BİYOMETRİK KİMLİK DOĞRULAMA İÇİN ALAN TABANLI TUŞA BASMA DİNAMİKLERİ ANALİZİ

Neşe AGUN
Yüksek Lisans Tezi

Bilgisayar Mühendisliği Anabilim Dalı
Şubat-2016

JÜRİ VE ENSTİTÜ ONAYI

Neşe AGUN'un **"Biyometrik Kimlik Doğrulama İçin Alan Tabanlı Tuşa Basma Dinamiklari Analizi"** başlıklı **Bilgisayar Mühendisliği** Anabilim Dalındaki, Yüksek Lisans Tezi 03.02.2016 tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

	Adı-Soyadı	**İmza**
Üye (Tez Danışmanı)	**: Prof. Dr. YUSUF OYSAL**	**……...**
Üye	**: Doç. Dr. CÜNEYT AKINLAR**	**……...**
Üye	**: Yrd. Doç. Dr. MUSTAFA ATANAK**	**……..**

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun …… tarih ve …… sayılı kararıyla onaylanmıştır.

Enstitü Müdürü

ÖZET

Yüksek Lisans Tezi

**BİYOMETRİK KİMLİK DOĞRULAMA İÇİN
ALAN TABANLI
TUŞA BASMA DİNAMİKLERİ ANALİZİ**

Neşe AGUN

**Anadolu Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı**

Danışman: Prof. Dr. Yusuf OYSAL
2016, 58 Sayfa

Bu tezde, tuşa basma dinamikleri kullanılarak biyometrik kimlik doğrulama sistemi ele alınmıştır. Tuşa basma verisinin toplanması ve modellenmesi için bir sunucu-istemci modeli geliştirilmiştir. Önceki tuşa basma dinamiklerine ilgili çalışmalardan farklı olarak sınıflandırma modeli için alan tabanlı bir özellik kullanılmıştır. Sanal klavye kullanılarak geliştirilen alan tabanlı özellik kullanıcıları ayırt etmede kullanılmıştır. Klavye tasarımının yapılabildiği ve alanların seçildiği bir yazılım geliştirilmiştir. Farklı alanlara sahip çeşitli klavye tasarımlarının oluşturduğu özellikler ile çeşitli kullanıcı grupları üzerinde makine öğrenmesi tekniklerinin kullanıldığı deneyler yürütülmüştür. Kullanıcı grupları 10, 25, 50 ve 100 kişiden oluşmaktadır. Doğruluğun kullanıcı sayısı ile ter orantılı olduğu görülmüştür. Daha yüksek başarı oranları alan tabanlı özelliklerin kullanılması durumunda gözlemlenmiştir. Doğruluğun en yüksek olduğu sonuçlar alan tabanlı özelliklerin kullanıldığı yapay sinir ağları sınıflandırıcısı ile elde edilmiştir.

Anahtar Kelimeler: Biyometrik Kimlik Doğrulama, Tuşa Basma Dinamikleri, Sanal Klavye, Makine Öğrenmesi

ABSTRACT

Master of Science Thesis

**AREA BASED
KEYSTROKE DYNAMICS ANALYSIS
FOR BIOMETRIC AUTHENTICATION**

Neşe AGUN

**Anadolu University
Graduate School of Sciences
Computer Engineering Program**

**Supervisor: Prof. Yusuf OYSAL
2016, 58 pages**

In this thesis, a biometric authentication system for user identification through keystroke dynamics is discussed. A server-client system is developed to gather and model user data. Different from previous research on keystroke dynamics an area based feature is introduced to the classification model. Area based feature is developed and used as a discriminating feature via a virtual keyboard design. A software is developed for designing virtual keyboards and selecting areas. Experiments are conducted by designing different keyboards with different areas for classification of different users among several groups through machine learning techniques. User groups are formed from 10, 25, 50 and 100 persons. It is observed that accuracy is inversely correlated with group size. A higher accuracy is reported when area based feature is used. Top accurate results are obtained with area based features via Multilayer Perceptron classifier.

Keywords: Biometric Authentication, Keystroke Dynamics, Virtual Keyboard, Machine Learning

TEŞEKKÜR

Veri toplama aşamasında zaman ayırarak veri girişlerinde bulundukları için Bilgisayar Mühendisliği ile Havacılık ve Uzay Bilimleri Fakültesi öğrencilerine teşekkür ediyorum.

Lisans öğrenimimde bana emeği geçen ve ayrıca tezin hazırlanmasında tüm desteği ile yardımcı olan değerli danışman hocam Prof. Dr. Yusuf OYSAL'a, bilgisayar bilimleri ile ilgili tüm bilgimi ve motivasyonumu kazandıran değerli hocam Doç Dr. Cüneyt AKINLAR'a, savunmamda yorumlarıyla katkı sağlayan Yrd. Doç. Dr. Mustafa ATANAK'a çok teşekkür ediyorum.

Beni her konuda destekleyen ve yanımda olan diğer yarım canım eşime, hayatım boyunca beni en iyi anlayan herşeyimi paylaştığım canım kardeşime, sonsuz sevgilerini her zaman yanımda bildiğim ve hissettiğim canım anneme ve canım babama sonsuz teşekkür ediyorum.

Canım Aşkım ANNEM'e…

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

SİMGELER VE KISALTMALAR DİZİNİ

ARFF : Attribute Relation File Format

ASCII : American Standard Code for Information Interchange

PIN : Personal Identification Number

WEKA : Waikato Environment for Knowledge Analysis

YKO : Yanlış Kabul Oranı

YRO : Yanlış Red Oranı

1. GİRİŞ

Günümüzde bilişim sistemleri bilginin tutulup verimli kullanıldığı sistemlerdir. Bu bilgilerden önemli bir kısmı hassastır ve kendilerine özgü kişiler tarafından bilinmelidirler. Birçok bilişim sistemine ulaşım kullanıcı adı ve şifre ikilisi ile güvenliklendirilmektedir. Ancak bu ikili bir kez yetkilendirilmemiş kullanıcıların eline geçtiğinde tüm sistemi kullanabilirler ve bu da finansal kayıplara ve bilgi güvenliği açıklarına neden olur. Bu ihmaller, bilgisayar tabanlı işlemler için kullanılan bilgisayar sistemlerinde artmış bir savunmasızlık algısı oluşturur. Bu durum, işlemlerin halka açık olarak olduğu bilgisayar sistemlerinde şiddetli olan bir düşüncedir. Araştırmacılar bu tehditlerin farkındalığında, bilgisayar güvenliğini artırmak için yollar araştırmışlardır. Bu araştırma çabaları yeni bir sektör ortaya çıkarmıştır. Bilgisayar güvenliğini artırma amaçlı çözümler sağlayan bir sektör olan biyometrik sektörüdür (Zhou, 2008).

Kişileri davranışsal veya fiziksel özelliklerini kullanarak ayırt etmeye yarayan bilime biyometri denilmektedir (Daugman, 1993). Biyometri, insanları birbirlerinden ayırt etmede kullanılmaktadır. Konu hakkındaki bilimsel çalışmalar, kullanıcıya ait biyometrik özelliklerin çalınamayacağı ve taklit edilemeyeceği öngörüsü üzerine yapılandırılmıştır (Monrose ve Rubin, 2000). Biyometrik tabanlı kimlik doğrulama, yaşamımızda önemli bir yere sahip güvenli bir kimlik doğrulamadır. Günümüzde biyometri giderek yaygın olarak kullanılır hale gelmektedir (Ergen ve Çalışkan, 2011).

Biyometrik sistemde, kişinin davranışsal ya da fiziksel özelliği analiz edilerek veritabanında bulunan kayıtlar ile karşılaştırmaktadır. Sistemin bu işlemi gerçekleştirirken çok hassas olması, doğru ve birbirini tekrar eden ölçümler yapması gerekmektedir. Çünkü sistem kişinin parmak izini, elini, yüzünü, avuç içini, irisini ya da sesini incelemektedir (Woodward ve ark., 2003).

Fiziksel biyometrik, kişilerin fiziksel ölçülerini dikkate almaktadır. Davranışsal biyometrik ise, kişilerin davranışlarına göre yapılmaktadır. Her sınıf biyometriğin avantajları ve dezavantajları vardır. Fizyolojik biyometrikler daha güçlü ve daha güvenlikli olarak algılanmakta ya da düşünülmektedir. Herkesin kendine özgü parmak izi vardır ve bu ölçüler dört dörtlük olarak düşünülmektedir.

Ama parmak izi yanıltılabilir. Parmak izi tarayıcıları daha güvenilir ve ucuz olmasına rağmen, kullanılıp aşınıp yıpranabilirler. Yılda bir kez değiştirilmesi gerekir ve uzaktan erişim sistemlerine konuşlandırılması zordur. İris tarayıcıları gürültüye daha toleranslıdır ve daha doğrudur, ancak pahalıdır. Davranışsal biyometrikler kendi halinde, dikkat çekmezler, ama daha çok hataya düşebilir gözükmektedirler. İmzaların sahtesi yapılabilir, konuşma taklit edilebilir.

Kullanıcılar kullandığı bir sistemin güvenliğinden emin olmak isterler. Sistem tarafında ise güvenlik için, çoğunlukla bir kullanıcı adı, parola, güvenlik onay imgesi kullanılmakta veya kredi kartına ait bilgilerle birlikte şifre ile sağlanmaktadır. Fakat bu sayılan güvenlik doğrulama bilgileri sahte kullanıcıların ellerine geçerse izinsiz kullanabilirler. Günümüzde biyometrik teknolojiler kimlik doğrulamada daha güvenilir ve verimli oldukları için yaygınlık kazanmışlardır (Shanmugapriya ve Padmavathi, 2009).

Tuşa basma dinamiği yüksek güvenlikli, parmak izi sistemleri ile eşit düzeyde, kullanım için hiçbir şeyden ödün vermeye gerek duymayan ve pahalı donanım gerektirmeyen davranışsal bir biyometriktir (Revett ve ark., 2007). Tuşa basma dinamiklerinin biyometri özelliği; kullanıcıların bilgisayar sistemlerine giriş yaparlarken klavyede tuşlara basma özellikleri ile ayırt edilebilmesini ifade etmektedir. Dolayısıyla biyometri kullanıcıdan kullanıcıya göre değişiklik gösterebilmektedir. Tuşa basma dinamiği de bir kullanıcının yazma sitilini yakalamaktadır. Yazma sitili kişiye özgüdür. Yazma sitili bir sisteme giriş yaparken kullanıcı adı ve şifresinin ne kadar hızlı girildiği, bir tuşa ne kadar süre basıldığı, ikili/üçlü tuşlar arasında ne kadar zaman geçirildiği gibi farklı özellikleri kapsamaktadır. Yazma sitili, bilinçli yapılan bir eylem olmadığından motor kontrol sisteminin bir refleksi olarak kabul edilmiştir. Bundan dolayı tuşa basma eyleminin taklit edilmesi fizyolojik olarak mümkün görülmemektedir (Anonim, 2015c).

1.1. Önceki Çalışmalar

1990 yılında Bleha ve ark. (1990) yaptıkları çalışmada, kullanıcı tanımlama analizi için, metin olarak sadece kullanıcı adını kullanmışlardır. Kullandıkları yöntemler ise en küçük uzaklık sınıflandırıcısı ve Bayes sınıflandırıcısıdır.

Kullanıcı eğer iki sınıflandırıcıdan da geçemezse kullanıcının sisteme kabulü yapılmamaktadır. Yapılan deneylerde 10 gerçek kullanıcı ve 22 sahte kullanıcı kullanılmıştır. Alınan sonuçlarda sahte kullanıcıların kendilerini sisteme kabul ettirme ihtimallerinin oldukları ve bilgisayar kullanmaya yatkın olmayan kullanıcıların daha çok hata yaptıkları ortaya çıkmıştır.

Aynı yıl Joyce ve Gupta (1990) yaptıkları çalışmada her kullanıcı için kullanıcı adı, şifre, ad ve soyad dörtlüsünden bir imza oluşturan sistem hazırlamışlardır. Testleri 33 kullanıcı ile yapmışlardır. Her kullanıcı bu yukarıda sayılan dörtlüyü kullanarak sisteme 8 kez giriş yaparak kendi imza referanslarını oluşturmuştur. Referans imza oluşturulduktan sonra her kullanıcı sisteme 5 kez giriş yapmıştır. Toplamda 165 (5 x 33) giriş elde edilmiştir. Bir oturum için referans imzalar ve girişler toplanmıştır. Rastgele 6 kullanıcı seçilmiştir. Geriye kalan 27 kullanıcı her bir 6 kullanıcının hesaplarına 5 er kez giriş yapmaya çalışmıştır. Yazarların ellerinde 810 (27x6x5) tane sahte kullanıcı atağı bulunmaktadır. Toplamda 975 deneme olmuştur. 810 tane sahte kullanıcı içerisinden 2 tanesi sistemden geçmiştir, 165 tane gerçek kullanıcı sistemden geçememiştir.

1993 yılında Bleha ve Obaidat (1993) tarafından bilgisayar kullanıcılarının kimliklerini doğrulamak için Doğrusal Algılayıcı (Linear Perceptron) algoritması kullanılmıştır. Kullanıcıların tuşları basma süreleri veri olarak toplanmıştır. Bunun için 10 gerçek kullanıcı ve 14 sahte kullanıcı kullanılmıştır. Veri toplama işlemi 8 haftada tamamlanmıştır. Toplanan verinin yarısı sistemi eğitmek için kullanılmış, diğer yarısı sistemi test etmek için kullanılmıştır. Sonuç olarak %8,5 gibi bir hata oranı elde edilmiştir.

1997 yılında Obaidat ve Sadoun (1997) istatistik ve yapay sinir ağları kullanan birçok sınıflandırıcı sistemi geliştirmeye çalışmışlardır. Çalışmalarda bir tuşa basma süresi ve iki tuşa basma arasındaki süreler üzerine yoğunlaşılmıştır. Bir tuşa basma süresi önceden kullanılmadığı için yeni bir özelliktir. Deneylerde 15 gerçek ve 15 sahte kullanıcı ile çalışılmıştır. Yapılan deneylerle, tuşa basma süresinin tuşlar arası geçiş süresinden daha ayırt edici bir özellik olduğu ve yapay sinir ağları kullanılarak yapılan ölçümlerin istatistik yöntemler kullanılarak yapılan ölçümlerden daha iyi sonuçlar verdiği ortaya çıkmıştır. En başarılı örüntü tanıma yöntemi Bayes kuralıdır. En başarısısz olan Cosine Ölçüm algoritmasıdır. En

başarılı yapay sinir ağları yöntemleri ise LVQ, RBFN ve bulanık ARTMAP tir. En başarısız yapay sinir ağları yöntemleri ise CPNN ve BP dir.

2000 yılında Monrose ve Rubin (2000) çalışmalarında istatistiksel sınıflandırıcıları kullanmışlardır. Verileri 11 ay gibi bir zamanda 63 kullanıcı ile toplamışlardır. Bu çalışmada önceki çalışmalarda yapılmayan bir şey yapılmıştır. Katılımcılar uygulamayı kendi bilgisayarlarına indirip verileri elde etmişlerdir ve yazarlara e-posta yolu ile dönüş yapmışlardır. Öklid uzaklık ölçüsü, ağırlıklandırılmamış olasılık ölçüsü ve ağırlıklandırılmış olasılık ölçüsü sınıflandırıcıları kullanılmış ve karşılaştırılmıştır. En iyi sonuç %87.18 ile ağırlıklandırılmış olasılık ölçüsü sınıflandırıcısına aittir. Bu performansı %85.63 ile ağırlıklandırılmamış olasılık ölçüsü ve %83.22 ile öklid uzaklığı yaklaşımları takip etmektedir. Veriler kullanıcılar tarafından yapılandırılmış sabit bir metin üzerinden elde edilmiştir. Serbest metin üzerinde aynı iyi performans gösterilememiştir. Bayes sınıflandırıcıları ile de çalışmalar yapılmıştır ve bu sınıflandırıcı %92.14 doğruluk oranına sahip olmuştur.

Aynı yıl Cho ve ark. (2000), değişik bir yaklaşım kullanarak tuşa basma ritimleri ile kimlik doğrulama olayına katılmışlardır. Sistemi kullanmak için kullanıcılardan en az 7 karakterli, kendileri için yakın olan bir şifre seçmeleri istenmiştir. Sistemi denemeye 25 kullanıcı ile başlanmış, ancak 4 kullanıcının verileri çok farklı olduğu için yok sayılmış ve 21 tecrübeli kullanıcı ile devam edilmiştir. Her bir kullanıcı istenilen şifreyi 150-400 kez girmişlerdir. Son 75 zaman vektörü test için toplanmıştır. Otomatik ilişkilendiricili çok katmanlı algılayıcı (Autoassociative MultiLayer Perceptron) algoritması kullanılmıştır. MLP yaklaşımının k-NN (Nearest Neighbour) yaklaşımından daha etkili olduğu görülmüştür.

2002 yılında Bergadano ve ark. (2002) yaptıkları çalışmada 154 birey üzerinde %4 yanlış red oranı ve %0.01 yanlış kabul oranı elde etmişlerdir. Yapılan çalışmada, her kullanıcıdan verilen sabit metni girmeleri istenmiştir ve kullanıcıların yazma hataları yapmalarına da izin verilmiştir. Gerçek kullanıcıların sahte kullanıcılardan ayırt edilebilmesi için parametreler üzerinde herhangi bir eğitim modeli kullanılmamıştır. Kullanıcılardan girilmesi istenen metin şifre

tabanlı kimlik doğrulama sistemlerine göre uzun tutulmuştur. Bununla birlikte üçlü harf zamanlaması uyarlanırken ikili harf gecikme zamanı kullanılmamıştır.

2003 yılında Guven ve Soğukpınar (2003) yaptıkları çalışmada, vektör tabanlı en küçük uzaklık sınıflandırıcısına benzeyen bir metot önermişlerdir. Önerilen metotta 4 ana basamak bulunmaktadır. Bunlar; tanımlama basamağı, onaylama basamağı, karar verme basamağı ve güncel tutma basamağından oluşmaktadır. Tanımlama basamağında, kullanıcılardan kullanıcı adlarını ve şifrelerini girmeleri istenmiş ve böylece her bir kullanıcının tuş vuruşu parametreleri toplanmıştır. Onaylama basamağında, önerilen algoritma ile kullanıcıların parametreleri önişlemden geçirilmiş ve yeni parametreler üretilmiştir. Karar verme basamağında, bir önceki basamakta hesaplanan yeni değerler karar verme fonksiyonuna aktarılmıştır. Karar verme fonksiyonu bu parametreleri kullanarak bir sayısal değer üretmektedir ve bu üretilen sayısal değer veritabanındaki değer ile karşılaştırılmaktadır. Eğer değerler uyuşuyorsa kullanıcı sisteme kabul edilir aksi takdirde sistem kullanıcıyı reddeder. En son basamak olan güncel tutma basamağında eğer kullanıcı sisteme giriş yapabilmiş ise veritabanındaki bilgiler güncellenir ve bu güncellenen parametreler kullanıcının bir dahaki sisteme girişi işleminde kullanılmak üzere saklanır. Bu ise sistemin her zaman kullanıcının tuş vuruş ritimlerini öğrenmekte olduğunu gösterir.

2004 yılında Yu ve Cho (2004) çalışmalarında Cho ve arkadaşlarının (2000) yaptıkları çalışmaya yeni düzenlemeler, iyileştirmeler getirmek istemişlerdir. 2000 yılında yapılan bu çalışmanın bazı eksiklikleri ve zorlukları vardır. Bunlar, modelin eğitiminin çok zaman alması, veri ön işlemesinin makine ile yapılmaması, büyük bir veri kümesine gereksinim duyulması gibi sıralanabilir. İlk olarak destek vektör makinasını (SVM) denemişlerdir. Yapay sinir ağları yöntemi performansı ile hemen hemen aynı bir performansa sahip olmuştur. Ama eğitim süresi ondan 1000 kat daha az zaman almıştır. İkinci olarak özellik çıkarmak için önerilen GA-SVM modelini kullanmışlardır. SVM doğrulukta da öğrenme hızında da daha iyi sonuçlar vermiştir. Üçüncü olarak eğitim verisinin yetersizliği üzerinde durulmuştur. Bunun için FS-Ensemble önerilmiştir. Sonuç olarak önerilen yeni yöntemlerin yukarıda bahsedilen problemleri çözdüğü gözlenmiştir.

2007 yılında Lee ve Cho (2007) çalışmalarında, kimlik doğrulama işleminin doğruluğunu arttırmak için, sahte kullanıcı örüntüleriyle tekrar eğitim yapan yeni bir algılayıcı sistem kurmaya çalışmışlardır. Deney sonuçlarına göre tekrar eğitim yapmak kimlik doğrulamanın performansını arttırmıştır. Yeni yapılan öğrenme vektör niceleme algılayıcısının diğer algılayıcıları performans açısından aşmakta olduğu gözlenmiştir. Duruş süresi ve uçuş süresi özelliklerinin her ikisi de analizlerde kullanılmıştır. Ayrıca elimizde çalışılacak olan sınırlı sayıda örüntünün olduğu gibi durumlarda şifre, tuş vuruşu örüntülerinin tekil ve tutarlı olması için en iyi yoldur.

Aynı yıl Teh ve ark. (2007) tarafından yapılan çalışmaya 50 kullanıcı katılmıştır. Her kullanıcı kullanıcı adını, şifresini ve belirlenen sabit metni 10 kez sisteme giriş yapmışlardır. İlk olarak aynı tuşu basış anından bırakış anına kadar geçen süreyi (duruş süresi) ve bir tuşu bırakış anından ikinci tuşu basış anına kadar geçen süreyi (uçuş süresi) kayıt etmişlerdir. Daha sonra elde edilen verilerin ortalamalarını ve standard sapmalarını hesap etmişlerdir. Her kullanıcı için biri duruş süresi diğer üçü uçuş süresi olmak üzere dört çeşit özellik bulunmaktadır. İşlenmiş veriler öncelikle Gauss olasılık yoğunluk fonksiyonuna tabi tutulmuşlardır. Daha sonra yazarların bulduğu yöntem olan benzerlik ölçüm yönü ile işlenmişlerdir. İki olay da birleştirilerek %6.36'lık bir eşit hata oranına ulaşılmıştır. Sonuç olarak duruş ve uçuş sürelerini birleştirerek yapılan kombinasyonlar ile daha iyi sonuçlar elde edilmiştir.

Yine aynı yıl Subramaniam ve ark. (2007) var olan kimlik doğrulama metodolojisine, tuşa basma analizi tekniğini katarak daha güvenli bir sistem yapmaya çalışmışlardır. Uygulamalarında sabit bir kullanıcı adı ve şifre kullanmışlardır. Bu sabit girilecek metinler arasındaki ikili, üçlü, dörtlü tuşlar arasındaki geçen süreyi baz almışlardır. Veritabanında kullanıcıya ait olan tuşa basma ritmindeki sıralama ve sisteme giriş yapma anında oluşan sıralama da bir özellik olarak sisteme katılmıştır. Bu özelliğin de dikkate alınmasının daha güvenilir bir sistem yapmada bir etken olduğu görülmüştür.

Yine aynı yıl Revett ve ark. (2007) yaptıkları çalışmalarda 50 kullanıcı kullanmışlardır. Kullanıcıları 20 gerçek kullanıcı ve 30 sahte kullanıcı olmak üzere ikiye bölmüşlerdir. Her kullanıcıdan 6-15 karakter arasında değişen kullanıcı adı ve

şifre belirlemeleri istenmiştir. Kullanıcılar seçtikleri bu giriş verilerini kaydolma safhasında 10 kez girmişlerdir. Veriler 14 günde ve her gün içerisindeki üç oturumda toplanmıştır ve ortalamaları alınmıştır. Sahte kullanıcı grubundan bir haftada 20 hesaba 100 kez giriş yapmaları istenmiştir. Bir sahte kullanıcı bir hesaba 4 kez girmeye çalışmıştır, yani bir hesaba 120 (4x30) kez girilmeye çalışılınılmıştır. Rastgele 100 sahte kullanıcı girişimi seçilmiştir. Bunlar da yanlış kabul oranı hesaplaması için kullanılmıştır. Ayrıca her gerçek kullanıcıdan hesaplarına 100'er kez girmeleri istenmiştir. Bu da yanlış red oranı hesaplaması için kullanılmıştır. Sonuç verisinde 2000 sahte kullanıcı ve 2000 gerçek kullanıcı girişimi olmuştur. Bu veri olasılık yapay sinir ağı (PNN) algoritmasında kullanılmıştır. Ayrıca bu algoritmanın biraz modifiye edilmiş olanı da kullanılmıştır. Bu ikinci modifiye edilmiş olandan daha iyi sonuçlar elde edilmiştir. Veri kümesi eğitim ve test için rastgele olarak 50-50 bölünmüştür. MLFN tekniği ile PNN karşılaştırılmıştır. Sınıflandırma doğruluğunda ve eğitme zamanında PNN'nin daha iyi olduğu sonucuna varılmıştır. Ayrıca türetilmiş özelliklerin ilk özelliklere göre daha iyi sonuç verdiği gözlenmiştir.

2009 yılında Killourhy ve Maxion (2009) yaptıkları çalışma için 51 kullanıcı seçmişlerdir, her bir kullanıcı 400 kez belirlenen metni girmiştir. Kullanıcıları ayırt etmede sabit bir şifre (.tie5Roanl) metni kullanılmıştır. Her birinde 50 tekrar olmak üzere 8 oturum düzenlenmiştir. (.tie5Roanl) sifresi kuvvetli bir 10 karakterli sifre olması sebebiyle seçilmiştir. Ayrıca on dört tane sınıflandırıcı dedektör ile gözlemler yapılmıştır. En iyi sonuçlar, Manhattan ve Mahalanobis dedektörlerinde saptanmıştır.

2011 yılında Abualgasim ve Osman (2011), PIN (rakamsal şifre) ve şifrelerin güvenliği için tuşa basma dinamiğini kullanan bir uygulama geliştirmişlerdir. Uygulamalarında daha iyi sonuç verdiği için istatistiksel sınıflandırma tekniğini kullanmışlardır. Şifrenin uzunluğuna göre alınan sonuçlardaki başarının değiştiğini gözlemlemişlerdir.

Yine aynı yıl Shanmugapriya ve Padmavathi (2011), çalışmalarına yeni bir özellik katarak, doğruluğun ne kadar değiştiğini görmeye çalışmalardır. Bu yeni özelliğe sanal klavye özelliği demişlerdir. Klavye düzenindeki aradaki tuşlar için karmaşıklığı 0, diğer tuşlar için ise 1 olarak almışlardır. Tuş pozisyonlarını ve

tuşların birbirleri arasındaki uzaklığı gözönüne almışlardır. 103 kullanıcı ile çalışmışlardır ve belirledikleri 3 kelimeyi her bir kullanıcıya 26 kez giriş yaptırmışlardır. Yeni özelliğin yannda kullanılan diğer özellikler ise bir tuşa basma bırakma arasında geçen zaman (dwell time), iki tuş arasında bir tuşu bırakıp diğer tuşu basma arasında geçen zaman (flight time), 2ngraph ve 3ngraph'dan oluşmaktadır. Performans, genetik algoritması ve geri yayılımlı yapay sinir ağları ile ölçülmüştür. Katılan yeni özellik ile doğruluğun %1 oranında arttığı, öğrenme ve test zamanlarında da bir azalma olduğu sonucuna varılmıştır.

2012 yılında Zhong ve ark. (2012) çalışmalarında, Killourhy ve Maxion (2009)'un yaptıkları çalışmaya yeni bir metrik katarak daha da iyileştirmeye çalışmışlardır. Yapılan çalşmada Manhattan ve Mahalanobis sınıflandırıcılarını karşılaştırmışlardır. Bu sınıflandırıcılardan bir sınıflandırıcının diğerinin tümleyicisi olarak alındığı yeni bir yaklaşım katılmıştır. Getirdikleri bu yeni yaklaşım ile sistem öncekinden daha iyi sonuç vermiştir.

2014 yılında Can ve Alagöz (2014) de Killourhy ve Maxion (2009)'un kullandıkları veri kümesini kullanarak bir çalışma yapmışlardır. Özellik olarak bir tuşa basma zamanı, bir tuştan diğer tuşa geçiş zamanı ve iki tuşu basma süresi arasında geçen zaman kullanılmıştır. D değişkenli Gauss, kNN ve karar ağacı algoritmaları üzerinde çalışılmıştır. Önerilen algoritmalar arasında en başarılı sınıflandırma doğruluk oranı sonucu veren k en yakın komşu algoritması olmuştur. k değeri 8 iken başarılı sonuçlar alınmıştır.

Aşağıdaki tabloda 1990 yılından günümüze yapılan çalışmaların kullandıkları sınıflandırma teknikleri ve üzerinde çalıştıkları kullanıcı sayılarına göre YKO ve YRO perfromans değerleri özetlenmiştir.

Çizelge 1.1. Önceki çalışmalar.

Çalışma	Yıl	Sınıflandırma Tekniği	Kullanıcı Sayısı	(%) YKO	(%) YRO
Bleha ve Ark.	1990	İstatistiksel	32	2.8	8.1
Joyce & Gupta	1990	İstatistiksel	33	0.25	16.36
Bleha & Obaidat	1993	Örüntü Tanımlama	24	8	9
Obaidat & Sadoun	1997	İstatistiksel	30	0.7	1.9
		Yapay Sinir Ağları		0	0
Monrose & Rubin	2000	İstatistiksel	63	7.9 (eşit hata oranı)	
Cho ve Ark.	2000	Yapay Sinir Ağları	21	0	1
Bergadano ve Ark.	2002	Yapay Sinir Ağları	154	0.01	4
Güven & Soğukpınar	2003	İstatistiksel	12	1	10.7
Soğukpınar & Yalçın	2004	İstatistiksel	40	0.6	60
Yu & Cho	2004	Yapay Sinir Ağları	21	0.3	3.69
Gunetti & Picardi	2005	Yapay Sinir Ağları	205	0.005	5
Lee & Cho	2007	Yapay Sinir Ağları	21	0.43 (eşit hata oranı)	
Teh ve Ark.	2007	İstatistiksel	50	6.36 (eşit hata oranı)	
Subramaniam ve Ark.	2007	İstatistiksel	36	3.9 (eşit hata oranı)	
Revett ve Ark.	2007	Yapay Sinir Ağları	50	0.04	0.03
Chuda & Durfina	2009	İstatistiksel	15	8.4	2.5
				3.6	4.7
Killourhy & Maxion	2009	Yapay Sinir Ağları	51	0.096 (eşit hata oranı)	
Abualgasim & Osman	2011	İstatistiksel	26	0.13	0
Shanmugapriya & Padmavathi	2011	Yapay Sinir Ağları	103	9,33	
Zhong ve Ark.	2012	Yapay Sinir Ağları	51	0.084 (eşit hata oranı)	
Can & Alagöz	2014	Örüntü Tanımlama	51	0.8 (eşit hata oranı)	

1.2. Tez Organizasyonu

Bu tez çalışmasında kullanıcının klavye üzerindeki tuşlara basıp bırakma, tuşlar arasındaki geçiş zamanlamaları ve tuş/parmak kombinasyonları üzerinden etkili bir davranışsal kimlik doğrulama modeli geliştirilmiştir. Bu çalışmada, önceki çalışmalardan farklı olarak, tuşlara basma dinamiklerinde kullanılmış olan özelliklere yeni özellikler katılarak doğruluğun artırılmaya çalışılması amaçlanmıştır. Yeni özellik olarak da alan tabanlı ayırt etme seçilmiştir. Klavye sanal olarak alanlara bölünmüş ve kullanıcıların alanlar üzerindeki davranış özelliklerinden yola çıkılarak daha iyi bir ayırt edicilik sağlanmaya çalışılınılmıştır. Bu alanda ülkemizde yapılacak çalışmalara da kaynaklık ve öncülük etmesi beklenmektedir.

Tez altı bölümden oluşmaktadır. Bölümlerin içerikleri aşağıda kısaca açıklanmıştır.

Birinci bölümde, tuşa basma dinamiğinin ne olduğu ve tuşa basma dinamiklerine neden gereksinim duyulduğu ve bu çalışmada getirilen yeni özellik açıklanmaya çalışılmıştır. Ardından önceki çalışmalara değinilmiştir. Her çalışmada kullanılan özellikler, yöntemler ve sonuçları açıklanmıştır.

İkinci bölümde, kimlik doğrulamaya bir giriş yapılmıştır. Yüz tanımadan tuş vuruşu tanımaya kadar bir çok biyometrik kimlik doğrulama yöntemlerinden ve biyometrik sistemlerin özelliklerinden bahsedilmiştir.

Üçüncü bölümde, tuşa basma dinamiğine daha ayrıntılı bir bakış açısı ile bakılmıştır. Biyometrik sistem özelliklerinin tuşa basma dinamiklerindeki açıklamaları yapılmıştır. Ayrıca kullanılagelen doğruluk ölçüm metrikleri ve tuşa basma dinamiklerinde bu metriklerin ne anlama geldiği açıklanmıştır.

Dördüncü bölümde, tezde gerçekleştirilen çalışma aşama aşama anlatılmıştır. Sistem mimarisinin çalışması, verilerin toplanması, getirilen yeni özellik için sanal klavyenin yapılması ve kullanılması, toplanan verilerin işlenerek özellik vektörlerinin çıkarılması ile makine öğrenmesinde kullanımı açıklanmıştır.

Beşinci bölümde, kullanılan yapay zeka yöntemleri ve makine öğrenmesi teknikleri ile ilgili bilgiler verilmiştir. Bu bölümde Naïve Bayes, Karar Ağaçları ve Yapay Sinir Ağları ile ilgili genel kavramlar anlatılmıştır.

Altıncı bölümde, iki ayrı gruptan toplanan verilerden elde edilen deneylerin nasıl yapıldığı, geliştirilen alan tabanlı özelliklerin doğruluğa olan etkisi ve elde edilen tüm sonuçlar çizelgeler halinde sunulmuştur. Ayrıca kullanıcı sayıları değiştirilerek sonuçlara etkileri gözlemlenmiştir. İleride yapılacak çalışmalar için yeni öneriler verilmiştir.

2. KİMLİK DOĞRULAMA

Kimlik doğrulama işlemi kullanılan sisteme kim olduğunuzu söylemeniz, sisteme kendinizi tanıtmanızdır. Bilgisayar kullanımında kimlik doğrulama işlemi, kullanıcının kim olduğuna karar verme işlemine denir. Günümüzde de gelişen teknolojilerle bilgisayar kullanımında kimlik doğrulama işleminin önemi artmıştır (Karnan ve ark., 2011). Kimlik doğrulama kullanıcılar için çoğu sistemlerde bankacılıktan sosyal hesaplara kadar gerekli bir işlemdir. Her kullanıcı sistemin güvenliğinin yeterli bir seviyede olmasını ister. Kimlik doğrulama işlemini; Bilgi tabanlı, Nesne tabanlı ve Biyometrik tabanlı olmak üzere üçe ayırabiliriz.

Şekil 2.1. Kimlik doğrulama.

2.1. Bilgi Tabanlı Kimlik Doğrulama

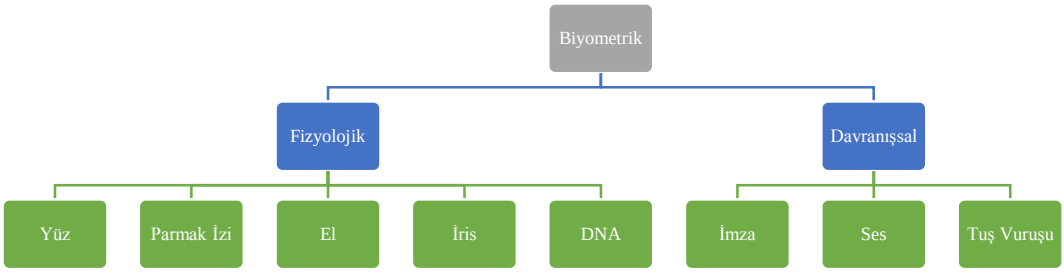
Bilgi tabanlı kimlik doğrulamada, kullanıcılar sisteme kendilerini ellerinde bulunan kullanıcı adı, şifre veya PIN gibi bilgilerle tanıtmaktadırlar. Kullanıcılar sisteme bu bilgileri girerek erişebilmektedirler. Ancak öncelikle sistem veri tabanında bu bilgilerin kayıtlı olması gerekmektedir. Sistemde kayıtlı olan bilgilerle, sisteme giriş yapmak isteyen kullanıcının girdiği bilgiler eşleşirse, sistem yöneticisi kullanıcının girişine onay vermektedir. Ancak kullanıcı adı, şifre veya PIN bilgisinin başka kişilerin eline geçebilmesi ya da bu bilgilerin unutulması gibi durumlar bu tür sistemlerin dezavantajlarını oluşturmaktadır (Oysal ve ark., 2012).

2.2. Nesne Tabanlı Kimlik Doğrulama

Nesne tabanlı kimlik doğrulamada, sisteme erişmek için kullanıcıların ellerinde anahtar veya manyetik kart gibi bir nesne bulunmaktadır. Kullanıcılar bu nesneyi kullanarak sisteme erişebilmektedirler. Nesne içerisinde kullanıcının doğru kullanıcı olduğunu gösteren bilgiler içermektedir. Ancak bu tür sistemlerde de nesnelerin unutulma, kaybolma ya da çalınma gibi durumları bulunmaktadır (Oysal ve ark., 2012).

2.3. Biyometrik Tabanlı Kimlik Doğrulama

Biyometrik tabanlı kimlik doğrulamada, diğer iki kimlik doğrulama türünün aksine kullanıcılar ya da kişiler herhangi bir şifreye ya da anahtara gereksinim duymadan sisteme kendilerini tanıtabilmektedirler. Bu tür sistemler, kişilerin fiziksel veya davranışsal özelliklerinden faydalanarak sisteme kabullerini yapmaktadırlar. Ayrıca, fiziksel veya davranışsal özelliklerin kaybolma, çalınma ya da unutulma tehlikesi bulunmamaktadır (Oysal ve ark., 2012). Biyometrik tabanlı kimlik doğrulama kendi içerisinde, fiziksel kimlik doğrulama ve davranışsal kimlik doğrulama olarak iki alt gruba ayrılmaktadır. Bu gruplar da kendi içlerinde farklı yöntemlerde kendilerini göstermektedirler.



Şekil 2.2. Biyometrik kimlik doğrulama.

2.3.1. Yüz Tanıma

Yüz biyometrisi, biyometrik teknolojinin hızlı büyüyen alanlarından birisidir. Bu yöntemin diğer yöntemlerden farkı diğer yöntemlere göre daha zor

olmasıdır. Bunun nedeni ise yüz tanıma sisteminde cihazla kişinin birebir temasının olmamasıdır. Gelişmekte olan teknolojiler sayesinde yüz tanıma sistemi bir fotoğraftaki ya da bir videodaki yüzün tamamını algılayabilmektedir. Algılama adımından sonra farklı yöntemlerle yüz tanıma işlemi yapılabilmektedir. Bilgisayar yazılımı algılama yapılan yüzdeki girinti ve çıkıntıları okuyarak bireyi belirlemeye çalışmaktadır. İnsan yüzünde 80 düğüm noktası vardır. Ancak yazılım, belirleme yapabilmek için 15-20 noktaya gereksinim duymaktadır (Anonim, 2015a). Yüzde özelliklerin sayısının fazla olması nedeniyle farklı özellikleri kullanan çeşitli yüz tanıma yöntemleri (Eigenfaces, Fisherfaces, Hidden Markov Models, Evolutionary Pursuit, vb.) kullanılmaktadır. Bu yöntemlerin tanımladığı ağız, burun, göz, kaş, çene, elmacık kemikleri gibi çizgiler arasındaki mesafeler kodlanarak veri tabanına yazılmaktadır. Veri tabanında tutulan bu kayıt kullanıcı kamera karşısına geldiğinde alınan yeni görüntüyle karşılaştırılmaktadır. (Musayeva ve Yahyayev, 2015). Yüzün dudaklar ve şakaklar arasındaki kalan kısım yüz tanımada önemli olmaktadır. Bu kısım kişide ömür boyu hiç değişmemektedir (Anonim, 2015a).

2.3.2. Parmak İzi Tanıma

Parmak uçları değişik deri yapısından oluşmaktadır. Her insanın parmak uçlarındaki deri, girintili çıkıntılı birbirinden farklı desenlerden oluşan bir yapıya sahiptir. Parmak uçlarındaki bu desenlerin temas sonucu yüzeylerde bıraktığı ize parmak izi denilmektedir. Parmak izi değişmeyen ve benzersiz olan biyometrik ölçülerinden biridir. (Musayeva ve Yahyayev, 2015). Üst deri tabakası sırtlardan, alt deri tabakası vadilerden oluşmaktadır. Parmak izinin benzersizliği sırtların desenlerinden ve kırışıklıklardan anlaşılabilmektedir. Bir parmak izi kemer, çadır kemer, sol döngü, sağ döngü, ağırşak gibi beş kalıptan oluşmaktadır. Bu kalıplardan döngüler parmak izinin %60'ını, ağırşaklar %30'unu ve kemerler %10'unu oluşturmaktadır. Genellikle iki parmağın aynı dermal sırt karakteristiklerini taşımadığı varsayılmaktadır. Bu varsayımdan yola çıkılarak parmak izi eşsiz kabul edilmektedir (Anonim, 2015a).

2.3.3. El Tanıma

Kullanıcıları ellerinin şeklinden tanıma yaparak kimlik doğrulama sağlayan bir yöntemdir. Bu yöntemde parmakların uzunluğu, genişliği ve avuç içi özellikleri kullanıcıyı tanımlamada kullanılabilen başlıca özelliklerdendir. Doğrulama sürecinin uzun sürmesi ve maliyetinin de yüksek olması bu yöntemin dezavantajları arasında yer almaktadır (Şamlı ve Yüksel, 2009).

2.3.4. İris Tanıma

Kimlik belirlemede insanları ayırt etmek için iris tanıma yönteminin kullanılmasının nedeni göz bebeklerinin her insanda farklı olmasından dolayıdır ve bu durum kişide ömür boyu değişmemektedir. Her insanda iris desenleri farklı olmaktadır. İris tanıma sistemlerinde ayrıntılı, zengin, karmaşık yapıların görüntülerini oluşturmak ve dışbükey korneadan speküler yansımayı azaltmak için kamera teknolojisi ve ince IR aydınlatma kullanılmaktadır (Anonim, 2015a). İris tanıma sistemlerinde temasa gerek kalmadan kullanıcı, tarama yapan CCD kameralarına 15-20 cm uzaklıkta durarak kendini tanıtabilmektedir (Anonim, 2015b). Bu nedenle iris tanıma sistemlerinde yüz tanıma sistemlerinden farklı olarak kullanıcının izni olmadan kimlik doğrulaması yapılamaz. İris tanıma etkinliği nadiren gözlük veya kontakt lensten etkilenmektedir (Musayeva ve Yahyayev, 2015).

2.3.5. DNA Tanıma

Deoksi Ribonükleik Asit (DNA) tanıma bir kişiyi tanımlamada kullanılan en kesin yöntemdir. DNA insanların fiziksel ve zihinsel kimliğini tanımlayan bir yapıdır. Bir insanın ikizi olmadığı sürece başka bir insanla aynı gen setine sahip olması mümkün değildir (Anonim, 2015a). Kan, saç, tırnak, ağız bezleri, kan lekesi, tükürük ve vücuda herhangi bir zamanda temas eden herhangi bir kaynak gibi pek çok kaynak DNA kontrolünü sağlamaktadır. DNA kontrolü çok güvenlikli kimlik doğrulama yöntemlerinden biridir ancak biyolojik dokunun kirlenmesi, 24 saat

içerisinde işlemin yapılması zorunluluğu ve maliyetinin yüksek olması gibi dezavantajları bulunmaktadır (Şamlı ve Yüksel, 2009). Bu dezavantajları ortadan kaldırmak için son zamanlarda DNA çipler ve mikrodiziler kullanıla gelmiştir. (Özkaya ve Sağıroğlu, 2015). Özellikle cinayet işlerinde, babalık davalarında DNA tanıma yaygın olarak kullanılan bir yöntemdir (Musayeva ve Yahyayev, 2015).

2.3.6. İmza Tanıma

İmza kimlik doğrulamasında belki de en sık kullanılan yöntemdir. İmza bir kişinin herhangi bir belgeyi yazdığını, okuduğunu ve onayladığını belirtmek için belge sonuna kişi tarafından kullanılan kelime ve sembollerdir. Hukuki açıdan önemi oldukça büyüktür. Kimlik doğrulamasında belki de en sık kullanılan yöntem olan bu imzanın gerçekten o kişi tarafından atılıp atılmadığının belirlenmesi çok önemlidir (Şan, 2013). İmza tanıma sistemleri imzayı iki farklı açıdan belirlemeye çalışmaktadırlar. İlki davranışsal özellikler, diğeri desen özellikleridir. İmza bazı durumlarda desen olarak taklit edilebilir. Ancak imzalama süresi, hızı, ivmesi, kalemin basım şiddeti gibi imzanın atış şeklini belirleyen davranışsal özellikler her kişi de farklılık göstermektedir. Bu farklılık gösteren özelliklerin ayırt edilmesinde gerçek zamanlı imza tanıma yöntemi kullanılmaktadır. Gerçek zamanlı tanıma yönteminde matematiksel veya algoritmik yaklaşımlar yer almaktadır. Gerçek zamanlı olmayan tanıma sistemleri imzayı sadece desen özelliklerine göre analiz edebilmektedir. Akıllı bir imza tanıma sisteminin imzaları iyi ayırt edebilmesi için çok sayıda örneğe ihtiyacının olması imza tanıma sistemlerinin bir dezavantajıdır. (Oysal ve ark., 2012).

2.3.7. Ses Tanıma

Ses yolunun yapısı her insanda kendine özgü olmaktadır. Dolayısıyla, her insanın konuşurken çıkardığı sesler de birbirinden farklıdır. Bu sebeple, konuşma sonucu oluşan ses verisi, güvenlik sistemlerinde kimlik doğrulama için kullanılabilmektedir (Oysal ve ark., 2012).

2.3.8. Tuş Vuruşu Tanıma

Tuşa basma dinamiği kullanıcının tuşa vurma sitilini tutan, kullanıcının tuşları nasıl kullandığı ile kullanıcıyı tanıyan güçlü bir davranışsal biyometrik kimlik doğrulamadır. Bilgisayar sistemlerine giriş işlemi genellikle kullanıcı hesaplarının kullanıcı adı ve şifresi ile kontrol edilmektedir. Eğer giriş için gerekli bilgiler yanlış kişilerin ellerine geçerse bu sistemin güvenliği için büyük bir risk demektir. Biyometrik sistemler, mesela parmak izi tanıma, güçlü güvenlik içermektedirler. Ancak bu sistemler için gerekli donanımlar oldukça pahalıdır. Fakat tuşa basma dinamiği sistemi ek bir donanım gerektirmemektedir, sadece bir bilgisayar ve klavye olması yeterlidir. Ayrıca kullanıcılar böyle bir sistem içinde bulunduklarının farkına bile varmazlar (Ilonen, 2003).

2.4. Biyometrik Sistemler

2.4.1. Biyometrik Sistemlerin Çalışma Mantığı

Biyometrinin geliştirilmesinde diğer yeni teknolojilerin geliştirilmesinde olduğu gibi güvenlik başrol oynamaktadır (Ergen ve Çalışkan, 2011). Biyometrik sistemlerin çalışması temel olarak iki kısma ayrılmaktadır. Birinci kısımda öncelikle sisteme kişinin tanımı yapılmaktadır. Kişinin ilgili yönteme ait bilgileri gerekli yazılımlar ve donanımlarla dijital ortama geçirilmektedir. Bu bilgiler yine yönteme ait özel algoritmalar kullanılarak incelemeler yapılıp, incelemeler sonunda kişiyi tanımlayan özellik parametrelerinin veri tabanına kayıtları oluşturulmaktadır. İkinci kısımda ise kişinin kimlik doğrulama isteği yapılmaktadır. Bu kısımda yine önceki kısımda olduğu gibi aynı araçlar kullanılarak sisteme girilen bilgiler, yine aynı algoritmalar kullanılarak analiz edilmektedir. Bu bilgilerin veri tabanında daha önceden oluşturulan kayıtlarla eşleşip eşleşmediğine bakılmaktadır. Eşleşme durumunda kişi sisteme kabul edilmekte aksi takdirde kişinin isteği reddedilmektedir (Zhang ve Shu, 1999).

2.4.2. Biyometrik Sistemlerin Özellikleri

Tüm biyometrik sistemler aşağıda açıklanmış olan beş özelliğe sahip olmalıdır : (Chellappa ve ark.,1995)

Evrensellik: Tüm bireyler biyometrik özelliklere sahip olmalıdır.

Eşsiz olma: Biyometrik karakteristiğin her insanda farklı bir şekilde yer almasıdır.

Süreklilik: Karakteristiğin zamanla değişmemesidir.

Elde edilebilirlik: Biyometrik özelliklerin bazı pratik cihazlarla ölçülebilir olmasıdır.

Kabul edilebilirlik: Bireylerin biyometriğin ölçüm ve toplanmasında itirazları olmamalı ve her bireyin de biyometrik özelliklerin ayırt edici özellikler olduğunu kabul etmesi.

Ayrıca şu iki özellik de değerlendirmelerde kullanılmaktadır:

Performans: Biyometriğin doğruluk oranıdır.

Yaygınlık: Biyometrik çözümün kullanım durumudur.

Aşağıdaki tabloda her biyometrik ölçünün hangi özellikte ne derece başarılı olduğu özetlenmiştir.

Çizelge 2.1. Biyometrik sistemlerin özellikleri.

Biyometrik Ölçüler	**Evrensellik**	**Eşsiz Olma**	**Süreklilik**	**Elde Edilebilirlik**	**Kabul Edilebilirlik**	**Performans**	**Yaygınlık**
DNA	Y	Y	Y	D	D	Y	D
Yüz	Y	D	O	Y	Y	D	Y
Parmak İzi	O	Y	Y	O	O	Y	O
El	O	O	O	Y	O	O	O
İris	Y	Y	Y	O	D	Y	D
İmza	D	D	D	Y	Y	D	Y
Ses	O	D	D	O	Y	D	Y
Tuş Vuruş	Y	O	D	Y	O	O	D
Y-Yüksek, O-Orta, D-Düşük							

2.4.3. Biyometrik Sistem Yaklaşımı

Biyometride birbirinden ayırt edilmesi gereken önemli iki terim vardır. Bunlar, doğrulama ve tanımlamadır. Kimlik doğrulama işleminde kişinin kim olduğunu iddia ettiği sistem tarafından bilinmektedir ve karşılaştırma sistemdeki sadece bir kayıtla yapılmaktadır. Kimlik tanımlama işleminde ise ek bir bilgi yoktur ve karşılaştırma sistemdeki tüm kayıtlar ile yapılmaktadır (Ilonen, 2003). Kimlik doğrulamada kullanıcı doğru kişi olduğunu kanıtlamak için talepte bulunur. Elinde kendisine önceden sağlanmış olan bir anahtar bulunmaktadır. Bu anahtar bir kullanıcı adı veya ID numarası olabilir. Kimlik talebinden sonra, kullanıcının biyometrik verisiyle kayıtlı olan veri karşılaştırılır. Daha sonra sistem doğru veya yanlış diye bir cevap dönmektedir. Sonuç olarak kimlik doğrulamada karşılaştırma birebir yapılır. Çünkü elde edilmiş olan biyometrik veri sadece kullanıcının talep edilen kayıtlı olan biyometrik verisiyle karşılaştırılır. Kimlik tanımlamada ise durum biraz farklıdır. Sisteme giriş yapmak isteyen kullanıcının biyometrik verisi bir dizi kullanıcının biyometrik verisi ile karşılaştırılır. Bu yüzden, tanımlamada birden çoğa başvurulur. Daha sonra sistem kullanıcı adı veya kimlik numarası gibi bir kimliği geri döndürmektedir (Ergen ve Çalışkan, 2011).

Tuşa basma dinamiği kimlik doğrulama için uygulanabilirdir, ayrıca kimlik tanımlama için de uygulanabilirdir. Yapılan çalışmaların çoğu kimlik doğrulama üzerinedir.

3. TUŞA BASMA DİNAMİKLERİ

Günümüzde kullanıcıların doğrulanması amacıyla sistemler üzerinde genel olarak kullanıcı adı ve şifre ikilisi birlikte kullanılmaktadır. Sisteme girişlerde bu ikilinin kullanımı ezbere dayalı olarak kullanılagelmekte ise de bu durumun kendi içerisinde bazı eksiklikleri vardır. Bu eksiklikler, kullanıcı adı ve şifre ezberlenme zorunluluğu; şifrelerin "doğum yılı" gibi tahmini kolay olarak tercihi olmaktadır. Bu eksikliklere ilave olarak belirlenen bir şifrenin birden fazla sistemde kullanılmasıdır. Bu durumda bir şifreyi ele geçiren diğer sistemlere de ulaşabilmektedir. Şifreler sözle ve/veya yazılı olarak kolayca aktarılabilmektedir. Bu durumda kullanıcıların sistemlere kendilerini tanıtmak için şifrelerini girerlerken tuşlara basıp bırakma şekilleriyle yani davranışsal biyometrik tabanlı kimlik doğrulama ile bu eksikliklerin çözülebilmesi öngörülmektedir.

Güvenlik ve mahremiyet konusu günümüzde bilgi teknolojilerinde daha çok önem kazanmıştır. Hemen hemen her yerde ve her zaman anlık olarak sistemlere giriş ve bilgilerin transferi yapılabilmektedir. Bu sıklıktan dolayı, sistemlere giriş çıkış yapan kullanıcıların girişlerinin doğrulanması, dolayısıyla sistemin güvenliği çok önemlidir. Tuşa basma dinamiklerinin yakalanması ile insanlara şifre ezberlemenin getirdiği yükün hafifletilmesi ve böylelikle kendilerini bilgisayar sistemlerine kolayca doğrulatabilmeleri düşünülmüştür. Ayrıca tuşa basma dinamiklerinin kullanıldığı sistemlere girmek isteyen sahte kullanıcıların şifreleri bilseler dahi böyle sistemlere girebilmeleri engellenmiş ve kullanılan sistemlerin güvenliği arttırılmış olmaktadır.

3.1. Biyometri Özelliklerinin Tuşa Basma Dinamiklerindeki Karşılıkları

Tüm biyometrik sistemlerin sahip olması gereken özelliklerin tuş basma dinamikleri üzerindeki karşılıkları aşağıda madde madde olarak açıklanmıştır. (Anonim, 2015d)

Evrensellik: Tuş basma dinamiği, klavye kullanmasını bilen her bir birey tarafından yararlanılabilen bir biyometrik çözümdür.

Eşsiz Olma: Fiziksel biyometrikler gibi kişiye özel olduğunu ispat etmek mümkün değildir. Ancak tuşa basma eyleminin fizyolojik olarak taklit edilmesinin mümkün olmadığı kabul edilmektedir.

Süreklilik: Tuş vuruş dinamiğinde karşılaşılabilen büyük bir problemdir. Bir kişinin tuş vuruş ritmi günden güne ya da aynı gün içerisinde bile değişkenlik gösterebilir. Yorgunluk, klavye değişimi, ruh hali, ilaç ya da alkol kullanımı nedenler arasında sayılabilir.

Elde Edilebilirlik: Tuş vuruş dinamiğinin önemli bir avantajıdır. Diğer biyometrik çözümler gibi ayrı özel bir donanım gerektirmez. Sadece standard bir bilgisayar klavyesi yeterlidir. Ayrıca tuş vuruş dinamiği ile kullanıcının ritimlerini arka planda tutmak ve kullanıcı değiştiğinde uyarı vermek de mümkündür.

Kabul Edilebilirlik: Ülkeye ve kanunlara göre değişiklik gösterebilir. Mutlaka deneylerden ve uygulamayı yürütmeden önce kullanıcıların tuş vuruş dinamiklerinin alınmasına razı olmaları gerekmektedir.

Performans: Tuş vuruş dinamiği diğer fiziksel tabanlı biyometriklere göre daha fazla değişkenlik gösterir. Bu biyometrik çözümde, bu yüzden yüksek YKO ve yüksek YRO değerlerine rastlamak mümkündür.

Yaygınlık: Kullanımı yeni yeni gelişmekte olan bir biyometrik çözümdür. İleriki zamanlarda kullanımının artması beklenmektedir.

3.2. Doğruluk Ölçüm Metrikleri

Biyometrik literatüründe kimlik doğrulama işleminin ölçümünde faydalı olan iki ana metrik vardır. Bunlar; tip 1 hatası olan Yanlış Red Oranı (YRO) ve tip 2 hatası olan Yanlış Kabul Oranı (YKO) dır.

Yanlış Red Oranı (YRO): Gerçek kullanıcıyı sahte kullanıcı gibi görme oranı

Yanlış Kabul Oranı (YKO): Sahte kullanıcıyı gerçek kullanıcı gibi görme oranı

Bir diğer ölçü de Eşit Hata Oranıdır (EHO). Bu oran, sahte kullanıcının sisteme ulaşma oranını azaltırken, gerçek kullanıcının da sisteme kabulünü dengeleyip rahatlatır. Günümüze kadar ulaşan bütün biyometrik sistemler, bu iki oran arasında şu şekilde bulunurlar. Ya düşük YKO yüksek YRO ya da tam tersidir.

Eldeki öznitelikler ve hata oranını ölçen uygun metrikler, son safha kimlik doğrulama işleminde operasyonel bakış açıları geliştirmeye neden olur.

Kaydolma işlemi kullanıcıların yazma sitili ile örnek bir istatistik imza oluşturmak için sisteme girişlerini yapmalarının istendiği etaptır. Biyometrik sistem birincil ve ikincil öznitelikleri kaydolma safhasında çıkartır. Kaydolma işleminde kullanılan farklı yöntemler vardır. Bazıları istenilen güvenlik seviyesine bağlı olarak 400-1500 karakter arasında değişen bir metnin girilmesini istemektedir. Diğerleri ticari ürün Biopassword gibi kullanıcıdan giriş ID/şifresini 10-15 kez girmesini isteyebilmektedir. Bazı yöntemler her iki stratejinin birleşimini kullanmaktadır. Bunlar; biopassword gibi bir kaydolma ve onu takip eden periyodik aralıklarla tuşa vuruşları gözlemlemedir. Hangi yöntem kullanılırsa kullanılsın, kaydolma işleminde iyi bir denge bulunmalıdır. Eğer çok uzun olursa kullanıcılar sıkılabilir, kısa olursa da doğruluk tam sağlanmayabilir. Bazı kullanıcılar periyodik olarak onların yazma sitilinin kontrol edilmesini rahatsız edici bulmakta ve bunun onların özeli olduğunu düşünmektedirler. Bir kere veri toplandığında o kullanıcı için bir imza referansı oluşturulur. Kullanıcının bundan sonraki girişlerinde girilen tuşa vuruşları bu referans ile karşılaştırılır. Eğer bunlar tolerans aralığında ise kullanıcı sisteme kabul edilir. Değilse sistem kilitlenir ya da başka bir şey yapılır. Biyometrik sistemlerde bu çözüm yapılırken iki durum vardır. Ya çok sıkı olup her giriş hareketi reddedilecek ya da çok rahat olup her giriş hareketi kabul edilecek. Buradaki denge EHO ile sağlanır. Eğer tuşa vuruş analizi kimlik doğrulama işlemi için uygun bir yöntem olacaksa şu iki soru cevaplandırılmalıdır:

1. Kullanıcı tabanlı tuşa vurma sitilinden hangi özellikler çıkartılmalı?
2. Kullanıcıların tuşa vurma sitilleri arasındaki farkı anlamayı arttırmak için hangi kimlik doğrulama algoritmaları kullanılmalı?

Tuşa vurma özelliği ile ilgili, gerçekten de parmak izi gibi kişiye özgü mü? Bunu cevaplamak için, kullanıcının tuşa vurma biçiminden özellikler çıkartılmalıdır. Bu özellikler kullanıcı tuşa vurduğu zaman kayıt edilen tekil ölçümlerin toplamıdır. Örneğin ne kadar hızlı yazıyor, tuşa basma süresi, bir tuştan diğer tuşa ne kadar sürede geçiyor, tuş basma sırası, shift tuşunu kullanışı ve tuşa vuruş sitili nasıl gelişiyor gibi. Bu özellikler her kullanıcı için toplanmalı ve kimlik doğrulamada kullanılmak için saklanmalıdır. Her özellik için yeterli miktarda örnek

elde edebilmek için, çoğu tuşa vurma dinamiği sistemleri kaydolma safhasını içermektedir. Kaydolma verisiyle her kullanıcının tuşa vurma sitili modeli oluşturulabilir. Daha sonra bu modelin parametreleri verilen sınıflandırıcı için girdi olur. Doğru parametreleri seçmek hiç şüphesiz kritik bir konudur. Özellikler genelde kaydolma işlemi sırasında toplanmaktadır. Özellikler birincil ve türemiş olarak sınıflandırılabilir. Birincil özellikler bir tuşa basma süresi, bir tuşu bırakıp diğer tuşa geçme süresi olarak gruplandırılabilir. Birincil özelliklerden ikincil özellikler türetilebilir. Bunlar tuşlara uygulanan kuvvet miktarı, yazma hızı, hataların oluşma sıklığı, kullanılan tuş kombinasyonları, hataları düzeltme yöntemi olarak sıralanabilir. Tuşa basma dinamiğini kullanan biyometrik sistemlerde sınıflandırma kimlik doğrulamaya indirgenmiştir.

3.3. Kullanılan Ölçüm Metrikleri

Tuşa basma dinamiğinde kullanılan metrikler yanlış red oranı (YRO), gerçek bir kullanıcıyı sahte bir kullanıcı olarak görme oranı, yanlış kabul oranı (YKO), sahte bir kullanıcıyı gerçek bir kullanıcı olarak görme oranı olarak tanımlanmaktadır.

Bu ölçüm metriklerine ilaveten aşağıda tuşa basma dinamiklerinde doğruluğu ölçmek için iki metrik daha gösterilmektedir. Bu ölçüm metriklerinin formülasyonları (3.1) ve (3.2)'de verilmiştir.

$$Kesinlik = \frac{TP}{TP+FP} \tag{3.1}$$

Yukarıda belirtilen formülasyon, kullanıcıların kendi şifreleriyle yaptıkları girişlerin doğru olarak kabul edilmesi oranıdır. Örneğin; bir kullanıcının kendi şifresi ile 10 giriş yapması durumunda bunlardan kaç tanesinde kullanıcıyı doğru olarak sınıflandırdığı oranıdır.

$$Hassasiyet = \frac{TP}{TP+FN} \tag{3.2}$$

Yukarıda belirtilen formülasyon ise kullanıcının başka bir kullanıcı olarak şifre girmesi durumunu ayırt edebilme oranıdır. Kısaca, bir kullanıcı başka bir kullanıcı gibi sisteme girmek istediğinde, bu kullanıcının girişlerinden kaçını doğru bir şekilde ayırt edebildiği ve aynı zamanda doğru kullanıcıyı tespit edebildiği oranıdır.

Bu metriklerde kullanılan ifadelerin tanımlamaları aşağıdaki tabloda yapılmıştır.

Çizelge 3.1. Tanımlar.

	Gerçek kullanıcı	Sahte kullanıcı
Sistem tarafından doğru saptanması	Sisteme kabul edilmesi TP	Sisteme kabul edilmemesi TN
Sistem tarafından yanlış saptanması	Sisteme kabul edilmemesi FP	Sisteme kabul edilmesi FN

Kesinlik; sisteme kayıtlı kişilerin doğru sınıflandırma oranını ölçerken, doğru kullanıcıyı ne kadar, sisteme kayıtlı başka bir kullanıcı ile karıştırdığı durumunu göz önünde bulundurur.

Hassasiyet ise, sisteme kayıtlı olan ya da olmayan bir kullanıcının sistemde başka bir kullanıcı gibi saptanmasının oranını göz önünde bulundurur.

$$F_1 = \frac{2xKesinlikxHassasiyet}{Kesinlik+Hassasiyet} \tag{3.3}$$

Bu iki değerin ortak bir formülle değerlendirildiği F-Ölçü yukarıdaki şekilde tanımlanmıştır.

F-Ölçü görüldüğü gibi Kesinlik ve Hassasiyet değerlerinin harmonik ortalamasıdır.

Ayrıca kullanılabilen başarı ölçütlerinden biri de doğrulukdur. Doğruluk, tüm doğru sonuçların genel populasyona oranıdır. Doğruluk (3.4)'de gösterildiği gibi hesaplanmaktadır.

$$Doğruluk = \frac{TP+TN}{TP+FP+TN+FN} \tag{3.4}$$

4. ALAN TABANLI TUŞA BASMA DİNAMİKLERİ

4.1. Verilerin Toplanması

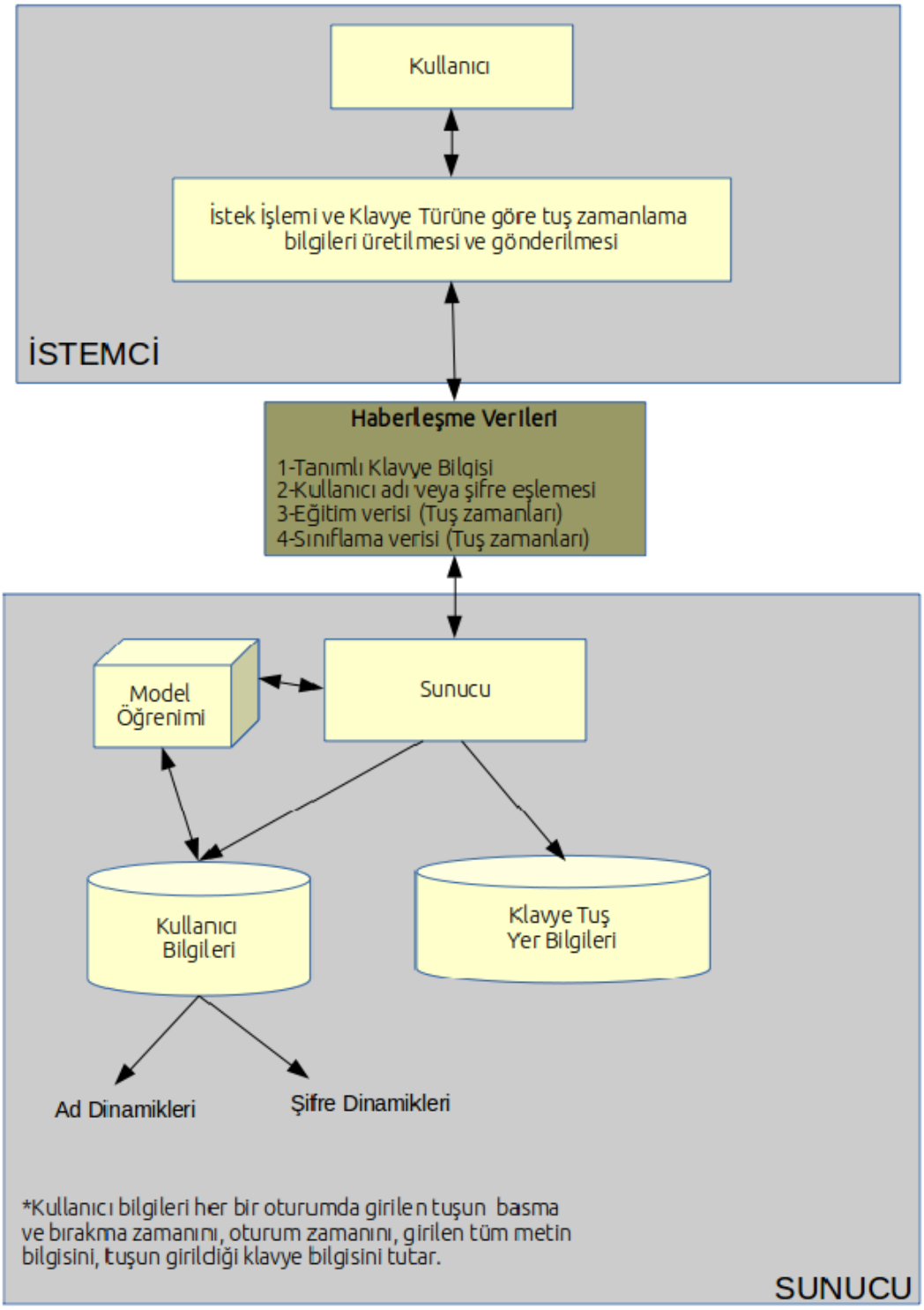
Tuş vuruşları dinamiği verisinin toplanması işlemi için sunucu-istemci modeli seçilmiştir. Bu modelde kullanıcılar istemci gibi davranarak sisteme bağlanmaya çalışırlar. İstemci sunucuya girilen tuş dinamiklerini ileterek sunucu tarafında kaydedilmesini sağlar. Sunucu ise kendisine iletilen tuş dinamiği verisini dosya sisteminde saklar. Bu şekilde veri tek bir merkezde toplanarak işlenebilir hale gelir. Bu modelde uygulama programı olarak Java programlama dili kullanılmıştır.

Tuş dinamikleri verisinin toplanmasında istemci tarafında iki aşamalı bir giriş söz konusudur. Kullanıcı ilk etapta sistemde mevcut mu diye test edilir. Eğer kullanıcı sisteme belirli bir kullanıcı adı ile giriş yaptıysa o zaman kullanıcı doğrulanmış olur. İlk doğrulama aşamasında tuş dinamikleri verisi toplanmaz. İkinci aşama kullanıcının belirlediği kullanıcı adını birden fazla kez gireceği ve tuş dinamiklerinin toplanacağı kısımdan oluşur. Kullanıcılar tuş dinamiklerini bu ekranda enter tuşuna basarak birden fazla kez girebilir. Bu sistemde kullanıcılar her seferinde girdikleri kullanıcı adını başta girilen kullanıcı adı ile eşleştirmek zorundadırlar. Bu yöntemle yanlış girdiler engellenmiş olur. Her doğru girişi belirten bir sayı kullanıcılara gösterilerek kaç sefer giriş yaptıkları belirtilmiş olur.

Şekil 4.1.'de istemci sunucu işlemleri belirtilmiştir. İstemci işlemleri tuşların basma ve bırakma zamanlamasını, kullanıcının girdiği metinin tutulmasını ve kullanıcının sisteme giriş zamanını sunucuya iletmekten oluşur. Sunucu bu bilgileri alarak iki kısıma ayırır. Birinci kısım eğitim için kullanılacak modelin kendisi ikinci kısım ise bu modeli oluşturacak ham verinin saklanması. Sunucu bu iki bilgi dışında kullanıcının giriş yaptığı klavyedeki tuşların lokasyon bilgisini de tutmaktadır. Bu kullanıcıdan gelen tuşların lokasyonlarının bulunmasında kullanılır.

Sunucu elde edilen tuş basma dinamiklerini ham metin verisi olarak saklar. Bu verilerin işlenmesinde klavye lokasyon bilgileri kullanılmaktadır. Her bir tuş kullanıcının belirttiği klavyede tablo koordinatı ve alan bilgisi ile eşlenir. Basılan bir tuş için tuş lokasyon bilgisi iki farklı sayısal değerden (x,y) ve alan bilgisi ise

bir etiketten oluşur. Şekil 4.2.'de istemciden gelen tek bir ham tuş verisinin hali gösterilmektedir.

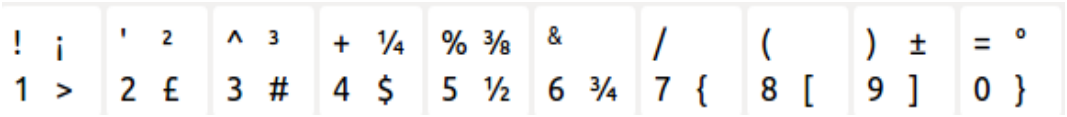


Şekil 4.1. İstemci sunucu yapısı.

```
1-78-78||1427357395178-1427357395271-93
```

Şekil 4.2. Tek tuş belirteci ve dinamikleri.

İstemcide toplanan veri iki kısımdan oluşmaktadır. Birinci kısım tuş belirtecidir. İkinci kısım ise vuruş zamanlarıdır. Birinci kısımda klavyedeki lokasyon bilgisi, tuşun ASCII kodu ve tuşun Extended-ASCII kodu bulunur. Vuruş zamanları ise ikinci kısımda basma, bırakma ve basma-bırakma arasında geçen zaman olmak üzere üç bilgiden oluşur. Ardışık olarak gelen tuş bilgileri eğer yardımcı tuşlar barındırıyorsa linear olarak soldan sağa işlenerek tek tuş bilgisi elde edilir. Tek tuş bilgisi birden fazla tuşun basımından veya tek bir tuşa basımdan oluşan metinsel bir bilgidir. Örneğin "SHIFT + a = A" iki tuşun basımından oluşan klavyede tek bir lokasyona karşılık gelen bir bilgidir. Bu şekilde klavye üzerindeki tuşların lokasyon bilgileri ile eşleştirilmesi sağlanmış olur. Yardımcı tuşların da (SHIFT, CAPS, ALT, FN) işin içine dâhil edilmesi ile klavyede bulunan ve birden fazla lokasyonda bulunan tuşların doğru şekilde ayırt edilmesi sağlanmış olur. Şekil 4.3.'de birden fazla yerde bulunan tuşlar klavye düzeni üzerinde işaretlenmiş olarak gösterilmiştir.



Şekil 4.3. Q klavye rakam tuşları.

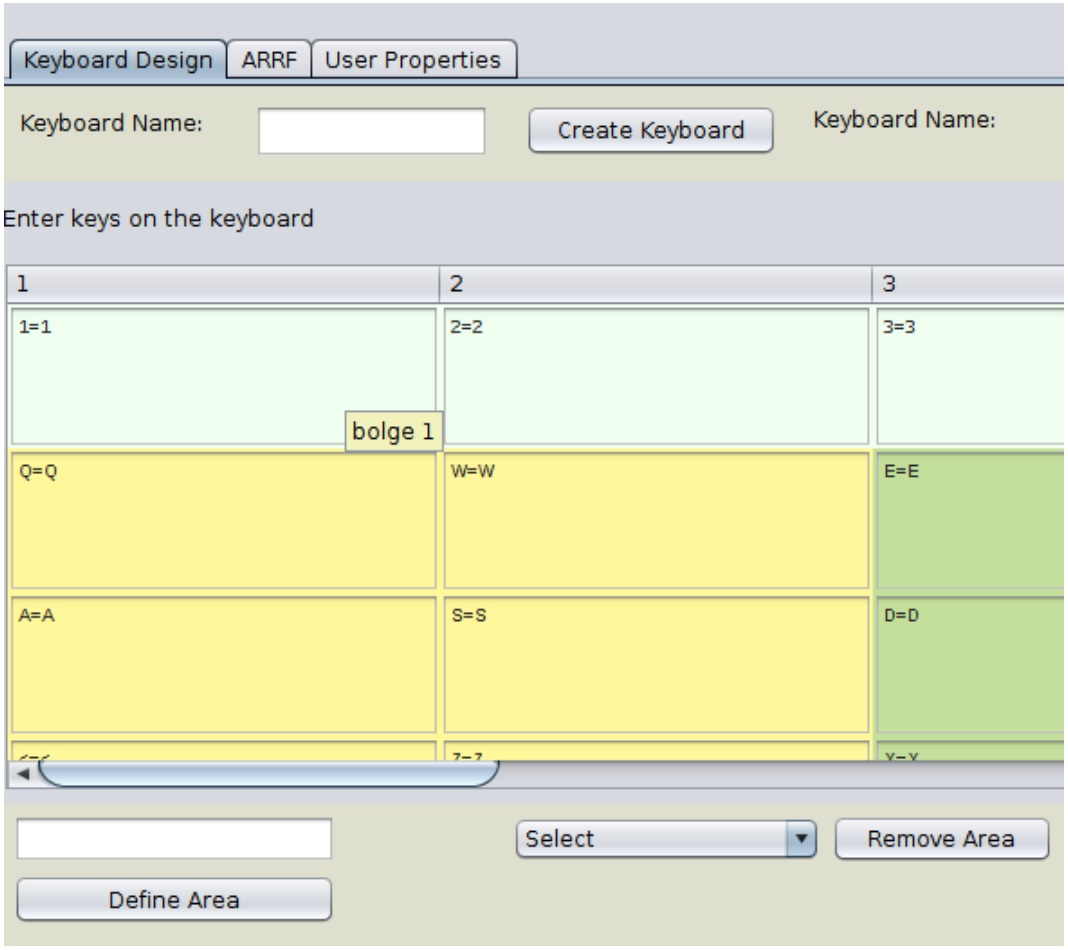
Şekil 4.3. görüldüğü üzere 1,2,3.. gibi tuşlarla birlikte (,[… gibi tuşlar aynı lokasyonda yer almaktadır. Örneğin "SHIFT+7 = /" aslında görüldüğü üzere iki tuşa basımdan oluşmakta ancak lokasyon olarak 7 tuşunun bulunduğu yerde yer almaktadır. Bu yöntemle kullanıcı SHIFT+7 tuşlarına bastığı zaman SHIFT ve 7 tuşlarının bulunduğu lokasyonlarda gezinmiş olacaktır. Bu eşlemeyi yapabilmek için kullanıcının SHIFT ve 7 tuşlarına bastığı zaman "/" karakterini yazdığı bilinerek metin bilgisi üzerinden lokasyon bilgileri eşleşmiş olacaktır.

4.2. Klavye Düzeni

Kullanıcılar giriş yapmadan önce bir yönetici tarafından kullanacakları klavyeler sisteme kaydedilmiş olmalıdır. Kullanıcıların giriş yapacakları klavye düzenini seçerek bu klavye üzerinde giriş yaptıklarını belirtmeleri gerekmektedir. Yönetici klavye düzenini girerek tuşlar için lokasyon bilgilerini belirtmiş olur. Şekil 4.4.'de klavye düzeni ekranı gösterilmiştir.

Şekil 4.4.'de klavye düzeni Q klavye için sadece rakam ve harfleri barındıran bir formatta yazılmıştır. Farklı renkler yöneticinin belirlediği alanları göstermektedir. Bu alanlar tuş lokayon bilgileri ile birlikte her bir hücreye ayrı ayrı kesişen kümeler şeklinde tanımlanabileceği gibi birbirinden ayrı hücreler için farklı şekilde de tanımlanabilir. Şekil 4.4.'de üç alan gözükmektedir. Birinci alanda 1, 2 ve 3 tuşları, ikinci alanda Q, W, A, S, < ve Z tuşları, üçüncü alanda ise E, D, ve Y tuşları bulunmaktadır. Bu giriş ekranını tablo formatı olarak adlandırıyoruz. Yönetici bu alanları bir etiket vermek koşulu ile tablo görünütüsünde tuşları seçerek oluşturabilir. Alan adları farklı klavyeler için aynı girildiğinde klavye bağımsız tuş dinamiklerinin yakalanması için kullanılabilir. Örneğin bir klavye için "A" tuşunun yeri diğer bir klavye üzerinde farklı lokasyonda bulunabilir. Ancak "A" tuşunun diğer klavyede aynı alan adına karşılık gelen bölgede yer alması durumunda iki klavye için de "A" tuşu aynı alanda yer alacaktır. Benzer şekilde lokasyon eşleştirme işlemlerinde kullanıcılar farklı klavyeler üzerinde farklı tuş dinamiklerine sahip olabilirler ancak alanlar üzerinde tuştan bağımsız olarak benzer tuş dinamikleri sergileyebilirler.

Tablo formatında her bir hücrede birden fazla tuş bulunabilir. Hücrelerin koordinatları basılan klavyedeki lokasyon bilgilerini, gösterilen değerler ise metin bilgisini ifade etmektedir. Tablo formatının oluşturulmasında kullanıcının kullanacağı klavyenin bir örneğinin kullanılması gerekir. Çünkü ASCII ve extended-ASCII kodlarının eşleşmesi için bu gereklidir. Yönetici tanımladığı klavyeler için tuş dinamiklerinin toplanmasından sonra dilediği gibi alan adlarını değiştirebilir. Bu şekilde tüm kullanıcılar için değişik alanlar yaratarak en uygun doğruluk oranını sağlayabilir. İleride yapılabilecek başka bir çalışmada alanların otomatik olarak tüm kullanıcılar için en uygun doğruluğu verecek şekilde oluşturulması düşünülebilir.



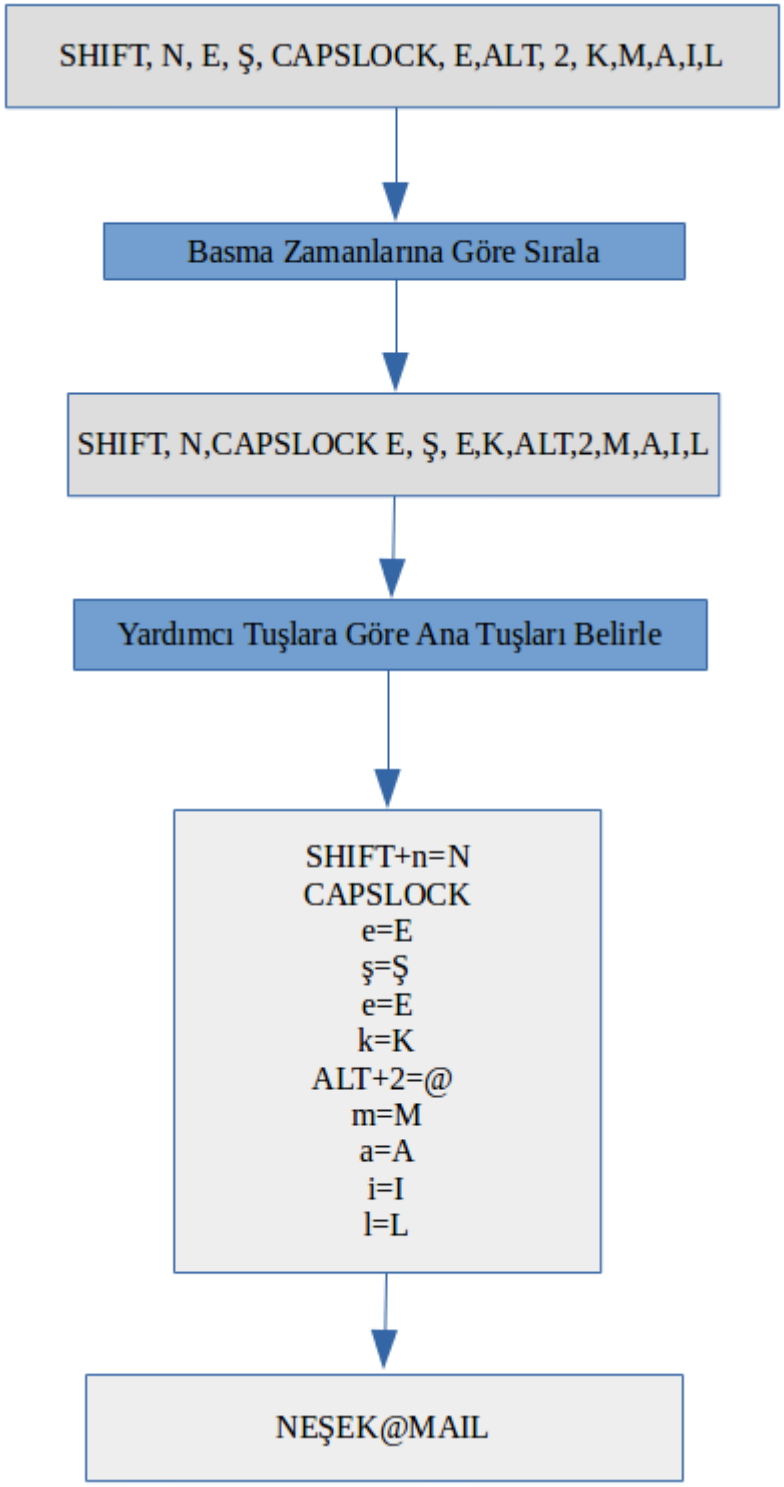
Şekil 4.4. Klavye tablo ekranı.

Yönetici sistem üzerinde gerekli gördüğü klavye tasarımlarını Şekil 4.4.'de gösterilen ekran üzerinde yaptıktan sonra İstemci'de kullanıcı verisi toplanması

sağlanır. Toplanan verilerin Şekil 4.1.'de belirtilen "Model Öğrenimi" için hazırlanması gerekir. Model öğrenimi kullanıcılardan gelen ham verinin işlenmesinden sonra ortaya çıkan veri üzerinde olacaktır. Bu kısıma Özellik Vektörlerinin Çıkarılması denir. Bir sonraki bölümde Özellik Vektörlerinin çıkarılması işlemi anlatılmaktadır.

4.3. Özellik Vektörlerinin Çıkarılması

Ham veri tuş belirteçlerinin işlenerek model oluşturulması işleminden önce klavye tanımlanması gereklidir. Harf dizilerinin birden fazla klavye tuşuna denk geldiği veya tuş dizilerinden tek tuşa denk geldiği tuşlar için tuş alan ve lokasyon bilgilerinin klavye düzeni kullanılarak saptanması gerekir. Bu işleme "tuş lokasyonu anlamlaştırılması" denmektedir. Tuş lokasyonu anlamlaştırması özellikle yardımcı tuşların kullanıldığı alanlar için gereklidir. Örneğin kullanıcının büyük harfle yazdığı tuş dizisinde SHIFT tuşu yardımcı tuşlardandır. Bu tuşların belirteçleri sıralı olarak işlenerek tuş lokasyonları elde edilir. Şekil 4.5.'de bu işlemdeki sıralama gösterilmektedir.



Şekil 4.5. Harf dizisinin işlenmesi.

Şekil 4.5.'de girilen bir tuş dizisinin basma zamanlamasına göre sıralanması ve sonrasında yardımcı tuşların belirlenerek orijinal tuş girdilerinin elde edilmesi işlemi gösterilmektedir. Bu şekilde gösterilen girdi en son olarak

“NEŞEK@MAIL” olarak elde edilir. Özellik vektörleri oluşturulurken en önemli aşama olan farklı klavyeler için tuş lokasyonlarının bulunması, bu aşamada oluşan ana tuş belirteçlerinden “ALT+2” tuşu ile elde edilen “@” karakteri “ALT+2 = @” şeklinde ifade edilmektedir. Eğer klavyemizde birden fazla lokasyonda @ işareti varsa bu tuşun lokasyonu ALT+2 tuşlarının bulunduğu lokasyon ile elde edilecektir. Özellik vektörleri makine öğrenmesinde verinin işlenmesinden sonra oluşan vektörlerdir. Tuş dinamiklerinin elde edilmesinden sonra ham veri oluşmaktadır. Bu veri az önce anlatıldığı gibi işlenerek tuş dizeleri oluşur. Tuş dizelerinin oluşmasından sonra her bir tuşun klavye ile eşleştirilmesi gerekir. Bu eşleştirme işlemi tamamlandıktan sonra tuş veri yapısı olacaktır. Tuş veri yapısı klavye tuşu ve kullanıcı tuşu olarak ikiye ayrılır. Kullanıcı tuşu klavye bilgisi, tuş çıktısı ve basma-bırakma zamanlarından oluşmaktadır. Klavye tuşu ham verinin işlenmesinden sonra oluşan tuş bilgisidir. Klavyedeki tuş bilgisinin üç ana özelliği bulunmaktadır. Bunlar tuşun klavye tablo formatındaki satır ve sütün koordinatları, bulunduğu alanlar ve basma-bırakma süreleridir.

Klavye tuş bilgisi oluştuktan sonra klavye dizisi için n-gram modeller oluşturulmaktadır. Bu birden fazla ardışık tuş grubu için oluşan tuş bilgileridir. Aşağıda kullanılan tüm özellikler sıralanmıştır.

- **1. Tuş basma-bırakma zamanı (Key Dwell):** Herhangi bir tuş için aynı tuşa basma ile aynı tuşu bırakma arasında geçen süredir.
- **2. Ardışık iki tuş arasında uçma zamanı (Key Flight):** Herhangi iki tuş için ilk tuşu bırakma ve ikinci tuşa basma arasında geçen süredir.
- **3. Ortalama Zaman (Average Time):** Girilen metinlerin ortalama giriş süresidir.
- **4. Yarı Zaman (Middle Time):** Girilen metnin orta tuşu ile ilk tuşu arasında geçen süredir.
- **5. Maksimum Zaman (Maximum Time):** Girilen tüm metinlerin giriş sürelerinden en uzun olanıdır.
- **6. Minimum Zaman (Minimum Time):** Girilen tüm metinlerin giriş sürelerinden en kısa olanıdır.
- **7. 1ngraph-Ortalama Bir Tuşa Basma-Bırakma:** Tüm tuşlar için basma ve bırakma arasında geçen ortalama süredir.

- **8. 2ngraph-Ortalama İki Tuşa Basma-Bırakma:** Tüm tuşlar için iki tuşa basma ve bırakma arasında geçen ortalama süredir.
- **9**. **3ngraph-Ortalama Üç Tuşa Basma-Bırakma:** Tüm tuşlar için üç tuşa basma ve bırakma arasında geçen ortalama süre. Birinci tuşa basma ve son tuşu bırakma arasında geçen ortalama süredir.
- **10. Ortalama Alanda Durma Zamanı (Area Dwell):** Tüm alanlar için alandaki mevcut geçirilen süre. Alandaki herhangi bir tuşa basma ve aynı alanda herhangi bir başka tuşu bırakma arasında geçen süresidir.
- **11. Ortalama Alan Uçuş Zamanı (Area Flight):** Tüm alanlar için iki alan arasındaki uçuş zamanı. Birinci alandaki herhangi bir tuşu bırakma ve ikinci alandaki tuşa basma arasında geçen ortalama zamanıdır.
- **12. Lokasyon Basma-Bırakma Zamanı (Location Dwell):** Herhangi bir lokasyona (koordinata) basma ve bırakma arasında geçen süredir.
- **13. Ardışık İki Lokasyon Arasında Uçma Zamanı (Location Flight):** Herhangi iki lokasyon (koordinat) için ilk lokasyondaki tuşu bırakma ve ikinci lokasyondaki tuşa basma arasında geçen süredir.

Kullanılan özellikler WEKA arff dosya formatına dönüştürülür. Şekil 4.6.'da oluşan ARFF formatındaki metin dosyası gözükmektedir.

```
@RELATION keystroke
@ATTRIBUTE MIDDLE_TIME NUMERIC
@ATTRIBUTE AVERAGETIME NUMERIC
@ATTRIBUTE AreaFlight-harf2-harf3 NUMERIC
@ATTRIBUTE KeyFlight-69-78 NUMERIC
@ATTRIBUTE Dwell-78 NUMERIC
@ATTRIBUTE 1NGRAPH_TIME NUMERIC
@ATTRIBUTE 2NGRAPH_TIME NUMERIC
@ATTRIBUTE AreaFlight-harf1-harf2 NUMERIC
@ATTRIBUTE KeyFlight-83-69 NUMERIC
@ATTRIBUTE Dwell-69 NUMERIC
@ATTRIBUTE AreaFlight-harf2-harf1 NUMERIC
@ATTRIBUTE KeyFlight-69-83 NUMERIC
@ATTRIBUTE Dwell-83 NUMERIC
@ATTRIBUTE MAXIMUM_TIME NUMERIC
@ATTRIBUTE MINIMUM_TIME NUMERIC
@ATTRIBUTE KeyFlight-71-65 NUMERIC
@ATTRIBUTE Dwell-65 NUMERIC
@ATTRIBUTE AreaFlight-harf3-harf2 NUMERIC
@ATTRIBUTE KeyFlight-85-71 NUMERIC
@ATTRIBUTE Dwell-71 NUMERIC
@ATTRIBUTE AreaFlight-harf3-harf3 NUMERIC
@ATTRIBUTE KeyFlight-78-85 NUMERIC
@ATTRIBUTE Dwell-85 NUMERIC
@ATTRIBUTE KeyFlight-0-66 NUMERIC
@ATTRIBUTE Dwell-66 NUMERIC
@ATTRIBUTE KeyFlight-82-0 NUMERIC
@ATTRIBUTE Dwell-0 NUMERIC
@ATTRIBUTE Dwell-82 NUMERIC
@ATTRIBUTE class {nese,nese-IMPOSTER}
@DATA
1108.5,1905.5,292.0,90.0,195.0,1459.0,1108.0,125.0,1
1181.5,2007.5,320.0,130.0,205.0,1500.5,1116.5,135.5,
1065.0,1915.0,250.0,70.0,215.0,1430.0,1085.0,140.0,1
1127.0,1869.0,249.5,75.5,201.0,1371.5,1028.5,130.0,1
725.0,1598.5,171.0,31.0,273.0,965.5,770.75,124.5,124
1288.5,2194.0,358.5,118.5,240.0,1643.5,1179.25,110.0
810.0,1685.0,210.0,40.0,265.0,1105.0,870.0,120.0,120
1261.0,2135.5,434.0,74.0,196.5,1720.5,1335.25,120.0,
760.0,1675.0,260.0,90.0,205.0,1080.0,880.0,100.0,100
848.0,1372.0,179.5,179.5,186.0,950.5,794.5,153.0,153
443.41666666666663,711.5,58.5,58.5,161.5,367.5000000
IMPOSTER
```

Şekil 4.6. ARFF dosyası.

4.4. İstemci Kullanıcı Profilleri

Bu bölümde istemci tarafında verinin nasıl toplandığı anlatılmaktadır. Kullanıcıların profillerinin oluşturulması her ne kadar sunucu tarafında olsa da kullanıcılar verilerini istemci tarafında oluşturulurlar. Şekil 4.1.'de klavye verisinin oluşturulması görüldüğü gibi istemci tarafında olmaktadır. İstemci tarafında iki farklı giriş ekranı mevcuttur. Bu giriş ekranlarından ilki sisteme kullanıcıyı kaydederken ikinci giriş ekranı kayıtlı kullanıcı için veri oluşturulmasını sağlar. Şekil 4.7. ve 4.9.'da istemcide yer alan kayıt oluşturma ve giriş ekranları sırasıyla gösterilmektedir.

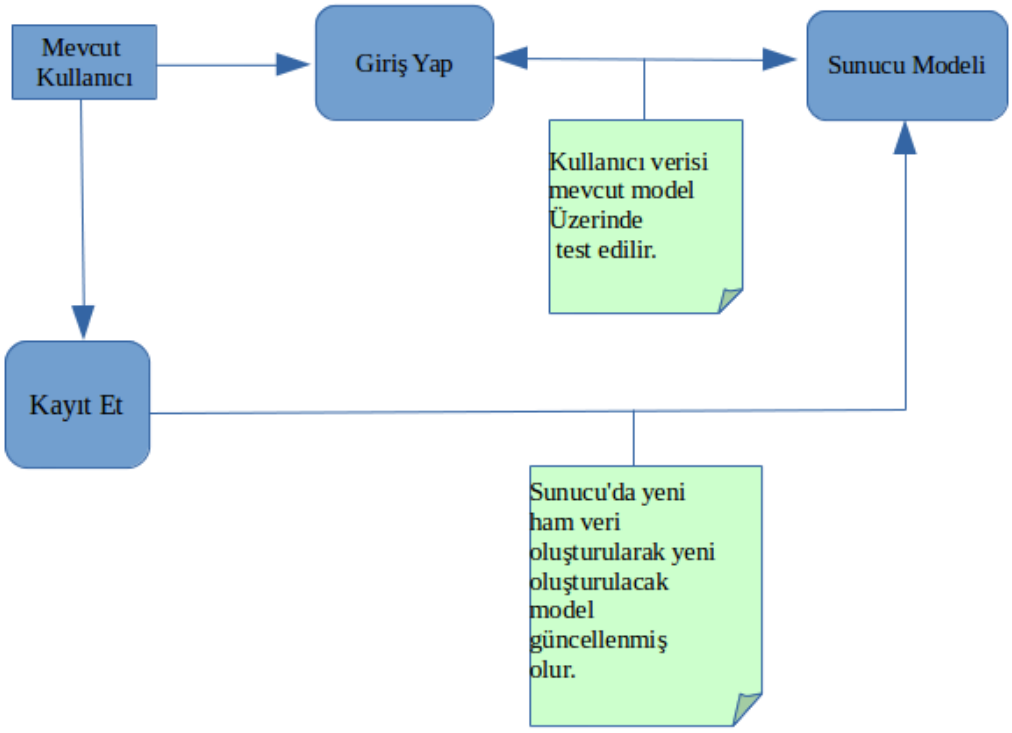
Şekil 4.7. Kullanıcı kayıt ekranı.

Kullanıcı kaydının yapılabilmesi için sunucu tarafında sunucunun istekleri kabul etmesi gereklidir. Sunucu ve istemci arasında ağ bağlantısı ile bu gerçekleşir. İstemci sunucuya girilen kullanıcı adına ait bir klasör mevcut mu diye sorar eğer mevcutsa kullanıcı "Kayıt Et" butonuna bastığında yeni bir klasör oluşturulmaz ve veri giriş ekranına geçiş yapılır. Kullanıcı "Giriş Yap" ile mevcut sistemde kayıtlı ise bu girişi yapıp yapamayacağını test edebilir. Bu şekilde kullanıcının kullanıcı adı giriş verisi model üzerinde test edilmiş olur ve kullanıcı yeni veri girişi oluşturup oluşturmayacağına karar verir. Bu işlem ancak sunucu tarafında kullanıcı için bir model oluşturulmussa yapılabilir. Kullanıcı sistemde kayıtlı değilse giriş

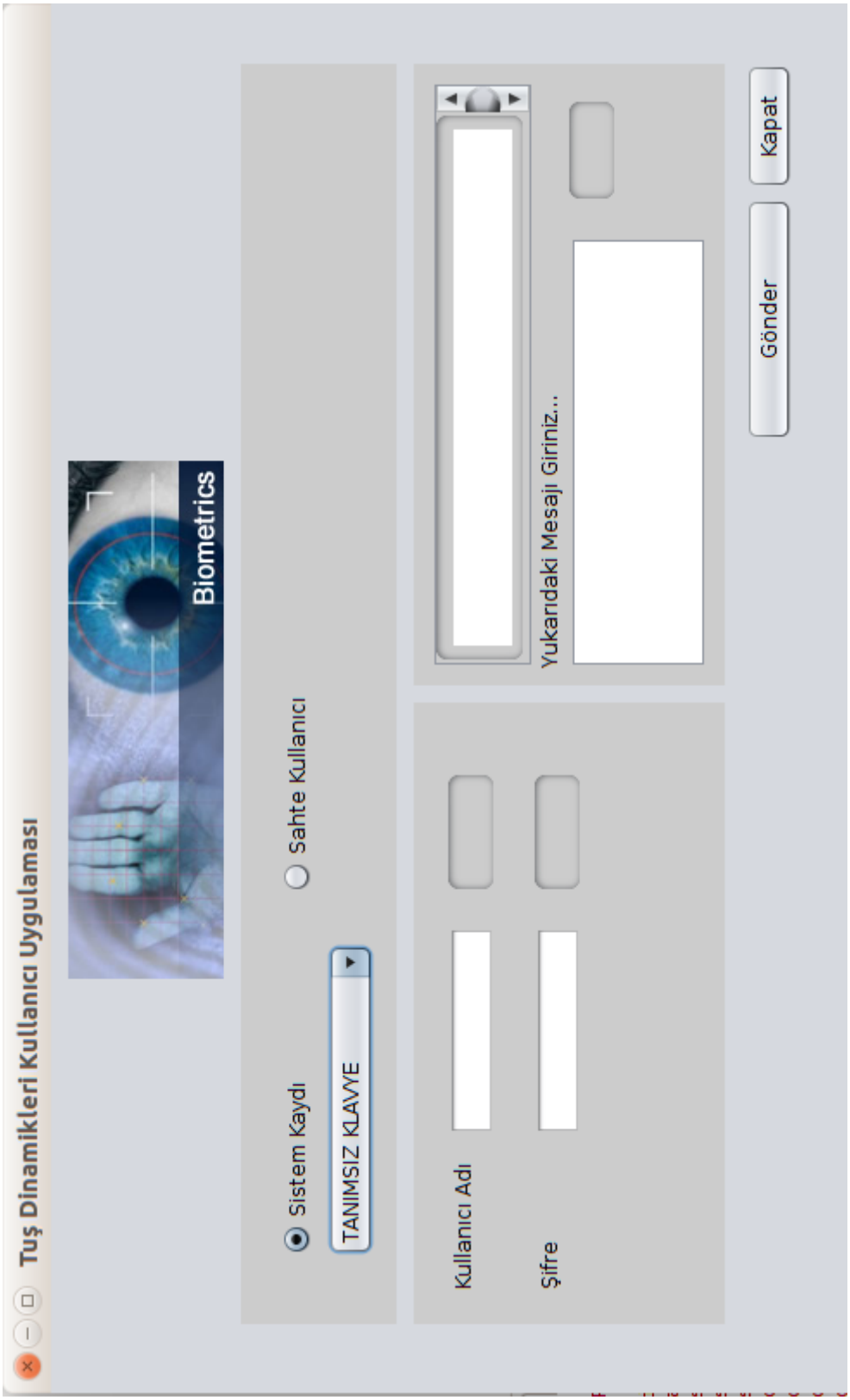
yap butonu aktif olmayacaktır. Tüm toplanan verilerin güvenli bir bilgisayar laboratuarında toplandığı kullanıcıların kimlik bilgileri ile bu laboratuara ulaştığı farz edildiğinde kullanıcı veri oluşturma ekranı başka kullanıcılar tarafından manipüle edilemez. Başka bir kullanıcı mevcut kullanıcının verisini silemez ve değiştiremez.

Şekil 4.9.'da yatay olarak gösterilen kullanıcı verisi giriş ekranı üç adet metin girişinden oluşmaktadır. Bunlar sırasıyla "Kullanıcı Adı", "Şifre" ve "Metin" girişleridir. Kullanıcı adı kullanıcının sisteme kayıt edeceği metin bilgisidir. "Şifre" kullanıcı tarafından belirlenen şifredir. Metin ise her kullanıcıya sunucu tarafından belirlenen sabit metin bilgisidir.

Kullanıcıları birbirinden farklı metinler girmiş olsa da ayırt edebilmek için önceden tanımlanan klavye bilgisi istemci tarafında seçilmelidir. Kullanıcı hangi klavyede giriş yaptıysa o klavyeyi seçmelidir. Şekil 4.9.'da klavye seçimi TANIMSIZ olarak belirtilmiştir. Kullanıcılar girdikleri metin bilgisi üzerinde herhangi bir yineleme ya da değişiklik yapamamaktadır. Bu bağlamda kullanıcılar başta belirledikleri kullanıcı adlarını, tuş dinamiklerini değiştiremezler. Bu işlem için yönetici kullanıcıların belirlediği tarihlerde oluşturduğu ham veri bilgisini seçerek veya silerek yeni ARFF verisi oluşturmalıdır.



Şekil 4.8. Kullanıcı işlemleri.



Şekil 4.9. Kullanıcı verisi oluşturma ekranı.

4.5. Programın Sağladıkları

Sunucu ve istemci mimarisi, toplanan verilerin tek bir merkezde oluşturulmasını ve kullanıcıların bu verilere ulaşamaz olmasını sağlamaktadır. Verilerin güvenliği için herhangi bir şifreleme algoritması kullanılmamış olmasına karşın verilerin toplanacağı bilgisayar veya donanımın bir laboratuar ortamında olması düşünülmüştür. Bu şekilde kullanıcılar güvenli bir şekilde veri girişi yapabileceklerdir.

Klavye tanımlama süreci ile sanal klavye tanımlarının kullanıcıların tuş dinamikleri üzerine etkisinin araştırılabileceği bir veri toplama ve veri işleme programı hazırlanmıştır. Oluşan verilerin işlenmesi ve günümüzde çok sık kullanılan WEKA makine öğrenmesi kütüphanesinin anlayabileceği formata dönüştürülmesi bu programla sağlanmıştır. Makine öğrenmesinde kullanılacak özellikler önceden belirlenerek tüm özellikler bu programda oluşturulmuştur. Bu bağlamda WEKA kütüphanesi kullanılarak farklı özelliklerin sonuca etkisi cabuk bir şekilde gözlenebilir.

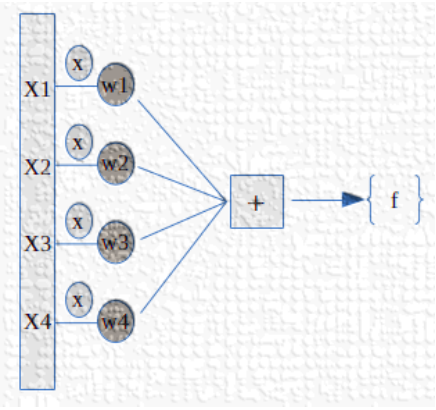
Ayrıca kullanıcılardan birden fazla oturumla veri toplama işlemi gerçekleştirilebildiği için kullanıcıların gün ve gün içerisindeki performanslarını karşılaştırma olanağına sahip olunabilmektedir. Bu şekilde günün farklı zamanlarında kullanıcıları ayırt etme işlemi kolaylaştırılabilir.

5. KULLANILAN YÖNTEMLER

5.1. Yapay Sinir Ağları (Multilayer Perceptron)

Yapay sinir ağları sinir hücrelerindeki öğrenmeyi taklit eden bir öğrenme modelidir. Yapay sinir ağları modeli katmanlardan oluşmaktadır. Her bir katman çeşitli sayıda algılayıcılardan (perceptron) oluşmaktadır. İlk güncel algılayıcı modeli Frank Rosenblatt tarafından 1958 yılında formüle edilmiştir. Ancak 1975 yılında Paul Werbo tarafından geri beslemeli öğrenme modeli geliştirilene kadar makine öğrenmesi modeli olarak kullanılamamıştır. Günümüzde bilgisayar mimarisinde büyük ölçekli tasarımların yaygınlaşması ve öğrenme modelinin paralel olabilmesi sebebiyle kolayca donanım tabanlı hale getirilmiştir (Haykin, 1999).

Yapay sinir ağı modeli aslında çok sayıda algılayıcının belirli bir katman tasarımında ağırlıklandırılarak birleşmesinden oluşur. Algılayıcılar lineer sınıflandırıcılardır. Kısaca bir girdi vektöründen 1 ya da 0 değerlerini alabilen bir çıktıya dönüştürürler. Algılayıcı yapısı Şekil 5.1.'de ve formülasyonu (5.1)'de gösterilmektedir.



Şekil 5.1. Basit algılayıcı yapısı.

$$o = f(\sum_{k=1}^{n} i_k . W_k) \quad (5.1)$$

Şekil 5.1.'de belirtilen yapı bir vektörün ağırlık değeri ile iç çarpımı olarak düşünülebilinir. Bu bağlamda (5.1)'de belirtilen f fonksiyonu değer olarak bir sabit değer almaktadır. Bu fonksiyon çıktı olarak 1 ya da 0 üretecektir. Bunun için genelde aşağıda fonksiyonel ifadesi verilen sigmoid fonksiyonu tercih edilir:

$$f(x,w) = \frac{1}{1+e^{-wx}} \tag{5.2}$$

Bu ifadede w ve x çarpımı iç çarpımdır. Sonucun 1 ya da 0 olması için kesin sınırlayıcı kullanılmalıdır. Örneğin 0.5 değerinden büyük bir sonuç 1'e çevrilirken 0.5'den küçük bir sonuç 0'a çevirilir.

Yapay sinir ağlarında katmanlardaki algılayıcılar geri besleme kullanılarak gradyan azalma (gradyan azalma) öğrenme optimizasyonu ile eğitilmektedir. Gradyan azalma optimizasyonu bir hata fonksiyonun küçültülmesi (minimization) prensibine dayanır. Yapay sinir ağlarında her bir katmandaki girdiye karşılık bir çıktı üretilir. Bu çıktıdaki hata oranı ise çıktının değeri ve bir sonraki katmandaki hatanın bir fonksiyonu olarak geri beslenir. Bu şekilde son çıktı katmanından ilk girdi katmanına doğru tüm çıktılar ve oluşan hatalar hesaplanır (Haykin, 1999).

Sigmoid fonksiyonu ve kesin sınırlandırıcı ile oluşturulan bir algılayıcı iç bükey ya da lineer bir hata düzlemine sahip olacaktır. Bu yüzden öğrenme optimizasyonu için gradyan azalma kullanılabilinir. Gradyan azalma ile fonksiyonun türevi ve hata oranı kullanılarak küçük adımlarla güncellemeler yapılır. Aşağıdaki şekilde sigmoid fonksiyonu kullanan bir algılayıcı için hata güncellemesi verilmiştir.

1. Adım : w vektörü rastgele belirlenir.

2. Adım : t örneği için f fonksiyonun türevi t'de y(t) olarak hesaplanır.

$$w_i(t+1) = w_i(t) + \alpha\left(d_j - y_j(t)\right)x_{j,i} \tag{5.3}$$

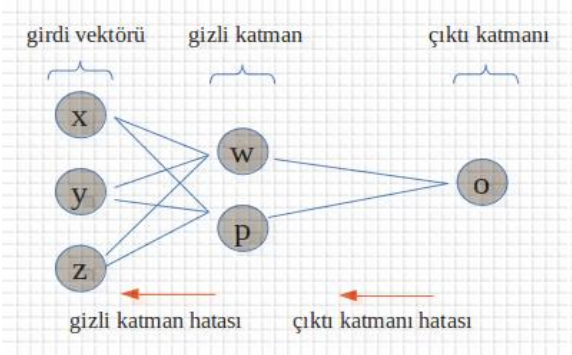
3. Adım : w vektörü i elemanı için güncellenir.

Şekil 5.2. Algılayıcı eğitimi.

Şekil 5.2.'de belirtilen güncelleme algoritmasının 2. ve 3. adımları sonuç yakınsayana kadar her bir örnek için tekrarlanmaktadır. Sonucun yakınsaması artık

fonksiyonun kararlı hale gelmesi ve hata oranının değişmemesi anlamına gelmektedir. Formülde belirtilen alpha değişkeni her bir güncellemenin adımını belirtmektedir. Alpha genelde 0.001 gibi küçük değerler seçildiğinde öğrenme yavaş olacaktır. Tüm örnekler için sırayla güncelleme yapıldığı için bu bir çevirimiçi (online) öğrenme modelidir. Tüm örneklerdeki hata bir kere hesaplanarak yapılan güncellemede ise öğrenme algoritması batch öğrenme olarak adlandırılır.

Şekil 5.3.'de bir algılayıcılardan oluşan bir yapay sinir ağı modeli gösterilmektedir.



Şekil 5.3. Yapay sinir ağı modeli.

Yapay sinir ağları doğrusal olmayan çok sınıflı öğrenme yapabilen sınıflandırıcılardır. Bu yönüyle yapay sinir ağları çok karmaşık sınıflandırma fonksiyonlarına yakınsayabilir. Her bir yapay sinir ağı birden fazla algılayıcının birden fazla katman ile birleşmesinden oluşur. Şekil 5.3.'de belirtilen yapay sinir ağı modeli 3 girdi alarak 1 çıktı üretmektedir. Yukarıda Şekil 5.3.'de gösterilen yapay sinir ağı modeli 2 katmanla ifade edilmektedir. Yapay sinir ağlarında her bir algılayıcıdaki hata oranı çıktı düğümünde hesaplanarak girdi düğümlerine doğru her bir düğüm için hesaplanır. Bu hesaplama yöntemi geri yayılım (backpropagation) olarak adlandırılır. Geri yayılım algoritmasında kullanılan fonksiyonun türevi kullanılarak hata oranı hesaplanır ve güncelleme değeri şu şekilde hesaplanır:

$$\Delta w_{hl}^{(n)} = \mu \sum_p delta_l^{(n)} . out_h^{(n-1)} \qquad (5.4)$$

Delta güncellemesinin hesaplamaları çıktı katmanı hatası ve gizli katman hatası için sırasıyla şu şekilde belirtilebilir:

$$delta_k^{(N)} = \left(targ_k - out_k^{(N)}\right).out_k^{(N)}.(1 - out_k^{(N)}) \qquad (5.5)$$

$$delta_k^{(n)} = \left(\sum_k delta_k^{(n+1)}.w_{lk}^{(n+1)}\right).out_k^{(n)}.(1 - out_k^{(n)}) \qquad (5.6)$$

Bu eşitliklerde belirtilen $out_k^{(N)}.(1 - out_k^{(N)})$ değeri k düğümündeki fonksiyonun belirtilen çıktı cinsinden türevidir. Bu sigmoid fonksiyonun o çıktıdaki türevidir. Farklı fonksiyonlar için bu türev farklı hesaplanacaktır. (5.5)'de belirtilen $(targ_k - out_k^{(N)})$ yapay sinir ağının son katmanında oluşan hatadır. Bir yapay sinir ağı çoklu sınıflandırma özelliğine sahip olabilmesi için birden fazla çıktı vermelidir. (5.5)'de belirtilen $delta_k^{(N)}$'de N ile belirtilen bu değer hangi çıktı düğümünün güncellemesi olduğunu belirtmektedir (Haykin, 1999).

5.2. Karar Ağaçları

Karar Ağaçları; parametrik olmayan yani bir parametreye yakınsamadan öğrenen hiyerarşik sınıflandırıcılardandır. Hem regresyonda hem de sınıflandırmada kullanılırlar. Karar ağaçları ara düğüm ve yapraklarda bulunan uç düğüm noktalarından oluşur. Her bir ara düğüm bir özelliğe göre tek bir sınıflandırma yaparak girilen bir değeri kökten yapraklara doğru sınıflandırır. Düğümlerde sınıflandırma için seçilen özellikler tepede entropisi en yüksek olandan aşağıda entropisi en düşük olana doğrudur. Çünkü bir özelliğin entropisinin yüksek olması özelliğin ayırt ediciliğinin de yüksek olduğunun bir göstergesidir. Ağaçın tepesindeki bir düğümde yapılan ayrıştırma daha çok veriyi bir çırpıda sınıflandırırken ağacın yapraklarında bu ayrıştırma sınırlı olur. Dolayısıyla tepedeki düğüme gelecek özelliği doğru seçmek önemlidir. Karar ağaçları bu sınıflandırmayı yaparken belirsizliği en aza indirme hedefine yakınsamaktadır. Belirsizliğin hesaplanmasında aşağıda belirtilen entropi fonksiyonu kullanılır (Alpaydın, 2014):

$$H[D] = -\sum_{j=1}^{|C|} P(c_j) \log_2 P(c_j) \quad (5.7)$$

Bu ifadede belirtilen D veriseti örneklemi her bir c_j sınıfı için hesaplanır. Örneğin bir veri setinde H1N1 virüsünün kandaki oranlara göre test edilmesi gerçekleştirilsin. Veri setimizin %50'si pozitif ve %50'si negatif olsun. Bu durumda entropi şu şekilde hesaplanır.

$$entropy(D) = -0.5xlog_2 0.5 - 0.5xlog_2 0.5 = 1 \quad (5.8)$$

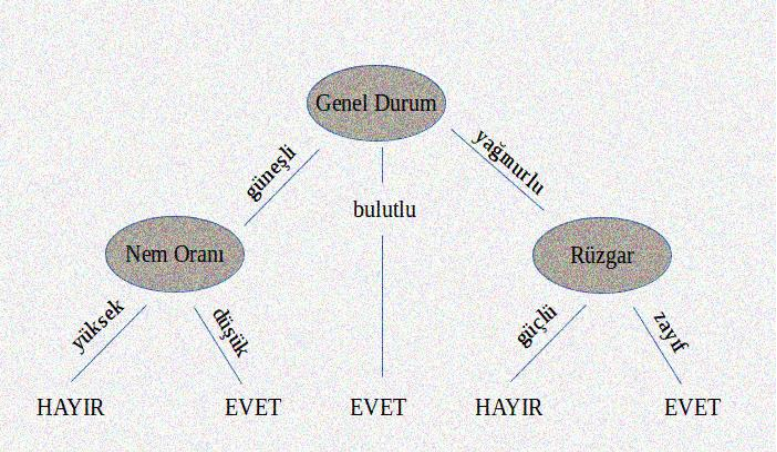
Karar ağaçlarında bir düğümde birden çok sınıflandırma yapılabilinir. Böyle bir durumda düğümdeki entropi değeri yapılan sınıflandırmanın veri setinin sınıflarını kaç örneğe böldüğüne göre entropi hesabı yapılarak bulunur. Bunun için şu oran kullanılmıştır:

$$H_{A_i}[D] = \sum_{j=1}^{v} \frac{|D_j|}{|D|} H[D_j] \quad (5.9)$$

Burada v veri setinin o düğüme göre dallanma sayısıdır. $|D_j|$ düğümde dallanmış veri setinin boyutu ve |D| tüm veri setinin boyutudur. $H[D_j]$ ise veri setinin o daldaki entropisidir. Aşağıdaki eşitlikte ise belirli bir özelliğe göre oluşturulan düğümdeki dalın ayrıştırma kazancı ölçülmektedir.

$$gain(D, A_i) = H[D] - H_{A_i}[D] \quad (5.10)$$

Ayrıştırma kazancı yüksek ise, o zaman seçilen özellik, ağacın yukarısında yer alır. Aşağı doğru diğer daha az ayrıştırıcı olan özellikler de düğümlerde yerini alır.



Şekil 5.4 Karar ağacında sınıflandırma.

Şekil 5.4.'de belirtilen karar ağacı üç özelliğe bakarak ikili bir sınıflandırma yapmaktadır. Karar ağaçları sınıflandırma hatası en aza inene kadar arama uzayını belirli özelliklerin aldığı değerlere göre alt parçalara bölerek sınıflandırma yapar. Şekil 5.4'de, tüm yaprak düğümleri bir sınıfa atama yapacak şekilde etiketlenmiştir. Bu yolla varılan yaprak düğümler sınıfları belirlemiş olur (Alpaydın, 2014).

5.3. Naive Bayes

Naive Bayes sınıflandırıcısı a_i gibi belirli özellikler verildiğinde en olası sınıfı bulan istatistiksel bir sınıflandırıcıdır. (5.11)'de verilen fonksiyonda her bir sınıfın tüm özelliklerinin olasılığı çarpılarak elde edilen bir fonksiyon verilmiştir. Bu fonksiyonda P($a_i|v_j$) ile belirtilen koşullu olasılığın (m-yakınsama olasılığı) hesaplanması (5.12)'de gösterilmektedir (Bishop, 2006).

$$V_{nb} = argmax_{v_j \epsilon V} P(v_j) \pi P(a_i|v_j) \tag{5.11}$$

$$\pi P(a_i|v_j) = \frac{n_c + mp}{n+m} \tag{5.12}$$

Denklem (5.12) için terimler;

***n_c:** Verilen bir sınıf ve özelliğin birlikte geçme sayısını

***n :** Verilen bir sınıfın geçme sayısını

***m :** Herhangi bir katsayıyı

***p :** Verilen bir sınıfın önsel olasılığını

ifade etmektedir.

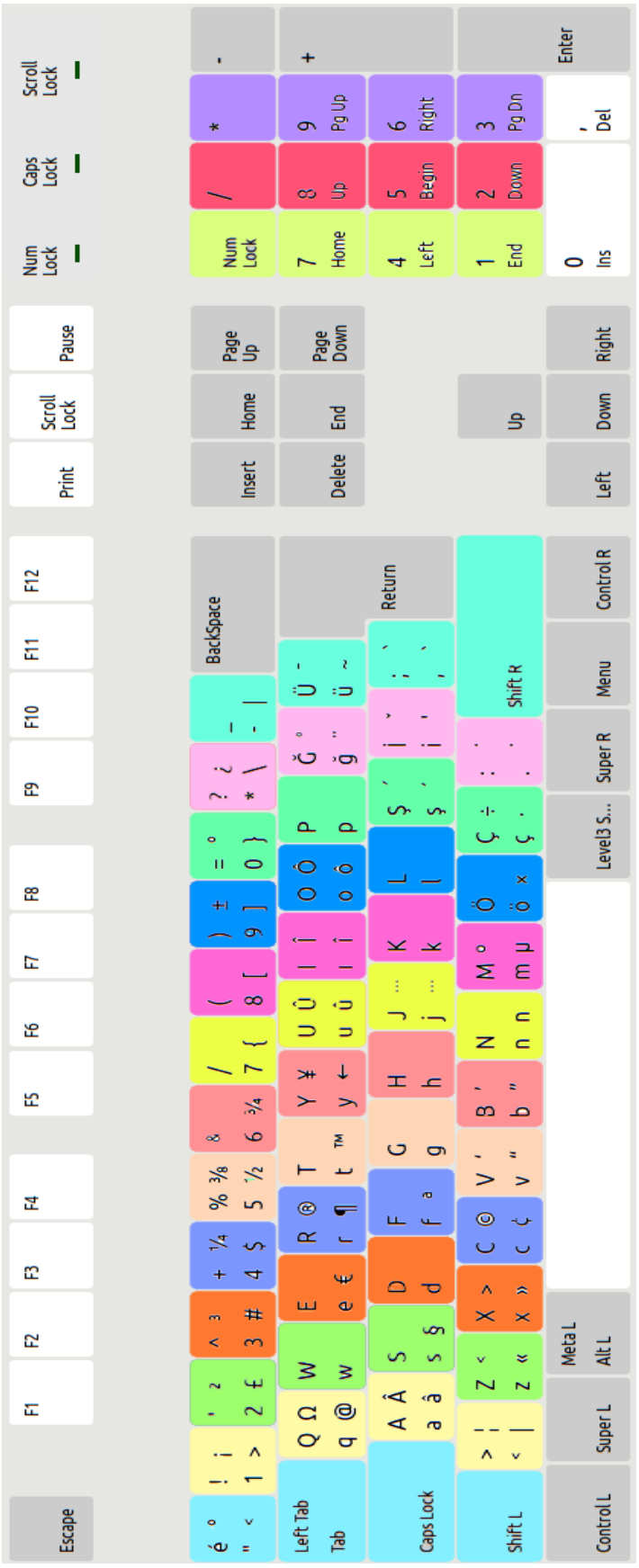
Naive Bayes sınıflandırıcılarda özelliklerin birbirleri ile korelasyonu göz önünde bulundurulmadığı gibi herhangi bir parametreye yakınsamazlar. Bu yönleriyle Naive Bayes sınıflandırıcılar basit ama etkili sınıflandırıcılardır. Sayısal verilerin örneğin tuşa basma dinamiklerinde uçuş süresinin naïve bayes sınıflandırıcı ile kullanılması için veri çiftleme işlemi yapılmalıdır. Veri çifleme bir fonksiyon kullanarak sayısal veriyi kategorize etmek için kullanılır. Veri çiftleme işlemi dışında (5.12) için bir olasılık yoğunluk fonksiyonu da veriyi olasılıklandırmada kullanılabilinir (Bishop, 2006).

6. SONUÇLAR VE ÖNERİLER

Deneyler farklı kullanıcılardan oluşan iki grup tarafından gerçekleştirilmiştir. Her grup 50 kullanıcıdan oluşmaktadır. Tuş dinamiklerini ölçmek için her bir kullanıcıdan, sisteme, 12 karakterden oluşan '*anSmV*5kdo3.*' metnini giriş yapmaları istenmiştir. Veri toplama işlemi 5 oturumda gerçekleştirilmiştir. Her oturumda da belirtilmiş olan sabit metnin 5 kez sisteme girilmiş olması gerekmektedir. Oturum sonunda 5 girişin ortalaması alınarak bir özellik vektörü çıkartılmaktadır. 5 oturum sonunda her kullanıcı için 5 özellik vektörü bulunmaktadır. Toplamda her kullanıcı bu metni sisteme 25 kez girmiş olmaktadır.

Testler iki farklı grup verileri için WEKA makine öğrenmesi yazılımı üzerinde yapılmıştır. Test sonuçları WEKA'nın Naïve Bayes, J48, RBFNetwork ve MultiLayer Perceptron makine öğrenmesi yöntemleri kullanılarak elde edilmiştir. Bu çalışmada YRO ve YKO metrikleri dışında doğruluk oranı ölçülmüştür. Her kullanıcı için tüm veri girişleri ve testler aynı tür klavye üzerinde gerçekleştirilmiştir. Testler için iki farklı klavye tasarımı oluşturulmuştur. Bunlardan biri klavyedeki alanların yatay olarak üçerli gruplar halinde 20 alana bölünmesi ile oluşturulan tasarım olan Şekil 6.1.'de görülen Tasarım 1, diğeri klavyedeki alanların dikey olarak dörtlü gruplar halinde 16 alana bölünmesi ile oluşturulan tasarım olan Şekil 6.2.'de görülen Tasarım 2'dir.

Şekil 6.1. Tasarım 1 (Klavye düzeninin yatay olarak 20 alana bölünmüş olan tasarımı).



Şekil 6.2. Tasarım 2 (Klavye düzeninin dikey olarak 16 alana bölünmüş olan tasarımı).

Aşağıdaki tablolarda oluşturulan her bir grup için her bir tabloda 4 farklı sınıflandırıcı için elde edilen lokasyonlu alanlı (tasarım1 ve tasarım2), lokasyonlu alansız, lokasyonsuz alanlı (tasarım1 ve tasarım2) ve lokasyonsuz alansız olmak üzere 6 değer gösterilmektedir.

I. deney grubu

Kullanıcı Sayısı : 50

Çizelge 6.1. Simülasyon sonuçları I.

YÖNTEM	SINIFLANDIRICI	LOKASYONLU			LOKASYONSUZ		
		ALANLI		ALANSIZ	ALANLI		ALANSIZ
		TASARIM 1	TASARIM 2		TASARIM 1	TASARIM 2	
İstatiksel Yöntemler	Naive Bayes	62.5	65.35	61.07	61.07	63.57	55.71
Karar Ağaçları	J48	50	51.42	43.92	46.42	51.42	42.14
Yapay Sinir Ağları	RBF Network	65.35	70.35	64.64	65.35	69.28	61.07
Yapay Sinir Ağları	MultiLayer Perceptron	78.57	84.64	76.42	77.14	81.07	72.85

II. deney grubu

Kullanıcı Sayısı : 50

Çizelge 6.2. Simülasyon sonuçları II.

YÖNTEM	SINIFLANDIRICI	LOKASYONLU			LOKASYONSUZ		
		ALANLI		ALANSIZ	ALANLI		ALANSIZ
		TASARIM 1	TASARIM 2		TASARIM 1	TASARIM 2	
İstatiksel Yöntemler	Naive Bayes	59.40	65.68	59.40	58.67	63.83	56.82
Karar Ağaçları	J48	39.85	50.18	39.85	39.85	49.81	40.59
Yapay Sinir Ağları	RBF Network	59.77	62.36	58.67	59.77	60.88	55.71
Yapay Sinir Ağları	MultiLayer Perceptron	77.85	81.54	77.49	72.69	80.07	73.06

Aynı kullanıcı sayısına sahip fakat farklı kullanıcılardan oluşan her iki grup için de birbirine yakın sonuçlar elde edilmiştir. Alan tabanlı özellikler iki grupta da doğruluk oranının yüksek değerlerde çıkmasını sağlamıştır.

Aşağıdaki tablolarda kullanıcı sayıları farklı olarak yapılan deney sonuçları verilmiştir. Çizelge 6.3.'de her iki grubta olan tüm kullanıcıların katıldığı simülasyon sonuçları bulunmaktadır. Çizelge 6.4.'de her gruptan rasgele seçilerek oluşturulan 25 kullanıcının katıldığı simülasyon sonuçları, aynı şekilde Çizelge 6. 5.'de ise rasgele seçilen 10 kullanıcının katıldığı simülasyon sonuçları bulunmaktadır.

İki Grubun Birleşimi

Kullanıcı Sayısı : 100

Çizelge 6.3. Simülasyon sonuçları III.

YÖNTEM	SINIFLANDIRICI	LOKASYONLU			LOKASYONSUZ		
		ALANLI		ALANSIZ	ALANLI		ALANSIZ
		TASARIM 1	TASARIM 2		TASARIM 1	TASARIM 2	
İstatiksel Yöntemler	Naive Bayes	55.95	55.41	54.33	54.15	56.13	49.63
Karar Ağaçları	J48	35.55	35.55	35.55	34.29	42.77	34.11
Yapay Sinir Ağları	RBF Network	58.30	61.01	56.31	57.94	59.92	51.26
Yapay Sinir Ağları	MultiLayer Perceptron	70.93	73.64	70.75	70.03	74.54	66.06

Her gruptan seçilen rasgele kullanıcılar

Kullanıcı Sayısı : 25

Çizelge 6.4. Simülasyon sonuçları IV.

YÖNTEM	SINIFLANDIRICI	LOKASYONLU			LOKASYONSUZ		
		ALANLI		ALANSIZ	ALANLI		ALANSIZ
		TASARIM 1	TASARIM 2		TASARIM 1	TASARIM 2	
İstatiksel Yöntemler	Naive Bayes	71.52	70.19	69.53	72.18	70.19	67.54
Karar Ağaçları	J48	52.31	67.54	50.99	67.54	67.54	51.65
Yapay Sinir Ağları	RBF Network	71.52	72.84	70.19	73.51	73.51	66.88
Yapay Sinir Ağları	MultiLayer Perceptron	86.09	86.75	83.44	86.09	86.09	78.14

Her gruptan seçilen rasgele kullanıcılar

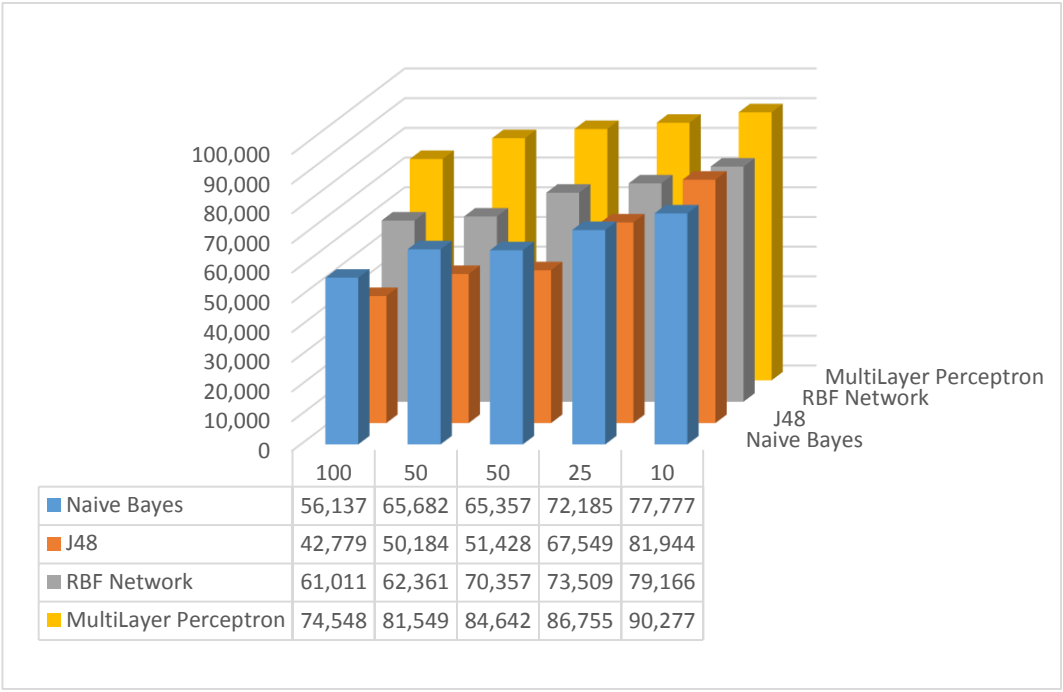
Kullanıcı Sayısı : 10

Çizelge 6.5. Simülasyon sonuçları V.

YÖNTEM	SINIFLANDIRICI	LOKASYONLU			LOKASYONSUZ		
		ALANLI		ALANSIZ	ALANLI		ALANSIZ
		TASARIM 1	TASARIM 2		TASARIM 1	TASARIM 2	
İstatiksel Yöntemler	Naive Bayes	75	77.77	73.61	76.38	76.38	73.61
Karar Ağaçları	J48	73.61	81.94	73.61	81.94	81.94	69.44
Yapay Sinir Ağları	RBF Network	75	79.16	72.22	75	75	72.22
Yapay Sinir Ağları	MultiLayer Perceptron	79.16	87.5	76.38	90.27	90.27	77.77

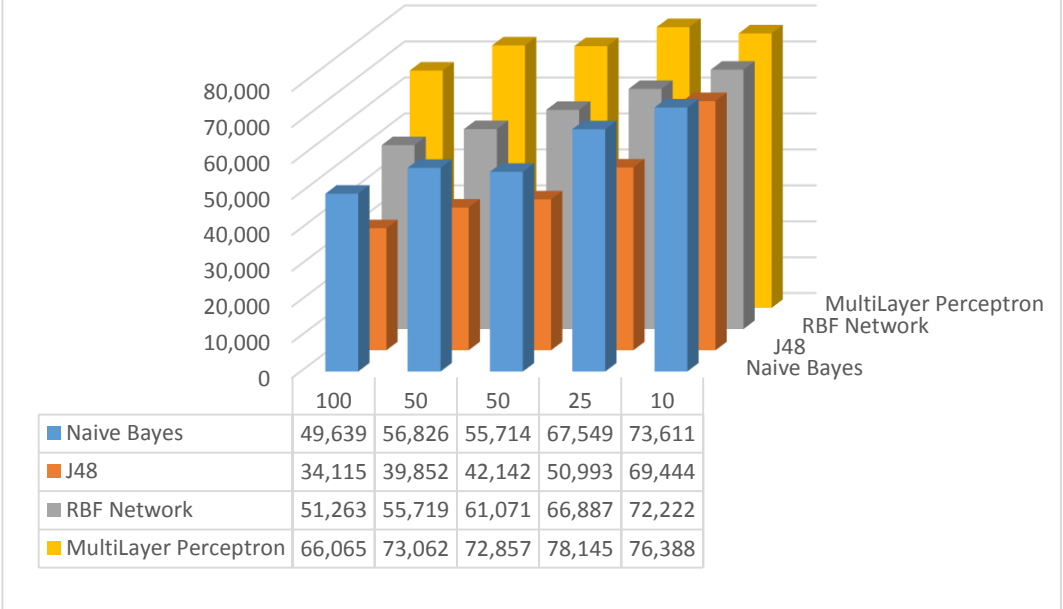
Farklı sayıda kullanıcılardan oluşturularak yapılan deneylerden elde edilen sonuçlara göre de alan tabanlı özelliklerin, farklı kullanıcı sayısındaki deneylerde ve/veya farklı yapay zeka makine öğrenmesi yöntemlerinde, doğruluğu arttırmada etkili olduğu ortaya çıkmıştır.

Aşağıda elde edilen tüm çizelge değerlerinin kullanıcı sayılarına göre karşılaştırılmış olan durumları grafik şeklinde gösterilmiştir. Grafiklerdeki çizelgelerde, sütunların başlıkları deneylere katılan kullanıcı sayılarını göstermektedir. Sırayla Şekil 6.3. ve Şekil 6.4.'de her deney grubunun her yöntemdeki aldığı en yüksek ve en düşük değerlere göre oluşturulan grafikler bulunmaktadır.



	100	50	50	25	10
Naive Bayes	56,137	65,682	65,357	72,185	77,777
J48	42,779	50,184	51,428	67,549	81,944
RBF Network	61,011	62,361	70,357	73,509	79,166
MultiLayer Perceptron	74,548	81,549	84,642	86,755	90,277

Şekil 6.3. Her tablodaki her yöntem için alınan en yüksek değerlerin kullanıcı sayılarına göre karşılaştırılması.



	100	50	50	25	10
Naive Bayes	49,639	56,826	55,714	67,549	73,611
J48	34,115	39,852	42,142	50,993	69,444
RBF Network	51,263	55,719	61,071	66,887	72,222
MultiLayer Perceptron	66,065	73,062	72,857	78,145	76,388

Şekil 6.4. Her tablodaki her yöntem için alınan en düşük değerlerin kulalnıcı sayılarına göre karşılaştırılması.

Grafiklerden elde edilen sonuçlar doğruluk oranının her bir sınıflandırıcıda kullanıcı sayısı ile ters orantılı olduğunu göstermektedir. Deneye katılan kullanıcı sayısı azaldıkça başarı oranı artmaktadır.

Yapılan tüm deneylerde dört farklı makine öğrenmesi metodunun doğruluk üzerinden başarısı ölçülmüştür. Aynı sayıda kullanıcı içeren iki farklı kullanıcı grubunda da yaklaşık aynı sonuçlar ortaya çıkmıştır. Tüm tablolardaki değerlere bakıldığında yeni getirilen özelliklerin olduğu durumlardaki doğruluk oranlarının daha fazla olduğu görülmektedir. Alan tabanlı özelliklerin de katılarak yapıldığı testlerde dört metod için de daha etkili sonuçlar elde edilmiştir. Alan tabanlı özellikler klavye tasarımlarına göre oluşmaktadır. Bu tasarımlar da çok farklı sonuçlar üretmektedir. Bu yüzden klavye üzerinde alanların tasarımı çok önemlidir. İki farklı tasarım kullanılarak elde edilen özellikler arasından dikey bölümlendirme ile elde edilen tasarım olan Tasarım 2 daha başarılı olmuştur. Test sonuçları arasından alan bilgisi ve lokasyon bilgisinin yer aldığı sonuçlar en başarılı olanlarıdır. Klavye tasarımının getirdiği bu özellikler başarıyı fark edilir ölçüde artırmıştır. Sonraki çalışmalarda kullanıcılar için farklı türde klavye tasarımları oluşturularak kullanıcıya özgü analizler yapılabilinir.

KAYNAKLAR

Abualgasim, S. D., & Osman, I. (2011). An Application of the Keystroke Dynamics Biometric for Securing PINs and Passwords. *World of Computer Science and Information Technology Journal (WCSIT)*, 2221-0741.

Alpaydın, E., Introduction To Machine Learning, The MIT Press, London, England, 2014.

Anonim (2015a), Biyometrik Sistemler Nelerdir http://www.genvatek.com.tr/bilgi-biyometrik-sistemler-nelerdir-44.html

Anonim (2015b), Biyometrik Tanıma Sistemleri http://www.guvenlikdanismanlik.com/biyometrik-tanima-sistemleri.htm

Anonim (2015c), Şifresiz Güvenlik http://www.bilimania.com/bilisim-teknolojileri/35-bilisim- teknolojileri/3523- sifresiz-guevenlik

Anonim (2015d), Keystroke Dynamics http://www.biometric-solutions.com/solutions/index.php?story=keystroke_dynamics

Bergadano, F., Gunetti, D., & Picardi, C. (2002). User authentication through keystroke dynamics. *ACM Transactions on Information and System Security (TISSEC)*, *5*(4), 367-397.

Bishop, C.M., Pattern Recognition and Machine Learning, Microsoft Research Ltd, Cambridge U.K., 2006.

Bleha, S., Slivinsky, C., & Hussien, B. (1990). Computer-access security systems using keystroke dynamics. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *12*(12), 1217-1222.

Bleha, S. A., & Obaidat, M. S. (1993). Computer users verification using the perceptron algorithm. *Systems, Man and Cybernetics, IEEE Transactions on*,*23*(3), 900-902.

Can, Y. S., & Alagoz, F. (2014, April). User identification using Keystroke Dynamics. In *Signal Processing and Communications Applications Conference (SIU), 2014 22nd* (pp. 1083-1085). IEEE.

Chellappa, R., Wilson, C. L., & Sirohey, S. (1995). Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, *83*(5), 705-741.

Cho, S., Han, C., Han, D. H., & Kim, H. I. (2000). Web-based keystroke dynamics identity verification using neural network. *Journal of organizational computing and electronic commerce*, *10*(4), 295-307.

Daugman, J. G. (1993). High confidence visual recognition of persons by a test of statistical independence. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *15*(11), 1148-1161.

Ergen, B., & Çalışkan, A. (2011, May). Biyometrik Sistemler ve El Tabanlı Biyometrik Tanıma Karakteristikleri. In *6th International Advanced Technologies Symposium (IATS'11)* (pp. 16-18).

Guven, A., & Sogukpinar, I. (2003). Understanding users' keystroke patterns for computer access security. *Computers & Security*, *22*(8), 695-706.

Haykin, S., NEURAL NETWORKS A Comprehensive Foundation, Prentice Hall International, Toronto, Canada, 1999.

Ilonen, J. (2003). Keystroke dynamics. *Advanced Topics in Information Processing–Lecture*, 03-04.

Joyce, R., & Gupta, G. (1990). Identity authentication based on keystroke latencies. *Communications of the ACM*, *33*(2), 168-176.

Karnan, M., Akila, M., & Krishnaraj, N. (2011). Biometric personal authentication using keystroke dynamics: A review. *Applied Soft Computing*,*11*(2), 1565-1573.

Killourhy, K. S., & Maxion, R. (2009, June). Comparing anomaly-detection algorithms for keystroke dynamics. In *Dependable Systems & Networks, 2009. DSN'09. IEEE/IFIP International Conference on* (pp. 125-134). IEEE.

Lee, H. J., & Cho, S. (2007). Retraining a keystroke dynamics-based authenticator with impostor patterns. *Computers & Security*, *26*(4), 300-310.

Monrose, F., & Rubin, A. D. (2000). Keystroke dynamics as a biometric for authentication. *Future Generation computer systems*, *16*(4), 351-359.

Musayeva, G., Yahyayev M., (2015), Biyometrik Güvenlik Sistemleri http://www.researchgate.net/publication/271210599_Biyometrik_Gvenlik

Obaidat, M. S., & Sadoun, B. (1997). Verification of computer users using keystroke dynamics. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, *27*(2), 261-269.

Oysal, Y., Polat, H., Akınlar, C., Şora Günal, E., "Biyometrik Temelli Güvenlik Sistemleri", "Güvenlik Sistemleri" (Ed: Oysal, Y.), Anadolu Üniversitesi Açıköğretim Fakültesi Yayını, Eskişehir, Türkiye, 126-144, 2012.

Özkaya, N., Sağıroğlu, Ş., (2015), Açık Anahtar Altyapısı ve Biyometrik Teknikler http://www.researchgate.net/publication/251743851_AIK_ANAHTAR_ALTYAPISI_VE_BYOMETRK_TEKNKLER

Revett, K., Gorunescu, F., Gorunescu, M., Ene, M., Magalhaes, S., & Santos, H. (2007). A machine learning approach to keystroke dynamics based user authentication. *International Journal of Electronic Security and Digital Forensics*, *1*(1), 55-70.

Shanmugapriya, D., & Padmavathi, G. (2009). A survey of biometric keystroke dynamics: Approaches, security and challenges. *arXiv preprint arXiv:0910.0817*.

Shanmugapriya, D., & Padmavathi, G. (2011). Virtual key force—a new feature for keystroke. *Int J Eng Sci Technol*, *3*(10), 7738-7743.

Subramaniam, K. S., Bharath, S. R., & Ravinder, S. (2007, March). Improved authentication mechanism using keystroke analysis. In *Information and Communication Technology, 2007. ICICT'07. International Conference on* (pp. 258-261). IEEE.

Şamlı, R., & Yüksel, M. E. (2009). Biyometrik güvenlik sistemleri. *Akademik Bilişim, 9.*

Şan, S., *Parmak damar tanıma teknolojisi*, Yüksek Lisans Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elazığ, 2013.

Teh, P. S., Teoh, A. B. J., Ong, T. S., & Neo, H. F. (2007, December). Statistical fusion approach on keystroke dynamics. In *Signal-Image Technologies and Internet-Based System, 2007. SITIS'07. Third International IEEE Conference on* (pp. 918-923). IEEE.

Woodward, J. D., Orlans, N. M., & Higgins, P. T. (2003). *Biometrics:[identity assurance in the information age]*. New York: McGraw-Hill/Osborne.

Yu, E., & Cho, S. (2004). Keystroke dynamics identity verification—its problems and practical solutions. *Computers & Security*, *23*(5), 428-440.

Zhang, D., & Shu, W. (1999). Two novel characteristics in palmprint verification: datum point invariance and line feature matching. *Pattern Recognition*, *32*(4), 691-702.

Zhong, Y., Deng, Y., & Jain, A. K. (2012, June). Keystroke dynamics for user authentication. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on* (pp. 117-123). IEEE.

Zhou, C. (2008). *A Study of Keystroke Dynamics as a Practical Form of Authentication* (Doctoral dissertation, MSc Thesis, Pomona College).