix2Pix yaklaşımının farklı maliyet fonksiyonlarıyla kullanıldığındaki etkisini anlamak için Denim2Bıyık veri kümesindeki görseller kullanılmıştır. Böylece önerilen sentezleme

yöntemlerinin görüntü üretimindeki performansları değerlendirilmektedir. Denim2Bıyık veri kümesi sadece kot nesnelere ait görselleri içermekte ve çıktı gri ölçekli bir resim (Bıyık motifi) olmaktadır. Bu veri kümesine ait elde edilen sonuçlar **Şekil 4.4**'te gösterilmektedir. Ayrıca sayısal benzerlik sonuçları **Çizelge 4.1**'de verilmektedir. Denim2Bıyık veri kümesinde klasik benzerlik ölçüm metriklerine bakıldığında FSIM, HaarPSI, MS-SSIM, PSNR, SSIM, VSI'a göre CONTENT yönteminin; VIFp için ise LPIPS fonksiyonunun diğer yöntemlerle kıyaslandığında daha iyi sonuç verdiği gözlemlenmiştir. **Çizelge 4.1**'deki güncel FID ve KID metriklerinden LPIPS fonksiyonunun daha verimli olduğu gözlemlenmiştir.

Sonuç olarak, veri kümesinde yapılan sentezleme sonuçları genel olarak değerlendirildiğinde CONTENT ve LPIPS fonksiyonlarının diğer ölçütlere kıyasen tatmin edici düzeyde doğruluk sağladığı görülmüştür. Böylece ÇÜA mimarileri için CONTENT, DISTS ve LPIPS fonksiyonlarının maliyet ölçümü için kullanılabilmesi görülmüştür.

**Çizelge 4.1** : Görüntü sentezleme doğrulukları (Satırlar: ESA tabanlı kullanılan yöntemler, Sütunlar: Klasik ve güncel benzerlik ölçütleri).

Yöntemler	Benzerlik Ölçütleri								
	Klasik							Güncel	
	FSIM	HaarPSI	MS_SSIM	PSNR	SSIM	VIFp	VSI	FID	KID
<b>L<sub>1</sub></b>	0.824	0.561	0.854	16.96	0.842	0.28	0.92	207.1	0.121
<b>LPIPS</b>	0.824	0.559	0.855	16.87	0.846	<b>0.30</b>	0.92	<b>130.7</b>	<b>0.042</b>
<b>L<sub>1</sub>LPIPS</b>	0.822	0.556	0.852	16.84	0.843	0.29	0.92	157.7	0.058
<b>DISTS</b>	0.825	0.551	0.847	16.84	0.839	0.27	0.92	149.7	0.052
<b>L<sub>1</sub>DISTS</b>	0.826	0.557	0.852	16.91	0.844	0.29	0.92	133.9	0.043
<b>CONTENT</b>	<b>0.848</b>	<b>0.565</b>	<b>0.858</b>	<b>17.05</b>	<b>0.849</b>	0.29	<b>0.93</b>	165.8	0.075
<b>L<sub>1</sub>CONTENT</b>	0.824	0.562	0.855	17.01	0.846	0.29	0.92	176.8	0.086



Şekil 4.4 : Denim2Bıyık veri kümesi sentezleme çıktıları. Satırlar: Girdi, Çıktı, L<sub>1</sub>, LPIPS, L<sub>1</sub>\_LPIPS, DISTS, L<sub>1</sub>\_DISTS, CONTENT, L<sub>1</sub>\_CONTENT.

## 5. ÇÜA İLE GÖRÜNTÜ ÜRETİMİNDE YENİ BİR MİMARİNİN GELİŞTİRİLMESİ

Koşullu görüntü sentezleme aynı boyutlara sahip farklı alanlardaki görüntülerin birbirine dönüşümüdür. Bu alanda yapılan çalışmalarda ÇÜA mimarileri yaygın bir şekilde kullanılmaktadır. Klasik ÇÜA yaklaşımıyla görüntü dönüşümünde üretici ağır kodlayıcı ve kod çözücü kısımları arasında veri aktarımı yapılmaktadır. Yapılan bu veri aktarımı, üretilen görüntü kalitesini arttırırken, diğer yandan veri bağımlılığına yol açmaktadır. Bu bağımlılığın iki olumsuz etkisi bulunmaktadır: 1) üretilen görsellerdeki hataya kodlayıcı yada kod çözücü kısmın mı neden olduğunun anlaşılmasını engeller; 2) görüntüdeki üretim kalitesinin ağır parametre artışına bağlanmasına sebep olur. Bu çalışmada, farklı alanlar arası görüntü çevirisinde yüksek seviyedeki öznelik parametrelerini eşitleme ilkesine dayanarak geliştirilen iki farklı mimari (dY-Net, uY-Net) önerilmektedir. Bu mimarilerin ilki görüntü üretim hızına diğeri görüntü üretim kalitesine odaklanmaktadır. Üretimdeki hız performansını odaklanan dY-Net mimarisinde, veri bağımlılığı bulunmaz ve parametre uzayında önemli derecede daralma söz konusudur. Görüntü üretim kalitesine odaklanan uY-Net mimarisinde SSIM ve PSNR gibi kaliteyi ölçen metriklerin sonuçlarını maksimize etmeye çalışılmaktadır. Önerilen mimariler ile mevcut sentezleme yaklaşımlarının performans testleri için üç farklı veri kümesi (Maps, Cityscapes ve Denim2Bıyık) kullanılmıştır.

### 5.1 Ölçüm Metrikleri

Bu çalışmada sıkça kullanılan ölçüm metrikleri verilmiştir. Bu metrikler SSIM, MSE, PSNR, MAE ve Hinge gömüllü maliyetidir. SSIM ve MAE metriği bölüm 2.4'te anlatılmıştır. MSE ve PSNR metriği bölüm 3.4'te açıklanmıştır.

#### 5.1.1 Hinge gömülü maliyeti

Genellikle doğrusal olmayan veya yarı denetimli veri blokları arasındaki benzerliği hesaplamak için kullanılan bir metriktir [48]. Bu metrik bir gerçek girdi vektörü ( $x$ ) ile etiket vektörü (1 veya -1 içeren) arasındaki mesafeyi ölçer. Metriğin ana formülü Denklem 5.1'de verilmiştir. Denklemdeki  $L$  değeri girdi ve etiket listesini içermektedir. Listedeki özellik liste sayısı  $n$  olarak ifade edilir.

$$L_{HE} = \begin{cases} \text{mean}(L), & \text{if reduction} = \text{'mean'}, \\ \text{sum}(L), & \text{if reduction} = \text{'sum'}, \end{cases} \quad (5.1)$$

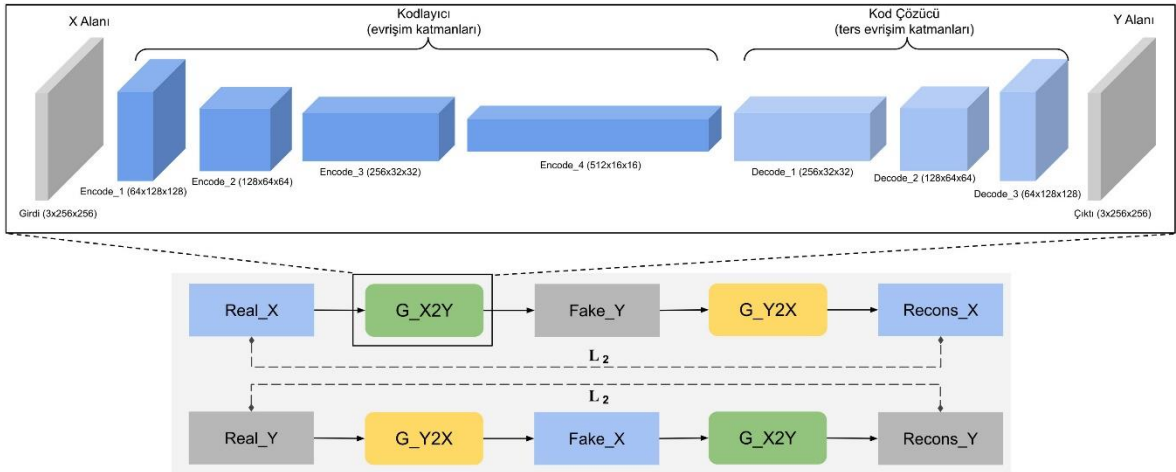
$$L = \sum_i^n \ell(x_i, \hat{x}_i) = \begin{cases} L_2(x_i, \hat{x}_i), & \text{if } \hat{x}_i = 1, \\ \max(0, m - L_2(x_i, \hat{x}_i)), & \text{if } \hat{x}_i = -1, \end{cases}$$

## 5.2 DiscoGAN Mimarisi ile Görüntü Üretimi

DiscoGAN mimarisi farklı veri kümeleri arasındaki ilişkiyi keşfetmek ve kaliteli görüntüler sentezleyebilmek için 2017 yılında geliştirilmiş koşullu bir ÇÜA mimarisidir [31]. Bu mimari denetimli veya denetimsiz veri kümeleri üzerinde çalışabilen bir yapıya sahiptir.

### 5.2.1 Üretici ağ

DiscoGAN mimarisi aynı yapıya sahip iki üretici ağı sahiptir. Bu mimari iki ayrı dönüşüm işlevini öğrenir, birinci ağ ( $G_{X2Y}$ ) X alanındaki görüntüyü ( $Real_X$ ) Y alanındaki sahte bir görüntüye ( $Fake_Y$ ) dönüştürmeyi öğrenirken, ikinci ağ Y alanındaki görüntüyü ( $Real_Y$ ) X alanındaki sahte bir görüntüye ( $Fake_X$ ) dönüştürmeyi öğrenir. Mimari ilk işlemde sonra tam bir öğrenme için tahmini  $Fake_Y$  görüntüsünü  $G_{Y2X}$  ağından geçirerek yeniden inşa edilmiş  $Recons_X$  görüntüsünü oluştururken,  $Fake_X$  görüntüsünü  $G_{X2Y}$  ağından geçirerek yeniden inşa edilmiş  $Recons_Y$  görüntüsünü oluşturmaktadır. DiscoGAN mimarisindeki üretici ağların ana şeması Şekil 5.1'de gösterilmektedir.



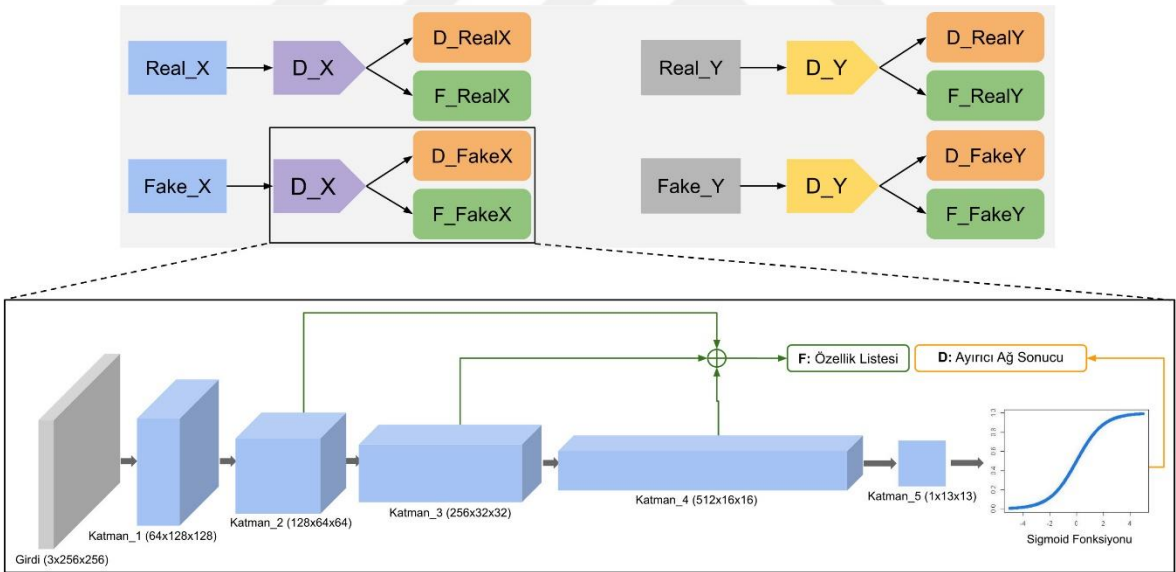
Şekil 5.1 : DiscoGAN üretici ağların şeması.



## 5.2.2 Ayırıcı ağ

DiscoGAN mimarisi aynı yapıya sahip iki ayırıcı ağa sahiptir. Bu ağlardan ilki ( $D_X$ ) X alanındaki gerçek görüntü olan  $Real_X$  ile sahte görüntü olan  $Fake_X$  ait özellikleri öğrenmeye çalışarak, iki görüntünün çıktısını gerçek veya sahte olarak sınıflandırmaya çalışır. Diğer ağ ( $D_Y$ ) Y alanındaki gerçek görüntü olan  $Real_Y$  ile sahte görüntü olan  $Fake_Y$  ait özellikleri öğrenmeye çalışarak iki görüntünün çıktısını gerçek veya sahte olarak sınıflandırmaya çalışır.

Ayırıcı ağların genel şeması Şekil 5.2’de verilmiştir. Bu şemaya göre ayırıcı ağa girdi olarak verilen görüntüden iki çıktı elde edilir. İlk çıktı ayırıcı ağın tartışmalı maliyet için kullandığı aktivasyon fonksiyonundan geçirilmiş çıktıdır. İkinci çıktı ağın 2,3 ve 4 numaralı katmanlarından elde edilen özellik (Feature) listesidir. Bu özellik listesi üretici ağın maliyet fonksiyonu için kullanılır. Örneğin  $Real_X$  görüntüsü  $D_X$  ağına verilerek elde edilen  $D_{RealX}$  sonucu ayırıcı ağın çıktısını ifade ederken,  $F_{RealX}$  değeri bu görüntüye ait özellik listesini ifade etmektedir.



Şekil 5.2 : DiscoGAN ayırıcı ağların şeması.

## 5.2.3 Maliyet fonksiyonu

DiscoGAN mimarisinin üretici ağların maliyet fonksiyonu yeniden yapılandırma (reconstructed), tartışmalı (adversarial) ve özellik (feature) kayıpları üzerine inşa edilmiştir. Mimarideki ayırıcı ağların maliyeti geleneksel ÇÜA mimarilerindeki tartışmalı kayıp değeri üzerine kurulmuştur.

### 5.2.3.1 Üretici ağ maliyeti

Bu mimaride yeniden yapılandırma kaybı için  $L_2$  metriği kullanılır. Bu işlem giriş görüntüleri  $Real\_X$  ve  $Real\_Y$  ile yeniden oluşturulmuş  $Recons\_X$  ve  $Recons\_Y$  görüntüleri arasında hesaplanır. Formülasyonu Denklem 5.2 gibi ifade edilir.

$$\mathcal{L}_{const_X} = L_2(Real\_X, Recons\_X) \quad (5.2)$$

Özellik maliyeti hesaplamasında hinge embedding hata fonksiyonu kullanılır. Bu işlem ayırıcı ağdan elde edilen gerçek ( $F\_RealX$ ) ve sahte özellik ( $F\_FakeX$ ) listesi arasında hesaplanır. Denklem 5.3'te gösterilmiştir. Denklemdaki  $\lambda_1$  değeri 0.9 alınarak hesaplanır.

$$\mathcal{L}_{feature_X} = \lambda_1 * L_{HE}(F\_RealX, F\_FakeX) \quad (5.3)$$

Tartışmalı maliyet ayırıcı ağın aktivasyon fonksiyonundan geçirilmiş sahte çıktı ( $D\_FakeX$ ) sonucu üzerinde hesaplanır. Bu işlem Denklem 5.4'te gösterilmiştir. Denklemdeki  $\lambda_2$  değeri 0.1 alınarak hesaplanır.

$$\mathcal{L}_{adv_X} = \lambda_2 * \underset{G_{X2Y}}{arg\ min}(\log(1 - D\_FakeX)) \quad (5.4)$$

Ağın X alanı için gerekli maliyet fonksiyonu Denklem 5.5 gibi ifade edilir. Denklemdaki başlama oranı başlangıçta  $\mathcal{r} = 0.01$  olarak verilir. Bu değer 10000 iterasyon eğitim işleminden sonra  $\mathcal{r} = 0.5$  olarak değiştirilir.

$$\mathcal{L}_{G_{X2Y}} = \mathcal{r} * \mathcal{L}_{const_X} + (1 - \mathcal{r}) * (\mathcal{L}_{feature_X} + \mathcal{L}_{adv_X}) \quad (5.5)$$

Üretici ağların genel maliyeti her iki alan için hesaplanıp toplanılır. Toplam maliyet fonksiyonu Denklem 5.6 gibi yazılır.

$$\mathcal{L}_G = \mathcal{L}_{G_{X2Y}} + \mathcal{L}_{G_{Y2X}} \quad (5.6)$$

### 5.2.3.2 Ayırıcı ağ maliyeti

Ayırıcı ağların X alanı için maliyeti gerçek veriye ait  $D\_RealX$  ve dönüştürmek istediğimiz sahte veriye ait  $D\_FakeX$  blokları arasında hesaplanır. Bu işlem Denklem 5.7'de gösterilmektedir.

$$\mathcal{L}_{adv_{D_X}} = \underset{D_X}{arg\ max}(\log(D\_RealX) - \log(1 - D\_FakeX)) \quad (5.7)$$

İki ayırıcı ağın toplam maliyeti Denklem 5.8 gibi yazılır.

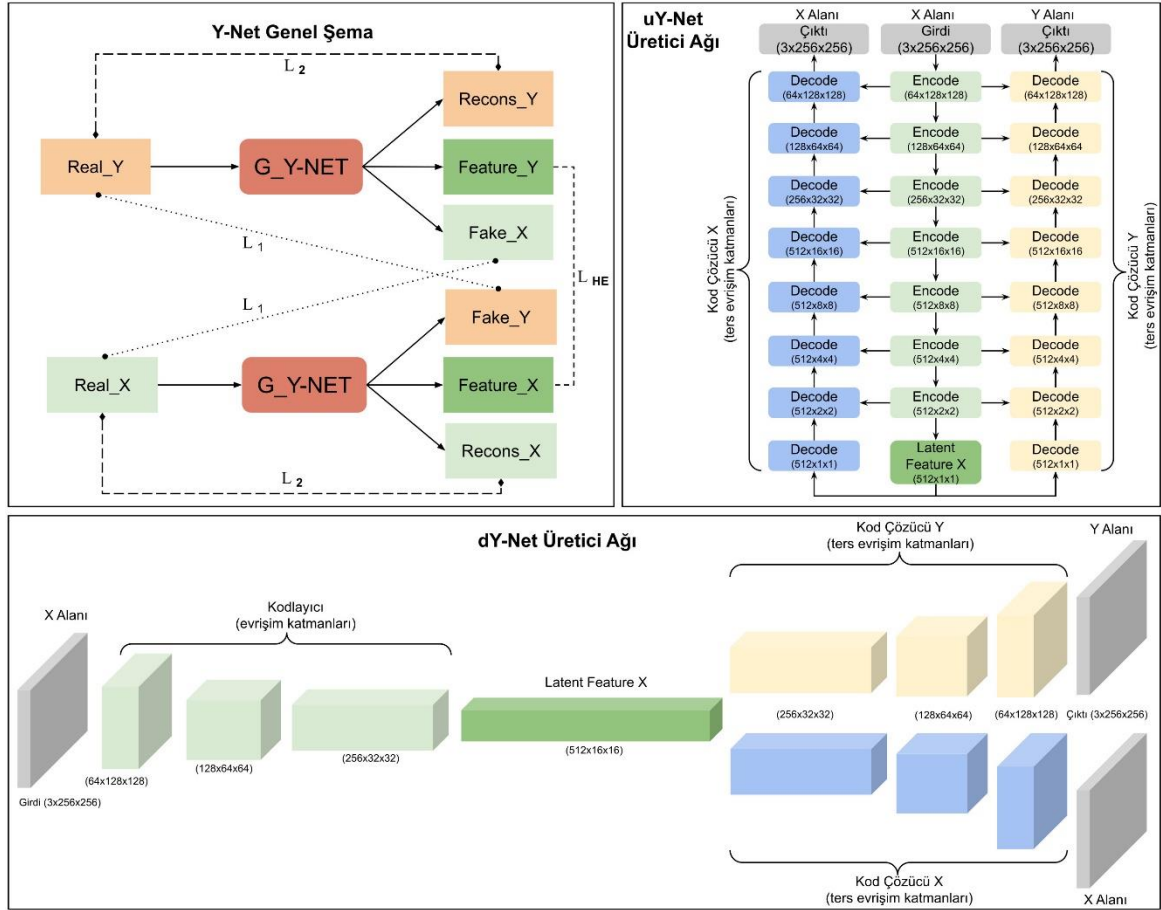
$$\mathcal{L}_D = \mathcal{L}_{adv_{D,X}} + \mathcal{L}_{adv_{D,Y}} \quad (5.8)$$

### 5.3 Y-Net: Önerilen Mimari

ÇÜA mimarilerinde bir görüntü alanından diğer alana geçiş yapmak istediğimizde, otomatik kodlayıcıların aksine, üretici ağımızın kodlayıcı (encoder) ve kod çözücü (decoder) kısımları birlikte değerlendirilmektedir. Bunun dezavantajı asıl hatanın kodlayıcıdan veya kod çözücüden olduğu anlaşılammamaktadır. Bu çalışmadan önerilen Y-Net mimarisi ile bir görüntü alanından diğer görüntü alanına geçiş yapılırken iki alandaki görüntülerin özniteliklerini ortak paydaya indirgemeye çalışarak bu problemi anlamamızı sağlar. Aynı zamanda iki alana ait verilerin öznitelik çıkartımını tek bir ağa devretmekteyiz. Bunun avantajı mimarinin daha hafif ve hızlı olmasını sağlarken, ek olarak modele alanlar arasındaki geçişin nasıl yapılacağına göstermektedir.

#### 5.3.1 Üretici ağın iyileştirilmesi

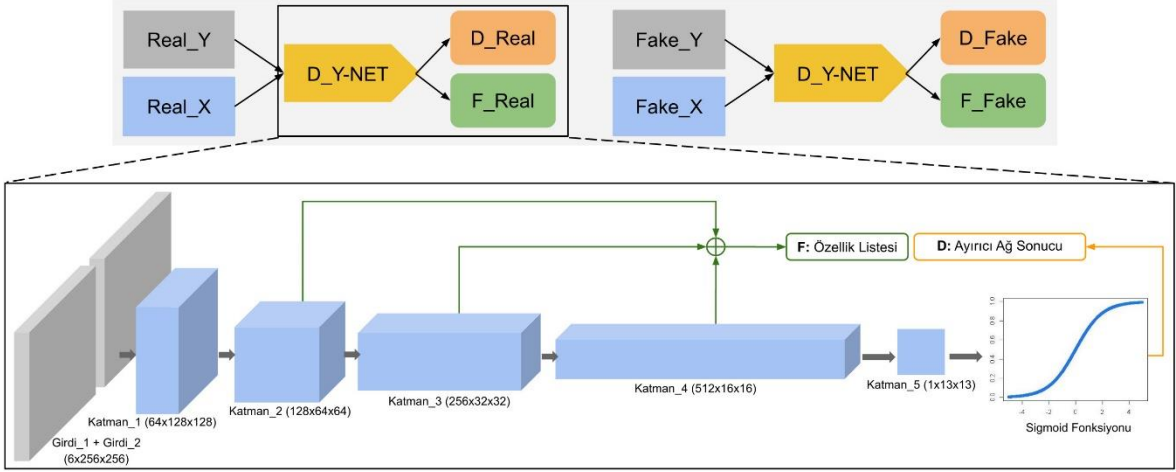
Y-Net mimarisi bir kodlayıcı ve iki kod çözücü parçalı bir üretici ağa sahiptir. Kodlayıcı parça X ve Y alanlarına ait verilerin özniteliklerini öğrenmeye çalışır. Kod çözücü parçalardan biri X alanına ait özniteliklerden yeniden bu alana ait veriyi oluşturmaya çalışır, diğeri aynı işlemi Y alanı için yapar. Bu mimari iki ayrı dönüşüm işlevini öğrenir, ilk dönüşüm X alanından Y alanına, ikinci dönüşüm Y alanından X alanına geçiştir. İlk dönüşümde üç çıktı elde edilir, sırasıyla *Fake\_Y*, *Feature\_X*, *Recons\_X* verileridir. İlk çıktı X alanındaki görüntünün Y alanına dönüşmüş imajıdır. İkinci çıktı, X alanına ait verinin kodlayıcı ağdan geçtikten sonraki öznitelik haritasıdır. Üçüncü çıktı, X alanındaki verinin yeniden oluşturulmuş görüntüsüdür, bu görüntü ile çift taraflı doğrulama işlemi yapılmaktadır. Bu dönüşüm işleminin aynısı Y alanı için de yapılır. Bu çalışmada önerilen Y-Net mimarisinin iki farklı kullanımı bulunmaktadır: 1) DiscoGAN üretici ağ tabanlı dY-Net; 2) U-Net üretici ağ tabanlı uY-Net. dY-Net mimarisi zamansal ve hiper parametre optimizasyonuna odaklanırken, uY-Net mimarisi ise görüntü kalitesine odaklanmaktadır. Y-Net mimarisinin iki farklı tasarım prototipi **Şekil 5.3**'te gösterilmiştir.



Şekil 5.3 : Y-Net mimarisinin üretici ağ şeması.

### 5.3.2 Ayırıcı ağı iyileştirilmesi

Bir üretici ağına sahip Y-Net mimarisi geleneksel ÇÜA mimarilerinin aksine gerçek görüntüleri ( $Real_X$ ,  $Real_Y$ ) ve sahte görüntüleri ( $Fake_X$ ,  $Fake_Y$ ) birlikte giriş verisi olarak almaktadır. Bu işlem üretici ağ ile uyumlu olması içindir, çünkü üretici ağ iki alana ait öznelikleri aynı paydaya indirgemeye çalışırken aynı şekilde ayırıcı ağda bu alanlara ait görüntüleri aynı paydaya indirgemeye çalışmaktadır. Böylece iki alan arasındaki dönüşüm işleminin dengeli olması sağlanmaktadır. Ayırıcı ağın genel şeması Şekil 5.4'te verilmektedir. Bu şemanın ilk aşamasında gerçek görüntülere ait ayırıcı ağın çıktısı ( $D\_Real$ ) ve özellik listesi ( $F\_Real$ ), ikinci aşamada ise sentezlenen sahte görüntülere ait  $D\_Fake$  ve  $F\_Fake$  verileri gösterilmektedir. Y-Net ayırıcı ağ mimarisi DiscoGAN tabanlı olarak geliştirilmiştir.



Şekil 5.4 : Y-Net mimarisinin ayırıcı ağ şeması.

### 5.3.3 Maliyet fonksiyonu

Y-Net mimarisinin üretici ağ maliyet fonksiyonu yeniden yapılandırma (reconstructed), benzerlik, tartışmalı (adversarial) ve özellik (feature) kayıpları üzerine inşa edilmiştir. Mimarideki ayırıcı ağın maliyeti geleneksel tartışmalı kayıp değeri üzerine kurulmuştur. Tartışmalı maliyet olarak min-max algoritması yerine hinge maliyet tekniği [49] kullanılmıştır.

#### 5.3.3.1 Üretici ağ maliyeti

Bu mimaride yeniden yapılandırma kaybı için  $L_2$  metriği kullanılarak hesaplanır. Bu işlem giriş görüntüleri  $Real_X$  ve  $Real_Y$  ile yeniden oluşturulmuş  $Recons_X$  ve  $Recons_Y$  görüntüleri arasında hesaplanır.

Benzerlik kaybı  $L_1$  metriği kullanılarak hesaplanır. Bu işlem giriş görüntüleri  $Real_X$  ve  $Real_Y$  ile bu görüntülerin karşı alana dönüştürülmüş imajları olan  $Fake_X$  ve  $Fake_Y$  görüntüleri arasında hesaplanır. Denklem 5.9'da gösterilmektedir.

$$\mathcal{L}_{similarity_X} = L_1(Real_X, Fake_X) \quad (5.9)$$

Özellik maliyeti hesaplamasında hinge gömülü hata fonksiyonu kullanılır. Bu işlem ayırıcı ağdan elde edilen gerçek ( $F_{Real}$ ) ve sahte özellik ( $F_{Fake}$ ) listesinin sonucu ile üretici ağdan gelen X alanına ait  $Feature_X$  ve Y alanına ait  $Feature_Y$  özellik haritalarının sonucunun toplanmasıdır. Bu işlem Denklem 5.10'da gösterilmiştir. Denklemdeki  $\lambda_1$  değeri 0.9 alınarak hesaplanır.

$$\mathcal{L}_{feature} = \lambda_1 * (L_{HE}(F\_Real, F\_Fake) + L_{HE}(Feature\_X, Feature\_Y)) \quad (5.10)$$

Tartışmalı maliyet ayırıcı ağın aktivasyon fonksiyonundan geçirilmiş sahte çıktı ( $D\_Fake$ ) sonucu üzerinde hesaplanır. Bu işlem Denklem 5.11 gibi ifade edilir. Denklemdaki  $\lambda_2$  değeri 0.1 alınarak hesaplanır.

$$\mathcal{L}_{adv} = \lambda_2 * mean(-D\_Fake) \quad (5.11)$$

Üretici ağı genel maliyeti Denklem 5.12 gibi ifade edilir. Denklemdaki başlama oranı başlangıçta  $r = 0.01$  olarak verilir. Bu değer 10000 iterasyon eğitim işleminden sonra  $r = 0.5$  olarak değiştirilir.

$$\mathcal{L}_G = r * (\mathcal{L}_{const_X} + \mathcal{L}_{const_Y} + \mathcal{L}_{similarity_X} + \mathcal{L}_{similarity_Y}) + (1 - r) * (\mathcal{L}_{feature} + \mathcal{L}_{adv}) \quad (5.12)$$

### 5.3.3.2 Ayırıcı ağ maliyeti

Ayırıcı ağı maliyeti gerçek veriye ait  $D\_Real$  ve dönüştürmek istediğimiz sahte veriye ait  $D\_Fake$  blokları arasında hesaplanır. Bu işlem Denklem 5.13'te gösterilmektedir. Ortalaması alınan veri bloğu Relu aktivasyon fonksiyonundan geçirilerek sonuç elde edilir.

$$\mathcal{L}_D = relu(mean(1 - D\_Real)) + relu(mean(1 + D\_Fake)) \quad (5.13)$$

## 5.4 Yeni Mimarinin Analizinde Kullanılan Veri Kümeleri

Mimarilerin performans karşılaştırması için Maps [6], Cityscapes [50] ve Denim2büyük veri kümeleri kullanılmıştır. Denim2büyük veri kümesi ile ilgili detaylı açıklama bölüm 3.3'te verilmiştir.

### 5.4.1 Cityscapes

Cityscapes veri kümesi, 50 farklı şehrin sokaklarından çeşitli stereo video görsellerini içermektedir [50]. Motorlu araç, yol, bina, insan gibi 30 farklı sınıfa ait anlamsal segmentasyon haritasını içermektedir. Bu çalışmanın eğitim aşamasında 2975, test aşamasında 500 adet görüntü çifti (şehir manzarası-anlamsal segmentasyon haritası) kullanılmıştır. Bu veri kümesine ait örnek görüntüler Şekil 5.5'te gösterilmektedir.

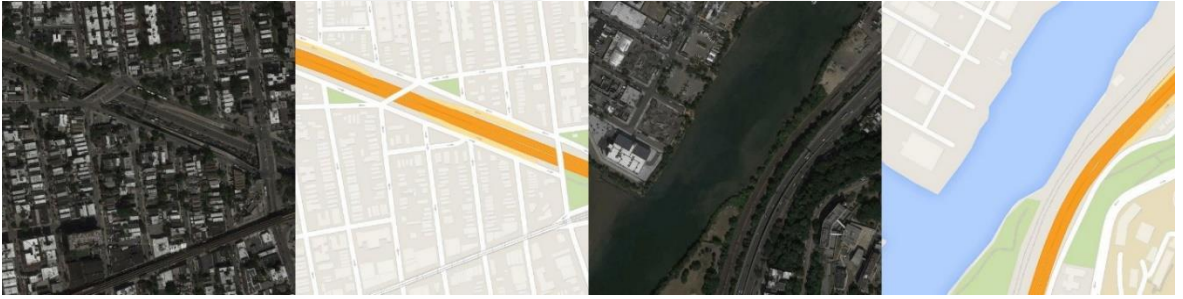




Şekil 5.5 : Cityscapes veri kümesinden örnek görüntüler.

#### 5.4.2 Maps

MAPS veri kümesi, New York çevresinden elde edilen uydu görüntülerinden ve bunlara karşılık gelen Google harita görüntülerinden oluşmaktadır [6]. Bu uydu görüntülerinde bina, park, yol gibi nesnelere ve bölgelere yer verilmektedir. Bu çalışmanın eğitim aşamasında 2095, test aşamasında 98 adet görüntü çifti (uydu-harita) kullanılmıştır. Bu veri kümesine ait örnek görüntüler Şekil 5.6'da gösterilmektedir.



Şekil 5.6 : Maps veri kümesinden örnek görüntüler.

#### 5.5 Uygulama Sonuçları

Görüntü sentezleme alanındaki en güncel mimarilerden biri DiscoGAN'dır. Bu bölümde önerilen Y-Net mimarisiyle DiscoGAN mimarisinin görüntü sentezlemedeki başarımları ve performansları kıyaslamalı bir şekilde sunulmaktadır. Çizelge 5.1'de mimarilerin yapıları ve özellikleri (parametre sayısı, depolama alanı ve parametre boyutu) kapsamlı bir şekilde gösterilmektedir.

İkinci aşamada mimarilerin kıyaslanması için Cityscapes, Maps ve Denim2büyük veri kümeleri üzerinde eğitim ve test işlemleri gerçekleştirilmiştir. Mimarilerin eğitim ve test işlemleri PyTorch derin öğrenme kütüphanesiyle RTX 2080 ekran kartına sahip bir sunucu üzerinde 150 iterasyon (epoch) boyunca part boyutu (batch size) 1 alınarak gerçekleştirilmiştir. DiscoGAN mimarisi üzerinde hiperparametre, iterasyon ve görüntü

boyutları gibi ayarlamalar ve düzenlemeler yapılmıştır. Böylece bu modellerin uygun koşullar altında başarımlarını performans karşılaştırması için gerekli ortam hazırlanmıştır.

Başarım performansı kriteri olarak görüntü üretim kalitesi ve eğitim sürecinin zamansal maliyeti kullanılmıştır. Sentetik görüntü üretim kalitesinin tespiti için test kümesindeki görüntü çiftleri üç farklı benzerlik ölçütü (SSIM, PSNR ve MSE) kullanılmıştır.

**Çizelge 5.1 : Mimarilerin nicel özellik karşılaştırması.**

Ağ Tipi	Özellik Adı	Model Adı		
		DiscoGAN	dY-Net	uY-Net
Üretici	Ağ sayısı	2	1	1
	Eğitilebilir Parametreler	11,022,336	8,266,752	89,283,398
	İleri/geri geçiş boyutu (MB)	164.00	127.00	135.96
	Parametre boyutu (MB)	42.04	31.54	340.59
	Tahmini toplam boyut (MB)	207.54	159.29	477.30
Ayırıcı	Ağ sayısı	2	1	1
	Eğitilebilir Parametreler	5,527,552	2,766,848	2,766,848
	İleri/geri geçiş boyutu (MB)	74.00	37.00	37.00
	Parametre boyutu (MB)	21.08	10.55	10.55
	Tahmini toplam boyut (MB)	96.60	47.56	47.56

Modellerin test veri kümelerindeki görüntü çiftleri üzerindeki başarımlarını performansları **Çizelge 5.2**'de gösterilmektedir. Bu tabloya göre dY-Net mimarisinin DiscoGAN'a göre üretici ağ yapısında yaklaşık %25 oranında, ayırıcı ağda %50 oranında daha verimli olmasına rağmen görüntü kalitesi olarak oldukça benzer görseller ürettiği görülmektedir. Bununla birlikte, görüntü kalitesine odaklanan uY-Net mimarisinin de DiscoGAN'a göre görüntü üretiminde daha başarılı sonuçlar ürettiği görülmektedir. Bu mimarilerin test veri kümelerindeki örnek çıktıları **Şekil 5.7-9**'da gösterilmektedir. Bu şekillerde her veri kümesinden üç farklı görüntü çiftinin sonuçları verilmektedir.



**Çizelge 5.2 : Mimarilerin performans karşılaştırması.**

Veri Kümesi Adı	Model Adı	Eğitim Süresi (dakika)	Metrik Adı	Alanlar arası görüntü dönüşüm yönü			
				$X \rightarrow X$	$X \rightarrow Y$	$Y \rightarrow Y$	$Y \rightarrow X$
Cityscapes	DiscoGAN	715	SSIM $\uparrow$	0.451	0.384	0.676	0.676
			PSNR $\uparrow$	18.85	15.73	19.92	18.22
			MSE $\downarrow$	943.96	2047.1	716.25	1108.1
	dY-Net	<b>475</b>	SSIM $\uparrow$	0.835	0.397	0.871	0.688
			PSNR $\uparrow$	27.04	16.32	28.01	18.88
			MSE $\downarrow$	143.61	1781.1	109.40	955.9
	uY-Net	815	SSIM $\uparrow$	0.945	0.437	0.944	0.696
			PSNR $\uparrow$	32.45	16.43	35.13	18.50
			MSE $\downarrow$	43.32	1744.7	22.21	1051.9
Maps	DiscoGAN	495	SSIM $\uparrow$	0.268	0.245	0.589	0.516
			PSNR $\uparrow$	18.02	15.80	28.15	26.48
			MSE $\downarrow$	1093.3	1809.2	129.76	205.5
	dY-Net	<b>335</b>	SSIM $\uparrow$	0.742	0.255	0.739	0.515
			PSNR $\uparrow$	23.44	16.14	33.25	26.93
			MSE $\downarrow$	320.57	1691.4	34.89	191.1
	uY-Net	635	SSIM $\uparrow$	0.968	0.252	0.763	0.541
			PSNR $\uparrow$	28.73	16.13	33.64	26.87
			MSE $\downarrow$	89.60	1676.7	31.63	204.4
Denim2bıyık	DiscoGAN	210	SSIM $\uparrow$	0.712	0.673	0.922	0.849
			PSNR $\uparrow$	19.05	19.06	25.26	19.44
			MSE $\downarrow$	956.1	1923.9	235.09	1049.9
	dY-Net	<b>145</b>	SSIM $\uparrow$	0.814	0.688	0.970	0.819
			PSNR $\uparrow$	28.30	16.53	32.42	19.18
			MSE $\downarrow$	143.8	1735.6	41.73	1166.8
	uY-Net	265	SSIM $\uparrow$	0.977	0.739	0.983	0.869
			PSNR $\uparrow$	36.85	16.69	38.08	20.96
			MSE $\downarrow$	27.46	1725.2	12.92	777.9

	Real X	Y→X	X→X	Real Y	X→Y	Y→Y
DiscoGAN						
dY-Net						
uY-Net						
DiscoGAN						
dY-Net						
uY-Net						
DiscoGAN						
dY-Net						
uY-Net						

Şekil 5.7 : Cityscapes veri kümesinden örnek test sonuçları.

	Real X	Y→X	X→X	Real Y	X→Y	Y→Y
DiscoGAN						
dY-Net						
uY-Net						
DiscoGAN						
dY-Net						
uY-Net						
DiscoGAN						
dY-Net						
uY-Net						

Şekil 5.8 : Maps veri kümesinden örnek test sonuçları.



	Real X	Y→X	X→X	Real Y	X→Y	Y→Y
DiscoGAN						
dY-Net						
uY-Net						
DiscoGAN						
dY-Net						
uY-Net						
DiscoGAN						
dY-Net						
uY-Net						

Şekil 5.9 : Denim2Bıyık veri kümesinden örnek test sonuçları.

## 6. YÜKSEK ÇÖZÜNÜRLÜKTE BİYİK DESENİ ÜRETİMİ

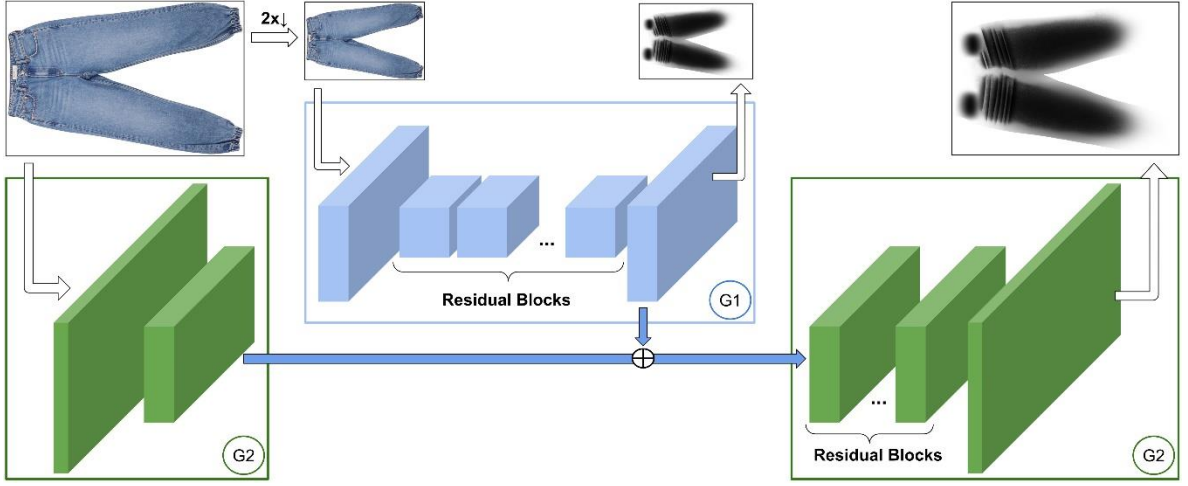
ÇÜA mimarileri görüntü üretiminde etkin bir performans sergilemesine rağmen yüksek parametre uzayına sahiptir. Bundan dolayı eğitim aşamasında yüksek işlemci gücüne ihtiyaç duymaktadır. Yüksek işlemci gücü ihtiyacı bu alanda yapılan çalışmaları sınırlandırmaktadır. Bu yüzden düşük çözünürlüklü görüntü üretimi için birçok ÇÜA mimarisi olmasına rağmen yüksek çözünürlükte görüntü üretimi yapabilen mimari sayısı çok daha azdır. Tezin bu bölümünde kaliteli ve yüksek çözünürlüklü görüntü üretimi için Pix2PixHD mimarisi kullanılmıştır.

### 6.1 Pix2PixHD Mimarisi

Anlamsal görüntü haritalarından yüksek çözünürlükte gerçekçi görüntüler üretebilmek için Pix2PixHD mimarisi 2018 yılında Wang ve diğerleri tarafından geliştirilmiştir [12]. Pix2Pix tabanlı olarak geliştirilen bu mimari 2048x1024 boyutunda görüntü üretimi yapabilmektedir.

#### 6.1.1 Üretici ağ

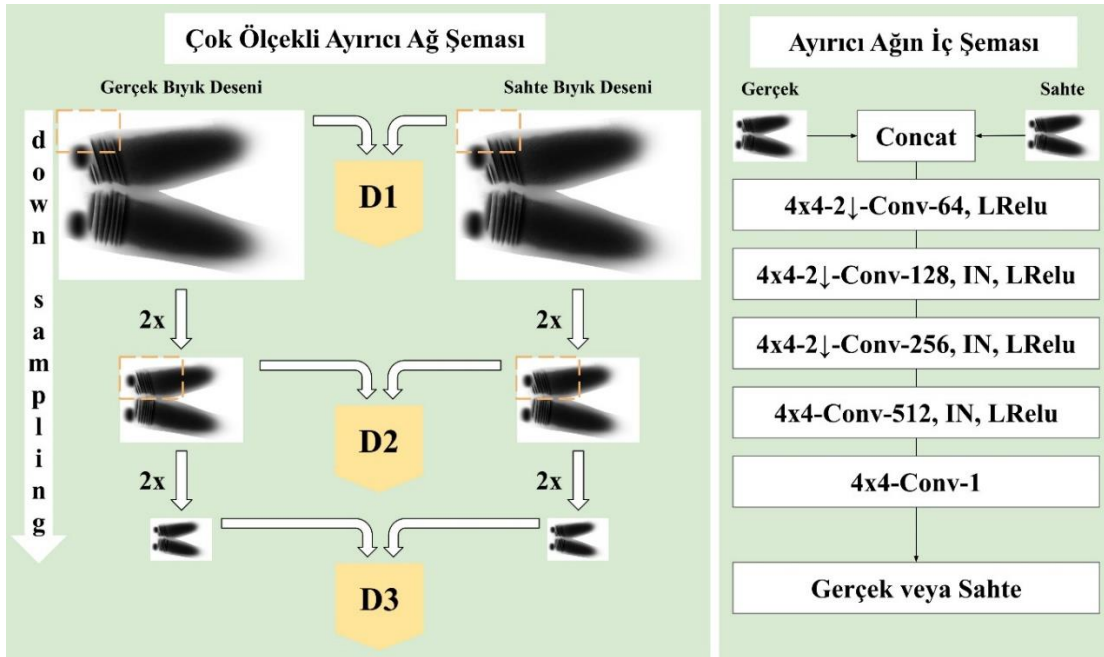
Pix2pixHD mimarisi 2 üretici ağa (G1, G2) sahiptir. Bu ağlar birbiriyle etkileşimli olarak çalışmaktadır [12]. İlki küresel ağdır (Global network = G1). Bu ağın temel görevi 2x küçültülmüş görüntünün X alanından Y alanına (Denim kumaşı → Biyık deseni) dönüşümünü sağlamaktır. Ağın çalıştığı görüntü boyutu 1024x512'dir. Görüntü kodlayıcı bloklardan işlendikten sonra 9 ayrı blokta (residual blocks) geçerek kod çözücü bloklara aktarılır [12]. İkincisi yerel güçlendirici (Local enhancer = G2) ağdır. Bu ağın temel görevi yüksek çözünürlüklü (2048x1024) görüntü üretimidir. Ağ görüntüleri kodlayıcı bloklardan geçirdikten sonra küresel ağın çıktısı ile birleştirir. Ardından bu çıktılarının birleşimini 3 ayrı blokta (residual blocks) işleyerek kod çözücü bloklara aktarır ve böylece yüksek çözünürlüklü görüntü üretimi sağlanmış olur. Yerel güçlendirici ağ ile eğitim yapılabilmesi için öncelikle küresel ağın eğitilmesi lazımdır. Bu ağların temel şeması **Şekil 6.1**'de gösterilmektedir.



Şekil 6.1 : Pix2PixHD mimarisinin üretici ağ şeması.

### 6.1.2 Ayırıcı ağ

Yüksek çözünürlüklü görüntü üretimindeki temel problemlerden biride ağın görüntüdeki detayları atlayabilmesidir. Ağın detaylara odaklanabilmesi için görüntüyü farklı ölçeklerde inceleyebilmesi gerekir. Bu yüzden Pix2PixHD mimarisinin ayırıcı ağı çok ölçekli PatchGAN mimarisi kullanmaktadır [12]. Ayırıcı ağ gerçek ve üretici ağ tarafından üretilen sahte görüntüyü 3 farklı ölçekte ( $D1$ ,  $D2$  ve  $D3$ ) değerlendirmektedir. Herbir ölçekte görüntülerin 2x küçültülmüş formunun bir parçası ayırıcı ağa verilmektedir. Böylece ayırıcı ağ görüntünün hem genel hatlarına hemde iç hatlarına odaklanabilmektedir. Ayırıcı ağın temel çalışma mekanizması Şekil 6.2’de verilmektedir.



Şekil 6.2 : Pix2PixHD mimarisinin ayırıcı ağ şeması.

### 6.1.3 Maliyet fonksiyonu

Pix2PixHD mimarisinin üretici ağ maliyeti 3 ayrı kayıptan oluşur. Bunlar tartışmalı (adversarial), özellik eşleştirme (feature matching) ve algısal (perceptual) kayıplardır [12]. Algısal kayıpta Vgg ağı kullanılmaktadır [12]. Ayırıcı ağın maliyet fonksiyonu tartışmalı kayıp tabanlıdır.

#### 6.1.3.1 Üretici ağ maliyeti

Tartışmalı kayıp, üretici ağ tarafından üretilen sahte görüntünün ( $\hat{x}$ ) ayırıcı ağa verilip elde edilen çıktısı ile aynı boyuttaki 1'lerden oluşan vektör arasında hesaplanır. Bu işlem birler vektörü ile sahte görüntü arasındaki piksel farkının karesi alınarak hesaplanır. Bu hesaplamaların formülü Denklem 6.1'de verilmiştir.

$$\mathcal{L}_{adv} = (1 - D(\hat{x}))^2 \quad (6.1)$$

Özellik eşleştirme kaybı, üretici ağdan gelen sahte görüntü ( $\hat{x}$ ) ile gerçek görüntünün ( $x$ ) ayırıcı ağa verilerek elde edilen özellik haritalarının  $L_1$  metriği ile karşılaştırmasıyla hesaplanır. Bu işlem Denklem 6.2'de gösterilmektedir. Denklemdaki  $T$  değeri ayırıcı ağdaki özellik katmanlarının toplam sayısını vermektedir.

$$\mathcal{L}_{FM} = \sum_{i=1}^T L_1(D(x_i) - D(\hat{x}_i)) \quad (6.2)$$

Algısal kayıp, üretici ağdan gelen sahte görüntü ( $\hat{x}$ ) ile gerçek görüntünün ( $x$ ) özellik çıkartıcı ağa (Vgg) verilerek elde edilen özellik haritalarının  $L_1$  metriği ile karşılaştırmasıyla hesaplanır. Bu işlem Denklem 6.3'de gösterilmektedir. Denklemdaki  $T$  değeri özellik çıkartıcı ağdaki kullanılan katmanların toplam sayısını vermektedir.

$$\mathcal{L}_{perceptual} = \sum_{i=1}^T L_1(Vgg(x_i) - Vgg(\hat{x}_i)) \quad (6.3)$$

Üretici ağın toplam kayıp fonksiyonu Denklem 6.4 gibi tanımlanır. Denklemdaki  $\lambda$  değerleri orijinal çalışmada 10 olarak alınmıştır.

$$\mathcal{L}_G = \lambda_1 * \mathcal{L}_{adv} + \lambda_2 * \mathcal{L}_{FM} + \lambda_3 * \mathcal{L}_{perceptual} \quad (6.4)$$

### 6.1.3.2 Ayırıcı ağ maliyeti

Ayırıcı ağın maliyeti tartışılmalı kayıp üzerine kurulmuştur. Bu işlemde birler vektörü ile gerçek ( $x$ ) görüntü arasındaki piksel farkının karesi alınır. Ardından sıfırlar vektörü ile sahte görüntü ( $\hat{x}$ ) arasındaki piksel farkının karesi alınır. Bu iki işlem sonucunun toplamı ayırıcı ağın maliyetini vermektedir. İşlemin formülü Denklem 6.5'te gösterilmektedir.

$$\mathcal{L}_D = (1 - D(x))^2 + (0 - D(\hat{x}))^2 \quad (6.5)$$

### 6.2 Denim2BıyıkHD: Yüksek Çözünürlüklü Denim2Bıyık Veri Kümesi

Denim2Bıyık veri kümesinin ilk versiyonunda 976 görüntü çifti bulunmaktadır (Bakınız bölüm 2.3). Bu veri kümesi belirli bir standartta olmadığı için görüntü üretiminde yeterli kalitede başarımlar sağlanılamadı. Veri kümesinin üzerinde çeşitli düzenlemeler ile 950 görüntü çiftine indirgenerek belirli bir standartlaştırma seviyesine çıkartılmıştır (Bakınız bölüm 3.3). Bu görüntülerin 256x256x3 ebatlarında olmasından dolayı yüksek çözünürlüklü görüntü üretiminde kullanılması uygun değildir. Yüksek çözünürlüklü bıyık desen üretiminde kullanılacak veri kümesi için Baykan Denim firmasının İstanbuldaki üretim fabrikasından 195 denim kumaş görüntüsü elde edilmiştir. Bu görüntüler aynı koşullar (çözünürlük, ışık dağılımı, çekim ortamı, kamera, v.b) altında çekimlenmiştir. Görüntülerdeki bıyık desen görselleri firmanın Malatya lokasyonundaki üretim fabrikasında bir uzman personel tarafından PhotoShop programı ile hazırlanmıştır. Nihai durumda 195 denim-bıyık görüntü çifti elde edilmiştir. Bu görsellerin ÇÜA mimarileri işlenebilmesi için PhotoShop ve OpenCV programları ile görüntüler üzerinde çeşitli düzenlemeler (döndürme, arkaplan rengi, boyut, bölgesel eşleşme, renk ayarları, v.b) yapılmıştır. Veri kümesinin yeni formunda 512p (1024x512x3) ve 1024p (2048x1024x3) versiyonları mevcuttur. Veri kümesinden örnek birkaç denim kumaşı ve bıyık deseni çifti **Şekil 6.3**'te gösterilmektedir.





Şekil 6.3 : Denim2BıyıkHD veri kümesinden örnek görüntüler.

### 6.3 Uygulama Sonuçları

Denim2BıyıkHD veri kümesinin eğitim işlemi RTX 2080 Titan ekran kartına sahip bir sunucuda gerçekleştirilmiştir. 100 iterasyon gerçekleşen eğitim işlemi yaklaşık olarak 10 saat sürmüştür. Eğitim ve test işlemleri için Python programlama dili ve PyTorch kütüphanesi kullanılmıştır. Veri kümesi %90 eğitim ve %10 test olarak bölümlenmiştir.

Gerçek ve üretilen görüntü arasındaki benzerliği tam değerlendirebilmek için SSIM metriği ile görüntülerin yapısal benzerliği, PSNR yöntemi ile görüntü piksellerindeki maksimum sinyal gücü ve MSE yöntemi ile iki görüntü arasındaki piksel farklılık maliyeti hesaplanmıştır. Mimarilerin eğitim ve test işlemlerindeki başarımları **Çizelge 6.1**'de gösterilmektedir.

**Çizelge 6.1** : Denim2BıyıkHD veri kümesinin eğitim ve test aşamasında Pix2PixHD mimarisi ile elde edilen başarımların sonuçları.

Model Adı	Aşama	Metrik Adı (Benzerlik)		
		SSIM ↑	PSNR ↑	MSE ↓
Pix2PixHD	Eğitim	0.99	39.15	33.888
	Test	0.89	17.59	1792.1

Pix2PixHD mimarisinin eğitim aşamasında yüksek çözünürlüklü bıyık desen görsellerini 99% oranında başarıyla ürettiği gözlemlenmiştir. Ancak eğitim ile test aşamalarındaki çıktılar henüz aynı kalitede üretilmediği görülmektedir. Bu problem, Denim2BıyıkHD veri kümesindeki örnek sayısının yetersiz olmasından kaynaklanmaktadır. Veri kümesindeki örnek sayısı artırılırsa test aşamasındaki görüntü üretim kalitesinde artacaktır. Eğitim tamamlandıktan sonra görsel test sonuçları elde edilmiştir. Denim2BıyıkHD veri kümesinin eğitim ve test kümelerindeki örneklerin bıyık deseni üretim sonuçları **Şekil 6.4-5**'te gösterilmektedir. İlki eğitim ikincisi test kümesindeki üretim sonucudur. Şekillerde sırasıyla ilk sütun: denim kumaşı, ikinci sütun: gerçek bıyık deseni, üçüncü sütun: üretim sonucunu göstermektedir.



**Şekil 6.4 :** Denim2BıyıkHD veri kümesinin eğitim setindeki üretim sonuçları (Denim kumaşı, Gerçek bıyık deseni, Pix2PixHD üretim sonucu).



**Şekil 6.5 :** Denim2BıyıkHD veri kümesinin test setindeki üretim sonuçları (Denim kumaşı, Gerçek bıyık deseni, Pix2PixHD üretim sonucu).

## 7. SONUÇ VE ÖNERİLER

Tezin ikinci bölümünde, denim kumaş görselleri üzerine bulunan bıyık desenlerinin otomatik üretilmesi hakkında bir ön çalışma yapılmıştır. Böylece uzman personellerin denim kumaşlarına bakarak yaptıkları manuel çizimlere (ortalama 2-3 saat) gerek kalmadan hızlı ve doğru bir şekilde bıyık görüntüleri üretilbileceğini göstermektedir. Bu çalışmanın 2 ana katkısı bulunmaktadır. Birincisi otomatik üretim için denim kumaş görselleri ve bıyık desen modellerini içeren bir veri kümesi inşa edilmiştir. Diğeri veri kümesi Pix2Pix olarak bilinen bir ÇÜA mimarisinin eğitiminde kullanılmıştır. Ek olarak yapılan deneysel çalışmalar neticesinde klasik mimarinin aşırı öğrenme sağladığı ve test verileri üzerinde istenmeyen gürültülü sonuçlar ürettiği gözlemlenmiştir. Mimari üzerinde yapılan güncelleme sonucunda aşırı öğrenmenin ortadan kaldırıldığı ve gürültü etkisinin temizlendiği görülmüştür.

Tezin üçüncü bölümünde, görüntüden görüntüye dönüşüm yapan çekişmeli üretici ağ mimarilerinin performans karşılaştırması yapılmıştır. Denim2Bıyık veri kümesi ön işlemlerden geçirilerek daha kararlı hale getirilmiştir. Veri kümesinin yeni formu 950 adet denim-bıyık desen çiftini içermektedir. Bu çalışma ile bıyık desen üretimini otomatikleştiren, üretim süresini yaklaşık 1 saniyeye düşüren, personel maliyetlerini azaltan ve farklı kalitede bıyık motiflerinin çıkmasını engelleyen otonom bir sisteme doğru önemli bir adım atılmıştır. Literatürde üretken ağ mimarilerinin denim ürünlerine uygulamasını içeren farklı bir çalışmanın olmadığı görülmektedir. Bu durum yapılan çalışmanın akademik özgün değerini yükseltmektedir. Yapılan deneysel çalışmalarda Pix2Pix mimarisinin (yaklaşık bir saniyede 86% oranında) zamansal ve doğruluk olarak en iyi sonuçları ürettiği görülmüştür.

Tezin dördüncü bölümünün ana konusu görüntü sentezleme probleminde yaygın olarak kullanılan ve bir çekişmeli üretken ağ olan Pix2Pix mimarisinin farklı maliyet metrikleri ile performansını değerlendirmektir. Bunun için Pix2Pix'in orijinal  $L_1$  maliyeti yerine ESA tabanlı maliyet fonksiyonları (CONTENT, DISTS ve LPIPS) kullanılmıştır. Maliyet fonksiyonundaki çekişmeli maliyet terimi ve düzgünselleştirme (regülerizasyon) terimlerine ESA tabanlı yapıların eklenmesinin etkileri analiz edilmiştir. Yapılan eğitim ve test faaliyetleri sonucunda sentezleme doğrulukları klasik ve güncel ölçütlerle tablolara aktarılmıştır. Elde edilen deneysel sonuçlara incelendiğinde ESA tabanlı bu ölçütlerin ÇÜA mimarileri ile görüntü sentezlemede kullanılabileceğini göstermektedir.

Tezin beşinci bölümünde görüntü sentezleme alanında kullanılmak üzere dY-Net ve uY-Net isimlerinde iki farklı mimari önerilmektedir. dY-Net mimarisi düşük parametre uzayına sahiptir ve üretim hızındaki performansa odaklanır. uY-Net mimarisi görüntü üretim kalitesine odaklanır. Önerilen mimarilerin mevcut benzerleriyle olan karşılaştırmaları üretim zamanı ve görüntü kalitesi şeklindedir. Y-Net mimarilerinin temelinde giriş ve çıkış görsellerinin öznitelik uzaylarını eşitleme ilkesi yer almaktadır. Böylece parametrelerin ortak kullanımı söz konusu olmakta ve hiper parametre sayısında ciddi bir azalma gerçekleşmektedir. Önerilen mimarilerin üretim performansı ve başarımlarının literatürdeki konumunu öğrenmek için DiscoGAN mimarisiyle karşılaştırmalar yapılmıştır. Çift yönlü eğitim yapısıyla DiscoGAN mimarisi, görüntü sentezleme alanında yaygın bir şekilde kullanılmaktadır. Elde edilen karşılaştırma sonuçlarına göre, dY-Net mimarisinin parametre sayısı DiscoGAN'nın %66'sı kadar olmasına rağmen benzer kalitede görüntü üretebilmektedir. uY-Net mimarisinin diğer mimarilerden daha kaliteli görüntü sentezlediği görülmüştür. Doğrulama faaliyetlerinde üç farklı veri kümesi (Maps, Cityscapes ve Denim2Bıyık) kullanılmıştır. Bu sonuçlar, sentezleme faaliyetlerinde performansı veya görüntü kalitesini önceleyen uygulamalar için iki farklı mimarinin kullanılabileceğini de ortaya koymaktadır.

Tezin altıncı bölümünde, yüksek çözünürlükte görüntü üretimi yapan Pix2PixHD mimarisi ile bıyık desen üretimi gerçekleştirilmiştir. Denim2Bıyık veri kümesinin yüksek çözünürlükteki versiyonu olan Denim2BıyıkHD veri kümesi hazırlanmıştır. Veri kümesinin yeni formu 195 adet yüksek çözünürlüklü standartlaştırılmış denim-bıyık desen çiftini içermektedir. Bu çalışma ile yüksek kalitede bıyık desen üretimi için önemli bir adım atılmıştır. Hem zamansal maliyet olarak 1 saniyenin altında üretim hızı sağlanmış hemde yüksek kalitede bıyık desen üretimi (yaklaşık 90 oranında) gerçekleştirilmiştir.

Yapılan çalışmalar neticesinde, denim-bıyık dönüşüm probleminin üretken ağ temelli mimariler kullanılarak başarılı bir şekilde çözülebildiği, önerilen özgün üretken ağ mimarilerin eşleştirme doğrulukları göz önüne alındığında ümit verici çıktılar elde edilebildiği görülmektedir. Önümüzdeki yıllarda endüstriyel problemlerin çözümlerinde yapay öğrenme yaklaşımlarının artan bir ivmeyle kullanılacağı öngörülmektedir.



## KAYNAKLAR

- [1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative Adversarial Nets. *Advances in neural information processing systems*, 27.
- [2] Huang, H., Yu, P. S., & Wang, C. (2018). An Introduction to Image Synthesis with Generative Adversarial Nets. *arXiv preprint arXiv:1803.04469*.
- [3] Goodfellow, I. (2016). Nips 2016 tutorial: Generative Adversarial Networks. *arXiv preprint arXiv:1701.00160*.
- [4] Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.
- [5] Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., ... & Shi, W. (2016). Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4681-4690).
- [6] Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1125-1134).
- [7] Zhu, J. Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 2223-2232).
- [8] Karras, T., Aila, T., Laine, S., & Lehtinen, J. (2017). Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.
- [9] Karras, T., Laine, S., & Aila, T. (2018). A style-based generator architecture for generative adversarial networks. *arXiv preprint arXiv:1812.04948*.
- [10] Zhang, H., Goodfellow, I., Metaxas, D., & Odena, A. (2018). Self-attention generative adversarial networks. In *International conference on machine learning* (pp. 7354-7363). PMLR.
- [11] Brock, A., Donahue, J., & Simonyan, K. (2018). Large scale GAN training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*.
- [12] Wang, T. C., Liu, M. Y., Zhu, J. Y., Tao, A., Kautz, J., & Catanzaro, B. (2018). High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 8798-8807).
- [13] Park, T., Liu, M. Y., Wang, T. C., & Zhu, J. Y. (2019). Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 2337-2346).
- [14] Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., & Aila, T. (2019). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 8110-8119).
- [15] Donahue, J., & Simonyan, K. (2019). Large scale adversarial representation learning. *arXiv preprint arXiv:1907.02544*.

- [16] Li, Y., Singh, K. K., Ojha, U., & Lee, Y. J. (2020). Mixnmatch: Multifactor disentanglement and encoding for conditional image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 8039-8048).
- [17] Dundar, A., Sapra, K., Liu, G., Tao, A., & Catanzaro, B. (2020). Panoptic-based image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 8070-8079).
- [18] Wang, X., Li, Y., Zhang, H., & Shan, Y. (2021). Towards Real-World Blind Face Restoration with Generative Facial Prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9168-9178).
- [19] Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., & Aila, T. (2021). Alias-free generative adversarial networks. *arXiv preprint arXiv:2106.12423*.
- [20] Wang, X., Xie, L., Dong, C., & Shan, Y. (2021). Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 1905-1914).
- [21] Suvorov, R., Logacheva, E., Mashikhin, A., Remizova, A., Ashukha, A., Silvestrov, A., ... & Lempitsky, V. (2021). Resolution-robust Large Mask Inpainting with Fourier Convolutions. *arXiv preprint arXiv:2109.07161*.
- [22] Zou, X., Wong, W. K., & Mo, D. (2018). Fashion Meets AI technology. In *International Conference on Artificial Intelligence on Textile and Apparel* (pp. 255-267). Springer, Cham.
- [23] Jucienė, M., Urbelis, V., Juchnevičienė, Ž., & Čepukonė, L. (2014). The Effect of Laser Technological Parameters on The Color and Structure of Denim Fabric. *Textile Research Journal*, 84(6), 662-670.
- [24] Zhong, T., Dhandapani, R., Liang, D., Wang, J., Wolcott, M. P., Van Fossen, D., & Liu, H. (2020). Nanocellulose from Recycled Indigo-dyed Denim Fabric and Its Application in Composite Films. *Carbohydrate Polymers*, 240, 116283.
- [25] Golden Laser. (2021). Jeans Laser Engraving Machine. Retrieved from: <https://www.goldenlaser.cc/jeans-laser-engraving-machine.html>, accessed: 21 June 2021.
- [26] Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.
- [27] Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, 30(1), 79-82.
- [28] Baykan Denim A.Ş. (2021) Baykan Denim, <https://www.baykandenim.com/home-page/>, accessed: 2021-11-26.
- [29] Nilsson, J., & Akenine-Möller, T. (2020). Understanding ssim. *arXiv preprint arXiv:2006.13846*.
- [30] Ravuri, S., & Vinyals, O. (2019). Classification Accuracy Score for Conditional Generative Models. *arXiv preprint arXiv:1905.10887*.
- [31] Kim T., Cha M., Kim H., Lee J. K., Kim J. 2017. Learning to discover cross-domain relations with generative adversarial networks. *34th International Conference on Machine Learning, ICML 2017*, 4, 2941–2949.



- [32] Mejjati Y A., Richardt C., Tompkin J., Cosker D. 2018. Unsupervised Attention-guided Image-to-Image Translation. *NeurIPS 2018*, 1–11.
- [33] Mihelich M., Dognin C., Shu Y., Blot M. 2020. A Characterization of Mean Squared Error for Estimator with Bagging. *ArXiv, abs/1908.02718*.
- [34] Fardo F A., Conforto V H., Oliveira F C., Rodrigues P. 2016. A Formal Evaluation of PSNR as Quality Measurement Parameter for Image Segmentation Algorithms. *ArXiv, abs/1605.07116*.
- [35] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.
- [36] Gatys, L. A., Ecker, A. S., & Bethge, M. (2015). A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*.
- [37] Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 586-595).
- [38] Ding, K., Ma, K., Wang, S., & Simoncelli, E. P. (2020). Image quality assessment: Unifying structure and texture similarity. *arXiv preprint arXiv:2004.07728*.
- [39] Zhang, L., Zhang, L., Mou, X., & Zhang, D. (2011). FSIM: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8), 2378-2386.
- [40] Reisenhofer, R., Bosse, S., Kutyniok, G., & Wiegand, T. (2018). A Haar wavelet-based perceptual similarity index for image quality assessment. *Signal Processing: Image Communication*, 61, 33-43.
- [41] Wang, Z., Simoncelli, E. P., & Bovik, A. C. (2003, November). Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003* (Vol. 2, pp. 1398-1402). Ieee.
- [42] Sheikh, H. R., & Bovik, A. C. (2006). Image information and visual quality. *IEEE Transactions on image processing*, 15(2), 430-444.
- [43] Zhang, L., Shen, Y., & Li, H. (2014). VSI: A visual saliency-induced index for perceptual image quality assessment. *IEEE Transactions on Image processing*, 23(10), 4270-4281.
- [44] 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. *Advances in neural information processing systems*, 30.
- [45] Bińkowski, M., Sutherland, D. J., Arbel, M., & Gretton, A. (2018). Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*.
- [46] Ding, K., Ma, K., Wang, S., & Simoncelli, E. P. (2021). Comparison of full-reference image quality models for optimization of image processing systems. *International Journal of Computer Vision*, 129(4), 1258-1281.
- [47] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

- [48] Bailer, C., Varanasi, K., & Stricker, D. (2017). CNN-based patch matching for optical flow with thresholded hinge embedding loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3250-3259).
- [49] Lim, J. H., & Ye, J. C. (2017). Geometric gan. *arXiv preprint arXiv:1705.02894*.
- [50] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., ... & Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3213-3223).



## ÖZGEÇMİŞ

**Ad-Soyad :** Emrullah ŞAHİN

### ÖĞRENİM DURUMU:

- **Lisans :** 2019, İnönü Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü

### MESLEKİ DENEYİM:

- 2022-Devam Dumlupınar Üniversitesi Mühendislik Fakültesi Yazılım Mühendisliği Bölümü / Kütahya.
- 2021-2022 Baykan Denim ve Konfeksiyon / Malatya.
- 2019-2021 Sadıkoğulları Teknoloji ve Yazılım / Malatya.

### YÜKSEK LİSANS VEYA DOKTORA TEZİNDEN TÜRETİLEN ÇALIŞMALAR (Makaleler, Projeler)

#### SCI

- **Şahin, E., & Talu, M.F. (2021).** Y-Net: A New Approach to Image Synthesis with GANs. Parallel Computing (Değerlendirme aşamasında)(Gönderim tarihi: 12/03/2022).
- **Şahin, E., Altun, S., & Talu, M.F. (2021).** Çekişmeli Üretken Ağların ESA Tabanlı Algısal Benzerlik Metrikleri ile Görüntü Sentezlemesi. Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi (Değerlendirme aşamasında)(Gönderim tarihi: 21/10/2021).

#### TRDİZİN

- **Şahin, E., & Talu, M.F. (2022).** Bıyık Deseni Üretiminde Çekişmeli Üretici Ağların Performans Karşılaştırması. Bitlis Eren Üniversitesi Fen Bilimleri Dergisi, 10(4), 1575-1589.

#### ULUSAL DERGİ

- **Şahin, E., & Talu, M.F. (2021).** Automatic Mustache Pattern Production on Denim Fabric with Generative Adversarial Networks. Bilgisayar Bilimleri (Yayın sürecinde)(Gönderim tarihi: 5/11/2021, Kabul tarihi: 3/12/2021).

#### ÜSİ

- Üniversite Sanayi İşbirliği (ÜSİ) kapsamında, Baykan Denim Konfeksiyon A.Ş. ve İnönü Üniversitesi Bilimsel Araştırma ve Koordinasyon Birimi (BAP) tarafından “FKP-2021-2144” nolu proje numarasıyla desteklenmektedir.