



GAZİ ÜNİVERSİTESİ  
MÜHENDİSLİK FAKÜLTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BM496 BİTİRME PROJESİ II  
DÖNEM SONU RAPORU

181180030- İsmail ERTAYLAN  
181180006- Büşra ARIK

Dr. Öğr. Üyesi Çağrı ŞAHİN

2023

Kelime Sayısı: 4558


## İNTİHAL BEYANI

Bu çalışmadaki tüm bilgilerin akademik kurallara ve etik davranışa uygun olarak alındığını ve sunulduğunu ve bu belgede alıntı yaptığımı belirttiğim yerler dışında sunduğum çalışmanın kendi çalışmam olduğunu, Yükseköğretim Kurumları Bilimsel Araştırma Ve Yayın Etiği Yönergesinde belirtilen bilimsel araştırma ve yayın etiği ilkelerine uygun olduğunu beyan ederim.

Numara : 181180006

Ad Soyad : Büşra Arık


Tarih : 19.05.2023

İmza : 

Numara : 181180030

Ad Soyad : İsmail Ertaylan

Tarih : 19.05.2023

İmza : 

## İÇİNDEKİLER

<b>ÖZET .....</b>	<b>1</b>
<b>1. GİRİŞ .....</b>	<b>1</b>
1.1. Kapsam .....	1
1.2. Amaç .....	1
1.3. Hedef Kitle .....	2
<b>2. LİTERATÜR TARAMASI .....</b>	<b>2</b>
2.1. Forged Face Detection Using ELA and Deep Learning Techniques .....	2
2.2. Methods of Deepfake Detection Based on Machine Learning .....	2
2.3. Exposing AI Generated Fake Face Videos by Detecting Eye Blinking .....	2
2.4. A Detection Method of Operated Fake-Images Using Robust Hashing .....	3
2.5. Detecting Fake Images on Social Media Using Machine Learning .....	4
2.6. Compare Analysis of Deepfake Image Detection Method Using Convolutional Neural Network .....	5
2.7. Fake Image Detection Using Machine Learning .....	5
2.8. Image Forgery Detection Using Machine Learning .....	6
2.9. A Landscape View of Deepfake Techniques And Detection Methods .....	7
2.10. Deep Fake Image Detection Based on Pairwise Learning .....	7
<b>3. GELİŞTİRİLEN YAKLAŞIM VE BULGULAR .....</b>	<b>8</b>
3.1. Ürün Perspektifi .....	8
3.2. Sistem Arayüzü .....	8
3.3. Ürün İşlevleri .....	8
3.4. Gerekli Durum ve Modlar .....	9
3.5. Fonksiyonel Gereksinimler .....	9
3.6. Arayüz Gereksinimleri .....	9
3.7. Veri Seti Gereksinimleri .....	9
3.8. Performans Gereksinimleri .....	9
3.9. Donanım Gereksinimleri .....	10
3.10. Yazılım Kalite Faktörleri .....	10
3.10.1. Güvenlik .....	10
3.10.2. Yeniden Kullanılabilirlik .....	10
3.10.3. Güvenilirlik .....	10
3.10.4. Test Edilebilirlik .....	10
3.10.5. Taşınabilirlik .....	10
3.10.6. Esneklik .....	10
3.10.7. Erişilebilirlik .....	10
3.11. Tasarım Görünümleri .....	10
3.12. Tasarım Bakış Açıları .....	10
3.13. Tasarım Öğeleri .....	11

3.14.	Bağlam Bakış Açısı.....	12
3.15.	Kompozisyon Bakış Açısı.....	12
3.16.	Mantıksal Bakış Açısı .....	13
3.17.	Arayüz Bakış Açısı .....	13
<b>4.</b>	<b>TESTLER, SONUÇLAR VE ÖNERİLER.....</b>	<b>15</b>
4.1.	Test Senaryoları ve Sonuçlar .....	15
4.2.	Tartışma ve Öneriler .....	16
<b>5.</b>	<b>REFERANSLAR.....</b>	<b>17</b>

**ŞEKİLLER LİSTESİ**

Şekil 1. Sistem Use-Case Diyagramı .....	8
Şekil 2. Dağıtım Diyagramı .....	12
Şekil 3. Görsel Yükleme Penceresi.....	13
Şekil 4. Metadata Analizi.....	14
Şekil 5. ELA Analizi.....	14
Şekil 6. Yapay Zekâ Analizi .....	14
Şekil 7. Test Senaryoları ve Sonuçları Tablosu .....	15

## SAHTE FOTOĞRAF ANALİZİ

### ÖZET

Sahte Fotoğraf Analizi projesi, meta veri analizi, Hata Seviyesi Analizi (ELA) ve Derin Öğrenme teknikleri kullanılarak sahte fotoğrafların tespitine odaklanmaktadır. Günümüzün dijital ortamında manipüle edilmiş görüntülerin yaygınlığı, bu tür örnekleri tanımlamak ve işaretlemek için etkili yöntemler gerektirmektedir. Dijital verilerdeki sıkıştırma yapaylıklarının analizini sağlayan Hata Seviyesi Analizi ve Derin Öğrenme algoritmalarının gücünün yanı sıra zaman damgaları, cihaz bilgileri ve yazılım bilgileri gibi görüntü meta verilerine gömülü bilgilerden yararlanan bu proje, sahte fotoğraf algılama için kapsamlı bir çözüm sunmayı amaçlıyor. Bu metodolojilerin entegrasyonu, hata seviyelerindeki tutarsızlıkları tespit ederek ve manipülasyonları gösteren kalıpları öğrenmek için derin bir sinir ağını eğiterek manipüle edilmiş görüntülerin tanımlanmasını sağlar. Kapsamlı deneyler ve değerlendirmeler yoluyla bu proje, yanlış bilgilerin yayılmasıyla mücadele etmek ve görsel içeriğin bütünlüğünü korumak için güvenilir ve sağlam bir yaklaşım sunarak sahte fotoğraf tespitinin ilerlemesine katkıda bulunmayı amaçlamaktadır.

### 1. GİRİŞ

#### 1.1. Kapsam

Sahte Fotoğraf Analizi projesi, kullanıcıların programa yüklediği görsellerin belirli metotlara göre sahte olup olmadığının analizini gerçekleştiren bir uygulama olarak geliştirilmiştir. Uygulamada programlama dili olarak Python tercih edilmiştir. Proje iki öğretim dönemini kapsayan bir projedir. Projenin ilk döneminde araştırmalar, literatür taramaları, gereksinim belirlemeleri ve planlamalar gerçekleştirilmiştir. Ayrıca kullanılacak veri kümesi olan CASIA veri kümesi belirlenmiştir. Bu veri seti üzerinde araştırmalar gerçekleştirilmiştir. Veri ön işleme operasyonları gerçekleştirilmiştir. Hata seviyesinde analiz (ELA) işlemi tamamlanmıştır. İkinci dönemde ise meta veri analizi ve CASIA veri kümesiyle eğitilen CNN bazlı modelin seçilen görsele uygulanması aracılığıyla tahmin işlemleri yapılacaktır.

#### 1.2. Amaç

Projede amaç kullanıcının girdi olarak kullandığı görsellerin derin öğrenme yöntemleri aracılığıyla sahtelik analizinin gerçekleştirilmesi ve sonucun kullanıcıya basit bir şekilde yansıtılmasıdır.

### 1.3. Hedef Kitle

Proje genel olarak bir görselin sahteliğini analiz etmek isteyen tüm kitlelere hitap etmektedir.

## 2. LİTERATÜR TARAMASI

### 2.1. Forged Face Detection Using ELA and Deep Learning Techniques

Qurat-ul-ain [1], sahte fotoğraf analizi için CNN'i kullanan bir teknik önermiştir. Başlangıçta tüm veri kümesinin (128\*128) piksel olarak yeniden boyutlandırıldığı ve normalleştirildiği pre-processing işlemi yapılmaktadır. Daha sonra ELA kullanılarak extraction işlemi yapılmıştır. İşlemden geçen görüntüler training ve test setlerine bölünüp, gerçek ve sahte görüntüleri tanımak için Deep-CNN modellerine iletilir. Bu modeller VGG-16, ResNet-50, InceptionV3 ve VGG-19'dir. VGG-16 ve 19 modelleri %91,97 ve %92,09 oranında training doğruluğu verirken, VGG-16'nın aynı veri setlerinde diğer önceden eğitilmiş modellere kıyasla daha iyi olan %64,49 test seti doğruluğu verdiği gözlemlenmiştir [1].

### 2.2. Methods of Deepfake Detection Based on Machine Learning

Makalede face swapping AI tabanlı algoritmalarla videonun/fotoğrafın değiştirilip değiştirilmediğine karar vermek için kullanılabilir indikatörler açıklanmıştır. Bunlar; çok pürüzsüz bir cilt, sentezlenen yüz ile orijinal yüz arasındaki renk uyumsuzluğu, baş pozisyonu, göz kırpma oranı, küçük hareketli parçalardaki artefaktlar ve yüz çarpıtma artefaktlarıdır. Yüz çarpıtma artefaktları, düşük çözünürlüklü yüz çıktısına sahip algoritmalar (64x64 veya 128x128) tarafından oluşturulan sahte videoların en iyi indikatörlerindendir. Model olarak DenseNet169 ile yüz çarpıtma artefakt indikatörü kullanılmıştır. Modeli değerlendirmek için Celeb-DF veri seti kullanılmıştır. Bu veri setindeki içeriklerin test sonucunda doğru çıktılar verip vermediği anlaşılması için üzerinde değişiklikler yapılmıştır. İçeriklere gürültü eklenip Gauss bulanıklığı, exponential bulanıklığı ve Rayleigh bulanıklığı test edilmiştir. En yüksek AUC değerine sahip model %60,1 ile DenseNet169 + Rayleigh blur modeli olmuştur [2].

### 2.3. Exposing AI Generated Fake Face Videos by Detecting Eye Blinking

Bu çalışmada, sinir ağları ile oluşturulan sahte yüz videolarının tespiti için göz kırpmaya dayalı bir yöntem anlatılmıştır. Spontane göz kırpma, refleks olarak göz kırpma ve istemli göz kırpma olmak üzere 3 göz kırpma türü vardır. Burada kullanılan yöntem, göz kırpma sürecindeki fenomenolojik ve zamansal düzenlilikleri yakalamak için CNN'i, RNN ile birleştiren bir derin öğrenme modeline dayanmaktadır. Deneyde yapılan işlemler pre- processing, LRCN ve model eğitimi olmak üzere 3 başlıkta toplanmaktadır. Pre-processingte, yer işareti tabanlı yüz hizalama algoritmaları kullanılarak yüz bölgeleri hizalanır ve yüz dedektörü kullanılarak yüz

işaretleri çıkarılmaktadır. LRCN aşamasında model özellik çıkartma, dizi öğrenme ve durum tahmini işlemlerinden geçirilip eğitim aşamasına gönderilmektedir. LRCN modeli, gözün açık halinin görüntü veri kümelerine göre eğitilmiştir. Daha sonra DeepFake algoritması ile oluşturulan gerçek ve sahte videolarda göz kırpmayı algılayan algoritma test edilmiştir. Deneyde CEW veri seti kullanılmıştır. LRCN metodu, EAR ve CNN metotları ile karşılaştırılarak değerlendirildiğinde CNN görüntü sınıflandırıcısı, farklı sınıfları ayırt etmek için görüntü alanında eğitilmiştir. Göz durumunu ayırt etmek için CNN modeli olarak VGG16 kullanılmıştır. EAR metodu üst ve alt kapak mesafesi ile sol ve sağ köşe noktası arasındaki mesafe arasındaki oran açısından göz durumunu analiz etmek için göz işaretlerine yanıt vermektedir. En büyük dezavantajı tamamen göz işaretlerine bağlı olmasıdır. Hesaplamalara bakıldığında LRCN %99 başarı oranıyla en iyi sonucu verirken CNN %98 ve EAR %79 oranında başarılıdır. Ama göz durumuna göre tespit konusunda CNN olağanüstü başarılı bir sonuç vermektedir. Mevcut çalışmada eksiklik olarak dinamik göz kırpma modeli dikkate alınmamaktadır. Göz kırpma, sahte yüz videolarını tespit etmede kolay bir ipucudur ve daha gelişmiş modeller, daha fazla eğitim verisi ile hala gerçekçi yanıp sönme efektleri oluşturabilmektedir. Bu nedenle, önerilen metot eksiklikler barındırmaktadır [3].

#### **2.4. A Detection Method of Operated Fake-Images Using Robust Hashing**

Bu makalede, görüntü işlemlerinden kaynaklanan bozulmalar da dahil olmak üzere sahte görüntüleri tespit etmek için bir yöntem önerilmiştir. Makalede, Robust hash yöntemi kullanılarak referans görüntülerden robust hash değerleri hesaplanır ve değerler veri tabanında saklanır. Referans görüntülere benzer şekilde robust hash yöntemi kullanılarak bir sorgu görüntüsünden robust hash değeri hesaplanmaktadır. Sorgunun hash değeri, veritabanında depolananlarla karşılaştırılıp, hash değerleri arasındaki mesafeye göre sorgu görüntüsünün gerçekliğine karar verilmektedir. Sahte görüntü algılamaya yönelik hash değerlerinin, sıkıştırma ve yeniden boyutlandırma gibi bir dizi görüntü işlemi türüne karşı yeterince sağlam olması gerekir çünkü bu tür bir işlem, görüntülerin kalitesini düşürmesine rağmen görüntülerin içeriğini değiştirmez. Bu nedenle, sorgu görüntülerine benzer görüntüleri sağlam bir şekilde almayı amaçlayan robust hashing yöntemi kullanılmıştır. Buna karşılık, robust hash yöntemi kullanılarak oluşturulan hash değerlerin, kopyala-taşı ve GAN'lar gibi sahte görüntüler oluşturmak için kullanılan manipülasyonun etkisine duyarlı olması gerekmektedir. Bu gereksinimler altında, Li et al.'s yönteminin sahte görüntü tespiti için uygun bir performansa sahip olduğu anlaşılmıştır. Deneyde Görüntü Manipülasyon Veri Kümesi, UADFV, CycleGAN ve StarGAN veri setleri kullanılmıştır. Orijinal görüntüler referans olarak kullanılmış, her deney için farklı sahte görüntü veri seti ile oluşturulan ayrı bir referans veri



seti hazırlanmıştır. Sorgu görüntüleri olarak hem orijinal görüntüler hem de sahte görüntüler kullanılmıştır. Bu deneyde, önerilen yöntem, gerçek sorgu görüntüleri olmasına rağmen, veri kümelerinden gelen sorgu görüntülerinin herhangi bir ek işlem yapılmadan doğrudan kullanıldığı Wang'ın yöntemi ve Xu'nun yöntemiyle karşılaştırılmıştır. Wang'ın yöntemi, sınıflandırıcının ProGAN kullanılarak eğitildiği GAN modelleriyle birlikte çeşitli CNN'ler tarafından oluşturulan görüntüleri tespit etmek için önerilmiştir. Önerilen yöntemin neredeyse tüm kriterler açısından daha yüksek bir doğruluğa sahip olduğu gösterilmektedir. Ayrıca, Görüntü Manipülasyonu ve UADFV veri kümeleri kullanıldığında geleneksel yöntemlerin doğruluğu oldukça azalmıştır. Bunun nedeni, geleneksel olanların CNN'ler kullanılarak oluşturulan sahte görüntüleri tespit etmeye odaklanmasıdır. Görüntü Manipülasyonu Veri Kümesi, GAN'larla oluşturulan görüntülerden oluşmaz. Ayrıca UADFV derin sahte videolardan oluşsa da veri setindeki videolar zaten video sıkıştırma etkisine sahiptir. Sıkıştırma için orijinal bir hash kod olduğunda, önerilen yöntem, geleneksel yöntemlere göre sahte görüntüleri daha iyi bir şekilde tespit edebilmektedir, eğer bir hash kodu yoksa görüntü tespiti yapılamamaktadır. Deneyde, önerilen yöntemin diğerlerinden daha iyi performans gösterdiği ve aynı zamanda birden fazla işlemi birleştirirken de iyi bir sonuç verdiği gözlemlenmiştir [4].

## 2.5. Detecting Fake Images on Social Media Using Machine Learning

Bu makalede sosyal medya üzerindeki sahte fotoğrafların makine öğrenmesiyle tespiti incelenmiştir. Araştırmacı, makine öğrenimi algoritmalarını kullanan ve CNN aracılığıyla bu tespiti sağlayan sınıflandırıcı bir model önermiştir. İlgili modelde normal görsel ve sahte görsel olmak üzere iki sınıf vardır. Araştırmacı, CNN aracılığıyla derin öğrenme tekniğini kullanmıştır. Yönteme göre önce Instagram'daki IJACSA veri setinden tespiti yapılacak görüntüler elde edilir. CNN aracılığıyla geleneksel matematiksel işlemler kullanılır ve görüntü özellikleri çıkartılıp aktivasyon fonksiyonu oluşturulur. Görüntü verilerinde doğrusallık olmadığından RELU işlevi kullanılmaktadır. Dizi boyutunu küçültmek amacıyla max pooling algoritması kullanılmaktadır. Bu aşamalardan sonra tahmin gerçekleştirilir ve bir sinir ağı ile görüntünün eşleşip eşleşmediğine karar verilir. SoftMax ile çıktının olasılıklar halinde görünmesi sağlanır. Sinir ağı eğitimi tamamlandığında da veri seti test edilir ve doğruluğun hesaplandığı değişkenleri içeren karışıklık matrisi çıkarılır. Araştırmada performans metrikleri 3 ağı göre incelenir: Alexnet, klasik CNN ve AlexnetTL. Eğitim verilerine göre ortalama sonuçlar değişse de her seferinde sıralama Alexnet- AlexnetTL-klasik CNN şeklinde olmuştur. Eğitim verilerinden yola çıkıldığında Alexnet %99,3, AlexnetTL %94 ve klasik CNN ise %83,9 oranında başarı sağlamıştır. Sonuçlardan da anlaşıldığı üzere klasik CNN modeline göre Alexnet/AlexnetTL'in kullanımı, daha başarılı çıktılar elde etmektedir [5].

## 2.6. Compare Analysis of Deepfake Image Detection Method Using Convolutional Neural Network

Bu çalışmanın amacı, derin sahte görüntüleri tespit etmenin güvenilir bir yolunu bulmak ve CNN mimarisiyle başarılı sonuçlar elde etmektir. Çalışmada, büyük bir veri setinden derin sahte görüntüleri tespit etmek için 8 CNN mimarisi kullanılmaktadır. Bunlardan üçü DenseNet mimarisi (DenseNet121, DenseNet169 ve DenseNet201), ikisi VGGNet mimarisi (VGG16, VGG19), biri ResNet50 mimarisi, biri VGGFace mimarisi ve sonuncusu ise özel bir CNN mimarisidir. CNN, özellik çıkartma ve sınıflandırma kısımlarından oluşmaktadır. Deneyde veriler, Kaggle üzerinden toplanan bir veri kümesinden elde edilir ve daha sonra evrişim katmanına gönderilir. Bu katman, girdi olarak alınan fotoğraflardan çok sayıda özellik çıkartır. Daha sonra havuzlama katmanına geçilir. Bu katmanın amacı, evrişimli özellik katmanının boyutunu en aza indirmektir. Önceki seviyelerden gelen girdiler düzleştirilir ve girdi FC katmanına gönderilir. Düzleştirilmiş vektör üzerinde matematiksel fonksiyonel işlemleri yürütmek için başka FC katmanları kullanılır. Bu aşama fotoğrafların sınıflandırma sürecini başlatmaktadır. Deney sonunda VGGFace, doğruluk, kesinlik, F1 puanı ve ROC eğrisi altındaki alan gibi ölçümlerde en iyi performansı göstermiştir. En kötü performansı ise VGG16 göstermiştir, %92 doğruluk elde etmiştir. ResNet50 de %97 doğruluk elde etmiştir. DenseNet201 ve DenseNet169, sırasıyla %96 ve %95 doğruluk elde etmiştir. En yüksek AUC puanı, %99,8 ile VGGFace mimarisi tarafından elde edilirken, en düşük AUC puan DenseNet121 mimarisi tarafından elde edilmiştir. Yazarlar tarafından önerilen model, %90 doğruluk elde etmiştir. Genele bakıldığında VGGFace en iyi performansı göstermiştir [6].

## 2.7. Fake Image Detection Using Machine Learning

Çoğu görüntü dosyası resim hakkında bilgi veren meta verilerini de barındırmaktadır. Meta veriler, dosyanın nasıl oluşturulduğu ve işlendiği ile ilgili bilgiler vermektedir. Meta veride aranması gereken bilgiler şunlardır;

1. Model ve yazılım: Bunlar, resmi oluşturan cihazı veya uygulamayı tanımlar. Kameralar, EXIF bilgisi olarak marka ve model içermektedir.
2. Görüntü boyutu: Meta veriler genellikle resmin boyutlarını kaydeder. İşlenen görüntü boyutu, meta verilerdeki diğer boyutlarla eşleşiyor mu diye kontrol edilir.
3. Zaman bilgisi: Bunlar genellikle fotoğrafın çekim ve değişim tarihlerini içerir. Zaman bilgilerinin beklenen zaman dilimine uyumu kontrol edilir.
4. Meta veri türü: Meta veri türlerinin bazıları sadece kameralar tarafından üretilirken, diğerleri yalnızca uygulamalar tarafından üretilir.

5. Açıklamalar: Gömülü ek açıklamalar içerir.
6. Eksik meta veri: Belirli meta verilerin olmaması genellikle orijinal bir fotoğrafın değil, resmin kaydedildiğini göstermektedir.
7. Değiştirilmiş meta veriler: Kasıtlı olarak meta verileri değiştirilebilmektedir.

Makalede ELA ve Meta veri analizi yöntemleri birlikte kullanılmıştır. Bir JPEG'in kalitesi kaydedildikçe düşmektedir. ELA, fotoğraftaki görüntü kalite farklarına bakarak manipülasyonu anlayabilmektedir. ELA, ImageJ kütüphanesi aracılığıyla yapılmaktadır. ImageJ, görüntüyü belirli bir sıkıştırma yüzdesiyle JPEG formatında kaydetme seçeneği sunmaktadır. Sistem önce görüntüyü kayıpsız kaydeder. Daha sonra aynı görüntü ImageJ kullanılarak %90 kaliteli görüntüye dönüştürülür. Aradaki fark, fark yöntemiyle bulunmaktadır. Elde edilen görüntü, giriş görüntüsünün gerekli ELA görüntüsüdür. Bu görüntü, arabelleğe alınmış bir görüntü olarak kaydedilir ve daha sonraki işlemler için sinir ağına gönderilir. Eğitim sırasında dizi, çok katmanlı algılayıcı ağına girdi olarak verilir ve çıktı nöronları ayarlanır. MLP, tamamen bağlı bir sinir ağıdır ve 2 çıkış nöronu vardır. İlk sahte, ikincisi gerçek görüntüyü temsil etmektedir. Verilen görüntü sahte ise, sahte nöron bire, gerçek ise sıfıra ayarlanır. Testte, görüntü dizisi giriş nöronlarına beslenir ve çıkış nöronlarının değerleri alınır. Meta veri analizinde ise önce meta verilerin çıkarılma işlemi yapılmaktadır. Sonra meta veri metni, meta veri analizi modülüne gönderilir. Bu analiz temelde bir etiket arama algoritmasıdır. Metinde Photoshop, Gimp, Adobe vb. kelimeleri arar. Sahtelik ve gerçeklik olarak adlandırılan ve gerçek ve sahteyi temsil eden iki değişken oluşturulur. Bir etiket alındığında, analiz edilir ve karşılık gelen değişken önceden tanımlanmış belirli bir ağırlıkla artırılır ve buradan alınan sonuçlarla ELA yönteminden alınan sonuçlar birleştirilir. Bu analiz, çok küçük bir işlem altında dahi tüm 'photoshopped' veya 'gimped' görüntülerde sahteliği tespit edebilmektedir ama WhatsApp, Google+ vb. üzerinden paylaşılan görsellerde hata vermektedir. Sinir ağı CASIA veri seti ile eğitilmiştir. Eğitilmiş sinir ağı, görüntüyü %83 başarı oranıyla tanıyabilmiştir [7].

## **2.8. Image Forgery Detection Using Machine Learning**

Bu yazıda, çift görüntü sıkıştırma bağlamında görüntü sahteliğini belirlemek için robust derin öğrenme tabanlı bir sistem anlatılmıştır. Bir görüntünün orijinal ve yeniden sıkıştırılmış sürümleri arasındaki fark, modeli eğitmek için kullanılmıştır. Görüntü yeniden sıkıştırıldığında, sahtelik içeriyorsa, orijinal görüntünün kaynağı ile sahte bölümün kaynağı arasındaki fark nedeniyle görüntünün sahte kısmı görüntünün geri kalanından farklı şekilde sıkıştırılmaktadır. Orijinal görüntü ile yeniden sıkıştırılmış versiyonu analiz edilir. Makalede, CNN mimarisi yaklaşımını vurgulayan, sinir ağları ve derin öğrenmeye dayalı bir görüntü

sahteciliği tespit sistemi sunulmuştur. Bu yöntem, görüntü sıkıştırmasındaki varyasyonları içeren CNN mimarisini kullanmaktadır. Modeli eğitmek için orijinal ve yeniden sıkıştırılmış görüntüler arasındaki fark kullanılmıştır. Önerilen teknik, birleştirme ve kopyala-taşı ile değişiklik yapılmış fotoğraflardaki sahteliği saptayabilmektedir. Deney sonuçları, %92,23 genel doğrulama oranı göstermektedir. Mevcut teknik, minimum 128x128 çözünürlük gerektirmektedir [8].

## 2.9. A Landscape View of Deepfake Techniques And Detection Methods

Bu makalede derin sahte çalışma ve kavramları, teknikleri ve algoritmaları incelenmiştir. Derin sahte içerikler manipülasyon derecesine göre tüm yüzün sentezi, kimlik değişikliği, özellik manipülasyonu ve ifade değişimi olarak 4 sınıfta toplanmaktadır. Tüm yüz sentezi, StyleGAN kullanarak aslında var olmayan tam yüz görsellerini üretmektedir. Bu sentez video oyunları, 3 boyutlu modelleme, fotoğrafçılık gibi çeşitli alanlarda avantajlar sağlamaktadır. Kimlik değiştirme yöntemi, FaceSwap6 ve DeepFakes7 gibi yöntemler ile yüz değişimini sağlar. Özellik manipülasyonu, bir GAN ve StarGAN yöntemi kullanılarak yüzde saç/ten rengi, sakal, bıyık, yaş, cinsiyet değişiklikleri gibi rötuşlara olanak sağlamaktadır. İfade değişimine bakıldığında ise, standart GAN mimarileri aracılığıyla bir kişinin mimiklerinin değişimi sağlanmaktadır. Çalışmalarda odaklanılan çeşitli noktalarla farklı sonuçlar elde edilebilmektedir. NIST MFC2018 veri setinde renk farklılıklarına odaklanılarak yapılan bir çalışma ile %70 AUC elde edilmiştir. Sinirsel davranışı izleyen başka bir çalışmada ise FakeSpotter yöntemleri kullanarak bir SVM eğitilmiştir. Önerilen teknikler CelebA-HQ, FFHQ veri setlerinden gerçek yüz görselleri; InterFaceGAN ve styleGAN'ın ürettiği sentetik yüz görselleri kullanarak test edilmiş ve %84,7 oranında doğruluk elde edilmiştir [9].

## 2.10. Deep Fake Image Detection Based on Pairwise Learning

Bu makalede, kontrast kaybı kullanarak sahte görüntülerin tespiti için derin öğrenmeye dayalı bir yaklaşım önerilmektedir. Sahte-gerçek görüntü çiftlerini oluşturmak için son teknoloji GAN'lar kullanılmıştır. İndirgenmiş DenseNet, girdi olarak ikili bilgiye izin vermek için iki akışlı bir ağ yapısına dönüştürülmüştür. Ardından, önerilen ortak sahte özellik ağı, görüntüler arasındaki özellikleri ayırt etmek için ikili öğrenme kullanılarak eğitilmiştir. Son olarak, sahteliğini algılamak için önerilen ortak sahte özellik ağına bir sınıflandırma katmanı eklenmiştir. Yöntemi doğrulamak için, sahte yüz ve genel görüntüleri tanımlamak için önerilen DeepFD uygulanmıştır. Deneysel sonuçlar, yöntemin yöntemlerden daha iyi performans sağladığını göstermiştir. Önerilen ikili öğrenme stratejisi, eğitilmiş sahte görüntü dedektörünün, eğitim aşamasına dahil edilmemiş olsa bile, yeni bir GAN tarafından oluşturulan sahte görüntüyü algılama yeteneğine sahip olmasını sağlayan sahte özellik öğrenmesini

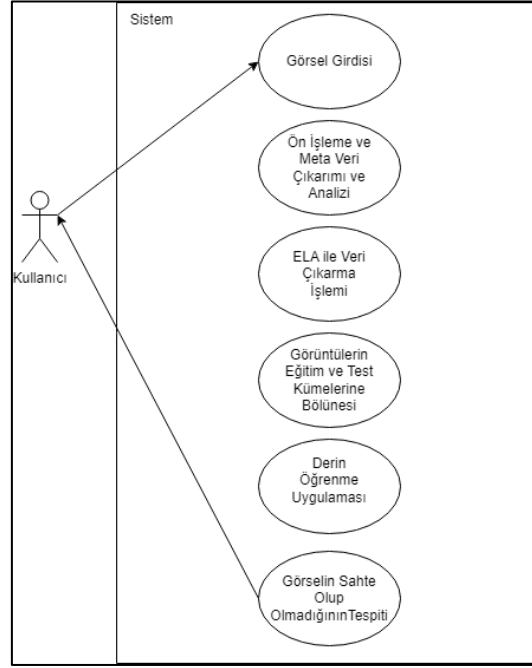
sağlamaktadır. Deneysel sonuçlar, yöntemin kesinlik ve geri çağırma oranı açısından daha başarılı olduğunu göstermektedir. Yöntemin dezavantajı, eğitim örneklerinin toplanması ile ilgilidir. Bazı sahte görüntü oluşturucuların teknik detayları açıklanmadığından eğitim örneklerinin toplanması zor olabilmektedir. Bunu aşabilmek için CFF küçük bir eğitim setinden birkaç aşamalı olarak öğrenilmelidir [10].

### 3. GELİŞTİRİLEN YAKLAŞIM VE BULGULAR

#### 3.1. Ürün Perspektifi

Sahte fotoğraf analizinde sistem tek bir kullanıcı tipine sahiptir. Bu sebeple ilgili aşamalar değişkenlik göstermez. Akış kullanıcıdan sisteme ve sistemden kullanıcıya doğrudur. Akış kullanıcının sonucu elde etmesiyle son bulur.

#### 3.2. Sistem Arayüzü



Şekil 1. Sistem Use-Case Diyagramı

#### 3.3. Ürün İşlevleri

- Uygulama açıldığında kullanıcı URL kısmına ilgili görselin URL'ini girer ya da istediği görseli bilgisayarından seçer.
- Kullanıcı görseli seçtikten sonra görsel ekranda görünür.
- Kullanıcı analiz yapmak istediği metodun butonuna tıklar.
- Sistem, ilgili metodun algoritmalarını gerçekleştirdikten sonra sonuçları ekrana yansıtır.
- Kullanıcı fotoğrafın sahte olup olmadığı bilgisine erişir.
- Kullanıcı derin öğrenme analizinin sonucunu ekranda pasta grafiği biçiminde olasılığıyla

birlikte görüntüler.

- Kullanıcı ELA analizini kıyaslamalı olarak ekranda görebilir. Kullanıcı metadata analizi sonucunu EXİF bilgileriyle birlikte ekranda görebilir.

### 3.4. Gerekli Durum ve Modlar

Sistemin değişken herhangi bir modu bulunmamaktadır. Sistem her an tek bir çalışma modundadır ve yüklenen görselin tespitini yapmaya hazır haldedir. Her tespit sonrasında başka bir görsel seçerek de tekrar tespit yapabilir duruma gelmektedir.

### 3.5. Fonksiyonel Gereksinimler

Fonksiyonel olarak sistem, yüklenen görselin belirli aşamalardan geçmesiyle hata seviyelerini analiz etmelidir. Bu analiz sonuçlarında görseller eğitim ve test aşamalarından geçmelidir. Ardından görseller, sahteliğin tespit edilebilmesi için derin öğrenme modellerine iletilmelidir. Sistem başarı oranlarını sonuç olarak vermelidir.

- Sistem kullanıcının belirlenen boyutlar(50kb-5mb) arasındaki fotoğrafları veya görselleri JPG ve PNG formatında yüklemesine izin vermelidir.
- Sistem veri kümesini ön işleme(pre-processing) aşamasından geçirmelidir.
- Sistem metadata analiziyle görselin alt bilgilerini yansıtma işlemini gerçekleştirmelidir.
- Sistem daha önceden oluşturulmuş derin öğrenme modelini ilgili görsele uygulamalıdır.
- Sistem derin öğrenme modeli sonucunda sahtelik analizini yapmalı ve belirli bir yüzde ile sonucu kullanıcıya yansıtmalıdır.

### 3.6. Arayüz Gereksinimleri

- Yüklenen görsel program üzerinde görünür hale gelmelidir.
- Yüklenen görselin belirli aşamalardan sonraki varsa değişimleri ekrana yansıtılmalıdır.
- Görselin analiz sonuçları gerek sayısal oranlarla gerekse grafiklerle kullanıcıya yansıtılmalıdır.

### 3.7. Veri Seti Gereksinimleri

- Sistem, eğitim ve test aşamalarında kullanılmak üzere uygun veri setini barındırmalıdır.
- Kullanıcının yüklediği görseller, sistem tarafından tek seferlik kullanılacak olup veri setine dahil edilmeyecektir.
- Sistem, veri seti elemanlarını önceden belirlenmiş ölçüde yeniden boyutlandırarak algoritmalarda kullanılmak üzere sabit boyutlu hale getirmelidir

### 3.8. Performans Gereksinimleri

- Sistem, fotoğrafın yüklenmesi işlemini kullanıcının da internet hızına ve yüklediği fotoğrafın boyutuna göre 15 saniyeden kısa bir süre içerisinde tamamlamalıdır.

- Sistem, yüklenen fotoğrafın analizini 15 saniyeden kısa bir süre içerisinde gerçekleştirmelidir.
- Sistem, 5 megabayttan daha fazla boyutta bir fotoğrafı kabul etmemelidir.
- Sistem 50 kilobyttan daha ufak boyutta bir fotoğrafı kabul etmemelidir.

### **3.9. Donanım Gereksinimleri**

- Sistemde gerekli işlemlerin yapılacağı bir bilgisayar ve müşterinin çıktıları görebileceği bir monitör olmalıdır.
- Bilgisayarın çalışacağı sistemde 1.8 GHz ve üzeri işlemci hızına sahip olmalıdır.
- Bilgisayarın çalışacağı sistem, en az 2 GB RAM'e sahip olmalıdır.
- Bilgisayarın çalışacağı sistem, ilgili derin öğrenme modelini de barındıracağından en az 1 GB depolama alanına sahip olmalıdır.

### **3.10. Yazılım Kalite Faktörleri**

#### **3.10.1. Güvenlik**

Sistemin herhangi bir güvenlik açığı sorunu bulunmamaktadır.

#### **3.10.2. Yeniden Kullanılabilirlik**

Sistem, her kullanıcı tarafından kolayca kullanılma yeteneğine sahip olmalıdır.

#### **3.10.3. Güvenilirlik**

Sistem tarafından üretilen çıktı algoritmanın sonuçlarına göre doğru ve tutarlı olmalıdır.

#### **3.10.4. Test Edilebilirlik**

Sistem, veri setinden elde ettiği sonuçlara dayanarak çıktıyı test edebilmelidir.

#### **3.10.5. Taşınabilirlik**

Sistem, gereksinimleri karşılayan her bilgisayarda kullanılabilir durumda olmalıdır.

#### **3.10.6. Esneklik**

Sistem, olası güncellemelere kolayca uyum sağlamalıdır.

#### **3.10.7. Erişilebilirlik**

Sistem, her an kullanıcı tarafından kullanıma açık halde olmalıdır.

### **3.11. Tasarım Görünümleri**

Projede yapısal olarak modüler bir tasarım uygulanmıştır. Nesne yönelimli programlama ilkelerine dayanarak birbirinden etkilenmeyen ve birbirleriyle uyumlu çalışabilen yapılar inşa edilmiştir. Bu sayede olası hataların çözümü kolaylaşmıştır.

### **3.12. Tasarım Bakış Açıları**

Bu bölümde, her tasarım bakış açısı kısaca açıklanmaktadır.

Bağlam bakış açısı, kullanıcılar ve sistem arasındaki ilişkileri ve etkileşimleri tanımlar.

Her işlevi temsil etmek için use-case diyagramları kullanılır.

- Kompozisyon bakış açısı, uygulamanın ana yapısını açıklar. Sistemin bileşenleri arasındaki etkileşimleri gösterir. Genel sistem mimarisi, bileşen diyagramı kullanılarak gösterilmiştir.
- Bağımlılık bakış açısı, sistemdeki bileşenlerin birbirleri arasındaki kapsamlarını belirtir.
- Mantıksal bakış açısı, temel olarak majör arabirimleri ve bu arabirimlerin arasındaki iletişimi temsil eder. Bu bakış açısı sayesinde projenin büyük parçalarına genel bir bakış yapılmış olur.
- Arayüz bakış açısı, sistemde kullanılan arayüzler açıklanır.
- Etkileşim bakış açısı, her kullanıcı işlemi için etkileşimleri ve ilişkileri tanımlar. Bu ilişkileri temsil etmek için sıra diyagramı kullanılır.

### 3.13. Tasarım Öğeleri

Tasarım öğeleri bileşen diyagramı eklenecektir

Analiz

- Tür: Sistem
- Açıklama: Bu diyagramda analiz bileşeni, sistemin temel amacını temsil eder. 5 ana bileşenden oluşmaktadır. Bunlar kullanıcı, istemci, algoritma ve sonuç bileşenleridir.

Kullanıcı

- Tür: Bileşen
- Açıklama: Sistemin hizmet edeceği temel bileşendir. Kullanıcı istemci aracılığıyla algoritmadan analiz hizmeti ister. Analizin sonucunu istemci üzerinden alır.

İstemci

- Tür: Bileşen
- Açıklama: Sistemde kullanıcı ile sunucu arasında iletişimi sağlayan köprü niteliğindeki bileşendir. Kullanıcının isteğini alıp algoritmaya, algoritmadan gelen sonucu ise kullanıcıya iletir.

Algoritma

- Tür: Bileşen
- Açıklama: Sistemde en karmaşık bileşendir. Görselin analizini sağlayan yapay zekâ unsurlarının tamamı bu bileşenin içerisindedir. Geliştiriciler en çok bu bileşenin arka planındaki algoritmalar üzerinde çalışır.

Sonuç

- Tür: Bileşen



- Açıklama: Algoritmaların gerçekleştirdiği analizlerin sonucudur. Kullanıcının ulaşmak istediği bileşendir.

### 3.14. Bağlam Bakış Açısı

Sistemde tek tip kullanıcı bulunmaktadır. Sistem, kullanıcılara sahte fotoğraf analizi hizmeti sunar. Kullanıcı bu hizmeti aynı anda tek bir görsel için kullanabilir. Kullanım durumlarıyla ilgili daha ayrıntılı bilgi, SRS dokümanında belirtilmiştir.

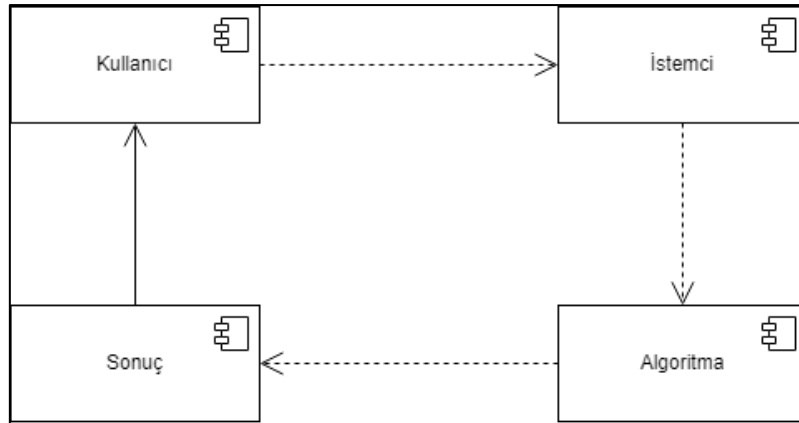
Sistemi kullanan bir aktör vardır. Öncelikle programı çalıştırmalıdır ve ardından aşağıdaki durumlara sahip olmalıdır:

- Görsel yükleme: Kullanıcı analizini gerçekleştirmek istediği görseli programa yükler.
- Metadata analizi gerçekleştirme: Kullanıcı yüklediği görselin metadata analizini gerçekleştirir ve ilgili sonuçlara erişir.
- ELA analizi gerçekleştirme: Kullanıcı yüklediği görselin hata seviyesi analizi gerçekleştirilir. Sıkıştırma yapaylıklarının sonuçlarına erişir.
- Yapay zekâ ile sahtelik analizi gerçekleştirme: Veri setinin eğitilmesi sonrasında yüklenen görselin sahtelik analizi gerçekleştirilir.
- Sahtelik çıktısı elde etme gerçekleştirme: Kullanıcı sonuç olarak grafik üzerinden yüklediği görselin sahtelik analizi sonuçlarını görmüş olur.

Sistemin örnek dillerini açıklayan use-case diyagramı gereksinim belgesinde (SRS) bulunmaktadır.

### 3.15. Kompozisyon Bakış Açısı

Bileşenler arasındaki mantıksal ilişki, kullanıcı, program ve çıktı üzerinden kurulmuştur. Kompozisyon bakış açısına göre kullanıcı girdiyi programa yükler. Program ilgili görseli aldıktan sonra belirli aşamalardan geçirir. Bir analiz uygular. Analiz sonucunda elde edilen mesaj aynı yol üzerinden kullanıcıya dönüş yapar.



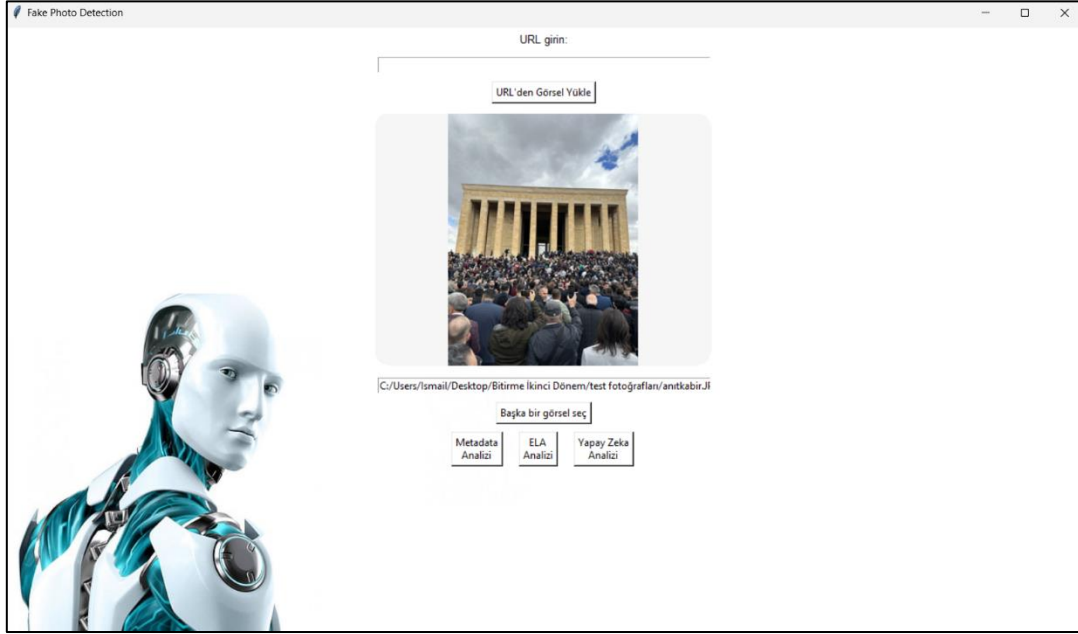
Şekil 2. Dağıtım Diyagramı

Yukarıdaki diyagram sistemin ana bileşenlerini ve bunların aralarındaki ilişkilerini göstermek amacıyla oluşturulmuştur.

### 3.16. Mantıksal Bakış Açısı

Sistem, alt sistemlere sınıflar aracılığıyla parçalanmıştır. Her sınıfın yerine getirdiği işlev aracılığıyla sistem işlevini yerine getirir.

### 3.17. Arayüz Bakış Açısı



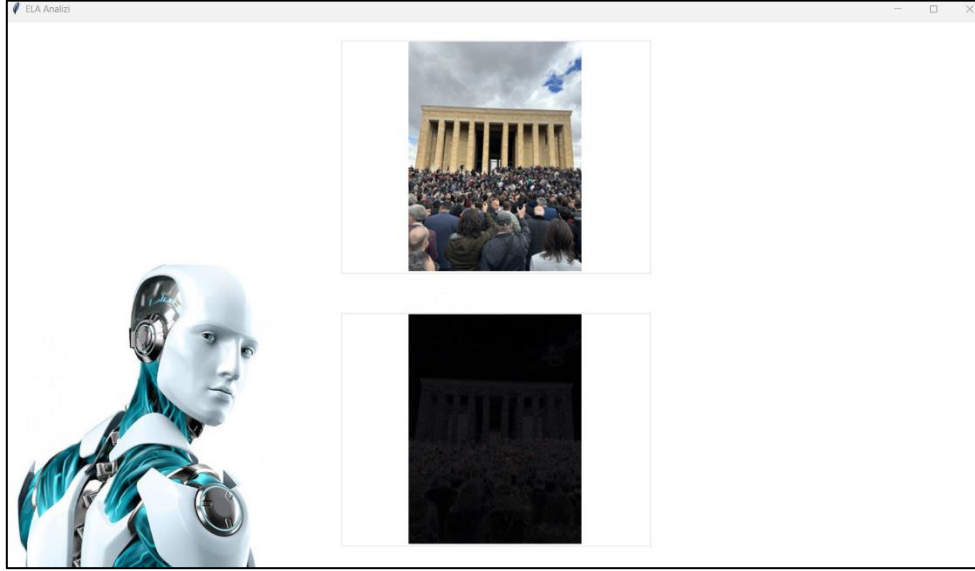
Şekil 3. Görsel Yükleme Penceresi

Yukarıdaki temsili görselde sahte fotoğraf dedektörü uygulamasının giriş penceresinin bir görüntüsü bulunmaktadır. Kullanıcı görsel yükle butonuna tıklayarak ya da URL girerek seçtiği görseli görüntüler ve yükler. Ardından analiz işlemine geçilir.



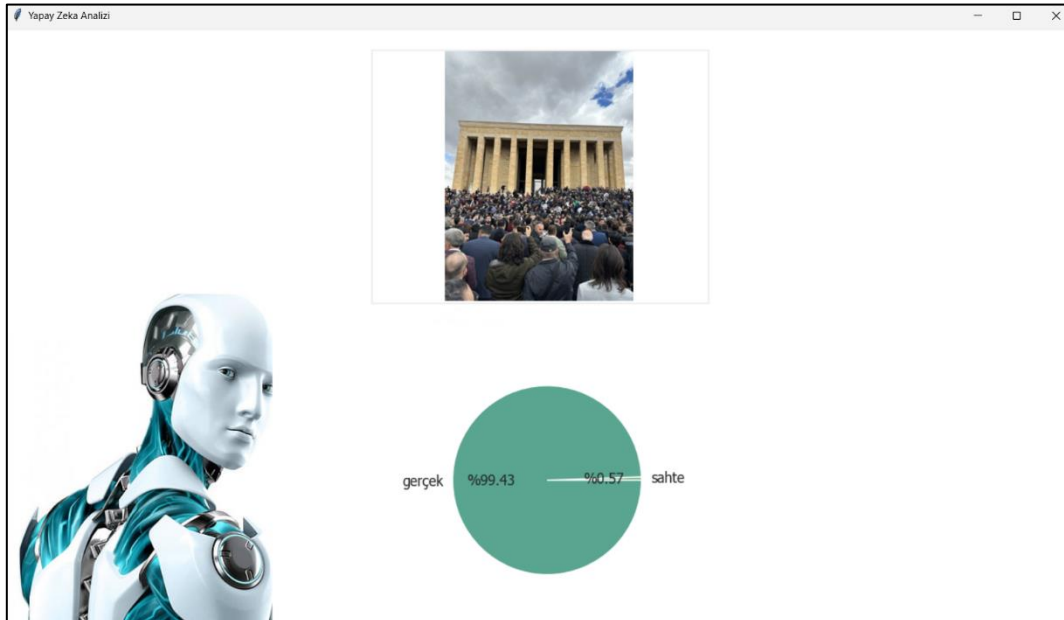
#### Şekil 4. Metadata Analizi

Seçilen görselin EXIF bilgileri kullanıcıya yansıtılır ve “software” kelimesi aranarak metadata analizi gerçekleştirilir. Kullanıcı tekrar analiz işlemi yapmak isterse önceki sayfaya dönüş yapabilir.



#### Şekil 5. ELA Analizi

Seçilen görsele ELA analizi uygulanır. Görselin ELA analizi sonrasındaki hali kullanıcıya yansıtılır. Kullanıcı tekrar analiz işlemi yapmak isterse önceki sayfaya dönüş yapabilir.



#### Şekil 6. Yapay Zekâ Analizi

Seçilen görsele daha önceden ilgili veri setiyle eğitilmiş CNN bazlı model uygulanır. Görselin yapay zekâ analizi sonrasındaki gerçeklik oranları pasta dilimi grafiği halinde kullanıcıya

yansıtılır. Kullanıcı tekrar analiz işlemi yapmak isterse önceki sayfaya dönüş yapabilir.

#### 4. TESTLER, SONUÇLAR VE ÖNERİLER

##### 4.1. Test Senaryoları ve Sonuçlar

Projenin test aşamasına gelindiğinde gerçek dünyada sahte fotoğrafların tespit edilebilmesi için birçok senaryo olabileceğinden test yöntemi olarak Blackbox metodu tercih edilmiştir. Bu metot ile testlerin teknik detaylarına girilmeden yüzeysel olarak sonuç odaklı testler gerçekleştirilmiştir. Senaryoların testlerine geçilmeden önce daha önceki dokümanlarda değinilen uygulama özelliklerinden hangilerinin test edilip, hangilerinin test edilmeyeceğine karar verilmiştir.

Toplamda biri çoklu görsel testi olmak üzere 10 farklı senaryoda sahte fotoğraf analizi uygulamasının beklentileri gerçekleştirip gerçekleştirmediği analiz edilmiştir. Çeşitli senaryolardaki bu 10 testin sonuçları aşağıdaki tabloda özetlenmiştir.

Testler/Sonuçlar	Sonuç	Başarı Durumu
Çoklu Görsel Testi	Gerçek: %73 Sahte: %86 Genel: %79,5	Başarılı
Öz Çekim Testi	Gerçek: %98,31 Sahte: %0	Başarısız
Photoshop Bulanıklık Testi	Gerçek: %99,43 Sahte: %90,57	Başarılı
Photoshop/Renk Manipülasyonu Testi	Gerçek: %98,86 Sahte: %78,14	Başarılı
Photoshop/Işık Manipülasyonu Testi	Gerçek: %97,59 Sahte: %98,71	Başarılı
İnternette Alınan Görsel Testi	Gerçek: %98,44	Başarılı
Telefondan Kaydedilen ve Üzerine Metin Eklenen Görsel Testi	Sahte: %99,67	Başarılı
Birleştirilen Görsel Testi	Sahte: %95,77	Başarılı
Video Alıntısı Testi	Gerçek: %92,95	Belirsiz
Gizli Ekleme Testi	Sahte: %9	Başarısız

Şekil 7. Test Senaryoları ve Sonuçları Tablosu

#### 4.2. Tartışma ve Öneriler

Projenin sonuna gelindiğinde teorik ve uygulama açısından birçok sonuca varılmıştır. Literatür taramasında elde edilen bilgilere göre sahtelik analizi üzerine kapsamlı olarak birçok çalışma gerçekleştirilmiştir. Bu çalışmaların bazıları manipülasyon değişikliklerinin tespiti üzerine olabildiği gibi bazıları ise DeepFake ve FaceSwap gibi daha profesyonel ve tehlikeli uygulamaları kapsamaktadır. İlgili proje görsellerdeki birçok manipülasyonun tespiti ve değişikliklerin analiz edilmesi üzerine geliştirilmiştir. İnternet üzerinden veya kullanıcının bilgisayarından seçebildiği görsellerin metadata analizi, ELA analizi ve CNN bazlı bir yapay zekâ analizi aracılığıyla sahteliği analiz edilebilmektedir. Kullanıcılara basit bir arayüz sunarak işlevin öne çıkması sağlanmıştır. Uygulamanın test aşamasında gerek yapay zekanın çoklu görseller üzerindeki başarısı gerekse çeşitli senaryolardaki olumlu sonuçlar projenin başlangıç aşamasındaki beklentileri karşılamaktadır. Eğitim dönemi sonlandıktan sonra da proje geliştiricileri, projede iyileştirmeler ve geliştirmeler üzerine çalışmaya devam etmeyi planlamaktadır.

## 5. REFERANSLAR

1. Qurat-ul-ain, Nida, N., Irtaza, A., & Ilyas, N. (2021). Forged Face Detection using ELA and Deep Learning Techniques. *2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST)*. <https://doi.org/10.1109/ibcast51254.2021.9393234>
2. Maksutov, A. A., Morozov, V. O., Lavrenov, A. A., & Smirnov, A. S. (2020). Methods of Deepfake Detection Based on Machine Learning. *2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. <https://doi.org/10.1109/eiconrus49466.2020.9039057>
3. Yuezun Li, Ming-Ching Chang, & Siwei Lyu. (2018). In Ictu Oculi: Exposing AI Generated Fake Face Videos by Detecting Eye Blinking. *ArXiv: Computer Vision and Pattern Recognition*. <http://export.arxiv.org/pdf/1806.02877>
4. Tanaka, M., Shiota, S., & Kiya, H. (2021). A Detection Method of Operated Fake-Images Using Robust Hashing. *Journal of Imaging*, 7(8), 134. <https://doi.org/10.3390/jimaging7080134>
5. AlShariah, N. M., & Khader, A. (2019). Detecting Fake Images on Social Media using Machine Learning. *International Journal of Advanced Computer Science and Applications*, 10(12). <https://doi.org/10.14569/ijacsa.2019.0101224>
6. Shad, H. S., Rizvee, M. M., Roza, N. T., Hoq, S. M. A., Monirujjaman Khan, M., Singh, A., Zaguia, A., & Bourouis, S. (2021). Comparative Analysis of Deepfake Image Detection Method Using Convolutional Neural Network. *Computational Intelligence and Neuroscience*, 2021, 1–18. <https://doi.org/10.1155/2021/3111676>
7. Villan, M., Kuruvilla, A., Paul, J., & Elias, E. (2017). Fake Image Detection Using Machine Learning. *IRACST-International Journal of Computer Science and Information Technology & Security (IJSITS)*.
8. Ali, S. S., Ganapathi, I. I., Vu, N. S., Ali, S. D., Saxena, N., & Werghi, N. (2022). Image Forgery Detection Using Deep Learning by Recompressing Images. *Electronics*, 11(3), 403. <https://doi.org/10.3390/electronics11030403>
9. Ahmed S Abdulreda, & Ahmed J. Obaid. (2022). A landscape view of deepfake techniques and detection methods. *International Journal of Nonlinear Analysis and Applications*, 13(1), 745–755. <https://doi.org/10.22075/ijnaa.2022.5580>
10. Hsu, C. C., Zhuang, Y. X., & Lee, C. Y. (2020). Deep Fake Image Detection Based on Pairwise Learning. *Applied Sciences*, 10(1), 370. <https://doi.org/10.3390/app10010370>