T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BULUT TABANLI OTONOM SÜRÜŞ SISTEMLERINDE GÖRÜNTÜ İŞLEME IÇIN TAM HOMOMORFIK ŞIFRELEME YAKLAŞIMI

DOKTORA TEZİ

Kamran SAEED

Bilgisayar Mühendisliği Anabilim Dalı

ARALIK 2025

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BULUT TABANLI OTONOM SÜRÜŞ SISTEMLERINDE GÖRÜNTÜ İŞLEME IÇIN TAM HOMOMORFIK ŞIFRELEME YAKLAŞIMI

DOKTORA TEZİ

Kamran SAEED

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Doç. Dr. Muhammed Fatih ADAK

ARALIK 2025

Kamran Saeed tarafından hazırlanan "Bulut Tabanlı Otonom Sürüş Sistemlerinde Görüntü İşleme Için Tam Homomorfik Şifreleme Yaklaşımı" adlı tez çalışması 04.12.2025 tarihinde aşağıdaki jüri tarafından oy birliği/oy çokluğu ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Doktora tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı :	**Doç. Dr. M. Fatih ADAK** Sakarya Üniversitesi	.....
Jüri Üyesi :	**Prof. Dr. Ünal ÇAVUŞOĞLU** Sakarya Üniversitesi	.....
Jüri Üyesi :	**Dr. Öğr. Üyesi Musa BALTA** Sakarya Üniversitesi	.....
Jüri Üyesi :	**Doç. Dr. Süleyman EKEN** Kocaeli Üniversitesi	.....
Jüri Üyesi :	**Dr. Öğr. Üyesi M. Ali Nur ÖZ** Sakarya Uygulamalı Bilimler Üniversitesi	.....

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Sakarya Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğine ve Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesine uygun olarak hazırlamış olduğum "BULUT TABANLI OTONOM SÜRÜŞ SISTEMLERINDE GÖRÜNTÜ İŞLEME IÇIN TAM HOMOMORFIK ŞIFRELEME YAKLAŞIMI" başlıklı tezin bana ait, özgün bir çalışma olduğunu; çalışmamın tüm aşamalarında yukarıda belirtilen yönetmelik ve yönergeye uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, bu tezi başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve 20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince Sakarya Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Enstitü tarafından belirlenmiş ölçütlere uygun rapor alındığını, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun ortaya çıkması halinde doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi beyan ederim.

(04/12/2025).

Kamran Saeed

Sevgili eşime ve çocuklarıma sonsuz sabırları için, anne ve babama sarsılmaz destekleri için, öğretmenlerime ise merakımı ve bilgimi besledikleri için teşekkür ederim.

TEŞEKKÜR

Tez danışmanım **Doç. Dr. M. Fatih Adak**'a, bu araştırma süresince sunduğu paha biçilmez rehberlik, teşvik ve sürekli desteği için en derin şükranlarımı sunarım. Onun uzmanlığı ve danışmanlığı, bu tezin hem yönünün hem de kalitesinin şekillenmesinde belirleyici olmuştur.

Ayrıca, değerli jüri üyeleri **Prof. Dr. Ünal Çavuşoğlu** ve **Dr. Öğr. Üyesi Musa Balta**'ya, bu çalışmayı değerlendirirken göstermiş oldukları yapıcı eleştiriler, kıymetli öneriler ve ayırdıkları değerli zamanları için içten teşekkürlerimi sunarım. Sakarya'daki kalışım boyunca bana her anlamda destek olan **Müdür Paşa Odabaş'a** da özel olarak teşekkür ederim. Kendilerinin yardım ve kolaylaştırıcı katkıları, bu sürecin çok daha sorunsuz ilerlemesini sağlamıştır.

Sevgili eşim **Dr. Samiah Ejaz**'a ve çocuklarım **Aayra Kamran** ile **Muhammad Bin Kamran**'a, bu yolculuk boyunca bana güç veren sabırları, sevgileri ve teşvikleri için kalpten teşekkür ederim. Babam **Muhammad Saeed**'e, annem **Shamim Akhter**'e, kayınpederim **Sardar Ijaz Afzal Khan**'a ve kayınvalidem **Razia Sultana**'ya sonsuz duaları ve destekleri için derin minnettarlığımı sunarım. Kardeşlerim **Usman Saeed** ve **Rizwan Saeed**'e, ablalarım **Iram Saeed** ve **Maryam Saeed**'e her zaman yanımda oldukları için ayrıca teşekkür ederim.

Dostlarım **M. Awais**, **Sajan Ali**, **Hassan Niazi** ve meslektaşım **M. Kashif Naseer**'e verdikleri destek, dostluk ve teşvikleri için özel olarak teşekkür etmek isterim.

Bu araştırmanın yürütülmesi için gerekli imkânları sağlayan **Sakarya Üniversitesi, Bilgisayar ve Bilişim Mühendisliği Bölümü laboratuvarına** ve destekleri için **Sakarya Üniversitesi Bilimsel Araştırma Projeleri Komisyonu (BAPK)**'na **2024-26-61-50 numaralı proje** kapsamında teşekkür ederim.

Son olarak, bu tezin başarıyla tamamlanmasına doğrudan ya da dolaylı olarak katkıda bulunan tüm dostlarıma, meslektaşlarıma ve kurumlara şükranlarımı sunarım.

Kamran Saeed

İÇİNDEKİLER

Sayfa

KISALTMALAR

AES : Gelişmiş Şifreleme Standardı (Advanced Encryption Standard)

ASPP : Atrous Uzamsal Piramit Havuzu

BFV : Brakerski, Fan ve Vercauteren

BGV : Brakerski Gentry Vaikuntanathan

BOF : Serbestler Çantası (Bag of Freebies)

BOS : Özelikler Çantası (Bag of Specials)

CNNs : Evrişimsel Sinir Ağları

CKKS : Cheon Kim Kim Song

CQA : Cochrane Kalite Değerlendirmesi

CSP : Bulut Servis Sağlayıcı (Cloud Service Provider)

CSP : Bulut Servis Sağlayıcı (Cloud Service Provider)

CBIR : İçerik Tabanlı Görüntü Getirme

CIoU : Tam Kesişim Üzerinden Birleşim

DPM : Deforme Edilebilir Parça Modeli

DES : Veri Şifreleme Standardı

DCT : Ayrık Kosinüs Dönüşümü

DSS : Dijital İmza Standardı

DIoU : Mesafe Kesişim Üzerinden Birleşim

DFL : Dağılım Odaklı Kayıp (Distribution Focal Loss)

E-ELAN : Genişletilmiş Verimli Katman Toplama Ağı

EPCBIR : Verimli ve Mahremiyet-Duyarlı İçerik Tabanlı Görüntü Getirme

EIoU : Verimli Kesişim Üzerinden Birleşim

ECC : Eliptik Eğri Kriptografisi

FPS : Saniyedeki Kare Sayısı (Frames Per Second)

FHE : Tam Homomorfik Şifreleme

FPGA : Saha Programlanabilir Kapı Dizileri (Field Programmable Gate Arrays)

FPN : Özellik Piramidi Ağı (Feature Pyramid Network)

GPU : Grafik İşlem Birimi (Graphics Processing Unit)

GIoU : Genelleştirilmiş Kesişim Üzerinden Birleşim

IoT : Nesnelerin İnterneti

IoV : Araçların İnterneti

IoU : Kesişim Üzerinden Birleşim (Intersection over Union)

LBP : Doğrusal İkili Örüntü

LWE : Hatalar ile Öğrenme (Learning With Errors)

MD5 : Mesaj Özeti Algoritması 5

mAP : Ortalama Doğruluk Oranı (mean Average Precision)

NC : Ağ Kodlama (Network Coding)

PHE : Kısmi Homomorfik Şifreleme

PRISMA : Sistematik İncelemeler ve Meta Analizler için Tercih Edilen Raporlama Öğeleri

PPSIFT : Ölçekten Bağımsız Özellik Dönüşümünü Korumalı (Preservin SIFT)

PIC : Mahremiyet Korumalı Görüntü Arama Bileşeni

PPCBIR : Mahremiyet Korumalı Kümeleme Tabanlı Görüntü Getirme

PQC : Kuantum Sonrası Kriptografi

PGI : Programlanabilir Gradyan Bilgisi

PAN : Yol Toplama Ağı (Path Aggregation Network)

PETs : Mahremiyet Artırıcı Teknolojiler

RSA : Rivest Shamir Adleman

RC4 : Rivest Şifreleme 4

RFB : Artık Filtre Bankası

SHA-1 : Güvenli Hash Algoritması 1

SPP : Uzamsal Piramit Havuzu (Spatial Pyramid Pooling)

SPP : Basit Ön İşleme (Simple Pre-Processing)

SIoU : Ölçeklenebilir Kesişim Üzerinden Birleşim

SHE : Kısmen Homomorfik Şifreleme

SIFT : Ölçekten Bağımsız Özellik Dönüşümü

TDHPPIR : İki Aşamalı Derin Hashing Mahremiyet Korumalı Görüntü Getirme

VHE : Doğrulanabilir Homomorfik Şifreleme

YOLOv9c : You Only Look Once sürüm 9

V2X : Araçtan-Her Şeye (Vehicle-to-everything)

VANETs : Araç Ad-hoc Ağları (Vehicular Ad-hoc Networks)

YZ : Yapay Zekâ

SİMGELER

$\mathbf{X}$	: Girdi vektörü
$\mathbf{E_x}$	: Şifrelenmiş girdi vektörü
$\mathbf{E_i}$	: Vektörün şifrelenmiş elemanı
$\mathbf{K_s}$	: Gizli anahtar
$\mathbf{K_p}$	: Açık anahtar
$\mathbf{M}$	: Düz metin mesajı / Görüntü matrisi
$\mathbf{c_0, c_1}$	: Şifreli metin bileşenleri
$\mathbf{u}$	: Rastgele polinom (şifreleme)
$\mathbf{e, e_1, e_2}$	: Gauss dağılımından örneklenen hata terimleri
$\mathbf{a}$	: Uniform dağılımdan örneklenen polinom
$\mathbf{\Delta}$	: Şifrelemede ölçekleme faktörü
$\mathbf{\rho^2(b, b\_gt)}$	: Tahmin edilen ve gerçek kutu merkezleri arasındaki kareli Öklid mesafesi
$\mathbf{c^2}$	: En küçük kapsayıcı kutunun köşegen uzunluğu
$\mathbf{\alpha}$	: CIoU kaybındaki ağırlık faktörü
$\mathbf{v}$	: CIoU'daki en-boy oranı tutarlılık terimi
$\mathbf{F(i,j)}$	: (i,j) noktasındaki öznitelik haritası çıktısı
$\mathbf{I(i,j)}$	: (i,j) noktasındaki giriş görüntü pikseli
$\mathbf{K(m,n)}$	: Evrişim çekirdeği ağırlığı
Box	: Sınır kutusu (x,y,w,h)
$\mathbf{t_x, t_\gamma}$	: Tahmin edilen merkez koordinatları ofseti
$\mathbf{c_x, c_\gamma}$	: Izgaradaki hücre ofseti
t_w, t_h	: Tahmin edilen genişlik ve yükseklik ofseti
p_w, p_h	: Ön genişlik ve yükseklik (çapa kutusu)
$\mathbf{\hat{x}, \hat{y}}$	: Tahmin edilen sınır kutusu merkez koordinatları
$\mathbf{\hat{w}, \hat{h}}$	: Tahmin edilen sınır kutusu genişlik ve yüksekliği
$\mathbf{C}$	: Güven puanı
$\mathbf{\hat{C}}$	: Tahmin edilen güven puanı
P(Object)	: Nesne varlığı olasılığı
$\mathbf{P(Class_i)}$	: i sınıfının olasılığı
λ_coord	: Yerelleştirme kaybı ağırlığı

λ_noobj : Nesne olmayan güven kaybı ağırlığı

IoU : Kesişim-Birleşim oranı

B_pred : Tahmin edilen sınır kutusu

B_truth : Gerçek sınır kutusu

⌊·⌋ : Aşağı yuvarlama fonksiyonu

⌊·⌉ : En yakın tam sayıya yuvarlama

[·]y : y modunda indirgeme

Img_v : Görüntü vektörü

O : Nesne görüntüsü

img_v_i : Görüntü I'nin düzleştirilmiş vektörü

img_v_o : Nesne O'nun düzleştirilmiş vektörü

img_v(o_rot) : Döndürülmüş nesne vektörü

img_v(o_rep) : Görüntü boyutuna uyacak şekilde tekrarlanan nesne vektörü

img_v(pt_l) : Tespit edilen sıfırlarla desen konum vektörü

img_v() : Nesnenin tespit edildiği ve diğer değerlerin 266'ya ayarlandığı yeni görüntü vektörü

T : Üçlü sıfır sayısı

T_Zero_C : Şifre çözülmüş sonuçta bulunan üçlü sıfırların sayısı

FRV : İlk satır vektörü (bilinmeyen nesne aramasında kullanılır)

FSV : Son çıkarım vektörü (doğrulama adımı)

TABLO LİSTESİ

Sayfa

ŞEKİL LİSTESİ

Sayfa

BULUT TABANLI OTONOM SÜRÜŞ SISTEMLERINDE GÖRÜNTÜ İŞLEME IÇIN TAM HOMOMORFIK ŞIFRELEME YAKLAŞIMI

ÖZET

Akıllı ulaşım sistemlerinin hızlı ve ileri düzeydeki gelişimi, otonom araçları günümüz yeniliklerinin bir parçası haline getirmiştir. Bu araçlar, çevrelerini sürekli olarak taramak ve anlık gelişen olaylara göre karar vermek için sensörler, kameralar ve iletişim modüllerinden oluşan bir ağa bağımlıdır. Bu sensörlerin taşıdığı bilgiler yolcu bilgileri, yol koşulları ve konum verilerini içermesi nedeniyle oldukça hassastır. Araçlarda sınırlı depolama alanı bulunduğundan, bu verilerin bulut depolama gibi harici bir ortama aktarılması gerekmektedir. Ancak veriler buluta gönderilip işlenmek üzere saklandığında, yetkisiz erişim, içeriden gelen tehditler ve çıkarıma dayalı saldırılar gibi risklere açık hale gelmektedir.

Üçüncü taraf depolama alanlarında saklanan bu verilerin korunması için kriptografi teknikleri kullanılabilir, ancak bu durumda verilerin geri çağrılması (retrieval) sorunu ortaya çıkmaktadır. Araç üzerine monte edilen kameralar tarafından yakalanan tüm görüntü tabanlı verilerin saklanması genellikle başka bir kaynağı gerektirir; çünkü bu verilerin tamamının araç depolamasında tutulması imkânsızdır. Bu çalışmada, verilerin bulutta güvenli bir şekilde saklanması ve şifre çözme yapılmadan geri çağrılabilmesi için homomorfik şifreleme tekniği önerilmiş ve doğrulanmıştır. Bu kapsamda Brakerski–Fan–Vercauteren (BFV) tam homomorfik şifreleme (FHE) şeması kullanılarak görüntü tabanlı veri setleri piksellere dönüştürülmüş ve iletimden önce şifrelenmiştir. Geleneksel şifreleme yöntemlerinde olduğu gibi bilgilerin çözülüp analiz edilmesine gerek kalmadan, bu yöntemde süreç boyunca gizlilik korunmaktadır. Şifrelenmiş veriler bulutta işlenebilmekte ve güvenliğini korumakta, dolayısıyla ele geçirilse bile saldırgan için kullanılabilir olmamaktadır.

Bu çalışmanın en önemli katkılarından biri, görüntülerin çok yönlü ve güvenli bir şekilde geri çağrılmasını sağlamak amacıyla üç farklı algılama tekniğinin bir araya getirilmesidir. Benzerlik yüzdesine dayalı yöntem, şifrelenmiş görüntüler arasında yapısal benzerliklerin karşılaştırılmasını ve eşleştirilmesini sağlayarak benzer veya yinelenen görüntülerin tespitinde etkili olmaktadır. Bilinen nesne tabanlı yaklaşım, bulutta önceden şifrelenmiş nesne desenlerinden yararlanarak belirli kategori sorgularında hızlı ve kesin çözümler sunmaktadır. Bilinmeyen nesne tabanlı yaklaşım ise yeni, daha önce görülmemiş nesnelerin tanımlanabilmesi için nesne şekillerinin oluşturulup şifrelenmesiyle sistemin esnekliğini artırmaktadır. Bu kombinasyon sayesinde çerçeve hem rutin sorgularda hem de beklenmedik yol durumlarında etkili olmaktadır.

Bununla birlikte, gerçek zamanlı nesne tespitinde yüksek doğruluk ve verimlilik sağlamak için YOLOv9c modeli uygulanmıştır. Model, araçların, yayaların, trafik işaretlerinin ve diğer kritik trafik unsurlarının yerini belirlemede %90 ortalama hassasiyet elde etmiştir. Kriptografik şema ile YOLOv9c'nin birleşimi, gizlilik, doğruluk ve performans arasında bir denge sağlamıştır. Deneysel sonuçlar, bulut tabanlı görüntü işlemede homomorfik şifrelemenin uygulanabilir olduğunu

kanıtlamıştır. Şifreleme, daha fazla hesaplama ve depolama maliyetine neden olsa da, bu maliyetler mevcut bulut ortamlarında karşılanabilir düzeydedir. Ayrıca, şifrelenmiş görüntülerde yapısal bütünlük korunduğundan, şifreleme tespit sonuçlarını bozucu bir etki yaratmamıştır.

Çalışmada dikkat çekilen bir diğer önemli nokta, kötü hava koşulları gibi zorlu ortamlarda da çerçevenin dayanıklılığını korumasıdır. Benzerlik tabanlı tekniklerin doğruluğu bu durumlarda düşse de sistem işlevselliğini sürdürebilmiştir. Sonuç olarak, bu çalışma homomorfik şifreleme ve derin öğrenmenin, otonom araçlar için bulut depolarda güvenli ve gizliliği koruyan bir arama kabiliyeti sağlayabileceğini göstermektedir. YOLOv9c modeli, karmaşık araç veri setlerinde nesneleri yüksek doğrulukla tespit ederek 0.901 F1 skoru, 0.948 hassasiyet ve 0.86 duyarlılık elde etmektedir. Önerilen BFV tabanlı şifreleme yöntemi, önceki çalışmalarda bildirilen sürelere kıyasla çok daha düşük işlem yüküne sahiptir; şifreleme süresi yalnızca 0.18 saniye, çözme süresi ise 0.09 saniyedir. Şifreli görüntü arama performansı da oldukça yüksektir; 100 görüntülük bir veri setinde arama işlemi 14.04 saniyede tamamlanmakta olup, bu süre literatürde rapor edilen sürelerden önemli ölçüde daha kısadır. Bilinen nesne tespiti, 50 nesneye kadar 4.06 saniyelik arama süresi ile ölçeklenebilirlik göstermektedir. Bilinmeyen nesne tespitinde ise 1.17 KB boyutundaki nesneler, 1000 görüntü içeren bir veri setinde 10 saat 29 dakika içinde bulunabilmektedir.

Önerilen bu sistem, yalnızca akıllı ulaşım sistemlerine değil, aynı zamanda büyük miktarda verinin saklanmasını gerektiren tüm sistemlere, güvenli ve düşük maliyetli üçüncü taraf depolama çözümleri sunmaktadır. Araştırmanın bulguları, veri gizliliği ve güvenliğinin kritik önem taşıdığı sağlık, finans, kamu, gözetim ve akıllı şehirler gibi diğer sektörler için de değerlidir.

A FULLY HOMOMORPHIC ENCRYPTION APPROACH TO IMAGE PROCESSING IN CLOUD BASED AUTONOMOUS DRIVING SYSTEMS

SUMMARY

Nowadays there is an advancement of smart transportation which has boosted the growth of autonomous vehicles. Autonomous vehicles use sensors, cameras, and networked Internet of Things (IoT) devices to understand their environment. These devices help to take appropriate decisions on time. All these devices mounted on vehicles captured enormous amounts of sensitive data like real-time video feed, location data, and traffic status. It is essential to the safety of the passengers and their effective navigation. Because of this large amount of data there is need for third-party storage where all this data should be securely stored like cloud storage. If this data is sent to the cloud and stored there, then it is very necessary to ensure the privacy preserving of that data and should be protected from any insider attacker and unauthorized access. Cryptography is the technique through which data can be secured but at the same time administrator required some computation on the data for analyzation of any incident or research. This is the primary concern of autonomous vehicles automation system. This research provides a solution to this concern about how the data can easily secure on cloud or any third-party storage and can also perform the computation without revealing any information of data. The suggested method proposed the integration of homomorphic encryption through which administrator or researcher can perform any type of computation without decryption of data on cloud. In this work this technique is tested on the image-based dataset which is captured by the camera mounted on the vehicle. As the image data takes too much storage, it is impossible to store on vehicle that's why it is necessary to send this data to the other storage. Images data can be encrypted per pixel and securely processed in the cloud without decryption. The scheme by Brakerski, Fan, and Vercauteren (BFV) was selected as a cryptographic basis since it enables the implementation of efficient integer-based operations and computations can be done directly on encrypted messages. This technique can be applied to recognize very similar images or similar situations despite the use of diverse environmental conditions. The second is an established object-based approach, in which objects are extracted and encrypted and stored in the cloud along with the corresponding images. The queries are handled by comparing the desired object with the encrypted objects and allow users to download all images linked with that object. This method is computationally cheap and is especially effective where the objects in question are already in the system database. The third one is a method of objects whose nature is unknown, created to process the objects do not present in the prior dataset. Here the object vector is repeated and shifted to create various patterns that correspond to the length of the image vector, and which are then encrypted and compared with encrypted images. Being more computationally intensive but offers flexibility to find new or unexpected object-based images, this method is important in the real world of autonomous driving. All the processing on the cloud was performed on encrypted data, that's why this method ensures privacy

preserving. The encrypted detection framework was combined with the (You Only Look Once version 9) YOLOv9c deep learning model that has become one of the most popular algorithms in real-time object detection. YOLOv9c attained an average Precision of approximately 90 percent when using traffic related objects such as vehicles, pedestrians, road signs, and lights. Its ability to improve small and partially blocked object detection, combined with speed, prequalified it as a supplement to the cryptographic layer. YOLO offered quick predictions of bounding boxes and homomorphic encryption ensured that all comparisons and validations were conducted in a secure way. The methodology started with preprocessing in which the raw images were resized, normalized and turned into one-dimensional pixel vectors. Such vectors were subsequently coded with the use of BFV scheme and sent to the cloud. In the case of known objects, sub vectors were extracted, encrypted, and stored to permit queries later. In the case of undetermined objects, repetitive designs were created and coded. Subtraction operations were implemented directly on the ciphertexts on the cloud, which resulted in encrypted output. Upon their reversion to the administrator, they were decrypted by using the private key and verified by searching through consecutive zero sequences that were the indicators of presence of an object. This methodology made sure that there was no point in exposure of unencrypted data at any point of processing. It was implemented by a multi-layer hardware configuration comprising of a high-performance Ubuntu server with 128 GB of RAM and multi-core processors to operate the cloud, an administrator system with Windows and Xeon processors to decrypt and validate, and an on-vehicle system with light-weight hardware to encrypt and transmit the data. The cryptographic tasks were programmed with the use of the TenSEAL library, and YOLOv9c was trained and tested on traffic data. These findings showed that this combined model was effective. YOLOv9c has a good precision and recall rate which proves that it can be relied on in actual time. The encrypted detection of the known object-based method was the fastest and most efficient, and low computational overhead. The unknown object-based approach, though slower, was necessary in terms of flexibility, which enabled the system to identify new objects that were not in the driving environment before. The similarity-based approach that uses percentile allowed us to make encrypted comparisons of the images, but the similarity scores dropped during bad weather conditions because of low image quality. Notably, the structural integrity of encrypted images was maintained in all methods, which avoided distortions caused by encryption, which otherwise would obstruct detection. It was also found that homomorphic encryption added both time and storage overhead to the analysis. Encrypted vectors were much larger than plaintext and encrypting and decrypting vectors introduced latency when compared to single-stage image processing. Nevertheless, subtraction on the encrypted side was still light, and the current cloud infrastructure contributed to the elimination of storage heavy loads. The cost was determined as being acceptable when weighed against the imperative privacy trims obtained. While classical schemes like Rivest Shamir Adleman (RSA) and Paillier offer limited homomorphic capabilities, they do not support the full range of efficient integer-based operations required by the framework. In contrast, the BFV scheme was selected because it enables extensive computations directly on encrypted messages, providing the end-to-end security processing needed to ensure data is never exposed to risk during analysis. The BFV scheme on the other hand provided end-to-end security processing. According to the Artificial Intelligence (AI) point of view, YOLOv9c had better performance compared to the previous versions as well as other alternatives like faster R-CNN in speed and applicability in real-time situations. Homomorphic encryption integration with

YOLOv9c produced a hybrid system that could not be implemented either by pure cryptographic or pure AI. The strengths of the framework that were identified in the discussion are that the framework provides privacy and detection performance, adjusts to known and unknown objects, and scales in a cloud setting. Limitations also were admitted, including the complexity of computations, the need to store more information, and the dependence on the connection with the cloud. They can, however, be overcome by several enhancements such as hardware acceleration with GPUs and FPGAs, parallel and distributed computing to minimize pattern evaluation delays, and integration of edge and fog computing to minimize latency in regions with low network coverage. Future work also looks forward to improving robustness in adverse weather with image enhancement schemes and multi-sensor fusion, compressing encrypted data schemes, and preparing the framework to cryptography resistant to quantum computing with the advancing of computing technology. An additional potentially useful direction is the federated learning approach where vehicles would learn their local models on encrypted data and submit only encrypted updates to the cloud. This would minimize the overhead in communication and would further increase privacy. The suggested model can be extended to healthcare imaging, surveillance, industrial surveillance, and smart cities, which are all based on sensitive visual data that must be analyzed without revealing privacy. This research gives a future research path that how computation on cloud storage can be done securely by demonstrating that practical and effective image processing which has been done on encrypted images. This implementation also shows that intelligent transportation systems can easily use any storage for image data for encrypted image detection by maintaining privacy-preserving. Integration of BFV homomorphic encryption scheme and YOLOv9c, the system was effective in accomplishing a reliable and efficient object-based images detection. The percentile similarity-based, known object-based, and unknown object-based methods offered multiple options of searching the desired images. As the encryption drawback is both computation and storage load, but these costs are less than the advantages of secure computation on cloud. The findings of this research validate the proposed idea of secure sensitive image information on cloud and facilitate intelligent analytics. This validation opens doors to autonomous vehicles and IoT systems expand more by utilization of cheap cloud storage as well those don’t provide highly secure features because the data itself is secure.

1. GİRİŞ

Nesnelerin İnterneti (IoT), modern dijital ekosistemde verilerin nasıl oluşturulduğunu, iletildiğini ve işlendiğini değiştiren bir paradigmadır. Onun yaygınlaşması, ağ bağlantılı cihazların hızla artışıyla açıklanabilir. Sağlık sektörü ve akıllı şehirler dahil olmak üzere farklı endüstrilere IoT sensörlerinin yaygın olarak dahil edilmesi, verimlilikte ve kullanım kolaylığında büyük gelişmelerle sonuçlanacaktır. Ancak bu büyüme, en başta birbirine bağlı cihazlar tarafından üretilen hassas bilgilerin kitlesel üretiminin doğurduğu güvenlik ve gizlilik tehditleri olmak üzere kritik sorunları da beraberinde getirmektedir. Ağları ve uç noktaları olası saldırılara, yetkisiz erişimlere ve veri sızıntılarına karşı güvence altına almak; mülkiyet bilgilerini korumak ve suç faaliyetlerini engellemek için güvenlik duvarları, kimlik doğrulama sistemleri ve şifreleme sistemlerini içeren güçlü güvenlik tasarımlarını gerektirir. Bu güvenlik önlemleri aynı zamanda işlevsellik ve güvenilirliği sağlamanın yanı sıra veri bütünlüğünü, gizliliğini ve erişilebilirliğini güvence altına almak için de gereklidir. Siber saldırıların artan oranları nedeniyle, ağların ve cihazların çok sıkı güvenlik kontrolleri kullanılarak korunması, tüketici güvenini ve genel olarak dijital altyapıların bütünlüğünü sağlamak açısından son derece önemlidir. Geleneksel güvenlik sistemleri, bazı durumlarda faydalı olsa da, IoT ortamının kendine özgü koşullarında kullanıldığında hâlâ belirgin sınırlamalarla karakterize edilmektedir (Malhotra ve ark, 2021).

IoT ekosistemlerinin merkezi olmayan yapısı, verilerin gizliliğini, bütünlüğünü ve erişilebilirliğini güvence altına alabilecek yeni ve yenilikçi çözümler gerektirmektedir. Bu doğrultuda mevcut araştırma önerisi, homomorfik şifreleme tekniğini ve YZ birleştiren çok boyutlu bir yaklaşım geliştirerek IoT verilerinin güvenliğini ve anomali tespitini iyileştirmek suretiyle bu sorunların aşılmasına yardımcı olacaktır. Mevcut araştırma, hem gizliliği koruyan hem de güvenli olan görüntü işleme yöntemlerinin oluşturulmasına adanmıştır. Artan miktarda görüntü bilgisinin yanı sıra bu verilerin güvenli bir şekilde işlenmesi ve depolanması gerekliliği, söz konusu çalışmanın önemini açıkça ortaya koymaktadır. Ayrıca bulut tabanlı veri merkezlerinde depolanan

görüntü verileriyle ilişkili gizlilik ve güvenlik sorunlarına da çözüm getirmesi hedeflenmektedir. Projeye yapılan en önemli katkı, görüntü verilerinin bulut bilişimde iletilmesini ve değerlendirilmesini mümkün kılan yeni bir görüntü şifreleme algoritmasıdır. Bu algoritma, homomorfik şifreleme tabanlı arama yapılabilir bir yapıdadır ve şifre çözme işlemi yapılmadan bir görüntüdeki nesnenin tanınmasına olanak sağlar. Böylece, görüntü verisi sahipleri, verilerinin nasıl işlendiği üzerinde kontrol sahibi olabilir ve bu da verilerinin korunmasını garanti altına alır.

Çalışmanın geniş kapsamlı odak noktası, bulut tabanlı veri merkezlerinin görüntü verilerini güvenli bir şekilde işleyip analiz etmesine olanak tanıyacak yeni bir görüntü şifreleme algoritması geliştirmektir. Bu yöntem, homomorfik tabanlı aranabilir şifreleme tekniği kullanarak görüntü verilerinin güvenliğini sağlamakta ve veri sahiplerinin işleme operasyonları üzerindeki kontrolünü korumaktadır. Otonom araçlar, çevresel farkındalığı artırmak için büyük miktarda görüntü verisi depolamakta ve sınırlı yerleşik bellek nedeniyle bu bilgileri bulut sunucularına aktarmaktadır (Charroud ve ark, 2024). Araçlar; kamera, GPS cihazları ve iletişim donanımları gibi sensörler ile donatılarak yol koşulları, yol engelleri, trafik işaretleri ve diğer araçlar hakkında önemli bilgileri toplar ve dağıtır. Vehicle-to-everything (V2X) iletişimi Adnan Yusuf ve ark. (2024), Wi-Fi, 4G/5G ve diğer kablosuz teknolojiler, bu araçların internet ve diğer cihazlarla bağlantı kurmasını sağlar. Elde edilen veriler, direksiyon, fren ve hızlanma ile ilgili kararların alınmasını mümkün kılan karmaşık algoritmalar ve makine öğrenimi yöntemlerine dayalı yerleşik hesaplama birimleri tarafından işlenmektedir. Sürücüsüz araçlar, akıllı trafik ışıkları, altyapı ağları ve ev otomasyon sistemleri gibi diğer IoT sistemleriyle iletişim kurabilen IoT sistemleridir. Otonom araçlar tarafından üretilen bilgiler, ek işleme, arşivleme ve paylaşım için bulut sunucularına yüklenebilir. Bulut, gerektiğinde güncellemeler, eğitim sayesinde geliştirilmiş haritalama verileri ve artırılmış işlem kapasitesi sağlayabilir. Bulut bilişim, kullandıkça öde modeline dayalı olarak depolama ve işlem hizmetlerine isteğe bağlı erişim sağlanmasıdır (Fernanda ve ark, 2023). Kullanıcı verileri yerel bakım maliyetlerini ortadan kaldırmak amacıyla bulut sunucularına taşındığında, veri gizliliği, bütünlüğü ve erişim kontrolü sorunu artmaktadır. Ayrıca, yetkisiz veri değişikliklerine izin verilmemesi ve uygun erişim kontrol prosedürlerinin oluşturulmasının sağlanması da büyük önem taşımaktadır.Veri gizliliği ciddi bir problemdir çünkü en iyi niyetli katılımcılar bile verilere erişmeye çalışabilir. Uçtan

uca şifreleme, veriye yüksek düzeyde güvenlik sağlar; örneğin, Gelişmiş Şifreleme Standardı (AES) algoritmasında CBC modunun kullanılması Adak ve ark. (2023), ancak aynı zamanda şifrelenmiş verilerde arama işlevlerini sınırlar. Gizlilik koruma yöntemleri ve bulut hizmetlerinin çeşitli güvenli yöntemleri geniş çapta incelenmiş bir alandır Silva ve ark. (2021) ve günümüzdeki eğilimler, Gizlilik Artırıcı Teknolojiler (PETs) Y. Shen ve ark. (2011) ve homomorfik şifreleme kullanılarak aranabilir şemalardır (Tebaa ve ark, 2012). PET'ler her ne kadar çok güvenli olsa da, yüksek maliyet, yüksek hesaplama yoğunluğu ve kaynak tüketimi ile ayırt edilirler; bu da onların uygulanmasını zorlaştırır, kullanıcı hatalarına yol açabilir ve ciddi bant genişliği sorunlarına sebep olabilir. Ayrıca, PET'lerin kullanımı sahte bir güvenlik algısı yaratabilir ve bu da düzenleyici ve yasal standartlara aykırı eylemlerin teşvik edilmesine yol açabilir (Burghardt ve ark, 2008; Stalder, 2002). Buna karşılık, homomorfik şifreleme şemaları veriler üzerinde şifre çözmeden işlem yapmaya izin verir; bu da verimliliği artırırken kaynak tüketimi gerektirmez (Liu ve ark, 2020). Pratikte, şifrelenmiş düz metin üzerindeki işlemler, şifrelenmemiş düz metin üzerindeki işlemlerle aynı sonucu verir ve böylece verilerin bütünlüğü ve güvenliği korunur. Sonuç olarak, şifrelenmiş verilerde arama yapmak verimli hale gelir ve hassas bilgiler açığa çıkmaz. Homomorfik tabanlı aranabilir şifreleme şemalarının uygulamaları blockchain Iqbal ve ark. (2022), Liu ve ark. (2020), Yi ve ark. (2014), tele-tıp Kocabas ve ark. (2020) ve araç tabanlı ad-hoc ağlar (VANETs) X. Li ve ark. (2021) gibi çeşitli alanlarda gösterilmiştir. Bazı şemalar güvenli veri arama özellikleriyle ilgilenirken Wei ve ark. (2021), bazıları ise gizliliğin korunması, veri bütünlüğü ve karşılıklı kimlik doğrulama içeren güvenli anahtar anlaşmasıyla ilgilenmektedir.

1.1. Araştırma Bağlamı ve Motivasyonu

Akıllı cihazların ve ağa bağlı sistemlerin yaygın olarak kullanılması, insan yaşam biçimini, çalışma düzenlerini ve sosyal hayatı kalıcı olarak değiştirmiştir. Modern hastaneler, giyilebilir cihazları uzaktan hasta takibinde kullanmaya başlamış; şehirler, yollarda otonom araç filolarını uygulamaya koymuş; evler ise halihazırda otonom şekilde kontrol edilen aydınlatma ve enerji kullanımıyla donatılmıştır. Bu gelişmeler, teknolojinin günlük yaşamın içine ne kadar köklü ve karmaşık bir şekilde girdiğinin canlı bir göstergesidir. Bu tür bir bağlantılılığın beklenen faydaları, daha fazla kolaylık, artırılmış güvenlik ve operasyonların daha verimli hale gelmesidir. Ancak bu

avantajları sağlayan karşılıklı bağımlılık aynı zamanda riski de beraberinde getirmektedir: korunmasız bir iletişim kanalı ya da etkilenmiş tek bir cihaz bile, toplumu uzun vadeli etkilerle sarsabilecek geniş çaplı siber saldırıların kapısı olabilir. Dikkat çekici bir gerçek ise IoT cihazlarının her yerde bulunmasıdır; önümüzdeki yıllarda on milyarlarca cihazın birbirine bağlanacağı tahmin edilmektedir.

Otonom araçlar, en fazla veri üreten teknolojiler arasında yer almakta olup saniyeler içinde büyük miktarlarda konum, sensör ve görsel veri üretmektedir. Bu araçlar her ne kadar araç üzerinde gerçek zamanlı kararlar almak zorunda olsalar da bilgilerin depolanması, modellerin güncellenmesi ve kapsamlı analizlerin yapılması için bulut bilişim altyapısına bağımlıdırlar. Ancak bu tür üçüncü taraf kaynaklara bağımlılık, ciddi bir zayıflık oluşturmaktadır; çünkü bulut hizmetlerinde depolanan hassas verilere tamamen güvenilemez. Bu nedenle siber güvenlik geçici bir tehdit değil, sürekli bir sorun haline gelmiştir. Ağlara, bulut hizmetlerine ve yapay zekâ (YZ) modellerine yönelik saldırılar giderek daha sofistike hale gelmiş ve çoğu durumda sistemlerdeki en küçük açıklar bile istismar edilmiştir. En modern sistemler bile bağışık değildir; çünkü YZ algoritmalarını yanıltmak için özel olarak üretilen düşmanca veriler (adversarial inputs), otonom sürüşte yanlış sınıflandırmalara, çarpışmalara veya ciddi arızalara yol açabilmektedir. Bu durum doğrudan insan güvenliği ve ekonomik kayıplarla ilişkilidir. Bu gerçekler, güvenliğin bu sistemlerin tasarımına ve işleyişine sonradan eklenen bir unsur değil, en baştan entegre edilmesi gereken temel bir unsur olduğunun altını çizmektedir.

Simetrik şifreleme (AES) ve asimetrik açık anahtarlı şifreleme (Rivest Shamir Adleman (RSA), Eliptik Eğri Kriptografisi (ECC)) teknikleri, verilerin depolanması ve iletimi için güvenlidir; verilerin gizliliğini ve bütünlüğünü korur, yakalanan verileri okunamaz hale getirir ve değiştirmeye karşı dirençli kılar. Bununla birlikte, bu geleneksel yöntemlerin temel dezavantajı, önemli hesaplamalar yapılmadan önce verilerin şifrelerinin çözülmesi gerekmesidir. Bu durum, büyük miktarlarda kişisel veya hassas veriyi işleyen bulut tabanlı sistemler için ciddi bir engel teşkil etmektedir. Kullanıcılar ya hesaplamalar sırasında risk almakta ya da kendilerini tamamen bulut sağlayıcısına teslim etmek zorunda kalmaktadır.

Uygun bir alternatif ise homomorfik şifrelemedir. Bu yöntem, şifre çözmeye gerek kalmadan doğrudan şifrelenmiş veriler üzerinde işlem yapılmasına olanak tanır. Bu da, özel bilgilerin teorik olarak her zaman gizli tutulabileceği anlamına gelmektedir.

Ancak homomorfik şifreleme oldukça maliyetli bir hesaplama gerektirmekte ve özellikle FHE (Tam Homomorfik Şifreleme) bağlamında yaygın kullanım için henüz pratik değildir (Rauthan, 2022). Bu engelleri aşmak için kriptografinin diğer ileri teknolojilerle bütünleştirilmesine yönelik yeni yaklaşımlara ihtiyaç vardır.

Aksine, YZ veri yorumlama, desen tanıma ve gerçek zamanlı karar desteği alanlarında dikkate değer yetenekler göstermiştir. Örneğin, otonom araçlardaki gelişmiş algoritmalar, (You Only Look Once) YOLO tabanlı nesne tespit modelleri sayesinde engelleri, araçları ve yayaları neredeyse anında tanıyabilmektedir (Diwan ve ark, 2023). Ancak bu potansiyele rağmen, YZ sistemleri genellikle şifrelenmemiş veriler kullanır; bu durum gizlilik gereksinimlerine doğrudan aykırıdır. Ayrıca YZ manipülasyona açıktır. Zehirlenmiş eğitim veri setleri veya adversarial örnekler gibi saldırılar tehlikeli hatalara yol açabilir. Bu nedenle, YZ'nın kritik altyapılarda son derece hassas verileri şifreleme koruması olmadan işlemesi güvenilir değildir.

YZ ile şifreleme arasındaki olası örtüşme bu araştırmayı motive etmiştir. Her biri sorunun bir kısmını çözer: YZ zekâ ve esneklik sağlarken, kriptografi gizliliği korur. Birlikte kullanıldıklarında bu iki yaklaşım, verimlilikten ödün vermeden güvenli ve gizli hesaplamaları mümkün kılar. Dolayısıyla, bulut tabanlı ortamlarda bile otonom araçlar nesneleri tespit edebilir ve buna dayanarak yön bulma kararları verebilir; ancak görsel verilerin gizliliği ve şifrelenmesi ihmal edilmez. Bu durum yalnızca kendi kendine giden araçlarla sınırlı değildir. Kriptografi ve YZ'nın etkileşimi, IoT ağları, sağlık sektörü, akıllı şehirler ve kritik altyapılarda benimsenebilecek, gizliliğe duyarlı yeni bir bilişim çağının başlangıcıdır. Böyle bir yakınsama, hem güvenliği ve işlevselliği artıran hem de gizliliğe saygılı ve güvenilir bir teknolojik geleceği garanti edecektir.

1.2. Güvenli Veri İşlemede Karşılaşılan Zorluklar

Ağların, Nesnelerin İnterneti'nin (IoT) ve otonom sistemlerin hızlı gelişimi yalnızca önemli fırsatlar yaratmakla kalmamış, aynı zamanda ciddi zorlukları da beraberinde getirmiştir. Akıllı şehirler, daha güvenli ulaşım ve daha verimli sağlık hizmetleri vaat eden bu teknolojiler, geleneksel güvenlik araçlarıyla kolayca yönetilemeyen yeni güvenlik tehditleri doğurmaktadır. Buradaki temel çatışma, hassas verilerin yönetilmesi, taşınması ve depolanması için gereken gizlilik ve güven ile gerçek zamanlı işlevselliğin korunması gerekliliğidir. Bu bölümde, otonom araçlarda ve bulut

entegrasyon sistemlerinde çevrimiçi olarak erişilebilen çok sayıda hassas kişisel verinin yaratacağı zorluklar vurgulanmaktadır. Ayrıca, geleneksel tekniklerin sınırlamaları ile YZ modellerinin zayıflıkları ve kısıtları da öne çıkarılmaktadır.

1.2.1. IoT ve otonom araçlarda veri büyümesi ve gizlilik zorlukları

Modern ağ bağlantılı cihazlar rekor hızlarda veri üretmektedir. Örneğin, bir otonom araç tek bir günde terabaytlarca veri üretebilir. Bu veriler GPS izlerini, LiDAR yoğunluk bulutlarını, radar imzalarını ve araç üzerindeki kameralarla elde edilen yüksek çözünürlüklü görüntüleri içermektedir. Bu veri akışı, yayaların tespit edilmesi, trafik ışıklarının algılanması ve engellerin belirlenmesi gibi gerçek zamanlı karar verme süreçlerinde kullanılmaktadır. Bununla birlikte, aynı veri akışı genellikle plaka numaraları, yüz ifadeleri ve konum işaretleri gibi son derece hassas bilgileri de barındırmaktadır. Büyük hacim ile hassas içeriğin birleşmesi potansiyel tehditleri daha da artırmaktadır. Bu tür veriler, kötü niyetli manipülasyon, kimlik hırsızlığı veya izleme amacıyla ele geçirildiğinde istismar edilebilir. Ayrıca, veriler merkezi bulut sunucularına aktarıldıktan sonra mülkiyeti ve yönetimi sıklıkla belirsiz hale gelmektedir. Bu durum önemli soruları gündeme getirmektedir: Bu verilere erişim hakkı kime aittir? Tüketiciler, bilgilerinin devlet veya ticari gözetim amacıyla kullanılmayacağından emin olabilir mi? Bu soruların hâlen net bir cevabının olmaması, IoT ve otonom araç ekosistemlerinde gizliliğin neden büyük bir sorun olarak ortaya çıktığını vurgulamaktadır (Alam, 2024).

1.2.2. Bulut depolamada güvenlik riskleri ve açıkları

Bulut platformları, otonom sistemler ve IoT için hesaplama ve depolama altyapısını sağlamaktadır. Neredeyse sınırsız kapasiteleri sayesinde, ağ bağlantılı cihazların ürettiği devasa veri hacmini yönetmek için cazip bir seçenek haline gelmiştir. Bununla birlikte, bulut ortamları yarı güvenilir kabul edilmektedir (Butt ve ark, 2023). Sağlayıcılar güvenlik önlemleri uygulasa bile riskler devam etmektedir:

- İçeriden kötüye kullanım, özel erişime sahip çalışanların depolanmış verileri kendi çıkarlarına kullanmasıyla ortaya çıkar.
- Yanlış yapılandırmalar, bulut ayarlarının hatalı olması durumunda özel verilerin istemeden açığa çıkmasına yol açar.
- Gelişmiş sürekli tehditler, suçluların uzun süreler boyunca gizli bir şekilde faaliyet göstermesini içerir.

Gizlilik, iletim ve depolama sırasında geleneksel şifreleme ile sağlanmaktadır. Ancak, bulut ortamı önemli bir hesaplama işlemi (örneğin şifrelenmiş araç görüntülerinde nesne tanıma) gerçekleştirebilmek için önce verileri çözmek zorundadır. Bu durum şu ikilemi ortaya çıkarır: Veriler şifreli tutulur ve işlevsellik kaybolur ya da veriler çözülür ve gizlilik kaybolur. Her iki seçenek de askeri gözetim, tıbbi tanı veya otonom sürüş gibi kritik sistemler için uygun değildir.

1.2.3. Geleneksel kriptografinin dezavantajları (AES, RSA, ECC)

Dijital güvenlik uzun süredir AES, RSA ve ECC gibi kriptografik yöntemlere dayanmaktadır. Bu yöntemler depolanan verilerin güvenliğini sağlamak, kullanıcıları doğrulamak ve iletişim bağlantılarını korumak için güvenilirdir. Ancak görev iletim ve depolamadan işleme geçtiğinde bu yaklaşımlar yetersiz kalmaktadır. AES verileri etkili bir şekilde şifrelese de analiz başlamadan önce mutlaka çözülmesi gerekir. RSA ve ECC, kimlik doğrulama ve anahtar değişimi için güvenlik sağlasa da, büyük ölçekli şifreli hesaplamalar için uygun değildir. Gerçek zamanlı ve mahremiyeti koruyan hesaplama gereksinimleri, avantajlarına rağmen hibrit sistemlerle karşılanamamaktadır. Otonom araçlarda milisaniyeler içinde güvenlik açısından kritik kararların alınması gerekirken, şifre çözme ihtiyacı gecikmelere ve risklere neden olur. Örneğin, bir yaya tespit edilmeden önce sensör verilerinin çözülmesi gerekirse güvenlik riske girebilir. Bu sınırlama, hesaplamaların doğrudan şifrelenmiş veriler üzerinde yapılmasına olanak tanıyan gelişmiş yöntemlerin gerekliliğini vurgulamaktadır (Savadatti ve ark, 2025).

1.2.4. Gerçek zamanlı şifrelenmiş hesaplamanın sınırlamaları

Verileri şifrelemek en büyük zorluklardan yalnızca biridir; bir diğeri ise bunu gerçek zamanlı sistemlerde güvenli bir şekilde kullanmaktır. Teorik bir çözüm, sonsuz sayıda hesaplamayı deşifre etmeye gerek kalmadan şifrelenmiş veriler üzerinde yapmaya izin veren Tam Homomorfik Şifreleme (FHE) tarafından sunulmaktadır. Bu sayede bir bulut sunucusu, verileri hiçbir zaman düz metin olarak görmeden analiz edebilir. Ancak, FHE hâlâ hesaplama açısından çok maliyetlidir ve şu dezavantajlara sahiptir:

- Büyük şifreli metinler, iletim ve depolama için daha fazla bant genişliği gerektirir.
- Yavaş şifreleme ve şifre çözme hızları, hızlı işlem gerektiren uygulamalarda kullanımını sınırlar.

- Tekrarlanan işlemler sırasında artan gürültü, hatalı sonuçlara yol açabilir veya maliyetli yenileme işlemleri gerektirebilir.

FHE'nin sınırsız hesaplama özelliklerine rağmen, gerçek zamanlılık gerektiren otonom sürüş gibi uygulamalarda yaygın olarak benimsenmesi hâlâ sınırlıdır. Araştırmacılar, gizliliği koruyan hesaplama ile gerçek zamanlı tepki verebilirlik arasında denge bulmaya çalışmaktadır (Acar ve ark, 2019a).

1.2.5. YZ modeli güvenlik riskleri

Günümüz sistemlerinde YZ, hem güvenliği artıran bir araç hem de potansiyel bir zafiyet unsuru olarak hizmet etmektedir. YZ, eğilimleri bulmada, düzensizlikleri tespit etmede ve bağımsız karar vermeyi kolaylaştırmada oldukça başarılıdır. Ancak, YZ modelleri bağışık değildir ve birkaç şekilde zarar görebilir:

- **Adversarial saldırılar:** Küçük girdi değişiklikleri (örneğin, dur işaretine yapıştırılan etiketler) YZ modellerinin nesneleri yanlış sınıflandırmasına neden olabilir ve bu da zararlı sonuçlar doğurabilir (Kaviani ve ark, 2022).
- **Veri zehirlenmesi:** Eğitim veri setlerine kötü niyetli girdiler eklendiğinde, modeller önyargılı hale gelebilir ve güvenilirliklerini kaybedebilir (Ahmed ve ark, 2021).
- **Model tersine mühendislik saldırıları:** Eğitilmiş bir modeli sorgulayan saldırgan, özel eğitim verilerini (örneğin, bir kişinin görüntüsü gibi) yeniden oluşturabilir (W. Yang ve ark, 2025).

1.3. Araştırma Problemi Tanımı

Bilginin hem gizliliği hem de kullanılabilirliği aynı anda karşılaması gerekmektedir; bu ihtiyaç, akıllı sistemlerin ve ağ bağlantılı cihazların hızlı gelişimiyle ortaya çıkmıştır. IoT cihazları, akıllı altyapılar ve otonom araçlar tarafından üretilen büyük miktarda hassas verinin önemli bir kısmı, analiz, depolama ve uzun vadeli model güncellemeleri için bulut sunucularına aktarılmak zorundadır. Klasik kriptografi yöntemleri, verilerin aktarım veya depolama sırasında korunmasını sağlayabilmektedir; ancak şifrelenmiş veriler üzerinde hesaplama yapılması gerektiğinde yetersiz kalmaktadır. Buna karşılık, YZ büyük veri kümeleri üzerinde çalışarak anormallikleri, desenleri ve karar destek unsurlarını tespit etmede çok daha üstündür; ancak YZ'nin çoğunlukla şifresiz girdilere ihtiyaç duyması büyük bir

paradoksu ortaya çıkarmaktadır: özel bilgiyi ifşa etmeden nasıl işleyebiliriz? Verinin değerinin artışıyla birlikte, rakiplerin onu ele geçirme, değiştirme veya kötüye kullanma motivasyonu da artmaktadır. Bu nedenle, veri bütünlüğü ve gizliliği yalnızca teknik bir gereklilik değil, aynı zamanda insan güvenliğiyle de doğrudan bağlantılıdır; özellikle de otonom sürüşte, yapılacak en küçük hata veya gecikme fiziksel zararlara yol açabilir. Mevcut güvenlik paradigmaları, analiz gerçekleştirmek için ya şifrelemeyi ya da şifre çözmeyi gerektirmektedir – bu da, şifrelenmiş veriler üzerinde güvenli işlem yapılmasını destekleyecek yöntemlerin önemini daha da vurgulamaktadır.

YZ nesne tanıma, saldırı tespiti ve kestirimsel analiz gibi alanlarda etkileyici ilerlemeler kaydetmiş olsa da, şifresiz veriye bağımlılığı ciddi bir sınırlamadır. Kötü niyetli girdilere ve veri zehirlenmesine karşı savunmasızlık, YZ sistemlerinin kritik senaryolardaki zayıf noktasıdır. Bu zayıflık, sürekli özel görsel veriler toplayan ve bu verileri hem yerel olarak işlemek hem de buluta ileterek gerçek zamanlı bilgi ve toplu zekâ sağlamak zorunda olan otonom araçlarda açıkça görülmektedir. Bu tür görüntüler genellikle coğrafi konum bilgileri, plaka numaraları ve tanınabilir yüzler içerdiğinden, gizlilik ihlalleri özellikle ciddi bir tehdit oluşturmaktadır. Mevcut güvenlik yöntemleri bu alanda parçalıdır: YZ tabanlı tespit sistemleri gizlilik pahasına faydalıdır; kriptografi yöntemleri ise depolama güvenliği sağlasa da hesaplama kabiliyetinden ödün vermektedir. Şifrelenmiş görüntü verilerinin gerçek zamanlı olarak aranmasına, sınıflandırılmasına ve analiz edilmesine olanak sağlayan, genel kabul görmüş bir çözüm mevcut değildir. Bu sorun yalnızca sürücüsüz araçlarla sınırlı değildir; akıllı şehirlerde (gözetim kameraları tarafından toplanan verilerin gizliliği), sağlık sistemlerinde (hasta verilerinin kimlik açığa çıkmadan işlenmesi) ve endüstriyel IoT'lerde (sensör kayıtlarının gizli veya tescilli olabilmesi) de benzer durumlar söz konusudur. Şifrelenmiş veriler üzerinde işlem yapabilme kabiliyeti olmadan, kurumlar güvenlik ile işlevsellik arasında bir seçim yapmak zorunda kalmaktadır. Ağların ve cihazların, kriptografik güvenliği zedelemeden YZ tabanlı içgörüleri güvenle kullanabilmesi, bu çalışmanın ortaya koyduğu kritik nitelikteki araştırma boşluğunu oluşturmaktadır.

YZ ve şifreleme alanındaki yeniliklere rağmen, otonom sistemler ve IoT ağlarının gizliliğe duyarlı ortamlarında hassas görüntü verilerini gerçek zamanlı olarak işleyebilmek için iki alanın avantajlarını etkin bir şekilde birleştiren mevcut bir

çerçeve bulunmamaktadır. YZ tabanlı yöntemler daha derinlemesine analizler sağlarken, ham verileri açığa çıkarmaktadır; klasik kriptografik yöntemler ise gizliliği sağlasa da kullanılabilirlikten ödün vermektedir. Bu boşluğu kapatmak için bu tez, YZ destekli nesne tanıma ile FHE'nin (Tam Homomorfik Şifreleme) yeni bir hibritini sunmaktadır. Böylece bulut destekli otonom sistemlerde güvenli depolama, şifrelenmiş veri üzerinde hesaplama ve gizliliği koruyan analiz mümkün hale gelmektedir.

1.4. Araştırma Amaçları

Bu çalışmanın genel amacı, en yeni kriptografik yöntemlerle YZ'yi birleştirerek, özellikle IoT sistemlerinde ve otonom araçlarda şifrelenmiş görüntü verilerinin işlenmesiyle ağları ve cihazları güvence altına alan, gizlilik uyumlu bir çerçeve oluşturmaktır. Güvenilmeyen ortamlar olan bulut altyapılarında, bu çerçevenin amacı hassas bilgilerin gizliliğini sağlamak ve gerçek zamanlı analiz ile karar almayı mümkün kılmaktır. Araştırma bu hedefe ulaşmak için aşağıdaki amaçlar doğrultusunda yönlendirilecektir:

- FHE (özellikle BFV şeması) uygulayarak, şifrelenmiş veriler üzerinde deşifre etmeye gerek kalmadan hesaplamalar yapmak.
- FHE'deki tipik zorlukların üstesinden gelmek: şifreli metin şişmesi, gürültü artışı, hesaplama yükü; çerçevenin bu unsurları dikkate aldığından emin olmak.
- Dağıtık ağlar ve IoT ortamlarında işleyen etkili anahtar yönetim planları oluşturmak.
- Derin öğrenme modellerini, özellikle YOLOv9c'yi, görüntü verilerinin gizliliğini koruyarak nesne tespitine imkân tanıyacak bir iş akışına entegre etmek.
- YZ iş akışlarını, gizliliği koruyan veri temsilleri üzerinde veya şifreli alan içinde nesne tanımaya olanak sağlayacak şekilde uyarlamak.
- Çerçeveyi, yapay zekâ tarafından üretilen çıktıları (örneğin nesne vektörleri) şifreli alanda işleyebilecek, böylece orijinal görüntüleri açığa çıkarmadan bulutta güvenli karşılaştırmalar ve doğrulamalar yapabilecek şekilde tasarlamak.
- Benzerlik kullanarak şifrelenmiş veri kümeleri içinde arama yapmaya yönelik yenilikçi yöntemler önermek.

- Nesne tanıma görevlerinin gizliliği koruyan bir şekilde doğrulanabildiği bir sistem geliştirmek. Bu, bulut sunucusunun ham, şifrelenmemiş görüntülere asla erişmeden güvenli analiz ve arama yapabilmesini, YZ tespit adımının güvenli hesaplama adımından ayrılmasını sağlar.
- Arama stratejilerinin, verimliliğin yanı sıra çıkarım saldırılarına karşı güçlü güvenlik garantileri sunduğundan emin olmak.
- Önerilen çerçevenin, şifreli metin analizi ve düşmanca manipülasyon gibi bilinen saldırı yöntemlerine karşı savunmalarını incelemek.
- Depolama ihtiyaçlarını, arama gecikmesini, şifreleme süresini ve şifre çözme süresini ölçerek hesaplama performansını değerlendirmek.
- Çerçevenin daha büyük veri kümelerine ve daha karmaşık ağ senaryolarına uygulanmasıyla ölçeklenebilirliğini test etmek.
- Otonom araçların görüntü veri kümelerini (örneğin Kaggle trafik işaretleri ve nesne veri kümeleri) kullanarak sistemi doğrulamak.
- Çerçevenin uygulanabilirliğini, gizliliği koruyan hesaplamanın aynı derecede önemli olduğu sağlık, akıllı şehirler ve endüstriyel IoT gibi diğer alanlara genişletmek.
- Kritik verilere erişemeyen hizmet sağlayıcıların bulunduğu bulut destekli IoT sistemlerinde benimsenme potansiyelini vurgulamak.

Bu araştırmanın amacı, güvenli YZ'nin kriptografi ile entegrasyonunun yalnızca mümkün değil, aynı zamanda gerçek zamanlı ve yüksek riskli durumlarda da faydalı olduğunu göstermektir. Bu hedeflere ulaşarak, tez hem akademik anlayışı ilerletmekte hem de güvenilir, gizliliği koruyan akıllı sistemlerin gerçek dünyadaki uygulanabilirliğine katkı sağlamaktadır.

1.5. Temel Araştırma Noktaları

İyi tanımlanmış hedeflere sahip olmanın yanı sıra, iyi yapılandırılmış bir araştırma projesi, araştırma çalışmasının yönünü belirleyen özel sorulara da sahip olmalıdır. Problem ifadesinde belirtilen sorunlar ve hedeflerde ifade edilen amaçlar, bu tezdeki soruların doğrudan kaynağını oluşturur. Bu sorular, hem kuramsal temelleri hem de ağları ve cihazları güvence altına almak için YZ ile şifrelemenin birlikte

kullanılabilirliğini araştırmayı amaçlamaktadır. Araştırma çalışmasına rehberlik eden temel sorular şunlardır:

- **Homomorfik şifreleme kullanılarak şifrelenmiş görüntü verileri üzerinde güvenli gerçek zamanlı hesaplama, verimlilik veya gizlilikten ödün vermeden nasıl mümkün kılınabilir?**

Bu araştırma projesi, hassas verilerin ve gerçek zamanlı karar almanın kritik olduğu otonom araçlar gibi ortamlarda FHE'nin uygulanabilirliğini incelemeyi amaçlamaktadır. FHE'nin hesaplama karmaşıklığı ve şifreli metin büyümesi önemli engellerdir, her ne kadar FHE teorik olarak şifrelenmiş veriler üzerinde sonsuz işlem yapılmasına izin verse de. Bu soruya yanıt vermek için araştırılması gerekenler:

Hangi homomorfik şifreleme sistemi (ör. BFV) performans ve güvenlik arasında en uygun dengeyi sağlar?

Şifreleme ve şifre çözme sürelerinin gerçek zamanlı veri akışlarına etkisi nedir?

Güçlü gizlilik garantilerini korurken hesaplama maliyetlerini düşürmek için hangi iyileştirmeler yapılabilir?

- **Şifrelenmiş veri işleme çerçeveleri, YZ modelleriyle — özellikle YOLOv9c gibi nesne tespit algoritmalarıyla — gizliliği koruyarak nasıl entegre edilebilir?**

Bu araştırma, YZ ile kriptografi arasındaki bağı kurmaktadır. Amaç, normalde şifrelenmemiş veriyle çalışmak için tasarlanmış olan nesne tespit modellerini, gizlilik koruma senaryolarına uyarlamaktır. Bu sorunun incelenmesi şunları içerir:

Derin öğrenme modellerinin, şifrelenmiş görüntü vektörlerini ya da gizlilik koruyan verileri nasıl analiz edebileceğini keşfetmek.

Şifreli alanlarda çalışmanın tespit doğruluğunu önemli ölçüde etkileyip etkilemediğini belirlemek.

YZ'nin zafiyetlerinden faydalanan kötü niyetli saldırılara karşı güvenli bir çerçevenin nasıl korunabileceğini araştırmak.

- **Şifrelenmiş görüntüler üzerinde bilinen-nesne tespiti, bilinmeyen-nesne tespiti ve benzerlik tabanlı aramayı güvenli ve etkili bir şekilde gerçekleştirmek için hangi yöntemler geliştirilebilir?**

Gerçek dünya uygulamalarında şifrelenmiş veriler üzerinde arama yapabilme yeteneği çok önemlidir. Bu sorunun ana odağı, şifrelenmiş veri kümelerinden anlamlı geri getirme yöntemleri geliştirmektir. Bu kapsamda:

Şifresi çözülmeden şifrelenmiş vektör karşılaştırması için matematiksel yöntemler geliştirmek.

Şifrelenmiş kalıpları verilen bilinen nesneleri tanıyacak teknikler oluşturmak.

Daha önce açıkça etiketlemeye gerek kalmadan yeni kalıpları tanıyabilen esnek bilinmeyen-nesne tespit yöntemleri geliştirmek.

Bu sistemlerin büyük, bulut tabanlı veri kümeleri için yeterince verimli olduğundan emin olmak.

- **Önerilen yapıda, güvenlik, hesaplama yükü, depolama ihtiyaçları ve tespit doğruluğu arasındaki ödünleşimler nelerdir?**

Her güvenlik sistemi, kullanım kolaylığı ile koruma arasında bir denge kurmak zorundadır. Bu soru, önerilen çözümün uygulanabilirliğini, şu tür ödünleşimleri belirleyerek değerlendirmektedir:

Şifreli metin büyümesinin depolama maliyetlerine etkisi.

Arama ve şifreleme yöntemlerinin getirdiği ek işlem süresi.

Gizliliği korumanın, YZ tabanlı tespit doğruluğunu önemli ölçüde azaltıp azaltmadığı.

Bu ödünleşimlerin, yalnızca güvenliği ya da yalnızca hızı önceliklendiren mevcut yöntemlerle nasıl karşılaştırıldığı.

- **Önerilen metodoloji, akıllı şehirler, sağlık hizmetleri, otonom araçlar ve endüstriyel IoT gibi gerçek dünya alanlarına ne ölçüde uygulanabilir ve ölçeklenebilir?**

Bu soru, çerçevenin yalnızca teknik uygulanabilirliğini değil, aynı zamanda daha geniş etkisini ve esnekliğini de ele almaktadır. Şunları araştırır:

Çözümün, yoğun trafik ve büyük veri kümelerinin bulunduğu ortamlara ölçeklenebilir olup olmadığı.

Önerilen yöntemlerin ek hassas alanlara (ör. güvenlik kameraları, tıbbi görüntüleme) nasıl uyarlanabileceği.

FHE ile YZ'nin birleşiminin, farklı sektörlerde gizliliği koruyan hesaplama için bir model olarak kullanılıp kullanılamayacağı.

1.6. Araştırmanın Kapsamı

Bu araştırma çalışmasının kapsamı, ağların ve cihazların güvenliği alanına hedeflenmiş ve faydalı bir katkı sağlamak amacıyla özellikle belirlenmiştir. Çalışma, şifreleme ve YZ'nin entegrasyonunun geniş etkileri olsa da, ağırlıklı olarak otonom araçlar ve IoT sistemleri için gizlilik korumalı görüntü işleme üzerine odaklanmakta, sağlık hizmetleri ve akıllı şehirler gibi yakın alanlara da uzanmaktadır. Araştırmanın kapsamı aşağıda açıklanmaktadır:

- Araştırma çalışmasının odak noktası, FHE'nin YZ tabanlı nesne tespitiyle entegrasyonudur.
- Çalışma, güvenlik ve gizliliğin kritik olduğu cihazlar tarafından üretilen görsel verilere (ör. otonom araçlar) odaklanmaktadır.
- Çerçeve, gizli şifrelenmiş verilerin işlenmek ve depolanmak üzere dışarıya aktarıldığında dahi özel kalması gereken bulut destekli ortamlara yönelik olarak tasarlanmıştır.
- YOLOv9c ve diğer nesne tanıma modelleri, bu alandaki YZ uygulamalarına örnek olarak ele alınmaktadır.
- Tezin konusu, veri gizliliğidir; yani özel bilgilerin depolama, iletim veya hesaplama sırasında asla açık hâlde ortaya çıkmamasını sağlamaktır.
- Araştırma çalışması ayrıca bütünlüğü de dikkate alır; bu, hesaplama sonuçlarının ve şifrelenmiş verilerin değiştirilmediğinden emin olmayı ifade eder.
- Bununla birlikte, benzer ilkeler kabul edilse de, kullanıcı kimlik doğrulama, erişim kontrolü ve büyük, federatif ağlarda anahtar dağıtımı gibi unsurlar kapsam dışında bırakılmıştır.
- Güvenilen ortamlardaki içeriden gelen tehditler ayrıntılı olarak incelenmemekte; bunun yerine odak, dış ve bulut tabanlı saldırganlara karşı savunmaya yöneliktir.
- Çalışma, diğer mimarilerin kapsamlı incelenmesi yerine, YOLOv9c'yi temel nesne tespit modeli olarak kullanmaktadır.

- Projenin amacı, tamamen yeni modeller geliştirmekten ziyade mevcut derin öğrenme modellerini şifreli ortamlara uyarlamaktır.
- Doğal dil işleme veya video dizi analizi gibi diğer YZ görevleri dikkate alınmamış; kapsam yalnızca görüntülerde nesne tespiti ile sınırlandırılmıştır.
- Gerçekçi otonom sürüş ortamlarını simüle etmek için Kaggle yol ve trafik görüntü veritabanları gibi kamuya açık görüntü veri kümeleri kullanılmaktadır.
- Çalışmanın odak noktası, tamamen işlevsel otomobillerde veya endüstriyel sistemlerde dağıtımdan ziyade, simülasyon ve kavram kanıtlama (proof-of-concept) gösterimleridir.
- Değerlendirme ölçütleri, arama verimliliği, tespit doğruluğu, depolama yükü ve şifreleme/şifre çözme süresinden oluşmaktadır.
- Çerçeve ölçeklenebilirlik düşünülerek geliştirilmiş olsa da, milyonlarca IoT cihazı arasında geniş ölçekli endüstriyel uygulama mevcut deneysel kapsamın dışında bırakılmıştır.
- Milisaniyeler içinde çarpışma önleme gibi kritik güvenlik senaryoları için ultra düşük gecikme garantileri, mevcut araştırma çalışmasının odak noktası değildir; gerçek zamanlı performans kavram kanıtlama düzeyinde değerlendirilmektedir.
- Araştırma çalışması, YZ ve şifrelemenin entegrasyonuna odaklanmakta; güvenli donanım gibi tamamlayıcı unsurları kapsamamaktadır.

1.7. Araştırmanın Önemi

Bu araştırma, YZ ve kriptografi alanları arasındaki kritik bir boşluğu kapatmayı hedeflediği için önemlidir. Her iki alan kendi başına önemli ilerlemeler kaydetmiş olsa da, gizliliği koruyan hesaplama için bu iki alanın entegrasyonu hâlâ sınırlıdır. Bu tez, homomorfik şifreleme (HE) ile YZ tabanlı nesne tanıma yöntemlerinin nasıl bir araya getirilebileceğini ve böylece hassas görsel verilerin kullanılabilirlikten ödün verilmeden güvence altına alınabileceğini göstermektedir. Araştırmada kullanılan BFV homomorfik şifreleme algoritması, teorik bir çerçevenin ötesine geçerek gerçek dünyadaki görüntü verileri üzerinde şifreli hesaplama yapılabileceğini kanıtlamaktadır. Çalışma, nesne tespit modellerinin verilerin şifreli kalması gereken ortamlara uyarlanabileceğini ortaya koymakta ve böylece gizlilik dostu YZ alanına

katkı sunmaktadır. Bu durum, güvenilir ve dayanıklı akıllı sistemlere giden yolu açmaktadır. Araştırma, YZ, otonom sistemler ve siber güvenlik kesişiminde disiplinler arası bir katkı sunmaktadır. Özellikle, otonom araçlarda plaka, yaya veya konum bilgisi gibi hassas verilerin gerçek zamanlı ve güvenli şekilde işlenmesi sorununu doğrudan ele almaktadır. Bunun yanı sıra, sınırlı kaynaklara sahip IoT cihazlarının verilerini şifreli biçimde buluta aktarmasına imkân tanımaktadır. Bu yaklaşım, gizlilik endişelerinin bulut benimsenmesini yavaşlattığı birçok sektörde güveni artırmaktadır. Tezin sunduğu yöntemler, yalnızca otonom sürüş ve IoT ile sınırlı kalmayıp sağlık (ör. hasta kayıtları), şehir gözetim sistemleri, bankacılık, ulusal güvenlik ve hatta askeri istihbarat gibi hassas verilerin güvenli şekilde analiz edilmesi gereken diğer alanlara da uygulanabilir. Bu yönüyle çalışma, farklı sektörlerde gizlilik dostu analitik uygulamalarına ışık tutmaktadır. Çalışma ayrıca, performans kısıtlamaları nedeniyle genellikle uygulanamaz görülen homomorfik şifrelemenin pratik kullanımına dair kanıt sunarak endüstrideki paydaşları bu teknolojiyi değerlendirmeye teşvik etmektedir. Geliştirilen çerçeve, GDPR gibi uluslararası veri koruma yasalarıyla uyumlu güvenli YZ boru hatlarının mümkün olduğunu göstermektedir. Bunun ötesinde, çalışma etik YZ uygulamalarını desteklemekte, modellerin ham veriye erişimini engelleyerek önyargı ve kötüye kullanım risklerini azaltmaktadır. Ayrıca, kuantum saldırılarına karşı dirençli örgü tabanlı homomorfik şifreleme sayesinde gelecekte ortaya çıkabilecek tehditlere karşı da dayanıklılık sağlamaktadır. Özetle, bu çalışma gelecek nesil gizlilik koruyucu teknolojilerin temellerini atmakta, YZ'nin şifreleme ile bütünleştiği yeni araştırmalara yol göstermekte ve güvenilir, gizlilik odaklı otonom sistemler ile IoT ağlarının yaygınlaştırılmasına katkı sunmaktadır.

1.8. Araştırma Katkıları

Bu tezdeki temel değer, YZ ile en güncel şifreleme yöntemlerini bir araya getirerek, şifrelenmiş görüntü verilerinde gizlilik korumalı nesne tespiti ve arama için yeni bir modelin geliştirilmiş olmasıdır. Önceki çalışmalarda ya kullanılabilirlikten ödün vererek yalnızca şifrelemeye odaklanılmış ya da gizliliği ihmal ederek yalnızca YZ yöntemleri öne çıkarılmıştır. Bu çalışma ise hem işlevselliği hem de güvenliği bir arada sağlayan bütüncül bir çözüm ortaya koymaktadır. Katkılar şunlardır:

- Şifrelenmiş görüntü verisi üzerinde doğrudan hesaplama yapabilen, tamamen homomorfik şifreleme tabanlı bir sistem tasarlanmış ve uygulanmıştır.

- Büyük ölçekli görüntü vektörlerinde BFV tekniği kullanılarak hesaplama gecikmeleri, gürültü artışı ve şifreli metinlerin büyümesi gibi gerçek dünya problemleri ele alınmıştır.
- Gerçek görüntü verilerinin hiçbir zaman açığa çıkarılmadan gizlilik korumalı şekilde analiz edilebileceği gösterilmiştir.
- YOLOv9c derin öğrenme modeli, şifrelenmiş veri işleme hattında kullanılabilecek şekilde uyarlanmıştır.
- Ham, şifrelenmemiş görüntülere erişim olmaksızın da nesne tespitinin mümkün olmasını sağlayan yöntemler geliştirilmiştir.
- Düz metin ve şifreli alanlarda tespit doğruluğu sistematik biçimde analiz edilmiş, performans ile gizlilik arasındaki denge ortaya konmuştur.
- Şifrelenmiş görüntü koleksiyonları üzerinde benzerlik tabanlı aramaların yapılabilmesi için yeni teknikler önerilmiştir.
- Hem bilinen nesnelerin (referans şifrelenmiş alt vektör verildiğinde) hem de bilinmeyen nesnelerin (sistemin yeni desenleri kendi başına tanıdığı durumlarda) tespitini sağlayan yöntemler geliştirilmiştir.
- Bu arama yöntemlerinin çıkarım saldırılarına karşı güçlü güvenlik güvenceleri sunduğu ve hem erişim verimliliğini hem de gizliliği koruduğu doğrulanmıştır.
- Kaggle yol ve trafik veri kümeleri gibi herkese açık veri setleri kullanılarak kapsamlı deneysel değerlendirmeler yapılmıştır.
- Şifreleme ve çözme süreleri, depolama yükü, tespit doğruluğu ve arama verimliliği ölçülerek sistem performansına dair bütünsel bir tablo elde edilmiştir.
- Daha karmaşık veri kümeleri üzerinde çerçeve test edilerek ölçeklenebilirlik gösterilmiş, doğruluk, hız ve güvenlik arasındaki denge değerlendirilmiştir.
- Önerilen mimari, gizlilik korumasının kritik olduğu ancak gerçek zamanlı nesne tespitinin güvenlik için zorunlu olduğu otonom araç senaryolarında uygulanmıştır.
- Çalışma, ayrıca endüstriyel IoT (gizli sensör analitiği), akıllı şehirler (şifrelenmiş gözetim analizi) ve sağlık (güvenli tıbbi görüntüleme) gibi alanlara da genişletilmiştir.

- Yaklaşımın farklı ve hassas alanlarda uyarlanabilirliği ve genellenebilirliği gösterilmiştir.
- Teorik düzeyde, gizlilik korumalı hesaplama için tam homomorfik şifreleme ile YZ nın sistematik entegrasyonu ortaya konmuştur.
- Pratik düzeyde ise, çalışmada kavramsal bir uygulama gerçekleştirilerek yöntemin uygulanabilirliği kanıtlanmış ve güvenli YZ tabanlı sistemlerin gerçek dünyada kullanımının önü açılmıştır.

1.9. Tezin Organizasyonu

Bu tez, araştırma sürecinin anlatısını netlik ve mantıksal akış sağlayacak şekilde kademeli olarak inşa eden bölümlere ayrılmıştır. Temel kavramlarla başlayıp yöntem, uygulama, değerlendirme ve sonuçlara ilerleyerek yapılandırılmıştır. Bölümlerin genel çerçevesi şu şekildedir:

- **Bölüm 1: Giriş**

Bu bölümde araştırmanın arka planı, motivasyonu ve problem tanımı sunulmaktadır. Araştırma konusu tanımlanır, amaçlar ve yönlendirici sorular ortaya konur, çalışmanın kapsamı ve sınırlamaları açıklanır, önemi vurgulanır ve temel katkılar listelenir. Son olarak, tezin organizasyonuna dair özet verilir.

- **Bölüm 2: Arka Plan ve Literatür Taraması**

Bu bölümde gerekli teorik temeller ve mevcut literatür incelenmektedir. YZ , kriptografi ve gizlilik korumalı hesaplamanın temel kavramları ele alınır. Ardından, literatür taraması kapsamında şifrelenmiş veri işleme, güvenlik uygulamalarında YZ kullanımı ve mevcut kriptografi–YZ entegrasyonları incelenir. Literatürdeki boşluklar ortaya konularak bu araştırmanın gerekliliği vurgulanır.

- **Bölüm 3: Kriptografi ve Yapay Zekâ Temelleri**

Bu bölümde tez için gerekli teorik altyapı sunulmaktadır. Öncelikle kriptografinin temelleri ve hiyerarşisi anlatılır; açık anahtar altyapıları, simetrik ve asimetrik şemalar ile homomorfik şifreleme gibi ileri teknikler ele alınır. Homomorfik şifreleme aileleri (BFV, CKKS, BGV) için güvenlik varsayımları ve parametreleme yöntemleri açıklanır. YZ perspektifinden, özellikle evrişimsel sinir ağları (CNN) ve YOLO ailesi üzerinden nesne tespitinin temelleri, özellik temsili ve hesaplama karmaşıklığı

açısından değerlendirilir. Tutarlılık sağlamak amacıyla tez boyunca kullanılacak matematiksel altyapı ve gösterimler de bu bölümde tanıtılır.

- **Bölüm 4: Sistem Tasarımı ve Önerilen Yöntem**

Bu bölümde çalışmada kullanılan metodolojik çerçeve sunulmaktadır. Genel araştırma konsepti, veri seti seçimi, deney ortamı, YZ modeli (YOLOv9c) ve kriptografik yaklaşım (BFV FHE) açıklanır. Yöntem; gizlilik korumalı arama algoritmalarının geliştirilmesi, YZ nın şifrelenmiş verilerle entegrasyonu ve sistemin doğruluk, ölçeklenebilirlik ve güvenlik açısından değerlendirilmesini kapsar. Ayrıca bilinen nesne, bilinmeyen nesne ve yüzde benzerliğe dayalı şifreli arama teknikleri akış şemaları, algoritmalar ve sözde kodlarla açıklanır.

- **Bölüm 5: Deneysel Kurulum ve Uygulama**

Homomorfik şifreleme parametre ayarları, ön işleme adımları, eğitim/ince ayar kararları, donanım–yazılım altyapıları ve veri setleri (örneğin otonom sürüş görüntü koleksiyonları) tanıtılır. Şifreleme, çözme/doğrulama ve şifrelenmiş temsiller üzerinde çıkarım/arama adımlarının uygulanışını ayrıntılı biçimde açıklar. Yeniden üretilebilirlik materyalleri ve karmaşıklık analizleri de sunulur.

- **Bölüm 6: Sonuçlar ve Analiz**

YOLO performansı, şifreli yüzde benzerliği tespiti, bilinen ve bilinmeyen nesne tespiti, BFV gecikme ve depolama yükü gibi nicel sonuçlar sunulmaktadır. Ayrıca istatistiksel analizler, hata örnekleri, hassasiyet testleri ve RSA, Paillier, DGK ile düz metin tabanlı yaklaşımlar gibi temellerle karşılaştırmalar yer alır. Bu karşılaştırmalar, önerilen yöntemin hem güvenlik hem de hesaplama verimliliği açısından mevcut yaklaşımlara göre avantajlarını açıkça ortaya koymaktadır. Elde edilen bulgular, mahremiyet korunarak gerçekleştirilen nesne tespitinin gerçek zamanlı ve pratik uygulamalar için uygulanabilir olduğunu göstermektedir.

- **Bölüm 7: Tartışma**

Elde edilen sonuçlar, araştırmanın amaçları ve soruları ışığında değerlendirilmektedir. Gürültü artışı, hesaplama maliyetleri, anahtar yönetimi gibi zorluklar ve avantajlar ele alınır. Ayrıca daha geniş IoT/akıllı şehir ve sağlık bağlamlarındaki uygulanabilirlik, dağıtım ve etik konular tartışılır. Buna ek olarak, sistemin ölçeklenebilirliği ve gerçek zamanlı uygulamalara entegrasyonu açısından karşılaşılabilecek sınırlamalar analiz

edilmektedir. Son olarak, gelecekte yapılabilecek iyileştirmeler ve bu alandaki olası araştırma yönleri değerlendirilmektedir.

- **Bölüm 8: Sonuç ve Gelecek Çalışmalar**

Araştırma soruları ele alınır ve katkıların özeti sunulur. Video akışlarına genişleme, kuantum sonrası geliştirmeler, YZ destekli anahtar yönetimi, FHE için GPU/ASIC hızlandırma, kafes parametre optimizasyonu ve federatif öğrenme entegrasyonu gibi olası gelecek yönelimler tartışılır.

Kaynakça: Çalışmada atıfta bulunulan tüm eserlerin kapsamlı bir listesi

2. ARKA PLAN VE LİTERATÜR TARAMASI

Bu araştırmanın temeli, kriptografi, YZ, bulut bilişim platformlarının güvenlik gereksinimleri ve otonom araç ekosistemleri alanındaki önceki çalışmaları inceleyen literatür taramasıyla atılmaktadır. Literatür taraması; geleneksel şifreleme yöntemleri, homomorfik şifrelemenin gelişimi, YZ'nin görüntü tanıma ve anomali tespitindeki kullanımı ile IoT cihazlarının bulut tabanlı bilişim ortamlarına entegrasyonunda karşılaşılan özgün zorlukları kapsamaktadır. Taramanın sonunda araştırma boşluklarına değinilerek, otonom araçlarda şifrelenmiş görüntü işleme için YZ destekli, ölçeklenebilir ve güvenli bir homomorfik şifreleme çerçevesine duyulan ihtiyaç vurgulanmaktadır.

2.1. Nesnelerin İnterneti (IoT) ve Veri Güvenliği

IoT kavramına göre, gündelik nesneler çeşitli sensörler, aktüatörler ve iletişim özellikleriyle donatılarak çevrimiçi bağlantı kurabilir ve bilgi paylaşabilir hale gelmektedir. IoT ile YZ'nin birleşimi; otomasyon, izleme ve veriye dayalı karar verme süreçlerini mümkün kılarak dijital platformların ve fiziksel cihazların birlikte etkin bir şekilde çalıştığı akıllı ve ağ tabanlı bir ortam oluşturmaktadır. Güncel tahminler, dünya genelinde milyarlarca IoT cihazının kullanımda olduğunu ve akıllı evler, sağlık, tarım, endüstriyel otomasyon ve ulaşım altyapısı gibi çeşitli alanlarda büyük miktarda veri ürettiğini göstermektedir (Zikria ve ark, 2021). Otonom araçlar ve akıllı ulaşım altyapıları, mobilite ve taşımacılık sektöründe IoT'nin belkemiğini oluşturmaktadır. Sürücüsüz araçlarda kamera, lidar, radar, ultrasonik sensörler ve GPS modülleri gibi birçok sensör yer almaktadır. Bu sensörler, trafik ışıkları, yaya hareketleri, yol koşulları ve araç hızını da içeren çevresel verileri sürekli üretmektedir. Toplanan veriler, engel önleme veya acil frenleme gibi düşük gecikme gerektiren işlemler için araç üzerinde işlendikten sonra, geniş çaplı depolama, veri analitiği ve YZ modeli eğitimi için bulut sunucularına aktarılmaktadır. Bu veri akışı şu şekilde özetlenebi

- Lidar, kamera ve radar gibi sensörler ham çevresel verileri toplar
- Kritik görevler araç üzerindeki birimler tarafından ön işleme tabi tutulur.

- Kritik görevler araç üzerindeki birimler tarafından ön işleme tabi tutulur.
- 5G, araçtan araca veya araçtan altyapıya bağlantılar aracılığıyla veri iletimi yapılır.
- Nesne tanıma ve büyük ölçekli veri analizi bulut platformları ve analitik katmanda gerçekleştirilir.

Bu veri akışı, otonom araçlardaki IoT çerçevesinin yalnızca yerel cihazlarla sınırlı kalmayıp, bulut platformları tarafından desteklenen zekâya kadar uzandığını göstermektedir. Bu katmanda makine öğrenmesi algoritmaları, büyük miktarda veriyi kullanarak güvenlik özelliklerini, algılama kabiliyetini ve navigasyonu sürekli olarak geliştirmektedir. Ancak IoT teknolojileri tüm bu avantajlarına rağmen önemli güvenlik ve gizlilik sorunları da doğurmaktadır. IoT cihazları, dağıtık yapıları ve sınırlı kaynaklara sahip olmaları nedeniyle saldırılara karşı özellikle savunmasızdır. Geleneksel sistemlerle kıyaslandığında, maliyet ve enerji kısıtları nedeniyle IoT cihazları çoğu zaman güçlü güvenlik modüllerinden yoksundur. Bunun bir sonucu olarak otomotiv IoT ekosisteminde saldırı yüzeyi önemli ölçüde artmakta ve saldırganların özel araç verilerine erişimini kolaylaştırmaktadır. Başlıca tehditler şunlardır:

- **Dinleme (Eavesdropping):** Saldırganların, iletişim kanallarını ele geçirerek GPS koordinatları veya görüntüler gibi özel sensör verilerini elde etmesi (H. Wu ve ark, 2024).
- **Aldatma (Spoofing) saldırıları:** Navigasyonu bozmak amacıyla kötü niyetli düğümlerin güvenilir sensör veya araç kimliğiyle kendini tanıtıp sahte çevresel ya da konum verisi göndermesi (Dasgupta ve ark, 2022).
- **Tekrar (Replay) saldırıları:** Saldırganların, sistemin yanıltıcı ya da geçersiz bilgileri doğruymuş gibi kabul etmesini sağlamak için geçerli veri paketlerini yeniden göndermesi.
- **Ortadaki adam (Man-in-the-Middle) saldırıları:** Altyapı ile araç arasındaki iletişimlerin ele geçirilerek değiştirilmesi ve sahte komutların gönderilmesi.
- **Gizli bilgilerin açığa çıkması:** Araç plakaları, sürücü kimlikleri veya seyahat alışkanlıkları gibi özel bilgilerin şifrelenmeden aktarılması veya depolanması sonucu kamuya sızması.

- **Yan kanal saldırıları:** Saldırganlar, güç tüketimi, zamanlama veya sensör verileri gibi dolaylı bilgileri analiz ederek araç veya kullanıcıya dair hassas bilgilere ulaşabilir. Bu saldırılar doğrudan şifrelemeyi hedeflemez ancak IoT cihazlarının gizli verilerini açığa çıkarabilir.
- **Erişim desenlerinin saptırılması:** Saldırganlar, belirli veri veya servislerin ne sıklıkla erişildiğini inceleyerek araç güzergâhları, kullanıcı alışkanlıkları veya davranış kalıpları hakkında çıkarım yapabilir. Bu durum kötü niyetli aktörlere araç faaliyetlerini öngörme imkânı sunar.
- **Bulut saldırgan modeli:** "Dürüst ama meraklı" modelde bulut sağlayıcıları protokollere uyar ancak şifreli veya şifresiz araç verilerinden gizli bilgi çıkarmaya çalışabilir. "Kötü niyetli" modelde ise saldırganlar sisteme veri enjekte edebilir veya veri bütünlüğünü bozabilir.
- **Seçilmiş şifreli metin saldırıları (CCA):** Saldırganlar, sisteme özel olarak tasarlanmış girişler sağlayarak hassas verilere erişmeye veya veriyi dolaylı olarak çözmeye çalışabilir. Sistemin kötü niyetli girdilere karşı dayanıklı olmaması bu saldırı türünü mümkün kılar.
- **Anahtar döndürme riskleri:** Güvenliği sağlamak için kriptografik anahtarların periyodik olarak yenilenmesi gerekir; ancak bu süreç doğru yönetilmezse geçici veri sızıntılarına veya güvenlik açıklarına yol açabilir.
- **Bootstrap maliyeti ve güvenlik etkisi:** Çok yoğun şifrelenmiş sistemler, gürültüyü sıfırlamak için zaman zaman bootstrapping işlemine ihtiyaç duyar. Bu işlem yüksek hesaplama maliyeti getirir ve optimize edilmezse zamanlama bilgisi sızmasına veya sistem performansının düşmesine neden olabilir.

Araç IoT çerçevesinin ele geçirilmesi; kazalar, trafik sıkışıklıkları ya da ulaştırma altyapısına yönelik koordineli siber saldırılar gibi olaylara yol açabileceğinden, bu tür güvenlik açıklarının sonuçları yalnızca kişisel mahremiyetle sınırlı kalmaz, aynı zamanda kamu güvenliğini de tehdit eder. Bu nedenle, yalnızca IoT bağlantılarını korumakla kalmayıp aynı zamanda bulut platformlarında işlenen hassas araç verilerini de güvence altına alan uçtan uca güvenlik çerçevelerine ihtiyaç vardır. Simetrik ve asimetrik şifreleme teknikleri depolama ve aktarım sırasında gizliliği sağlasa da bazı kısıtları mevcuttur. IoT cihazları, kaynak sınırlılıkları nedeniyle karmaşık kriptografik teknikleri verimli bir şekilde çalıştıramaz. Bu da gerçek zamanlı sistem tepkilerinde gecikmelere neden olur. Ayrıca, geleneksel şifreleme yöntemleri verilerin işlenmeden

önce çözülmesini gerektirdiği için bulut sağlayıcılarına ham (şifrelenmemiş) veriler açığa çıkar. Buradaki temel sorun güvenlik ile işlevsellik arasındaki bu çelişkiden kaynaklanmaktadır: Verilerin mahremiyetini nasıl koruyarak bulut ortamında pratik analitiklere izin verilebilir?

Araştırmacılar, bu endişeleri gidermek için veriler üzerinde doğrudan şifreli hâlde hesaplama yapılmasına imkân tanıyan homomorfik şifreleme gibi yeni teknikler üzerine yoğunlaşmaktadır. Bu paradigma, işlem sırasında çözme ihtiyacını ortadan kaldırarak güvenilmeyen ortamlarda veri sızıntısı riskini azaltır. Otomotiv IoT çerçevesinde, homomorfik şifreleme hassas görüntülerin ve sensör akışlarının ömür döngüsünün her aşamasında hesaplama, depolama ve iletim dahil mahremiyetini güvence altına alabilir. Sonuç olarak, homomorfik şifreleme; IoT tabanlı otonom araçlarda akıllı analitik ile veri gizliliği arasındaki çelişkili gereksinimleri karşılamada uygulanabilir bir çözüm olarak giderek daha fazla kabul görmektedir.

2.2. Otonom Araçlar ve Veri Üretimi

Otonom araçlar, modern ulaşımda en dönüştürücü yeniliklerden biridir. Bu araçlar, sensörler, hesaplama platformları ve YZ algoritmalarının birleşimine dayanarak doğrudan insan kontrolü olmadan hareket edecek şekilde tasarlanmıştır. Geleneksel araçların büyük ölçüde insan algısına dayanmasının aksine, otonom sistemler çevrelerini sürekli algılama ve işleme yoluyla dijital olarak kavrar. Bu da güvenli ve verimli bir şekilde yönetilmesi gereken son derece büyük veri hacimlerinin ortaya çıkmasına yol açar.

Bir otonom aracın işleyişi, zengin bir sensör topluluğuna bağlıdır. Kameralar, çevredeki ortamın yüksek çözünürlüklü görüntülerini ve videolarını yakalayarak yayaların, diğer araçların, yol işaretlerinin ve şerit çizgilerinin algılanmasını sağlar. Lidar olarak bilinen ışık tespiti ve mesafe ölçüm sistemi, lazer ışınları kullanarak üç boyutlu nokta bulutları üretir ve bu sayede nesnelerin tanımlanmasına ve araçtan uzaklıklarının belirlenmesine yardımcı olur. Radar sistemleri, nesneleri tespit etmek ve hızlarını ölçmek için radyo dalgaları yayar; bu da özellikle görüşe dayalı sensörlerin başarısız olabileceği kötü hava koşullarında önemlidir. Ultrasonik sensörler genellikle park yardımı ve yakın mesafedeki engellerin algılanması gibi kısa menzilli görevlerde kullanılır. Son olarak, küresel konumlama sistemi (GPS) ve ataletsel ölçüm birimleri gibi konumlandırma sistemleri doğru navigasyon için gerekli jeolokasyon ve yönelim

bilgilerini sağlar. Bu sensörlerin her biri farklı bir amaca hizmet eder ve birlikte çalıştıklarında güvenilir ve yedekli bir algı sistemi oluştururlar. Çeşitli bilgi kaynaklarının birleştirilmesiyle otonom araçlar, şehir yolları ve otoyollar gibi dinamik ortamlarda daha yüksek doğruluk ve dayanıklılık kazanır.

Bir otonom araçtaki sensör topluluğu, son derece büyük ve çeşitli veri üretir. Kameralar sürekli görüntü ve video akışları oluşturur ve bu veriler tek bir gün sürüşte terabaytlarca boyuta ulaşabilir. Lidar, yoğun üç boyutlu nokta bulutları üretir ve bu veriler saniyede yüzlerce megabayta çıkabilir. Radar sistemleri, görsel verileri tamamlayan hız ve mesafe ölçümleri sağlar. GPS ve ataletsel ölçüm birimleri ise araç rotası, hızı ve konumuna dair bilgiler üretir. Ayrıca yol yüzey koşulları, trafik yoğunluğu ve hava durumu gibi çevresel bilgiler de kaydedilir (Ayala ve ark, 2021). Görsel, mekânsal ve bağlamsal verilerin bu birleşimi, otonom sistemlerin karar verme süreçleri için vazgeçilmezdir. Bununla birlikte, depolama, bant genişliği ve güvenli işleme konularında ciddi zorluklar ortaya çıkar.

Otonom araçların ürettiği veri ölçeği, yalnızca yerel depolamaya güvenmeyi pratik olmaktan çıkarır. Araçlar, engel tespiti veya çarpışma önleme gibi görevler için bazı gerçek zamanlı işlemleri yerel olarak gerçekleştirse de, uzun vadeli depolama ve büyük ölçekli analizler genellikle buluta aktarılır. Merkezi bulut altyapıları, sınırsız sayılabilecek depolama kapasitesi (Khan ve ark, 2024), derin öğrenme modellerinin yeniden eğitimi için güçlü hesaplama kaynakları ve binlerce araçtan gelen verilerin birleştirilmesi imkânı sunar. Bu birleştirme, nesne tanıma doğruluğunu, harita güncellemelerini ve rota optimizasyonunu geliştirir. Bulut altyapısının kullanımı büyük faydalar getirse de, tam anlamıyla güvenilemeyecek dış platformlara bağımlılık da yaratır. Yolların ve yayaların görüntüleriyle birlikte hassas konum bilgileri yüksek derecede gizli verilerdir ve ele geçirildiklerinde ya da kötüye kullanıldıklarında özel bilgilerin açığa çıkmasına neden olabilir.

Bulut platformları, otonom sürüş teknolojilerinin ölçeklendirilmesi için vazgeçilmezdir. Bu nedenle verilerin güvenliğini sağlayacak gelişmiş koruma tekniklerine ihtiyaç duyulmaktadır. Temelde, otonom araçlar daha geniş bir iletişim ekosistemine bağlıdır ve bu ekosistem, yoldaki diğer nesnelerle bilgi alışverişini gerekli kılar. Araçtan araca iletişim, yakın çevredeki araçlarla hız ve konum gibi bilgilerin paylaşılmasına imkân tanır (Zeadally ve ark, 2020). Araçtan altyapıya iletişim ise trafik ışıkları, yol kenarı birimleri ve gişelerle etkileşimi mümkün kılar.

Sürücüsüz araçların üçüncü taraflarla olan iletişimi ise verilerin bulut gibi merkezi sunuculara iletilmesini sağlar. Ayrıca bu araçlar, yayaların taşıdığı akıllı cihazları kullanarak araç-yaya iletişimini de gerçekleştirmektedir.

Bu iletişim mimarileri, verimliliği ve güvenliği artırarak siber saldırı riskini azaltmayı hedefler. Ancak saldırganlar zaman zaman geçmiş kayıtları ya da sahte bilgileri kullanarak bu iletişim mekanizmalarını bozabilir. Örneğin, ortadaki adam (man-in-the-middle) saldırılarıyla bulut sunucuları ile sürücüsüz araçlar arasındaki iletişim engellenip manipüle edilebilir (Gothwal ve ark, 2023). Yanıltıcı veri enjeksiyonları da önemli bir tehdittir; saldırganlar sahte trafik ışığı bilgileri, hava durumu ya da yol koşulları verileri sunabilir. Bu tür girişimler, araçtaki yolcular açısından ciddi güvenlik riskleri doğurur.

Otonom araçların ürettiği devasa veri hacimleri her zaman güvenlik ve mahremiyet açıklarıyla bağlantılıdır. Yüksek çözünürlüklü görüntüler, yüzler, plaka numaraları ya da diğer kişisel ayrıntıları yakalayabilir. Konum ve navigasyon verileri, sürücünün rutinlerini ve hareket kalıplarını ortaya koyabilir. Bu bilgilerin sızması durumunda, gözetim ya da kötü niyetli amaçlarla kullanılma ihtimali yüksektir. Geleneksel şifreleme yöntemleri, örneğin Gelişmiş Şifreleme Standardı (AES) ya da Rivest–Shamir–Adleman (RSA), iletişim kanallarını güvence altına almak için yaygın şekilde kullanılmaktadır. Ancak bu yöntemlerde verilerin işlenmeden önce çözülmesi gerekir. Bu durum, verilerin bulut ortamına aktarılıp YZ modellerinde analiz edilmesi veya eğitilmesi sırasında önemli bir güvenlik açığı yaratır. Tam homomorfik şifreleme (FHE) bu noktada potansiyel bir çözüm sunar. FHE, verilerin şifreli hâlleri üzerinde doğrudan hesaplama yapılmasına imkân tanır (Rauthan, 2022). Böylece veriler toplama, iletim ve işleme aşamalarının tamamında şifreli kalır; bu da çok daha güçlü bir gizlilik ve saldırılara karşı dayanıklılık sağlar.

2.3. Bulut Platformları, Hesaplama ve Güvenlik Zorlukları

Bulut bilişim, modern dijital altyapının vazgeçilmez bir parçası hâline gelmiştir. Kullanıcıların pahalı donanımlara yatırım yapmasına gerek kalmadan geniş depolama, ölçeklenebilir kaynaklar ve güçlü hesaplama yeteneklerine erişim sağlar. Akıllı ulaşım ve otonom sürüş bağlamında bulut platformları, araçlar tarafından üretilen büyük veri hacimlerinin depolanması ve işlenmesi için merkezi bir merkez görevi görür. Bu platformlar gelişmiş analizleri mümkün kılar, YZ eğitimini destekler ve gerçek

zamanlı karar desteği sağlar. Otonom araçlar günlük olarak terabaytlarca sensör verisi üretmektedir. Engel algılama gibi bazı görevler araç üzerindeki bilgisayarlar tarafından yerel olarak gerçekleştirilmek zorundadır. Ancak pek çok işlem, tek bir aracın kapasitesini aşan büyük ölçekli hesaplama gerektirir. Bulut platformları bu sorunları şu yollarla çözer:

- Esnek depolama kapasitesi sayesinde çok büyük hacimli görüntüleri ve yüksek kaliteli görüntü akışlarını işleyebilir.
- Çok sayıda araçtan toplanan veriler üzerinde derin öğrenme modellerinin yeniden eğitilmesi için verimli hesaplama olanağı sunar.
- Birden fazla kaynaktan gelen verilerin paylaşımı ve iş birliği sayesinde güvenlik özelliklerinin, trafik tahminlerinin ve haritaların geliştirilmesine katkı sağlar.

Bulut altyapısının sürücüsüz araçlarla bütünleştirilmesi pek çok avantaj sağlamaktadır (Silva ve ark, 2021). Öncelikle, depolama ve hesaplama maliyetlerini tekil araçlar açısından azaltır. Böylece her araçta pahalı donanım bulundurulmasına gerek kalmaz; yalnızca verileri buluta aktaran daha basit donanımlar yeterli olur. İkinci olarak, merkezi sunucular hava koşullarını ve yol yoğunluğunu izleyerek araçların daha verimli güzergâh seçmesine ve görevleri güvenli biçimde yerine getirmesine yardımcı olur. Üçüncü olarak, araştırmacılar ve üreticiler buluttaki veriye erişerek araç performansına dair sonuçlar elde edebilir; bu da sürüş modellerinin geliştirilmesine ve farklı ortamlara uyarlanmasına katkı sağlar. Bir yandan bulut sunucuları yüksek düzeyde ve hızlı hesaplama avantajı sunarken, diğer yandan aşırı kullanım güvenlik açıklarını da beraberinde getirmektedir. Bulut sunucuları çoğunlukla çok amaçlı olarak kullanılmakta; kimi zaman bireysel kullanıcıların ihtiyaçları, kimi zaman da işletmelerin merkezi sistemleri bu altyapı üzerinden çalıştırılmaktadır. Bu durum, güvenlik açısından ek riskler doğurmaktadır.

Bulut üzerindeki temel sorunlardan biri, herhangi bir saldırganın verilere erişim sağlaması hâlinde özel verileri kolayca değiştirebilmesidir. Bulut sunucularının yönetimi, bu sunuculara erişim hakkına sahip olduğundan, yönetici konumundaki kişiler de potansiyel bir saldırgan olabilir. Bir diğer önemli sorun ise çıkarım saldırılarıdır. Bu saldırılarda, saldırganlar meta verilerden, trafik hacminden veya erişim desenlerinden bilgi çıkarsayabilir. Örneğin, veriler belirli bir konumdan buluta yüklenirken, sunucu yöneticileri bu yüklemenin hangi konumdan yapıldığını

öğrenebilir (Hussain Akbar ve ark, 2023). Böylece saldırganlar araçların güzergâhlarını takip edebilir. Otonom araçlarda, bu tür meta veri sızıntıları aracılığıyla araç konumunun ve diğer kritik bilgilerin açığa çıkma riski oldukça yüksektir. Bu nedenle, otonom araçlarda bulut kullanımında güvenlik ile verimlilik arasında bir denge sağlanmalıdır. Bulut sunucuları merkezi sistem üzerinden araçların zamanında karar almasına yardımcı olsa da, yaşam döngüsünün her aşamasında üst düzey güvenliğe ihtiyaç vardır. Geleneksel şifreleme yöntemleri, hesaplama yapılabilmesi için verilerin önce çözülmesini gerektirdiğinden, her ne kadar iletim ve depolama sırasında gizlilik sağladıklarını iddia etseler de risk taşımaktadır.

Bu verilerin güvenliğini sağlamak için, şifreli veriler üzerinde doğrudan hesaplama yapabilen yöntemlere ihtiyaç vardır. Tam homomorfik şifreleme (FHE), veriler üzerinde şifre çözmeden işlem yapılmasına olanak tanır. Bu sayede hassas görüntüler ve sensör okumaları hiçbir zaman bulut hizmet sağlayıcısına ifşa edilmez (Yi ve ark, 2014). Her ne kadar bu yöntem günümüzde daha fazla hesaplama kaynağı gerektirse de, otonom araçlar için güvenli ve güvenilir bulut tabanlı bir yapı oluşturma yolunda umut verici bir yaklaşım olarak değerlendirilmektedir.

2.4. Kriptografi Temelleri

Bilgisayar dünyasında güvenli iletişimin temeli kriptografidir. Kriptografi, yetkisiz erişime karşı koruma sağlarken gizlilik, bütünlük ve kimlik doğrulama güvencesi verir. Otonom araçlar ve bulut tabanlı sistemler bağlamında kriptografi, hassas sensör verilerini, görüntü akışlarını ve navigasyon kayıtlarını korumak açısından kritik öneme sahiptir. Bu bölümde, günümüzde akıllı ulaşım sistemlerinde kullanılan başlıca kriptografi yöntemleri, avantajları ve zayıf yönleri ele alınmaktadır. Simetrik kriptografide, hem şifreleme hem de şifre çözme için aynı gizli anahtar kullanılır. En bilinen simetrik algoritmalar Veri Şifreleme Standardı (DES) ve Gelişmiş Şifreleme Standardı'dır (AES) (Sood ve ark, 2023). Bu yöntemlerin yaygın olmasının temel nedeni, hızlı olmaları ve işlem gücü sınırlı cihazlarda da kullanılabilmeleridir. Simetrik şifrelemede, düz metin gizli anahtar ile şifrelenerek şifreli metin (ciphertext) elde edilir. Alıcı, orijinal mesajı çözmek için aynı anahtara ihtiyaç duyar. Anahtar gizli tutulduğu sürece bu yöntem etkili ve güvenlidir. Ancak, iletişim kuran taraflar arasında gizli anahtarın güvenli bir şekilde dağıtılması zorluk teşkil etmektedir. Anahtarın ele geçirilmesi durumunda, tüm sistem savunmasız hâle gelir.

Simetrik şifreleme, otonom araçlarda yerleşik sensörler ile yerel işlemciler arasındaki kısa süreli iletişimin korunması için kullanılabilir. Ayrıca, görüntü verilerinin iletimden önce hızlı bir şekilde şifrelenmesine de uygulanabilir. Ancak, bu yöntemin en zayıf noktası, özellikle araçların çok sayıda dış sistemle iletişim kurmak zorunda kaldığı büyük ağlarda anahtar yönetimi alanındadır. Açık anahtar şifrelemesi (asimetrik şifreleme olarak da bilinir), biri şifreleme, diğeri ise şifre çözme için kullanılan bir açık anahtar ve bir özel anahtar gerektiren kriptografik bir sistemdir. Rivest-Shamir-Adleman algoritması ve Eliptik Eğri Kriptografisi iyi bilinen algoritmalardır (Savadatti ve ark, 2025). Bu şema, simetrik tekniklerde olduğu gibi tek bir gizli anahtarın paylaşımını gerektirmez ve bu da onu açık ağlarda daha uygun hale getirir. Bir gönderici, alıcının açık anahtarını kullanarak bilgiyi şifreler. Veriler yalnızca alıcıda güvenli bir şekilde saklanan özel anahtar ile çözülebilir. Bu model, şifreleme anahtarı herkese açık olarak dağıtılmış olsa bile iletişim güvenliği sağlar. Asimetrik şifreleme ayrıca dijital imzaları da kolaylaştırır; bunlar mesajların gerçekliğini doğrulamak için kullanılır. Otonom araçlarda, araçtan araca (V2V) ve araçtan altyapıya (V2I) iletişimlerde, anahtarların değişimi ve kimlik doğrulama gerektiğinde asimetrik şifreleme kullanışlıdır. Ancak, asimetrik algoritmalar simetrik algoritmalara kıyasla daha yavaş çalışır ve daha fazla işlem gücü gerektirir. Bu da onları gerçek zamanlı video şifreleme gibi yüksek hızlı işlemler için uygun olmayan hale getirir.

Her iki teknik de yaygın olarak kullanılmasına rağmen, ortak bir zorlukları vardır: şifrelenmiş verilerin işlenebilmesi için çözülmesi gerekir. Bu çözme işlemi, araç verilerinin bulut sunucular üzerinden analiz edildiği veya YZ modelleri eğitildiği durumlarda hassas bilgileri ortaya çıkarır. Veriler çözüldüğünde, bulut sağlayıcısı tarafından kötüye kullanılmaya veya dış saldırganların müdahalesine açık hale gelir. Bu zorluk, kullanılabilirlik ile güvenlik arasında önemli bir kopukluk yaratır. Otonom araçlarda sensör ve görüntü verileri etkin bir şekilde korunmalı, ancak aynı zamanda doğru tanıma ve güvenli navigasyon sağlamak için güçlü bulut işlemeye de izin verilmelidir. Geleneksel kriptografi her iki gereksinimi aynı anda karşılayamaz. Eski tekniklerin zayıflıkları, bilginin tüm yaşam döngüsü boyunca korunmasını sağlayacak yeni kriptografik çözümlerin gerekliliğini ortaya koymaktadır. Tam homomorfik şifreleme (FHE), veriler çözülmeden doğrudan şifrelenmiş veriler üzerinde hesaplama yapılmasına imkân tanıdığı için öne çıkan potansiyel bir çözüm olarak görülmektedir

(Marcolla ve ark, 2022a). Bu yenilik, simetrik ve asimetrik yöntemlerin sınırlamalarını ortadan kaldıracak ve otonom sürüşte bulut tabanlı analizler için güvenli bir temel sunacaktır.

2.5. Homomorfik Şifreleme

Homomorfik şifreleme, çağdaş kriptografinin en önemli katkılarından biridir. Bu yöntem, ilgili açık metni ortaya çıkarmadan şifrelenmiş veriler üzerinde matematiksel işlemlerin yapılmasına olanak tanır (Chase ve ark, 2017). Bu işlemlerin çıktısı yine şifreli biçimde kalır ve çözüldüğünde, sanki işlemler doğrudan şifrelenmemiş veriler üzerinde yapılmış gibi doğru sonuçlar elde edilir. Bu özellik, hassas verilerin güvenilir olmayan sunucular tarafından işlendiği bulut bilişim sistemlerinde özellikle faydalıdır.

Geleneksel şifreleme yöntemlerinde veriler yalnızca aktarım sırasında veya depolama aşamasında korunur. Ancak analiz yapılabilmesi için verilerin çözülmesi gerekir; bu durum verileri kötüye kullanıma açık hale getirir. Homomorfik şifreleme ise bu zayıflığı ortadan kaldırır. Veriyi yaşam döngüsü boyunca şifreli halde tutarak, aktif hesaplamalarda bile gizliliğin korunmasını sağlar. Bu yaklaşım, güvenli veri işleme konusuna bakış açısını kökten değiştirmiştir. Artık gizlilik uğruna işlevsellikten vazgeçmeye gerek yoktur; homomorfik şifreleme sayesinde her ikisi aynı anda mümkün hale gelmiştir. Homomorfik şifrelemenin farklı türleri bulunmaktadır:

- **Kısmi Homomorfik Şifreleme (PHE):** Yalnızca toplama veya çarpma işlemlerinden birinin şifreli veriler üzerinde yapılmasına izin verir. Örneğin, Rivest Shamir Adleman algoritması çarpma, Paillier şeması ise toplama işlemlerini desteklemektedir. Pratik olmakla birlikte, bu yöntemler gerçek dünyada sıkça ihtiyaç duyulan karmaşık işlem kombinasyonlarını desteklemediği için sınırlıdır (T. Wu ve ark, 2021).
- **Kısmen Homomorfik Şifreleme (SHE):** Ara bir çözüm olarak sunulmuştur. Hem toplama hem çarpma işlemlerini yapabilir, ancak sınırlı sayıda işlemle kısıtlıdır. Belirli bir seviyenin ötesinde şifre çok gürültülü hale gelir ve geçerli sonuçlar üretmez. Bu yöntem bir gelişme sağlasa da, YZ veya büyük veri analitiği gibi alanlarda gerekli olan karmaşık hesaplamaları desteklemekte yetersizdir (Subramaniyaswamy ve ark, 2022).

- **Tam Homomorfik Şifreleme (FHE):** Gerçek anlamda bir atılım olmuştur. 2009 yılında Craig Gentry, şifreli veriler üzerinde sınırsız işlem yapılabilmesini mümkün kılan ilk yapıyı ortaya koymuştur. Onun stratejisi, şifreli metinlerdeki gürültü seviyesini düşürerek tekrar tekrar hesaplama yapılmasını sağlayan *bootstrapping* kavramını geliştirmiştir. İlk tasarım hesaplama açısından çok maliyetli olsa da, sonraki araştırmaların önünü açmış ve hızlı ilerlemelere yol açmıştır (Rauthan, 2022).

Verimliliği ve kullanılabilirliği artırmak için farklı homomorfik şifreleme şemaları geliştirilmiştir. Bu şemaların her biri belirli veri türleri ve işlemler için optimize edilmiştir:

- **Brakerski Fan Vaikuntanathan (BFV) Şeması:** Özellikle tamsayı işlemlerine uyarlanmıştır. Hızlıdır, yaygın olarak kullanılmaktadır ve birçok uygulamanın temelini oluşturur. BFV, aritmetik doğruluğu ve dengeli yapısı nedeniyle araştırmalarda sıklıkla tercih edilmektedir (Kuo ve ark, 2023).
- **Brakerski Gentry Vaikuntanathan (BGV) Şeması:** BFV gibi tamsayı işlemlerini destekler, ancak modüler aritmetiğin nasıl ele alınacağı konusunda ek esneklik sağlar. Yapılandırılmış sayısal hesaplamaların bulunduğu uygulamalarda güçlü bir şekilde kullanılabilir (Geelen ve ark, 2023).
- **Cheon Kim Kim Song (CKKS) Şeması:** Gerçek sayılar üzerinde yaklaşık hesaplama yapabilmek üzere tasarlanmıştır. Bu özelliği, kesin doğruluğun her zaman gerekli olmadığı makine öğrenmesi ve YZ uygulamaları için uygun hale getirir (Acar ve ark, 2019b).

Bu planların birleşimi, homomorfik şifrelemenin artık soyut bir kavram olmaktan çıkıp, uygulanabilir bir teknolojiye dönüştüğünü ve farklı uygulamalara göre çeşitlenmiş alternatifleri olduğunu göstermektedir. Akıllı araçlar; görüntüler, LiDAR nokta bulutları ve konum kayıtları gibi hassas veriler üretmektedir. Bu veriler buluta aktarıldığında güvenli kalmalı, ancak aynı zamanda YZ modellerinin eğitilmesi ve güvenlik yeteneklerinin artırılması için de kullanılabilmelidir.

Geleneksel şifreleme, bir ikilem ortaya koyar: Veriyi şifreli tutmak güvenliği artırır ama kullanılabilirliği azaltır; şifreyi çözmek ise kullanılabilirliği artırır ama güvenlik risklerini beraberinde getirir. Homomorfik şifreleme bu ikilemi ortadan kaldırır. Bulut sunucuları, veriyi hiç çözmeden, şifreli haldeyken gerçek zamanlı analiz, nesne tespiti

ve navigasyon güncellemeleri yapabilir. Bu yaklaşım hem gizliliği hem de işlevselliği korur. Örneğin, eğitilmiş bir model ile bulut platformu şifreli kamera görüntülerinde nesneleri algılayabilir. Çıktılar yine şifreli olarak araca gönderilir ve yalnızca araç tarafından çözülür. Böylece ham veriler hiçbir zaman açığa çıkmaz. Bu özellik, kişisel gizlilik, yol güvenliği ve yasal düzenlemelere uyumun aynı anda önceliklendirildiği bir ortamda özellikle kritik bir yetenektir. Bununla birlikte, homomorfik şifreleme zorluklardan arınmış değildir. Hesaplamalar, geleneksel yöntemlere kıyasla çok daha maliyetlidir. Şifreli verilerle çalışmak daha yavaş işlem yapılmasına yol açar ve daha fazla bellek tüketir (Acar ve ark, 2019b). Ayrıca şifreli metinlerin büyümesini yönetmek de bir diğer zorluktur; bootstrapping ve modulus switching gibi yöntemler bu soruna çözüm sunmaktadır. Araştırmacılar, daha üstün algoritmalar, optimize edilmiş kütüphaneler ve donanım hızlandırmaları ile verimliliği artırmaya devam etmektedir. Nihai amaç, homomorfik şifrelemenin otonom sürüş gibi büyük ölçekli gerçek zamanlı sistemlerde uygulanabilir hale gelmesidir.

2.6. Nesnelerin İnterneti (IoT) ve Bulutta Şifreleme

IoT ve bulut tabanlı sistemlerde güvenliğin anahtarı şifrelemedir. Milyarlarca cihazın birbirine bağlı olduğu bir ortamda, gizliliği ve güveni sağlamak için etkili yöntemlere ihtiyaç duyulmaktadır. IoT; sensörler, kameralar ve akıllı cihazlar şeklinde bitmek tükenmek bilmeyen veri kaynakları üretirken, bulut bu verilerin depolanıp işlenmesinde merkezi bir alan olarak işlev görmektedir. Şifreleme olmadığı durumda, hassas bilgiler her iletim ve depolama noktasında saldırılara karşı savunmasız hale gelir. Bu bölümde, şifrelemenin IoT ve buluta uygulanışı, karşılaşılan çevresel zorluklar ve neden daha gelişmiş önlemlere ihtiyaç duyulduğu incelenmektedir. IoT, modern altyapının bir parçasıdır. Akıllı evlerde, tıbbi cihazlarda, endüstriyel kontrol sistemlerinde ve otonom araçlarda sensörlerin kullanımı giderek yaygınlaşmaktadır. Bu cihazlardan gelen bilgiler ağlar üzerinden uygulamalara aktarılır; burada işlenir, analiz edilir ve sonrasında aksiyon alınır. Bu veri akışını güvence altına almak için kullanılan başlıca kriptografi araçları, uzun süredir AES ve RSA gibi geleneksel şifreleme yöntemleridir (Sood ve ark, 2023).

Simetrik şifreleme, küçük ölçekli uygulamalarda yaygındır çünkü hafif ve hızlıdır. Örneğin, giyilebilir cihazlar veya akıllı sayaçlar ölçümlerini şifreleyip merkezi sunuculara gönderebilir. Bu yöntem, düşük işlem gücüne ve sınırlı pil ömrüne sahip

cihazlar için önemli bir verimlilik sağlar. Asimetrik şifreleme ise doğrulama ve anahtar dağıtımı gerektiğinde kullanılır. Örneğin, akıllı bir otomobil, karşılaştığı trafik kontrol sistemi güvenilir mi diye karar vermek için açık anahtar kriptografisini kullanabilir. Bu sayede sahtecilik önlenir ve yalnızca yetkilendirilmiş kişiler bilgi paylaşabilir. Bu sistemler, iletim ve depolama sürecinde yüksek koruma sağlasa da ortak bir zayıflıkları vardır: analiz yapılmadan önce şifrelenmiş verilerin çözülmesi gerekir. Küçük yerel sistemlerde bu büyük bir sorun oluşturmaz, ancak bulut söz konusu olduğunda kritik bir faktör hâline gelir.

Bulut platformları, organizasyonların veri işleme ve analiz yöntemlerini kökten değiştirmiştir. Bulut, ölçeklenebilirlik, yüksek hızlı hesaplama ve yerel cihazlarla kıyaslanamayacak ortak altyapı sağlar (Mathur, 2024). IoT açısından, bulut, her gün üretilen büyük veri miktarını yönetmek için tek mantıklı seçenektir. Veriler, buluta iletilmeden ve uzak sunucularda depolanmadan önce şifrelenir. Asimetrik şifreleme, güvenli iletişim kanalları sağlar ve kimlik doğrulamaya yardımcı olurken, simetrik şifreleme depolanan büyük veri miktarını güvence altına alır. Ayrıca, birçok hizmet sağlayıcı, veri aktarımı sırasında güvenliği artırmak için Güvenli Yuva Katmanı (SSL) veya Taşıma Katmanı Güvenliği (TLS) gibi ek koruma katmanları kullanır (I. Gupta ve ark, 2022). Ancak zorluk, bulutun kendisinin şifrelenmiş verileri işlemesi gerektiğinde ortaya çıkar. Analitik platformlar, derin öğrenme modelleri ve istatistiksel araçlar, çalışabilmek için düz metne ihtiyaç duyar. Bu, bulutun verileri çözmesini ve ardından hesaplamaları yapmasını gerektirir. İşte bu noktada kullanıcı, hizmet sağlayıcıya tam güvenmek zorundadır. Eğer sağlayıcı ihlal edilirse veya kötü niyetli iç çalışanlar varsa, hassas bilgiler açığa çıkabilir.

2.7. Bulut Tabanlı Sistemlerde Homomorfik Şifreleme

Bulut bilişimin hızlı gelişimi, verilerin depolanma, işlenme ve paylaşılma biçimini değiştirmiştir. Bulut platformları, yerel sürücülerde yönetilmesi pratik olmayan büyük veri miktarını işlemek için organizasyonlar ve bireyler açısından vazgeçilmez hâle gelmiştir. Otonom araçlar ve diğer Nesnelerin İnterneti (IoT) sistemleri için de bulut artık bir gerekliliktir. Sonsuz depolama, hızlı hesaplamalar ve sınırsız cihazla veri birleştirme imkânı sunar. Bununla birlikte, bu bağımlılık güven ile ilgili ciddi soruları da beraberinde getirir. Veriler buluta yüklendiği anda, kullanıcı verilerin gizli ve güvenli olacağına inanmak zorundadır. Geleneksel şifreleme, verileri depolama ve

iletim sırasında korurken, bulutun hesaplama yapması gerektiğinde bu güvenliği sağlamaz. Bu dezavantaj, araştırmacıları bulutta güvenli işlem yapmak için homomorfik şifreleme yöntemlerine yöneltmiştir (Mahato ve ark, 2023).

Bulut sistemleri, büyük ölçekli veri analizlerini gerçekleştirecek şekilde tasarlanmıştır. Bu sistemler derin öğrenme modellerini çalıştırır, öngörüsel analizler yapar ve gerçek zamanlı karar destek mekanizmalarını sağlar. Otonom araçlar açısından bulut, binlerce görüntü akışı, sensör verisi ve seyahat rotasının bir araya getirildiği ve bu sayede tanıma modelleri ile trafik tahminlerinin güncellendiği bir merkezdir. Bu hizmetleri sunabilmek için bulut, ham veriyi işlemek zorundadır. Geleneksel yöntemlerle veriler şifrelenmiş olduğunda, bulut veriye şifre çözmeden erişemez. Bu durum, yüzler, plaka numaraları ve seyahat alışkanlıkları gibi hassas bilgileri, sahibinin yetkisinin ötesinde sunucuya açık hâle getirir. Sağlayıcılar ıyi niyetli davransalar bile, veri ihlali, içeriden tehditler veya devlet gözetimi durumunda gizlilik tehlikeye girebilir.

Homomorfik şifreleme bu sorunun çözümüdür çünkü bulut, şifrelenmiş veriler üzerinde işlem yapabilir (Prabu Kanna ve ark, 2019). Ham veriler asla bulut tarafından görünmez, fakat kullanıcıya faydalı sonuçlar sunulabilir. Bulut sistemlerinde homomorfik şifrelemenin uygulanması kavramı teoride ilk kez ortaya atılmıştır. Araştırmacılar, şifreli arama, şifreli oylama ve özel sorguların temel bilgileri açığa çıkarmadan yapılabileceğini göstermiştir. Zamanla, sağlık, finans ve iletişim hizmetleri alanlarında deneysel sistemler geliştirilmiştir. Örneğin, tıp araştırmacıları, hastalarının verilerini analiz ederken şifreleyebildiklerini ve böylece gizlilik yasalarına uygun kaldıklarını göstermiştir. Benzer şekilde, bankalar, kişisel kayıtları açığa çıkarmadan şifreli dolandırıcılık tespiti üzerinde deneyler yapmıştır. Ulaşım alanında, bulut bilişimin şifrelenmiş konum ve sensör verilerini işleyerek navigasyon hizmetlerini destekleme potansiyeli incelenmiştir.

Bu yayınlar, homomorfik şifrelemenin gerçek sistemlerde uygulanabilir olduğuna dair kanıt sağlamıştır, ancak performans hâlâ büyük ölçekli uygulamalar için bir engel teşkil etmektedir (Tebaa ve ark, 2012). Bulut bilişimde gizlilik tek önemli sorun değildir; doğruluk da kritik bir meseledir. Hesaplamaların buluta devredildiği durumlarda, veri sahibi sonuçların doğruluğundan emin olmalıdır. Bu sorunu aşmak için doğrulanabilir homomorfik şifreleme (VHE) geliştirilmiştir. VHE, kullanıcıya bulutun işlemleri dürüstçe ve manipülasyonsuz gerçekleştirdiğini doğrulama imkânı verir. Örneğin, otonom bir araç şifrelenmiş video karelerini yol işaretlerini tanımak

için gönderdiğinde, VHE sayesinde kullanıcı dönen sonuçların şifrelenmiş girdiyle uyumlu olduğunu doğrulayabilir. Bu sistem güvenini sağlar ve bulut sağlayıcısı sahtekarlık yapamaz veya kısmi sonuç sunamaz.

Buna rağmen, homomorfik şifrelemenin bulut bilişim sistemlerindeki temel zorluğu verimliliktir. Şifrelenmiş veriler, düz metne göre çok daha yavaş işlenir (Benzekki ve ark, 2016). Şifreli veriler depolama ve iletim için daha fazla alan ve bant genişliği gerektirir. Gerçek zamanlı sistemlerde, örneğin otonom araçlarda, bu ek yükler göz ardı edilemez. Araştırmacılar bu yükü azaltmak için çeşitli yöntemler önermiştir. Bootstrapping, şifreli veriyi yenileyen ve gürültüyü azaltan bir mekanizmadır, ancak maliyetlidir. Modül değişimi ve yeniden lineerleştirme gibi yöntemler, hesaplamaların karmaşıklığını azaltarak verimliliği artırır. Ayrıca, Grafik İşlem Birimleri (GPU) ve Alan Programlanabilir Kapı Dizileri (FPGA) gibi donanım hızlandırmaları kullanılarak şifreli işlemler hızlandırılmaya çalışılmıştır (Renugadevi ve ark, 2022).

Buna rağmen, teorik olarak mümkün olan ile zaman açısından hassas uygulamaların gereksinimleri arasındaki fark hâlâ mevcuttur. Güvenli bulut bilişimin en yüksek öncelikli gereksinimlerinden biri otonom araçlardır (Muthanna ve ark, 2024). Bu araçlar her gün terabaytlarca görsel ve mekansal veri üretir; bu veriler analiz edilip tanıma modelleri güncellenmeli, navigasyon geliştirilip filo arasında paylaşılmalıdır. Bu verilerin buluta şifrelenmemiş biçimde gönderilmesi, kabul edilemez gizlilik ve güvenlik sorunları oluşturur. Homomorfik şifreleme, işlevsellik ve gizlilik arasında bir denge sağlar.

2.8. Şifrelenmiş Görüntü İşleme ve Arama

Dijital sistemlerde görüntü kullanımının artması, veri güvenliği açısından hem yeni fırsatlar hem de yeni zorluklar ortaya çıkarmıştır. Sağlık taramalarından uydu görüntülerine veya otonom araçların video akışlarına kadar olan görseller, çoğu zaman paylaşılması ve işlenmesi gereken değerli bilgiler içerir. Aynı zamanda, bu tür görüntüler kimlikler, konumlar veya alışkanlıklar gibi hassas bilgileri açığa çıkarabilir. Bu bilgileri güvence altına almak ve analiz sırasında kullanılabilir hâlde tutmak, artık başlıca araştırma alanlarından biri olmuştur.

Şifrelenmiş görüntüler üzerinden görüntü işleme ve arama, bu zorluğun üstesinden gelmeye yönelik bir çabadır; yani sistemlerin görüntüleri çözülmemiş hâliyle ifşa etmeden hatırlayabilmesi veya inceleyebilmesini sağlamaktır. Başlangıçta

görüntülerin korunması klasik şifreleme yöntemlerine dayanıyordu. Standart şifreleme algoritmaları, görüntüleri şifrelemek ve güvenli sunucularda depolamak için kullanıldı. Arama veya analiz gerektiğinde görüntülerin şifresi çözülmek zorundaydı (Kamal ve ark, 2021). Bu yaklaşım depolama ve iletim sırasında güvenliydi, ancak işlem sırasında başarısız oluyordu; görüntüler şifresi çözüldüğünde kötüye kullanım riski ortaya çıkıyordu. Bu sorunu çözmek amacıyla araştırmacılar, gizliliği koruyan görüntü arama yöntemleri geliştirmeye başladılar. İlk yaklaşım, özellik çıkarımı tabanlıydı. Sistemler tüm görüntüyü kullanmak yerine, kenarlar, dokular veya anahtar noktalar gibi özellikleri çıkarıp bunlar üzerinden arama yapıyordu. Kullanılan standart yöntemler arasında Scale Invariant Feature Transform (SIFT), Local Binary Patterns (LBP) ve Discrete Cosine Transform (DCT) yer alıyordu (Das ve ark, 2023; Jiang ve ark, 2021). Bu teknikler, yalnızca türetilmiş özellikler depolandığı veya gönderildiği için bir tür koruma sağlıyordu, tüm görüntü paylaşılmıyordu.

Buna rağmen, saldırganlar birçok durumda özelliklerden kaba görüntüler oluşturabilir ve böylece gizlilik sızabilir. Daha iyi koruma ihtiyacı, gizliliği artıran teknolojilerin geliştirilmesini zorunlu kıldı. Bu yöntemler, hassas bilgileri gizlemeyi ve aramayı mümkün kılmayı amaçlıyordu. Bazı yöntemlerde, görüntü özellikleri depolamadan önce şifrelenirken, bazılarında görüntülerin dönüştürülmüş hâlleri oluşturularak eşleme ve benzerlik algoritmalarıyla uyumlu hâle getiriliyordu. Bu yaklaşımlar umut verici olsa da, birçok teknoloji kullanıcıya yanıltıcı bir güvenlik sağlıyordu. Şifrelenmiş özellikler bile, belirli durumlarda desen çıkarımı için kullanılabiliyordu. Ayrıca, bazı saldırganlar, arama sonuçları veya erişimlere dayanarak görüntü içeriklerini tahmin eden çıkarım saldırıları gerçekleştirebiliyordu. Bu nedenle, bu yaklaşımlar tıbbi görüntüleme veya akıllı ulaşım gibi yüksek güvenlik gerektiren alanlarda yeterli görülmedi. Derin öğrenmenin gelişmesiyle, araştırmacılar gizliliği koruyan görüntü arama için sinir ağlarını değerlendirdi. Konvolüsyonel Sinir Ağları (CNN) da şifrelenebilen veya gizlenebilen sıkıştırılmış özellikler üretmek üzere eğitildi ve depolandı (W. Li, 2021). Bu temsil biçimleri, benzerlik aramalarını etkin hâle getirirken görüntünün hassas bilgilerini gizliyordu. Gizliliği koruyan SIFT ve CNN tabanlı arama teknikleri, önceki yöntemlere göre daha iyi performans sağladı. Çalışmada önerilen metodoloji, bu araştırma alanında bir adım ileriye gitmeyi ve hem gizlilik hem de performans gereksinimlerini karşılayacak esnek bir şifreli arama algoritması sunmayı amaçlamaktadır.

2.9. Otonom Araçlar için Uygulamalar

Otonom araçlar da güvenli ve akıllı kararlar alabilmek için verilere bağımlıdır. Sensörler, kameralar ve iletişim sistemleri her saniye devasa miktarda bilgi üretmektedir. Bu bilgiler, navigasyon, engel tespiti ve karar alma gibi işlemlere yardımcı olmak için gerçek zamanlı olarak toplanmalı, işlenmeli ve analiz edilmelidir. Araçlar bu hesaplamaların tamamını yerel olarak gerçekleştirecek kapasitede olmadığından, bulut sistemleri araçlara destek sağlamakta merkezi bir rol oynar. Bu sırada, gizlilik ve güvenlik konusu kritik öneme sahiptir. Bilgiler açığa çıkarsa, sürücüler, yolcular ve trafik durumu hakkında hassas bilgiler ifşa olabilir. Homomorfik şifreleme ve gelişmiş yapay zeka, bu verileri otonom araçlarda güvenli ve etkili hâle getirmemize yardımcı olur. Araçlar Arası İnternet (Internet of Vehicles), IoT'nin bir uzantısıdır ve araçlar ile ulaşım sistemlerinin ağlar üzerinden birbirine bağlanmasını sağlar. Araçlar sürekli olarak diğer araçlar, yol altyapısı ve bulut platformları ile veri paylaşır. Bu tür iletişim güvenliği ve verimliliği artırır; çünkü araçlar birbirlerini tehlikelere, trafik sıkışıklıklarına veya kazalara karşı uyarabilir. Bununla birlikte, bu ortam gizlilik kaybına da açıktır. Kart kayıtları, video akışları ve iletişim günlükleri açıldığında kötüye kullanılabilir. Saldırganlar seyahat yollarını yeniden oluşturabilir, alışkanlıkları takip edebilir veya kişileri izleyebilir. Bu bilgileri nasıl koruyup aynı zamanda faydalı hâle getireceğimiz büyük bir zorluktur. Homomorfik şifrelemenin sağladığı çözüm şudur: veriler işlenmek üzere yüklendiğinde süreç boyunca gizli kalır ve kaybolmaz.

Nesne tanıma, otonom sürüşün en kritik faaliyetlerinden biridir. Araçların yol işaretlerini, yayaları, diğer araçları ve beklenmedik tehlikeleri doğru şekilde okuması gerekir. Bu görevde Konvolüsyonel Sinir Ağları (CNN) ve YOLO derin öğrenme modelleri standart araçlar hâline gelmiştir. Bu modeller, gerçek zamanlı görüntü akışlarını işleyebilir ve yüksek doğruluk sağlar. YOLO, yıllar içinde çeşitli yükseltmelerden geçmiş, her güncelleme hız, doğruluk ve esnekliği artırmıştır. En son sürümlerden YOLO sürüm dokuz, karmaşık sürüş ortamları için güçlü performans sunar. Bu modeller, bir karede birden çok nesneyi yüksek hatırlama ve ortalama doğruluk ile tanıyabilir. Homomorfik şifreleme sayesinde bu modeller bulut sağlayıcıları üzerinde çıplak görüntüleri açığa çıkarmadan çalıştırılabilir. Araç kameraları şifrelenmiş kareleri buluta gönderir ve model, doğrudan şifrelenmiş veriler

üzerinden nesneleri tanır. Sonuçlar şifreli formatta geri gönderilir ve sadece araç tarafından çözülür. Bu, güvenlik ve gizlilik sağlar.

Tüm nesne tespit algoritmaları arasında YOLO sürüm dokuz, doğruluk ve hız açısından en dengeli olarak bilinir. Yüksek çözünürlüklü görüntüleri kısa sürede yüksek tespit oranları ile desteklemek üzere geliştirilmiştir. Otonom araçlar için bu kritik öneme sahiptir; çünkü kararların saniyenin birkaç kesri içinde alınması gerekir. Yaya veya yol işaretinin algılanmasındaki gecikmeler kazalara yol açabilir. YOLO sürüm dokuz ayrıca önceki sürümlere göre daha iyi hatırlama ve ortalama doğruluk sunar. Bu, küçük veya kısmen gizlenmiş nesnelere karşı daha hassas olduğunu gösterir ve bu tür nesneler gerçek hayatta yaygındır. Bu model, homomorfik şifreleme ile birleştirildiğinde, araçlar nesneleri güvenli şekilde tespit edebilir ve bulutun hesaplama gücünden yararlanabilir. Homomorfik şifreleme, otonom araçlarda nesne tanımanın ötesinde de faydalıdır. Trafik desenleri analizi, rota optimizasyonu ve öngörücü bakım gibi işlemler de şifrelenmiş veriler kullanılarak gerçekleştirilebilir.

2.10. Tekniklerin Karşılaştırmalı İncelemesi

Homomorfik şifreleme, adım adım ilerleyen bir süreçtir ve her yeni adım kendine özgü yetenekler ve sınırlamalar taşır. Bu seviyeler genellikle Kısmi Homomorfik Şifreleme (PHE) T. Wu ve ark. (2021), Biraz Homomorfik Şifreleme (SHE) Subramaniyaswamy ve ark. (2022) ve Tam Homomorfik Şifreleme (FHE) Rauthan. (2022) olarak sınıflandırılır ve her biri kendi alanında önemli bir ilerleme olarak kabul edilmiştir. Karşılaştırmalı analiz büyük önem taşır; çünkü bu, güvenlik, hesaplama verimliliği ve kullanım kolaylığı arasındaki dengeyi anlamamıza yardımcı olur. IoT ve otonom araç sistemlerinde kullanılacak belirli bir şifreleme yöntemi seçimi, doğrudan performans ölçütlerini, depolama ihtiyaçlarını ve bulut tabanlı verilerin güvenli işlenebilme yeteneğini etkiler. Homomorfik korumanın en basit türü PHE'dir. Bu yöntem, yalnızca tek bir işlem türüne toplama veya çarpma izin verir ve diğerini dışlar. Örneğin, RSA şeması şifreli metinlere çarpımsal homomorfizm özelliği kazandırırken; Paillier şeması Mohammed ve ark. (2023) toplamsal homomorfizm özelliği sunar. Bu mekanizmalar, kaynakları kısıtlı cihazlarda kullanıma uygun olacak şekilde verimli ve hafiftir; bu nedenle özellikle IoT ortamlarında tercih edilirler.

Bu sistemlerde PHE, temel istatistiksel hesaplamaları gerçekleştirmek için kullanılmıştır; örneğin sensör verilerinin toplanması gibi işlemler, veriyi çözmeden

yapılabilir. Ancak, yalnızca tek bir işlemi desteklemesi, karmaşık bulut tabanlı uygulamalarda örneğin derin öğrenme veya görüntü tanıma kullanımını sınırlar. SHE (Biraz Homomorfik Şifreleme), fonksiyonel kapsamı genişleterek hem toplama hem de çarpma işlemlerine izin verir, ancak sadece sınırlı sayıda ardışık işlem gerçekleştirilmesine olanak tanır. Her işlem yapıldığında, şifreli metnin gürültü bileşeni artar ve sonunda, bir yenileme işlemi (refreshing) uygulanmadan mesajı çözmek imkânsız hâle gelir. Bu durum, şifreli metinlerin bootstrapping işlemi gerektirmeden çalışabileceği maksimum hesaplama derinliğini sınırlar.

SHE, pratikte orta düzeyde karmaşık görevler için kullanılabilir; örneğin küçük makine öğrenimi modellerinin eğitimi veya bir veritabanına güvenli sorgular gönderilmesi gibi. Bu yöntem, tıp verilerinin analizi, finansal risk değerlendirmesi ve temel görüntü işleme gibi alanlarda uygulanmıştır. Otonom araç platformları bağlamında SHE, boyutu sınırlı şifreli sensör veri kümelerinin işlenmesini destekleyebilir; ancak video akışları veya gerçek zamanlı karar alma süreçleri için uygun değildir.

Otonom araç ekosisteminde, FHE güvenli nesne tanıma, trafik akışı tahmini ve filo genelinde makine öğrenimi uygulamalarını destekler. Araçlar şifreli görüntüleri buluta aktarabilir, böylece bulut derin öğrenme modellerini ham verilere erişmeden çalıştırabilir. Bu tür bir koruma, gizliliğin pratik işlevsellikle uzlaştırılabilmesi için gereklidir; ancak bununla ilişkili hesaplama maliyetleri önemli bir sınırlamadır. Her homomorfik şifreleme paradigmasının avantajları ve dezavantajları farklıdır ve bunlar ifade yeteneği ile işlem yükü açısından değerlendirilir. PHE hızlı ve kolaydır, ancak yalnızca tek bir işlem türüne izin verir. SHE her iki işlemi de sınırlı derinlikte destekler. FHE tam esneklik sunar, ancak yüksek hesaplama ve depolama maliyeti gerektirir. Bu nedenle, IoT uygulamaları sınırlı enerjiye sahipse PHE veya SHE kullanılabilir; ancak otonom araç uygulamaları bulutlara bağımlıysa, gerekli güvenlik ve işlevsellik gereksinimlerini karşılayabilecek tek çözüm FHE'dir. En önemli farkları vurgulamak için, Tablo 2.1 üç yöntemin temel özelliklerini özetlemektedir:

FHE (Tam Homomorfik Şifreleme), homomorfik fonksiyonelliğin zirvesidir; yani şifreli metinler üzerinde istenilen uzunlukta toplama ve çarpma zincirleri gerçekleştirilebilmesine olanak tanır. Bu tarihi ilerleme, kamu bulut sağlayıcılarının karmaşık algoritmaları—örneğin derin öğrenme çıkarımı—hiçbir zaman açık verilere dokunmadan çalıştırabilmesini sağlar. FHE, ilk olarak Gentry tarafından geliştirilmiş

ve daha sonra Brakerski Vaikuntanathan BV şeması, BGV şeması Geelen ve ark. (2023) ve CKKS şeması Lee ve ark. (2023) ile performans iyileştirilmiş ve gürültü büyümesi azaltılmıştır. Ancak hesaplama maliyetleri ve depolama kullanımı hâlâ kritik sorunlar arasındadır.

Tablo 2.1. Üç tür homomorfik şifrelemenin karşılaştırması

Yöntem	Desteklenen İşlemler	Hesaplama Derinliği	Verimlilik	Uygulamalar	Sınırlamalar
Kısmi Homomorfik Şifreleme (PHE)	Sadece toplama veya sadece çarpma	Tek Bir işlem için sınırsız	Çok verimli	Sensör değerlerinin toplanması, IoT'de basit istatistikler	Her iki işlemi birden yapamaz, karmaşık görevler için uygun değil
Biraz Homomorfik Şifreleme (SHE)	Toplama ve çarpma	Sınırlı derinlik (gürültü artar)	Orta düzey verimli	Tıbbi sorgular, küçük modeller, temel şifreli arama	Sınırlı sayıda işlem, gerçek zamanlı uygulamalarda ölçeklenemez
Tam Homomorfik Şifreleme (FHE)	Toplama ve çarpma	Sınırsız	Düşük verimlilik (araştırmalarla gelişiyor)	Şifreli derin öğrenme, bulutta görüntü işleme,	Yüksek hesaplama, büyük şifreli veri boyutu, yanıt süresi daha yavaş

Karşılaştırmalı çalışmalar göstermektedir ki, tüm uygulama ihtiyaçlarını karşılayabilecek tek bir şifreleme yöntemi bulunmamaktadır. PHE, temel işlemlerin yeterli olduğu hafif uygulamalarda tercih edilirken; SHE her iki işlemi de yapabilme olanağı sağlayarak orta bir çözüm sunar; FHE ise en güçlü gizlilik garantilerini ve en geniş algoritmik yetenekleri sağlayarak özellikle bulut tabanlı otonom araç sistemlerinde uygulanabilir hâle gelmektedir. Performans hâlâ bir sorun olsa da, araştırmalar bu boşluğu daraltmaya devam etmektedir. Bu tür avantaj ve dezavantajların iyi anlaşılması, araştırmacı ve mühendislerin sistemlerinde en uygun yöntemi seçmelerine yardımcı olacaktır. Otonom araçlar ve IoT uygulamalarında, hassas ve yüksek hacimli verilerin işlendiği durumlarda, FHE en umut verici çözüm olarak öne çıkar. Her veri işleme aşamasında gizliliği koruyabilme yeteneği, bulut üzerinde güvenli akıllı ulaşımın temelini oluşturur. Homomorfik şifreleme, şifrelenmiş veriler üzerinde, veriyi çözmeye gerek kalmadan hesaplamalar yapabilmeyi sağlayan

kriptografi alanında yeni bir kavramdır. Homomorfik şifreleme sistemleri, 1978'de Rivest, Adleman ve Dertouzos tarafından yapılan ilk çalışmalarla ortaya çıkmıştır (Issaoui ve ark, 2023).

2009 yılında Craig Gentry, şifrelenmiş veriler üzerinde çözme işlemi yapmadan rastgele hesaplamaların yapılmasını sağlayan bir FHE algoritması geliştirdi; böylece hassas verilerin gizliliği ve mahremiyeti garanti altına alındı (Kölsch ve ark, 2019). Homomorfik şifrelemenin önemi, toplama ve çarpma gibi işlemleri şifrelenmiş veriler üzerinde veri gizliliğini etkilemeden gerçekleştirebilmesidir (C. P. Gupta ve ark, 2013). Bu özellik, güvenli bulut bilişim mimarileri, bulutlarda özel veri işleme ve elektronik oylama sistemleri gibi birçok alanda uygulama bulmasına yol açmıştır (A. Abu Aziz ve ark, 2018; Benzekki ve ark, 2016; C. P. Gupta ve ark, 2013). Ayrıca, homomorfik şifreleme, VHE (Verifiable Homomorphic Encryption) gibi şifrelenmiş veriler üzerinde hesaplama bütünlüğünü koruyarak gizliliği kaybetmeden işlem yapılmasını sağlayan yöntemlerin geliştirilmesini de mümkün kılmıştır (Lai ve ark, 2014). Homomorfik şifrelemenin potansiyeli, IoT gibi alanlarda performans simülasyonları ile algoritmaların verimliliğinin test edilmesi yoluyla daha da artmıştır (Benzekki ve ark, 2016). Bulut depolamada daha gelişmiş FHE anahtarlarının geliştirilmesi de, homomorfik şifrelemenin gerçek dünyada daha pratik ve güvenli hâle getirilmesi için süregelen araştırmaların göstergesidir (R. Hu ve ark, 2016).

Alloghani ve ark. (2019), homomorfik şifrelemenin bulut bilişim ve büyük veri alanlarındaki gizlilik ve güvenlik sorunlarını çözmedeki etkisini sistematik bir şekilde incelemişlerdir. Analizlerinde PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) kontrol listesini ve Cochrane kalite değerlendirme (CQA) unsurlarını kullanmış ve çalışmaların çoğunlukla güvenlik konusuna odaklandığını tespit etmişlerdir. Homomorfik şifreleme genellikle bir çözüm olarak önerilse de, bazı çalışmalar başka konulara da dikkat çekmektedir. Analiz, makalelerin %38'inin yöntemlerin tanımı, araştırma amacı ve finansman konularında belirsizlik içerdiğini ortaya koymuştur. Bu inceleme, homomorfik şifreleme algoritmalarının sözlüksel analizini, kullanım alanlarını ve gelişim yönlerini ele almaktadır; çünkü bu algoritmalar, verilerin güvenliğini sağlayabilir, gizliliğini koruyabilir ve bulut bilişim ile büyük verilerde CIA üçlüsünü (Gizlilik, Bütünlük, Erişilebilirlik) sürdürebilir.

Song ve ark. 2018), FHE tabanlı bir gizlilik korumalı IoT mimarisi sunmaktadır. Önerilen yapı, verilerin gizliliğini korurken aynı zamanda IoT cihazlarındaki

hesaplama yükünü de azaltabilmektedir. IoT'nin yaygınlaşması, çok sayıda cihazın internete bağlanmasını ve bunların birbirleriyle etkileşimde bulunmasını sağlayarak birçok fırsat yaratmıştır (Nižetić ve ark, 2020). Bu sinerji, bulut platformlarıyla birleştiğinde büyük veri depolama, güçlü hesaplama sistemleri ve çift yönlü iletişimi destekler. Son araştırmalara göre, Ağ Kodlama (Network Coding – NC) kablosuz ağlara kıyasla daha yüksek dayanıklılık ve veri aktarım kapasitesi sunarken, homomorfik şifreleme bulut üzerinde güvenli hesaplamalar yapılmasını ve veri gizliliğinin korunmasını sağlar (Peralta ve ark, 2019). Bu, özellikle iletişim ve hesaplama maliyetleri açısından en verimli olan, ECC tabanlı homomorfik şifreleme şemalarına odaklanmaktadır. Benzer parametreler kullanılarak yapılan çeşitli teknik analizler sonucunda, ECC tabanlı homomorfik şifrelemenin 0,25 saniye gibi minimum hesaplama süresiyle etkili bir çözüm olduğu tespit edilmiştir (A. Chaudhari ve ark, 2021; Kumari ve ark, 2022; Prabu Kanna ve ark, 2019; Sathish Kumar ve ark, 2023).

Homomorfik şifreleme ve Scale-Invariant Feature Transform (SIFT) yöntemlerini şifreli görüntü verisi aramasında ilk kullanan çalışma Hsu ve ark. (2011) tarafından yapılmıştır. Bu yaklaşım, bazı işlem yükü ve gizlilik sorunları ile karşılaşmış; bu sorunlar Chao-Yung Hsu ve ark. (2012) tarafından çoklu bulut modeli içinde SIFT yöntemini tutarak aşılmış ve kullanıcı gizliliği korunmuştur. S. Hu ve ark. (2016), Paillier kriptosistemine dayalı Privacy-Preserving Scale-Invariant Feature Transform (PPSIFT) yöntemini sunmaktadır. Bilgisayar biliminde gizliliği koruyan görüntü işleme tasarım zorlukları ve uygulama sorunları Qin ve ark. (2024) tarafından tartışılmıştır. Linear Binary Pattern (LBP) tabanlı gizliliği koruyan yöntemler Sultana ve ark. (2017) tarafından ele alınmış, bu yöntem şifreleme ve matris dönüştürmeyi kapsamaktadır. Bu kapsamda, KNN ve LSH kullanılarak hesaplama açısından yoğun olan, gizliliğe duyarlı içerik tabanlı görüntü arama (EPCBIR) yapılabilmektedir (Xia ve ark, 2017). Meraoumia ve ark. (2021), güvenli biyometrik kriptosistemlerin verimliliğine örnek sunmuş; bu sistemler, bireyleri el palmiyeleri biyometrisi ve kaotik sistemler kullanarak nesnelerle ilişkilendirmektedir.

(T. Yang ve ark, 2018), görüntü yeniden oluşturma sürecinde gizliliği güvence altına aldıklarını belirtir ve SHE tabanlı, Hahn Moment yöntemine dayalı bir yaklaşım sunar. Ayrıca, bulut tabanlı ortamlarda Dünya Taşıyıcı Mesafesi (Earth Mover-Distance) ve indeks oluşturma temelli özellik benzerliği aramaları üzerinde çalışırlar (Y. Li ve ark, 2022). (Y. Wang ve ark, 2019), kullanıcı gizliliğini koruyan bir yöntem önerir; bu

yöntemde görüntü verileri dış kaynaklara aktarılır ve K-means algoritması ile indeksler oluşturulur. Şifrelenmiş görüntü tabanlı güvenli arama tekniği tanıtılmıştır; bulut servis sağlayıcı (CSP), içerik tabanlı aramaları gerçekleştirmek ve indeksleri oluşturmak için Ayrık Kosinüs Dönüşümü (Discrete Cosine Transform, DCT) kullanır (Xia ve ark, 2019). Yeni bir gizlilik koruyucu görüntü arama aracı (PIC), (L. Zhang ve ark, 2017) tarafından önerilmiştir; ayrıca (Du ve ark, 2020) tarafından bulut üzerinde 4-Boyutlu hiperkaotik sistem ile, gizlilik koruyucu içerik tabanlı görüntü arama (CBIR) kullanılarak büyük veri kümelerinin aranması sağlanmıştır. PIC, veri sahiplerinin etkili erişim kontrollerini kurmasıyla, yöneticilerin şifrelenmiş görüntü koleksiyonlarını arayabilmesini kolaylaştıracaktır. Mobil bulut bağlamında şifrelenmiş görüntü arama (Zou ve ark, 2017) tarafından ele alınmıştır. (Abdulsada ve ark, 2019a), CNN ve PHE tabanlı, şifrelenmiş tıbbi veriler üzerinde gizlilik koruyucu arama önerisi sunmuştur. (Duan ve ark, 2022) çalışmasında ise, k-NN ile görüntüler arasındaki benzerliği değerlendirip Öklidyen mesafe kullanarak özellikleri çıkaran CNN tabanlı bir tıbbi görüntü arama yöntemi önerilmektedir.

Yeni bir çok kullanıcı bulut-bağlantılı görüntü verisi şeması ele alınmıştır; bu şema, şifreleme ve görüntü veri arama için ikili doğrusal eşleme, CNN tabanlı özellik çıkarımı ve vekil yeniden şifreleme (proxy re-encryption) içermektedir. İki aşamalı derin karma gizlilik koruyucu görüntü arama (Two-stage Deep Hashing Privacy-Preserving Image Retrieval, TDHPPIR), en yüksek kalitede görüntü hash kodları üretir ve CNN tabanlı karma şemeleri aracılığıyla görüntülerin hızlı ve güvenli bir şekilde alınmasını sağlayan kullanışlı bir indeksleme yöntemi sunar (C. Zhang ve ark, 2020). Geometrik alan aramalarıyla ilgili güvenlik sorunlarını aşmak için, (Miao ve ark, 2024), Mobil Nesnelerin İnterneti (Internet of Vehicles, IoV) için etkili bir gizlilik koruyucu mekânsal-konum sorgu altyapısı tanıtmaktadır. Yaklaşımları, çift-bulut yapısı kullanarak veri erişim desenini korur ve Paillier kriptosistemi ile polinom uyumlama kullanarak rastgele geometrik sorgulara izin verir. Çözüm, geleneksel tek sunuculu homomorfik şifreleme tekniklerine göre %90'dan fazla daha düşük maliyetlidir. Bu çalışma, IoV uygulamaları kapsamında güvenli coğrafi veri işleme için önemlidir. (Miao ve ark, 2024)'ün konum tabanlı gizlilik üzerine odaklanmasının aksine, önerilen çalışma, gizliliğin kapsamını otonom araçlardaki görüntü tabanlı arama algoritmalarına genişletmektedir. Bulut tabanlı otonom araç sistemleri, şifrelenmiş görüntü verileri üzerinde hesaplamalar yapabilme yeteneği sayesinde

verinin uçtan uca gizliliğini sağlamak için FHE kullanmaktadır. Mevcut araştırma önerileri, ham görüntü verilerinin açığa çıkmadığını ortaya koymakta ve bu nedenle, özellikle gerçek zamanlı otonom sürüş ile ilgili durumlarda uygulanabilirliği yüksek olmaktadır. Liang ve ark, IoT uygulamalarının bulut destekli kullanımlarına dayalı olarak gizlilik odaklı kümeleme tabanlı bir görüntü arama sistemi (Privacy-sensitive Clustering-Based Image Retrieval, PPCBIR) sunmaktadır (Liang ve ark, 2025). Önerdikleri çözüm, veri getirme verimliliğini artırmak için kümeleme tabanlı hiyerarşik bir indeksleme şeması, benzerliği güvenli şekilde hesaplamak için geliştirilmiş bir k-NN algoritması ve özellik çıkarımı için CNN'ler kullanmaktadır. Ayrıca, güvenli arama desenlerini ele almakta, çıkarım saldırılarını engellemekte ve çok kullanıcılı bir ortamda anahtarları korumak için matris yeniden şifreleme yöntemini kullanmaktadırlar.

Buna rağmen, bu yöntem, anlık (on-the-fly) güvenli görüntü işleme gereken uygulamalarda yeterli kabul edilemez; çünkü sadece indeks tabanlı, özellik eşleştirme temelli arama ile sınırlıdır. Mevcut çalışma, bilinmeyen nesne tespiti, benzerlik tabanlı tespit ve şifrelenmiş görüntüler üzerinde doğrudan güvenli görüntü işleme sağlamak amacıyla FHE'yi uygulamıştır; bu da indeks tabanlı aramayı ortadan kaldırmaktadır. Buna rağmen, bu yöntem, anlık (on-the-fly) güvenli görüntü işleme gerektiren uygulamalar için yeterli kabul edilemez; çünkü yalnızca indeks tabanlı ve özellik eşleştirmeye dayalı arama ile sınırlıdır ve esnek bir analiz imkânı sunmamaktadır. Bu durum, özellikle gerçek zamanlı karar verme gerektiren sistemlerde önemli bir kısıt oluşturmaktadır. Mevcut çalışma ise, bilinmeyen nesne tespiti ve benzerlik tabanlı algılama gibi daha karmaşık senaryoları ele alarak, şifrelenmiş görüntüler üzerinde doğrudan güvenli görüntü işleme gerçekleştirmeyi hedeflemiştir. Bu amaçla FHE kullanılarak, veriler çözülmeden analiz edilmiştir. Bu sayede, veriler çözülmeden analiz edilebilmekte ve kullanıcı mahremiyeti en üst düzeyde korunmaktadır. Ayrıca önerilen yaklaşım, gerçek dünya senaryolarında karşılaşılan dinamik ve öngörülemeyen nesnelerin güvenli şekilde işlenmesine olanak tanıyarak sistemin esnekliğini ve uygulanabilirliğini artırmaktadır. Böylece, indeks tabanlı arama yaklaşımına olan bağımlılık ortadan kaldırılmış ve gizlilik korunarak daha genel ve güçlü bir nesne tespit çerçevesi sunulmuştur Tam homomorfik şifreleme kullanarak güvenli bulut tabanlı görüntü veri işleme yaklaşımı, yakın tarihli çalışmamızda sunulmuştur (Saeed ve ark, 2025b). Bu çalışmada, BFV şeması ve YOLOv9c

kullanılarak şifrelenmiş görüntülerin alınması ve nesne tespiti gösterilmektedir. Bir diğer önceki çalışmamızda, BFV homomorfik şifreleme kullanılarak şifrelenmiş görüntülerden nesne tabanlı güvenli görüntü arama yöntemi önerilmiştir (Saeed ve ark, 2025a). Bu teknik, güvenli desen eşleştirme ve gizlilik koruyucu görüntü tespiti sağlar; böylece, yakın tarihli çalışmamızda (Saeed ve ark, 2025b) sunulan bulut tabanlı FHE çerçevesini tamamlar. Tablo 2.2, mevcut görüntü tabanlı arama yöntemlerinin karşılaştırmasını göstermektedir.

Tablo 2.2. Farklı araştırma çalışmalarındaki tekniklerin karşılaştırması

Araştırma Makalelerinde Kullanılan Teknik	İndeks Tabanlı	Arama Deseni Güvenliği	Bilinmeyen Nesne Tabanlı Tespit	Yüzdelik ve Benzerlik Tabanlı Tespit	Homomorfik Şifreleme	Gizlilik Koruma
CNN & PHE (Abdulsada ve ark, 2019a)		✓		✓	✓	✓
CNN based Hash (C. Zhang ve ark, 2020)	✓					✓
DCT (Xia ve ark, 2019)	✓					
SIFT (Chao-Yung Hsu ve ark, 2012)	✓					
SIFT (S. Hu ve ark, 2016)	✓				✓	✓
Dual-Cloud Spatial Query Framework (Miao ve ark, 2024)					✓	✓

Tablo 2.2. (Devam) Farklı araştırma çalışmalarındaki tekniklerin karşılaştırması

Araştırma Makalelerinde Kullanılan Teknik	İndeks Tabanlı	Arama Deseni Güvenliği	Bilinmeyen Nesne Tabanlı Tespit	Yüzdelik ve Benzerlik Tabanlı Tespit	Homomorfik Şifreleme	Gizlilik Koruma
Hahn Moments & PHE (T. Yang ve ark, 2018)					✓	✓
K means Index Generation (Y. Wang ve ark, 2019)	✓					✓
KNN means & MHE (L. Zhang ve ark, 2017)	✓				✓	✓
Linear Binary Pattern (Sultana ve ark, 2017)	✓					
LSH & KNN (Xia ve ark, 2017)	✓					
LSH & SIFT (Zou ve ark, 2017)	✓					✓
MSC & SIRS-IR (Du ve ark, 2020)						✓
PHE & Euclidean distance (Duan ve ark, 2022)					✓	✓

Tablo 2.2. (Devam) Farklı araştırma çalışmalarındaki tekniklerin karşılaştırması

Araştırma Makalelerinde Kullanılan Teknik	İndeks Tabanlı	Arama Deseni Güvenliği	Bilinmeyen Nesne Tabanlı Tespit	Yüzdelik ve Benzerlik Tabanlı Tespit	Homomorfik Şifreleme	Gizlilik Koruma
PPCBIR (Liang ve ark, 2025)	✓	✓			✓	✓
RSA & PPSIFT (Qin ve ark, 2024)					✓	✓
Earth Mover Distance Metric (Y. Li ve ark, 2022)	✓			✓		
Önerilen Yöntem FHE (BFV)(Saeed ve ark, 2025a, 2025b)		✓	✓	✓	✓	✓

3. KRİPTOGRAFİ VE YAPAY ZEKA TEMELLERİ

Günümüz bilişim sistemlerinin temel taşlarından bazıları, verilerin korunması ve bu verilerden anlamlı bilgiler çıkarabilme yeteneğidir. Kriptografik çözümler, gizli bilgiyi önceden belirlenmiş kişilerden saklamayı mümkün kılar; yapay zeka ise büyük ve karmaşık veri kümelerini analiz edip çözümlemeyi sağlar. Bu iki alan bir araya geldiğinde, güvenli ve akıllı sistemler ortaya çıkar: Bu, özellikle bulut tabanlı sistemler ve otonom araç platformları için sürekli bir ihtiyaçtır; zira bu sistemler tarafından oluşturulan ve işlenen hassas veriler giderek artmaktadır. Otonom araçların ortaya çıkışıyla birlikte, ulaşım artık bir veri etkinliği haline gelmiştir. Her araç, etrafındaki dünyayı 24 saat gözlemleyen kameralar, LiDAR, radar ve GPS sensörleri taşır. Bu sensörler, durmaksızın görsel ve uzamsal-veri akışı üretir ve bu veriler olmadan araç, yol tespiti yapma, çevredeki canlı ve cansız nesneleri tanıma, karar verme gibi gerekli işlemleri gerçekleştiremez. Araç üzerinde sınırlı işlem kaynakları bulunurken, bulutlar bu bilgilerin depolanması ve analizinde önemli bir rol oynar. Ancak, yolcuların gizliliği ve güvenliği için veriler araç dışına çıktığında da korunmalıdır.

Geleneksel kriptografik mekanizmalar olan şifreleme, simetrik ve asimetrik şifreleme, ilk savunma hattını oluşturur. Simetrik şifreleme, asimetrik şifrelemeden daha hızlıdır ve gerçek zamanlı iletişimde daha pratiktir; ancak tek bir anahtara bağımlıdır. Asimetrik şifreleme çoklu anahtar kullanımı sağlar, fakat anahtar yönetimi gerekir. Mesaj bütünlüğü, hash fonksiyonlarıyla sağlanır ve manipülasyon tespiti mümkün olur. Tüm bu avantajlara rağmen temel bir zayıflık vardır: verilerin işlenebilmesi için önce çözülmesi gerekir. Veriler çözüldüğünde, güvenlik riski ortaya çıkar ve saldırılara açık hale gelir. Bu durum, bulut üzerinde şifrelenen görüntü akışları, konum bilgileri veya diğer hassas veriler için ciddi bir tehlike oluşturur. Homomorfik şifreleme, şifreli veriler üzerinde doğrudan hesaplama yapılabilmesini sağlayan bir tekniktir ve bu, en ilgi çekici çözümlerden biri olarak öne çıkmaktadır. Bulutta veriler şifrelendiğinde, orijinal verilere artık erişilemez; yani veriler asla ham hâlde bulutta saklanmaz, yalnızca şifreli metin olarak tutulur. Hesaplama sonucunda elde edilen

veriler tekrar şifrelenerek araca gönderilir ve yalnızca araç içinde çözülür. Böylece, görüntü analizi veya nesne tespiti gibi görevler, ham verileri bulut sistemlerine açmak zorunda kalmadan gerçekleştirilebilir. Homomorfik şifreleme, bu nedenle gizlilik bilincinin korunması konusunda büyük bir ilerleme sağlar. İnteraktif görsel-işitsel hizmetler için kriterler belirlenirken, yapay zekanın burada çok önemli bir rolü vardır (Arrojo, 2025). Otonom araçlar, yayaları, diğer araçları, yol işaretlerini ve öngörülemeyen diğer tehlikeleri tespit etmek için bu tür modelleri kullanır. En hızlı ve etkili nesne tespitini sağlayan en iyi modeller, YOLO ailesi olarak bilinir (Sirisha ve ark, 2023). İşleme aşamalarının ayrılması sayesinde, ham sensör verilerinin gizliliği korunabilir. Yapay zeka modelleri, ilk tespiti araç üzerinde gerçekleştirebilir ve ortaya çıkan veriler buluta gönderilmeden önce şifrelenebilir; bu sayede güvenli bir analiz ve doğrulama sağlanmış olur.

İki kavramın ilişkisini mantıksal argümanlar ve örneklerle açıklamak gerekirse, yapay zeka ve homomorfik şifreleme birbirini tamamlayan fakat ayrı işlevlere sahip sistemlerdir. Mantıksal olarak birleşim noktaları, veri gizliliğini korurken YZ modellerinin nesne tespiti gibi görevleri yerine getirmesidir. Örneğin, YZ modelinin araç üzerinde ilk tespiti yapması ve ardından verilerin şifrelenerek bulutta işlenmesi, her iki teknolojinin birlikte çalışmasının mantıksal bir örneğidir. Varoluşsal açıdan ise, YZ ve homomorfik şifreleme ayrı sistemler olarak da çalışabilir; biri veriyi analiz ederken diğeri veriyi korur, ancak ortak bir amaç doğrultusunda entegrasyon sağlandığında güvenli ve etkili bir sistem elde edilir. Homomorfik şifreleme, işlemci kaynakları üzerinde ciddi bir yük oluşturur; oysa derin öğrenme modelleri, araçların zamanında kendilerini kontrol edebilmesi için yüksek işlem hızı gerektirir. Bu bağlamda, veri gizliliği ve model verimliliği arasında bir denge bulmak, bu çalışmanın temel hedeflerinden biri olmuştur. Bu dengeyi sağlamak için kullanılan bazı teknikler şunlardır: şifreli verinin optimize edilmiş temsili, daha saf bootstrapping teknikleri ve paralel sinir ağları kullanımı. Bu yaklaşımlar, yüksek güvenlik gereksinimleri ile hızlı karar alma süreci arasındaki farkı giderek azaltmaktadır.

Bu bölümde, önerilen sistem için teorik altyapı açıklanmaktadır. Öncelikle, kriptografinin sistematik bir incelemesi ve geleneksel ile daha olgun post-kuantum yaklaşımlarına ilişkin çeşitli kriptografi disiplinlerinin genel bir tanımı sunulmaktadır. Ardından, homomorfik şifreleme ve şifreli veriler üzerinde hesaplama yapmanın teorik ve matematiksel temelleri ele alınmaktadır. Daha sonra, günümüzdeki nesne

tanıma süreçlerini oluşturan CNN ve YOLO modelleri açısından yapay zekanın temelleri açıklanmaktadır. Son olarak, kriptografi ve yapay zekanın bulut üzerinde nasıl entegre edilebileceği ve bu sayede güvenli ve akıllı çözümler aracılığıyla bilgisayar destekli sürüşün geliştirilmesi, uygulanması, yönetimi ve yaşam döngüsünün nasıl güvence altına alınabileceği anlatılmaktadır.

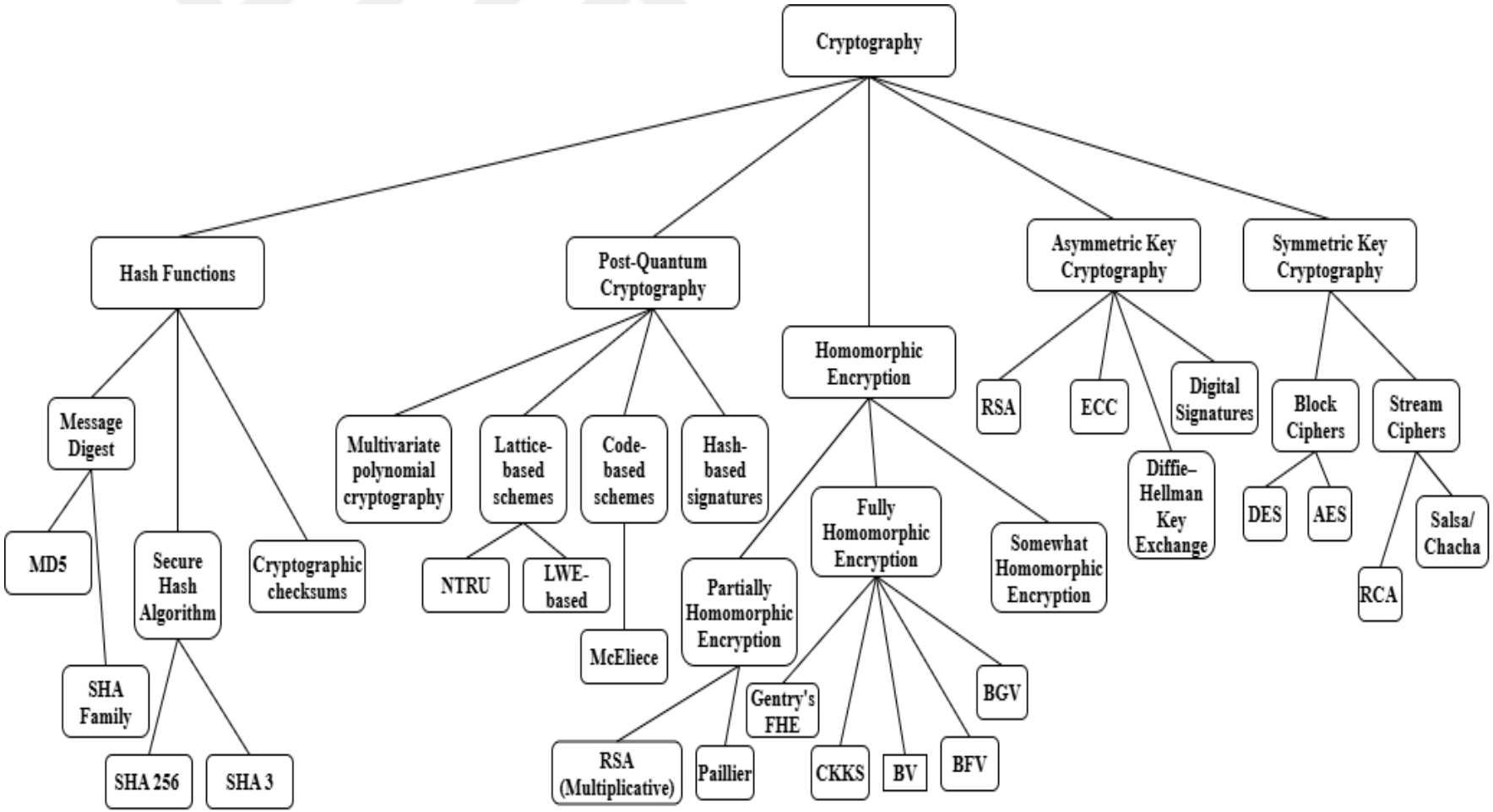
3.1. Kriptografinin Hiyerarşisi

Kriptografi, farklı veri güvenliği ihtiyaçlarına cevap veren birçok alt dalı olan karmaşık bir alan haline gelmiştir. Geleneksel olarak mesaj gizliliğine odaklanan kriptografi, dijital sistemler, iletişim ağları ve bulut hizmetlerinin hızlı gelişimiyle birlikte çok daha ileri bir seviyeye taşınmıştır. Günümüzde kriptografi, yalnızca gizlilik ve bütünlüğün temeli olmakla kalmayıp, aynı zamanda güvenli hesaplama, dijital imzalar ve gelecekteki kuantum saldırılarına karşı dayanıklılık gibi sofistike özellikleri de kapsamaktadır. Bu alanın kapsamını daha iyi anlamak için kriptografik araçları piramidal bir sistem içerisinde yapılandırmak uygun olur. Şekil 3.1, kriptografinin hiyerarşisini göstermektedir; bu hiyerarşinin her seviyesi, gelişmiş gizlilik odaklı işlemler yoluyla mesaj gizliliği de dahil olmak üzere veri korumanın belirli bir yönüne odaklanır. Takip eden bölümde, bu hiyerarşinin her bir dalı ayrıntılı olarak açıklanacaktır.

3.1.1. Simetrik anahtarlı kriptografi

1970'li yıllar, asimetrik anahtar kriptografisinin icat edildiği dönemdir; bu dönemde kullanılan temel şifreleme yöntemi simetrik anahtarlı kriptografiydi. Bazen buna geleneksel veya tek anahtarlı şifreleme de denir. Simetrik anahtarlı algoritmalarda, şifreleme ve çözme işlemleri aynı tek anahtar ile gerçekleştirilir. Sadece tek bir anahtarın yeterli olması, bu kriptografiyi veri korumada hayati bir yöntem hâline getirir.Simetrik anahtarlı kriptografide temel teknikler, yerine koyma (substitution) ve yer değiştirme (transposition) yöntemleridir.

- **Yerine koyma teknikleri**, düz metindeki harf veya karakterleri, gizli anahtardan türetilen kurallara göre yenileriyle değiştirir.
- **Yer değiştirme teknikleri**, düz metindeki öğelerin sırasını değiştirir ancak öğelerin kendilerini değiştirmez (Parekh ve ark, 2023).



Şekil 3.1. Kriptografinin hiyerarşisini

Simetrik anahtarlı kriptografi iki türde incelenir:

3.1.1.1. Akış şifreleri (Stream Ciphers)

Akış şifrelerinde şifreleme süreci, algoritmanın şifreleme anahtarı ve nonce (benzersiz rastgele üretilmiş bir sayı) ile sahte rastgele bir anahtar akışı (keystream) oluşturmasıyla başlar. Sonuç olarak, düz metin uzunluğuna eşit rastgele bir bit dizisi elde edilir. Düz metin de bireysel bitlere ayrılır ve XOR bit düzeyinde işlemleri kullanılarak, düz metindeki her bir bit, anahtar akışının bitleriyle sırayla birleştirilir. Böylece düz metin kademeli olarak şifreli metne dönüştürülür. Alıcı taraf, şifreli metni çözmek için şifreleme sırasında aynı şekilde yeni bir anahtar akışı üretmek zorundadır ve her karakter tek tek çözülür. En yaygın akış şifre algoritmaları şunlardır:

- **Rivest Cipher 4 (RC4):** RC4, etkin yapısı ve değişken uzunluktaki veri akışlarını yönetebilme yeteneğiyle dikkat çekmiştir. Ancak, ortaya çıkan güvenlik açıkları nedeniyle çoğu uygulamada artık güvenli kabul edilmemektedir. Kriptografik standart kuruluşları, kullanımına karşı uyarılarda bulunmaktadır (Biswas ve ark, 2024).
- **Salsa20:** Basit ama sofistike bir tasarıma sahiptir ve hızlı ve etkili çalışır. Tanınmış tehditlere karşı güçlü koruma sağlar. Ayrıca, çeşitli kriptografik algoritmalarda yapı taşı olarak kullanılabilir. Salsa20, güvenlik ve performansın dengelendiği birçok uygulamada yaygın olarak tercih edilen bir akış şifresidir (Biswas ve ark, 2024).

3.1.1.2. Blok şifreleri (Block ciphers)

Blok şifreleri, şifreleme sürecinde bir dizi blok üretir ve bu bloklar daha sonra anahtar ile şifrelenir. Önceden belirlenmiş bir düzen içinde şifrelenmiş veri blokları çıktıyı oluşturur. Şifreli metin alıcıya ulaştığında, alıcı aynı kriptografik anahtarı kullanarak blokları çözerek düz metin hâline getirir.

En yaygın blok şifre algoritmaları şunlardır:

- **Advanced Encryption Standard (AES):** AES, 128, 192 ve 256 bit olmak üzere üç farklı anahtar boyutunu destekler; çoğunlukla 128 bit anahtar kullanılır. AES uygulamaları arasında, multimedya telif haklarının korunması için dijital hak yönetimi, depolama cihazlarında veri şifreleme ve güvenli iletişim ile uygulama yönetimi yaygın olarak kullanılmaktadır (Buell, 2021).

- **Data Encryption Standard (DES):** DES, 64 bitlik düz metin bloklarını şifrelemek için 56 bitlik bir anahtar kullanır. Kısa anahtar boyutunun yarattığı güvenlik zafiyetini aşmak için, daha güvenli bir algoritma olan AES geliştirilmiştir (Buell, 2021).

3.1.2. Asimetrik anahtarlı kriptografi (Asymmetric key cryptography)

Asimetrik anahtarlı kriptografi, şifreleme ve şifre çözme işlemleri için iki anahtarın kullanıldığı kriptografi türüdür: herkese açık bir anahtar (public key) ve özel bir anahtar (private key). Şifreleme işlemi, genellikle herkesin erişebileceği halka açık anahtar ile yapılır. Ancak bu veriyi çözebilecek tek kişi, eşleşen özel anahtara sahip olan alıcıdır.

Bu nedenle, asimetrik anahtarlı kriptografi bazen açık anahtarlı kriptografi (public-key cryptography) olarak da adlandırılır.Bu kriptografi türünün iki temel sorunu vardır: dijital imzalar ve anahtar dağıtımı. Asimetrik algoritmalar, veriyi bir anahtarla şifreler ve başka bir anahtar ile çözer (Parekh ve ark, 2023).

Asimetrik algoritmaların temel özellikleri şunlardır:

- Şifreleme anahtarı ve algoritma tek başına, şifre çözme anahtarını belirlemek için kullanılamaz.
- Bir anahtar şifre çözme için kullanılırken, diğer anahtar şifreleme için kullanılabilir.

Simetrik anahtarlı kriptografi, şifreleme ve şifre çözme için sembol değiştirme (substitution) ve yer değiştirme (permutation) yöntemlerini kullanırken, asimetrik kriptografi düz metni (plaintext) ve şifreli metni (ciphertext) matematiksel fonksiyonlar aracılığıyla işler ve bunları şifreleme ve çözme için tamsayılar (integer) şeklinde temsil eder. Asimetrik kriptografi, şifreleme ve çözme işlemleri için kodlama ve dekodlama süreçlerini gerektirir, çünkü düz metin ve şifreli metin tamsayılar olarak işlenir. En yaygın asimetrik kriptografi algoritmaları şunlardır:

3.1.2.1. Rivest–Shamir–Adleman (RSA)

RSA, sıklıkla dijital imzalar oluşturmak ve güvenli iletişimi sağlamak için kullanılır. Büyük tamsayı asal sayılar kullanarak anahtarlar üretir. Veriyi şifrelemek için herkese açık anahtar, şifre çözmek için ise özel anahtar gereklidir. Güçlü bir güvenlik sağlasa da, bazı diğer algoritmalara göre daha yavaştır (Savadatti ve ark, 2025).

3.1.2.2. Eliptik eğri kriptografisi (Elliptic curve cryptography – ECC)

ECC'de daha kısa anahtar boyutları, RSA ile aynı güvenlik seviyesini sağlar. Bu yaklaşım, eliptik eğrilerin matematiksel özelliklerine dayanır. Kaynakları sınırlı cihazlar için daha hızlı ve etkilidir. IoT ve mobil güvenlik alanlarında giderek yaygınlaşmaktadır (Savadatti ve ark, 2025).

3.1.2.3. Diffie-Hellman

Diffie-Hellman doğrudan veri şifrelemek yerine, güvenli iletişim için paylaşılan gizli bir anahtar oluşturur. Bu anahtar iki kişi arasında hiç değiştirilmeden oluşturulur. Anahtar değişimi için sıklıkla RSA gibi diğer tekniklerle birlikte kullanılır (van Oorschot, 2022).

3.1.2.4. Dijital imza standardı (Digital signature standard – DSS)

DSS, ElGamal şifrelemesine dayalı bir varyant kullanır. Dijital imzalar için tasarlanmıştır ve mesajların bütünlüğünü ve geçerliliğini sağlar. Gönderen, iletişimi kendi özel anahtarıyla imzalarken, alıcı iletişimi gönderenin herkese açık anahtarı ile doğrular. Yaygın uygulamalar arasında kaynak kodu veya yazılım imzalama (yazılım güvenliği için) ve e-posta imzalama (e-posta güvenliği için) bulunur (van Oorschot, 2022).

3.1.3. Karma fonksiyonları (Hash functions)

Hash fonksiyonlarına dayalı kriptografi algoritmalarında, sabit bir hash değeri girdi alır ve karakter dizisi üretir. Bu hash fonksiyonları hızlı ve tek yönlü olacak şekilde tasarlanmıştır; girdi üzerinde yapılan çok küçük bir değişiklik bile tamamen farklı bir hash değeri üretir. Bu özellikler, dijital verilerin kimlik doğrulamasını (orijin veya geçerlilik doğrulaması) kolaylaştırdığı için bilgi güvenliğinin temelini oluşturur (Boyanov, 2024).

- Kriptografik hash fonksiyonları, sistemlerin herhangi bir dosya veya mesaj için benzersiz bir değer veya "parmak izi" oluşturmasını sağlar; böylece verilere yapılan istenmeyen değişiklikler hemen tespit edilebilir ve veri bütünlüğü korunur.
- Dijital imzalar ve sertifika doğrulamasının temelini oluştururlar. İletişim ve yazılımın güvenilirliğini doğrulamanın güvenli bir yolunu sağlarlar.
- Şifreler depolanmadan önce hashlenerek güvenli bir şekilde saklanır; böylece veriler sızdırılsa bile kullanıcı kimlik bilgileri doğrudan açığa çıkmaz.

- Dağıtık defter teknolojilerinde (ör. blockchain) şeffaflık sağlamak için yaygın olarak kullanılır; hash fonksiyonları blokların hash değerlerini birbirine bağlayarak kırılmaz ve değiştirilemez bir defter oluşturur.

3.1.3.1. Mesaj özeti algoritması 5 (MD5)

MD5, en yaygın kullanılan hash algoritmalarından biridir. Eskiden dijital imzalar ve veri bütünlüğü için yaygın olarak kullanılmıştır, ancak artık güvensiz kabul edilir; çünkü saldırganların hash çakışmaları oluşturmasını kolaylaştıran zayıflıklar bulunmuştur. Hızı ve kullanım kolaylığı nedeniyle bir dönem popüler olsa da, güvenlik açısından kritik uygulamalarda artık önerilmez (Boyanov, 2024).

3.1.3.2. Güvenli hash algoritması 1 (SHA-1)

SHA-1, daha büyük hash boyutu ve daha yüksek çakışma direnci ile MD5'i geride bırakmıştır. Ancak hesaplama gücü ve kriptoanaliz ilerledikçe zayıflıkları ortaya çıkmış ve çakışma saldırıları mümkün olmuştur. Bu nedenle dijital imzalar ve SSL/TLS sertifikaları gibi birçok güvenlik uygulaması artık SHA-1'i kullanmamaktadır (Boyanov, 2024).

3.1.3.3. SHA-2 Ailesi (SHA-256, SHA-512)

Kriptografik hashleme için endüstri standardı artık SHA-2 ailesidir ve güçlü güvenlik sağlar. SHA-2, 256 ve 512 bit uzunluğunda hash çıktısı üretir. Bu algoritmalar, şifrelerin hashlenmesi, blockchain teknolojileri ve güvenli iletişim protokolleri için yüksek çakışma ve ön-imal direnci nedeniyle tercih edilmektedir (Boyanov, 2024).

3.1.3.4. SHA-3 (Keccak)

SHA-3, NIST'in en son standardıdır ve özel bir sünger (sponge) mimarisi kullanarak SHA-2'ye göre güvenlik ve esneklik sunar. Uzun vadeli güvenlik gerektiren uygulamalar için uygundur ve belirli saldırılara karşı geliştirilmiş koruma sağlar (A. C. H. Chen, 2024).

3.1.3.5. BLAKE2 ve BLAKE3

BLAKE2 ve BLAKE3, yüksek hızlı ve güvenli alternatiflerdir. SHA-2 ve SHA-3'ün güvenliğini sunarken, hashleme işleminin hızından ödün vermezler. BLAKE3, artımlı hashleme ve paralel işlem yeteneği sayesinde günümüz sistemleri için hem hızlı hem de güçlü kriptografik güvenlik sağlar (A. C. H. Chen, 2024).

3.1.4. Post-kuantum kriptografi (PQC)

Post-kuantum kriptografi veya kuantuma dayanıklı kriptografi, kuantum bilgisayar saldırılarına karşı veri güvenliğini sağlar. Kuantum bilgisayarlar**,** kuantum fiziği prensiplerini kullanarak hesaplamaları çok hızlı bir şekilde gerçekleştirebilir. Kuantum mekaniği, maddenin atom altı ve çok küçük ölçeklerdeki doğasını inceler. Henüz başlangıç aşamasında olsalar da, kuantum bilgisayarlar bazı problemleri geleneksel bilgisayarlardan daha hızlı çözebilir. Bu problemlerin en bilineni tam sayı çarpanlara ayırma**d**ır. Post-kuantum kriptografi, özel bilgilerin kuantum bilgisayar saldırılarına karşı korunması açısından kritik öneme sahiptir (Dam ve ark, 2023). Tipik PQC teknikleri şunlardır:

3.1.4.1. Lattice tabanlı kriptografi

Lattice (ızgara) noktalarından oluşan geometrik yapılardır. Lattice tabanlı kriptografi, yüksek boyutlu lattice yapıları üzerinde işlem yapmayı gerektirir; bu da kuantum bilgisayarlar için çözülmesi zor bir problemdir. Lattice tabanlı kriptografi, NTRU olarak bilinen açık anahtar kriptosisteminin temelini oluşturur.

NTRU, diskret logaritma ve çarpanlara ayırma gibi sayı teorisine dayalı problemler yerine lattice problemlerinin zorluğuna dayanır. Kuantum bilgisayar saldırılarına karşı dayanıklıdır ve RSA ile ECC'den farklıdır. LWE (Learning With Errors**)** tabanlı şifreleme, gürültülü lineer denklemleri çözmenin zorluğuna dayanır; küçük rastgele hatalar eklenerek, gizli anahtar olmadan çözülmesi imkânsız hale gelir (Nisha ve ark, 2024).

3.1.4.2. Kod tabanlı kriptografi

Kod tabanlı kriptografi, hata düzeltme kodlarının matematiksel özelliklerine dayanır. Hata düzeltme kodları, veri iletimindeki hataları tespit edip düzeltir. Kod tabanlı kriptografi yöntemleri, kuantum bilgisayarlar için anlaşılması zor olacak şekilde tasarlanmıştır. McEliece kriptosistemi, kod teorisine dayalı bir açık anahtar şifreleme tekniğidir. 1978'de Robert McEliece tarafından önerilmiş olup, genel lineer bir kodun çözümleme probleminin NP-zor olmasına dayanır (Weger ve ark, 2024).

3.1.4.3. Multivariant kriptografi

Bu tür kriptografinin temeli, çok değişkenli polinom denklem sistemlerinin çözümüne dayanır (Dey ve ark, 2023).

3.1.4.4. Hash tabanlı kriptografi

Hash tabanlı kriptografi, hash fonksiyonlarının matematiksel özelliklerine dayanır. Bu algoritmalar, hash fonksiyonlarının çakışmaya karşı dirençli olmasına dayandığı için kuantum bilgisayarların çözmesini zorlaştırır. Aynı hash değerini üreten iki girdi bulmak neredeyse imkânsızdır (Fathalla ve ark, 2024).

3.1.5. Homomorfik kriptografi

Homomorfik şifreleme**,** şifreli veriler üzerinde doğrudan hesaplama yapmayı mümkün kılan bir kriptografi tekniğidir. Standart şifrelemede, anlamlı her işlem için verilerin çözülmesi gerekirken, homomorfik şifreleme sayesinde şifreli veriler üzerinde hesaplamalar yapılabilir. Bu hesaplamalar çözüldüğünde elde edilen sonuç, aynı işlemlerin açık veriler üzerinde yapılmasıyla elde edilen sonuçla aynıdır (Chase ve ark, 2017). Böylece, yüksek derecede hassas veriler hala şifreli olarak incelenebilir, değiştirilebilir ve işlenebilir; bu da gizlilik ve veri güvenliğini sağlar.

3.1.5.1. Homomorfik şifreleme uygulamaları

- **Güvenli Dış Kaynak Kullanımı (Secure Outsourcing):** Homomorfik şifrelemenin en ikna edici uygulamalarından biri, güvenli dış kaynak kullanımıdır. İş senaryolarına yönelik hesaplamalar, kritik veriler açığa çıkmadan güvenilmeyen sunuculara devredilebilir. Örneğin, bir sağlık kuruluşu, bulutta saklanan şifreli hasta kayıtları üzerinde karmaşık analizler yapabilir ve bu sırada herhangi bir hastanın klinik bilgilerini üçüncü taraflara açmaz (Marcolla ve ark, 2022b).
- **Güvenli Analiz (Secure Analysis):** Homomorfik şifreleme, kuruluşların şifreli veri setlerini gizliliği koruyarak analiz etmesine olanak tanır. Bu, veri gizliliğinin kritik önemde olduğu hükümet ajansları veya finans kurumları gibi ortamlar için son derece faydalıdır. Gelişmiş algoritmalar ve istatistiksel analizler sayesinde, hassas veriler gizli kalır; altta yatan istatistiksel bilgiler açığa çıkmaz (Marcolla ve ark, 2022b).
- **Makine Öğrenmesi Gizliliği (Machine Learning Confidentiality):** Modellerin etkili bir şekilde eğitilebilmesi için genellikle büyük veri kümelerine erişim gerekir. Homomorfik şifreleme sayesinde, farklı kurumlar ham verileri paylaşmadan modeller geliştirebilir. Bu, özellikle sağlık ve finans gibi veri ve gizlilik hassasiyeti yüksek alanlarda kurumlar arası güvenli iş birliklerini mümkün kılar (Marcolla ve ark, 2022b).

- **Güvenli Veri Paylaşımı (Secure Data Sharing):** Homomorfik şifreleme, güvenli bilgi paylaşımını da mümkün kılar. Kurumlar, aynı şifreli verileri kullanarak hesaplamalar yapabilir ve böylece hassas bilgileri açığa çıkarmadan iş birliği sağlayabilir. Bu, verilerin gizli kalması gerekirken iş birliğine olanak tanıyan durumlar için özellikle önemlidir (Marcolla ve ark, 2022b).

3.1.5.2. Homomorfik şifreleme türleri

Homomorfik şifreleme, çeşitli kullanım senaryoları ve güvenlik gereksinimlerini karşılamak için farklı şekillerde uygulanabilen popüler bir kriptografi yöntemidir. Temel homomorfik şifreleme türleri şunlardır:

- **Kısmi Homomorfik Şifreleme (Partially Homomorphic Encryption – PHE):** Bu şifreleme türü, belirli işlemlerin şifreli veriler üzerinde gerçekleştirilmesine izin verir. Toplam homomorfizmi (Additive Homomorphism): Şifreli değerlerin toplanmasını veya çıkarılmasını sağlar. Çarpımsal homomorfizm (Multiplicative Homomorphism): Şifreli değerlerin çarpılmasına veya bölünmesine olanak tanır. Bu türde yalnızca belirli işlemler yapılabilir (Medileh ve ark, 2023).
- **Kısmen Homomorfik Şifreleme (Somewhat Homomorphic Encryption – SHE):** Bu şifreleme türü, şifreli veriler üzerinde sınırlı sayıda işlem yapılmasına izin verir. SHE şemaları, FHE'ye kıyasla sınırsız işlem yapamaz ve güvenliği riske atmadan çalışamaz. Ancak, hesaplama verimliliği ile güvenlik arasında bir denge sağlar (Subramaniyaswamy ve ark, 2022).
- **Tam Homomorfik Şifreleme (Fully Homomorphic Encryption – FHE):** En güçlü şifreleme türüdür. FHE, şifreli veriler üzerinde toplama ve çarpma işlemlerinin yanı sıra Boolean işlemlerini (AND, OR, NOT) de gerçekleştirmeye olanak tanır. FHE çok güçlüdür ancak yüksek hesaplama maliyetine sahiptir çünkü şifre çözmeden doğrudan şifreli veriler üzerinde işlem yapmayı mümkün kılar (Rauthan, 2022).

3.1.5.3. Örnekler: Toplama ve çarpma işlemleri

Toplama İşlemi

Örnek olarak giriş vektörü şu şekilde olsun: X= [1,3,5,9]

Bu vektör, public key (açık anahtar) ile şifrelenir ve şifreli vektör şu şekilde gösterilir:

$$\mathrm{E}\,(\mathrm{X}\,[1,3,5,9]) = E_x\,[E_1, E_2, E_3, E_4] \qquad (3.1)$$

Skaler toplama örneği: E_x+ 3

$$E_x\ [E_1, E_2, E_3, E_4] = E_x\ [E_1, E_2, E_3, E_4] + 3 \quad (3.2)$$

Bu şifreli vektör, secret key (gizli anahtar) kullanılarak çözüldüğünde sonuç E_x:

X = [4,6,8,12]

Aynı boyutta bir vektör ile toplama: E_x+ [3,4,1,2]

Şifre çözme işleminden sonra sonuç:

X = [4,7,6,11]

Çarpma İşlemi

Skaler çarpma: E_x * 3

$$E_x\ [E_1, E_2, E_3, E_4] = E_x\ [E_1, E_2, E_3, E_4] * 3 \quad (3.3)$$

Gizli anahtar kullanılarak şifre çözme sonrası sonuç:

X = [3,9,15,27]

Aynı boyutta bir vektör ile çarpma: E_x* [3,4,1,2]

Şifre çözme sonrası sonuç:

X = [3,12,5,18]

3.2. Yapay Zekâ Tabanlı Nesne Tespiti ve Güvenlik

YZ, modern güvenlik sistemlerinin ve akıllı ulaşım çözümlerinin ayrılmaz bir parçasıdır. Bu nedenle, veri odaklı olma, desenleri tanıma ve geleneksel algoritmaların ötesinde kararlar alabilme gibi avantajlara sahiptir. YZ, yalnızca uygulamaları etkinleştiren bir araç değil, aynı zamanda robot otomobiller ve bulut destekli pazar işlemleri için bir risk koruma aracıdır. YZ , siber güvenliğin birçok prensibini içerir; anomal davranışların tespiti, izinsiz giriş algılama ve şifrelenmiş veri desenlerinin analizi gibi görevlerde kullanılır. Model tabanlı otomatik koruma, YZ sayesinde adaptiftir; yani geçmiş saldırılardan öğrenir ve her saldırı ile kendini geliştirir, bu da statik savunma biçimlerine kıyasla çok daha üstündür. Bulut tabanlı hizmetlerde ve ağlarda, YZ tehdit tespitini iyileştirmek ve yanlış alarm oranını düşürmek için hızlı tehdit algılama sağlar. Ayrıca YZ, otonom araçların güvenliğini artırmak için iletişim ağını analiz eder ve iletilen mesajların bütünlüğünü doğrular.

3.2.1. Derin öğrenme temelleri

Derin öğrenme, çok katmanlı sinir ağlarını kullanan bir makine öğrenimi dalıdır. Bu farklı katmanlar, ham veriler üzerinde giderek daha karmaşık özellikleri öğrenir. Görüntü analizinde, alt katmanlar kenar ve dokuları, üst katmanlar ise nesneleri ve sahneleri algılar. Derin öğrenme modelleri hesaplama açısından maliyetlidir ve büyük veri gerektirir, ancak doğruluk açısından eşsizdir (O'Mahony ve ark, 2020). Bu özellikleri, gerçek dünyadan gelen görüntü akışlarından hızlı ve güvenilir kararlar alması gereken otonom sürüş için ideal kılar.

3.2.2. Konvolüsyonel sinir ağları (CNN'ler)

CNN'ler, modern görüntü tanımanın temel yöntemidir ve bu uygulama için ana çerçeveyi oluşturur. Konvolüsyonel filtreler, görüntülerden mekânsal özellikler çıkarmak için kullanılır. Pooling katmanları boyutları küçültür, tamamen bağlı katmanlar ise özellikleri birleştirerek tahminlerde bulunur. CNN'ler görsel desenleri tanımada çok başarılıdır ve yüz tanıma, medikal görüntüleme ve otonom navigasyon gibi alanlarda yaygın olarak kullanılır. Araçlar için, CNN'ler yayaları, trafik ışıklarını, yol işaretlerini ve diğer araçları tespit etmekte kullanılmıştır (Issaoui ve ark, 2023; Ross Girshick, 2015). CNN'ler, ham piksellerden doğrudan öğrenmede geleneksel el ile tasarlanmış özellik yöntemlerinden daha başarılıdır.

3.2.3. YOLO ailesi modelleri

Daha önce bahsedilen yöntemler, nesne tespitini iki aşamalı bir problem olarak ele alıyordu. Bu yöntemlerde, ilk aşama görüntüden ROI'leri (Region of Interest – İlgi Alanı) belirlemek, ikinci aşama ise bu ROI'leri bilinen sınıflara ayırmaktı. Ancak, iyileştirmelere rağmen bu süreç, gerçek zamanlı nesne tespiti uygulamaları için bile yavaştır. YOLO, görüntünün farklı parçalarını analiz etmek yerine, tüm sahneyi tek seferde işler. YOLO modeli, derin bir konvolüsyonel sinir ağıdır ve monolitik bir mimari kullanır (Redmon ve ark, 2016). Bu sayede, görüntüdeki tüm nesneler için sınıf olasılıkları ve sınırlayıcı kutuları (bounding box) aynı anda tahmin eder. Nesne tespitini tek bir regresyon problemi olarak ele alarak, çok aşamalı tespit yöntemlerinin karmaşık boru hatlarını (pipeline) ortadan kaldırır ve böylece önemli ölçüde hız kazanır. YOLO modülü, tek bir sinir ağıdır (yalnızca bir konvolüsyonel katmana sahiptir; diğer çok katmanlı ağlara kıyasla çok daha basittir) ve görüntünün her bir bölgesi için hem bounding box hem de nesne sınıflarını aynı anda tahmin eder. Diğer algoritmaların nesneleri sınıflandırmak için ayrı sınıflandırıcılar kullandığına karşın,

YOLO nesne tespitini regresyon problemi olarak ele alır. Model çok hızlı bir şekilde eğitilmiş ve temel model için gerçek zamanlı olarak saniyede 45 FPS hızında görüntü işlenebilmiştir (Redmon ve ark, 2015).

Darknet'te mevcut olan sürümü, Cuda ve C tabanlı düşük seviyeli bir derin öğrenme çerçevesidir ve bu sürüm YOLO olarak adlandırılır. YOLO, deformable parts model ve R-CNN gibi diğer nesne tespit algoritmalarından katbekat daha hızlıdır ve çok daha yüksek ortalama doğruluk (mean Average Precision – mAP) sağlar. Diğer algoritmalara göre daha fazla bounding box hatası yapabilir, ancak var olmayan bir nesneyi tespit etme olasılığı çok düşüktür. Üçüncüsü, YOLO'nun iyi bir genelleme yeteneği vardır; çünkü nesne tespiti için uçtan uca bir hedef fonksiyonuna sahiptir. YOLO'nun eğitimi basittir; çünkü nesne tespitinin farklı aşamaları tek bir sinir ağına entegre edilmiştir. İlk olarak, görüntüdeki tüm özellikler modele beslenir. Model daha sonra tüm görüntüyü S x S boyutunda bir ızgaraya böler; burada S, rastgele seçilebilir. Her ızgara hücresi, merkezinin bu hücrede bulunduğu nesneyi algılar. Her ızgara hücresi, güven skoru (confidence score), bounding box ve sınıf olasılığını belirleyen bir kalite vektörü tahmin eder. Bu vektörler, YOLO modelinin çıktısını oluşturur. Vektörün ölçek faktörü SxS (Bx5 + C) şeklindedir; burada S, ortalama görüntü ölçek faktörü, B bounding box sayısı ve C, nesneye bağlı sınıfların koşullu olasılığıdır. Her bounding box ile ilişkili beş sayı vardır: güven skoru, x koordinatı, y koordinatı, yükseklik ve genişlik. Güven skoru, modelin belirli bir bounding box içinde bir nesne olduğuna olan güvenini gösterir. Güven skoru, IoU (Intersection over Union – tahmin ve gerçek bounding box'ların kesişim oranı) kullanılarak hesaplanır. Eğitim sırasında yalnızca nesne merkezine sahip ızgara hücreleri 1 güven skoruna sahip olur; diğer vektör girişleri 0'dır. Nesne merkezi olmayan hücreler ise 0 güven skoruna ve sıfır vektörüne sahiptir. Örneğin, 10 sınıflı nesnelere sahip bir veri kümesi varsa, C = 10 olur. Her ızgara hücresi için seçilen bounding box sayısı ne olursa olsun, yalnızca bir sınıf olasılıkları seti bulunur (Redmon ve ark, 2015).

3.2.3.1. YOLOv2

Ön eğitim için YOLOv2 (You Only Look Once Versiyon 2), değiştirilmiş bir Darknet-19 modelidir. Model, 19 konvolüsyon katmanı ve aralarına yerleştirilmiş 5 max-pooling katmanından oluşur. Nesne tespiti için ağın sonunda toplam 6 birleştirme (concatenation) yapılmıştır ve böylece YOLOv2, 30 katmanlı bir mimariye dönüştürülmüştür. YOLOv2, daha önce Faster R-CNN'de kullanılan anchor box (çapa

kutuları) kavramını ilk kez tanıtmıştır. Bu kutular, kalabalık bir sahnede küçük nesnelerin tahmin edilmesini sağlar. YOLO'da çıkış, tamamen bağlı bir katmandan (fully connected layer) beklenirken, YOLOv2'de son çıkış anchor box'lar üzerinden yapılır. YOLOv2, her bounding box için sınıf olasılıklarını hesaplayarak grid hücresi başına B adet nesne tahmin eder. Faster R-CNN'deki anchor box'lar elle seçilirken, YOLOv2 eğitim görüntülerinde k-means kümeleme kullanarak daha iyi önceden uyumlu anchor box'lar arar. Başlangıçta YOLOv2, görüntüdeki küçük nesneleri tanımakta zorlanıyordu; çünkü işlem tamamlandığında görüntü oldukça küçülmüş olur ve katman üstüne katman eklenirken küçük özelliklerin korunması zorlaşır. Bunun çözümü, identity mapping adı verilen bir mimari uygulamaya dayanır: Önceki katmanlardan biri, son katmanlardan biriyle birleştirilir ve böylece görüntüden düşük seviyeli bilgiler öğrenilebilir. Ayrıca, Batch normalization eklenerek modelin genelleme yeteneği artırılmış ve eğitim sırasında daha iyi performans sağlanmıştır. Batch normalization, tüm konvolüsyon katmanlarının ardından uygulanır ve dropout katmanı kullanılmadan aşırı öğrenmeyi önler (Redmon ve ark, 2016).

Darknet-19 modeli öncelikle ImageNet sınıflandırma veri seti (girdi boyutu 248x248) üzerinde 160 epoch boyunca eğitilmiştir. Optimizer olarak stochastic gradient descent kullanılmış ve öğrenme oranı 0.1 olarak belirlenmiştir. Eğitim örnekleri için kırpma (cropping), yakınlaştırma (zooming), döndürme (rotation), pozlama değişikliği (exposure change) gibi veri artırma teknikleri uygulanmıştır. Ayrıca aynı model, 448x448 giriş boyutlu görüntülerle sınıflandırma amacıyla 10 epoch daha eğitilmiştir; bu, modelin tam boyutlu görüntülerde daha fazla filtre öğrenmesine yardımcı olmuştur. Nesne tespiti için, Darknet-19'un son konvolüsyon katmanı, rastgele ağırlıklı 6 konvolüsyon katmanı ile değiştirilmiş ve giriş görüntü boyutu 416x416 olarak ayarlanmıştır. Bu 6 konvolüsyon katmanı şunlardır:

- 3 adet 3x3 boyutunda, her biri 1024 filtreye sahip katman
- 3 adet 1x1 boyutunda, her grid hücresi için r filtre sayısına sahip katman

Pooling ile 416 boyutundaki görüntü, 13x13/N boyutuna sıkıştırılır; burada N, çözülmesi gereken sınıf sayısıdır. Değerlendirme verisi olarak Pascal VOC 20 sınıf veri seti kullanılmıştır. Her bounding box için 5 anchor box üzerinden, koordinatlar ve güven skoru dahil olmak üzere 5 değer atanır. Toplamda her grid hücresi için 5 kutu x

5 değer + 20 sınıf olasılığı = 125 uzunluğunda vektör oluşur. Anchor box'lar ile bounding box regresyonu şu şekilde güncellenir:

$$\hat{x} = \sigma(tx) + cx, \ \hat{y} = \sigma(ty) + cy \quad (3.4)$$

$$\hat{w} = pw * e\hat{}(tw), \ \hat{h} = ph * e\hat{}(th) \quad (3.5)$$

YOLOv2, yüksek verimlilik göstermiştir (çıkarım süresi yalnızca 45 FPS) ve yüksek doğruluk elde etmiştir. Ancak birkaç yıl sonra, SSD (Sun ve ark, 2021) ve RetinaNET (Oczak ve ark, 2022) gibi modeller mAP açısından YOLOv2'yi geçmiştir. Öte yandan, YOLO bazen küçük nesneleri tespit edememektedir; bunun nedeni, upsampling ve residual blok eksikliği, ve yalnızca 30 katmanlı yanlış bir ağ mimarisi kullanılmasıdır (Redmon ve ark, 2016).

3.2.3.2. YOLOv3

YOLOv3, YOLO model ailesinin bir sonraki sürümüdür ve ilk olarak Joseph Redmon ve ekibi tarafından 2018 yılında önerilmiştir (Zaidi ve ark, 2022). Tasarlanan algoritma, güncel yöntemlerle aynı doğruluğu korurken hızlı bir şekilde çalışacak şekilde geliştirilmiştir; yani performans için doğruluktan ödün verilmez. YOLOv3, Darknet-53 olarak bilinen 53 katmanlı özgün bir mimari üzerine kurulmuş ve ImageNet 1000 sınıf veri seti üzerinde önceden eğitilmiştir. Ağ derinleştikçe önemli özelliklerin kaybolmaması için YOLOv3, upsampling katmanları eklemiştir. YOLOv3'ün en iyi özelliklerinden biri, yalnızca doğru tespit yapması değil, aynı zamanda nesneleri üç farklı ölçekte tespit edebilmesidir.

YOLOv2'de olduğu gibi, YOLOv3 de çıktı üretmek için 1x1 konvolüsyon katmanı kullanır; filtre sayısı veri seti ve yapılandırmaya bağlıdır. Tespit işlemi, ağın üç farklı noktasında yer alan üç farklı boyuttaki özellik haritalarına üç 1x1 konvolüsyon dalı uygulanarak üç ölçekte gerçekleştirilir. YOLOv3, COCO 80 sınıf veri seti üzerinde test edilmiştir. Burada:

- B = 3 (anchor box sayısı)
- C = 80 (sınıf sayısı)
- Her bounding box için 5 değer

Bu nedenle, grid hücresinin vektör uzunluğu **255** olur. Bu, tespit katmanı konvolüsyon kernel boyutunu 1x1x255 yapar (Ayoosh Kathuria, 2018).

Ağ, giriş görüntüsünü sıkıştırma oranına göre üç farklı ölçekte tespit gerçekleştirir. YOLOv3'te bu ölçekte görüntüler, sırasıyla 32, 16 ve 8 faktörleriyle down-sampling yöntemi ile işlenir.

- **Büyük nesne tespiti:** 32 faktör ile küçültülen özellik haritası 13x13 boyutuna indirilir (giriş görüntüsü 416x416'dan 13x13'e küçültülür, katman 82).
- **Orta boy nesne tespiti:** Özellik haritası katman 79'dan geçer, upsample edilerek 26x26 boyutuna getirilir. Bu 26x26 derinlik, katman 61 özellik haritasıyla birleştirilir ve 1x1 konvolüsyon bloklarına gönderilir. Katman 94'ten ikinci tahmin alınır: 26x26x255 (kernel 1x1x255).
- **Küçük nesne tespiti:** Katman 91'in özellik haritası konvolüsyon katmanlarından geçirilir, 52x52 boyutuna upsample edilir. Katman 36'nın özellik haritası ile birleştirilip 1x1 konvolüsyon uygulanır. Üçüncü tahmin katmanı: 52x52x255.

Bu sayede grid çıktıları:

- 13x13 → büyük nesneler
- 26x26 → orta nesneler
- 52x52 → küçük nesneler

K-means kümeleme, anchor box'ları sıralamak için kullanılır.

- En büyük 3 anchor box → 13x13 grid
- Orta 3 anchor box → 26x26 grid
- En küçük 3 anchor box → 52x52 grid (Pulkit Sharma, 2023)

YOLOv3, sum squared loss yerine cross-entropy loss fonksiyonu kullanır. Ayrıca, softmax yerine logistic regression aktivasyonu kullanılır; bu, her bounding box için bir güven skoru ve sınıf olasılığı üretir. Çoklu sınıf sınıflandırma modeli, bir nesne için birden fazla sınıf tahmini yapabilir; yani her sınıf için olasılık eşiğin üzerinde ise o sınıf da nesneye atanabilir (Ayoosh Kathuria, 2018).

3.2.3.3. YOLOv4

YOLOv4 dışında, tüm önceki YOLO sürümleri çok büyük batch ile eğitim gerektirdiğinden, tek bir GPU ile eğitilmesi oldukça zordu. YOLOv4, çok az veri ile öğrenebilen bir nesne tespit modelidir ve eğitimi için yalnızca tek bir GPU yeterlidir. YOLOv4 çok hızlıdır ve yüksek doğruluk elde etmek için diğer tüm YOLO

sürümlerinde olduğu gibi Mish aktivasyonu, Mosaic veri artırma, cross stage partial connections, CIoU loss gibi özelliklere sahiptir (Bochkovskiy ve ark, 2020). YOLOv4 model mimarisi üç ana bileşenden oluşur:

Backbone: Modelin backbone kısmı için ImageNet sınıflandırma veri seti gibi veri setleri kullanılarak önceden eğitilmiş bir model alınır ve fine-tuning ile nesne tespiti için yeniden eğitilir. Nesne tespit ağı geleneksel olarak önceden eğitilmiş modellerin iki aşamalı bir sürecini içerir. Önceden eğitilmiş modellerden elde edilen özellik haritaları, downstream görevler için oldukça iyi çalışır. YOLOv4 backbone'u için CSPResNext50, CSPDarknet53, EfficientNet-B3 gibi çeşitli modeller mevcuttur. Bazı modeller sınıflandırma için, bazıları tespit için kullanılır. Backbone olarak CSPDarknet53 tercih edilmiştir; bu, nesne tespit görevlerinde ResNet'e göre daha iyi genel performans sağlar. Ayrıca backbone modeli küçük nesneleri daha iyi öğrenirse, giriş boyutu arttıkça küçük nesnelerin tespiti de gelişir.

Neck: Neck bölümü, backbone ve head ağları arasındaki katmanlardan oluşur ve bir özellik çıkarıcı görevi görür. Backbone ağının farklı noktalarından özellikler çıkarır. FPN, bi-FPN ve PAN gibi diğer neck modellerinden farklı olarak YOLOv4, Spatial Pyramid Pooling (SPP) kullanır; bu, receptive field'i genişletir ve sabit boyutlu çıktı üretir. SPP, motion pooling tabanlıdır; dinamik boyutlu filtrelerle feature map üzerinde max pooling uygular ve padding ile uzamsal boyutu korur. Katman çıktıları daha büyük bir feature map ile birleştirilir ve model performansı artırılır. YOLOv4 CNN backbone'da addition yerine PAN concatenation kullanır. PAN'ın iki uygulaması vardır: bottom-up ve top-down.

Head: YOLOv4'te head, YOLOv3 ile aynıdır ve üç farklı ölçekte tespit yapmak için anchor box kullanır. İnsan gözü farklı boyutlardaki nesneleri algılayabileceği gibi, model de farklı ölçeklerde nesneleri algılar.

YOLOv4 nesne tespit performansını artırmak için iki popüler bag-of-words stratejisi uygulanmıştır: Bag of Freebies (BOF) (Kirthika ve ark, 2021) ve Bag of Specials (BOS) (Nepal ve ark, 2022). Freebies Bag, modelin eğitiminde kullanılan bazı özellikleri içerecek şekilde oluşturulmuştur. Veri artırma: BOF, modelin daha önce görülmemiş yeni veriler üzerinde daha iyi genelleme yapmasına yardımcı olan veri artırma özelliklerini içerir. Görüntülere farklı artırma teknikleri uygulanır; örneğin, döndürme, doygunluk değişikliği, renk tonu (hue) değişikliği gibi. Benzer artırma

teknikleri bounding box'lara da uygulanır. Ayrıca, cutout ve random erasing gibi bazı yeni veri artırma yöntemlerinin konvolüsyonel ağların performansını da iyileştirebileceği gösterilmiştir. DropBlock ve DropConnect (Moayed ve ark, 2023) DropOut ile tam olarak aynı değildir; bunlar eğitim verisinde overfitting'i önlemede oldukça etkili iki başka regularizasyon tekniğidir. Backbone'da regularizasyon: YOLOv4 backbone'u DropBlock ile regularize edilmiştir. Ayrıca CIoU loss, YOLOv4 araştırma tabanlı BOF nesne dedektöründe de kullanılmıştır.

$CIoU$ Loss:

$$L\_CIoU = 1 - IoU + (\rho^2(b, b\_gt)/c^2) + \alpha v \quad (3.6)$$

CIoU, bounding box kaybını, ground truth ve tahmin edilen bounding box merkez noktaları arasındaki mesafenin ortalaması, ground truth ve tahmin edilen bounding box'ların kesişim oranı farkı (yani IoU) ve alan oranı olarak tanımlar. Ancak CIoU, modelin yakınsamasını ve doğruluğunu artırabilir. Bags of Specials (BOS), daha iyi performans için çıkarım ile ilgili özellikler içerir. Receptive field'i genişletmek için özellik öğrenmede kullanılan BOS ağlarından bazı örnekler şunlardır: SPP (Simple Pre-Processing), RFB (Residual Filter Bank), ASPP (Atrous Spatial Pyramid Pooling). Ayrıca YOLOv4, ince ayarlı bir özellik haritası oluşturmak için Spatial Attention Module'un geliştirilmiş bir versiyonunu da kullanır. Bu değişikliğin yanı sıra, YOLOv4 backbone'da standart ReLU aktivasyon fonksiyonu yerine Mish aktivasyon fonksiyonunu kullanır ve bu, test doğruluğunda iyileşme göstermiştir.

3.2.3.4. YOLOv5

YOLOv4'ün yayınlanmasından sadece iki ay sonra, Glenn Jocher, YOLO ailesine en son ek olarak YOLOv5'i (well-known Yoga Object Localizer'ın beşinci versiyonu) tanıttı. Model, mimari açısından YOLOv4'e oldukça benzeyen YOLOv5 tabanlıdır (ikisi de aynı backbone, neck ve head'i paylaşır). YOLOv5, Darknet yerine Pytorch ile yazılmıştır. Büyük Sınıflar – YOLOv5, önceki sürümdeki .cfg yerine .yaml formatında yapılandırma kullanır. YOLOv5'in beş versiyonu vardır: YOLOv5s, YOLOv5l, YOLOv5m, YOLOv5n ve YOLOv5x. En küçük model YOLOv5s, en büyük model ise YOLOv5x'dir. YOLOv5, kullanıcıların donanım ve doğruluk gereksinimlerine göre model seçebilmesi için ölçeklenebilir model boyutları (küçük, orta, büyük ve ekstra büyük) özelliğini tanıtmıştır. Backbone motoru hız için optimize

edilmiş ve Focus katmanları, CSP (Cross Stage Partial connections) ve PANet gibi modern yapı taşlarını desteklemiştir.

Çerçeve, veri artırmaya odaklanmıştır (ör. mosaic augmentation, adaptive image scaling, otomatik öğrenen bounding box anchor'ları). Bu, aşırı öğrenmeyi azaltmaya ve küçük ve çeşitli nesneleri daha iyi tespit etmeye yardımcı olmuştur. Kayıp Fonksiyonu: YOLOv5, bounding box regresyonu için GIoU (Generalized IoU), CIoU (Complete IoU) ve DIoU (Distance IoU) kayıplarını kullanır. Bu yeni IoU metrikleri, box hizalamasının ground-truth hesaplama doğruluğunu artırmıştır. Etkisi: YOLOv5, basitliği, modüler yapısı ve açık kaynak ekosistemi sayesinde hızla en yaygın kullanılan araç haline gelmiştir. Otonom araçlardan tıbbi görüntülemeye veya üretim süreçlerindeki karmaşık ekipmanların incelenmesine kadar çeşitli uygulamalarda kullanılmıştır (Jani ve ark, 2023).

3.2.3.5. YOLOv6

YOLOv6, endüstriyel/edge dağıtımı göz önünde bulundurularak geliştirilmiştir. Amaç, sınırlı kaynaklara sahip cihazlarda düşük çıkarım maliyeti ile dünya standartlarında doğruluk sunmaktı. Ayrıca, YOLOv6'ya RepVGG tabanlı backbone ve hafif head eklenmiştir. Yüksek hız ve gecikmesiz doğruluk sağlanmıştır. Kayıp Fonksiyonu: SIoU (Scalable IoU) kullanılmıştır; bu, IoU tabanlı kaybın daha gelişmiş bir versiyonudur. Regresyon offset eklenmesi sayesinde, SIoU ile bounding box tahmini, Almanca derinlik bounding box tahmin görevinde temel yöntemlere kıyasla daha hızlı yakınsamaya sahiptir. Bir araştırmacı ayrıca, ground truth ve tahmin edilen planer kutular arasındaki bearing açıların yakınsamayı hızlandırmada etkili olduğunu göstermiş ve SIoU'nun lokalizasyondaki potansiyelini ortaya koymuştur. Ayrıca, YOLOv6, quantization-engine-aware eğitimini de destekleyerek, düşük hassasiyetli hesaplama donanımıyla uyumlu hale gelmiştir (C. Li ve ark, 2022). Bu nedenle, mobil işlemciler veya gömülü bilgisayarlar gibi ürünlerde kullanılabilir. Hız ve doğruluk arasındaki dikkat çekici dengesi sayesinde, YOLOv6, gerçek zamanlı gözetim sistemleri, robotik uygulamalar ve edge bilişimin özellikle gecikmeye duyarlı olduğu görüntüleme sistemleri için son derece uygundur.

3.2.3.6. YOLOv7

YOLOv7, YOLO ailesinde önemli bir dönüm noktasıdır. Hem hız hem de doğruluk açısından mümkün olan sınırları zorlayan birçok yeni fikir sunmuştur. Genişletilmiş

Verimli Katman Birleştirme Ağı (E-ELAN) ile eğitim sırasında gradyan daha sık geri iletilmiş ve özellikler daha sık yeniden kullanılmıştır. Bu, yeni karmaşıklık katmanları eklemeden tüm ağı daha derin ve kararlı hale getirmiştir. YOLOv7, modelin karmaşıklığını çıkarım sırasında azaltmak için re-parametrizasyon tekniklerini uygulamış, ancak eğitim sırasında ağın karmaşıklığını korumuştur. Bu, yüksek ölçeklenebilir veri tabanlarına dağıtım sürecini hesaplama açısından hafif hale getirmiştir (H. Yang ve ark, 2023). En son güncelleme, Efficient IoU (EIoU) kullanarak mesafe, ölçek ve örtüşmeyi daha etkili şekilde ağırlıklandıran bounding box regresyonu kullanmıştır. YOLOv7, önceki sürümleri hem ortalama hassasiyet (mAP) hem de saniye başına kare (FPS) açısından geçerek benchmarklarda en son teknoloji sonuçlarını göstermektedir. Akademik araştırmalarda, trafik izleme, drone görselliği ve ileri seviye otonom sürüş sistemlerinde giderek daha fazla benimsenmiştir.

3.2.3.7. YOLOv8

YOLOv8, YOLO modellerinin işlevselliğini tek bir yapı içinde çoklu görev eğitimine kadar genişletmiştir. YOLOv8, yalnızca nesne tespiti yapmakla kalmayıp aynı zamanda görüntü segmentasyonu ve görüntü sınıflandırması görevlerini de yerine getirecek şekilde tasarlanmıştır. Daha iyi bir backbone, daha iyi özellik çıkarımı ve daha optimize edilmiş bir detection head ile birleştirilmiştir. YOLOv8 tarafından bounding box regresyonu için önerilen Distribution Focal Loss (DFL), kutu değerlerini sabit miktarlar olarak tahmin etmek yerine, bunları olasılık dağılımları olarak modelleyerek daha iyi konumlandırma sağlar. DFL, Complete IoU (CIoU) ile birleştirilerek küçük ve üst üste binen nesneler için daha yüksek doğruluk elde edilmiştir. YOLOv8'in çeşitli uygulamaları öne çıkarılmış ve esnekliği için övgü almıştır (Q. Zhou ve ark, 2024). Sağlık teşhisi, tarım izleme, otonom araç algısı gibi daha fazla kullanım alanına uygulanabilir hale gelmiştir. Açık kaynak sürümü, kullanımının daha da yayılmasına yardımcı olmuştur.

3.2.3.8. YOLOv9 ve YOLOv9c (2024)

YOLOv9, Mini modunda, YOLO'nun en son yinelemesidir. StreamOSL, yoğun görsel ortamlarda gerçek zamanlı verimliliği korurken doğruluğu artırmada önemli ilerlemeler sağlar. YOLOv9'da backbone ve detection head üzerinde yapısal ayarlamalar önerilmiş ve yoğun nesne yönetimi geliştirilmiştir. YOLOv9c versiyonu ise bulut destekli ve otonom sürüş sistemleri için daha da optimize edilmiştir. Bu sürüm, Distribution Focal Loss (DFL), CIoU ve Programmable Gradient Information

(PGI) kombinasyonunu kullanır. PGI, eğitim modunda gradient akışını iyileştirerek daha hızlı yakınsama ve daha yüksek doğruluk sağlar. DFL + CIoU + PGI kombinasyonu, milyonlarca nesne içeren veri setlerinde bile güvenilir ve doğru eğitim sağlar. YOLOv9 ve YOLOv9c, mevcut en doğru gerçek zamanlı tespit sistemlerinden bazılarıdır. Yüksek doğruluk, recall ve ortalama hassasiyet (mAP) sağlarken, düşük çıkarım süresini de korurlar (Nguyen ve ark, 2025). Mimari yapıları, onları otonom araçlar, akıllı şehir uygulamaları, robotik ve bulut tabanlı büyük video analitiği için son derece uygun kılar.

3.2.4. YOLO'nun matematiksel temelleri ve evrimi (v1–v9)

Bu bölüm, YOLO model ailesinin matematiksel temellerini sunar. CNN ve nesne tespiti için temel oluşturan genel denklemlerle başlar ve ardından YOLO v1'den v9'a kadar yapılan iyileştirmeleri açıklar.

Konvolüsyon İşlemi:

$$F(i,j) = \Sigma\,\Sigma\, I\,(\,i + m, j + n\,) * K(\,m,n\,) \tag{3.7}$$

Burada $F(i,j)$ çıktı feature map, III giriş görüntüsü ve KKK kernel'dir.

Bounding Box Tahmini:

$$Box = (x,y,w,h) \tag{3.8}$$

$$\hat{x} = \sigma(tx) + cx,\ \hat{y} = \sigma(ty) + cy \tag{3.9}$$

$$\hat{\mathrm{w}} = pw * e\hat{}(tw),\ \hat{\mathrm{h}} = ph * e\hat{}(th) \tag{3.10}$$

Güven Skoru (Confidence Score):

$$C = P(Object) \times IoU(pred, truth) \tag{3.11}$$

Sınıf Tahmini (Class Prediction):

$$Final\ class\ confidence = P(Class\_i) \times IoU(pred, truth) \tag{3.12}$$

Kayıp Fonksiyonu (YOLO Base):

$$L = \lambda coord\ \Sigma\Sigma\ 1ij\hat{}obj\ [(x - \hat{x})^2 + (y - \hat{y})^2] + \lambda coord\ \Sigma\Sigma\ 1ij\hat{}obj\ [(\sqrt{w} - \sqrt{\hat{\mathrm{w}}})^2 + (\sqrt{h} - \sqrt{\hat{\mathrm{h}}})^2] + \Sigma\Sigma\ 1ij\hat{}obj\ (C - \hat{\mathrm{C}})^2 + \lambda noobj\ \Sigma\Sigma\ 1ij\hat{}noobj\ (C - \hat{\mathrm{C}})^2 + \Sigma\ 1i\hat{}obj\ \Sigma\ (p(c) - \hat{p}(c))^2 \tag{3.13}$$

Intersection over Union (IoU):

$$IoU = Area(B\_pred \cap B\_truth) / Area(B\_pred \cup B\_truth) \quad (3.14)$$

3.3. Şifrelenmiş Verilerle Yapay Zekâ Entegrasyonu

YZ şifrelenmiş verilerle entegre etmenin en büyük engeli verimliliktir. Şifreleme, derin öğrenme modelleri için hesaplama açısından maliyetlidir ve bu modellerin çok hızlı çıkarım sürelerine ihtiyacı vardır. Homomorfik şifreleme, YZ ve gizliliği entegre etmek için temel bir teknolojidir; teorik olarak şifreli veriler üzerinde hesaplamalar yapılmasını sağlar. Bu çerçevede, sinir ağları tarafından üretilen çıktılar üzerinde güvenli işlemler örneğin vektör karşılaştırmaları ve analiz bulut sunucusu ham, şifrelenmemiş verilere erişmeden gerçekleştirilebilir. Şifrelenmiş AI, düz metin işlemlerine göre daha yavaş olsa da, algoritma verimliliği ve donanım hızlandırmaları gelecekte bunu uygulanabilir bir yöntem haline getirmektedir. Otonom araçlar bağlamında, bu entegrasyon hassas görüntü akışlarının ve coğrafi koordinatların bulutta güvenli bir şekilde analiz edilmesini sağlar; bunlar üçüncü taraflarla paylaşılmaz.

Kriptografik yapıların YZ ile entegrasyonunu zorunlu kılan iki temel gereklilik vardır:

- **Veri gizliliği:** Otonom araçlar, medikal cihazlar ve akıllı şehir uygulamaları, bilgi akışlarının sızdırılmadan tamamen güvenli olmasını sağlamak zorundadır.
- **Akıllı karar verme:** Şifrelenmiş veriler sürekli olarak kilitli olsa bile, sofistike işlemler yapılmadan ve veriler değerli bilgiye dönüştürülmeden kullanışlı hâle gelmez.

Homomorfik şifreleme, YZ modellerinin verileri çözmeden şifreli veriler üzerinde çalışmasına olanak sağlar. Ham veriler bulut sağlayıcısına bile açılmaz. Şifrelenmiş bir veri kümesinden elde edilen sonuçlar, dış tarafların verileri değiştirmediğine dair güven sağlar. Bulut üzerinde çalışan YZ modelleri, büyük hacimli şifreli verileri işleyebilir; bu, araç navigasyonu gibi zaman hassas uygulamalar için kritiktir. YZ algoritmaları, verileri çözmeden şüpheli faaliyetleri, kötü amaçlı yazılımları veya saldırı girişimlerini izlemek için kullanılabilir. Homomorfik şifrelemenin yüksek hesaplama yükü vardır ve derin öğrenme modelleri, aynı modeli düz metin üzerinde çalıştırmaya göre şifreli veriler üzerinde çok daha yavaş çalışır. Bu, düşük gecikme gerektiren sistemler için (örneğin otonom araçlar) kritik önemdedir. Anahtar yönetimi

sorunu vardır. Şifreleme anahtarları geniş ölçüde paylaşıldığında (örneğin bulutlarda), sızdırılmamaları için iyi yönetilmelidir.

4. SİSTEM TASARIMI VE ÖNERİLEN YÖNTEM

Güvenli ve akıllı bir otonom araç sistemi geliştirmek, iki önemli teknolojinin birleşimini gerektirir. İlk teknoloji, araçların çevresini tanımasını ve anlamasını sağlayan YZ dır. İkincisi ise, bu süreçte kullanılan bilgileri gizli ve güvenli hâle getiren kriptografidir. Bu teknolojiler entegre edildiğinde, araçların gerçek yaşam koşullarında güvenli bir şekilde kullanılabileceği ve hassas verilerin korunarak saklanabileceği bir platform sağlanır. Otonom araçlar, çeşitli sensörlerden üretilen sürekli veri akışına dayanır. Kameralar, yolları, insanları ve trafik ışıklarını canlı olarak kaydeder. LiDAR ve radar, nesneleri algılar ve mesafeleri üç boyutta ölçer. GPS, araçların konumunu sağlar ve bulut ile diğer araçlar ve altyapılarla koordinasyonu destekler. Bu büyük miktardaki navigasyon verisinin güvenliği gereklidir, aksi takdirde yasa dışı kullanım ve saldırılara maruz kalabilir. Güvenlik ve gizlilik olmadan, saldırganlar verilere erişebilir, verileri değiştirebilir veya gizlilik ihlallerine ve kazalara yol açabilir. Bu risklere karşı, bu sistem tasarımında homomorfik şifreleme kullanılmıştır. Arama, eşleştirme ve sınıflandırma işlemleri, bu tür şifreleme sayesinde doğrudan şifrelenmiş veriler üzerinde uygulanabilir. Böylece, hassas bilgiler herhangi bir yerde çözülmesine gerek kalmadan işlenebilir. Bu, araçlar tarafından toplanan bilgilerin bulutta analiz edilmesini güvenli hâle getirir, gizliliği korur ve bilgi sızıntısını önler. Bu arada, YZ araçların karar verme kapasitesini artırmak için kullanılır.

Güvenli sürüşün en önemli bileşenlerinden biri nesne tespitidir. Önerilen sistem, derin öğrenmeye dayalı en son nesne tanıma modellerinden biri olan YOLOv9c modelini içerir. Bu model, gerçek zamanlı nesneleri tanır, sınırlayıcı kutular sağlar ve araçların navigasyonu ile engellerden kaçınmasına dair bilgi sunar. Karmaşık senaryolarda, örneğin şehir içi trafik veya otoyollarda, tespit doğruluğunu artırır. Tasarım aynı zamanda araçlar arası, araç-bulut sunucu ve araç-idari kontrol sistemi arasındaki tüm mimari iletişimi de kapsar. Araçlar verileri toplar, şifreler ve gönderir. Büyük ölçekli depolama ve şifreli işlem bulut tarafından yapılır ve yöneticiler erişim kontrollerini ve anahtarları yönetir. Bu rol ayrımı, araçlar üzerine

hesaplama yükünü en aza indirirken, sistem genelinde gizlilik ve güveni korur. Bu sistemin amacı, güvenli, güvenilir ve ölçeklenebilir bir yapı sağlayarak akıllı ulaşımı desteklemektir. Çözüm, otonom araçların operasyonel gereksinimlerini karşılayacak ve en son kriptografi ve yapay zeka gelişmelerini entegre ederek gizlilik koruma gereksinimlerini artıracaktır. Bu sayede şifrelenmiş fotoğraflarda nesnelerin tanınması ve doğrulanması mümkün olur, tipik siber saldırılara karşı direnç sağlanır ve performans ile güvenlik arasında bir denge kurulur.

4.1. Sistem Mimarisi

Önerilen sistem mimarisi üç katmana ayrılmıştır: araç tarafı, bulut tarafı ve yönetim kontrol tarafı. Her katmana otonom araçlar için güvenli, verimli ve akıllı nesne algılama sağlamak üzere farklı görevler atanmıştır.

4.1.1. Araç tarafı

Bilginin ilk kaynağı otonom araçlardır. Bu taraftaki temel işlevler şunlardır:

Veri toplama

- Araçlara monte edilmiş kameralar yol, trafik ışıkları ve yayaların gerçek zamanlı görüntülerini alır.
- LiDAR, engellerin 3D haritalarını oluşturur.
- Radar, farklı hava koşullarında hareketli nesneleri tespit eder.
- GPS, araçların gerçek konumunu sağlar.

Veri ön işleme

- Ham veriler analiz edilecek şekilde yapılandırılmış vektörlere dönüştürülür.
- Ön işleme sırasında gürültü ve alakasız bilgiler azaltılır.

Veri şifreleme

- Bilgi iletimi öncesinde homomorfik şifreleme kullanılır.
- Şifreleme sayesinde ham görüntüler veya sensör verileri araçta düz metin halinde asla bırakılmaz.
- İletişim kanallarına yönelik saldırılara karşı hassas bilgiler korunur.

4.1.2. Bulut tarafı

Bulut, büyük ölçekli veri işleme ve güvenli depolama sağlar. Ana görevleri şunlardır:

Güvenli veri depolama

- Araçlardan gelen şifrelenmiş veriler, çözülmeye gerek kalmadan depolanır.
- Homomorfik şifreleme, hassas bilgilerin ifşasını önler.

Şifreli işleme

- Çıkarma, benzerlik kontrolü ve doğrulama gibi işlemler şifrelenmiş vektörler üzerinde doğrudan yapılır.
- Bu süreçte bulutta verilerin çözülmesine gerek yoktur.

Yapay Zeka entegrasyonu

- YOLOv9c gerçek zamanlı bir nesne algılayıcıdır.
- Model, şifrelenmiş veri akışlarındaki nesneleri tanır ve kategorize eder.
- Ham görüntüler açığa çıkarılmadan yüksek doğruluk sağlanır.

Araçlara geri bildirim

- İşlenmiş sonuçlar araçlara gönderilir ve karar alma süreçlerini destekler.
- Engellerden kaçınma ve navigasyon gibi faaliyetleri kolaylaştırır.
- Trafik akışına gerçek zamanlı dinamik yanıt verilmesini sağlar.

4.1.3. Yönetim kontrol tarafı

Bu katman, sistem içinde yönetişim, güven ve hesap verebilirlik sağlar. İşlevleri şunlardır:

Anahtar yönetimi

- Şifreleme ve çözme anahtarları güvenli bir şekilde saklanmalıdır.
- Yetkisiz erişim veya içeriden kötüye kullanımı önler.

Erişim kontrolü

- Hangi kullanıcı veya aracın hangi verilere erişebileceğini belirler.
- Ağın yalnızca yetkili araçlar ve bulut hizmetlerinden oluştuğunu doğrular.

Kimlik doğrulama

- Araçların ve sunucuların sisteme girişini doğrular.
- Sahte veya taklit girişlere karşı koruma sağlar.

İzleme ve denetim

- Veriler sistem katmanları arasında izlenir.
- Güvenlik denetimleri ve uyum kayıtları tutulur.
- Hesap verebilirlik ve şeffaflık sağlanır.

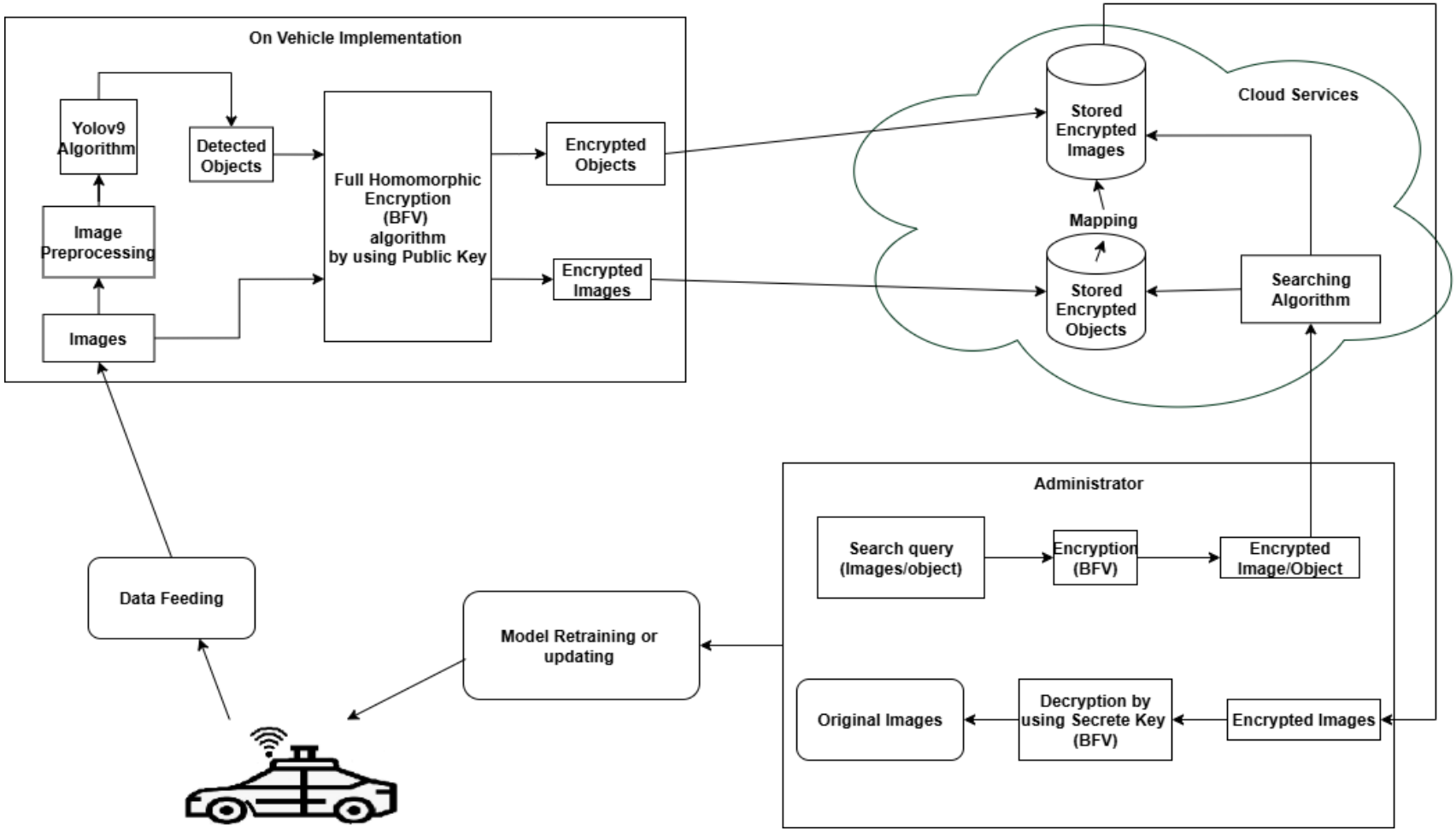
Yönetici bulut sunucusuna belirli bir sorgu gönderdiğinde, örneğin belirli bir resmi aramak istediğinde, arama algoritması bu talebe karşılık gelen şifrelenmiş görüntüleri sağlar. Bu görüntüler, otonom sürüş özelliklerinin en yüksek doğrulukla elde edilmesi amacıyla ileri araştırmalar veya model yeniden eğitimi için kullanılabilir. Sadece gizli anahtara sahip yönetici, herhangi bir şifrelenmiş veriyi çözebilir. Önerilen sistem mimarisinin genel akışı Şekil 4.1'de gösterilmektedir.

4.1.4. Bileşenlerin entegrasyonu

Bu üç bileşenin birleşimi, sistemin gücünü oluşturur. Otonom araçlar veriyi oluşturur, şifreler ve buluta gönderir; burada veri gizli bir şekilde işlenir ve sistemin yönetim tarafı kuralları uygular ve güveni denetler. Bu katmanların birleşimi, verimli, akıllı ve güvenli bir ekosistem oluşturur. Bu güvenli iletim tasarımı, yalnızca otonom araçların teknik gereksinimlerini karşılamakla kalmaz, aynı zamanda güncel siber tehditlere karşı da direnç sağlar. Mimari, ölçeklenebilir, sağlamdır ve bileşenler arasında görevlerin dikkatli dağıtımı sayesinde gizliliği korur.

4.2. YOLO'nun Çerçeveye Entegrasyonu

İnsan gözü bir görüntüdeki nesneleri anında tanıyabilir. Bilgisayarla görme (computer vision) artık bu tür zorlu görevlerin gerçekleştirilme şeklini tamamen dönüştürdü. Nesne tespit algoritmaları, insan müdahalesi olmadan otonom sürüşün her zaman ve her hava koşulunda yapılabilmesine olanak tanır. Bu algoritmalar ayrıca sürücülere gerçek zamanlı yol bilgisi de sağlayabilir. Bir görüntüdeki nesnelerin konumlandırılması (localization) ve sınıflandırılması (classification), nesne tespitinde birlikte çalışır. Nesne konumlandırma, bir görüntüdeki nesneyi veya nesneleri kapsayan sınır kutuları (bounding boxes) kullanarak tespit etme sürecidir; nesne sınıflandırma ise görüntüdeki nesneye belirli bir sınıf etiketi atama sürecidir.



Şekil 4.1. Genel çerçevenin akışı

Günümüze kadar, CNN'ler kullanılarak birçok gelişmiş derin öğrenme nesne tespit algoritması geliştirilmiştir. Önceki modellerde, deformable parts model (DPM) dahil, nesneleri tespit etmek için kaydırmalı pencereler (sliding windows) kullanılıyordu. Daha sonra, R-CNN en etkili CNN tabanlı nesne tespitçilerinden biri olarak geliştirildi. Potansiyel bölgeler belirlendikten sonra, R-CNN her bölgenin özelliklerini çıkarır ve ardından çıkarılan özellikler bir sınıflandırıcı (classifier) modeli ile sınıflandırılır. R-CNN'deki eksiklikler, fast R-CNN ve daha sonra Faster R-CNN gibi R-CNN'in iki uzantısının geliştirilmesine yol açtı. Bu yöntemler nispeten zorluydu ve zaman alıcıydı çünkü birçok bileşenin ayrı ayrı eğitilmesi gerekiyordu. Tüm bu sorunlara çözüm, YOLO'nun tanıtılmasıyla geldi. Adından da anlaşılacağı gibi, YOLO seleflerinden farklıdır; çünkü nesnenin nerede olduğunu ve türünü belirlemek için görüntüye yalnızca tek bir bakışta bakar.

Basit işlem hattı sayesinde YOLO çok hızlıdır, ancak R-CNN ailesindeki diğer modellere göre biraz daha az hassastır. YOLO'nun yüksek hızı, temel modelde saniyede 45 kare ve geliştirilmiş sürümde 150 kareyi geçen çıkarım süresiyle hızlı gerçek zamanlı tespiti mümkün kılar. YOLO modeli, nesne tespitini bir regresyon görevi olarak çözmek için görüntüyü özellik seti olarak, bir sınır kutusunu ve sınıf olasılıklarını çıktı olarak uygular. YOLO birçok kez geliştirilmiş ve V1'den en son YOLO V9'a kadar sürümler ortaya çıkmıştır. Mimari ve model eğitimi üzerinde yapılan bir dizi iyileştirme ile en yeni sürümler, geliştirilmiş sonuçlar göstermiştir. Tablo 4.1, çeşitli çalışmalarda YOLO modellerinin performans karşılaştırmasını içermektedir. Tablo, farklı YOLO nesne tespit modellerini F1 skoru, yani 0.5 ve 0.75 IoU eşiklerinde ortalama doğruluk (mAP), geri çağırma (recall) ve hassasiyet (precision) değerlerine göre karşılaştırmaktadır. YOLOv6s (Nusari ve ark, 2024) ve YOLOv9c (Nusari ve ark, 2024), nesne tespitinde en yüksek doğruluk oranına sahip sistemlerdir (sırasıyla 0,948 ve 0,96) ve yanlış pozitif sayıları en azdır. YOLOv6s'in (Nusari ve ark, 2024) geri çağırma oranı 0,897 olup, bu da modelin gerçek nesnelerin çoğunu tanıyabildiğini göstermektedir. YOLOv8s'in (Nusari ve ark, 2024) F1 skoru 0,922 ile en iyi dengeyi göstererek geri çağırma ve hassasiyet arasında iyi bir uyum sağlar. YOLOv7 (Nusari ve ark, 2024) ve YOLOv9 (Hasan Ali Akyürek ve ark, 2024) her iki eşikte de yüksek mAP değerleri sergileyerek çeşitli tespit koşullarında güvenilirliklerini kanıtlamaktadır.

Tablo 4.1. YOLOv5'ten YOLOv9'a modellerin performans karşılaştırmalı analizi

Modeller	Geri Çağırma (Recall)	mAP @ 0.5	mAP @ 0.75	Hassasiyet (Precision)	F1 Skoru
YOLOv9n (X. Chen ve ark, 2024)	76.7	85.3	-	82.5	0.79494
YOLOv9c (Nusari ve ark, 2024)	0.922	0.977	0.925	0.96	0.94062
YOLOv9 (Hasan Ali Akyürek ve ark, 2024)	0.801	0.921	0.721	0.929	0.86027
YOLOv8s (Nusari ve ark, 2024)	0.908	0.966	0.918	0.936	0.92179
YOLOv8 (H. Wang ve ark, 2024)	0.785	0.779	-	0.916	0.84546
YOLOv8 (Hasan Ali Akyürek ve ark, 2024)	0.837	0.899	0.646	0.873	0.85462
YOLOv7 (Yu ve ark, 2023)	0.601	0.64	0.338	0.629	0.61468
YOLOv7 (Xing ve ark, 2023)	0.72	0.75	0.36	0.78	0.7488
YOLOv7 (Hasan Ali Akyürek ve ark, 2024)	0.776	0.844	0.466	0.836	0.80488
YOLOv7 (Nusari ve ark, 2024)	0.885	0.964	0.889	0.966	0.92373
YOLOv6s (Nusari ve ark, 2024)	0.897	0.965	0.917	0.948	0.9218
YOLOv6 (Xing ve ark, 2023)	0.62	0.72	0.39	0.75	0.67883
YOLOv6 (Hasan Ali Akyürek ve ark, 2024)	0.801	0.894	0.585	0.878	0.83773
YOLOv5 (Yu ve ark, 2023)	0.606	0.633	0.321	0.62	0.61292
YOLOv5 (Xing ve ark, 2023)	0.66	0.71	0.37	0.77	0.71077

YOLOv8 (H. Wang ve ark, 2024) düşük mAP değerlerine ve 0,75 IoU eşikinde mAP değerine sahip olmasa da, F1 skoru 0,845 ile hâlâ yüksek bir performans göstermektedir. YOLOv9n (X. Chen ve ark, 2024) bağlamında, alışılmadık hassasiyet ve geri çağırma değerlerine (sırasıyla 82,5 ve 76,7) rağmen yüksek performans bildirilmektedir ve YOLOv6 (56) iyi hassasiyet ve mAP göstermektedir. Daha eski modellerin, örneğin YOLOv5 (Yu ve ark, 2023) ve YOLOv7 (Yu ve ark, 2023),

performansı daha düşük olup, bu durum yeni YOLO sürümlerinin tespit ve genel performanstaki önemli gelişimini ortaya koymaktadır. YOLOv9c (Nusari ve ark, 2024), ortalama nesne tespiti için en uygun model olarak en yüksek hassasiyet (0,96), geri çağırma (0,922), mAP (IoU 0,5'te 0,977) ve F1 skoru (0,941) değerlerini sunmaktadır. Bu araştırma karşılaştırmasına dayanarak, nesne tespiti için bu çerçevede YOLOv9c seçilmiştir. Bilinen nesne tabanlı aramalarda YOLOv9c önerilmektedir.

Görsellerdeki nesneleri bulmak için YOLOv9c nesne tespit algoritması kullanılır. Görsellerdeki nesneleri tanımak ve sınıflandırmak için YOLOv9c nesne tanıma algoritması uygulanır. YOLOv9c, gerçek zamanlı çalışan bir nesne tespit sistemidir; görüntüyü bir ızgaraya böler ve her hücre için sınıf olasılıkları ile sınırlayıcı kutuları tahmin eder. Bu sayede nesne tanıma işlemi bir regresyon problemi olarak ele alınır. YOLOv9c eğitimi için Kaggle üzerinde erişilebilen büyük ve etiketli bir görsel koleksiyonu seçilmiştir (Alin Cijov, t.y.). Bu çalışmanın temelini, IoT uygulamalarında karşılaşılan farklı senaryoları kapsayan bir otonom araç nesne tanıma veri setinin seçimi oluşturur. Bu amaçla Kaggle veri seti tercih edilmiştir çünkü oldukça kapsamlıdır. Veri setinde 22.2 bin, toplamda 935 MB dosya bulunmaktadır. Model, eğitildikten sonra yeni görsellerdeki nesneleri tanımak için kullanılabilir. YOLO uygulamasının ardından, her görüntüdeki tüm nesneler çıkarılır ve yeni bir obj_image olarak kaydedilir. Bu obj_image şifrelendikten sonra, hem şifreli görsel hem de obj_image, eşlemesiyle birlikte bulut sunucuda saklanır. Şifreleme için BFV algoritması kullanılır.

4.3. Homomorfik Şifreleme Şeması

Homomorfik şifreleme, şifrelenmiş veri üzerinde doğrudan hesaplama yapılabilmesini sağlayan bir şifreleme yöntemidir. Diğer yöntemlerde veriyi kullanmak için önce çözmek gerekirken, homomorfik şifreleme sayesinde veriler hiçbir zaman ham formunda açığa çıkmaz ve gizlilik korunur. Bu özellik, çok yüksek miktarda hassas verinin güvenli bir şekilde işlenmesi gereken otonom araçlarda bulut işlemleri için son derece uygundur.

4.3.1. Homomorfik şifreleme kütüphaneleri

Homomorfik şifreleme kütüphaneleri (Doan ve ark, 2023; Pulido-Gaytan ve ark, 2021), uzmanlar ve araştırmacıların biyometri gibi çeşitli uygulamalarda homomorfik şifrelemeyi kullanmalarını desteklemek için gereklidir. Son yıllarda bu kütüphanelerin

evrimi ve optimizasyonu önemli ölçüde ilerlemiş, böylece uygulamaların verimliliği artırılmıştır (Abreu ve ark, 2022). Aşağıda bazı önemli homomorfik şifreleme kütüphaneleri özetlenmiştir:

- **HElib**: IBM tarafından 2013'te geliştirilen HElib, CKKS ve BGV homomorfik şifreleme şemalarını kullanır ve C++ destekler. 2024 başında yayınlanan en son sürümü 2.2.2 ile katmanlı homomorfik şifreleme işlemlerinde yüksek verimliliğiyle dikkat çekmektedir (Halevi ve ark, 2014).
- **Python-Paillier**: Paillier kriptosisteminin Python uygulamasıdır. 2014'te yayımlanmış, 2022'de 1.5.1 sürümü yayınlanmıştır ve Python uygulamaları için uygundur (Data61, 2013).
- **Java-Paillier**: Java ile yazılmış, Paillier kriptosisteminin toplamsal homomorfik şifreleme implementasyonudur. Orta 2010'larda kullanıma sunulmuş ve Java uygulamalarını destekler. Son sürümü 1.0'dır.
- **Microsoft SEAL**: 2015'te Microsoft tarafından yayımlanan SEAL (Simple Encrypted Arithmetic Library), C++ ile geliştirilmiş olup Python, .NET ve diğer dillerde kullanılabilir. BFV ve CKKS algoritmalarını uygular. 2024 başında 4.1.1 sürümü yayınlanmıştır (Titus ve ark, 2018).
- **FHEW**: 2014'te yayımlanan FHEW (FHE over the Torus), ikili devre seviyesinde hızlı homomorfik şifrelemeyi sağlar. C++ ile geliştirilmiştir ve 1.1 sürümü en güncel olanıdır (Ducas ve ark, 2015).
- **TFHE**: 2016'ta yayımlanan TFHE, torus üzerinde hızlı FHE bootstrapping sağlar. C++ ile düşük gecikmeli ikili kapı işlemlerini destekler ve sürümü 1.0.2'dir (Chillotti ve ark, 2016; Z. Li ve ark, 2016).
- **Pyfhel**: SEAL ve HElib için Python arayüzü sağlayan kütüphanedir. 2018'te yayımlanmış ve CKKS, BFV şemalarını destekler. Güncel sürümü 2.5.0'dır (Pyfhel Library, 2023).
- **PALISADE**: 2017'de yayımlanan tam teşekküllü lattice tabanlı kriptografi kütüphanesi. BFV, BGV, CKKS şemalarını destekler ve C++ ile kullanılabilir. Sürümü 1.11.8, güvenli çok taraflı hesaplama ve yüksek performanslı homomorfik şifrelemeye odaklanır (Polyakov ve ark, 2017).

- **Lattigo**: 2019'da yayımlanan lattice tabanlı bir kütüphane olup Go dili için CKKS ve BFV şemalarını uygular. 2024 itibarıyla sürümü 2.2.3'tür (Mouchet ve ark, 2020).
- **TenSEAL**: 2020'de geliştirilen TenSEAL, makine öğrenmesi uygulamalarını homomorfik şifreleme üzerinde çalıştırmak için tasarlanmıştır. Python ile PyTorch gibi derin öğrenme modelleri ile kullanılabilir ve CKKS şemasına katkıda bulunur. Güncel sürümü 0.3.10'dur (Benaissa ve ark, 2021).
- **OpenFHE**: PALISADE'ın açık kaynaklı bir fork'u. 2022'de yayımlanmış, BFV, BGV, CKKS şemalarını destekler ve C++ ile kullanılabilir. Sürümü 1.0.2'dir (Al Badawi ve ark, 2022).
- **HEANN**: 2016'da yayımlanan Python kütüphanesi, CKKS tabanlı yaklaşım homomorfik şifrelemeye odaklanır. Gerçek sayı verilerinin gizlilik korumalı işlemleri için uygundur. En güncel sürümü 1.3.1'dir (Cheon ve ark, 2017).

Bu çerçeve için BFV seçilmiştir çünkü:

- Görsellerdeki piksel değerleri tam sayılardır ve BFV bunlarla uyumludur.
- Aşırı hesaplama maliyeti olmadan yüksek güvenlik sunar.
- Vektör karşılaştırması yoluyla nesne tanımada kritik olan şifreli çıkarma işlemlerini destekler.
- Bulut sunucularına verileri güvenli şekilde işleme imkânı verir, ham veriler gizli kalır.

4.3.2. BFV tam homomorfik şifreleme şeması

Bu çalışmada, şifrelenmiş veri üzerinde sınırsız işlemin şifre çözmeden yapılabilme imkânı nedeniyle FHE (Rauthan, 2022) tercih edilmiştir. Gizliliği koruyan işleme ihtiyaç duyan bulut tabanlı otonom araç uygulamalarında, AES veya RSA gibi standart şifreleme şemaları yeterli değildir; çünkü bunlar yalnızca veri gizliliğini sağlar, ancak şifreli veri üzerinde doğrudan hesaplama yapılmasına imkân vermez. FHE, çok sayıda homomorfik şifreleme algoritması arasında PHE (Partially Homomorphic Encryption) (Medileh ve ark, 2023) ve SHE (Somewhat Homomorphic Encryption) (Subramaniyaswamy ve ark, 2022) üzerinde tercih edilmektedir; çünkü FHE sınırsız toplama ve çarpma işlemlerine izin verir. PHE ve SHE, çok boyutlu şifreli görüntü arama ve nesne tespiti için uygun değildir: PHE yalnızca bir tür işlem (toplama veya

çarpma) destekler, SHE ise sınırlı sayıda işlemden sonra şifrenin çözülmesini gerektirir. FHE ise istatistiksel analiz, özellik çıkarımı, filtreleme, benzerlik ölçümü ve mesafe hesaplamaları gibi sınırsız sayıda işlemin şifreli veriler üzerinde yapılmasına olanak tanır.

Bu sınırlı şemalardan farklı olarak, FHE veri gizliliğini korurken herhangi bir hesaplama türüne izin verir. Ayrıca, indeks tabanlı şifreli arama yöntemlerinden daha güvenlidir; çünkü bu yöntemler genellikle arama ve erişim desenlerini veya önceden hesaplanmış meta verileri açığa çıkararak kişisel bilgilerin sızmasına neden olabilir. Önerilen çözüm, düz metin veya ara sonuçlar asla açığa çıkmadığı için çıkarım saldırılarına ve pasif dinlemeye karşı dirençlidir. FHE ayrıca RSA gibi klasik şifreleme tabanlı sistemlere göre hem kaba kuvvet hem de kuantum saldırılarına karşı oldukça dayanıklıdır; RSA, tam sayı çarpanlarına dayalı iken FHE güvenliği, iyi bilinen post-kuantum güvenli kriptografik bir problem olan Learning With Errors (LWE) problemine dayanır (Janani ve ark, 2023). Bu özellikler sayesinde FHE, bulut tabanlı şifreli görüntülerin güvenli bir şekilde işlenmesini sağlarken, hiçbir düz metin bilgisinin açığa çıkmasına izin vermez. Ayrıca, üçüncü tarafların şifrelenmiş veri setlerini çekip almasına ve asıl verilere erişmeden hesaplamalar yapmasına imkân tanır. Önerilen FHE tabanlı çözüm, düz metin sızıntısı olmadan anlık şifreli arama ve hesaplama yapılmasını destekler. Bu, indeks tabanlı şifreli aramalardan farklıdır; çünkü indeks tabanlı aramalarda, aramayı desteklemek için önceden hesaplanmış meta veriler kullanılır ve bu durum kısmi bilgi sızıntısına yol açabilir. Bu esneklik, gizlilik ve gerçek zamanlı işlem gereksinimlerinin dengelenmesi gereken senaryolarda, örneğin otonom sürüşte, kritik öneme sahiptir.

FHE, yetkilendirilmiş üçüncü tarafların komutla şifreli veriler üzerinde hesaplama yapabilmesini ve ihtiyaç duydukları bilgiyi elde etmelerini sağlarken, bulutlarda saklanan verilerin gizliliğini tamamen koruduğu için hem gizlilik hem de hesaplama esnekliği sunar. BFV şeması, Brakerski'nin şifreleme şeması olan bir FHE şemasıdır (J. Fan ve F. Vercauteren, 2012; Z. Brakerski, 2012). BFV, şifrelenmiş girdiler üzerinde çarpma ve toplama işlemleri yapılmasına imkân tanır ve şifre çözme gerektirmeden güvenli hesaplamayı sağlar. BFV FHE şemasının düz metni ve şifreli metni sırasıyla Rx ve Ry halkaları ile gösterilir; burada x ve y tam sayılardır ve y, x'ten büyüktür, x ise 1'den büyüktür. x ve y'nin asal veya aralarında asal olması gerekmez. Şema aşağıdaki işlemleri kullanır:

- $\lfloor \cdot \rfloor$: taban alma (floor) fonksiyonunu gösterir.
- $\lceil \cdot \rceil$: en yakın sayıya yuvarlama fonksiyonunu gösterir.
- $[\cdot]_y$: mod y ile indirgeme işlemini gösterir.

$a \leftarrow S$ ifadesi, a'nın S kümesinden eşit olasılıkla seçildiğini gösterir ve N ise kesilebilen (truncated) bir ayrık Gauss dağılımını temsil eder. Δ=⌊y/x⌋. Polinom modülünün derecesi (z), düz metin tam sayısı ve şifreli metin modülü asal sayısı sırasıyla z, y ve x ile gösterilir.

Gizli Anahtar:

Gizli anahtar K_s, R_2^z alkası üzerinden uniform olarak seçilir $K_s \leftarrow R_2^z$. ile gösterilir.

Açık Anahtar:

R_y^z halkasından uniform örnekleme, polinom a ile yapılır ve $a \leftarrow R_y^z$ ile gösterilir. Hata terimi eee ise ayrık Gauss dağılımı N ile örneklenir ve $e \leftarrow N$ olarak yazılır. Açık anahtarı oluşturan iki polinom $p0$ ve $p1$ aşağıdaki şekilde elde edilir:

$$(p0, p1) = ([-(e + S\,.\,a)]_y, a) \tag{4.1}$$

Şifreleme (Encryption):

Düz metin mesajı M bir Rx halkasında şifrelenirken, öncelikle R_2^z halkasından rastgele bir polinom u seçilir, yani $u \leftarrow R_2^z$. İki hata terimi $e1$ ve $e2$ ise ayrık Gauss dağılımı N ile örneklenir: $e1, e2 \leftarrow N$. Şifreli metin $(c0, c1)$ aşağıdaki formüllerle hesaplanır:

$$(c0, c1) = ([p0.u + \Delta.M + e1]_y, \lceil e2 + p1.u]_y) \tag{4.2}$$

Şifre Çözme (Decryption):

Gizli anahtar SK ile, şifreli metin $(c0, c1)$ şu formül kullanılarak çözümlenir ve mesaj M elde edilir:

$$Mesaj\ (M) = [[[c0 + c1\,.\,K_s]_y * \ (^x/_y)]].x \tag{4.3}$$

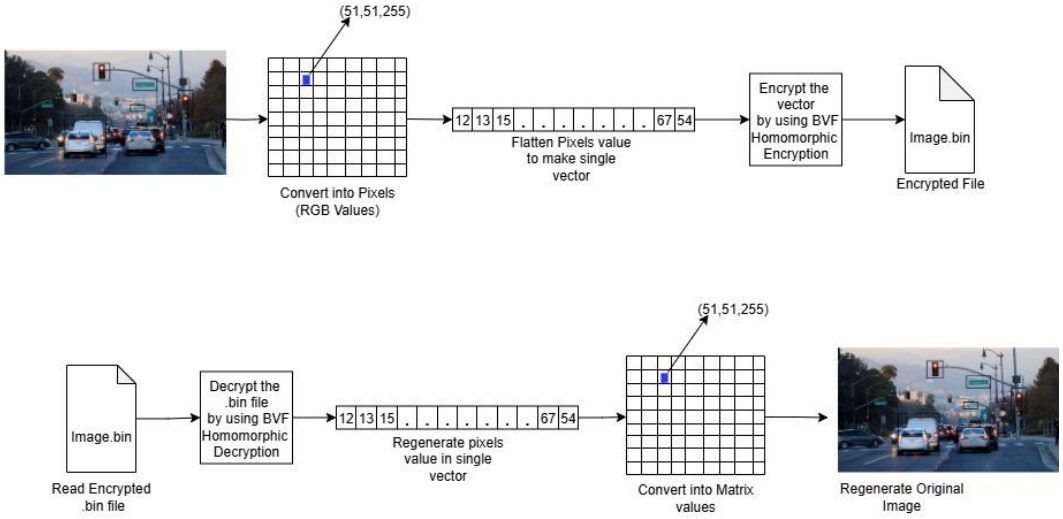
BFV şeması sayesinde, düz metin verisi açık anahtar kullanılarak şifreli metne dönüştürülür. Bu yöntem, yalnızca ilgili gizli anahtarın sahibi tarafından orijinal mesajın geri alınmasını ve çözülmesini garanti eder. Görseller, .bin dosyalarında saklanır ve BFV tekniği kullanılarak şifrelenir ve çözülür. BFV şifreleme ve şifre çözme süreci, öncelikle resimlerin piksel değerlerine dönüştürülmesiyle başlar.

Tablo 4.2. Şifreleme algoritması

Şifreleme algoritması: Secure_Image_Encoding(image_path, Kp, context)

Input: image_path → giriş görselinin dizini
context → TenSEAL BFV şifreleme bağlamı
Kp →açık şifreleme anahtarı
Output: cipher_vector → görselin şifrelenmiş temsili
Mat ← Read_Image(image_path)
Vec ← Convert_To_Vector(Mat)
CipherVec ← Apply_Encryption(Vec, context, Kp)
Write_To_File(CipherVec, "Encrypted_Image. bin")
return CipherVec

Her piksel, RGB değerleri üçlüsü (51, 51, 255) ile temsil edilir. 2D resim matrisi, piksel değerleri tek boyutlu bir vektöre dönüştürüldüğünde, aslında tek boyutlu bir piksel vektörüne indirgenmiş olur. Bu tek boyutlu vektör daha sonra açık anahtar ve BFV şifreleme tekniği kullanılarak şifrelenir.



Şekil 4.2. BFV kullanarak görselin şifrelenmesi ve çözülmesi

Bu işlem sonucunda oluşturulan şifreli vektör, Image.bin dosyası olarak kaydedilir. Şekil 4.2, şifreleme ve şifre çözme işlemlerinin akış diyagramlarını göstermektedir. encrypted.bin dosyası okunur; bu dosya 1D şifreli bir vektör içerir. Daha sonra gizli anahtar kullanılarak BFV algoritması ile çözülür.

Bu veriyi yalnızca güvenli gizli anahtara sahip yönetici çözebilir. Şifre çözme işleminin çıktısı, orijinal tek boyutlu piksel vektörüdür. Bu tek vektör daha sonra 2D RGB matrisine dönüştürülerek orijinal resim yeniden oluşturulur.

Tablo 4.3. Deşifreleme algoritması

Deşifreleme algoritması:
Secure_Image_Decoding(bin_file_path, Ks, context)
Input: bin_file_path → şifrelenmiş ikili dosyanın konumu context → TenSEAL BFV deşifreleme bağlamı Ks → gizli deşifreleme anahtarı **Output**: RestoredMat → yeniden oluşturulmuş görsel matrisi CipherVec ← Read_From_File(bin_file_path) PlainVec ← Apply_Decryption(CipherVec, context, Ks) PixelVec ← Restore_Pixel_Values(PlainVec) RestoredMat ← Convert_Vector_To_Matrix(PixelVec) Save_Image(RestoredMat, "Decrypted_Image. png") **return** RestoredMat

4.4. Şifreli Görsellerden Şifreli Görsel Arama

Yöneticilerin istedikleri veriyi arayabilmesi için üç farklı türde arama algoritması önerilmiştir:

4.4.1. Yüzdelik ve benzerlik tabanlı arama

Yönetici, şifrelenmiş bir görsel ile bulut sunucusuna bir istek göndererek ilişkili görselleri arayabilir. Arama görseli bulut sunucusuna gönderilmeden önce, yönetici BFV tekniğini kullanarak görseli şifreler ve .bin dosyası olarak kaydeder. Şifrelenmiş arama görseli, bulutta depolanan her şifreli görselden çıkarılır. Görsel indekslenirken, bu işlem şifreli çıkarma sonuçları (vektörler) oluşturur ve daha sonra yöneticiye gönderilir. Her görsel sonucu vektöründeki ardışık üç sıfır örnekleri, çıkarma sonuçları deşifre edildikten sonra sayılır. Bu sayı, her depolanan görsel ile arama görseli arasındaki benzerlik yüzdesini hesaplamak için kullanılır. Benzerlik yüzdesi seçildikten sonra yönetici bir istek oluşturur. Görseller filtrelendikten sonra bir zip dosyası hâline getirilir ve indirilmek üzere sunulur.

Arama işlemi için: Arama görseli $E_{img\_v(search)}$ şifrelenir ve bulut depolamaya gönderilir. Burada, bulut depolamada şifrelenmiş görseller $E_{img\_v(i-n)}$ üzerinde yüzdelik benzerlik tabanlı arama gerçekleştirilir:

$$E_{img\_v(i-n)} = Enc(Img\_V_{(i-n)}, C) \tag{4.4}$$

Burada C, anahtarları ve şifreleme parametrelerini içeren bağlamdır ve bulut depolamada her görselin şifrelenmesi için kullanılır. Arama görseli img_{search} $M \times N$

boyutunda bir vektör olarak temsil edilir. Görsellerin vektöre dönüştürülmesi şu şekilde yapılır:

$$img\_v_{img\_search} = reshape(img_{search}, [M \times N, 1]) \quad (4.5)$$

reshape fonksiyonu, 2D matrisi 1D vektöre düzleştirir. Arama vektörü düzleştirilip şifrelenir ve buluta yüklenir.

$$E_{img\_V_{i\_search}} = Enc(img\_v_{img\_search}, C) \quad (4.6)$$

Arama vektörü, tüm şifrelenmiş görsellerden çıkarılır ve $E_{img\_V(n\_R\_Vector)}$ içinde saklanır:

$$E_{img\_V(i\_R\_Vector)} = E_{img\_v(i-n)} - E_{img\_V_{i\_search}} \quad (4.7)$$

Her görseldeki benzerlik oranını belirleyin, $E_{img\_V(i\_R\_Vector)}$ değerini deşifre ederek ardışık üç sıfırları (T) doğrulayın ve sonucu gösterin:

$$img\_v_{i\_R\_Vector} = Dec(E_{img\_V(i\_R\_Vector)}, C) \quad (4.8)$$

Bir görselin yüzdelik benzerliğini hesaplamak için:

$$T_{Zero_{Count}} = \sum_{i=0,i+=3}^{n-2} 1\big(img\_v_{i\_R\_Vector}[\,i] = 0 \; and \; img\_v_{i\_R\_Vector}[\,i+1] = 0 \; and \; img\_v_{i\_R\_Vector}[\,i+2] = 0\,\big) \quad (4.9)$$

$$Percentile\_Similarity = \frac{T_{Zero_{Count}}}{Total\ pixels\ count\ in\ image\ (R,G,B)} \times 100 \quad (4.10)$$

BFV tekniğini kullanarak görseli şifreler ve .bin dosyası olarak kaydeder. Şifrelenmiş arama görseli, bulutta depolanan her şifreli görselden çıkarılır. Görsel indekslenirken, bu işlem şifreli çıkarma sonuçları (vektörler) oluşturur ve daha sonra yöneticiye gönderilir. Her görsel sonucu vektöründeki ardışık üç sıfır örnekleri, çıkarma sonuçları deşifre edildikten sonra sayılır. Bu sayı, her depolanan görsel ile arama görseli arasındaki benzerlik yüzdesini hesaplamak için kullanılır.

Tablo 4.4. Arama algoritması 1

Arama Algoritması 1: Find_Similar_Img(Arama_Gorseli, Sifreli_Veritabani[], Esik_Degeri)
Input: Arama_Gorseli → eşleşme için sorgu görseli $Sifreli_{Veritabani[\ ]}$ → şifrelenmiş görsellerin koleksiyonu $Esik_{Degeri}$ → minimum benzerlik yüzdesi **Output**: Archive.zip → eşleşen görsellerin sıkıştırılmış klasörü ResultVectors ← []

Tablo 4.4. (Devam) Arama algoritması 1

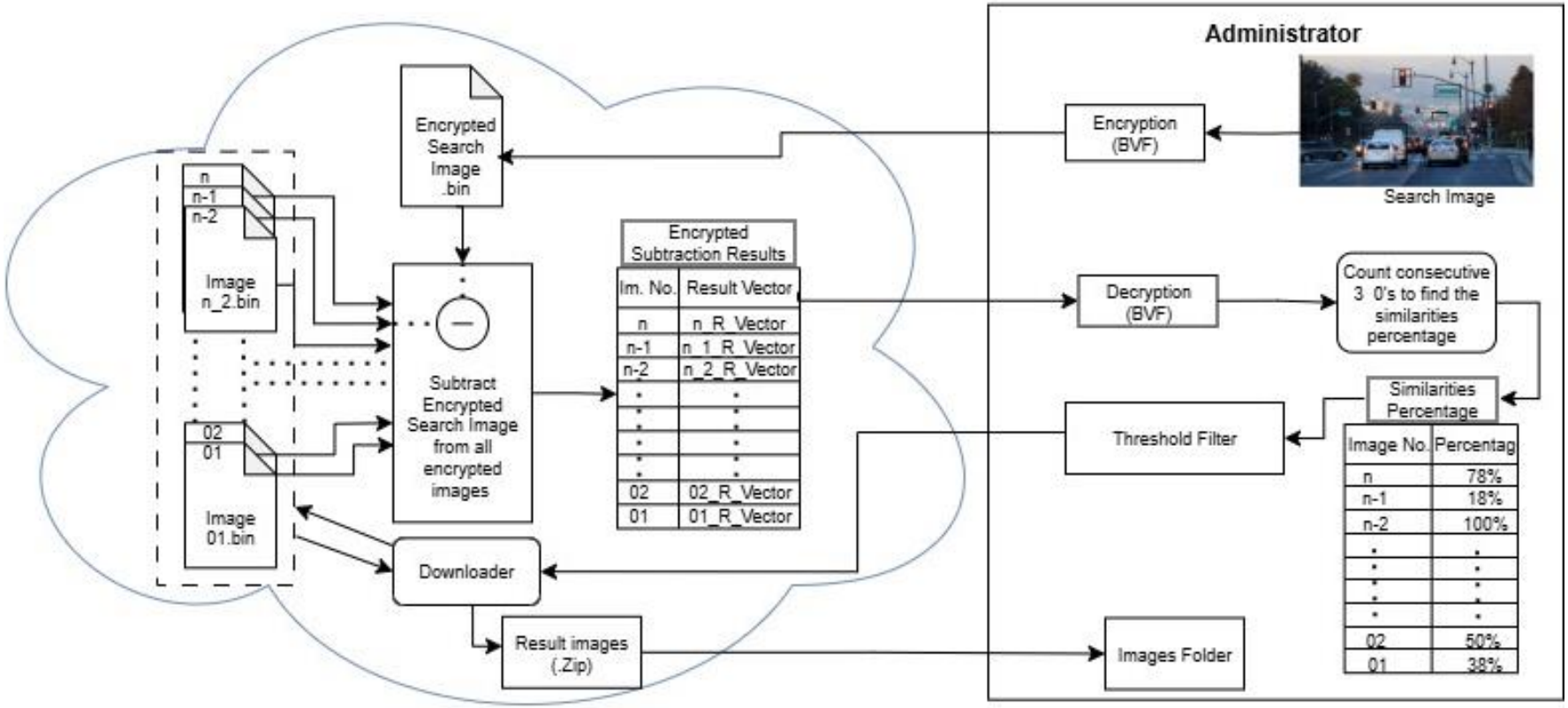
Arama Algoritması 1:
Find_Similar_Img(Arama_Gorseli, Sifreli_Veritabani[], Esik_Degeri)

```
context ← TenSEAL BFV şifreleme bağlamı
Kp ← açık anahtar
EncQuery ← Encrypt(Arama_Gorseli, context, Kp)
For each EncImg in Sifreli_Veritabani[] do
    DiffVec ← Subtract(EncImg, EncQuery)
    Append DiffVec to ResultVectors
DecodedVectors ← Decrypt(ResultVectors, Ks)
MatchedImages ← []
For each Vec in DecodedVectors do
    Score ← Count_Triplet_Zeros(Vec) / Length(Vec)
    If Score ≥ Esik_Degeri then
        Add corresponding image to MatchedImages
Compress MatchedImages into "Archive.zip"
                                return Archive.zip
```

Şekil 4.3, yüzdeye dayalı ve benzerlik tabanlı arama algoritmasının uygulama akışını göstermektedir.

4.4.2. Bilinmeyen nesnelere dayalı görüntü arama

Bu algoritma, YOLOv9c tarafından daha önce tanımlanmamış bir nesneyi bulmak için teorik bir yöntem önermektedir. BFV yalnızca aynı uzunluktaki şifreli vektörler üzerinde çıkarma yapılmasına izin verdiğinden, nesnenin boyutu görüntülerden farklı olabilir. İlk adım olarak nesnenin piksel RGB değerleri dönüştürülür. Nesneyi bulmak için, görüntüde herhangi bir konumda bulunabileceği varsayılarak nesne piksel değerlerinin ilk satırı çıkarılır. Kavramsal yaklaşım, nesnenin olabileceği her konum için şifreli bir arama deseni oluşturmaktır. Bulut sunucusu daha sonra bu şifreli desenlerin her birini hedef şifreli görüntüden homomorfik olarak çıkararak eşleşmeyi bulur. Bu brute-force yöntemi, homomorfik çıkarma yeteneklerini gösterse de, yüksek hesaplama ve veri yükü nedeniyle gerçek zamanlı uygulama için pratik değildir. Piksel değerleri düzleştirildikten sonra, isteği bulut sunucusuna gönderin ve tüm olası kombinasyonları aynı yöntemle şifreleyin. Şifreli görüntüler, her şifreli desen vektöründen çıkarılır. Bu yöntem, görüntüyü indeksleyerek şifreli çıkarma sonuçları üretir ve bunlar yöneticiye geri gönderilir. Şifre çözme sonrasında yönetici, ardışık sıfır sayısının desenin ilk satır piksel değerlerinin uzunluğuna eşit olup olmadığını kontrol ederek nesnenin konumunu belirler. Nesneyi doğrulamak için, yeni vektör başlatılır; uzunluğu görüntü vektörüne eşit olacak şekilde, nesnenin doğru piksel.



Şekil 4.3. Yüzdeye dayalı ve benzerlik tabanlı arama akış şeması

değerleri tespit edilen konuma yerleştirilir. Geri kalan değerler 266 ile değiştirilir çünkü çıkarma sonucu, nesne değerleri dışındaki diğer görüntü değerleri için sıfır olmayacaktır. Seçilen görüntüden aynı çıkarma işlemini gerçekleştirmek için, yeni vektör şifrelenir ve bulut sunucusuna geri gönderilir. Yönetici veriyi çözdükten ve bulut sunucusu sonuçları gönderdikten sonra, çıkarma sonucunu inceleyerek nesnenin doğrulanıp doğrulanmadığını belirler. Ardışık sıfır sayısı, nesne piksel değerlerinin sayısına eşitse nesne doğrulanmış ve görüntüde bulunmuş olur. Şekil 4.4'te, kapsamlı akış diyagramı gösterilmektedir. Görüntü ve nesne vektörleri, sırasıyla boyutları $M \times N$ ve $m \times n$ olan görüntü I ve nesne O ile gösterilir:

$$img\_v_i = reshape(I, [M \times N, 1]) \tag{4.11}$$

$$img\_v_o = reshape(O, [m \times n, 1]) \tag{4.12}$$

reshape fonksiyonu 2D matrisi 1D vektöre dönüştürür. Anahtarlar ve şifreleme parametrelerini içeren bağlam C, hem görüntü hem de nesne vektörlerini şifrelemek için kullanılır:

$$E_{v\_i} = Enc(img\_v_i, C) \tag{4.13}$$

$$E_{v\_o} = Enc(img\_v_0, C) \tag{4.14}$$

Nesne vektörünü görüntü vektörü boyutuna eşit hale getirmek için nesne vektörü döndürülür ve tekrarlanır. S pozisyonunda döndürülen vektör:

$$img\_v_{o\_rot}(S) = img\_v_{o\_rot}[-S:] + img\_v_{o\_rot}[:-S] \tag{4.15}$$

Tekrarlanan vektör deseni ile görüntü vektörünün uzunluğu eşleştirilir:

$$img\_V_{o,rep} = \left(img\_v_{o\_rot}(S) \times \left[\frac{M \times N}{m \times n}\right]\right)[:M \times N] \tag{4.16}$$

Desenler şifrelenir:

$$E_{img\_v_{o\_rep}} = Enc\left(img\_v_{o\_rep}, C\right) \tag{4.17}$$

Şifreli nesne desenlerini şifreli görüntüden çıkararak nesnenin konumu bulunur:

$$E_{results} = E_{v\_i} - E_{img\_v_{o\_rep}} \tag{4.18}$$

Ardışık sıfırlar incelenerek çıkarma sonucu vektör şifresi çözülür ve nesnenin konumu belirlenir:

$$img\_v_{result} = Dec(E_{results}, C) \tag{4.19}$$

Nesnenin tam konumunu içeren deseni belirledikten sonra, diğer tüm değerler 266 ile değiştirilir:

$$img\_v_{new\_obj\_img}[i] = \begin{cases} img\_v_o[i-j], & if\ j \le i < j + l\ and\ img\_v_{pt\_l}[i] = 0 \\ 266, & else \end{cases} \tag{4.20}$$

Yeni nesne görüntüsü şifrelenir:

$$E_{img\_v\_new\_obj\_image} = Enc\left(img\_v_{new\_obj\_img}, C\right) \tag{4.21}$$

$$E_{img\_v\_pt\_l} = Enc\left(img\_v_{pt\_l}, C\right) \tag{4.22}$$

Doğrulama yapılır:

$$E_{validate} = E_{img\_v\_pt\_l} - E_{img\_v\_new\_obj\_image} \tag{4.23}$$

$$V_{Validation\_result} = Dec(E_{validate}, C) \tag{4.24}$$

Tam nesneyi doğrulamak için desen:

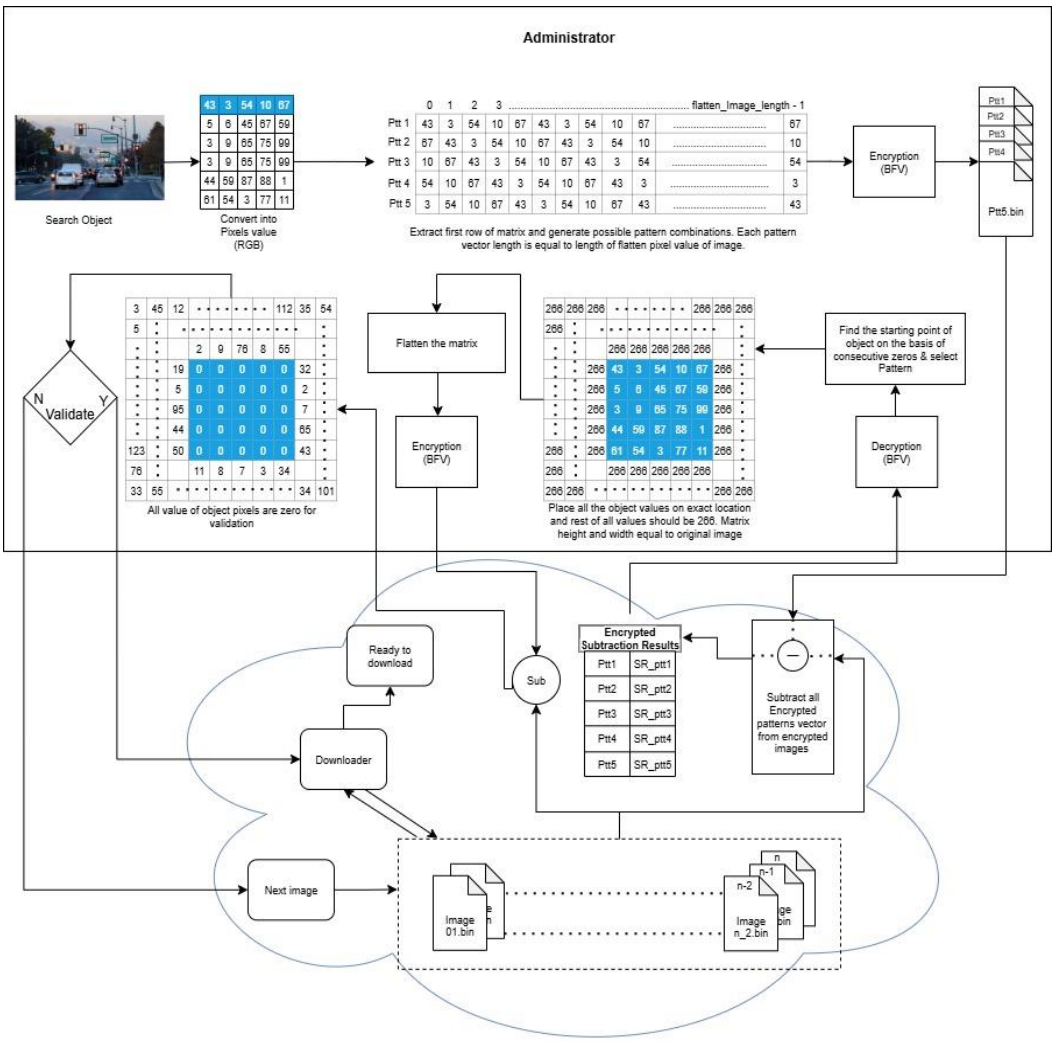
$$Validation\_Result = \begin{cases} T, & if\ count\left(V_{Validation\_result} = 0\right) = 3 \times m \times \mathrm{n} \\ F, & else \end{cases} \tag{4.25}$$

4.4.3. Bilinen nesnelere dayalı görüntü arama

Araç üzerinde yapılan uygulamada, YOLOv9c modelleri, kendi kendine giden araçlara takılı kameradan alınan görüntüyü işleyerek resimdeki nesneleri tespit eder. Daha sonra nesne ve görüntü şifrelenir ve bulut sunucuda, örneğin "görüntü x, bu y nesneleri içeriyor" şeklinde bir eşleme ile saklanır. Her ikisi de şifreli olduğundan, saldırgan bulut sunucusuna erişse bile bu eşlemeyi çözemeyecektir. Nesne tabanlı sorgu kullanılarak, yönetici ilgili görüntüleri arayabilir ve indirebilir. Yöntem, yönetici tek bir şifrelenmiş nesneyi bulut sunucusuna gönderdiğinde, resim arama algoritmasıyla aynı şekilde çalışır; ancak arama şifrelenmiş nesneler üzerinde yapılır.

Nesne ile eşleştirme yapıldıktan sonra, ilgili görüntüler ayrılır ve indirilmeye hazır hale gelir. Bu yaklaşım, tüm görüntüler üzerinde arama yapmaktan çok daha hızlı ve verimlidir. Küçük, şifrelenmiş nesne vektörleri üzerinde doğrudan arama yaparak sistem hızlı bir şekilde eşleşmeleri tespit edebilir. Önceden oluşturulmuş eşlemeyi kullanarak yönetici yalnızca tespit edilen nesnelere karşılık gelen belirli görüntüleri alabilir ve böylece indirilecek ve incelenecek veri miktarı büyük ölçüde azalır. Şekil

4.5'te, bu sürecin akış diyagramı gösterilmektedir. Bulut depolamada saklanan daha az ve bilinen şifrelenmiş nesneler olduğundan, işlem daha kısa sürer.



Şekil 4.4. Bilinmeyen nesnelere göre görüntü arama akış diyagramı

Tablo 4.5. Arama algoritması 2

Arama Algoritması 2:
Bilinmeyen_Nesne_Arama(Search_Object, Encrypted_Images[])

Input: Search_Object → tespit edilecek sorgu nesnesi
Encrypted_Images[] → şifrelenmiş görüntü dosyalarının koleksiyonu
Output: Result_Image. png → tespit edilen nesneyi vurgulayan yeniden oluşturulmuş görüntü
ResultVectors ← []
FRV ← Extract_First_Row(Search_Object[0 : len(FRV)])
PatternSet ← Generate_Possible_Patterns(No. of FRV)
context ← TenSEAL BFV şifreleme bağlamı
Kp ← genel şifreleme anahtarı
EncPatterns ← Encrypt(PatternSet, context, Kp)
For each EncPattern **in** EncPatterns **do**

Tablo 4.5. (Devam) Arama algoritması 2

Arama Algoritması 2:
Bilinmeyen_Nesne_Arama(Search_Object, Encrypted_Images[])

```
DiffVec ← Subtract(Encrypted_Images, EncPattern)
    Append DiffVec to ResultVectors
DecodedVectors ← Decrypt(ResultVectors, Ks)
Result_Image ← None
For each Vec in DecodedVectors do
    StartPos ← Find_Consecutive_Zeros(Vec)
    If StartPos exists then
        Map object pixels to Result_Image at StartPos
        Assign value 266 to remaining pixels
Enc_Result ← Encrypt(Convert_To_Vector(Result_Image), context, Kp)
FSV ← Subtract(Enc_Image_i, Enc_Result)
Dec_FSV ← Decrypt(FSV, Ks)
For i ← 0 to len(Dec_FSV) do
    If Count_Object_Pixels(Search_Object) =
                = Zero_Count(Dec_FSV[i]) then
                        Validation ← True
                return Result_Image. png
```

Bulut sunucuda şifrelenmiş nesne $E_{img\_v\_obj(i-n)}$, şifrelenmiş arama nesnesi $E_{img\_v(o\_search)}$'den çıkarılır:

$$E_{img\_v\_obj(i-n)} = Encrypt\left(img\_v_{obj_{(i-n)}}, C\right) \tag{4.26}$$

Arama nesnesi vektörleri $img\_v_{o\_search}$ ile O_v_{search} nesnesi $m \times n$ boyutunda gösterilir. Arama görüntülerinin vektöre dönüştürülmesi şu şekilde yapılır:

$$img\_v_{o\_search} = reshape(O\_v_{search}, [m \times n, 1]) \tag{4.27}$$

2D matris reshape fonksiyonu ile 1D vektöre dönüştürülür. Arama vektörü yeniden şekillendirilir, şifrelenir ve ilgili görüntüleri bulmak için buluta yüklenir:

$$E_{img\_V_{o\_search}} = Enc\left(img\_v_{o\_search}, C\right) \tag{4.28}$$

Bulutta saklanan $E_{img\_V\_obj(i-n)}$ ile şifrelenmiş arama nesnesi $E_{img\_V_{o\_search}}$ çıkarılarak $E_{img\_V\_obj(n\_R\_Vector)}$ elde edilir:

$$E_{img\_V\_obj(i\_R\_Vector)} = E_{img\_V\_obj(i-n)} - E_{img\_V_{o\_search}} \tag{4.29}$$

Ardından $E_{img\_V\_obj(i\_R\_Vector)}$ çözülür:

$$img\_v_{o\_R\_Vector} = Dec(E_{img\_V\_obj(i\_R\_Vector)}, C) \tag{4.30}$$

Üçlü sıfır sayısını belirlemek için:

$$T_{Zero_{Count}} = \sum_{i=0,i+=3}^{n-2} 1\big(img\_v_{o\_R\_Vector}[\,i] = 0 \; and \; img\_v_{o\_R\_Vector}[\,i+1] = 0 \; and \; img\_v_{o\_R\_Vector}[\,i+2] = 0\,\big) \quad (4.31)$$

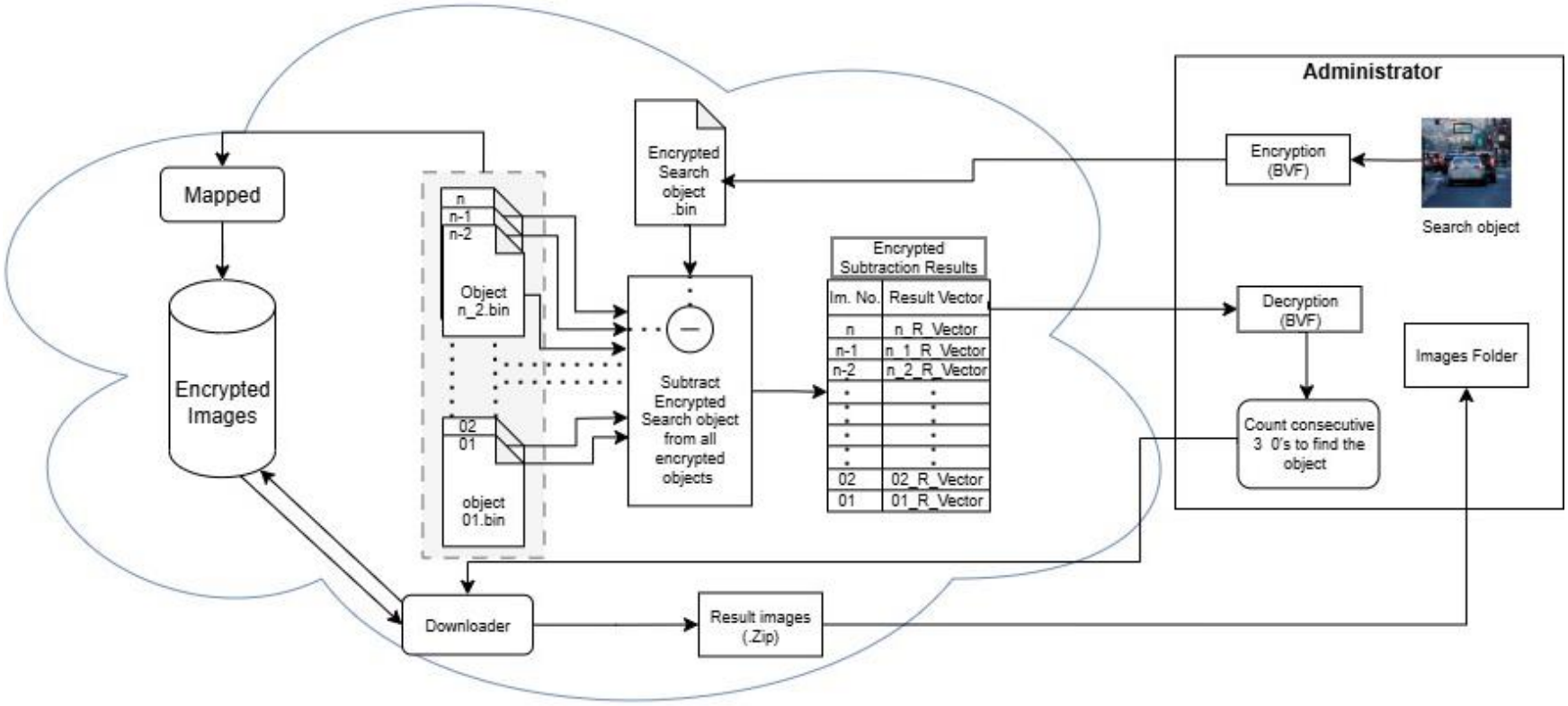
Tespit edilen nesne:

$$Detected\_Object = \begin{cases} T, & if\ T\_Zero\_Count = 3 \times m \times \mathrm{n} \\ F, & else \end{cases} \quad (4.32)$$

Tablo 4.6. Arama algoritması 3

Arama Algoritması 3:
Obj_Based_Search(Search_Obj, Encrypted_Images[], Enc_Obj[])

Input: Search_Object → sorgu objesi
Encrypted_Images[] → şifrelenmiş resim veri seti
Encrypted_Objects[] → şifrelenmiş aday obje veri seti
Output: FilteredImages.zip → eşleşen resimlerin arşivi
ResultVectors ← []
context ← TenSEAL BFV şifreleme bağlamı
Kp ← açık anahtar
EncQueryObj ← Encrypt(Search_Object, context, Kp)
For each EncObj **in** Encrypted_Objects[] **do**
DiffVec ← Subtract(EncObj, EncQueryObj)
Append DiffVec to ResultVectors
DecodedVectors ← Decrypt(ResultVectors, Ks)
For each Vec **in** DecodedVectors **do**
If Zero_Count(Vec) == Zero_Count(Search_Object) **then**
Mark Obj(Vec) as Selected
FilteredImages ← Map_Selected_Objects(Encrypted_Images, Selected_Objs)
return Compress(FilteredImages, "FilteredImages.zip")



Şekil 4.5. Bilinen objeler üzerinden resim arama akış diyagramı

5. DENEYSEL DÜZENEK VE UYGULAMA

Otonom araçlarda kullanılan şifreli görüntülerden güvenli ve akıllı bir nesne tespit sistemi geliştirmek ve bu görüntüler üzerinden arama yapmak için deneysel bir tasarım oluşturulmalıdır. Bu bölümün amacı, önceki bölümde tartışılan soyut çerçeveyi test edilebilir ve değerlendirilebilir bir modele dönüştürmektir. Otonom araçlar, sürekli olarak toplanan ve değerlendirilen sensör verilerine dayanır. Kameralar, LiDAR, radar ve GPS birimleri, güvenli bir navigasyon sağlamak için saniyenin kesirleri içinde işlenmesi gereken veri akışları üretir. Bu sensör okumaları ve görüntüler, yayalar, araçlar ve trafik işaretleri gibi nesneleri tanımlamak için nesne tanıma görevlerinin temelini oluşturur. Buna karşın, bu kesintisiz veri akışı güvenlik ve gizlilik endişelerini de beraberinde getirir. Saldırganlar, uygun koruma olmadan bilgilere erişebilir veya verileri değiştirebilir; bu durum hem güvenlik hem de gizlilik açısından risk oluşturur. Deneysel tasarımda, bu sorun iki tamamlayıcı teknoloji ile çözülmüştür. Birincisi, şifreli veriler üzerinde hesaplama yapmayı mümkün kılan homomorfik şifreleme (BFV) yöntemidir. Bu sayede hassas yol görüntüleri gizli tutulurken, bulutta analiz edilebilir. İkincisi, YOLOv9c ile uygulanmış derin öğrenmeye dayalı nesne tespitidir. Bu model nesneleri dinamik olarak tespit ve sınıflandırabilir, bu nedenle otonom sürüş ortamında kullanılabilir.

Deneyler, sistemin çeşitli yönlerini test etmeyi hedeflemektedir:

- **Güvenlik Etkinliği:** Şifreleme, verileri koruyarak yetkisiz erişimi engelleyebilir mi ve hassas bilgileri saklarken iletim sırasında güvenliği sağlayabilir mi?
- **Tespit Doğruluğu:** YOLOv9c ve önceki sürümleri, nesne tespitinde veya diğer tespit yöntemlerinde ne kadar başarılıdır?
- **Sistem Verimliliği:** Şifreleme, çıkarma, şifre çözme ve derin öğrenme hesaplamalarının zaman maliyeti nedir?
- **Ölçeklenebilirlik:** Birçok araç aynı anda şifreli veri akışlarını buluta yüklediğinde ne olur?

Bölüm, deneyler sırasında kullanılan donanım ve yazılım yapılandırmalarının açıklamasıyla başlar. Bu bilgiler kritik öneme sahiptir çünkü hem derin öğrenme hem de şifreleme kaynak tüketen yöntemlerdir. İşlemciler, grafik birimleri, bellek, işletim sistemleri ve yazılım çerçeveleri ayrıntılı olarak açıklanmıştır. Ardından eğitim ve test veri seti gösterilmektedir. Veri, farklı trafik koşulları, ışık durumları ve nesne tiplerini içeren geniş bir yol görüntüsü çeşitliliği ile temsil edilmektedir. Bunu takiben uygulama süreci açıklanır. Yöntemin tüm bileşenleri görüntü ön işleme ve şifreleme, alt vektör desenlerinin oluşturulması, çıkarma işlemi ve YOLOv9c tespiti pratik bir şekilde anlatılmıştır. İşlemlerin gerçek akışı algoritmalar ile gösterilmiş ve önerilen tasarıma dayalı pseudocode ile desteklenmiştir. Son olarak, performans ölçüm stratejileri sunulmuştur. Doğruluk, ortalama Ortalama Hassasiyet (mAP), şifreleme ve çözme süreleri ile depolama yükü gibi ölçümler, sistemin dengeli bir değerlendirmesini sağlamak için seçilmiştir. Bu ölçümler, bu bölümde açıklanan sonuçların güvenilirliğini garanti etmeye yardımcı olur.

5.1. Donanım ve Yazılım Kurulumu

Önerilen çerçeveyi uygulamak için üç farklı ortamda (sunucu tarafı, yönetici sistemi ve araç içi sistem) güçlü donanım ve kararlı yazılımlar gerekmekteydi. Her ortam, çalışmanın amacına hizmet edecek şekilde hazırlanmıştır. Sunucu, büyük veri hesaplamalarını ve depolamayı üstlenirken; yönetici sistemi, sistem seviyesini koordine etmekle; araç içi sistem ise uygulamada gerçek zamanlı aktiviteleri gerçekleştirmekle sorumludur.

5.1.1. Sunucu tarafı sistemi

Sunucu, yüksek performanslı hesaplama ve geniş depolama altyapısı sağlamaktadır. Kurulumun özellikleri şunlardır:

- **İşletim Sistemi:** Ubuntu 24.04 LTS (64-bit)
- **CPU:** AMD Ryzen 97950X3D, 32 çekirdek, soket başına 16 çekirdek, maksimum CPU frekansı 5759 MHz
- **Bellek:** 128 GB RAM
- **Depolama:** 1 TB SSD

Bu yapı, homomorfik çıkarma gibi kriptografik görevlerin sorunsuz uygulanmasına ve bulut tabanlı sistem alt bileşenlerinin yönetilmesine olanak tanır. Büyük bellek

kapasitesi, şifreli görüntü vektörlerini etkin bir şekilde işlemeye imkan verir; çünkü şifrelenmiş veriler, düz metne kıyasla hızla büyüyebilir.

5.1.2. Yönetici sistemi

Yönetici sistemi, bulut süreçlerini denetlemeli, nesne desenlerini şifrelemelerle kontrol etmeli ve güvenli iletişim sağlamalıdır. Yapılandırması şu şekildedir:

- **İşletim Sistemi:** Windows 10 (64-bit)
- **İşlemci:** Intel Xeon 4114 CPU, 2.20–2.30 GHz, 10 çekirdek/soket
- **Bellek:** 32 GB RAM
- **Depolama:** 2 TB SSD

Bu sistem, izleme ve hesaplama yetenekleri arasında bir denge sunar. Şifreleme bağlamlarının, anahtar dağıtımının ve nesne doğrulama işlemlerinin yönetilmesinde bir ara katman olarak görev yapar ve sonuçları araçlara iletmeden önce koordine eder.

5.1.3. Araç içi sistem

Araç içi sistem, hafif hesaplamaları gerçekleştirmek ve gerçek dünya sürüş ortamıyla etkileşimde bulunmak için tasarlanmıştır. Görevleri arasında sensör görüntülerinin şifrelenmesi, bulut sonuçlarının çözülmesi ve doğrulama yer alır. Sistem gereksinimleri şunlardır:

- **İşletim Sistemi:** Windows 10 (64-bit)
- **İşlemci:** Intel Core i5-4200U, 1.60–2.30 GHz, 2 çekirdek/soket
- **Bellek:** 8 GB RAM
- **Depolama:** 512 GB SSD

Araç içi yapı, sunucu ve yönetici sistemlerine kıyasla daha zayıf olsa da; şifreleme, şifre çözme ve karar verme süreçlerinde dinamik bir performans sunar. Gömülü araç sistemleri, hafif tasarımlarıyla kaynak kısıtlıdır.

5.2. Yazılım Çerçeveleri

Üç sistemin tamamı, uyumluluğu sağlamak amacıyla ortak bir yazılım ortamına dayandırılmıştır:

- **Programlama Dili:** Python 3.10

- **Kriptografi Çerçevesi:** BFV tabanlı
- **Homomorfik Şifreleme Kütüphanesi:** TenSEAL
- **Derin Öğrenme Mimarisi:** PyTorch, YOLOv9c çalıştırmak için
- **Görüntü İşleme Araçları:** OpenCV, NumPy
- **Görselleştirme Araçları:** Matplotlib

5.3. Veri Seti Açıklaması

Herhangi bir nesne tespit sisteminin performansı, eğitim ve test için kullanıldığı veri setinin kalitesine ve çeşitliliğine bağlıdır. Önerilen sistem durumunda, Kaggle üzerinde herkese açık bir veri seti seçilmiştir; çünkü bu veri seti, trafik ile ilgili geniş bir görüntü yelpazesi sunmaktadır. Veri setine dahil edilen nesneler, otonom araçlar için kritik öneme sahiptir: yayalar, araçlar, yol işaretleri ve trafik ışıkları. Böyle bir veri setine sahip olmak, sistemin gerçek sürüş koşullarında test edilebilmesini sağlamıştır. Veri seti, çeşitli yol ortamlarını ve farklı nesne türlerini içerir. Araçlar arasında otomobiller, otobüsler, motosikletler ve kamyonlar yer almakta olup, farklı pozisyonlarda hareketli ve duran araçlar bulunmaktadır. Yayalar ve bisikletliler, sokakta yürüyen yayalar, caddeyi geçen insanlar ve bisiklet süren kişilerden oluşmaktadır. Daha geniş bir genelleme sağlamak için çeşitli kıyafet stilleri ve vücut pozisyonları kullanılmıştır. Trafik işaretleri ve sinyaller arasında yön levhaları, hız sınırı tabelaları ve dur işaretleri yer almakta; kırmızı, sarı ve yeşil trafik ışıkları mevcuttur. Çevresel çeşitlilik açısından, aydınlık ve karanlık günlerde, yağmur ve sis gibi farklı hava koşullarında çekilmiş fotoğraflar; şehir, otoyol ve kırsal sahneleri içermektedir. Bu çeşitlilik, sistemin zorlayıcı koşullarda bile nesneleri etkili bir şekilde tanıyabilmesini sağlamıştır. Veri setinde toplam 22.241'den fazla etiketli görüntü bulunmaktadır. Her görüntüye bir bounding box eklenmiş ve hem nesnenin konumu hem de görüntüdeki nesnenin türü etiketlenmiştir.

YOLOv9c modeli, verilerin %70'i üzerinde eğitilmiştir. Eğitim sırasında performansı takip edebilmek için %15'i hiperparametreleri ayarlamak amacıyla kullanılmıştır. Kalan %15 ise tespit doğruluğunu ölçmek için en sona ayrılmıştır. Bu bölünme, modelin çeşitli örneklerle eğitilmesini ve gizli verilerle değerlendirilerek adil bir ölçüm yapılmasını garanti etmiştir. Veri setinin eğitimi ve şifrelemeden önce birkaç ön işleme adımı uygulanmıştır. Her bir resim, YOLOv9c tarafından kullanılan 640 ×

640 piksel boyutuna küçültülmüştür. Yeniden boyutlandırma, tutarlılığı garanti etmiş ve hesaplama yükünü azaltmıştır. Piksel değerleri, sinir ağını eğitmek için kullanılabilecek standart değerlere ölçeklenmiştir. Bu, model optimizasyonunda yakınsamayı artırmıştır. Bu işlem, yatay çevirme, rastgele döndürme, ölçekleme ve parlaklık değişiklikleri gibi tekniklerle gerçekleştirilmiştir. Bu önlem, eğitim setine çeşitlilik katmış ve aşırı öğrenme (overfitting) olasılığını en aza indirmiştir. Görseller işlendikten sonra piksel vektörlerine dönüştürülmüştür. Veriler, her pikselin bir tamsayı olarak temsil edildiği BFV şifreleme formatında bulunmaktaydı. Bu dönüşüm, önceki bölümde açıklanan şifreli işlem çıkarma ve doğrulama adımlarının uygulanmasını mümkün kılmıştır. Böylesine büyük bir veri setine rağmen bazı sınırlamalar belirlenmiştir. Nesnelerin kısmen görünür veya örtülü olduğu görüntüler tespit edilmesini zorlaştırmıştır. Nadir görülen trafik işaretleri ve bazı alışılmadık nesneler daha az bulunduğu için bu durum bazı doğruluk kayıplarına yol açabilir Şifreli metin genişlemesi (ciphertext expansion), büyük ölçekli şifreli hesaplamaların çok fazla depolama alanı tüketmesine sebep olmuştur. Buna rağmen, veri seti sistemin test edilmesi için mükemmel bir temel sağlamıştır. Önerilen çerçevenin uygulanmasında sistematik bir iş akışı izlenmiş ve homomorfik şifreleme ile derin öğrenmeye dayalı nesne tespiti birleştirilmiştir. Sistem, hem bilinen nesnelere dayalı aramalar hem de bilinmeyen nesnelere dayalı aramalar için geliştirilmiş; böylece otonom araçlarda kullanıldığında hem esneklik hem de güvenlik sağlanmıştır.

Veri seti, ham veya işlenmiş görüntüleri elde etmek ve bunları istenen boyuta ölçeklendirmek için kullanılmıştır. Görseller normalleştirilmiş ve çeşitliliği ve dayanıklılığı artırmak için artırılmıştır (augmentation). Her bir görüntünün pikselleri, piksel değerlerine dönüştürülmüş ve bir boyutlu vektörlere düzleştirilmiş, şifrelemeye hazır hâle getirilmiştir. TenSEAL kütüphanesi yardımıyla bir BFV şifreleme bağlamı (encryption context) oluşturulmuştur. Şifreleme ve şifre çözme, açık ve özel anahtarlar kullanılarak yapılmıştır. Görüntü vektörleri ve nesne vektörleri şifrelenerek şifreli metin hâline getirilmiştir. Buluta gönderilen tüm veriler şifrelenmiş şekilde iletilmiş, böylece gönderilen veriler orijinal görüntü olmamıştır. Şifrelenmiş nesne vektörleri doğrudan bulut veri tabanında saklanmış ve ilgili görüntülerle ilişkilendirilmiştir. Bilinmeyen nesneler durumunda, desenler nesne vektörünün tekrarlanması ve kaydırılmasıyla oluşturulmuş, vektörün boyutu tüm görüntü vektörü ile eşitlenmiştir. Bu desenler de şifrelenmiş ve güvenli şekilde buluta gönderilmiştir. Bulut tarafında,

şifreli görüntü vektörlerindeki nesne desenleri, şifreli nesne desenleri kullanılarak çıkarılmıştır. Homomorfik çıkarma işlemi, bulutun verilerin gerçek içeriğine (yani düz metin değerlerine) hiç maruz kalmadan işlemleri gerçekleştirebilmesini sağlamıştır.

Bilinen nesneler için çıkarma işlemi doğrudan saklanan nesne vektörleri üzerinde yapılmıştır. Bilinmeyen nesneler için, çıkarma adımı her oluşturulan desen ile tekrarlanmıştır. Çıkarma sonucu, şifreli modda yöneticiye veya araca geri iletilmiştir. Sonuçlar, özel anahtar kullanılarak çözülmüştür. Şifre çözülmüş vektörlerde ardışık sıfırlar kontrol edilerek doğrulama yapılmıştır. Nesne tespit edildiğinde, sıfır dizisinin uzunluğu nesne vektörünün uzunluğuna eşit olduğunda nesne doğrulanmıştır. YOLOv9c modeli, nesneleri tespit edip sınıflandırabilmek için eğitilmiş ve konuşlandırılmıştır. YOLO, gerçek zamanlı tespitlerde sınıf etiketleri ve bounding box'lar oluşturmuştur. Bu çıktılar, hem hassasiyeti hem de güvenilirliği artırmak için kriptografik çıkarma sonuçlarına eklenmiştir. Buluttaki şifrelenmiş eşlemelere doğrudan erişilerek nesne tespitleri öğrenilmiştir. Bilinmeyen nesne tespitlerini doğrulamak için çıkarma ve sıfır dizisi yöntemi kullanılmıştır. Kriptografik doğrulamalar ve YOLO tespitleri birleştirilerek, hız, doğruluk ve gizlilik arasında dengeli sonuçlar elde edilmiştir.

5.4. Algoritmalar

Bu çalışmada yöntemi sunan algoritmalar, veri işleme güvenliğini ve doğru nesne tanımayı sağlamak amacıyla uygulanmıştır. Tüm algoritmalar, çalışma sürecindeki adımları ele alacak şekilde uygulanmıştır; ilk algoritma görüntülerin şifrelenmesine, son algoritma ise kriptografik doğrulama ile YOLOv9c tespitinin entegrasyonuna karşılık gelmektedir. İlk algoritma, görüntülerin ön işleme ve şifrelenmesine ayrılmıştır. Veri setindeki ham görüntüler yeniden boyutlandırılmış, normalleştirilmiş ve tek boyutlu piksel vektörlerine dönüştürülmüştür. Bu, görsel bilgiyi kolayca şifrelenebilecek sayısal bir forma çevirmiştir. BFV şeması kullanılarak bir şifreleme bağlamı oluşturulmuş ve hem açık hem de özel anahtarlar üretilmiştir. Piksel vektörleri daha sonra açık anahtar kullanılarak şifrelenmiş ve oluşturulan şifreli metinler güvenli bir şekilde depolanmış veya taşınabilmiştir. Bu algoritma, hassas yol bilgilerinin boru hattının başında tamamen güvence altına alınmasını sağlamıştır.

İkinci algoritma, alt vektörlerin alınması ve şifrelenmesi ile ilgilenmiştir. İlgi alanları ve nesneler belirlenmiş ve vektörlere dönüştürülmüştür. Bu alt vektörler şifrelenmiş

ve bilinen nesneler olarak bulutta saklanmış ve ilgili görüntülere eşleştirilmiştir. Bu, sistemin sorgulandığında belirli şifreli nesneleri hızlı bir şekilde depolamasına ve geri çağırmasına olanak sağlamıştır. Süreç, nesne vektörünü ana görüntü vektörünün boyutuna uyacak şekilde çoğaltma ve ayarlama ile alt vektörün çeşitli desenlerini üretmeyi içermektedir; böylece bilinmeyen nesne görüntüye sığdırılmıştır. Oluşturulan tüm desenler şifrelenmiş ve gelecekte karşılaştırma yapmak için saklanmıştır. Bu iki yönlü strateji, çerçevenin hem sistemde zaten bilinen nesneleri hem de daha önce hiç görülmemiş nesneleri işleyebilmesini sağlamıştır. Üçüncü algoritma, bulutta homomorfik çıkarma işlemini gerçekleştirmiştir. Görüntü ve nesne vektörleri şifrelendikten sonra, sunucu şifreli alanda eleman bazlı çıkarma yapmıştır. Çıkarma işleminin çıktısı bu işlem sırasında şifreli kalmış, yani bulut ham verilere erişememiştir. Bilinen nesneler durumunda, çıkarma doğrudan şifrelenmiş sorgu nesnesi ile saklanan şifreli nesneler arasında yapılmıştır. Bilinmeyen nesneler için, her desen üretildiğinde çıkarma işlemi tekrar edilmiştir. Bu, nesne ile görüntü arasındaki olası eşleşmelerin göz önüne alındığından emin olmak için yapılmıştır. Dördüncü algoritma, şifre çözme ve doğrulamaya odaklanmıştır. Çıkarma sonuçları şifreli olarak yöneticiye veya araca geri gönderilmiştir. Sonuçlar daha sonra özel anahtar ile çözülmüş ve sistem çıktı vektörlerinde ardışık sıfır dizilerini aramıştır. Sıfır dizisi oluştuğunda, nesne vektörünün ana görüntüde başarıyla bulunduğu anlamına gelmektedir. Sıfır dizisinin uzunluğu nesne vektörünün boyutuna eşit olduğunda, tespit geçerli kabul edilmiştir. Bu, tespitleri kontrol etmek için matematiksel olarak sağlam bir yöntemdir ancak hassas görüntü verilerini açığa çıkarmaz. Son algoritma, kriptografi yaklaşımı ile YOLOv9c tabanlı nesne tanımayı birleştirmiştir. YOLOv9c veri üzerinde eğitilmiş ve nesne tespitinde gerçek zamanlı olarak uygulanmış; bounding box'lar ve sınıf etiketleri elde edilmiştir. Bu tespitler, çıkarma ve şifre çözme süreci sonuçları ile doğrulanmıştır. Eğer bir nesne YOLO tarafından tespit edilmişse, varlığı kriptografik doğrulama ile teyit edilerek yanlış pozitifler elimine edilmiştir. Çerçeve, her iki sistemin çıktısını birleştirerek doğruluk, hız ve gizlilik arasında bir denge sağlamıştır.

Bu algoritmalar bir araya gelerek, görüntü verilerini güvenli bir şekilde işleyen uçtan uca bir boru hattı oluşturmuştur; nesnelerin esnek tespitini sağlar ve kriptografik doğrulama ile mevcut derin öğrenmeyi birleştirir. Sistem, şifreli formda hem bilinen hem de bilinmeyen nesneleri etkili bir şekilde tespit edecek şekilde tasarlanmış;

YOLOv9c ise otonom sürüş uygulamasını gerçek zamanlı olarak desteklemek için doğruluk ve hız sağlamıştır.

5.5. Performans Ölçütleri

Önerilen gizliliği koruyan nesne tespit sistemini ölçmek için birkaç performans ölçütü seçilmiştir. Bu metrikler, sistemin hem kriptografik hem de yapay zeka kısmını kapsamaktadır. Bu, doğruluk, verimlilik ve güvenliği kapsayan dengeli bir değerlendirme sağlamayı amaçlamaktadır.

5.5.1. Doğruluk

Bir nesne tespit sisteminin en değerli ölçütlerinden biri doğruluktur. Doğruluk, doğru şekilde tanımlanan nesnelerin, temel gerçek (ground truth) etiketlerine oranı olarak tanımlanır. Bu çalışmada, doğruluk YOLOv9c tarafından yapılan tahminlerin referans etiketli veri seti ile karşılaştırılmasıyla değerlendirilmiştir. Doğrulukta görülen artış, sistemin farklı ortamlar altında araçları, yayaları ve yol işaretlerini tespit etmede etkili olduğunu göstermiştir.

5.5.2. Ortalama doğruluk oranı (mAP)

Nesne tespit modelleri genellikle ortalama doğruluk oranı (mean average precision, mAP) kullanılarak değerlendirilir. Bu metrik, hem doğruluğu (doğru tespit edilen nesne sayısı) hem de geri çağırmayı (ground truth nesnelerden tespit edilenlerin sayısı) dikkate alır.

Her nesne türü için precision-recall eğrisinin altındaki alan hesaplanmıştır. Tüm sınıfların ortalaması alınarak nihai mAP skoru elde edilmiştir. Bu, çeşitli modellerin makul karşılaştırmalarını sağlamış ve sistemin belirli bir nesne türüne yanlı olmamasını temin etmiştir.

5.5.3. Kesişim üzerine birleşim (IoU) ve güven skoru

YOLOv9c tarafından oluşturulan tüm bounding box'lara bir güven skoru atanmıştır; bu skor, nesnenin var olma olasılığını ifade eder.

Sadece önceden belirlenmiş eşik değerin (örneğin 0,5) üzerindeki skorlar tespit olarak kabul edilmiştir. IoU, tahmin edilen ve ground truth bounding box'lar arasındaki örtüşmeyi ölçmek için kullanılmıştır. IoU'nun 1 olması, daha doğru bir yerelleştirmeye karşılık gelir.

5.5.4. Şifreleme ve şifre çözme süresi

Sistem homomorfik şifrelemeyi kullanmaktadır; şifreleme ve şifre çözme hızı temel bir rol oynamaktadır. Şifreleme süresi, ön işleme tabi tutulmuş bir görüntüyü şifreli metne dönüştürmek için geçen süreyi ifade eder. Şifre çözme süresi ise bulut işlemlerinden sonra sonucu geri almak için geçen süreyi tahmin etmekte kullanılmıştır. Bu süreler boyunca farklı görüntü boyutları kaydedilmiş ve karşılaştırılmıştır; bu, ölçeklenebilirliği test etmek için yapılmıştır. Azalan süreler, otonom sürüş uygulamalarında daha yüksek performansın bir göstergesidir.

5.5.5. Homomorfik hesaplama süresi

Şifreleme ve şifre çözme işlemlerinin yanı sıra, bulutta homomorfik çıkarma süresi de ölçülmüştür. Çıkarma, operasyonel olarak ucuz bir işlem olmasına rağmen, binlerce piksel ve çok sayıda alt vektör deseni üzerinde binlerce kez tekrar edilmiştir. Hesaplama süresini kaydetme yeteneği, sistemin gerçek dünyadaki uygulanabilirliğini test etmeyi sağlamıştır. Bulgular, BFV şemasının nesnelerin neredeyse gerçek zamanlı doğrulanmasını mümkün kılabileceğini göstermiştir.

5.5.6. Depolama yükü

Homomorfik şifrelemenin sınırlaması, çok miktarda veriyi şifrelemesidir. Depolama yükü, şifrelenmiş görüntülerin boyutu ile düz metin (plaintext) görüntülerin boyutu karşılaştırılarak belirlenmiştir. Bu metrik, kaynak gereksinimleri ile gizlilik arasındaki dengeyi göstermektedir. Şifreli metinler daha fazla alan gerektirse de, bulutta sağlanan depolama alanı göz önüne alındığında yük hâlâ makul seviyededir.

5.5.7. Ölçeklenebilirlik ve sistem yükü

Ölçeklenebilirlik, çok sayıda görüntü kullanılarak test edilmiştir. Sistem yükü, bellek kullanımı ve işlem gecikmeleri açısından ölçülmüştür. Bu, sistemin birden fazla otonom aracın veri akışlarıyla aynı anda başa çıkarken göstereceği performans hakkında fikir vermiştir.

6. SONUÇLAR VE ANALİZ

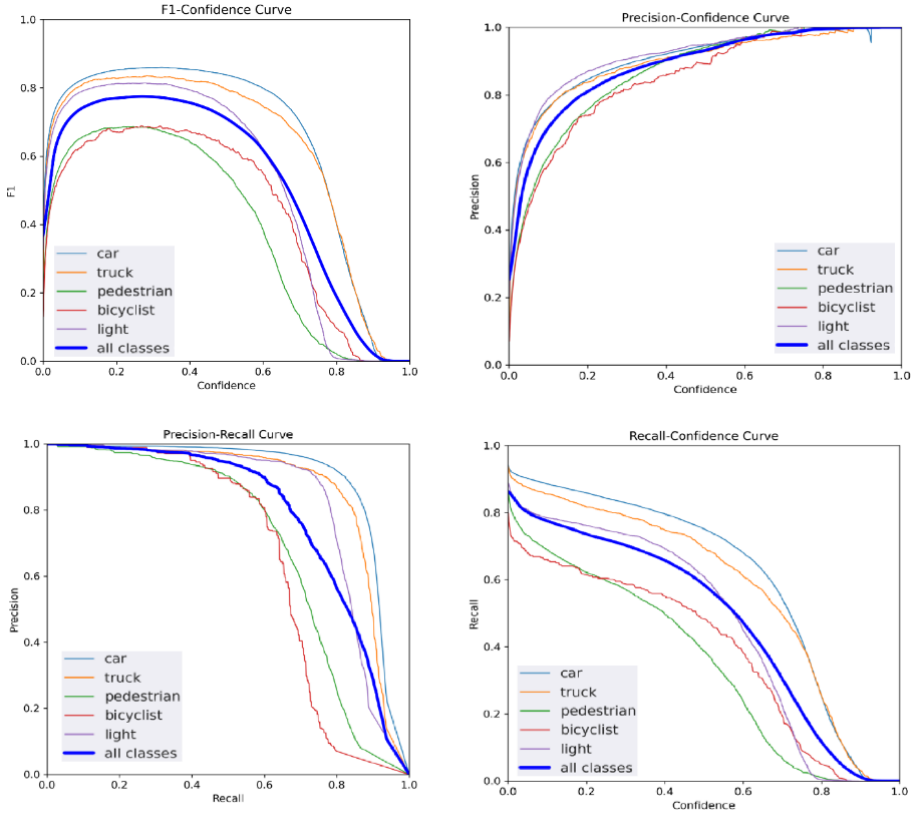
Bu bölüm, önerilen gizliliği koruyan nesne tespit sisteminin uygulanmasının bulgularını göstermektedir. Sonuçlar, hem kriptografik hem de derin öğrenme açısından doğruluk, verimlilik ve ölçeklenebilirlik temelinde değerlendirilmiştir. Sonuçlar, bilinen nesne tespiti, benzerliğe dayalı tespit, bilinmeyen nesne tespiti ve YOLOv9c performansı üzerinde tartışılmış ve bu noktada şifrelemeden kaynaklanan hesaplama yükü de değerlendirilmiştir. Ayrıca, önerilen modelin faydalı yönlerini ortaya koymak için mevcut çalışma yöntemleri ile karşılaştırılmıştır.

6.1. YOLOv9c Uygulama Analizi

YOLOv9c verilerinin performansı aşağıdaki tabloda sunulmuştur: YOLOv9c modeli, trafikle ilgili hazır resim seti kullanılarak eğitilmiş ve test edilmiştir. Model, araçları, yayaları, trafik ışıklarını ve yol işaretlerini tanımada mükemmel bir görsel performans göstermiştir. Distribution Focal Loss ve Complete IoU Loss gibi son derece gelişmiş yöntemler kullanılmış ve YOLO'nun önceki sürümlerine kıyasla daha yüksek doğruluk ve geri çağırma (recall) sağlamıştır. Model, çoğu nesne kategorisinde %90'dan fazla ortalama Doğruluk Oranı (mAP) elde etmiştir. Model, Kaggle üzerinde listelenmiş büyük bir veri seti üzerinde eğitilmiştir (Alin Cijov, n.d.). Bu veri seti, araçta bir webcam ile yakalanan 22241 görüntüden oluşmakta ve videodan kareler çıkarılmaktadır. Görüntüler farklı araçlar, trafik işaretleri, yayalar vb. gibi çeşitli nesnelerle etiketlenmiştir. YOLOv9c, sunucu tarafı sistemini eğitmek için kullanılmıştır. Modelin etkinliği, bir dizi görüntünün performans ölçütleri incelenerek test edilmiştir. Şekil 6.1'in gösterdiği gibi, bu veri setindeki görüntüler oldukça çeşitlidir. Tanınabilir birden fazla nesne içeren görüntüler olduğu gibi, aracın algılama menzilinin dışında kalan uzak nesneler veya tespit edilemeyen nesneler de mevcuttur. Modelin geri çağırma (recall) skoru 0,86, doğruluk (precision) skoru ise 0,948 olmuştur. Doğruluk ve geri çağırmanın birleşimi F1 skorunu yaklaşık 0,901 olarak üretmiştir. Bu sonuçlar, modelin doğru pozitiflerin önemli bir kısmını tespit edebildiğini ve yüksek doğruluk sağladığını göstermektedir. Bu durum, modelin çeşitli güven skoru eşiklerinde performansını gösteren Şekil 6.2'de grafiksel olarak da

gösterilmiştir. Precision-recall eğrisi ve F1 skoru, modelin nesne tespit doğruluğunun ayrıntılı bir değerlendirmesini sunmakta ve yanlış pozitifler ile yanlış negatifler arasındaki dengeyi temsil etmektedir.

Şekil 6.2.. YOLOv9c, bir görüntüde birden fazla nesneyi tespit etti.



Şekil 6.1. YOLOv9c sonuçlarının grafiksel gösterimi

Şifrelenmiş görüntülerin yüzdelik benzerliğe dayalı tespiti yeterli sonuçlar vermektedir. Görüntü arama algoritmasının detaylı bir analizi, görüntüler arasındaki benzerlik yüzdesi ve arama süresine özel dikkat gösterilerek veri seti boyutu arttıkça Tablo 6.1'de sunulmuştur. Arama süresi, görüntü sayısına oldukça bağımlıdır; tek bir görüntünün arama süresi 0,14 saniye iken, arama altındaki veri seti 22.241 görüntü olduğunda arama süresi yaklaşık 46 dakikadır. Benzerlik sonuçları dört aralıkta sınıflandırılmıştır: %0-30, %31-60, %61-80 ve %81-100. Daha küçük veri setlerinde, yakından eşleşen görüntülerin bulunma olasılığı daha yüksektir, çünkü arama sonuçlarının çoğu üst benzerlik aralığına (%81-100) düşmektedir.

Tablo 6.1. Yüzdelik ve benzerliğe dayalı görüntü arama sonuçları

Görüntü Sayısı	Bir Görüntü İçin Arama Süresi (sec)	Benzerlik Yüzdesi (Görüntü Sayısı)			
		0 to 30%	31 to 60%	61 to 80%	81 to 100%
1	0.14	0	0	0	1
10	1.14	0	4	3	3
100	14.04	22	59	5	14
1000	151	766	138	0	96
22241	2758	16536	3110	0	2595

Ancak, veri seti boyutu arttıkça dağılım oldukça farklılaşmaktadır; çoğu görüntü en düşük benzerlik aralığında (%0-30) yoğunlaşmaktadır. En büyük veri setinde, en düşük benzerlik aralığına düşen görüntü sayısı 16.536'dır. Buna karşılık, büyük bir veri setinde yüksek benzerlik puanı alan görüntü sayısı daha azdır. Bu tablo, veri seti boyutu ile görüntü arama algoritmalarının etkinliği ve doğruluğu arasındaki ilişkiyi vurgulamaktadır. Tablo, veri seti büyüdükçe yakından eşleşen görüntülerin bulunmasının zorlaştığını ve zaman alıcı hâle geldiğini, benzerlik yüzdesinin azaldığını göstermektedir. Yukarıdaki bilgiler, görüntü arama sistemlerinin ölçeklenmesiyle ilgili takasları anlamak için gereklidir.

6.2. Bilinmeyen Nesnelere Dayalı Görüntü Arama Analizi

Tablo 6.2, farklı boyutlardaki nesneleri bulmak için farklı büyüklükteki veri setlerinde yapılan arama sürelerini göstermektedir. Dört veri seti boyutu kullanılmıştır: 1, 10, 100 ve 1.000 görüntü. Nesne boyutları 664B ile 1,17KB arasında değişmektedir. Veri setinin boyutuna bakılmaksızın, nesne boyutu ve karmaşıklığı arttıkça arama süresi önemli ölçüde artmaktadır. Örneğin, tek bir görüntüde 664B boyutundaki bir nesneyi aramak yalnızca 8,29 saniye sürerken, aynı nesneyi 1.000 görüntüden oluşan bir veri

setinde aramak iki saatten fazla zaman almaktadır. En büyük nesne (1,17KB) ise 1.000 görüntüyü işlemek için 10 saat 29 dakika sürebilmektedir. Bu tablo, nesne boyutu, veri seti boyutu ve arama süresi arasındaki ilişkiyi açıkça göstermekte ve büyük ölçekli görüntü veri setlerinde nesne arama algoritmalarının hesaplama karmaşıklığını vurgulamaktadır.

Tablo 6.2. Bilinmeyen nesnelere dayalı görüntü arama sonuçları

Nesne Boyutu	Boyut (Dimension)	1 Görüntüden Arama Süresi (sec)	10 Görüntüden Arama Süresi (sec)	100 Görüntüden Arama Süresi (min)	1000 Görüntüden Arama Süresi (min)
664B	10X20	8.29	78	13.75	130
730B	20X40	16.07	110	26.89	252
683B	30X10	23.44	228	40.5	379
741B	40X20	31.68	307	53.70	445
1.17Kb	50X100	40.76	397	66	629

6.3. Bilinen Nesnelere Dayalı Görüntü Arama Analizi

Tablo 6.3, farklı boyutlardaki nesnelerin şifrelenmesi, çözülmesi ve aranması ile ilgili performans ölçütlerini sunmaktadır. Tablo, nesne sayısının sadece şifrelenmiş ve gerçek veri boyutunu değil, aynı zamanda nesnenin şifrelenmesi, çözülmesi ve aranması süresini nasıl etkilediğini göstermektedir.

Tablo 6.3. Bilinen nesnelere dayalı görüntü arama sonuçları

Nesne Sayısı	Gerçek Boyut (Kb)	Şifrelenmiş Boyut (MB)	Şifreleme Süresi (s)	Çözme Süresi (s)	Nesne Arama Süresi (s)
5	3.51	2.06	0.0654	0.075	0.23
10	9.48	4.53	0.154	0.201	0.57
20	45.3	17.3	1.308	0.943	1.27
30	87.4	31.3	1.401	1.027	2.08
40	122	40.4	1.98	1.868	2.64
50	171	63.5	3.87	2.934	4.06

Şifrelenmiş veri boyutu, 50 nesne durumunda maksimum 63,5 MB'a kadar artarken, gerçek veri boyutu toplam nesne sayısının artmasıyla (5'ten 50'ye) 3,51 KB ile 171 KB arasında değişmektedir. Nesne sayısı arttıkça şifreleme ve çözme işlemleri için gereken süre de artmaktadır. Örneğin, 5 nesneyi şifrelemek 0,0654 saniye sürerken, 50 nesneyi şifrelemek 3,87 saniye sürmektedir. Benzer şekilde, beş nesnenin çözülme süresi 0,075 saniye ile 2,934 saniye arasında değişmektedir. Daha fazla nesne olduğunda, nesneyi aramak için gereken süre de artmaktadır. 5 nesneyi aramak 0,23

saniye alırken, 50 nesneyi aramak 4,06 saniye almaktadır. Aşağıdaki tablo, şifrelenmiş nesne verileri bağlamında veri boyutu, işlem süresi ve arama etkinliği arasındaki dengeyi göstermeye yardımcı olur; çünkü sistemin performansı nesne sayısına bağlı olarak gözlemlenebilir.

6.4. BFV Şifrelemenin Görüntü Boyutu ve İşleme Süresine Etkisi

BFV homomorfik şifrelemenin dezavantajlarından biri, görüntü sayısı arttıkça işleme süresi ve depolama kapasitesinin de artmasıdır. Pratikte, homomorfik şifreleme göz önünde bulundurulmalıdır, çünkü şifrelenmiş veriler üzerinde hesaplama yapılmasına izin vererek önemli güvenlik avantajları sağlar; ancak özellikle büyük veri setlerinde performans maliyeti vardır. Tablo, görüntüler BFV şifreleme şeması kullanılarak şifrelendiğinde performans ölçümlerini göstermektedir. Bu şema, HE'nin (Homomorfik Şifreleme) önemli bir parçası olarak, hassas uygulamalarda verilerin gizliliğini korur ve veriler şifrelenmiş haldeyken bile işlem yapılmasına izin verir. Şifrelenmiş verilerin boyutu ve verilerin şifrelenip çözülmesi için geçen süre, görüntü sayısı arttıkça önemli ölçüde artmaktadır. Bu durumu göstermek için, BFV şeması kullanılarak tek bir 38,2 KB boyutundaki görüntü şifrelendiğinde, boyutu 21,8 MB'a kadar büyümüştür. Bu kadar büyük bir artış, HE'de yaygındır ve verilerin güvenliğini sağlamak için karmaşık matematiksel algoritmalar kullanılmasını gerektirir; böylece hesaplamanın verimliliği tehlikeye atılmaz.

Görüntü sayısı ayrıca şifreleme süresini de etkiler. Tek bir görüntünün şifrelenmesi yaklaşık 1,05 saniye sürerken, 22.241 görüntüden oluşan büyük bir veri setini şifrelemek yaklaşık 6,5 saat almaktadır. BFV şifreleme ayrıca önemli işlem gücü gerektirir; bu nedenle hesaplama gereksinimleri artmaktadır. Çözme işlemi genellikle şifrelemeden daha hızlıdır, ancak veri seti boyutu arttıkça çözme süresi de artar. Örneğin, bir görüntüyü çözmek yaklaşık 0,4052 saniye sürerken, 22.241 görüntüyü çözmek yaklaşık üç saat almaktadır. BFV şifreleme şeması çözme açısından tasarımla optimize edilmiş olsa da, şifreli veri boyutu arttıkça birçok kaynağı tüketir. Şifrelenmiş görüntüler çok fazla depolama alanı gerektirir, özellikle çok sayıda araç olduğunda, ve BFV şeması hesaplama açısından maliyetlidir. Bu problem, güvenlik, hesaplama ve depolama arasında bir denge kurularak çözülebilir. Veri depolama maliyetini azaltmak için, veriler zaten şifreli olduğu için güvenli olmayan, ucuz bulut tabanlı depolama hizmetleri kullanılabilir. Ayrıca, şifreli verilerin paketlenmesi ve paralellik, bir grup

pikselin hesaplanmasına izin vererek hesaplama maliyetini minimuma indirir (Kim ve ark, 2024). Dinamik ölçekleme, iş yükünü gerçek zamanlı ve dinamik olarak işlemek için kullanılır. Büyük miktarda şifreli veri, bulut tıkanıklığını önlemek için toplu yükleme sistemi ile verimli şekilde yüklenebilir.

Tablo 6.4. Görüntü boyutu ve işlem süresine etkisi

Görüntü Sayısı	Gerçek Boyut (MB)	Şifrelenmiş Görüntü Boyutu (MB)	Şifreleme Süresi (min)	Çözme Süresi(min)
1	0.037	21.8	0.0175	0.00675
10	0.38	218	0.1858	0.061
100	4.52	2303	2.12	0.6125
1000	43.3	23470.08	18.78	8.8
22241	952.32	512216	390	176

6.5. Performans Karşılaştırması

Mevcut şifreli görüntü işleme ve görüntü arama yöntemlerinin birkaç dezavantajı vardır. Güvenli ters indeksler gibi sabit indeks yapılarının kullanımı en büyük zayıflıklardan biridir; çünkü bu, aranabilecek sorguları sınırlayarak arama esnekliğini kısıtlar (Boucenna ve ark, 2019). Ayrıca, benzerlik tabanlı görüntü geri getirmede adaptasyon ve arama esnekliğindeki eksiklik, blok k-boyutlu ağaçlar (Bkd-trees) gibi indeksleme yöntemlerinden kaynaklanır (Tian ve ark, 2024). Bu yaklaşımlar, meta verilerin açığa çıkması nedeniyle çıkarım saldırılarına karşı savunmasızdır ve güvenlik saldırılarına karşı hassastır. Erişim desenleri veya arama sorgularının güvenliğini sağlayamayan yöntemlerin uygulanması, istem dışı olarak hassas bilgilerin yeniden keşfedilmesine yol açabilir. Bu nedenle, arama esnekliğini, geri getirme doğruluğunu ve güvenlik kontrollerini artıracak yeni ve sofistike yöntemlere ihtiyaç vardır. İndeks tabanlı şifreli arama ayrıca önceden tanımlanmış yapılar içerir ve arama faaliyetlerinin kapsamını sınırlar. Uygulamalarda kullanılan SHE ve PHE sistemleri, verilerin şifre çözülmeden önce üzerinde yapılabilecek işlem sayısını sınırlar. Bu dezavantajlar, şifreli arama ve nesne tespitinin otonom araç ortamlarında uygulanmasını kısıtlar; burada esnek, güvenli ve gerçek zamanlı arama teknikleri hayati öneme sahiptir.

Bu sınırlamaları ele almak için, önerilen şema BFV kullanır ve FHE ile şifrelenmiş veriler üzerinde sınırsız toplama ve çarpma işlemlerine izin verir; şifre çözmeye gerek yoktur. Bu özellik, benzerlik tabanlı arama, nesne tanıma ve güvenli görüntü geri kazanımı için daha dinamik ve esnek bir destek sağlar. Önceden tanımlanmış indeks

tabanlı geleneksel şifreli arama yöntemleriyle karşılaştırıldığında, önerilen çözüm; yüzde benzerlik, bilinmeyen nesnelerin tespiti ve bilinen nesnelerin tespiti açısından arama yapılmasına olanak tanır ve şifreli verilerin daha kapsamlı ve etkili bir şekilde analiz edilmesini sağlar. Ayrıca, FHE diğer homomorfik şifreleme yöntemlerine kıyasla daha yüksek güvenlik sunar. Tüm hesaplamalar şifreli veriler üzerinde yapılır ve arada şifre çözme yapılmadığı için veri sızdırma olasılığı minimuma indirilir. Bu durum, üçüncü taraf hizmet sağlayıcılarının şifreli verilere müdahale edebileceği ancak düz metne erişemeyeceği bulut ortamlarında özellikle faydalıdır. Önerilen teknik, FHE'nin güvenlik ve hesaplama avantajlarını kullanarak, akıllı ulaşım sistemlerinde güvenli görüntü aramasını sağlamak için gizliliği koruyan ve ölçeklenebilir bir arama yöntemi sunar ve geleneksel şifreli arama yaklaşımlarının sınırlamalarını aşar. Büyük görüntü verileri bir uygulamada kullanıldığında, şifrelemenin zaman yükü, özellikle şifreleme tekniklerinin etkinliği değerlendirilirken önemlidir.

Tablo 6.5. Önceki çalışmalar ile işlem süresi karşılaştırması

	(Rani ve ark, 2023)		(Mansoor ve ark, 2023)		Proposed Method	
Görüntü Boyutu	Şifreleme (sec)	Şifre Çözme (sec)	Şifreleme (Sec)	Şifre Çözme (sec)	Şifreleme (sec)	Şifre Çözme (sec)
256x256	2.34	3.66	1.42	0.8	0.18	0.09
512x512	9.51	14.95	4.8	3.37	0.96	0.47
1024x1024	38.5	60.13	-	-	3.97	1.71

Şifreleme ve şifre çözme süreleri, farklı şifreleme teknikleri ile ilişkili performans etkilerini tam olarak anlamak için analiz edilebilir. (Rani ve ark, 2023) gri tonlamalı ve renkli görüntülerin güvenliğini, diferansiyel şifreleme, kaotik eşleme ve sihirli karelerin birleşimiyle artıran yeni bir şifreleme şeması sunmuştur. Buna karşılık, (Mansoor ve ark, 2023) kaotik bir sistemin orijinal koşullarını ve kontrol parametrelerini ortalama ve varyans gibi istatistiksel parametrelerin yardımıyla değiştiren hibrit adaptif görüntü şifreleme (HAIE) şeması önermiştir. Bu çalışmada önerilen yaklaşım, BFV olarak adlandırılan homomorfik şifrelemeyi uygular ve şifreli veriler üzerinde şifre çözmeden hesaplamalar yapılmasına olanak tanır. Bu özellik, mevcut bilgilerin gerçek niteliğini açığa çıkarmadan hassas verilerin işlenmesini gerektiren uygulamalarda özellikle önemlidir. Tablo 6.5, (Rani ve ark, 2023), (Mansoor ve ark, 2023) ve önerilen yöntemin farklı görüntü boyutlarındaki şifreleme ve şifre çözme performansını göstermekte ve önerilen homomorfik şifreleme

tekniğinin performans avantajlarını ortaya koymaktadır. Tüm uygulamalar, Intel Core i5-7500 işlemci hızı 3.40 GHz ve 8 GB RAM'e sahip Windows 10 PC üzerinde gerçekleştirilmiştir. Tablo gösterdiği üzere, üç yaklaşımın şifreleme ve şifre çözme hızları büyük ölçüde farklıdır. Diğer yöntemlerle karşılaştırıldığında, önerilen işlem, daha küçük görüntü boyutlarında (256x256) çok daha hızlı bir şifreleme (0.18 saniye) ve şifre çözme (0.09 saniye) süresi sunmaktadır. Daha büyük görüntülerde (512x512 ve 1024x1024) önerilen yaklaşım, hem şifreleme hem de şifre çözme sürelerinde her zaman üstün bir performans faktörü ile karakterize edilmektedir. Bu sonuçlar, BFV homomorfik şifreleme yönteminin yüksek düzeyde güvenlik ve daha yüksek verimlilik sağladığını ve gerçek dünyada görüntü şifrelemesi için uygun bir yöntem olduğunu göstermektedir. Verimlilik ve gizlilik sağlanması gereken durumlarda, önerilen yaklaşım belirgin bir avantaj sunar; çünkü şifreli veriler üzerinde hesaplamaların yapılmasına olanak tanır. (Abdulsada ve ark, 2019b) tarafından kullanılan şema Paillier şemasıdır (Mohammed ve ark, 2023), Damgard, Geisler ve Kroigaard (DGK) tarafından kullanılan şema ise semantik olarak güvenli, toplamsal homomorfik şifreleme şemalarıdır (Kuo ve ark, 2023). Görüntülerin benzerliklerini belirlemek için global hiyerarşik renk histogramları arasındaki mesafeyi hesaplamak amacıyla Öklidyen mesafe tekniği uygulanmıştır. Bu yöntemin çıktısı, 4GB RAM'e sahip Intel i5 2.2 GHz işlemcide çalıştırılmıştır. Benzer şekilde, (Iqbal ve ark, 2022) telemedisinde BGV homomorfik şifrelemeyi tanıtmış ve şifreli bulut tabanlı depolama verilerinde hedef bilgiyi almak için arama yöntemini eğitmiştir. Ses örnekleri kullanılmış ve aşağıdaki sistem özelliklerinde test edilmiştir: OS Ubuntu 18.04.5 LTS (64 bit) ile 16GB RAM, Intel Core i7-7700 CPU 3.6GHz x 8, ve 1TB SSD disk.

Tablo 6.6, görüntü arama sürelerinin karşılaştırmasını sunmaktadır. Kriptografik işlemler genellikle anahtarları, şifrelenmiş verileri ve verilerin güvenli işlenmesini sağlamak için ek meta verileri saklama gereği nedeniyle bellek yükü oluşturabilir. Daha sofistike şifreleme algoritmaları ve daha büyük veri setleri, bu yükü artırır, sistem performansını etkiler ve büyük depolama, özellikle bulut depolama gerektirir. Bu durum, özellikle gerçek zamanlı işleme ve kaynak kısıtlı cihazlar için kaynak taleplerinde zorluk yaratabilir; çünkü karmaşık şifreleme yöntemleri daha yüksek bellek ve işlem kapasitesi talep edebilir. Veri miktarı ve şifreleme algoritmalarının karmaşıklığı arttıkça, sistemin verimliliği ile gizliliğin korunması arasındaki ilişki daha belirgin hale gelir.

Tablo 6.6. Önceki çalışmalar ile arama karşılaştırması

Dosya Sayısı	(Abdulsada ve ark, 2019a)(Sec)	(Iqbal ve ark, 2022) (sec)	Önerilen Yöntem (sec)
100	72.7	775	14.04
200	-	1160	27.75
600	395.4	-	83.65
1000	695.6	-	151

Önerilen yaklaşım, BFV ile birleşmiş FHE'nin, nesne algılama yeteneklerini etkilemeden veri güvenliğini sağlayabileceğini ortaya koymaktadır; ancak şifreleme yükü, önerilen yaklaşımın performansı üzerinde bazı etkiler yapmaktadır. YOLOv9c ile üç diğer arama algoritmasının kombinasyonu, yüksek doğruluk sağlar ve veri gizliliğini etkilemez. Aşırı hava koşulları ve aydınlatma değişiklikleri ile kamera kalitesi, benzerlik tabanlı algılama yöntemini doğrudan etkileyebilir; bu durum, sistemin verimliliğini artıracak önlemler alınarak giderilebilir. Ayrıca, şifreli verilerin merkezi sunuculardan taşınarak veri kaynağına daha yakın noktalarda işlenmesi ile işlem yüklerinin azaltılması sağlanabilir; bu sayede edge ve fog computing entegrasyonu, büyük miktardaki şifreli veri ile ilişkili yükü hafifletebilir. Gizliliği etkilemeden şifreli nesne algılama verimliliğini optimize edecek teknikler araştırılabilir (örneğin, görüntü sıkıştırma ve paralel işlem).

7. TARTIŞMA

Günümüzde akıllı ulaşım sistemlerinde en güncel sorunlardan biri, önerilen sistemin çözmeyi amaçladığı, otomatik araçlarda nesnelerin nasıl tanımlanıp doğrulanacağı ve aynı zamanda gizliliğin nasıl korunacağıdır. Geleneksel nesne algılama yöntemleri, ham görüntülerin doğrudan iletimine dayanır; bu nedenle değerli bilgiler yetkisiz kullanım ve kötüye kullanıma karşı savunmasız hale gelir. Buna karşılık, bu çalışma, homomorfik şifreleme ve gelişmiş derin öğrenme algoritmalarını, yani YOLOv9c'yi birleştirerek gizliliği koruyan nesne algılama gerçekleştirmiştir. Bu bölüm, bulguların daha geniş etkilerini tartışmakta, önerilen çerçevenin güçlü ve zayıf yönlerini değerlendirmekte ve bunun otonom araçlar, bulut bilişim ve IoT'nin geleceği açısından önemini ele almaktadır. Sistemin en büyük avantajlarından biri, güvenlik ve performansı birleştirmesidir. YOLOv9c'nin inanılmaz derecede doğru olduğu (ortalama Ortalama Hassasiyet, mAP %90'ın üzerinde) ve homomorfik şifreleme sisteminin sürecin veri gizliliğini korumak için kullanıldığı bulunmuştur. Hem güvenlik hem de doğruluk avantajı mevcut çalışmalarda yaygın değildir. Mevcut sistemlerin çoğu, algılama doğruluğuna çok duyarlıdır, ancak gizliliği ihmal eder; çünkü şifreleme odaklı sistemler performansa çok duyarlıdır. Önerilen bu çerçeve, her iki gerekliliği de dengeler ve bu nedenle pratikte uygundur. Diğer bir güçlü yön, çift nesne algılama yolları sayesinde esneklik sunmasıdır. İyi bilinen nesneler için sistem, bulutta önceden şifrelenmiş desenleri hızlı bir şekilde belirleyebilir ve böylece bunları çok hızlı bir şekilde doğrulayabilir. Bilinmeyen nesneler durumunda, desen kaydırma yöntemi sistemi yeni girişleri öğrenmeye olanak tanır; böylece daha önce bilgisi olmayan yeni nesneler de algılanabilir. Bu uyarlanabilirlik, otonom araçların yolda karşılaşılabilecek öngörülemeyen durumlara anında tepki vermesi gereken otonom sürüşte özellikle faydalıdır.

Şifreli veriler, araçtaki yerel sistemi zorlamadan bulut tabanlı tasarım aracılığıyla işlenip depolanabilir. Güvenlik açısından BFV şeması çok önemli bir adımdır. Bu şemada homomorfik çıkarma mümkün hale gelir ve böylece şifrelenmiş veriler üzerinde hesaplamalar yapılabilir. BFV şeması ayrıca, RSA veya Paillier gibi

geleneksel şifreleme şemalarının aksine, işleme başlamadan önce görüntülerin şifresini çözmeyi gerektirmez; böylece veriler açığa çıkmadan hesaplama yapılabilir. Sisteme yönelik özel bulut ve IoT tehditleri de ele alınmıştır. Şifreli veriler, düşman bulut altyapısına erişse bile özel anahtar kullanmadan okunamaz. Bu tür bir gizlilik, bir yüz, plaka veya konum gibi hassas kişisel veriler içerebilecek fotoğrafların bulunduğu otonom araçlarda son derece önemlidir.

Yapay zeka (YZ) ve homomorfik şifrelemenin birleşimi, bu çalışmanın ayırt edici bir özelliğidir. YOLOv9c tarafından yapılan nesne algılama hızlı ve hassas olsa da, şifrelemeye dayalı doğrulama verilerin güvenilir ve gizliliği koruyan bir şekilde işlenmesini sağlamıştır. Bu tür bir entegrasyon, gelecekte YZ çalışmalarının büyük potansiyel taşıdığını göstermektedir. YZ sistemlerinin yüksek doğruluk göstermesi tek başına yeterli olmayacaktır; veri gizliliği konusu giderek daha kritik hale gelmektedir. Gizlilik, kullanıcıya duyarlı olmalıdır. YZ ve homomorfik şifreleme birlikte, ulaşımın ötesinde sağlık, gözetim ve akıllı şehirler gibi alanlarda da kullanılabilecek, gizlilik odaklı yeni bir akıllı sistem sınıfı oluşturmaktadır. Önerilen çerçevenin güçlü yönleri olsa da, sınırlamaları da vardır. En büyük zorluk hesaplama yüküdür. Şifreleme ve şifre çözme, düz metin verilerini işlemeye kıyasla gecikme ekler ve bilinmeyen nesneler çeşitli desenlerde üretilmek zorunda olduğunda ek gecikmeler ortaya çıkar. Sistem makul bir performans sergilese de, yüksek hızlı otonom sürüşte gerçek zamanlı gereksinimler daha da optimize edilebilir. Diğer bir sınırlama ise depolama yüküdür. Düz metin görüntüler, şifreli versiyonlarına kıyasla çok daha küçüktür, bu da bulut depolamada bir yük oluşturur. Bulut teknolojisi büyük miktarda bilgi işleyebilse de, milyonlarca araç için ölçeklendirme mali ve organizasyonel sorunlar doğurabilir. Gelecekteki çalışmalarda, şifreleme tekniklerinin optimizasyonu ve şifreli verilerin boyutunun güvenliği etkilemeden sıkıştırılması düşünülmelidir. Üçüncü bir kısıtlama, bulut altyapısına bağımlılıktır. Bulut dış kaynak kullanımı ölçeklenebilirlik sunsa da, gecikme ve bağlantı riskleri gibi zorluklar da beraberinde gelir. Araçlar, düşük ağ kapsama alanına sahip bölgelerde çalışırken bulut işlem kullanımı algılama performansını etkileyebilir. Bu sorun, yerel kenar (edge) işlemleri ile bulut tabanlı hesaplamayı birleştiren hibrit çözümlerle aşılabilir. BFV tabanlı yöntem, gizlilik ve hesaplama bağlamında RSA, ECC ve Paillier gibi standart şifreleme yöntemlerinden açıkça daha işlevseldi. Önceki yöntemler gizlilik sağlasa da, verilerin şifre çözülüp analiz edilmesi gerektiğinden, hesaplama sürecinde veriler manipülasyona açık hale

geliyordu. Buna karşılık, BFV, şifreli veriler üzerinde homomorfik işlemlere izin vererek, süreç boyunca güvenliği korudu. YZ açısından, YOLOv9c, önceki sürümler (YOLOv3 ve YOLOv4) ile kıyaslandığında çok daha doğru, yüksek geri çağırma (recall) değerine sahip ve daha hızlı algılama yapabilmektedir. Faster R-CNN gibi diğer nesne algılama yaklaşımları da doğru sonuç verir, ancak gerçek zamanlı performans sağlayamazlar. YOLOv9c, otonom araç ortamları için en uygun dengeyi sunmaktadır. Bu yaklaşım, şifreleme ile birlikte, sisteme önceki çalışmalarda görülmeyen bir güvenlik ve verimlilik sağladı. Bu durum, otonom araçların geleceği için önemli etkiler taşımaktadır. Gizliliği koruyan nesne algılama, araçların verilerini bulut sunucusuna göndermesine ve bu verilerin analiz edilmesine olanak tanırken, hassas bilgilerin açığa çıkmasını engeller. Böyle bir yetenek, insanların otonom sürüş sistemlerine güvenmesini sağlamak için kritik öneme sahiptir. İnsanlar, konum geçmişleri veya yolcu kimlikleri gibi kişisel bilgilerinin izlenmediğinden emin olduklarında, bu tür teknolojileri kullanma olasılıkları artar.

Ayrıca, hem bilinen hem de bilinmeyen nesneleri tanıyabilme yeteneği, sürücüsüz araçların güvenliğini artırır. Yollar yalnızca dinamik ve öngörülemez değildir, aynı zamanda araçların sadece trafik ışıkları ve araçlar gibi yaygın nesneleri değil, nadir veya sıradışı nesneleri de tanımasını gerektirir. İki yönlü algılama yaklaşımı, araçları bu tür durumlarda esnek kılar ve böylece yol güvenliğini artırır. Bu sonuçlar, otonom araçların ötesinde, çok sayıda Nesnelerin İnterneti (IoT) sistemine de genişletilebilir. Birçok IoT cihazı hassas bilgileri toplamak için kullanılır ve bu verilerin bulut ortamında analiz edilmesi gerekir. Bu veriler homomorfik şifreleme ile şifrelenebilir ve anlamlı hesaplamalar yapılabilir. Yapay zeka ve şifrelemenin birleştirilmesi, akıllı şehir altyapılarında, sağlık izleme cihazlarında ve endüstriyel IoT ağlarında gizlilik odaklı analizlerin kapısını açabilir. Gelecekte yapılabilecek araştırmalar için birkaç alan vardır. Potansiyel bir yön, hesaplama maliyetlerini düşürmek amacıyla GPU veya FPGA kartı kullanılarak homomorfik şifrelemenin donanım hızlandırmalı modellerle uygulanmasıdır. Diğer bir yön, kuantum bilgisayarların klasik şifrelemeye tehdit oluşturması durumunda gerekli olacak kuantuma dayanıklı şifreleme şemalarının entegrasyonudur. Yapay zeka ayrıca federated learning (federatif öğrenme) yaklaşımında kullanılabilir ve gizliliği daha da artırabilir. Şifreli bilgilerin buluta gönderilmesi yerine, modellerin yerel olarak araçta eğitilmesi ve yalnızca toplu güncellemelerin buluta iletilmesi sağlanabilir. Bu yaklaşım, hem şifreleme hem de

merkezi olmayan öğrenmenin avantajını sunar ve ek yük ile gizlilik risklerini en aza indirir.

8. SONUÇ VE GELECEK ÇALIŞMALAR

Bu çalışma, sürücüsüz araçlarda gizliliği koruyan yeni bir görüntü algılama modeli önermiştir. BFV şeması ile tam homomorfik şifreleme (FHE) kullanılarak, şifrelenmiş görüntü bilgilerinin güvenli bir şekilde işlenebileceği ve bu süreçte tespit yeteneklerinden ödün verilmediği gösterilmiştir. Sistem, bulut altyapısının hesaplama gücünden faydalanırken, otonom araçların topladığı hassas görsel veriler işleme boyunca gizli tutulmuştur. Çözüm, üç farklı görüntü algılama yöntemini kullanmıştır: yüzdelik benzerlik tabanlı algılama, bilinen nesne tabanlı algılama ve bilinmeyen nesne tabanlı algılama. Her iki yaklaşım da çerçeveye farklı katkılar sağlamış ve çeşitli algılama durumlarının yönetilmesinde esneklik sunmuştur. Yüzdelik benzerlik yöntemi, şifrelenmiş görüntülerin benzerlik düzeyini tahmin ederek karşılaştırılmasına olanak tanımış ve ilişkili veya yakın kopya görüntülerin bulunmasında yardımcı olmuştur. Nesne tabanlı yaklaşım ise, bulutta önceden şifrelenmiş nesneleri hızlı ve verimli bir şekilde bulmayı sağlayan pratik bir sistem sunmuştur. Gerçek dünya sürüş koşullarının öngörülemezliğiyle başa çıkmak için bilinmeyen nesne tabanlı yaklaşım, orijinal veri setinde bulunmayan nesneleri de algılayabilme imkânı sağlayarak sistemin esnekliğini artırmıştır. Bu kriptografik araçlara ek olarak, yüksek doğruluk elde etmek amacıyla YOLOv9c derin öğrenme modeli de kullanılmıştır. YOLOv9c, araçlar, yayalar, trafik işaretleri ve sürüş dünyasının diğer temel bileşenlerinde %90 ortalama doğruluk (mAP) ile güçlü bir performans göstermiştir. Bu işlevsellik, üstün bilgisayarla görme yöntemlerinin homomorfik şifreleme ile entegre edilmesinin güvenli ve verimli bir algılama sistemi oluşturmadaki kararını doğrulamıştır. Bulgular, çerçevenin işlem süresi, bellek kullanımı ve doğruluk arasında bir denge sağladığını göstermiştir. Homomorfik şifrelemenin ek hesaplama ve depolama yükleri getirmesine rağmen, elde edilen yüksek gizlilik ve güvenlik bu maliyeti telafi etmiştir. Deneyler, şifrelenmiş görüntülerin yapısal bütünlüğünün korunduğunu ve zor koşullarda bile şifrelenmiş görüntülerin anlamlı karşılaştırmalarının ve doğru tespitlerinin mümkün olduğunu göstermiştir. Daha da önemlisi, sistem nesneleri bulut veya üçüncü taraflara şifrelenmemiş verileri açığa çıkarmadan tutarlı bir şekilde tanıyabilmiştir. Bu çalışma, BFV şeması kullanılarak tam homomorfik şifreleme (FHE) ile şifrelenmiş görüntü

bilgilerinin güvenli ve emniyetli bir şekilde işlenebileceğini, tespit yeteneklerinden ödün verilmeden göstermek amacıyla gerçekleştirilmiştir. Sistem, bulut altyapısının hesaplama kapasitesinden faydalanırken, otonom araçların topladığı hassas görsel bilgiler işleme sırasında açığa çıkarılmamıştır. Çözüm kapsamında kullanılan üç görüntü algılama yöntemi şunlardır: yüzdelik benzerlik tabanlı algılama, bilinen nesne tabanlı algılama ve bilinmeyen nesne tabanlı algılama. Her iki yöntem de çerçeveye yenilik katmış ve farklı algılama senaryolarıyla başa çıkmak için esneklik sağlamıştır. Yüzdelik benzerlik yöntemi, şifrelenmiş görüntülerin benzerlik derecesini tahmin ederek karşılaştırılmasına olanak tanımış ve benzer ya da yakın kopya görüntülerin bulunmasında faydalı olmuştur. Kurulan nesne tabanlı sistem ise, bulutta önceden şifrelenmiş nesneleri hızlı ve etkili bir şekilde bulmayı sağlayan pratik bir yöntem sunmuştur. Gerçek dünya sürüş ortamlarının belirsizliklerini azaltmak için, bilinmeyen nesne tabanlı yöntem sistemin esnekliğini artırmıştır; böylece başlangıç veri setinde temsil edilmeyen nesneler bile algılanabilmiştir. Bu kriptografik araçlara ek olarak, yüksek doğruluk sağlamak amacıyla YOLOv9c derin öğrenme modeli de sisteme entegre edilmiştir. %90 ortalama doğruluk (mAP) performansına sahip YOLOv9c, araçlar, yayalar, trafik işaretleri ve sürüş dünyasının diğer önemli unsurlarında güçlü bir performans sergilemiştir. Bu işlevsellik, en iyi bilgisayarla görme tekniklerinin homomorfik şifreleme ile birleştirilerek güvenli ve verimli bir algılama mekanizması oluşturulmasının gerekçesini ortaya koymaktadır. Sonuçlar, çerçevenin işlem süresi, bellek kullanımı ve doğruluk arasında bir denge sağladığını doğrulamıştır. Homomorfik şifreleme, hem hesaplama hem de depolamada bazı ek yükler getirse de, elde edilen yüksek gizlilik ve güvenlik bu maliyeti telafi etmiştir. Deneyler, şifrelenmiş görüntülerin yapısal bütünlüğünün korunduğunu ve zorlu koşullar altında bile anlamlı karşılaştırmaların ve doğru tespitlerin yapılabileceğini göstermiştir. Daha da önemlisi, sistem nesneleri düzenli olarak tanıyabilir ve şifrelenmemiş verileri bulut veya üçüncü taraflara açmadan çalışabilir. Bu çalışmanın önemi, gizliliği koruyan yöntemlerin, en son derin öğrenme teknikleriyle etkin bir şekilde kullanılabileceğini ve uygulamanın ciddi şekilde kısıtlanmadan gerçekleştirilebileceğini göstermesidir. Şifreleme ve algılama yöntemlerinin birleştirilmesi, sürüş koşulları, yayalar ve trafik ışıkları gibi hassas bilgilerin üçüncü taraflara açılmadan gizlenebileceğini ve yine de etkin bir şekilde analiz edilebileceğini göstermektedir. Bu başarı, bulut bilişimdeki en büyük zorluklardan biri olan güvenlik ve kullanım kolaylığı arasındaki denge sorununa doğrudan işaret etmektedir. Model,

orijinal görüntüleri şifrelenmiş ortamda tutmakta ve böylece insanların otonom araç sistemlerine olan güvenini artırmakta ve veri işleme tercihlerinin korunmasını sağlamaktadır. Benzerlik tabanlı yüzdelik arama, zorlu hava koşullarında gerçekleştirilmek zorunda olduğu için daha zordur; ayrıca, araç kameralarıyla alınan görüntülerin düşük kalitesi nedeniyle benzerlik yüzdeleri düşük çıkmıştır. Bu durum, gerçek dünyada şifrelenmiş görüntü algılamanın yeteneklerini ve sınırlamalarını ortaya koymuştur. Nesne tabanlı algılama ise en etkili yöntem olarak belirlenmiş, en düşük işlem yüküne sahip olmuş ve hızlı sonuçlar üretmiştir. Öte yandan, daha önce tanınmamış nesnelerin başarılı bir şekilde tespit edildiği durumlarda, hesaplama açısından daha maliyetli olan nesne tabanlı algılama yöntemi de haklı çıkarılmıştır. Bu bulgular, FHE gibi gelişmiş kriptografik sistemlerin, kullanım kolaylığı veya güvenlikten ödün vermeden bulut ortamında görüntü işlemek için kullanılabileceğini göstermektedir. Bu araştırma sadece otonom araçlar açısından değil, genel olarak da önem taşımaktadır. Önerilen yöntem, şifreleme sırasında görüntülerin bozulmasını önlemenin yanı sıra, homomorfik şifreleme sayesinde şifrelenmiş veriler üzerinde doğrudan işlem yapılmasına olanak sağlamaktadır; böylece görüntüler piksel seviyesinde korunmaktadır. Bu yenilik, görsel verilerin sağlık, gözetim ve akıllı şehirler gibi farklı alanlarda analiz ve algılama görevlerinde daha etkili bir şekilde kullanılabileceği anlamına gelmektedir. Sonuçlar, gizlilik korumalı nesne algılamanın yalnızca teorik olarak değil, dağıtık bilişim sistemleri, örneğin bulutlar üzerinde de uygulanabilir olduğunu göstermektedir. Benzerlik tabanlı yüzdelik arama zorlu hava koşullarında gerçekleştirildiği için zorlayıcı olmuş, fakat görüntülerin kalitesi düşük olduğundan benzerlik yüzdeleri düşük çıkmıştır. Nesne tabanlı algılama, en düşük işlem yükü ve hızlı sonuçlar ile en verimli yöntem olarak ortaya çıkmıştır. Öte yandan, hesaplama açısından daha maliyetli olan nesne tabanlı algılama, bilinmeyen nesneler başarılı bir şekilde tespit edildiğinde haklı bulunmuştur. Birlikte ele alındığında, bu sonuçlar, FHE gibi sofistike kriptografik araçların bulut üzerinde görüntüleri işlerken hem güvenlik hem de kullanım kolaylığı açısından avantaj sağlayabileceğini kanıtlamaktadır. Önerilen yöntem, şifreleme sırasında görüntülerin bozulmasını önlemek ve şifrelenmiş veriler üzerinde doğrudan işlem yapılmasını sağlamak suretiyle görsel verilerin farklı uygulamalarda daha etkili kullanılmasına imkan tanımaktadır. Sonuçlar, gizliliği koruyan nesne algılamanın yalnızca teorik bir model olmadığını, dağıtık bulut sistemlerinde de uygulanabilir olduğunu göstermektedir.

8.1. Gelecek Yönelimler

Önerilen çerçevenin ümit verici sonuçlarına rağmen, gelecekte iyileştirilebilecek ve araştırılabilecek çeşitli alanlar bulunmaktadır. Bu fırsatlar, performansı en üst düzeye çıkarmayı, hesaplama yükünü en aza indirmeyi ve sistemin daha geniş bağlamlarda uygulanabilirliğini artırmayı hedeflemektedir. FHE'nin getirdiği depolama ve hesaplama yükü, temel sorunlardan biri olarak tespit edilmiştir. Gelecekteki çalışmalarda, şifreleme, çıkarma ve çözme işlemlerini hızlandırmak için Grafik İşlem Birimi (GPU), Alan Programlanabilir Kapı Dizileri (FPGA) veya özel kriptografik hızlandırıcılar gibi donanım optimizasyon teknikleri araştırılabilir. Bu tür optimizasyonlar, şifrelenmiş algılama için gecikmeyi, yüksek hızlı sürüş senaryolarında bile gerçek zamanlı yanıt gereksinimlerini karşılayacak seviyelere düşürebilir. Yağmur, sis, kar veya düşük ışık gibi dış faktörler genellikle görüntü kalitesini etkiler ve dolayısıyla algılama doğruluğunu düşürür. Önerilen FHE tabanlı görüntü tespit sistemi, gelecekte işlem verimliliği ve ölçeklenebilirlik açısından daha da geliştirilebilir. Batch işlemleri, ciphertext paketleme ve SIMD gibi paralel işleme teknikleri kullanılarak çok sayıda verinin tek bir şifreli metin içinde işlenmesi sağlanabilir ve büyük çaplı şifreli görüntü veri setlerinde işlem süresi önemli ölçüde azaltılabilir. Ayrıca, bootstrapping gerektirmeyen bir çalışma yapısının tasarlanması, gürültü yönetimi maliyetini ortadan kaldırarak derin homomorfik işlemlerin daha verimli biçimde yürütülmesini sağlar. Donanım hızlandırıcılarının (GPU, TPU ve FPGA) entegrasyonu ise şifreleme, çözme ve görüntü arama süreçlerini daha da hızlandırarak, otonom araçlardan gelen büyük ölçekli veriler için gerçek zamanlı işleme kapasitesi sunabilir. Bu iyileştirmeler, FHE'nin performansını ve gizliliği koruyan görüntü analizindeki uygulanabilirliğini önemli ölçüde artırma potansiyeline sahiptir.Önerilen sistem dirençli olsa da, kötü hava koşullarında benzerlik yüzdeleri düşmüştür. Gelecekteki çalışmalar, işleme öncesi şifrelenmiş görüntülerin kalitesini artırmak için yüksek dereceli görüntü iyileştirme yöntemlerini, derin öğrenmeye dayalı gürültü giderme ve süper çözünürlük sinir ağlarını içermelidir. Ayrıca, olumsuz koşullarda algılamayı geliştirecek özel sensörler ve kameralar uygulanabilir, böylece gizliliği koruyan algılama her ortamda etkili olur. Bilinmeyen nesne tabanlı algılamada hesaplama gecikmesini aşmak için, sonraki uygulamalarda paralel ve dağıtılmış işleme yaklaşımları kullanılabilir. Bulut veya uç (edge) ortamda birden fazla düğüm kullanılarak iş yükü dağıtıldığında, sistem aynı anda birden fazla şifreli deseni

işleyebilir. Bu yöntem, algılama gecikmesini azaltacak ve sistemi daha ölçeklenebilir hâle getirecektir; bu, akıllı ulaşım sistemlerinde büyük ölçekli uygulamalarda avantajlıdır. Mevcut sistem büyük ölçüde bulut altyapısı kullanmaktadır; bu durum gecikme ve bağlantı riskleri yaratmaktadır. Gelecekte, uç ve sis (fog) bilişim entegrasyonu umut verici bir yaklaşım olacaktır. Şifrelenmiş veriler kaynağa daha yakın işlenebilir, böylece gecikme azaltılır ve araçlar tamamen uzak sunuculara bağımlı olmadan daha hızlı karar verebilir. Uç, sis ve bulut katmanlarıyla oluşturulmuş mimariler, hız, ölçeklenebilirlik ve gizlilik açısından ideal bir denge sunabilir. Şifrelenmiş veriler, düz metin verilere göre daha fazla depolama kapasitesi gerektirecektir. Gelecekteki çalışmalar, şifreli verilerde özel sıkıştırma algoritmalarının kullanımını araştırabilir. Bu algoritmalar, depolama yükünü azaltırken şifreli verilerin bütünlüğünü ve güvenliğini koruyabilir. Donanım gelişimiyle birleştirildiğinde, bu yöntem, büyük ölçekli şifrelenmiş görüntülerin depolanmasını ve işlenmesini daha uygulanabilir ve uygun maliyetli hâle getirecektir.

Kuantum bilişim sistemlerinin gelişimi devam ettikçe, klasik şifreleme sistemleri savunmasız hâle gelecektir. Gelecek araştırmalar, kuantum algoritmalarıyla saldırıya uğramayan kuantum dirençli homomorfik şifreleme şemalarının uygulanmasını göz önünde bulundurmalıdır. Çerçevenin post-kuantum döneme hazır olması, uzun vadede gizliliği koruyan algılama sistemlerinin güvenliğini ve sürdürülebilirliğini garanti edecektir. YOLOv9c oldukça verimli olsa da, YOLO ailesinin bir sonraki nesilleri veya diğer sofistike algılama modelleri daha yüksek doğruluk ve verimlilik sağlayabilir. Ayrıca, şifrelenmiş işleme ile kullanılabilecek hafif YOLO versiyonlarının araştırılması önemlidir; böylece şifreleme yükü olsa bile nesne algılama hâlâ hızlı kalabilir. Adaptif öğrenme teknikleri, örneğin transfer öğrenme, sistemin yeni nesne sınıflarına hızlı bir şekilde uyum sağlamasına olanak tanıyabilir, sistemin yeniden eğitilmesi gerekmeden. Mevcut araştırma, otonom araçlara odaklansa da, çerçevenin diğer alanlarda da geniş uygulanabilirliği vardır. Gizliliği koruyan şifreli görüntü algılama, sağlık görüntüleme, gözetim sistemleri, endüstriyel izleme ve akıllı şehir uygulamalarında da faydalı olabilir. Gelecek çalışmalar, bu sistemin bu alanlarda güncellenmesini ve doğrulanmasını sağlayarak, sistemin topluma etkisini ve uygulanabilirliğini gösterebilir. Gelecekteki yönelimler ayrıca federated learning (federatif öğrenme) yöntemlerini de içerebilir; burada modeller yerel olarak şifreli verilerle eğitilir ve ham bilgiler gönderilmez. Araçlar kendi şifreli

verilerini kullanabilir ve modele güncellemeleri buluta iletebilir. Bu, homomorfik şifrelemenin avantajlarını dağıtık öğrenme ile birleştirir, gizliliği artırır ve iletişim maliyetlerini en aza indirir.

KAYNAKLAR

A. Abu Aziz, A., N.Qunoo, H., & A. Abu Samra, A. (2018). Using Homomorphic Cryptographic Solutions on E-voting Systems. *International Journal of Computer Network and Information Security*, *10*(1), 44–59. doi: 10.5815/ijcnis.2018.01.06

A Chaudhari, & R Bansode. (2021). Survey on Securing IoT Data using Homomorphic Encryption Scheme. International Journal of Engineering. International Journal of Engineering, 4, 76–81.

Abdulsada, A. I., & Taha, N. A. (2019a). Towards efficient privacy-preserving image similarity detection. 050005. doi: 10.1063/1.5123121

Abdulsada, A. I., & Taha, N. A. (2019b). Towards efficient privacy-preserving image similarity detection. 050005. doi: 10.1063/1.5123121

Abreu, Z., & Pereira, L. (2022). Privacy protection in smart meters using homomorphic encryption: An overview. WIREs Data Mining and Knowledge Discovery, 12(4). doi: 10.1002/widm.1469

Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2019a). A Survey on Homomorphic Encryption Schemes. ACM Computing Surveys, 51(4), 1–35. doi: 10.1145/3214303

Acar, A., Aksu, H., Uluagac, A. S., & Conti, M. (2019b). A Survey on Homomorphic Encryption Schemes. ACM Computing Surveys, 51(4), 1–35. doi: 10.1145/3214303

Adak, M. F., Kose, Z. N., & Akpinar, M. (2023). Dynamic Data Masking by Two-Step Encryption. 2023 Innovations in Intelligent Systems and Applications Conference (ASYU), 1–5. doi: 10.1109/ASYU58738.2023.10296545

Adnan Yusuf, S., Khan, A., & Souissi, R. (2024). Vehicle-to-everything (V2X) in the autonomous vehicles domain – A technical review of communication, sensor, and AI technologies for road user safety. Transportation Research Interdisciplinary Perspectives, 23, 100980. doi: 10.1016/j.trip.2023.100980

Ahmed, I. M., & Kashmoola, M. Y. (2021). Threats on Machine Learning Technique by Data Poisoning Attack: A Survey (pp. 586–600). doi: 10.1007/978-981-16-8059-5_36

Al Badawi, A., Bates, J., Bergamaschi, F., Cousins, D. B., Erabelli, S., Genise, N., Halevi, S., Hunt, H., Kim, A., Lee, Y., Liu, Z., Micciancio, D., Quah, I., Polyakov, Y., R.V., S., Rohloff, K., Saylor, J., Suponitsky, D., Triplett, M., … Zucca, V. (2022). OpenFHE. Proceedings of the 10th Workshop on Encrypted Computing & Applied Homomorphic Cryptography, 53–63. doi: 10.1145/3560827.3563379

Alam, T. (2024). Data Privacy and Security in Autonomous Connected Vehicles in Smart City Environment. Big Data and Cognitive Computing, 8(9), 95. doi: 10.3390/bdcc8090095

Alin Cijov. (n.d.). Self-Driving Cars (R-CNN, SSD, Yolo - Object Detection Dataset). Https://Www.Kaggle.Com/Datasets/Alincijov/Self-Driving-Cars.

Alloghani, M., M. Alani, M., Al-Jumeily, D., Baker, T., Mustafina, J., Hussain, A., & J. Aljaaf, A. (2019). A systematic review on the status and progress of homomorphic encryption technologies. Journal of Information Security and Applications, 48, 102362. doi: 10.1016/j.jisa.2019.102362

Arrojo, M. J. (2025). Impact of Artificial Intelligence in the Audiovisual Industry. A Taxonomy of Uses and Applications (pp. 183–194). doi: 10.1007/978-981-96-0426-5_16

Ayala, R., & Mohd, T. K. (2021). Sensors in Autonomous Vehicles: A Survey. Journal of Autonomous Vehicles and Systems, 1(3). doi: 10.1115/1.4052991

Ayoosh Kathuria. (2018). What's new in YOLO v3? Towards Data Science .

Benaissa, A., Retiat, B., Cebere, B., & Belfedhal, A. E. (2021). TenSEAL: A Library for Encrypted Tensor Operations Using Homomorphic Encryption.

Benzekki, Kamal, Abdeslam El Fergougui, & Abdelbaki El Belrhiti El Alaoui. (2016). A secure cloud computing architecture using homomorphic encryption. International Journal of Advanced Computer Science and Applications , 7(2).

Biswas, D. G., Das, S., Kairi, A., Roy, A., Saha, T., & Samanta, M. (2024). Comparative Performance Analysis of Security Encryption Algorithms (pp. 145–161). doi: 10.1007/978-3-031-71125-1_12

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection.

Boucenna, F., Nouali, O., Kechid, S., & Tahar Kechadi, M. (2019). Secure Inverted Index Based Search over Encrypted Cloud Data with User Access Rights Management. Journal of Computer Science and Technology, 34(1), 133–154. doi: 10.1007/s11390-019-1903-2

Boyanov, P. (2024). PRACTICAL APPLICATIONS OF HASH FUNCTIONS MD5, SHA-1, AND SHA-256 USING VARIOUS SOFTWARE TOOLS TO VERIFY THE INTEGRITY OF FILES. Journal Scientific and Applied Research, 27(1), 120–137. doi: 10.46687/jsar.v27i1.413

Buell, D. (2021). Modern Symmetric Ciphers—DES and AES (pp. 123–147). doi: 10.1007/978-3-030-73492-3_9

Burghardt, Thorben, Erik Buchmann, & Klemens Böhm. (2008). Why do privacy-enhancement mechanisms fail, after all? a survey of both, the user and the provider perspective. In Workshop W2Trust, in Conjunction with IFIPTM, 8.

Butt, U. A., Amin, R., Mehmood, M., Aldabbas, H., Alharbi, M. T., & Albaqami, N. (2023). Cloud Security Threats and Solutions: A Survey. Wireless Personal Communications, 128(1), 387–413. doi: 10.1007/s11277-022-09960-z

Chao-Yung Hsu, Chun-Shien Lu, & Soo-Chang Pei. (2012). Image Feature Extraction in Encrypted Domain With Privacy-Preserving SIFT. IEEE Transactions on Image Processing, 21(11), 4593–4607. doi: 10.1109/TIP.2012.2204272

Charroud, A., El Moutaouakil, K., Palade, V., Yahyaouy, A., Onyekpe, U., & Eyo, E. U. (2024). Localization and Mapping for Self-Driving Vehicles: A Survey. Machines, 12(2), 118. doi: 10.3390/machines12020118

Chase, M., Ding, J., Goldwasser, S., Gorbunov, S., Hoffstein, J., Lauter, K., Lokam, S., Moody, D., & Morrison, T. (2017). Security of homomorphic encryption. Homomorphic Encryption. Org, Redmond WA, Tech. Rep. .

Chen, A. C. H. (2024). Report on Hash Algorithm Performance ─ A Case Study of Cryptocurrency Exchanges Based on Blockchain System. 2024 5th International Conference on Data Intelligence and Cognitive Informatics (ICDICI), 30–35. doi: 10.1109/ICDICI62993.2024.10810832

Chen, X., Wang, C., Liu, C., Zhu, X., Zhang, Y., Luo, T., & Zhang, J. (2024). Autonomous Crack Detection for Mountainous Roads Using UAV Inspection System. Sensors, 24(14), 4751. doi: 10.3390/s24144751

Cheon, J. H., Kim, A., Kim, M., & Song, Y. (2017). Homomorphic Encryption for Arithmetic of Approximate Numbers (pp. 409–437). doi: 10.1007/978-3-319-70694-8_15

Chillotti, I., Gama, N., Georgieva, M., & Izabachène, M. (2016). Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds (pp. 3–33). doi: 10.1007/978-3-662-53887-6_1

Dam, D.-T., Tran, T.-H., Hoang, V.-P., Pham, C.-K., & Hoang, T.-T. (2023). A Survey of Post-Quantum Cryptography: Start of a New Race. Cryptography, 7(3), 40. doi: 10.3390/cryptography7030040

Das, D., Naskar, R., & Chakraborty, R. S. (2023). Image splicing detection with principal component analysis generated low-dimensional homogeneous feature set based on local binary pattern and support vector machine. Multimedia Tools and Applications, 82(17), 25847–25864. doi: 10.1007/s11042-023-14658-w

Dasgupta, S., Rahman, M., Islam, M., & Chowdhury, M. (2022). A Sensor Fusion-Based GNSS Spoofing Attack Detection Framework for Autonomous Vehicles. IEEE Transactions on Intelligent Transportation Systems, 23(12), 23559–23572. doi: 10.1109/TITS.2022.3197817

Data61, C. (2013). Python Paillier Library. Https://Github.Com/Data61/Python-Paillier.

Dey, J., & Dutta, R. (2023). Progress in Multivariate Cryptography: Systematic Review, Challenges, and Research Directions. ACM Computing Surveys, 55(12), 1–34. doi: 10.1145/3571071

Diwan, T., Anirudh, G., & Tembhurne, J. V. (2023). Object detection using YOLO: challenges, architectural successors, datasets and applications. Multimedia Tools and Applications, 82(6), 9243–9275. doi: 10.1007/s11042-022-13644-y

Doan, T. V. T., Messai, M.-L., Gavin, G., & Darmont, J. (2023). A survey on implementations of homomorphic encryption schemes. The Journal of Supercomputing, 79(13), 15098–15139. doi: 10.1007/s11227-023-05233-z

Du, A., Wang, L., Cheng, S., & Ao, N. (2020). A Privacy-Protected Image Retrieval Scheme for Fast and Secure Image Search. Symmetry, 12(2), 282. doi: 10.3390/sym12020282

Duan, Y., Li, Y., Lu, L., & Ding, Y. (2022). A faster outsourced medical image retrieval scheme with privacy preservation. Journal of Systems Architecture, 122, 102356. doi: 10.1016/j.sysarc.2021.102356

Ducas, L., & Micciancio, D. (2015). FHEW: Bootstrapping Homomorphic Encryption in Less Than a Second (pp. 617–640). doi: 10.1007/978-3-662-46800-5_24

Fathalla, E., & Azab, M. (2024). Beyond Classical Cryptography: A Systematic Review of Post-Quantum Hash-Based Signature Schemes, Security, and Optimizations. IEEE Access, 12, 175969–175987. doi: 10.1109/ACCESS.2024.3485602

Fernanda, A., Huda, M., & Fadri Geovanni, A. R. (2023). Application of Learning Cloud Computing Technology (Cloud Computing) to Students in Higher Education. International Journal of Cyber and IT Service Management, 3(1), 32–39. doi: 10.34306/ijcitsm.v3i1.121

Geelen, R., & Vercauteren, F. (2023). Bootstrapping for BGV and BFV Revisited. Journal of Cryptology, 36(2), 12. doi: 10.1007/s00145-023-09454-6

Gothwal, R., Dharmani, G., Reen, R. S., & AbdAllah, E. G. (2023). Evaluation of Man-in-the-Middle Attacks and Countermeasures on Autonomous Vehicles. 2023 10th International Conference on Dependable Systems and Their Applications (DSA), 502–509. doi: 10.1109/DSA59317.2023.00070

Gupta, C. P., & Sharma, I. (2013). A fully homomorphic encryption scheme with symmetric keys with application to private data processing in clouds. 2013 Fourth International Conference on the Network of the Future (NoF), 1–4. doi: 10.1109/NOF.2013.6724526

Gupta, I., & Singh, A. K. (2022). A Holistic View on Data Protection for Sharing, Communicating, and Computing Environments: Taxonomy and Future Directions.

Halevi, S., & Shoup, V. (2014). Algorithms in HElib (pp. 554–571). doi: 10.1007/978-3-662-44371-2_31

Hasan Ali Akyürek, Hasan İbrahim Kozan, & Şakir Taşdemir. (2024). Surface Crack Detection in Historical Buildings with Deep Learning-based YOLO Algorithms: A Comparative Study. CRPASE: Transactions of Electrical, Electronic and Computer Engineering , 10(3), 1–14.

Hsu, C.-Y., Lu, C.-S., & Pei, S.-C. (2011). Homomorphic encryption-based secure SIFT for privacy-preserving feature extraction (N. D. Memon, J. Dittmann, A. M. Alattar, & E. J. Delp III, Eds.; p. 788005). doi: 10.1117/12.873325

Hu, R., Zhang, L., & Qin, Y. (2016). Improved Fully Homomorphic Encryption Algorithm for Cloud Storage. Proceedings of the 2016 International Conference on Communications, Information Management and Network Security. doi: 10.2991/cimns-16.2016.87

Hu, S., Wang, Q., Wang, J., Qin, Z., & Ren, K. (2016). Securing SIFT: Privacy-Preserving Outsourcing Computation of Feature Extractions Over Encrypted Image Data. IEEE Transactions on Image Processing, 25(7), 3411–3425. doi: 10.1109/TIP.2016.2568460

Hussain Akbar, Muhammad Zubair, & Muhammad Shairoze Malik. (2023). The Security Issues and challenges in Cloud Computing. International Journal for Electronic Crime Investigation, 7(1), 13–32. doi: 10.54692/ijeci.2023.0701125

Iqbal, Y., Tahir, S., Tahir, H., Khan, F., Saeed, S., Almuhaideb, A. M., & Syed, A. M. (2022). A Novel Homomorphic Approach for Preserving Privacy of Patient Data in Telemedicine. Sensors, 22(12), 4432. doi: 10.3390/s22124432

Issaoui, H., ElAdel, A., & ZAIED, M. (2023). B-CNN: Betadeep Convolutional Neural Network over encrypted data. J. (Jessica) Zhou, W. Osten, & D. P. Nikolaev (Eds.), Fifteenth International Conference on Machine Vision (ICMV 2022) (p. 18). SPIE. doi: 10.1117/12.2679393

J. Fan and F. Vercauteren. (2012). Somewhat practical fully homomorphic encryption.

Janani, M., Jeevitha, R., Jaikumar, R., Suganthi, R., & Jhansi Ida, S. (2023). Multivariate Cryptosystem Based on a Quadratic Equation to Eliminate the Outliers Using Homomorphic Encryption Scheme. In Homomorphic Encryption for Financial Cryptography (pp. 277–302). Cham: Springer International Publishing. doi: 10.1007/978-3-031-35535-6_13

Jani, M., Fayyad, J., Al-Younes, Y., & Najjaran, H. (2023). Model Compression Methods for YOLOv5: A Review.

Jiang, D., & Kim, J. (2021). Image Retrieval Method Based on Image Feature Fusion and Discrete Cosine Transform. Applied Sciences, 11(12), 5701. doi: 10.3390/app11125701

Kamal, S. T., Hosny, K. M., Elgindy, T. M., Darwish, M. M., & Fouda, M. M. (2021). A New Image Encryption Algorithm for Grey and Color Medical Images. IEEE Access, 9, 37855–37865. doi: 10.1109/ACCESS.2021.3063237

Kaviani, S., Han, K. J., & Sohn, I. (2022). Adversarial attacks and defenses on AI in medical imaging informatics: A survey. Expert Systems with Applications, 198, 116815. doi: 10.1016/j.eswa.2022.116815

Khan, A. Q., Matskin, M., Prodan, R., Bussler, C., Roman, D., & Soylu, A. (2024). Cloud storage cost: a taxonomy and survey. World Wide Web, 27(4), 36. doi: 10.1007/s11280-024-01273-4

Kim, D., Park, J., Kim, J., Kim, S., & Ahn, J. H. (2024). HyPHEN: A Hybrid Packing Method and Its Optimizations for Homomorphic Encryption-Based Neural Networks. IEEE Access, 12, 3024–3038. doi: 10.1109/ACCESS.2023.3348170

Kirthika, N., & Sargunam, B. (2021). YOLOv4 for Multi-class Artefact Detection in Endoscopic Images. 2021 3rd International Conference on Signal Processing and Communication (ICPSC), 73–77. doi: 10.1109/ICSPC51351.2021.9451761

Kocabas, O., & Soyata, T. (2020). Towards Privacy-Preserving Medical Cloud Computing Using Homomorphic Encryption. In Virtual and Mobile Healthcare (pp. 93–125). IGI Global. doi: 10.4018/978-1-5225-9863-3.ch005

Kölsch, J., Heinz, C., Ratzke, A., & Grimm, C. (2019). Simulation-Based Performance Validation of Homomorphic Encryption Algorithms in the Internet of Things. Future Internet, 11(10), 218. doi: 10.3390/fi11100218

Kumari, K. A., Sharma, A., Chakraborty, C., & Ananyaa, M. (2022). Preserving Health Care Data Security and Privacy Using Carmichael's Theorem-Based Homomorphic Encryption and Modified Enhanced Homomorphic Encryption Schemes in Edge Computing Systems. Big Data, 10(1), 1–17. doi: 10.1089/big.2021.0012

Kuo, T.-H., & Wu, J.-L. (2023). A High Throughput BFV-Encryption-Based Secure Comparison Protocol. Mathematics, 11(5), 1227. doi: 10.3390/math11051227

Lai, J., Deng, R. H., Pang, H., & Weng, J. (2014). Verifiable Computation on Outsourced Encrypted Data (pp. 273–291). doi: 10.1007/978-3-319-11203-9_16

Lee, J., Duong, P. N., & Lee, H. (2023). Configurable Encryption and Decryption Architectures for CKKS-Based Homomorphic Encryption. Sensors, 23(17), 7389. doi: 10.3390/s23177389

Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., Li, Y., Zhang, B., Liang, Y., Zhou, L., Xu, X., Chu, X., Wei, X., & Wei, X. (2022). YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications.

Li, W. (2021). Analysis of Object Detection Performance Based on Faster R-CNN. Journal of Physics: Conference Series, 1827(1), 012085. doi: 10.1088/1742-6596/1827/1/012085

Li, X., Liu, J., Obaidat, M. S., Vijayakumar, P., Jiang, Q., & Amin, R. (2021). An Unlinkable Authenticated Key Agreement With Collusion Resistant for VANETs. IEEE Transactions on Vehicular Technology, 70(8), 7992–8006. doi: 10.1109/TVT.2021.3087557

Li, Y., Ma, J., Miao, Y., Wang, Y., Liu, X., & Choo, K.-K. R. (2022). Similarity Search for Encrypted Images in Secure Cloud Computing. IEEE Transactions on Cloud Computing, 10(2), 1142–1155. doi: 10.1109/TCC.2020.2989923

Li, Z., Galbraith, S. D., & Ma, C. (2016). Preventing Adaptive Key Recovery Attacks on the GSW Levelled Homomorphic Encryption Scheme (pp. 373–383). doi: 10.1007/978-3-319-47422-9_22

Liang, J., Li, P., Liu, Z., & He, H. (2025). Privacy-Preserving Clustering-Based Image Retrieval in Cloud-Assisted Internet of Things. IEEE Internet of Things Journal, 1–1. doi: 10.1109/JIOT.2025.3546416

Liu, J., Li, X., Jiang, Q., Obaidat, M. S., & Vijayakumar, P. (2020). BUA: A Blockchain-based Unlinkable Authentication in VANETs. ICC 2020 - 2020 IEEE International Conference on Communications (ICC), 1–6. doi: 10.1109/ICC40277.2020.9148863

Mahato, G. K., & Chakraborty, S. K. (2023). A Comparative Review on Homomorphic Encryption for Cloud Security. IETE Journal of Research, 69(8), 5124–5133. doi: 10.1080/03772063.2021.1965918

Malhotra, P., Singh, Y., Anand, P., Bangotra, D. K., Singh, P. K., & Hong, W.-C. (2021). Internet of Things: Evolution, Concerns and Security Challenges. Sensors, 21(5), 1809. doi: 10.3390/s21051809

Mansoor, S., & Parah, S. A. (2023). HAIE: a hybrid adaptive image encryption algorithm using Chaos and DNA computing. Multimedia Tools and Applications, 82(19), 28769–28796. doi: 10.1007/s11042-023-14542-7

Marcolla, C., Sucasas, V., Manzano, M., Bassoli, R., Fitzek, F. H. P., & Aaraj, N. (2022a). Survey on Fully Homomorphic Encryption, Theory, and Applications. Proceedings of the IEEE, 110(10), 1572–1609. doi: 10.1109/JPROC.2022.3205665

Marcolla, C., Sucasas, V., Manzano, M., Bassoli, R., Fitzek, F. H. P., & Aaraj, N. (2022b). Survey on Fully Homomorphic Encryption, Theory, and Applications. Proceedings of the IEEE, 110(10), 1572–1609. doi: 10.1109/JPROC.2022.3205665

Mathur, P. (2024). Cloud Computing Infrastructure, Platforms, and Software for Scientific Research (pp. 89–127). doi: 10.1007/978-981-97-1017-1_4

Medileh, S., Laouid, A., Hammoudeh, M., Kara, M., Bejaoui, T., Eleyan, A., & Al-Khalidi, M. (2023). A Multi-Key with Partially Homomorphic Encryption Scheme for Low-End Devices Ensuring Data Integrity. Information, 14(5), 263. doi: 10.3390/info14050263

Meraoumia, M. A. A., Laimech, L., & Bendjenna, H. (2021). Biometric Cryptosystem to Secure Smart Object Communications in the Internet of Things. Kuwait Journal of Science. doi: 10.48129/kjs.11169

Miao, Y., Song, L., Li, X., Li, H., Choo, K.-K. R., & Deng, R. H. (2024). Privacy-Preserving Arbitrary Geometric Range Query in Mobile Internet of Vehicles. IEEE Transactions on Mobile Computing, 23(7), 7725–7738. doi: 10.1109/TMC.2023.3336621

Moayed, H., & Mansoori, E. G. (2023). Improving Regularization in Deep Neural Networks by Co-adaptation Trace Detection. Neural Processing Letters, 55(6), 7985–7997. doi: 10.1007/s11063-023-11293-2

Mohammed, S. J., & Taha, D. B. (2023). Paillier cryptosystem enhancement for Homomorphic Encryption technique. Multimedia Tools and Applications, 83(8), 22567–22579. doi: 10.1007/s11042-023-16301-0

Mouchet, C. V., Bossuat, J. P., Troncoso-Pastoriza, J. R., & Hubaux, J. P. (2020). Lattigo: A multiparty homomorphic encryption library in go. In Proceedings of the 8th Workshop on Encrypted Computing and Applied Homomorphic Cryptography, 64–70.

Muthanna, M. S. A., Elkin, D., Likhtin, S., & Al-Sveiti Malik, A. M. (2024). Multilevel Edge Computing System for Autonomous Vehicles (pp. 3–12). doi: 10.1007/978-3-031-51097-7_1

Nepal, U., & Eslamiat, H. (2022). Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. Sensors, 22(2), 464. doi: 10.3390/s22020464

Nisha, F., Lenin, J., Saravanan, S. K., Rohit, V. R., Selvam, P. D., & Rajmohan, M. (2024). Lattice-Based Cryptography and NTRU: Quantum-Resistant Encryption Algorithms. 2024 International Conference on Emerging Systems and Intelligent Computing (ESIC), 509–514. doi: 10.1109/ESIC60604.2024.10481608

Nguyen, H. T., Nguyen, B. P., & Su, A. K. (2025). YOLOv9c: A Robust Framework for Insect Detection (pp. 30–43). doi: 10.1007/978-981-96-0695-5_3

Nižetić, S., Šolić, P., López-de-Ipiña González-de-Artaza, D., & Patrono, L. (2020). Internet of Things (IoT): Opportunities, issues and challenges towards a smart and sustainable future. Journal of Cleaner Production, 274, 122877. doi: 10.1016/j.jclepro.2020.122877

Nusari, A. N., Ba Alawi, A. E., Bozkurt, F., & Özbek, İ. Y. (2024). Comparison of YOLO Algorithms for Vehicle Accident Detection and Classification. 2024 4th International Conference on Emerging Smart Technologies and Applications (ESmarTA), 1–8. doi: 10.1109/eSmarTA62850.2024.10638929

Oczak, M., Bayer, F., Vetter, S. G., Maschat, K., & Baumgartner, J. (2022). Where is the sow's nose: RetinaNet object detector as a basis for monitoring the use of rack with nest-building material. Frontiers in Animal Science, 3. doi: 10.3389/fanim.2022.913407

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., & Walsh, J. (2020). Deep Learning vs. Traditional Computer Vision (pp. 128–144). doi: 10.1007/978-3-030-17795-9_10

Pan, J., Sui, T., Liu, W., Wang, J., Kong, L., & Zhao, Y. (2023). Secure Control Using Homomorphic Encryption and Efficiency Analysis. Security and Communication Networks, 2023, 1–12. doi: 10.1155/2023/6473497

Parekh, A., Antani, M., Suvarna, K., Mangrulkar, R., & Narvekar, M. (2023). Multilayer symmetric and asymmetric technique for audiovisual cryptography. Multimedia Tools and Applications, 83(11), 31465–31503. doi: 10.1007/s11042-023-16401-x

Peralta, G., Cid-Fuentes, R. G., Bilbao, J., & Crespo, P. M. (2019). Homomorphic Encryption and Network Coding in IoT Architectures: Advantages and Future Challenges. Electronics, 8(8), 827. doi: 10.3390/electronics8080827

Polyakov, Y., Rohloff, K., Ryan, G. W., & Cousins, D. (2017). Palisade lattice cryptography library user manual. Cybersecurity Research Center,. New Jersey Institute OfTechnology (NJIT), Tech. Rep, 15.

Prabu Kanna, G., & Vasudevan, V. (2019). A fully homomorphic–elliptic curve cryptography based encryption algorithm for ensuring the privacy preservation of the cloud data. Cluster Computing, 22(S4), 9561–9569. doi: 10.1007/s10586-018-2723-9

Pulido-Gaytan, B., Tchernykh, A., Cortés-Mendoza, J. M., Babenko, M., Radchenko, G., Avetisyan, A., & Drozdov, A. Y. (2021). Privacy-preserving neural networks with Homomorphic encryption: Challenges and opportunities. Peer-to-Peer Networking and Applications, 14(3), 1666–1691. doi: 10.1007/s12083-021-01076-8

Pulkit Sharma. (2023). A Step-by-Step Introduction to the Basic Object Detection Algorithms (Part 1). Analytics Vidhya.

Pyfhel Library. Available online: https://github.com/ibarrond/Pyfhel . (2023).

Qin, Z., Weng, J., Cui, Y., & Ren, K. (2024). Privacy-Preserving Image Processing in the Cloud. IEEE Cloud Computing, 1–1. doi: 10.1109/MCC.2018.111121403

Rani, N., Mishra, V., & Sharma, S. R. (2023). Image encryption model based on novel magic square with differential encoding and chaotic map. Nonlinear Dynamics, 111(3), 2869–2893. doi: 10.1007/s11071-022-07958-7

Rauthan, J. S. (2022). Fully homomorphic encryption: A case study. Journal of Intelligent & Fuzzy Systems, 43(6), 8417–8437. doi: 10.3233/JIFS-221454

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection.

Redmon, J., & Farhadi, A. (2016). YOLO9000: Better, Faster, Stronger.

Renugadevi, N., Thangallapally, S., Vemula, S. C., Julakanti, S., & Bhatnagar, S. (2022). Methods for improving the implementation of advanced encryption standard hardware accelerator on field programmable gate array-A survey. SECURITY AND PRIVACY, 5(6). doi: 10.1002/spy2.254

Ross Girshick. (2015). Fast R-CNN. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 1440–1448.

Saeed, K., & Adak, M. F. (2025a). Enhancing Privacy in Image Search: Secure Object-Based Retrieval from Encrypted Images Using BFV Homomorphic Encryption. 2025 International Conference for Artificial Intelligence, Applications, Innovation and Ethics (AI2E), 1–6. doi: 10.1109/AI2E64943.2025.10983163

Saeed, K., & Adak, M. F. (2025b). Secured cloud-based image data processing of self-driving vehicles using full homomorphic encryption. Kuwait Journal of Science, 52(4), 100449. doi: 10.1016/j.kjs.2025.100449

Sathish Kumar, G., Premalatha, K., Uma Maheshwari, G., & Rajesh Kanna, P. (2023). No more privacy Concern: A privacy-chain based homomorphic encryption scheme and statistical method for privacy preservation of user's private and sensitive data. Expert Systems with Applications, 234, 121071. doi: 10.1016/j.eswa.2023.121071

Savadatti, S. G., Dhariwal, S. K., Krishnamoorthy, S., & Delhibabu, R. (2025). Analyzing RSA, AES, and ECC Across 13 Critical Factors in the Healthcare Domain. 2025 International Conference on Next Generation Communication & Information Processing (INCIP), 593–598. doi: 10.1109/INCIP64058.2025.11019733

Silva, P., Monteiro, E., & Simoes, P. (2021). Privacy in the Cloud: A Survey of Existing Solutions and Research Challenges. IEEE Access, 9, 10473–10497. doi: 10.1109/ACCESS.2021.3049599

Sirisha, U., Praveen, S. P., Srinivasu, P. N., Barsocchi, P., & Bhoi, A. K. (2023). Statistical Analysis of Design Aspects of Various YOLO-Based Deep Learning Models for Object Detection. International Journal of Computational Intelligence Systems, 16(1), 126. doi: 10.1007/s44196-023-00302-w

Song, W.-T., Hu, B., & Zhao, X.-F. (2018). Privacy Protection of IoT Based on Fully Homomorphic Encryption. Wireless Communications and Mobile Computing, 2018, 1–7. doi: 10.1155/2018/5787930

Sood, R., & Kaur, H. (2023). A Literature Review on RSA, DES and AES Encryption Algorithms. In Emerging Trends in Engineering and Management (pp. 57–63). Soft Computing Research Society. doi: 10.56155/978-81-955020-3-5-07

Stalder, F. (2002). The Failure of Privacy Enhancing Technologies (PETs) and the Voiding of Privacy. Sociological Research Online, 7(2), 25–39. doi: 10.5153/sro.718

Subramaniyaswamy, V., Jagadeeswari, V., Indragandhi, V., Jhaveri, R. H., Vijayakumar, V., Kotecha, K., & Ravi, L. (2022). Somewhat Homomorphic Encryption: Ring Learning with Error Algorithm for Faster Encryption of IoT Sensor Signal-Based Edge Devices. Security and Communication Networks, 2022, 1–10. doi: 10.1155/2022/2793998

Sultana, S. F., & Shubhangi, D. C. (2017). Privacy preserving LBP based feature extraction on encrypted images. 2017 International Conference on Computer Communication and Informatics (ICCCI), 1–4. doi: 10.1109/ICCCI.2017.8117707

Sun, C., Ai, Y., Wang, S., & Zhang, W. (2021). Mask-guided SSD for small-object detection. Applied Intelligence, 51(6), 3311–3322. doi: 10.1007/s10489-020-01949-0

Tebaa, M., Hajji, S. El, & Ghazi, A. El. (2012). Homomorphic encryption method applied to Cloud Computing. 2012 National Days of Network Security and Systems, 86–89. doi: 10.1109/JNS2.2012.6249248

Tian, M., Zhang, Y., Zhang, Y., Xiao, X., & Wen, W. (2024). A privacy-preserving image retrieval scheme with access control based on searchable encryption in media cloud. Cybersecurity, 7(1), 22. doi: 10.1186/s42400-024-00213-z

Titus, A. J., Kishore, S., Stavish, T., Rogers, S. M., & Ni, K. (2018). PySEAL: A Python wrapper implementation of the SEAL homomorphic encryption library.

van Oorschot, P. C. (2022). Public Key Cryptography's Impact on Society: How Diffie and Hellman Changed the World. In Democratizing Cryptography (pp. 19–56). New York, NY, USA: ACM. doi: 10.1145/3549993.3549997

Wang, H., Han, X., Song, X., Su, J., Li, Y., Zheng, W., & Wu, X. (2024). Research on automatic pavement crack identification Based on improved YOLOv8. International Journal on Interactive Design and Manufacturing (IJIDeM), 18(6), 3773–3783. doi: 10.1007/s12008-024-01769-3

Wang, Y., Miao, M., Shen, J., & Wang, J. (2019). Towards efficient privacy-preserving encrypted image search in cloud computing. Soft Computing, 23(6), 2101–2112. doi: 10.1007/s00500-017-2927-6

Weger, V., Gassner, N., & Rosenthal, J. (2024). A Survey on Code-Based Cryptography.

Wei, F., Zeadally, S., Vijayakumar, P., Kumar, N., & He, D. (2021). An Intelligent Terminal Based Privacy-Preserving Multi-Modal Implicit Authentication Protocol for Internet of Connected Vehicles. IEEE Transactions on Intelligent Transportation Systems, 22(7), 3939–3951. doi: 10.1109/TITS.2020.2998775

Wu, H., Zhang, Y., Shen, Y., Jiang, X., & Taleb, T. (2024). Achieving Covertness and Secrecy: The Interplay Between Detection and Eavesdropping Attacks. IEEE Internet of Things Journal, 11(2), 3233–3249. doi: 10.1109/JIOT.2023.3296368

Wu, T., Zhao, C., & Zhang, Y.-J. A. (2021). Privacy-Preserving Distributed Optimal Power Flow With Partially Homomorphic Encryption. IEEE Transactions on Smart Grid, 12(5), 4506–4521. doi: 10.1109/TSG.2021.3084934

Xia, Z., Lu, L., Qiu, T., J. Shim, H., Chen, X., & Jeon, B. (2019). A Privacy-Preserving Image Retrieval Based on AC-Coefficients and Color Histograms in Cloud Environment. Computers, Materials & Continua, 58(1), 27–43. doi: 10.32604/cmc.2019.02688

Xia, Z., Xiong, N. N., Vasilakos, A. V., & Sun, X. (2017). EPCBIR: An efficient and privacy-preserving content-based image retrieval scheme in cloud computing. Information Sciences, 387, 195–204. doi: 10.1016/j.ins.2016.12.030

Xing, J., Liu, Y., & Zhang, G.-Z. (2023). Improved YOLOV5-Based UAV Pavement Crack Detection. IEEE Sensors Journal, 23(14), 15901–15909. doi: 10.1109/JSEN.2023.3281585

Y. Shen, & Siani Pearson. (2011, January). Privacy enhancing technologies: A review. Hewlet Packard Development Company.

Yang, H., Liu, Y., Wang, S., Qu, H., Li, N., Wu, J., Yan, Y., Zhang, H., Wang, J., & Qiu, J. (2023). Improved Apple Fruit Target Recognition Method Based on YOLOv7 Model. Agriculture, 13(7), 1278. doi: 10.3390/agriculture13071278

Yang, T., Ma, J., Wang, Q., Miao, Y., Wang, X., & Meng, Q. (2018). Image Feature Extraction in Encrypted Domain With Privacy-Preserving Hahn Moments. IEEE Access, 6, 47521–47534. doi: 10.1109/ACCESS.2018.2866861

Yang, W., Wang, S., Wu, D., Cai, T., Zhu, Y., Wei, S., Zhang, Y., Yang, X., Tang, Z., & Li, Y. (2025). Deep learning model inversion attacks and defenses: a comprehensive survey. Artificial Intelligence Review, 58(8), 242. doi: 10.1007/s10462-025-11248-0

Yi, X., Paulet, R., & Bertino, E. (2014). Homomorphic Encryption and Applications. Cham: Springer International Publishing. doi: 10.1007/978-3-319-12229-8

Yu, G., & Zhou, X. (2023). An Improved YOLOv5 Crack Detection Method Combined with a Bottleneck Transformer. Mathematics, 11(10), 2377. doi: 10.3390/math11102377

Zaidi, S. S. A., Ansari, M. S., Aslam, A., Kanwal, N., Asghar, M., & Lee, B. (2022). A survey of modern deep learning based object detection models. Digital Signal Processing, 126, 103514. doi: 10.1016/j.dsp.2022.103514

Zeadally, S., Guerrero, J., & Contreras, J. (2020). A tutorial survey on vehicle-to-vehicle communications. Telecommunication Systems, 73(3), 469–489. doi: 10.1007/s11235-019-00639-8

Zhang, C., Zhu, L., Zhang, S., & Yu, W. (2020). TDHPPIR: An Efficient Deep Hashing Based Privacy-Preserving Image Retrieval Method. Neurocomputing, 406, 386–398. doi: 10.1016/j.neucom.2019.11.119

Zhang, L., Jung, T., Liu, K., Li, X.-Y., Ding, X., Gu, J., & Liu, Y. (2017). PIC: Enable Large-Scale Privacy Preserving Content-Based Image Search on Cloud. IEEE Transactions on Parallel and Distributed Systems, 28(11), 3258–3271. doi: 10.1109/TPDS.2017.2712148

Z. Brakerski. (2012). Fully homomorphic encryption without modulus switching from classical GapSVP . In Advances in Cryptology—CRYPTO 2012, Berlin, Germany:Springer, 868–886.

Zhou, Q., Wang, Z., Zhong, Y., Zhong, F., & Wang, L. (2024). Efficient Optimized YOLOv8 Model with Extended Vision. Sensors, 24(20), 6506. doi: 10.3390/s24206506

Zikria, Y. Bin, Ali, R., Afzal, M. K., & Kim, S. W. (2021). Next-Generation Internet of Things (IoT): Opportunities, Challenges, and Solutions. Sensors, 21(4), 1174. doi: 10.3390/s21041174

Zou, Q., Wang, J., Ye, J., Shen, J., & Chen, X. (2017). Efficient and secure encrypted image search in mobile cloud computing. Soft Computing, 21(11), 2959–2969. doi: 10.1007/s00500-016-2153-7

ÖZGEÇMIŞ

Ad-Soyad : Kamran SAEED

ÖĞRENİM DURUMU:

- **Lisans** **:** 2015, University of Engineering and Technology, Taxila, Mühendislik Bilimleri Fakültesi, Elektrik Mühendisliği Bölümü.
- **Yükseklisans** **:** 2019, Bahria University, Islamabad, Pakistan, Mühendislik Bilimleri Fakültesi, Bilgisayar Mühendisliği Bölümü.

TEZDEN TÜRETİLEN ESERLER:

- K. Saeed and M. F. Adak. Secured Cloud-Based Image Data Processing of Self-Driving Vehicles Using Full Homomorphic Encryption. *Kuwait Journal of Science*, Vol. 52, No. 4, Oct 2025.
- K. Saeed and M. F. Adak. Enhancing Privacy in Image Search: Secure Object-Based Retrieval from Encrypted Images Using BFV Homomorphic Encryption. 2025 *International Conference for Artificial Intelligence, Applications, Innovation and Ethics* (AI2E), 2025