



POLİTEKNİK DERGİSİ

*JOURNAL of POLYTECHNIC*

ISSN: 1302-0900 (PRINT), ISSN: 2147-9429 (ONLINE)

URL: <http://dergipark.org.tr/politeknik>



# Derin öğrenme modellerinin sinirsel stil aktarımı performanslarının karşılaştırılması

## *Comparison of neural style transfer performance of deep learning models*

Yazar(lar) (Author(s)): Batuhan KARADAĞ<sup>1</sup>, Ali ARI<sup>2</sup>, Müge KARADAĞ<sup>3</sup>

ORCID<sup>1</sup>: 0000-0002-4661-6607

ORCID<sup>2</sup>: 0000-0002-5071-6790

ORCID<sup>3</sup>: 0000-0002-5722-5441

**Bu makaleye şu şekilde atıfta bulunabilirsiniz (To cite to this article):** Karadağ B., Arı A. ve Karadağ M., “Derin öğrenme modellerinin sinirsel stil aktarımı performanslarının karşılaştırılması”, *Politeknik Dergisi*, 24(4): 1611-1622, (2021).

**Erişim linki (To link to this article):** <http://dergipark.org.tr/politeknik/archive>

**DOI:** 10.2339/politeknik.885838

# Derin Öğrenme Modellerinin Sinirsel Stil Aktarımı Performanslarının Karşılaştırılması

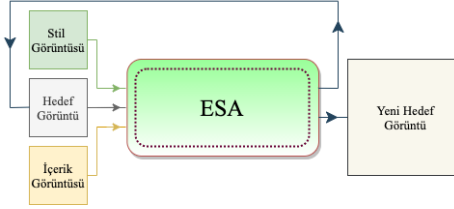
## Comparison of Neural Style Transfer Performance of Deep Learning Models

### Önemli noktalar (Highlights)

- ❖ Farklı derin öğrenme modellerinin sinirsel stil aktarım performansları karşılaştırılmıştır. / Neural style transfer performance of different deep learning models was compared.
- ❖ Farklı optimizasyon algoritmaları ile derin öğrenme modellerinin performansları karşılaştırılmıştır. / The performances of different optimization algorithms and deep learning models were compared.

### Grafik Özet (Graphical Abstract)

Farklı Evrişimsel Sinir Ağı (ESA) modelleri ile Sinirsel Stil Aktarımı (SSA) gerçekleştirilmiştir. Farklı optimizasyon algoritmaları kullanılarak ESA modellerinin performansları incelenmiştir. Elde edilen sonuçlar karşılaştırılarak en iyi model belirlenmiştir. / Neural Style Transfer (NST) has been carried out with different Convolutional Neural Network (CNN) models. The performances of CNN models were examined using different optimization algorithms. The best model was determined by comparing the obtained results.



Şekil A. Sinirsel stil aktarımı mimarisi / Figure A. Neural style transfer architecture

### Amaç (Aim)

SSA çalışmalarında kalite ve performans artırımı en çok amaçlanan hedeflerdendir. Bu çalışmada da farklı ESA modellerinin SSA performansları incelenmiş ve farklı optimizasyon algoritmalarının ESA modellerinin SSA çalışmalarındaki etkisi değerlendirilmiştir. / Quality and performance enhancement is one of the most aimed goals in NST studies. In this study, NST performances of different CNN models were examined and the effect of different optimization algorithms on the NST studies of CNN models was evaluated.

### Tasarım ve Yöntem (Design & Methodology)

İçerik ve stil olarak adlandırılan iki görüntü, seçilen ESA modeline aktarılmıştır. Optimizasyon algoritmaları kullanılarak belirlenen iterasyon adımı sonunda hedef görüntü üretilmiştir. / Two images, named content and style, were transferred to the selected CNN model. The target image was generated at the end of the iteration step determined by using optimization algorithms.

### Özgünlük (Originality)

Yapılan bu çalışmada VGG16, VGG19 ve ResNet50 ESA modellerinin SSA performansları karşılaştırılmıştır. Adam, RMSprop ve SGD optimizasyon algoritmalarının ESA modelleri üzerindeki etkisi yapılan çalışmalarla değerlendirilmiştir. / In this study, NST performances of VGG16, VGG19 and ResNet50 CNN models were compared. The effect of Adam, RMSprop and SGD optimization algorithms on CNN models has been evaluated by the studies.

### Sonuç (Conclusion)

Yapılan SSA çalışmalarında, görsel açıdan en iyi performans VGG19 ağ modelinden SGD optimizasyon algoritması kullanılarak elde edilmiştir. Zaman açısından en hızlı SSA ResNet50 ESA modelinde SGD optimizasyon algoritması kullanılarak elde edilmiştir. / In the NST studies, the best visual performance was obtained from the VGG19 network model by using the SGD optimization algorithm. In terms of time, the fastest NST was obtained by using SGD optimization algorithm in ResNet50 CNN model.

### Etik Standartların Beyanı (Declaration of Ethical Standards)

Bu makalenin yazar(lar)ı çalışmalarında kullandıkları materyal ve yöntemlerin etik kurul izni ve/veya yasal-özel bir izin gerektirmediğini beyan ederler. / The author(s) of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

# Derin Öğrenme Modellerinin Sinirsel Stil Aktarımı Performanslarının Karşılaştırılması

*Araştırma Makalesi / Research Article*

**Batuhan KARADAĞ<sup>1</sup>, Ali ARI<sup>2\*</sup>, Müge KARADAĞ<sup>3</sup>**

<sup>1,2</sup>İnönü Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Türkiye

<sup>3</sup>İnönü Üniversitesi, Fen-Edebiyat Fakültesi, Matematik Bölümü, Türkiye

(Geliş/Received :24.02.2021; Kabul/Accepted :03.03.2021 ; Erken Görünüm/Early View : 08.03.2021)

## ÖZ

Sinirsel stil aktarımı günümüzde hem akademik hemde endüstriyel alanda en fazla çalışılan konulardan biridir. Yapılan çalışmalarda kalite ve performans artırımı en çok amaçlanan hedeflerdendir. Bu çalışmada da farklı ESA modellerinin sinirsel stil aktarımında performansları incelenmiştir. VGG16, VGG19 ve ResNet50 modelleri kullanarak derin öznetelikler elde edilmiştir. Bu öznetelikler sayesinde içerik görüntüsünün içeriğini, stil görüntüsünün de stilini alıp yeni bir hedef görüntü oluşturulmuştur. Adam, RMSprop ve SGD optimizasyon algoritmaları kullanılmıştır. Yapılan sinirsel sinir aktarımı çalışmalarında, görsel açıdan en iyi performans VGG19 ağ modelinden SGD optimizasyon algoritması kullanılarak elde edilmiştir. Zaman açısından en hızlı sinirsel stil aktarımı ResNet50 evrişimsel sinir ağı modelinde SGD optimizasyon algoritması kullanılarak elde edilmiştir.

**Anahtar Kelimeler:** Sinirsel stil aktarımı, evrişimli sinir ağı, derin evrişimli sinir ağı.

## Comparison of Neural Style Transfer Performance of Deep Learning Models

### ABSTRACT

Neural style transfer is one of the most studied topics in both academic and industrial fields. Quality and performance enhancement are among the most targeted goals in the studies. In this study, the performance of different CNN models in neural style transfer was investigated. Deep features were obtained using VGG16, VGG19 and ResNet50 models. Thanks to these attributes, a new target image is created by taking the content of the content image and the style of the style image. Adam, RMSprop and SGD optimization algorithms are used. In neural transfer studies, the best visual performance was obtained from VGG19 network model by using SGD optimization algorithm. The fastest neural style transfer in terms of time was obtained using the SGD optimization algorithm in the ResNet50 convolutional neural network model.

**Keywords:** Neural style transfer, convolutional neural network, deep convolutional neural network.

### 1. GİRİŞ (INTRODUCTION)

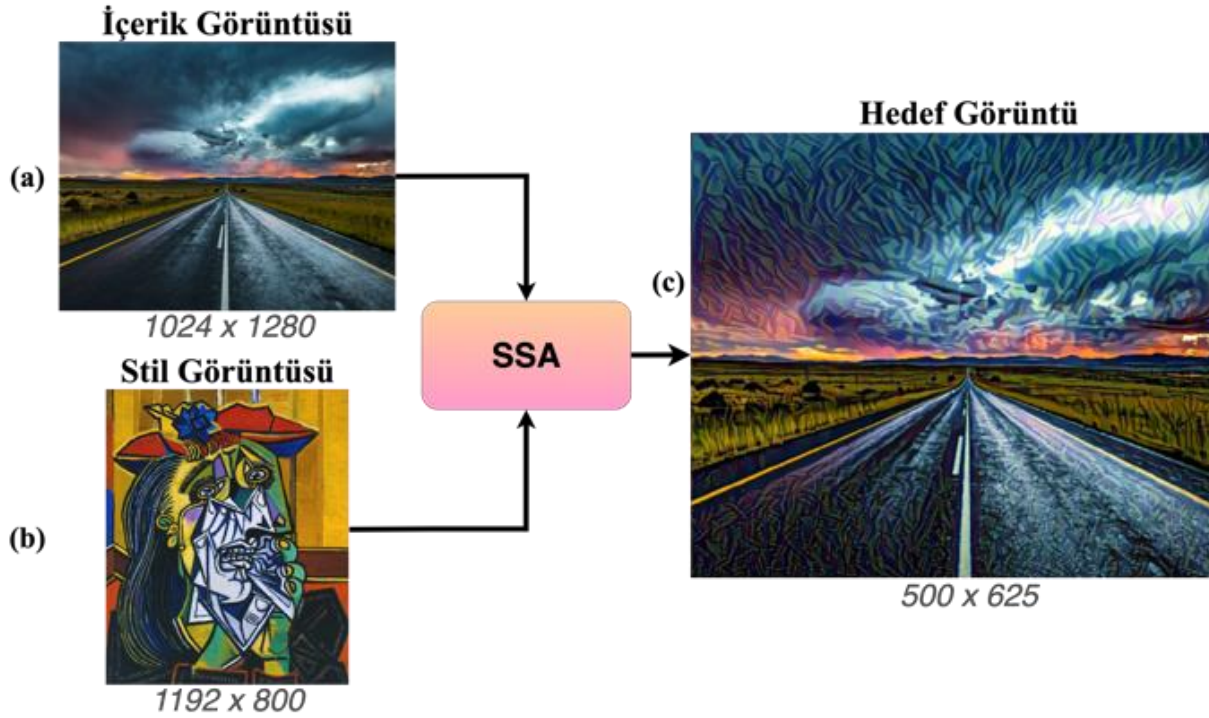
Resim çizme, yıllardır popüler bir sanat biçimi olmuştur [1]. İnsanlar, bir fotoğrafı sanki bir sanatçının elinden çıkmış yağlı boya tablosu gibi görünmesini sağlayabilir miydi? Gatys ve arkadaşları 2016 yılında bu soruya odaklanıp, fotoğrafların tıpkı sanatçı elinden çıkmış gibi görünmesini sağlamışlardır. Bu yöntem literatürde "Sinirsel Stil Aktarımı" (SSA) olarak adlandırılmıştır.

ESA, çok katmanlı ağ yapısı sebebiyle son günlerde yaygın kullanılmaya başlanması ve öznetelik çıkarma konusunda ortaya koyduğu başarı sayesinde stil aktarımında olumlu sonuçlar elde etmeyi başarmıştır. Bu bulgulara dayanarak, Gatys ve ark. [3] öncelikle, bir fotoğrafın içeriğini ve ünlü sanat eserlerinin tarzını

yeniden birleştirmek için ESA modelinin öznetelik çıkarımından yararlanmayı önermişlerdir [1]. Görüntüden içeriği ve stili ayırmak oldukça zor bir problemdir. Bununla birlikte, ESA [4] modelinde yaşanan gelişmelere bağlı olarak, doğal görüntülerden yüksek seviyeli anlamsal bilgi çıkarmayı öğrenen bilgisayarlı görme sistemleri üretilmiştir. Yeterli etiketlere sahip bir veri setiyle eğitilen ESA modeli, nesne tanıma, görüntü içeriği çıkarma, görüntü stilini çıkarma ve stil sınıflandırma çalışmalarında başarı ile kullanılmaktadır [5-12].

SSA çalışmalarında, girişe uygulanan iki görüntüden biri aktarma yapmak istenilen görüntü (içerik görüntüsü) diğeri ise stili kopyalanmak istenilen görüntüdür (stil görüntüsü). SSA'da amaç ise bu iki görüntüden yeni bir hedef görüntü oluşturmaktır. SSA çalışma prensibi Şekil 1'de gösterilmiştir.

\*Sorumlu Yazar (Corresponding Author)  
e-posta : ali.ari@inonu.edu.tr



Şekil 1. Sinirsel stil aktarımı ile hedeflenen durum şeması (The targeted state diagram with neural style transfer)

Şekil. 1’de görüldüğü gibi (b) görüntüsündeki sanatçının stili, (a) görüntüsü üzerine aktarılıp (c) hedef görüntüsü elde edilmiştir. Ayrıca Şekil. 1 incelendiğinde, stil aktarımının farklı görüntü boyutlarıyla yapılabildiğini görülmektedir. Günümüzde SSA tabanlı birçok başarılı endüstriyel uygulama (örneğin, Prisma, Ostagram ve Deep Forger) kullanılmaktadır [1].

Bu çalışmada, stil aktarımı genel olarak incelenmiş ve metodolojisi anlatılmıştır. Stil aktarımı, farklı ESA modelleri üzerinde (VGG16 VGG19, ResNet50) test edilmiştir. Modellerden geçirilen görüntüler ile, içerik görüntüsünün içeriğini ve stil görüntüsünün stilini kopyalayarak yeni bir hedef görüntü oluşturulmuştur. Görsel açıdan en iyi performans VGG19 ağ modelinden SGD optimizasyon algoritması kullanılarak elde edilmiştir. Zaman açısından en hızlı SSA ResNet50 ESA modelinde SGD optimizasyon algoritması kullanılarak elde edilmiştir.

## 2. SİNİRSEL STİL AKTARIMI (NEURAL STYLE TRANSFER)

SSA, temelinde derin öğrenme modellerinin öznelik çıkarımındaki başarımından faydalanarak bir görüntünün içeriğini bozmadan stil görüntüsü ile yeniden üretilmesi tekniğidir. Bu yöntem ile, verilen görüntülerden algısal olarak çekici (İlgi çeken, güzel görünen) görüntüler oluşturulabilir [2].

### 2.1. Evrimsel Sinir Ağı (Convolutional Neural Network)

Bilgisayar görmesi alanında, öznelik çıkarımı ve sınıflandırma her zaman önemli bir çalışma alanı olmuştur. ESA, görüntülerin analiz edilmesine başarıyla

uygulanmış derin ileri beslemeli yapay sinir ağıdır. Genellikle görüntü işleme problemlerinde kullanılmaktadır. Giriş katmanı, evrişim katmanı, havuzlama katmanı, tam bağlantı katmanı ve çıkış katmanı adı verilen katmanlardan oluşur. ESA, görüntü tanıma uygulamalarında kullanılmaktadır. Her bir evrişim katmanında özellik haritaları bulunabilir. Bu özellik haritaları sayesinde ağ, farklı özellikleri öğrenebilmektedir [13].

ESA, sayılan bu avantajlarının dışında önemli bir dezavantaja sahiptir. Bu dezavantaj ise, modelin eğitim ve test aşamasında başarı ile çalışabilmesi ihtiyaç duyulan yüksek bellekli güçlü GPU kartlarıdır. Son günlerde üreticilerin geliştirmiş oldukları yüksek bellekli GPU kartları sayesinde bu sorunun önüne geçilmesine çalışılmaktadır [14].

Bu çalışmada, VGG16, VGG19 ve ResNet50 ağ mimarileri kullanılmıştır. SSA sırasında sınıflandırma işlemi yapılmadığı için tam bağlantı ve çıkış katmanları modellere dahil edilmemiştir.

### 2.2. VGGNet

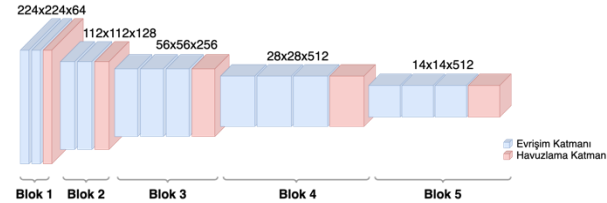
VGGNet modeli, ImageNet veri setinden alınmış bir milyondan fazla görüntüler ile eğitilmiş bir ESA modelidir [15].

Yapılan bu çalışma, VGGNet’ler arasında VGG16 ve VGG19 ağ mimarileri üzerinde gerçekleştirilmiştir. VGG modelleri diğer ağ modellerinden ayıran temel fark evrişim katmanlarının ikili, üçlü ve dördü kullanılmasıdır. Modelin her evrişim katmanı çıkışında ağırlıkları farklı olan filtreler hesaplanmaktadır ve katman sayısı arttıkça bu filtrelerde oluşan öznelikler görüntünün derinliğini simgelemektedir [16]. VGG16 ve

VGG19 modelleri evrişim katmanlarında 3x3 filtre içermektedir.

### 2.2.1. VGG16

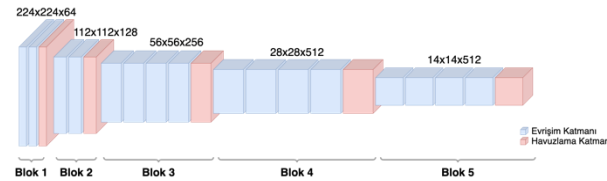
VGG16 modeli Görsel Geometri Grubu (Visual Geometry Group) tarafından ILSVRC-2014 yarışmasında daha iyi sonuçlar elde etmek için geliştirilen bir ağ modelidir [17]. Girdi katmanında yer alacak görüntü 224x224x3 boyutundadır. ImageNet veri seti ile %89 doğruluk yakalamış bir derin öğrenme ağ modelidir [18]. Şekil. 2’de sunulan VGG16 ağ mimarisinde, 13 evrişim katmanı ve 5 havuzlama katmanı ile toplamda 18 katman bulunmaktadır [17].



Şekil 2. VGG16 modeli mimarisi (VGG16 model architecture)

### 2.2.2. VGG19

VGG19 modeli, VGG16 algoritmasının akabinde geliştirilen bir ağ modelidir [17]. VGG16’da olduğu gibi giriş görüntüsü 224x224x3 boyutundadır. ImageNet veri seti ile %88 doğruluk yakalamış bir derin öğrenme ağ modelidir [18]. Şekil. 3’te sunulan VGG19 ağ mimarisinde, 16 evrişim katmanı ve 5 havuzlama katmanı ile toplamda 21 katman bulunmaktadır [17].

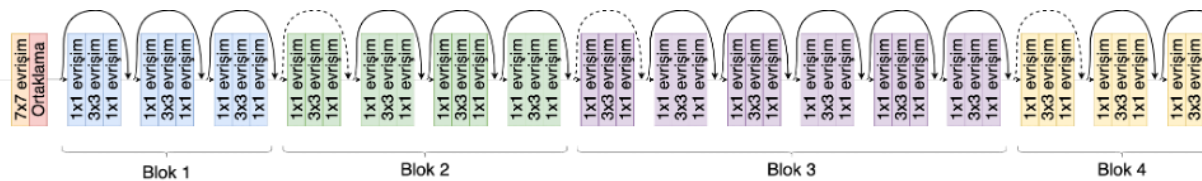


Şekil 3. VGG19 modeli mimarisi (VGG19 model architecture)

### 2.3. ResNet50

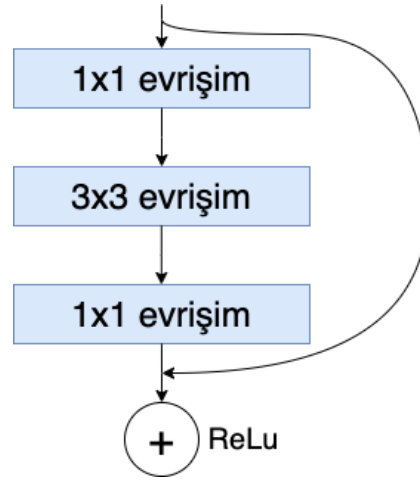
ResNet50 mimarisi, katman sayısı arttıkça eğitim başarısı artar yanlıgısını ortadan kaldırmak için çıkarılmış bir mimaridir. ResNet yapısında, “resudial” (artık) olarak adlandırılan bloklardan oluşmaktadır. Bu artık blokları sayesinde, her blok sonunda başarısız bir eğitim olsa bile en kötü ihtimalle bir önceki kadar öğrenim devam etmektedir.

ResNet, diğer mimari yapılarına göre daha fazla katman içermektedir ve aynı VGGNet’lerde olduğu gibi giriş görüntüsü 224x224x3 boyutundadır. ResNet50 mimarisi Şekil. 4’te sunulmuştur

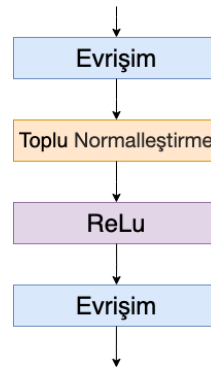


Şekil 4. ResNet50 modeli mimarisi (ResNet50 model architecture)

ResNet50 dört bloktan oluşur. Dört blokta bulunan evrişim katmanları iki adet 1x1 ve bir adet 3x3 filtre içermektedir. Şekil. 5’te ResNet50 blok yapısı ve Şekil. 6’da ResNet50 modelinde bulunan her evrişim katmanında gerçekleştirilen aşamalar sunulmuştur.



Şekil 5. ResNet50 blok yapısı (ResNet50 block structure)

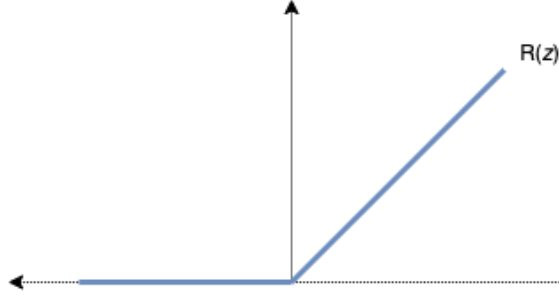


Şekil 6. ResNet50 belirtilen her evrişim katmanından sonra gerçekleştirilen işlemler (Operations performed after each convolution layer specified in ResNet50)

Şekil. 5’te gösterilen yapı, ResNet50 modelinin bir bloğunda yer alan adımları bize göstermektedir. Daha önce bahsedildiği gibi bu yapı sayesinde, başarısız bir eğitim olsa bile en kötü ihtimalle bir önceki kadar öğrenim devam ederek eğitim başarısızlığı ortadan kaldırılmıştır.

Şekil. 5’te gösterilen her bir evrişim katmanından sonra gerçekleştirilen aşamalar Şekil. 6’da sunulmuştur. Buna göre, her evrişim katmanından sonra sırasıyla toplu normalleştirme (Batch Normalization) ve ReLu

katmanları yer almaktadır. Normalleştirme işlemi, modelin daha iyi performans sağlaması için kullanılan bir yöntemdir. ReLu katmanında ise, giriş değeri sıfırın altında iken çıktı sıfır, sıfırın üzerinde iken çıktı değeri giriş değerine eşittir. Şekil. 7'de ReLu aktivasyon fonksiyonunun grafiği sunulmuştur.



Şekil 7. ReLu aktivasyon fonksiyonu (ReLU activation function)

Eş. 1'de gösterilen şekilde hesaplanmaktadır.

$$R(z) = \max(0, z) \quad (1)$$

## 2.4. Optimizasyon Algoritmaları (Optimization Algorithms)

Yapılan bu çalışmada, üç farklı optimizasyon algoritması kullanılmıştır. Bu algoritmalar Stokastik Gradyan İnişi (SGD), RMSprop ve Adam algoritmalarıdır. Optimizasyon algoritmaları, eğitimde oluşan hataları minimize etmek için kullanılmaktadır.

### 2.4.1. SGD

SGD, derin öğrenme modellerinde kademeli düşüş gerçekleştirerek, en uygun değeri bulmaya çalışan bir yöntemdir. Diğer yöntemlere göre biraz yavaş olduğu bilinmektedir. Mevcut gradyanı ( $\partial L / \partial w_t$ ), öğrenme katsayısı ( $a$ ) ile çarparak mevcut ağırlığı ( $w_t$ ) güncellemektedir [19-25]. Yeni ağırlıklar, Eş. 2'de sunulan denklem ile güncellenir.

$$w_{t+1} = w_t - a \frac{\partial L}{\partial w_t} \quad (2)$$

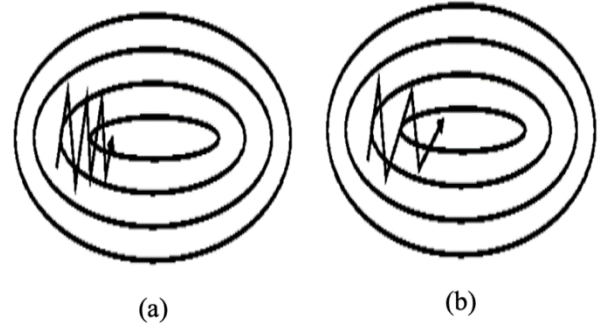
### 2.4.2. Momentum

Momentum, SGD optimizasyon algoritmasında optimum değer aranırken karşılaşılan salınımları azaltmak ve hedeflenen en uygun değeri bulma hızını artırmak için kullanılan algoritmadır [19-25]. Eş. 3,4'te belirtilen denklemler ile hesaplanmaktadır.

$$w_{t+1} = w_t - a V_t \quad (3)$$

$$V_t = \beta_1 V_{t-1} + (1 - \beta_1) \frac{\partial L}{\partial w_t} \quad (4)$$

Eş. 3,4'te görülen,  $V$ 'nin başlangıç değeri 0'dır.  $\beta$  (momentum), 0 ile 1 arasındadır ve genelde kullanılan değeri ise 0.9'dur. Daha önce kullanılan gradyanların ne kadar işleme dahil edileceğini belirleyen bir araçtır [19-25]. Şekil 8.'de momentumlu ve momentumuz SGD'nin yapmış olduğu salınımlar gösterilmiştir.



Şekil 8. (a) momentumuz SGD, (b) momentumlu SGD [25]

Şekil. 8 (a)'da SGD'nin çözüme giderken yapmış olduğu salınımların çok fazla olduğu görülmektedir. Şekil. 8 (b)'de ise SGD'de görülen salınımların azaldığı görülmektedir [25].

### 2.4.3. RMSprop

RMSprop (Karelerin ortalamasının karekökü yayılımı) temelinde, gradyanların karelerinin ortalamasının karekökü ile değerini, optimal seviyeye getirilmesini hedeflemektedir. Sabit öğrenme katsayısı problemi çözmek için önerilmiştir [19-25]. Eş. 5,6'da sunulan denklemler ile hesaplanmaktadır.

$$w_{t+1} = w_t - \frac{a}{\sqrt{S_t + \epsilon}} \frac{\partial L}{\partial w_t} \quad (5)$$

$$S_t = \beta S_{t-1} + (1 - \beta) \left[ \frac{\partial L}{\partial w_t} \right]^2 \quad (6)$$

### 2.4.4. Adam

Adaptif Moment Tahmini (Adam) RMSprop algoritmasına momentum ekleyerek, en uygun değeri bulmayı hedefleyen bir algoritmadır. Momentum yöntemindeki  $V$  ile RMSprop'daki  $S$ 'i kullanmaktadır [19-25]. Eş. 7-11'de sunulan denklemler ile güncellenmektedir.

$$w_{t+1} = w_t - \frac{a}{\sqrt{\hat{S}_t + \epsilon}} \hat{V}_t \quad (7)$$

$$\hat{V}_t = \frac{V_t}{1 - \beta_1^t} \quad (8)$$

$$\hat{S}_t = \frac{S_t}{1 - \beta_2^t} \quad (9)$$

$$V_t = \beta_1 V_{t-1} + (1 - \beta_1) \frac{\partial L}{\partial w_t} \quad (10)$$

$$S_t = \beta_2 S_{t-1} + (1 - \beta_2) \left[ \frac{\partial L}{\partial w_t} \right]^2 \quad (11)$$

Burada  $S$  ve  $V$  başlangıç olarak 0,  $a = 0.001$ ,  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$  ve  $\epsilon$  ise  $10^{-8}$  olarak tercih edilmektedir [19-25].

### 3. VERİ SETİ (DATASETS)

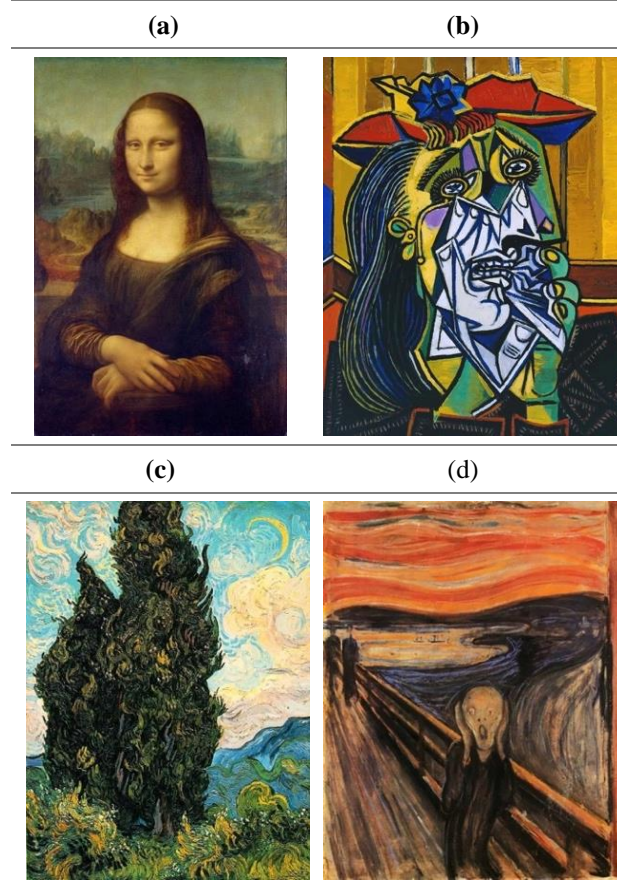
Stil aktarımı yapmak için iki görüntüye ihtiyaç duyulmaktadır. Bu görüntülerden biri “İçerik Görüntüsü”, diğeri de “Stil Görüntüsü” olarak adlandırılmıştır. Şekil. 9’da içerik görüntüsü, Şekil. 10’da stil görüntüleri sunulmaktadır.



Şekil 9. İçerik görüntüsü (Content image) [26]

Şekil. 9’da, bulutlu bir gün batımına ait iki yanı çayırılık olan asfalt bir yol fotoğrafını içerik görüntüsü olarak kullanılmıştır.

- (a) İtalyan ressam Leonardo da Vinci’nin 16. yüzyılda Francesco del Giocondo’nun karısı Lisa Gherardini’nin yağlı boya portresi olan Mona Lisa adlı tablosudur.
- (b) İspanyol ressam Pablo Picasso’nun 1937 yılında yapmış olduğu Ağlayan Kadın adlı tablosudur.
- (c) Hollandalı ressam Vincent van Gogh’un 19. yüzyılın sonlarına doğru Fransa bahçelerinden esinlenip tuvale resmettiği Selviler adlı yağlı boya tablosudur.
- (d) Norveçli ressam Edvard Munch’un 1893’te yapmış olduğu Çığlık adlı yağlı boya tablosudur.

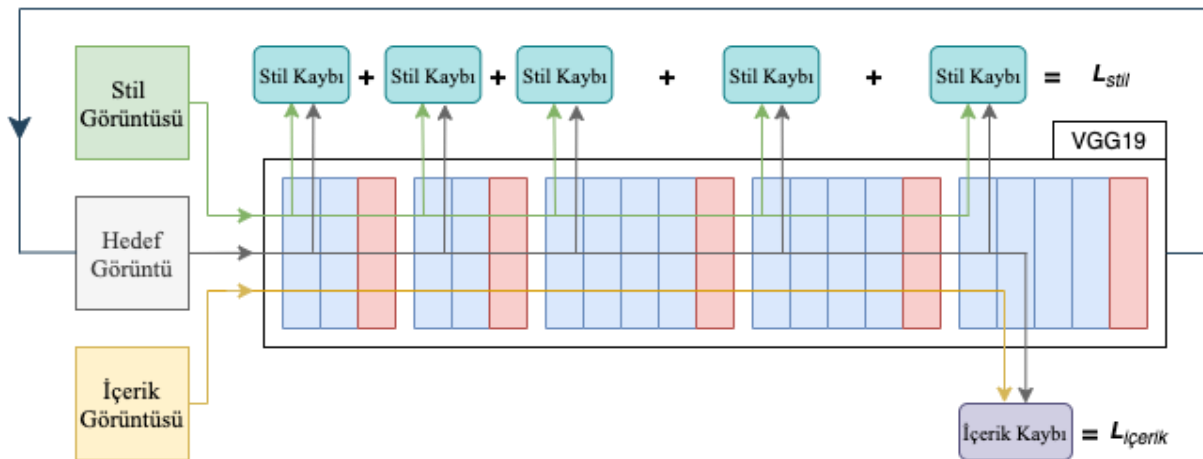


Şekil 10. Stil görüntüleri (Style images) [27-30]

Bu çalışmada, dört farklı sanatçının stilleri belirlenen içerik görüntüsüne aktarılmaya çalışılmıştır.

### 4. METODOLOJİ (METHODOLOGY)

İçerik ve stil olarak adlandırılan iki görseli önce ESA modeline verilmektedir. Ardından modele aktarılan görüntülerden önce içerik kaybı, ardından stil kaybı hesaplanmaktadır. Elde edilen bu iki kayıp değerlerini küçük sayılar ile güncelleyerek her iterasyon sonunda hedef görüntü elde edilmiştir. SSA mimarisi VGG19 mimarisi üzerinde Şekil. 11’de gösterilmiştir.



Şekil 11. Sinirsel stil aktarımının VGG19 modeli ile gerçekleştirilme yapısı (Structure of realization of neural style transfer with VGG19 model)

Mimaride görüldüğü gibi görseller VGG19 modeline aktarıldıktan sonra kayıp değerleri hesaplanır. Stil kaybı, VGGNet'ler için ilk bloklarında bulunan birinci evrişim katmanlarında, ResNet50 modeli için ise ikinci blokta bulunan ikinci evrişim katmanında, üçüncü blokta bulunan birinci evrişim katmanında, ardından gelen normalizasyon katmanında ve dördüncü bloktaki altıncı evrişim katmanında hesaplanmaktadır. İçerik kaybı, üçüncü blokta bulunan ikinci evrişim katmanında hesaplanmaktadır. Hesaplanan içerik ve stil kayıpları küçük sayılarla güncellenerek toplanır. Ardından optimizasyon algoritmaları kullanılarak gradyanlar güncellenir ve hedef görüntü üzerinde SSA gerçekleştirilir. Bu işlem belirlenen iterasyon sayısı kadar devam ettirilir.

#### 4.1. İçerik Kaybının Hesaplanması (Calculating of Content Loss)

İçerik kaybını hesaplamak için, her modelin son bloğunda yer alan ikinci evrişim katmanında hem içerik görüntüsü hemde hedef görüntü ele alınmıştır. Ardından bu iki görüntü ReLU aktivasyon fonksiyonundan geçirilmiştir. Daha sonra, bu iki aktivasyon matrisinin her ikisinin içeriği arasındaki farkı en aza indirerek, oluşturulan görüntüdeki orijinal içeriğin korunması hedeflenmiştir.  $L_{giriş}$  hesaplanması için Eş. 12 kullanılmaktadır.

$$L_{giriş}(\vec{p}, \vec{x}, l) = \frac{1}{2} \sum_{i,j} (F_{ij}^l - P_{ij}^l)^2 \quad (12)$$

Burada  $\vec{p}$  içerik görüntümüzü,  $\vec{x}$  hedef görüntümüzü,  $l$  katmanı,  $F$   $l$ . katmanında bulunan hedef görüntüyü,  $P$   $l$ . katmanında bulunan içerik görüntümüzü temsil etmektedir.

#### 4.2. Stil Kaybının Hesaplanması (Calculating of Style Loss)

Öznitelik çıkarımındaki başarılarından ötürü stil kaybının hesaplanacağı katmanlar dikkatle seçilmiştir. İçerik kaybından farklı olarak burada stil ve hedef görüntülerinin farklarını almak yerine gram matrisleri üzerindeki farka bakmamız gerekmektedir. Gram matris, görüntünün dokusu, şekli ve ağırlıkları gibi bilgilerin kıyaslanmasını sağlayacaktır. Eş. 13'te gram matrisinin nasıl hesaplanacağı gösterilmektedir.

$$G_{ij}^l = \sum_k F_{ik}^l F_{jk}^l \quad (13)$$

Denklem 13'de,  $G$  gram matrisini,  $F$ 'te iki boyutlu (2D) görüntüyü temsil etmektedir. Gram matris hesabını yapabilmek için öncelikle üç boyutlu (3D) görüntüyü 2D hale dönüştürmesi gerekmektedir. Ardından görüntüyü transpozese ile çarparak elde edilen değer bir matris vermektedir, bu matris "Gram Matris" olarak adlandırılır.

Şekil. 12'de  $F_{ij}^l$  ve  $F_{ij}^{lT}$  matrisleri gösterilmiştir.

$$F_{ij}^l = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \dots & \alpha_{1m} \\ \alpha_{21} & \alpha_{22} & \dots & \alpha_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n1} & \alpha_{n2} & \dots & \alpha_{nm} \end{bmatrix}_{n \times m} \quad F_{ij}^{lT} = \begin{bmatrix} \alpha_{11} & \alpha_{21} & \dots & \alpha_{n1} \\ \alpha_{12} & \alpha_{22} & \dots & \alpha_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{1m} & \alpha_{2m} & \dots & \alpha_{nm} \end{bmatrix}_{m \times n}$$

Şekil 12.  $F_{ij}^l$  ve  $F_{ij}^{lT}$  matrisleri ( $F_{ij}^l$  and  $F_{ij}^{lT}$  matrices)

Şekil. 12'de görüldüğü üzere  $n \times m$  'lik bir matris ile  $m \times n$  'lik matrisin çarpımı  $n \times n$  'lik bir matris üretecektir.  $n$  olarak temsil edilen aslında kanal sayısıdır,  $m$  ile temsil edilen de görüntünün boyutudur. Şekil. 12'de verilen matrisler ile gram matrisi, Çiz. 1'de gösterildiği gibi hesaplanmaktadır.



**Çizelge 1.** Gram matrisinin hesaplanması (Calculation of gram matrix)

i \ j	1	2	...	n
1	$\alpha_{11} \cdot \alpha_{11} + \alpha_{12} \cdot \alpha_{12} + \dots + \alpha_{1m} \cdot \alpha_{1m}$	$\alpha_{11} \cdot \alpha_{21} + \alpha_{12} \cdot \alpha_{22} + \dots + \alpha_{1m} \cdot \alpha_{2m}$	...	$\alpha_{11} \cdot \alpha_{n1} + \alpha_{12} \cdot \alpha_{n2} + \dots + \alpha_{1m} \cdot \alpha_{nm}$
2	$\alpha_{21} \cdot \alpha_{11} + \alpha_{22} \cdot \alpha_{12} + \dots + \alpha_{2n} \cdot \alpha_{1m}$	$\alpha_{21} \cdot \alpha_{21} + \alpha_{22} \cdot \alpha_{22} + \dots + \alpha_{2m} \cdot \alpha_{2m}$	...	$\alpha_{21} \cdot \alpha_{n1} + \alpha_{22} \cdot \alpha_{n2} + \dots + \alpha_{2m} \cdot \alpha_{nm}$
⋮	⋮	⋮	⋮	⋮
n	$\alpha_{n1} \cdot \alpha_{11} + \alpha_{n2} \cdot \alpha_{12} + \dots + \alpha_{nm} \cdot \alpha_{1m}$	$\alpha_{n1} \cdot \alpha_{21} + \alpha_{n2} \cdot \alpha_{22} + \dots + \alpha_{nm} \cdot \alpha_{2m}$	...	$\alpha_{n1} \cdot \alpha_{n1} + \alpha_{n2} \cdot \alpha_{n2} + \dots + \alpha_{nm} \cdot \alpha_{nm}$

Gram matrisini elde ettikten sonra stil kaybı artık hesaplanabilecektir. Stil kaybı, Eş. 14'te gösterildiği gibi hesaplanmaktadır.

$$E_l = \frac{1}{4N^2M^2} \sum_{i,j} (G_{ij}^l - A_{ij}^l)^2 \quad (14)$$

Her  $l$ . katmanda elde edilen kayıp  $E_l$  ile ifade edilir. Bir sonraki katmana geçmeden önce kayıp fonksiyonu bir ağırlık değeri ile güncellenir ve toplam stil kaybı, Eş. 15'de gösterildiği gibi hesaplanmaktadır.

$$L_{stil}(\vec{a}, \vec{x}) = \sum_l \omega_l E_l \quad (15)$$

#### 4.3. Toplam Kaybın Hesaplanması (Calculating of Total Loss)

Toplam kayıp değeri daha önce bulunmuş olan  $L_{giriş}$  ve  $L_{stil}$  değerlerinin  $a$  ve  $\beta$  değerleri ile güncellenmesi sonucu oluşan toplam değer ile ifade edilir. Eş. 16'da gösterildiği gibi hesaplanmaktadır.

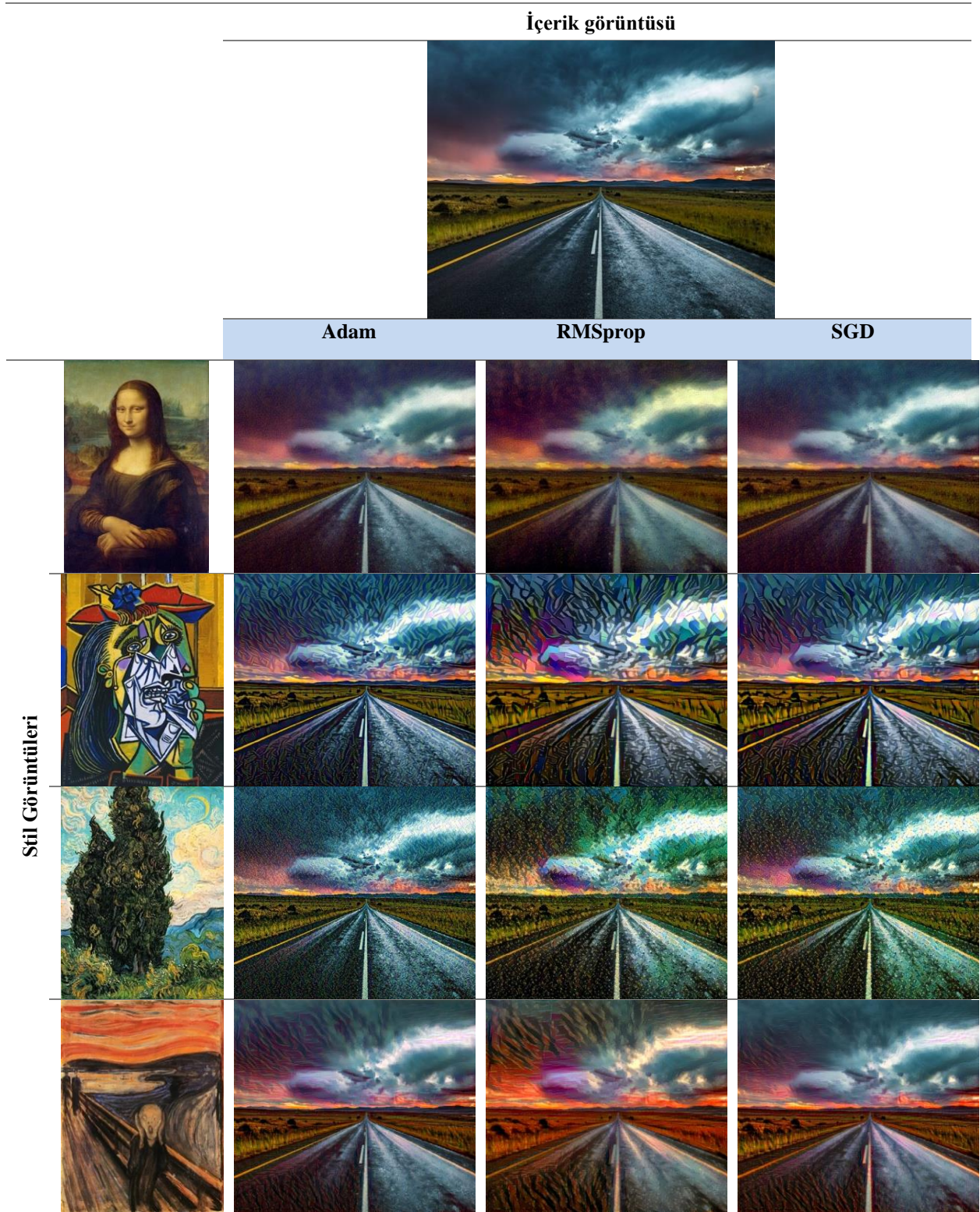
$$L_{toplam}(\vec{p}, \vec{a}, \vec{x}) = aL_{içerik}(\vec{p}, \vec{x}) + \beta L_{stil}(\vec{a}, \vec{x}) \quad (16)$$

Toplam kaybı elde ettikten sonra optimizasyon algoritmaları kullanılarak gradyanlar güncellenir. Belirlenen iterasyon sayısı sonunda hedef görüntü üzerinde SSA gerçekleştirilmiş olacaktır. Her adım sonunda kayıp değerimizin giderek düştüğü gözlemlenmiştir.

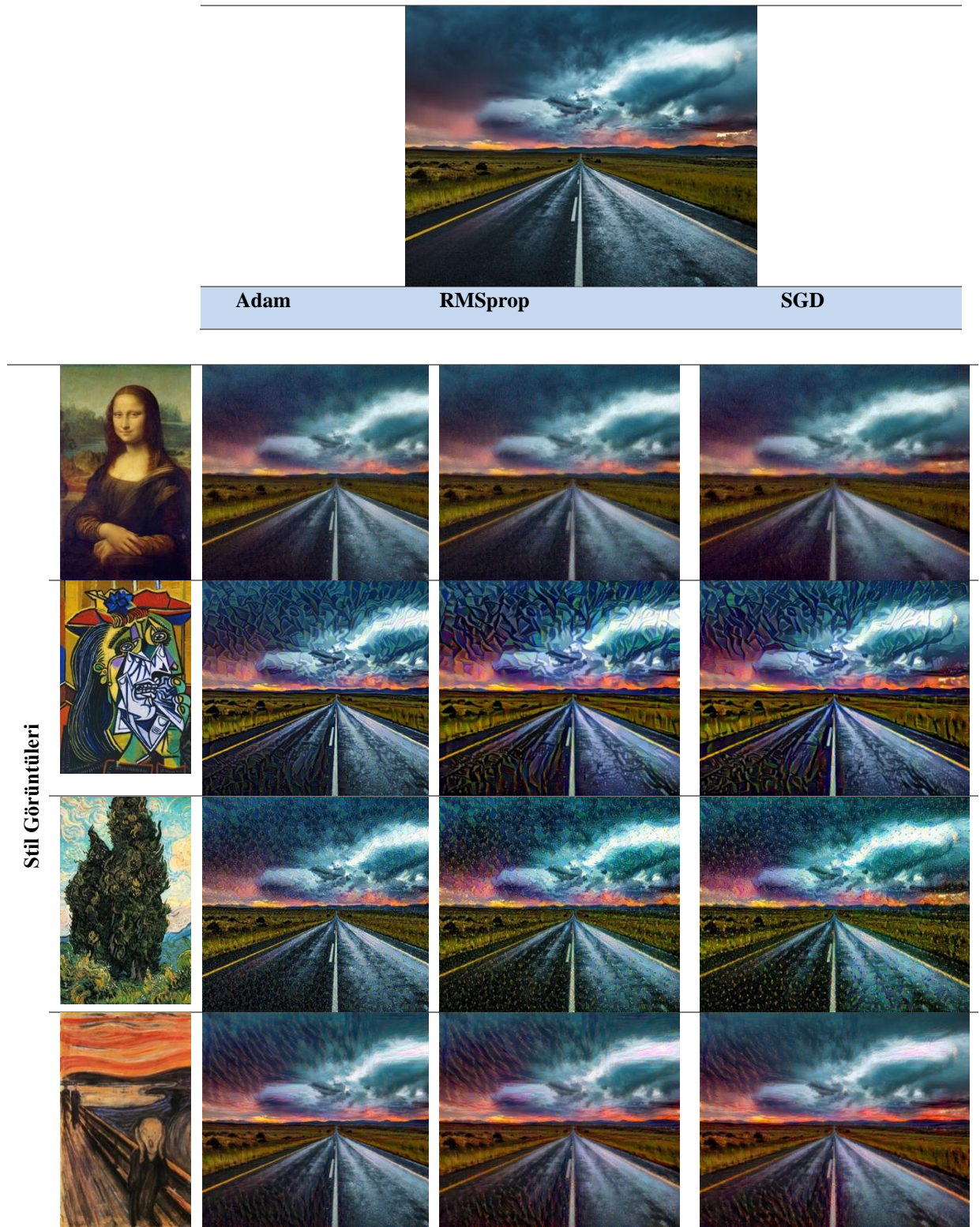
#### 5. SONUÇLAR (RESULTS)

Bu çalışmada SSA, üç farklı ESA modeli (VGG16, VGG19 ve ResNet50) ve üç farklı optimizasyon algoritması (Adam, RMSprop ve SGD) ayrı ayrı kullanılarak değerlendirilmiştir. 3000 iterasyon sonunda elde edilen hedef görüntüler karşılaştırılmıştır. Çiz. 2-4'te yeni oluşturulan hedef görüntüler sunulmuştur.

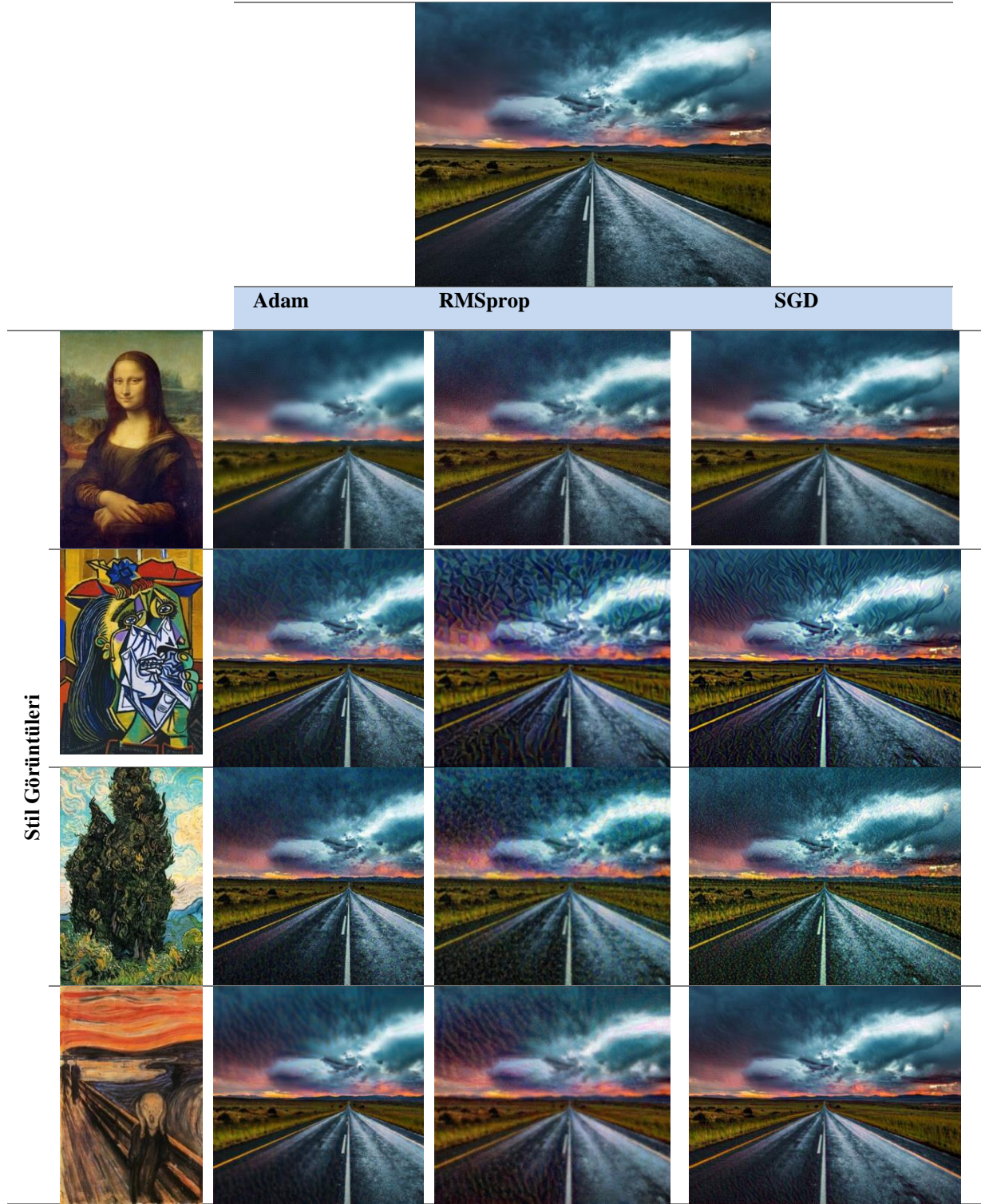
Çizelge 2. VGG16 sonuçları (VGG16 results)



**Çizelge 3.** VGG19 sonuçları (VGG19 results)



Çizelge 4. ResNet50 sonuçları (ResNet50 results)



Çiz. 2-4 görsel olarak incelendiğinde, VGG19 ağ modeli diğer modellere göre içerik görüntüsünün içeriğini bozmadan stil görüntüsünün stilini kopyalayarak hedef görüntü başarılı şekilde oluşturulmuştur. Ardından daha yakın hedef görüntü üreten ise VGG16 modeli olmuştur. ResNet50 modelinin sonuçlarına baktığımızda ise başarılı performans göstermediği görülmüştür.

Yeni üretilen hedef görüntülerin oluşturulma süreleri milisaniye (ms) cinsinden Çiz. 5-7'de sunulmuştur. Çiz. 5-7'de satırlar stil görüntülerini ifade ederken sütunlar ise optimizasyon algoritmalarını ifade etmektedir. Hedef görüntü oluşturulmasında en hızlı olan yeşil, en yavaş olan kırmızı ve ortanca hızda olan ise sarı renk ile belirtilmiştir.

**Çizelge 5.** VGG16 hedef görüntü üretimi süresi (VGG16 target image generation time)

		Optimizasyon Algoritmaları		
		Adam	RMSprop	SGD
Stil Görüntüleri	a	867.4	868.8	862.2
	b	869.5	873.8	866.9
	c	869.0	872.2	867.9
	d	870.6	871.4	870.6

**Çizelge 6.** VGG19 hedef görüntü üretimi süresi (VGG19 target image generation time)

		Optimizasyon Algoritmaları		
		Adam	RMSprop	SGD
Stil Görüntüleri	a	1020.1	1022.8	1016.3
	b	1020.2	1021.2	1019.7
	c	1018.9	1020.2	1018.7
	d	1017.5	1019.7	1016.9

**Çizelge 7.** ResNet50 hedef görüntü üretimi süresi (ResNet50 target image generation time)

		Optimizasyon Algoritmaları		
		Adam	RMSprop	SGD
Stil Görüntüleri	a	484.2	485.5	480.2
	b	486.1	487.8	488.2
	c	486.2	488.2	487.0
	d	486.2	487.8	485.1

Çiz. 5-7’de görüldüğü üzere, VGG16 ve VGG19 ağ modellerinde SGD optimizasyon algoritması daha hızlı sonuç elde etmiştir. Ardından bu hızı Adam takip etmektedir. RMSprop ise bu iki ESA modelinde en yavaş hedef görüntü üretimi başaran optimizasyon algoritması olmuştur. ResNet50 ağ modelinde ise durum diğer iki modele göre değişiklik göstermektedir.

## 6. DEĞERLENDİRME (EVALUATION)

Güncel ESA modellerinde SSA kullanılarak hedef görüntü üretilmesi gerçekleştirilmiştir. VGG19 modeli hız açısından geride kalsada hedef görüntünün görünüm başarısında oldukça iyi sonuç elde edilmiştir. Hız açısından geride kalmasının sebebi ise öznetelik çıkarımı konusunda daha hassas davranmasıdır. ResNet50 ağ modeli mimari olarak daha fazla katmana sahip olmasına

rağmen içerik ve stil görüntülerinin bilgilerini çıkarma konusunda geride kaldığı gözlemlenmiştir. Performans açısından ele alındığında VGG16 modeli daha iyi sonuç vermektedir. Hız açısından ortalarda kalmış olsada üretilen hedef görüntüler, sanki sanatçı elinden çıkmış bir tablo görünümünü vermektedir.

Üç farklı optimizasyon algoritmasının değerlendirilmesinde ise SGD optimizasyon algoritması diğerlerine göre daha gerçekçi sonuçlar vermiştir. SGD optimizasyon algoritmasını Adam algoritması izlemektedir. Adam ile üretilen görüntülerde başarılı sayılmaktadır. Lakin RMSprop optimizasyon algoritması ile elde edilen görüntüler, görsel olarak pek başarılı sonuçlar verememiştir. Genel hat itibari ile içerik görüntüsünden daha uzak sonuçlar elde edilmiştir.

## ETİK STANDARTLARIN BEYANI

### (DECLARATION OF ETHICAL STANDARDS)

Bu makalenin yazar(lar)ı çalışmalarında kullandıkları materyal ve yöntemlerin etik kurul izni ve/veya yasal-özel bir izin gerektirmediğini beyan ederler.

## YAZARLARIN KATKILARI (AUTHOR’S CONTRIBUTIONS)

**Batuhan KARADAĞ** ve **Ali ARI**: Yazılımın gerçekleştirilmesinde ve deneysel sonuçların yorumlanmasında katkı sağlamıştır.

**Müge KARADAĞ**: Metodolojinin matematiksel denklemlerinin yazımında ve yorumlanmasında katkı sunmuştur.

## ÇIKAR ÇATIŞMASI (CONFLICT OF INTEREST)

Makalenin yazar/yazarları, çalışma kapsamında herhangi bir kişisel ve finansal çıkar çatışması bulunmamaktadır.

## KAYNAKÇA (REFERENCES)

- [1] Jing Y., Yang Y., Feng Z., Ye J., ve Song M., “Neural style transfer: A review”, *IEEE Transactions on Visualization and Computer Graphics*, 26: 3365–3385, (2020).
- [2] Gatys L. A., Ecker A. S., Bethge M., Hertzmann A., ve Shechtman E., “Controlling perceptual factors in neural style transfer”, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul, (2017).
- [3] Gatys L. A., Ecker A. S. ve Bethge M., “A neural algorithm of artistic style”, *Journal of Vision*, 16: 326, (2016).
- [4] Krizhevsky A., Sutskever I. ve Hinton G. E., “Imagenet classification with deep convolutional neural networks”, *In Advances in neural information processing systems*, 1097–1105, (2012).
- [5] Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., ve Darrell, T., “DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition”, *31st International Conference on Machine Learning (ICML)*, (2014).

- [6] Cimpoi M., Maji S., ve Vedaldi A., “Deep filter banks for texture recognition and segmentation”, *International Journal of Computer Vision*, 118: 65–94, (2016).
- [7] Karayev S., Trentacoste M., Han H., Agarwala A., Darrell T., Hertzmann A., ve Winnemoeller H., “Recognizing image style”, *Proceedings of the British Machine Vision Conference*, (2014).
- [8] Kümmerer, M., Theis, L. ve Bethge M., “Deep Gaze I: Boosting Saliency Prediction with Feature Maps Trained on ImageNet”, *In ICLR*, (2015).
- [9] Chen L. C., Papandreou G., Kokkinos I., Murphy K., ve Yuille A. L., “Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs”, *In ICLR*, (2015).
- [10] Berning, M., Boergens, K. M. and Helmstaedter, M., “SegEM: Efficient Image Analysis for High-Resolution Connectomics”, *Neuron*, 87: 1193–1206 (2015).
- [11] Eigen D. ve Fergus R., “Predicting Depth, Surface Normals and Semantic Labels With a Common Multi-Scale Convolutional Architecture”, *IEEE International Conference on Computer Vision (ICCV)*, (2015).
- [12] Long J., Shelhamer E. ve Darrell T., “Fully convolutional networks for semantic segmentation”, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (2015).
- [13] Ari A. ve Hanbay D., “Deep learning based brain tumor classification and detection system”, *Turkish J. Elect. Eng. Comput. Sci.*, 26: 2275–2286, (2018).
- [14] Türkoğlu, M., Hanbay, K., Saraç Sivrikaya, I. ve Hanbay, D., “Derin Evrişimsel Sinir Ağı Kullanılarak Kayısı Hastalıklarının Sınıflandırılması”, *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, 9: 334-345, (2020).
- [15] Akılotu B. N., Kadiroğlu Z., Şengür A. and Kayaoğlu M., “Malaria Detection using both Convolutional Neural Networks and Transfer Learning Method”, *IESS*, (2019).
- [16] Kızrak, M.A., “DERİNE DAHA DERİNE: Evrişimli Sinir Ağları”, <https://ayyucekizrak.medium.com/derine-daha-derine-evrişimli-sinir-ağları-2813a2c8b2a9>, (2018), Erişim Tarihi: 10.02.2021
- [17] Doğan F. and Türkoğlu İ., “Comparison of Leaf Classification Performance of Deep Learning Algorithms”, *Sakarya University Journal of Computer and Information Sciences*, 1: 10-21, (2018).
- [18] Simonyan K. ve Zisserman A., “Very Deep Convolutional Networks For Large-scale Image Recognition”, *ICLR*, (2015).
- [19] Ruder S., “An overview of gradient descent optimization algorithms”, *CoRR*, (2016).
- [20] Akca, M.F., “Gradient Descent Nedir?”, <https://medium.com/deep-learning-turkiye/gradient-descent-nedir-3ec6afcb9900>, (2020), Erişim Tarihi: 10.02.2021
- [21] Yazan E., ve Talu M.F., “Comparison of the stochastic gradient descent based optimization techniques”, *Artificial Intelligence and Data Processing Symposium (IDAP)*, (2017).
- [22] Zeiler M. D., “ADADELTA: an adaptive learning rate method”, arXiv preprint arXiv:1212.5701, (2012).
- [23] Karim R., “10 Stochastic Gradient Descent Optimisation Algorithms + Cheat Sheet”, <https://towardsdatascience.com/10-gradient-descent-optimisation-algorithms-86989510b5e9>, (2018), Erişim Tarihi: 10.02.2021
- [24] Alpaydın, E., “Yapay Öğrenme”. *Boğaziçi Üniversitesi Yayinevi*, İstanbul, (2018).
- [25] Seyyarer E., Ayata F., Uçkan T. and Karcı A., “Derin öğrenmede kullanılan optimizasyon algoritmalarının uygulanması ve kıyaslanması”, *Anatolian Journal of Computer Sciences*, 5: 90-98 (2020).
- [26] İçerik Görüntüsü  
[https://images.wallpaperscraft.com/image/road\\_marking\\_evening\\_clouds\\_horizon\\_120298\\_1280x1024.jpg](https://images.wallpaperscraft.com/image/road_marking_evening_clouds_horizon_120298_1280x1024.jpg), (2020), Erişim Tarihi: 10.02.2021
- [27] Vinci L., “Mona Lisa” <https://www.sanatabasla.com/wp-content/uploads/2017/06/098-Mona-Lisa-Leonardo-da-Vinci.jpg>, Italy, (1503), Erişim Tarihi: 10.02.2021
- [28] Picasso P., “The Weeping Woman” <https://i.pining.com/originals/b5/cd/cb/b5cddb278146767d41a64d12cdc68fda.jpg>, (1937), Erişim Tarihi: 10.02.2021
- [29] Gogh V. V., “Cypresses” [https://upload.wikimedia.org/wikipedia/commons/thumb/1/11/Vincent\\_Van\\_Gogh\\_0016.jpg/1920px-Vincent\\_Van\\_Gogh\\_0016.jpg](https://upload.wikimedia.org/wikipedia/commons/thumb/1/11/Vincent_Van_Gogh_0016.jpg/1920px-Vincent_Van_Gogh_0016.jpg), (1889), Erişim Tarihi: 10.02.2021
- [30] Munch E., “Skrik”, [https://upload.wikimedia.org/wikipedia/commons/f/f4/The\\_Scream.jpg](https://upload.wikimedia.org/wikipedia/commons/f/f4/The_Scream.jpg), (1893), Erişim Tarihi: 10.02.2021