İSTANBUL OKAN ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

BİR HIZLI PARMAK İZİ DOĞRULAMA SİSTEMİ

YÜKSEK LİSANS TEZİ

MÜSLİM GÜLER

tarafından

YÜKSEK LİSANS

derecesi şartını sağlamak için hazırlanmıştır.

Mayıs 2019

Program: Bilgisayar Mühendisliği

BİR HIZLI PARMAK İZİ DOĞRULAMA SİSTEMİ

YÜKSEK LİSANS TEZİ

MÜSLİM GÜLER

tarafından

İSTANBUL OKAN ÜNİVERSİTESİ

Bilgisayar Mühendisliği Anabilim Dalına

Yüksek Lisans

derecesi şartını sağlamak için sunulmuştur.

Danışman
Prof. Dr. Bekir Tevfik AKGÜN

Üye
Doç. Dr. Pınar YILDIRIM

Üye
Doç. Dr. Erchan APTOULA
(Gebze Teknik Üniversitesi)

Mayıs 2019

Program: Bilgisayar Mühendisliği

ÖZET

Bu tez çalışması, parmak izlerinden kişilerin teşhis için geliştirilmiş bir uygulamadır. Çalışmada kişilerin alınan parmak izleri alınarak küme oluşturulmuştur. Bu küme içerisinde parmak izleri bölümlendirilerek saklanmıştır. Herhangi bir kişinin parmak izleri alınıp, K-Means kümeleme algoritması ile hızlı şekilde teşhis etmesi hedeflenmiştir.

Öncelikle konu ile ilgili daha önce yapılmış olan çalışmalar araştırılmış ve konunun gerçekleşmesine temel teşkil edecek biyometrik sistemleri, parmak izi doğrulama, teşhis sistemleri ve parmak izi minutiae çıkarım algoritmaları araştırılmıştır. Parmak izi teşhisi için uygulama geliştirilmiş, uygulamada parmak izi alım ve parmak izi teşhis modülü oluşturulmuştur.

Bu uygulama ile parmak izlerinin minutiaeları çıkarma algoritmaları denemiştir. Uygun parmak izi minutiae çıkaran algoritma seçilerek, minutiaelar ile K-Means kümeleme algoritmasına kümelenerek parmak izi görüntüsü oluşturulmuştur. Oluşturulan parmak iz üzerinde teşhis işlemi uygulanmış ve örnek senaryolar incelenmiştir. Son olarak ise uygulama sonrası elde edilen sonuçlar ve değerlendirmeler özetlenmiştir.

Anahtar Kelimeler: Parmak İzi, Teşhis Sistemleri, K-Means Kümeleme, Minutiae

ABSTRACT

This thesis is an application developed for the identification of persons from fingerprints. In this study, the fingerprints of the individuals were taken and a cluster was formed. In this cluster, fingerprints were partitioned and stored. It is aimed that any person can get fingerprints and idenfication them quickly with K-Means clustering algorithm.

Firstly, the previous studies on the subject have been researched and biometric systems, fingerprint verification, idenfication systems and fingerprint minutiae inference algorithms have been investigated.

The application was developed for fingerprint idenfication, and fingerprint acquisition and fingerprint idenfication module was created. With this app you have tried the extraction algorithms of fingerprints minutiae. A fingerprint image was created by using the appropriate fingerprint minutiae extracting algorithm and clustering to the K-Means clustering algorithm. Applied to a idenfication procedure on the generated fingerprint and sample scenarios were examined. Finally, the results and evaluations obtained after the application are summarized.

Keywords: Fingerprint, Idenfication Systems, K-Means Clustering, Minutiae

TEŞEKKÜR

Tez çalışmam sırasında kıymetli bilgi, birikim ve tecrübeleri ile bana yol gösterici ve destek olan değerli danışman hocam Sayın Prof. Dr. Bekir Tevfik AKGÜN' e sonsuz teşekkür ve saygılarımı sunarım.

Yüksek lisans eğitimim boyunca yardım, bilgi ve tecrübeleri ile bana sürekli destek olan Sayın Üzeyir Hakan TOPCU olmak üzere iş arkadaşlarıma teşekkür ederim.

Çalışmalarım boyunca maddi manevi destekleriyle beni hiçbir zaman yalnız bırakmayan aileme de çok çok teşekkür ederim.

MÜSLİM GÜLER

MAYIS 2019

İÇİNDEKİLER

TABLO LİSTESİ

ŞEKİL LİSTESİ

KISALTMALAR

T.C		Türkiye Cumhuriyeti	Republic of Turkey
K-Means	**:**	K-Ortalama	K-Clustering
LFS	**:**	Sıra kontrölü hesaplamak için bir araç	A tool for calculating the sequenche
IQS	**:**	Görüntü Kalitesi Belirtmeleri	Image Quality Specifications
IAFIS	**:**	Bütünleşmiş Otomatikleştirilmiş Parmak İzi Tanımlama Sistemi	Integrated Automated Fingerprint Identification System
WSQ	**:**	FBI WSQ da görüntü sıkıştırma	Wavelet Scalar Quantization
BOZORTH	**:**	Standart FBI algoritmasına dayanan parmak izi karşılaştırması için bir araç	A tool for fingerprint comparison based on the standard FBI algorithm
AFIS		Otomatikleştirilmiş Parmak İzi Tanımlama Sistemi	Automated Fingerprint Identification System
SourceAFIS	**:**	Açık kaynak parmak izi teşhis	Opensource fingerprint matcher

DPI	Dijital görüntüleme sistemlerinde, çözünürlük olarak kullanılan, "DOTS PER INCH" kelimelerinin baş harfleri ile kısaltılmış bir terim.	Dots per inch is a measure of spatial printing, video or image scanner dot density, in particular the number of individual dots that can be placed in a line within the span of 1 inch (2.54 cm)
ORM	İlişkisel veritabanı ile nesneye yönelik programlama için kullanılan terimdir	Object Relational Mapping

1. GİRİŞ

Sosyal yaşam konforunu arttırmak için kullanılan iletişim teknolojileri geliştikçe güvenliğe duyulan ihtiyaç da giderek daha fazla önem kazanmaktadır. Örneğin; kredi kartı ile bir ödeme yapabilmemiz için kullandığımız sayısal sisteminin bir şekilde bize güven vermesi gerekir. Bunu sağlayan araçlardan bazıları; kredi kartları, akıllı kartlar gibi fiziksel nesneler ve bunlarla birlikte kullanılan PIN numaraları, şifreler, sayısal şifreleme algoritmaları ve benzerleri güvenlik araçlarıdır. Bu tür güvenlik tedbirleri için kişiler devamlı olarak kart taşımalı, bunları aktif hale getirmek için de PIN numaraları, şifreler gibi birtakım sembol ve karakterleri akıllarında tutabilmelidirler. Ayrıca kartların kaybolması veya şifrelerin unutulması da günlük hayatımızı olumsuz yönde etkileyen en önemli sorunlardan biridir. Bu sorunlarının önüne geçebilmek için kullanılan güvenlik sistemlerinde, en önde gelen ve halen üzerinde araştırmaların devam ettiği [çözüm, ya da güvenlik sistemi] Biyometridir.

Biyometri veya Biyometrik terimi Yunancadan gelme olup Bio (canlı) ve Metrikos (ölçme) kelimelerinin birleşimi olan canlı-ölçme anlamındadır. Biyometrik sistemlerin; suçlu tespitinden, akrabalık tespitine, bilgisayar ağ güvenliğinden PDA'lerin güvenliğine, yüksek seviyede güvenlik gerektiren alanların giriş çıkış kontrolünden, Personel Yoklama ve Giriş-Çıkış kontrolleri ağırlıklı olmak üzere birçok alanda kullanım olanağı bulunmaktadır. Sayısal donanım ve yazılım teknolojilerindeki gelişmelere paralel olarak Biyometrik Sistemler, sosyal yaşamda geleneksel

anahtarların, şifre-giriş kontrollü sistemlerin, akıllı kart uygulamalarının, pin kodlarının yerini almaktadır.[1]

Biyometrik modeller, bireylerin davranışsal veya fiziksel özellikleri ile otomatik tanımlama yapmamızı sağlar. Biyometrik yaklaşımlar içinde yüz, iris ve avuç içi, diğer modellere göre yeni bir biyometrik özellikler olup, yüz, avuç içi ve iris ise son yıllarda ilginin arttığı ve kullanılmaya başlanan özellikler olmuşlardır. Bunda özelliklerin işlenmek üzere elde edilmesinin kolaylaşması önemli bir etkendir. Biyometrik tanıma sistemleri için öncelikle bir görüntü kaydedilir. Kaydedilen bu görüntü sayısal koda çevrilir. Bu kod da yapılan işleme göre şifrelenir ve bilgisayara kaydedilir. Daha sonra kullanıcı herhangi bir cihaz kullanarak kendini sisteme tanıtır. Kullanıcının kendini sisteme tanıttığı andaki duruşu ve çevre koşullarından dolayı sistem de kayıtlı olan sayısal kod ile doğrulama aşamasında üretilen kodun birbiriyle tamamıyla aynı olma olasılığı yoktur. Çünkü yüz tanıma ve iris tanıma sistemlerinde kullanıcının bakış açısı ve ortam ışıklandırması, parmak izi, el tanıma ve avuç içi tanıma sistemlerinde kullanıcının parmağını veya elini sisteme tanıtma açısı, cihazın kirliliği, parmağın kirliliği veya nem gibi etkenlerden dolayı birebir aynı kod üretilemez. Bunun için, iki kod sistem yöneticisi tarafından belirlenmiş olan belli bir yüzde tutuncaya kadar karşılaştırılır [2]. Sistemin güvenilirliği için düzenlenen algoritmaya göre istenilen orana ulaşıldığında doğrulama işlemi tamamlanmış olur ve sisteme giriş için onay verilir.

Biyometrik uygulamalar için de en yaygın olarak kullanılmaya başlanan özellik parmak izi tanıma yöntemi olmuştur. Parmak izinin kolay elde edilebilir olması tanıma

sisteminde yapılan çalışmaları kolaylaştırmıştır. Yapılan çalışmalara bakıldığında parmak izinin güvenilir bir ayırt ediciliğe sahip olduğunu görülmektedir [3]. Bir bireyi ayırt edilecek özellik parmak izidir. Bireyin parmak izi diğer bireylerden farklıdır. Oluşturulan yöntemde bir bireyi on parmak izlerini sınıflandırılarak modele tanıtılır. Bu modelle parmak izi kümesinin bir alt kümesinde kayıtlı olup olmadığı sorgulanarak teşhis işlemi gerçekleştirilir. Bu yöntem geliştirilirken minutiae tabanlı tanıma algoritma modeli kullanılmıştır. Modeli kullanan iki ayrı kütüphane ile daha yüksek başarım elde etmeyi hedeflenmiştir. Bunlar SOURCEAFIS ve LFS kütüphanesidir [4].

2. BİYOMETRİK TÜRLER VE SİSTEMLER

Bu bölümde genel olarak biyometrik türler, biyometrik sistemler ve parmak izi hakkında kısaca karşılaştırmalı bilgi verilecektir.

2.1. Biyometrik Türler

Biyometri, insanları fizyolojik, davranışsal ve biyolojik özellikleri ile tanımlamaktadır. Biyometri iki kategoriye ayrılır: Fizyolojik biyometri ve davranışsal biyometri [6]. Fizyolojik biyometri, yüz, iris, parmak izi, parmak damarı, el geometrisi vb. fizyolojik veya biyolojik özelliklerden bireyleri tanıyanlardır. Öte yandan davranışsal biyometri, el yazısı, imza veya ses gibi insan davranışlarından bireyleri tanıyanlardır.

Günümüzde biyometrik tanımlama sistemi kullanılmaktadır. Bu tür sistemlerin çoğu parmak izi, yüz, iris, sese dayanmaktadır. Her teknolojinin biri birine göre avantaj ve dezavantajları bulunmaktadır. Bu sebeple uygulanacak yere, alana uygun bir biyometrik tanımlayıcı seçimi söz konusudur.

Teorik olarak kişiye ait fizyolojik veya davranışsal bir özelliğin kimlik tespitinde kullanılabilmesi için bu özelliğin aşağıdaki şartları sağlaması gerekmektedir [17].

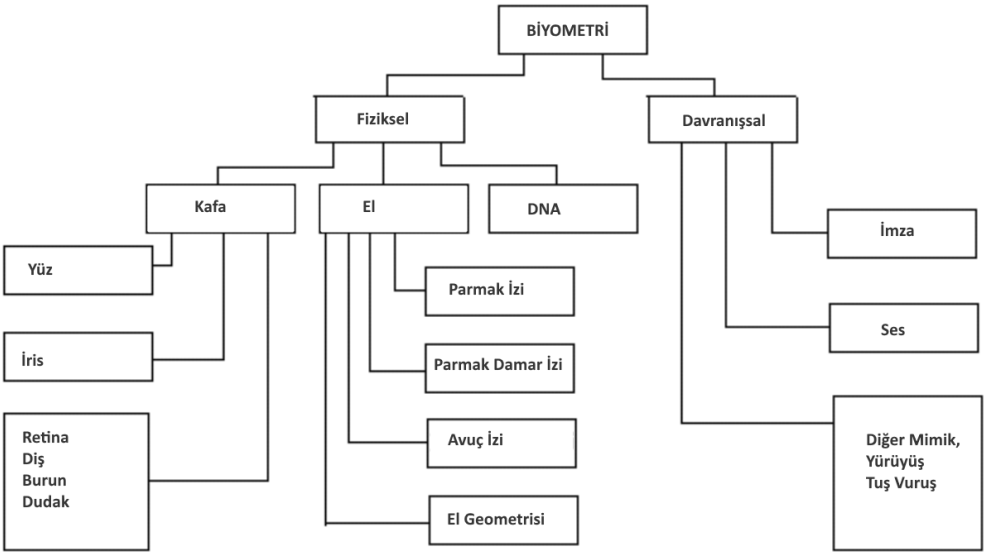
- Genellik: Bir biyometrik özellik her insanda bulunan genel bir özellik olmalıdır.
- Eşsizlik: Biyometrik özelliğin karakteristiği her insanda farklı olmalıdır.

- Süreklilik: Biyometrik özellik zamanla değişmemeli, en azından hayatın büyük bölümünde sabit kalan parametreleri bulunmalıdır.
- Ölçülebilirlik: Biyometrik özellik kişilerden kolayca alınabilir, ölçülebilir, işlenebilir, kaydedilebilir ve karşılaştırılabilir olmalıdır.

Yukarıda belirtilen şartları sağlayan herhangi bir özellik biyometrik özellik olarak kabul edilebilirken, bir biyometrik sistemin pratikte kullanılabilmesi için aşağıdaki şartları sağlaması beklenmektedir[18].

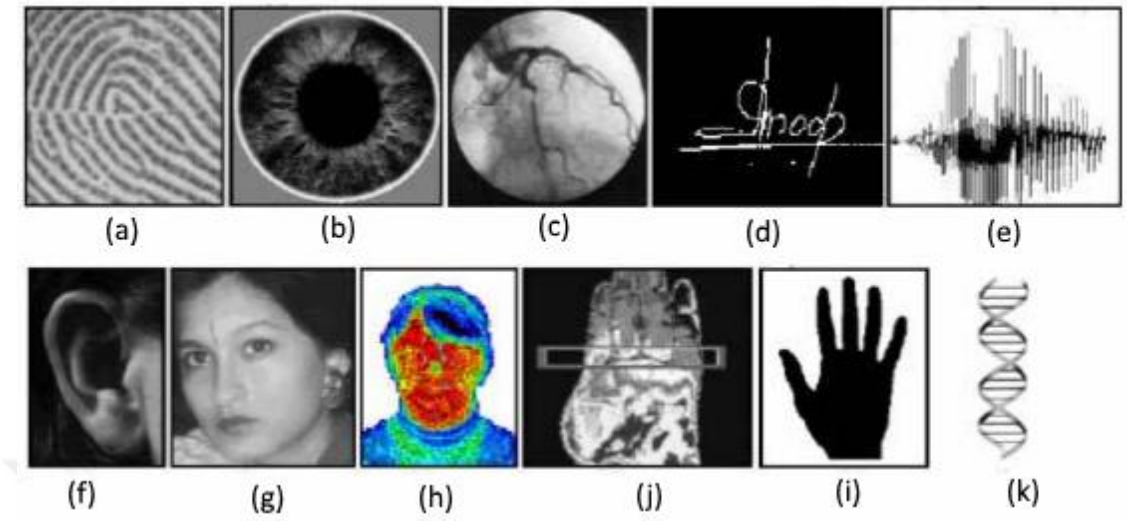
- Performans ve Doğruluk: Doğru kişiyi yanlış kişi olarak algılayıp kabul etmeme oranı veya yanlış kişiyi doğru kişi olarak algılayıp kabul etme oranının sıfır (kabul edilebilir ölçülerde) olması gerekmektedir.
- Kabul Edilebilirlik: Bir biyometrik sistem, sistemi kullanacak insanların çekinmeden ve kolayca kullanabilmelerine olanak sağlayacak şekilde tasarlanmalıdır.
- Güvenilirlik: Sistemin başka kişiler tarafından devre dışı bırakılma olasılığının sıfır (kabul edilebilir ölçülerde) olması gerekmektedir.
- Uygunluk: İşlem zamanı ve maliyeti açısından uygun olmalıdır.

Şekil 2.1.' de biyometrik sistemlerin sınıflandırılması verilmiştir. Bunlardan en çok kullanılanları; fizyolojik özelliklerden parmak izi, retina, yüz ve davranışsal özelliklerden ise imza ve sestir.



Şekil 2.1. Biyometri türlerinin sınıflandırılması

Biyometrik türler, şifre yaklaşımının aksine kişilerin bildiği bir bilgi veya taşıdığı bir objeyle değil, kişilerin kendisinde var olan ve fiziksel olarak kendilerine sıkı sıkıya bağlı olan bir özellikle kimliklendirilmelerine olanak sağlamaktadır. Biyometrik temelli tanıma türlerinin klasik yaklaşımlarda var olan kaybedilme, unutulma, çalınma, tahmin veya taklit edilebilme, diğer kullanıcılarla ortak kullanılma, paylaşılma riskini neredeyse yok eden veya çok karmaşık hale getiren bir teknolojidir.



Şekil 2.2. Biyometrik karakteristikler a) Parmak izi b) İris c) Retina d) İmza e) Ses f) Kulak g) Yüz h) Yüz- ısı dağılımı j) El ısı dağılımı i) El geometrisi k) DNA

Biyometrik tanıma türlerinin çeşitlerinin taradıkları özellikler tablo 2.1'de verilmiştir.

Tablo 2.1. Biyometrik tanıma sistemlerinin çeşitlerinin taradıkları özellikler

Biyometrik Karakteristik	**Özellikler**
Parmak İzi	Parmak satırları, gözenek yapısı
İmza Tanıma	Basınç ve hız ile yazma farkları
Yüz geometrisi	Göz, burun vs. arası uzaklıklar
İris Tanıma	İris deseni
Retina	Retina yapısına (desenine) göre
El Geometrisi	Parmak ve avuç içi ölçülerine göre
Parmak geometri	Parmak ölçme
El Damar yapısı	Elin arkası, parmak veya avuç içi damar yapısı
Kulak formu	Kulağın belirgin boyutları
Ses	Ton ya da ses rengi
DNA	Kalıtsal bir taşıyıcı olan DNA
Koku	Kokunun kimyasal bileşimi
Klavye vuruş	Klavye vuruşlarının ritmi

2.1.1. Yüz Tanıma

Yüz tanıma konusunda yapılan çalışmaların temeli 30 yıl öncesine dayanmakla birlikte son 15 yıldır bu çalışmalar cazibesini arttırmış son yıllarda askeri, ticari ve yasal uygulama alanlarının artması nedeniyle yüzlerin otomatik olarak tanınması çok popüler bir konu haline gelmiştir. Görüntü işleme, yapay zekâ ve yapay sinir ağları gibi alanlar bu konuyla yakından ilgilenmektedir. Dolayısıyla son yıllarda yüz resimlerindeki bilgiyi işleyip resmi analiz edebilecek tam otomatik bir yüz tanıma sistemi için güvenilir, hızlı ve verimli birçok algoritma geliştirilmiştir. Yüz işleme ile ilgili işlemler yüz tanıma, yüz takibi, poz kestirimi, yüz ifadesi analizi şeklinde gruplandırılabilmektedir. Yüz tanımada gerçekleştirilen işlemler aşağıda verildiği şekilde sıralanabilmektedir [7].

1. Resimlerdeki yüzlerin sezilmesi.
2. Yüze konumlandırılması.
3. Yüz sınırlarının belirlenmesi.
4. Özniteliklerinin bulunması.
5. Özniteliklerin kullanılarak yüzlerin tanınması.

Yüz tanıma teknolojisinin avantajlarından biri, bireyleri uzaktan tanımlama; bu nedenle, herhangi bir cihaza dokunarak insanları rahatsız etmeyecektir. Video kameralar gibi farklı aygıtlar tarafından görüntüler yakalanabilir. Öte yandan, bu teknolojinin bazı dezavantajları vardır. Örneğin, çekilen görüntünün kalitesi, ışığa, arka plana ve kameranın açısına bağlıdır. Kullanıcıların görünümleri zaman içinde

değişebilir ve gözlük, sakal, bıyık, makyaj veya farklı saç stillerine sahip olmanın doğruluk üzerinde etkisi olur. Ayrıca, daha yüksek hassasiyete sahip olmak, kullanıcıların daha fazla hafıza alanı gerektiren farklı görüntülerin kaydedilmesini gerektirir.

2.1.2. İris Tanıma

İris tanıma 1990'ların başında geliştirilmiştir. Kişilerin iris desenlerinin analiz edilmesine dayalı bir sistem olan iris tanıma sistemleri, kişinin sahip olduğu iris şeklinin kişinin yaşamı süresince değişmediği gerçeğinden yola çıkılarak geliştirilmiştir. Yapılan çalışmalarda irisin 400 farklı karakteristik özelliğe sahip olduğu bunlardan 173 tanesinin iris tanıma sistemlerinde kullanıldığı rapor edilmektedir. İris tanıma sistemlerinde genellikle yakın infrared ışıklı kamerayla yakından çekilmiş (0.1 ile 1 metre arası) monokrom resimler kullanılmaktadır. İris tanımada gerçekleştirilen işlemler aşağıda verildiği şekilde sıralanabilmektedir.

1. İris resminin alınması.
2. İrise konumlandırılması.
3. İrisin iç ve dış sınırlarının belirlenmesi
4. Analizde kullanılabilecek yapının filtrelenmesi.
5. Analizde kullanılabilecek yapının özellik vektörlerinin elde edilmesi.
6. İris kodunun hesaplanması.
7. İris kodunun karşılaştırılması.

Bir iris resminde, irise konumlandırılması, irisin iç ve dış sınırlarının belirlenmesi ve 2048 bitlik bir alfa nümerik ifade olan iris kodunun grafiksel gösterimi Şekil 2’ de verilmektedir. İris kodunun karşılaştırılmasında genellikle Hamming mesafesi kullanılmaktadır.

Şekil 2.3. İrisin sınırları ve iris kodunun grafiksel gösterimi

Genellikle havaalanları gibi kimlik doğrulama gerektiren giriş çıkış kontrol noktalarında iris tanıma yöntemi kullanılmaktadır. Bu yöntemle gözleri görmeyen, nistagmus hastalığına sahip (gözleri titreyen) veya irisleri olmayan kişilerin kimliklendirilmesi mümkün değildir. Ayrıca iris resmi alınırken gözlerin durumu, göz kapaklarının ve/veya kirpiklerin iris desenini bozması gibi faktörler sistemi olumsuz yönde etkilemektedir.

2.1.3. Ses Tanıma

Ses taraması için frekans, kısa süreli diyalog spektrumu ve spektrogramlar (zaman frekansı-enerji modelleri) gibi çeşitli vokal nitelikleri kullanılır [8]. Cihazlardan örnek verilirse bu teknoloji için kullanılan mikrofonlar oldukça ucuz. Bu sistem, kullanıcıların bir ifade seçmesine ve tanımlama ve doğrulama sırasında tekrar etmesine izin verir. Bu da aynı cümleyi tahmin edip sisteme girme riskini azaltır. Ancak gürültü ve yankılar sistem doğruluğunu azaltabilir. Ayrıca ses, hastalık sırasında veya farklı ruh hallerinde değişebilir ve bu da kimlik doğrulamasını sorunlu hale getirir. Ses tabanlı sisteme kayıt diğer biyometrik sistemlere göre daha uzundur, çünkü kullanıcıların ifadeyi birçok kez tekrarlamaları gerekir.

2.1.4. Parmak İzi

Parmak izi, dünyada en çok kabul görmüş, biyometrik doğrulama ve teşhis işlemlerinde kullanılan tanımlayıcı sistemlerdir [9]. Değişik kullanım amaçları için geliştirilmiş algılayıcılar kullanılarak parmak izine ait görüntü elde edilir ve doğrulama/teşhis metotlarından birisi aracılığıyla işlem gerçekleştirilir. Ayırt ediciliği çok yüksektir.

Tüm biyometrik tanımlama sistemleri sadece üstün olan tek bir özellikleriyle değerlendirilmezler. Sistemin evrensel oluşu, toplanabilirliği (koleksiyon), genel kabul görürlüğü vb. biyometrik tanımlayıcılar bir bütün olarak değerlendirilmektedir. Bu konu hakkında detaylı bilgi bölümün ilerleyen kısımlarında verilecektir.

2.2. Biyometrik Sistemler

Biyometri, biyolojik verileri yani bireyin kişisel bir nitelik ya da davranışını analiz ederek kimliğini doğrulama bilimidir [11]. Hayatımızda büyük önem taşıyan biyometrik tabanlı doğrulama güvenilir kimlik doğrulaması için güçlü bir metottur.

Biyometrik sistem, söz konusu şahsın sahip olduğu spesifik fiziksel veya davranışsal özelliklerinin gerçekliğine karar veren örüntü tanımlama sistemidir. Kullanışlı bir biyometrik sistemde önemli bir husus da bireyin nasıl tanımlandığına karar vermektedir. Uygulama durumuna bağlı olarak, biyometrik sistem doğrulama ya da kimliklendirme sistemi olarak adlandırılmaktadır [12]. Doğrulama sistemi, elde edilen biyometrik özelliklerin daha önce sistemde kayıtlı olarak saklanan veri kalıplarıyla karşılaştırılması sonucu bireyin kimliğinin gerçekliğini bulmaktadır. İddia edilen şahıs olunup olunmadığına karar vermek için birebir karşılaştırma yapılmaktadır. Doğrulma sistemi iddia edilen kişi olunup olunmadığını kabul ya da ret etmektedir [13]. Kimliklendirme sistemi, karşılaştırma için bütün şablon veri tabanlarını araştırarak bireyi tanımlamaktadır. Bireyin kimliğini kurmak için çoklu seçenek arasından birebir karşılaştırma yapmaktadır. Kimliklendirme sisteminde, özne ne olup olmadığını iddia etmeden onun ne olduğunu tespit etmektedir.

Gerçekleme terimi biyometrik alanda sıklıkla kullanılır ve bazen doğrulama ile aynı anlamda kullanılmaktadır. İşin aslı, bilgi teknolojileri terminolojisinde, kullanıcıyı gerçekleme sistemi moddan (doğrulama, kimliklendirme) bağımsız olarak kimliği bilme izni vermektedir [14].

Kayıt modülü, şahısları biyometrik sistemin veritabanına kaydetmekten sorumludur. Kayıt aşamasında özelliğin ham dijital gösterimini elde etmek için ilk önce bireyin biyometrik özelliği biyometrik okuyucu aracılığıyla taranmaktadır. Elde edilen örneğin başarılı bir içleme sürecinden geçmesi için genellikle güvenilir bir kalite kontrole tabi tutulmaktadır. Karşılaştırmayı kolaylaştırmak amacıyla, şablon (kalıp) denilen yoğun ama açıklayıcı bir sunum elde etmek için ham dijital gösterim genellik özellik saptayıcılar tarafından işlenmektedir. Uygulamaya bağlı olarak şablon (kalıp) biyometrik sistemin merkezi veritabanında saklanır veya şahsa verilen manyetik ya da akıllı karta kaydedilmektedir.

Doğrulama görevi bireyi giriş anında doğrulamaktan sorumludur. İşlem sırasında kullanıcı adı ya da şifre klavye aracılığıyla sisteme girilmektedir. Biyometrik okuyucu bireyin tanımlanması gereken özelliklerini alır ve onları dijital bir formata dönüştürmektedir. Bu özellikler daha sonra daha yoğun dijital sunum elde etmek için özellik saptayıcı tarafından işlenmektedir [15]. Nihai gösterim, kalıpları tek kullanıcının kalıbıyla karşılaştıran özellik saptayıcıya yüklenmektedir. Gerçekleme görevinde şifre gerekmez ve sistem veri biyometrinin gösterimini sistem veritabanına kayıtlı bütün diğer kullanıcıları şablonlarıyla karşılaştırmaktadır. Büyük veri tabanlarında kimliklendirme pahalı olduğundan girilen veriyle karşılaştırılacak şablon sayısını sınırlandırmak için sınıflandırma ve dizinleme teknikleri sık sık kullanılmaktadır.

Sonuç ya kayıtlı bir kullanıcı olduğu ya da “Kullanıcı tanımlanamadı” şeklinde bir uyarı mesajı olmaktadır.

Uygulama alanına bağlı olarak biyometrik sistem çevrimiçi ve çevrimdışı sistem iki grupta işleyebilmektedir. Çevrimiçi sistem, tanımlamanın hızlı bir şeklide yapılmasını ve acil karşılığın zorunluluğunu gerekli kılmaktadır. Öte taraftan çevrimdışı sistem, tanımlanma talebinin acil bir şekilde cevaplanmasını gerektirmez ve göreve uzun karşılık için beklemeye izin verilir. Tipik olarak çevrimiçi sistemler tam otomatiktir ve biyometrik özelliğin algılanması için canlı tarayan tarayıcıları kaydı yapılmamış içlemlerde kullanılmasını gerektirmektedir. Çevrimdışı sistem, "Tipik olarak yarı otomatiktir." Her ne kadar biyometrik elde edim çevrimdışı tarayıcılarla olsa bile kaydın denetlenmesi gerekmektedir. İyi bir kalite elde etmek içim manuel kalite kontrol yapılmalı ve nihai karara (şahıs) ulaşmak için karşılaştırıcı, yasal uzman tarafından daha sonra manuel olarak denetlenen adayların listesine dönüş yapılmalıdır.

2.2.1. Biyometrik Sistemlerin Temel Çalışma Prensipleri

Bir biyometrik sistem en genel anlamda bireyin biyometrik özelliğini alan, bu özellikten kişinin kimliklendirilmesinde kullanılacak olan özellik setini çıkaran ve kişiyi temsil eden bu anlamlı veri seti ile daha önceden aynı prensiplerle elde edilip veritabanına kaydedilmiş veri seti/setleri arasında karşılaştırma yapabilen bir tanıma/onaylama veya sınıflandırma sistemi olarak tarif edilebilmektedir [20].

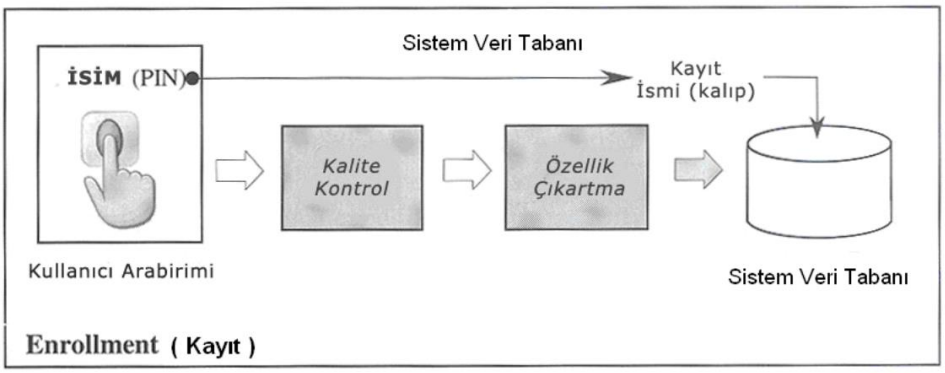
Biyometrik sistemler çalışma şekli ve uygulama durumuna göre 4 gruba ayrılırlar. Bunlar;

- kayıt modu (enrollment),
- onaylama modu (verification),
- tanıma modu (identification)
- izleme modu (screening)

şeklinde sıralanmaktadır[21]. Temelde bu modların tamamının çalışma prensibi aynı olmakla birlikte sözü edilen bu modlar uygulama şekli ve kullanım alanı konusunda bazı farklılıklara sahiptirler.

2.2.1.1. Tanıtma ve Kayıt Oluşturma

Kalıp oluşturma, tanıtma, doğrulama ve teşhis işlemleri için gereklidir. Bu modül bireyi sisteme tanıtmak ve fizyolojik karakteristiğinin veri tabanına kaydedilmesi için kullanılmaktadır. Ham sayısal verinin elde edilmesi için bireyin fizyolojik karakteristiği, biyometrik bir algılayıcı (sensör) tarafından okunur, ardından diğer safhalarda güvenilir bir kayıt teşkil etmesi açısından bir kalite kontrol işlemi devreye girer. Uyumun gerçekleşebilmesi için (Matching) genellikle bir özellik çıkartma modülü devreye girer ve işlem sonucunda “kalıp” (Template) olarak kayıt verisi oluşur.

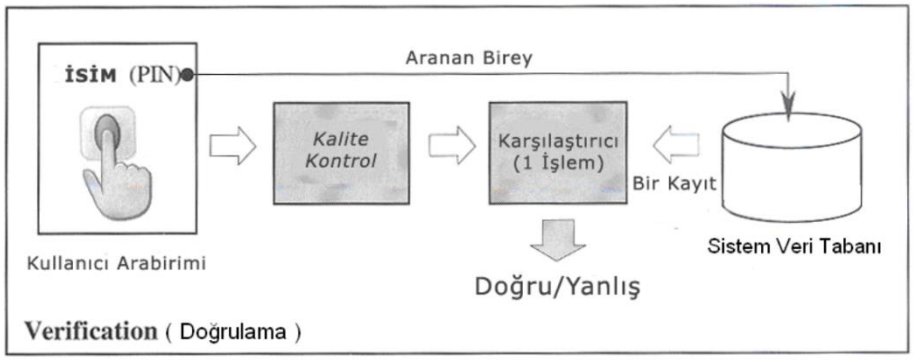


Şekil 2.4. Kayıt yapma (Enrollment) işlemi

Uygulamaya bağlı olarak, kalıplar biyometrik sistemin merkezi veri tabanına veya Manyetik Kart ya da Smartcard (Akıllı kart)'lara yüklenirler. Bu işlem Şekil 2.4.' de görülmektedir [22].

2.2.1.2. Doğrulama

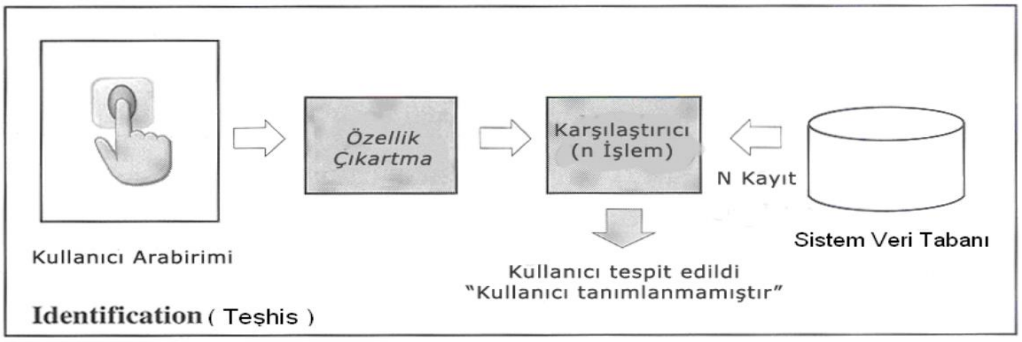
Kişinin erişim noktasında, doğruluğunun saptanması için doğrulama işlemi kullanır. İşlem esnasında Kullanıcı Adı veya PIN kodu bir klavye veya tuş takımından girilir. Biyometrik bir okuyucu ile bireyin karakteristiği (parmak izi) alınır ve sayısallaştırılarak özellik çıkartma işleminden geçirilip algılamanın sayısal gösterimi gerçekleştirilir. Son olarak özellik karşılaştırıcı ile PIN kodu girilmiş olan bireyin tek bir işlemle mevcut kayıttaki kalıpla uyumluluğu saptanır. Bu işlem Şekil 2.5.' de görülmektedir [23].



Şekil 2.5. Doğrulama işlemi

2.2.1.3. Teşhis Tanıma

Veri tabanındaki tüm kullanıcılara ait fizyolojik karakteristik bilgilerinin yer aldığı kalıplarla tek tek karşılaştırması ile tamamlanan bir işlemdir. Kullanıcı Adı veya PIN kodu bilgisine gerek yoktur. Bu işlem Şekil 6' da görülmektedir [24].



Şekil 2.6. Kullanıcı teşhis (Idenfication) işlemi

İşlem giriş bilgisinin tüm veri tabanı ile karşılaştırılarak kimlik tespitini yapmaya çalıştığından zaman alan bir hesaplama süreci söz konusudur. Bu sebeple büyük veri

tabanlarına sınıflandırma ve indeksleme gibi tekniklere başvurularak işlem süresi kısaltılmaya çalışılır.

Uygulama orijinli bir biyometrik sistem çevrimiçi (online) ya da çevrimdışı (offline) olarak çalışabilir. Çevrimiçi çalışan bir sistem algılamanın hızlı gerçekleşmesini ve sonucun hemen üretilmesini gerektirir. Örneğin, bir bilgisayar ağına giriş işlemi. Çevrimdışı çalışan bir sistemde algılamanın görece hızlı gerçekleşmesine gerek duyulmadığı gibi üretilen sonucun bekleme toleransı da yüksektir. Çevrimiçi sistemler tam otomatik olarak çalışır ve bu tarz çalışabilecek bir tarayıcıya ihtiyaç duyarlar. El yordamıyla bir kalite kontrol mekanizması mevcut olmayıp tüm işlem safhaları; Uyum ve Karar Verme mekanizması tam otomatik olarak gerçekleşir.

2.2.1.4. Pozitif ve Negatif Tanımlama

Pozitif tanımlama durumunda sistem şahsın iddia ettiği kişi olup olmadığı belirler. Pozitif tanımlamanın amacı aynı kimliği birden fazla kişinin kullanmasını engellemeyi amaçlamaktadır. Örneğin eğer sadece Hakan denilen bir kişi belirli bir güvenli alana girme noktasında yetkili ise, sistem giriş yetkisini sadece Hakan'a verecektir. Eğer sistem Hakan'ın kayıtlı şablonuyla girilen verinin karşılaştırılmasında başarısız olursa sonuç ret olmaktadır [25].

Negatif tanımlama uygulamasında sistem, şahsın olmayı reddettiği kişi olup olmadığını belirlemektedir. Negatif tanımlamanın amacı, bir tek şahsın birden çok kimliği kullanmasını önlemektir. Örneğin; Hakan her zaman devletten yoksulluk yardım parası alıyordu ve şimdi de Mehmet olduğunu iddia ederek Mehmet'in da yardım paralarını almak istemektedir. Bu durumda sistem Mehmet' in, onun olduğunu

iddia ettiği kişi olmadığını belirlemektedir. Eğer sistem Mehmet' in biyometrik verilerini siteme kayıtlı yardım alan şahıslarla karşılaştırırken başarısız olursa kabul gerçekleşir, aksi takdirde sonuç ret olmaktadır.

Bunu unutmayalım ki giriş kodu, şifre, anahtarlar, jetonlar gibi geleneksel kullanıcı doğrulamalarında pozitif tanımlama yöntemleri ile çalışıldığı halde, negatif tanımlama yöntemi sadece biyometriklerle çalışabilmektedir. Dahası pozitif yanılmama uygulamaları hem doğrulama hem de kimliklendirme durumlarında içe yararken, negatif tanımlama ve doğrulama durumlarında çalışmamaktadır. İçin aslı sistem söz konusu verinin o an mevcut olmadığını kanıtlamak için bütün arşivi araştırmak zorundadır.

2.2.2. Biyometrik Sistem Uygulama Çeşitleri

Biyometrik uygulamalar karakteristik özelliklerine göre aşağıda anlatılan 7 kategoriye ayrılmaktadır [26].

Ortak olan ve ortak olmayan (cooperative - non-cooperative) uygulamalar

Sistem kullanıcılarının sistem ile etkileşiminin tam olarak tanımlandığı uygulamalardır. Elektronik bankacılık ortak bir uygulamaya örnek; Mehmet' in havaalanlarında teröristleri tanımlamayı amaçlayan bir uygulama ortak olmayan uygulamaya bir örnektir.

Açık ve gizli (overt - covert) uygulamalar

Pek çok ticari uygulamada olduğu gibi kullanıcı biyometrik özelliğinin alındığının farkındaysa bu açık uygulama, farkında değilse gizli uygulamadır.

Alıştırılmış ve alıştırılmamış (habituated - non-habituated) uygulamalar

Giriş çıkış kontrolünün biyometrik özelliklerle sağlandığı kapıdan giriş çıkış yapmak gibi sisteme kayıtlı olan kullanıcılardan ne sıklıkla ve ne zaman biyometrik karakter bilgisi istendiğinin belirli olduğu uygulamalar alıştırılmış uygulamalara örnektir. Ehliyet alınırken biyometrik özellik kullanımı gibi durumlar ise alıştırılmamış uygulamaya örnek olarak gösterilebilir.

Katılımlı ve katılımsız (attended - non-attended) uygulamalar

Sisteme bir bilir kişinin dahil olup olmaması ile ilgilidir. Kullanıcıya ATM kartının ilk verilmesi katılımlı; ATM kartının daha sonraki kullanımları katılımsız uygulamaya örnektir.

Standart ve standart olmayan çevresel uygulamalar (Standard – non-Standart operating environment)

Bilgisayara giriş yapmak gibi çevresel şartların belli olduğu uygulamalar standart uygulamaya örnek; gece dışarıda yüz takibi yapmak standart olmayan uygulamaya örnektir.

Genel ve özel (public - private) uygulamalar

Şirket içinde bir ağa dahil olup olmama bilgi işlem yöneticisinin kontrolündedir ve bu bir özel uygulamadır. Ancak biyometrik özelliğin yüklü olduğu elektronik kartların kullanılması genel bir uygulamadır.

Açık ve kapalı (open - closed) uygulamalar

Bir kullanıcı hem fiziksel olarak güvenli girişin gerektiği bölgelerde, hem elektronik bankacılıkta hem de bilgisayara girişte parmak izi kullanabilir. Tüm bu sistemlerde

kişinin parmak izi farklı veritabanlarında saklanıyor ve buradan karşılaştırılıyorsa bu kapalı bir sistemdir. Ancak tüm bu uygulamalar için ortak bir veritabanı kullanılıyorsa bu açık uygulamaya bir örnek olur.

3. PARMAK İZİ VE PARMAK İZİ TANIMA SİSTEMLERİ

Bu bölümde parmak izi tarihçesi, parmak izinin oluşumu, parmak izi algılanması ve saklanması, parmak izi görüntüsünü, parmak izlerinin sınıflandırılması, özellik noktası çıkarımı hakkında detaylı bilgi verilecektir.

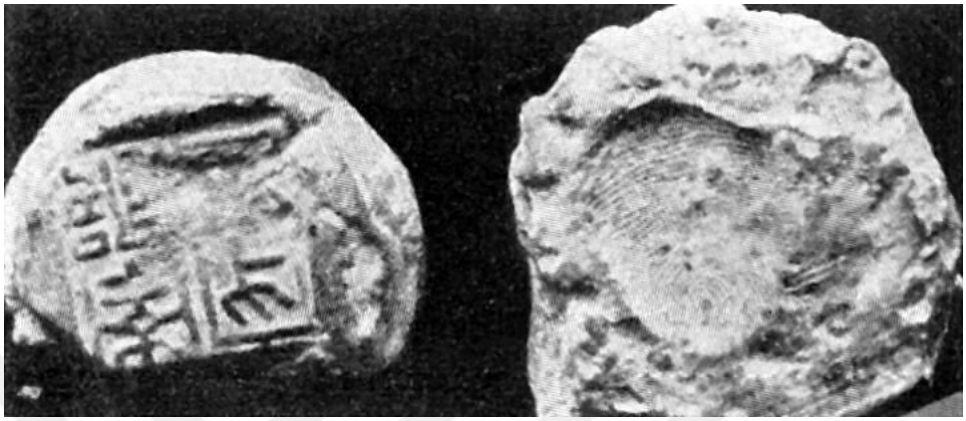
3.1. Parmak İzi Tarihi

En eski parmak izleri Babiller' e uzanır. Babiller' den kalan bazı kil tablet, mühür, tuğla ve seramik kaplarda parmak izleri bulunmuştur. Bazı izler kazara çamurun üzerinde kalmış olsa da bazıları oldukça derin olduğu tespit edilmiştir. Derin parmak izlerinin, seramiği yapan ustalarca, kimliklerinin bilinmesi için bırakıldığına inanılıyor.

M.Ö. 1300'lerde ticari kontratların yazıldığı kil tabletlere iki tarafın da imza amacıyla parmak bastığı belirlenmiştir. Çin'de resmi dokümanlara parmak basmak oldukça eski bir gelenek olduğu ortaya çıkmıştır. M.Ö. 246'da Çin'de, resmi görevliler kil mühürler üzerine parmak bastıkları görülmüştür. Daha sonra, ipek ve kağıda yazılan kontratlara her iki taraf da mürekkepli parmaklarını basmaya başladı. Bu yöntem bazı ülkelerde okuma yazma bilmeyenler için imza yerine kullanılır.

M.S. 851' de Zeyd Hasan, Çin'de tüccarların senetlere parmak bastığını yazmıştı. Japonya'da 702 yılına ait boşanma belgesinde, okuma yazma bilmeyen çiftin parmak

izleri basıldığı bilinmektedir. İranlı R. Hamadani, 1300'lerde Çin'de parmak izinden teşhis yapıldığını ve her insanın farklı parmak izi olduğunu belirtmiştir.



Şekil 3.1. Babiller' in ticari kil tabletindeki parmak izi

Avrupa'da 1600'lerden itibaren parmak izleri ile ilgili bilimsel makaleler yayınlandı. Sir W. J. Herschel 1858'de Hindistan'da görevli iken, ticari kontratlara tarafların mürekkepli parmak ve el basmalarını mecburi hale getirdi. Herschel, mahkûmların parmak ve el izlerini de dosyalandığı bilinmektedir. Fransız kimyacı P. J. Couler, 1863'te kağıtta kalan parmak izlerini iyot buharı ile görünür hale getirdiğini açıklamıştır. İskoç bilim adamı H. Foulds, 1880'de matbaa mürekkebiyle alınan parmak izleri ile kimlik tespit edilebileceğini göstermiştir. Ancak polis yetkilileri bu sistemi kullanmayı reddeder. Konu tanınmış biyolog Charles Darwin'e iletilir ancak o hasta olduğu için yeğeni F. Galton'a konuyu inceletir. Galton, iki insanın aynı parmak izine sahip olma olasılığının 64 milyarda 13 olduğunu hesaplar ve parmak izlerini gruplara ayırmıştır.

Her insanın parmak izinin farklı olduğu 1880'lerde anlaşılmaya başlamıştır. Parmak izi yardımıyla bir cinayetin çözülmesi ilk kez 1892'de Arjantin'de gerçekleşmiştir. Arjantin'de emniyet amiri olan Juan Vucetich, 1892'de dünyanın ilk parmak izi bürosunu kurmuştur. Aynı yıl, iki oğlunun ölümünden dolayı komşusunu suçlayan bir kadının asıl katil olduğunu annenin kapıdaki kanlı parmak izi yardımıyla bulmuştur. Anne, genç biriyle evlenmesine karşı çıkan çocuklarını öldürdüğünü itiraf etmek zorunda kaldı. Bu olay parmak izinin kimlik tespitinde kullanılmaya başlanmasını hızlandırdı. Ardından Hindistan'da da bir parmak izi bürosu kuruldu. Hintli uzmanların hazırladığı parmak izi gruplaması, Henry Sınıflandırması olarak bilinir. İngiltere'de Scotland Yard, 1901'de parmak izi bürosunu kurdu ve Henry Sınıflandırmasını kabul etti. New York polis yetkilileri de 5 yıl sonra ilk büroyu kurdu.

Türkiye'de parmak izi alma işlemi 1910'da Macar kökenli Yusuf Cemil tarafından başlatılmıştır.

ABD'de 1924'te FBI Başkanı olan J. E. Hoover, 48 yıl görev yaptı ve dünyanın en büyük parmak izi arşivini oluşturdu [27].

3.2. Parmak İzinin Oluşumu, Algılanması, Taranması, Görüntüsü ve Saklanması

Bu bölümde parmak izinin oluşumu, parmak izinin algılanması, parmak izinin taranması, taranan parmak izinin görüntüsü ve parmak izinin saklanması hakkında kısaca bilgiler verilecektir.

3.2.1. Parmak İzi Oluşumu

Parmak izi fetüs yedi aylık bir gelişimi tamamladığında tamamen oluşur, ömür boyu değişmeden kalır. Bu özellik parmak izini biyometrik tanımlayıcı olarak ön plana çıkarmaktadır.

Parmak izinin oluşumunda çok sayıda olasılık olduğundan dolayı görsel olarak parmak izlerinin biri birinin aynısı olması imkânsızdır. Parmak izleri oluşumu atadan gelen genlerle alakalı olduğundan oluşan parmak izi benzer özellikler taşır [13].

Tek yumurta ikizlerinin DNA testiyle ayrımı yapılamamaktadır. Ses, vücut şekli, yüz ve el geometrisi gibi birçok fiziksel karakteristik tanımlamalı otomatik tanıma sistemleri, tek yumurta ikizlerinin ayrımında hatalı sonuçlar üretebilmektedir. Bununla birlikte parmak izine ait detaylar (minutiae) tek yumurta ikizlerine farklılık göstermektedir [13].

Dermatolojik çalışmalar parmak izi yapısının farklı ırklara mensup insanlarda çok daha belirgin olduğunu göstermektedir. Aynı ırktan akrabalık ilişkisi olmayan insanlarda bazı benzerlikler göze çarpmaktadır. Ebeveyn ve çocuk ilişkisinde benzerlik artmakta, en büyük benzerlik ise en yakın genetik ilişkinin olduğu mono zigot (tek yumurta) ikizlerinde olduğu belirtilmektedir [13].

3.2.2. Parmak İzi Algılama

Parmak izi algılamada, tarama modu baz alındığında; çevrimdışı ve çevrimiçi tarama şeklinde bir sınıflandırmadan söz edilir. Çevrimdışı tarama, kağıt üzerindeki bir mürekkeple elde edilen parmak izinin taranması olarak örneklendirilebilir. Bu tip bir

tarama ya optik tarayıcı ya da yüksek çözünürlüklü bir kamera tarafından yapılabilir. Bu uygulama güvenlik birimleri ve adli makamlarca sıklıkla kullanılır. Parmak izi yağlı yapısından dolayı dokunduğu yerde bir iz bırakır. Kimyasal yöntemler kullanarak izler görünür hale getirilir ve optik bir tarayıcı veya kamera kullanmak suretiyle görüntüler sayısallaştırılır.

Parmak izinin karakterize edilmesindeki ana parametreler; çözünürlük, alan, piksel sayısı, geometrik doğruluk, kontrast ve geometrik gürültüdür.

Çevrimiçi tarama için optik, kapasitif, termal, basınç duyarlı, ultrasonik vb. kullanılan çeşitli algılama sistemleri vardır. Optik tarayıcılar yıllardır kullanılmakta olan teknolojidir. Yeni solid-state (katı-hal) tarayıcılar küçük yapılarıyla birçok elektronik cihaza monte edilebilir boyutlara indirgenmiştir.

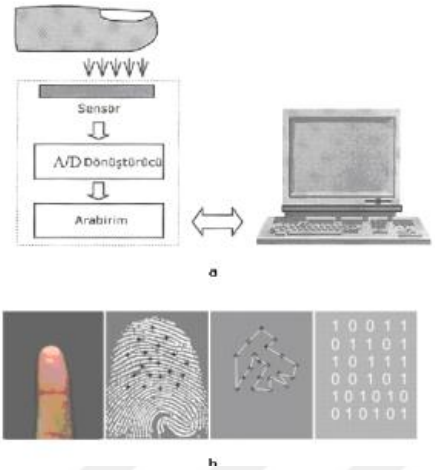
Parmak izi algılama sistemleri hızlı gelişme göstermekte olan sistemlerdir. Çok çeşitli alanlarda kullanım imkanı bulmuşlardır. Suçlu tespitinden elektronik veri güvenliğine kadar değişik sektörlerde kullanılan bazı önemli uygulamaların isimleri Tablo 3.1.’ de görülmektedir.

Tablo 3.1. Parmak izi algılama sistemlerinin uygulama alanları

Adli Kurumlar	**Vatandaşlık Uygulamaları**	**Ticari**
Suçlu Tespiti	Kimlik Kartı	Bilgisayar Ağ Güvenliği
Kadavra Tespiti	Sürücü Belgesi	Elektronik Veri Güvenliği
Terörist Tanımlama	Sosyal Güvenlik	E-Ticaret
Akrabalık Tespiti	Pasaport Kontrol	İnternet Erişimi
Kayıp Çocuk Tespiti	ATM, Kredi Kartı	
Cep Telefonları		
PDA		
Medikal Kayıt Yönetimi		

3.2.3. Parmak İzi Tarama

Parmak izi taramasında özellikle emniyet teşkilatlarında kullanılan yöntem "mürekkep tekniği" olarak adlandırılabilecek, parmağın mürekkeple beyaz kağıda izinin çıkarılması prensibine dayanmaktadır. Sayısal görüntünün elde edilmesi kağıdın taranmasıyla gerçekleşir ki bu sistem çevrimdışı sistem olarak bilinir. Mürekkebe ihtiyaç duymayan çevrimiçi sistemler ise direkt olarak parmak izinin taranıp sayısallaştırılması yöntemidir. Düşük fiyat ve küçük boyutlu tasarımlar her zaman endüstride temel amaçtır. Ancak doğru, güvenilir algılama hedefinden uzaklaşılmamalıdır.



Şekil 3.2. Parmak izi tarayıcı sistemi

Şekil 3.2’ de görüldüğü gibi tarayıcının genel yapısında; algılayıcı (sensör), Analog Sayısal dönüştürücü ve iletişim için bir arabirim bulunmaktadır. Algılayıcılar; optik, katı- hal, ultrasonic vs. modelden biri olabilirler. Bazı gömülü (embedded) sistemler tamamen bağımsız olarak tüm işlemleri icra edebilmektedirler [15]. Birçok kişisel algılama sistemi parmak izi görüntüsünü saklamayıp, kenar çıkartma sonucu elde edilen sayısal değerleri saklar. Bu durum parmak izleri verilerinin sayısal ortamda daha az yer kaplaması anlamına gelir. Ancak bazı uygulamalarda parmak izi görüntüsünün bütününün saklanması gerekebilmektedir.

3.2.4. Parmak İzi Görüntüsü

Sayısal parmak izi görüntüsünün karakterize edilmesi için baz alınan temel parametreler;

Çözünürlük (Resolution): İnç başına düşen piksel veya nokta sayısıdır. 500 DPI çözünürlük tercih edilen değerdir. 250-500 DPI arası değerlerde tarama işlemi sonucu doğru, güvenilir uyum/uyumsuzluklar gerçekleşir. Detay, parmak izi uyumunda öncelikli rollerden birini icra eder, parmak izinin aynı kişiye ait olup olmadığını gösterir. Düşük çözünürlükte elde edilen görüntülerde parmak izine ait tepe ve çukurların ayrımı zorlaşmaktadır. 200-300 DPI çözünürlükteki görüntülerde parmak izi uyumu genellikle Korelasyon tekniği kullanılarak gerçekleştirilir [15].

Alan (Area): Alan, parmak izinin tarandığı dikdörtgensel alanın boyutudur. Ne kadar büyük tarama alanı olursa o kadar ayırt edici, kaliteli veri elde edilir. Küçük parmak izi tarayıcıları parmağın tamamına ait görüntü almazlar, bu durumun sonucu olarak kullanıcılar tarama bölgesi konusunda hassas olmak durumundadırlar. Minimum 2,54 x 2,54 cm'lik bir tarama alanında elde edilen görüntü tam anlamıyla parmak izine ait verilerin elde edilmesine imkan verir. Bununla beraber algılayıcı tarama alanı ne kadar küçük olursa o kadar düşük maliyet söz konusu olacaktır.

Piksel Sayısı: Piksel sayısı çözünürlük ve tarama alanlarından elde edilir.

r : çözünürlük

h : Tarama alanı yüksekliği

w : Tarama alanı genişliği

r= 500 DPI

h= 30,22 mm

w=25,18 mm

değerlerine göre piksel sayısı aşağıdaki gibi hesaplanabilir.

Piksel Sayısı = 500 x (30,22/25,4) x 500 x (25,18/25,4) = 594,88 x 495,66

≈ 600 x 500

Dinamik Aralık (derinlik): Her pikselin işlenmesi için gerekli olan bit sayısını gösterir. Parmak izi algılamada renkli tarama kullanılmamaktadır. Hemen tüm sistemler 8 bitlik piksel derinliği kullanarak 256 gri renk seviyesi tarama gerçekleştirirler. Bazı sistemler 2-3 bitlik tarama gerçekleştirip yazılım desteğiyle 8 bite ulaşabilmektedir [5]. 1 bitlik derinliğin üzerindeki tüm taramalarda kenar çıkarma işlemi için yeterli veri elde edilebilmektedir. Örneğin 8 bitlik bir taramada 256 gri renk seviyesi elde edilir.

Geometrik Doğruluk: x ve y yönlerindeki maksimum geometrik bozunumdur.

Görüntü Kalitesi: Tam olarak parmak izinin kalite düzeyini tarif etmek zordur. Aynı zamanda her parmak izinden kaliteli görüntü elde etmek de pek mümkün değildir. Eğer parmak izine ait tepeler (ridge) yüksek değilse, izler belirgin değilse (yaşlı insanların ve el kullanarak bedensel iş yapanların vs.) ve parmak çok nemli ve ıslak ise veya parmak tarayıcıya yanlış odaklanarak dokundurulmuşsa düşük kaliteli görüntüler elde edilir. Şekil 3.3.’ de çeşitli ortamlarda elde edilmiş parmak izi görüntüleri yer almaktadır.

Şekil 3.3. Optik algılayıcılarla elde edilmiş çeşitli parmak izi görüntüleri

Şekil 3.3.' de sırasıyla soldan sağa; İyi kalitede görüntü, çok kuru bir parmak ile elde edilen görüntü, ıslak parmakla elde edilen görüntü ve kötü bir parmak izi görüntüsü verilmiştir [16].

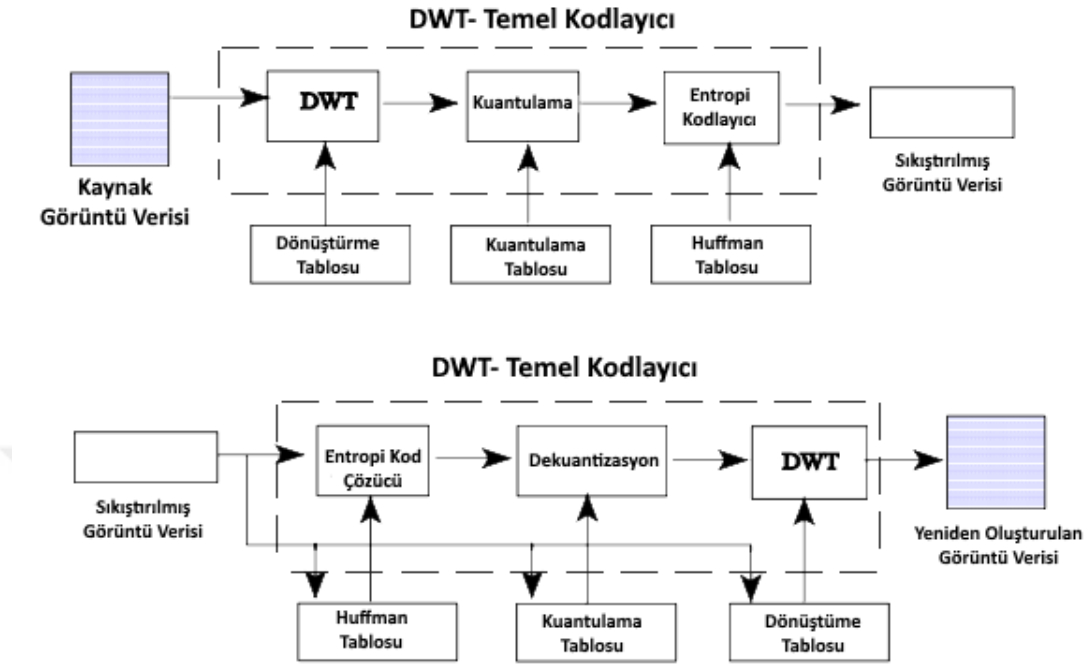
Tüm bu özellikler algılama sisteminin doğruluğuyla direkt olarak ve bir bütün olarak alakalıdır. Örneğin; 500 DPI tarama derinliğinden 400 DPI derinliğe düşüşte doğrulukta %1 lik azalma olacağı düşünülürse ve dinamik derinliğin de 8 bitten 4 bite indirildiğinde yine %1 lik kayıp olacağı farz edildiğinde; her iki durumun birden uygulanması sonucunda kayıp %2 den daha fazla olacaktır. Sonuç olarak tüm bu özellikler birbiriyle ilişkilidir.

3.2.5. Parmak İzinin Saklanması ve Sıkıştırılması

1995 yılı kayıtları itibariyle FBI biriminde 200 milyon civarında kayıt olduğu bilinmektedir. Bu bilgiler sayısallaştırıldığında verilerin saklanması önemli bir problem olmaktadır. 500 DPI lık bir görüntü sıkıştırmasız sayısallaştırıldığında 589,824 bytes

yer kaplamakla birlikte (encode) işlemi için 10 MByte'lık bir alana ihtiyaç duymaktadır. Sayısallaştırılmış parmak izi verilerini saklamak için düşük kayıplı JPEG resim sıkıştırma formatı tatminkar bir sonuç verememektedir. Bunun için sıkıştırılmış gri tonlama tekniği olan Wavelet Scalar Quantization (WSQ) kullanılmaktadır [14].

WSQ kodlayıcı sınıfı, parmak izi görüntüsünün, her biri belirli bir frekans bandındaki bilgiyi temsil eden bir dizi alt bantta ayrışmasını içerir. Alt bant ayrışımı, parmak izi görüntüsünün ayrı bir dalgacık dönüşümü ile sağlanır. Alt bantların her biri daha sonra bir niceleme tablosundaki değerler kullanılarak nicelenir. Bu şartnamede, niceleme tabloları için varsayılan değerler verilmemiştir. Ölçülen katsayılar daha sonra verileri sıkıştıran bir Huffman kodlama prosedürüne geçirilir. Huffman tablo özellikleri encode sağlanmalıdır. Şekil 3.4. WSQ kodlama ve kod çözme için ana prosedürleri gösterir. Belirli bir görüntüyü sıkıştırmak için kullanılacak bir kodlayıcı için belirtilen aynı tabloların, o görüntüyü yeniden oluşturmak için bir kod çözücüye sunulması gerekir.



Şekil 3.4. WSQ kodlaması

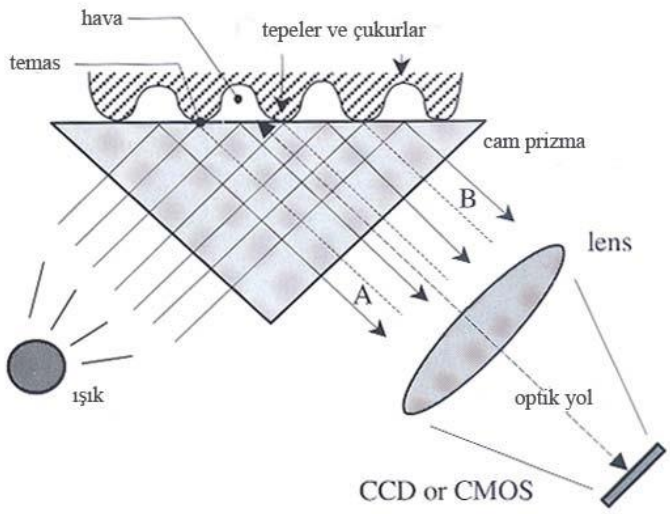
3.3. Parmak İzi Okuyucuları

Parmak izi algılama sistemlerinin en önemli parçası parmak izi algılayıcılarıdır. Parmak izi algılama sistemlerinde kullanılan hemen tüm algılayıcılar; Optik, Katı-hal ve Ultrasonik olmak üzere 3 gruba ayrılır;

3.3.1. Optik Algılayıcılar

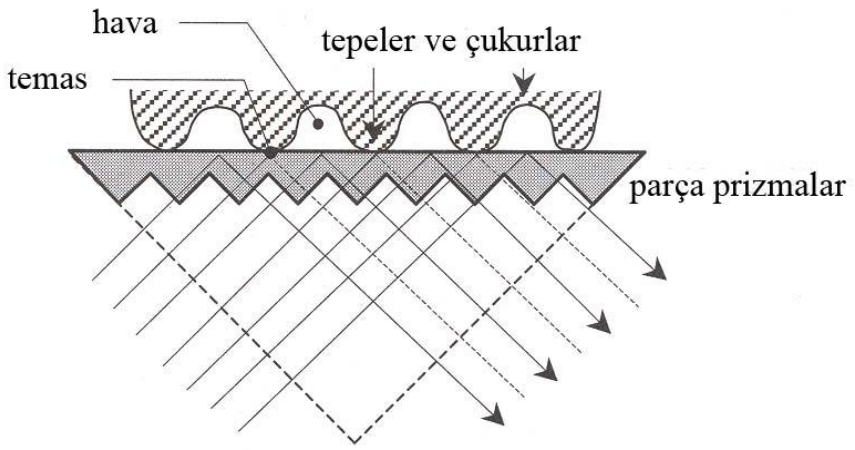
FTIR (Frustrated Total Internal Reflection) en eski ve çok kullanılmakta olan canlı-tarama (live-scan) ile algılama tekniğidir [16]. Parmak cam prizmanın üzerine

dokunduğu anda tepeler (ridges) cam yüzeye temas eder, ancak çukurlar (valleys) belirli bir mesafede temas etmeden kalır. LED'lerden oluşan ışık kaynağından saçılan ışınlar "tepeler" tarafından soğrulur, "çukurlara" çarpanlar yansıma yaparlar. Işınlar son olarak görüntünün elde edilmesini sağlayacak olan CCD veya CMOS görüntü algılayıcılarına odaklandırılır. Görüntüde karanlık yerler parmak izindeki tepeleri aydınlık yerler ise çukurları temsil eder. Bu durum Şekil 3.5 de grafiksel olarak görülmektedir. Genellikle en iyi görüntü kalitesi üretmeleri ve en geniş tarama alanı sunmasına karşın FTIR tabanlı cihazlar diğer optik cihazlar gibi küçültülememektedir. Teknolojik ilerlemeler paralelinde maliyetin azaltılması amacıyla cam yerine plastik parçalar kullanılır olmuştur. Pahalı CCD'ler yerine daha ucuz olan CMOS görüntü algılayıcılar kullanılmaktadır.



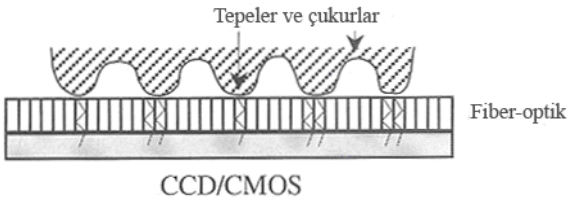
Şekil 3.5. Optik parmak izi algılayıcı

Küçük prizmaların Şekil 3.6. de görüldüğü gibi bitişik olarak kullanılmasıyla elde edilen bir sistemdir. Fiziksel olarak küçük parça prizmaların kullanımı ile sistemin hacmi küçültülmüştür. Görüntü kalitesi, tek prizmanın kullanıldığı FTIR'lere oranla düşüktür.



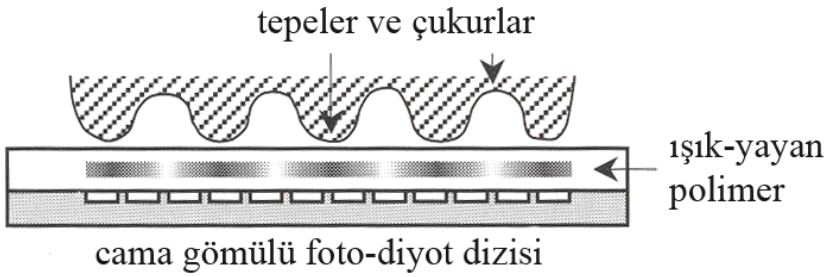
Şekil 3.6. Parça prizmalarla optik parmak izi algılayıcı

Fiber optik yüzey kullanımıyla çok küçük alanlara sığdırılabilir algılayıcılar yapılabilmektedir. Bu sistemlerde lens ve prizma kullanılmadığından algılayıcı boyutu çok büyük oranda azaltılmıştır. Parmak plakanın üst yüzeyiyle temas halindedir. Bu sistemde alt tabakaya yayılmış ara bir lens kullanılmadan direkt CCD/CMOS kameralar yerleştirilmiştir [6]. Arada herhangi bir lens kullanılmadığından dolayı CCD/CMOS'lar tüm yüzeyi kaplama zorunluluğu ortaya çıkmaktadır. Bu durum maliyetin yükselmesine sebep olan bir faktördür.



Şekil 3.7. Fiber yapılı optik okuyucu

Elektro optikte ise iki temel katmandan oluşmaktadırlar. İlk katman polimer bir yapıdır. Voltaj uygulandığında yüzeydeki gücün şiddetine göre ışık iletimi değişkenlik gösterir. Ve bu parmak izi örüntüsünün oluşmasını sağlar. İkinci katman ise, ilk polimer yüzeye sabitlenmiş cama gömülü şekilde duran fotodiyot dizisinden oluşmaktadır. Bunlar görüntünün sayısallaştırılmasını sağlamaktadırlar. Birçok ticari kullanımda ilk katman alınarak, bir lens ve CMOS görüntü algılayıcısıyla uygulanmaktadır. Çok küçük boyutlarda olmasına rağmen bu sistem FTIR yapılı algılayıcıların üretmiş olduğu görüntü kalitesiyle karşılaştırılabilecek düzeyde değildir.



Şekil 3.8. Elektro-Optik parmak izi algılayıcı

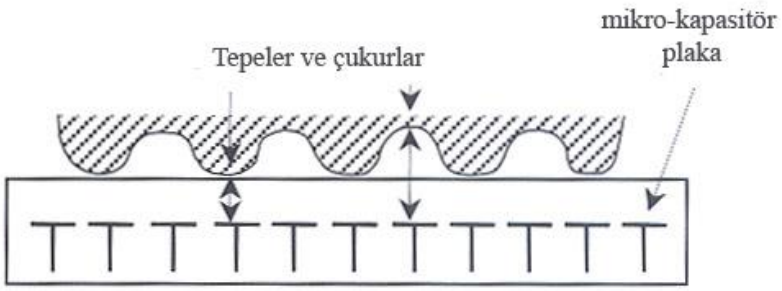
Direkt okumada ise, parmak izine odaklanabilecek yüksek çözünürlüklü bir kamera kullanılır. Parmak herhangi bir yüzeye temas etmemektedir. Ancak parmak izine kameranın odaklanabilmesi için belirli bir mesafe aralığında olması gerekmektedir.

Tüm sistemlerin olumlu ve olumsuz özellikleri mevcuttur. Herhangi bir yüzeye temas söz konusu olmadığından bu sistem hijyenik olup periyodik yüzey temizliği gerektirmez, ancak bu sistemde iyi-odaklama ve yüksek kontrastlı görüntülerin elde edilmesi zordur.

3.3.2. Katı-Hal Algılayıcılar

Silikon algılayıcılar olarak da bilinmekte olup 1990'ların ortasında ticari olarak kullanım imkanı bulmuşlardır. Boyut ve maliyet konusuna çözüm olacağı düşüncesiyle geliştirilmiş olan bu algılayıcılar beklendiği sonucu verememişlerdir. Silikon bazlı algılayıcılar bir dizi pikselden oluşmakta ve her piksel içerisinde küçük algılayıcılar taşımaktadır. Herhangi harici optik bir donanıma veya CCD/CMOS algılayıcılarına ihtiyaç duymamaktadır. Dört temel etki fiziksel bir bilgiyi elektrik sinyallerine çevirmektedir: Kapasitif, termal, piezo elektrik olarak 3 grupta incelenmiştir.

Kapasitif, silikon algılayıcılar içerisinde en çok kullanılanlarındandır. Bir yonga (chip) içerisine gömülü iki boyutlu mikro-kapasitör plakalarından oluşur. Kapasitör plakalarından biri algılayıcının tümsekli yüzeyidir. Bu yüzeydeki tümseklerin ebadı derideki tepe-çukur boyutundan küçüktür. Plakanın diğeri ise parmak derisinin kendisidir. Silikon plakaya parmak teması gerçekleştiği anda küçük elektrik yükleri ortaya çıkar. Bu yüklerin büyüklüğü parmak yüzeyi ile mikro plaka arasındaki mesafeye bağlı olarak değişkenlik gösterir. Bu değişkenlik parmaktaki tepeler ve çukurlardan kaynaklanan değişikliklerdir.



Şekil 3.9. Kapasitif algılayıcı

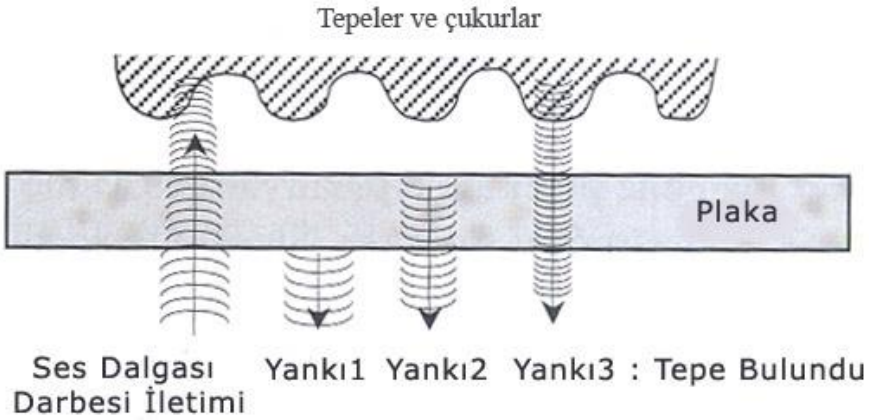
Termal algılayıcı, pyro-elektrik malzemeden yapıldığından dolayı, ısıya karşı duyarlı olup akım oluşturur. Parmak izinin yüzey ile teması sırasında tepeler dokunmuş olacağından oluşan ısı, çukurlarınkinden farklı olacaktır. Kapasitif algılayıcılarla karşılaştırıldığında ESD dayanımı daha yüksek olup koruma kılıfı daha kalın olabilmektedir. Zira ısı kalın yüzeyden de yayılım yapabilmektedir.

Piezo elektrik algılayıcı, basınca duyarlı algılayıcılar yüzeylerine mekanik bir bası uygulandığında bir elektrik sinyali üretir. Algılayıcı, yüzeyi yalıtkan dielektik maddeden yapılmış olup parmağın teması esnasında basınç etkisiyle bir akım meydana gelir. Bu etki piezo elektrik etkisidir. Oluşan akım basıncın gücüyle doğru orantılıdır. Parmak teması esnasında her zaman için tepelerin ve çukurların etkisi farklı olacaktır, bunun sonucunda farklı elektrik alanları üreteceklerdir.

3.3.3. Ultrasonik Algılayıcılar

Parmak yüzeyine akustik sinyalleri gönderip, yüzeyden gelen yankılarını algılama mantığıyla çalışmaktadır [7]. Yankıyla (echo), parmak izine ait tepe ve çukurların

uzaklığı ölçülür. Algılayıcı iki temel bileşeni vardır: Kısa akustik darbeleri oluşturan darbe gönderici (transmitter) ve yankıları yakalayan alıcı (receiver). İyi kalitede sonuç elde edilebildiği gibi diğer sistemlere oranla yavaş çalışmaktadır ve fiziksel olarak büyük bir mekanik yapıya sahiptir. Şekil 3.10. da Ultrasonik algılayıcıların yapısı görülmektedir.



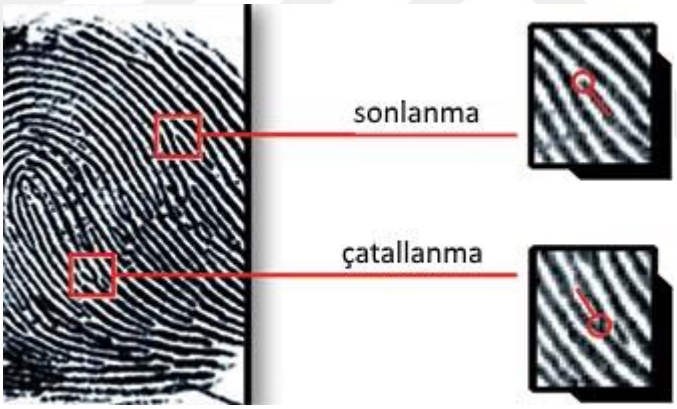
Şekil 3.10. Ultrasonik algılayıcı

3.4. Parmak İzi İçin Terminoloji

Kullanılan parmak izi terminolojisi iki başlıkta açıklanmıştır.

3.4.1. Karık Çizgileri, Bitiş Noktası ve Çatal Noktası

Karık çizgisi parmak izi olan tek bir dairesel çizgidir. Karık çizgilerinin toplamı parmak izi desenini ortaya koymaktadır. Parmak izi tanıma sistemlerinde kullanılır. Kontur hatları çatal noktaları aniden kesilmesi ve çatallanma (bifurkasyon) ile oluşturulmaktadır.

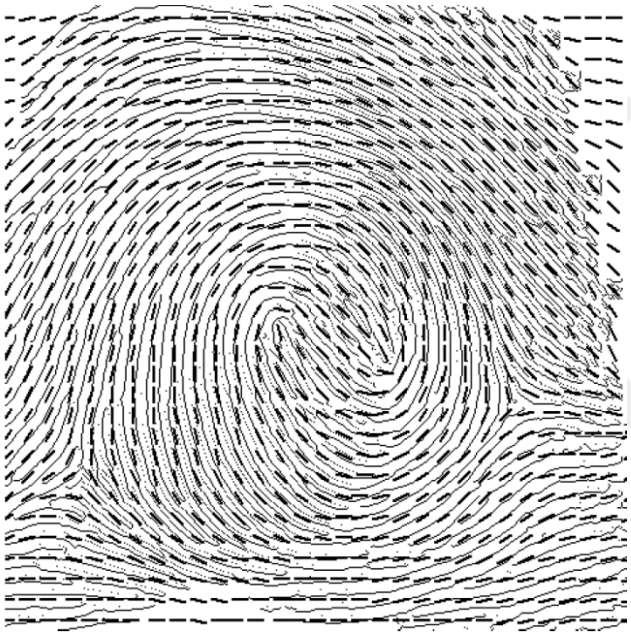


Şekil 3.11. Parmak izi çizgileri

3.4.2. Oryantasyon Haritaları

Oryantasyon haritası, karık çizgileri yön haritasının yönünü gösterir, görüntüleri oluşturan küçük parçalara bölünür. Her parçada yerel yönlendirme işlemi yapılır. Birim vektörünün diğer tarafındaki eğimin yönünü belirler. Pistteki tüm noktaların ortalaması

hesaplanır. Oluşan vektör, bileşenin yönünü belirler. Düşük çözünürlüklü parmak izinin oryantasyon haritası genel bir sunumdur. İzin verilen oryantasyon haritasının parmaklarınızla sınıflandırılması, bir referans noktasının bulunmaması, tanımlama ve parmak izi tanıma sistemleri ve yapay sinir ağı için kullanılabilir. Şekil 3.12.' de oryantasyon haritası gösterilir.



Şekil 3.12. Örnek oryantasyon haritası

3.5. Parmak İzlerinin Sınıflandırılması

Parmak izlerinin ve vadilerin ortasındaki çizgi, birbirinden çok sayıda farklı özel şekiller oluşturur [29]. Bu şekilde, farklı parmak izi sistematik bir sınıflandırma yapmak için yeterince küçüktür. Böylece oluşan şekilleri sınıflandırmak mümkündür. Bazı parmak izi sınıflarını açıklanmıştır.

Şekil 3.13. Parmak izlerinin sınıflandırılması (Soldan sağa kemer, kabarık, döngü)

3.5.1. Kemer

İnsanların yaklaşık %6'sı bu modeli sergiler. Çizgiler, parmağın ortasındaki pürüzsüz veya yukarı doğru hareket eder. Alt sınıflar düz ve çadır kemerdir.

Düz kemer, tüm parmak izi desenlerinin en basiti. Baskının bir tarafından giren ve karşı tarafından çıkan sırtlardan oluşur. Bu çıkıntılar desenin merkezinde yükselme eğilimindedir ve dalga benzeri bir desen oluşturur.

Şekil 3.14. Düz kemer

Çadır kemer ise parmağın yanından başlar ve düz bir kemerinkine benzer bir şekilde akar. Aradaki fark, merkezdeki sırtlarda, sürekli değildir (birbirlerine bitişik, eğimli bir çadır izlenimi vererek yukarı doğru salınırlar).

Şekil 3.15. Çadır kemer

3.5.2. Kıvrım

İnsanların yaklaşık %34'ü bu kalıbı sergiler. Çizgiler eş merkezli daireler, jakuziler veya spiraller oluşturur. Alt sınıflar sade, merkezi cep, çift döngü ve kazayla içerir.

Sade kıvrımlar, çıkıntılar bir tam devrenin dönüşünü ve dolayısıyla dairesel veya spiral şekillidir. İki deltaları var. Bu kıvrım parmak izi en basit ve en yaygın olanıdır.

Şekil 3.16. Sade kıvrım

Merkezi kıvrımlar; parmak izi kalıbı, bir dış döngü içinde bir iç cep (whorl) oluşturan ikinci bir kez nükseder. Minimum iki delta ve yazı tipine sahiptir. Erkek tavus kuşu tüyü üzerindeki göz şablonuna benzerliğinden dolayı bazen tavus kuşu gözü olarak bilinir.

Şekil 3.17. Merkezi kıvrım

Çift döngü kıvrım, birbirlerine sarılmaya çalışan iki ayrı ilmek oluşumu vardır. İki ayrı delta seti vardır.

Şekil 3.18. Çift döngü kıvrım

Kaza kıvrımları, iki farklı desen türünün birleşimi (Düz Kemerler hariç). İki veya daha fazla delta vardır. Kombinasyonları olabilir.

Şekil 3.19. Kaza kıvrımları

3.6. Parmak İzinde Özellik Noktalar

Parmak izi sınıflandırmasına ilişkin ilk bilimsel çalışmalar, parmak izlerini üç ana sınıfa bölen (Galton, 1892) tarafından yapılmıştır. Daha sonra, (Henry, 1900), sınıf sayısını artırarak Galton'un sınıflandırmasını geliştirdi. Şu anda polis teşkilatları tarafından kullanılan tüm sınıflandırma planları, Henry'nin sınıflandırma planının bir çeşididir [28].

Minutiae'nın ana kategorileri aşağıdaki gibidir:

- Sonlanma - aniden biten bir sırt;
- Çatallanma - iki sırta bölünen tek bir sırt;
- Göl veya çevre - tek bir sırt olarak devam etmek için kısa bir süre sonra çatallaşan ve tekrar birleşen bir sırt;
- Kısa sırt, ada veya bağımsız sırt - başlayan, kısa mesafeli yolculuk yapan ve ardından biten bir sırt;
- Nokta - yaklaşık olarak eşit uzunluk ve genişlikte bağımsız bir sırt;
- Mahmuz - daha uzun bir sırttan dallanan kısa bir sırt ile bir çatallanma;
- Geçit veya köprü - iki paralel sırt arasında uzanan kısa bir sırt.

Şekil 3.20. Minutiae noktaları

3.7. Parmak İzinde Görüntü İyileştirme

Görüntü geliştirmede kullanılan yöntemler arasında birçok filtre vardır [22]. Bu filtrelerin örnekleri ortalama değer filtrelemesi Laplacian filtresi ve medyan filtresi uygulanabilir. En sık kullanılan filtrelerden biri, medyan filtrelemede bilgisayarlı görme alanında kullanılmaktadır. Filtreler, uygulanan her pikselin etrafındaki komşu piksellerle birlikte alınır. 3x3 piksel boyutundaki bir filtrenin merkezi ve çevresindeki pikseller bir dizi diziyi sıralayıp atıyor. Sıralanan dizinin orta öğesi (5 öğe) merkez pikselin yeni değeri olarak atanır.

Parmak izi görüntüleri genellikle gri seviye resimlerdir. Bu, görüntü üzerinde 8 bitlik gri hareketsizlik seviyesinden oluşur; özellik noktalarının çıkarılması oldukça zor bir iştir. Görüntü analizi, daha kolay gerçekleştirilebilecek siyah ve beyaz renk değerlerinden oluşan ikili görüntüye dönüştürülmelidir. Bu eşiğin altındaki değerler yerine siyah olan parmak izi, görüntüyü bir nesnenin büyük miktarda beyaz yerine siyah-beyaza dönüştürmek için görüntü renk değerlerinin ortalamasını dikkate alarak bir eşik değeri tanımlar [18].

3.8. Parmak İzi Tanıma Algoritma Çeşitleri

Parmak izi tanıma sistemlerinin en temel üç algoritmasında kullanılan "Minutiae" tabanlı, "Ridge" tabanlı ve "Korelasyon" tabanlı algoritmalarıdır. Yöntemde kullanılan Minutiae tabanlı algoritma detaylı şekilde anlatılacaktır. [24].

3.8.1. Minutiae Tabanlı

En yaygın kullanılan parmak izi sunum tekniğidir ve yapılandırması oldukça belirgindir. Diğer korelasyon tabanlı sistemlere göre daha doğrudur ve minutiae dayalı parmak izi sunumunda şablon boyutu daha küçüktür. Bu sistemde, iki parmak izi minutiae noktaları eşleşirse eşleşir. Minutiae tabanlı parmak izi tekniği, şu anda mevcut olan en fazla parmak izi tanımanın omurgasıdır. Diğer parmak izi özelliklerine kıyasla, karşılık gelen yönlendirme haritalarına sahip minutiae nokta özellikleri, parmak izlerini sağlam şekilde ayırt edebilecek kadar belirgindir. Minutiae özelliğini kullanarak

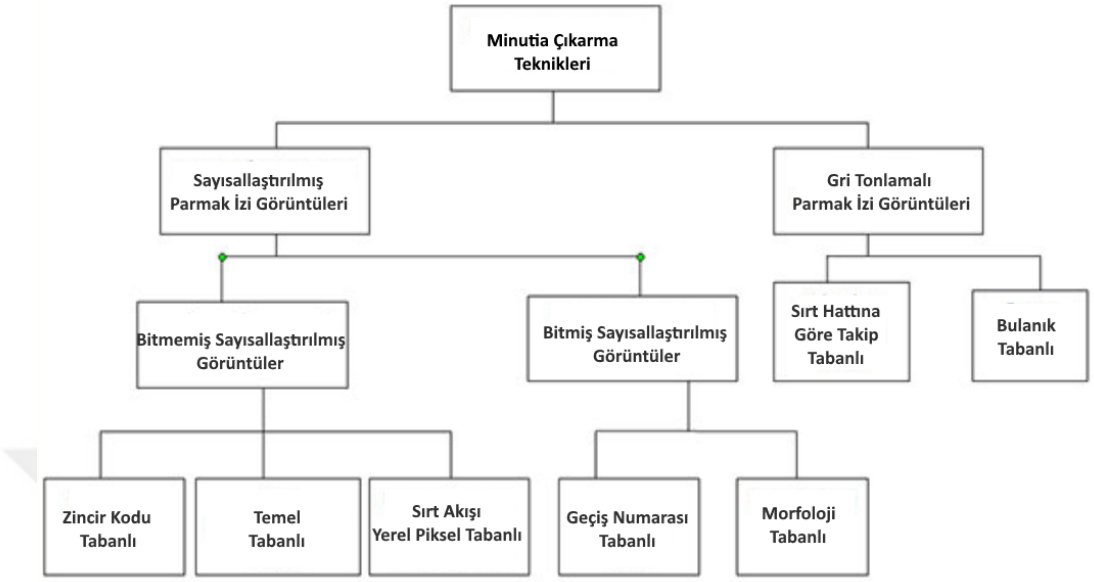
parmak izi gösterimi, parmak izi tanıma karmaşık sorununu nokta desen eşleştirme sorununa indirgiyor.

Orijinal görüntü sadece minutiae bilgileri kullanılarak yeniden oluşturulamadığından, minutiae dayalı parmak izi tanıma sistemleri gizlilik konularına da yardımcı olabilir ve minutiae parmak bireyselliklerini kanıtlamak için yeterlidir. Buna karşılık, görüntü çözünürlüğü ve küresel bozulma açısından, minutiae diğer parmak izi eşleştirme şemalarına göre daha kararlı ve sağlamdır. Bununla birlikte, birincil zorluk, minutiaenin düşük kaliteli bir görüntüden çıkarılmasıdır.

Minutiae çıkarımı için iyi bir görüntü kesinlikle gereklidir. Bununla birlikte, bazen görüntü kalitesi çeşitli nedenlerden dolayı düşük olabilir ve bu yüzden parmak izlerinin minutiae eşleşmesinden önce parmak izi görüntülerinin geliştirilmesi gerekli hale gelir. Minutiae çıkarım yöntemleri iki geniş kategoride sınıflandırılır. Bunlar

- Binarize parmak izi görüntüleri üzerinde çalışan yöntemler
- Doğrudan gri ölçekli parmak izi görüntülerinde çalışan yöntemler

Aşağıda verilen farklı minutiae çıkarımı teknikleri kategorilerini gösteren bir diyagramdır.

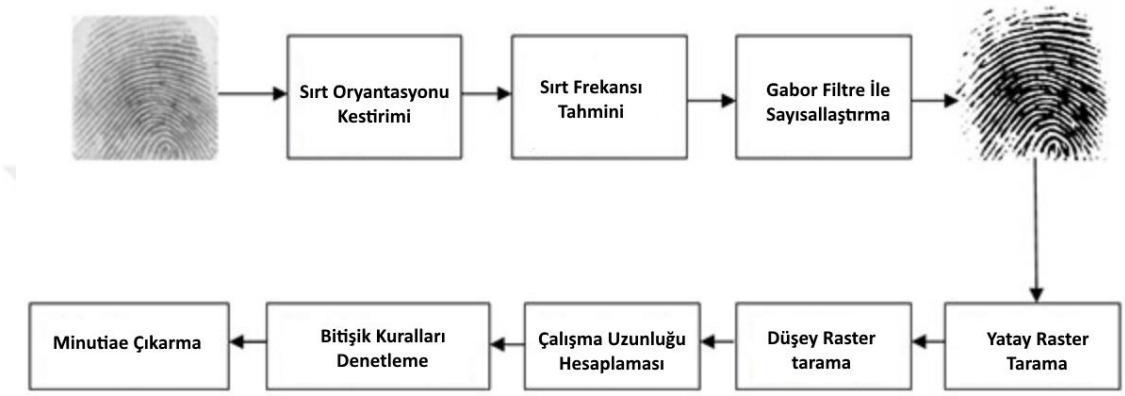


Şekil 3.21. Minutiae çıkarım teknikleri

Zincir kodu tabanlı algoritmasında nesne konturlarının zincir kodu temsiline dayanır ve piksel görüntüsü, tam olarak konturunun zincir kodundan geri kazanılabilir. Bu yöntemde, beyaz arka plandan siyah ön plana geçişler, görüntü yukarıdan aşağıya ve sağdan sola taranarak tanımlanır. Daha sonra konturu saatin tersi yönünde takip ederek bir kontur elemanı dizisi olarak ifade edilir ve her bir eleman konturdaki bir pikseli temsil eder. Sınır boyunca bir sırt çizgisinin saatin tersi yönünde izlenmesiyle, sırt çizgisi kayda değer bir sola dönüş yaptığında bir minutiae ucu bulunur. Benzer şekilde, eğer iz sağa dönerse, çatallanma minutiae tespit edilir.

Temel tabanlı yöntemde ise, ikili görüntülerden kodlanan yatay ve dikey çalışma uzunluğunu temel alır. Bu işlem, hesaplama açısından pahalı bir inceltme işlemine ihtiyaç duymadan hızlı minutiae ekstraksiyonu ile sonuçlanır. Çalışma uzunluğu kodlamasından sonra, parmak izi görüntüleri bir basamaklar dizisi ile tasvir edilir ve

karakteristik çizgiler, yerlerin dizilerin bitişi kontrol edilerek belirlenir. Karakteristik çalışmaların tümü gerçek minutiae değildir ve geçerliliğinin bazı geometrik kısıtlamalar ile kontrol edilmesi gerekir. Bu minutiae çıkarma tekniği aşağıda gösterilen diyagramla açıklanmıştır.



Şekil 3.22. Minutiae çıkarma tekniği

Yukarıdaki şekilde gösterildiği gibi, görüntü geliştirme için önceden işlenir. Görüntü ilk olarak arka plandan bölümlendirerek çıkarılır ve önceden tanımlanmış bir ortalama ve varyansa sahip olmak için normalleştirilir. Her pikselin etrafındaki yerel yönlendirme ve çıkma frekansı hesaplanır ve yerel yönlendirme yönünde yönlendirilmiş sırtları güçlendiren Gabor filtresi uygulanır [28]. Böylece ön plan ile arka plan sırtları arasındaki kontrast artar ve gürültü etkili bir şekilde azalır. Bir sonraki adım, bir eşik değerin seçildiği ve eşiğin üstünde değerlere sahip olan tüm piksellerin beyaz olarak sınıflandırıldığı, diğer tüm piksellerin siyah olarak sınıflandırıldığı görüntü sayısallaştırılmasıdır. Her görüntü alanı için en uygun eşiğin seçildiği uyarlamalı görüntü ikilileştirmesi kullanılarak doğru bir eşik seçilir. Çalışma uzunluğu

gösterimi bellek alanını azalttığı ve işlem süresini de hızlandırdığı için ikili veya etiketli görüntülerde çok verimli olduğu kabul edilir.

3.8.2. Sırtı Tanıma Tabanlı

Sırt akışı tekniğinde, parmak izi görüntüsündeki her pikselin etrafında 3 x 3 kare maskenin oluşturulduğu ve piksellerin ortalamasının hesaplandığı inceltilmemiş binary görüntülerden minutiaeları çıkarmak için kare tabanlı bir yöntemdir. Piksel, ortalama 0.25'ten düşükse bir sırt sonlandırma minutiae ve ortalama 0.75'ten büyükse bir çatallanma minutiae olarak kabul edilir.

3.8.3. Korelasyon Tabanlı

Parmak izi görüntüdeki tepeler ve vadiler çizgisi özellikleri parametreler kullanılarak karşılaştırılır. Kayıt noktası korelasyonu tabanlı teknikler kesin pozisyon verilerini bilmek gerektirir [30].

4. KULLANILAN TEKNOLOJİLER

Bu bölümde uygulamayı gerçekleştirmekte kullanılan teknolojiler, uygulamada kullanılan algoritma ve algoritma örneklerinden bahsedilecektir.

4.1. SourceAfis

SourceAFIS, insan parmak izlerini tanıyan bir algoritmadır. İki parmak izini 1: 1 karşılaştırabilir veya parmak izini eşleştirmek için büyük bir veritabanını 1: N arayabilir. Girişte parmak izi görüntüler ve çıktıda benzerlik puanı üretir. Benzerlik puanı daha sonra özelleştirilebilir eşik eşiğiyle karşılaştırılır [31].

SourceAFIS algoritması, Java ve .NET'te açık kaynak kodlu uygulamaya sahiptir. SourceAFIS API, uygulama geliştiricileri tarafından maksimum basitlik ve kullanım kolaylığı için tasarlanmıştır. Doğruluk ve eşleşme hızı çoğu uygulama için yeterlidir.

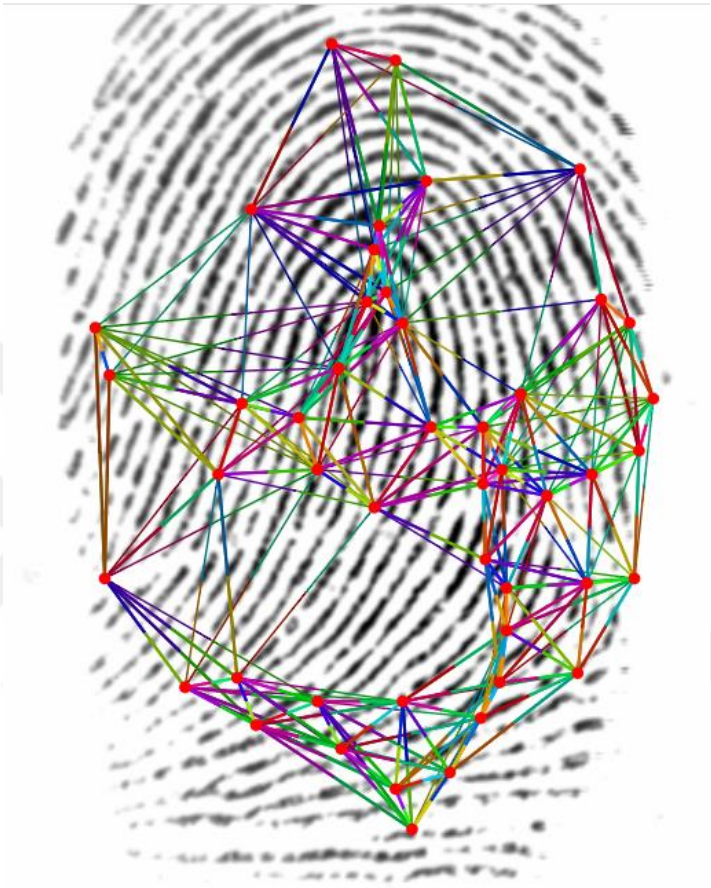
4.1.1. Algoritma

SourceAFIS parmak izinde minutiaeda sırt sonları ve çatallanmaları kullanır. Minutiae, Şekil.4.1'de ilişkili açı açısı olan noktalardır.

Şekil 4.1. Parmak izi görüntüsünde minutiae

Esasen şablonda kaydedilen budur. Parmak izi görüntüsünden minutiae listesine (şablon) kadar birçok küçük soyutlama olur. Minutiae' dan sonra, kenarları oluşturan bir soyutlama daha var. Kenar iki minutiae bağlayan bir çizgidir. Kenarın uzunluğu ve minutiaedan miras kalan iki açısı vardır. Kenar açıları, kenara göre ifade edilir. Kenarın bu üç özelliği (uzunluk ve iki göreceli açı), kenar hareket ettiğinde veya döndüğünde değişmez ve tam olarak eşleşmek için ihtiyacımız olan şey budur [32]. Renk kenar

uzunluğu ve açılarla belirlenir. Benzer kenarlar benzer renklere sahiptir. Şekil.4.2'de gösterilmiştir.



Şekil 4.2. Kenar uzunluğu ve açılar

SourceAFIS daha sonra eşleştirilen iki parmak izi ile paylaşılan en az bir kenarı bulmaya çalışır. Bu, karma tabloyla karşılaştırılabilir performansa sahip en yakın komşu algoritması kullanılarak çok hızlı bir şekilde yapılır. Bu, her parmak izinden birer tane eşleşen minutiae çifti olan kök çifti verecektir.

Kök çiftinden başlayarak, SourceAFIS kenarları dışa doğru tarar ve birkaç eşleştirilmiş minutiae ve eşleştirilmiş kenarlardan oluşan bir eşleştirme oluşturur.

Şekil 4.3. Eşleştirme ağacı

Kök minutiae mavidir. Eşleştirme ağacı yeşildir. Destek kenarlarının grafiği sarıdır.

SourceAFIS'in algoritmanın son kısmı olan skoru çalıştırdığı yer burasıdır. Temel fikir, eşleştirilmiş her minutiaenın veya kenarın, rastgele gerçekleşmesi muhtemel olmayan bir olay olmasıdır. Bu tür olası olmayan olaylar ne kadar fazlaysa, eşleşmenin sadece bir tesadüf olması o kadar düşüktür. Bu yüzden algoritma aslında çeşitli eşleşme özelliklerini sayar ve aynı zamanda ne kadar çok eşleştiklerini de puanlar. Kısmi puanların son toplamı, bazı makul ölçeklere hizalanacak şekilde biçimlendirilir ve algoritmadan geri döndürülür. Uygulama puanı alır ve eşleşip eşleşmediğine karar vermek için bazı eşik değerlerle karşılaştırır [33].

SourceAFIS parmak izlerini karşılaştırdığında, gerçekten düşük seviyeli biyometrik özelliklerini, özellikle de minutiaeları (sırtlar ve çatallanmalar) karşılaştırır. Her

minutiae konumu, türü (bitiş veya çatallanma) ve karşılık gelen bitiş veya çatallanma yönü ile karakterize edilir. Bunları template olarak saklar.

4.1.2. Template (Öz Nitelik) Çıkarma

Öz nitelik çıkarma, parmak izi görüntüsünden biyometrik özelliklerin çıkarılması işlemidir. Öz nitelik çıkarımı, esas olarak biyometrik özellikleri kodlayan bir veri yapısı olan parmak izi şablonunu üretir. Şablon daha sonra gerçek eşleşme için eşleştiriciye beslenir. Bu ayrılmanın nedeni, özellik çıkarımının, tek bir şablon çifti eşleştirmekten daha düşük maliyetli olmasıdır. Uygulamalar, bir kereden fazla kullanılmaları muhtemel olduğunda şablonları önbelleğe almalıdır.

Şablonlar hafıza içi veri yapısıdır. Daha etkili önbelleğe almak için, özellikle büyük parmak izi veritabanlarını ararken, parmak izi şablonu seri hale getirilebilir ve sürekli olarak saklanabilir. Şablonu kalıcı bellekten yeniden yüklemek, parmak izi görüntüsünden yeniden oluşturmaktan genellikle daha hızlıdır.

SourceAFIS şablonları, yalnızca onları oluşturan belirli SourceAFIS sürümleriyle birlikte kullanılabilir. Eski SourceAFIS' te daha eski şablonları kullanmaya çalışmak, seri kaldırma veya eşleştirme sırasında istisnalara veya eşleşme sırasında doğruluk kaybına neden olabilir [34].

Uygulamalar her zaman orijinal parmak izi görüntülerini saklamalı ve iletmelidir. Serileştirilmiş şablonlar yalnızca yerel önbellek olarak ve yalnızca belirli bir sürüme tutturulmuş SourceAFIS kitaplığı ile kullanılmalıdır. SourceAFIS' in yükseltilmesi gerektiğinde, tüm şablonların parmak izi görüntülerinden çıkarılması gerekir.

SourceAFIS, ISO 19794-2 şablonlarını içe aktarmak için çok temel bir desteğe sahip ancak ISO 19794-2'nin kullanımı önerilmemektedir, çünkü ISO açık kaynaklı projelere düşmandır. ISO 19794-2 şablonları için dışa aktarma işlevi yoktur, bu nedenle ISO 19794-2 yine de kalıcı şablon ön belleklemesi için kullanılamaz.

4.1.3. Şablon

Seri hale getirilmiş SourceAFIS şablonu, JSON kodlu bir veri yapısıdır. Alan genişliği ve yüksekliği, girilen görüntünün 500 DPI 'ye yeniden ölçeklendirilmesinden sonraki boyutunu tanımlar. JSON verilerinin geri kalanı bir minutiae listesidir. Her minutiae aşağıdaki alanlar tarafından tanımlanmıştır:

X, Y - Minutiaenın 500 DPI piksel cinsinden konumu. Eksen y görüntünün en üstünde başlar ve alta doğru yükselir. Eksen x, görüntünün sol kenarında başlar ve sağa doğru yükselir.

Direction – Minutiae açısı. Sırtı uçları sırtı işaret eder. Sırttaki çatallanmalar çatallanma işleminin bölünmüş tarafına işaret eder. Açı, sıfır açısı sağa dönük ve saat yönünde artan radyanlarla ölçülür.

Type - Minutiae tipi, biten veya bifürkasyon.

Tablo 4.1. Örnek şablon yapısı


```
{
 "width": 452,
 "height": 325,
 "minutiae": [
  {
   "x": 65,
   "y": 365,
   "direction": 1.925456854072617,
   "type": "ending"
  },
  {
   "x": 36,
   "y": 125,
   "direction": 1.45896352529278375,
   "type": "ending"
  },
  {
   "x": 132,
   "y": 25,
   "direction": 6.136524856570089,
```

```
    "type": "bifurcation"
   },
   {
    "x": 245,
    "y": 365,
    "direction": 4.36548568251726,
    "type": "ending"
   },
   {
    "x": 152,
    "y": 256,
    "direction": 3.42365843933,
    "type": "ending"
   }
 ]
}
```


“Tablo 4.1.” de şablon yapısı gösterilmiştir.

4.1.4. Şeffaflık

SourceAFIS 'deki algoritma şeffalığı, uygulamaların, çeşitli filtreleme aşamalarındaki görüntüler, çıkarılan minutiaelar ve sırtlar, eşleştirici tarafından oluşturulan ağacı eşleştirme ve puanlama dağılımı dahil olmak üzere, özellik çıkarma ve eşleştirme sırasında oluşturulan tüm ara veri yapılarını yakalamalarına olanak sağlayan bir API' dir.

4.2. LFS Kütüphanesi

Bu kütüphane parmak izi görüntüsünden minutiae elde etmenize olanak sağlar ve "BOZORTH" kütüphanesinin bir girişi olarak kullanılır. Dizi kontrolü için kullanılır. "BOZORTH" kütüphanesi iki set minutiae arasında bir karşılaştırma yapar. "LFS" kütüphanesinin çıktısını alır ve bir eşleşme puanı döndürür. Eşleşen puan, iki parmak izi olma olasılığı artar. Bu araç FBI "Bozorth3" standardına dayanmaktadır. Şablon aşağıdaki gibidir [35].

X - Tespit edilen minutiae piksel x koordinatı.

Y - Tespit edilen minutiae piksel y koordinatı.

Direction - Tespit edilen minutiae yönü. Minutiae yönü, 0 birimi yukarı bakacak şekilde dikey bir şekilde başlıyor ve tam bir daireyi tamamlayan 11.25 derecelik artışlarla saat yönünde artıyor. Bu şema kullanılarak, tespit edilen bir minutiae açısı, sağa yatay temsil eden 8, düşey aşağıya işaret eden 16 ve sola yatay olan 24 temsil eden 24 ile 0 ila 31 aralığında ölçülür.

Reliability - Tespit edilen minutiae tayin edilen güvenilirlik ölçüsü, 0.0 ile 1.0 aralığında bir kayan nokta değeridir; 0.0, en düşük minimum kalitesini temsil eder ve 1.0, en yüksek minimum seviyeyi temsil eder [36].

Type - biten veya bifürkasyon.

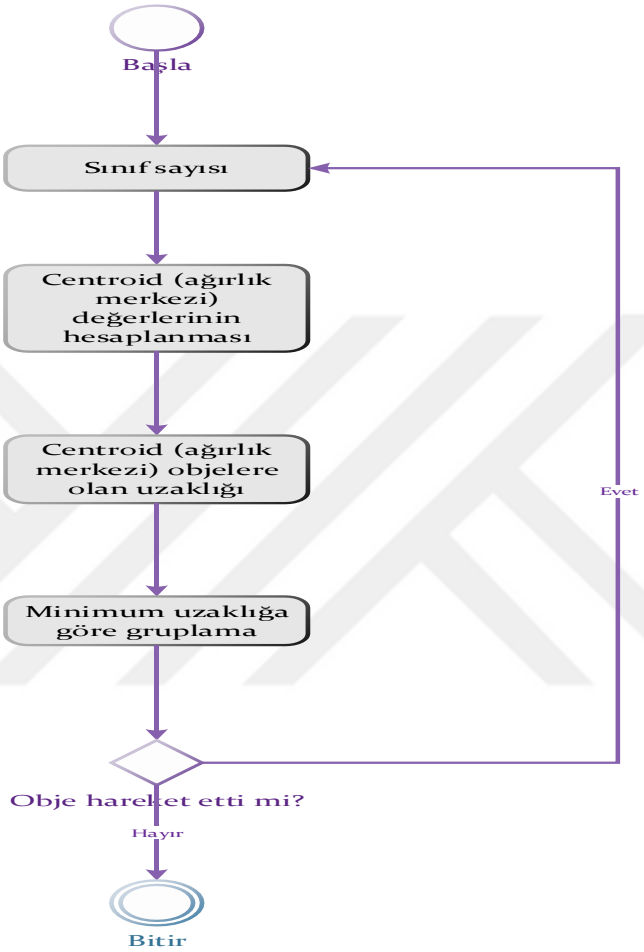
4.3. K-Means Clustering Algoritması

En eski kümeleme algoritmalarından olan K-Means, 1967 yılında J.B. MacQueen tarafından geliştirilmiştir. K-Means'in atama mekanizması, her verinin sadece bir kümeye ait olabilmesine izin verir. Merkez noktanın kümeyi temsil etmesi ana fikrine dayalı bir metottur. Verileri belirlenen küme sayısına göre, her küme için belirlenen ortalama değer doğrultusunda kümelenmesi üzerine çalışır. K-Means algoritması eldeki verileri k adet kümede ve kümelerin ortalamalarına göre kümelere ayırır. K küme sayısı kullanıcı tarafından verilir. Kısaca n tane nesneyi –küme içi benzerlik maksimum, kümeler arası benzerlik minimum olacak şekilde- k tane kümeye böler [37].

4.3.1. Çalışma Mantığı

K-Means algoritmasının çalışma mekanizmasına göre öncelikle her kümenin merkez noktasını (centroid) veya ortalamasını temsil etmek üzere k adet nesne -rasgele- seçilir. Kalan diğer nesneler, kümelerin ortalama değerlerine olan uzaklıkları dikkate alınarak en benzer oldukları kümelere dahil edilir. Daha sonra, her bir kümenin ortalama değeri hesaplanarak yeni küme merkezleri belirlenir ve tekrar nesne-merkez uzaklıkları

incelenir. Herhangi bir değişim olmayıncaya kadar algoritma ötelenmeye devam eder. En yaygın olarak kullanılan uzaklık hesaplama formülü öklit uzaklık formülüdür.

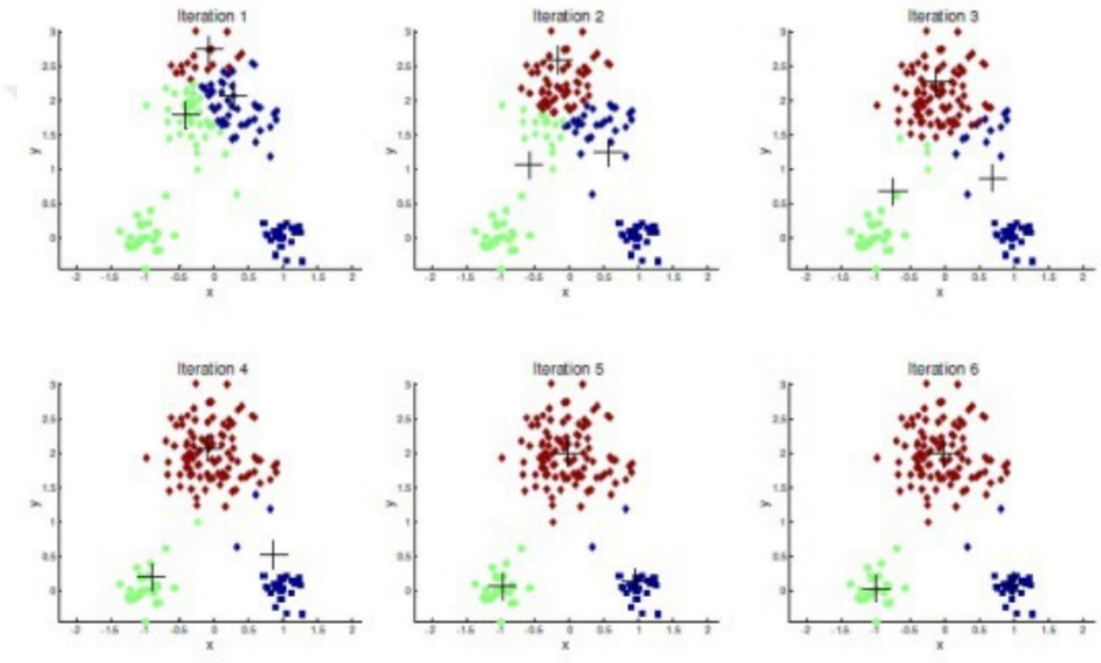


Şekil 4.4. K-Means çalışma mantığı

4.3.2. İşlem Adımları

Merkez noktaların belirlenmesi, başlangıç küme merkezlerinin seçimi k-Means' in sonucunu önemli oranda etkiler. Başlangıç noktalarının belirlenmesinde çeşitli teknikler vardır. Bu tekniklerden bazıları aşağıdaki gibidir.

1. K sayısı kadar rastgele veri seçilip küme merkezleri olarak atanır.
2. Veriler rastgele k tane kümeye atanır ve küme ortalamaları alınarak başlangıç küme merkezleri belirlenir.
3. En uç değerlere sahip veriler küme merkezleri olarak seçilir.
4. Veri setinin merkezine en yakın noktalar başlangıç noktaları olarak seçilir.



Şekil 4.5. K-Means işlem adımları

K- Means Algoritmasına göre kümeleme yapılırken, ilk olarak karışık halde verilmiş olan veri seti sıralanır. (K-Means Kümeleme işlem adımı - 1) Sıralama işleminden sonra, her verinin başlangıçta rastgele belirlenmiş olan merkez noktalarına göre uzaklığı alınır. Veriler en yakın olduğu merkez noktasının kümesine dahil olur. (K-Means Kümeleme işlem adımı - 3) Bu adımdan sonra her küme için küme elemanlarının ortalaması alınır. Bu ortalama yeni merkez noktasıdır. (K-Means Kümeleme işlem

adımı - 4) Sonraki adımda, tekrar her verinin merkez noktalarına olan uzaklığı hesaplanır ve veriler en yakın olduğu merkez noktasının kümesine dahil edilir. Küme elemanlarının ortalaması alınıp yeni merkez noktaları belirlenir. Kümeleme işleminin sonucu, bir sonraki adımda aynı çıkına kadar bu işlem tekrarlanır.

4.3.3. Avantaj – Dezavantajları

Uygulanabilirliği kolaydır ve büyük veri kümelerinde hızlı çalışabilir. Veri sayısı çok fazla olan hesaplamalarda, K-Means, küme sayısı küçük ise hesaplamaları, hiyerarşik kümelemeden daha hızlı yapar.

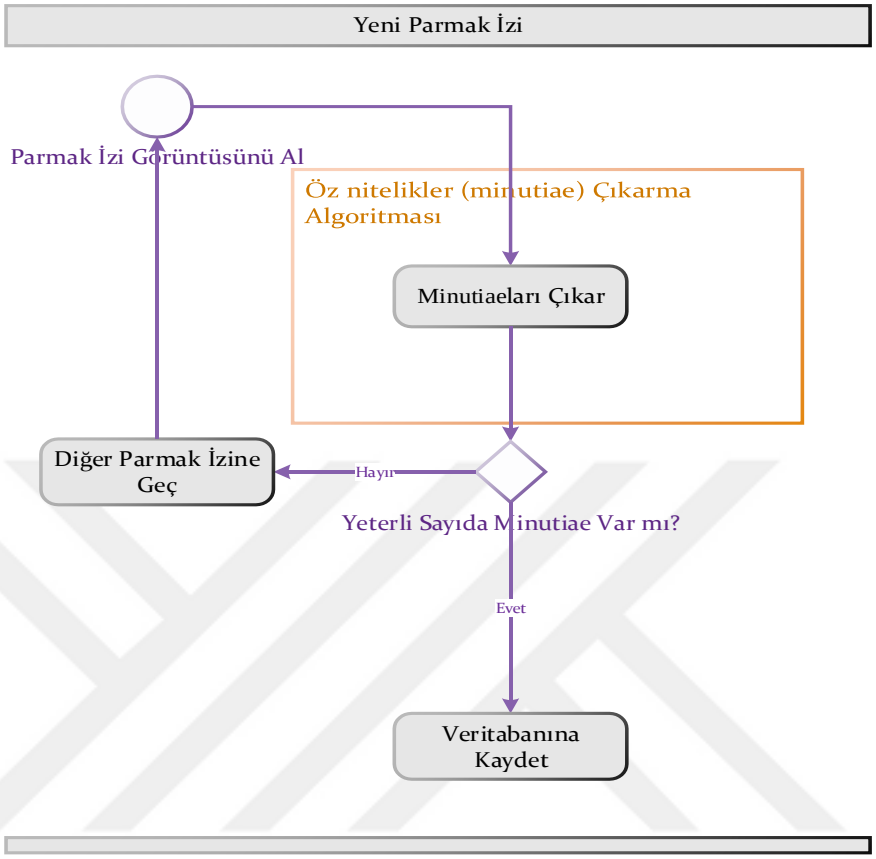
Dezavantajı, K-Means algoritması k küme sayısını tespit edememektedir. Bu nedenle uygun k sayısını bulana kadar bir deneme yanılma süreci gerçekleşmektedir. Gürültülü verilere duyarlıdır. Bu veriler de kümelere dahil edilir.

5. BİR HIZLI PARMAK İZİ DOĞRULAMA SİSTEMİ TASARIMI

Bu bölümde bir hızlı parmak izi bulma sistemi tasarımı açıklanacaktır. Tasarımda parmak izi alımını ve teşhis yöntemi senaryolara göre diyagramlarla açıklanmaktadır.

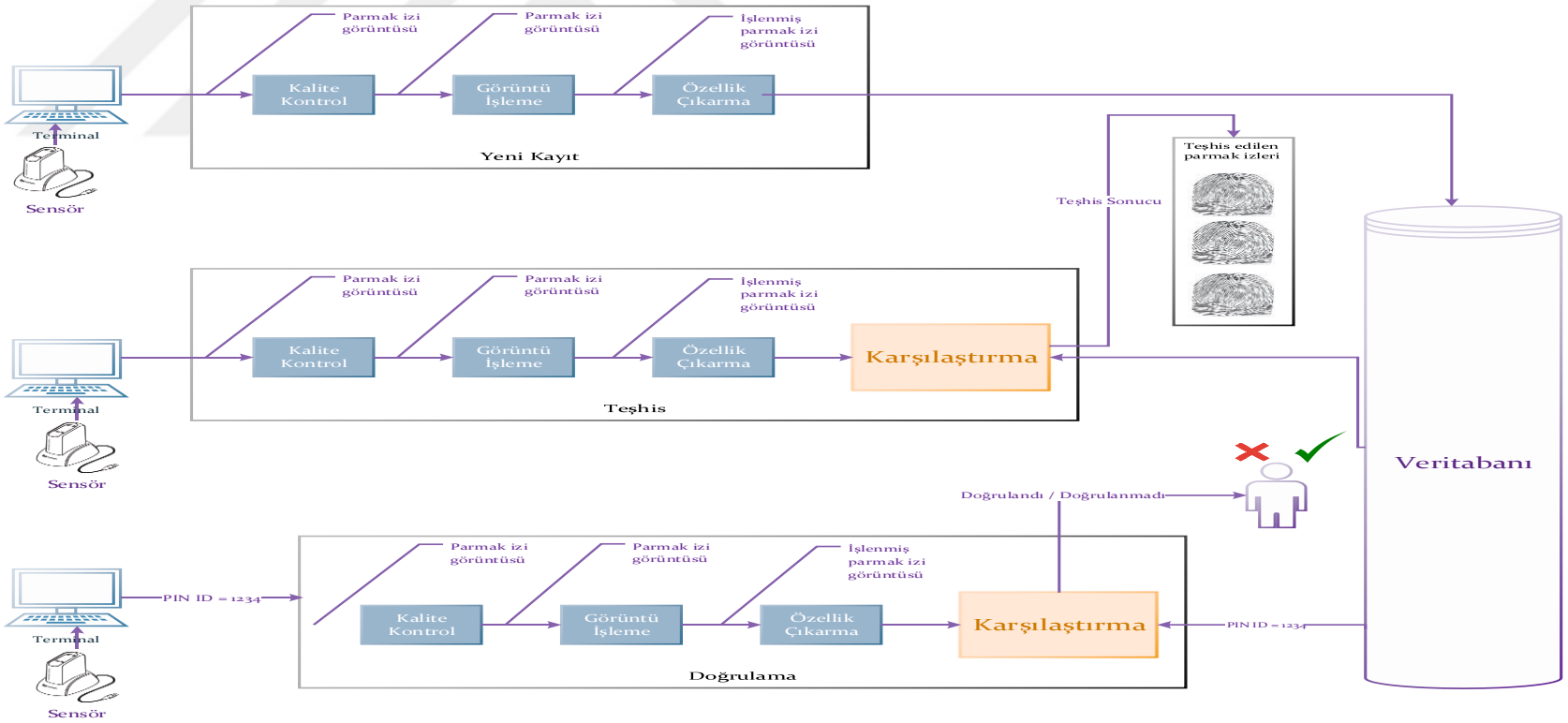
5.1. Alışılmış Parmak İzi Sistemi

Dijital sensörler aracılığıyla parmak izini alınır. Parmak izi görüntüsü kalitesiz ise işlem tekrarlanır Alınan parmak izi görüntü kalitesini daha iyi hale getirilir. Yüksek kaliteli bir görüntüye dönüştürmek için hasar alanı belirlenir ve ortadan kaldırılmaya çalışılır. Eksik minutiaeları ortaya çıkarmak için parmak izi üzerinde iyileştirme algoritmalar kullanılır. Parmak izinin farklı özelliklerini çıkarılır. Çıkarılan özellikler (kalıplar) veritabanına kaydedilir. Parmak izi kaydetme işlemi 5.1. diyagramında gösterilmiştir.



Şekil 5.1. Alışılmış sistemde yeni parmak izi kaydetme

Teşhiş işleminde, veritabanındaki tüm kullanıcılara ait fizyolojik karakteristik bilgilerinin yer aldığı kalıplarının karşılaştırıldığı bir işlemdir. Doğrulama işleminde özellik karşılaştırıcı ile PIN kodu girilmiş olan kişinin sensörden alınan parmak izinden alınan kalıp ile kayıttaki kalıpla uyumluluğu saptanır. Bütün süreç Şekil.5.2. de diyagramında gösterilmiştir.



Şekil 5.2. Alışılmış parmak izi sistemi

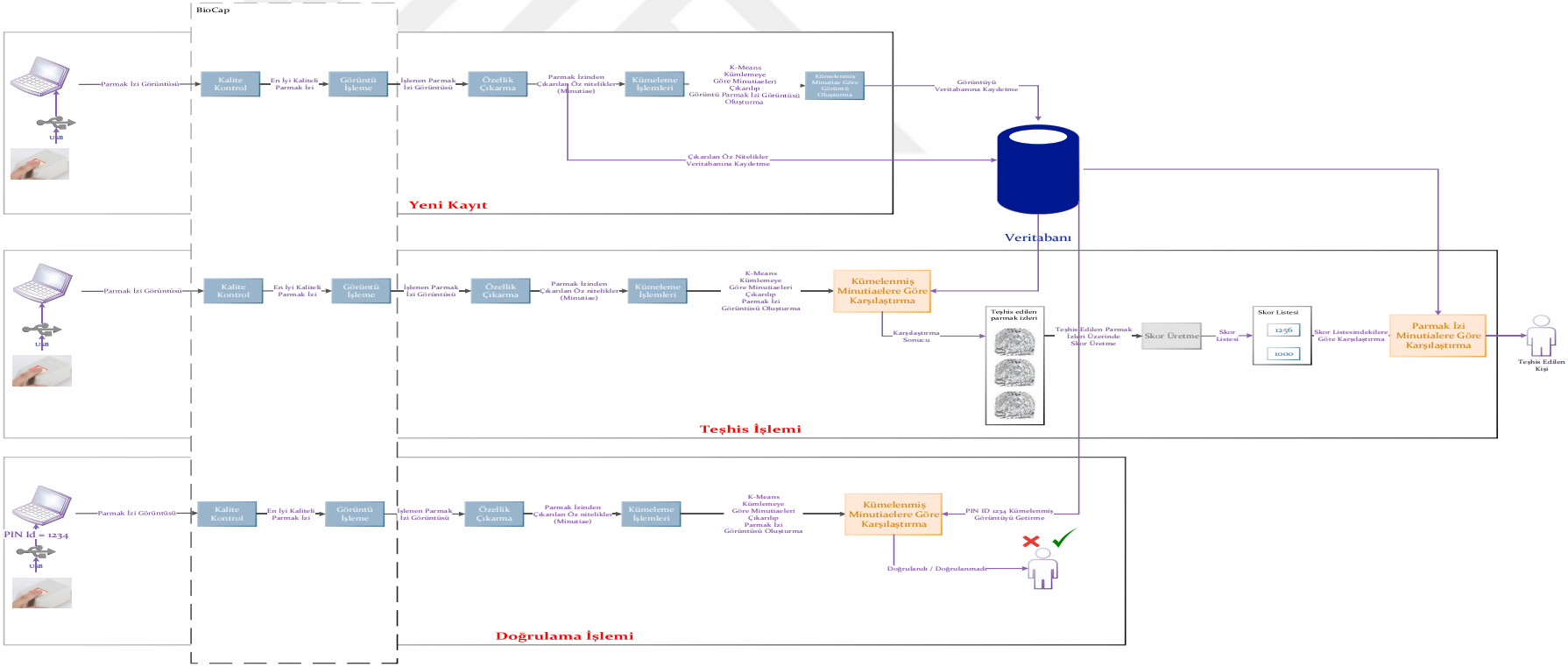
5.2. Önerilen Sistem Mimarisi

Önerilen sistemde parmak izleri kendisi ve parmak izlerinin minutiaeların kümeleme algoritması ile kümelenen minutiaelara göre oluşturulan parmak izi görüntüsü saklanmasına dayalı bir sistemdir. Kümeleme algoritması olarak "K-Means Clustering" algoritması kullanılmıştır. K means clustering algoritması "Kullanılan Teknolojiler Bölümünde" değinilmiştir.

Parmak izinin orijinal görüntüsü veritabanında tutulduğundan parmak izi veri bütünlüğü bozulmamıştır.

Sistemde alışılmış sitemler gibi 3 ana modülden oluşmuştur. Bunlar kayıt (enrollment), teşhis (idenfication) ve doğrulama (verification) modülünden oluşmaktadır.

Sistemin modüllerinde ortak olan BioCAP modülü ise sensörden alınan parmak izi görüntüsü kalitesiz ve minutiae sayısı yetersiz ise parmak izi tekrar alınır. Bu tekrar alma işlemi kullanıcı fark edilmeden yapılmaktadır. Belli bir alımdan sonra alınan parmak izleri karşılaştırılıp en iyisi olanı kabul edilir.

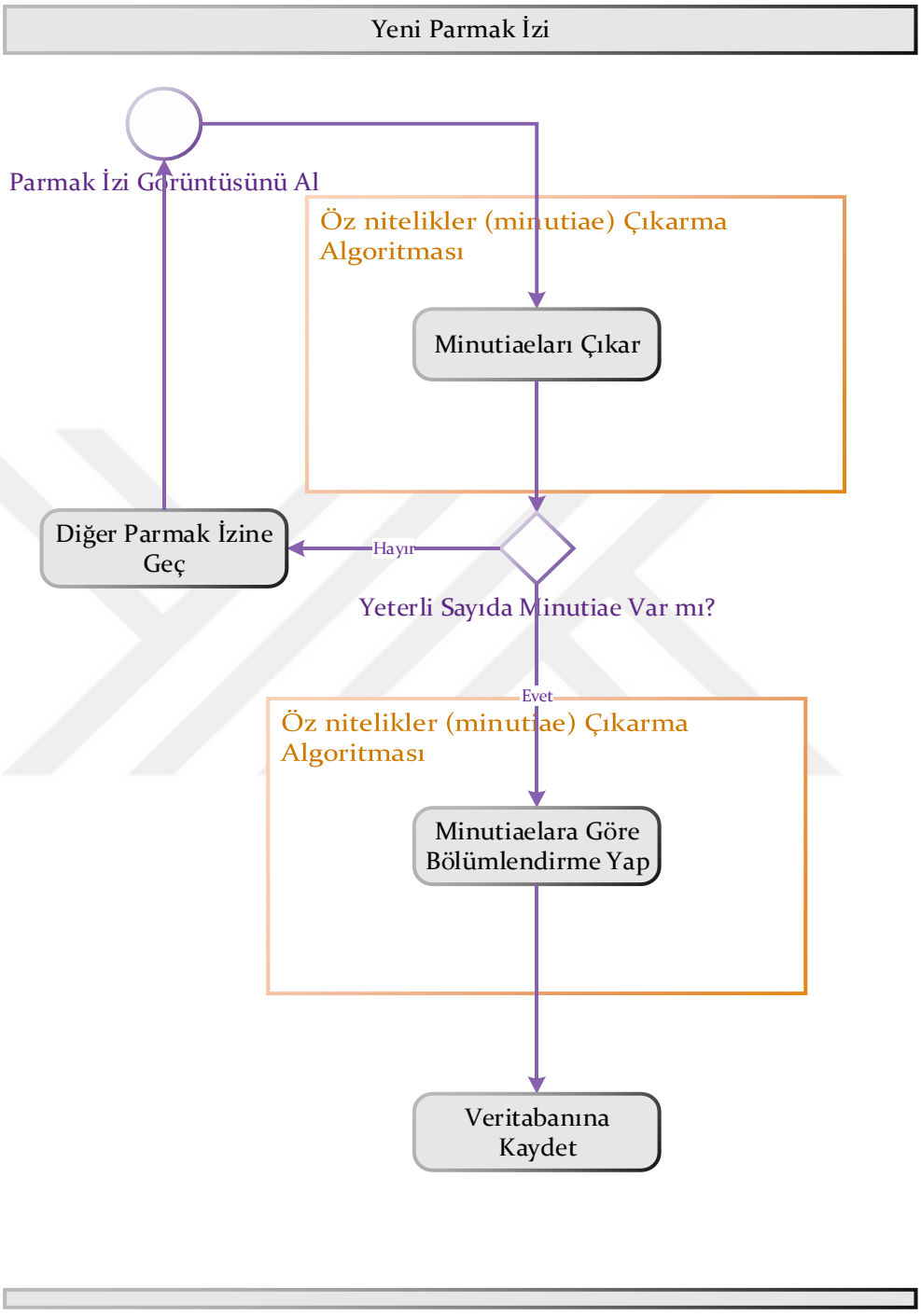


Şekil 5.3. Önerilen sistem diyagramı

Parmak izi onayı geldiğinde veri tabanında en öncelikli bölümlerde (segmentte) arama yapılır. Çakışma olması durumunda sonuç bulunana dek daha öz öncelikli bölümlerde (segmentlerde) aramaya devam edilir.

5.3. Parmak İzi Verisi Hazırlama

Dijital sönserden alınan ya da dosyadan okunan parmak izi görüntüsü iki aşamadan geçmektedir. Bunlar minutiae çıkarımı ve minutiaelarda yapılan kümeleyip bölümlendirilerek parmak izi görüntüsünü elde etme işlemidir.

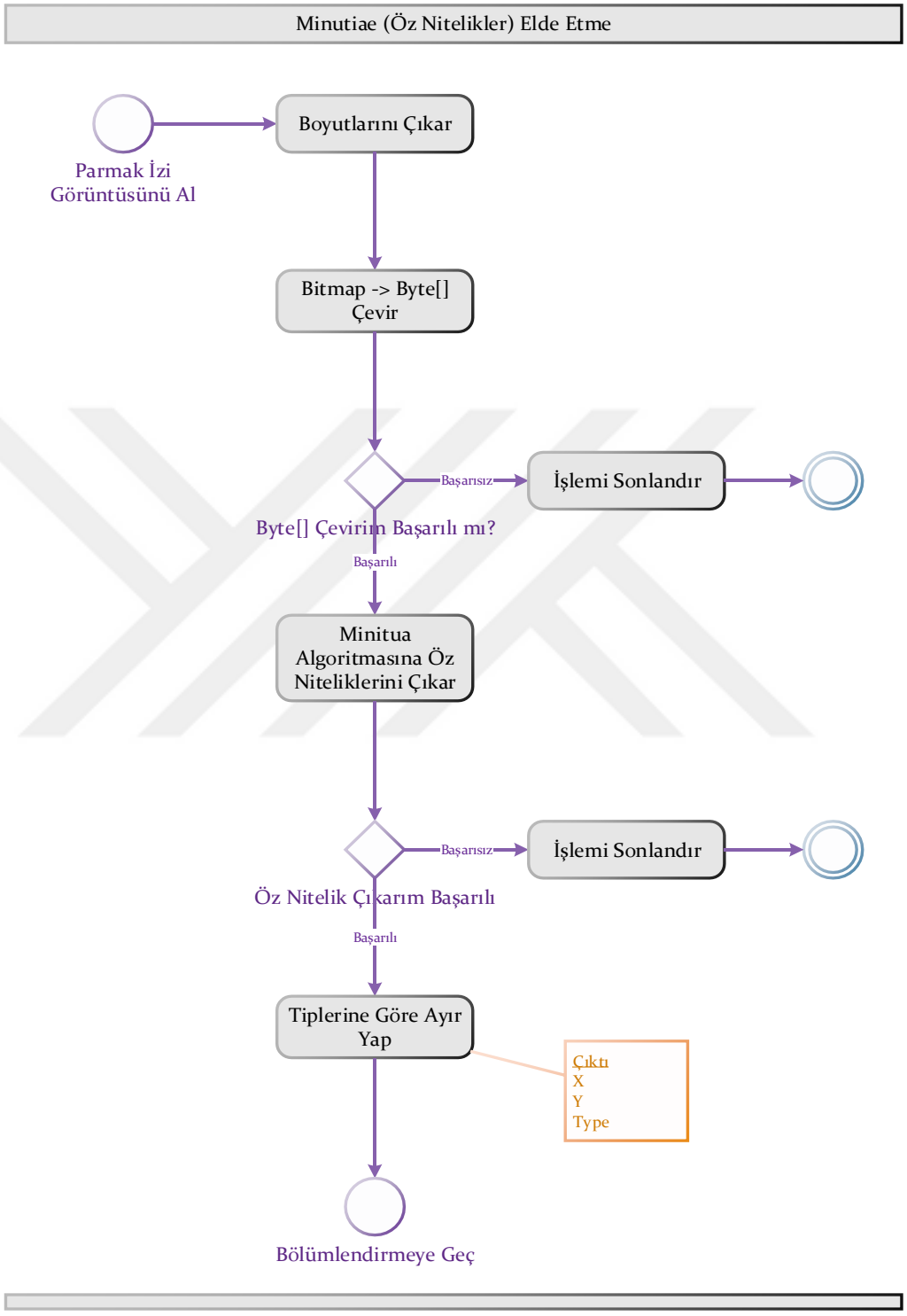


Şekil 5.4. Parmak izi hazırlama

5.3.1. Minutiaeları Çıkarma Yöntemi

Parmak izi görüntüsünde bölümlendirme yapılması için minutiaelar elde edilmelidir. Şekil 5.5.' de diyagramda açıklanmıştır.

Diyagrama göre parmak izi görüntüsü boyutları çıkarılır. Boyutlarına göre 8 bit gray scale parmak izi görüntüsü byte dizisine çevrilir. Çevirim başarılı ise minutiae algoritmasına göre öz nitelikler çıkarılır. Byte dizisi çevirim başarısız ise işlem sonlandırılır. Öz nitelikler çıkarma işlemi tamamlandıktan sonra öz nitelik dizisi elde edilir. Öz nitelik dizisinde X, Y öz nitelik tipi ve öz nitelik kalite derecesi bilgileri tutulur. Öz nitelik çıkarım işlemi başarısız ise işlem sonlandırılır. Öz nitelik çıkarım işlemi başarılı olduktan sonra bölümlendirmeye geçilir.

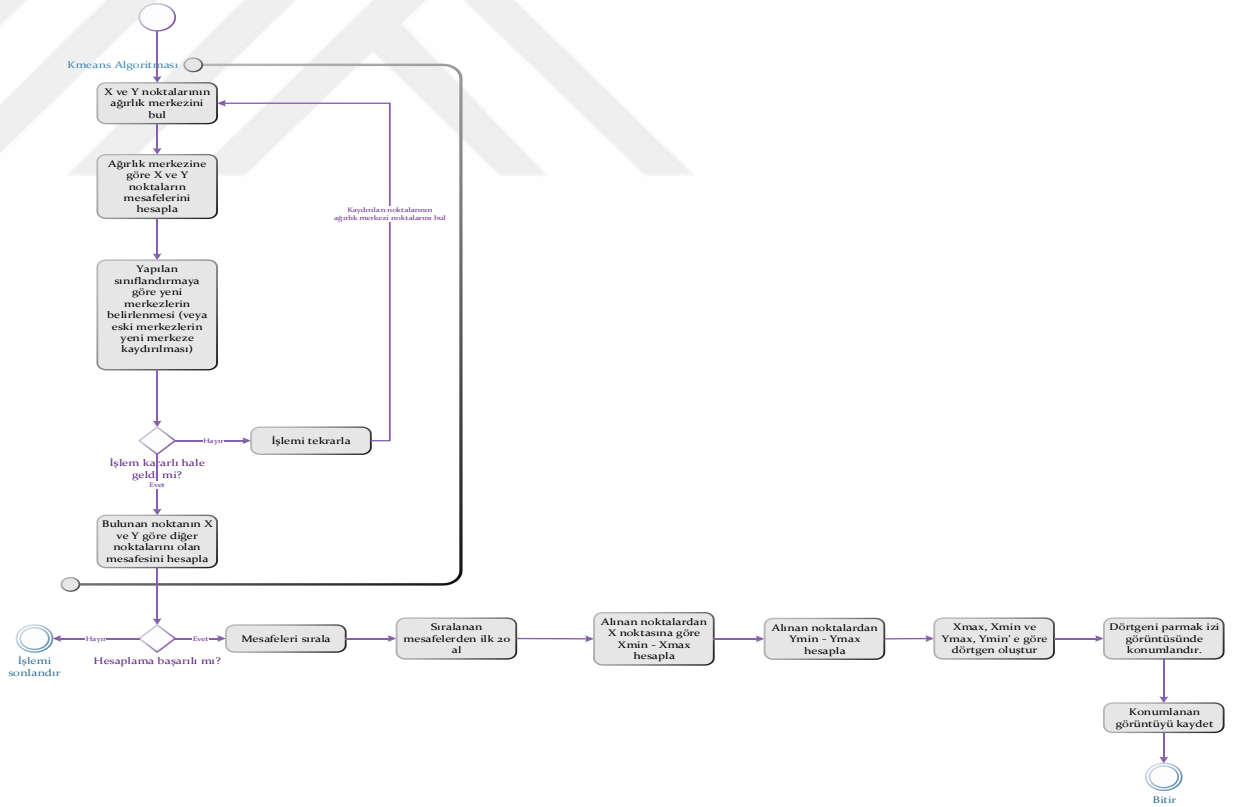


Şekil 5.5. Minutiae elde etme

5.3.2. Minutiaeları Kümeleyerek Bölümlendirme Yöntemi

Parmak izi görüntüsünden minutiaelar bölümlendirmesi Şekil 5.6. diyagramda açıklanmıştır.

Diyagrama göre minutiaelardan elde edilen X ve Y noktalarının ağırlık merkezi bulunur. Bulunan ağırlık merkezine göre X ve Y noktalarının mesafeleri hesaplanır. Hesaplanan mesafelere göre sınıflandırma yapılır. Sınırlandırmalara göre yeni merkezler belirlenir. Belirleme, eski merkezlerinin yeni merkezlere kaydırılması işlemidir. İşlem kararlı hale gelmesi için işlemler tekrarlanır. İşlem kararlı hale geldiğinde elde edilen X ve Y noktası kararlı bir noktadır. Kararlı noktanın diğer noktalara göre mesafeler hesaplanır. Mesafeler küçükten büyüğe doğru hesaplanır. Sıralanan mesafeler belli bir sayıya göre listeye alınır. Oluşturulan listede X noktasına göre Xmin ve Xmax hesaplaması yapılır. Aynı işlem Y noktası için gerçekleştirilir. Y noktasına göre Ymin ve Ymax hesaplaması yapılır. Hesaplanan Xmin, Xmax, Ymin ve Ymax dörtgeni oluşturulur. Dörtgen orijinal parmak izi görüntüsü üzerine konumlandırılır. Konumlandırılan dörtgen içindeki görüntü alınır ve kişinin parmağıyla ilişkilendirilerek kişinin biyometrisine kaydedilir.

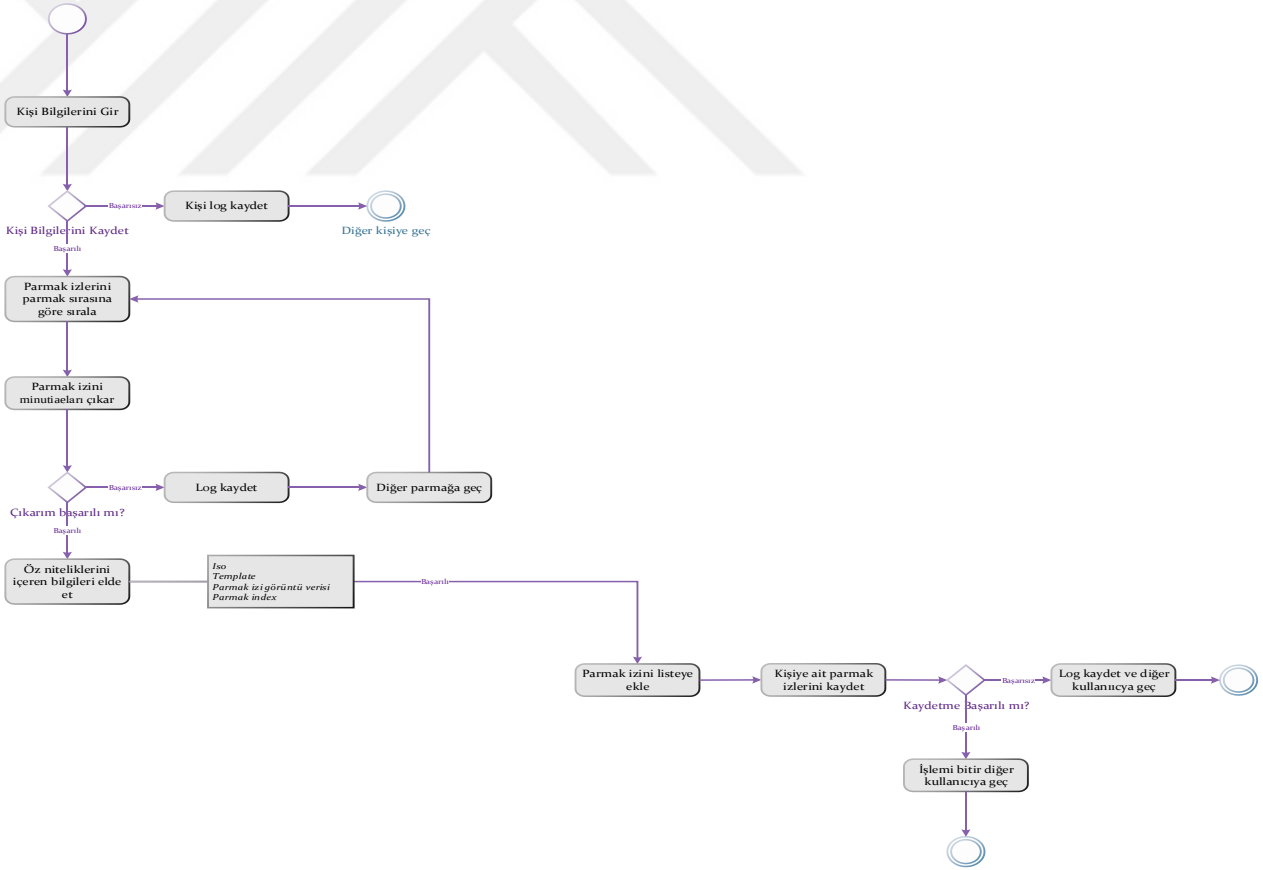


Şekil 5.6. Parmak izi görüntüsünü kümeleyip bölümlendirme

5.4. Yeni Kayıt (Enrollment)

Yeni kayıt oluşturma Şekil 5.7.'te diyagramda açıklanmıştır.

Kişi bilgileri girilerek kaydedilir. Kaydetme işlemi başarılı ya da başarısız ise log kaydedilir. Kaydetme işlemi başarısız ise diğer kişiye geçilir. Dijital sensörler aracılığıyla kişinin parmak izleri alınır. Alınan parmak izleri kalitesiz, minutiae sayısı yetersiz ise parmak izi alımı tekrarlanır. Belli bir sayıda alınan parmak izlerinin içinden en iyisi olan parmak izi kabul edilir. Alım tamamlandıktan sonra kişinin parmak izleri parmak sırasına göre sıralanır. Parmak izinin minutiaeları çıkarılır. Minutiae çıkarım işlemi log kaydedilir. Minutiae çıkarım işlemi başarısız ise diğer parmağa geçilir. Başarılı ise öz niteliklerin bilgileri ISO, template, parmak izi orijinal görüntüsü ve parmak izi indeksi elde edilir. Minutiaelara göre kümeleme yapılır. Kümelenen parmak izi indeksi ile ilişkilendirilir. İlişik başarılı ise diğer bölümlendirme ilişkilendirilir. Böylece elde edilen, parmak indeksine ait parmak izinin kümelenerek bölümlendirilmiş listesidir. Bölümlendirme parmak izi indeksiyle ilişkilendirme tamamlandığında veriler kaydedilerek diğer parmak için aynı işlemler yapılır. Böylece orijinal parmak izi ile kümelenen parmak izi veritabanından tutulmuş olur. Her işlem adımı loglanarak veritabanında kaydedilmektedir. Kullanıcıya ait parmak izi işlemleri tamamlandığında kullanıcı tam anlamıyla kaydedilmiş olur ve diğer kişiye geçilir.



Şekil 5.7. Yeni Kayıt

5.5. Teşhis (Idenfication) İşlemi

Herhangi bir kişi için dijital sensörden alınan kaliteli parmak izleri ya da dosyadan okunan parmak izi görüntüleri, parmak izi verisi hazırlama modülünden geçer. Bu modülden kümelenerek bölütlenmiş parmak izi görüntüsü ile orijinal parmak izi görüntüsü elde edilir. Veritabanından bölütlenmiş parmak izi verileri getirilir. Bu parmak izi verileri karşılaştırılır. Eğer çakışma yok ise kişi teşhis edilmiştir. Eğer çakışma çakışma olduğu takdirde çakışan parmak izleri ile bölümlendirilmiş parmak izlerinden skorlar üretilir. Eğer skorlar eşik değerinden büyük olup tek skor ise kişi teşhis edilmiştir. Eğer tek skor olmayıp skor dizesi var ise skor dizesindeki orijinal parmak izleri veritabanından getirilir ve karşılaştırılarak kişi teşhis edilir.

Durumları daha iyi anlaşılması için senaryolar bölümünde anlatılmıştır.

5.6. Doğrulama (Verification) İşlemi

Kişi için PIN kodu girilir. Kişiye ait parmak izleri dijital sensörden ya da dosyadan okunur. Okunan ya da alınan parmak izi verisi hazırlama modülünden geçer. Bu modülden kümelenerek bölütlenmiş parmak izi görüntüsü ile orijinal parmak izi görüntüsü elde edilir. Veritabanından kişinin girmiş olduğu PIN koduna ait bölütlenmiş parmak izi verileri getirilir. Bu parmak izi verileri karşılaştırılarak skor üretilir. Skor sıfırdan büyük ise kişi doğrulanmış olur.

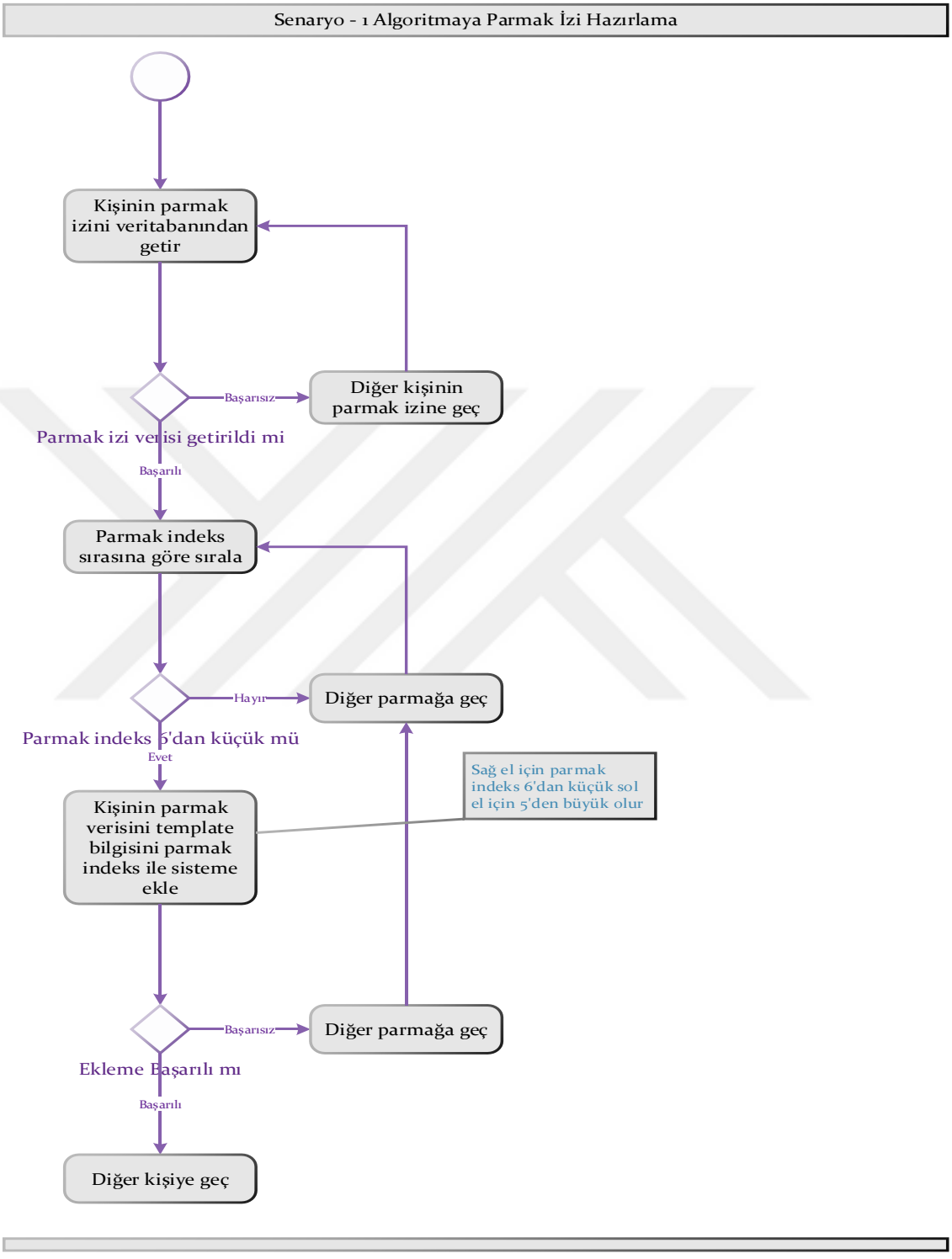
5.7. Senaryolar

Kişiyi teşhis ederken karşılaşılabilecek durumlar için senaryolar üretilmiştir. Bu bu bölümde senaryolara değinilecektir.

5.7.1. Senaryo - 1 Algoritmaya Parmak İzi Hazırlama

Senaryo – 1 için algoritmaya göre parmak izi hazırlama Şekil 5.8.'de diyagramda açıklanmıştır.

Kişilere ait parmak izi veritabanından getirilir. Parmak izi yok ise diğer kişiye geçilir. Parmak indeksine göre parmak izleri sıralanır. Ele göre işlemler ayrı yapılır. Parmak indeksi 6'dan küçük ise sağ el 5'den büyük ise sol el anlamına gelmektedir. Parmak indeksine göre kişinin parmak template verisi ile sisteme eklenir. Ekleme başarılı ise diğer parmak indeksine geçilir. Başarısız ise diğer kişiye geçilir ve işlemler tekrarlanır.

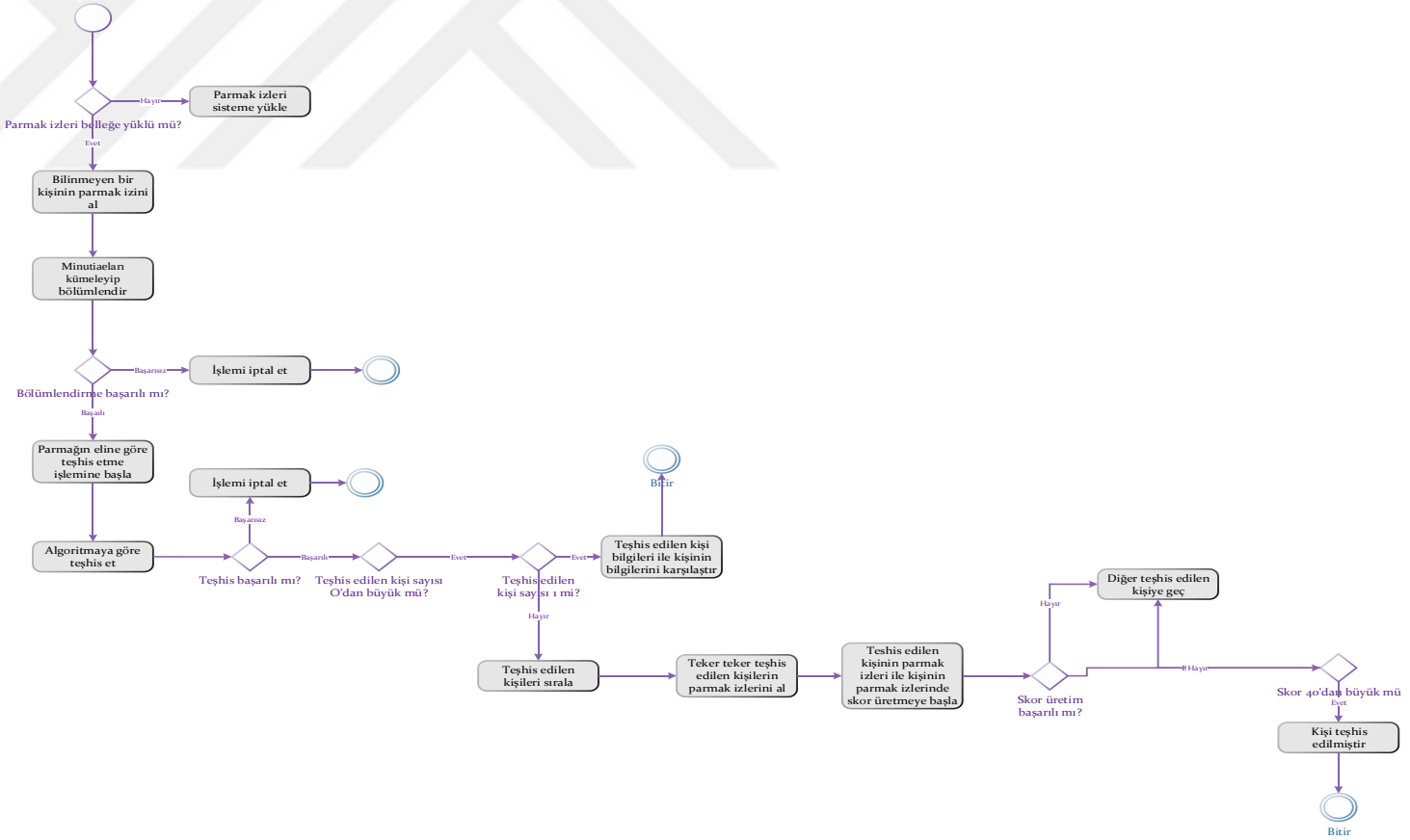


Şekil 5.8. Senaryo – 1 göre algoritmaya parmak izi hazırlama

5.7.2. Senaryo - 1 Teşhis (Idenfication) İşlemi

Senaryo – 1 için teşhis işlemi Şekil 5.9.'da diyagramda açıklanmıştır.

Parmak izlerinin sisteme yüklenmesi gerekir. Parmak izi sisteme yüklü değilse sistemin yüklenmesi için beklenir. Bilinmeyen bir kişinin parmak izi görüntüsü alınır. Parmak izi minutiaeları için bölümlendirme işlemi yapılır. Bölümlendirme işlemi başarısız ise işlem iptal edilir. Başarılı ise parmağın eline göre teşhis etme işlemine başlanır. Teşhis işlemi algoritmaya göre işlemektedir. Teşhis işlemi başarısız ise işlem iptal edilir. İşlem başarılı olduğunda teşhis edilen kişi sayısı bir kişi ise teşhis edilen kişi bilgileri kişinin bilgileri karşılaştırılır. Teşhis edilen kişi sayısı birden fazla ise teşhis edilen kişilerin bilgileri parmak izleri getirilir. Teşhis edilen kişilerin parmak izleri ile kişi bazında kişinin parmak izleriyle skor üretilir. Skor üretim başarısız ise diğer kişiye geçilir. Skor 40'dan büyükse ise kişi teşhis edilmiş olup işlem tamamlanır.

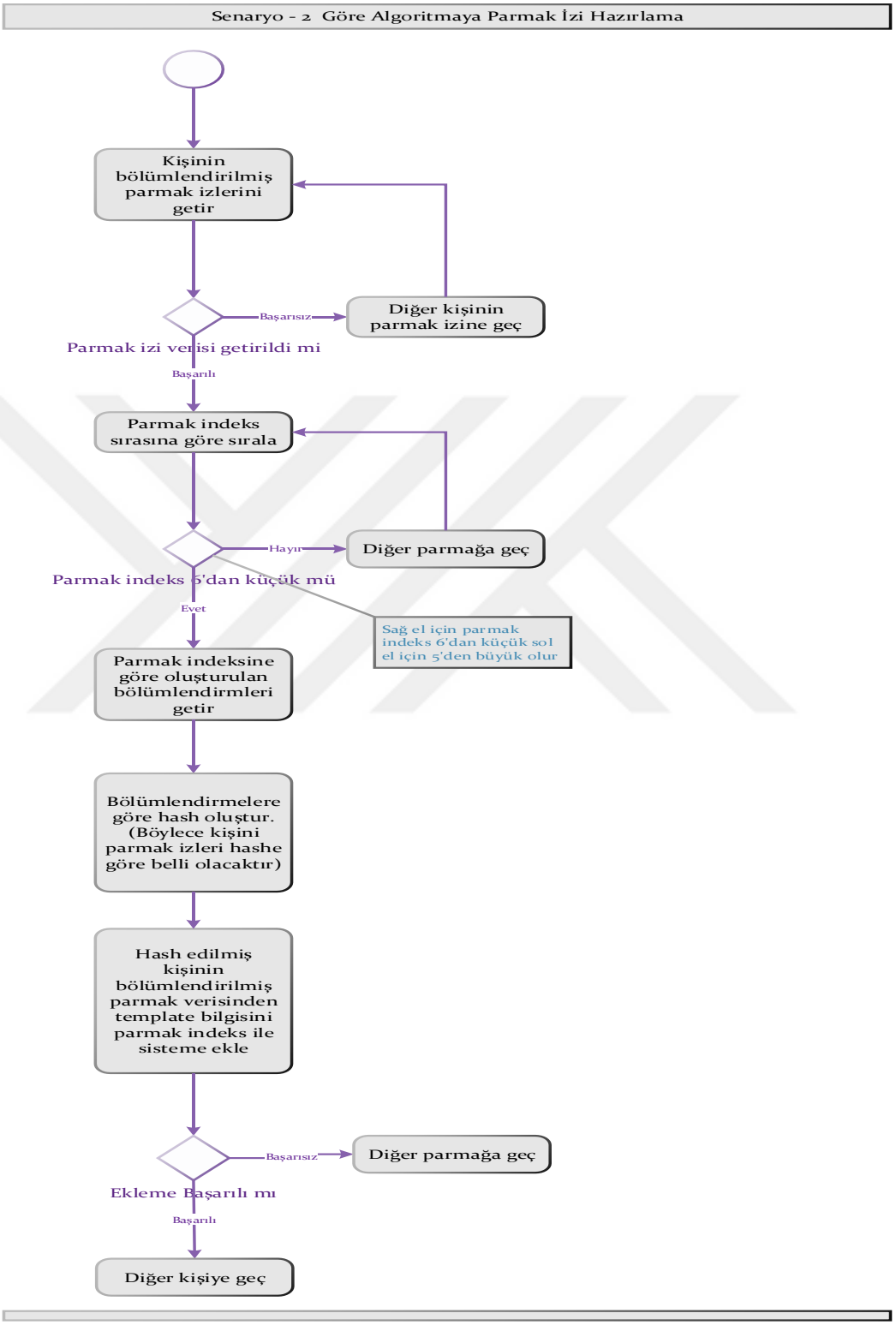


Şekil 5.9. Senaryo – 1 göre teşhis

5.7.3. Senaryo - 2 Algoritmaya Parmak İzi Hazırlama

Senaryo – 2 göre algoritmaya parmak izi hazırlama Şekil 5.10. diyagramda açıklanmıştır.

Kişinin bölümlendirilmiş parmak izi veritabanından getirilir. Parmak izi verisi yoksa veya parmak izi veritabanından getirilemezse işlem iptal edilir. Parmak indeksine göre parmak izleri sıralanır. Ele göre işlemler ayrı yapılır. Parmak indeksi 6'dan küçük ise sağ el 5'den büyük ise sol el anlamına gelmektedir. Parmak indeksine göre veritabanından getirilen bölümlendirmeler için hash oluşturulur. Böylece parmak izleri hashe göre belli olacaktır. Parmak indeksi bazında hash edilmiş kişinin bölümlendirilmiş parmak izi verisinden template bilgisini parmak indeksi ile sisteme eklenir. Ekleme başarılı ise diğer parmak indeksine geçilir. Başarısız ise diğer kişiye geçilir ve işlemler tekrarlanır.

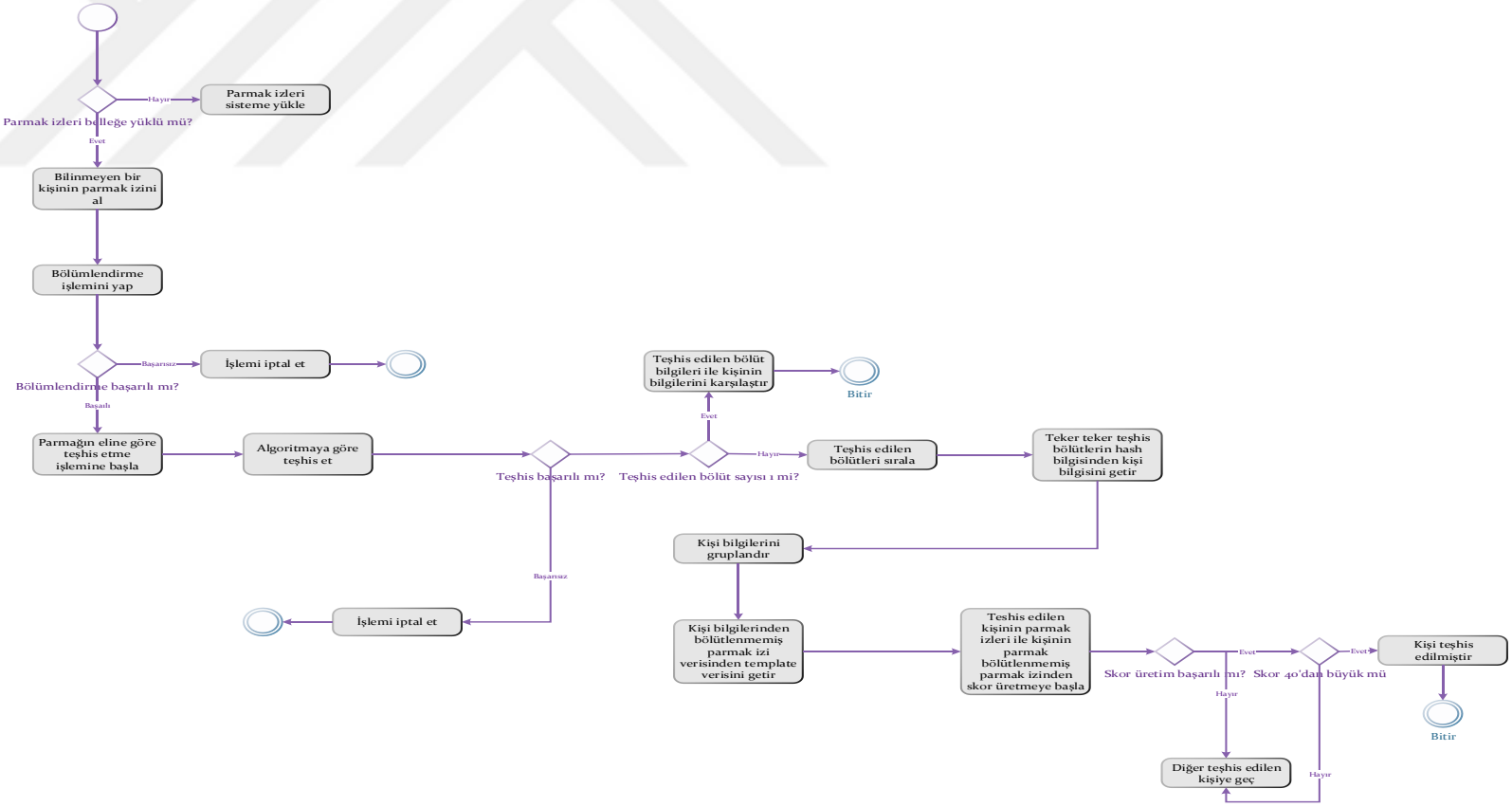


Şekil 5.10. Senaryo – 2 göre algoritmaya parmak izi hazırlama

5.7.4. Senaryo - 2 Göre Teşhis (Idenfication) İşlemi

Senaryo – 2 için teşhis işlemi Şekil 5.11 ’de diyagramda açıklanmıştır.

Parmak izlerinin sisteme yüklenmesi gerekir. Parmak izi sisteme yüklü değilse sistemin yüklenmesi için beklenir. Bilinmeyen bir kişini parmak izi görüntüsü alınır. Parmak izi minutiaeları için bölümlendirme işlemi yapılır. Bölümlendirme işlemi başarısız ise işlem iptal edilir. Başarılı ise parmağın eline göre teşhis etme işlemine başlanır. Teşhis işlemi algoritmaya göre işlemektedir. Teşhis işlemi başarısız ise işlem iptal edilir. İşlem başarılı olduğunda teşhis edilen bölüt sayısı bir ise bölüte sahip kişi bilgisiyle kişinin bilgileri karşılaştırılır. Teşhis edilen bölüt sayısı birden fazla ise teşhis bölüt bilgisinin hash bilgisinden kişi bilgisine ulaşılır. Ulaşılan kişi bilgileri gruplanır. Kişilerin bilgileri parmak izleri bilgilerinden template bilgisine getirilir. Teshis edilen kişinin parmak izleri ile kişinin parmak bölütlenmemiş parmak izinden skor üretilir. Skor üretim başarısız ise diğer bölüte geçilir. Eğer skor üretimi başarılı ve skor 25’den büyükse ise kişi teşhis edilmiş olup işlem tamamlanır.



Şekil 5.11. Senaryo – 2 göre teşhis işlemi

5.8. Uygulama Tasarımı

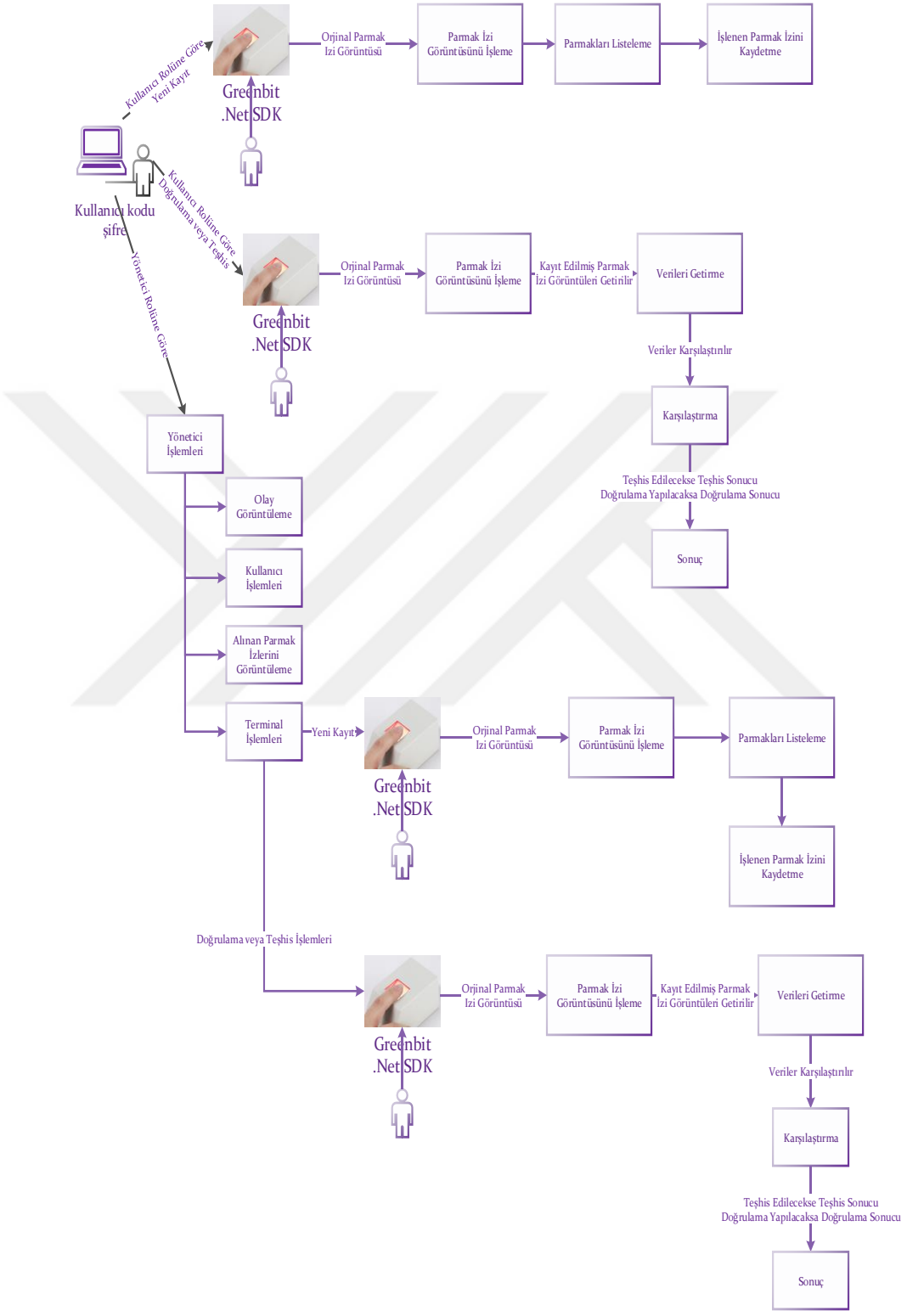
Bu bölümde, önerilen sistemin modülerine göre uygulama tasarlanmıştır. Tasarlanan uygulama Windows ortamında çalışan X64 ve X86 mimariye destek veren masaüstü uygulamasıdır. Uygulama kullanıldığında gereksinim olarak Greenbit USB Driver ve .Net Framework yüklü olması gerekmektedir.

Uygulamaya girildiğinde kullanıcı adı ve şifresi sormaktadır. Girilen kullanıcı sistemde yok ise uyarı vermektedir. Kullanıcı sistemde ise kullanıcı rolüne göre ekranlara yönlendirilir. Kullanıcı rolleri; yönetici, yeni kayıt ve doğrulama ya da teşhis olup özelliklerine uygulama dinamik yapıdadır. Kullanım kolaylığı ve öğrenilmesi kolaydır. Tasarım 5.12. diyagramında açıklanmıştır.

Uygulama rollerine göre 3 ana kısımdan oluşmaktadır.

1. **Yeni Kayıt Rolü:** Bu role sahip kullanıcılar yeni kayıt yaparlar. Kayıt yaparken parmak izini sensör ya da dosyadan okuyarak kaydederler. Kayıt sonucunu ekranda gösterilmektedir. Daha önceden kayıtlı olan kişi tekrar kayıt olmak isterse kullanıcıyı uyarır. İşlemi iptal eder.
2. **Teşhis ve Doğrulama Rolü:** Bu role sahip kullanıcılar gelen kişiyi teşhis etmek için kullanılır. Teşhis ederken parmak izini sensör ya da dosyadan okur. Doğrulama yapacaksa kişiden PIN numarası girilmesi istenir. Daha önceden kaydedilmiş veriler yüklü olduğu için teşhis ya da doğrulama sonucunu ekranda gösterilmektedir. Teşhis edilen kişi bilgileri ekranda görülmektedir.

3. **Yönetici Rolü:** Bu role sahip kullanıcılar kullanıcı yönetimi, terminal işlemlerini, yeni kayıt rolüne ait alınan parmak izlerini görüntüleme ve olay görüntüleme işlemlerini yapmaktadır. Olay görüntüleme ile terminallerden oluşan hata loglarını görüntüleyip müdahale edebilir. Kullanıcı işlemleri ile kullanıcı listeleme, ekleme, çıkarma ve rol atama işlemlerini yapar. Terminal işlemleri ile terminallere IP ataması yapar. Terminallere bağlantı olup olmadığını kontrol edebilir.



Şekil 5.12. Uygulama tasarımı

5.9. Yazılım Tasarımı

Yazılım Visual Studio 2019 ortamında C# dilinde yazılmıştır. C# ile .Net 4.7 Framework kullanılmıştır. Uygulama Windows desktop uygulamasıdır. Uygulama çok katmanlı mimariye sahiptir. Verilerin saklanması için veritabanı olarak MSSQL 2017 Standart sürümü kullanılmıştır.

ORM olarak Entity Framework ORM (Object Relational Mapping) araçları kullanılmıştır. ORM, ilişkisel veritabanı ile nesneye yönelik programlama (OOP) arasında bir köprü görevi gören araçtır. Bu köprü, ilişkisel veritabanındaki bilgileri yöneterek nesne modelleri için kullanılan bir yapıdır.

Veritabanı tabanı ile haberleşmesi için DbContext kullanılmıştır. DbContext veritabanında karşılık gelen obje yapısıdır. İçinde tablo yapısında karşılık gelen DbSet objelerini bulundurur. DbContext kullanarak tablo ve view yapılarına erişilmiştir. DbSet yapısını kullanarak tablo üzerinde CRUD işlemlerini gerçekleşmiştir.

Repository Design Pattern yapısı kullanılmıştır. Repository temel olarak veritabanı sorgulama işlemlerinin bir merkezden yapılmasını sağlar. İş katmanına bu işlerin taşınmasını önler. Bu şekilde sorgu ve kod tekrarına engel olmuş olur. Yani asıl amaç veri işlem ve sorgulamaların tekrarlardan kaçınılarak merkezi bir yapıya çekilmesidir. Bu sayede veritabanı işlemlerimizi tekrar ve tekrar iş katmanı içinde yazmak durumunda kalmaktan uzak durulmuştur.

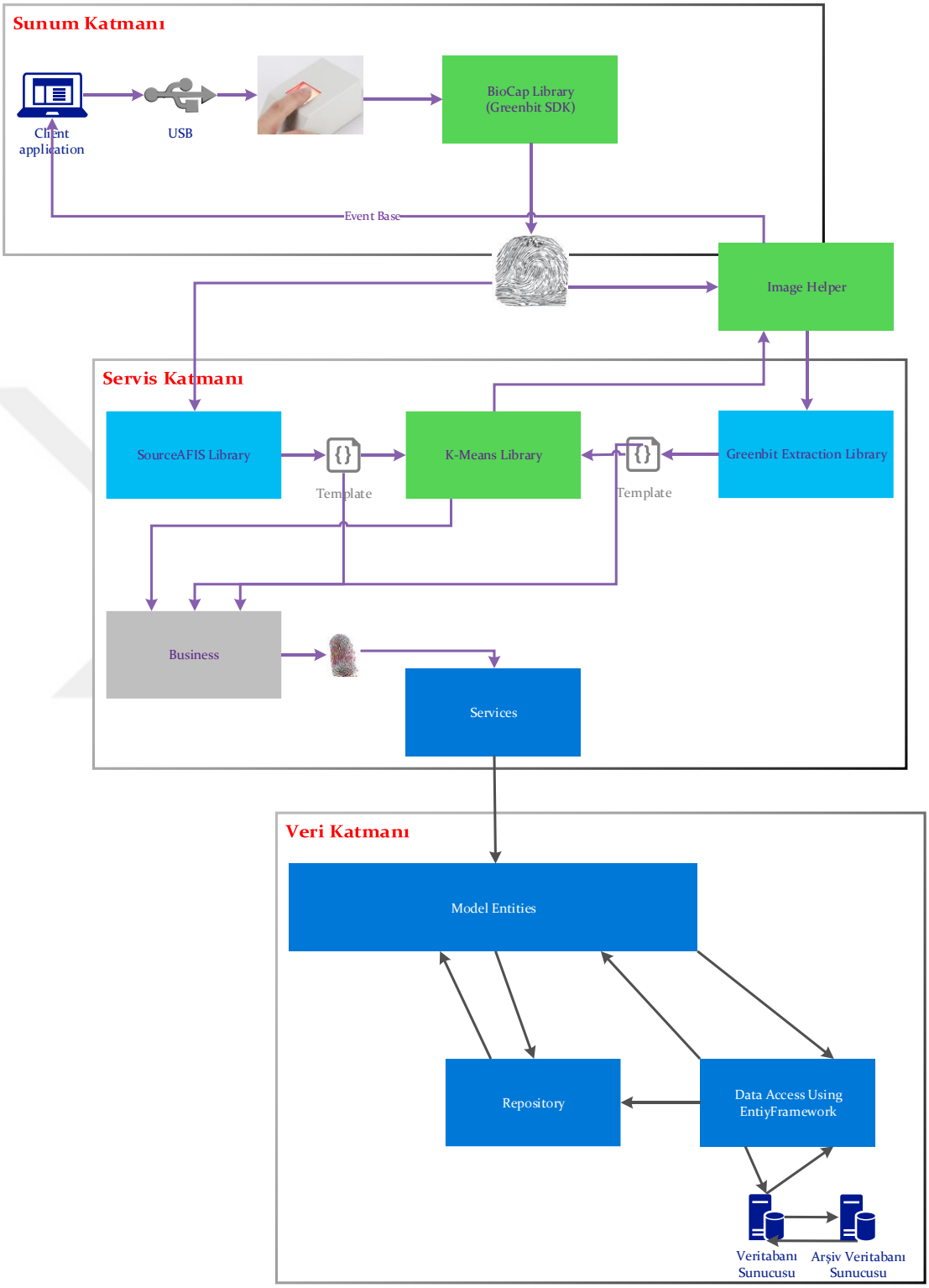
Repository tasarım kalıbının ilk ve en önemli amacı budur, bunun yanında yukarıdaki tanıma ek olarak repository tasarım kalıbı, programda asıl işi yapan bölümler ile veriye erişen bölümlerin birbirinden soyutlanması sağlanmıştır. Yani veri katmanı

ve bu katmanı kullanan iş katmanı arasında bir arabirim olarak yer alır ve bu iki katman arasında soyutlama görevi de üstelenmiştir.

Yazılım mimarisi şekil 5.14. görüldüğü gibi 3 katmandan oluşmaktadır. Bunlar sunum katmanı, servis katmanı ve veri katmanıdır.

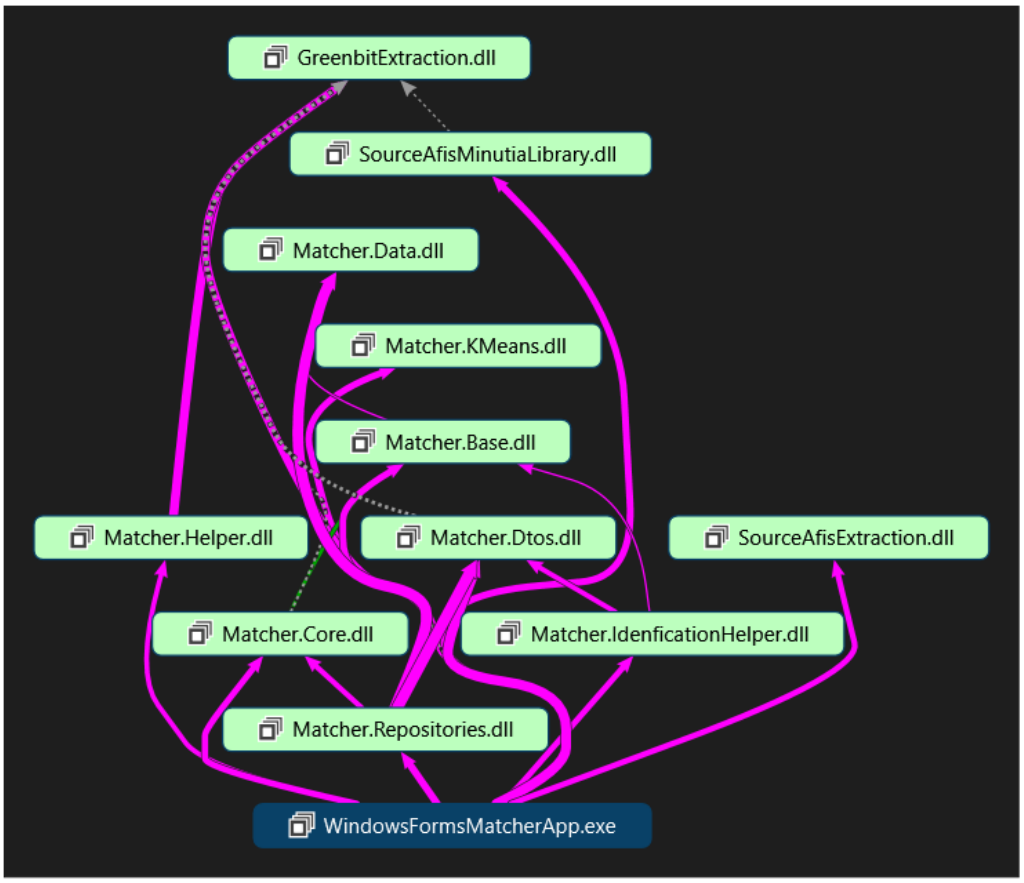
Diyagram yukarıdan aşağıya göre client (terminal) uygulaması USB ile sensöre bağlanır. Terminal de dijital sensörün driver, yüklü olmalıdır. Sensör ile haberleşmesi BioCAP kütüphanesi ile bağlanır. Sensörle ile ilgili işlemleri BioCAP yapar. Sensörden parmak izi görüntüsü gelir. Parmak izi görüntüsünü Greenbit Exreaction Library kullanacaksa Image Helper kütüphanesi ile parmak izi görüntüsü raw image çevrilir. Grenbit Extraction Library ile parmak izi görüntüsünün kalite değeri hesaplanır ve minutiaeları elde edilir. SourceAFIS library ile minutiaeları elde edilebilir. Böylece iki algoritmaya göre minutiaeları karşılaştırılmış olur. Elde edilen minutiaeları kümelemek için K-Measn Library ile kümeleme yapılarak bölütlenir. Kümeleme işlemini kullanıcıya göstermek için parmak izi görüntüsünü elde etmek için Image Helper kütüphanesi kullanarak event base olarak ara yüzde kullanıcı bilgilendirilir. Kümeleme işlemi başarılı şekilde tamamlandıktan sonra elde edilen minutiaeları business katmanında modellere çevrilir. Modeller services katmanı ile repository katmanından hazırlanan metotları kullanarak Entityframework yardımıyla veritabanına haberleşir. Böylece veritabanına veriler eklenmiş olur. Bu işlemler çift yönlü yapılabilmektedir.

Bu mimari ile veritabanı izole edilmiş olur. İşlemler ayrı ayrı kütüphanelerde gerçekleşmiştir.



Şekil 5.13. Yazılım mimarisi

Şekil.5.14. kod diyagramında görüldüğü üzere uygulamanın kullandığı .dll kütüphaneleri ve ortak yapıları gösterilmiştir

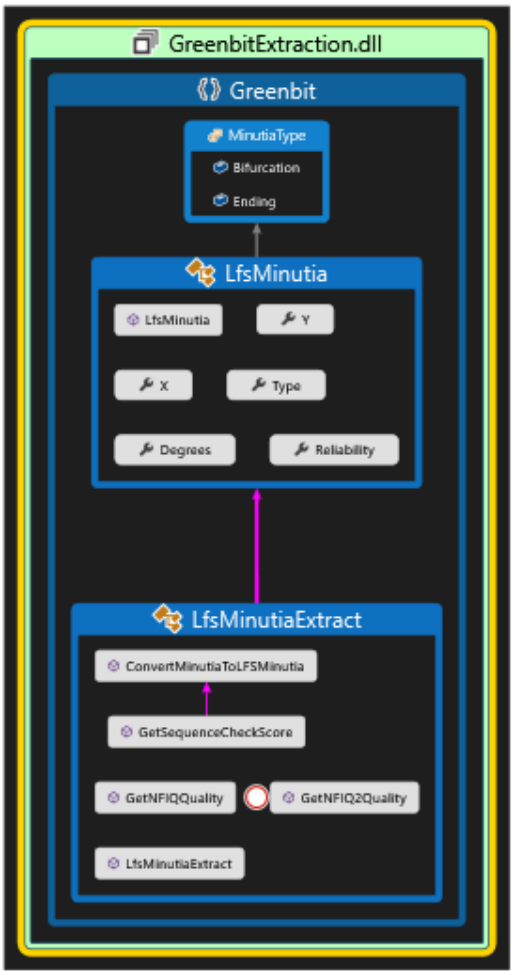


Şekil 5.14. Kod diyagramı

Kod diyagramında kullanılan ve oluşturulan kütüphaneler ayrı ayrı açıklanacaktır.

5.9.1. GreenbitExraction Kütüphanesi

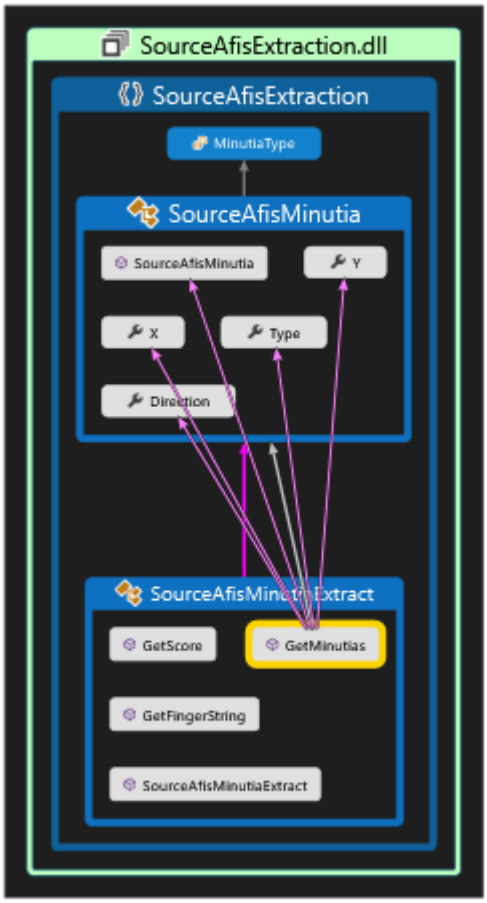
Bu kütüphane, parmak izini Greenbit kütüphanesinde minutiaeları çıkarmak için kullanır. Ayrıca parmak izinin NFIQ ve NFIQ2 hesaplaması bu kütüphanede gerçekleşir. Minutiaeların tip ve değerleri bu kütüphane yardımıyla elde edilir. Kullanılan sınıflar şekil 5.15. ‘de görülmektedir.



Şekil 5.15. GreenbitExraction kütüphanesi

5.9.2. SourceAfisExtraction Kütüphanesi

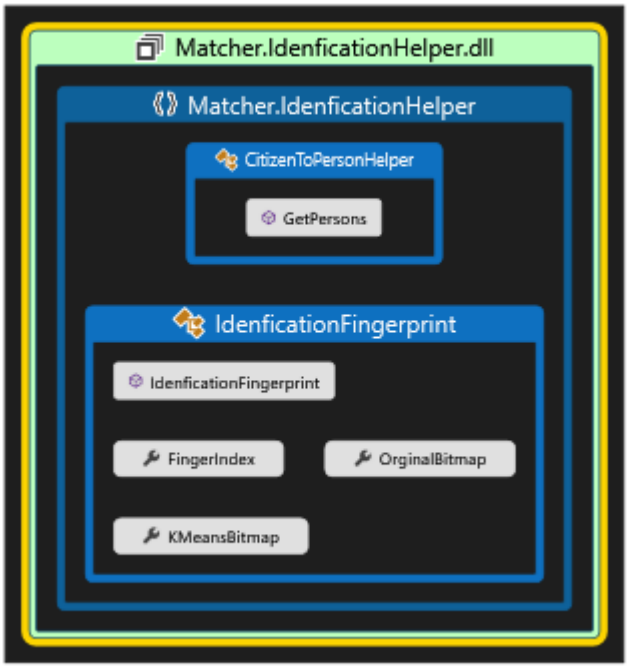
Bu kütüphane, parmak izini sourceAFIS kütüphanesini kullanarak minutiaeları çıkarmak için kullanır. Minutiaeların tip ve değerleri bu kütüphane yardımıyla elde edilir. Bu kütüphane yardımıyla parmak izlerini siteme tanıtma, skor üretme ve teşhis işlemi bu kütüphane yardımıyla gerçekleşir. Kullanılan sınıflar şekil 5.16. ‘da görülmektedir.



Şekil 5.16. SourceAfisExtraction kütüphanesi

5.9.3. Matcher.IdenficationHelper Kütüphanesi

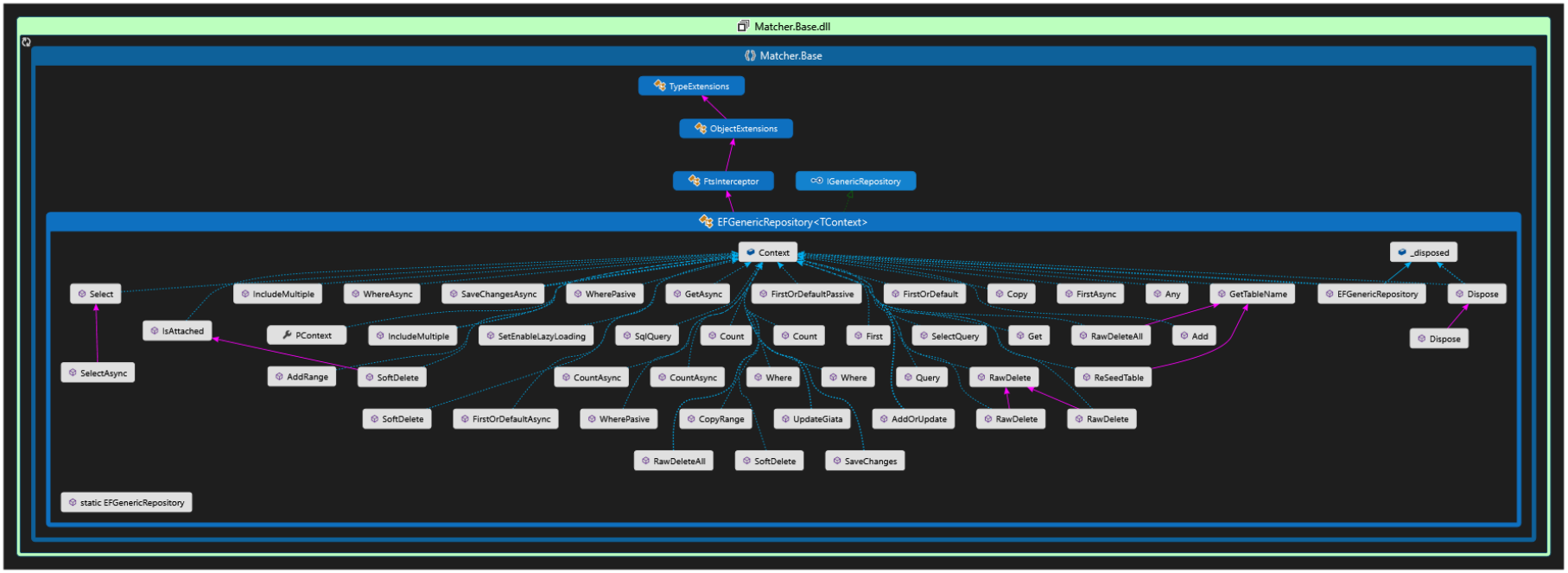
Bu kütüphane, veritabanından çekilen parmak izlerini teşhis, skor işlemleri için yardımcı olan bir kütüphanedir.



Şekil 5.17. Matcher.IdenficationHelper kütüphanesi

5.9.4. MatcherBase Kütüphanesi

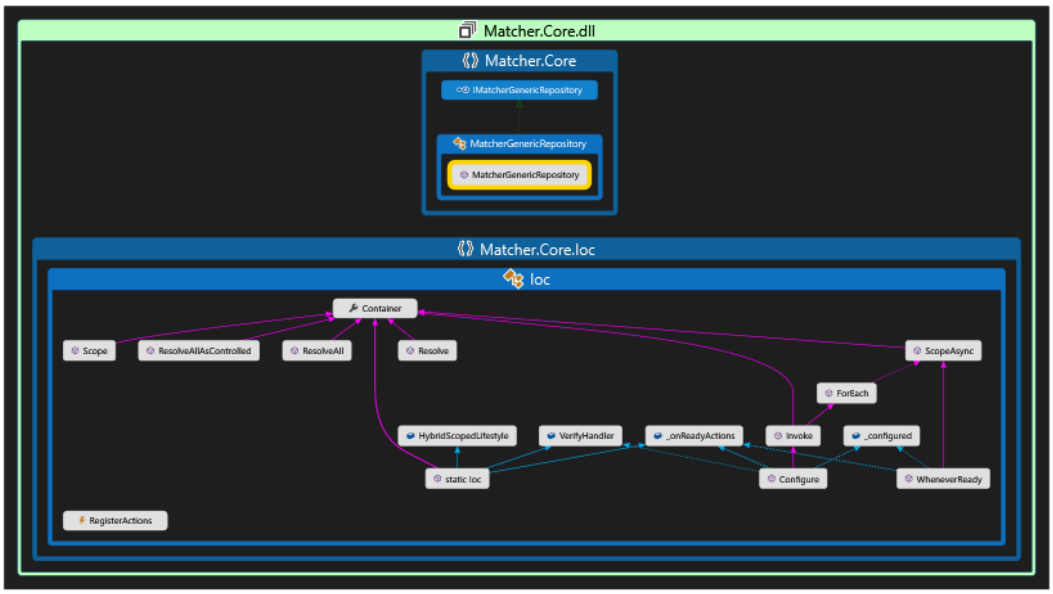
Veritabanı ile haberleşmesini sağlayan Repository Design Pattern yapısına uygun metotları bulunduran, metotlarla veritabanına sorgu atan kütüphanedir. Şekil 5.18 görülmektedir.



Şekil 5.18. MatcherBase Kütüphanesi

5.9.5. Matcher.Core Kütüphanesi

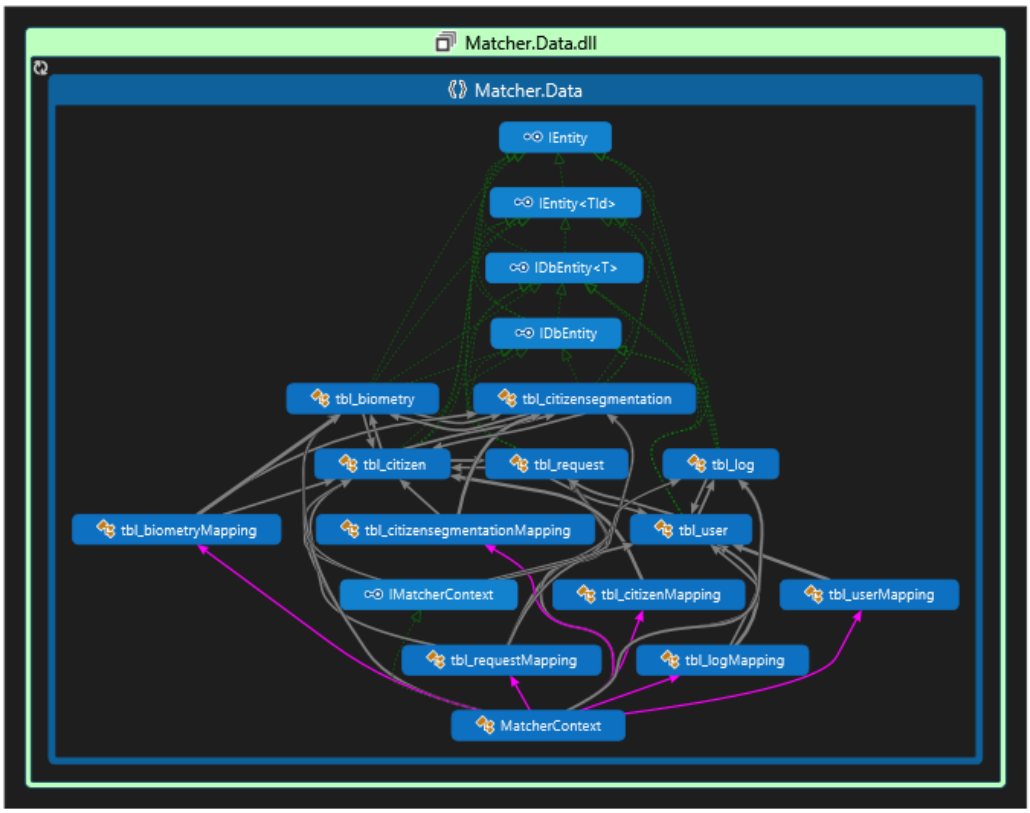
Bu kütüphane Dependency Injection gerçekleştirmeyi sağlar. Dependency Injection, Inversion of Control adını uygulama akışının değiştirme yönteminin özelleşmiş bir halidir. Dependency Injection ile uygulamanın çalışacağı ve bağımlı olduğu akışları dışarıdan enjekte ederek uygulama akışını dinamik olarak değiştirilmektedir. Böylece uygulamanın genişletilebilmesi ileri zamanlarda gelebilecek olan yeni geliştirmelerde uygulama en az oranda etkilenmesini mümkün kılınmaktadır. Şekil 5.19' da görülmektedir.



Şekil 5.19. Matcher.Core kütüphanesi

5.9.6. Matcher.Data Kütüphanesi

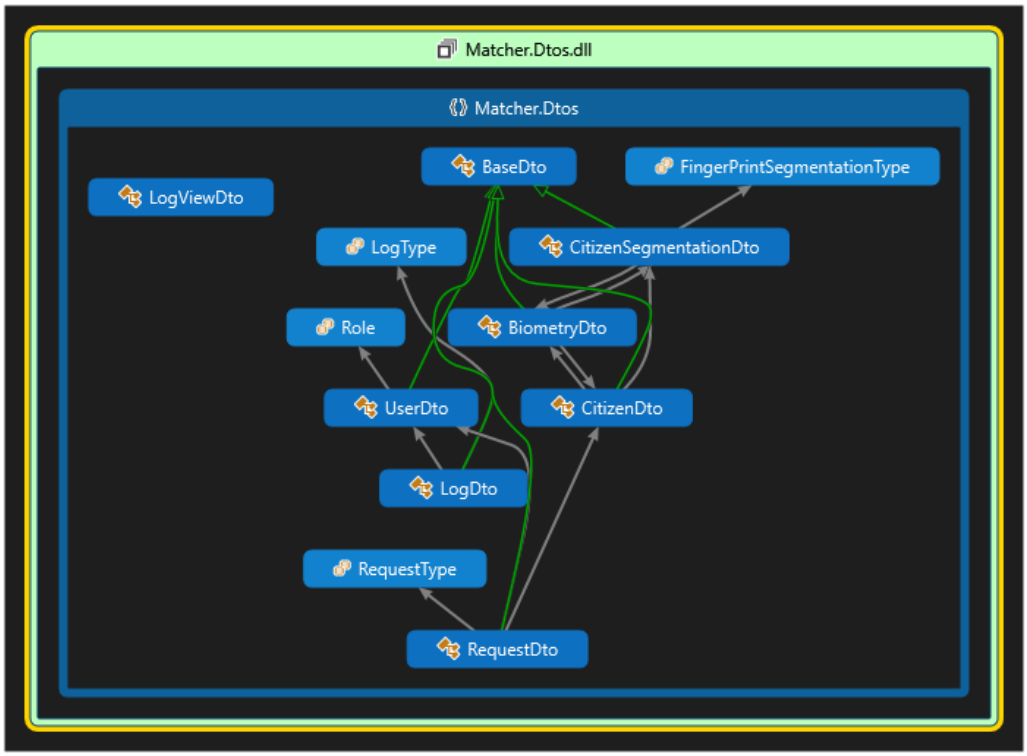
Bu kütüphene, EntityFramework kütüphanesini kullanarak veritabanına eirişir. Veritabanındaki tablo, view ve storedprocedure alır. Veritabanına ilk haberleşen kısım burasıdır. Şekil 5.20’ de görülmektedir.



Şekil 5.20. Matcher.Data kütüphanesi

5.9.7. Matcher.Dtos Kütüphanesi

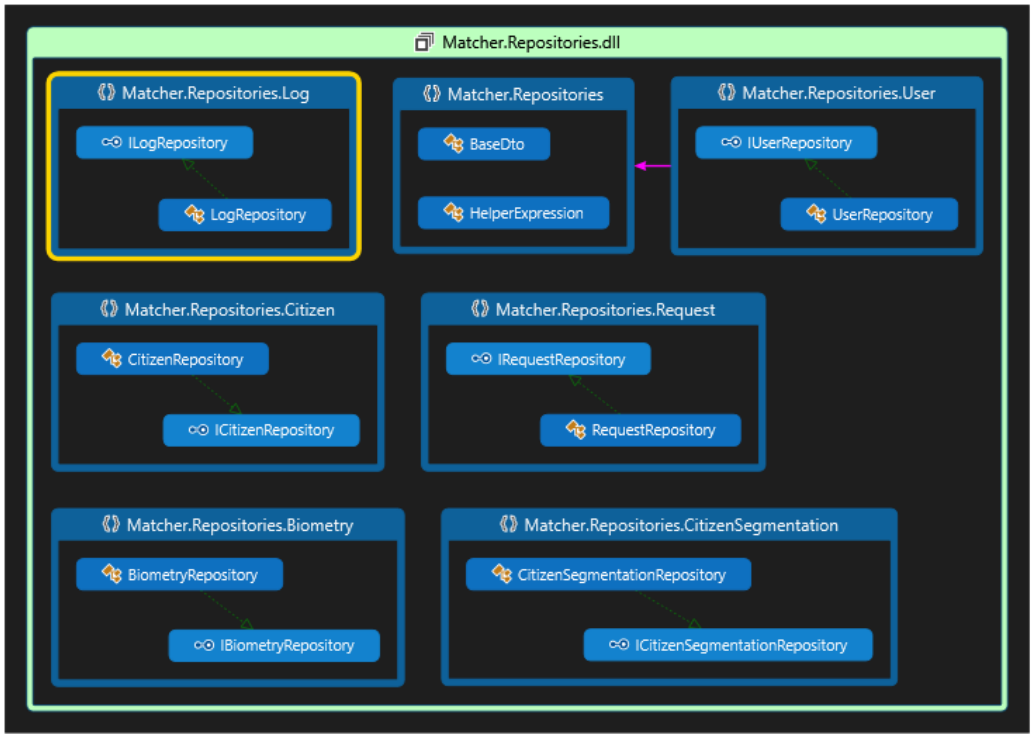
Bu kütüphande DTO (Data Transfer Object) sınıfları bulunur. Veritabanı üzerinde yapılacak işlemleri yüklenen sınıflardır. DTO içine veritabanı işlemleri yazıp programda karışıklık ve tekrar tekrar kullanmanın (code reuse) önünü açmak için kullanır. Şekil.5.21' de görülmüştür.



Şekil 5.21. Mathcer.Dtos kütüphanesi

5.9.8. Matcher.Repositories Kütüphanesi

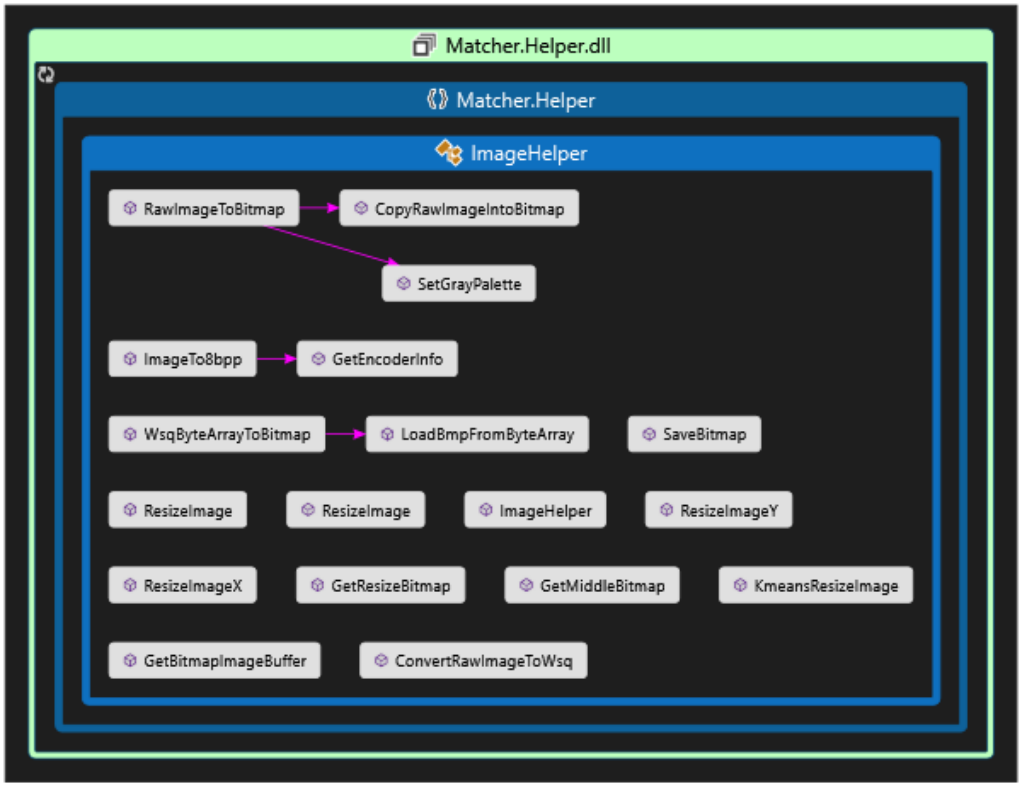
Bu kütüphane ile veri merkezli uygulamalarda veriye erişimin ve yönetimin tek noktaya indirilmesini sağlayan kütüphanedir. Böylece kod tekrarlarından kaçınılmıştır. Hata yakalama kolaylaşmıştır. Kod yazımını ve okunuşu kolaylaşmıştır. Test edilebilir bir yapı oluşturulmuştur. Şekil.5.22' de görüldüğü üzere her metot bağımsızdır.



Şekil 5.22. Matcher.repositories kütüphanesi

5.9.9. Matcher.Helper Kütüphanesi

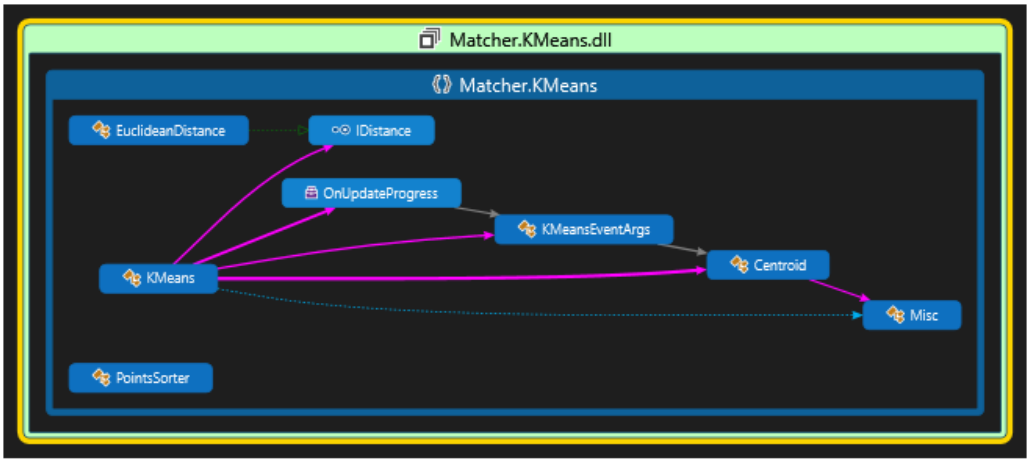
Bu kütüphane ile parmak izi görüntüsü üzerine iyileştirmeler yapılmasını sağlayan kütüphanedir. Şekil.5.23' de görüldüğü üzere raw veriden bitmap çevrimleri görülmektedir.



Şekil 5.23. Matcher.Helper kütüphanesi

5.9.10. Mather.KMeans Kütüphanesi

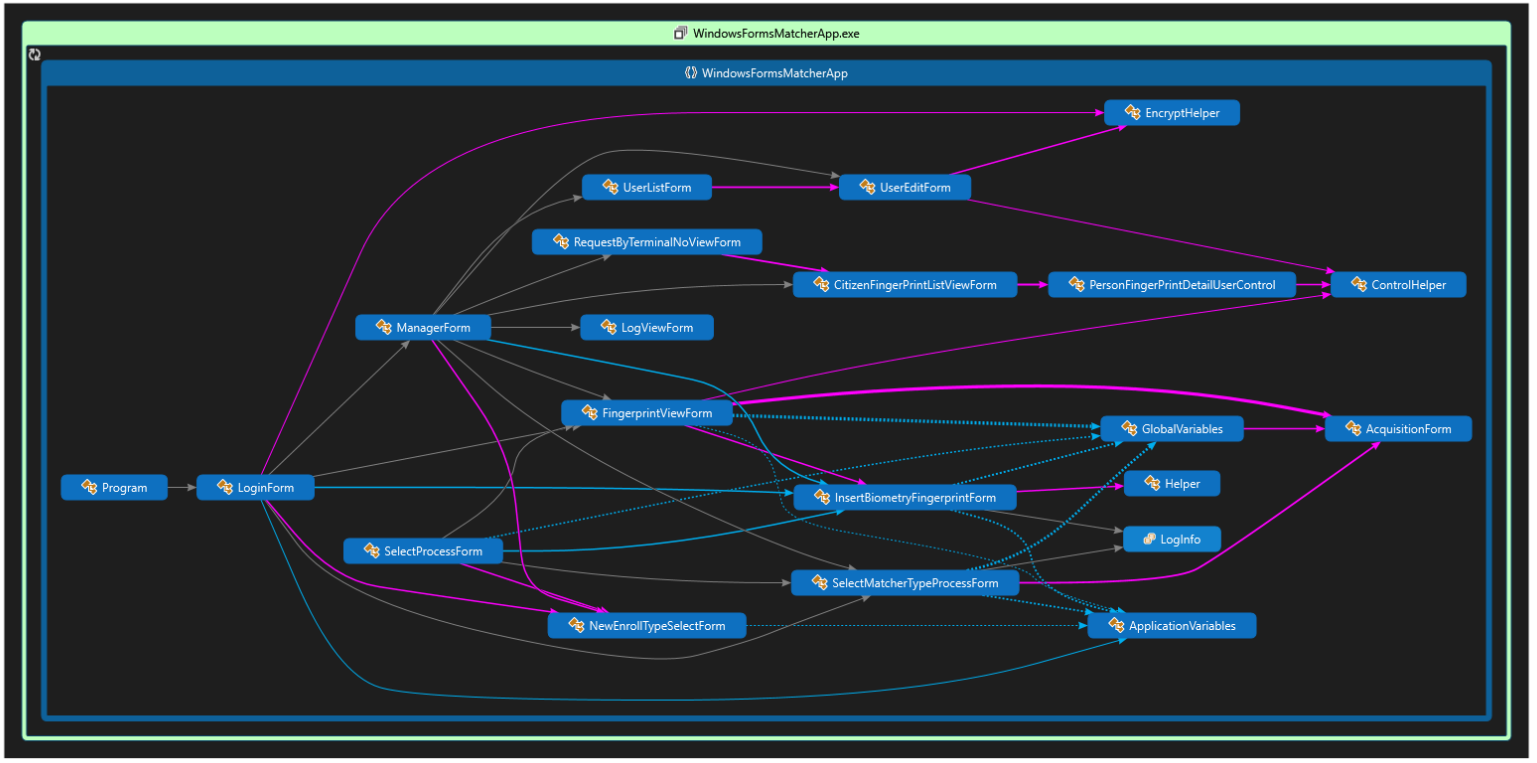
Bu kütüphane ile parmak izi görüntüsünde çıkarılan minutiaeları kümelemek için hesaplama yapılan kütüphanedir.



Şekil 5.24. Matcher.KMeans kütüphanesi

5.9.11. Uygulama Ekranların Kod Tasarımı

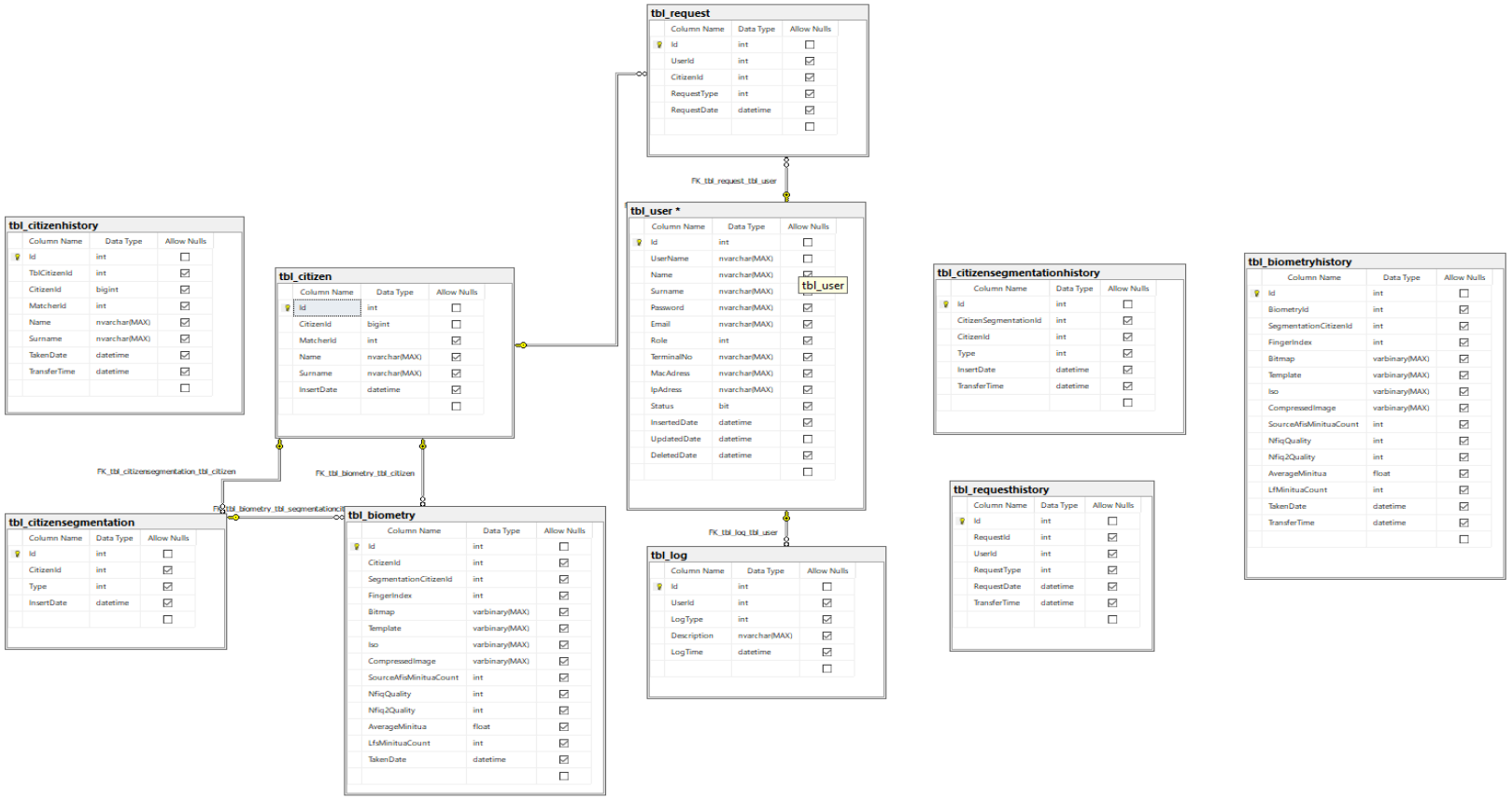
Uygulamadaki ekranların bulunduğu, ekranların birbiriyle olan ilişkilerini gösterir. Şekil 5.25 görüldüğü üzere uygulama login ekranında başlayıp dallanır.



Şekil 5.25. Uygulama ekran kod tasarımı

5.10. Veritabanı Tasarımı

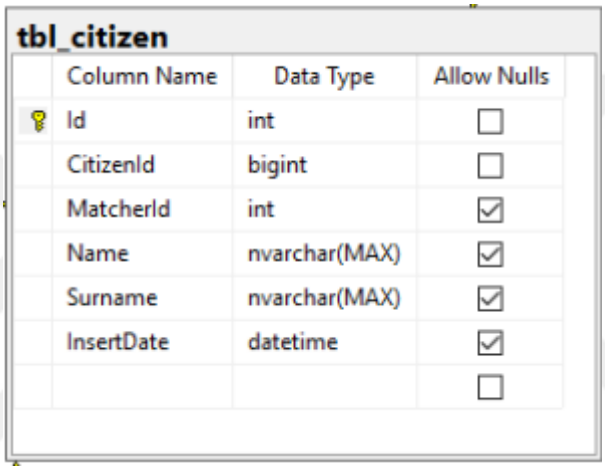
Bu bölümde, önerilen sistemin veritabanı tasarımındaki tablolar ayrı ayrı başlıklarda açıklanacaktır. Tasarımın bütün olarak tasarımı Şekil 5.10.'da tasarım gösterilmiştir. Veritabanında; tbl_citizen, tbl_citizensegmentation, tbl_biometry, tb_user, tbl_request ve tbl_log tablolarından oluşmaktadır. Kişiye ait parmak izi verisi aramada tbl_segmentation göre bölütlenmiş ya da bölütlenmemiş görüntüsü getirilebilmektedir.



Şekil 5.26 Veritabanı tasarımı

5.10.1. tbl_citizen Tablosu

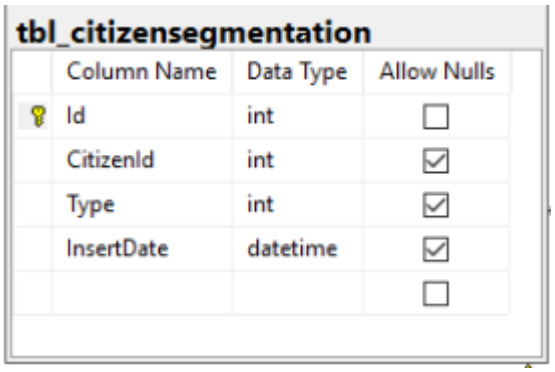
Kişi bilgilerin tutulduğu tablodur. Bu tabloda kişi adı, soyadı, sisteme kayıt olma tarihi ve sistemde unique olan matcherid bilgileri tutulmaktadır. Bu tabloda id sütunu primary key olup otomatik artmaktadır. Şekil 5.27.'de tablo bilgileri gösterilmiştir.



Şekil 5.27. tbl_citizen tablosu

5.10.2. tbl_citizensegmentation Tablosu

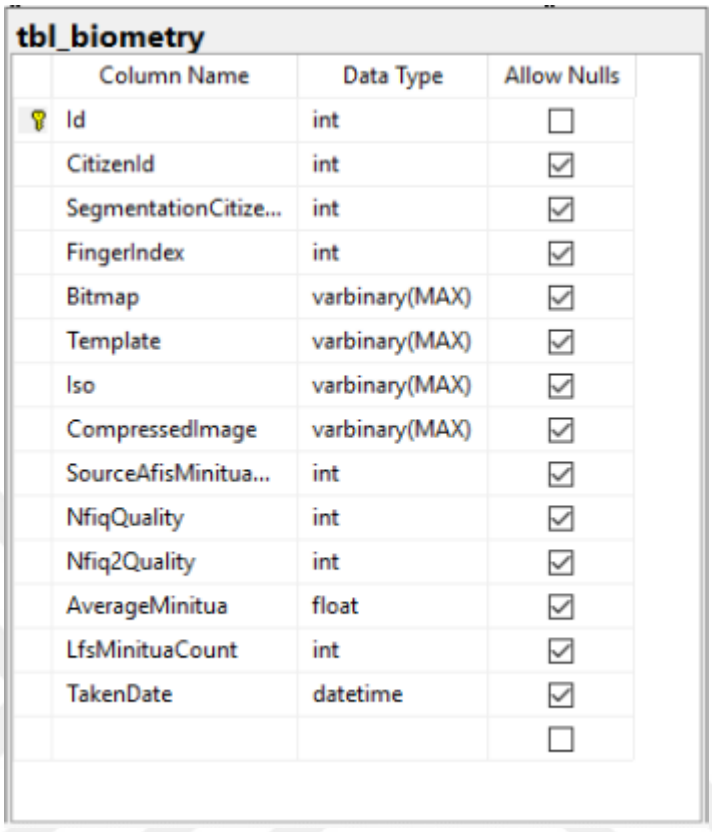
Kişinin parmak izi bölüt bilgilerinin tutulduğu tablodur. Kişi id ile ilişkilidir. Parmak izinin gerçek görüntüsü ile bölütlenmiş bilgisinin ayrımı type sütunu ile yapılmaktadır. Type alanı 0 ise parmak izi görüntüsünün kendisi, 1 ise K-means göre kümelenmiş görüntüdür. Şekil 5.28.'de tablo bilgileri gösterilmiştir.



Şekil 5.28. tbl_citizensegmentation tablosu

5.10.3. tbl_biometry Tablosu

Kişiye ait biyometrilerin tutulduğu tablodur. Citizen tablosundaki id ile citizensegmentation tablosundaki id ile ilişkilidir. Bu tabloda parmak indeksi (1-10), parmak izi görüntüsü, SourceAFIS kütüphanesinin ürettiği template parmak izi verisi, ISO/IEC 19794-8:2006 formatındaki parmak izi verisi, SourceAFIS kütüphanesinin çıkardığı minutiaeların sayısı, LFS kütüphanesinin çıkardığı minutiae sayısı, parmak izinin minutiaeların ortalama kalitesi, parmak izinin NFIQ (1-5) ve NFIQ2 (0-100) kalite değeri ve parmak izinin alınma tarihi tutulur. Şekil 5.29. ‘de tablo bilgileri gösterilmiştir.



tbl_biometry

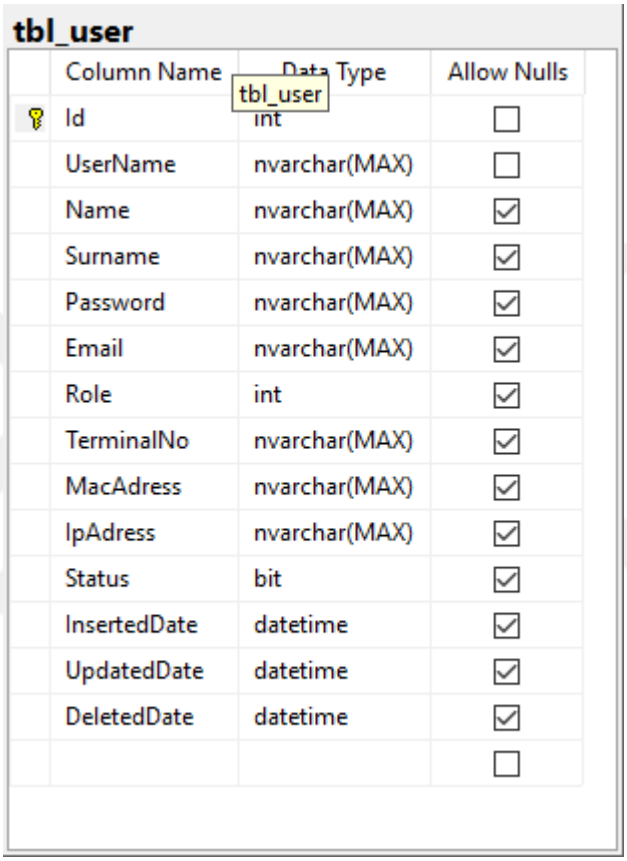
Column Name	Data Type	Allow Nulls
Id	int	☐
CitizenId	int	☑
SegmentationCitize...	int	☑
FingerIndex	int	☑
Bitmap	varbinary(MAX)	☑
Template	varbinary(MAX)	☑
Iso	varbinary(MAX)	☑
CompressedImage	varbinary(MAX)	☑
SourceAfisMinitua...	int	☑
NfiqQuality	int	☑
Nfiq2Quality	int	☑
AverageMinitua	float	☑
LfsMinituaCount	int	☑
TakenDate	datetime	☑
		☐

Şekil 5.29. tbl_biometry tablosu

5.10.4. tbl_user Tablosu

Sisteme giriş yapacak olan kullanıcıların tanımlı olduğu tablodur. Bu tabloda kullanıcı bilgileri olarak sisteme giriş için kullanıcı adı, adı, soyadı, şifresi, mail adresi ve rolü tutulmaktadır. Rol olarak sistem yöneticisi, yeni kayıt ve teşhis rolüdür. Ayrıca kullanıcılar işlemlerini terminallerde yapmaktadır. Bu yüzden kullanıcılarla terminal bilgileri ilişkilendirilmiştir. Terminal bilgileri olarak; terminal adı, ip adresi ve MAC adresi tutulmaktadır. Kullanıcılar pasif olduklarında durumu statü alanında

tutulmaktadır. Kullanıcıların eklenme tarihi, güncellenme tarihi ve silme tarihi tutulmaktadır. Kullanıcı silindiğinde veritabanından silinmez. Durumu pasife çekilir. Şekil 5.30.' da tablo bilgileri gösterilmiştir.

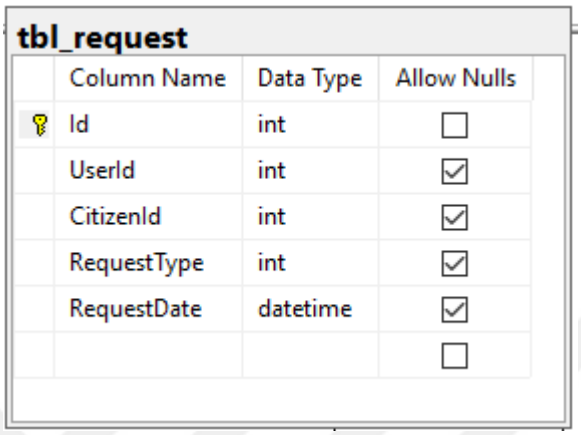


Şekil 5.30. tbl_user tablosu

5.10.5. tbl_request Tablosu

İşlemlerin tutulduğu tablodur. Bu işlemler yeni kayıt ve teşhis işlemleridir. İşlemler hangi kişi için yapıldığı ve hangi kullanıcının yaptığı bilgisi yer almaktadır. Bunun için

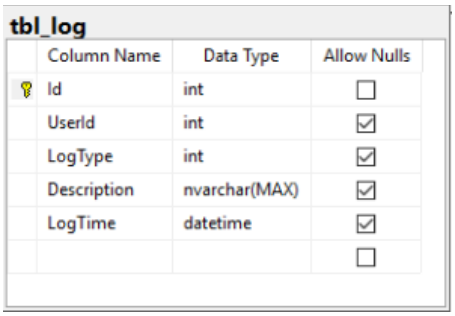
tbl_user ve tbl_citizen ile ilişkilidir. Her işlem yapıldığında bu tabloya kayıt tarihiyle birlikte kayıt atılmaktadır. Şekil 5.31. ‘de tablo bilgileri yer almaktadır.



Şekil 5.31. tbl_request tablosu

5.10.6. tbl_log Tablosu

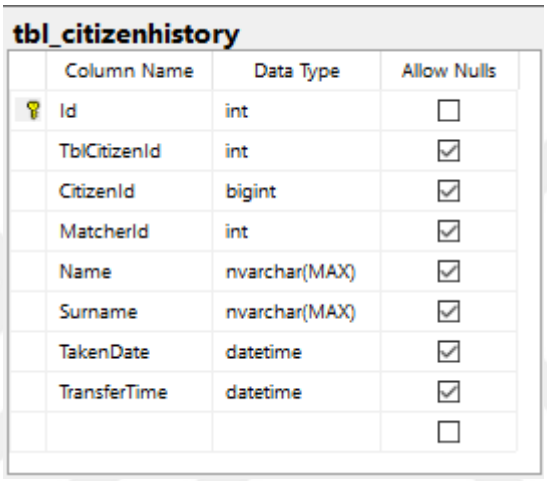
Log tablosunda yapılan işlemler tutulmaktadır. Diğer tablolarla ilişkisi yoktur. Type alanında işlemin tipi, description alanında açıklama bilgisi tutulmaktadır. Logtime ise işlem zamanı tutulmaktadır. Sistemdeki kullanıcılarla ilişkili olup kullanıcı bazında işlemler tutulmaktadır. Şekil 5.32. ‘de tablo bilgileri gösterilmiştir.



Şekil 5.32. tbl_log tablosu

5.10.7. tbl_citizenhistory Tablosu

Arşivlenen kişi bilgilerin tutulduğu tablodur. Şekil 5.33.'de tablo bilgileri gösterilmiştir. TblcitizenId alanı tbl_citizen tablosundan silinen kayıtın id' sidir. Böylece arşivden çıkarılıp işlem yapılmasına olanak sağlar.



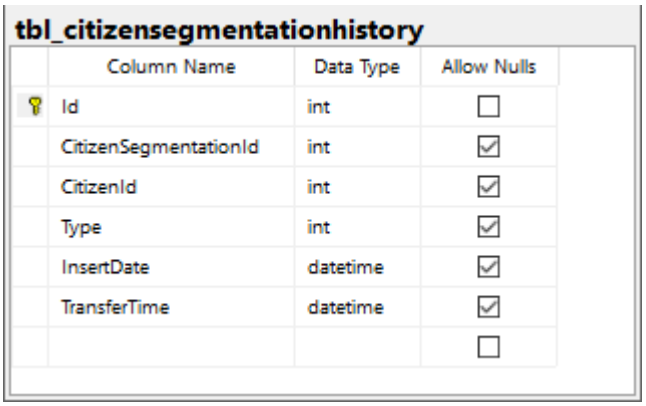
tbl_citizenhistory

Column Name	Data Type	Allow Nulls
Id	int	☐
TblCitizenId	int	☑
CitizenId	bigint	☑
MatcherId	int	☑
Name	nvarchar(MAX)	☑
Surname	nvarchar(MAX)	☑
TakenDate	datetime	☑
TransferTime	datetime	☑
		☐

Şekil 5.33. tbl_citizenhistory tablosu

5.10.8. tbl_citizensegmentationhistory Tablosu

Arşivlenen kişinin parmak izi bölüt bilgilerinin tutulduğu tablodur. Citizensegmantationid alanı tbl_citizensegmentation tablosundan silinen kayıtın id' sidir. Böylece arşivden çıkarılıp işlem yapılmasına olanak sağlar. Şekil 5.34.'de tablo bilgileri gösterilmiştir.



Şekil 5.34. tbl_citizensegmentationhistory tablosu

5.10.9. tbl_biometryhistory Tablosu

Arşivlenen kişinin parmak izi bilgilerinin tutulduğu tablodur. Biometryid alanı tbl_biometry tablosundan silinen kayıtın id' sidir. Böylece arşivden çıkarılıp işlem yapılmasına olanak sağlar. Şekil 5.35.'de tablo bilgileri gösterilmiştir.

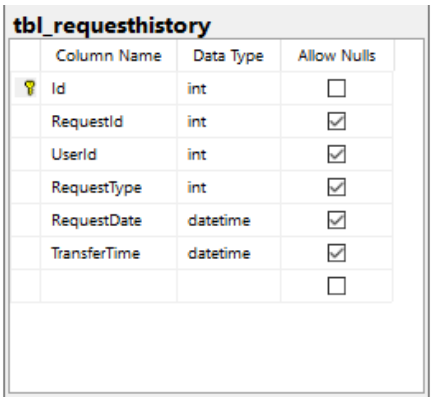
tbl_biometryhistory

Column Name	Data Type	Allow Nulls
Id	int	☐
BiometryId	int	☑
SegmentationCitizenId	int	☑
FingerIndex	int	☑
Bitmap	varbinary(MAX)	☑
Template	varbinary(MAX)	☑
Iso	varbinary(MAX)	☑
CompressedImage	varbinary(MAX)	☑
SourceAfisMinituaCount	int	☑
NfiqQuality	int	☑
Nfiq2Quality	int	☑
AverageMinitua	float	☑
LfMinituaCount	int	☑
TakenDate	datetime	☑
TransferTime	datetime	☑
		☐

Şekil 5.35. tbl_biometryhistory tablosu

5.10.10. tbl_requesthistory Tablosu

Arşivlenen kiişlem bilgilerinin tutulduğu tablodur. RequestId alanı tbl_request tablosundan silinen kayıtın id' sidir. Böylece arşivden çıkarılıp işlem yapılmasına olanak sağlar. Şekil 5.36.'de tablo bilgileri gösterilmiştir.



Şekil 5.36. tbl_requesthistory tablosu

6. SİSTEMİN GERÇEKLENMESİ

6.1. Sistem Mimarisi

Sistem mimarisi bulut üzerine kurulmuştur. Mimariyi yukarıdan aşağıya doğru incelenirse istemci (terminaller), host adresini girerek 80 port ile DNS yardımıyla web IIS serverlere bağlanırlar. Böylece IP adresi girmeden uygulamayı indirebilirler. DNS arkada web server aracılığıyla isteği load blancer iletir. Sunucuların önünde bulunan load blancer ile sistemdeki yük paylaşılır. Bu yük paylaşımı bulut servisleri ile web sunuculara paylaşılır. Web sunucuları güvenlik protokol yani sertifika (https) ile firewallden geçip istekleri parmak izi sunucusuna iletir. Parmak izi sunucusu isteğin türüne göre istekleri kayıt sunucusu ya da eşleştirme sunucusuna yönlendirir. Yeni kayıt master sunucu aracılığıyla hangi sunucunun yükü az ise yükü node sunucusuna paylaştırır. Böylece parmak izi görüntüleri node sunucuya aktarılmış olur. Parmak izleri kayıt edilmesi için node sunucusu orijinal parmak izi görüntü ve kümelenmiş segmente edilen parmak izi görüntülerini veritabanı sunusuna aktarır. Veritabanı sunucusu verileri kaydederek master sunucusuna bilgilendirir. Master sunucusu, node sunucularına parmak izinin verisinin RAM hafızada tutulmasını sağlamak için istek oluşturur. Veritabanı sunucusundan kaydedilen parmak izi verisi çekilir ve RAM hafızaya yüklenmiş olur.

Teşhis işlemi için (istemci) terminaller parmak izi verisini alır. Parmak izi verisi büyük olduğu için bant genişliği yüksek olmalıdır. Parmak izi verisi web server sunucunda istek oluşturur. İsteğin içerisinde parmak izi verisi bulunmaktadır. Parmak izi verisini load blancer yardımıyla işlemi buluttaki web sunuculara iletir. Web sunucuları firewall geçerek isteği parmak izi sunucusuna aktarır. İstek teşhis olduğu için isteği master sunucusuna aktarır. Master sunucusu isteği paralel olarak node sunucularda teşhis işlemi yapmasını ister. Node sunucuları paralel olarak karşılaştırma işlemi yapar. Karşılaştırma sonucunu master sunucusunu gönderir. Master sunucusu sonucu güvenlik duvarıyla buluttaki sunuculara gönderir. Buluttaki web serverler sonucu load blancere gönderir. Load blancer web sunusu yardımıyla cevabı terminale gönderir. Bu teşhis işlemi yapılırken yeni kayıta göre daha fazla sürede yapılmaktadır.

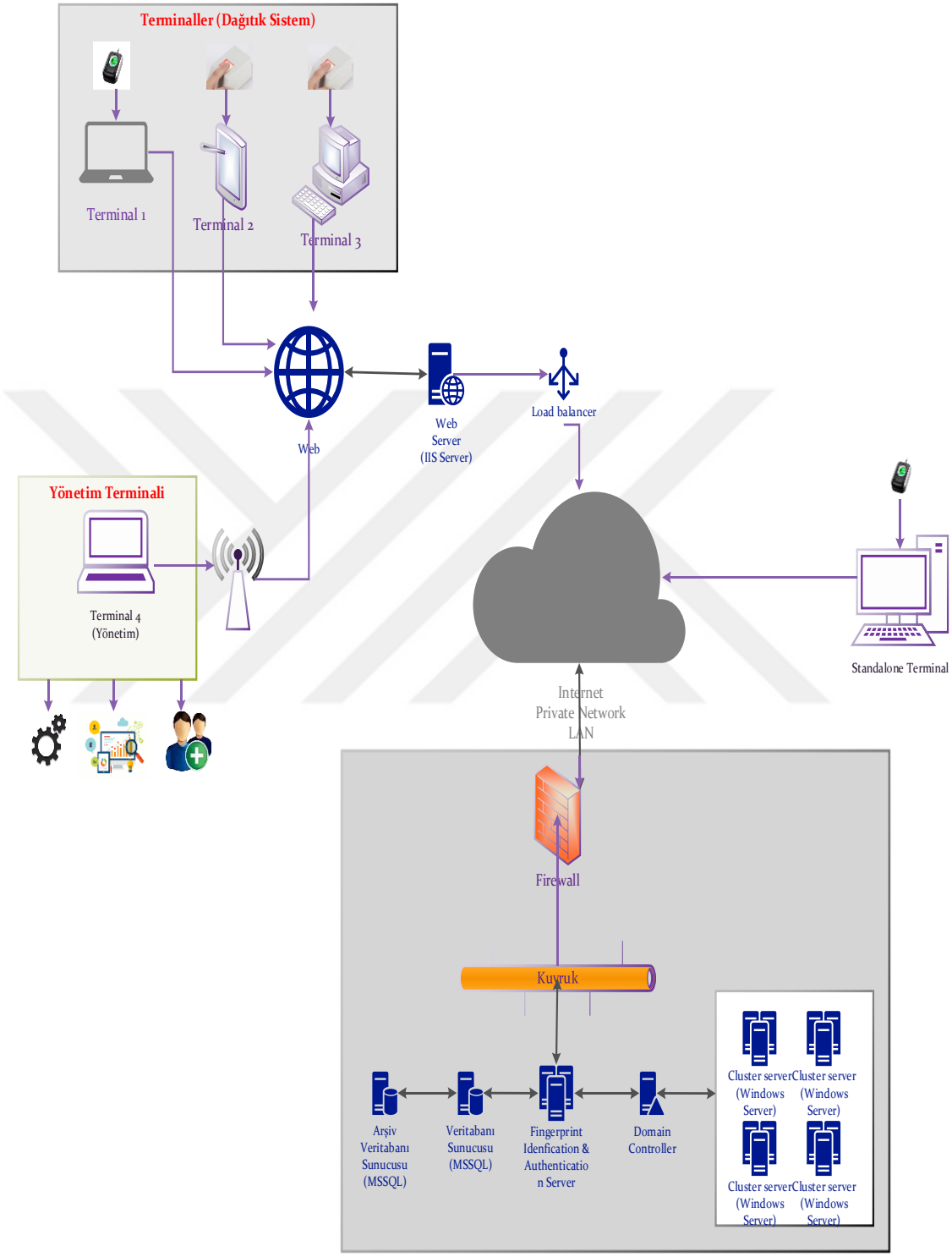
Teşhis işlemi sırasında yeni kayıt istekleri geldiğinde istekleri bekletir. Bu işlem için kuyruk yapısı kullanılmıştır. Kuyruk yapısı FIFO mantığı ile istekleri sunuculara aktarmaktadır. Görüldüğü gibi işlemler dağıtık sistemde yapılmıştır. Dağıtık sistemlerde sunuculardan biri hasar çevrimdışı durumuna düştüğünde otomatik olarak yükü diğer sunuculara aktarır. Node sunucusu aktif olduğunda verilerin yüklenmesi zaman alacaktır. Çünkü veriler veritabanından çekilip RAM taşınması gerekir. Bu bir risk oluşturmaktadır.

Veritabanı sunusunda verileri kaydeder. Kullanılmayan verileri ve belli bir zaman geçtikten sonra verileri arşivleyerek arşivleme sunucusuna aktarır. Böylece sunucu üzerinde yük paylaşılmış olur. Eski verilere ihtiyaç duyulduğunda arşiv veritabanından verileri getirerek veritabanı sunucusun aktarır.

Standolane sistemlerde veriler kendi terminal içerisinde bulunabilir. Veriler şifreli olarak terminalde saklanabilir. Çevrimdışı durumlarda veriler terminalde olduğundan teşhis işlemi yapılır. Yeni kayıt yapıldığında veriler kendi localde şifrelenerek saklanabilir. Çevrimiçi durumuna düştüğünde veriler wifi ya da network kablosuyla sisteme aktarılır. Yönetim panelinde sitem hakkında loglar, sunucu durumları, web servis, RAM ve fiziksel hafıza durumları takip edilir.

Sistem güvenliği konusunda Https üzerinde sertifikalı olarak haberleşir. Sistemin ağ yapısı özel ağ üzerinde kurulmuştur. Bu yüzden dışarıdan erişim yoktur. Dışarıdan erişim VPN bağlantısıyla gerçekleşir. Ama VPN bağlantısı sisteme erişme konusunda kısıtlıdır.

Sistem mimarisi diyagramı Şekil 6.1.'de açıklanmıştır.



Şekil 6.1. Sistem mimarisi diyagramı

6.1. Ölçeklenebilirlik

Sistem ölçeklenebilirlik konusunda bulut platformları sistemi host edildiği sunucuların donanımlarını birkaç dakika içerisinde artırmaya veya uygulamanızın host edildiği sunucunun yanına ek sunucular ekleyerek sistemi daha verimli ve sağlıklı çalışmasını sağlar. Ayrıca bu sunucuların load balancing (yük dağılımları) ayarlarını da otomatik yapar. Ek ayarlar yaparak yük dağılımlarını yönetmeyi sağlar. Diğer taraftan veritabanı ve master node sunucularında veriler arttığında sunucular eklenebilir. Parmak izi verisini RAM hafızada tutan sunucularda RAM miktarı yüksek olmalıdır. RAM yerine alternatif olarak SSD hafıza türleri düşünülebilir.

6.2. Uygulamanın Yazılım Özellikleri

Bu bölümde uygulamadaki bölümleri ara yüzle beraber, bölümlerdeki önemli kod parçalarına değinilmiş ve bu kodların işlevlerinden bahsedilmiştir. Uygulama kullanıcı rolüne göre modüllere ayrılmıştır. Uygulama yönetim, yeni kayıt ve teşhis modülü olmak üzere üç ana modülden oluşur.

6.2.1. Ekranlar

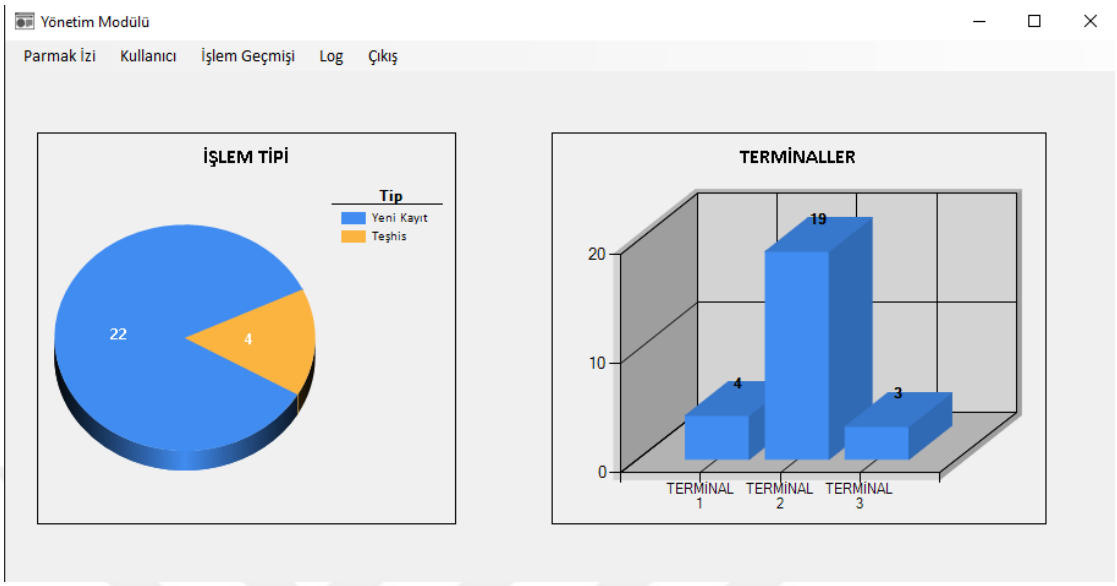
Uygulama açılırken giriş modülüyle kullanıcı giriş yapılmaktadır. Kullanıcı, kullanıcı adı ve şifresini girer. Girişe tıklar. Kullanıcı adı ve şifresi kontrol edildikten sonra rolüne göre yönlendirmeler yapılır. Kullanıcı adı ve şifresi yanlış ise kullanıcıya uyarı vermektedir. Şekil 6.2.’ de giriş modülü görülmektedir.

Şekil 6.2. Giriş modülü

6.2.1.1. Yönetim Modülü

Giriş yapan kullanıcı yönetici rolünde ise yönetici modülüne giriş yapar. Girişte sistemdeki kayıtlarla ilgili analiz gelir. Şekil 6.3' de olduğu gibi terminal ve işlem tipi olmak üzere iki tane grafik ile kayıtlar analiz edilmektedir. Her işlem yapıldıkça grafikler güncellenmektedir.

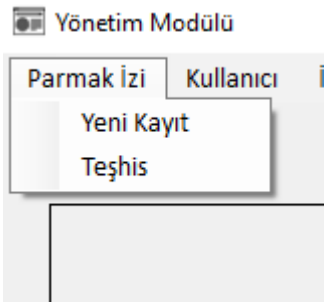
Yönetim modülünde parmak izi, kullanıcı, işlem geçmişi ve log olmak üzere 4 ana menüden oluşmaktadır. Menüdeki işlemler ayrı başlıklarda incelenecektir.



Şekil 6.3. Yönetim modülü giriş ekranı

- **Parmak İzi**

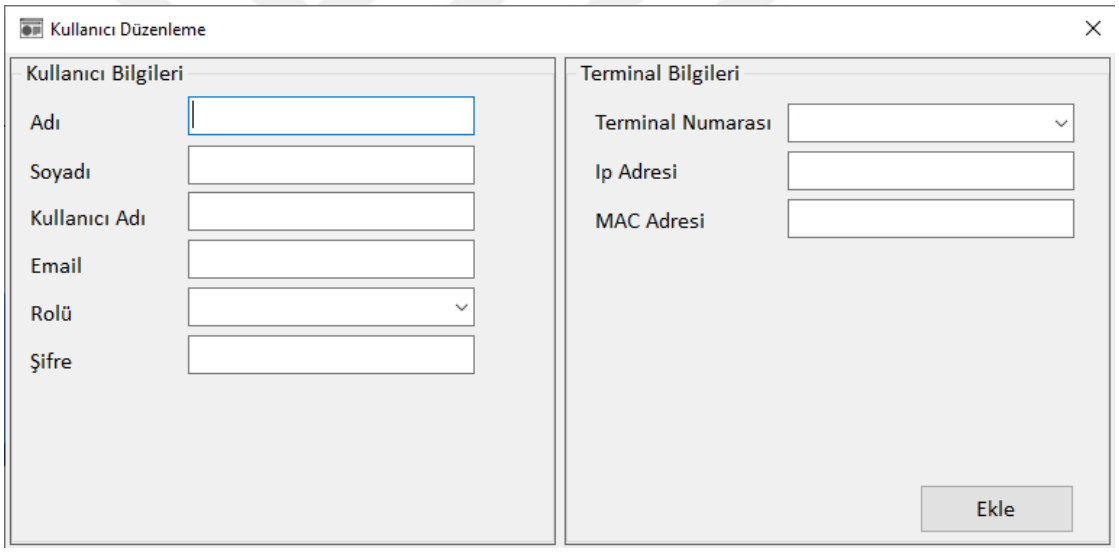
Parmak izi menüsünde yeni kayıt ve teşhis işlemleri bulunmaktadır. Bu işlemler ileride yeni kayıt ve teşhis modülü olarak anlatılacaktır. Bu menün işlevleri diğer modüllerde ortaktır.



Şekil 6.4. Parmak izi modülü

- **Yeni Kullanıcı**

Yeni kullanıcı, kullanıcı menüsü altında bulunmaktadır. Sisteme yeni kullanıcı eklenecekse bu ekrandan yapılmaktadır. Kullanıcı bilgileri olarak; adı, soyadı, kullanıcı adı, şifre, mail ve rolü girilmektedir. Kullanıcı terminallerle ilişkilendirilmesi bu ekrandan sağlanmaktadır. Terminal bilgileri olarak terminal numarası, ip adresi ve MAC adresi girilmektedir. Şekil 6.5 'te görüldüğü üzere kullanıcı bilgileri bu ekranda yapılarak kullanıcı eklenir. Kullanıcı düzenleme sonucu kullanıcıyı bilgilendirir.

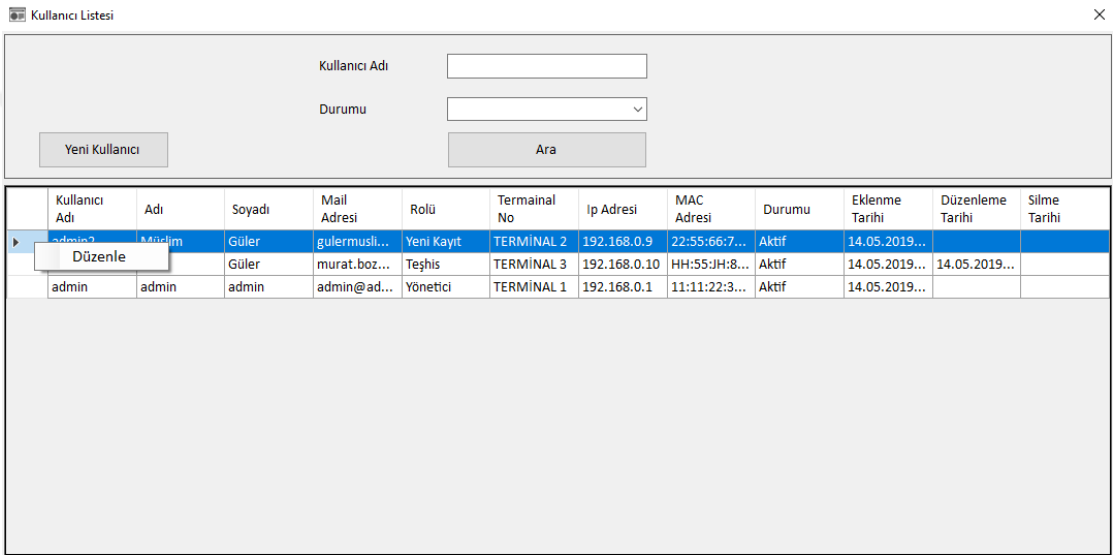


Şekil 6.5. Kullanıcı düzenleme

Kullanıcı silme ve güncelleme aynı ekran üzerinden yapılmaktadır. Sistem yeni ya da güncellenen kullanıcı olduğunu anlayıp ona göre işlem yapmaktadır.

- **Kullanıcı Listeleme**

Kullanıcı listeleme, kullanıcı menüsü altında bulunmaktadır. Sistemdeki kullanıcılar bu ekranda listelenmiştir. Sorgulama kriterine olarak kullanıcı adı ve rolüne göre arama yapılabilmektedir. Yeni kullanıcı eklenecekse bu ekrandan yapılabilmektedir. Sağ click yapılarak kullanıcı detay bilgileri görüntülenebilir.



Şekil 6.6. Kullanıcı listesi

Şekil.6.6’ da görüldüğü üzere sistemdeki kullanıcılar listelenmiştir.

- **Kişi Bazında İşlemler**

Kişi bazında işlemler, işlemler menüsü altında bulunmaktadır. Sisteme yeni kaydedilen kişileri görüntülemek için kullanılır. Yönetici rolüne sahip kullanıcı sistemdeki yeni kaydedilen parmak izlerini ve bölütlenmiş parmakları izleri inceleyebilir. Kaydı oluşturan kullanıcı bilgilerini ekranda görülmektedir. Böylece

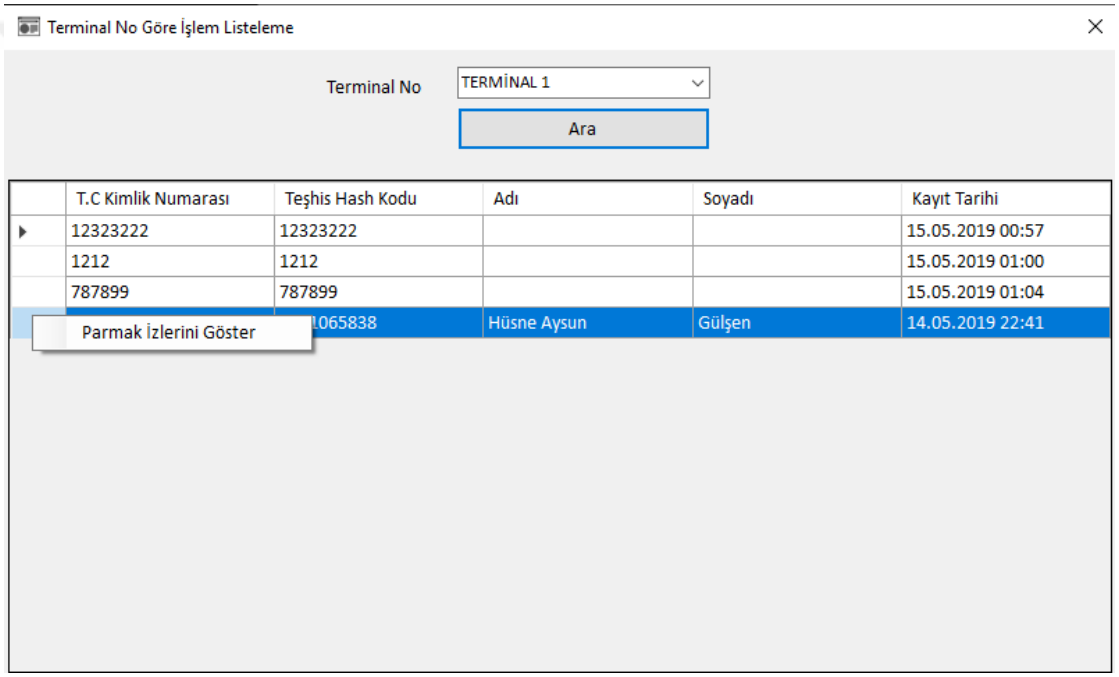
kişi kaydını hangi kullanıcının aldığı, hangi terminalden ve ip adresinden aldığını görebilir. Böylece kişi ve kullanıcıyı takip edebilir.

Şekil 6.7. Parmak izlerini görüntüleme

Şekil.6.7’ de örnek bir kayıt görüntülenmiştir.

- **Terminal Bazında İşlemler**

Terminal bazında işlemler, işlem menüsü altında bulunmaktadır. Terminallere göre isteme yeni kaydedilen kişileri görüntülemek için kullanılır. Yönetici rolüne sahip kullanıcı sistemdeki yeni kişileri listeleyebilir. Listelenen kayıta sağ click yaparak parmak izlerini, bölütlenmiş parmak izlerini ve terminal bilgisini görüntüleyebilir. Böylece kişi ve kullanıcıyı terminal bazında takip edebilir.



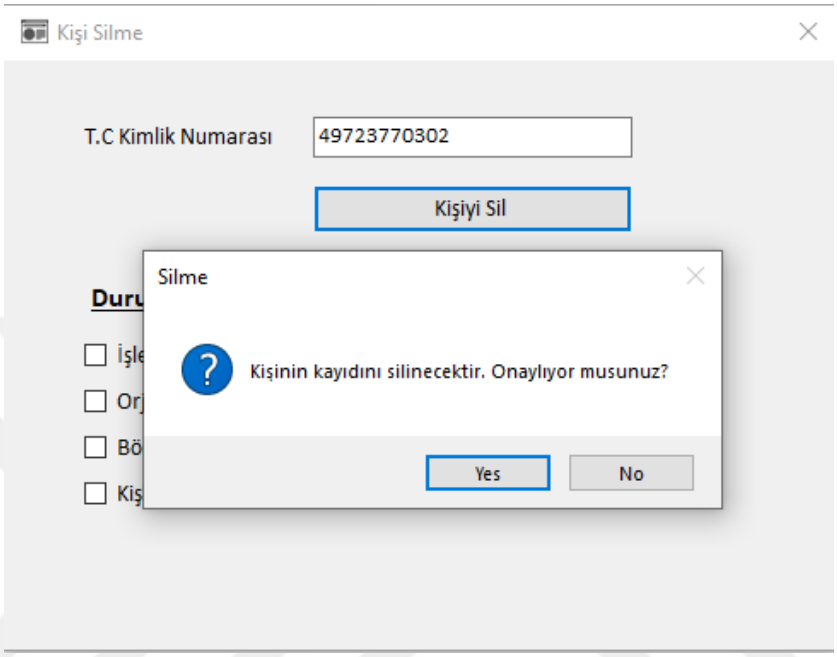
Şekil 6.8. Terminal no göre yeni kayıt listeleme

Şekil.6.8’ de örnek bir kayıt görüntülenmiştir.

- **Kişi Silme**

Kişi silme, işlemler menüsü altında bulunmaktadır. Kişiyi silmek için T.C kimlik numarası girilir. Sile tıklanır. Onay mesajından sonra girilen T.C kimlik numarasına ait işlem kaydı, bölütlenmiş parmak izi bilgileri, orijinal parmak izi ve kişi bilgileri

arşivlenir. Arşiv işlem sonucu durumdaki kutucuklarda belli olur. Hepsi check olmuşsa durum başarılıdır. Örnek bir kayıt şekil.6.9.'da gösterilmiştir.

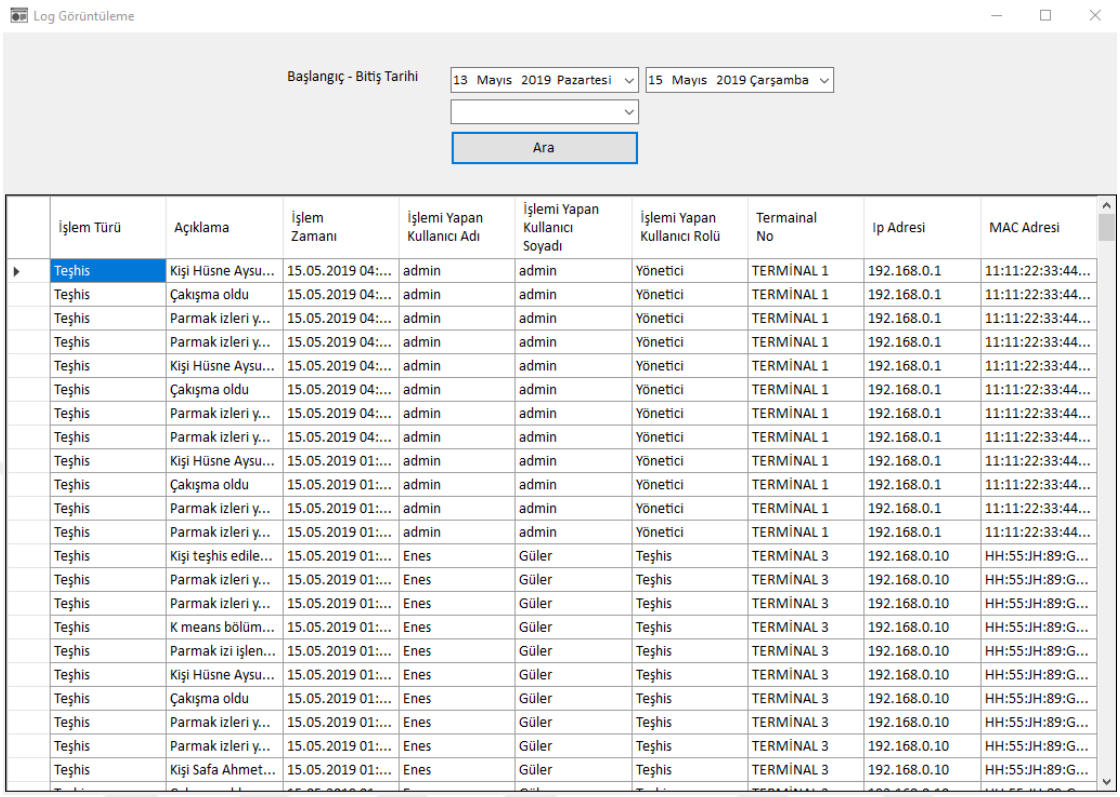


Şekil 6.9. Kişi silme

- **Log Görüntüleme**

Log görüntüleme, log menüsü altında bulunmaktadır. Yeni kayıt ve teşhis işlemlerinin işlem türüne göre detaylı şeklide görüntülendiği ekrandır. Tarih aralığı seçilerek veriler getirilir. Performans açısından verileri hızlı getirmek amacıyla kullanıcıdan tarih girilmesi beklenmektedir. Log detayında açıklama, işlem türü, işlem zamanı, işlemi yapan kullanıcı adı ve soyadı, işlemi yapan kullanıcı rolü işlem yapılan terminal no, ip adresi ve MAC adresi bulunmaktadır. Sistemde beklenmeyen bir surum karşılandığında log detayından görülmektedir.

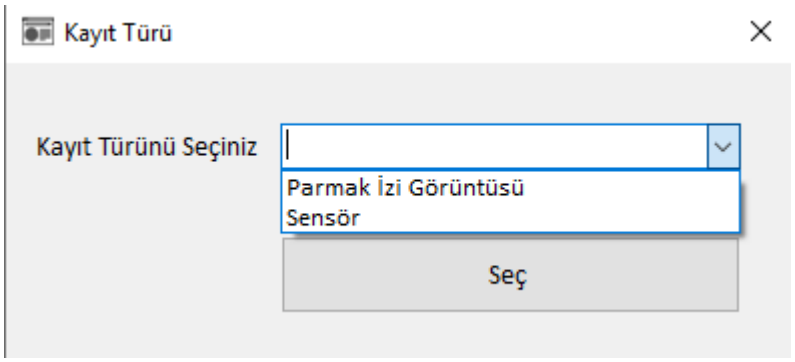
Şekil.6.10.' da örnek kayıtlar görüntülenmiştir



Şekil 6.10. Log ekranı

6.2.1.2. Yeni Kayıt

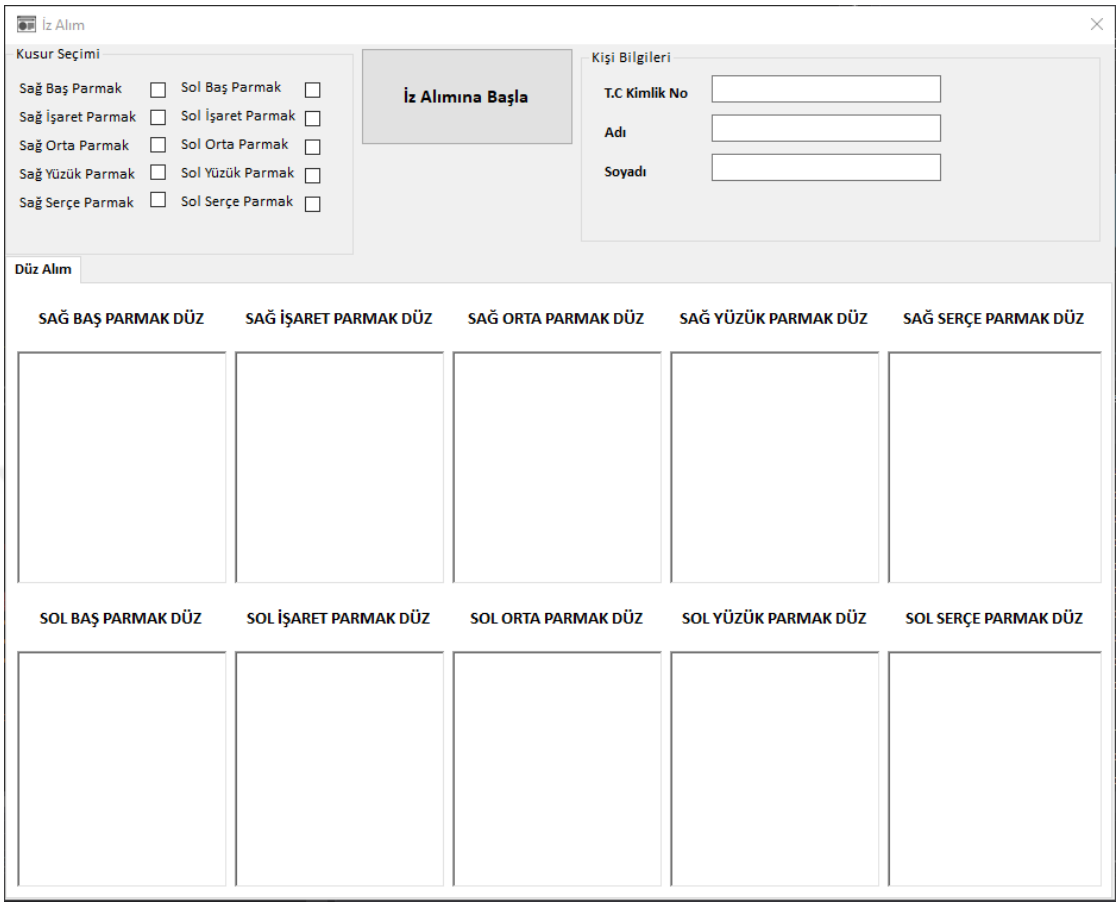
Yeni kayıt modülünde parmak izler sisteme girilmesini sağlar. Parmak izleri iki yolla sisteme alınabilmektedir. Bunlar sensörden ya da dosyadan okunarak yapılabilmektedir. Şekil 6.11.' de görüldüğü gibidir.



Şekil 6.11. Yeni kayıt

- **Sensörden Yeni Kayıt**

Sensörden yeni kayıt olduğunda, T.C. Kimlik numarası, adı ve soyadını girer. Kullanıcının parmakları kusur var ise kusurlu parmakları seçer. Parmaklarında kusur varsa kusurları seçip ‘İz Alımına Başla’ tıklar. Alım sırasında kusurlu parmaklar atlatılıp işleme devam edilir. Şekil 6.12.’ de alım sensörden yeni kayıt için ekrandır.



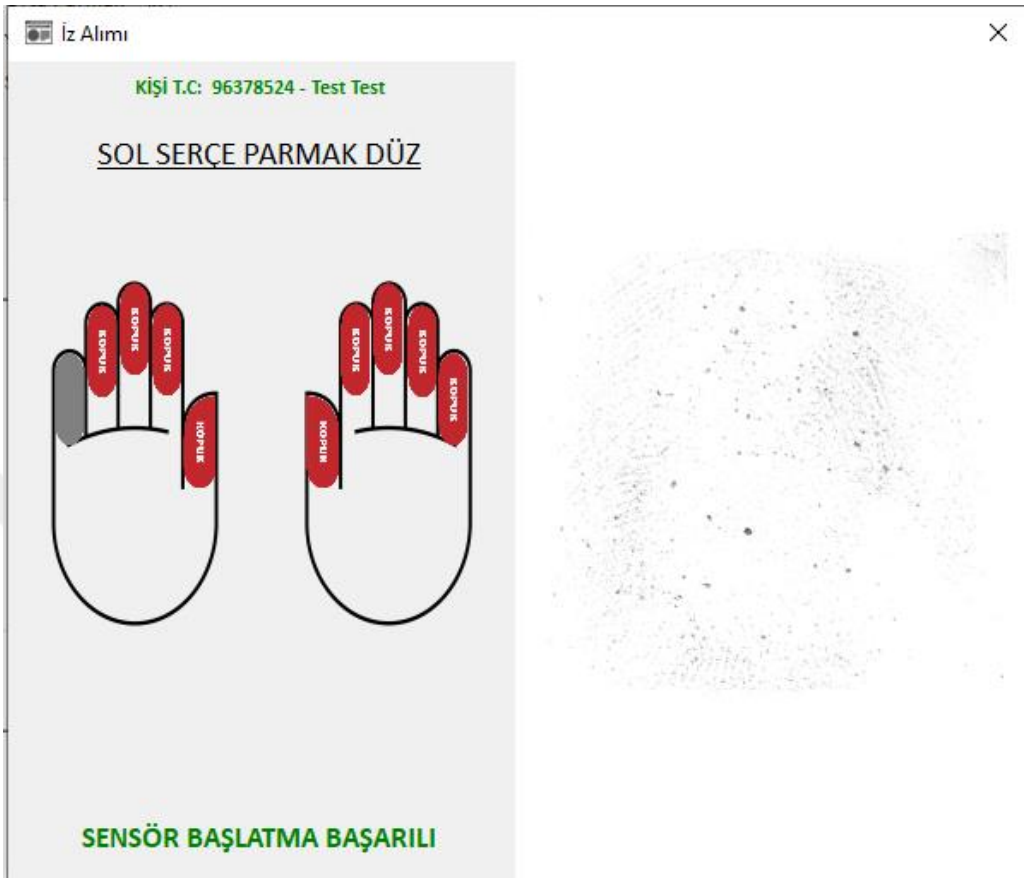
Şekil 6.12. Sensörden yeni kayıt

- **Sensörden Parmak İzi Alımı**

Kusur seçilen parmaklardan atlatıldığı için kalan parmaklar sırasıyla sensörden alım yapılır. Kusur seçilmeden yapılan alım şekil 6.13.'deki gibidir.

Şekil 6.13. Parmak izi alımı

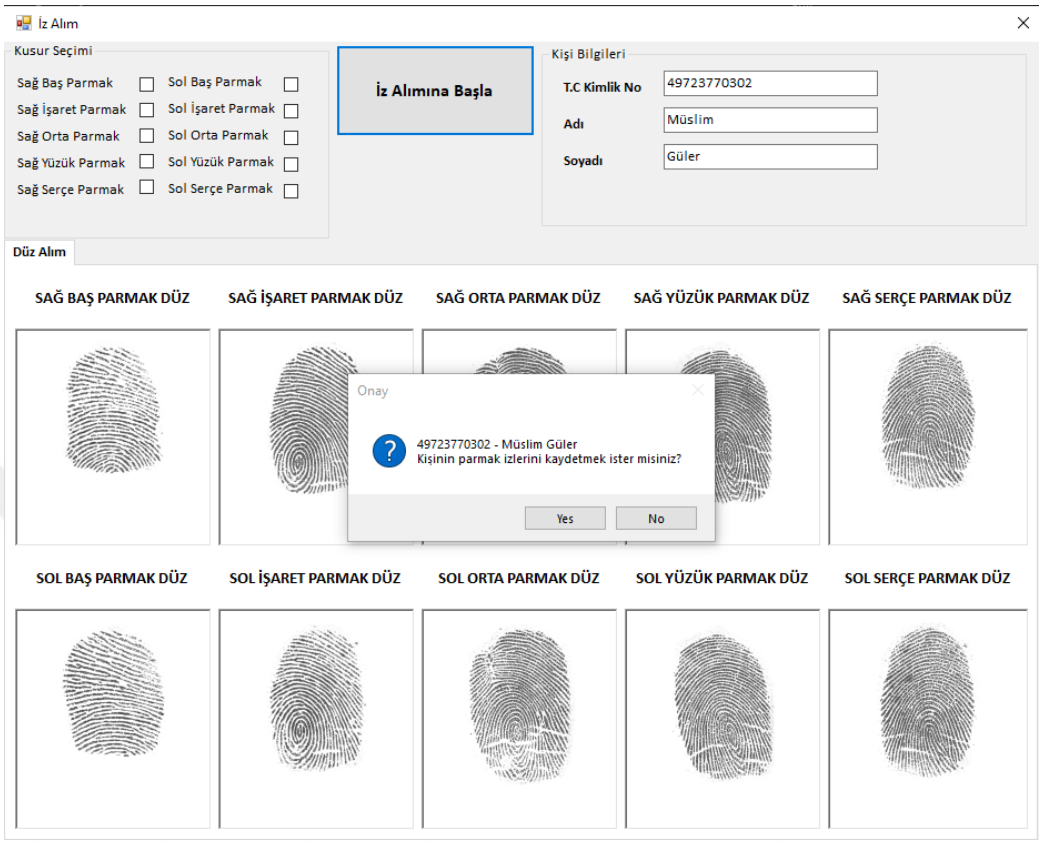
Kusurlu parmaklar olduğunda şekil 6.14.de görüldüğü gibi parmaklar kopuk olarak işaretlenir.



Şekil 6.14. Kusurlu parmak alımı

- **Ön İzleme Ekranı**

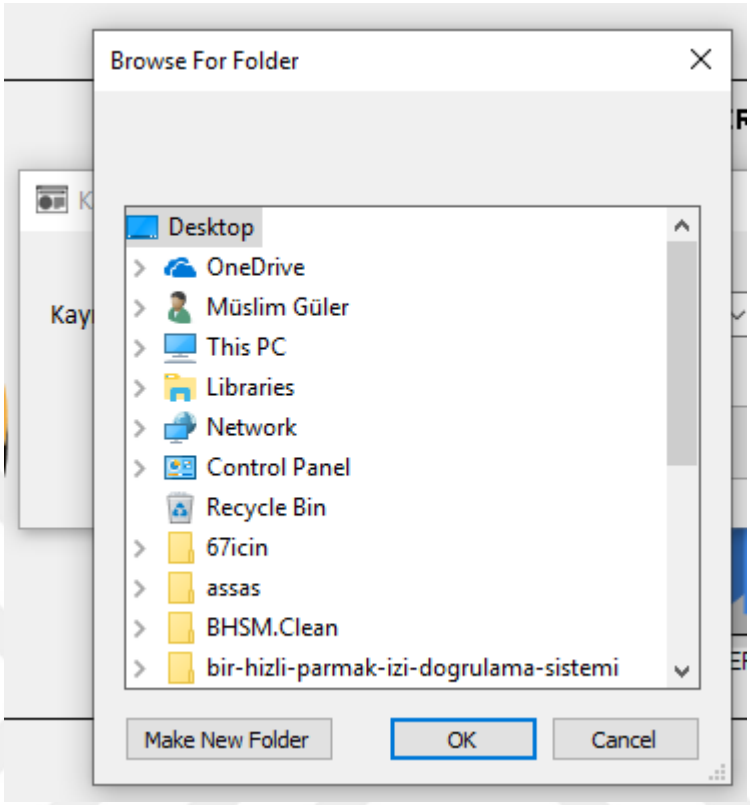
Alınan parmak izleri sırasıyla ön izleme ekranında görülür. Bu ekran ile alınan izler kıyaslanabilir. Alım bittikten sonra kullanıcıyı uyarıp parmak izi alımı tekrarlanabilir.



Şekil 6.15. Ön izleme ekranı

- **Klasörden Yeni Kayıt**

Klasörden yeni kayıt seçildiğinde, kullanıcıdan parmak izi görüntüsü bulunan klasör seçilmesi isteyecektir. Klasörün içinde parmak izi görüntüsü olan dosyaların adı "T.C Kimlik Numarası"+"-"+"Parmak indeks" şeklinde olmalıdır. Örneğin; 49723770302-1 gibi olmalıdır. Şekil 6.16.'da görüldüğü üzere parmak izi görüntüsü bulunan klasör seçilmesi beklenmektedir.



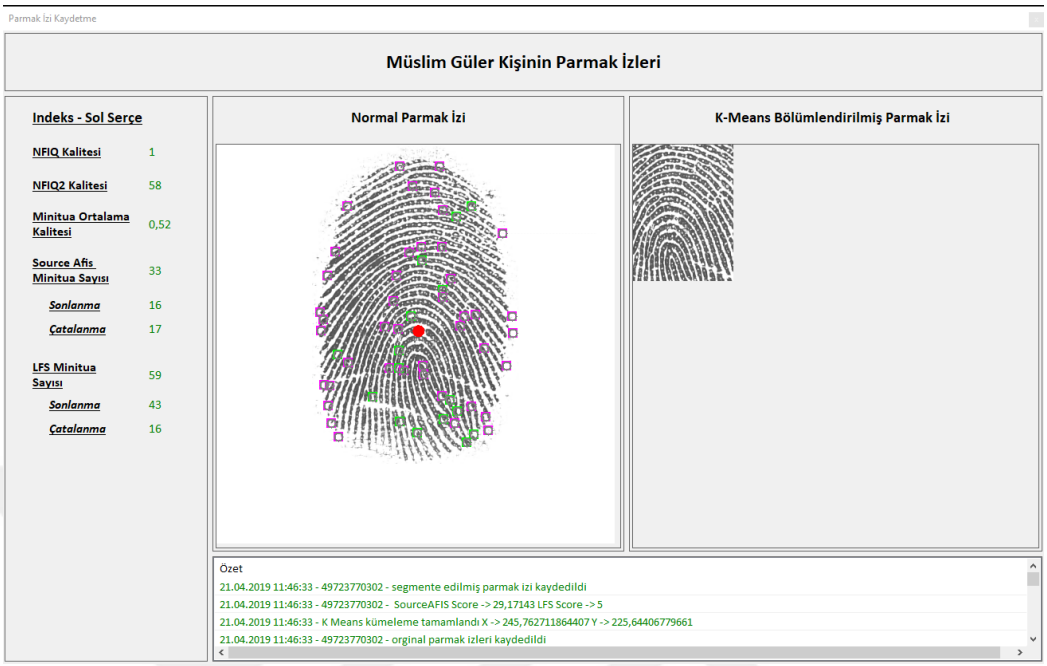
Şekil 6.16. Klasörden parmak izi seçimi

- **Parmak İzi Görüntüsü K-Means Bölümlendirilmesi**

Sensörden ya da klasörden seçilen parmak izleri kullanıcıyla ilişkilendirilip kaydedilir. Parmak izleri kalite ve iki ayrı algoritmaya minutiae hesaplaması yapılır. Sonuçlar kaydedilir. K-Means bölümlendirme bu ekranda yapılır. K-Means yapılırken kullanıcı parmak izinin nereden hangi noktada ağırlık noktası olduğu görebilir. Bunun parmak izinde görsellik sağlanmıştır. Minutiae tiplerine göre şekiller çizilmiştir. Şekil 6.17.'de görüldüğü üzere parmak izi üzerinde belirtilmiştir.

Şekil 6.17. K-Means ile işlenen parmak izi

K-Means ile bölümlendirilmiş parmak izi kişi ile ilişkilendirilip kaydedilir. Kaydedilen parmak izleri yönetici rolüne sahip kullanıcı tarafından kişi geçmişinden görüntüleyebilir. Şekil 6.18.'de görüldüğü üzere kişiye ait parmak izleri bölütlenmektedir.

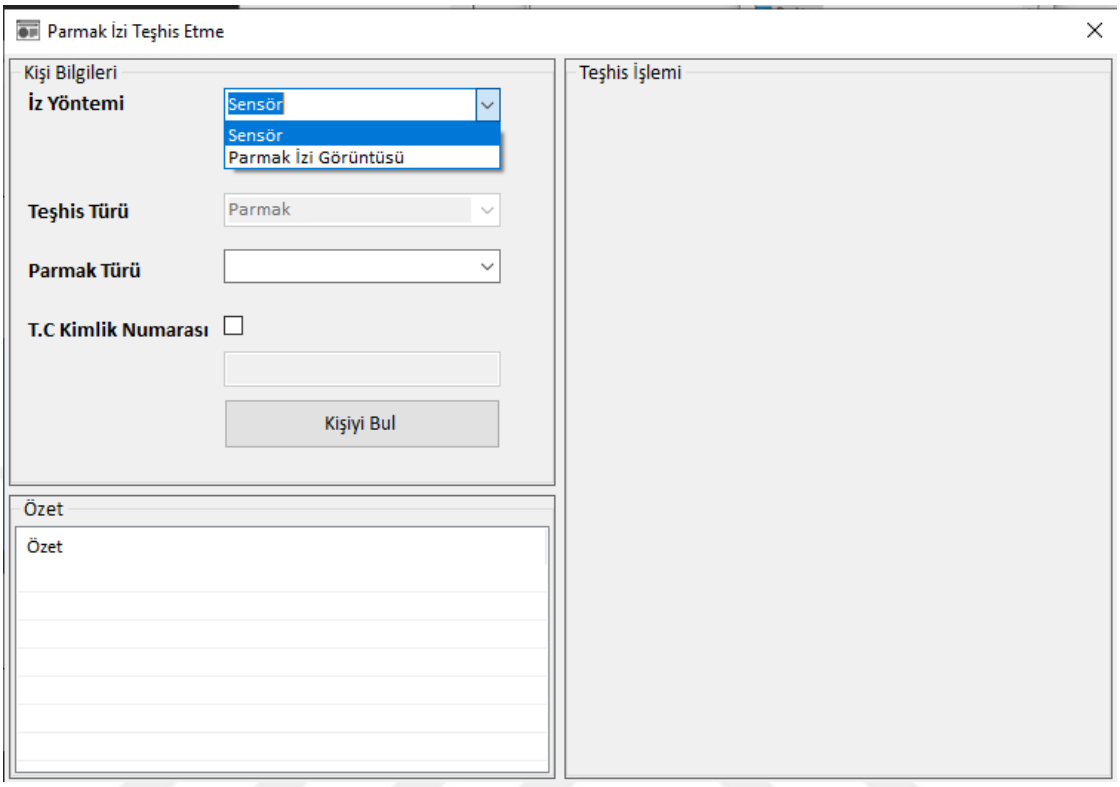


Şekil 6.18. K-Means bölümlendirmesi

Yapılan işlemler aynı ekranın alt tarafında loglanır. Loglanan kayıtlar log menüsünden görülmektedir. Şekil 6.18'de alt bölümde loğlar görülmektedir.

6.2.1.3. Teşhis İşlemi

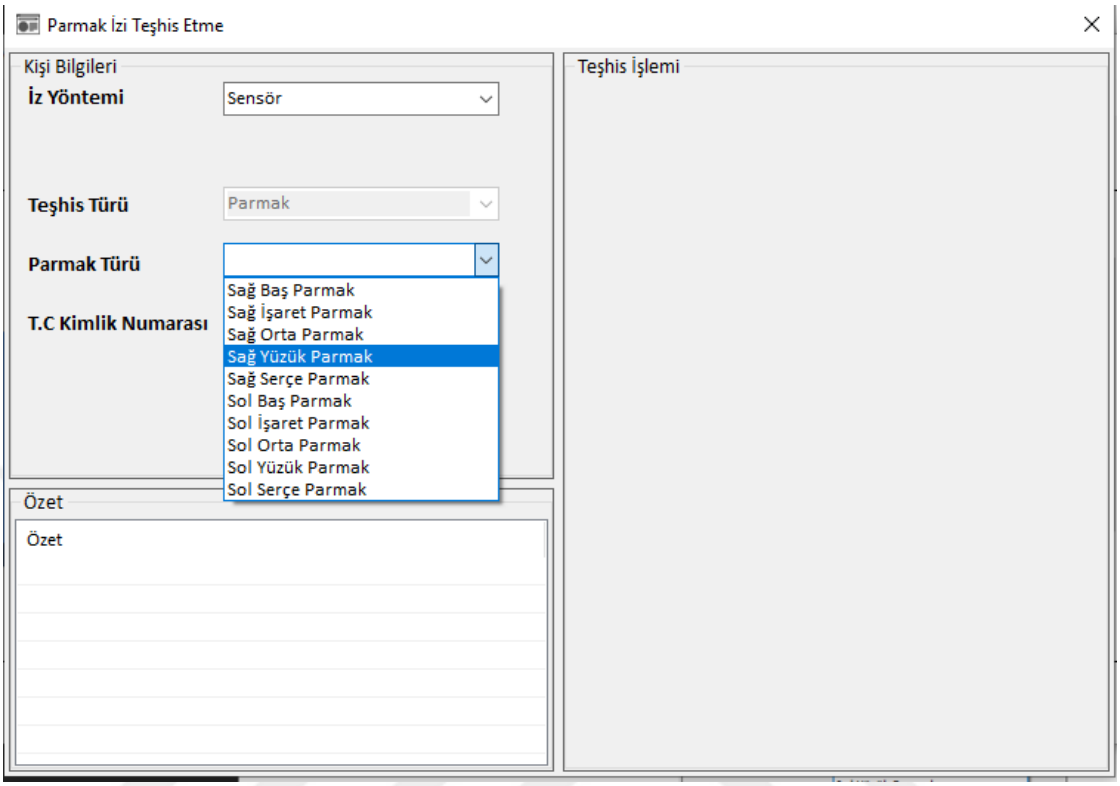
Teşhis işlemi, verilen parmak izinin kime ait olduğunu teşhis edilen aşamadır. Teşhis işlemi ya sensörden ya da parmak izi görüntüsünden olabilir.



Şekil 6.19. Teshis işlem türü seçimi

- **Sensörden Parmak İzi Alımı**

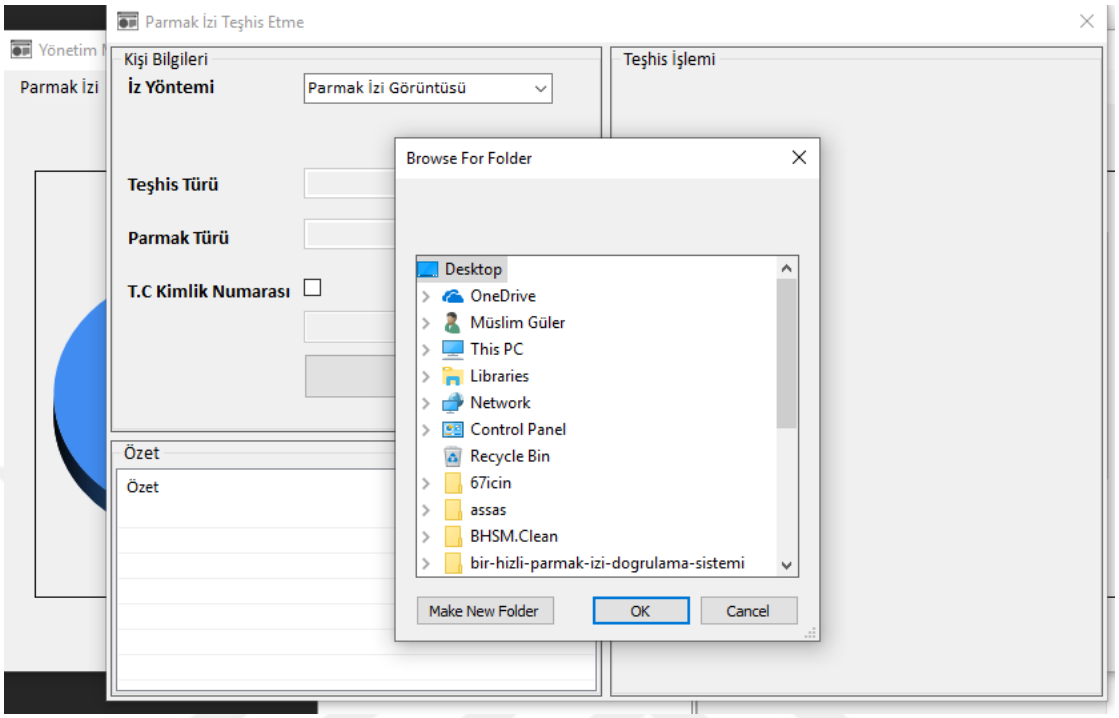
İz yönteminde sensör seçildiğinde teşhis türü parmak otomatik olarak yüklenir. Parmak izi seçilerek parmak izi alım adımına geçilir. Yeni kayıtta sensörden alınan parmak izi adımları burada aynı şekilde uygulanmaktadır. Buradaki fark bütün parmaklar değil de seçilen parmak için parmak izi alımı yapılır. Şekil 6.20.' de görülmektedir.



Şekil 6.20. Parmak seçimi

- **Klasörden Parmak İzi Görüntüsü Seçimi**

İz yönteminde parmak izi görüntüsü seçildiğinde parmak izleri içeren klasör seçilmelidir. Klasör içindeki parmak izi sayısı bir ise parmak türü seçilmesi istenir. Değil ise otomatik parmak türünü bulacaktır.

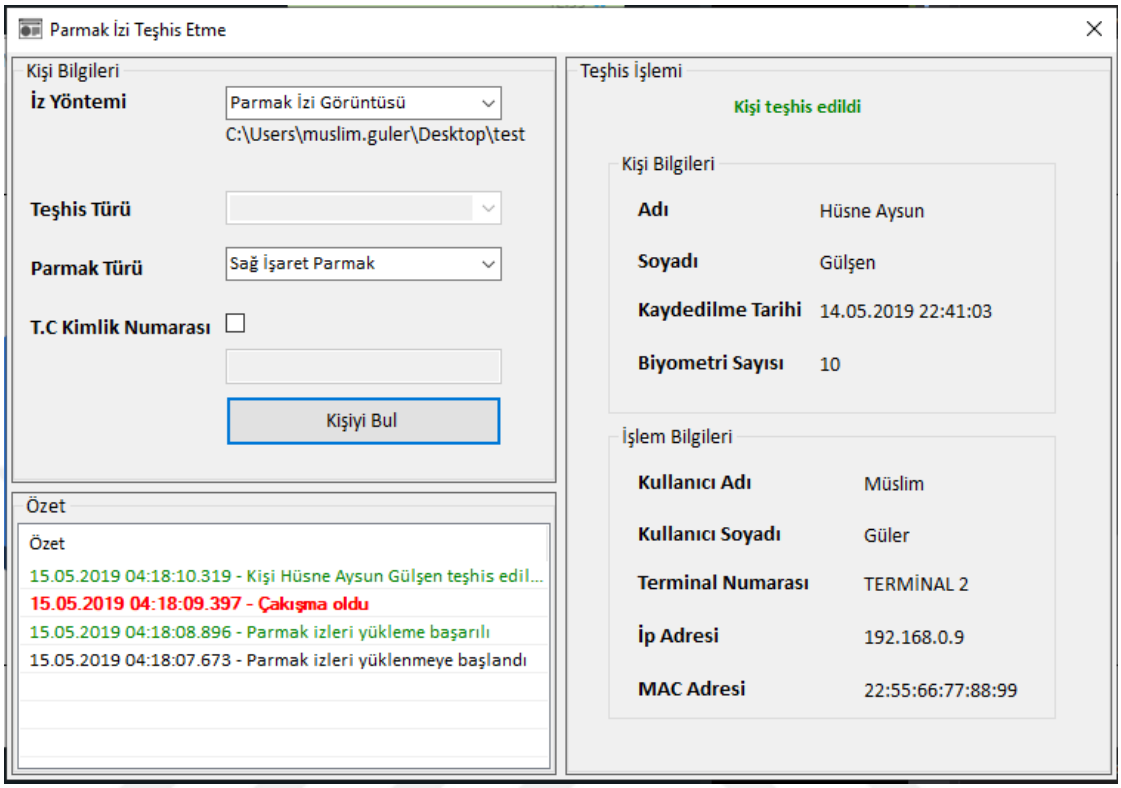


Şekil 6.21. Parmak izini klasörden yükleme

- **Teşhis İşlemi**

Sensör ya da klasörden okunan parmak izi görüntüsü K-Measn bölümlendirilerek teşhis işlemine başlanır. Veritabanından yüklenen kişilerin parmak izleri için teşhis edilen kişinin T.C kimlik numarası varsa kişiler filtrelenir. Filtreleme işlemi bittikten sonra parmak izleri teşhis için işleme sokulur. Aslında kişi T.C kimlik numarası girdiğinde kişi kendisini doğrulamış olur.

Teşhis işlemi sonucunda teşhis edilen kişi bilgileri ve kişiyi kayıt yapan kullanıcı ve terminal bilgileri ekranda gösterilir.



Şekil 6.22. Teşhis işlemi

Teşhis işleminde yapılan adımlar özet kısmında detaylı şekilde loglanır.

6.2.2. Kodlar

Uygulamada önemli görülen kodlar başlıklarıyla eklenmiştir.

- **Yeni Kullanıcı ve Biyometriler Ekleme**

Aşağıdaki kod parçasında yeni eklenecek olan kişi ve biyometrilerini eklemeyi sağlayan kod parçası görülmektedir.

```
public int InsertCitizen(CitizenDto citizenDto)
    {
      var result =
        _matcherGenericRepository.FirstOrDefault<tbl_citizen>(m=>
m.CitizenId == citizenDto.CitizenId);
      if (result == null)
      {
        var tbl_citizen = new tbl_citizen
        {
          CitizenId = citizenDto.CitizenId,
          InsertDate = DateTime.Now,
          Name = citizenDto.Name,
          Surname = citizenDto.Surname,
          MatcherId = citizenDto.MatcherId
        };
        var result = _matcherGenericRepository.AddOrUpdate(tbl_citizen);
        var result1 = _matcherGenericRepository.SaveChanges(false);
```

```
            return result.Id;
        }
        return 0;
    }
var tblBiometry = new tbl_biometry()
        {
            CitizenId = biometryDto.CitizenId,
            Bitmap = biometryDto.Bitmap,
            FingerIndex = biometryDto.FingerIndex,
            ImageData = biometryDto.ImageData,
            Iso = biometryDto.Iso,
            TakenDate = DateTime.Now,
            Template = biometryDto.Template,
            Id = biometryDto.Id,
            AverageMinitua = biometryDto.AverageMinitua,
            Nfiq2Quality = biometryDto.Nfiq2Quality,
            NfiqQuality = biometryDto.NfiqQuality,
            SourceAfisMinituaCount = biometryDto.SourceAfisMinituaCount,
            LfsMinituaCount = biometryDto.LfsMinituaCount,
            SegmentationCitizenId = biometryDto.SegmentationCitizenId
        };
        var aa = _matcherGenericRepository.AddOrUpdate(tblBiometry);
```

```
var bb = _matcherGenericRepository.SaveChanges(false) > 0;
return aa.Id;
```

Aşağıdaki kod parçacığı ile LFS ve SourceAFIS algoritmalarına göre minutiae hesaplaması yapılmaktadır.

- **LFS minutiae hesaplaması**

```
NW_GBNFIQ_ImageQuality_Info_InputData Input = new NW_GBNFIQ_ImageQuality_Info_InputData();
    NW_GBNFIQ_ImageQuality_Info Info;
    NW_GBNFIQ_ImageQuality_Info_OutputData Output;
    Minutiae = null;
    Input.Frame = Frame;
    Input.Width = SizeX;
    Input.Height = SizeY;
    Info = new NW_GBNFIQ_ImageQuality_Info(Input);
    Output = Info.NW_GBNFIQ_GetImageQuality();
    List<LfsMinutia> minutias = new List<LfsMinutia>();
    foreach (var minutiaeItem in Output.MinInfo)
    {
       var minutia = new LfsMinutia()
       {
          Degrees = minutiaeItem.Direction_Degrees,
```

```
                Reliability = minutiaeItem.reliability,
                Type = minutiaeItem.type == 0 ? MinutiaType.Bifurcation :
MinutiaType.Ending,
                X = minutiaeItem.x,
                Y = minutiaeItem.y
            };
            minutias.Add(minutia);
        }
        Minutiae = minutias;
        return Output.NFIQ_Quality;
```

- **SourceAFIS minutiae hesaplaması**

```
SourceAFIS.Simple.Fingerprint fingerprint = new SourceAFIS.Simple.Fingerprint();
fingerprint.AsBitmap = bitmap;
SourceAFIS.Extraction.Extractor extractor = new SourceAFIS.Extraction.Extractor();
var templateBuilder = extractor.Extract(fingerprint.Image, 500);
List<SourceAfisMinutia> sourceAfisMinutias = new List<SourceAfisMinutia>();
        foreach (var item in templateBuilder.Minutiae)
        {
            SourceAfisMinutia sourceAfisMinutia = new SourceAfisMinutia();
            sourceAfisMinutia.Direction = item.Direction;
            sourceAfisMinutia.X = item.Position.X;
```

```
            sourceAfisMinutia.Y = item.Position.Y;
            sourceAfisMinutia.Type = (MinutiaType)((int)item.Type);
            sourceAfisMinutias.Add(sourceAfisMinutia);
        }
        return sourceAfisMinutias;
```

Aşağıdaki kod parçacığı ile K-Means ile bölümlendirme yapılmaktadır. Çıkan kümeye göre image olarak kaydedilmektedir.

```
List<Centroid> centroidList = new List<Centroid>();
        for (int i = 0; i < _k; i++)
        {
            Centroid centroid = new Centroid(dataSet, Misc.centroidColors[i]);
            centroidList.Add(centroid);
        }
        OnUpdateProgress(new KMeansEventArgs(centroidList, dataSet));
        while (true)
        {
            foreach (Centroid centroid in centroidList)
                centroid.Reset();
            for (int i = 0; i < dataSet.GetLength(0); i++)
            {
```

```
    double[] point = dataSet[i];
    int closestIndex = -1;
    double minDistance = Double.MaxValue;
    for (int k = 0; k < centroidList.Count; k++)
    {
        double distance = _distance.Run(centroidList[k].Array, point);
        if (distance < minDistance)
        {
            closestIndex = k;
            minDistance = distance;
        }
    }
    centroidList[closestIndex].addPoint(point);
}
foreach (Centroid centroid in centroidList)
    centroid.MoveCentroid();
OnUpdateProgress(new KMeansEventArgs(centroidList, null));
bool hasChanged = false;
foreach (Centroid centroid in centroidList)
    if (centroid.HasChanged())
    {
        hasChanged = true;
```

```
            break;
        }
    if (!hasChanged)
        break;
}
return centroidList.ToArray();
```

Aşağıdaki kod parçacı ile veritabanından biyometrileri çekip sourceAFIS sistemine tanıtmaktadır.

```
List<BiometryDto> biometryDtos = new List<BiometryDto>();
var tblBiometry = _matcherGenericRepository.IncludeMultiple<tbl_biometry>(m => m.tbl_citizen, m => m.tbl_citizensegmentation);
if (citizenId.HasValue)
    tblBiometry = tblBiometry.Where(m => m.tbl_citizen.CitizenId == citizenId.Value);
if (segmantationType.HasValue)
    tblBiometry = tblBiometry.Where(m => m.tbl_citizensegmentation.Type == (int)segmantationType.Value);
if (fingerIndex.HasValue)
    tblBiometry = tblBiometry.Where(m => m.FingerIndex == fingerIndex.Value);
```

```
            var config = new MapperConfiguration(cfg =>
        {
            cfg.CreateMap<tbl_citizen, CitizenDto>();
            cfg.CreateMap<tbl_citizensegmentation, CitizenSegmentationDto>();
            cfg.CreateMap<tbl_biometry, BiometryDto>().ForMember(x => x.CitizenDto,
opt => opt.MapFrom(x => x.tbl_citizen)).ForMember(x =>
x.CitizenSegmentationDto, opt => opt.MapFrom(x => x.tbl_citizensegmentation));
        });
            IMapper iMapper = config.CreateMapper();
            biometryDtos = iMapper.Map<IQueryable<tbl_biometry>,
List<BiometryDto>>(tblBiometry);
            return biometryDtos;
       List<Person> persons = new List<Person>();
            foreach (var citizenDto in citizenDtos)
            {
                if (citizenDto.BiometryDtos.Any())
                {
                    Person person = new Person();
                    person.Id = citizenDto.Id;
                    person.Fingerprints = new List<Fingerprint>();
                    ConcurrentBag<Fingerprint> fingerprints = new
ConcurrentBag<Fingerprint>();
```

```
            Parallel.ForEach(citizenDto.BiometryDtos, m =>
            {
               Fingerprint fingerprint = new Fingerprint();
                fingerprint.Template = m.Template;
               fingerprint.Finger = (Finger)m.FingerIndex;
               fingerprints.Add(fingerprint);
            });
            person.Fingerprints.AddRange(fingerprints);
            fingerprints.Clear();
            persons.Add(person);
         }
      }
      return persons;
```

Aşağıdaki kod parçacığı teşhis işlemi yapmaktadır. Dönen sonuç kişi listesidir.

```
SourceAFIS.Simple.AfisEngine afisEngine = new AfisEngine();
      afisEngine.Extract(person);
      afisEngine.Threshold = threshold;
      return afisEngine.Identify(person, persons).ToList();
```

Aşağıdaki kod parçacığı doğrulama işlemi yapmaktadır. Doğrulamadan çıkan skor dönmektedir.

SourceAFIS.Simple.AfisEngine afisEngine = new AfisEngine();

return afisEngine.Verify(probe, candidate);

7. SONUÇLAR

Bu çalışmanın birinci bölümünde biyometrik türler ve biyometrik sistemler incelenmiştir. Biyometrik sistemlerin temel çalışma prensiplerindekinden olan yeni kayıt ve teşhis aşamaları irdelenmiştir. Herhangi bir kişinin sistemde teşhis ve doğrulamak için hedef belirlenmiştir. Bu hedef doğrultusunda parmak izi tanıma algoritmalarından Minutiae tabanlı algoritmasını kullanan SourceAFIS ile kütüphanesi hedef gerçeklenmiştir.

Çalışmanın ikinci bölümünde parmak izlerinin iki LFS ve SourceAFIS kütüphaneleriyle Minutiae çıkarımları sağlanmıştır. Çıkarılan minutiaelardan "K-Means Kümeleme" algoritmasını kullanarak parmak izinin tümünü ile kümenlenenen minutiae bölütlenerek elde dilen parmak izi görüntüsü ilişkisel veritabanında saklanmıştır. Sistemden teşhis onayı geldiğinde ilk önce bölütlenmiş veritabanında arama yapılır. Çakışma olduğu zaman olarak parmak izlerinin tümü aranmayıp sadece çakışma olan parmak izleri üzerinde teşhis işlemi gerçekleştirilir.

Sonuçta, arama uzayı hafızaya taşınarak hız sağlanmıştır. Böylece zaman ve yer bakımından kazanç elde edilmiştir. Arama uzayında akıllı indeksler kullanılarak aramada daha kazançlı hale dönüştürülebilir. Bu teşhis sistemi kullanılarak, standolane ve dağıtılmış yöntemde sistemler geliştirilebilir. Bu sistemlerin tasarlanması tez kapsamında belirtilmiş ama dağıtılmış sistemlerin uygulanması gerçekleştirilmemiştir. Standolane sistem için uygulama hazırlanmıştır. Uygulama tasarımı ilişkisel veritabanı kullanılarak inşa edilmiştir. Dağıtık sistemlerde başka parmak izi sensörleri entegre

edilebilir. Parmak izi biyometrisi alternatifi olarak başka biyometri kullanılıp sistem genişletilebilir. Ayrıca kümeleme algoritmaların performans açısından diğer kümeleme algoritmaları sisteme entegre edilip performans testleri yapılabilir. Sistemde ilişkisel veritabanı yerine NoSQL kullanarak hız ve yönetimi bakımından değerlendirilebilir. İşte, bu maddelerin tasarlanması yüksek lisans tez kapsamının dışında tutulmuş olup sonraki çalışmalarda yapılması planlanmaktadır.

8. KAYNAKLAR

[1] Chikkerur, S.S., Cartwright, A.N. ve Govindaraju, V.(2007). "Fingerprint Image Enhancement Using STFT Analysis." *Journal Pattern Recognition, 40,* (1).

[2] Jain, A.K., Prabhakar, S. ve Hong, L. (1999). "A Multichannel Approach to Fingerprint Classification." *Journal IEEE Transactions on Pattern Analysis and Machine Intelligence, 21, (4),* 349-359.

[3] J. G. Daugman, "High confidence visual recognition of persons by a test of statistical independence", *IEEE Trans. Pattern Analysis and MachineIntelligence, Vol. 15, No. 11, pp. 1148-1161, 1993*

[4] D. Zhang, X. Jing, J. Yang, "Biometric Image Discrimination Technologies," *80-95, Idea Group Publishing, 2006.*

[5] Hemant V., (2012), "Authentication Using Finger-Vein Recognition", *M.Sc. Thesis, University of Johannesburg, pp.* 1-185.

[6] Ashbourn J., (2004), "Practical Biometrics: from Aspiration to Implementation", *Business & Economics, pp.* 1-159.

[7] Jain A. K., Hong L., and Pankanti S., (2000), "Biometrics: Promising Frontiers for Emerging Identification Market*", Commun. ACM, Vol. 43, No. 2, pp.* 91–98.

[8] Sheng W., Howells G., Fairhurst M. and Deravi F., (2007), "A Memetic Fingerprint Matching Algorithm", *IEEE Transactions on Information Forensics and Security, Vol. 2, No. 3,* pp. 402-412.

[9] Jain A. K., Ross A., and Pankanti S., "Biometrics: A Tool for Information Security", *IEEE Transactions on Information Forensics Security, Vol. 1, No. 2,2006,* pp. 125–143.

[10] Lee H. C., and Gaensslen E. R., (2001), "Advanced in Fingerprint Technology*", CRC Pres, London, Second Edition,* pp. 85-456.

[11] Maltoni, D., Maio, D., Jain, A.K., and Prabhakar, S., "Handbook of Fingerprint Recognition", *Berlin, Germany: Springer-Verlag, 2003*

[12] Jain, A.K., Ross, A., Prabhakar, S.,"An introduction to biometric recognition, IEEE Transaction on Circuits and Systems for Video Technology," *14(1), 4-19, 2004.*

[13] Indovina, M., Uludag, U., Snelick, R., Mink A., and Jain, A., "Multimodal biometric authentication methods: a cots approach, Proc." *MMUA 2003, Workshop on Multimodal User Authentication, 99-106, 2003.*

[14] Fierrez-Aguilar, J., Ortega-Garcia, J., Gonzalez-Rodriguez, J., Bigun, J., "Discriminative multimodal biometric authentication based on quality measures, Pattern Recognition," *38(5), 777–779*, 2005.

[15] K.Delac, M.Grgic, "A Survey Of Biometric Recognition Methods", *46th International Symposium Electronics In Marine, ELMAR-2004, 16-18 Haziran 2004, Zadar, Croatia*, 184, 193

[16] O.G. Martinsen, S. Clausen., J. B. Nysæther, And S. Grimnes., "Utilizing Characteristic Electrical Properties Of The Epidermal Skin Layers To Detect Fake Fingers In Biometric Fingerprint Systems—A Pilot Study", *Ieee*

Transactions On Biomedical Engineering, Cilt 54, No. 5, Mayıs 2007, Sayfa 891-894

[17] Kuribayashi, M. and Tanaka, H. (2005). "Fingerprinting protocol for images based on additive homomorphic property". *IEEE Transactions on Image Processing, 14(12):*2129–2139.

[18] Dumer, I. "Equal-Weight Fingerprinting Codes, Second International workshop on codeing and cryptology", *Lecture Notes in Computer Science, Volume 5557*, 2009, pages 43-51.

[19] Jing-Wein Wang, Ngoc Tuyen Le, Chou-Chen Wang, and Jiann-Shu Lee," Enhanced Ridge Structure for Improving FingerprintImage Quality Based on a Wavelet Domain", *IEEE signal processing letters, VOL. 22, NO. 4*, April 2015, pp 390-394

[20] Debiao He, and Ding Wang, "Robust Biometrics-Based Authentication Scheme for Multiserver Environment", *IEEE systems journal, pp. 1-8, 2014*

[21] Josef Strom Bart unek, Mikael Nilsson, Benny Sallberg, and Ingvar Claesson,"Adaptive Fingerprint Image Enhancement With Emphasis on Preprocessing of Data‖", *IEEE transactions on image processing, vol. 22*, no. 2, pp. 644-656, February 2013

[22] Weiguo Sheng, Gareth Howells, Michael Fairhurst, and Farzin Deravi, "A Memetic Fingerprint Matching Algorithm" *IEEE transactions on information forensics and security, vol. 2, no. 3*, pp. 402-412, september 2007

[23] D. Zhang, X. Jing, J. Yang, "Biometric Image Discrimination Technologies," *80-95, Idea Group Publishing, 2006.*

[24] Xudong Jiang, Manhua Liu, and Alex C. Kot, "Fingerprint Retrieval for Identification" *IEEE transactions on information forensics and security, vol. 1, no. 4*, pp. 532-542 december 2006.

[25] Arun Ross, Jidnya Shah, and Anil K. Jain, "From Template to Image: Reconstructing Fingerprints from Minutiae Points" *IEEE transactions on pattern analysis and machine intelligence, vol. 29, no. 4*, pp. 544-560, april 2007.

[26] Weiguo Sheng, Gareth Howells, Michael Fairhurst, and Farzin Deravi, "A Memetic Fingerprint Matching Algorithm*" IEEE transactions on information forensics and security, vol. 2*, no. 3, pp. 402-412, september 2007

[27] Lavanya B N, K B Raja, Venugopal K R and L M Patnaik, "Minutiae Extraction in Fingerprint using Gabor Filter Enhancement*" 2009 international conference on advances in computing, control, and telecommunication technologies, IEEE*, pp. 54-56, 2009 13

[28] David Zhang , Feng Liu, Qijun Zhao, Guangming Lu, and Nan Luo, "Selecting a Reference High Resolution for Fingerprint Recognition Using Minutiae and Pores" *IEEE transactions on instrumentation and measurement, vol. 60*, no. 3, pp. 863-871, march 2011.

[29] https://www.bayometric.com/minutiae-based-extraction-fingerprint-recognition/ (Ziyaret tarihi Mart 2019)

[30] https://www.gemalto.com/govt/biometrics/afis-history (Ziyaret tarihi Mart 2019)

[31] https://sourceafis.machinezoo.com/algorithm. (Ziyaret tarihi Şubat 2019)

[32] https://sourceafis.machinezoo.com/transparency/ (Ziyaret tarihi Şubat 2019)

[33] https://www.nist.gov/srd/shop/special-database-catalog (Ziyaret tarihi Şubat 2019)

[34] https://sourceafis.machinezoo.com/template (Ziyaret tarihi Mart 2019)

[35] https://ekilavuz.com/greenbit-sls442--parmak-izi-tarayici--tarayici--scanner--587a840ca31f9cb8-10 (Ziyaret tarihi Mart 2019)

[36] https://ekilavuz.com/greenbit-dactyscan-26i-parmak-izi-tarayici-tarayici--scanner--042b55081f54b7f5-7 (Ziyaret tarihi Şubat 2019)

[37] www.mmfdergi.gazi.edu.tr/article/downloadSuppFile/ (Ziyaret tarihi Mart 2019)

ÖZ GEÇMİŞ

Müslim GÜLER, 2009 yılında Şanlıurfa Harran Üniversitesi Bilgisayar Mühendisliği bölümünü tamamladıktan sonra, 2016 yılında Okan Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği programında tezli yüksek lisans eğitimine başlamıştır.

İş deneyimleri arasında 2013 yılında Pro-line Bilişim Sistemleri firmasında İstanbul ana merkezinde Yazılım Uzmanı olarak görev yapmaya başlamıştır Şu anda ise aynı firmada Biyometri ve İstemci Uygulamaları Ekip Liderliği olarak halen devam etmektedir.