

## İnsan Yüzü Modifikasyonu için Farklı Bir GAN Modeli

Emre KARDAL<sup>1\*</sup>, Vasif NABIYEV<sup>2\*</sup>

### Öz

Günümüzde aktif olarak kullanılan Üretken Çekişmeli Ağlar (GAN'lar), makine öğrenmesi ve yapay zeka alanlarında son teknoloji yöntemlerden biridir. GAN'lar, iki sinir ağının (Üretici ve Ayırt Edici) rekabetçi bir şekilde birbirlerini eğiterek yüksek karmaşıklığındaki veri örneklerini işlemelerine ve bu sayede gerçekçi yapay görüntüler, sesler veya videolar üretmelerine olanak tanır. Genel olarak GAN algoritması kullanan modeller, rastgele gürültü örnekleri ile rastgele sonuçlar üretmektedir. Ancak, bu çalışmada geliştirilen farklı bir GAN modeli, belirli koşullara uygun olarak modifiye edilmesi istenen hedef yüzlerin gerçekçi sonuçlar oluşturmaya odaklanmaktadır. Bu modelin tasarımında, hedef yüz verileri girdi olarak kullanılarak, bu yüzlerin istenilen özelliklere göre (örneğin, sakal ekleme, kellik, vb.) modifiye edilmesi sağlanmıştır. Deneysel sonuçlar, belirli koşullar altında üretilen çıktının kayda değer başarılar elde ettiğini göstermiştir. Özellikle, geriye yönelik bir eğitim süreci olmamasına rağmen, modelin çıktısı giriş olarak tekrar kullanıldığında, eski fotoğrafın yeniden oluşturulmasında %62 başarı elde edilmiştir. Ayrıca, fotoğrafların arka planı silinerek sadece yüz için yapılan hesaplamalar sonucunda bu başarı oranı ortalama %85'e yükselmiştir. Bu çalışma, GAN modellerinin sadece rastgele gürültü ile sonuç üretmekten öte, belirli koşullara uygun gerçekçi modifikasyonlar yapabilme potansiyelini göstermektedir. Ulaşılan bu başarı oranları, özellikle güvenlik sistemleri, estetik cerrahi, film endüstrisi ve bilgisayar yaratıcılığı gibi alanlarda GAN modellerinin kullanım potansiyelini artırmaktadır.

**Anahtar Kelimeler:** Üretken Çekişmeli Ağlar, Makine Öğrenmesi, Yapay Zekâ

## Different GAN Model for Human Face Modification

### Abstract

Generative Adversarial Networks (GANs) are among the state-of-the-art methods in the fields of machine learning and artificial intelligence. GANs enable the processing of highly complex data samples and thus the generation of realistic artificial images, sounds, or videos through the competitive training of two neural networks: the Generator and the Discriminator. Typically, models using the GAN algorithm produce random results with random noise samples. However, in this study, a different GAN model was developed, focusing on generating realistic results for target faces to be modified according to certain conditions. In the design of this model, target face data were used as input, and modifications (e.g., adding a beard, balding, etc.) were made according to desired characteristics. Experimental results showed that the outputs generated under certain conditions achieved remarkable success. Notably, even without a retrospective training process, the model achieved a 62% success rate in reconstructing the old photograph when the model's output was reused as input. Additionally, when the background of the photos was removed and calculations were performed only on the face, this success rate increased to an average of 85%. This study demonstrates the potential of GAN models to perform realistic modifications according to specific conditions, beyond merely producing results with random noise. The achieved success rates highlight the potential applications of GAN models in fields such as security systems, aesthetic surgery, the film industry, and computer creativity.

**Keywords:** Generative Adversarial Networks (GANs), Machine Learning (ML), Artificial Intelligence (AI)

<sup>1,2</sup>Karadeniz Teknik Üniversitesi, Bilgisayar Mühendisliği Bölümü Mühendislik Fakültesi, Trabzon, Türkiye, emrekd1314@gmail.com vasif@ktu.edu.tr

\*Sorumlu Yazar/Corresponding Author

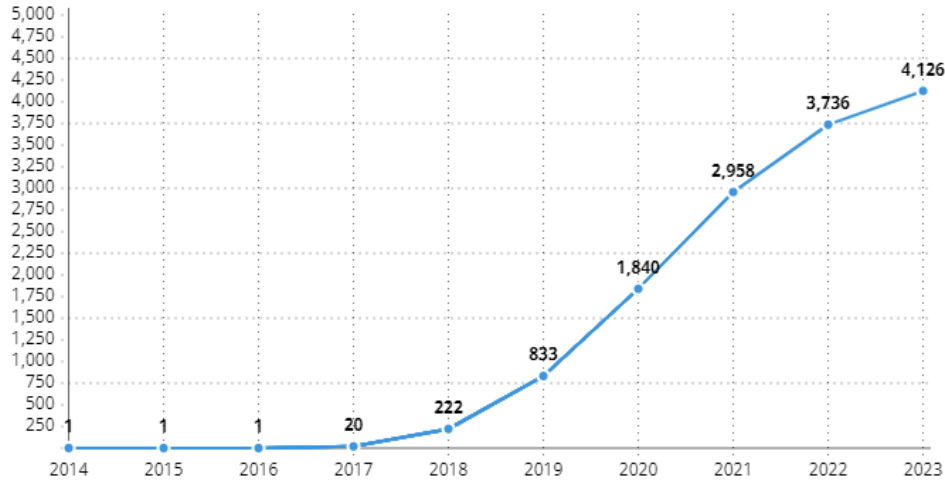
Geliş/Received: 06.04.2023

Kabul/Accepted: 06.06.2024

Yayın/Published: 18.06.2024

## 1. Giriş

Günümüzde güvenlik sistemleri, yüz estetiği, film endüstrisi, bilgisayar yaratıcılığı açısından yüz modifikasyonu önem taşımaktadır. Bu tarz problemlerin çözümü, yeni yöntemler ve akıllı algoritmalar gerektirmektedir. Yüz modifikasyonu, yüz tanıma teknolojilerinin yanı sıra, kişisel fotoğraf düzenleme uygulamalarında ve sinema endüstrisinde özel efektler için de kritik bir rol oynamaktadır. Örneğin, film yapımcıları, karakterlerin görünümünü değiştirmek için gelişmiş yüz modifikasyon tekniklerini kullanarak izleyicilere daha etkileyici deneyimler sunmaktadır. Yapay zekâ modellerinden biri olan GAN ise bu alanda kullanılabilecek etkili yöntemlerden biridir. GAN, Ian Goodfellow tarafından, yarı denetimli ve denetimsiz öğrenme yöntemi olarak önerildi (Goodfellow, I. ve ark. 2014). Ve gün geçtikçe daha popüler hale geldi (Salehi, P. ve ark. 2020). Şekil 1’de 2014 yılından 2023 yılına kadar olan makale sayısının grafiği gösterilmiştir.



Şekil 1. GAN’ın gelişimi (WOS topic: Generative Adversarial Networks)

GAN, temel olarak iki ana bileşenden oluşur: Üretici (Generatör) ve Ayırt Edici (Discriminator). Üretici, rastgele gürültüden yeni veri örnekleri üretirken, Ayırt Edici bu örneklerin gerçek mi yoksa sahte mi olduğunu belirlemeye çalışır. Bu iki ağ arasındaki rekabet, Nash dengesi olarak bilinen oyun teorisi prensibine dayanır (Goodfellow I. 2016). Ayırt Edici, üretilen rastgele sonuçlar ile veri setinden gelen rastgele örneklerden hangisinin gerçek, hangisinin sahte örnekler olduğunu çözmeye çalışır. Bu ağ sonucu 0 ile 1 arasında puanlar. Bu puan ağa gelen örnekler ne kadar gerçekse 1’e yakın, ne kadar sahteyse 0’a yakın olarak değerlendirilir. Başta üretilen sonuçlar tamamen rastgele olabilir. Eğitim devam ettikçe Ayırt Edici artık daha net bir şekilde gerçek ile sahte örnekleri ayırt etmeye başlar. Üretici ağ ise Ayırt Edicinin gerçek olduğunu sandığı örneklere yakınsamaya, sahte olduğunu anladığı örneklerden uzaklaşmaya çalışır. Bu rekabet durumu GAN’ın diğer sinir ağı modellerinden daha performanslı ve gerçekçi sonuçlar üretmesini sağlar. Bu ağların eğitim algoritması ise Algoritma 1’de görülmektedir (Goodfellow, I. ve ark. 2014).

Algoritma 1: (Goodfellow, I. ve ark. 2014)

for “eğitim sayısı” do:

for “Ayırt Ediciye uygulanacak adım sayısı” do:

$m$  adet gürültü örneği  $\{z^{(1)}, \dots, z^{(m)}\}$

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

Ayırt Ediciyi artan değişken gradyan yöntemiyle güncelle:

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^m [\log D(x^{(i)}) + \log(1 - D(G(z^{(i)})))]$$

end for

$m$  adet gürültü örneği  $\{z^{(1)}, \dots, z^{(m)}\}$

Üreticiyi azalan değişken gradyan yöntemiyle güncelle:

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^m \log(1 - D(G(z^{(i)})))$$

end for

*Güncellemelerde herhangi bir gradyan tabanlı güncelleme yöntemi kullanılabilir.* (Goodfellow, I. ve ark. 2014).

Günümüzde, GAN'lar sadece görüntü işleme alanında değil, aynı zamanda ses ve video üretimi gibi diğer medya türlerinde de geniş uygulama alanı bulmaktadır. Örneğin, sanal gerçeklik (VR) ve artırılmış gerçeklik (AR) uygulamalarında, kullanıcı deneyimini artırmak için gerçekçi görüntüler ve videolar üretmek amacıyla GAN'lar kullanılmaktadır. Bu makalede özellikle GAN'ın tercih edilme sebebi rekabetten doğan performansı ve daha gerçekçi sonuç üretmesidir. Ancak orijinal modelde sadece rastgele gürültüler ile rastgele sonuçlar üretilmektedir. Bu yüzden orijinal GAN'dan farklı olarak, bu çalışmada modifiye edilmesi istenen insan yüzlerinin koşul verileri girdi olarak kullanılmış ve böylece daha kontrollü ve hedeflenmiş sonuçlar elde edilmiştir. Bu modelin amacı, belirli koşullara uygun olarak hedef yüzlerin istenilen özelliklerde modifiye edilmesini sağlamaktır. GAN'ların insan yüzü modifikasyonu gibi karmaşık görevlerde nasıl etkili bir şekilde kullanılabileceğini göstermektir. Çalışmada kullanılan yöntemler, veri setleri ve deneysel sonuçlar incelenmiş ve GAN modelinin bu alandaki potansiyeli vurgulanmıştır. Bu bağlamda, GAN'ların mevcut ve gelecekteki uygulamaları için önemli bir referans noktası oluşturulmuştur. Böylece farklı bir GAN modeli önerilmiştir.

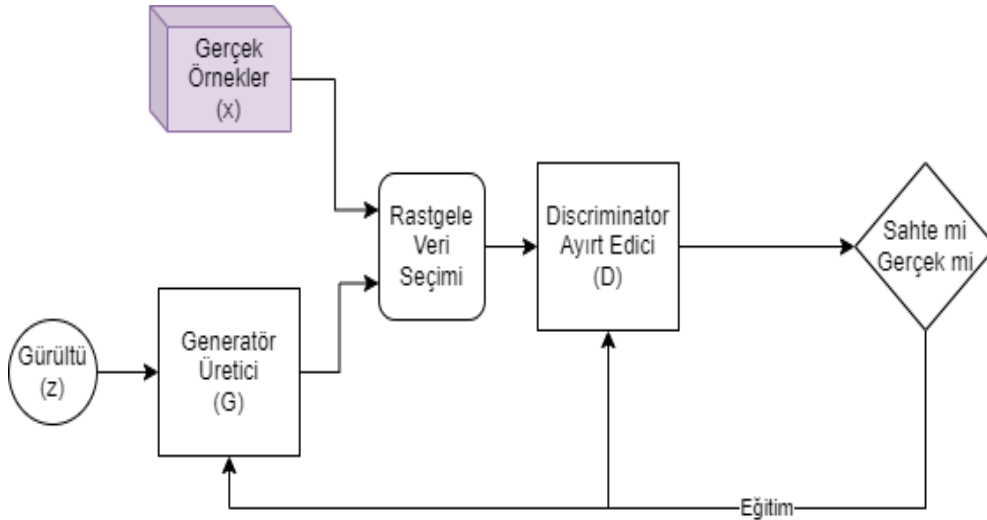
## 2. Literatür Taraması

### 2.1. Üretken Çekişmeli Ağlar (GANs)

Orijinal GAN modeli oyun teorisine dayanmaktadır. Denklem (1)'de (Mirza, M. ve ark. 2014) görülen minmax fonksiyonunda, aynı anda eğitilen Üretici G olarak, Ayırt Edici ise D olarak temsil edilmektedir. Bu denklemde  $E_{x \sim p_{data}}$  kısmı beklenen değeri temsil eder.  $E_{x \sim p_{data}(x)}$ , beklenen gerçek değerleri,  $E_{x \sim p_{data}(z)}$  ise beklenen sahte değerleri temsil etmektedir. Bu denklemdeki amaç en iyi sonuçları en az maliyetle üreterek eğitimin tamamlanmasıdır. Ayırt Edici D bu denklemin

sonucunu maksimum hale getirmeye çalışır. Üretici ise tam tersi şekilde, denklemin sonucunu minimum hale getirmeye çalışır. Denklemdaki  $D(x)$  ve  $D(G(z))$ 'nin alabileceği değerler “0” ile “1” arasındadır. Bu yüzden denklemin maksimum değeri  $\log 1 + \log(1 - 0)$ 'dan “0” olmaktadır. Üretici ise sadece denklemin ikinci kısmına etki edebilmektedir. Bu yüzden bu denklemin minimum değeri ikinci kısımdan  $\log(0)$ 'dan yaklaşık  $-\infty$  olmaktadır. Bu iki durum ağların rekabetinden ve kullanılan aktivasyon fonksiyonlarının sonucunun 0'a ulaşmamasından dolayı asla gerçekleşmemektedir. İdeal durumda  $D(x) = 0.5$  ve  $D(G(z)) = 0.5$  olmalıdır bu durumda ise bu denklemin sonucu yaklaşık olarak  $\log 0.5 + \log(1 - 0.5) = 2 \log 0.5 \cong -0.602$ 'dir. Bu hesaplama sadece tek değer için örnek olarak yapılmıştır. Şekil 2'de GAN'ın eğitim şeması gösterilmektedir.

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_{data}(z)} [\log (1 - D(G(z)))] \quad (1)$$



Şekil 2. Orijinal GAN akış şeması

Bu şemada GAN, rastgele verilen bir gürültüyle oluşturulan Üretici çıkışını gerçek örneklerle karıştırarak Ayırt Ediciye vermektedir. Ayırt Edici örneğin sahte veya gerçek olma durumunu değerlendirir. Sonuca göre ise geri yayılım (Boué, L. 2018) algoritmasının yardımıyla Ayırt Edici ve Üretici ağın katmanlarındaki ağırlıklar güncellenir. Bu algorithmada artan/azalan gradyan yöntemiyle her bir katmandaki düğümün hata üzerindeki etkisi, kısmi türevlerin yardımıyla bulunur. Bunun için kullanılan aktivasyon fonksiyonunun türevi olması gerekmektedir. Denklem (2)'de kullanılabilir aktivasyon fonksiyonlarından biri olan Sigmoid fonksiyonu ve onun türevi gösterilmektedir. Denklem (3)'te ise bir diğer aktivasyon fonksiyonu olan Tanh fonksiyonu ve türevi verilmiştir (Karabayır, İ. 2018). Denklem (4) ve Denklem (5)'de (Xu, B. ve ark. 2015) ise doğrusal aktivasyon fonksiyonları olan ReLU, ReLU'dan türeyen LReLU ve bunların türevleri bulunmaktadır. LReLU, ReLU fonksiyonunun negatif alanlarda da türevlenebilir halidir.

$$S(t) = \frac{1}{1+e^{-t}}, \quad S'(t) = S(t)(1-S(t)) \quad (2)$$

$$\tanh t = \frac{e^t - e^{-t}}{e^t + e^{-t}}, \quad \tanh' t = 1 - \tanh^2 t \quad (3)$$

$$y = f(x) = \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases}, \quad y' = f'(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad (4)$$

$$y = f(x) = \begin{cases} a \cdot x, & x < 0 \\ x, & x \geq 0 \end{cases}, \quad y' = f'(x) = \begin{cases} a, & x < 0 \\ 1, & x \geq 0 \end{cases} \quad (5)$$

*a* değişkeni 0 ila 1 arasında değer almaktadır, genellikle pratikte 0'a oldukça yakın değerler kullanılır.

### 2.1.2. Farklı GAN Modelleri

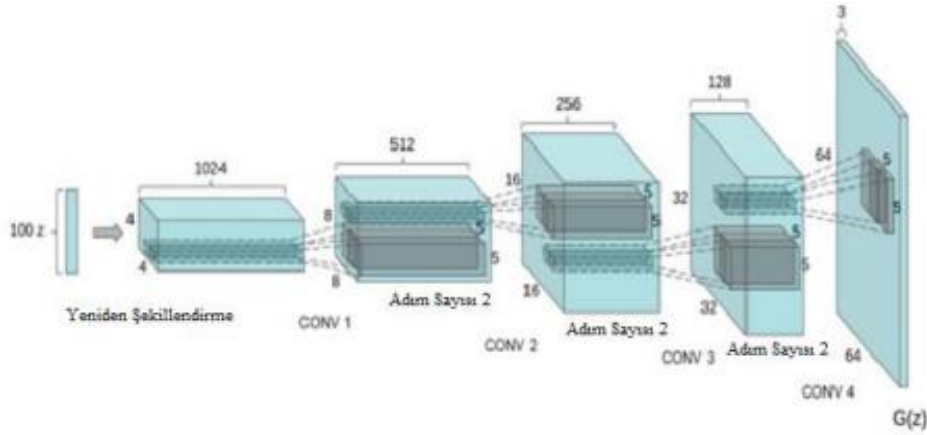
GAN'ın birçok farklı modeli bulunmaktadır. Her model problem veya amaca göre değişiklik göstermektedir. Çıkan sonuçların daha iyi olması üzerine çalışan bazı modeller bulunmaktadır. Bunlardan biri ağ yapısında Convolution (Evrışim) ve DeConvolution (Ters Evrışim) kullanan bir modele sahip olan DCGAN'dır (Radford, A. ve ark. 2015). Bu model özellikle resimler, fotoğraflar oluşturma üzerinedir. Ayırt Edici evrışimsel ağ yapısına sahip olup aldığı görüntüyü evrışimsel katmanlardan ve filtrelerden geçirerek bir değer döndürür. Üretici ise ters evrışimsel ağ yapısını kullanarak gelen rastgele gürültü dizisini filtreler ile katmanlardan geçirerek çıkışa bir görüntü olarak verir. Bu ağa benzer bir diğer model BigGAN'dır (Brock, A. ve ark. 2018). Bu model kademeli olarak görüntünün çözünürlüğünü arttırarak daha dengeli bir yapı elde etmek üzerinedir. Katmanlar halinde olan Üretici ağ, her katmanda bir önceki katmanın çıkışını alır ve sonra görüntünün boyutunu iki katına çıkarıp bir sonraki katmana verir. Ayırt Edici ise bunun tam tersi olarak her katmanda görüntüyü küçültür. Bunu yaparken her katmanda özellik çıkarır ve son karar katmanında bunları kullanarak bir sonuca varır. Bu sonuç orijinal GAN modelindeki gibi 0 ila 1 arasında bir sayıdır.

Eğer problem ayrıca koşullu sonuçlar üretilmesini gerektiriyorsa başka bir GAN modelinden faydalanılmalıdır. CGAN (Mirza, M. ve ark. 2014) bu modellerden biridir. CGAN diğer GAN modellerinden farklı olarak öğrenme sonucunda rastgele sonuçlar üretmektense belli bir kural çerçevesinde sonuçlar üretilmesi üzerinedir. Orijinal GAN modelinde, Ayırt Edici ağa sahte veya gerçek örnekler verilir ve bunları ayırt etmesi istenir. Üretici ise rastgele oluşturulmuş gürültü dizisiyle ürettiği sahte örneklerin, ne kadarının Ayırt Edici tarafından tespit edildiğine göre kendini günceller. CGAN'da ise Üretici ve Ayırt Edicinin girişlerine ekstradan belirlenen kurala göre oluşturulan bir veri eklenir. Böylece Ayırt Edici oluşturulan örnekleri, sahte olup olmadığını yanı sıra belirtilen kurala da uyup uymadığını kontrol eder. Ayırt Edicinin bu özelliği sayesinde Üretici

gerçekçi örnekler üretse de belirtilen kurala uymadığında Ayırt Edici tarafından düzenlenmesi için zorlanır. Buradaki kurallar örneklerin türleri modele ve istenen sonuca göre değiştirilebilir. Aynı amaçla MCGAN (Park, H. ve ark. 2018) ise tek bir kural yerine çok koşullu kurallar ile sonuçlar üretilmesi içindir.

GAN'da oluşabilen problemlere karşı ise farklı GAN modeli önerilmiştir. Bunlardan biri WGAN'dır (Arjovsky, M. ve ark. 2017). Bu modelde orijinal GAN modelindeki kayıp/hata fonksiyonu olan Jensen-Shannon Uyuşmazlığı yerine Wasserstein Mesafesi kullanılır. Bu sayede bazı durumlarda oluşan mod çökmesi, gradyanların kaybolması gibi eğitimi etkileyen etkenler minimize edilmiş olur. WGAN'a benzer olarak yine aynı sorunların önüne geçmek için LSGAN (Mao, X. ve ark. 2017) modelinde en küçük kareler hata fonksiyonunun kullanılmaktadır.

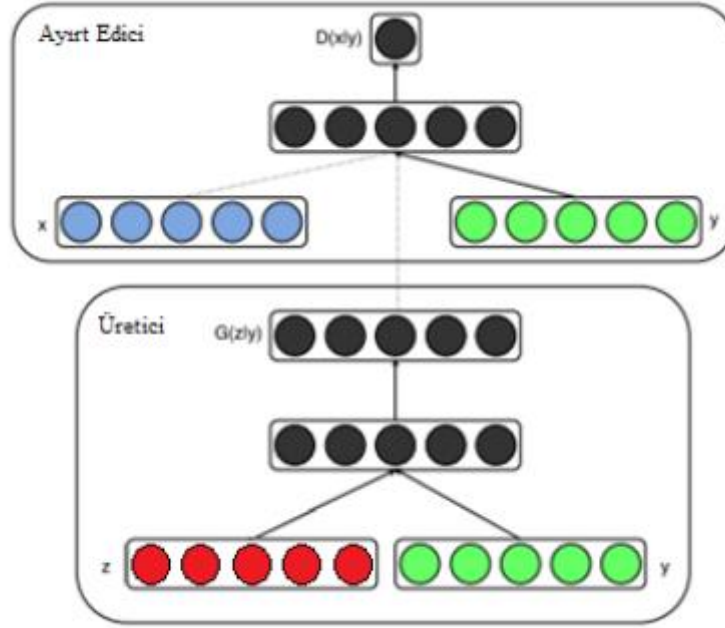
DCGAN, resim/fotoğraf gibi veriler oluşturmak üzere evrimsel katmanlar ve filtreler kullanır. Bu katmanlar ve filtreler veriler üzerinden bir takım özellik matrisi çıkarır. Her katmanda giriş bir önceki katmanın çıkarttığı özellik matrislerini de kullanarak görüntüyü oluşturmaya ya da ayrıştırmaya çalışır. Şekil 3'te (Radford, A. ve ark. 2015) DCGAN modelinin Üretici (G) ağının nasıl katman ve adımlardan geçtiği gösterilmiştir.



Şekil 3. DCGAN Üretici Ağ modeli (Radford, A. ve ark. 2015)

Şekil 3'te de görüldüğü gibi Üretici (G) rastgele üretilen gürültü dizisi olan  $z$  verisini ilk olarak  $4 \times 4 \times 1024$  boyutunda yeniden şekillendirdikten sonra özellik evrişim katmanları ile sonuçta  $64 \times 64 \times 3$  boyutlu bir görüntüye dönüştürmektedir. Ayırt Edici ağ ise Üreticinin tersini, devrik evrişim kullanarak yapar ve karar katmanına çıkardığı özellikleri vererek 0 ila 1 arasında bir puanlama yapar. Bu puanlama sonucuna göre orijinal GAN modelindeki gibi hatalar hesaplanır, istenilen hata oranına gelene kadar ağdaki her katmanın ağırlığı güncellenerek eğitime devam edilir.

CGAN, orijinal GAN modelindeki gibi rastgele sonuçlar üretmektense belli kriterleri sağlayan sonuçların üretilmesi üzerinedir. CGAN'ın ağ şeması Şekil 4'teki gibidir (Mirza, M. ve ark. 2014).



Şekil 4. CGAN, Üretici ve Ayırt Edici ağ şeması (Mirza, M. ve ark. 2014)

Şekil 4'te görüldüğü gibi orijinal GAN modelinden farklı olarak girişe ayrıca y adında bir dizi eklenir. Bu dizini herhangi bir şekilde olabilir (Mirza, M. ve ark. 2014). Bir sınıflandırma etiketi eklenmesi durumunda ağ bu sınıflandırma kriterlerine bağlı bir öğrenme yapacak ve istenilen sonucun üretilmesini sağlayacaktır.

WGAN modeli Denklem (1)'de kullanılan fonksiyon yerine daha stabil, daha az maliyetli bir öğrenme yöntemi olarak farklı bir fonksiyon önermiştir. Wasserstein Mesafesi olarak adlandırılan bu fonksiyon Denklem (6)'de görüldüğü gibidir (Arjovsky, M. ve ark. 2017).

$$W(P_r, P_g) = \inf_{\gamma \in \Pi(P_r, P_g)} E(x, y) \sim \gamma [\|x - y\|] \quad (6)$$

$\Pi(P_r, P_g)$  tüm eşzamanlı dağılımların kümesini temsil eder, burada marjinal dağılımlar sırasıyla  $P_r$  ve  $P_g$ 'dir. Sezgisel olarak,  $\gamma(x, y)$ , dağılımları  $P_r$ 'yi  $P_g$ 'ye dönüştürmek için x'ten y'ye taşınması gereken "kütle" miktarını gösterir. Wasserstein mesafesi, en iyi taşıma planının "maliyeti" dir (Arjovsky, M. ve ark. 2017).

Tablo 1'de ayrıca bazı GAN modelleri ve FID metriğine yani fréchet başlangıç mesafesine göre hangi veri setinde ne kadar başarılı oldukları gösterilmektedir (Li, Z. ve ark. 2022). FID oluşturulan görüntülerin dağıtımını bir dizi gerçek görüntünün dağıtımıyla karşılaştırır.

**Tablo 1.** FID metriğine göre (düşük olan daha iyi) bazı GAN modellerinin FFHQ-babies, FFHQ-sunglasses, face sketches veri setlerinde başarıları gösterilmektedir (Li, Z. ve ark. 2022).

Modeller	FFHQ-babies	FFHQ-sunglasses	face sketches
TGAN (Wang, Y. ve ark. 2018)	104.79	55.61	53.41
TGAN+ADA (Karras, T. ve ark. 2020)	102.58	53.64	66.99
MineGAN (Wang, Y. ve ark. 2020)	98.23	68.91	64.34
KD-GAN (Hou, X. ve ark. 2022)	68.67	34.61	35.87
StyleGAN2 (Karras, T. ve ark. 2020)	184.77	-	94.16
StyleGAN2+MDL (Kong, C. ve ark. 2022)	93.15	-	40.02
FastGAN (Liu, B. ve ark. 2020)	-	-	76.28

### 2.1.3. Diğer Yöntemler ve Araştırmalar

Sakal ekleme üzerine olan farklı çalışmalardan biri olan “Intuitive, Interactive Beard and Hair Synthesis with Generative Models” (Üretken Modellerle Sezgisel, İnteraktif Sakal ve Saç Sentezi) (Olszewski, K. ve ark. 2020) adlı makalede farklı bir yöntem olarak girişten maske olarak alınan sakal veya saç konumu ile eklenecek sakal veya saçın kılavuz vuruşları denen çizimini giriş olarak almaktadırlar. Bunu iki parçadan oluşan bir ağ kullanarak oluşturmuşlardır. Ayrıca bu makale de ki model interaktif olarak görüntü üzerinde değişiklik yapmaya izin vermektedir. Bunun için kullanıcıdan yine saç veya sakal olduğu bölgenin seçilmesi istenmektedir. Böylece hangi bölgede çalışılacağı bilinerek yeni girdi olarak alınan saç veya sakal çizimi fotoğraf üzerinde yeni girişe göre güncellenmesi sağlanır.

Görüntüler üzerinde istenmeyen gölgeler, filigran vb. durumların kaldırılması için “Unsupervised Portrait Shadow Removal via Generative Priors” (Üretken Öncelikler aracılığıyla Denetimsiz Portre Gölgesini Kaldırma.) (He, Y. ve ark. 2021) adlı makalede önerilen model ise eğitilmiş StyleGAN2 ile tamamen gölgelendirilen görüntülerin MaskNet ağı ile tersine çevrilmesi işlemi yapılarak eğitim yapılmıştır.

Bir diğer yöntem olarak “Generating Synthetic Disguised Faces with Cycle-Consistency Loss and an Automated Filtering Algorithm” (Döngü Tutarlılığı Kaybı ve Otomatik Filtreleme Algoritması ile Sentetik Gizlenmiş Yüzler Oluşturma) (Ahmad, M. Ve ark. 2021) adlı makalede CNN ağlar kullanarak yüz fotoğraflarına saç, sakal, maske vb. eklemeler yapılmıştır. Çalışmada Döngü Tutarlılığı Kaybının kullanılması ile görüntünün tamamının öğrenmek yerine bir alandan öğrenme



sağlanmış olundu. Bu durum sadece saç, sakal gibi bölgelerin eğitime dahil edilmesini sağladığı görülmüştür.

GAN kullanarak yazılı bir açıklamadan resim oluşturma üzerine yapılan bir çalışma olan “Optimal text-to-image synthesis model for generating portrait images using generative adversarial network techniques” (Üretken Çekişmeli Ağ tekniklerini kullanarak portre görüntüleri oluşturmak için en uygun metinden görüntüye sentez modeli) (Berrahal, M. ve ark. 2022) adlı makalede ise bazı GAN modellerinin iyileştirilmesi ile metin tabanlı resim üretimi üzerinde çalışılmıştır. Bu makalede DF-GAN (Tao, M. ve ark. 2022) modeli için ekstra ağlar gerekmeden üreticiyi geliştirmek amacıyla ayırt edici ağı hedefe duyarlılığını arttırmak üzerine bir öneride bulunmuştur. Attn-GAN (Xu, T. ve ark. 2018) için ise ilk aşamada düşük çözünürlüklü görüntüler oluşturmak için metin vektöre dönüştürülür ve ardından bu makalede önerilen portre görüntüler oluşturmak için metinden görüntüye sentez modeli ile birleştirilir. Böylece modelde iyileşme sağlanmış olur.

### 3. Materyal ve Metotlar

#### 3.1. Ön işlemler ve Araçlar

Bu çalışma insan yüzlerinde belirlenen kurallara göre saç, sakal, kellik gibi modifikasyonların yapılması üzerinedir. Bunun için CelebA (Liu, B. ve ark. 2015) veri setinden bu türlere göre rastgele seçilen yaklaşık 8200 fotoğraf ile eğitim gerçekleştirilmiştir. Eğitim için Python Tensorflow Keras ve Cuda'dan yararlanılmıştır. Ön işlem olarak seçilen rastgele fotoğraflar 178x218 boyutunda olduğundan ortalanarak 178x178 şekilde kırpılmış ve 64x64'e sıkıştırılmıştır. Eğitim sırasında ayrıca rastgele düşük Gauss gürültüleri eklenilmiştir. Bu gürültüler az örnekler ile eğitim yaparken veri çeşitliğinin artırılması ve Üretici-Ayırt Edici dengesinin sağlanması içindir. Kısıtlı bellek miktarından, eğitim sırasında sürekli yeniden görüntüleri işleme durumu olmaması için ekstra veri artırma işlemleri uygulanmamıştır. Var olan yaklaşık 8200 fotoğraf da görüntüler döndürülmüş, sağdan, soldan çekilmiş olarak bulunmaktadır. Bu yüzden eğitimde her döngüde gauss gürültüsü oranı rasgele bir şekilde bir miktar artırılıp azaltılmıştır.

#### 3.2. Veri Seti

Bu çalışmada CelebA (Liu, B. ve ark. 2015) veri seti kullanılmıştır. Bu veri setinde her görüntünün hangi özelliklere sahip olduğu etiketlenmiş bir şekilde bulunmaktadır. Bu özelliği nedeniyle bu veri seti seçilmiştir. Veri setinde görüntüdeki insanın sakalı var ise sadece etiket olarak sakal etiketine 1 yok ise -1 olarak yazılmıştır. Sakalın fotoğrafın neresinde olduğu bilgisi

verilmemiştir. Bu eğitim sırasında sadece fotoğraflar arasındaki farka dayalı olarak saç, sakal verisini ağı kendi başına öğrenmesi sağlamaktadır. Fakat bu veri seti kare görüntülerden oluşmamaktadır. Bu nedenle ayrıca görüntüler kırılıp sıkıştırılmıştır. Bu veri setinin bir diğer dezavantajı ise görüntülerin büyük bir kısmının bozuk olması. Şekil 5'te bu tarz bozuk görüntülere örnek verilmiştir.



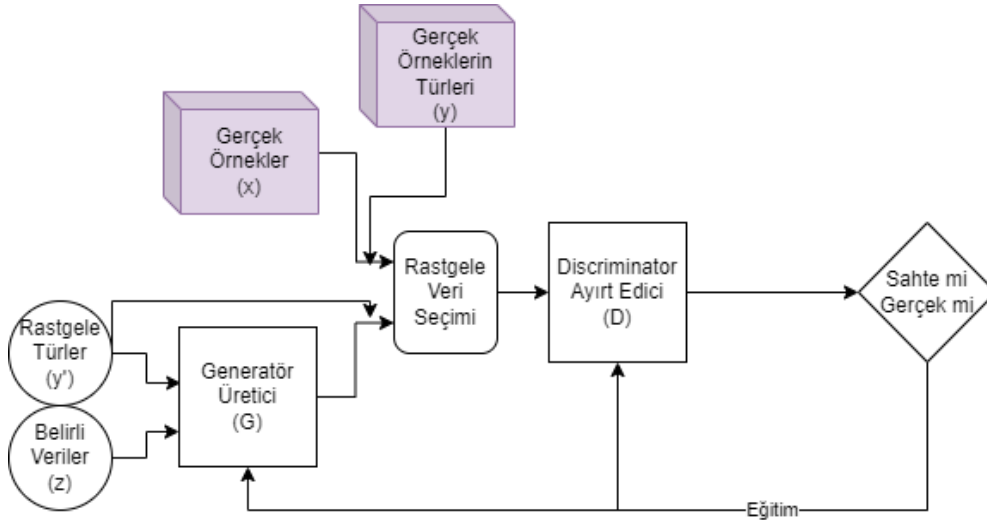
**Şekil 5.** Veri setindeki bazı bozuk görüntülere örnek

Şekil 5'te de görüldüğü gibi bazı görüntüler bozuk olmaktadır. Bu durum ağı eğitimi olumsuz etkilemektedir. Fakat veri setinin etiketleme durumu ve bazı bozuk görüntülerin ayıklanması sonucu bu veri setinin çalışma için uygun olduğu görülmüştür.

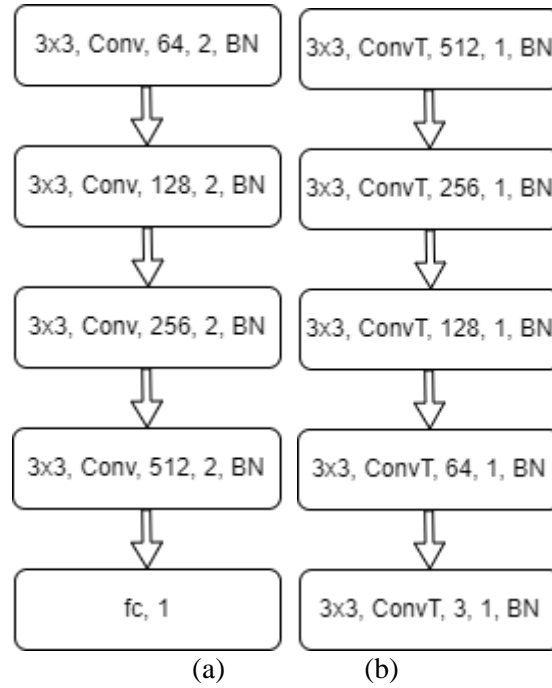
### 3.3. Uygulanan Yöntemler

Ağı yapısı olarak Ayırt Edici de evrişimli (convolutional) ağı ile üretilen görüntüler 64x64 ve rgb renk uzayında olup 4 katman evrişimli ve 1 katman tam bağlantılı (Fully Connected) olarak tasarlanmıştır. Üretici ağı da ise giriş 64x64 rgb uzayında bir görüntü olup 5 katman devrik evrişim (convolution transpose) kullanılmıştır. İki ağıda da her katmanda filtre boyutu 3x3 olarak kullanılmıştır. Ayrıca her katmanda eğitimi daha stabil hale getirmek için normalizasyon yapılmıştır. Oluşturulan farklı GAN modelinin eğitim sırasındaki yapısı Şekil 6'da görüldüğü gibidir. Şekil 7 (a) ve (b)'de uygun ağı şeması gözükmemektedir. Son katmanlar hariç her katmanda LReLU aktivasyon fonksiyonu kullanılmıştır. Üretici ağıda son katmanda tanh, Ayırt Edici ağı da ise son katmanda Logistic/Sigmoid aktivasyon fonksiyonları ile Denklem (7)'de gösterilen ikili çapraz entropi hata fonksiyonu kullanıldı (Ho, Y. ve ark. 2019).

$$J_{BCE} = -\frac{1}{M} \sum_{m=1}^M [y_m \log(h_\theta(x_m)) + (1 - y_m) \log(1 - h_\theta(x_m))] \quad (7)$$



Şekil 6. Oluşturulan farklı GAN modelinin çalışma şeması



Şekil 7. Ağların şeması  
(a) Ayırt Edici ağ (b) Üretici ağ

Orijinal GAN modelinden farklı olarak gürültü yerine modifiye edilecek fotoğraflar  $z$  Üretici ağına, değiştirilmesi istenilen özellikler  $y'$  ile verilir. Üretici gelen örnekler ile koşul verilerini bir matris de birleştirerek istenilen türde resmi oluşturmaya çalışır. Bunun için Şekil 7 (b)'de ki Üretici ağ yapısı kullanıldı. Üretici tarafından üretilen görüntüler ile istenilen türler çıkışta birleştirilir, aynı durum gerçek örnekler üzerinde de uygulanır. Sonra bu veriler karıştırılarak ağ yapısı Şekil 7 (a)'da ki gibi olan Ayırt Edici ağına gönderilir. Bu ağ gelen örneklerin gerçek olmasına ve ayrıca verilen türle uyumlu olup olmadığına 0 ila 1 arası bir puanlama ile karar verir. "0" durumu uyumsuz veya sahte anlamına gelirken, "1" durumu uyumlu ve gerçek anlamına gelmektedir. Ayırt Edici için hata, gerçek görüntülere ne kadar sahte dediği ve sahte görüntüleri ne kadar gerçek olarak tahmin ettiğinin

toplamıdır. Üretici için ise hata, ürettiği görüntülerin Ayırt Edici tarafından ne kadar sahte olarak fark edildiğinin oranı kadardır. Bu işlemlerden sonra eğitim aşamasına geçilerek geri yayılım algoritmaları ile iki ağ, bu hata oranlarına göre yeniden ağırlıklandırılır. Bu işlem her döngüde tekrar edilir. Ayırt Edici ağ artık gerçek görüntüler ile sahte görüntüleri ayırt edemediğinde eğitim tamamlanmış olur.

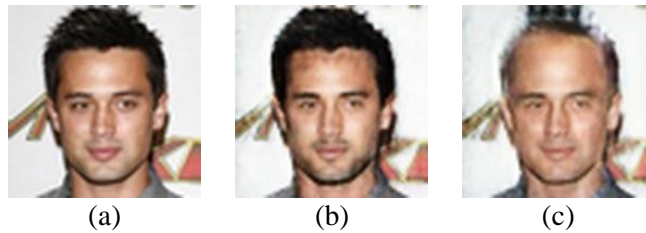
#### 4. Sonuçlar ve Öneriler

Çoğu GAN modelinden farklı olarak rastgele gürültüler ile rastgele sonuçlar üretmektense gerçek fotoğraflar üzerinde istenilen durumların oluşturulması bu çalışmanın temel amacıdır. Bu modele benzer CNN ağları bulunmakla birlikte çoğu ticari amaçlı olduğundan model yapısı paylaşılmamıştır. Herkesin erişiminde olan modeller için eğitim sırasında bu çalışmadaki gibi sakal ekleme veya kelleştirme için aynı kişinin farklı fotoğraflarını gerektirmektedir. Bu durum veri seti bulmakta ya da hazırlamakta zorluklara sebep olmaktadır. Fakat bu çalışmada hiçbir morfolojik filtre de kullanılmadan kellik ve sakallı durumları öğrendiği görülmektedir. Bu süreç, sadece fotoğraflar arasında ortak benzerliklerin ağ tarafından fark edilmesi ile gerçekleştirilmiştir. Bu sayede sadece fotoğraf ve o fotoğrafın saç sakal bilgisinin etiketi eğitim için yeterli olmaktadır. Ayrıca sakalın, saçın işaretlenmesi veya aynı kişinin sakallı, sakalsız, kel, saçlı gibi fotoğraflarına eğitim aşamasında kullanılmasına gerek kalmamıştır.

##### 4.1. Deneysel Sonuçlar

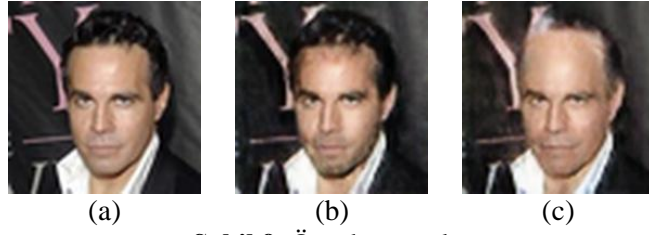
Şekil 7 (a) ve (b) de görünen modeller kullanılarak oluşturulan ağlar yaklaşık olarak her eğitim döngüsünde 203 saniye harcamıştır. Kullanılan makinede işlemci i7 8750H, ekran kartı Nvidia GTX 1060, 2x8gb 2667 Mhz ddr4 ram bulunmaktadır.

Saç ve sakal ekleme için eğitilen ağın yaklaşık 140 eğitim döngüsü için verdiği bazı örnek sonuçlar Şekil 8’de ve Şekil 9’da gözükmemektedir. Şekil 10’da eğitim sayısına göre sakal eklemeye ve kelleştirilmeye çalışılan bir örnek için sonuçlar görülmektedir.



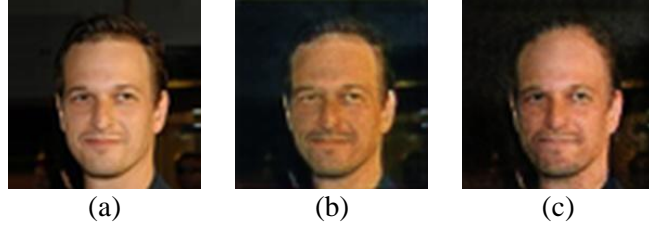
Şekil 8. Örnek sonuçlar

a) Girişten verilen fotoğraf b) Sakal eklenmiş çıkış c) Kelleştirme sonucu çıkış



Şekil 9. Örnek sonuçlar

a) Girişten verilen fotoğraf b) Sakal eklenmiş çıkış c) Kelleştirme sonucu çıkış

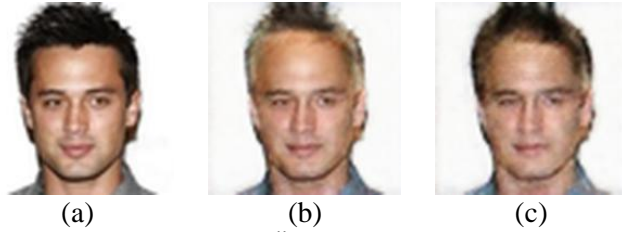


Şekil 10. Belli döngülerde sonuçlar

a) Girişten verilen fotoğraf b) 3. Döngüden sonra sonuç c) 60. Döngüden sonra sonuç

#### 4.1.1. Başarım Metriği ile Sonuçlar

Görüldüğü üzere eğitim süreci arttıkça görüntüdeki kelleştirme oranı ve sakal ekleme başarısı gözle görülür bir biçimde artmaya başlamıştır. Ayrıca geriye yönelik bir eğitim söz konusu olmamasına rağmen çıkan sonuç giriş olarak verilip eski fotoğrafı yeniden yapması istediğinde ortalama %62 bir başarıyla fotoğraf tekrar oluşturulabilmiştir. Fotoğrafların arka planı silinerek sadece yüz için hesaplandığında geriye döndürme ortalama %85 oranında olmuştur. Örnek bir fotoğraf Şekil 11’de görülmektedir.



Şekil 11. Örnek sonuçlar

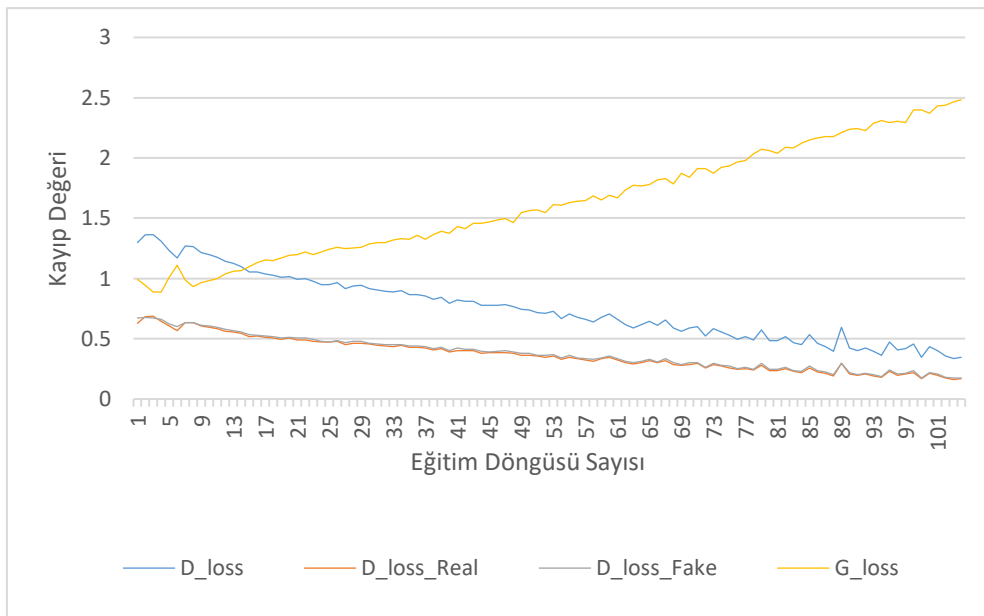
a) Girişten verilen fotoğraf b) Kelleştirme sonucu c) Tekrar saç ekleme sonucu

Bu oranların hesabında Denklem (8)’de ki Yapısal benzerlik indeksi ölçüsü diğer bir adıyla SSIM (Wang, Z. ve ark. 2004) fonksiyonu kullanılmıştır.

$$SSIM(x, y) = \left( \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \right) \left( \frac{2\sigma_{xy} + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \right) \quad (8)$$

## 4.2. Öneriler

Sonuçlarda görüldüğü gibi ağ kelleştirme işleminde daha az başarı göstermekle birlikte sakal ekleme işleminde daha başarılı görülmektedir. Kelleştirme adımında saçlara eklenen beyazlıklar veri setinde bulunan kel insanların tamamen kel olmadığı ve kalan saçlarının beyaz olmasından kaynaklı olduğu sonucuna farklı örneklerde de test edilerek ulaşılmıştır. Veri setindeki her görüntünün iyi olmaması, sakallı/kel ise sakalı/kelliği ne kadar az veya çok yoğun olsa da aynı etiket değerine sahip olması ve sadece 8200 farklı örnek ile eğitim yapılmasından dolayı mükemmel sonuçlar elde edilememiştir. İleride daha yüksek örnek sayısı ile daha yumuşak etiketli bir veri setiyle daha yüksek başarılı sonuçlar elde edilebilir.



Şekil 12. Eğitim döngüsüne göre Üretici (G) ve Ayırt Edici(D) kayıp oranları ikili çapraz entropi ile hesaplanmıştır.

Şekil 12’de eğitim döngüsüne göre Üreticinin hata oranı ile Ayırt Edici ağı sahte ve gerçek görüntüler için olan hata oranı gösterilmiştir. Ayrıca yine Şekil 12’de görüldüğü gibi Ayırt Edici kayıp değeri sürekli düşmektedir. Yani Üretici ağ yeteri kadar güçlü değil ya da Ayırt Edici ağ çok güçlüdür. Bu durumda eğitim ilerleyen aşamalarda durma noktasına gelmektedir. Çözüm olarak daha güçlü bilgisayarlar daha gelişmiş bir Üretici ağ yapılabilir.

## Yazarların Katkısı

Tüm yazarlar çalışmaya eşit katkıda bulunmuştur.

## Çıkar Çatışması Beyanı

Yazarlar arasında herhangi bir çıkar çatışması bulunmamaktadır.

## Araştırma ve Yayın Etiği Beyanı

Yapılan çalışmada araştırma ve yayın etiğine uyulmuştur.

## Kaynaklar

- Ahmad, M., Cheema, U., Abdullah, M., Moon, S., & Han, D. (2021). Generating synthetic disguised faces with cycle-consistency loss and an automated filtering algorithm. *Mathematics*, 10(1), 4.
- Berrahal, M., & Azizi, M. (2022). Optimal text-to-image synthesis model for generating portrait images using generative adversarial network techniques. *Indonesian Journal of Electrical Engineering and Computer Science*, 25(2), 972-979.
- Boué, L. (2018). Deep learning for pedestrians: backpropagation in CNNs. arXiv preprint arXiv:1811.11987.
- Brock, A., Donahue, J., & Simonyan, K. (2018). Large scale GAN training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096.
- Goodfellow, I. (2016). Nips 2016 tutorial: Generative adversarial networks. arXiv preprint arXiv:1701.00160.
- Ho, Y., & Wooley, S. (2019). The real-world-weight cross-entropy loss function: Modeling the costs of mislabeling. *IEEE access*, 8, 4806-4813.
- Hou, X., Liu, B., Wan, F., & You, H. (2022). Exploiting Knowledge Distillation for Few-Shot Image Generation. <https://openreview.net/forum?id=vsEi1UMa7TC/>
- I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, 'Generative adversarial nets', in *Advances in neural information processing systems*, pp. 2672–2680, 2014.
- Karabayır İ. (2018). Gradyan ve Özel Bir Hiper Düzlem Temelli Yeni Bir Optimizasyon Algoritması: Evriştirilmiş Gradyan Yönü ile Optimizasyon. Doktora Tezi. İstanbul Üniversitesi Sosyal Bilimler Enstitüsü İşletme Anabilim Dalı Sayısal Bilim Dalı
- Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., & Aila, T. (2020). Training generative adversarial networks with limited data. *Advances in neural information processing systems*, 33, 12104-12114.
- Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., & Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 8110-8119).
- Li, Z., Xia, B., Zhang, J., Wang, C., & Li, B. (2022). A comprehensive survey on data-efficient GANs in image generation. *arXiv preprint arXiv:2204.08329*.
- Liu, Ziwei and Luo, Ping and Wang, Xiaogang and Tang, Xiaoou (2015). Proceedings of International Conference on Computer Vision (ICCV). Deep Learning Face Attributes in the Wild
- Mao, X., Li, Q., Xie, H., Lau, R. Y., Wang, Z., & Paul Smolley, S. (2017). Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 2794-2802).
- Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784.
- Olszewski, K., Ceylan, D., Xing, J., Echevarria, J., Chen, Z., Chen, W., & Li, H. (2020). Intuitive, interactive beard and hair synthesis with generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 7446-7456).
- Park, H., Yoo, Y., & Kwak, N. (2018). Mc-gan: Multi-conditional generative adversarial network for image synthesis. *arXiv preprint arXiv:1805.01123*.
- Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434.
- Salehi, P., Chalechale, A., & Taghizadeh, M. (2020). Generative adversarial networks (GANs): An overview of theoretical model, evaluation metrics, and recent developments. arXiv preprint arXiv:2005.13178.

- Tao, M., Tang, H., Wu, F., Jing, X. Y., Bao, B. K., & Xu, C. (2022). Df-gan: A simple and effective baseline for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 16515-16525).
- Wang, Y., Gonzalez-Garcia, A., Berga, D., Herranz, L., Khan, F. S., & Weijer, J. V. D. (2020). Minegan: effective knowledge transfer from gans to target domains with few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 9332-9341).
- Wang, Y., Wu, C., Herranz, L., Van de Weijer, J., Gonzalez-Garcia, A., & Raducanu, B. (2018). Transferring gans: generating images from limited data. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 218-234).
- Wang, Zhou & Bovik, Alan & Sheikh, Hamid & Simoncelli, Eero. (2004). Image Quality Assessment: From Error Visibility to Structural Similarity. *Image Processing, IEEE Transactions on*. 13. 600-612. 10.1109/TIP.2003.819861.
- Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. arXiv preprint arXiv:1505.00853.
- Xu, T., Zhang, P., Huang, Q., Zhang, H., Gan, Z., Huang, X., & He, X. (2018). AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1316-1324).
- WOS Topic: Generative Adversarial Networks
- Arjovsky, M., Chintala, S., & Bottou, L. (2017, July). Wasserstein generative adversarial networks. In *International conference on machine learning* (pp. 214-223). PMLR.
- Liu, B., Zhu, Y., Song, K., & Elgammal, A. (2020, October). Towards faster and stabilized gan training for high-fidelity few-shot image synthesis. In *International Conference on Learning Representations*.
- He, Y., Xing, Y., Zhang, T., & Chen, Q. (2021, October). Unsupervised portrait shadow removal via generative priors. In *Proceedings of the 29th ACM International Conference on Multimedia* (pp. 236-244).
- Kong, C., Kim, J., Han, D., & Kwak, N. (2022, October). Few-shot image generation with mixup-based distance learning. In *European Conference on Computer Vision* (pp. 563-580). Cham: Springer Nature Switzerland.