



**T.C.**  
**KONYA TEKNİK ÜNİVERSİTESİ**  
**LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**



**YOLO ALGORİTMASI İLE SCADA**  
**SİSTEMLERİNDE ERİŞİM DÜZEYİ**  
**KONTROLÜ**

İhsan Fırat Gülüm

**YÜKSEK LİSANS**

**Elektrik-Elektronik Mühendisliği Anabilim Dalı**

**KASIM-2024**  
**KONYA**

**Her Hakkı Saklıdır**

**TEZ KABUL VE ONAYI**

**İhsan Fırat GÜLÜM** tarafından hazırlanan “**YOLO Algoritması ile SCADA Sistemlerinde Erişim Düzeyi Kontrolü**” adlı tez çalışması 25/11/2024 tarihinde aşağıdaki jüri tarafından oy birliği ile Konya Teknik Üniversitesi Lisansüstü Eğitim Enstitüsü Elektrik-Elektronik Mühendisliği Ana bilim Dalı’nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

**Jüri Üyeleri**

**İmza**

**Başkan**

Dr. Öğr. Üyesi Sabri ALTUNKAYA

.....

**Danışman**

Prof. Dr. Ömer AYDOĞDU

.....

**Üye**

Doç. Dr. Akif DURDU

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Mevlüt Uyan  
Enstitü Müdürü

## TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

## DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

İhsan Fırat Gülüm

Tarih: 18.12.2024

## ÖZET

### YÜKSEK LİSANS

## YOLO ALGORİTMASI İLE SCADA SİSTEMLERİNDE ERİŞİM DÜZEYİ KONTROLÜ

**İhsan Fırat GÜLÜM**

**Konya Teknik Üniversitesi  
Lisansüstü Eğitim Enstitüsü  
Elektrik Elektronik Mühendisliği Anabilim Dalı**

**Danışman: Prof. Dr. Ömer AYDOĞDU**

**2024, 62 Sayfa**

Günümüz endüstriyel üretim sistemlerinde, kontrol ve otomasyon sistemlerindeki teknolojik gelişmeler neticesinde insan gücüne olan bağımlılık giderek azalmaktadır. Bu trend, üretim süreçlerinin verimliliğini artırırken, aynı zamanda operasyonel bağımsızlığı da güçlendirmektedir. Bununla birlikte fabrika ortamlarında, makine operatörlerinin etkin yönetimi ve denetimi, üretim süreçlerinin düzgün işleyişi açısından kritik öneme sahiptir.

Seri üretim yapan tesislerde, üretim hatlarının farklı ürünlere adaptasyonu sırasında, makine parametrelerinin operatörler tarafından, Merkezi Kontrol ve Veri Toplama Sistemi (SCADA) içerisinde bulunan İnsan-Makine Arayüzleri (HMI) üzerinden güncellenmesi gerekmektedir. Çoklu vardiya sistemlerinde çalışan operatörler arasında, bu parametre değişikliklerinin kimin tarafından yapıldığının tespiti zaman zaman zorluklara yol açabilmektedir. Geleneksel kullanıcı adı ve şifre tabanlı giriş sistemleri, operatörler arasında güvenlik riskleri ve operasyonel aksaklıklar doğurabilmektedir. Ayrıca artık günümüzde şifreler, insanların bilgilerini güvende tutmanın kusursuz bir yolu değildir. Microsoft her saniye yaklaşık 1.287 ya da günde yaklaşık 111 milyon parola saldırısı rapor etmektedir. Güvenlik için şifrelemede hem rakam hem de harf kullanımı önerilse de ikinci bir güvenlik katmanı olarak farklı kimlik doğrulama yöntemleri tavsiye edilmektedir.

Bu tez çalışması, yüz tanıma teknolojisini SCADA sistemlerinde bulunan HMI'lara entegre ederek, sistem girişlerinde operatörlerin kullanıcı kimlik doğrulama sürecini optimize etmeyi amaçlamaktadır. Önerilen sistem, operatörlerin sadece kameraya bakarak, yetkileri dahilindeki sayfalara erişimini sağlarken, aynı zamanda potansiyel kimlik sahteciliği girişimlerini de (örneğin, başka bir operatörün fotoğrafını kullanarak sisteme erişme teşebbüsü) tespit edip, alarm üretme ve kayıt altına alma özelliğine sahiptir.

Bu çalışmada, amaca yönelik olarak farklı bireylere ait yüzlerden oluşturulan kendi veri setimiz üzerinde eğitilen YOLOv8 modeli, 0,5 saniye tespit süresi ile yüzde 90 başarı oranına ulaşmıştır. Ek olarak yüz tanıma için kullanılan, LFW veri seti üzerinde yüzde 99,63 başarı oranına ulaşan FaceNet modeli uygulamamıza entegre edilmiş ve 0,6 saniye tespit süresi ile oldukça başarılı sonuçlar alınmıştır. Bu yaklaşım, sistem güvenliğinin artırılmasını, operatörlerin erişim süreçlerinin hızlandırılmasını ve

operasyonel kontrol düzeyinin yükseltilmesini hedeflemektedir. Tez kapsamında gerçekleştirilen denemeler, geliştirilen sistemin toplam 2 saniyeden az süren analiz süresinde (sahtecilik tespiti ve yüz analizi) yüzde 90 doğruluk oranıyla etkin bir güvenlik sağladığını göstermektedir. Endüstriyel otomasyon sistemleri için tasarlanan bu uygulamanın, diğer erişim kısıtlı endüstriyel alanlarda da etkili bir şekilde kullanılabileceği açıktır.

**Anahtar Kelimeler:** Erişim Kontrolü, HMI, Kontrol Kartı, PLC, SCADA, Yüz Tanıma



## **ABSTRACT**

**MS**

### **ACCESS LEVEL CONTROL IN SCADA SYSTEMS WITH YOLO ALGORITHM**

**İhsan Fırat GÜLÜM**

**Konya Technical University  
Institute of Graduate Studies  
Department of Electrical and Electronics Engineering**

**Advisor: Prof. Dr. Ömer AYDOĞDU**

**2024, 62 Pages**

In today's industrial production systems, dependence on manpower is gradually decreasing as a result of technological developments in control and automation systems. This trend increases the efficiency of production processes, while at the same time strengthening operational independence. In factory conditions, effective management and supervision of machine operators is critical for the appropriate functioning of production processes.

In mass production facilities, during the adaptation of production lines to different products, machine parameters must be updated by the operators via Human-Machine Interfaces (HMI) within the Supervisory Control and Data Acquisition (SCADA) systems. Among operators working in multi-shift systems, it can sometimes be difficult to determine who is making these parameter changes. Traditional username and password-based login systems can create security risks and operational disruptions among operators. And passwords are no longer a foolproof way to keep people's information secure. Microsoft reports about 1,287 password attacks every second, approximately 111 million per day. Although it is recommended to use both numbers and letters in encryption for security, different authentication methods are recommended as a second layer of security.

This thesis aims to optimize the user authentication process of operators at system logins by integrating face recognition technology into HMIs in SCADA systems. The proposed system enables operators to access authorized pages by simply looking at the camera, while at the same time detecting, alarming and recording potential identity fraud attempts (for example, an attempt to access the system using another operator's photo).

In this study, the YOLOv8 model, which was trained on our own dataset created from the faces of different individuals for the purpose, achieved a 90 percent success rate with a detection time of 0.5 seconds. In addition, the FaceNet model used for face recognition, which achieved a success rate of 99.63 percent on the LFW dataset, was integrated into our application and very successful results were obtained with a detection time of 0.6 seconds. This approach aims to increase system security, speed up the access processes of operators and increase the level of operational control. The tests conducted

within the scope of the thesis show that the developed system provides effective security with a 90 percent accuracy rate in a total analysis time of less than 2 seconds (spoofing detection and face analysis). It is clear that this application designed for industrial automation systems can be effectively used in other access-limited industrial areas.

**Keywords:** Access Control, Control Card, Face Recognition, HMI, PLC, SCADA



## ÖNSÖZ

Tez boyunca hem çalışmak hem de akademik çalışmalar yürütmek, pek çok fedakârlığı beraberinde getiren uzun ve zorlu bir süreç oldu. Bu süreçte, bilgilerini ve deneyimlerini bana cömertçe sunan, karşılaştığım problemlerde yardımlarını esirgemeyen ve zorluklarla başa çıkmamda her daim yanımda olan danışman hocama sonsuz teşekkür ederim.

Bu zorlu süreçte beni motive eden, her anlamda desteklerini hissettiren ve ilerlemem için cesaret veren aileme de çok minnettarım. Onların varlığı, bu yolculuğu daha katlanabilir ve anlamlı kıldı.

Bu yolculuğun, akademiye ve ülkemizin hızla büyüyen sanayi endüstrisine katkıda bulunmasını temenni ediyorum, çünkü ancak ülkemize katkı sağlayacak ve katma değer yaratacak bu tür çalışmalar, böylesine zorlu bir sürecin en değerli sonucu olabilir.

İhsan Fırat GÜLÜM  
KONYA-2024

# İÇİNDEKİLER

<b>ÖZET .....</b>	<b>iv</b>
<b>ABSTRACT.....</b>	<b>vi</b>
<b>ÖNSÖZ .....</b>	<b>viii</b>
<b>İÇİNDEKİLER .....</b>	<b>ix</b>
<b>KISALTMALAR .....</b>	<b>xi</b>
<b>1. GİRİŞ.....</b>	<b>1</b>
1.1. Tezin Amacı ve Önemi .....	4
<b>2. KAYNAK ARAŞTIRMASI .....</b>	<b>6</b>
2.1. Yüz Tespiti ve Sahte Yüz Tespiti ile İlgili Yapılan Çalışmalar .....	6
2.2. Yüz Tanımlama ile İlgili Yapılan Çalışmalar.....	9
<b>3. MATERYAL VE YÖNTEM.....</b>	<b>13</b>
3.1. SCADA Sistemi Bileşenleri.....	15
3.2. YOLO Modeli.....	18
3.2.1. mAP (ortalama doğrulukların ortalaması) .....	19
3.2.2. YOLOv8 .....	22
3.2.2.1. YOLOv8 mimarisi .....	22
3.2.2.1.1. Omurga (backbone) .....	23
3.2.2.1.2. Baş (head) .....	25
3.3. Yüz Tanımanın Temel Prensipleri .....	26
3.3.1. Yüz tespiti .....	26
3.3.2. Yüz hizalaması.....	26
3.3.3. Yüz temsili .....	27
3.3.4. Yüz doğrulanması .....	27
3.3.5. Yüz modeli performans karşılaştırılması .....	28
3.3.6. FaceNet modeli .....	31
3.4. SNAP7 Haberleşmesi .....	32
3.5. PROFINET Protokolü.....	33
3.6. Geliştirilen Sistem Arayüzü ve Kullanımı.....	34
<b>4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....</b>	<b>39</b>
4.1. YOLOv8 Eğitim Sonuçları .....	40
4.1.1. Karışıklık matrisi (Confusion matrix).....	42
4.1.2. F1 güven eğrisi (F1 confidence curve) .....	43
4.1.3. Doğruluk-güven eğrisi (Precision-confidence curve).....	43
4.1.4. Train/box_loss eğrisi.....	45
4.1.5. Train/cls_loss eğrisi .....	46
4.1.6. mAP50 ve mAP50-95 eğrisi .....	47

4.1.7. Gerçek ve sahte yüzler için yapılan test sonuçları .....	48
4.1.7. FaceNet modeli test sonuçları.....	51
<b>5. SONUÇLAR VE ÖNERİLER.....</b>	<b>52</b>
<b>KAYNAKLAR .....</b>	<b>55</b>
<b>EKLER .....</b>	<b>59</b>
<b>ÖZGEÇMİŞ .....</b>	<b>Hata! Yer işareti tanımlanmamış.</b>



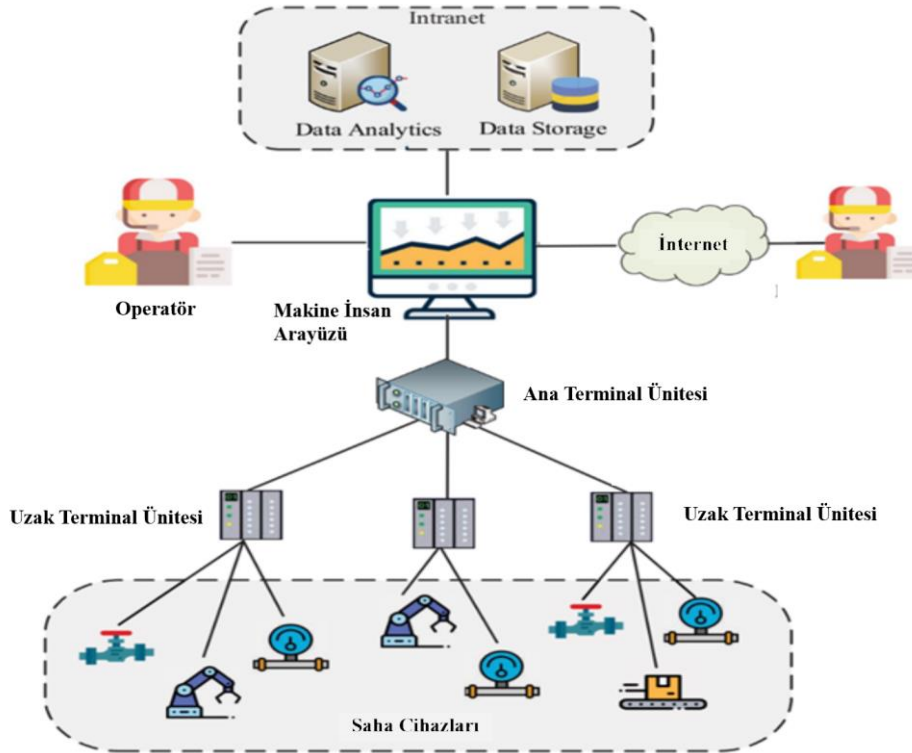
## KISALTMALAR

- AP : Ortalama Doğruluk  
CNN : Evrişimsel Sinir Ağları  
HMI : İnsan-Makine Arayüzü  
IoU : Kesişim Birleşim Oranı  
LDA : Doğrusal Ayırma Analizi  
LFW : Etiketlenmiş Yüzler Veri Tabanı  
mAP : Ortalama Doğrulukların Ortalaması  
PCA : Temel Bileşen Analizi  
PLC : Programlanabilir Mantıksal Denetleyici  
RTU: Uzak Uç Birimi  
RAM : Rastgele Erişimli Bellek  
SCADA: Merkezi Kontrol ve Veri Toplama Sistemi  
YOLO : Tek Seferde Algılama

## 1. GİRİŞ

SCADA (Merkezi Kontrol ve Veri Toplama) sistemleri, özellikle büyük ölçekli endüstriyel süreçlerde, izleme ve kontrol görevlerini yerine getiren teknoloji platformlarıdır. Bu sistemler, kritik altyapıların sorunsuz ve verimli çalışmasını sağlamak için çeşitli sensörlerden ve cihazlardan veri toplayarak, operatörlerin sistemin genel durumu hakkında bilgi sahibi olmasını sağlar. Enerji üretim tesisleri, telekomünikasyon ağları, ulaşım sistemleri, petrol ve doğalgaz boru hatları gibi çok çeşitli alanlarda kullanılır. SCADA sistemleri, gerçek zamanlı veri analizi ve otomatikleştirilmiş müdahaleler ile sistemde meydana gelebilecek aksaklıkları önceden tespit etmeye ve çözmeye yardımcı olur. Bu sayede hem operasyonel güvenlik artırılır hem de sistemin kesintisiz bir şekilde çalışması sağlanır.

Genel bir SCADA sistemi, HMI'lar, merkezi denetleyiciler, sensörler ve aktüatörler gibi bir dizi dağıtılmış saha cihazından oluşur. Kontrolör ve saha cihazları arasındaki veri alışverişi, endüstriyel uygulamalar için özel olarak geliştirilmiş belirli iletişim protokolleri ile sağlanır. Örnek bir SCADA Sistemi ve Bileşenleri (Pliatsios ve diğ., 2020) Şekil 1.1'de gösterilmiştir.



Şekil 1.1. Örnek bir SCADA sistemi

Şekil 1.1’de görülen HMI’lar, makineler ile operatörlerin iletişimini sağlayan arayüz ekranlarıdır. Endüstride sıklıkla kullanılan bu ekranlar, makinenin tüm kontrollerinin yapılabilmesine, parametrelerinin değiştirilebilmesine olanak sağlar. HMI’larda kontrol edilen çeşitli sayfalar vardır. Bunlara örnek olarak Ana Sayfa, Alarm Sayfası, Ayarlar Sayfası, Proses Kontrol Sayfası ve Reçete Sayfası gibi sayfalar örnek olarak verilebilir.

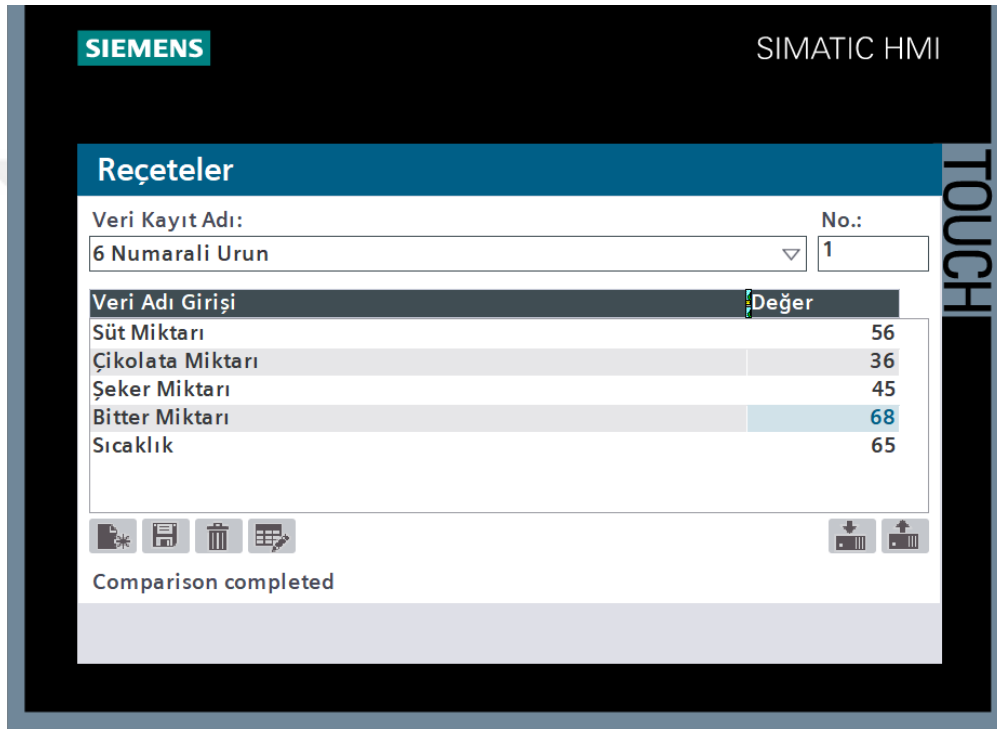
Büyük ölçekli fabrikalarda kullanılan fabrika otomasyonu sistemlerinde, kontrol arayüzüne erişimde farklı amaçlar için farklı kullanıcılar söz konusu olmaktadır. Her kullanıcının erişmesi ve müdahale etmesi gereken kısımlar farklıdır. Bu nedenle herhangi bir kullanıcının tüm sistem bilgisine ulaşması sakıncalı olabilmektedir. Sistem gizli reçete bilgileri içerebilir. Özellikle seri üretim yapan fabrikalarda, sıklıkla farklı reçeteleme sistemi kullanılır. Bu reçete sistemini anlamak için örneğin çikolata fabrikasındaki, çikolata üreten bir makineyi ele alalım. Bu makine de 10 çeşit çikolata üretebildiğinizi ve yine bu makinenin günde 2 vardiya çalıştığını düşünelim. Her çeşit çikolatanın kendine has bileşen oranları vardır ve bu bileşenler uzun deneme sürelerinden ve belli proseslerden geçtikten sonra tespit edilmiştir. Bu oranların kaydedildiği yere reçete sayfası denilmektedir. Yetkisiz bir vardiya çalışanı, vardiyasının sonuna yaklaşınca reçete sayfasındaki oranları yanlışlıkla değiştirdiğinde, bir sonraki gelen vardiyadaki operatör ciddi suçlamalar ile yüz yüze kalabilmektedir. Özellikle bu tarz fabrikalarda reçete sayfası ile reçete sayfasına erişim ciddi önem taşımaktadır.

Reçete sayfası gibi, her HMI’ın kendine has farklı kritik sayfaları olabilmektedir. Bu sayfalara erişim ataması yapılmaktadır. Makinenin elektrik aksanlarının manuel olarak kontrol edilebildiği sayfaya, elektrik teknikerinin kullanıcı adı tanımlanır ve diğer operatörler bu sayfalara erişemez. Günlük üretilen ürün miktarını veya makinenin günlük çalışma saatini gösteren sayfaya üretim mühendisinin adı tanımlanır. Böylece erişim düzeyleri ayrı ayrı atanmış olur. Farklı erişim seviyeleri oluşturabilmek için geleneksel olarak, her operatöre kullanıcı düzeyi ataması yapılır ve her kullanıcıya erişim için kullanabileceği şifre verilir. Ama operatörler şifrelerini bazen birbiri ile paylaşma durumunda kalabilmektedir veya şifrelerini unutup sürekli yeni bir şifre talebi oluşturmaktadır. Her dakikanın kıymetli olduğu seri üretim yapan fabrikalarda da bu tarz talepler vakit kaybına ve üretim verimliliğinin düşmesine sebebiyet vermektedir.

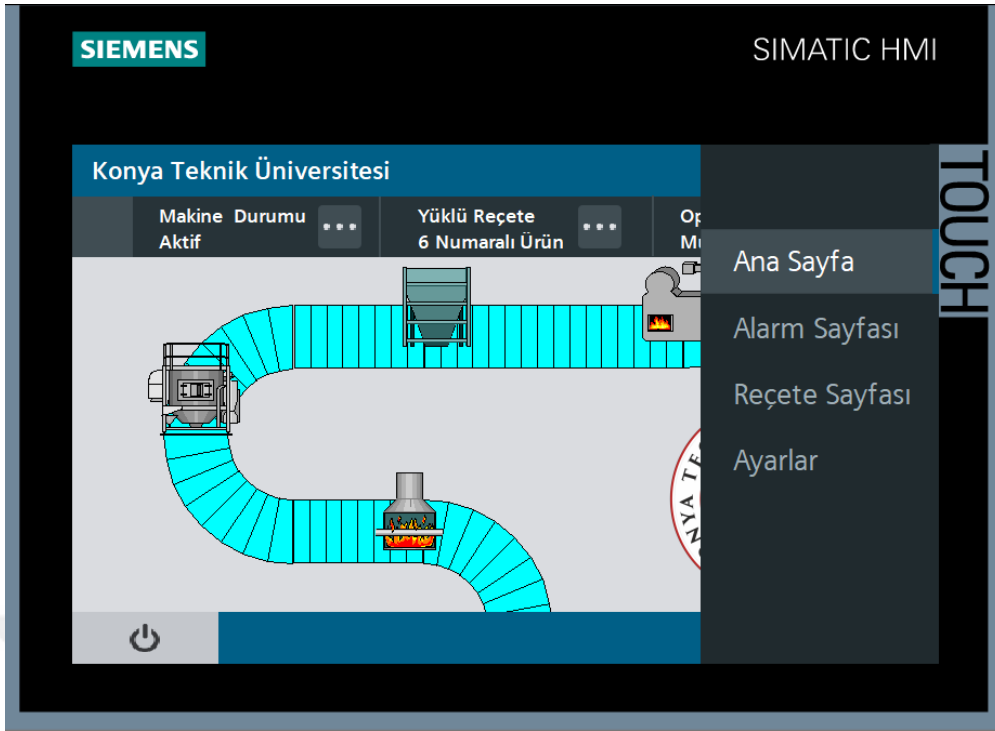
Parmak izi, yüz tanıma, ses tanıma ve yüz yapısı analizi gibi, modern güvenlik sistemlerinin etkinliğini artırmak için birçok güvenli yöntem mevcutken, hâlâ sıklıkla

kullanıcı adı ve parola gibi geleneksel güvenlik yöntemleri kullanılmaktadır. Bu durum, süreçleri yavaşlatmakta ve sistemin güvenilirliği ile sağlamlığını tehdit etmektedir. Dolayısıyla, daha etkili ve hızlı kimlik doğrulama yöntemlerinin entegrasyonu hem verimliliği artıracak hem de güvenlik seviyesini yükseltecek hem de zaman kayıplarını minimuma indirecektir.

Örnek bir reçete sayfası ve reçetenin kullanıldığı proses sayfası Şekil 1.2 ve Şekil 1.3'te gösterilmiştir.



Şekil 1.2. Reçete sayfası örneği



Şekil 1.3. Proses sayfası ve reçete sayfasına geçiş için kullanılan yan panel

### 1.1. Tezin Amacı ve Önemi

Bu çalışmanın temel hedefleri, Python programlama dili ile SCADA sistemlerinin entegrasyonunu güçlendirmek ve bu iş birliğini geniş kapsamlı bir şekilde artırarak endüstriyel otomasyon süreçlerinde verimlilik ve güvenliği iyileştirmektir. Çalışma kapsamında, SCADA sistemine kaydedilen operatörler için farklı erişim seviyeleri tanımlanarak yetkilendirme mekanizması güçlendirilecek ve kontrollü sayfa erişimi sağlanacaktır. Ayrıca, tanımlanmamış operatörlerin yetkisiz giriş denemeleri tespit edilerek alarm sistemleri devreye alınacak, operatör fotoğrafları veri tabanına kaydedilerek yetkili erişimler güvence altına alınacaktır.

Tez kapsamında, Python dili kullanılarak YOLO algoritmasıyla görüntü işleme yetenekleri SCADA sistemine entegre edilerek, Python aracılığıyla veri entegrasyonu sağlanacak ve bu sistem üzerinden erişim sağlayan operatörlerin aktiviteleri veri tabanında kaydedilecektir. Python ile PLC arasındaki iletişim kanalı genişletilerek endüstriyel otomasyon uygulamalarında daha etkin bir iş birliği sağlanıp, yapay zekâ tabanlı algoritmaların endüstriyel süreçlere entegrasyonu mümkün kılınacaktır.

Bu çalışmanın bir diğer hedefi, SCADA sistemlerinin güvenliğini artırmak, Python'un esnekliğini endüstriyel süreçlere entegre etmek ve yapay zekâ algoritmalarının

endüstriyel uygulamalardaki kullanımını kolaylaştırarak otomasyon alanında verimlilik, güvenlik ve iş birliği potansiyelini artırmaya yönelik önemli katkılar sunmaktır. Tezin sonuçları doğrultusunda, gelecekte yapılacak çalışmalarda Python uygulamaları ile SCADA sistemleri arasında sağlanan iletişim sayesinde, seri üretim yapılan fabrikalarda anlık üretim verilerinin SCADA sistemine iletilmesi ve makine öğrenmesi yöntemleri kullanılarak bu verilerin analiz edilmesi mümkün olacaktır. Bu sayede, operatörlere ve mühendislere üretim verimliliği hakkında bilgi sunulacak, olası verim düşüşleri tespit edilerek çözüm önerileri geliştirilebilecektir. Örneğin, verimliliğin düşmeye başladığı anlaşıldığında, ilgili mühendise uyarı mesajı iletilip, düşüşün muhtemel sebepleri ve alınması gereken önlemler sunulabilecektir.

Özetle bu tez çalışmasında, endüstride kullanılan makinelerde sıklıkla karşılaşılan, kullanıcının adı ve şifresi ile girilen geleneksel metottan farklı olarak, yüz tanıma sisteminin endüstriye kazandırılması hedeflenmiştir. Daha hızlı ve daha dengeli çalışmakta olan bu sistem ile parola unutulmalarının, parola paylaşımlarının, erişim düzeyi karmaşıklıklarının önüne geçilmiş olacak, buna ek olarak erişim izni olup sisteme giren operatörlerin hangi saatlerde sisteme girdiği anlaşılacak, erişim izni olmayıp girmeye çalışanların veya erişim izni olan operatörlerin fotoğrafını telefonlarından açıp kameraya göstererek girmeye çalışan operatörlerin fotoğrafları çekilip, usulsüz girmeye çalışan operatörlerin fotoğraflarının kaydedildiği klasöre kaydedilecektir. Bu tarz bir aldatmanın önüne geçmek için YOLOv8 algoritması kullanırken, yüz tanıma için de FaceNet kütüphanesi kullanılacaktır. Böylelikle çok daha şeffaf, kontrol altında tutulan operatör takip sistemi yapılmış olacaktır.

Tez çalışmasının ilerleyen bölümlerinde, öncelikle kapsamlı bir kaynak araştırması yapılarak literatürde yer alan benzer çalışmalar incelenecek ve bu çalışmaların özetleri ile elde ettikleri sonuçlar detaylıca açıklanacaktır. Literatürde kullanılan modeller karşılaştırılacak ve modellerin avantajları ve dezavantajları sebepleri ile açıklanacaktır.

Literatür bölümünün ardından Materyal ve Yöntem kısmında, tez çalışmamızda kullanılan malzemeler, kullanılan yaklaşımlar ve yöntemler detaylı bir şekilde ele alınacaktır. Bu bölümde, her bir malzemenin nasıl kullanıldığı, işlevleri açıklanarak tez çalışmasının temel yapısı ortaya konulacaktır.

Son bölüm olan Araştırma Sonuçları ve Tartışma kısmında ise, çalışmanın bulguları grafikler ve görüntüler eşliğinde ayrıntılı olarak değerlendirilecek, elde edilen sonuçlar analiz edilerek sistemin performansı ve çalışması hakkında kapsamlı bir değerlendirme yapılacaktır.

## 2. KAYNAK ARAŞTIRMASI

Yapılan çalışma da ilk etapta yüz tespiti ve tespit edilen yüzün sahte mi gerçek mi olduğunun analizi ardından da yüze tanımlama işlemi uygulandığından bu aşamalarda yapılan literatür çalışmaları ayrı ayrı başlıklar altında incelenmiştir.

Aşağıda, literatürde taranan önemli çalışmaların kısa özetleri verilmiştir. Güncel literatür çalışmalarında önerilen yaklaşımın avantajları ve dezavantajları vurgulanmıştır.

### 2.1. Yüz Tespiti ve Sahte Yüz Tespiti ile İlgili Yapılan Çalışmalar

Rehman ve diğerleri (2017), CASIA-FASD veri setinde bulunan 24.000 fotoğraf kullanarak VGG-11 mimarisine dayalı bir Evrişimsel Sinir Ağı modeli olan SpoofNet'i geliştirmişlerdir. Çalışmalarında, aşırı öğrenmeyi minimize etmek ve modelin genel performansını artırmak amacıyla 50RS-30SeC-1E adlı bir eğitim stratejisi sunmuşlardır. Bu strateji, her bir ileri geçişte CNN'e rastgele olarak seçilen 50 görüntünün sunulmasını sağlar. Bu işlem, modelin her seferinde farklı bir veri alt kümesi üzerinde öğrenme yapmasına olanak tanır. Ayrıca, 30 kez bu ileri geçişler yapıldığında bir epoch (eğitim döngüsü) tamamlanmış olur. Bu yaklaşım, modelin daha geniş bir veri çeşitliliği görmesini sağlayarak aşırı öğrenmeyi önler ve genel performansı artırır. Rehman ve diğerleri, bu yöntemle %89 doğruluk oranına ulaşmışlardır.

Alhanea ve diğerleri (2021), yaptıkları çalışmada derin öğrenme kullanarak bir yüz tanıma çalışması için GoogleNet modeli üzerinde transfer öğrenme yöntemi uygulamışlardır. 200 yüz görüntüsünden oluşan veri seti kullanılarak gerçekleştirilen bu çalışmada, %93,3'lük bir doğruluk oranına ulaşmışlardır. Bununla birlikte, model yalnızca yüz tanıma işlevi için tasarlanmış olup, herhangi bir sahtecilik tespiti (anti-spoofing) yöntemi entegre edilmemiştir. Bu durum, sistemin sahte yüzlerle (örneğin fotoğraf veya video) kolayca yanıltılabileceği anlamına gelir.

Alshaikhli ve diğerleri (2021), Face-Fake-Net adını verdikleri yüz sahteciliğini tespit etmek için yaptıkları çalışmalarında, ResNet-50 mimarisini temel alan ve uzamsal ile kanal-bazlı dikkat modüllerini entegre eden bir derin öğrenme modeli geliştirmişlerdir. Çalışmada model, CASIA-SURF (16.000 eğitim, 2.000 test) ve CelebA-Spoof (4.500 eğitim, 500 test) veri setleri kullanılarak test edilmiş ve sırasıyla %99.7 ve %99.4 doğruluk oranlarıyla başarılı sonuçlar elde etmiştir. Ancak çalışmada, modelde ulaşılan sahtecilik tespitinin hızına dair herhangi bir bilgiye yer verilmemiştir.

Aung ve diğerleri (2021), yüz tespiti için YOLOv2'yi VGG16 ön eğitilmiş evrişimsel sinir ağı ile birleştirerek, 5000 fotoğraftan oluşan ve her fotoğrafta birden fazla insanın olduğu veri seti üzerinde kullanarak Çizelge 2.1'deki sonuçları almıştır. Deneysel sonuçlar, önerilen yöntemin test görüntü setini %93 ortalama hassasiyetle tespit ettiğini göstermektedir. Ayrıca, önerilen yöntem gerçek zamanlı canlı videolarda yüz algılama hızını önemli ölçüde artırmıştır. Bu çalışma ile önerilen yöntemin veri setinin ön işleme tarafında kullanılmasını önermişlerdir.

**Çizelge 2.1.** YOLOv2 ile kombine edilen modellerin performans çizelgesi

	<i>Training time</i>	<i>Average precision on test image set</i>
YOLOv2+vgg16 with adam solver	5 hours 54 minutes	0.86 %
YOLOv2+vgg16 with sgd solver	5 hours 10 minutes	0.98 %

	<i>Accuracy</i>
YOLOv2+Alex network model	0.85 %
YOLOv2+Google network model	0.87 %
Our proposed model (YOLOv2+VGG16 network model)	0.93 %

Li ve diğerleri (2021), Stereo Eşleştirme ile Yüz Sahteciliği Önleme (Stereo Matching FAS) adını verdikleri yöntem ile kullanıcılardan iki farklı açıdan (sol ve sağ) yüz görüntüsü alınarak stereo eşleştirme işlemi yapılmaktadır. Bu işlem, yüzün daha doğru bir oranda temsil edilmesini ve derinlik bilgisinin elde edilmesini sağlamaktadır. Stereo eşleştirme ağı, disparite haritası adı verilen bir derinlik haritası oluşturur ve bu harita sınıflandırma ağına girdi olarak verilerek yüzün gerçek mi yoksa sahte mi olduğu belirlenir. Çalışmada kullanılan model, 140 gerçek ve 350 sahte yüz görüntüsünden oluşan, toplamda 114.231 görüntü içeren bir veri kümesi üzerinde eğitilmiştir. Bu yaklaşım, analiz süresini iki farklı açıdan görüntü alımına bağlı olarak uzatmaktadır.

Mun ve diğerleri (2022), ziyaretçi doğrulama amacıyla Tiny-YOLOv3 algoritmasını kullanarak yüz tespiti ve tanımlaması gerçekleştirmişlerdir. Çalışmalarında, 2.000 fotoğraftan oluşan bir veri seti üzerinde eğitim yapmışlar ve elde ettikleri model yaklaşık %86,3 doğruluk oranı ile 0.15 saniye gibi kısa bir sürede sonuç alınmıştır. Bu sonuçlar, yöntemin hız açısından verimli olduğunu ortaya koymaktadır. Ancak, çalışma

yalnızca yüz tespiti ve tanımlaması üzerine odaklanmış olup, anti-spoofing gibi sahteciliği önlemeye yönelik güvenlik önlemleri içermemektedir.

Ghimire ve diğerleri (2022), geniş bir yüz veri kümesi üzerinde yüz algılama algoritmalarının değerlendirilmesine yönelik çalışmaları kapsamında, küçük model kategorisinde en az FLOPS'a sahip olan (teorik olarak en hızlı olan algoritma) FaceBoxes algoritmasının, kolay veri setlerinde hassasiyetinin düşük olması nedeniyle sağlamlık sağlanamadığını ve bu nedenle uygun olmadığını göstermişlerdir. Bunun sonucunda, YOLOv5 algoritmasının hassasiyet ve hız açısından en sağlam model olduğunu belirtmişlerdir. Çizelge 2.2'deki sonuçlarına göre, YOLOv5 küçük veri setlerinde daha yüksek performans sergilemiş ve sistem entegrasyonu açısından da kolaylık sağlamıştır.

**Çizelge 2.2.** Modellerin performansının değerlendirildiği çizelge

Model Type	Detector	Performance		Parameters(M)	Flops(G)
		Easy	Hard		
Large	MTCNN	85.1	60.7	-	-
	DSFD	94.3	71.4	120.06	259.55
	TinaFace	95.6	81.4	37.98	172.95
	SCRFD-34GF	96.1	85.3	9.8	34.13
	Yolo-V5l	95.9	84.5	46.63	41.61
	SCRFD-2.5GF	93.8	77.9	0.67	2.53
Small	SCRFD-0.5GF	90.6	68.5	0.57	0.51
	RetinaFace	87.8	47.3	0.44	0.80
	FaceBoxes	76.2	24.2	1.01	0.28
	Yolo-V5n	93.6	80.5	1.73	2.11
	Yolo-V5n0.5	90.8	73.8	0.44	0.57

Tamang ve diğerleri, (2023) yüz tanıma üzerine yaptıkları çalışmada Yolov5 ve Yolov8 için aynı sayıda (2079) veri içeren veri setinde 200 epok ile eğitildiğinde Yolov5 için 0.85 mAP, Yolov8 için ise 0.94 mAP sonucunu almıştır. Bu çalışma, aynı şartlar altında test edilen YOLOv8'in YOLOv5'den daha başarılı sonuçlar ortaya koyduğunu göstermektedir.

Jeyaraj ve diğerleri (2024) tarafından geliştirilen YSAF (Yolo-Spatial Attention-FFT) yöntemi, 12000 fotoğraftan oluşan veri seti üzerinde uygulanarak mAP50 skoru için 0.994, recall skoru için 0.991 ve mAP skoru için 0.97 gibi dikkate değer sonuçlar elde etmiştir. Yazarlar, 25 yinelemeli (epoches) bir eğitim süreci benimsemişlerdir, ancak daha uzun bir eğitim sürecine neden başvurmadıkları konusunda detaylı bilgi vermemişlerdir. Önerilen YSAF yöntemi, Spatial Attention(Mekansal Dikkat), FFT (Hızlı Fourier Dönüşümü) ve YOLOv8 olmak üzere üç temel aşamadan oluşmaktadır. Mekânsal Dikkat ile girdi görüntülerindeki ilgili yüz özelliklerine (örneğin, gözler, burun,

ağız) odaklanarak, gereksiz bilgileri filtreler. FFT ile görüntüleri frekans domeninde analiz ederek gerçek yüzler ile sahte yüzler arasında ayırım yapmayı sağlayan desenleri ortaya çıkarır. YOLOv8 ile de Mekânsal dikkat ve FFT'den elde edilen bilgileri birleştirerek yüzleri tespit ederek gerçek veya sahte olarak sınıflandırır.

## 2.2. Yüz Tanımlama ile İlgili Yapılan Çalışmalar

Aşağıda, yüz tanımlama işlemi için literatürde taranan önemli çalışmaların kısa özetleri verilmiştir ve görülecektir ki yüz tanıma modelleri temelde üç kısma ayrılır. Manuel olarak elle hazırlanmış modeller, PCA ve LDA tabanlı özellik çıkarımına dayanarak öğrenme tabanlı modeller ve son yıllarda sıklıkla karşımıza çıkan CNN tabanlı derin öğrenme yoluyla oluşturulan modeller. Güncel literatür çalışmalarında önerilen yaklaşımın avantajları ve dezavantajları vurgulanmıştır.

Kumar ve diğerleri (2009), geliştirmiş oldukları, bir yüzün belirli bölgelerini (örneğin, gözler, burun, ağız) referans alınan kişilerin aynı bölgelerine olan benzerliklerine göre analiz eden Simile Classifiers (Benzetme Sınıflandırıcıları) yöntemi ile LFW veri seti üzerinde yüzde 84,14 doğruluk oranına erişmişlerdir. Kullandıkları bu yöntem yüzün farklı bölgelerinin belirli referans kişilere ne kadar benzediğini ölçerek kimlik doğrulaması yapar ve bu sayede iki görüntünün aynı kişiye ait olup olmadığını belirler.

Yin ve diğerleri (2011), İlişkendir- Tahmin Et (AP) ismini verdikleri modelleri ile yüzü bir bütün olarak ele almak yerine, yüzün önemli kısımlarını kapsayacak şekilde 12 parçaya ayırıp, bu parçaları eşleştirerek yüzlerin aynı kişiye ait olup olmadığına karar vermektedirler. Geliştirmiş oldukları AP modelini LFW veri setinde test etmişlerdir ve yüzde 90,57 başarı oranına erişmişlerdir.

Prince ve diğerleri (2012), Olasılıksal Doğrusal Ayrım Analizi (PLDA) ismiyle, yüz görüntülerindeki kimlik bilgilerini gizli değişkenler üzerinden modelleyen ve bayesyen bir yaklaşımla belirsizliği hesaba katan bir yöntem sunmuşlardır. Sınıf içi ve sınıflar arası varyansı Gauss dağılımlarıyla modelleyerek oluşturdukları bu yöntemi LFW veri seti üzerinde test etmişlerdir ve yüzde 87,3 başarı oranına erişmişlerdir.

Huang ve diğerleri (2012), yüz tanıma işlemi için yaptıkları iki aşamadan oluşan modelin ilk aşamasında yüz görüntüsünden en önemli özellikler çıkartılıyor ve ikinci aşamada bu özellikler derinlemesine karşılaştırılıyor ve yüzler arasındaki benzerlik

oranları bulunuyor. Yapmış oldukları bu model ile LFW veri seti üzerinde yüzde 92,58 başarı oranına ulaşmışlardır.

Simonyan ve diğerleri (2013), yüz tanıma işlemi için yaptıkları yöntem, yoğun örneklem alınmış Ölçek Değişmez Özellik Dönüşümü (SIFT) özelliklerinin Gauss Karışım Modeli (GMM) ile kodlanarak yüksek boyutlu Fisher vektörlerinden oluşmaktadır. Bu vektörler, Temel Bileşen Analizi (PCA) ile boyut indirgeme ve mekansal bilgi entegrasyonu ile zenginleştirilmiş, ardından normalizasyon işlemleri uygulanarak daha etkili ve kompakt yüz temsilleri elde etmişlerdir. Yapmış oldukları bu yöntem ile LFW veri seti üzerinde yüzde 93,03 başarı oranına ulaşmışlardır.

Lei ve diğerleri (2014), DFD (Discriminant Face Descriptor- Ayrımcı Yüz Tanımlayıcı) yöntemini geliştirerek, 5749 kişiye ait farklı pozlar ve aydınlatma koşulları altında çekilen 13.233 görüntüden oluşan Doğadaki Etiketli Yüzler (LFW) veri seti üzerinde test etmişlerdir ve yaklaşık yüzde 83,13 doğruluk oranına erişmişlerdir. 3 temel aşamadan oluşan DFD modelinin ilk aşaması ile filtrelenmiş görüntülerde aynı kişiye ait pikseller arasındaki farklar azaltılırken, farklı kişilere ait pikseller arasındaki farklar artırılır. İkinci aşaması ile pikseller arasındaki farklılıkların önemine göre ağırlıklandırma yaparak, yüz tanıma için daha anlamlı ve ayrımcı özelliklerin çıkarılmasını sağlar. Son aşaması ile de denetimsiz(unsupervised) bir öğrenme yöntemi kullanılarak, yüz görüntülerinde en çok rastlanan ve yüz tanımadaki kullanılacak en önemli özelliklerin oluşturulmasında kullanılacak, en anlamlı desenler belirlenir.

Lu ve Tang (2014), Gaussian Süreçleri (GPs) modelinden oluşan, Ayrımcı Gaussian Süreci Gizli Değişken Modelini (DGPLVM) temel alarak Gaussian Face modelini oluşturmuşlardır. DGPLVM modeli GPs modelinde kullanılan olasılıksal modelleme yeteneklerini kullanarak, verilerin gizli bir uzayda temsil edilerek, iki yüzün ortak ve farklı noktalarını tespit etmekte yardımcı olur. Geliştirdikleri Gaussian Face modelini LFW veri setinde test etmişlerdir ve yüzde 98,52 başarı oranına erişmişlerdir.

Taigman ve Yang (2014), DeepFace adını verdikleri modelde, önce yüz üzerindeki belirli noktalardan bir 2D model oluşturmuş, ardından bu modele daha detaylı noktalar ekleyerek 3D bir model elde etmişlerdir. Bu 3D modeli kullanarak Derin Sinir Ağları ile yüz özelliklerini çıkarmış ve yüz tanıma ile doğrulama işlemlerini gerçekleştirmişlerdir. Bu yaklaşımla, LFW veri setinde yüzde 97,35 başarı oranı elde etmişlerdir.

Schroff ve diğerleri (2015), FaceNet adını verdikleri modelde, yüz tanıma için doğrudan öklidyen bir uzaya gömü (embedding) sağlayan yenilikçi bir derin öğrenme

yaklaşımı sunmuşlardır. Bu model, CNN mimarisini kullanarak, her yüzü 128 boyutlu bir vektörle kompakt bir şekilde temsil etmektedir. FaceNet, yüzlerin benzerliklerini ve farklılıklarını bu gömüler arasındaki öklidyen mesafeler üzerinden değerlendirir; aynı kişiye ait yüzlerin mesafesini minimize ederken, farklı kişilere ait yüzlerin mesafesini maksimize edecek şekilde eğitilmiştir. Model, LFW veri setinde test edilerek yüzde 99,63 başarı oranına erişmiştir.

Parkhi ve diğerleri (2015), Çizelge 2.3 de verilen 11 bloktan oluşan CNN mimarisini kullanarak eğittiği modelini, LFW veri setinde test etmişlerdir ve yüzde 98,95 başarı oranına erişmişlerdir.

**Çizelge 2.3.** Parkhi ve diğerlerinin geliştirdikleri modelin CNN mimarisi

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
input	conv	relu	conv	relu	mpool	conv	relu	conv	relu	mpool	conv	relu	conv	relu	conv	relu	mpool	conv
-	conv1_1	relu1_1	conv1_2	relu1_2	pool1	conv2_1	relu2_1	conv2_2	relu2_2	pool2	conv3_1	relu3_1	conv3_2	relu3_2	conv3_3	relu3_3	pool3	conv4_1
-	3	1	3	1	2	3	1	3	1	2	3	1	3	1	3	1	2	3
-	3	-	64	-	-	64	-	128	-	-	128	-	256	-	256	-	-	256
-	64	-	64	-	-	128	-	128	-	-	256	-	256	-	256	-	-	512
-	1	1	1	1	2	1	1	1	1	2	1	1	1	1	1	1	2	1
-	1	0	1	0	0	1	0	1	0	0	1	0	1	0	1	0	0	1
19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37
relu	conv	relu	conv	relu	mpool	conv	relu	conv	relu	conv	relu	mpool	conv	relu	conv	relu	conv	softmax
relu4_1	conv4_2	relu4_2	conv4_3	relu4_3	pool4	conv5_1	relu5_1	conv5_2	relu5_2	conv5_3	relu5_3	pool5	fc6	relu6	fc7	relu7	fc8	prob
1	3	1	3	1	2	3	1	3	1	3	1	2	7	1	1	1	1	1
-	512	-	512	-	-	512	-	512	-	512	-	-	512	-	4096	-	4096	-
-	512	-	512	-	-	512	-	512	-	512	-	-	4096	-	4096	-	2622	-
1	1	1	1	1	2	1	1	1	1	1	2	1	1	1	1	1	1	1
0	1	0	1	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0

Basaran ve diğerleri (2018), Çok Ölçekli Yerel Zernike Momentleri (MS-LZM) ismini verdikleri özellik çıkarımına dayanan modellerinde, yerel Zernike momentlerinden (LZM) yararlanmışlardır. Çoklu ölçeklerdeki yüz özelliklerini yakalamak için görüntü piramidi kullandıkları bu modellerini LFW veri setinde test etmişlerdir ve yüzde 95,15 başarı oranına erişmişlerdir.

Deng ve diğerleri (2022), ArcFace adını verdikleri modellerinde, derin evrimsel sinir ağları ile yüz özelliklerini çıkararak, bu özellikleri 512 boyutlu bir vektöre gömmektedir. Özellikler, normalleştirilmiş bir hiperkürede açısal mesafeler yöntemi ile karşılaştırılarak, yüzlerin hangi sınıfa ait olduğu belirlenir. Yöntemin güçlü ayırt edici özellik çıkarma yeteneği sayesinde, LFW veri setinde yüzde 99,53 başarı oranına erişmişlerdir.

Sonuç olarak, incelenen tüm modellerin LFW veri seti üzerindeki doğruluk oranları Çizelge 2.4'te listelenmiştir. Çizelgeden de anlaşılacağı üzere, Schroff ve diğerleri tarafından yapılan FaceNet kütüphanesi en yüksek başarı oranına sahip yüz tanıma modelidir.

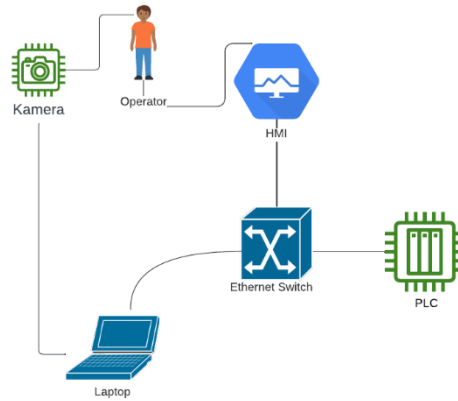
**Çizelge 2.4.** Yüz tanıma için geliştirilen modellerin çizelge gösterimi

<b>Yazar</b>	<b>Doğruluk (%)</b>
Kumar ve diğerleri (2009)	84,14
Yin ve diğerleri (2011)	90,57
Prince ve diğerleri (2012)	87,3
Huang ve diğerleri (2012)	92,58
Simonyan ve diğerleri (2013)	93,03
Lei ve diğerleri (2014)	83,13
Lu ve Tang (2014)	98,52
Taigman ve Yang (2014)	97,35
Parkhi ve diğerleri (2015)	98,95
Schroff ve diğerleri (2015)	99,63
Basaran ve diğerleri (2018)	95,15
Deng ve diğerleri (2022)	99,53

### 3. MATERYAL VE YÖNTEM

SCADA sistemleri, geniş bir alana yayılmış tesislerin tek bir merkezden izlenebildiği ve kontrol edilebildiği sistemlerdir. Bu sistemler temel olarak donanım bileşenleri ve yazılımdan oluşur. Tek bir cihazla kullanılabileceği gibi ağ bağlantılarıyla birden fazla bilgisayar ve taşınabilir cihazla kontrol ve izleme yapılabilir. Bu sistemlerde genellikle, birbirleriyle ethernet kabloları ile haberleşen PLC, HMI ve Bilgisayar ile sistem kurulup, görüntü işleme süreci ve operatör erişiminin kontrolü gibi yüksek hafıza isteyen işlemler bilgisayar tarafında yürütülüp, sadece gerekli bilgiler PLC'ye gönderilerek PLC ve HMI'nın daha efektif kullanılması sağlanır. Bu sistemlerde, HMI ve PLC'ler kendi aralarında profinet haberleşme altyapısını kullanırken, PLC ve bilgisayar ise SNAP7 gibi haberleşme protokollerini kullanır.

Tez çalışmasında uygulama amacıyla temel bileşenlere sahip bir SCADA sistemi tasarlanmış ve önerilen yaklaşımlar bu sistem üzerinde denenmiştir. PLC olarak proses otomasyonunda sıkça kullanılan Siemens'in S7-1200 serisinden CPU-1212 kullanılırken, HMI tarafında TP-900 Comfort Panel kullanılmıştır. Bilgisayar olarak 8 Gb Ram'e sahip olan HP-Envy marka dizüstü bilgisayar kullanılmıştır. Bu bilgisayarın yerine Jetson Nano Geliştirici Kitleri veya Raspberry Pi Kontrol kartları da kullanılabilir. Bu tez için hazırlanan sistemin topolojisi Şekil 3.1'de gösterilmiştir.



Şekil 3.1. Sistemin genel görünümü

Geliştirilen uygulamada; operatör HMI üzerinden Reçete sayfasına erişmek istediğinde, öncelikle Login Sayfasına erişip kameraya bakması gerekmektedir. Bilgisayar tarafından, kameradan çekilen fotoğraf alınıp işlendikten sonra, girmeye

çalışan operatörün erişim izninin olup olmadığı kontrol edilerek, işlem sonucu PLC tarafından sisteme gönderilir. Girmeye çalışan operatörün erişim izni varsa olumlu cevap ile operatörün şahsi bilgilerini de PLC'ye gönderir. HMI'dan gelen cevaba göre operatörün istenilen sayfaya geçip geçmeyeceğine karar verir. Ayrıca sisteme hangi operatörün hangi saatlerde giriş yaptığını kayıt altında tutmak amacıyla çekilen fotoğraf, erişim izni olup giren operatörlerin kaydedildiği sayfaya eklenir. Böylelikle üretimde çıkan bir sıkıntı da hangi operatörün ne zaman değişiklik yaptığının bilgisine bakılarak, kısa sürede çözülecektir.

Sistemin çalışır haldeki görünümü de Şekil 3.2'de gösterilmiştir. Şekilde de görüleceği üzere kullanıcı kamerası yüzünü gösterdikten sonra, yüzün veri tabanında olup olmadığı kontrol edilmekte, yüzüz 0,6 saniye içerisinde veri tabanında bulunmasının (monitör de bulunan Python uygulamasının sağ alt tarafında görülmektedir) ardından kullanıcının HMI'da bulunan Reçete Ekranına erişebileceği "Giriş" butonu ekranda görünür hale gelmektedir. Kullanıcı "Giriş" butonu ile Reçete ekranına erişebilmektedir.



Şekil 3.2. Yapılan uygulamanın gösterimi

Girmeye çalışan operatörün giriş izni yoksa olumsuz cevap ile operatörün çekilen fotoğrafı, erişim izni olmayıp girmeye çalışanların toplandığı klasöre kaydedilir. Ayrıca

yetkisiz bir operatör yetkili bir operatörün fotoğrafını kullanarak sistemi yanıltarak girmeye çalışırsa sistem alarm verir, bu şekilde girmeye çalışan operatörün fotoğrafını sisteme kaydeder. Bilgisayar tarafına gelen görüntülerin, YOLOv8 modeli ile gerçek olup olmadığı tespit edilir. Operatör vesikalık fotoğraf ya da telefonundan açtığı fotoğraf ile sistemi yanıltmaya çalışırsa YOLOv8 algoritması yardımıyla bu durum fark edilir ve hem alarm verilir hem de gösterilen fotoğraflar usulsüz fotoğrafların kaydedildiği klasöre eklenir. Operatör canlılık testinden geçerse, FaceNet modeli ile yüzün hangi operatöre ait olduğu tespit edilir. Bu algoritmayı kullanabilmek için operatörlere ait bir adet fotoğrafın sisteme kaydedilmesi gerekmektedir. FaceNet modeli, operatörün yüzünü sistemde bulursa operatörün isim bilgileri PLC'ye gönderilir ve yüz canlılık ve tespit yöntemi tamamlanmış olur.

### **3.1. SCADA Sistemi Bileşenleri**

SCADA sistemleri, günümüzde enerji üretim tesislerinde, su depolama ve nakil işlemlerinde, doğalgaz dağıtım hatlarında, ulaşım ve inşaat sektöründe, güvenlik ve bankacılık sistemlerinde, yem üretimi ve iklimlendirme gibi çeşitli alanlarda uygulanmaktadır. SCADA sistemleri, karmaşık ve geniş ölçekli operasyonların verimli bir şekilde yürütülmesini sağlayan bileşenler topluluğudur. Bu bileşenler şunlardır.

#### **3.1.1. PLC (programlanabilir mantıksal denetleyici)**

PLC'ler, endüstriyel kontrol uygulamalarının merkezi bileşenleridir. PLC'ler sahadaki sensörlerden gelen verileri işlemek, işlenen verilere göre diğer bileşenler üzerinde kontrol sağlamak için kullanılır. Ayrıca PLC'ler, motor, aktüatörler, pompalar gibi bileşenleri kontrol eden denetleyicilerdir. PLC'ler, SCADA sistemine operasyonların gerçek zamanlı durumunu bildirir ve merkezi kontrol sağlar.

#### **3.1.2. RTU (uzak uç birimi)**

RTU'lar, uzak lokasyonlarda bulunan cihazlar ile SCADA sisteminin merkezi arasında veri alışverişini sağlayan ünitelerdir. RTU'lar, sahadaki fiziksel değişkenlerin ölçümünü yapar ve bu verileri merkezi sisteme ileterek sahadaki operasyonların izlenmesini sağlar.

### 3.1.3. Sensörler

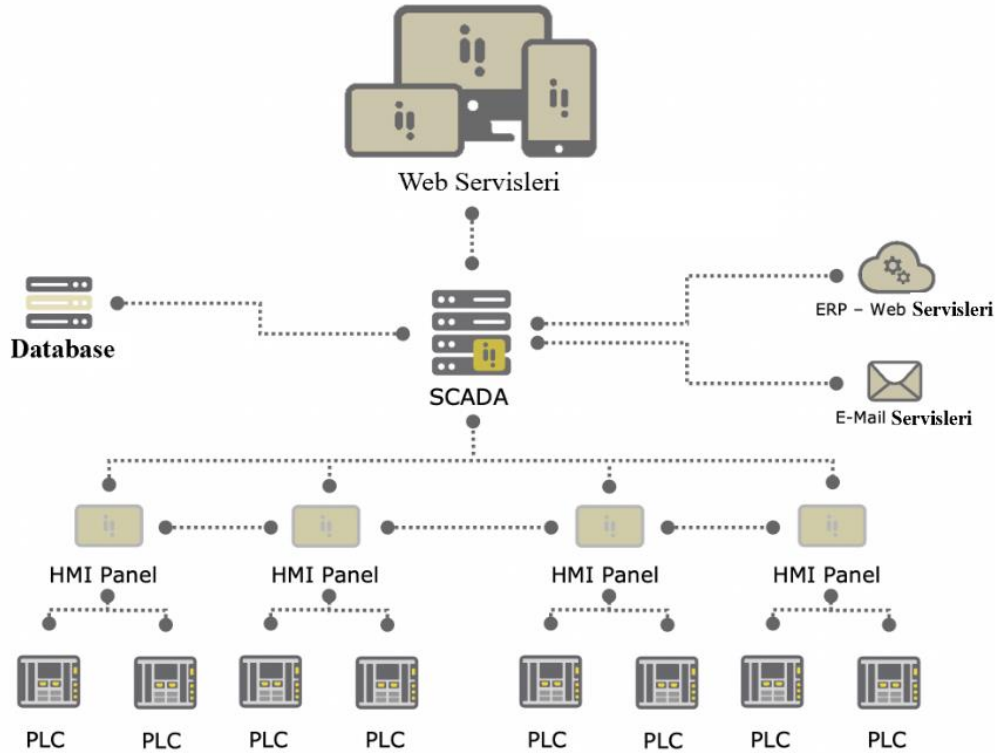
Sensörler, fiziksel parametrelerin ölçümünü gerçekleştiren cihazlardır. SCADA sistemleri için sıcaklık, basınç, akış hızı ve seviye gibi çeşitli parametrelerin ölçülmesini sağlar ve bu verileri PLC veya RTU'ya iletir.

### 3.1.4. İletişim altyapısı

İletişim altyapısı, SCADA sisteminin temelini oluşturan veri iletim ağlarıdır. Kablolulu (Ethernet, seri hat, fiber optik) veya kablosuz (radyo, mikrodalga, hücresel) iletişim protokolleri aracılığıyla veri aktarımı sağlanır ve sistemin entegrasyonu korunur.

### 3.1.5. Veri tabanı yönetimi

Veri tabanı yönetimi, SCADA sistemleri tarafından toplanan verilerin organize bir şekilde saklanması ve yönetilmesi için veri tabanı yönetim sistemleri kullanılır. Bu sistemler, veri analizi ve raporlaması için temel sağlar. Şekil 3.3'te (Anonim, 2024) SCADA sisteminin genel bir topolojisi gösterilmiştir.



Şekil 3.3- SCADA sistemlerinin genel topolojisi

Şekil 3.3'te görüleceği üzere sahadaki sensörleri ve bileşenleri kontrol eden PLC'ler, HMI'lar vasıtasıyla kontrol edilmektedir.

SCADA sistemleri, endüstriyel süreçlerin etkin ve güvenilir bir şekilde yönetilmesini sağlayarak, operatörlerin gerçek zamanlı kontrol sağlamalarına, süreçleri izlemelerine ve müdahalelerde bulunmalarına olanak tanır. Yukarıda açıklanan bileşenler, SCADA sistemlerinin tüm bu işlevleri yerine getirmesi için entegre şekilde çalışır. Bu bileşenlerin doğru ve güvenilir bir şekilde çalışması, SCADA sistemlerinin verimliliğini ve operasyonel güvenliğini artırmaktadır.

### **3.1.6. HMI'lar**

SCADA sistemlerinde operatör ile iletişim çoğunlukla HMI'lar ile yapılır. Operatörler HMI'ları kullanarak süreç görselleştirmeleri, kontrol fonksiyonları ve alarm yönetimleri gibi çeşitli görevleri yürütürler. HMI'lar, kullanıcı dostu ara yüzleri sayesinde operatörlerin sahadaki verilere erişimini kolaylaştırır ve hızlı karar alma süreçlerine olanak sağlar. SCADA sistemlerinde kullanıcı ile iletişimi sağlayan HMI'lar üç çeşide ayrılabilir.

#### **3.1.6.1. Donanım Tabanlı HMI'lar**

Endüstriyel ortamlar için özel tasarlanan, dokunmatik veya fare (Mouse) ile kontrol edilen HMI'lardır. Yazılım tabanlı HMI'lara göre daha sağlam tasarlanmıştır ve tozlu, nemli veya sıcak ortamlarda çalışabilmesi için özel olarak tasarlanırlar. Daha çok fabrika otomasyonunda, enerji santrallerinde ve petrol-gaz tesislerinde kullanılmaktadır. Yaptığımız bu tez çalışmasında Siemens tarafından geliştirilen, donanım tabanlı bir HMI olan TP-900 Comfort HMI kullanıldı.

#### **3.1.6.2. Yazılım Tabanlı HMI'lar**

Birçok konuda donanım tabanlı HMI'lara benzeyen yazılım tabanlı HMI'lar, donanım tabanlı HMI'lardan farklı olarak sahadan uzak yerlerde kurulan sunucular, sanal sunucular ya da bilgisayarlar üzerinde yazılım tabanlı olarak çalışmaktadır. İçerisinde

hazır grafik ve arayüzler barındıran bu modeller genellikle uzak kontrol merkezlerinde kullanılmaktadır.

### **3.1.6.3. Mobil HMI'lar**

Akıllı telefon, tablet gibi cihazlar üzerinden genellikle yazılım tabanlı veya donanım tabanlı bir HMI'a bağlı olarak, operatöre esneklik kazandırmak amacıyla kullanılan HMI'lardır. Bu HMI'lar ile uzaktan izleme ve kontrol yapılabilmektedir. Ayrıca sistemde meydana gelen herhangi bir alarm veya uyarı durumunda operatöre anında bildirim verebilmektedir.

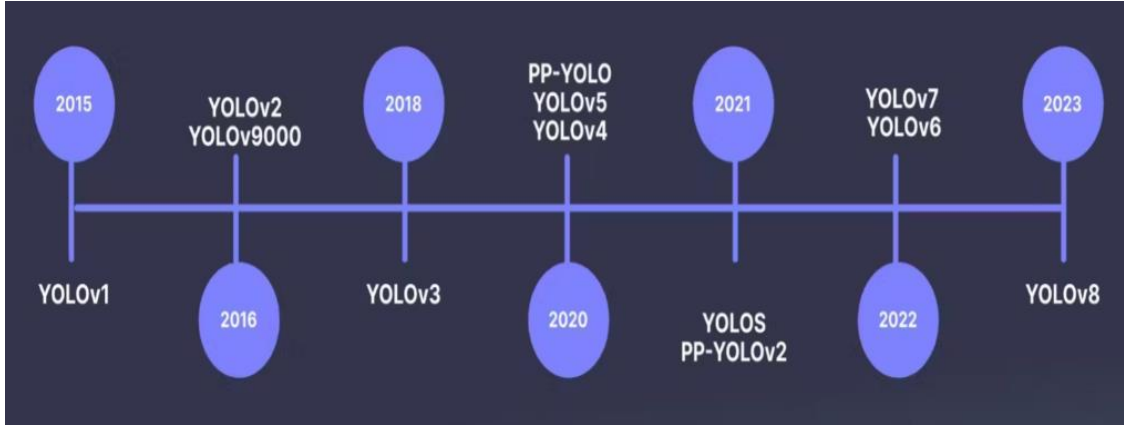
### **3.1.6.4. HMI Yazılımı**

HMI yazılımı, veri toplama, analiz, izleme ve kontrol işlevlerini yerine getiren merkezi kontrolün yazılım bileşenidir. Bu yazılım, operatörlerin verileri yönetmesini, alarmlar oluşturmasını ve süreçlerin durumunu izlemelerini sağlar. Yapılan tez çalışmasında TIA Portal V16 yazılımı kullanıldı.

## **3.2. YOLO Modeli**

Gerek hızlı olması gerek de doğruluk oranlarının fazla olması YOLO modelini ön plana çıkaran özellikleridir. YOLO, basit CNN (Convolutional Neural Networks) kullanarak sınıf olasılıklarını ve sınırlayıcı kutuları doğrudan giriş görüntüsünden tahmin eden tek kademeli bir derin öğrenme ağ modelidir (Redmon ve diğ., 2015).

YOLO, nesne ve yüz algılamada yenilikçi bir yaklaşım sunmakta olup, önceki çalışmalardan farklı olarak sınıflandırıcıları tekrarlı kullanmak yerine, uzamsal ayrıştırmayı takiben sınıflandırıcıyı doğrudan uygulamaktadır. Bu model, tek bir sinir ağı aracılığıyla, tam görüntüler üzerinde doğrudan sınırlayıcı kutular ve sınıf olasılıklarını tek bir değerlendirme ile tahmin ederek işlem süresini önemli ölçüde azaltmaktadır. YOLO'nun yıllar içinde kaydettiği gelişmeler, gelecekte daha geniş bir uygulama alanı bulacağını işaret etmektedir. Şekil 3.4'te, geçmişten günümüze geliştirilen YOLO modelleri sunulmuştur (Buhl, 2023).



Şekil 3.4. YOLO modellerinin tarihsel gösterimi

### 3.2.1. mAP (ortalama doğrulukların ortalaması)

mAP (Mean Average Precision), nesne tanıma alanında yaygın şekilde kullanılan bir değerlendirme metriğidir ve modelin doğruluğunu ve genel performansını ölçmek amacıyla kullanılır. Özellikle, farklı nesne tanıma algoritmalarının karşılaştırılmasında etkili bir kriterdir. mAP'in temelinde, Recall (duyarlılık) ve Precision (kesinlik) 4-gibi kavramlar yer alır; bu iki metrik, modelin tespit ettiği nesnelerin doğruluğunu ve gerçekte var olan nesnelere ne kadar iyi tespit ettiğini gösterir. mAP'i anlamak için Precision ve Recall'un nasıl hesaplandığını ve bu iki metrik arasındaki dengenin ne kadar önemli olduğunu kavramak gereklidir.

Kesinlik ve duyarlılık kavramlarını daha iyi anlamak için TP, FP, TN ve FN kavramlarını anlamak gereklidir.

TP (True Positive, Doğru Pozitif): Modelin nesneyi doğru tahmin etmesi

FP (False Positive, Yanlış Pozitif): Modelin nesne olmayan veriyi nesne olarak tahmin etmesi (aslında nesne olmayanları nesne olarak işaretleme).

TN (True Negative, Doğru Negatif): Modelin nesne olmayan veriyi nesne değil olarak doğru tahmin etmesi.

FN (Yanlış Negatif, Yanlış Negatif): Modelin gözden kaçırdığı, yani tespit etmesi gerekirken tespit etmediği gerçek nesnelere.

Bu kavramları örnek üzerinden anlatacak olursak, 100 araba ve 100 motorun olduğu veri setinde;

TP: Model motor olan bir nesneyi bulup bu nesne motordur tahmininde bulunması

FP: Motor olmayan bir nesneyi bulup bu nesne motordur tahmininde bulunması

TN: Araba nesnesini bulup, bu nesne motor değildir tahminde bulunması

FN: Araba nesnesini bulup, bu nesne arabadır tahminde bulunması

Bu kavramlar iyice anlaşıldıysa kesinlik ve duyarlılık kavramlarına geçilebilir.

Kesinlik (Precision), modelin test ettiği nesnelere arasından ne kadarının gerçekten doğru olduğunu gösterir. Yani, modelin nesne olarak tahmin ettiği verilerin ne kadarının aslında doğru şekilde tespit edilmiş nesnelere olduğunu belirtir.

$$\text{Kesinlik} = \text{TP} / (\text{TP} + \text{FP})$$

Duyarlılık (Recall), Duyarlılık ise modelin, verilen veriler içerisindeki nesnelere kaçını doğru bildiğini gösteren orandır. Duyarlılık arttıkça, verilen verilerdeki nesnelere bulunma oranı da artmaktadır. Bir diğer deyişle yüksek duyarlılık, daha az sayıda gerçek nesnenin gözden kaçtığını gösterir.

$$\text{Duyarlılık} = \text{TP} / (\text{TP} + \text{FN})$$

Duyarlılık ve kesinlik kavramlarını daha iyi anlamak için aşağıdaki örneği inceleyelim.

Örnek olarak 250 tane verinin olduğu veri setinde bu verilerden 150 tanesi sahte insan yüzüne sahipken 100 tanesi de gerçek insan yüzüne sahip olsun. Eğitilen model 125 tane sahte yüz bulduğunda ve bu bulduğu sahte yüzlerin yalnızca 100 tanesini aslında sahte yüz, 25 tanesini ise aslında gerçek yüz ise modelin kesinlik değeri 125 tahminden 100 tanesi doğru olduğu için %80 olacaktır.

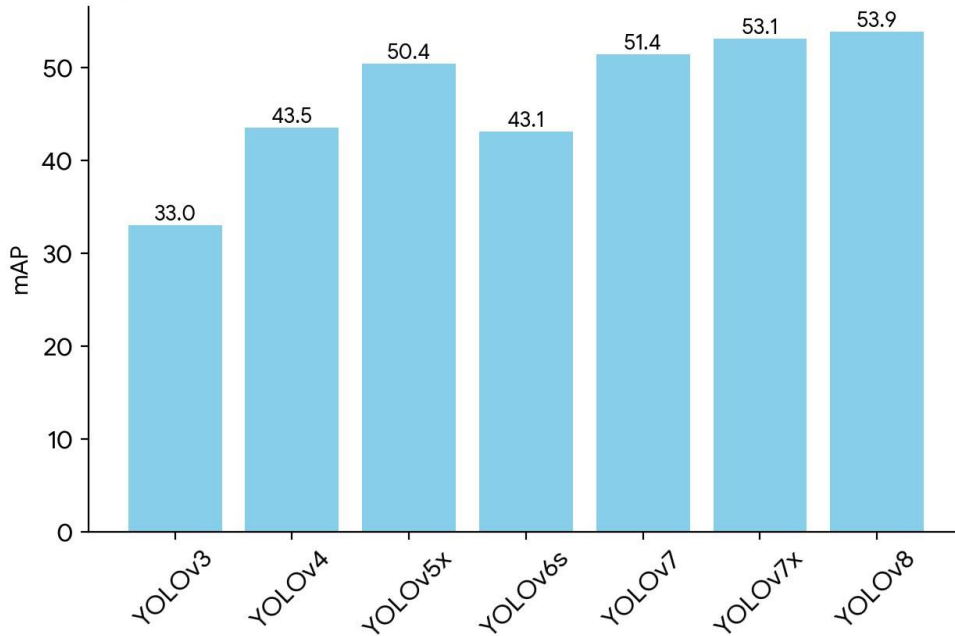
150 tane sahte yüzün olduğu veri setinde 125 tane sahte yüz bulduğu için ise duyarlılık değeri,  $125/150 = \%83,3$  olacaktır.

Kesinlik ve duyarlılık değerleri birbirinden farklı görünse de özünde birbirini destekleyen ve modelimizin ne kadar verimli çalıştığını anlamamız için bize yol gösteren çok önemli iki metriktir. Bu iki metriği bir arada değerlendirebilmek için AP (Ortalama Doğruluk) değeri kullanılır. Kesinlik ve duyarlılık metriklerini grafikte gösterirsek, eğrinin altında kalan alanın toplam değerini ifade etmek için AP değeri kullanılır.

AP bir sınıf için doğruluk ortalamasını verirken mAP modeldeki tüm sınıfların ortalama doğruluk oranını vermektedir. Örneğin sahte ve gerçek yüzlerden oluşan veri setinde, sahte ve gerçek yüzlerin kendilerine ait AP değeri varken, veri setinin genel doğruluğunu göstermek için mAP değeri kullanılır. Bu ikisi arasındaki ilişki, denklem 3.1'de gösterilmektedir.

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (3.1)$$

Bu denklem de N toplam sınıf sayısını göstermektedir. Bir sınıflık modelde  $AP = mAP$  olacaktır. 3 sınıfımızın olduğu bir modelde, AP değeri için sırasıyla 0.80, 0.90, 0.55 değerini alırlarsa bu durumda mAP değerimiz  $1/3(0.80+0.90+0.55) = 0.75$  çıkacaktır. Görüntü işleme verilerinde modelin tüm sınıflar bazında başarısını anlamak için mAP değeri bizim için çok önemlidir. Şekil 3.5'te YOLO modellerinin aynı veri seti üzerinde yapılan testlerinde mAP değeri bazında karşılaştırmaları gösterilmektedir. Grafikten de anlaşılacağı üzere mAP değeri bazında YOLOv8 en başarılı olan modeldir. Bu başarısından dolayı, bu tez çalışmasında, yüzün canlı olup olmadığını anlamak için YOLOv8 modeli kullanıldı.



Şekil 3.5. YOLO modellerinin mAP değerlerine göre karşılaştırılması

YOLOv3 için COCO veri seti üzerinde yapılan test ile %33 mAP oranı bulundu (Redmon ve Farhadi, 2018)

YOLOv4 için COCO veri seti üzerinde yapılan test ile %43,5 mAP oranı bulundu (Bochkovskiy ve diğ., 2020).

YOLOv5x için COCO veri seti üzerinde yapılan test ile %50,4 mAP oranı bulundu (Dong ve diğ., 2022).

YOLOv6s için COCO veri seti üzerinde yapılan test ile %43,1 mAP oranı bulundu (Yung ve diğ., 2022).

YOLOv7 için COCO veri seti üzerinde yapılan test ile %51,4 mAP oranı ve YOLOv7x için %53,1 mAP bulundu (Wang, 2022).

YOLOv8 için COCO veri seti üzerinde yapılan test ile %53,9 mAP oranı bulundu (Anonymous, 2023).

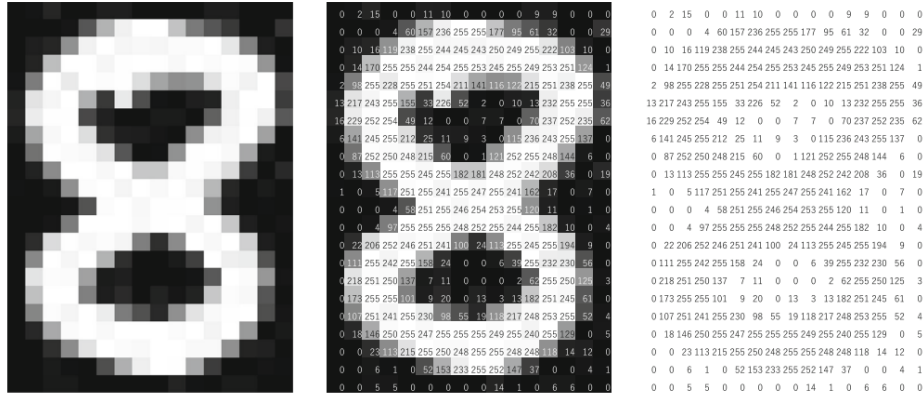
### 3.2.2. YOLOv8

Önceki YOLO modellerine kıyasla daha hızlı ve daha doğru olan YOLOv8, ölçeklenebilir mimarisi sayesinde farklı görevler ve performans gereksinimlerine uyarlanabilir bir yapı sunmaktadır. Yeni bir omurga ağı kullanarak, daha iyi özellik temsili ve daha yüksek doğruluk sağlamaktadır. Otonom sürüş, robotik, güvenlik, tıbbi görüntüleme ve tarım gibi geniş bir uygulama alanına sahip olan YOLOv8, yüksek hızı, doğruluğu, esnekliği ve kullanım kolaylığı ile görsel yapay zekâ alanındaki araştırmalar ve uygulamalar için güçlü bir araçtır. Ayrıca YOLOv8, nesne algılama, segmentasyon, poz tahmini, izleme ve sınıflandırma gibi çeşitli görsel görevleri desteklemektedir (Terven, 2023).

#### 3.2.2.1. YOLOv8 mimarisi

YOLO modellerinin mAP değerlerinin farklılık göstermelerindeki sebep, kullandıkları mimaridir. YOLOv8'in başarısının altında yatan temel bileşenler ise omurga (backbone) ve baş (head) olarak adlandırılan iki ana yapıdır. Bu yapılar, girdi görüntüsünden anlamlı özellikler çıkarılmasından, bu özelliklerin farklı ölçeklerde birleştirilmesine ve nihayetinde nesnelere konum ve sınıflarının tahmin edilmesine kadar tüm süreçte kritik rol oynarlar. (Li ve diğ., 2023) YOLOv8 mimarisi Şekil 3.6'da gösterilmiştir. (Anonymous, 2023)



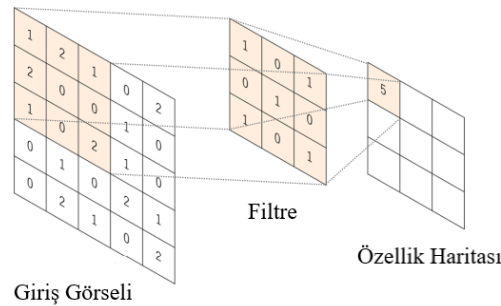


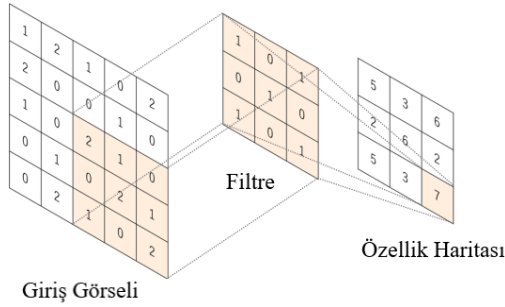
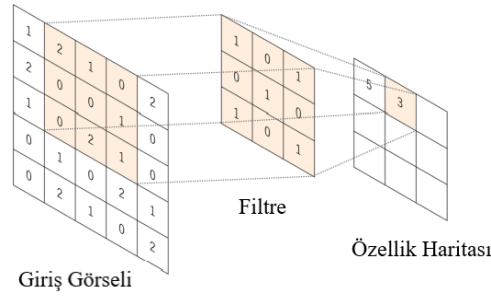
Şekil 3.7. Bilgisayarların görselleri nasıl algıladığını gösteren örnek bir şekil

Evrişim katmanı, Şekil 3.5'te gösterilen görselleri belirli filtrelerden geçirerek modellerin bu görselleri çok kısa süre içerisinde algılayıp, tahmin yapabilmelerini sağlar.

YOLO mimarisinde de yer alan bu katman, neredeyse tüm derin öğrenme modellerinde bulunmaktadır. Evrişim katmanlarının kullanıldığı modeller, yüksek doğruluk elde etmek için genellikle çok sayıda etiketlenmiş veriye ihtiyaç duymaktadır (Suganyadevi ve diğ., 2022). Dolayısıyla, evrişim katmanı kullanan mimariler gibi, YOLO modelleri de etkili bir şekilde çalışabilmek için geniş bir etiketlenmiş veri setine gereksinim duymaktadır.

Şekil 3.8'de örnek bir Evrişim Katmanının nasıl uygulandığı 3 adımda gösterilmiştir. (Yamashita ve diğ., 2018) Şekilde 5x5'lik görsele, 1 ve 0'lardan oluşan 3x3'lük filtre uygulanmıştır ve sonuç olarak üçüncü adımdaki özellik haritası ortaya çıkmıştır. Bu çıkan sonuca Özellik Haritası (Feature Map) denir.





Şekil 3.8. Üç adımdan oluşan örnek evrişim katmanı uygulaması

### 3.2.2.1.2. Baş (head)

YOLOv8'in baş bölümü, boyun bölümünden gelen bilgileri işleyerek nesnelerin sınıflarını ve konumlarını belirler. Bu bölüm, iki temel görevi yerine getirir:

**Sınıflandırma:** Tespit edilen her bir nesnenin hangi sınıfa ait olduğunu (örneğin, insan, araba, kedi) tahmin eder.

**Konumlandırma:** Nesnenin görüntüdeki yerini ve boyutunu tam olarak belirleyen sınırlayıcı kutuyu (bounding box) tahmin eder.

YOLOv8, nesnelerin farklı boyut ve en-boy oranlarına sahip olabileceğini göz önünde bulundurarak, çapa (anchor) kullanımı sayesinde daha esnek ve isabetli tahminler yapar. Çapalar, önceden tanımlanmış boyut ve en-boy oranlarına sahip kutular olup, modelin çeşitli ölçeklerdeki nesnelere daha etkili bir şekilde tespit etmesine olanak tanır. YOLOv8'in baş bölümü, çapa kullanımının getirdiği bu avantaj ile nesnelere daha doğru ve hızlı bir şekilde tespit ederek, gerçek zamanlı ve yüksek doğruluk gerektiren nesne tespiti uygulamaları için ideal bir çözüm sunar.

### 3.3. Yüz Tanımının Temel Prensipleri

Çoğunlukla yüz tanıma modelleri, tanıma sürecini dört ana aşamada gerçekleştirir: tespit, hizalama, temsil ve doğrulama. (Serengil ve diğ., 2020) Bu aşamalar, modern yüz tanıma sistemlerinin temel yapı taşlarını oluşturur ve her biri, sürecin doğruluğunu ve etkinliğini artırmak için kritik öneme sahiptir.

#### 3.3.1. Yüz tespiti

Yüz tanıma modelinin ilk adımı, görüntüdeki yüzlerin tespit edilmesidir. Bu aşamada, görüntü işleme teknikleri ve makine öğrenimi algoritmaları kullanılarak yüzler belirlenir. Özellikle Haar-cascade sınıflandırıcıları gibi yöntemler yüz ve göz tespiti için yaygın olarak kullanılır. Haar-cascade yöntemi, eğitilmiş sınıflandırıcıları kullanarak görüntüdeki yüzleri ve diğer nesnelere hızlı ve verimli bir şekilde tespit eder. Bu adım, görüntüdeki potansiyel yüz bölgelerinin belirlenmesi ve doğrulanmasını içerir. Tespit edilen yüzlerin konumu ve boyutu, sonraki aşamalar için temel bilgi sağlar. Yüz tespitinde kullanılan algoritmalar, yüksek doğruluk oranları ve düşük yanlış pozitif/negatif oranları ile çalışmalıdır.

#### 3.3.2. Yüz hizalaması

Tespit edilen yüzlerin hizalanması, yüz tanıma sürecinin ikinci kritik aşamasıdır. Yüz hizalama, yüzün temel özelliklerinin (gözler, burun, ağız) standart bir pozisyona getirilmesi işlemidir. Bu, yüzün farklı açılardan çekilmiş olmasının tanıma performansını olumsuz etkilememesi için gereklidir. Google araştırmacıları, yüz hizalama algoritmalarının yüz tanıma doğruluğunu önemli ölçüde artırdığını göstermiştir. (Parkhi ve diğ., 2015) Hizalama sürecinde, yüzün dönme açısını belirlemek için trigonometrik hesaplamalar yapılır. Gözlerin konumu belirlenerek bir dik üçgen oluşturulur ve kosinüs teoremi kullanılarak yüzün dönme açısı hesaplanır. Hesaplanan açı ile yüz görüntüsü döndürülür ve yüz, doğru bir şekilde hizalanır. Bu adım, yüzün farklı pozisyonlarda ve açılarda olmasının tanıma doğruluğunu olumsuz etkilemesini önler.

### 3.3.3. Yüz temsili

Yüz hizalamasının ardından, yüz görüntülerinin matematiksel olarak temsil edilmesi kritik bir adımdır. Bu aşamada, yüz özelliklerini çıkarmak ve yüzleri sayısal vektörler olarak temsil etmek için çeşitli yöntemler kullanılır. Derin öğrenme modelleri, örneğin evrişimsel sinir ağları, yüz özelliklerini hiyerarşik bir şekilde öğrenir ve bu özellikleri yüksek boyutlu vektörler olarak ifade eder. Örneğin, VGG-Face modeli yüzleri 2622 boyutlu bir vektörle, FaceNet modeli ise 128 boyutlu bir vektörle temsil eder. Bu vektörler, yüzün benzersiz özelliklerini kodlayarak tanıma sürecinde karşılaştırma amacıyla kullanılır.

Bunun yanında, geleneksel yöntemler de yüz özelliklerinin çıkarılmasında önemli bir rol oynar. Ana Bileşen Analizi (PCA) ve Lineer Ayırma Analizi (LDA) gibi istatistiksel yöntemler, yüz görüntülerinin boyutunu azaltarak en belirgin özellikleri çıkarır. Bu yöntemler, veri setinin varyansını maksimum düzeyde koruyarak yüzlerin daha kompakt bir şekilde temsil edilmesini sağlar.

Manuel özellik çıkarım teknikleri, HOG (Histogram of Oriented Gradients) ve LBP (Local Binary Patterns) gibi yöntemleri içerir. Bu teknikler, yüzlerdeki kenarlar, dokular ve diğer temel yapı taşlarını analiz ederek yüz özelliklerini belirler. HOG, yüz görüntüsündeki kenar yapılarını ve gradyan yönelimlerini analiz ederken, LBP yerel dokuları ve desenleri yakalar.

Tüm bu yöntemler, yüzlerin temsil edilmesi aşamasında birbirini tamamlayıcı şekilde kullanılabilir. Derin öğrenme algoritmaları, büyük veri setleri üzerinde eğitilerek yüzlerin karmaşık ve yüksek seviyeli özelliklerini öğrenirken, geleneksel ve manuel yöntemler daha basit ve doğrudan özellik çıkarımı sağlar. Bu süreçlerin her biri, yüz tanıma doğruluğunu artırmak amacıyla optimize edilir ve yüzlerin anlamlı ve ayırt edici bir şekilde temsil edilmesini sağlar.

### 3.3.4. Yüz doğrulanması

Son aşama, elde edilen yüz vektörlerinin doğrulanmasıdır. Bu aşamada, iki yüz vektörünün aynı kişiye ait olup olmadığını belirlemek amacıyla çeşitli metrikler kullanılır. En yaygın kullanılan metrikler arasında Öklid mesafesi (Euclidean distance) ve kosinüs benzerliği (cosine similarity) yer alır. Öklid mesafesi, iki vektör arasındaki düz çizgi mesafesini ölçerken, küçük mesafeler yüksek benzerlik anlamına gelir. Kosinüs

benzerliđi ise iki vektör arasındaki açıyı hesaplar ve 1'e yakın deđerler, iki vektörün yüksek düzeyde benzer olduđunu gösterir. Doğrulama sürecinde, bir eşik deđeri belirlenir ve iki vektör arasındaki mesafe bu eşik deđerine göre deđerlendirilir. Mesafe eşik deđerinin altındaysa, yüzlerin aynı kişiye ait olduđu kabul edilir. Kısacası, doğrulama katmanı, iki farklı yüz görüntüsünden elde edilen vektörlerin benzerliđini veya farklılıđını ölçer. Vektörler birbirine yeterince yakınsa, yüzlerin aynı kişiye ait olduđu sonucuna varılır; aksi halde, farklı kişilere ait oldukları anlaşılır.

Bu adım, yüz tanıma sürecinin nihai kararını verir ve sistemin genel doğruluđunu belirler. Doğrulama aşamasında kullanılan metriklerin hassasiyeti, sistemin genel performansını doğrudan etkiler ve doğru eşik deđerinin belirlenmesi, yanlış pozitif ve negatif oranlarını minimize etmek için kritik öneme sahiptir.

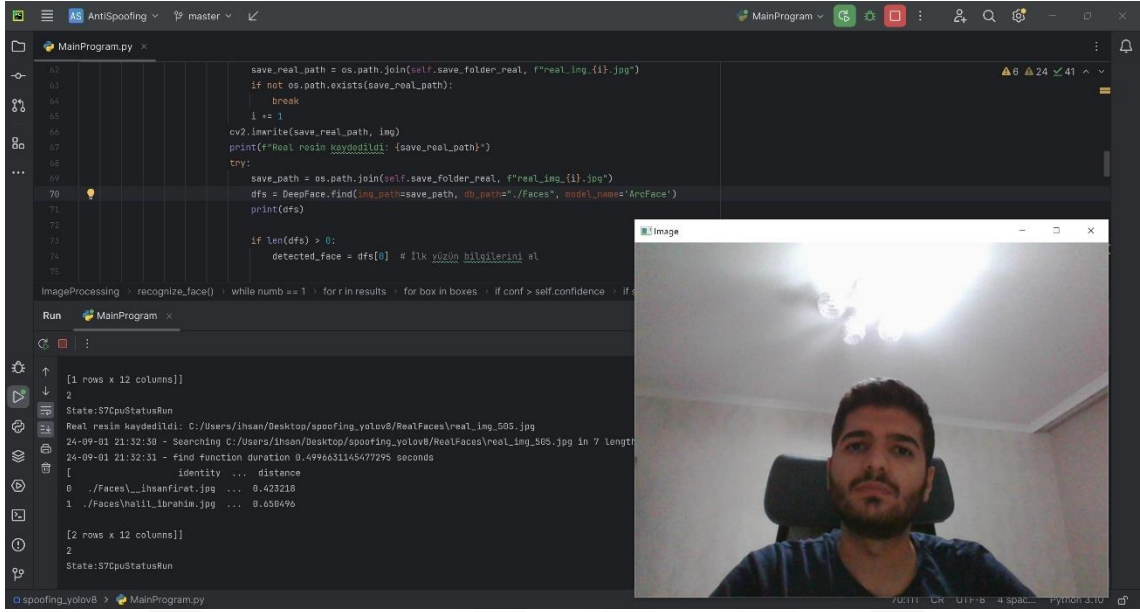
### 3.3.5. Yüz modeli performans karşılaştırılması

Bu tez çalışmasında ArcFace, Facenet512, Facenet ve VGG-Face tamamen aynı koşullar altında test edildi. Yapılan test sonuçları aşağıda gösterilmiştir. Yapılan testlerde, modelleri en çok yanıltan iki yüz, Şekil 3.9'da gösterilmiştir. Şekildeki sol taraftaki yüz ihsan\_firat kullanıcısına aitken, sağ taraftaki yüz halil\_ibrahim kullanıcıya aittir.



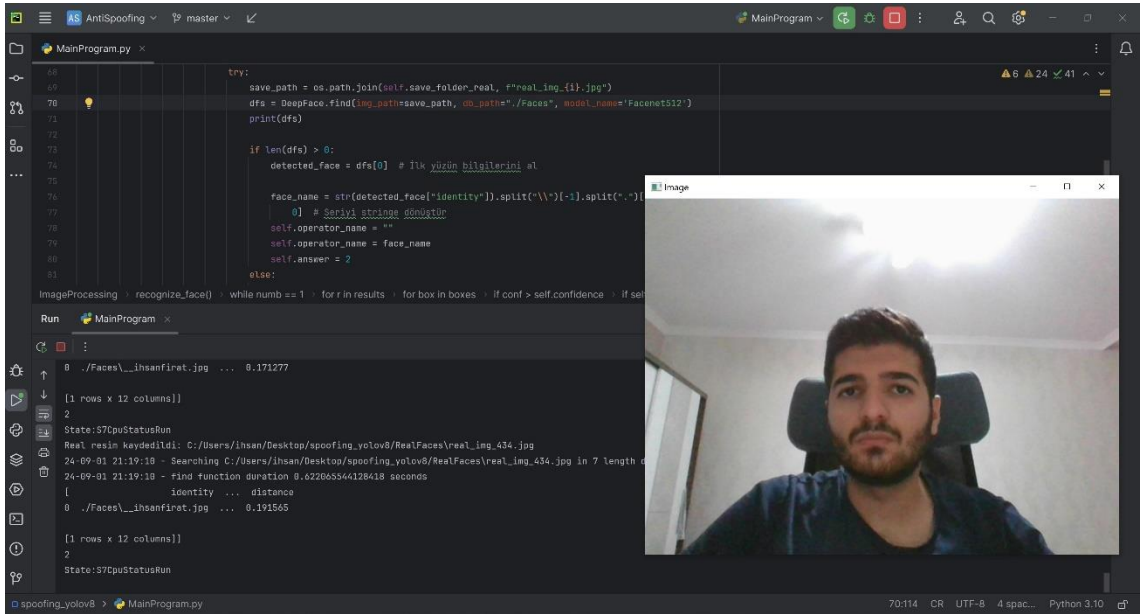
Şekil 3.9. Veri tabanındaki İhsan Fırat ve Halil İbrahim'e ait operatör yüzleri

Şekil 3.10'da görüleceđi üzere, ArcFace modeli, ihsan\_firat'a ait olan yüzü halil\_ibrahim kullanıcıya da benzetti. Süre olarak yaklaşık 0,5 saniye de bulsa da doğruluk olarak başarılı bir sonuç ortaya koymadı.



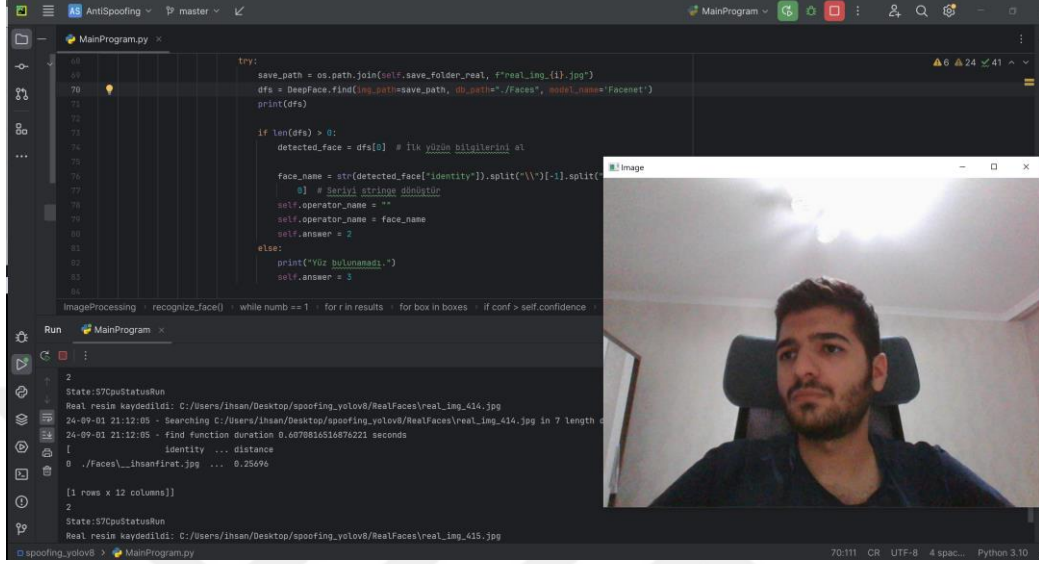
Şekil 3.10. ArcFace modeli başarımları

Şekil 3.11'de görüleceği üzere, Facenet512 modeli, ön taraftan çekilen ihсан\_fırat'a ait olan yüzü 0,62 saniye içerisinde analiz edip doğru olarak buldu ve halil\_ibrahim kullanıcıya da benzetmedi.



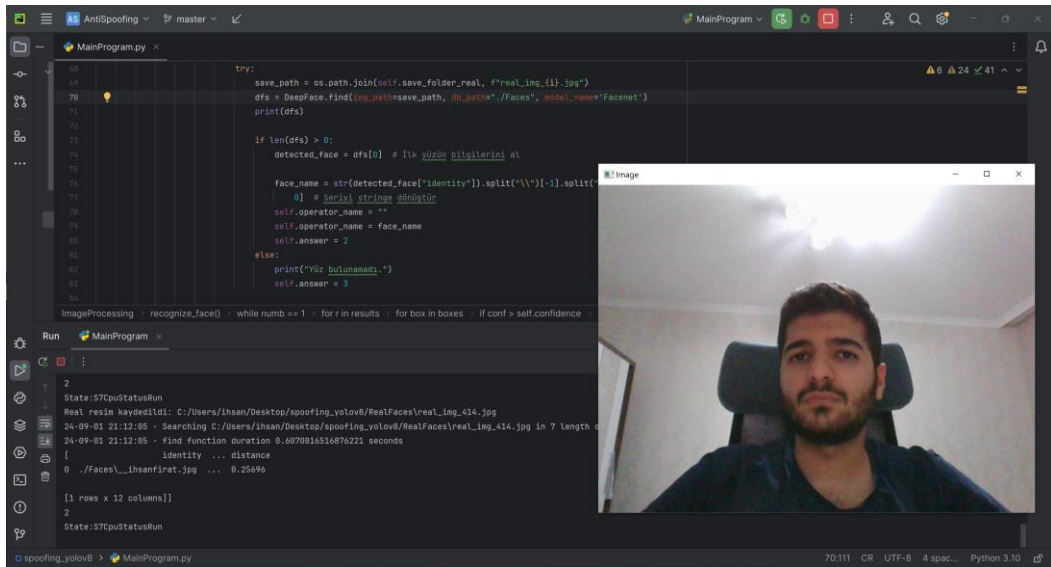
Şekil 3.11. Facenet512 modeli başarımları

Şekil 3.12’de görüleceği üzere, Facenet modeli, yan taraftan çekilen ihsan\_firat’a ait yüzü 0,60 saniye içerisinde analiz edip doğru olarak buldu ve halil\_ibrahim kullanıcılarına da benzetmedi.



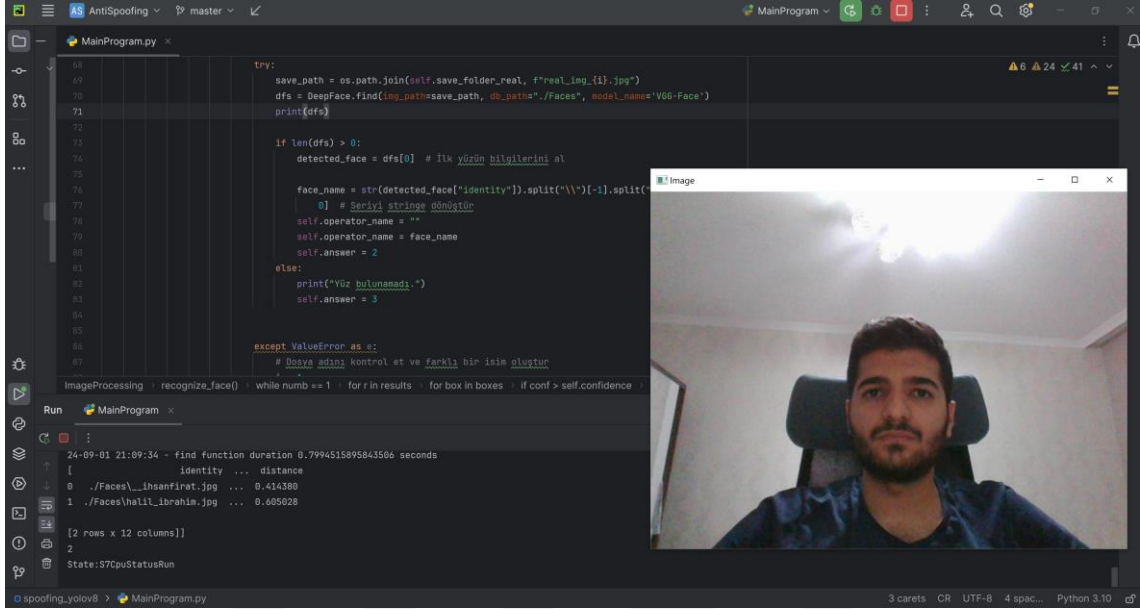
Şekil 3.12. Yan taraftan çekilen yüz için Facenet modeli başarımları sonuçları

Şekil 3.13’te görüleceği üzere, Facenet modeli, ön taraftan çekilen ihsan\_firat’a ait yüzü 0,60 saniye içerisinde analiz edip doğru olarak buldu ve halil\_ibrahim kullanıcılarına da benzetmedi.



Şekil 3.13. Ön taraftan çekilen yüz için Facenet modeli başarımları sonuçları

Şekil 3.14'te görüleceği üzere VGG-Face modeli, ön taraftan çekilen ihsan\_firat'a ait yüzü 0,79 saniye içerisinde analiz edip, hem ihsan\_firat kullanıcıasına hem de halil\_ibrahim kullanıcıasına benzetti.



Şekil 3.14. VGG-Face modeli başarımları sonuçları

Başarımları oranları karşılaştırıldığında, 0,5 saniye ile en hızlı model ArcFace olmasına rağmen doğruluk açısından beklenen performansı sağlayamadı. Buna karşın, 0,6 saniye ile Facenet modeli hem zaman hem de doğruluk açısından başarılı bir sonuç ortaya koydu. Bu nedenle tez çalışmamızda, daha dengeli bir performans sunan Facenet modeli tercih edilmiştir.

### 3.3.6. FaceNet modeli

2015 yılında, Schroff ve diğerleri tarafından geliştirilen FaceNet modeli 3 görevi yerine getirebilecek şekilde geliştirilmiştir. Bunlar kümeleme, yüz doğrulama ve yüz tanımadır.

Kümeleme, veri setindeki en yaygın yüzleri eşleştirip bir araya toplama görevini yapmaktadır.

Yüz Doğrulama, FaceNet modeline verilen 2 adet yüz fotoğrafının aynı kişiye ait olup olmadığının analizini yapmaktadır.

Son olarak Yüz Tanıma ise modele verilen yüz fotoğrafının, modele verilen veri seti içerisinde bulma işlemini yapmaktadır.

Bizim çalışmamızda FaceNet yüz tanıma görevini yerine getirmektedir. FaceNet bu yüz tanıma görevini getirirken PCA gibi özellik çıkarma yöntemini kullanan diğer yöntemlerden farklı olarak 128 boyutlu gömme vektörü oluşturma ve üçlü kayıp fonksiyonu metotlarını kullanır. FaceNet'in çalışma şeklini 5 adımda özetleyebiliriz.

Birincisi, FaceNet modeline bir yüze ait fotoğraf verilir.

İkincisi, CNN derin ağları ile bu yüzün özellikleri çıkartılır

Üçüncüsü, CNN özellikleri çıkardıktan sonra yüze ait özelliklerden oluşan 128 boyutlu bir vektöre dönüştürür. Bu vektör o yüze ait imza ya da başka bir deyişle o yüzün taslağı olur.

Dördüncüsü, üçlü kayıp fonksiyonu kullanılarak, aynı yüzleri birbirine yakın, farklı kişilere ait yüz vektörlerini ise uzak tutar.

Beşincisi, verilen her yeni yüz, yeni bir 128 boyutlu vektöre dönüştürülür ve yeni yüze ait vektör ile veri tabanındaki diğer yüzler arasında Denklem 3.2'de gösterilen Öklidyen mesafesi kullanılarak verilen yüzün, veri tabanındaki hangi yüze en çok benzediği bulunur ve FaceNet modelinin yaptığı analiz tamamlanır.

$$d(A,B)=\sqrt{(A_1 - B_1)^2 + (A_2 - B_2)^2 + \dots + (A_{128} - B_{128})^2} \quad (3.2)$$

Denklem 3.2'deki A ve B, 2 farklı yüzü temsil eden 128 boyutlu vektörlerdir.  $d(A,B)$  ise bu yüzlere ait vektörler arasındaki mesafeyi verir. FaceNet, verilen yüzü veri tabanında ararken, öklidyen mesafesinin belirlenen eşik değerinin aşağısında kalan yüzleri eşleştirir, eşik değerinin üzerinde kalanları ise farklıdır diye etiketler.

### 3.4. SNAP7 Haberleşmesi

Snap7, Siemens S7 PLC'leri ile endüstriyel Ethernet üzerinden çift yönlü veri alışverişini mümkün kılan açık kaynaklı bir haberleşme kütüphanesidir. Bu kütüphane, S7comm protokolünü uygulayarak, PLC'nin bellek alanlarına doğrudan erişim, okuma, yazma ve kontrol fonksiyonlarını yerine getirme yeteneği sunar. (Zheng ve diğ., 2017)

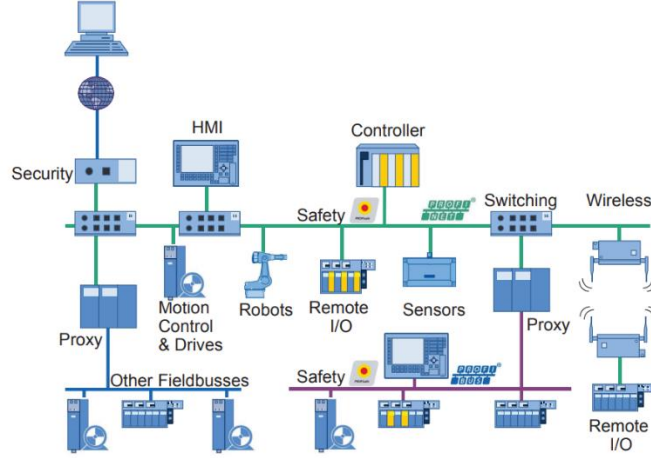
Snap7'nin platform bağımsız yapısı, Windows, Linux ve macOS gibi farklı işletim sistemlerinde çalışabilme avantajı sağlar. Bu sayede, endüstriyel otomasyon uygulamaları geliştirme sürecinde esneklik ve taşınabilirlik sunar. Ayrıca, Python, Java,

C# gibi farklı programlama dillerine yönelik arayüzler sunması, geliştiricilerin tercih ettikleri dilde Snap7'yi kullanarak projelerini hayata geçirmelerine olanak tanır. Bu da linux tabanlı Jetson Nano, Raspberry pi gibi kontrol kartlarında da bu protokolün kullanabildiği anlamında gelir. Snap7'nin en önemli özelliklerinden biri, Siemens S7 PLC'lerinin yerel iletişim protokolü olan S7comm'u tam olarak desteklemesidir. Bu sayede, PLC'nin tüm veri alanlarına (örneğin, girişler, çıkışlar, markerlar, veri blokları) doğrudan erişim sağlanabilir. Bu, PLC'nin durumunu izleme, kontrol etme ve veri toplama gibi işlemleri gerçekleştirmek için gerekli olan esnekliği ve kontrolü sağlar. Snap7, endüstriyel otomasyon uygulamalarında yaygın olarak kullanılan SCADA sistemleri ile kolay entegrasyon imkânı sunar. Bu sayede, PLC'den toplanan veriler gerçek zamanlı olarak izlenebilir, analiz edilebilir ve operatörlere anlık olarak sunulabilir. Ek olarak, Snap7'nin açık kaynaklı yapısı, sürekli geliştirilmesini ve güncellenmesini mümkün kılarak, endüstriyel otomasyon alanındaki yeni teknolojilere ve gereksinimlere uyum sağlamasını kolaylaştırır.

### 3.5. PROFINET Protokolü

PROFINET, endüstriyel otomasyon alanında yaygın olarak kullanılan açık standart bir Ethernet protokolüdür. Başlangıçta Siemens tarafından geliştirilmiş olsa da, bugün PROFIBUS & PROFINET International (PI) tarafından yönetilmektedir. PROFINET, endüstriyel cihazlar arasında gerçek zamanlı iletişim sağlayarak, endüstriyel süreçlerin daha verimli ve güvenilir bir şekilde yönetilmesine olanak tanır. Farklı topoloji seçenekleri ve iletişim hizmetleri sunan PROFINET, geniş bir endüstriyel uygulama yelpazesine kolayca uyarlanabilir. Ayrıca, diğer endüstriyel Ethernet protokolleri ve farklı üreticilerin cihazları ile uyumlu çalışabilmesi, PROFINET'in endüstriyel otomasyonda yaygınlaşmasını ve Endüstri 4.0'ın temel unsurlarından biri olmasını sağlamıştır. PROFINET haberleşmesinin kullanıldığı örnek bir sistem Şekil 3.15'te gösterilmektedir (Anonymous, 2014). Şekilde yeşil ile renklendirilen hat, PROFINET ağını temsil etmektedir.

Bu tezde, PLC tarafında kullanılan Siemens S7-1200 serisinden CPU-1212 ile HMI tarafında kullanılan Siemens TP-900 Comfort Panel arasında PROFINET haberleşmesi kullanıldı.



Şekil 3.15. Profinetin kullanıldığı örnek bir topolojinin görünümü

### 3.6. Geliştirilen Sistem Arayüzü ve Kullanımı

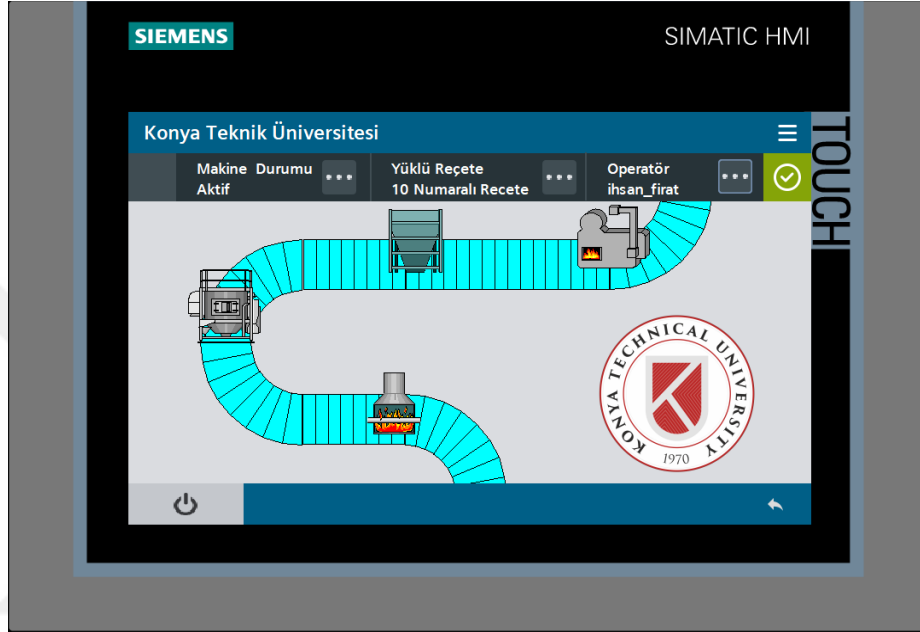
Tez çalışması için geliştirilen sistem arayüzü ve arayüz çalışma mantığı aşağıdaki gibi bir senaryo üzerinden anlatılabilir.

Sistemde HMI açıldığında karşımıza ilk olarak Şekil 3.16’da gösterilen ana ekran gelmektedir. Erişim Düzeyi Kontrol Sayfasında hangi operatörün hangi sayfalara erişebileceğinin ayarlandığı sayfaya geçilir. Buradan Proses Sayfası butonu ile makinedeki işlemlerin kontrol edilebildiği sayfaya geçilir.

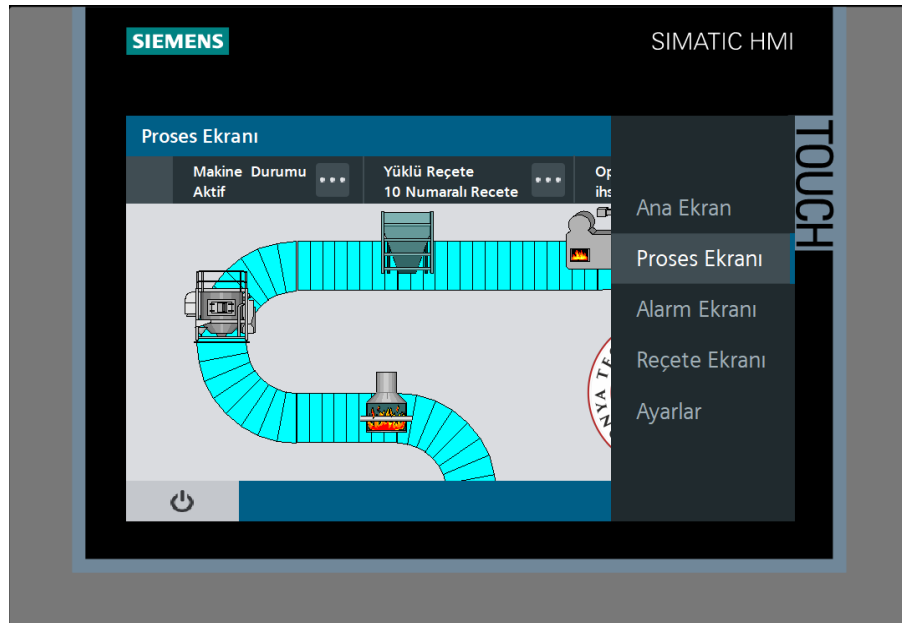


Şekil 3.16. Ana ekran

Şekil 3.17’de gösterilen Proses sayfasında Makine ile alakalı tüm durumlar izlenebilmekte ve kontrol edilebilmektedir. Sayfanın üst kısmında hangi operatörün giriş yaptığı ve hangi reçetenin seçili olduğu görülmektedir. Proses Sayfasında, sağ üstte bulunan 3 çizgi ile HMI da bulunan tüm sayfaların listelenmesi yapılır ve istenilen sayfaya geçilir. Operatör, Şekil 3.18’de gösterildiği gibi sağda listelenen alanda Reçete Ekranı butonuna tıkladığında öncelikle Login sayfasına yönlendirilir.



Şekil 3.17. Proses ekranı



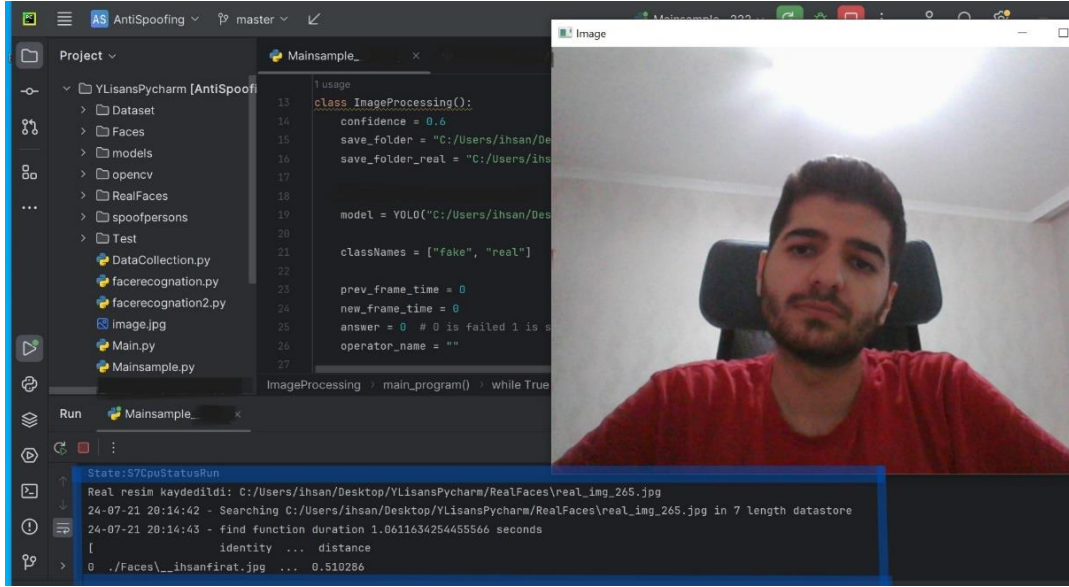
Şekil 3.18. Proses ekranı

Operatör Şekil 3.19’da gösterilen login ekranda ilk olarak yüz tespiti kontrolünden geçmesi gereklidir. Operatör kameraya yüzünü gösterdiğinde, Python tarafında ilk olarak yüzün gerçek mi sahte mi olduğu tespit edilir, yüzün gerçek olduğunun tespiti halinde, yüz için tanımlama işlemine geçilir.

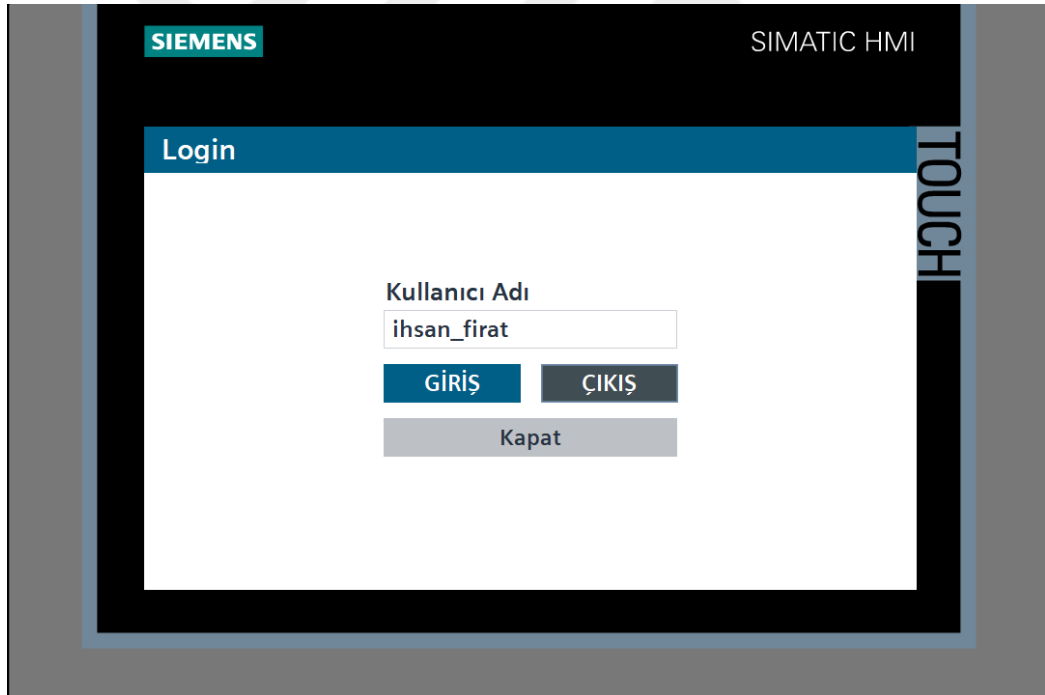


Şekil 3.19. Login ekranı

Şekil 3.20’de gösterilen yüz tanımlama kısmında, yüz veri tabanında kayıtlı kullanıcılar arasında bulunursa, PLC tarafına yüzün gerçek olduğunu ve kime ait olduğu bilgisi iletilir. Ayrıca operatörün ismi ve saat kaçta reçete sayfasına giriş yaptığı, veri tabanına kaydedilir. Kullanıcı güvenlik kısmını başarıyla geçerse, reçete sayfasına giriş yapabilir. Şekil 3.21’de gösterilen Login Ekranında görünen “Giriş” butonu ile kullanıcı Reçete Sayfasına geçiş yapar.

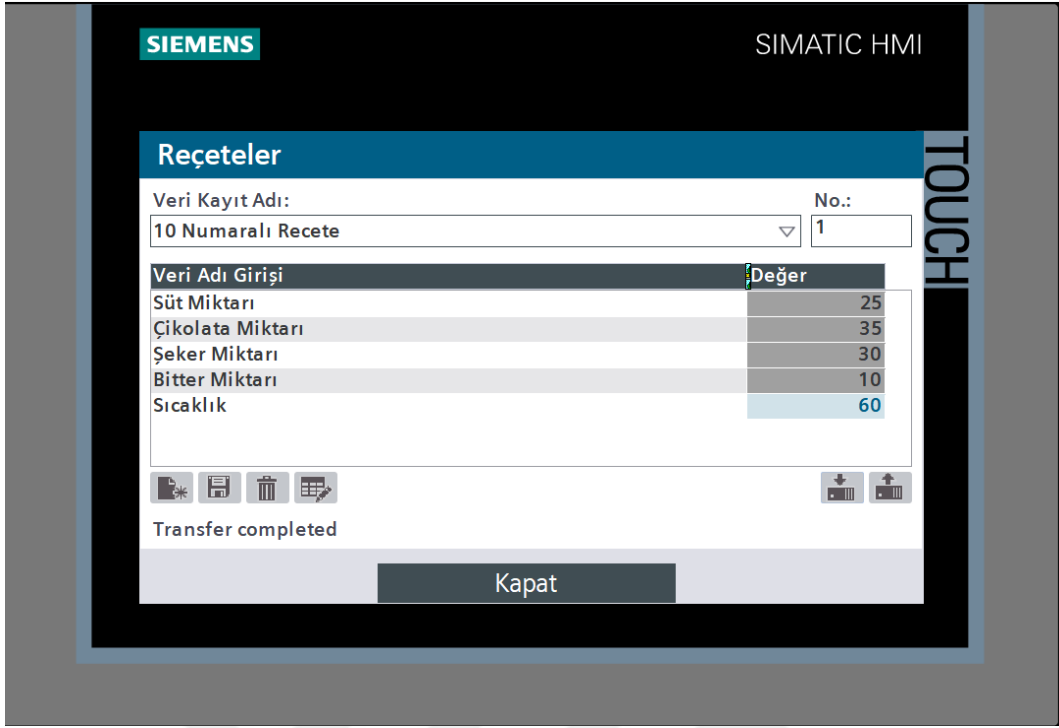


Şekil 3.20. Python'da gerçekleştirilen yüz tanımlama süreci



Şekil 3.21. Güvenlik sürecinden başarıyla geçildikten sonra login ekranı

Şekil 3.22'de gösterilen reçete sayfasında, istenirse yeni oranlar ile yeni bir reçete oluşturulabilir ya da kayıtlı reçeteler arasındaki bir reçete makineye yüklenilebilir. Yüklenen reçetenin adı, proses sayfasında görüntülenir. Böylelikle yeni gelen operatör hangi reçetenin yüklü olduğunu kısa sürede anlayabilir.



Şekil 3.22. Reçete ekranı

#### 4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

Yapılan proje kapsamında, yüzlerin gerçek mi yoksa sahte mi olduğunu tespit etmek amacıyla YOLOv8 modeli kullanılmıştır. Ardından, yüzlerin kimliğini belirlemek için FaceNet modeli devreye alınmıştır. Aşağıda, YOLOv8 ile eğitilen modelin eğitim süreci boyunca elde edilen grafikler detaylı bir şekilde açıklanacaktır. Sonrasında, FaceNet modelinin insan tanıma konusundaki hız ve etkinliği, yapılan testlerle somut bir şekilde gösterilecektir.

Kullanıcının yüzünün gerçek mi sahte mi olduğu, YOLOv8 algoritması kullanılarak yüzde 90 başarı oranıyla 0,5 saniye içinde tespit edilmektedir. Takip eden aşamada, FaceNet algoritması, veri tabanındaki yüzlerle karşılaştırma yaparak hangi yüze ait olduğunu %99,63 başarı oranı ile 0,5 ile 0,6 saniye arasında belirlemektedir. Toplam süreç, 2 saniyenin altında tamamlanmaktadır. Sistem hem hızlı hem de güvenli bir şekilde çalışmaktadır. Bu bölümde, YOLOv8 için gerçekleştirilen eğitim ve test sonuçları detaylı olarak sunulacaktır. Ayrıca, YOLOv8'in entegre edildikten sonraki performansı pratik uygulamalarla desteklenmiş ve ilgili görseller detaylıca açıklanacaktır.

Bir sonraki aşamada, FaceNet modelinin başarı oranı, gerçekleştirilen uygulamaların görselleri ile ortaya konulacaktır.

Bu uygulamalar, 8 GB RAM kapasitesine sahip bir bilgisayar ortamında gerçekleştirilmiştir. Dolayısıyla, elde edilen hız değerleri asgari seviyeler olup, daha yüksek performanslı bilgisayar ve kontrol kartlarında daha hızlı sonuçlar elde edilebilecektir. Tüm sürecin 2 saniye gibi kısa bir süre içinde sonuçlanabilmesi, fabrikalarda iş süreçlerinin daha hızlı tamamlanmasına katkıda bulunacak ve sistemin uygulanabilirliğini avantajlı kılacaktır.

Şekil 4.1'de YOLOv8 ve FaceNet modellerinin eğitilmesi, değerlendirilmesi, entegre edilmesine dair topoloji gösterilmektedir. Şekilde üzerinde verilen,

- a) Veri toplama, gerçek ve sahte görsellerden oluşan veri setinin oluşturulmasıdır.
- b) Veri hazırlama, Oluşturulan veri setindeki görsellerin sahte olanlarını sahte olarak, gerçek olanların gerçek olarak etiketlenmesidir. Ardından oluşan veri setindeki görsellerin bir kısmı test için bir kısmı eğitim için seçilmelidir.
- c) Model seçimi, ilgili görseller için en uygun modelin belirlenmesidir. Sahtecilik tespiti için bu model YOLOv8, yüz tanımlama için ise FaceNet modelidir.
- d) Model eğitimi, veri hazırlama kısmında eğitim için hazırlanan verilerin eğitilmesidir.

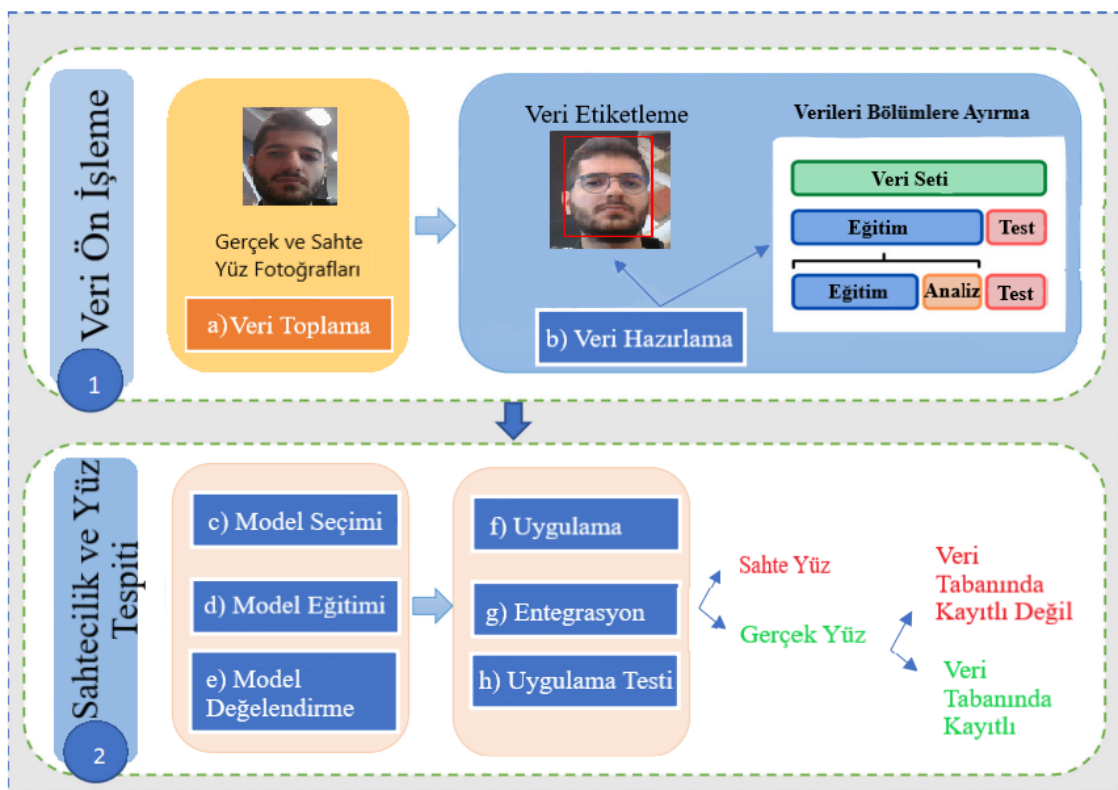
e) Model değerlendirme, veri hazırlama bölümünde test için ayrılan görsellerin test için kullanılmasıdır.

f) Uygulama, modelin uygulamalı olarak çalıştırılmasıdır.

g) Entegrasyon, modelin sisteme entegre edilmesidir.

h) Uygulama testi, uygulanan modelin sisteme entegre edildikten sonra sistem içinde test edilmesidir.

Son bölümde ise sistemde başarılı olarak test edilen modellerin, devreye alınmasıdır. Görseller ilk modelden başarılı olarak geçerse ardından yüz tanımlama geçer ve burada veri tabanına kayıtlı olup olmasına göre filtrelenir.

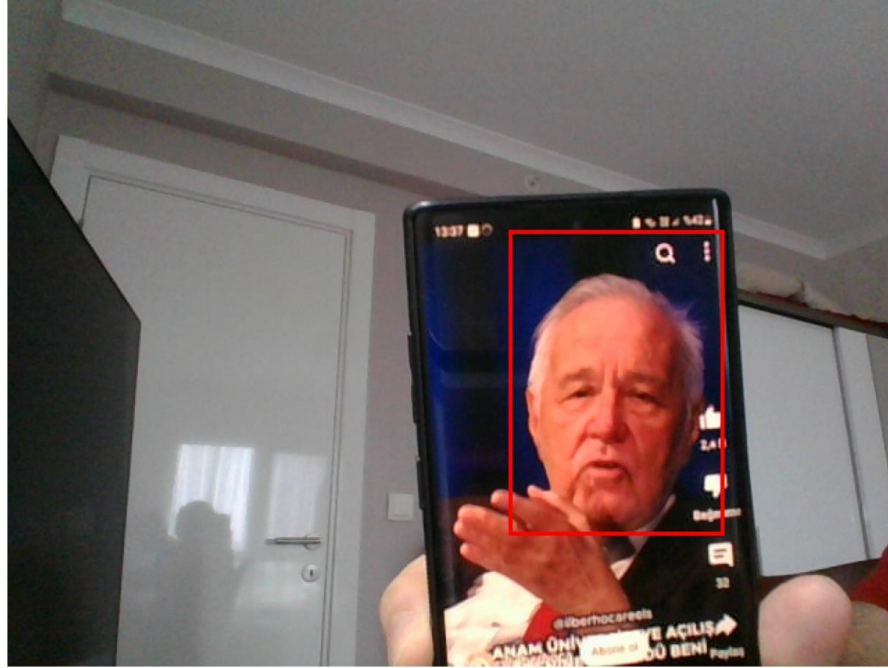


Şekil 4.1. Yüz için sahtecilik tespiti ve tanımlama işlemlerinin genel görünümü

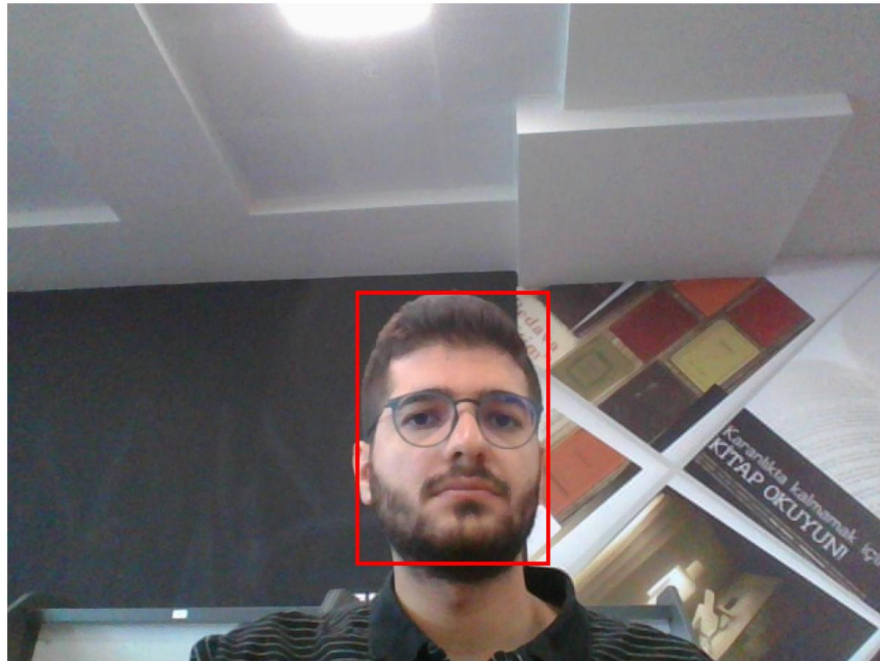
#### 4.1. YOLOv8 Eğitim Sonuçları

YOLOv8 modeli için 25 ve 100 yinelemeden (epoch) oluşan iki farklı eğitim yapıldı. 100 yineleme (epoch) için daha iyi sonuçlar alındığı için ona ait veriler gösterilecektir. Modelin eğitiminde 8736 adet, içerisinde işaretlenmiş yüzler olan fotoğraf kullanıldı. Eğitimde kullanılan örnek fotoğraflardan birisi Şekil 4.2'de gösterilmiştir,

buradaki yüz ‘‘Sahte’’ olarak etiketlenmiřtir. Őekil 4.3’te ise ‘‘Gerçek’’ olarak etiketlenen yüzler g rmekteyiz.



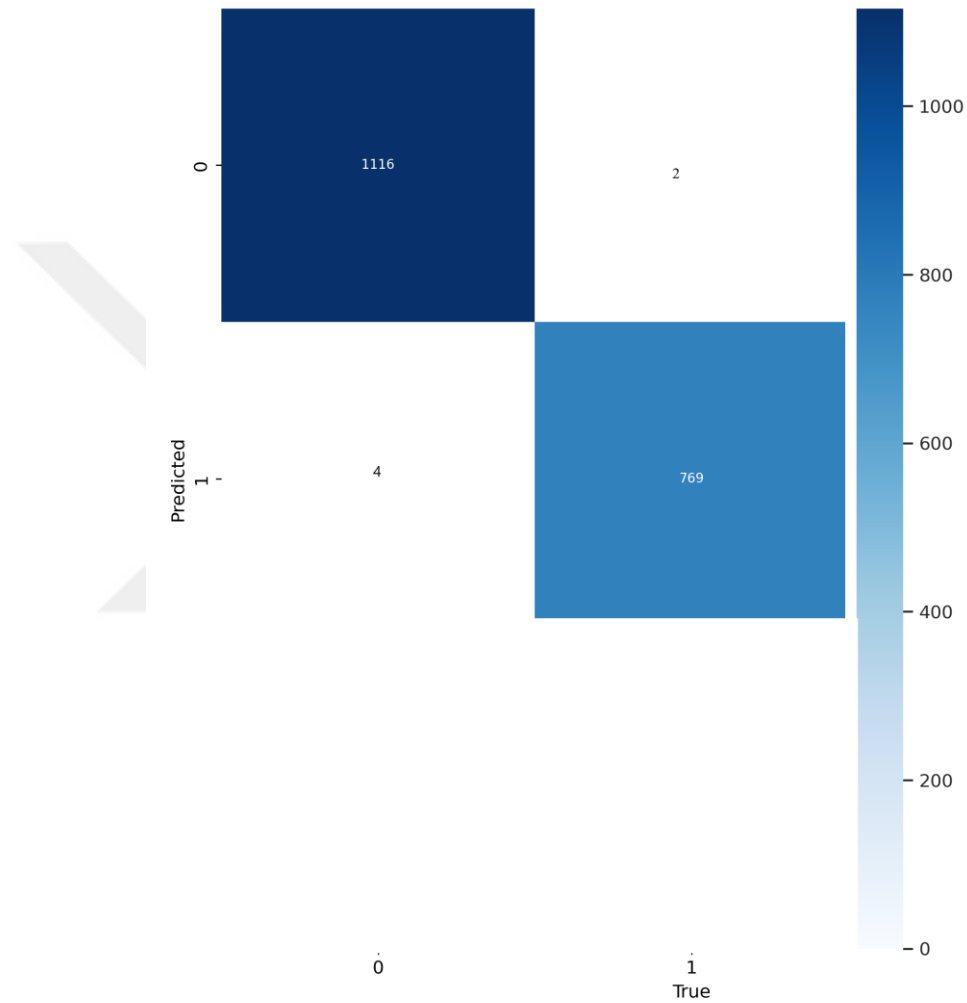
Őekil 4.2. Sahte olarak etiketlenen yüz



Őekil 4.3. Gerçek olarak etiketlenen yüz

#### 4.1.1. Karışıklık matrisi (Confusion matrix)

Karışıklık matrisi modelin genel performansını değerlendirebileceğimiz matristir. Kendi modelimizin eğitimi sonucunda oluşturulan Karışıklık Matrisi, Şekil 4.4 'te gösterilmiştir.

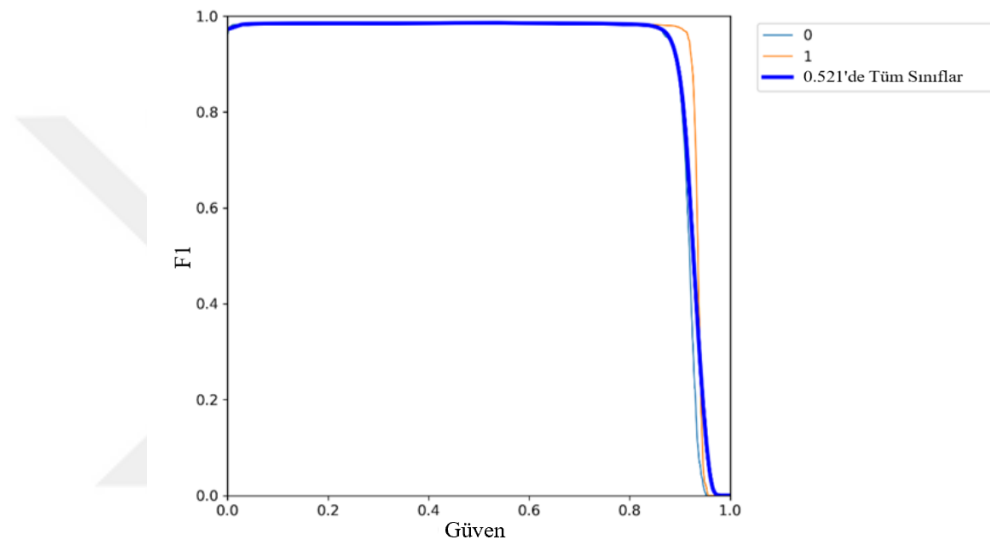


Şekil 4.4. Eğitim sonucunda oluşan karışıklık matrisi

Karışıklık matrisine göre; Eğitilen model Sınıf 0 için yani Sahte yüzler için test edilen 1120 fotoğraftan 1116'sını doğru 4 tanesini yanlış bilmiştir. Sınıf 1 için ise 771 fotoğraftan 769 tanesini doğru 2 tanesini ise yanlış bilmiştir.

#### 4.1.2. F1 güven eğrisi (F1 confidence curve)

F1-Güven Eğrisi (F1-Confidence Curve), bir modelin çeşitli güven seviyelerinde (confidence thresholds) F1 skorunun nasıl değiştiğini görselleştirir. F1 skoru, doğruluk (precision) ve hassasiyet (recall) metriklerinin harmonik ortalaması olup, modelin genel performansını tek bir metrik olarak özetler. Bu eğri, modelin farklı güven seviyelerinde nasıl performans gösterdiğini analiz etmeye yardımcı olur. Modelimize ait F1 Güven Eğrisi Şekil 4.5'te gösterilmiştir.



Şekil 4.5. Eğitim sonucunda oluşan F1 güven eğrisi

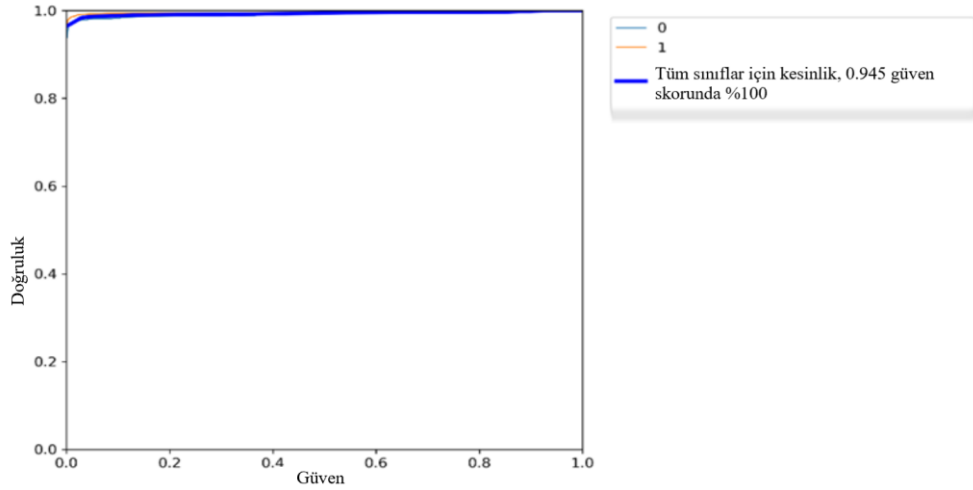
Bu şekilde; Mavi Eğri (Sınıf 0): Sınıf 0 için F1 skorunun farklı güven seviyelerinde nasıl değiştiğini gösterir. Turuncu Eğri (Sınıf 1): Sınıf 1 için F1 skorunun farklı güven seviyelerinde nasıl değiştiğini gösterir. Kalın Mavi Eğri (Tüm Sınıflar): Tüm sınıflar için ortalama F1 skorunun güven seviyelerine göre değişimini gösterir.

Eğitilen modelin F1 skorunun 0.99 değerine ulaşması, modelin çok iyi çalıştığını göstermektedir. Ayrıca her sınıf için de ayrı ayrı yüksek değerlere çıkması modelin dengeli olarak iyi çalıştığını göstermektedir.

#### 4.1.3. Doğruluk-güven eğrisi (Precision-confidence curve)

Doğruluk-Güven Eğrisi, bir modelin tahminlerinin güven seviyesine bağlı olarak doğruluk (precision) oranının nasıl değiştiğini gösteren bir grafikdir. Doğruluk oranı, modelin pozitif olarak sınıflandırdığı örneklerin ne kadarının gerçekten pozitif olduğunu

belirtir. Bu eğri, modelin tahminlerindeki kesinliği ve bu kesinliğin doğruluğa nasıl yansıdığını görselleştirmeye olanak tanır. Eğitim sürecinin sonunda elde edilen Doğruluk-Güven Eğrisi, Şekil 4.6'da gösterilmiştir.



Şekil 4.6. Eğitim sonucunda oluşan doğruluk-güven eğrisi

**X Eksen (Güven, Confidence):** Modelin tahminlerine olan güven seviyesini ifade eder ve 0 ile 1 arasında değişir. X eksen, modelin pozitif tahmin yaparken ne kadar emin olduğunu yansıtır. Örneğin, 0.8 güven seviyesinde, model sadece %80 emin olduğu tahminleri pozitif olarak kabul eder.

**Y Eksen (Doğruluk, Precision):** Doğruluk oranını gösterir ve 0 ile 1 arasında değişir. Y eksen, modelin pozitif olarak sınıflandırdığı tahminlerin ne kadarının gerçekten doğru olduğunu belirtir. 1'e yakın değerler, modelin pozitif tahminlerinin büyük çoğunluğunun doğru olduğunu ifade eder.

**Mavi Eğri (Sınıf 0):** Sınıf 0 için doğruluk oranının farklı güven seviyelerinde nasıl değiştiğini gösterir.

**Turuncu Eğri (Sınıf 1):** Sınıf 1 için doğruluk oranının farklı güven seviyelerinde nasıl değiştiğini gösterir.

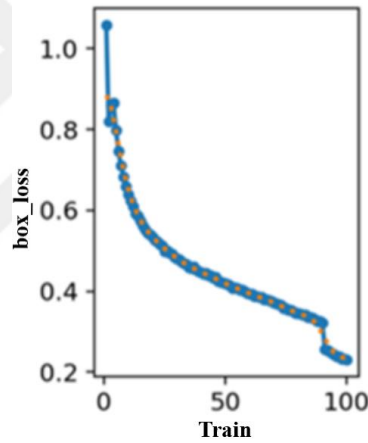
**Kalın Mavi Eğri (Tüm Sınıflar):** Tüm sınıflar için ortalama doğruluk oranının güven seviyelerine göre değişimini gösterir.

Şekil 4.5'te sistemin genel olarak 1'e yakın olarak çalıştığı görülmektedir. Bu da sistemin hem kararlı hem de doğru çalıştığını göstermektedir.

#### 4.1.4. Train/box\_loss eğrisi

YOLOv8 ile eğitilen modelin eğitim süreci boyunca sınırlayıcı kutu kaybının (box loss) nasıl değiştiğini gösterir. Sınırlayıcı kutu kaybı, modelin tespit ettiği sınırlayıcı kutuların, gerçek sınırlayıcı kutularla ne derece örtüştüğünü ölçen bir kayıp fonksiyonudur. Kayıp değeri düştükçe, modelin nesnelere doğru konum ve boyutlarda tespit etme performansı artar. Bu grafik, modelin eğitim sürecindeki iyileşmelerin bir göstergesi olarak değerlendirilir.

Sınırlayıcı kutu kaybı, modelin nesnelere doğru bir şekilde konumlandırıp konumlandırmadığını değerlendirir. Bu değer sıfıra yaklaştıkça tespitin iyi yapıldığı, bire yaklaştıkça tespitin o kadar kötü yapıldığı anlamına gelmektedir. Şekil 4.7’de eğitilen modelimizin eğitim boyunca gösterdiği box\_loss performansı gösterilmiştir.



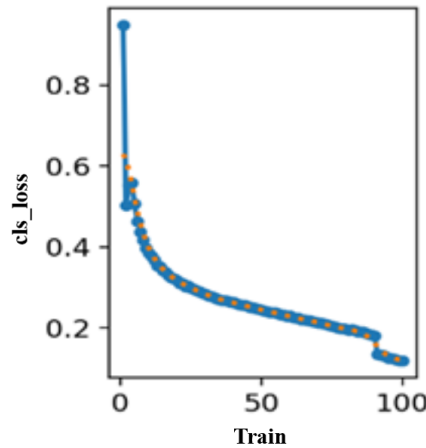
Şekil 4.7. Model eğitimi sonucunda oluşan train/box\_loss eğrisi

Grafikte, eğitim sürecinin başlangıcında box loss değerinin yüksek olduğu gözlemlenmektedir. Bu yüksek başlangıç değeri, modelin henüz doğru sınırlayıcı kutuları tahmin edemediğini ve büyük hatalar yaptığını göstermektedir. Bu durum, modelin öğrenme sürecinin başında olmasından kaynaklanmaktadır. Eğitim sürecinin ilerlemesiyle birlikte box loss değerinde belirgin bir azalma gözlemlenmektedir. Bu azalan trend, modelin nesne tespiti konusunda daha doğru tahminler yapmaya başladığını ve gerçek sınırlayıcı kutulara daha yakın tahminlerde bulunduğunu göstermektedir. Kaybın azalması, modelin eğitim boyunca daha da geliştiği ve daha doğru tahminler yaptığını gösterir. Grafikte bazı dalgalanmalar gözlemlenebilir, bu dalgalanmalar, modelin farklı minibatch'ler üzerinde eğitim yaparken karşılaştığı çeşitli veri

örneklerinden kaynaklanmaktadır. Eğitim sürecinde model, her minibatch'te farklı veri örnekleri üzerinde optimizasyon yaparken geçici hatalar yapabilir. Ancak, genel azalma trendi bu dalgalanmaların önemini azaltmaktadır. Sonuç olarak yaklaşık 100 epoch sonunda box loss değeri, 0.2 civarına düşmüştür. Bu düşük değer, modelin eğitim sürecinin sonunda sınırlayıcı kutuları oldukça doğru bir şekilde tahmin edebildiğini göstermektedir. Düşük box loss değeri, modelin nesne tespiti görevinde yüksek bir doğruluk oranına sahip olduğunu ve nesnelere doğru bir şekilde konumlandırabildiğini belirtir.

#### 4.1.5. Train/cls\_loss eğrisi

Train/cls\_loss grafiği, eğitim boyunca modelin sınıflandırma hatalarının değerinin nasıl değiştiğini 0 ile 1 arasında gösterir. Modelin tahmin ettiği sınıf ile, aslında olan sonucun ne oranda örtüştüğünü ifade eder. Örneğin, bir fotoğraftaki nesnenin doğru kategoriye (gerçek ya da sahte gibi) atanıp atanmadığını belirleyip, tüm nesnelere bu sonucunun doğruluğunun ne oranda olduğunu grafiksel olarak gösterir. YOLOv8 ile eğitilen modelimizin, eğitim süresince nasıl değiştiği Şekil 4.8'de gösterilmiştir.

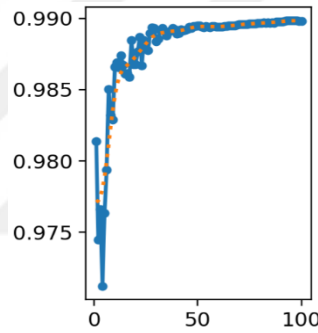


Şekil 4.8. Model eğitimi sonucunda oluşan cls/box\_loss eğrisi

Şekil 4.7'den anlaşılacağı üzere modelin, eğitimin ilk 50 epoch'a kadar olan kısmında yanlışlık oranı 0.6'lardayken, 100 epoch'a doğru 0.2'lerinde altına düşerek, güzel bir başarı oranını yakaladığını göstermektedir.

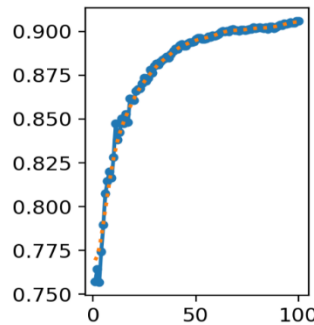
#### 4.1.6. mAP50 ve mAP50-95 eğrisi

mAP50 Eğrisi ile modelin IoU değerinin ortalama 50 olduğu zamandaki performansının grafiği gösterilirken, mAP50-95 eğrisi ile IoU değerinin 50 ile 95 arasında ortalama 0.05 aralıklarla arttırılarak nesnenin tanımlanmasının ne kadar doğru olduğu gösterilir. IoU değeri modelin tahmin ettiği sınırlayıcı kutunun (bounding box) gerçek sınırlayıcı kutu ile ne kadar örtüştüğünü ölçen bir metriktir. mAP50 eğrisi ile modelin genel olarak ne kadar doğru olduğuna bakılırken mAP50-95 ile modelin daha zorlu şartlarda ne kadar doğru çalıştığına bakılır. Eğitilen modelin mAP50 grafiği Şekil 4.9'da mAP50-95 grafiği de Şekil 4.10'da gösterilmiştir. Şekil 4.9'dan anlaşılacağı üzere, model ilk epoklardan itibaren doğru çalışmaya başlamıştır. Eğitimin sonunda da 0.990 gibi çok yüksek bir doğruluk oranına yaklaşmıştır.



Şekil 4.9. Model eğitimi sonucunda oluşan mAP50 eğrisi

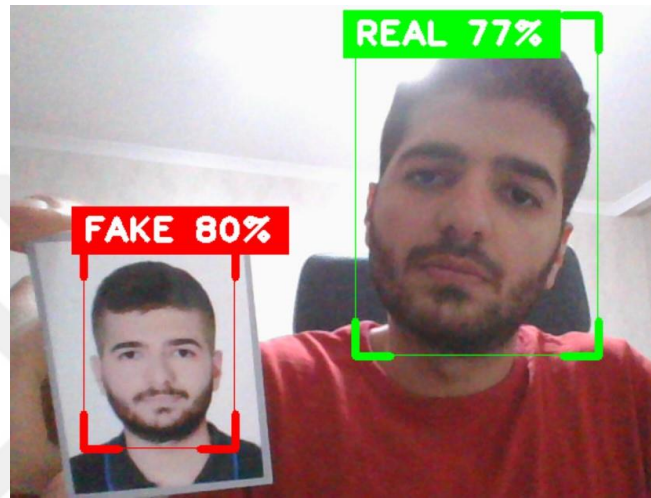
Şekil 4.10'dan anlaşılacağı üzere, model ilk epoklarda 0.75 gibi düşük sayılabilecek doğruluk oranları ile başlayıp, eğitimin sonunda 0.90 gibi yüksek değerlere kadar çıkmıştır. Bu da modelin doğruluk oranının kabul edilebilir seviyelerde olduğunu ispatlamaktadır.



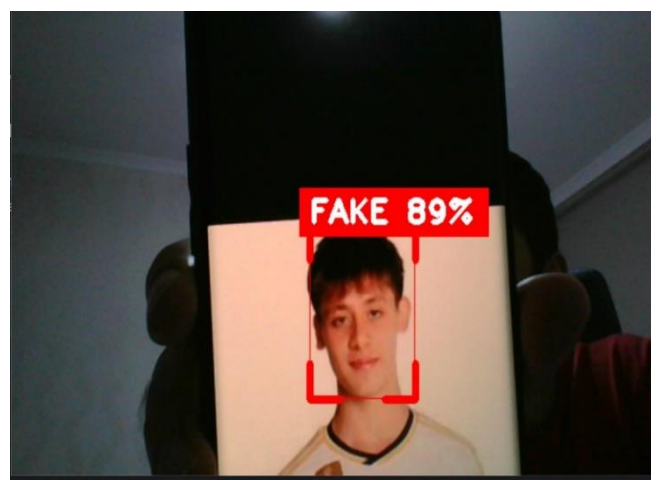
Şekil 4.10. Model eğitimi sonucunda oluşan mAP50-95 eğrisi

#### 4.1.7. Gerçek ve sahte yüzler için yapılan test sonuçları

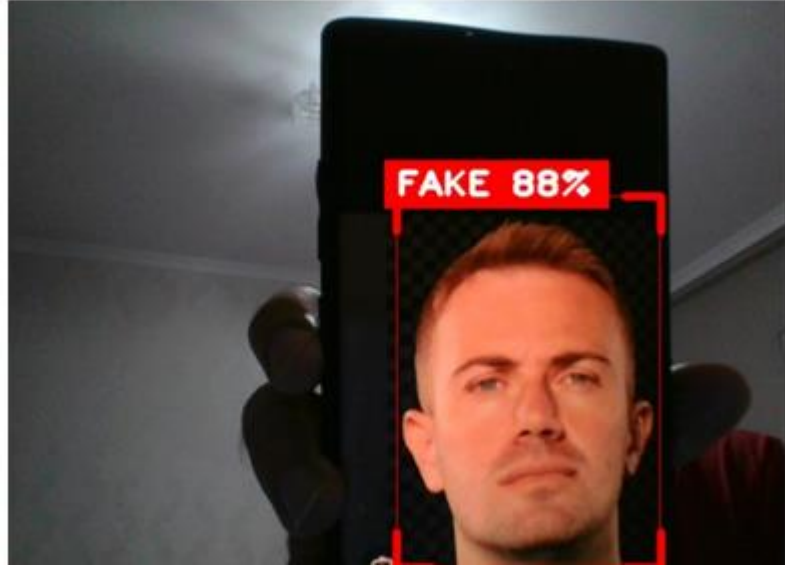
YOLOv8 ile eğitilen model ile yapılan test sonuçları Şekil 4.11, 4.12, 4.13, 4.14, 4.15 ve 4.16'da gösterilmiştir. Model gündüz ve gece şartlarında test edilmiştir ve oldukça iyi sonuçlar vermiştir, sonuçlar aşağıda gösterilmiştir. Test canlı kamera ile alınan görüntüler üzerinde eş zamanlı olarak bu tespiti yapmaktadır.



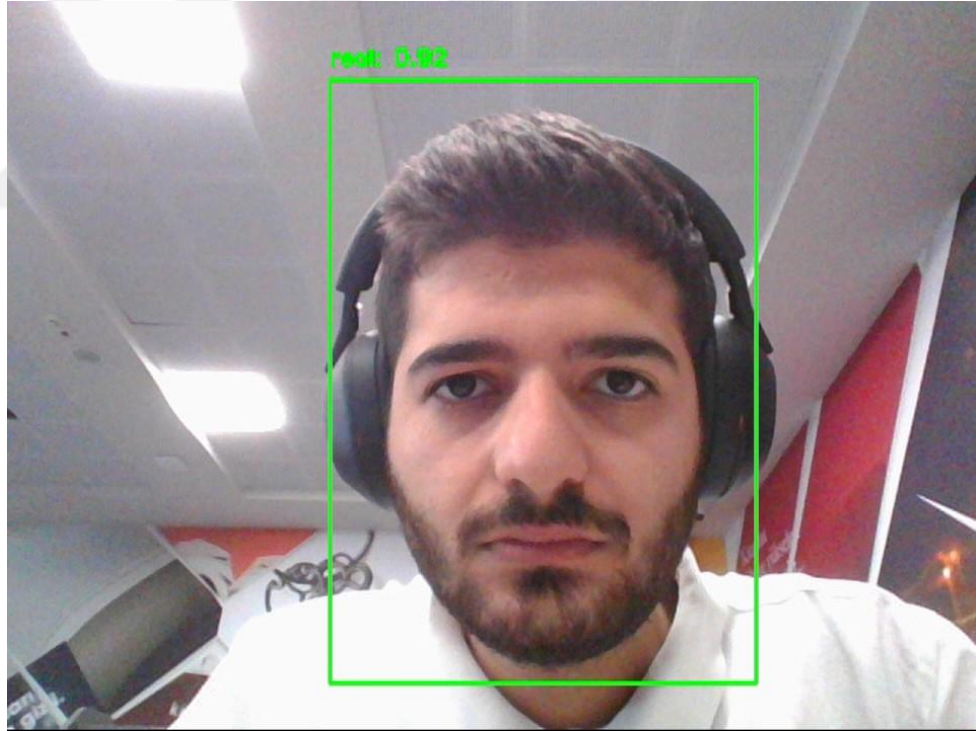
Şekil 4.11. Yüzde 80 güvenilirlik oranı ile bulunan sahte yüz ve yüzde 77 güvenilirlik oranı ile bulunan gerçek yüz tespitinin yapıldığı test sonucu



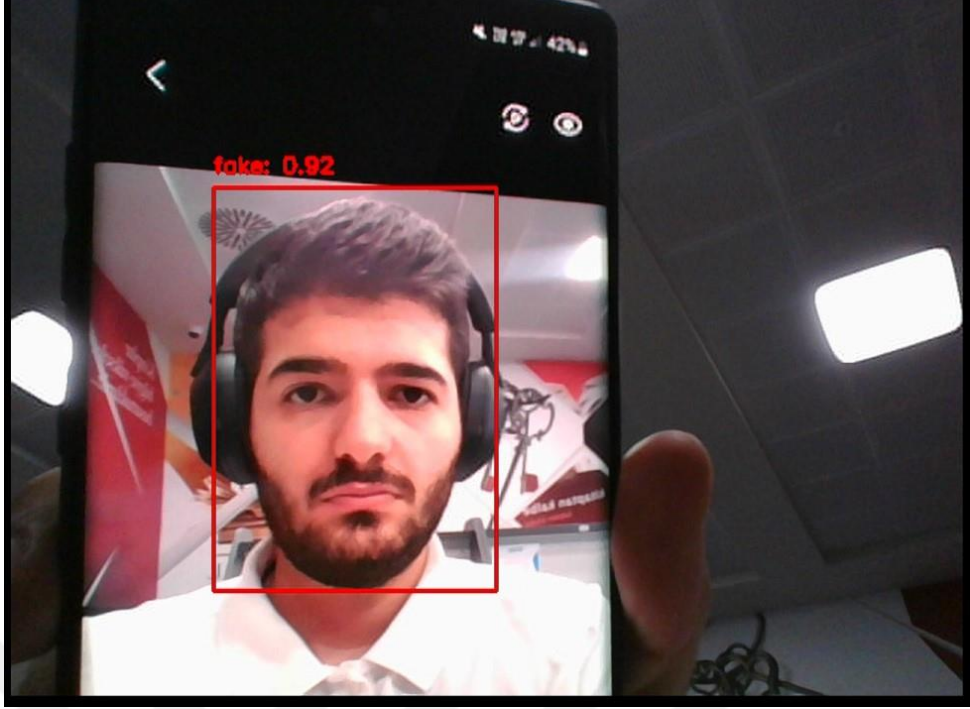
Şekil 4.12. Yüzde 89 güvenilirlik oranı ile sahte yüz tespitinin yapıldığı test sonucu



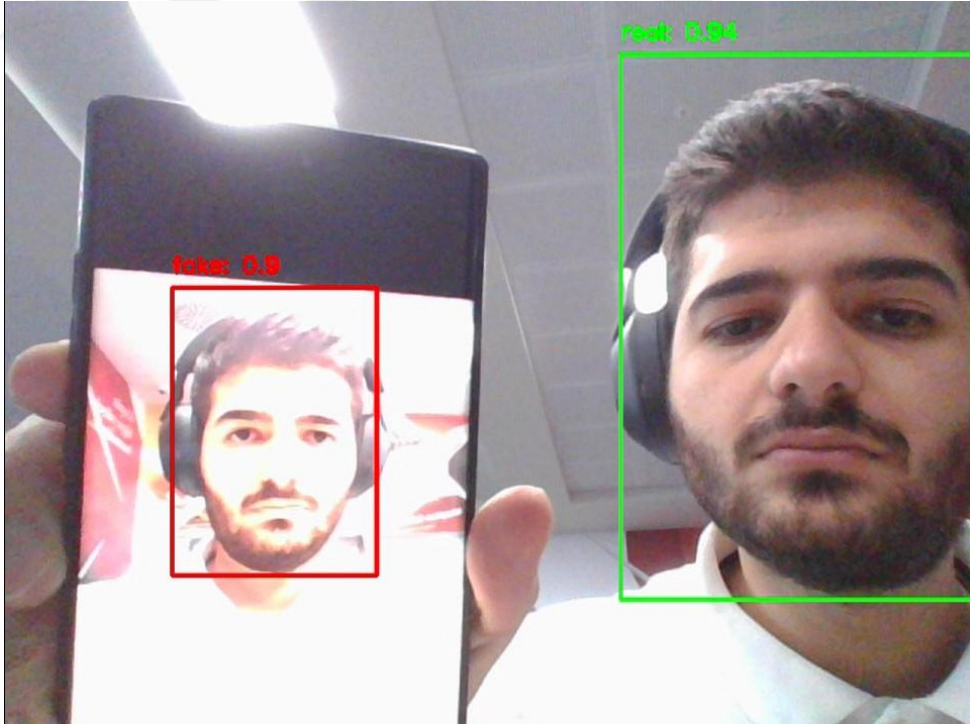
Şekil 4.13. Yüzde 88 güvenilirlik oranı ile sahte yüz tespitinin yapıldığı test sonucu



Şekil 4.14. Yüzde 92 güvenilirlik oranı ile bulunan gerçek yüz tespitinin yapıldığı test sonucu



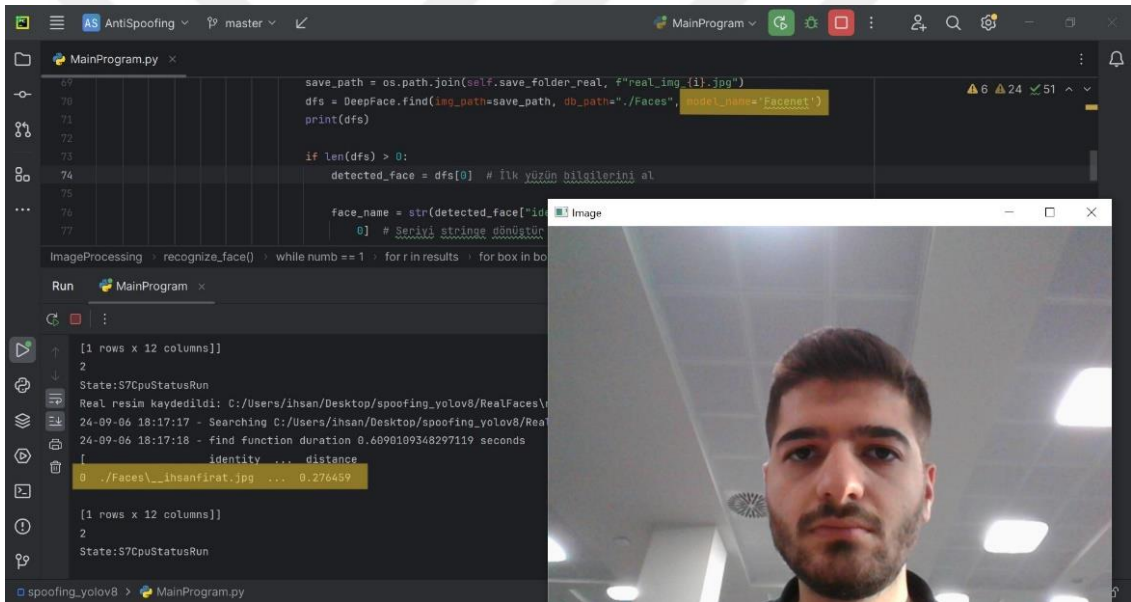
Şekil 4.15. Yüzde 92 güvenirlilik oranı ile bulunan sahte yüz tespiti yapıldığı test sonucu



Şekil 4.16. Yüzde 90 güvenirlilik oranı ile bulunan sahte yüz ve yüzde 94 güvenirlilik oranı ile bulunan gerçek yüz tespiti yapıldığı test sonucu

#### 4.1.7. FaceNet modeli test sonuçları

Çizelge 2.4 ve Bölüm 3.2.5'te gösterilen avantajlarından dolayı bu çalışmamızda FaceNet modeli kullandı. Kamera yardımıyla okunan yüzün, veri tabanında kayıtlı kullanıcılar arasında olup olmadığı kontrol edildi. Yüzün tespit edilip, veri tabanından bulunma süreci yaklaşık olarak 0,5 ile 0,6 saniye arasında sonuçlanmaktadır. Yapılan testlerden bir tanesinin sonucu da Şekil 4.16'da gösterilmiştir. Bir diğer avantajı ise kayıtlı operatörlerin yüzlerinin olduğu klasöre yeni bir yüz eklenildiğinde model bunu otomatik olarak analiz ediyor, modeli uzun uzun eğitmeye gerek kalmadan kısa bir tarama ile yüz FaceNet modeline eklenmiş oluyor.



Şekil 4.17. 0,6 saniye içerisinde doğru bir şekilde analizi gerçekleştirilen FaceNet test sonucu

## 5. SONUÇLAR VE ÖNERİLER

Bu tez çalışmasında, geleneksel olarak kullanıcı adı ve parola kullanılan fabrika otomasyon sistemlerinde güvenilirliğin ve çalışma hızının arttırılması için görüntü işleme tabanlı yüz analizi teknolojisinin kullanıldığı yeni bir topoloji geliştirildi.

Bu topolojinin kullanılabilmesi için öncelikle mevcut sisteme görüntü işleme amaçlı bir donanım ilave edilmelidir. Bu donanım kullanıcıların ihtiyacına göre basit bir jetson nano modülü veya bir bilgisayar olabilir. Ya da yüksek hafızaya sahip gelişmiş bir sunucu olabilir. Örnek olarak 5 farklı kullanıcının kullanacağı bir SCADA sistemi için jetson nano yeterli olabilirken, büyük bir fabrika da yüzlerce operatörün ve birden çok HMI'ın tek kontrol mekanizmasından kontrolü istenirse o zaman da gelişmiş bir bilgisayar ya da sunucu gerekli olmaktadır. Yaptığımız tez çalışmasında, onlarca farklı kullanıcının çalışabileceği bir senaryo için planlama yaptık ve bu sebeple görüntü işleme bilgisayar üzerinde yapıldı.

İkinci olarak, operatör ile haberleşmeyi sağlayacak bir HMI'ya ihtiyaç vardı ve bu sebeple Siemens tarafından geliştirilen TP900 Comfort panel kullanıldı. HMI operatörlerden aldığı bilgileri direkt olarak bilgisayara aktarmak yerine PLC'ye aktarmaktadır. Sistemin bu şekilde tasarlanmasının sebebi HMI işlemcisini haberleşme gibi işlem yükünü arttıracak işlemlerden koruyarak, yükü daha yüksek işlemcili PLC üzerinde gerçekleştirmektir.

Üçüncü olarak bir adet PLC'ye ihtiyaç vardır. Bu ihtiyacı karşılamak için Siemens tarafından geliştirilen S7-1200 serisinden CPU-1212 PLC kullanılmıştır. Bu PLC üzerinde 8 adet dijital giriş ve 6 adet dijital çıkış bulundurmaktadır. Daha fazla sayıda dijital giriş-çıkışlara ihtiyaç olunması halinde kolaylıkla daha fazla sayıda dijital giriş-çıkışa sahip CPU-1214, CPU-1215, CPU-1217 modülleri kullanılabilir. Modül değişiklikleri, içerisinde HMI ve PLC yazılımlarını oluşturduğumuz TIA Portal V16 uygulamasında kolaylıkla yapılabilir.

Dördüncü olarak operatörün yüzünü çekebilmek için bir kamera gerekli. Bu kameranın çözünürlüğünün yüksek olması ve karanlık şartlarda da iyi çekim kabiliyetine sahip olması önemli. Bir webcam ile de bu ihtiyaç giderilebilir bu da topolojinin esnekliği açısından katkı sağlamaktadır. Sistemimizde bilgisayar üzerinde bulunan dahili kamera kullanıldı.

Son olarak da haberleşmeyi sağlayabilmek için ethernet kabloları ve bu kabloları birleştirebileceğimiz bir switch gerekli. Vücuttaki damarlar vücut için ne denli önemliyse,

ethernet kabloları da bu topoloji için son derece önemlidir. Bazı şirketler ethernet kablolarına önem vermemekte ve kalitesiz kablolar kullanmaktadır. Bu da hem haberleşme kalitesini etkilemekte hem de ileride bozularak sistemin çalışmasını bozabilmektedir. Bu malzemeler sistemin topoloji için gerekli fiziksel malzemelerdir.

Sistemin yazılım tarafında ise görüntü işleme için PyCharm uygulaması ve Python dili, HMI ve PLC yazılımı için ise TIA Portal v16 uygulaması kullanıldı.

Yüzün sahte olup olmadığını analiz edebilmesi için YOLO v8 modeli kullanıldı. Bu modelin eğitimi için 8736 adet yüz içeren bir veri seti kullanıldı. Bu veri setinin sayısı daha da artırılabilir ve daha çeşitli yüzler ile çeşitlendirilebilir. Yüz çeşitliliğin artırılması ve çekilen yüzlerin çekildiği koşulların çeşitlendirilmesi sağlanırsa daha kaliteli sonuçlar elde edilebilir. YOLOv8 modelinin tercih edilmesi hızlı ve dengeli çalışıyor olmasıydı. Güvenliğin hayati öneme sahip olduğu yerlerde YOLOv8 kullanılması tartışmaya açık bir konudur çünkü az da olsa bu tür modellerin hata payı vardır. Alternatif olarak kullanılan yüzün farklı açılardan çekilerek sahte mi gerçek mi olduğunu tespit eden modeller mevcut ama bizim sistemimizi yavaşlatacağı için ve güvenliğin o kadar da hayati öneme sahip olmamasından kaynaklı YOLOv8 yeterli işi görmüştür, sebebi ise sisteme giriş yapan her kullanıcının yüzü çekilerek veri tabanına kaydedilmekte ve ek olarak güvenlik ile alakalı bir sıkıntının olması durumunda bu veri tabanı kontrol edilerek sıkıntı çözümlenebilmektedir. Sistemimizde kullandığımız YOLOv8 modeli 0,5 saniye içerisinde yüzün sahte mi gerçek mi olduğuna yüzde 90 doğruluk oranı ile karar verebilmektedir. Ulaşılan hız ve doğruluk yaptığımız bu uygulamayı oldukça avantajlı kılmaktadır.

Yüz tanımlama için LFW veri seti üzerinde yüzde 99,63 başarı oranına ulaşan FaceNet modeli kullanıldı ve sisteme entegre edildi. YOLOv8 modeli tarafından gerçekleştirilen yüz doğrulama işleminden sonraki aşamada kullanılan FaceNet modeli, entegre ettiğimiz sistemimizde 0,5- 0,6 saniye içerisinde kullanıcının çekilen yüzünün veri tabanındaki hangi kullanıcının yüzüne ait olduğunu bulmaktadır.

Böylelikle iki aşamadan oluşan yüz analizi, iki saniyenin altında bir sürede gerçekleşerek oldukça tatmin edici bir hıza ulaşmıştır. Bu sebeple endüstride kullanılması durumunda üretimin hızlanmasına ve üretimin güvenilirliğinin artırılmasına katkıda bulunacaktır.

Kurulan bu topoloji ile ilerleyen aşamalarda görüntü işleme alanında farklı uygulamalar da gerçekleştirilebilir. Örneğin, bir fabrikada üretilen hatalı ürünlerin kamera ile tespit edilmesi istenirse, bozuk ürünlerin tespitini yapan model, kontrol

ünitesine entegre edilerek sisteme dahil edilebilir. Bu tür örnekler daha da çeşitlendirilebilir.

Sonuç olarak günümüzde birçok alanda sıklıkla kullanılan parmak izi, yüz tanıma, ses tanıma ve yüz yapısı analizi gibi modern güvenlik sistemlerinden birisi olan yüz tanıma sistemine sahtecilik önleme özelliği de eklenerek, bu güvenlik yöntemlerine oldukça fazla ihtiyaç duyulan endüstriyel otomasyon alanına entegrasyonu başarılı bir şekilde gerçekleştirilmiştir.



## KAYNAKLAR

- Alhanaee, K., Alhammadi, M., Almenhali, N., ve Shatnawi, M., 2021, *Face Recognition Smart Attendance System using Deep Transfer Learning*, *Procedia Computer Science*, Volume 192, Pages 4093-4102, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2021.09.184>.
- Alshaikhli, M., Elharrouss, O., Al-Maadeed, S., & Bouridane, A., 2021, Face-Fake-Net: The Deep Learning Method for Image Face Anti-Spoofing Detection : Paper ID 45, 2021 9th European Workshop on Visual Information Processing (EUVIP), Paris, France, pp. 1-6, doi: 10.1109/EUVIP50544.2021.9484023.
- Anonymous, 2014, PI North America Nutzerorganisation e. V. (PNO), PROFINET System Description Technology and Application, PROFIBUS & PROFINET International (PI).
- Anonymous, 2023, Discussion on YOLOv8 Issues [online], GitHub, <https://github.com/ultralytics/ultralytics/issues/189>, [Ziyaret Tarihi: 26 Temmuz 2024].
- Anonymous, 2023, YOLOv8 Performance Metrics [online], <https://docs.ultralytics.com/models/yolov8/#performance-metrics> [Ziyaret Tarihi: 15 Haziran 2024].
- Anonim, 2024, Teslakontrol, SCADA Nedir? Endüstri 4.0'ın Altın Çağı [online], <https://www.teslakontrol.com/tr/scada-nedir> [Ziyaret Tarihi: 22 Temmuz 2024].
- Aung, H., Bobkov, A. V., ve Tun, N. L., 2021, Face Detection in Real Time Live Video Using Yolo Algorithm Based on Vgg16 Convolutional Neural Network, 2021 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), Sochi, Russia, pp. 697-702, doi: 10.1109/ICIEAM51226.2021.9446291.
- Basaran, E., Gökmen, M. & Kamasak, M. E., 2018, An Efficient Multiscale Scheme Using Local Zernike Moments for Face Recognition, *Applied Sciences*, vol. 8, no. 5, Art. no. 827.
- Bochkovskiy, A., Wang, CY., & Liao, HYM, 2020, YOLOv4: Optimal speed and accuracy of object detection, arXiv:2004.10934.
- Buhl, Nikolaj, 2023, YOLO Object Detection Explained: Evolution, Algorithm and Applications [online], <https://encord.com/blog/yolo-object-detection-guide/> [Ziyaret Tarihi: 17 Temmuz 2024].

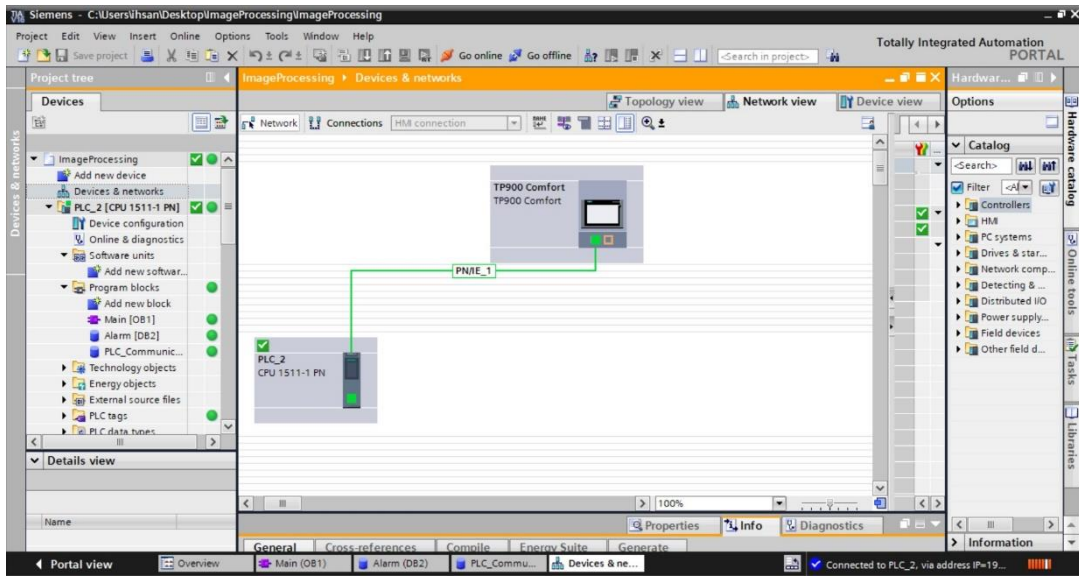
- Deng, J., Guo, J., Yang, J., Xue, N., Kotsia, I. & Zafeiriou, S., 2015, ArcFace: Additive Angular Margin Loss for Deep Face Recognition, *Journal of Latex Class Files*, vol. 14, no. 8.
- Dong, X., Yan, S., & Duan, C., 2022, A lightweight vehicles detection network model based on YOLOv5, *Engineering Applications of Artificial Intelligence*, 113, 104914.
- Ghimire, A., Werghi, N., Javed, S., & Dias, J., 2022, Real-Time Face Recognition System, arXiv preprint arXiv:2204.08978, <https://arxiv.org/abs/2204.08978>.
- Huang, C., Zhu, S. & Yu, K., 2012, Large Scale Strongly Supervised Ensemble Metric Learning, with Applications to Face Verification and Retrieval, arXiv:1212.6094.
- Jeyaraj, R., Subramanian, B., Yogesh, K., Jin, A., & Gohel, H. A., 2024, YSAF: YOLO with spatial attention and FFT to detect face spoofing attacks, 2024 IEEE 3rd International Conference on AI in Cybersecurity (ICAIC), 1-6. <https://doi.org/10.1109/ICAIC60265.2024.10433802>
- Kumar, N., Berg, A. C., Belhumeur, P. N. & Nayar, S. K., 2009, Attribute and simile classifiers for face verification, 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, pp. 365-372, doi: 10.1109/ICCV.2009.5459250
- Lei, Z., Pietikäinen, M. & Li, S. Z., 2014, Learning Discriminant Face Descriptor, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 2, pp. 289-302, doi: 10.1109/TPAMI.2013.112.
- Li, Z., Yuan, J., Jia, B., He, Y., & Xie, L., 2021, An Effective Face Anti-Spoofing Method via Stereo Matching, *IEEE Signal Processing Letters*, vol. 28, pp. 847-851, doi: 10.1109/LSP.2021.3072284.
- Lu, C. & Tang, X., 2014, Surpassing Human-Level Face Verification Performance on LFW with GaussianFace, arXiv preprint arXiv:1404.3840v3.
- Mun, H.-J., & Lee, M.-H., 2022, Design for visitor authentication based on face recognition technology using CCTV, *IEEE Access*, 10, 124604-124618. <https://doi.org/10.1109/ACCESS.2022.3223374>
- Parmar, D. N., & Mehta, B. B., 2014, Face Recognition Methods & Applications, arXiv preprint arXiv:1403.0485.
- Parkhi, O. M., Vedaldi, A., & Zisserman, A., 2015, *Deep Face Recognition*, pp. 41.1-41.12, doi: 10.5244/c.29.41.
- Parkhi, O. M., Vedaldi, A. & Zisserman, A., 2015, Deep Face Recognition, *Proceedings of the British Machine Vision Conference (BMVC)*, 1-12, University of Oxford.

- Pliatsios, D., Sarigiannidis, P., Lagkas, T., & Sarigiannidis, A. G., 2020, A Survey on SCADA Systems: Secure Protocols, Incidents, Threats and Tactics, *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1942-1976, doi: 10.1109/COMST.2020.2987688.
- Prince, S., Li, P., Fu, Y., Mohammed, U. & Elder, J., 2012, Probabilistic Models for Inference about Identity, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 144-157, doi: 10.1109/TPAMI.2011.104.
- Redmon, J., Divvala, S. K., Girshick, R. B., & Farhadi, A., 2015, You only look once: Unified, real-time object detection, *CoRR*, vol. abs/1506.02640.
- Redmon, J., ve Farhadi, A., 2018, YOLOv3: An incremental improvement, *arXiv preprint arXiv:1804.02767*.
- Schroff, F., Kalenichenko, D. & Philbin, J., 2015, FaceNet: A Unified Embedding for Face Recognition and Clustering, *arXiv preprint arXiv:1503.03832*.
- Serengil, S. I., & Ozpinar, A., 2020, LightFace: A Hybrid Deep Face Recognition Framework, 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), Istanbul, Turkey, pp. 1-5, doi: 10.1109/ASYU50717.2020.9259802.
- Simonyan, K., Parkhi, O. M., Vedaldi, A. & Zisserman, A., 2013, Fisher Vector Faces in the Wild, *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.
- Suganyadevi, S., Seethalakshmi, V., & Balasamy, K., 2022, A review on deep learning in medical image analysis. *Int J Multimed Info Retr*, 11, 19–38.
- Taigman, Y. & Yang, M., 2014, DeepFace: Closing the Gap to Human-Level Performance in Face Verification, 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, pp. 1701-1708, doi: 10.1109/CVPR.2014.220.
- Tamang, S., Sen, B., Pradhan, A., Sharma, K. & Singh, V. K., 2023, Enhancing COVID-19 Safety: Exploring YOLOv8 Object Detection for Accurate Face Mask Classification, *International Journal of Intelligent Systems and Applications in Engineering*, 11(2), 892–897.
- Terven, J., & Cordova-Esparza, D., 2023, A Comprehensive Review of YOLO: From YOLOv1 and Beyond, *arXiv preprint arXiv:2304.00501v5*.
- Rehman, Y. A., Po, L. M., & Liu, M., 2017, Deep learning for face anti-spoofing: An end-to-end approach, 2017 *Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, Poznan, Poland, pp. 195-200, doi: 10.23919/SPA.2017.8166863.

- Wang, C. Y., Bochkovskiy, A., & Liao, H. Y. M, 2022, YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696.
- Yamashita, R., Nishio, M., Do, R.K.G. , 2018, Convolutional neural networks: an overview and application in radiology, *Insights Imaging*, 9:4, 611-629, DOI: <https://doi.org/10.1007/s13244-018-0639-9>.
- Yin, Q., Tang, X. & Sun, J., 2011, An associate-predict model for face recognition, *CVPR 2011*, Colorado Springs, CO, USA, pp. 497-504, doi: 10.1109/CVPR.2011.5995494.
- Yung, N. D. T., Wong, W. K., Juwono, F. H., & Sim, Z. A., 2022, Safety Helmet Detection Using Deep Learning: Implementation and Comparative Study Using YOLOv5, YOLOv6, and YOLOv7, 2022 International Conference on Green Energy, Computing and Sustainable Technology (GECOST), Miri Sarawak, Malaysia, pp. 164-170, doi: 10.1109/GECOST55694.2022.10010490.
- Zheng, B., Xu, J., Li, H., Xing, J., Zhao, H., & Liu, G, 2017, Development of Remotely Monitoring and Control System for Siemens 840D sl NC Machine Tool Using Snap 7 Codes. In *Proceedings of the 2017 2nd International Conference on Electrical, Automation and Mechanical Engineering (EAME 2017)*, 108-112. Atlantis Press. DOI: 10.2991/eame-17.2017.26

## EKLER

### EK-1 TIA Portal v16'da kullanılan topoloji



## EK-2 PLC'de kullanılan tag tablosu

The screenshot displays the Siemens TIA Portal interface. The main window shows the 'PLC\_Communication' table for 'PLC\_2 [CPU 1511-1 PN]'. The table has columns for Name, Data type, Offset, Start value, Monitor value, Retain, Accessible f..., and Write... V... The following table represents the data shown in the screenshot:

Name	Data type	Offset	Start value	Monitor value	Retain	Accessible f...	Write... V...
Static							
RequestFromHMI	Bool	0.0	false	FALSE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
AnswerFromCamera	Int	2.0	0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
OperatorName	WString	4.0	WSTRING*	WSTRING*	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
LoginSuccessful	Bool	516.0	false	FALSE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
LoginFailed	Bool	516.1	false	FALSE	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

The interface also shows a project tree on the left with 'PLC\_2 [CPU 1511-1 PN]' selected, and a 'Find and replace' panel on the right. The status bar at the bottom indicates 'Connected to PLC\_2, via address IP=19...'.

### EK-3 PLC’de kullanılan ladder yazılımı

