



T.C.
NECMETTİN ERBAKAN ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



**ARAÇLARIN SEYİR ALIŞKANLIKLARININ
İZLENEREK SUÇA KARIŞMA
POTANSİYELLERİNİN YAPAY SİNİR
AĞLARIYLA TAHMİN EDİLMESİ**

Niyazi GÖKDOĞAN

YÜKSEK LİSANS TEZİ

Mekatronik Mühendisliği Anabilim Dalı

Mayıs-2021
KONYA
Her Hakkı Saklıdır

TEZ KABUL VE ONAYI

Niyazi GÖKDOĞAN tarafından hazırlanan “*ARAÇLARIN SEYİR ALIŞKANLIKLARININ İZLENEREK SUÇA KARIŞMA POTANSİYELLERİNİN YAPAY SİNİR AĞLARIYLA TAHMİN EDİLMESİ*” adlı tez çalışması 07/05/2021 tarihinde aşağıdaki jüri tarafından oy birliği / oy çokluğu ile Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü Mekatronik Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

Başkan

Doç. Dr. İbrahim ÇAYIROĞLU

Danışman

Prof. Dr. Mehmet KARALI

Üye

Dr. Öğr. Üyesi Yusuf UZUN

İmza

.....

.....

.....

Fen Bilimleri Enstitüsü Yönetim Kurulu’nun/.../2021 gün ve sayılı kararıyla onaylanmıştır.

Prof. Dr. İbrahim KALAYCI
FBE Müdürü

TEZ BİLDİRİMİ

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

DECLARATION PAGE

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

Niyazi GÖKDOĞAN

Tarih: 07/05/2021

ÖZET

YÜKSEK LİSANS TEZİ

ARAÇLARIN SEYİR ALIŞKANLIKLARININ İZLENEREK SUÇA KARIŞMA POTANSİYELLERİNİN YAPAY SİNİR AĞLARIYLA TAHMİN EDİLMESİ

Niyazi GÖKDOĞAN

Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü
Mekatronik Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Mehmet KARALI

2021, 80 Sayfa

Jüri

Prof. Dr. Mehmet KARALI
Doç. Dr. İbrahim ÇAYIROĞLU
Dr. Öğr. Üyesi Yusuf UZUN

Trafik denetim uygulamaları adli ve idari olarak ikiye ayrılmaktadır. Adli denetimlerde belirli araçlar durdurulurken, idari denetimlerde araç ve sürücünün anlık durumu göz önüne alınarak, tecrübeye dayalı uygulama gerçekleştirilmektedir.

Bu tez çalışmasında, araçların plaka bilgilerine dayalı olarak araçların geçmiş sürüş hareketleriyle yapay sinir ağlarını eğiterek, araç ve sürücüsü hakkında muhtelif öngörülerde bulunan, suçlu tespitini kolaylaştıran bir yazılım algoritması geliştirilmiştir. Konya Merkez'deki araç kullanıcılarının sosyal, kültürel ve mesleki seyir davranışları düşünülerek 124 kamera etiketi, ilgili mekân ve yerlere Yandex Maps aracılığıyla atanmıştır. Etiketler, Python programlama dili kullanılarak enlem ve boylam bilgilerine göre Konya haritası üzerine açıklamalarıyla yerleştirilmiştir. Riskli ve risksiz temsili 7942 araca ait 476520 veri üretilerek, büyük veri (Big Data) oluşturulmuştur. One Hot metoduyla kodlanan veriler 3 ara katmanlı yapay sinir ağı ile eğitime girerek yapay sinir ağı modeli tamamlanmıştır. Raspberry Pi ile optik karakter tanıma yapılarak temsili plakalar okunmuş, bir IoT (nesnelerin interneti) bulutu olan ThingSpeak'e aktarılmıştır. 2 modül arasındaki iletişim ThingSpeak üzerinden sağlanmıştır. Yapay sinir ağı eğitimi sonucunda teorik yaklaşık %95 doğruluk elde edilmiştir. Test verileri incelendiğinde bu doğruluk pratikte de elde edilmiştir. Kullanıcı tarafından kolayca anlaşılabilen arayüz hazırlanarak, arkasına iletişim ve risk tahminleme algoritması gömülmüştür. Arayüz yazılımı, görsel ve yazılı tavsiyelerde bulunarak kullanıcıya temsili gelen araç hakkında öneri sunabilmiştir.

Anahtar Kelimeler: Big Data, Görüntü İşleme, Trafik Denetleme, Yapay Sinir Ağı

ABSTRACT

MS THESIS

TRACKING OF THE VEHICLE NAVIGATION HABITS AND PREDICTION OF PROBABILITY OF INVOLVEMENT IN A CRIME BY USING ARTIFICIAL NEURAL NETWORKS

Niyazi GÖKDOĞAN

**THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE OF
NECMETTİN ERBAKAN UNIVERSITY
THE DEGREE OF MASTER OF SCIENCE
IN MECHATRONIC ENGINEERING**

Advisor: Prof. Dr. Mehmet KARALI

2021, 80 Pages

Jury

Prof. Dr. Mehmet KARALI

Assoc. Prof. Dr. İbrahim ÇAYIROĞLU

Asst. Prof. Dr. Yusuf UZUN

Traffic control practices are divided into two as judicial and administrative. While certain vehicles are stopped in forensic inspections, an experience-based practice is carried out by taking the instant situation of the vehicle and driver into consideration in administrative inspections.

In this thesis, a software algorithm has been developed that makes various predictions about the vehicle and its driver and facilitates criminal detection by training the artificial neural networks with the past driving movements of the vehicles based on the license plate information of the vehicles. Considering the social, cultural and professional navigation behaviors of vehicle users in Konya Center, 124 camera tags have been assigned to the relevant locations and locations via Yandex Maps. Tags are placed on the Konya map with their explanations according to latitude and longitude information using Python programming language. Big Data was created by producing 476 520 data belonging to 7942 vehicles representing risky and risk-free vehicles. The data encoded with the One Hot method were trained with 3 interlayer artificial neural networks and the artificial neural network model was completed. Representative plates were read by using optical character recognition with Raspberry Pi and transferred to ThingSpeak, an IoT (internet of things) cloud. Communication between the 2 modules is provided via ThingSpeak. As a result of the artificial neural network training, theoretical accuracy of approximately 95% has been achieved. When the test data were examined, this accuracy was also obtained in practice. An interface that can be easily understood by the user has been prepared, and a communication and risk estimation algorithm is embedded behind it. The interface software was able to offer suggestions to the user about the representative vehicle by providing visual and written advice.

Keywords: Artificial Neural Network, Big Data, Image Processing, Traffic Control

ÖNSÖZ

Bu tez çalışması Necmettin Erbakan Üniversitesi Mühendislik Fakültesi Mekatronik Mühendisliği Anabilim Dalı Öğretim Üyelerinden Sayın Prof. Dr. Mehmet KARALI yönetiminde gerçekleştirilmiş ve Necmettin Erbakan Üniversitesi Fen Bilimleri Enstitüsü'ne yüksek lisans tezi olarak sunulmuştur.

Yüksek öğrenimim boyunca teknik ve sosyal hayatıma ışık tutan, inanmak ile başarmak arasındaki bağı harekete geçmek olduğunu hissettiren pek kıymetli danışman hocam Prof. Dr. Mehmet KARALI'ya teşekkürlerimi sunarım.

Bu çalışmanın ve mesleki hayatımda gerçekleştirdiğim tüm projelerin emekleriyle tabanını oluşturan bölüm hocalarım sayın, Doç. Dr. İlhan İLHAN'a, Doç. Dr. İbrahim YILDIZ'a, Dr. Öğr. Üyesi Ümit ÖNEN'e, Dr. Öğr. Üyesi Ahmet MERAM'a, Arş. Gör. Dr. Emrehan YAVŞAN'a teşekkür ederim.

Tasarlama ve üretme odaklı yaşam ilkelerini benliğime yerleştiren, ismini gururla taşıdığım, Konya eşraflarından dedem Merhum Niyazi USTA'ya (ILDIRAR) minnettarlığımı sunarım.

Yaşamım boyunca maddi ve manevi desteklerini benden eksik etmeyen değerli annem Hatice GÖKDOĞAN başta olmak üzere, aileme şükranlarımı sunarım.

Çalışmalarım doğrultusunda şahsıma psikolojik destek veren sayın Uzm. Psk. Dan. Esra KAĞNICI'ya teşekkür ederim.

Niyazi GÖKDOĞAN
KONYA-2021

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÖNSÖZ	vi
İÇİNDEKİLER	vii
ŞEKİL LİSTESİ.....	ix
SİMGELER VE KISALTMALAR.....	xi
1. GİRİŞ	1
2. KAYNAK ARAŞTIRMASI	3
2.1. Trafik Denetleme Sistemleri ve Kullanılan Yöntemler	3
2.1.1. Polis trafik uygulamalarının uygulanma yöntemleri	5
2.1.2. Trafik denetleme sistemleri	7
2.1.3. Trafik denetleme sistemlerinde kullanılan yöntemler	8
2.2. Görüntü İşleme ile Plaka Tanımı	10
2.2.1. Raspberry Pi.....	10
2.3. Yapay Sinir Ağları	11
2.3.1. Yapay sinir ağında kullanılan aktivasyon fonksiyonları.....	14
2.4. Büyük Veri (Big Data).....	16
3. MATERYAL VE YÖNTEM.....	20
3.1. Konumların Oluşturulması	22
3.1.1. Kamera konumlarının belirlenmesi ve konumlar arası mesafenin hesaplanması.....	22
3.1.2. Konya haritasına kameraların yerleştirilmesi ve görselleştirilmesi.....	23
3.2. Big Data'nın Oluşturulması	25
3.2.1. Risksiz araç hareketlerinin oluşturulması	25
3.2.2. Riskli araç hareketlerinin oluşturulması	31
3.3. Risksiz ve Riskli Verilerin Birleştirilmesi	35
3.4. Yapay Sinir Ağlarının Eğitilmesi ve Oluşturulan Modelin Test Edilmesi	36
3.5. Raspberry Pi 3 ile Karakter Tanıma ve Temsili Plaka Okuma.....	38
3.5.1. Gerekli kütüphane kurulumları.....	39
3.5.2. Python diliyle karakter tanıma ve temsili plaka okuma yazılımı.....	41
3.6. Plaka Bilgisinin Raspberry Pi 3 ve Bilgisayar Arasında Alışverişi.....	41
3.6.1. Plaka bilgisinin bilgisayara gönderilmesi	42
3.6.2. Gönderilen plaka bilgisinin bilgisayardan okunması.....	43
3.6.3. Veri gönderim yazılımı ve temsili plaka okuma yazılımının entegre hale getirilmesi	44

3.7. YSA Modeliyle Öneri Yazılımının Yapılması ve ThingSpeak İletişim Entegrasyonu	44
3.8. Bilgisayar Kullanıcı Arayüzü Hazırlanması	45
4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA.....	46
4.1. Risksiz Araç Hareket Sonuçları	46
4.1.1. Sağlık çalışanlarına ait sonuçlar	46
4.1.2. Okul çalışanlarına ait sonuçlar.....	46
4.1.3. Kolluk kuvveti çalışanlarına ait sonuçlar.....	47
4.1.4. Diğer kamu çalışanlarına ait sonuçlar.....	47
4.1.5. Öğrencilere ait sonuçlar	48
4.1.6. Esnaf ve işçilere ait sonuçlar	48
4.1.7. Pazarlama çalışanlarına ait sonuçlar	49
4.1.8. Emeklilere ait sonuçlar	49
4.2. Riskli Araç Hareket Sonuçları	50
4.2.1. Sağlık çalışanlarına ait sonuçlar	50
4.2.2. Okul çalışanlarına ait sonuçlar.....	50
4.2.3. Kolluk kuvveti çalışanlarına ait sonuçlar.....	51
4.2.4. Diğer kamu çalışanlarına ait sonuçlar.....	51
4.2.5. Öğrencilere ait sonuçlar	52
4.2.6. Esnaf ve işçilere ait sonuçlar	52
4.2.7. Pazarlama çalışanlarına ait sonuçlar	53
4.2.8. Emeklilere ait sonuçlar	53
4.3. Yapay Sinir Ağının Eğitim ve Test Sonuçları	54
4.4. Temsili Plaka Okuma Yazılımı Sonucu.....	55
4.5. Raspberry Pi ve Bilgisayar Arası İletişim Sonuçları	56
4.5.1. Raspberry Pi'nin veri göndermesi ve bilgisayarın gönderilen veriyi okuması	56
4.5.2. Veri gönderim ve temsili plaka okuma yazılımlarının entegre hale getirilmesi sonucu	57
4.6. YSA Modeliyle Öneri Yazılımının Yapılması ve ThingSpeak İletişim Entegrasyonu Sonucu	58
4.6.1. Öneri yazılımı sonucu.....	58
4.6.2. Öneri yazılımının ThingSpeak ile entegrasyonu sonucu	59
4.7. Bilgisayar Kullanıcı Arayüzü Sonucu	59
5. SONUÇLAR VE ÖNERİLER.....	64
5.1 Sonuçlar	64
5.2 Öneriler	65
6. KAYNAKLAR	66
EKLER	71

ŞEKİL LİSTESİ

Şekil 2.1. ReLU Aktivasyon Fonksiyonu	15
Şekil 2.2. Sigmoid Aktivasyon Fonksiyon Grafiği	16
Şekil 3.1. Araştırma Yöntem Diyagramı	21
Şekil 3.2. Konum Bilgileri	22
Şekil 3.3. 124x6 Boyutlarında Oluşturulan DataFrame	23
Şekil 3.4. Koordinatlara Nokta Atanması	24
Şekil 3.5. Koordinatlara Atanmış Kameraların Yerleştirilmesi	24
Şekil 3.6. Kameraların İçerdiği Koordinatlara İlişkin Açıklamaların Gösterilmesi	25
Şekil 3.7. Sağlık Çalışanlarına Ait Seçilen Hastane Listesi	26
Şekil 3.8. Okul Çalışanlarına Ait Seçilen Okul Listesi	27
Şekil 3.9. Kolluk Kuvveti Çalışanlarına Ait Seçilen Polis Merkezi Listesi	28
Şekil 3.10. Diğer Kamu Çalışanlarına Ait Seçilen Kamu Binası Listesi	28
Şekil 3.11. Öğrencilere Ait Seçilen Yurt Binası Listesi	29
Şekil 3.12. Esnaf ve İşçilere Ait Seçilen Sanayi Listesi	30
Şekil 3.13. Pazarlama Çalışanlarına Ait Seçilen Konum Listesi	31
Şekil 3.14. Riskli Araç Kriterlerinde Kullanılan Yer DataFrame	32
Şekil 3.15. Riskli Araç Kriterlerinde Kullanılan Yer DataFrame	33
Şekil 3.16. Oluşturulan Veri Kümesi	35
Şekil 3.17. Yapay Sinir Ağı Modeli	36
Şekil 3.18. Raspberry Pi 3 Model B	37
Şekil 3.19. Raspberry Pi Kamera Modülü 2	38
Şekil 3.20. Temsili Plaka	40
Şekil 3.21. ThingSpeak Çalışma Prensipleri	40
Şekil 3.22. API Keys Sekmesi ve Kişisel API Key	41
Şekil 3.23. Veri Okuma API Key'i	42
Şekil 4.1. Sağlık Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü	45
Şekil 4.2. Okul Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü	45
Şekil 4.3. Kolluk Kuvveti Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü	46
Şekil 4.4. Diğer Kamu Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü	46
Şekil 4.5. Öğrencilere Ait Üretilen Risksiz Araç Sürüşü	47
Şekil 4.6. Esnaf ve İşçilere Ait Üretilen Risksiz Araç Sürüşü	47
Şekil 4.7. Pazarlama Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü	48
Şekil 4.8. Emeklilere Ait Üretilen Risksiz Araç Sürüşü	48
Şekil 4.9. Sağlık Çalışanlarına Ait Riskli Araç Sürüşü	49
Şekil 4.10. Okul Çalışanlarına Ait Riskli Araç Sürüşü	49
Şekil 4.11. Kolluk Kuvveti Çalışanlarına Ait Riskli Araç Sürüşü	50
Şekil 4.12. Diğer Kamu Çalışanlarına Ait Riskli Araç Sürüşü	50
Şekil 4.13. Öğrencilere Ait Riskli Araç Sürüşü	51
Şekil 4.14. Esnaf ve İşçilere Ait Riskli Araç Sürüşü	51
Şekil 4.15. Pazarlama Çalışanlarına Ait Riskli Araç Sürüşü	52
Şekil 4.16. Emeklilere Ait Riskli Araç Sürüşü	52
Şekil 4.17. İterasyon Sonuç Bilgileri	53
Şekil 4.18. Model Accuracy ve Model Loss'un İterasyona Bağlı Değişimi	54
Şekil 4.19. Test Sonuçları	54
Şekil 4.20. Temsili Plaka Okuma Yazılımı Sonucu	55
Şekil 4.21. ThinkSpeak'e Veri Gönderme Sonucu	56
Şekil 4.22. ThinkSpeak'ten Veri Okuma Sonucu	56
Şekil 4.23. Plaka Tanıma ve Plaka Gönderimi Sonucu	57

Şekil 4.24. Öneri Sonuçları	58
Şekil 4.25. Entegrasyon Sonucu	58
Şekil 4.26. Arayüz	59
Şekil 4.27. “ <i>Serbest</i> ” Tavsiyesi	60
Şekil 4.28. “ <i>Yogunluk Bazlı İşlem</i> ” Tavsiyesi	61
Şekil 4.29. “ <i>Durdur</i> ” Tavsiyesi	62



SİMGELER VE KISALTMALAR

Simgeler

- i : x de veri belirten değişkenin alt indisi
 j : Toplam sembolünde x deki sıra belirten değişkeni
 $f(x_i)$: Softmax fonksiyonu
 $F(x)$: ReLU fonksiyonu
 $F'(x)$: ReLU fonksiyon türevi
 x : X eksenindeki veri değişkeni
 x_i : x deki i . veri
 x_j : Toplam sembolündeki her bir j . veri

Kısaltmalar

- AF : Activation Function (Aktivasyon Fonksiyonu)
ANN : Artificial Neural Network (Yapay Sinir Ağı)
API : Application Programming Interface (Uygulama Programlama Arayüzü)
CNN : Convolutional Neural Network (Evrişimli Sinir Ağı)
CPU : Central Processing Unit (Merkezi İşlem Birimi)
CSI : Camera Serial Interface (Kamera Seri Veriyolu)
DSI : Display Serial Interface (Ekran Seri Arabirimi)
EDS : Elektronik Denetleme Sistemi
FES : Fuzzy Expert System (Bulanık Uzman Sistem)
FPS : Frame Per Second (Saniyedeki Kare Sayısı)
GPIO : General Purpose Input/Output (Genel Amaçlı Giriş Çıkış Portları)
IDE : Integrated Development Environment (Tümleşik Geliştirme Ortamı)
IoT : Internet of Things (Nesnelerin İnterneti)
ITS : Intelligent Transportation System (Akıllı Ulaşım Sistemi)
KGYS : Kent Güvenlik Yönetim Sistemi
LAN : Local Area Network (Yerel Ağ Bağlantısı)
MOBESE : Mobil Elektronik Sistem Entegrasyonu
OCR : Optical Character Recognition (Optik Karakter Tanıma)
OSA : Olasılıksal Sinir Ağı
PCA : Principal Component Analysis (Temel Bileşen Analizi)

PTS : Plaka Tanıma Sistemi

ReLU : Rectified Linear Unit (Doğrultulmuş Doğrusal Birim)

RGB : Red Green Blue (Kırmızı Yeşil Mavi)

SD : Secure Digital (Güvenli Sayısal)

TEDES : Trafik Elektronik Denetleme Sistemi

USB : Universal Serial Bus (Evrensel Seri Veriyolu)

YSA : Yapay Sinir Ağı



1. GİRİŞ

Trafikte yapılan denetimlerle araçlar durdurulmakta, başta, önceden belirlenen şahıslar olmak üzere birtakım suçlular ve potansiyel şüpheliler aranmaktadır.

Önleme aramasına bağlı çevirmeler, kabul edilebilir makul bir sebepten hareketle, kamu düzeni ile milli güvenliğin, genel ahlak ve sağlığın ya da başka insanların hak ve özgürlüklerinin korunması; bulundurulması ya da taşınması yasaklanmış her türlü silah, patlayıcı eşya ve maddelerin belirlenmesi; suç işlenmesine engel olunması, toplum için tehlikeli olabilecek kişilere ulaşmak üzere yapılabilmektedir [1]. Adli arama ve çevirmelerde, genel asayiş ile emniyetin korunması ya da tehlikeye karşı önlem alınması amacıyla, tehlike arz eden bir kişinin, eşyanın ele geçirilip polis gözetiminde tutulması amaçlanmaktadır [2]. Terör zanlısı aramaları, kaçakçılık aramaları, uyuşturucu madde ve zanlısı denetlemeleri, firari mahkum aramaları, kayıp insan araması, asker kaçağı denetimi gibi kontroller adli denetim altında yapılmaktadır.

Araç veya şahsa ait evrak kontrolü, alkol uygulamaları, kış lastiği kontrolü, göçmen denetimi, trafik kurallarına aykırı araç kullanım kontrolü gibi trafik denetlemeleri idari kontrol üzerine gerçekleştirilmektedir.

Denetim ekipleri, seyir halinde, sabit, ihbarlı ve seçici kontrol yöntemleriyle trafikte uygulama yapmaktadır [3]. TEDES, EDS, mobil radar, koridor hızı tespit sistemi gibi trafik kuralları odaklı teknik yapılar, aracın anlık trafik ihlallerini tespit ederek gerekli cezai işlemin önünü açmaktadır [4;5;6;7].

Trafik denetimlerinin birçoğunda PTS (Plaka Tanıma Sistemi) kullanılmaktadır. PTS sistemi araca ait tüm bilgiyi içeren plakayı tanıdığından, aracın konumunu öğrenmeyi, davranışlarını incelemeyi, belirli noktalardan geçtiğini tanımlamayı sağlayabilmektedir. PTS üzerine birçok akademik ve pratik çalışma mevcuttur. Genel yaklaşım, alınan görüntüde plaka yerini bulmak ardından karakter tanıma işlemleri gerçekleştirmektir [8]. Tanınan plaka verisi, denetim amacına uygun kullanılarak, gerekli işlemleri yürütmede taban bilgisini oluşturmaktadır.

Yapılan çevirmelerde, durdurulacak araçların seçimi tecrübeye, anlık kararlara veya tesadüfi kurallara dayanmaktadır. Araç geçiş yoğunluğunun fazla olmasıyla suçlu ve şüpheli araç durdurma isabet oranının düşebileceği ihtimal dahilindedir. Oysaki gelişen teknoloji ile potansiyel suçluların daha isabetli bir oranla durdurulabileceği öngörülmüştür.

Trafikte, sürüş sağlığı analizi, araç yönlendirme ve çizelgeleme işlemleri, sinyalizasyon işlemleri, araç akış kontrol uygulamaları gerçek zamanlı olarak yapılmaktadır. Bu işlemleri destekleyici Expert System (Uzman Sistem), Fuzzy Logic (Bulanık Mantık), Genetik Algoritma ile optimizasyon, Yapay Sinir Ağları'yla çıkarım yöntemleri ve sistemleri kullanılmaktadır [9;10;11]. Bu uygulamaların belirli araca ait geçmiş verileri kullanmadan, araçların veya trafik durumunun yalnız anlık durumu değerlendirilerek karar mekanizmasını çalıştırdığı anlaşılmaktadır.

Trafikteki zaman değişkenliği, bulanıklık, doğrusal olmama, belirsizlik nedeniyle geleneksel trafik modelleme ve kontrol yöntemlerini kullanarak trafik sistemleri verimliliğinin istenilen düzeyde iyileştirilemeyeceği, iyi sonuçlar alınamayacağı anlaşılmaktadır [12].

Belirli yol lokasyonlarında, geçmiş araç hareketlerinin kaydedilmesiyle oluşturulan Big Data kullanılarak, Yapay Sinir Ağlarıyla ileri bir zamandaki trafik yoğunluğunu ve bu yoğunluğa bağlı sinyalizasyon değişkenlerini tahmin edici, araç hareketlerine göre kaza oluşma potansiyelini tahmin eden araştırmalar mevcuttur [13].

YSA (Yapay Sinir Ağı), sayısal veya sözel sinyal, bilgi işleme, örüntü tanıma başta olmak üzere birçok doğrusal olmayan verileri, YSA'nın çok boyutlu ölçeklemeye uygun olması, öz uyarlamalı yapıya sahip olması, düzensiz verilerden genel bir düzen oluşturabilmesi ve kendi kendine öğrenme özellikleriyle işleyebilmektedir. Bu sebeple yapay zeka uygulamalarının trafik sistemlerine entegreli hale getirilmesi, trafik alanındaki problemlerin çözümünde verimli olabileceğini göstermektedir [9;14].

Bu tez çalışmasıyla doğrusal olmayan verileri kullanarak ve YSA'yı uygun parametrelerle eğiterek, mesleki tecrübesi olmayan ya da yetersiz olan güvenlik güçlerine, durdurulacak aracın geçmişi ve sürücü hakkında bilgi verecek sistemle bir karar destek mekanizması oluşturmasına zemin hazırlanması amaçlanmıştır.

Daha efektif sonuçlar almak üzere, yapay sinir ağları ve diğer yapay zeka algoritmaları için gerekli gerçek verileri, Big Data kurmak üzere temin etme, saklama, kullanma gibi yetkiler gerekli yasal düzenlemeyle araştırmacılara verilmesi tavsiye edilmektedir.

2. KAYNAK ARAŞTIRMASI

2.1. Trafik Denetleme Sistemleri ve Kullanılan Yöntemler

Kanunda denetim kavramı; araçları, geçerli kanun çerçevesinde araçlarda bulunması gereken belgeleri ve gereçleri, sürücüleri ve bunların belgelerini, sürücüler ile karayolunu kullanan diğer bireylerin kurallara uygun davranıp davranmadığını, trafik düzenlemeleri ve çeşitli tesislerin söz konusu kanun hükümleri çerçevesinde hareket edip etmediğini denetlemek, biçiminde tanımlanmaktadır [15].

Trafiğin insan, araç ve yol olmak üzere üç temel bileşeni bulunmaktadır. Yollar ve araçların da yapımında insanın inisiyatifinin söz konusu olduğu dikkate alınırsa “insan faktörünün” trafikte etken ve asıl bileşen olduğunu ifade etmek doğru bir ifade olacaktır. Bu yönü itibarıyla insanın trafiğin bileşenleri içindeki en önemli yere sahip olduğu ortadadır. Bu alanda yapılan istatistikler, en büyük oranın insan kusurunda olduğu anlaşılmaktadır. Bütün bunlara rağmen trafik kazalarında genel olarak, yalnızca bir tek faktörden söz etmek yanlış olacaktır. Trafik bileşenleri olan insan, araç ve yol unsurlarının hepsi ya da birkaçı kazanın etkenleri arasında yer alabilmektedir. 2019 yılında Türkiye’de meydana gelen ölümlü yaralanmalı trafik kazalarına yol açan toplam 204 bin 538 kusura bakıldığında zaman kusurların % 88,0’lık kısmının sürücü, %8,2’lik kısmının yaya, %2,0’lik kısmının taşıt, %1,3’lük kısmının yolcu ve %0,5’lik kısmının yol kaynaklı olduğu görülmüştür [16].

Trafik işleyişinde yolu kullananlarının kusur, ihlal ve ihmallerine polis denetimleri yapılmak suretiyle engel olunmaya çalışılmaktadır. Polis denetimleri, orta vadeli kampanya ve uzun vadeli eğitim programlarına göre insan davranışları üzerinde kısa sürede etkili olan, kısa sürede sonuç alınan bir yöntemdir. Trafik ihlallerinin acil müdahaleyi gerekli kılan durumlar olmasından dolayı, hedeflerinin gerçekleşmesini sadece uzun vadeli kampanyalar ve eğitimlerden beklemek pek de akılcı bir yaklaşım değildir. Bundan dolayı esas olan polis denetimlerini kesintisiz olarak sürdürmektir.

Denetimi kademeli olarak ilerleyen bir süreç olarak üç adımda düşünmek mümkündür [17].

Yasa yapmak, ilk adımdır. Birçok ülkede yol kullanıcıları tarafından uyulması gereken konuları tespit eden bir dizi yasa ve yönetmelik vardır. Ülkelerin bazılarında bu, karayolu kanunu gibi oldukça genel bir kurallar dizisi özelliği gösterebildiği gibi; kimi ülkelerde trafikle ilgili yasalar belli durumlarda uyulması gerekli belirli davranışlar üzerinde durulabilmektedir. Trafikle ilgili yasaların hangi ölçüde belirli olduğu,

otomasyon ihtimalleri bakımından büyük sonuçlar doğurmaktadır. Bundan dolayı uygun kanunlar belirlenmesi çok önemli bir adımdır [17].

Yolda uyulacak genel davranışların yasalar çerçevesinde olmasını sağlamak için polislerin kontrol faaliyetleri ikinci adımı oluşturmaktadır. Farklı durumlarla karşılaşıldığı zaman polis, ihlallerin derecesini dikkate alarak alternatif bazı işlemler uygulayabilmektedir. Polis, sürücülerini durdurarak kayıt altına almadan uyarıda bulunabilir, durdurarak para cezası uygulayabilir, bir saat süreyle aracı trafikten menden, araca el koyma veya sürücüyü tutuklamaya kadar farklılıklar gösterebilen yaptırımlara gidebilir. Yasada tespit edilen hükümler çerçevesinde, sürücüyü durdurma yoluna gitmeden olayı ve aracın plakasını kaydedebilir [17].

Üçüncü adımda, polis tarafından kaydedilen suçun sonucunda yapılan yasal işlemler yer almaktadır. Bu şamada birçok farklı işlem yapılabilmektedir. Hız ve park suçları, kimi ülkelerde idari ölçekte sabit ceza sistemleri kullanılarak cezalandırılmaktadır. Kiminde ise ceza puanı sistemi kullanılmaktadır. Daha ciddi suçlarla ilgili kararlar, mahkemelerde verilmektedir; davadan önce savcılar, çoğunlukla olayı mahkemeye taşımadan anlaşma yaparak davacı bulunur. Suçla ilgili dava açılması halinde ise davalı ve davacıya açık olan, çok çeşitli müracaat işlemleri bulunmaktadır. Bunun gibi davalar savcılığın da polis memurlarının da zamanını çok aldığından dolayı, savcının özellikle kesin kanıt bulmakta zorlanması gibi hallerde dava açılmayabilir. Netice itibariyle bunun gibi durumlar, polis memurlarının kararlarını da sadece bir uyarı cezası kesme istikametinde etkilemektedir [17].

“Yönergesi (2011) esasları çerçevesinde Türkiye’de, trafik denetimleri aşağıda verilen dört farklı biçimde yapılmaktadır [3].

Seyir halinde kontrol: Ekip araçlarının seyir halinde olmasından dolayı uzun bir karayolunda seyahat eden sürücülerin üzerinde “Algılanan Yakalanma Riski Duygusu” geliştirmek suretiyle kurallara uyma oranlarını artırmak, seyir halinde kontrolün temel amacıdır. Bu kontrol yöntemi, sürücü ihlalleri oranlarının belli noktalarda olmaktan ziyade, uzun yol kesimlerinde yoğunluk gösterdiği durumlarda kullanılmaktadır. Sorumluluk güzergahı üzerinde görevlerini seyir halinde resmi ya da sivil bir ekip aracıyla yapan görevliler, sürücülerin araçlarını kurallara uygun şekilde kullanmaları üzerinde olumlu bir etki yapmaktadır. Seyir halindeyken kontrol etme yöntemi, akan trafikte sürücülerin kurallara riayet etme oranlarını yükselterek kural dışı ve tehlikeli araç kullanılmasını önlemek amaçlanmaktadır [3].

Sabit kontrol: Sabit kontroller; bir ihbara bağılı olmadan sondajlama usulüyle araçların seçilip kontrol ceplerine alınması ve bu noktada; sürücü ve araca ait belgelerle araçta olması gerekli zorunlu gereçler, “Araçların İmal Tadil ve Montajı Hakkındaki Yönetmelik” çerçevesinde tespit edilmiş teknik koşullar ile trafik ve tescil belgelerine göre aracın uygunluk durumunun denetlenmesi amacıyla yapılmaktadır. Sabit kontrollerin kazaların önlenmesi noktasında ve sürücüleri davranışları üzerinde doğrudan bir etki yapmadığı bilinmektedir. Bu modelin uygulanmasında temel amaç, araçların kaza yapmalarında etken olabilecek teknik ve riskli koşulları taşımayan durumlarına engel olunması ve Zorunlu Mali Mesuliyet Sigortaları gibi, trafikte yer alacak araçlarda hukuken bulunması zorunlu belgelerin denetlemesidir [3].

İhbarlı kontrol: Bu denetleme yönteminde; sivil ya da resmi bir trafik ekibi ya da personeli, denetim yapılacak hedef bölgede görevlendirilmektedir. Sözü edilen ekip ya herhangi da personel, bir trafik kuralı ya da bir özellikle belirlenmiş trafik kuralı ihlali yapan sürücüleri belirleyerek, ileri bir noktada tedbir almış çevirme ekibine bildirmektedir. Hedef aracı alan çevirme ekibi, araca ve sürücüsüne yasal işlem yapar. Seyir halindeki sürücülerin işledikleri kural ihlallerinin tespit edilerek cezalandırılması esasına dayanan İhbarlı kontrol yöntemi, sık sık uygulanan bir yöntemdir. İhbarlı kontrol, hem maliyetinin düşük olması, hem de sürücüler üzerindeki etkisinden dolayı etkili ve sık kullanılan yöntemlerden biridir [3].

Seçici kontrol: Genellikle yoğun trafik ve fazla ihlalin olduğu ve trafik denetim personelinin, araç ve gerecin yetersiz olduğu hallerde, denetimin sürücü davranışlarındaki etkisini artırmak için seçici trafik kontrolleri uygulanmaktadır. Bu uygulamada hangi tipte ihlalin hangi koşullarda ve ne zaman yapıldığı; sayım ya da gözlem yöntemiyle belirlenmektedir. Bu bağlamda kazalara yol açan ihlallere odağa koyan ve söz konusu ihlallere engel olmaya yönelik; ihlalin saatini, süresini ve niteliğini göz önünde bulundurarak, bir denetleme programını hazırlanmakta ve uygulanmaktadır [3].

2.1.1. Polis trafik uygulamalarının uygulanma yöntemleri

Kentlerde hızlı yapılaşma, nüfusun hızlı artışıyla beraber, kentlerin genişlemesi ile zorunlu hale gelen ulaşım araçları kullanma zorunluluğu ve buna bağılı olarak artan trafik sorunları kentlerin çözülmesi gereken en önemli sorunlarından biri haline gelmiştir [18]. Sözü edilen trafik sorunu birey ve devletler için maddi, manevi birçok kayba yol açmaktadır. Bu nedenle trafik sorununu çözmek suretiyle karayollarında trafik güvenliğini sağlamak, devletleri yönetenler için aşılacak önemli toplumsal sorunlardan

biri haline gelmiştir. Karayolları trafik güvenliği; insan, çevre ve taşıt bileşenleri arasında oluşabilecek olası problemleri ve bu sorunlara karşı geliştirilmiş yöntemleri içine alan birçok yönü olan bir konudur [19].

Dünya geneli için önemli bir konu haline gelen trafik güvenliği, Birleşmiş Milletler 'de de ele alınmış ve 2010'da yapılan toplantı sonrasında 64/255 sayılı "Küresel Yol Güvenliğinin İyileştirilmesi" başlıklı karar alınmıştır. Türkiye, Birleşmiş Milletler tarafından alınan kararlara uygun olarak, karayolu trafik güvenliği strateji ve eylem planı çerçevesinde özet olarak aşağıda verilen ifadelerle yer verilmiştir [20].

- Trafik güvenliğiyle ilgili farkındalık oluşturmak üzere kampanya, proje, seminer, konferans, sergiler düzenlenecek; kitapçık, afiş ve broşürler hazırlanarak halka ulaştırılacaktır.

- Trafik güvenliğiyle ilgili bütün kurum ve kuruluşlar kendi görev alanlarında gereken tüm çalışmaları gerçekleştirecektir.

- Yerel ve ulusal basın ve yayın kuruluşları tarafından trafik güvenliğine katkısı olacak içeriklere sahip yayınların yapılacaktır.

- Trafik güvenliğiyle ilgili tehlike ve riskleri minimuma indirmek için özellikle altyapı çalışmaları yapılacaktır.

- Trafik yönetimi ve denetimlerinde; modern teknolojiden yararlanılacaktır.

- Başta hız, emniyet kemeri, alkol, cep telefonu ve kask kullanımı ve kırmızı ışık ihlalleri gibi denetimler olmak üzere bütün alanlarda etkin denetimler arttırılacaktır [20].

Ciddi bir denetim yapılmadan yalnızca bilgilendirme kampanyaları ve yol kullanıcıların trafik eğitimiyle bir sonuca ulaşmak mümkün değildir. Denetimlerin etkin ve başarılı olabilmesi için aşağıdaki unsurlara dikkat edilmesi gerekmektedir:

- Denetimin hangi amaçla yapıldığı iyi tespit edilmelidir. Denetimlere toplumun destek vermesi sağlanmalıdır.

- Denetim adil ve hukuk çerçevesinde olmalıdır. Teknolojik araçlardan faydalanılmalıdır.

- Yeterli sayıda personel görevlendirilmelidir. İhlalin yapılması ile cezalandırma arasındaki süre uzun olmamalıdır.

- Denetimde görevlendirilecek personel özenle seçilmeli özellikle tecrübeli personel tercih edilmelidir. Denetim uygulamaları da denetlenmelidir [20].

2.1.2. Trafik denetleme sistemleri

Her geçen gün trafiğe çıkan araç sayısının artmasına bağlı olarak hasarlı, yaralanmalı ve ölümlü trafik kazalarda da artışlar olmaktadır. Trafik kazalarına engel olmak ve kazaları en aza indirebilmek için ilgili birimler tarafından farklı birçok denetleme ve cezalandırma yöntemlerinden yararlanılmaktadır. Akıllı Trafik Sistemleri, son yıllarda yaygınlaşan teknolojik yöntemlerden biridir.

TEDES (Trafik Elektronik Denetleme Sistemi), karayollarında can ve mal güvenliğinin sağlanması ve düzenli bir şekilde kontrol altına alınmasını amaçlamaktadır. TEDES ile güvenli ve düzenli bir trafik akışı sağlanmak amaçlanmaktadır. Hız noktalarına yerleştirilmiş kameralarla araçların fotoğraflarını anlık olarak çekerek onların plakalarını, giriş ve çıkış zamanlarını kayıt altına alan TEDES, kaydettiği araçlara ait hızları, anlık olarak tespit etmektedir. Aynı anda araçların sorgulamasını yapmak suretiyle, araçların cinslerine göre hız limitlerinin aşılması halinde cezai işlem uygulanmaktadır. İhlali yapan aracın bilgileri ve ihlal anındaki görüntü, ceza makbuzuyla beraber araç sahibine postayla gönderilmektedir. Ayrıca sistemle birlikte hacizli-yakalamalı, kayıp-çalıntı ve adli olaylara karışan araçlar da belirlenmektedir [4].

EDS, Trafik akışının kontrolü, ulaşımın denetiminin yapılması ve trafikte düzeni bozan araçların neden olduğu kazaların önlenmek suretiyle can ve mal güvenliğinin sağlanması üzere geliştirilmiş ve kuralları ihlal eden araçların sensörler vasıtasıyla belirlenmesinden sonra görüntü işleme yazılımlarıyla plaka numaralarının kaydedilerek ilgili kanunda öngörülmüş cezanın tabi tutulması için tasarlanmış Elektronik Denetleme Sistemidir. EDS, trafik akışının kontrol edilmesi ve ulaşım optimize edilmesi, kent yaşantısının belli normatif ve kurallara uygun bir yapıya kavuşturulması için geliştirilmiş bir denetleme sistemidir [5]. 2918 Sayılı Karayolları Trafik Kanunu'nun ek 16. Maddesi EDS'nin kanuni dayanağıdır.

Mobil radarlar yeterli çözünürlükte PC, kameralı tablet gibi taşınabilir donanım ve yazılımlarla trafik denetimleri farklı bir boyutlar kazanmıştır. Kamera menzili içerisine giren araçların plakalarını otomatik olarak tanıma, sorgulama, gerekli bilgileri ve uyarıları anlık olarak yansıtabilen yazılımlarla yapılmış denetimin ayrıcalıklarıdır. Trafik uygulama noktalarında görüntü işleme teknolojisiyle donanıma eklenecek özel algoritmalarla tek kamerayla hız tespiti mümkün hale gelmiştir [6].

Koridor Hızı İhlal Tespit Sistemi, karayollarında araçların ortalama hızlarını tespit edilen iki nokta arasında hesaplamak suretiyle kural ihlalinde bulunan araçları saptamaktadır. Koridor Hızı İhlal Tespit Sistemleri, hız denetiminin yapılacağı yol

güzergâhının girişine ve çıkışına yerleştirilmiş plaka tanıma üniteleriyle bu ünitelerle ulaşılan plaka bilgilerinden yararlanarak ortalama hızları hesaplayan merkezi bir yazılımdan meydana gelmektedir. Tespiti yapılan koridorun arasında kalan noktalarda plakası tespit edilen araçların, söz konusu mesafedeki seyahat süreleri ölçülmekte ve bu değerler mesafe bilgileri de yararlanılarak ortalama hıza dönüştürülmektedir. Sistem, çok veya tek şeritli yollarda, yan yana ya da peş peşe veya giden araçları yüksek çözünürlüğe sahip kameralar sayesinde belirlememektedir. Bununla birlikte hız ihlalinin yapıldığı yer, zaman, yön, mesafe, tarih, hız sınırı, taşıt hızı, hız ihlal olayı gibi bilgileri, merkezi yazılımdan yararlanarak ceza makbuzuna aktarabilmektedir. Bu sistem, noktasal hız tespit radarlarından farklı olarak, sisteminde radarlardan yararlanmamakta ve böylece kalibrasyona ihtiyaç duymamaktadır. Bunun yanında, ortalama hız hesabı ile ilgili denetimler, araçların tüm güzergâhta hız kısıtlamalarına uymasını sağlamak ve aşırı hızın neden olduğu kazaları önlemektedir [7].

2.1.3. Trafik denetleme sistemlerinde kullanılan yöntemler

Şehir içi ve şehirlerarası ulaşımında yaşanan değişimler, insanların hareket tarzını etkiler, çevreyi değiştirir, kentlerin hızla büyümesine neden olur. Son yıllardaki teknolojiye yaşanan gelişmeler, herhangi bir kentin karşılaşılabileceği trafik kazaları, trafik sıkışıklıkları ve kara ulaşımının yol açtığı çevresel etkiler gibi birçok sorunu çözmek amacıyla uygun güçlü çözüm yolları bulunmasını sağlamıştır. Karayollarındaki araçların birbirleriyle en fazla kesişme olasılıklarının olduğu alanlar olan kavşaklarda sözü edilen karmaşıklıklara engel olmak amacıyla trafik denetiminde çeşitli tekniklerden yararlanılmaktadır. Türkiye’de son yıllarda trafik denetimlerinde çeşitli teknik ve yazılımlardan yararlanılmaktadır.

Yapay zeka teknik ve bilgisayar teknolojilerinin gelişmesine paralel olarak akış kontrol, çizelgeleme, yönlendirme gibi sorunların çözülmesine yönelik birçok akademik çalışma ve uygulama bulunmaktadır. Bu bağlamda, trafikte yaşanan sorunların çözülmesi amacıyla yapay zeka teknikleri kullanımında önemli gelişmeler yaşanmaktadır [21]. Sözü edilen kapsamda öne çıkan sistemler; Expert System (Uzman Sistem) ya da Fuzzy Logic (Bulanık Mantık) yöntemleriyle şehir içi ve şehirlerarası ulaşımında trafik akımının kontrol edilmesi, Artificial Neural Networks (Yapay Sinir Ağları) ile trafik akışının tahmin edilmesi, Genetic Algorithms (Genetik Algoritmalar) ile trafik optimizasyonu sistemler sayılabilmektedir [22].

Başta ABD olmak üzere birçok ülkede trafiğin kontrolünde, yapay zeka teknikleri ve Akıllı Ulaşım Sistemleri önemli ölçüde kullanılmaktadır. Trafiğin kontrol edilmesi, trafik akımlarının kontrol edilmesi olarak kabul edilirse, kapasite kullanım düzeylerinin en iyi noktaya getirilmesi, ana arterler, arter kavşakları ve şehir içi yolların kontrol edilmesiyle şehrin yollarını otoyollara ulaştıran katılım ve ayrılım noktalarında yapılan denetimle sağlanır. Bu noktada trafik sıkışıklığını en düşük düzeye indirmek, trafik sıkışıklığını engellemek, ulaşım hız kazandırmak gibi ölçütler lineer olmayan yapıda yer alan kompleks optimizasyon sorunlarının çözülmesini gerekli hale getirmektedir. Mevcut optimizasyon teknikleri vasıtasıyla çözilemeyen bu tipteki sorunlar için yapay sinir ağları, bulanık mantık, genetik algoritmalar, uzman sistemler gibi yapay zeka teknikleri, problemin tiplerine göre tercih edilmektedir [23].

Sözü edilen tekniklerden yapay sinir ağları problemlerin çözümü noktasında biyolojik nöron modelleri esas alınarak oluşturulmuş yapay nöron modeli üzerine geliştirilmiş algoritmalarından yararlanmakta; bulanık mantık ise belirsizliklerin modellenmesi tabanlı çalışmaktadır. Uzman sistemler, bilgileri ve deneyimleri model olarak kabul eden bir yöntemdir. Bir uzman sistemin performansının başarısı, bilgi tabanının etkililik durumuyla doğru orantılıdır. Genetik algoritma ise bir tür stokastik tarama algoritmasıdır. Çözülmesi genellikle lineer olmayan trafik akımlarının kontrol edilmesi, yapay zeka tekniklerinin uygulandığı alanlarından biridir. Söz konusu teknikler özellikle trafik uyarımlı kontrollerin yapılabilmesi için yapısı itibarıyla oldukça uygun tekniklerdir [23].

Trafik Denetimlerinde Trafik Kazalarında Alınacak Önlemlere İlişkin Yönergenin birinci kısım ikinci bölümünde yer alan “Trafik Denetim Şekilleri” başlıklı bölümde Denetim personelinin tablet bilgisayar, takograf, radar, alkolmetre, gibi denetim araç ve gereçlerini kullanmayı bilmesinin gerektiği vurgulanmaktadır. Trafik kural ihlali yapan araçlar, şüphe oluşturmayacak biçimde belirlenecek ve mümkün olması halinde fotoğraf makinesi, video, kamera, MOBESE (Mobil Elektronik Sistem Entegrasyonu), KGYS (Kent Güvenlik Yönetim Sistemi) ve EDS (Elektronik Denetleme Sistemi) ve gibi teknik cihazlar kullanılmasının gerekliliği aynı yönergede vurgulanmaktadır.

2.2. Görüntü İşleme ile Plaka Tanımı

Her alanda teknoloji ve otomasyondan yararlanılan çağımızda, bilgisayarlı görüntü uygulamaları sayesinde geliştirilmiş olan plaka tanıma sistemleri trafik denetimi, güvenlik uygulamaları, araç takipleri, köprü ve otoyol geçişleri, giriş çıkış sistemleri, araçların bulunması gibi değişik birçok amaçla kullanılabilmektedir. Söz konusu teknoloji, trafik ve güvenlik donanımları alanı için de büyük bir önem taşımaktadır [24].

Bu konuyla ilgili çalışmalar yapanların; kenar ayırıştırma, simetri özelliği, Hough dönüşümü, renk özelliği, Gabor süzgeçleri, histogram analizi gibi çeşitli görüntü işleme tekniklerinden yararlanıldığı bilinmektedir [25].

Tasarlanan sistemde standartlara uygun olmayan plakalar, ilk olarak kameranın ilettiği fotoğrafa önışlem uygulaması yapılmakta, plaka çıkarımı amacıyla morfolojik işlemler uygulanmak suretiyle köşeler tespit edilmekte ve tespit edilen köşeler tarafından kaplanan minimum dörtgen bölge çıkarılmaktadır. Çıkarılan bölgeye yeni bir önışlem uygulaması yapıldıktan sonra bölge, karakter ayırıştırma sistemi ile düşey ve yatay taranmaktadır. Nihayet karakterlerin tanınması basamağında karakterler, yapay zekâ algoritması olan yapay sinir ağlarıyla kodlanmış şablon eşleştirme yönteminden yararlanılarak plaka tanıma işlemi tamamlanmaktadır [24].

Plaka tanıma sistemleri genel olarak araç plakasının bölgesinin tespit edilmesi, plaka bölgesi dikkate alınarak karakter ayırıştırma işlemi ve plaka karakterlerinin tanınması işlemleri olmak üzere üç önemli basamaktan meydana gelmektedir. Bilgisayar tabanlı araç plakaları tanıma sisteminin yazılım uygulamalarında ilk olarak plakanın ait olduğu bölge tespit edilmiş, bunun için kenar bulma algoritmasından yararlanılmıştır. Bunun ardından bulunan plaka bölgesinden hareketle karakter ayırıştırma işlemine geçilerek blob coloring algoritması işleme konulmaktadır. Nihayet karakterleri ayırıştırılmış plakadan karakter tanımaya geçilmek suretiyle şablon ayırıştırma algoritması kullanılarak elde edilen sonuçlar dikkate alınarak plaka bölgesinin tespit edilmesinde %95’lik doğruluk oranına, karakter ayırıştırma noktasında %92’lik doğruluk oranına ve tüm karakterlerin okunmasında %87’lik doğruluk oranına ulaşılmaktadır [26].

2.2.1. Raspberry Pi

Raspberry Pi bilgisayar monitörüne veya televizyona takılarak standart bir klavye ve fare ile kullanılan kredi kartı boyutunda bir bilgisayardır [27]. Basit bir mikro denetleyiciden fazlasını gerektiren çok sayıdaki uygulama için kullanılmaktadır [28]. Raspberry’nin pek çok avantajı bulunmaktadır. Bunlardan bazıları şöyle sıralanabilir;

kullanıcının kapsamlı programlama deneyimine sahip olmasını gerektirmemektedir, enerji tüketimi açısından verimlidir, küçük boyutu sayesinde geri dönüşümü kolaydır [27].

Abirami ve Jasmine (2018), gerçek zamanlı araç plaka tahmini üzerine çalışma yapılmıştır. Yapılan çalışmada trafik plakaları, sinir ağı kullanılarak plakadaki karakterleri hatasız tanınması için farklı yazı tiplerini içermektedir. USB kamera kullanılarak yakalanan görüntüyü tanımak için Raspberry Pi kullanılmıştır. Özellik çıkarımı PCA algoritması ve sınıflandırma CNN algoritması ile yapılmıştır. Segmentasyon için ayrı bölge ayrıldıktan sonra sınıflandırma adımı gerçekleştirilmiştir. CNN sınıflandırıcı, araç plakasında bulunan karakterleri tanımak için kullanılmıştır. OCR ve OpenCV yöntemleri kullanılarak araç plaka tanımı gerçekleştirilmiştir [28].

Agrawal ve Pardakhe (2017), yapılan çalışmada Raspberry Pi işlemcisi kullanarak araç plakasının görüntüleri hakkındaki bilgileri okumak için Optik Karakter Tanıma (OCR) kullanan tamamen otomatik bir plaka tanıma sistemi uygulanmıştır. Araç plakasının görüntüleri alınarak plaka alanı çıkarılmış, plakayı bölümlere ayırmak ve ayrılan her bir parçayı tanımlama için karakter segmentasyonu yapılmış ve plaka numarası tanıma işlemi gerçekleştirilmiştir. Raspberry Pi işlemcisi tarafından doğrulanmıştır [29].

Fakhar, Saad, Fauzan, Affendi ve Aidil (2019), polis devriye arabalarına takılarak çalınan arabaları gerçek zamanlı olarak tespit etmek ve otoyollarda aşırı hız yapan araçları tespit etmek gibi erişim kontrolü ve güvenliğin otomatikleştirilmesinde kullanılan sisteme alternatif olarak OpenCV kütüphaneli bir Raspberry Pi üzerinde çalışan taşınabilir bir sistem sunulmuştur. Raspbian Jessie OS kurulu olan bir Raspberry Pi 3 üzerinde, görüntüleri yakalamak için bir kameranın ek çevre birimleri üzerinde çalıştırılmıştır. Tanınan plakalar, Raspberry Pi içindeki bir bulut veri tabanına da kaydedilmiştir. Sistemin taşınabilir olması amaçlandığı için bu nedenle bir mobil güç bankasına bağlanarak çalıştırılmıştır [30].

2.3. Yapay Sinir Ağları

Yapay Sinir ağları, insanın beynin temel özelliklerinden öğrenme işlevini yerine getiren bilgisayar sistemleridir. Bu sistem, öğrenme işlemini örneklerden yardım alarak gerçekleştirmektedirler. Söz konusu ağlar birbirlerine bağlı yapay sinir hücrelerinden (proses unsurlarından) oluşmaktadır. Bağlantılardan her birinin bir ağırlık değeri

bulunmaktadır. Yapay sinir ağındaki bilgi, belirtilen ağırlık değerlerinde saklanıp ağa yayılmış durumdadır [31].

Yapay sinir ağıları, bilinmekte olan hesaplama yöntemleri dışında bir hesaplama yöntemini işaret etmektedir. Bulundukları ortama uyumlu hale gelebilen, eksik bilgiyle çalışabilen, adaptif, belirsizlikler altında karara varabilen, hatalar karşısında toleransları olan bu hesaplama tekniğinin neredeyse yaşamın her alanında başarılı uygulamaları görülebilmektedir. Oluşturulan ağı yapısının tespit edilmesinde, ağ parametrelerinin tercihinde, bir standardın belirlenmemesi, sorunların yalnızca nümerik bilgilerle ortaya konulabilmesi, eğitimin ne şekilde sona erdirileceğinin bilinmemesi ve ağı davranışlarını açıklamasının mümkün olmamasına karşın söz konusu ağlara olan ilgi sürekli olarak artmaktadır. Yapay sinir ağlarının en çok öne çıkan teknikleri arasında özellikle; örüntü tanıma, sınıflandırma, optimizasyon, veri sıkıştırma ve sinyal filtreleme çalışmalarını saymak mümkündür [32].

Yapay Sinir Ağları, zanlıların sinir sistemlerinden yararlanılarak geliştirilmiştir. Sinir hücreleri birbirleriyle synapsler vasıtasıyla iletişim kurmaktadır. Bir sinir hücresi tarafından işlenen bilgiler, axonları yoluyla diğer hücelere gönderilmektedir. Bunun gibi yapay sinir hücreleri, hariçten ulaşan bilgileri, bir toplama işleviyle toplamakta ve aktivasyon işlevinden geçirmek suretiyle çıktıyı üreterek ağı bağlantılarını kullanarak diğer hücelere göndermektedir. Farklı toplama ve aktivasyon işlevlerine sahiptir. Yapay sinir ağlarının birbirine bağlanmasını sağlayan bağlantı değerlerine ağırlık değerleri adı verilmektedir [31].

Proses elemanları; birbirleriyle paralel; girdi katmanı, ara katmanlar ve çıktı katmanı olmak üzere üç katman şeklinde bir araya gelmek suretiyle bir ağ oluşturmaktadırlar. Bilgiler girdi katmanından ağa iletilmektedir. Ara katmanlarda işlenmek suretiyle çıktı katmanına gönderilmektedir. Bilgi işleme noktasından kasıt ağa ulaşan bilginin ağı ağırlık değerlerinden yararlanılarak çıktıya dönüştürülmesidir. Girdiler için ağı doğru çıktıları üretebilmesi, ağırlıkların değerlerinin doğru olmasına bağlıdır. Doğru ağırlıkların bulunması işlemi “ağı eğitilmesi” olarak bilinmektedir. Söz konusu değerler, başlangıç aşamasında rasgele atanmaktadır. Ardından eğitim sırasında her örnek, ağa gösterildiği zaman ağı öğrenme kuralı dikkate alınarak ağırlıklar değiştirilmektedir. Nihayet bir başka örnek ağa gösterilerek ağırlıklar tekrar değiştirilmekte ve en doğru değerleri tespit edilmeye çalışılmaktadır. Söz konusu işlemler, ağ eğitim setinde yer alan tüm örnekler için doğru çıktılar üretinceye dek yinelenmektedir. Bu sürecin ardından test setinde yer alan örnekler ağa gösterilir; ağ test

setinde bulunan örneklere doğru cevaplar verilmesi halinde ağ eğitilmiş sayılmaktadır. Ağın ağırlıkları tespit edildikten ağırlıkların tek tek anlamları bilinmemekle ama ağın girdilerle ilgili hakkındaki kararını bu ağırlıkları dikkate alarak vermesi, bu ağırlıklarda ağın zekasının saklı olduğunu ortaya koyabilir [31].

Rizwan, Krishnan, Karthikeyan ve Kumar (2016), yaptıkları araştırmada sabah saatlerinde giden araçların, akşam saatlerinde gelen araçlardan daha fazla olduğu, ters yöndeki araç hareketinin daha az olduğu bir şehirde trafik yoğunluğuna göre sinyallerin zamanlamasını bulmak ve zamanlamayı ayarlamak amaçlanmıştır. Yolu her iki tarafındaki sinyallerin zamanlamalarını aynı anda tahmin etmek ve ayarlamak için Yapay Sinir Ağı (YSA) yöntemi kullanılmıştır. Hafif motorlu araç, ağır motorlu araç, iki ve üç tekerlekli araç hareketlerinin gerçek zamanlı trafik araştırması yapılmıştır. Araç sayısı (arabalar, otomobiller, bisikletler, kamyonlar ve otobüsler) ve yolun genişliği girdi olarak işlenmiştir. Levenberg Marquardt algoritması kullanılmıştır. Farklı yol genişliklerinden oluşan şehir tabanlı bir trafik planlamasında yolun genişliği de göz önünde bulundurulmuştur. Araştırma sonucunda modelin trafik zamanlamalarını tahmin etmek için kullanılabilir olduğu görülmüştür [33].

Patel ve Ranganathan (2001), araştırmada kentsel trafik kontrol uygulamaları için gerçek zamanlı bir karar verme sistemi kullanılmıştır. Sistem, dinamik olarak değişen ortamı öğrenip adapte edebilen geri yayılım tabanlı bir YSA ile karar verme için bulanık bir uzman sistemi entegre eder. Önerilen akıllı karar verme sisteminin performansı, uyarlanabilir trafik ışığı kontrol problemi haritalanarak değerlendirilir. Uygulama YSA yaklaşımı, FES yaklaşımı ve önerilen entegre sistem yaklaşımı kullanılarak gerçekleştirilir. Üç yaklaşımı kullanan kapsamlı simülasyonların sonuçları, entegre sistemin daha iyi performans sağladığını ve diğer iki yaklaşımdan daha verimli bir uygulamaya yol açtığını göstermektedir. YSA ön karar verici FES ise asıl karar verici olarak tasarlanmıştır [34].

Kaplan, Sağıroğlu ve Çolakoğlu (2021), araştırmada üniversite kampüs giriş/çıkış kontrolü ve istatistiksel bilgi gereksinimleri için Araç Plaka Tanıma Sistemi gerçekleştirilmiştir. Sistem elektronik algılama ve görüntü değerlendirme modüllerinden oluşmuştur. Değerlendirme modülünde görüntü işleme için yapay sinir ağı ve diğer bazı görüntü işleme teknikleri kullanılmıştır. YSA, Levenberg-Marquardt yöntemi ile eğitilmiştir [35].

Anagnostopoulos, Kayafas, Loumos ve Anagnostopoulos (2006), çalışmada yeni bir adaptif görüntü segmentasyon tekniği (kayan eşmerkezli pencereler) ve bir karakter

tanıma sinir ağı ile birlikte bağlı bileşen analizi temelinde araç plakası tanımlaması için yeni bir algoritma önermiştir. Algoritma, farklı geçmişlere ve ortam aydınlatmasına sahip 1334 doğal sahne gri seviye araç görüntüsü ile test edilmiştir. Düzgün bir şekilde bölümlenmiş plakalar 1287'dir (% 96,5). Optik karakter tanıma sistemi, tüm plaka tanıma performansı% 89.1'e ulaşan, topoloji 108-180-36 olan iki katmanlı bir olasılıksak sinir ağıdır (OSA). OSA, algoritmik görüntü işlemeden elde edilen verilere dayanarak araç plakalarından alfanümerik karakterleri tanımlamak için eğitilmiştir. Yukarıdaki iki oranı birleştirerek, lisans plakası tanıma algoritması için genel başarı oranı % 86,0'dır. Bu yazıda sunulan ilgili literatürdeki bir inceleme, mesafe, görüş açısı, aydınlatma koşulları belirlendiğinde ve arka plan karmaşıklığının düşük olduğu durumlarda daha iyi performansın (% 95'ten % 90'a kadar) bildirildiğini ortaya koymaktadır. Böylelikle şartlara göre yapay zeka tabanlı öğrenme methodologyyla plaka tanınması doğruluğu artırılabilir [36].

Akgüngör ve Doğan (2010) çalışmalarında trafik kaza tahmin modellerini, regresyon analizi, yapay sinir ağları ve genetik algoritma yöntemleri kullanılarak geliştirmiştir. Modeller geliştirilirken nüfus, araç sayısı ve kaza sayısı model parametreleri olarak kullanılmış ve bu parametrelere ait 1986- 2005 yılları arasındaki verilerden faydalanılmıştır. Regresyon analiziyle oluşturulan modellerde Smeed ve Andreassen kaza model formları kullanılmıştır. YSA modelinde 2-5-1 ağ mimarisi en uygun mimari olarak belirlenmiş, ağların gizli katmanında sigmoid, çıkış katmanında da doğrusal fonksiyon kullanılmıştır. Ağın eğitiminde ise ileri beslemeli geri yayılım algoritmasından yararlanılmıştır. Genetik algoritma tekniği ile modeller oluşturulurken farklı formdaki modeller denenmiş ancak bu çalışma için en başarılı modelin üstel model olduğu görülmüştür. Geliştirilen bütün modellerin performansları ortalama mutlak yüzde hata ortalama mutlak hata ve ortalama karesel hataların karekökü ölçütleri içinde değerlendirilmiştir [37].

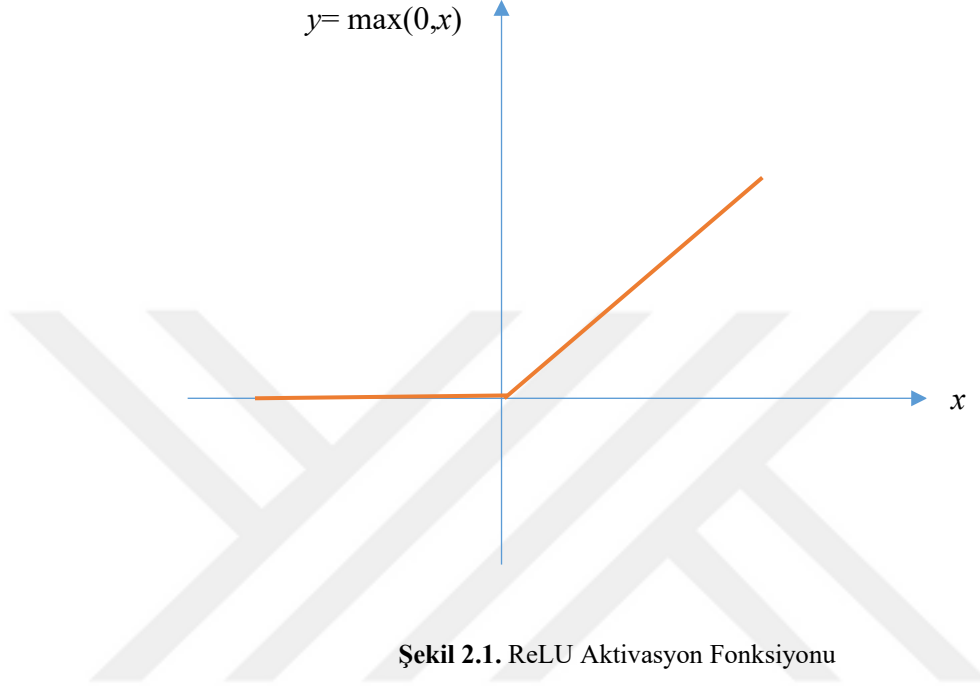
2.3.1. Yapay sinir ağında kullanılan aktivasyon fonksiyonları

2.3.1.1. ReLU

ReLU, doğrusal olmayan bir aktivasyon fonksiyonudur. ReLU işlevini kullanmanın üstünlüğü, tüm nöronların aynı anda etkinleştirilmemesidir. Bu, bir nöronun yalnızca doğrusal dönüşümün çıktısı sıfır olduğunda devre dışı bırakılacağı anlamına gelir. Negatif eksende sıfır değerlerini alması ağın daha hızlı çalışacağı anlamına da gelmektedir. Bu sebep ReLU'nun kullanımını yaygın hale getirmiştir [38].

Matematiksel olarak şu şekilde tanımlanabilir:

$$\begin{aligned} F(x) &= \begin{cases} 0, & x < 0 \\ x, & x \geq 0 \end{cases} \\ F'(x) &= \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases} \end{aligned} \quad (2.1)$$



Şekil 2.1. ReLU Aktivasyon Fonksiyonu

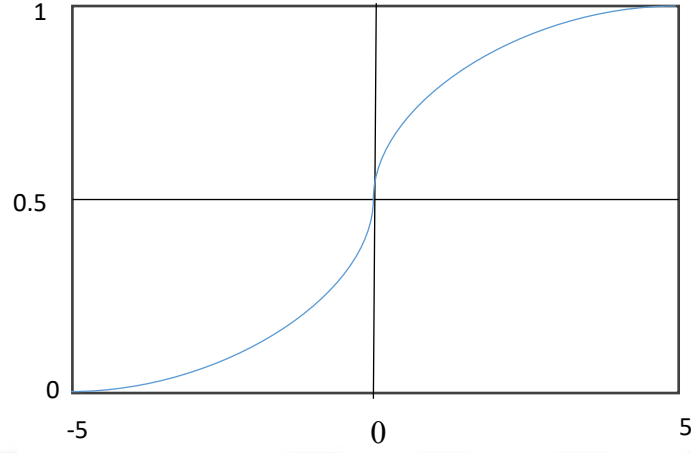
2.3.1.2. Sigmoid

Sigmoid genellikle ileri beslemeli sinir ağlarında kullanılan doğrusal olmayan aktivasyon fonksiyonudur. Gerçek girdi değerleri için tanımlanmış olan pozitif türevler ile bir dereceye kadar pürüzsüzlük ile sınırlı türevlenebilir gerçek bir fonksiyondur. Sigmoid fonksiyonu derin öğrenme mimarilerinin çıktı katmanında görünerek olasılığa dayalı çıktıyı tahmin etmek için kullanılır [39]. Sigmoid aktivasyon işlevini uygular. Küçük değerler için (<-5) sigmoid sifıra yakın bir değer verir ve büyük değerler için (>5) fonksiyonun sonucu bire yaklaşıır [40]. Matematiksel olarak şu şekilde tanımlanabilir:

$$\text{sigmoid}(x) = \frac{1}{(1+\exp(-x))} \quad (2.2)$$

Sigmoid aktivasyon fonksiyonu tanım kümesindeki elemanların her biri için 0 (sıfır) ile 1 (bir) arasında değer üretir. Sigmoidin en temel avantajı anlaşılması kolay ve

çoğunlukla derin olmayan ağlarda kullanılabilmesidir [40]. Sigmoid Aktivasyon Fonksiyon grafiği ise Şekil 2.2. de verilmiştir.



Şekil 2.2. Sigmoid Aktivasyon Fonksiyon Grafiği

2.3.1.3. Softmax

Softmax fonksiyonu, her sınıfın olasılıklarını döndürdüğü ve hedef sınıfın en yüksek olasılığa sahip olduğu çok sınıflı modellerde kullanılır. Softmax fonksiyonu genellikle kullanıldığı derin öğrenme mimarilerinin neredeyse tüm çıktı katmanlarında görünür. Sigmoid ile Softmax Activation Function (AF) arasındaki temel fark, Sigmoid'in ikili sınıflandırmada Softmax AF ise çok değişkenli sınıflandırmada kullanılmasıdır [39]. Matematiksel olarak şu şekilde tanımlanabilir:

$$f(x_i) = \frac{\exp(x_i)}{\sum \exp(x_j)} \quad (2.3)$$

2.4. Büyük Veri (Big Data)

Zaman içerisinde birçok farklı tanımlı yapılan Big Data (Büyük Veri) kavramı geleneksel veri işleme uygulama yeteneklerinin ötesindeki büyük, kompleks veri setlerini tanımlamak için kullanılır [41]. Bir başka tanımda ise Big Data insan veya makine aracılığıyla sayısal şekilde kodlanmış her türlü kurumsal veri ile sosyal medya paylaşımları ve internet vasıtasıyla ortaya çıkan kişisel verilerin işlenebilir ve anlamlı hale dönüştürülmesidir [42]. Big Data dijital ortamda etkinliğin artmasıyla beraber daha geniş kullanıcı kitlesine ulaşan çok büyük miktardaki veri kümelerini ifade etmektedir [43]. IBM'nın yayınladığı istatistiğe göre dünya genelinde günlük 2,5 Kentrilyon bayt

veri üretilmektedir. Bu boyuttaki bir veri üzerinde yapılacak işlemler bir veri depolama ünitesindeki ya da bir bilgisayardaki klasik veri tabanının kaldıramayacağı büyüklüktedir. Bu büyüklükteki bir veriyi işleme, veriyi transfer etme gibi yapılan işlerin tamamına Big Data adı verilir [41]. Big Data 3V olarak da ifade edilen Variety (Çeşitlilik), Velocity (Hız) ve Volume (Hacim) üç ana bileşenden oluşur [44]. Bazı kaynaklarda ise bu üç ana bileşene ek olarak Veracity (Gerçeklik) ve Value (Değer) bileşenleri de dahil edilmiştir [45].

Variety (Çeşitlilik): Big Data, geniş bir alanda herhangi bir türde oluşturulmakta ve bu mix veri türleri arasında standartlaştırılmış kural bulunmamaktadır. Veriler yapılandırılmış, yarı yapılandırılmış ve yapılandırılmamış olmak üzere üç farklı türde bulunmaktadır. Yapılandırılmış veri, veri tabanlarında kolaylıkla tutulmakta ve etiketlenmektedir. Geleneksel olarak veriler, özellikle operasyonel veriler; karakter, sayısal dalgali nokta vb. veri türüne dayalı olarak bir veritabanına konulmaktadır. Yarı yapılandırılmış veriler, genellikle yapılandırılmış veriler kadar katı bir şekilde tanımlanmayan bazı model veya yapılaraya sahip farklı veri türlerinin bir kombinasyonudur. Etiketleme, benzer verinin beraber gruplanmasını sağlamaktadır. Örneğin bir çağrı merkezi konuşma kayıtları, müşteri ismi, arama tarihi, şikayet bilgilerini içerebilir. Yapılandırılmamış veriler ise, ilişkisel bir veri tabanında yer alan sütun ve satırlarda depolanmamaktadır [44]. Bugüne kadar oluşturulan veriler içerisinde en fazla paya sahip olmakla birlikte en hızlı büyüyen veri kısmıdır [41].

Volume (Hacim): Veri hacmi, veri hacmini oluşturan çok sayıda işlem, olay veya geçmiş miktarı ile ölçülebilir, ancak hacim genellikle öznelitlikler, boyutlar veya öngörücü değişkenler tarafından daha da şiddetlenir. Tipik olarak, analitik, tahmine dayalı modeller oluşturmak için örnekler adı verilen daha küçük veri kümeleri kullanır. Veri hacmi, depolama veya hesaplama işlemleri kısıtlı olduğundan, çoğunlukla, iş kullanım durumu veya tahmine dayalı içgörü önemli ölçüde azalmıştır. Kuruluşlar, veri hacmi kısıtlamasını kaldırarak ve daha büyük veri kümeleri kullanarak tahmin modellerinin doğruluğunu artırma daha fazla gözlem veya değişken ekleyebilirler. Ek olarak, veriler üzerindeki bağları serbest bırakarak, kurumlar, birbiriyle ilişkili bilgi bitlerinin gerçek dünyadaki karmaşıklıklarını yansıtan daha doğru tahminler oluşturmak için daha uzun bir süre boyunca verilere bakabilirler [44].

Velocity (Hız): Veri hızı, verilerin oluşturulduğu, biriktirildiği, alındığı ve işlendiği hız ile ilgilidir [44]. Hız bileşeni Big Data'nın çok yüksek hızda üretildiğini ifade eder [41]. Her geçen zaman verinin üretilme hızı daha da artmaktadır [43]. Yüksek hızda

oluşturulan verilere Twitter mesajları veya Facebook gönderileri örnek olarak verilebilir [46].

Veracity (Gerçeklik): Veracity, analiz edilen verilerin kalitesini ifade eder. Yüksek doğruluk oranına sahip veriler, analiz edilmesi değerli ve genel sonuçlara anlamlı bir şekilde katkıda bulunan birçok kayda sahiptir. Öte yandan, düşük gerçekliğe sahip veriler, yüksek oranda anlamsız veriler içerir. Bu veri setlerinde değerli olmayan, gürültü olarak adlandırılır. Yüksek doğruluk oranına sahip bir veri kümesine örnek, tıbbi bir deney veya denemeden elde edilen veriler olabilir. Anlamlı bilgileri ortaya çıkarmak için yüksek hacimli, yüksek hızlı ve çok çeşitli veriler, gelişmiş araçlarla (analitik ve algoritmalar) işlenmelidir [46].

Value (Değer): Orijinal formda alınan veri genellikle ilk başta hacmine kıyasla düşük değere sahiptir yani Big Data çoğunlukla düşük yoğunluklu değeri ifade eder [43]. Big Data üzerinde yapılacak olan çalışmaların sonal hedefi değerli bilgi elde etmektir. Farklı kaynaklardan alınan çok miktardaki verilerin analiziyle elde edilen bilgilerle kurumlara sağlayacağı değer de bir o kadar büyük olması beklenir [41].

Amini, Gerostathopoulos ve Prehofer (2017), eş zamanlı trafik kontrolü için dağıtılmış bilgi işlem platformuna dayalı mevcut trafik kontrol sistemlerinin gereksinimlerinin sistem analizine dayandığı çalışmada; sanal ortamda gerçekleştirilen araştırmada araçların hız, konum, seyahat süresi, gerçek aracın etrafındaki araç sayısı ve o araçların ortalama hızı Kafka alıcısına gönderilmiştir. Kısmen somutlaştırılan veriler sonucunda trafik kontrolü için kapsamlı ve esnek Big Data mimarisi önerilmiştir [47].

Shengdong, Zhengxian ve Yixiang (2019), Cloud sistemi üzerinde yapay trafik kontrol sistemi ve Big Data madenciliği ile trafik yoğunluğunu tahmin edici sistem tasarlanmıştır. Program üstün akıllı ulaşım kontrol teknolojisi ve akıllı ulaşım ağı sanallaştırma teknolojisini içerir. Akıllı trafik akışı verilerine dayanarak bulut kontrol yönetimi sunucusunun merkezinde, kentsel yol kısa vadeli trafik akışını ve yoğunluğunu, trafik akışı verilerini tahmin etmek için derin öğrenme ve overrun öğrenme yöntemleri kullanılmıştır. Gerçek zamanlı trafik akış kontrol stratejisi için akıllı optimizasyon planlama algoritması kullanılmıştır [48].

Singh, Vishnu ve Mohan (2016), yapılan araştırmada şehir trafiğinde kask takmayan bisiklet sürücülerinin otomatik tespiti için görsel Big Data kullanılmıştır. Şehir ölçeğindeki bir gözetleme kamerası ağında kask kullanmadan motosiklete binen trafik kuralı ihlal edenlerin gerçek zamanlı tespiti için görsel büyük veri analitiğine dayalı bir yapı önerilmiştir. Önerilen yapı aynı zamanda trafik polisine, farklı çevre koşullarında bu

tür ihlalleri tespit etme konusunda yardımcı olması hedeflenmiştir. İşlenen videolar Cloud ortamına aktarılarak Big Datadan çekilmiştir. Deneysel sonuçlar, bisiklet sürücüsünün tespiti ve ihlal edenlerin tespiti için sırasıyla % 98.88 ve % 93.80 olan yüksek sınıflandırma performansını göstermektedir. Bir kareyi işlemek için geçen ortalama süre ≈ 11 ms'dir. Ayrıca önerilen çerçeve, gerekirse hafif ayarlamalarla yeni senaryolara otomatik olarak uyum sağlamaktadır [49].



3. MATERYAL VE YÖNTEM

Bu araştırmada kurgulanan donanım ve yazılımlarda aşağıda açıklaması verilen programlar, kütüphaneler, veri anlamlandırma yöntemleri kullanılmıştır.

Python, etkileşimli nesne yönelimli programlama dili olup çok yüksek seviyeli dinamik veri türlerini ve sınıfları içerir. Prosedürel ve fonksiyonel programlama gibi nesne yönelimli programlamanın ötesinde çoklu programlama paradigmalarını destekler. Birçok farklı problem sınıfına uygulanabilen üst düzey bir genel amaçlı programlama dilidir [50]. Guido van Rossum tarafından 1991’de oluşturulmuştur. Python kısa bir dildir ve kod okunabilirliği için tasarlanmıştır. Python programları, diğer programlama dillerinden daha az kod satırı gerektirme eğilimindedir. Python, en basit programlardan devasa büyük ölçekli projelere kadar her şey için kullanılabilen ölçeklenebilir bir dildir [51]. Kodlanan tüm yazılımlarda Python 3 programlama dili kullanılmıştır.

Anaconda, entegre geliştirme ortamı olarak da bilinen IDE (Integrated Development Environment) platformunda kod yazılmasına imkan verip farklı ortamlara erişim sağlayan yazılımdır. IDE’ler kod yazmak, kodu düzenlemek ve oluşan hataları ayıklamak, verileri görselleştirmek, verileri incelemek, değişkenleri depolamak, sonuçları sunmak ve projeler üzerinde iş birliği yapmak başta olmak üzere birçok yararlı özellik içerir [52]. Bu çalışmada IDE olarak Anaconda 3 platformu kullanılmıştır.

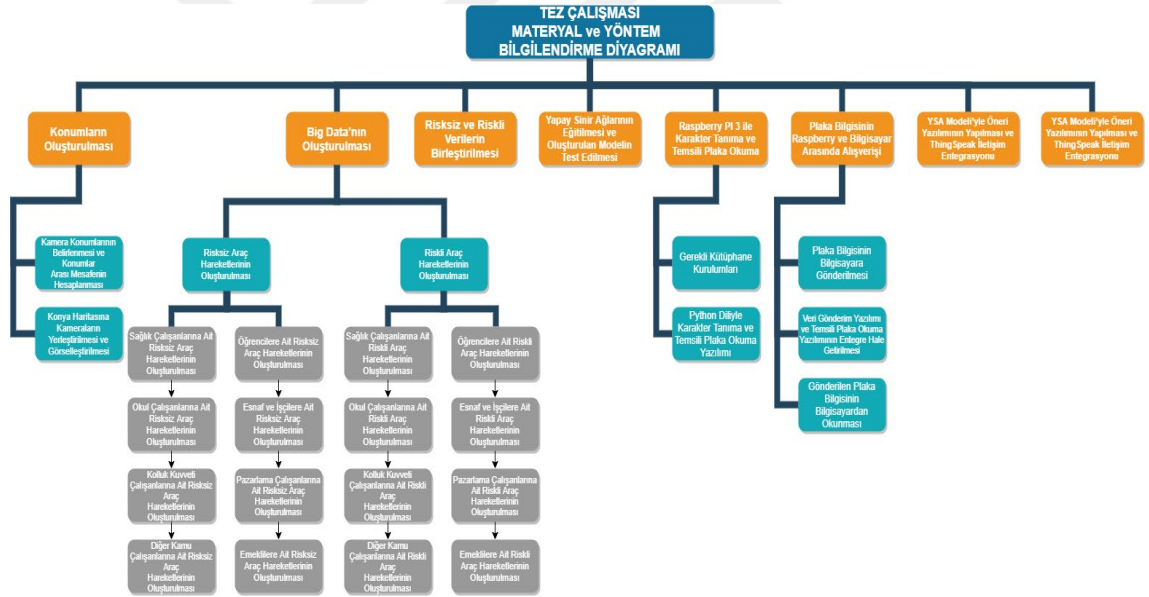
Spyder, betik kodları yazmak için bir düzenleyici, verileri kolayca incelemek için bir değişken gezgini, bir Ipython konsolu gibi özellikleri içerisinde barındıran IDE olarak tanımlanabilir. Özellikle veri analizi için tasarlanmış oldukça işlevsel bir IDE'dir. Arayüz özelleştirebilme avantajının yanı sıra kullanıcının hataları algılayarak hataları düzeltmek için kodu bütün ya da satır şeklinde analiz etmesine olanak tanıyan kod inceleme ve hata ayıklama için güçlü araçlar içerir [52]. Anaconda içerisinde dahili olarak gelen Spyder 3 bu çalışmada kullanılmıştır.

One Hot Coding, en yaygın kullanılan kodlama şemasıdır. Kategorik değişkenin her seviyesini sabit bir referans seviyesiyle karşılaştırır. One Hot Coding, n gözlem ve d farklı değer içeren tek bir değişkeni, her biri n gözlemli d ikili değişkene dönüştürür. Her gözlemde ikili değişkenin varlığını (1), yokluğunu ise (0) şeklinde göstermektedir [53]. Bu çalışmada YSA model eğitimine girecek veriler One Hot metoduna göre kodlanmıştır.

Tensorflow, Google ekibindeki araştırmacılar tarafından geliştirilen veri bilimi için bir derin öğrenme çerçevesi ve ücretsiz açık bir kaynak kitaplığıdır. Tensorflow çeşitli alanlarda kullanılabilir. Nesne algılanması alanında, bir nesne algılama modelinin

oluşturulması, nesnenin eğitilmesi ve yerleştirilme sürecini basitleştirmek için kullanılabilirliği kullanılan alanlardan bazılarıdır. Tensorflow da nesne algılama API çerçevesi bulunmaktadır. Tensorflow nesne algılama API çerçevesi, kullanıcı için önceden eğitilmiş bir nesne algılama modeli sağlamaktadır [54]. *Keras*, makine öğrenimi platformu TensorFlow üzerinde çalışan, Python'da yazılmış bir derin öğrenme API'sidir. Hızlı deney yapmaya odaklanılarak geliştirilmiştir. Keras'ın temel veri yapıları katmanlar ve modellerdir. En basit model türü, doğrusal bir katman yığını olan sequential modeldir [55]. Sequential (sıralı) model her katmanın bir giriş tensörüne ve bir çıkış tensörüne sahip olduğu düz bir katman yığını olduğu durumda kullanılabilir [56]. Bu çalışmada YSA modelinin oluşturulması ve eğitiminde YSA fonksiyon ile parametrelerini içererek bunları yalın bir komutla işlevselleştiren Tensorflow ve Keras kütüphaneleri kullanılmıştır.

Tez çalışmasında izlenen ise yol şekil 3.1. de gösterilmiştir.



Şekil 3.1. Araştırma Yöntem Diyagramı

df_distance - DataFrame

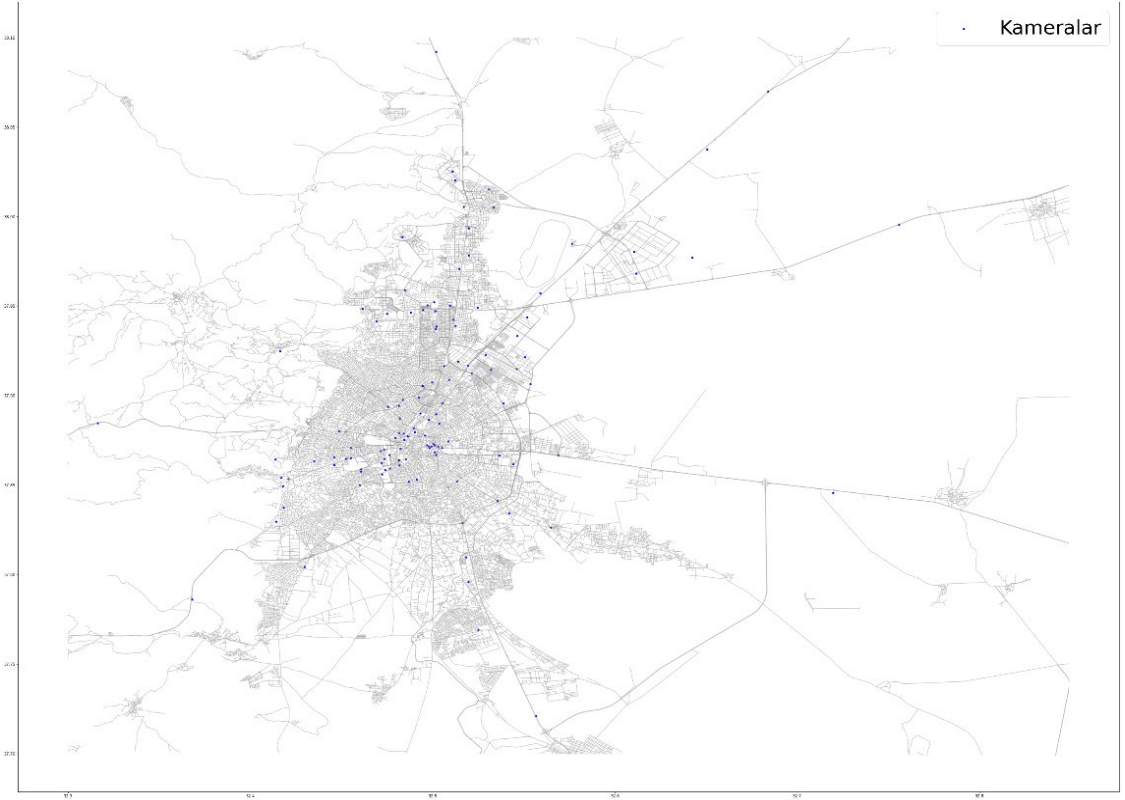
Index	Unnamed: 0	nereye	nereye	mesafe	zaman	bilgi_sinif	2bilgi_sinif	bilgi_nereye	bilgi_nereye
2866	2866	CM24	CM15	12555	1078	Pastane	Yurt		
10423	10423	CM85	CM8	4143	732	Okul_2	Kimyasal		
9319	9319	CM76	CM20	10698	1010	Sanayi	Ekmek_Firini		
3372	3372	CM28	CM25	3944	487	Okul_3	Sanayi		
11876	11876	CM96	CM97	11097	840	Polis_Merkezi	Polis_Merkezi		
2689	2689	CM22	CM86	4762	512	Ekmek_Firini	Kafe		
14213	14213	CM115	CM78	12289	1076	Fetih_Cad	Sanayi		
9708	9708	CM79	CM37	6639	632	Sanayi	Vakif_dernek		
7215	7215	CM59	CM24	27787	1482	Y_istanbul_c...	Pastane		
12098	12098	CM98	CM71	25855	1767	Okul_4	Park		
2450	2450	CM20	CM95	16048	1216	Ekmek_Firini	Polis_Merkezi		
8192	8192	CM67	CM9	33792	1749	Aksaray_cevr...	Silah_Satis		
5057	5057	CM41	CM98	12868	1255	Park	Okul_4		
6878	6878	CM56	CM59	26734	1932	Belediye_3	Y_istanbul_c...		
9342	9342	CM76	CM43	5465	914	Sanayi	Banka		
9840	9840	CM80	CM45	8809	902	Okul_3	Sanayi		
10684	10684	CM87	CM21	14955	1360	Kafe	Ekmek_Firini		
996	996	CM9	CM5	4860	508	Silah_Satis	Otel		
13563	13563	CM110	CM48	18949	1401	YeniMeram_Cad	Sanayi		
3810	3810	CM31	CM91	24082	1918	Kafe	Polis_Merkezi		
7911	7911	CM64	CM100	4590	463	Avm	Okul_4		
515	515	CM5	CM20	4040	275	Otel	Ekmek_Firini		
3667	3667	CM30	CM72	20934	1701	Bar	Belediye_4		
9979	9979	CM81	CM60	23674	1803	Okul_3	Antalya_Cevr...		
2685	2685	CM22	CM82	8397	819	Ekmek_Firini	Okul_3		
13341	13341	CM108	CM74	3820	589	AhmetOzcan_C...	Valilik_1		
5775	5775	CM47	CM72	9987	1047	Sanayi	Belediye_4		
12496	12496	CM101	CM97	12296	1035	Hastane	Polis_Merkezi		
2398	2398	CM20	CM43	7632	810	Ekmek_Firini	Banka		
4206	4206	CM34	CM115	3552	478	Noter	Fetih_Cad		
11229	11229	CM91	CM70	12041	1120	Polis_Merkezi	Park		

Format Resize Background color Column min/max OK Cancel

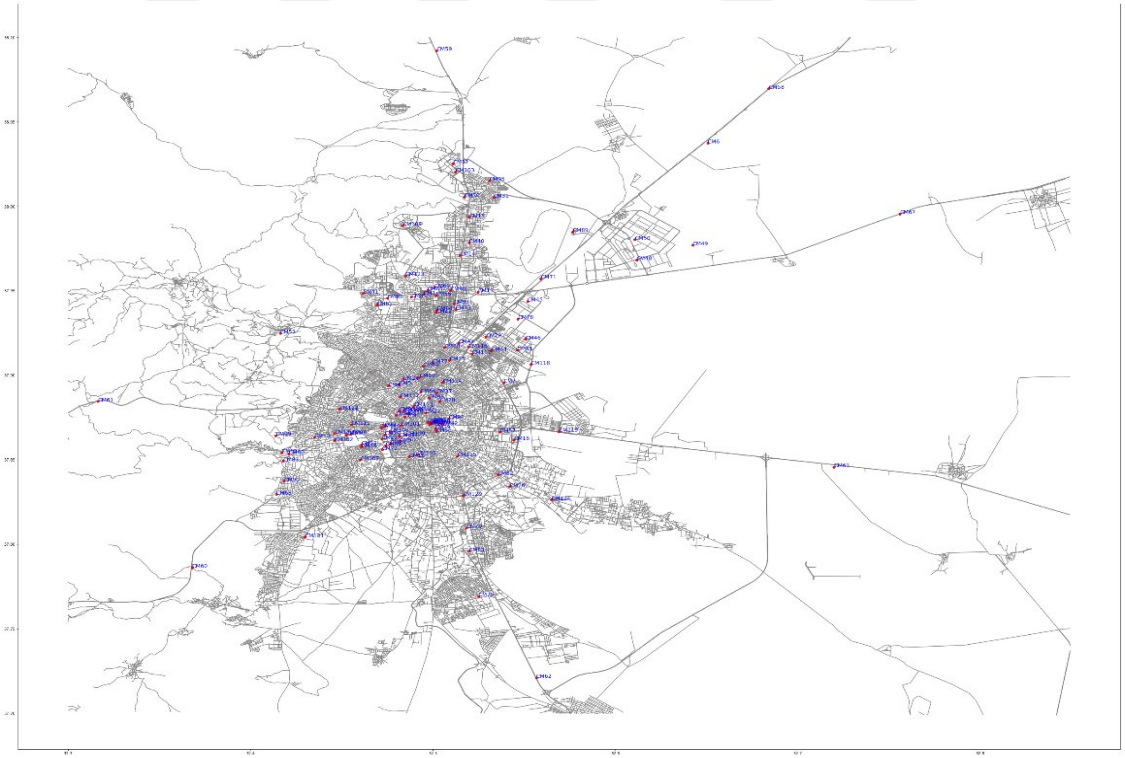
Şekil 3.3. 124x6 Boyutlarında Oluşturulan DataFrame

3.1.2. Konya haritasına kameraların yerleştirilmesi ve görselleştirilmesi

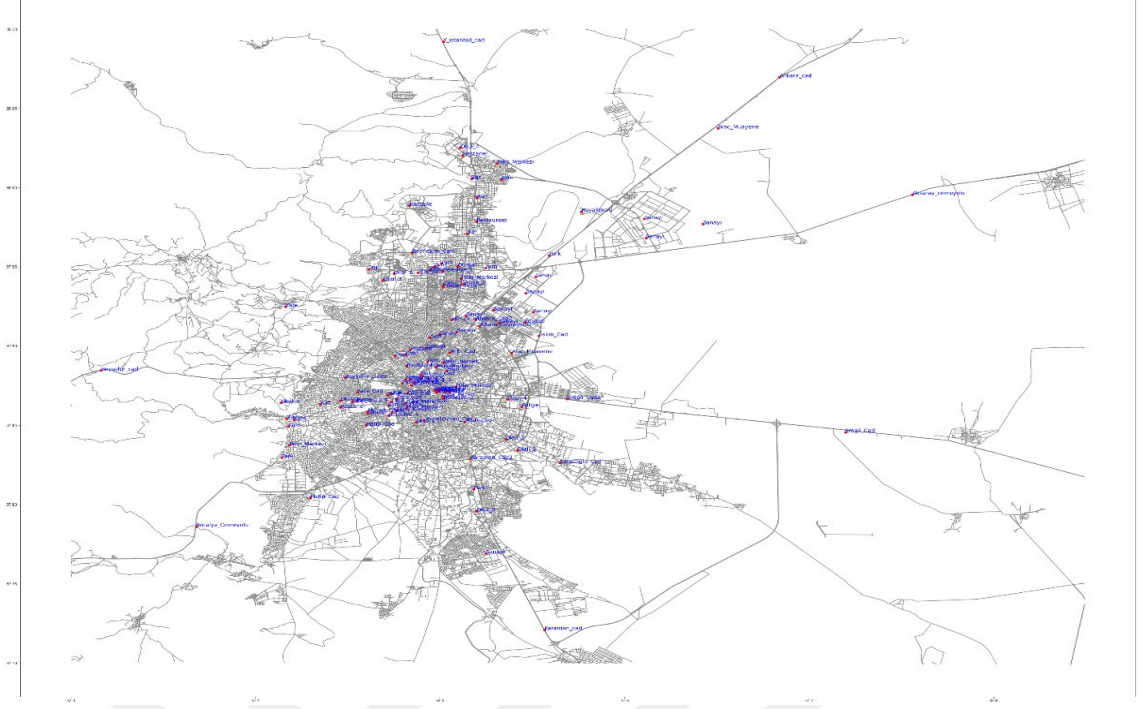
Konya merkez ilçelerinin 2 Boyutlu çizimini içeren .shp uzantılı harita dosyası Geopandas kütüphanesi ile okunarak her bir noktanın enlem ve boylam bilgilerini içerecek şekilde Matplotlib kütüphanesi yardımıyla çizdirilmiştir. Her bir kameranın konumunun bilgilendirmesini içeren df_distance DataFrame'inden açıklamalar çekilip çizdirilen haritaya eklenmiştir. Şekil 3.4. de enlem ve boylam bilgileri bilinen lokasyonların Konya haritasında noktasal gösterimi verilmiştir. Şekil 3.5. de noktasal gösterimi verilen lokasyonların kamera bilgileriyle konumları gösterilmiştir. Şekil 3.6. da ilgili kameraların df_distance DataFrame'deki "*bilgi_sinif*" sütunu bilgileri verilmiştir.



Şekil 3.4. Koordinatlara Nokta Atanması



Şekil 3.5. Koordinatlara Atanmış Kameraların Yerleştirilmesi



Şekil 3.6. Kameraların İçerdiği Koordinatlara İlişkin Açıklamaların Gösterilmesi

3.2. Big Data'nın Oluşturulması

Çalışmada araç hareketleri, belirli meslek sınıfları çerçevesinde risksiz ve riskli olarak günlük frekansta, 30 günlük periyotta üretilmiştir. Her meslek sınıfının hareketleri anlamlı sonuçlar alabilmek üzere ilgili meslek çalışanlarına sorularak ortalama bir hareket sınırlamaları oluşturulmuştur.

Üretilen araç verilerinde, aracın bir kameraya görünmesi veya zamanlama bilgisinin anlaşılması, ilgili kamera verisinin ve zaman sınıfının “1” olması ile belli olmaktadır. Araç görünmediği kamera ve zamanlarda ilgili sütunun verisi “0” değerini almıştır.

Yazılım sonrası 7942 araca ait 476520 satır, 137 sütun veri oluşturulmuştur. Her araca bir temsili plaka atanmıştır.

3.2.1. Risksiz araç hareketlerinin oluşturulması

Çalışan meslek grubuna ait her bir araç için oluşturulan rota,

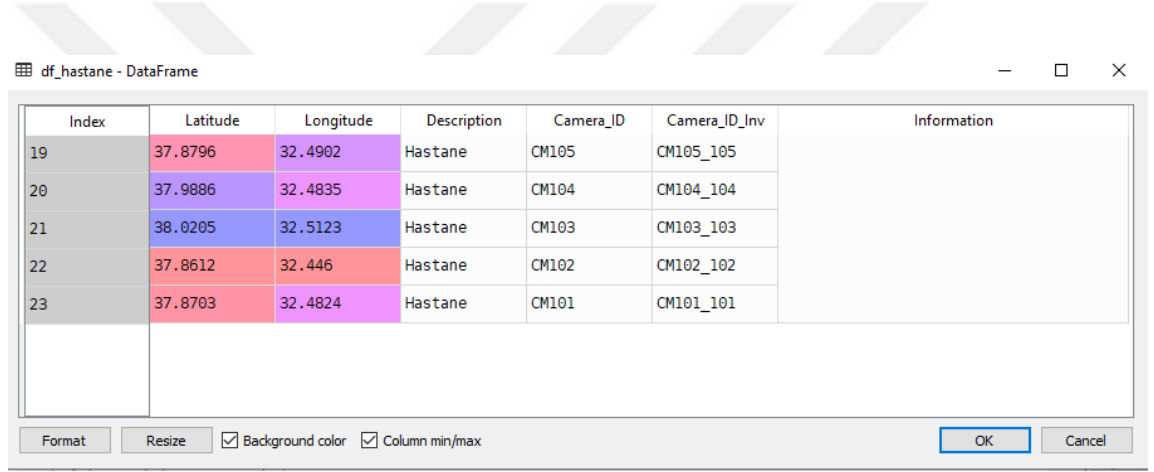
- Hız sınırları aşmama
- Toplumsal risk oluşturan (yanıcı, patlayıcı, kesici, delici vb.) ürünler satan iş yerlerine uğramama
- Hafta içi akşam vakitlerinde sosyal alanlara ilgili lokasyonlara mesai saati içerisinde ziyaret yapma

- Hafta içi bir günün, diğer hafta içi günleri saatlerine yakın zamanlarda mesai gidiş ve dönüş gerçekleştirme
- Hafta sonu sosyal alanlara ilgili konumların mesai saatleri içerisinde ziyaret gerçekleştirme

Kriterlerine bağlı olarak risksiz araç hareketleri oluşturulmuştur. DataFrame'deki "Durum" sütunu risksiz anlamını ifade eden "0" olarak belirlenmiştir.

3.2.1.1. Sağlık çalışanlarına ait risksiz araç hareketlerinin oluşturulması

Sağlık çalışanlarına ait araç hareketleri toplamda 5 yoğun hastane seçilerek 250 farklı araçla oluşturulmuştur. Yazılım sonucunda 15.000 eşsiz araç sürüşü üretilmiştir. Meslek sınıfı "0" olarak belirlenmiştir. Seçilen hastane listesi Şekil 3.7. de yer almaktadır.



Index	Latitude	Longitude	Description	Camera_ID	Camera_ID_Inv	Information
19	37.8796	32.4902	Hastane	CM105	CM105_105	
20	37.9886	32.4835	Hastane	CM104	CM104_104	
21	38.0205	32.5123	Hastane	CM103	CM103_103	
22	37.8612	32.446	Hastane	CM102	CM102_102	
23	37.8703	32.4824	Hastane	CM101	CM101_101	

Şekil 3.7. Sağlık Çalışanlarına Ait Seçilen Hastane Listesi

3.2.1.2. Okul çalışanlarına ait risksiz araç hareketlerinin oluşturulması

Okul çalışanlarına ait araç hareketleri toplamda 4 farklı eğitim grubunda 17 farklı okul seçilerek 540 farklı araçla oluşturulmuştur. Yazılım sonucunda 32400 değişik araç sürüşü ortaya çıkmıştır. Meslek sınıfı "1" olarak belirlenmiştir. Seçilen okul listesi Şekil 3.8. de yer almaktadır.

df_okul - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_ID_Inv	Information
24	37.8787	32.4842	Okul_4	CM100	CM100_100	
25	37.8643	32.4138	Okul_4	CM99	CM99_99	
26	37.8647	32.4523	Okul_4	CM98	CM98_98	
39	37.8584	32.4741	Okul_2	CM85	CM85_85	
40	37.9386	32.5022	Okul_2	CM84	CM84_84	
41	37.7958	32.5196	Okul_3	CM83	CM83_83	
42	37.8412	32.5354	Okul_3	CM82	CM82_82	
43	37.9415	32.4691	Okul_3	CM81	CM81_81	
44	37.9457	32.475	Okul_3	CM80	CM80_80	
67	37.8663	32.5368	Okul_4	CM57	CM57_57	
92	38.0252	32.5109	Okul_4	CM32	CM32_32	
95	37.8648	32.4735	Okul_3	CM29	CM29_29	
96	37.8844	32.5036	Okul_3	CM28	CM28_28	
97	37.8778	32.4957	Okul_3	CM27	CM27_27	
98	37.8341	32.5419	Okul_2	CM26	CM26_26	
107	37.8656	32.4461	Okul_1	CM17	CM17_17	
108	37.8518	32.4869	Okul_1	CM16	CM16_16	

Format Resize ☒ Background color ☒ Column min/max OK Cancel

Şekil 3.8. Okul Çalışanlarına Ait Seçilen Okul Listesi

3.2.1.3. Kolluk kuvveti çalışanlarına ait risksiz araç hareketlerinin oluşturulması

Kolluk Kuvvetleri çalışanlarına ait araç hareketleri toplamda 7 farklı Polis Merkezi seçilerek 350 ayrı özelleştirilmiş araçla oluşturulmuştur. Yazılım sonucunda 21000 farklı araç sürüşü üretilmiştir. Meslek sınıfı “2” olarak belirlenmiştir. Seçilen polis merkezi listesi Şekil 3.9. da yer almaktadır.

df_polism - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_ID_Inv	Information
27	37.9425	32.5115	Polis_Merkezi	CM97	CM97_97	
28	37.8765	32.4796	Polis_Merkezi	CM96	CM96_96	
29	38.0153	32.5309	Polis_Merkezi	CM95	CM95_95	
30	37.861	32.4818	Polis_Merkezi	CM94	CM94_94	
31	37.8865	32.4977	Polis_Merkezi	CM93	CM93_93	
32	37.8743	32.5087	Polis_Merkezi	CM92	CM92_92	
33	37.8373	32.4184	Polis_Merkezi	CM91	CM91_91	

Format Resize ☒ Background color ☒ Column min/max OK Cancel

Şekil 3.9. Kolluk Kuvveti Çalışanlarına Ait Seçilen Polis Merkezi Listesi

3.2.1.4. Diğer kamu çalışanlarına ait risksiz araç hareketlerinin oluşturulması

Diğer Kamu çalışanlarına ait araç hareketleri toplamda 13 farklı kamu binası seçilerek 220 farklı araçla oluşturulmuştur. Yazılım sonucunda 13200 farklı araç sürüşü üretilmiştir. Meslek sınıfı “3” olarak belirlenmiştir. Seçilen kamu binası listesi Şekil 3.10. da verilmiştir.

df_digerkamu - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_ID_Inv	Information
70	37.8773	32.4862	Belediye_1	CM54	CM54_54	
89	37.879	32.4816	Noter	CM35	CM35_35	
105	37.8716	32.498	Postahane	CM19	CM19_19	
81	37.8723	32.4968	Banka	CM43	CM43_43	
90	37.8707	32.4982	Noter	CM34	CM34_34	
18	37.8668	32.4764	Tren_gari	CM106	CM106_106	
35	37.9849	32.5766	Havalimani	CM89	CM89_89	
88	37.8723	32.5011	Noter	CM36	CM36_36	
52	37.856	32.4721	Belediye_4	CM72	CM72_72	
69	37.9473	32.5016	Belediye_2	CM55	CM55_55	
51	37.939	32.5124	Valilik_2	CM73	CM73_73	
50	37.8715	32.4993	Valilik_1	CM74	CM74_74	
68	37.8668	32.5018	Belediye_3	CM56	CM56_56	

Format Resize ☒ Background color ☒ Column min/max OK Cancel

Şekil 3.10. Diğer Kamu Çalışanlarına Ait Seçilen Kamu Binası Listesi

3.2.1.5. Öğrencilere ait risksiz araç hareketlerinin oluşturulması

Öğrencilere ait araç hareketleri toplamda 5 farklı üniversite kampüsü ve 4 farklı yurt binası seçilerek birbirinden farklı 75 araçla oluşturulmuştur. 75 aracın tümü sadece yurt kullanmamıştır. Diğer meslek sınıfları gibi ikamet kullanabileceği göz önüne alınmıştır. Yazılım sonucunda 4500 farklı araç sürüşü üretilmiştir. Meslek sınıfı “4” olarak belirlenmiştir. Seçilen yurt binası listesi Şekil 3.11. de verilmiştir.

df_uni - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_ID_Inv	Information
67	37.8663	32.5368	Okul_4	CM57	CM57_57	
25	37.8643	32.4138	Okul_4	CM99	CM99_99	
26	37.8647	32.4523	Okul_4	CM98	CM98_98	
24	37.8787	32.4842	Okul_4	CM100	CM100_100	
92	38.0252	32.5109	Okul_4	CM32	CM32_32	

df_yurt - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_ID_Inv	Information
109	37.9935	32.5199	Yurt	CM15	CM15_15	
110	37.9708	32.5147	Yurt	CM14	CM14_14	
111	37.8634	32.435	Yurt	CM13	CM13_13	
112	37.8541	32.417	Yurt	CM12	CM12_12	

Şekil 3.11. Öğrencilere Ait Seçilen Yurt Binası Listesi

3.2.1.6. Esnaf ve işçilere ait risksiz araç hareketlerinin oluşturulması

Esnaf ve işçilere ait araç hareketleri toplamda 14 farklı sanayi seçilerek 2100 değişik araçla oluşturulmuştur. Yazılım sonucunda 126000 farklı araç sürüşü üretilmiştir. Meslek sınıfı “5” olarak belirlenmiştir. Seçilen sanayi listesi 3.12. de yer almaktadır.

df_sanayi - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_ID_Inv	Information
45	37.9227	32.529	Sanayi	CM79	CM79_79	
46	37.9333	32.5465	Sanayi	CM78	CM78_78	
47	37.9073	32.4996	Sanayi	CM77	CM77_77	
48	37.8575	32.4607	Sanayi	CM76	CM76_76	
49	37.7692	32.525	Sanayi	CM75	CM75_75	
74	37.9802	32.6106	Sanayi	CM50	CM50_50	
75	37.977	32.6424	Sanayi	CM49	CM49_49	
76	37.9683	32.6116	Sanayi	CM48	CM48_48	
77	37.9191	32.514	Sanayi	CM47	CM47_47	
78	37.9214	32.5507	Sanayi	CM46	CM46_46	
79	37.9437	32.5518	Sanayi	CM45	CM45_45	
80	37.9146	32.532	Sanayi	CM44	CM44_44	
99	37.9089	32.509	Sanayi	CM25	CM25_25	
114	37.8989	32.4926	Sanayi	CM10	CM10_10	

Format Resize ☒ Background color ☒ Column min/max OK Cancel

Şekil 3.12. Esnaf ve İşçilere Ait Seçilen Sanayi Listesi

3.2.1.7. Pazarlama çalışanlarına ait risksiz araç hareketlerinin oluşturulması

Pazarlama mesleği noktasal lokasyonlara değil diğer meslek gruplarına bağlı olduğundan pazarlama çalışanlarına ait araç hareketleri oluşturulurken okul, avm, park, kafe, otel, yurt, belediye, banka, sanayi, postane gibi çoklu meslek gruplarının konumlarına gitmesi sağlanmıştır. 295 farklı araçla üretilen araç hareketleri, yazılım sonucunda 17700 araç sürüşü ortaya çıkarılmıştır. Meslek sınıfı “6” olarak belirlenmiştir. Seçilen konumlara ait liste Şekil 3.13. de yer almaktadır.

The figure displays four DataFrames, each representing a different category of locations. Each DataFrame has the following columns: Index, Latitude, Longitude, Description, Camera_ID, and Camera_Link. The data is color-coded by row.

df_okul - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_Link
24	37.8787	32.4842	Okul_4	CM100	CM100_10
25	37.8643	32.4138	Okul_4	CM99	CM99_99
26	37.8647	32.4523	Okul_4	CM98	CM98_98
39	37.8584	32.4741	Okul_2	CM85	CM85_85
40	37.9386	32.5022	Okul_2	CM84	CM84_84
41	37.7958	32.5196	Okul_3	CM83	CM83_83
42	37.8412	32.5354	Okul_3	CM82	CM82_82
43	37.9415	32.4691	Okul_3	CM81	CM81_81
44	37.9457	32.475	Okul_3	CM80	CM80_80
67	37.8663	32.5368	Okul_4	CM57	CM57_57
92	38.0252	32.5109	Okul_4	CM32	CM32_32
95	37.8648	32.4735	Okul_3	CM29	CM29_29
96	37.8844	32.5036	Okul_3	CM28	CM28_28
67	37.8778	32.4957	Okul_3	CM27	CM27_27

df_sanayi - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_Link
45	37.9227	32.529	Sanayi	CM79	CM79_79
80	37.9146	32.532	Sanayi	CM44	CM44_44
78	37.9214	32.5507	Sanayi	CM46	CM46_46
114	37.8989	32.4926	Sanayi	CM10	CM10_10
47	37.9073	32.4996	Sanayi	CM77	CM77_77
99	37.9089	32.509	Sanayi	CM25	CM25_25
74	37.9802	32.6106	Sanayi	CM50	CM50_50
49	37.7692	32.525	Sanayi	CM75	CM75_75
46	37.9333	32.5465	Sanayi	CM78	CM78_78
76	37.9683	32.6116	Sanayi	CM48	CM48_48
79	37.9437	32.5518	Sanayi	CM45	CM45_45
77	37.9191	32.514	Sanayi	CM47	CM47_47
48	37.8575	32.4607	Sanayi	CM76	CM76_76
75	37.977	32.6474	Sanayi	CM49	CM49_49

df_sosyal - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_Link
59	37.9504	32.4971	Avm	CM65	CM65_65
56	37.8296	32.4141	Park	CM68	CM68_68
53	37.9572	32.5591	Park	CM71	CM71_71
93	38.0053	32.5332	Kafe	CM31	CM31_31
54	37.8097	32.5183	Park	CM70	CM70_70
38	37.8535	32.4209	Kafe	CM86	CM86_86
37	37.8493	32.4178	Kafe	CM87	CM87_87
55	37.9521	32.5008	Park	CM69	CM69_69
60	37.9054	32.4945	Avm	CM64	CM64_64
72	37.9249	32.4163	Kafe	CM52	CM52_52
58	37.8899	32.4931	Avm	CM66	CM66_66
113	37.9491	32.5246	Avm	CM11	CM11_11
83	37.9486	32.4617	Park	CM41	CM41_41

df_digerisyerleri - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_Link
123	37.9478	32.4947	Otel	CM1	CM1_1
111	37.8634	32.435	Yurt	CM13	CM13_13
112	37.8541	32.417	Yurt	CM12	CM12_12
120	37.8936	32.4754	Otel	CM4	CM4_4
121	37.8714	32.503	Otel	CM3	CM3_3
119	37.8944	32.4815	Otel	CM5	CM5_5
109	37.9935	32.5199	Yurt	CM15	CM15_15
68	37.8668	32.5018	Belediye_3	CM56	CM56_56
69	37.9473	32.5016	Belediye_2	CM55	CM55_55
105	37.8716	32.498	Postahane	CM19	CM19_19
70	37.8773	32.4862	Belediye_1	CM54	CM54_54
52	37.856	32.4721	Belediye_4	CM72	CM72_72
81	37.8723	32.4968	Banka	CM43	CM43_43

Şekil 3.13. Pazarlama Çalışanlarına Ait Seçilen Konum Listesi

3.2.1.8. Emeklilere ait risksiz araç hareketlerinin oluşturulması

Emeklilere ait araç hareketleri halk pazarı, kafe, restoran, park, alışveriş merkezi, şehirlerarası seyahat merkezleri gibi sosyal, kamusal noktalar seçilerek 140 farklı araçla oluşturulmuştur. Yazılım sonucunda 8400 farklı araç sürüşü üretilmiştir. Meslek sınıfı “7” olarak belirlenmiştir.

3.2.2. Riskli araç hareketlerinin oluşturulması

Trafik denetim ekipleri, sürücünün ırkı, cinsiyeti, yaşı, sosyal sınıfı, ekonomik gelir kademesi gibi birçok farklı sebepleri göz önünde bulundurarak araç durdurma işlemiyle kontrol gerçekleştirmektedir [57;58;59;60;61]. Bu sebepler ve sebeplerin etkinliği, yetkili kişinin yaşına, etnik kökenine, deneyim süresine, cinsiyetine bağlı olarak değişmektedir [62].

Farklı bölgelerde yapılan trafik denetim uygulamalarının amacını da içeren veriler incelendiğinde, alkol, silah, uyuşturucu, hız aşımı veya anormal araç hareketi tabanlı durdurmalar gerçekleştirildiği görülmüştür [63;64;65;66].

Trafik denetim ekiplerinin kontrol amaçları, denetim lokasyonu ile yerel araçların durumuna göre farklılık göstermektedir [63]. Denetim yapılacak lokasyonun kültür, sosyal, gelir durumu gözönüne alınarak denetim amacına göre uygulama yapılması dolayısıyla, Konya ilinin yaşam şartları düşünülerek risk kriterleri aşağıdaki gibi seçilerek riskli araç verileri oluşturulmuştur.

Her bir araç için risk kriterleri:

- Hız sınırı aşımı
- Kimyasal ürün satış mağaza ziyareti
- Silah ve türevlerinin satış yeri
- Alkollü eğlence merkezi
- Çalışanlar için mesai sonrası mesleği ile ilgisiz yüksek sıklıktaki benzer işyeri ziyareti
- 23.00 – 06.30 arası kafe, restoran, park gibi kamusal alanlarda kısa veya uzun süreli bulunma eylemi olarak belirlenmiştir.

Bu kriterler bazı araçlar için çoklu kombinasyon şeklinde uygulanmıştır. Kriterleri içeren DataFrame Şekil 3.14. ve Şekil 3.15. de verilmiştir. DataFrame’deki “Durum” sütunu riskli anlamını ifade eden “1” olarak eklenmiştir.

df_tsosyal - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_ID_Inv	Information
49	37.7692	32.525	Sanayi	CM75	CM75_75	
114	37.8989	32.4926	Sanayi	CM10	CM10_10	
47	37.9073	32.4996	Sanayi	CM77	CM77_77	
59	37.9504	32.4971	Avm	CM65	CM65_65	
80	37.9146	32.532	Sanayi	CM44	CM44_44	
77	37.9191	32.514	Sanayi	CM47	CM47_47	
55	37.9521	32.5008	Park	CM69	CM69_69	
113	37.9491	32.5246	Avm	CM11	CM11_11	
74	37.9802	32.6106	Sanayi	CM50	CM50_50	
75	37.977	32.6424	Sanayi	CM49	CM49_49	
83	37.9486	32.4617	Park	CM41	CM41_41	
99	37.9089	32.509	Sanayi	CM25	CM25_25	
48	37.8575	32.4607	Sanayi	CM76	CM76_76	
46	37.9333	32.5465	Sanayi	CM78	CM78_78	
56	37.8296	32.4141	Park	CM68	CM68_68	
60	37.9054	32.4945	Avm	CM64	CM64_64	
45	37.9227	32.529	Sanayi	CM79	CM79_79	
78	37.9214	32.5507	Sanayi	CM46	CM46_46	
93	38.0053	32.5332	Kafe	CM31	CM31_31	
54	37.8097	32.5183	Park	CM70	CM70_70	
53	37.9572	32.5591	Park	CM71	CM71_71	
79	37.9437	32.5518	Sanayi	CM45	CM45_45	
58	37.8899	32.4931	Avm	CM66	CM66_66	
72	37.9249	32.4163	Kafe	CM52	CM52_52	
37	37.8493	32.4178	Kafe	CM87	CM87_87	
76	37.9683	32.6116	Sanayi	CM48	CM48_48	
38	37.8535	32.4209	Kafe	CM86	CM86_86	

Format Resize ☒ Background color ☒ Column min/max OK Cancel

Şekil 3.14. Riskli Araç Kriterlerinde Kullanılan Yer DataFrame

df_tehlike - DataFrame

Index	Latitude	Longitude	Description	Camera_ID	Camera_ID_Inv	Information
115	37.8753	32.4844	Silah_Satis	CM9	CM9_9	
116	37.8728	32.5004	Kimyasal	CM8	CM8_8	
94	38.0054	32.517	Bar	CM30	CM30_30	

Format Resize ☒ Background color ☒ Column min/max OK Cancel

Şekil 3.15. Riskli Araç Kriterlerinde Kullanılan Yer DataFrame

3.2.2.1. Sağlık çalışanlarına ait riskli araç hareketlerinin oluşturulması

Sağlık çalışanlarına ait riskli araç hareketleri, risksiz sürüşe ek olarak risk kriterlerinin her biri veya kombinasyonu rastgele seçilerek 250 değişik araca uygulanmıştır. Yazılım sonucunda 15000 farklı araç sürüşü üretilmiştir. Meslek sınıfı “0” olarak belirlenmiştir.

3.2.2.2. Okul çalışanlarına ait riskli araç hareketlerinin oluşturulması

Eğitim-Öğretim çalışanlarına ait riskli araç hareketleri, risksiz sürüşe ek olarak risk kriterlerinin her biri veya kombinasyonu rastgele seçilerek 540 bireysel araca uygulanmıştır. Yazılım sonucunda 32400 farklı araç sürüşü üretilmiştir. Meslek sınıfı “1” olarak belirlenmiştir.

3.2.2.3. Kolluk kuvveti çalışanlarına ait riskli araç hareketlerinin oluşturulması

Kolluk Kuvvetlerin çalışanlarına ait riskli araç hareketleri, risksiz sürüşe ek olarak risk kriterlerinin her biri veya kombinasyonu rastgele seçilerek 350 özel araca uygulanmıştır. Yazılım sonucunda 21000 farklı araç sürüşü üretilmiştir. Meslek sınıfı “2” olarak belirlenmiştir.

3.2.2.4. Diğer kamu çalışanlarına ait riskli araç hareketlerinin oluşturulması

Diğer Kamu çalışanlarına ait riskli araç hareketleri, risksiz sürüşe ek olarak risk kriterlerinin her biri veya kombinasyonu rastgele seçilerek 220 kişisel araca uygulanmıştır. Yazılım sonucunda 13200 farklı araç sürüşü üretilmiştir. Meslek sınıfı “3” olarak belirlenmiştir.

3.2.2.5. Öğrencilere ait riskli araç hareketlerinin oluşturulması

Öğrencilere ait riskli araç hareketleri, risksiz sürüşe ek olarak risk kriterlerinin her biri veya kombinasyonu rastgele seçilerek 75 eşsiz araca uygulanmıştır. 75 aracın tümü sadece yurt kullanmamıştır. Diğer meslek sınıfları gibi ikamet kullanabileceği göz önüne alınmıştır. Yazılım sonucunda 4500 farklı araç sürüşü üretilmiştir. Meslek sınıfı “4” olarak belirlenmiştir.

3.2.2.6. Esnaf ve işçilere ait riskli araç hareketlerinin oluşturulması

Esnaf ve işçilere ait araç hareketleri, risksiz sürüşe ek olarak risk kriterlerinin her biri veya kombinasyonu rastgele seçilerek 2100 değişik araca uygulanmıştır. Yazılım sonucunda 126000 farklı araç sürüşü üretilmiştir. Meslek sınıfı “5” olarak belirlenmiştir.

3.2.2.7. Pazarlama çalışanlarına ait riskli araç hareketlerinin oluşturulması

Pazarlamacılık mesleğine ait araç hareketleri, risksiz sürüşe ek olarak risk kriterlerinin her biri veya kombinasyonu rastgele seçilerek 295 farklı araca uygulanmıştır. Yazılım sonucunda 17700 farklı araç sürüşü üretilmiştir. Meslek sınıfı “6” olarak belirlenmiştir.

3.2.2.8. Emeklilere ait riskli araç hareketlerinin oluşturulması

Emeklilerin riskli araç hareketleri, risksiz sürüşe ek olarak risk kriterlerinin her biri veya kombinasyonu rastgele seçilerek 140 farklı araca uygulanmıştır. Yazılım sonucunda 8400 farklı araç sürüşü üretilmiştir. Meslek sınıfı “7” olarak belirlenmiştir.

3.3. Risksiz ve Riskli Verilerin Birleştirilmesi

Tez çalışmasının bu aşamasında, risksiz veri DataFrame’i ile riskli veri DataFrame’i alt alta eklenerek yeni bir veri kümesi oluşturulmuştur. Bu veri kümesi 476520 satır ve 137 sütundan oluşmaktadır.

Yeni veri kümesinde plaka, indeks, risksiz ve riskli verileri oluşturan etmenlerin kümedeki veriler ile uyumluluğu kontrol edilmiştir.

Oluşturulan risksiz her bir meslek sınıfının araç hareketlerinin sayısı aynı meslek sınıfının riskli araç hareketleri sayısına eşit olarak üretildiği bilinmektedir. Bu eşitliğin sağlanmasının amacı, verilerin büyük çoğunluğunun binary veri içerdiğinden birbirine oldukça benzemesidir. Yapay sinir ağındaki eğitimi sonucu oluşan modelin doğru tahminleme yapabilmesi üzere bu her meslek sınıfının riskli-risksiz veri sayılarının eşitliğini sağlama metodu oldukça önem taşıdığı farklı metotlarda uygulanan YSA eğitimleri sonucunda anlaşılmıştır.

Yapay sinir ağları eğitimi için “Plaka”, “Tarih”, “Gun” sütunları kaldırılarak Şekil 3.16. da gösterilen veri kümesi elde edilmiştir.

```

<bound method DataFrame.info of
0      0.0  0.0  0.0  0.0  ...      0.0      810.0      954.198112      0.0
1      0.0  0.0  0.0  0.0  ...      0.0      851.0      1017.113566      0.0
2      0.0  0.0  0.0  0.0  ...      0.0      810.0      877.329707      0.0
3      0.0  0.0  0.0  0.0  ...      0.0      666.0      738.715326      0.0
4      0.0  0.0  0.0  0.0  ...      0.0      810.0      923.665706      0.0
...      ...      ...      ...      ...      ...      ...      ...      ...
476515  0.0  0.0  0.0  0.0  ...      7.0      916.0      1032.644287      1.0
476516  0.0  0.0  0.0  0.0  ...      7.0      343.0      288.267355      1.0
476517  0.0  0.0  0.0  0.0  ...      7.0      512.0      485.547595      1.0
476518  0.0  0.0  0.0  0.0  ...      7.0      835.0      798.048932      1.0
476519  0.0  0.0  0.0  0.0  ...      7.0      834.0      799.605177      1.0

[476520 rows x 134 columns]>

```

Şekil 3.16. Oluşturulan Veri Kümesi

3.4. Yapay Sinir Ağlarının Eğitilmesi ve Oluşturulan Modelin Test Edilmesi

Bir önceki adımda oluşturulan yapay sinir ağına girmeye hazır DataFrame'in her bir dizisinin "Durum" sütuna unique() fonksiyonu ile benzersiz özelliği kazandırılmıştır. 2 Boyutlu araç hareket verilerinin X boyutu "Durum" sütunu silinmiş ve Y boyutu "Durum" sütunu olmak üzere ayrılmıştır. Eğitim ve test verileri aşağıda verilen eşitlikle ayrılmış, 4 farklı değişkene atanmıştır.

```
X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.3 ,random_state=42)
```

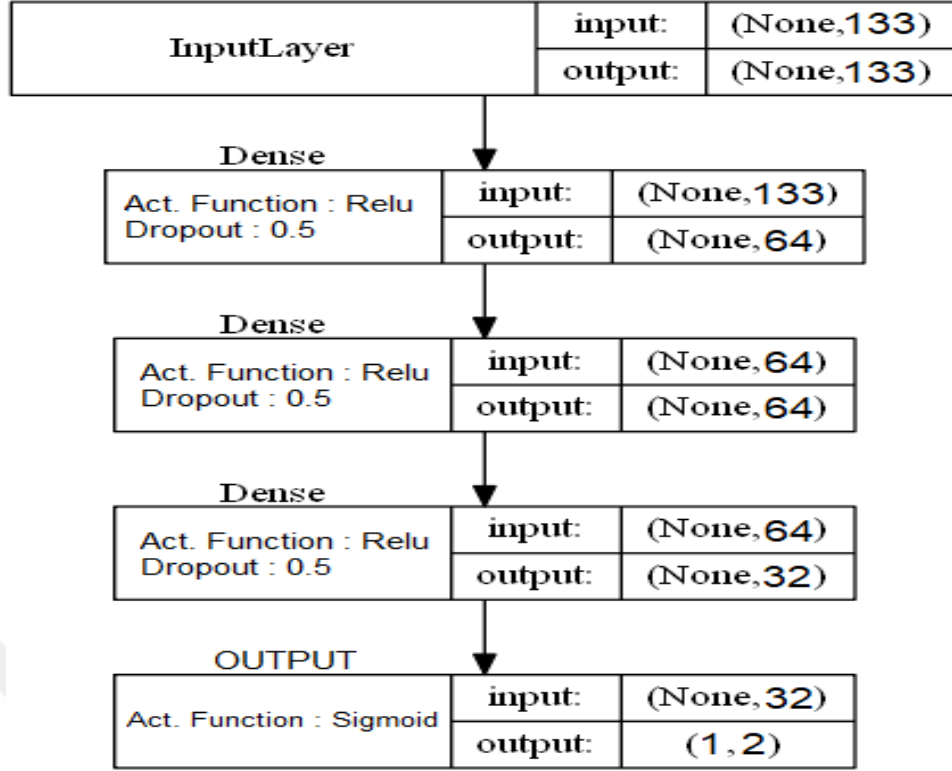
Değişkenlerin veri tipleri;

```

y_train = np.array(y_train).astype(int)
X_train = np.array(X_train).astype(float)
y_test = np.array(y_test).astype(int)
X_test = np.array(X_test).astype(float) olarak düzenlenmiştir.

```

Yapay sinir ağı modeli, Input Layer, 3 hidden layer ve Output layer olarak belirlenmiştir. Şekil 3.17. de ağa ait detaylı özellikler görülmektedir.



Şekil 3.17. Yapay Sinir Ağı Modeli

Yapay sinir ağını daha düzenli hale getirmek için varsayılan olarak tanımlanan BatchNormalization fonksiyonu kullanılmıştır. BatchNormalization fonksiyonun kullanılmasıyla, yapay sinir ağına eğitimi sırasında yok olma gradyanına bir direnç kazandırılmıştır. Böylece ağı eğitme süresinin optimum düzeye gelmesi ve modelin performans artışı amaçlanmıştır.

0.0-0.5 ve 0.5-1.0 aralığındaki değerlerde istenilen sonuç alınamamıştı, 'Dropout' değeri 0.5 alınmış, aşırı öğrenmenin önüne geçilmiştir.

Optimizör olarak Adam algoritması seçilmiştir. Stokastik gradyan iniş yöntemi olan Adam optimizasyonu değişkenlerinden, momentum '*beta_1*', '*beta_2*', '*epsilon*' değiştirilmemiş, sırasıyla 0.9, 0.999, 1e-7 olarak kullanılmıştır. Öğrenme oranı değişkeni '*learning_rate*', 0.001 seçilmiştir [67].

Yapay sinir ağı modelinin derlenmesinde optimizasyon fonksiyonu '*rmsprop*', kayıp fonksiyonu '*binary_crossentropy*' ve modelin performansını değerlendiren metrics parametresi '*accuracy*' olarak belirlenmiştir. Aşağıda verilen eşitlikte bu parametrelerinin python dilindeki yazımı gösterilmiştir.

```
model.compile(optimizer='rmsprop', loss='binary_crossentropy', metrics=['accuracy'])
```

Derlenen yapay sinir ağı mimarisi fit() fonksiyonu ile analiz etme sayısı kadar altta verilen kodun çalışmasıyla eğitime başlar.

```
history=model.fit(X_train_scaled, y_train, batch_size=64, verbose=1, epochs=20)
```

Yapay sinir ağındaki verilerin kodlanması one-hot tekniğinde olduğundan test ve gerçek sonuç çıktıları aşağıdaki gibi olmuştur.

Risksiz aracı, “Durum = 0” ifade eden kodlanmış veri [1 0] ‘dir.

Potansiyel riskli aracı, “Durum = 1” ifade eden kodlanmış veri [0 1] ‘dir.

Eğitilen yapay sinir ağı modeli, “model.save_weights('dosyaismi.h5')” komutu ile kaydedilerek, tez çalışmasının diğer yazılım modüllerinde gerekli kütüphaneler çağrılarak kullanılması sağlanmıştır.

3.5. Raspberry Pi 3 ile Karakter Tanıma ve Temsili Plaka Okuma

Raspberry Pi 3 model B’yi oluşturan bileşenler aşağıda yer almaktadır.

- Dört Çekirdekli 1.2GHz Broadcom BCM2837 64bit CPU
- 1 GB RAM
- Kart üzerinde BCM43438 kablosuz LAN ve Bluetooth Düşük Enerji (BLE)
- 100 Base Ethernet
- 40 pimli genişletilmiş GPIO
- 4 USB 2 bağlantı noktası
- 4 Kutuplu stereo çıkış ve kompozit video bağlantı noktası
- Tam boyutlu HDMI
- Raspberry Pi kamera bağlamak için CSI kamera bağlantı noktası
- Raspberry Pi dokunmatik ekran bağlamak için DSI ekran bağlantı noktası
- İşletim sistemini yüklemek ve verileri depolamak için Micro SD bağlantı noktası
- Mikro USB güç kaynağı portu [68].

olarak gelmektedir. OpenCV kütüphanesinin kurulumu aşağıda komutlarla gerçekleştirilmiştir.

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get install \
build-essential cmake unzip pkg-config gfortran \
libjpeg-dev libpng-dev libtiff-dev \
libavcodec-dev libavformat-dev libswscale-dev libv4l-dev \
libxvidcore-dev libx264-dev \
libhdf5-dev libhdf5-serial-dev \
libgtk-3-dev \
libatlas-base-dev libblas-dev liblapack-dev \
python3-dev \

$ pip3 install numpy
$ cd ~
$ wget -O opencv.zip https://github.com/Itseez/opencv/archive/3.2.0.zip
$ wget -O opencv_contrib.zip
https://github.com/Itseez/opencv_contrib/archive/3.2.0.zip
$ unzip opencv.zip
$ unzip opencv_contrib.zip

$ cd ~/opencv-3.2.0/
$ mkdir build
$ cd build
$ cmake -D CMAKE_BUILD_TYPE=RELEASE \
-D CMAKE_INSTALL_PREFIX=/usr/local \
-D OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib-4.1.2/modules \
-D BUILD_EXAMPLES=ON ..

make -j4

$ sudo make install
$ sudo ldconfig

$ python3 #kurulum_kontrol
>>> import cv2
>>> cv2.__version__
3.2.0
```

Tesseract kütüphanesinin kurulumunu sağlayan komutlar sırasıyla aşağıda yer almaktadır.

```
$sudo apt-get update
$sudo apt-get upgrade
```

```
$sudo apt install libatlas3-base libsz2 libharfbuzz0b libtiff5 libjasper1 $libilmbase12
libopenexr22 libilmbase12 libgstreamer1.0-0 libavcodec57 $libavformat57
libavutil55 libswscale4 libqtgui4 libqt4-test libqtcore4
```

```
$sudo pip3 install opencv-contrib-python libwebp6
```

```
$sudo apt install tesseract-ocr
```

```
$sudo apt install libtesseract-dev
```

```
$sudo pip install pytesseract
```

3.5.2. Python diliyle karakter tanıma ve temsili plaka okuma yazılımı

Çalışmanın bu adımında Raspberry Pi ve V2 kamera modülü Kullanılarak RGB 640x480 çözünürlüğünde, 30fps görüntü alınmıştır.

Her bir görüntü tesseract kütüphanesinin bir fonksiyonu olan `image_to_string`'e girdi olarak verilmiştir.

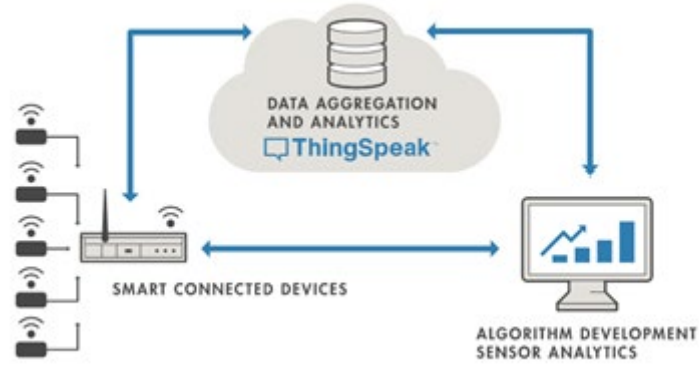
`Image_to_string` fonksiyonun döndürdüğü veri, tamsayı dönüşümü için bir karar mekanizmasına gönderilmiştir. Tam sayıya dönüşebilen veri gönderilmeye hazır olmuştur. Dönüşemeyen veri için ise yeni görüntü verisi beklenmektedir.



Şekil 3.20. Temsili Plaka

3.6. Plaka Bilgisinin Raspberry Pi 3 ve Bilgisayar Arasında Alışverişi

ThingSpeak, buluttaki canlı veri akışlarını toplanılmasına, verilerin görselleştirilmesine, verilerin analiz edilmesine olanak veren IoT analiz platform hizmetidir. IoT, internete bağlı olan cihazların insanlarla ve diğer nesnelerle iletişim kurmasını imkan vererek verilerin işlendiği ve analiz edildiği bulut depolama ve bulut bilişim kaynaklarına sensör verileri sağlar. ThingSpeak, Cihazlar tarafından gönderilen verilerin anında görselleştirilmesini sağlar. Verilerin geldiği anda çevrim içi analizi ve verilerin işlenmesi ise MATLAB ile gerçekleştirilir. Şekil 3.21. de yer alan diyagramda IoT sistemi açıklanmaya çalışılmıştır.



Şekil 3.21. ThingSpeak Çalışma Prensipleri

Şekil 3.21. de Sol tarafta yer alan ağın kenarında yaşayan akıllı cihazlar (IoT'deki "nesneler") veri toplamaktadır. Ortada, birçok kaynaktan gelen verilerin, genellikle bu amaç için tasarlanmış bir IoT analitik platformu tarafından eş zamanlı olarak toplandığı ve analiz edildiği bulut bulunmaktadır. ThingSpeak, diyagramın bulut kısmına sığar ve internete bağlı sensörlerden verileri hızlı bir şekilde toplamak ve analiz etmek için bir platform sağlar. Diyagramın sağ tarafı, IoT uygulamasıyla ilişkili algoritma gelişimini göstermektedir. Burada veriler üzerinde geçmiş analizler yapılarak toplanan veriler hakkında fikir edinilmeye çalışılır. Son olarak, bulutta veya akıllı cihazda çalışabilecek algoritmaların prototiplemesini sağlamak için veriler IoT platformundan bir masaüstü yazılım ortamına çekilir. Bir IoT sistemi tüm bu unsurları içerir [70]. Tez çalışmasında bu platform tespit edilen temsili plaka verilerinin aktarılması için kullanılmıştır.

3.6.1. Plaka bilgisinin bilgisayara gönderilmesi

Başarılı şekilde dönüştürülen veri bilgisayara gönderime hazırdır. Gönderim IoT teknolojisinde veri aktarımı ve veri analizi sağlayan ThingSpeak üzerinden gerçekleşmiştir. Plaka bilgisini tutması için ThingSpeak üzerinden “rpi2computer_number” adlı kanal oluşturulmuştur. Kanala ait veriyi ThingSpeak’e aktarmaya olanak sağlayan API Key’i ThingSpeak API Keys kısmından alınmıştır. Şekil 3.22. de veri göndermeyi sağlayan API Keys sekmesi ve kişisel API KEY sayfa bölümü verilmiştir.

rpizcomputer_number

Channel ID: 1311320

Author: ngneer

Access: Private

number of the read rpi

Private View

Public View

Channel Settings

Sharing

API Keys

Data Import / Export

Write API Key

Key

VGJQ4EVDC56

Generate New Write API Key

Şekil 3.22. API Keys Sekmesi ve Kişisel API Key

Alınan Key'in kullanılabilmesi ve Raspberry'nin ThingSpeak ile bağlantı kurabilmesi için "urllib" kütüphanesinin fonksiyonları çağırılmıştır. ThingSpeak'e veri gönderim kodu aşağıda verilmiştir.

```
import urllib #kutuphane cagrilmasi
import urllib.request

myAPI = 'VGJQ4EVDC56****' #API KEY tanimi
baseURL = 'https://api.thingspeak.com/update?api_key=%s' % myAPI #ana link
tanımı
textont = "95N404" #rastgele veri
conn = urllib.request.urlopen(baseURL + '&field1=%s&field2=%s' % (textont,0)) #
veri gönderme komutu
conn.close() # haberleşmeyi kapat
```

3.6.2. Gönderilen plaka bilgisinin bilgisayardan okunması

Gönderilen verinin okunabilmesi için ThingSpeak API Keys bölümünden aynı kanala ait okuma API Key'i alınmıştır. Şekil 3.23. de Okuma API Key'i görülmektedir.

Read API Keys

Key

HY4X0PIU01W7

Note

Save Note

Delete API Key

Add New Read API Key

Şekil 3.23. Veri Okuma API Key'i

ThingSpeak'ten veri okuma işlemi aşağıda verilen yazılım kodu sayesinde başarılı şekilde okunmuştur.

```
import urllib.request
import json
import time
while True:
    TS=urllib.request.urlopen("https://api.thingspeak.com/channels/1311320/feeds.json?api_key=HY4X0PIU01W7****&results=2")
    response = TS.read()
    data=json.loads(response)
    b = data['feeds'][1]
    c = b['field1']
    print (c)
    time.sleep(1)
    TS.close()
```

3.6.3. Veri gönderim yazılımı ve temsili plaka okuma yazılımının entegre hale getirilmesi

Anlamli verinin ThingSpeak'e gönderilmesini sağlayan gerekli kütüphaneler temsili plaka tanıma yazılımına eklenmiştir.

“myAPI” ve “baseURL” ifadeleri döngü dışında tanımlanmıştır.

Alınan her birim zamandaki anlamli görüntünün tesseract kütüphanesinin image_to_string fonksiyonuna girmesi sonucu alınan çıktı tamsayısı “textont” değişkenine atanmıştır.

“conn = urllib.request.urlopen(baseURL + '&field1=%s&field2=%s' % (textont,0))” komutu ile veri gönderimi sağlanmış, ardından “conn.close()” komutu ile haberleşme durdurulmuştur. Entegre edilmiş yazılım kodları EK-1’ de verilmiştir.

3.7. YSA Modeliyle Öneri Yazılımının Yapılması ve ThingSpeak İletişim Entegrasyonu

Her bir aracın DataFrame’de toplam 60 hareketi bulunmaktadır. 30 günün her bir günü için gidiş ve dönüş olarak düşünülmüştür.

Riskli kategoride bulunan her bir aracın 60 hareketi de risk içermeyebilir, aynı şekilde risksiz kategoride bulunan her bir aracın tüm hareketi risksiz olmayabilir. Bu doğrultuda YSA modeline temsili okunan plakaya ait tüm hareketlerin tahmini alınması planlanmıştır.

Spider IDE’ de (Tümleşik Geliştirme Ortamı) YSA modeli çağrılarak rastgele üretilen temsili gelen plakanın meslek bilgisinin DataFrame’den çekilmesi, toplam riskli hareketin sayısı, riskli araç hareketinin aynı plakaya ait tüm hareketine oranla yüzdesi işlemlerinin hesaplanması ve konsol ekranına yazılması ile ilgili kod EK-2’de belirtilmiştir.

ThingSpeak’le iletişim için yukarıdaki yazılıma gerekli kodlar eklenmiş yazılım EK 3’te verilmiştir.

3.8. Bilgisayar Kullanıcı Arayüzü Hazırlanması

Kullanıcı arayüzü, Spider IDE üzerinde Tkinter kütüphanesi kullanılarak hazırlanmıştır.

Arayüz yazılımı, okunan plaka bilgisi, araç kullanıcısının muhtemel mesleği, üç farklı eylem tavsiyesi ve bunun trafikte kullanılan kırmızı-sarı-yeşil renkleriyle ifade edilmesi, yazılımı duraklatıp istenildiği zaman kaldığı yerden devam etmesi düşünülerek oluşturulmuştur.

Arayüz üzerinde Label, Buton, Textbox, Listbox elemanları kullanılmıştır. Arayüze ait yazılım EK-4’te verilmiştir.

4. ARAŞTIRMA SONUÇLARI VE TARTIŞMA

4.1. Risksiz Araç Hareket Sonuçları

4.1.1. Sağlık çalışanlarına ait sonuçlar

Sağlık çalışanlarının araç sürüşlerine ait üretilen verilerin son 13'ü 4.1. de yer almaktadır.

Index	H_jci	H_sonu	Sabah	Ogle	Aksam	Diger_Yakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
14987	1	0	0	0	1	0	0	705	734.446	0	Thursday	2019-01-24 06:51:49
14988	1	0	1	0	0	0	0	800	904.911	0	Friday	2019-01-24 18:48:52
14989	1	0	0	0	1	0	0	1258	1349.4	0	Friday	2019-01-25 06:26:46
14990	0	1	0	0	1	0	0	254	301.221	0	Saturday	2019-01-25 17:19:10
14991	0	1	0	0	1	0	0	357	407.147	0	Saturday	2019-01-26 21:42:01
14992	0	1	0	0	1	0	0	1187	1288.77	0	Sunday	2019-01-27 17:48:29
14993	0	1	0	0	1	0	0	1346	1459.43	0	Sunday	2019-01-27 18:59:55
14994	1	0	1	0	0	0	0	800	848.358	0	Monday	2019-01-28 06:30:44
14995	1	0	0	0	1	0	0	1451	1739.02	0	Monday	2019-01-28 18:24:24
14996	1	0	1	0	0	0	0	800	873.069	0	Tuesday	2019-01-29 08:35:19
14997	1	0	0	0	1	0	0	1348	1486.64	0	Tuesday	2019-01-29 18:00:03
14998	1	0	1	0	0	0	0	800	886.541	0	Wednesday	2019-01-30 06:55:42
14999	1	0	0	0	1	0	0	705	837.11	0	Wednesday	2019-01-30 18:19:26
15000	0	0	0	0	0	0	0	0	0	0	0	0

Şekil 4.1. Sağlık Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü

4.1.2. Okul çalışanlarına ait sonuçlar

Okul çalışanlarının araç sürüşlerine ait üretilen son 13 veri Şekil 4.2. de yer almaktadır.

Index	H_jci	H_sonu	Sabah	Ogle	Aksam	Diger_Yakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
47387	1	0	0	0	1	0	1	1150	1228.86	0	Thursday	2019-01-24 18:09:08
47388	1	0	1	0	0	0	1	658	670.332	0	Friday	2019-01-25 08:04:47
47389	1	0	0	0	1	0	1	2407	2595.48	0	Friday	2019-01-25 17:38:48
47390	0	1	0	0	1	0	1	603	604.293	0	Saturday	2019-01-26 16:01:07
47391	0	1	0	0	1	0	1	587	607.465	0	Saturday	2019-01-26 17:54:48
47392	0	1	1	0	0	0	1	1854	2055.16	0	Sunday	2019-01-27 10:41:28
47393	0	1	0	1	0	0	1	1828	1831.1	0	Sunday	2019-01-27 12:38:47
47394	1	0	1	0	0	0	1	658	719.675	0	Monday	2019-01-28 07:49:49
47395	1	0	0	0	1	0	1	2407	2673.81	0	Monday	2019-01-28 18:54:23
47396	1	0	1	0	0	0	1	658	725.237	0	Tuesday	2019-01-29 07:19:09
47397	1	0	0	0	1	0	1	685	701.89	0	Tuesday	2019-01-29 17:18:03
47398	1	0	1	0	0	0	1	658	724.86	0	Wednesday	2019-01-30 08:54:54
47399	1	0	0	0	1	0	1	685	767.444	0	Wednesday	2019-01-30 17:53:16

Şekil 4.2. Okul Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü

4.1.3. Kolluk kuvveti çalışanlarına ait sonuçlar

Kolluk kuvveti çalışanlarının araç sürüşlerine ait üretilen verilerin bir kısmı Şekil 4.3. de yer almaktadır.

df_test - DataFrame

Index	H_ici	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
68386	1	0	1	0	0	0	2	1131	1350.03	0	Thursday	2019-01-24 07:07:55
68387	1	0	0	0	1	0	2	1268	1424.67	0	Thursday	2019-01-24 18:44:06
68388	1	0	1	0	0	0	2	1131	1282.26	0	Friday	2019-01-25 08:52:55
68389	1	0	0	0	1	0	2	1268	1403.96	0	Friday	2019-01-25 18:43:13
68390	0	1	0	0	1	0	2	1664	1753.4	0	Saturday	2019-01-26 16:10:31
68391	0	1	0	0	1	0	2	1267	1446.98	0	Saturday	2019-01-26 20:47:37
68392	0	1	0	0	1	0	2	1491	1608.55	0	Sunday	2019-01-27 16:11:34
68393	0	1	0	0	1	0	2	1379	1647.31	0	Sunday	2019-01-27 18:08:42
68394	1	0	1	0	0	0	2	1131	1320.24	0	Monday	2019-01-28 08:36:34
68395	1	0	0	0	1	0	2	1588	1615.37	0	Monday	2019-01-28 18:43:36
68396	1	0	1	0	0	0	2	1131	1301.47	0	Tuesday	2019-01-29 07:05:49
68397	1	0	0	0	1	0	2	1268	1413.78	0	Tuesday	2019-01-29 17:54:26
68398	1	0	1	0	0	0	2	1131	1301.68	0	Wednesday	2019-01-30 08:34:38
68399	1	0	0	0	1	0	2	1268	1351.03	0	Wednesday	2019-01-30 17:41:10

Format Resize Background color Column min/max OK Cancel

Şekil 4.3. Kolluk Kuvveti Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü

4.1.4. Diğer kamu çalışanlarına ait sonuçlar

Diğer kamu çalışanlarının araç sürüşlerine ait üretilen verilerin bir bölümü Şekil 4.4. de yer almaktadır.

df_test - DataFrame

Index	H_ici	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
81647	1	0	0	0	1	0	3	1017	1204.36	0	Thursday	2019-01-24 06:52:00
81648	1	0	1	0	0	0	3	687	795.084	0	Friday	2019-01-24 17:15:41
81649	1	0	0	0	1	0	3	839	952.083	0	Friday	2019-01-25 08:01:05
81650	0	1	1	0	0	0	3	2661	3042.29	0	Friday	2019-01-25 17:18:26
81651	0	1	0	1	0	0	3	2157	2225.68	0	Saturday	2019-01-26 09:16:49
81652	0	1	0	1	0	0	3	1989	2149.74	0	Saturday	2019-01-26 12:09:40
81653	0	1	0	0	1	0	3	2632	3079.4	0	Sunday	2019-01-27 15:16:08
81654	1	0	1	0	0	0	3	687	738.146	0	Sunday	2019-01-27 19:35:37
81655	1	0	0	0	1	0	3	839	926.85	0	Monday	2019-01-28 07:20:37
81656	1	0	1	0	0	0	3	839	926.85	0	Monday	2019-01-28 17:42:07
81657	1	0	0	0	1	0	3	687	739.721	0	Tuesday	2019-01-29 08:04:08
81658	1	0	0	0	1	0	3	839	840.068	0	Tuesday	2019-01-29 18:45:49
81659	1	0	1	0	0	0	3	687	712.824	0	Wednesday	2019-01-30 07:22:25
81659	1	0	0	0	1	0	3	839	898.952	0	Wednesday	2019-01-30 18:35:17

Format Resize Background color Column min/max OK Cancel

Şekil 4.4. Diğer Kamu Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü

4.1.5. Öğrencilere ait sonuçlar

Öğrencilerin araç sürüşlerine ait üretilen verilerin son 14'ü Şekil 4.5. de yer almaktadır.

df_test - DataFrame

Index	H_jci	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
86146	1	0	1	0	0	0	4	1664	1830.83	0	Thursday	2019-01-24 07:00:19
86147	1	0	0	0	1	0	4	1534	1679.08	0	Thursday	2019-01-24 18:59:40
86148	1	0	1	0	0	0	4	1664	1809.18	0	Friday	2019-01-25 08:36:12
86149	1	0	0	0	1	0	4	1534	1797.05	0	Friday	2019-01-25 18:07:18
86150	0	1	1	0	0	0	4	1403	1445.55	0	Saturday	2019-01-26 10:23:14
86151	0	1	0	1	0	0	4	1416	1511.06	0	Saturday	2019-01-26 13:11:53
86152	0	1	0	1	0	0	4	1278	1464.26	0	Sunday	2019-01-27 14:14:36
86153	0	1	0	0	1	0	4	1390	1554.82	0	Sunday	2019-01-27 17:42:26
86154	1	0	1	0	0	0	4	1664	1710.77	0	Monday	2019-01-28 07:47:19
86155	1	0	0	0	1	0	4	1721	1826.51	0	Monday	2019-01-28 18:38:05
86156	1	0	1	0	0	0	4	1664	1705.22	0	Tuesday	2019-01-29 06:34:16
86157	1	0	0	0	1	0	4	2148	2431.3	0	Tuesday	2019-01-29 17:43:27
86158	1	0	1	0	0	0	4	1664	1807.33	0	Wednesday	2019-01-30 06:43:23
86159	1	0	0	0	1	0	4	1534	1781.88	0	Wednesday	2019-01-30 19:17:11

Format Resize Background color Column min/max OK Cancel

Şekil 4.5. Öğrencilere Ait Üretilen Risksiz Araç Sürüşü

4.1.6. Esnaf ve işçilere ait sonuçlar

Esnaf ve işçilerin araç sürüşlerine ait üretilen verilerin bir kısmı 4.6. da verilmiştir.

df_test - DataFrame

Index	H_jci	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
212146	1	0	1	0	0	0	5	616	692.804	0	Thursday	2019-01-24 06:52:27
212147	1	0	0	0	1	0	5	1345	1448.99	0	Thursday	2019-01-24 17:36:19
212148	1	0	1	0	0	0	5	616	632.996	0	Friday	2019-01-25 06:20:26
212149	1	0	0	0	1	0	5	517	538.718	0	Friday	2019-01-25 17:18:30
212150	0	1	0	1	0	0	5	1725	2027.42	0	Saturday	2019-01-26 13:16:51
212151	0	1	0	1	0	0	5	1751	1957.24	0	Saturday	2019-01-26 14:57:14
212152	0	1	0	0	1	0	5	1246	1425.57	0	Sunday	2019-01-27 16:47:35
212153	0	1	0	0	1	0	5	1169	1299.88	0	Sunday	2019-01-27 21:52:17
212154	1	0	1	0	0	0	5	616	647.864	0	Monday	2019-01-28 06:07:39
212155	1	0	0	0	1	0	5	1498	1669.58	0	Monday	2019-01-28 18:50:09
212156	1	0	1	0	0	0	5	616	647.733	0	Tuesday	2019-01-29 08:39:46
212157	1	0	0	0	1	0	5	1345	1536.79	0	Tuesday	2019-01-29 18:32:41
212158	1	0	1	0	0	0	5	616	646.811	0	Wednesday	2019-01-30 07:48:23
212159	1	0	0	0	1	0	5	517	529.278	0	Wednesday	2019-01-30 17:55:05

Format Resize Background color Column min/max OK Cancel

Şekil 4.6. Esnaf ve İşçilere Ait Üretilen Risksiz Araç Sürüşü

4.1.7. Pazarlama çalışanlarına ait sonuçlar

Pazarlama çalışanlarının araç sürüşlerine ait üretilen veriler 4.7. de verilmiştir.

df_test - DataFrame

Index	H_ici	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
229846	1	0	1	0	0	0	6	1310	1546.91	0	Thursday	2019-01-24 06:26:11
229847	1	0	0	0	1	0	6	1795	1972.46	0	Thursday	2019-01-24 17:58:20
229848	1	0	1	0	0	0	6	1310	1494.43	0	Friday	2019-01-25 06:34:01
229849	1	0	0	0	1	0	6	1795	2018.15	0	Friday	2019-01-25 18:56:46
229850	0	1	1	0	0	0	6	1725	1868.57	0	Saturday	2019-01-26 10:29:47
229851	0	1	0	1	0	0	6	1751	1758.73	0	Saturday	2019-01-26 13:13:18
229852	0	1	0	1	0	0	6	757	898.82	0	Sunday	2019-01-27 14:03:44
229853	0	1	0	0	1	0	6	751	753.406	0	Sunday	2019-01-27 16:19:09
229854	1	0	1	0	0	0	6	1310	1363.09	0	Monday	2019-01-28 08:29:36
229855	1	0	0	0	1	0	6	1793	2117	0	Monday	2019-01-28 17:20:28
229856	1	0	1	0	0	0	6	1310	1510.59	0	Tuesday	2019-01-29 07:00:08
229857	1	0	0	0	1	0	6	1706	1747.03	0	Tuesday	2019-01-29 18:11:34
229858	1	0	1	0	0	0	6	1310	1556.66	0	Wednesday	2019-01-30 07:23:05
229859	1	0	0	0	1	0	6	1388	1654.42	0	Wednesday	2019-01-30 19:52:57

Format Resize Background color Column min/max OK Cancel

Şekil 4.7. Pazarlama Çalışanlarına Ait Üretilen Risksiz Araç Sürüşü

4.1.8. Emeklilere ait sonuçlar

Araç sürüşlerine ait üretilen veriler 4.8. de yer almaktadır.

df_test - DataFrame

Index	H_ici	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
238247	1	0	0	0	1	0	7	1042	1166.67	0	Thursday	2019-01-24 18:46:40
238248	1	0	0	1	0	0	7	1676	1752.02	0	Friday	2019-01-25 12:05:56
238249	1	0	0	0	1	0	7	1984	2379.74	0	Friday	2019-01-25 18:12:17
238250	0	1	0	1	0	0	7	1471	1534.36	0	Saturday	2019-01-26 11:30:56
238251	0	1	0	1	0	0	7	1516	1651.05	0	Saturday	2019-01-26 15:15:35
238252	0	1	0	1	0	0	7	615	629.768	0	Sunday	2019-01-27 12:53:32
238253	0	1	0	0	1	0	7	554	648.674	0	Sunday	2019-01-27 16:14:56
238254	1	0	0	1	0	0	7	678	760.285	0	Monday	2019-01-28 15:41:20
238255	1	0	0	0	1	0	7	498	567.595	0	Monday	2019-01-28 19:30:07
238256	1	0	0	1	0	0	7	969	985.364	0	Tuesday	2019-01-29 15:24:28
238257	1	0	0	0	1	0	7	1093	1174.27	0	Tuesday	2019-01-29 17:57:00
238258	1	0	0	1	0	0	7	555	613.732	0	Wednesday	2019-01-30 15:35:16
238259	1	0	0	0	1	0	7	1876	2140.52	0	Wednesday	2019-01-30 20:07:06

Format Resize Background color Column min/max OK Cancel

Şekil 4.8. Emeklilere Ait Üretilen Risksiz Araç Sürüşü

4.2. Riskli Araç Hareket Sonuçları

4.2.1. Sağlık çalışanlarına ait sonuçlar

Sağlık çalışanlarının araç sürüşlerine ait üretilen veriler Şekil 4.9. da yer almaktadır.

df_main - DataFrame

Index	H_ici	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
14987	1	0	0	0	1	0	0	430	418.219	1	Thursday	2019-01-24 18:38:18
14988	1	0	1	0	0	0	0	560	633.987	1	Friday	2019-01-25 06:15:49
14989	1	0	0	0	0	1	0	430	416.573	1	Friday	2019-01-25 23:22:41
14990	0	1	0	0	1	0	0	1544	1590.54	1	Saturday	2019-01-26 19:46:23
14991	0	1	0	0	1	0	0	1763	1893.74	1	Saturday	2019-01-26 21:32:56
14992	0	1	0	1	0	0	0	1625	1645.74	1	Sunday	2019-01-27 11:32:09
14993	0	1	0	1	0	0	0	1737	2073.91	1	Sunday	2019-01-27 14:54:23
14994	1	0	1	0	0	0	0	560	636.165	1	Monday	2019-01-28 07:52:48
14995	1	0	0	0	0	1	0	3025	3039.2	1	Monday	2019-01-28 20:15:34
14996	1	0	1	0	0	0	0	560	591.924	1	Tuesday	2019-01-29 08:12:29
14997	1	0	0	0	1	0	0	430	370.116	1	Tuesday	2019-01-29 18:11:10
14998	1	0	1	0	0	0	0	560	580.979	1	Wednesday	2019-01-30 07:50:16
14999	1	0	0	0	1	0	0	430	374.031	1	Wednesday	2019-01-30 18:45:58

Format Resize Background color Column min/max OK Cancel

Şekil 4.9. Sağlık Çalışanlarına Ait Riskli Araç Sürüşü

4.2.2. Okul çalışanlarına ait sonuçlar

Okul çalışanlarının araç sürüşlerine ait üretilen veriler Şekil 4.10. da yer almaktadır.

df_main - DataFrame

Index	H_ici	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
47386	1	0	1	0	0	0	1	1345	1186.04	1	Thursday	2019-01-24 07:37:55
47387	1	0	0	0	0	1	1	1481	1224.19	1	Thursday	2019-01-24 00:27:06
47388	1	0	1	0	0	0	1	1345	1183.31	1	Friday	2019-01-25 06:48:27
47389	1	0	0	0	0	1	1	1481	1234.64	1	Friday	2019-01-25 23:27:10
47390	0	1	0	0	0	1	1	1700	1822.36	1	Saturday	2019-01-26 00:27:24
47391	0	1	0	0	0	1	1	1767	1542.51	1	Saturday	2019-01-26 04:41:15
47392	0	1	0	0	0	1	1	1818	1985.11	1	Sunday	2019-01-27 21:10:18
47393	0	1	0	0	1	0	1	1868	1685	1	Sunday	2019-01-27 22:03:46
47394	1	0	1	0	0	0	1	1345	1160.11	1	Monday	2019-01-28 08:26:54
47395	1	0	0	0	1	0	1	1481	1230.16	1	Monday	2019-01-28 18:37:26
47396	1	0	1	0	0	0	1	1345	1096.96	1	Tuesday	2019-01-29 06:41:43
47397	1	0	0	0	0	1	1	1481	1458.74	1	Tuesday	2019-01-29 20:54:33
47398	1	0	1	0	0	0	1	1345	1162.13	1	Wednesday	2019-01-30 08:58:08
47399	1	0	0	0	1	0	1	1481	1387.5	1	Wednesday	2019-01-30 17:35:12

Format Resize Background color Column min/max OK Cancel

Şekil 4.10. Okul Çalışanlarına Ait Riskli Araç Sürüşü

4.2.3. Kolluk kuvveti çalışanlarına ait sonuçlar

Kolluk kuvveti çalışanlarının araç sürüşlerine ait üretilen veriler Şekil 4.11. de yer almaktadır.

df_main - DataFrame

Index	H_ici	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
68386	1	0	1	0	0	0	2	2164	2004.18	1	Thursday	2019-01-24 07:51:25
68387	1	0	0	0	1	0	2	3663	3709.34	1	Thursday	2019-01-24 18:38:52
68388	1	0	1	0	0	0	2	2164	1888.18	1	Friday	2019-01-25 07:29:44
68389	1	0	0	0	0	1	2	3492	3943.75	1	Friday	2019-01-25 23:37:21
68390	0	1	1	0	0	0	2	1601	1640.73	1	Saturday	2019-01-26 09:38:22
68391	0	1	0	1	0	0	2	2119	2174.09	1	Saturday	2019-01-26 11:25:24
68392	0	1	0	0	0	1	2	614	718.532	1	Sunday	2019-01-27 00:22:36
68393	0	1	0	0	0	1	2	499	588.99	1	Sunday	2019-01-27 05:04:46
68394	1	0	1	0	0	0	2	2164	2110.92	1	Monday	2019-01-28 08:45:52
68395	1	0	0	0	0	1	2	3152	3644.16	1	Monday	2019-01-28 20:46:50
68396	1	0	1	0	0	0	2	2164	1904.66	1	Tuesday	2019-01-29 07:44:14
68397	1	0	0	0	0	1	2	4313	4598.75	1	Tuesday	2019-01-29 20:23:55
68398	1	0	1	0	0	0	2	2164	1972.71	1	Wednesday	2019-01-30 07:22:20
68399	1	0	0	0	1	0	2	3152	3351.81	1	Wednesday	2019-01-30 17:13:09

Format Resize Background color Column min/max OK Cancel

Şekil 4.11. Kolluk Kuvveti Çalışanlarına Ait Riskli Araç Sürüşü

4.2.4. Diğer kamu çalışanlarına ait sonuçlar

Diğer kamu çalışanlarının araç sürüşlerine ait üretilen veriler Şekil 4.12. de yer almaktadır.

df_main - DataFrame

Index	H_ici	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
81646	1	0	1	0	0	0	3	408	331.987	1	Thursday	2019-01-24 06:07:29
81647	1	0	0	0	0	1	3	2276	2298.85	1	Thursday	2019-01-24 22:06:30
81648	1	0	1	0	0	0	3	408	376.658	1	Friday	2019-01-25 08:24:27
81649	1	0	0	0	0	1	3	3717	4377.14	1	Friday	2019-01-25 22:28:22
81650	0	1	1	0	0	0	3	1476	1646.02	1	Saturday	2019-01-26 09:11:51
81651	0	1	0	1	0	0	3	1251	1469.4	1	Saturday	2019-01-26 13:13:25
81652	0	1	0	1	0	0	3	678	783.642	1	Sunday	2019-01-27 11:27:31
81653	0	1	0	0	1	0	3	685	750.568	1	Sunday	2019-01-27 16:12:21
81654	1	0	1	0	0	0	3	408	361.362	1	Monday	2019-01-28 08:51:18
81655	1	0	0	0	0	1	3	2276	2316.21	1	Monday	2019-01-28 23:08:09
81656	1	0	1	0	0	0	3	408	340.92	1	Tuesday	2019-01-29 06:52:47
81657	1	0	0	0	1	0	3	2475	2855.51	1	Tuesday	2019-01-29 18:27:34
81658	1	0	1	0	0	0	3	408	350.177	1	Wednesday	2019-01-30 06:45:34
81659	1	0	0	0	0	1	3	2276	2669.28	1	Wednesday	2019-01-30 20:13:22

Format Resize Background color Column min/max OK Cancel

Şekil 4.12. Diğer Kamu Çalışanlarına Ait Riskli Araç Sürüşü

4.2.5. Öğrencilere ait sonuçlar

Öğrencilerin araç sürüşlerine ait üretilen veriler Şekil 4.13. de yer almaktadır.

df_main - DataFrame

Index	H_jci	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
86146	1	0	1	0	0	0	4	522	457.896	1	Thursday	2019-01-24 07:13:18
86147	1	0	0	0	0	1	4	469	441.936	1	Thursday	2019-01-24 21:00:45
86148	1	0	1	0	0	0	4	522	481.588	1	Friday	2019-01-25 07:19:40
86149	1	0	0	0	0	1	4	2585	2148.04	1	Friday	2019-01-25 19:17:56
86150	0	1	0	0	1	0	4	757	726.246	1	Saturday	2019-01-26 16:00:37
86151	0	1	0	0	1	0	4	812	963.709	1	Saturday	2019-01-26 20:41:45
86152	0	1	1	0	0	0	4	727	606.007	1	Sunday	2019-01-27 09:07:10
86153	0	1	0	1	0	0	4	667	787.911	1	Sunday	2019-01-27 11:06:02
86154	1	0	1	0	0	0	4	522	470.622	1	Monday	2019-01-28 06:48:44
86155	1	0	0	0	1	0	4	2276	2164.25	1	Monday	2019-01-28 17:02:55
86156	1	0	1	0	0	0	4	522	518.509	1	Tuesday	2019-01-29 07:54:34
86157	1	0	0	0	1	0	4	2126	2064.34	1	Tuesday	2019-01-29 18:35:53
86158	1	0	1	0	0	0	4	522	490.707	1	Wednesday	2019-01-30 07:28:46
86159	1	0	0	0	0	1	4	1389	1184.27	1	Wednesday	2019-01-30 23:44:48

Format Resize Background color Column min/max OK Cancel

Şekil 4.13. Öğrencilere Ait Riskli Araç Sürüşü

4.2.6. Esnaf ve işçilere ait sonuçlar

Esnaf ve işçilerin araç sürüşlerine ait üretilen veriler Şekil 4.14. de yer almaktadır.

df_main - DataFrame

Index	H_jci	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
212146	1	0	1	0	0	0	5	812	791.389	1	Thursday	2019-01-24 06:41:07
212147	1	0	0	0	1	0	5	2143	2465.51	1	Thursday	2019-01-24 17:35:26
212148	1	0	1	0	0	0	5	812	713.468	1	Friday	2019-01-25 06:09:15
212149	1	0	0	0	1	0	5	2143	2346.87	1	Friday	2019-01-25 18:30:15
212150	0	1	0	0	0	1	5	752	605.717	1	Saturday	2019-01-26 22:34:30
212151	0	1	0	0	0	1	5	852	784.686	1	Saturday	2019-01-26 00:22:22
212152	0	1	0	1	0	0	5	1713	1618.15	1	Sunday	2019-01-27 14:57:02
212153	0	1	0	0	1	0	5	1880	1828.75	1	Sunday	2019-01-27 17:21:26
212154	1	0	1	0	0	0	5	812	739.644	1	Monday	2019-01-28 07:19:52
212155	1	0	0	0	0	1	5	2503	2062.76	1	Monday	2019-01-28 19:44:48
212156	1	0	1	0	0	0	5	812	778.136	1	Tuesday	2019-01-29 06:55:01
212157	1	0	0	0	0	1	5	2503	2035.09	1	Tuesday	2019-01-29 20:49:45
212158	1	0	1	0	0	0	5	812	680.395	1	Wednesday	2019-01-30 07:29:03
212159	1	0	0	0	0	1	5	1488	1212.85	1	Wednesday	2019-01-30 20:01:19

Format Resize Background color Column min/max OK Cancel

Şekil 4.14. Esnaf ve İşçilere Ait Riskli Araç Sürüşü

4.2.7. Pazarlama çalışanlarına ait sonuçlar

Pazarlama çalışanlarının Araç sürüşlerine ait üretilen veriler Şekil 4.15. de yer almaktadır.

df_main - DataFrame

Index	H_jci	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
229846	1	0	1	0	0	0	6	921	792.83	1	Thursday	2019-01-24 07:41:16
229847	1	0	0	0	0	1	6	1485	1186.87	1	Thursday	2019-01-24 19:38:28
229848	1	0	1	0	0	0	6	921	740.501	1	Friday	2019-01-25 06:35:42
229849	1	0	0	0	0	1	6	1901	1712.61	1	Friday	2019-01-25 20:33:40
229850	0	1	0	0	1	0	6	1240	1293.18	1	Saturday	2019-01-26 19:36:04
229851	0	1	0	0	0	1	6	1363	1378.54	1	Saturday	2019-01-26 06:28:29
229852	0	1	0	1	0	0	6	1244	1222.55	1	Sunday	2019-01-27 14:54:34
229853	0	1	0	0	1	0	6	1383	1242.26	1	Sunday	2019-01-27 19:56:55
229854	1	0	1	0	0	0	6	921	717.081	1	Monday	2019-01-28 07:00:36
229855	1	0	0	0	0	1	6	2428	2851.71	1	Monday	2019-01-28 23:45:34
229856	1	0	1	0	0	0	6	921	845.161	1	Tuesday	2019-01-29 06:05:16
229857	1	0	0	0	0	1	6	1812	1564.7	1	Tuesday	2019-01-29 21:26:20
229858	1	0	1	0	0	0	6	921	713.58	1	Wednesday	2019-01-30 06:32:42
229859	1	0	0	0	0	1	6	1812	1593.23	1	Wednesday	2019-01-30 20:30:58

Format Resize Background color Column min/max OK Cancel

Şekil 4.15. Pazarlama Çalışanlarına Ait Riskli Araç Sürüşü

4.2.8. Emeklilere ait sonuçlar

Emeklilerin araç sürüşlerine ait üretilen veriler Şekil 4.16. da yer almaktadır.

df_main - DataFrame

Index	H_jci	H_sonu	Sabah	Ogle	Aksam	Diger_Vakitler	Meslek	Beklenen_Sure	Gerçeklesen_Sure	Durum	Gun	Tarih
238245	1	0	0	0	0	1	7	1030	1159.71	1	Wednesday	2019-01-23 00:48:18
238246	1	0	0	1	0	0	7	687	583.359	1	Thursday	2019-01-24 11:11:35
238247	1	0	0	0	1	0	7	2204	2162.7	1	Thursday	2019-01-24 17:48:45
238248	1	0	0	1	0	0	7	519	613.088	1	Friday	2019-01-25 14:14:39
238249	1	0	0	0	0	1	7	1030	1040.37	1	Friday	2019-01-25 21:26:54
238250	0	1	0	0	1	0	7	356	321.399	1	Saturday	2019-01-26 17:59:24
238251	0	1	0	0	1	0	7	425	410.311	1	Saturday	2019-01-26 18:21:11
238252	0	1	0	0	0	1	7	613	589.318	1	Sunday	2019-01-27 21:08:20
238253	0	1	0	0	0	1	7	566	520.007	1	Sunday	2019-01-27 00:38:06
238254	1	0	0	1	0	0	7	519	590.592	1	Monday	2019-01-28 15:42:13
238255	1	0	0	0	0	1	7	916	1032.64	1	Monday	2019-01-28 19:32:03
238256	1	0	0	1	0	0	7	343	288.267	1	Tuesday	2019-01-29 11:11:04
238257	1	0	0	0	0	1	7	512	485.548	1	Tuesday	2019-01-29 19:12:49
238258	1	0	1	0	0	0	7	835	798.049	1	Wednesday	2019-01-30 10:38:40
238259	1	0	0	0	0	1	7	834	799.605	1	Wednesday	2019-01-30 00:25:24

Format Resize Background color Column min/max OK Cancel

Şekil 4.16. Emeklilere Ait Riskli Araç Sürüşü

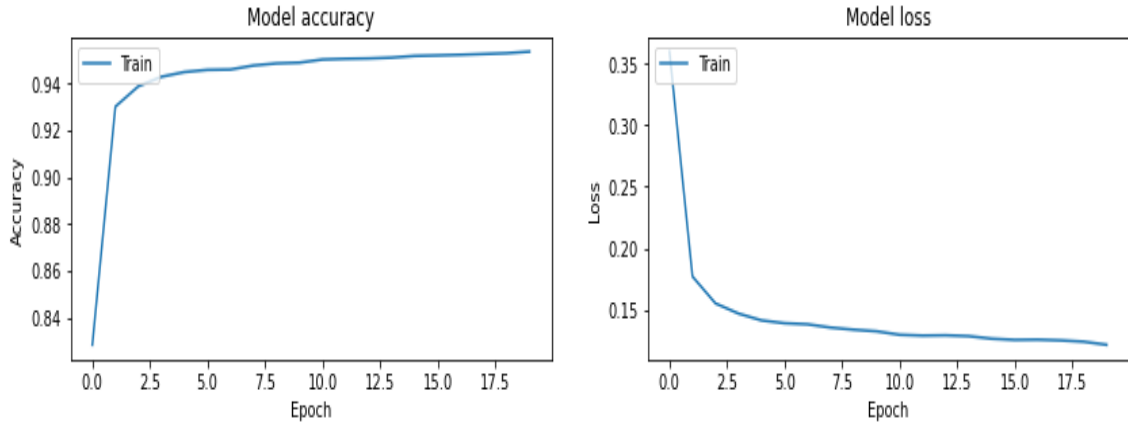
4.3. Yapay Sinir Ağının Eğitim ve Test Sonuçları

Yapay Sinir Ağının eğitimi 20 iterasyonda bitirilmiş ve her bir iterasyona ait detaylı bilgiler konsol ekranına yazdırılmıştır. Şekil 4.17. de iterasyonların sonuç bilgileri yer almaktadır.

```
Epoch 1/20
357390/357390 [=====] - 34s 95us/step - loss: 0.3580 - accuracy: 0.8298
Epoch 2/20
357390/357390 [=====] - 33s 93us/step - loss: 0.1760 - accuracy: 0.9298
Epoch 3/20
357390/357390 [=====] - 33s 92us/step - loss: 0.1517 - accuracy: 0.9402
Epoch 4/20
357390/357390 [=====] - 33s 93us/step - loss: 0.1468 - accuracy: 0.9421
Epoch 5/20
357390/357390 [=====] - 33s 94us/step - loss: 0.1397 - accuracy: 0.9446
Epoch 6/20
357390/357390 [=====] - 33s 94us/step - loss: 0.1376 - accuracy: 0.9465
Epoch 7/20
357390/357390 [=====] - 34s 94us/step - loss: 0.1350 - accuracy: 0.9477
Epoch 8/20
357390/357390 [=====] - 34s 94us/step - loss: 0.1322 - accuracy: 0.9488
Epoch 9/20
357390/357390 [=====] - 34s 94us/step - loss: 0.1324 - accuracy: 0.9486
Epoch 10/20
357390/357390 [=====] - 34s 94us/step - loss: 0.1283 - accuracy: 0.9508
Epoch 11/20
357390/357390 [=====] - 33s 94us/step - loss: 0.1267 - accuracy: 0.9515
Epoch 12/20
357390/357390 [=====] - 33s 92us/step - loss: 0.1258 - accuracy: 0.9524
Epoch 13/20
357390/357390 [=====] - 33s 92us/step - loss: 0.1252 - accuracy: 0.9521
Epoch 14/20
357390/357390 [=====] - 33s 92us/step - loss: 0.1238 - accuracy: 0.9529
Epoch 15/20
357390/357390 [=====] - 33s 92us/step - loss: 0.1248 - accuracy: 0.9528
Epoch 16/20
357390/357390 [=====] - 33s 92us/step - loss: 0.1235 - accuracy: 0.9535
Epoch 17/20
357390/357390 [=====] - 33s 92us/step - loss: 0.1231 - accuracy: 0.9530
Epoch 18/20
357390/357390 [=====] - 33s 92us/step - loss: 0.1237 - accuracy: 0.9536
Epoch 19/20
357390/357390 [=====] - 34s 94us/step - loss: 0.1232 - accuracy: 0.9537
Epoch 20/20
357390/357390 [=====] - 34s 95us/step - loss: 0.1226 - accuracy: 0.9538
```

Şekil 4.17. İterasyon Sonuç Bilgileri

Model Accuracy'nin (doğruluk oranı) ve Model Loss'un (kayıp oranı) 12. iterasyon sonrasında yaklaşık hep aynı değerlerde olduğu görülmektedir. Bu sebeple iterasyon sayısı 20 alınmıştır. Model Accuracy ve Model Loss'a ait iterasyona bağlı değişim oranı Şekil 4.18. de verilmiştir.



Şekil 4.18. Model Accuracy ve Model Loss'un İterasyona Bağlı Değişimi

X_test DataFrame değişkeninde bulunan verilerden sırasıyla X_test[0], X_test[10], X_test[100], X_test[1000], X_test[10000], X_test[20000], X_test[30000], X_test[40000], X_test[50000], X_test[55000], X_test[70000], X_test[76000] verisi, nihai modelde tahminleme işlemine tabi tutulmuştur. Yapay sinir ağındaki verilerin kodlanması one-hot tekniğinde olduğundan test ve gerçek sonuç çıktıları Şekil 4.19. deki gibi olmuştur.

Risksiz, “Durum = 0” ifade eden kodlanmış veri [1 0] ‘dır.

Riskli, “Durum = 1” ifade eden kodlanmış veri [0 1] ‘dir.

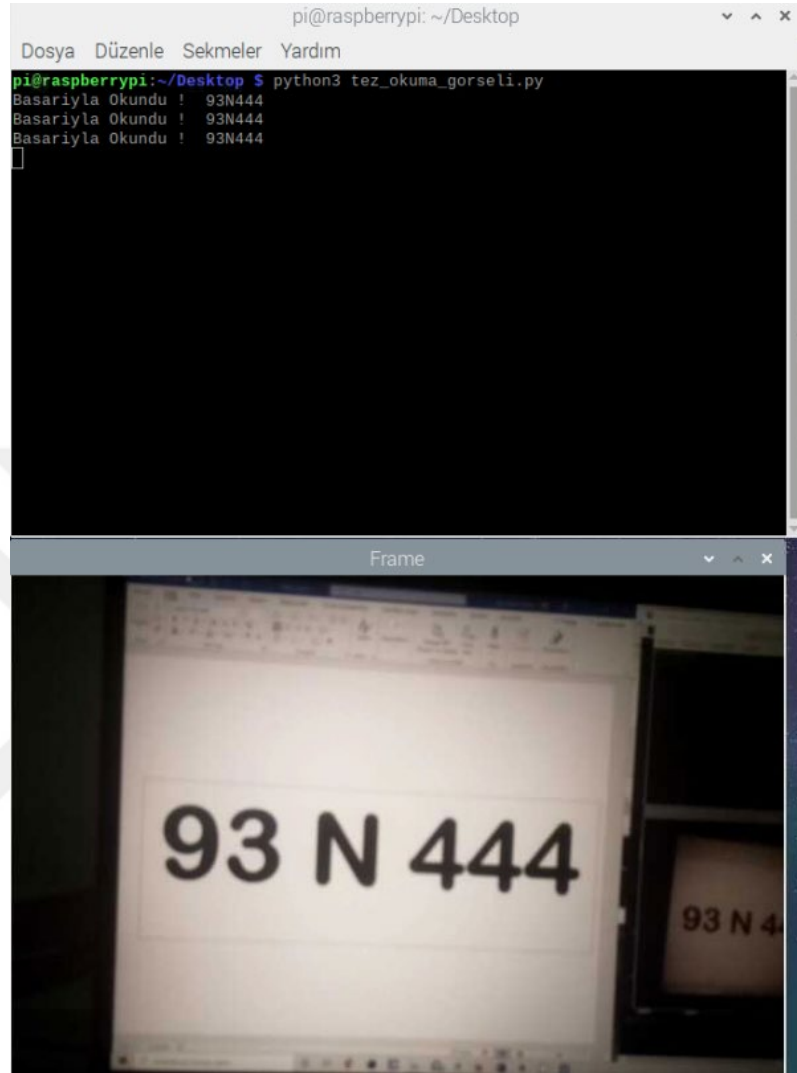
x[0]:	Tahmin sonucu: [[1. 0.]] Gerçek sonucu: [1 0]	x[30000]:	Tahmin sonucu: [[1. 0.]] Gerçek sonucu: [1 0]
x[10]:	Tahmin sonucu: [[1. 0.]] Gerçek sonucu: [0 1]	x[40000]:	Tahmin sonucu: [[0. 1.]] Gerçek sonucu: [0 1]
x[100]:	Tahmin sonucu: [[0. 1.]] Gerçek sonucu: [0 1]	x[50000]:	Tahmin sonucu: [[0. 1.]] Gerçek sonucu: [0 1]
x[1000]:	Tahmin sonucu: [[1. 0.]] Gerçek sonucu: [1 0]	x[55000]:	Tahmin sonucu: [[1. 0.]] Gerçek sonucu: [1 0]
x[10000]:	Tahmin sonucu: [[1. 0.]] Gerçek sonucu: [1 0]	x[70000]:	Tahmin sonucu: [[1. 0.]] Gerçek sonucu: [1 0]
x[20000]:	Tahmin sonucu: [[0. 1.]] Gerçek sonucu: [0 1]	x[76000]:	Tahmin sonucu: [[0. 1.]] Gerçek sonucu: [1 0]

Şekil 4.19. Test Sonuçları

4.4. Temsili Plaka Okuma Yazılımı Sonucu

Temsili plaka düzgün okunmuş ise okunan plaka konsol ekranına yazdırılmış, varsa boşluklar silinmiş ve konsol ekranına yazdırılmıştır. Plaka yoksa ya da

okunamadıysa yazılım boşa kalmış, plaka görünmesini beklemiştir. Şekil 4.20. de yazılım çıktısı görülmektedir.

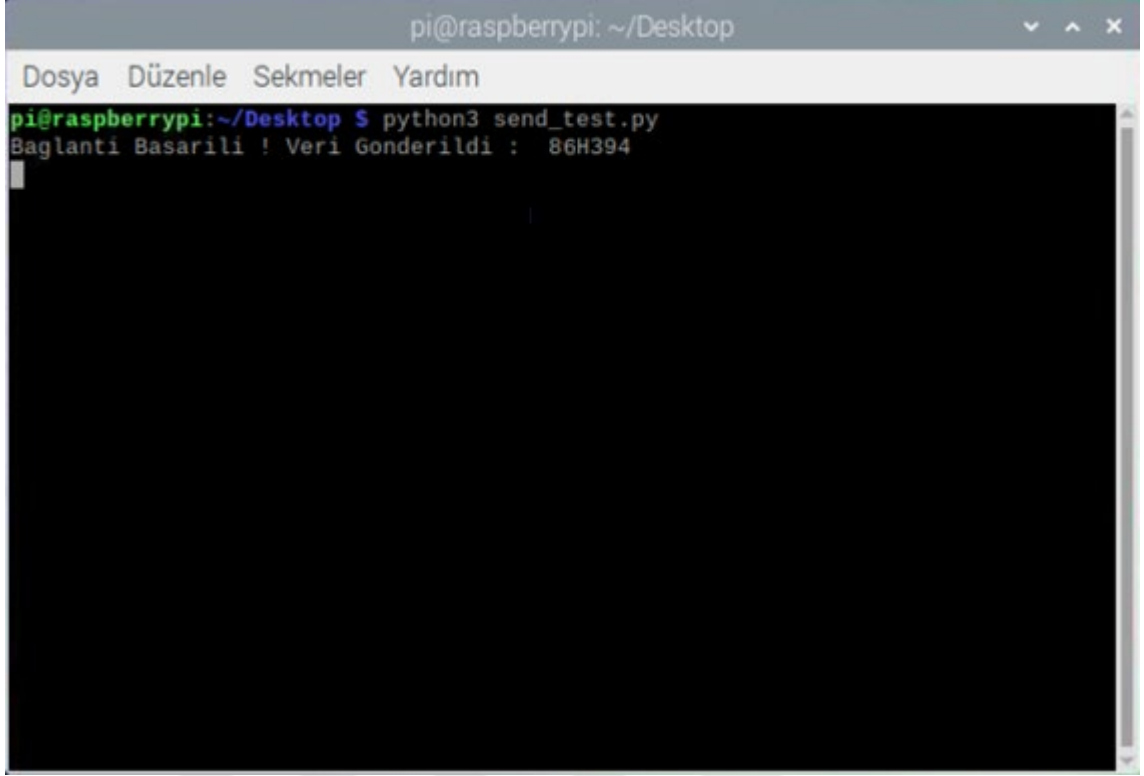


Şekil 4.20. Temsili Plaka Okuma Yazılımı Sonucu

4.5. Raspberry Pi ve Bilgisayar Arası İletişim Sonuçları

4.5.1. Raspberry Pi'nin veri göndermesi ve bilgisayarın gönderilen veriyi okuması

Şekil 4.21. de Raspberry Pi'nin ThingSpeak'e veri gönderdiğini gösteren konsol ekran görüntüsü verilmiştir.



Şekil 4.21. ThingSpeak'e Veri Gönderme Sonucu

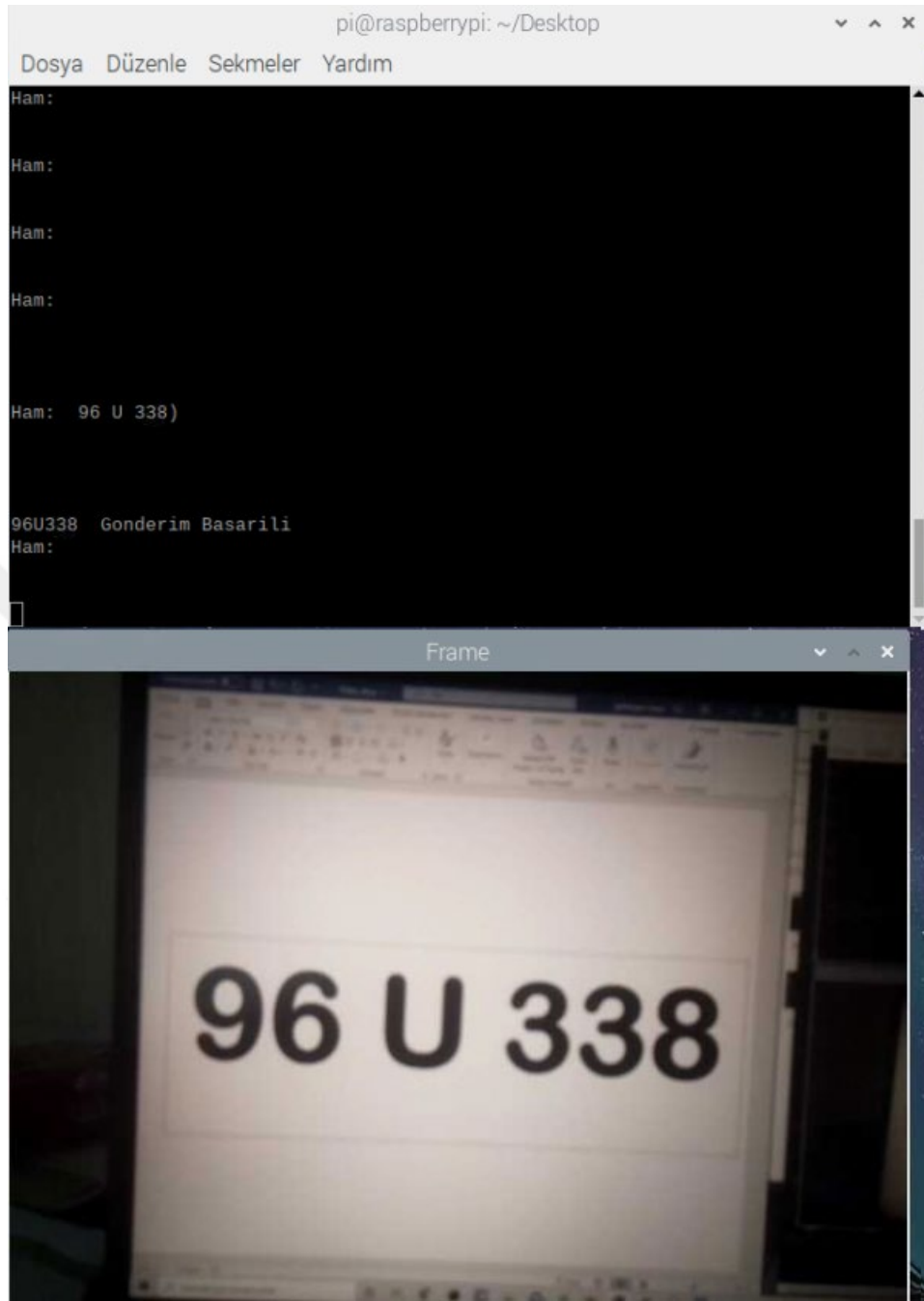
Şekil 4.22. de bilgisayarın ThingSpeak'ten veri okuduğu gösteren Spider konsol ekran görüntüsü verilmiştir.

```
In [5]: runfile('C:/Users/AsusTUF/Desktop/Niyazi/Y_L/TEZ/c_0/map_0/genel_arayuz/
YSA_SON/calisan_1/computer_read_values.py', wdir='C:/Users/AsusTUF/Desktop/Niyazi/
Y_L/TEZ/c_0/map_0/genel_arayuz/YSA_SON/calisan_1')
{'created_at': '2021-03-12T21:12:29Z', 'entry_id': 210, 'field1': '86H394'}
Baglanti Basarili ! Alinan veri : 86H394
{'created_at': '2021-03-12T21:12:29Z', 'entry_id': 210, 'field1': '86H394'}
Baglanti Basarili ! Alinan veri : 86H394
{'created_at': '2021-03-12T21:12:29Z', 'entry_id': 210, 'field1': '86H394'}
Baglanti Basarili ! Alinan veri : 86H394
{'created_at': '2021-03-12T21:12:29Z', 'entry_id': 210, 'field1': '86H394'}
Baglanti Basarili ! Alinan veri : 86H394
{'created_at': '2021-03-12T21:12:29Z', 'entry_id': 210, 'field1': '86H394'}
Baglanti Basarili ! Alinan veri : 86H394
```

Şekil 4.22. ThingSpeak'ten Veri Okuma Sonucu

4.5.2. Veri gönderim ve temsili plaka okuma yazılımlarının entegre hale getirilmesi sonucu

Veri gönderme kodları, temsili plaka okuma yazılımının uygun yerlerine eklenerek Şekil 4.23. deki sonuç alınmıştır.



Şekil 4.23. Plaka Tanıma ve Plaka Bilgisi Gönderimi Sonucu

4.6. YSA Modeliyle Öneri Yazılımının Yapılması ve ThingSpeak İletişim Entegrasyonu Sonucu

4.6.1. Öneri yazılımı sonucu

Araç sayısı kadar üretilen plakalardan rastgele dördünün seçimine göre öneri yazılımı sonucu Şekil 4.24. de verilmiştir.

```

PLAKA: 88T918
TOPLAM RISKLI HAREKET: 36 /60
DURDURMA YUZDESI: 60.0
Meslek: 7 Emekli
PLAKA: 91C927
TOPLAM RISKLI HAREKET: 9 /60
DURDURMA YUZDESI: 15.0
Meslek: 1 MEB Calisani
PLAKA: 88L672
TOPLAM RISKLI HAREKET: 49 /60
DURDURMA YUZDESI: 81.66666666666667
Meslek: 5 Esnaf_Isci
PLAKA: 88G003
TOPLAM RISKLI HAREKET: 11 /60
DURDURMA YUZDESI: 18.333333333333332
Meslek: 1 MEB Calisani

```

Şekil 4.24. Öneri Sonuçları

4.6.2. Öneri yazılımının ThingSpeak ile entegrasyonu sonucu

Şekil 4.25. de “*Baglanti Basarili*” çıktısıyla ilgili kodun çalışma sonucu verilmiştir. ThingSpeak’e en son gönderilen temsili plaka okunarak yazılım işlemleri gerçekleştirmiştir. Aynı plakayı üst üste okuması sonucu “*Yeni Plaka Yok, Tanimsiz Plaka*” uyarısı vermiştir.

```

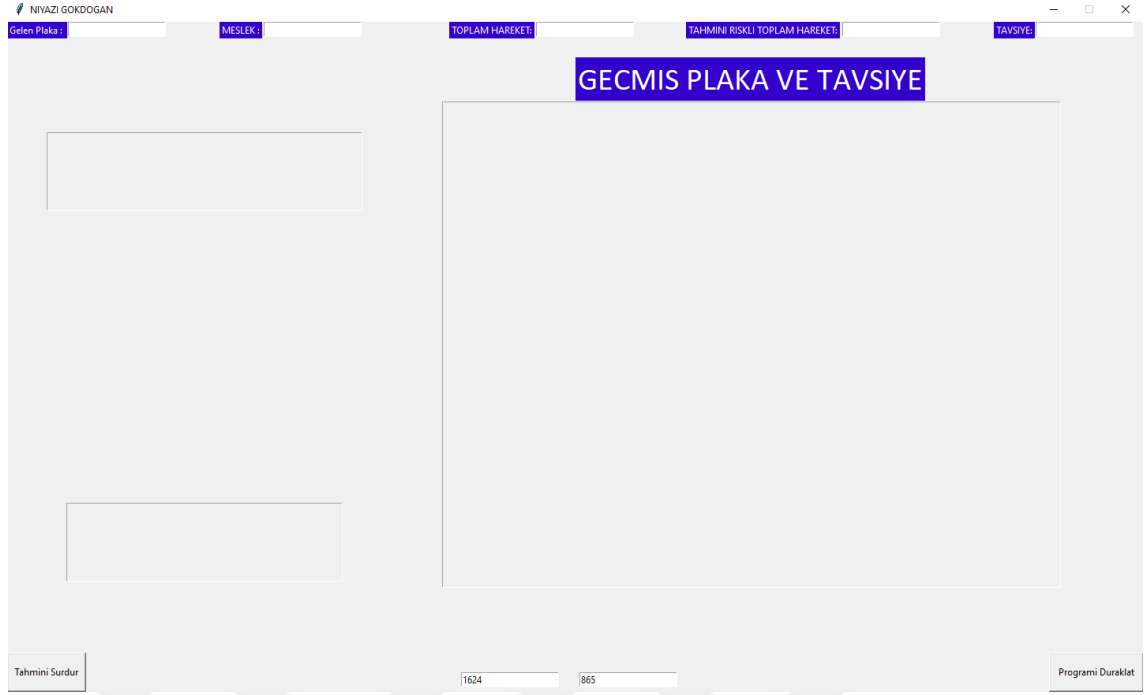
PLAKA: 96U338
TOPLAM RISKLI HAREKET: 52 /60
DURDURMA YUZDESI: 86.66666666666667
Meslek: 1 MEB Calisani
96U338
Yeni Plaka yok ,Tanimsiz Plaka
96U338
Yeni Plaka yok ,Tanimsiz Plaka
96U338
Yeni Plaka yok ,Tanimsiz Plaka

```

Şekil 4.25. Entegrasyon Sonucu

4.7. Bilgisayar Kullanıcı Arayüzü Sonucu

Arayüz çerçevesi 1440x850 piksel çözünürlükte oluşturulmuştur. Arayüzde 6 label, 9 textbox, 2 buton, 1 listbox kullanılarak, sade ve anlaşılır olması hedeflenmiş ve Şekil 4.26. daki arayüz sonucu alınmıştır.



Şekil 4.26. Arayüz

Tez çalışmasında beklenen çıktı Raspberry Pi'nin okuduğu plaka verisini bilgisayarına aktararak, YSA modelinin gelen araç plakasıyla bir tavsiyede bulunmasıdır. YSA modelinden alınan sonuç kullanıcıya 3 farklı tavsiye sunabilmektedir.

Şekil 4.27. de YSA modeli “*Serbest*” tavsiyesiyle temsilde gelen aracın suçla karşılaşma potansiyelinin düşük düzeyde olduğunu belirterek, aracın durmadan devam etmesi önerisinde bulunmaktadır.

NIYAZI GOKDOGAN

Gelen Plaka: 93 U 249 MESLEK: MEB Calisani TOPLAM HAREKET: 60 TAHMINI RISKLI TOPLAM HAREKET: 10 TAVSIYE: Serbest

GEÇMİŞ PLAKA VE TAVSİYE

93 U 249	Serbest	MEB Calisani
98 F 194	Serbest	MEB Calisani
88 U 695	Durdur	Emekli
99 N 643	Serbest	MEB Calisani
85 I 842	Durdur	Esnaf_Isci
97 U 802	Serbest	Esnaf_Isci
87 T 475	Yogunluk Bazli Islem	Hastane_Calisani
84 K 752	Durdur	Kolluk_Kuvveti
94 K 282	Durdur	Emekli
97 I 856	Durdur	Esnaf_Isci

93 U 249

Serbest

Tahmini Surdur 1359 825 Programi Duraklat

Şekil 4.27. “*Serbest*” Tavsiyesi

Şekil 4.28. de YSA modeli “Yogunluk Bazlı İşlem” tavsiyesiyle temsilde gelen aracın suça karışma potansiyelinin orta düzeyde olduğunu belirterek, aracın kontrol noktasındaki yoğunluğa göre kolluk kuvvetinin tecrübesini kullanması gerektiği önerisinde bulunmaktadır.

NİYAZI GOKDOĞAN	
Gelen Plaka: 96 V 624	MESLEK: Esnaf_Isci
TOPLAM HAREKET: 60	TAHMINİ RİSKLİ TOPLAM HAREKET: 17
TAVSİYE: Yogunluk Bazlı İşlem	
GEÇMİŞ PLAKA VE TAVSİYE	
96 V 624	Yogunluk Bazlı İşlem Esnaf_Isci
83 P 049	Serbest MEB Calisani
99 S 784	Durdur Esnaf_Isci
91 A 573	Serbest Esnaf_Isci
83 V 037	Serbest Esnaf_Isci
84 L 204	Serbest Esnaf_Isci
92 V 544	Serbest Esnaf_Isci
89 V 908	Serbest MEB Calisani
98 G 747	Durdur Emekli
93 F 653	Serbest Esnaf_Isci
96 A 040	Serbest Esnaf_Isci
83 K 522	Serbest Hastane_Calisani
85 E 767	Durdur Esnaf_Isci
87 O 888	Durdur Esnaf_Isci
95 Y 090	Yogunluk Bazlı İşlem Hastane_Calisani
Tahmini Sürdür	
1370	
822	
Programı Duraklat	

Şekil 4.28. “Yogunluk Bazlı İşlem” Tavsiyesi

Şekil 4.29. da YSA modeli “*Durdur*” tavsiyesiyle temsilde gelen aracın suçla karşılaşma potansiyelinin yüksek seviyede olduğunu belirterek, aracın durdurulması önerisinde bulunmaktadır.

NIYAZI GOKDOGAN

Gelen Plaka: 98 F 675 MESLEK: Esnaf_Isci TOPLAM HAREKET: 60 TAHMİNİ RİSKLİ TOPLAM HAREKET: 52 TAVSİYE: Durdur

GEÇMİŞ PLAKA VE TAVSİYE

98 F 675	Durdur	Esnaf_Isci
91 K 397	Serbest	Hastane_Calisani
93 U 249	Serbest	MEB Calisani
98 F 194	Serbest	MEB Calisani
88 U 695	Durdur	Emekli
99 N 643	Serbest	MEB Calisani
85 I 842	Durdur	Esnaf_Isci
97 U 802	Serbest	Esnaf_Isci
87 T 475	Yogunluk Bazli Islem	Hastane_Calisani
84 K 752	Durdur	Kolluk_Kuvveti
94 K 282	Durdur	Emekli
97 I 856	Durdur	Esnaf_Isci

98 F 675

Durdur

Tahmini Sürdür 1374 820 Programi Duraklat

Şekil 4.29. “*Durdur*” Tavsiyesi

5. SONUÇLAR VE ÖNERİLER

5.1 Sonuçlar

Bu tez çalışmasında araçların sık görülebileceği noktalar ve geçiş güzergahları temsili kameralar eklenerek belirlenmiştir. Kamera noktalarının enlem ve boylam bilgileriyle Spyder IDE kullanılarak python diliyle Konya merkez ilçelerini içeren harita üzerine yerleştirilmiştir. Potansiyel risk kriterleri dikkate alınarak 476520 Big Data özelliklerini taşıyan DataFrame oluşturulmuştur. DataFrame düzenlenerek YSA eğitime hazır hale getirilmiştir. YSA eğitimi gerçekleştirilerek Raspberry Pi temsili plaka okuma ve plaka bilgisi gönderim yazılımı yapılmıştır. YSA modeliyle tahminleme ve plaka bilgisini ThingSpeak'ten okuma çerçevesinde bilgisayar kullanıcı arayüzü tasarlanmıştır.

YSA kullanılarak araçların risk potansiyellerini tahmin etme hedefi ve sonucu kullanıcıya arayüz üzerinden kırmızı “*Durdur*”, turuncu “*Yogunluk Bazlı İşlem*”, yeşil “*Serbest*” görsel ve yazılı önerilerle aktarılmıştır.

Tez çalışmasının amacına uygun olarak araç hareketleri temsili olarak üretilmesine karşın YSA doğruluk oranı yaklaşık %95 olduğu görülmüştür. Uygulamadaki yüksek doğruluk oranı YSA eğitim sonucunu desteklediği görülmektedir.

Üretilen araç hareketlerinin toplamını oluşturan Big Data homojen bir yapıya sahip veri içermektedir. Buna rağmen YSA çıktısının oldukça başarılı olması, Yapay Sinir Ağı'nın eğitiminde parametrelerin önemini göstermektedir. YSA eğitiminin uygun parametrelerle gerçekleştirilmesi, farklı şehir ve bölgelerde bu çalışmanın kullanımının mümkün olacağını göstermektedir.

Potansiyel riski bulunan kimyasal ürün satış yeri, silah ve ekipmanları satış merkezi, alkollü eğlence merkezi lokasyonlarının sayısı az olmasına rağmen risk grubundaki araç hareketlerinin diğer kriterlerinin de etkisinin potansiyel suça karışma oranını etkilediği Yapay Sinir Ağı'nın yüksek doğruluğundan çıkarılabilmektedir. Bu sonuçla Türkiye'nin farklı şehirlerindeki kültürel ve sosyal yaşam alışkanlıklarına bağlı olarak Yapay Sinir Ağı'nın uygun parametrelerle eğitiminin araçların suça karışma potansiyellerini yüksek doğrulukla tahmin edebileceği anlaşılmaktadır.

5.2 Öneriler

Bu çalışmada ele alınan araç davranışları Konya merkez ilçelerinde yaşayan insanların kültürel, sosyal, iş yaşamları göz önüne alınarak oluşturulduğundan, farklı şehir ve bölgelerde bu çalışmanın uygulanması önerilmektedir.

Oluşturulan verilerin oldukça homojen bir yapıya sahip olmasından dolayı gerçek araç hareketleri verileriyle Big Data'nın oluşturulması ve YSA eğitim parametrelerinin buna göre belirlenmesi önerilmektedir. Big Data verileri 30, 90, 180, 360 gün gibi farklı periyotlarda alınabilir, YSA eğitim parametreleri buna göre düzenlenebilir.

Plaka veri gönderme sistemi ihtiyaca göre seçilmelidir. Hücresel veri aktarımının düşük olduğu lokasyonlarda kriptolu radyo frekansı tabanlı sistemlerle aktarım yapılabilir.

Plaka tanıma ve tanınmış veri aktarımında yüksek hız istenmesi durumunda yüksek işlem hızına sahip mikrodenetleyici veya plaka tanıma sistemi üzerine özelleştirilmiş tanıma sistemleri kullanılabilir.

6. KAYNAKLAR

- [1] Eryılmaz, M. Bedri., 2003, Türk ve İngiliz Hukukunda ve Uygulamasında Durdurma ve Arama, Seçkin Yayıncılık, Ankara.
- [2] Kuntbay, İ., 1990, İnsan Hakları ve Polis, TODAİE Yayınları, Ankara.
- [3] Trafik Kontrolleri Yönergesi., 2011, <http://trafik.gov.tr/trafik-mevzuat>, [Ziyaret Tarihi: 09.11.2020].
- [4][http://www.artemishaber.com/haberdetay/TRAFIK-ELEKTRONIK-DENETLEME-SISTEMI-\(TEDES\)-NEDIR_/901](http://www.artemishaber.com/haberdetay/TRAFIK-ELEKTRONIK-DENETLEME-SISTEMI-(TEDES)-NEDIR_/901), [Ziyaret Tarihi: 27.11.2020].
- [5] <http://www.tedesbilgi.com/eds/>, [Ziyaret Tarihi: 27.11.2020].
- [6] Geçin, M., Çömlekçi, S. ve Ülgen, Mehmet E., 2017, Taşınabilir Trafik Denetleme Sistemi. *Karayolu Trafik Güvenliği Sempozyumu ve Sergisi Seçilmiş Bildiriler*, Ankara.
- [7] ISSD. (2020). POINTR, <https://www.issd.com.tr/tr/https-www-issd-com-tr-tr-18924-Koridor-Hizi-Ihlal-Tespit-Sistemi>, [Ziyaret Tarihi: 29.11.2020].
- [8] Boztoprak, H., 2007, Gerçek zamanlı taşıt plaka sistemi, Yüksek Lisans Tezi, *Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü*, Isparta.
- [9] Halim, Z., Kalsoom, R., Bashir, S., and Abbas, G., 2016, Artificial intelligence techniques for driving safety and vehicle crash prediction. *Artificial Intelligence Review*, 46 (3), 351-387.
- [10] Olayode, O. I., Tartibu, L. K., and Okwu, M. O., 2020, Application of Artificial Intelligence in Traffic Control System of Non-autonomous Vehicles at Signalized Road Intersection. *Procedia CIRP*, 91, 194-200.
- [11] Khalid, M., 1996, Intelligent traffic lights control by fuzzy logic. *Malaysian Journal of Computer Science*, 9 (2), 29-35.
- [12] Liu, Z., 2007, A survey of intelligence methods in urban traffic signal control. *IJCSNS International Journal of Computer Science and Network Security*, 7 (7), 105-112.
- [13] Diamantini, C., Potena, D., and Storti, E., 2018, A Big Data Framework for Analysis of Traffic Data in Italian Highways. In *International Symposium on Methodologies for Intelligent Systems* (pp. 161-168). Springer, Cham.
- [14] Akin, D., and Akba, B., 2010, A neural network (NN) model to predict intersection crashes based upon driver, vehicle and roadway surface characteristics. *Sci. Res. Essays*, 5 (19), 2837-2847.

- [15] Karayolları Genel Müdürlüğü.,1997, *Karayolları Trafik Kanunu ve Ceza Rehberi*. TC Bayındırlık ve İskan Bakanlığı, Ankara.
- [16]<https://data.tuik.gov.tr/Bulten/Index?p=Karayolu-Trafik-Kaza-Istatistikleri-2019-33628>, [Ziyaret Tarihi: 09.11.2020].
- [17] Darçın, M., 2000, Trafik Denetiminin Etkinleştirilmesi. Yüksek Lisans Tezi. *Gazi Üniversitesi Fen Bilimleri Enstitüsü*, Ankara.
- [18] Akbulut, F., 2016, “Kentsel Ulaşım Sistemlerinin Planlanması ve Yönetiminde Sürdürülebilir Politika Önerileri.” *Kastamonu Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 11, 336-355.
- [19] Payam, M. M., 2012, İlk ve Ortaöğretim Okullarında Trafik Eğitimindeki Temel Sorunlar: Siirt İli Örneği. Doktora Tezi, *Polis Akademisi Başkanlığı, Güvenlik Bilimleri Enstitüsü*, Ankara.
- [20] Karayolu Trafik Güvenliği Stratejisi ve Eylem Planı., 2012, *Resmi Gazete*, Sayı. 28370, Ankara, <https://www.resmigazete.gov.tr/eskiler/2012/07/20120731-17.htm>, [Ziyaret Tarihi: 02.11.2020].
- [21] Aavani P., Mithun K. Sawant, Sawant,S. and Rohit S. D., 2017, A Review on Adaptive Traffic Controls Systems
- [22] Teodorovic D. and Vukadinovic, K., 1998, Traffic Control and Transport Planning A Fuzzy Setes and Neural Networks Approach.
- [23] Tektaş, M., Akbaş, A. ve Topuz V., 2002, Yapay Zekâ Tekniklerinin Trafik Kontrolünde Kullanılması Üzerine Bir İnceleme. *1. Uluslararası Trafik ve Yol Güvenliği Kongresi*. Gazi Üniversitesi, Ankara, 551-559.
- [24] Altun, B., Atalı, G., Özkan, S. ve Karayel, D., 2018, “Görüntü İşleme Teknikleri ve YSA ile Bir Plaka Tanımlama Sisteminin Oluşturulması”. *International Symposium on Innovative Technologies in Engineering and Science*, 09-11 November 2018, Antalya.
- [25] Atalı, G., Özkan, S. ve Karayel, D., 2016, “Morfolojik Görüntü İşleme Tekniği ile Yapay Sinir Ağlarında Görüntü Tahribat Analizi”. *APJES IV-I*.
- [26] Bingöl, O. ve Kuşcu, Ö., 2008, Bilgisayar Tabanlı Araç Plaka Tanıma Sistemi. *Bilişim Teknolojileri Dergisi*, 1 (3).
- [27] Vijayan, A. and Shaji, L., 2015, Raspberry Pi a Tiny Computer. *International Journal of Engineering Research & Technology (IJERT)*, 3 (28), 1-3.
- [28] Abirami, N. and Leena-Jasmine, J.S., 2018, Accurate Vehicle Number Plate Regognition and Real Time Identification Using Raspberry Pi. *International Research of Engineering and Technology*, 5 (4),2563-2569.

- [29] Agarwal, A. and Pardakhe, N., 2017, Automatic License Plate Recognition Using Raspberry Pi. In *International Interdisciplinary Conference on Science Technology Engineering Management Pharmacy and Humanities*.
- [30] Fakhar, S., Saad, H., Fauzan, K., Affendi, H. and Aidil, A., 2019, Development of Portable Automatic Number Plate Recognition (ANPR) System on Raspberry Pi. *International Journal of Electrical and Computer Engineering*, 9 (3), 1805-1813.
- [31] Öztemel, E., 2006, Yapay Sinir Ağları, İstanbul: Papatya Yayıncılık.
- [32] Şen, Z., 2004, Yapay Sinir Ağları İlkeleri, İstanbul: Su Vakfı Yayınları.
- [33] Rizwan, M., Krishnan, P.N., Karthikeyan, R. and Kumar, S.R., 2016, Multi Layer Perception Type Artificial Neural Network based Traffic Control. *Indian Journal of Science and Technology*, 9 (5), 1-6.
- [34] Patel, M. and Ranganathan, N., 2001, IDUTC: An Intelligent Decision-Making System for Urban Traffic-Control Applications, *IEEE Transactions on Vehicular Technology*, 50 (3), 816-828.
- [35] Kaplan, O., Sağiroğlu, Ş. ve Çolakoğlu, Ö.F., 2019, Araç plaka tanıma sistemi .https://www.researchgate.net/profile/Seref_Sagiroglu/publication/228437161_ARAC_PLAKA_TANIMA_SISTEMI/links/5498a2120cf2519f5a1de7cc/ARAC-PLAKA-TANIMA-SISTEMI.pdf [Ziyaret Tarihi: 20 Temmuz 2019].
- [36] Anagnostopoulos, C. N. E., Anagnostopoulos, I. E., Loumos, V. and Kayafas, E., 2006, A license plate-recognition algorithm for intelligent transportation system applications, *IEEE Transactions on Intelligent Transportation Systems*, 7 (3), 377–392.
- [37] Akgüngör, A. P. ve Doğan, E., 2010, Farklı yöntemler kullanılarak geliştirilen trafik kaza tahmin modelleri ve analizi, *International Journal of Engineering Research and Development*, 2 (1), 16-22.
- [38] Sharma, S. (2017). Activation functions in neural networks. *Towards Data Science*, 6.
- [39] Nwankpa, C.E., Ijomah, W., Gachagan, A. and Marshall, S., 2018, *Activation Functions: Comparison of Trends in Practice and Research for Deep Learning*. arXiv preprint arXiv:1811.03378
- [40] <https://keras.io/api/layers/activations/> [Ziyaret Tarihi:08.03.2021].
- [41] Özdemir, İ., 2019, Denetimde Büyük Veri Analitiği Kullanımı İçin Yol Haritası, Yüksek Lisans Tezi, *Gazi Üniversitesi, Bilişim Enstitüsü*, Ankara.
- [42] Tellan, D., 2014, Kontrol Toplumunun Yaşam Hücreleri ya da Büyük Verinin Ekonomi-Politiği, Yüksel, H (Editör). *LaborComm 2014 5. Uluslararası İşçi ve İletişim Konferansı*, Ankara, 61-72.

- [43] Aktan, E., 2018, Büyük Veri: Uygulama Alanları, Analitiği ve Güvenlik Boyutu. *Bilgi Yönetimi Dergisi*, 1 (1), 1-22
- [44] Minelli, M., Chambers, M., and DhiraJ, A., 2012, Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends For Today's Businesses. America: *John Wiley & Sons*, 38-70.
- [45] Demchenko, Y., Ngo, C. and Membrey, P., 2013, *Architecture Framework and Components for the Big Data Ecosystem* (SNE Technical Report SNE-UVA-2013-02). System and Network Engineering Group & Universiteit Van Amsterdam.
- [46] [https://www.bigdataframework.org/four-vs-of-big-data/#:~:text=Velocity%20of%20Big%20Data,distinct%20\(distributed\)%20processing%20techniques](https://www.bigdataframework.org/four-vs-of-big-data/#:~:text=Velocity%20of%20Big%20Data,distinct%20(distributed)%20processing%20techniques). [Ziyaret Tarihi:02.03.2021].
- [47] Amini, S., Gerostathopoulos, I. and Prehofer, C., 2017, Big Data Analytics Architecture for Real-Time Traffic Control. *2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, 710-715.
- [48] Shengdong, M., Zhengxian, X. and Yixiang, T., 2019, Intelligent Traffic Control System Based on Cloud Computing and Big Data Mining, *IEEE Transactions on Industrial Informatics*, 15 (12), 6583-6592.
- [49] D. Singh, D., Vishnu, C. and Mohan, C.K., 2016, Visual Big Data Analytics for Traffic Monitoring in Smart City, *15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 886-891.
- [50] <https://docs.python.org/3/faq/general.html#what-is-python> [Ziyaret Tarihi: 03.03.2021].
- [51] Dow, C., 2018, Internet of Things Programming Projects, Packt Publishing, Birmingham UK
- [52] Rolon-Merette, D., Ross, M., Rolon-Merette, T. and Church, K., 2020, Introduction to Anaconda and Python: Installation and setup. *The Quantitative Methods for Psychology*, 16 (5), 3-11.
- [53] Potdar, K., Pardawala, T.S. and Pai, C.D., 2017, Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers. *International Journal of Computer Applications*, (175) 4, 7-9.
- [54] Manajang, D.J., Sompie, S.R. and Jacobus, A., 2020, Implementasi Framework Tensorflow Object Detection Dalam Mengklasifikasi Jenis Kendaraan Bermotor. *Jurnal Teknik Informatika*, 15 (3), 171-178.
- [55] <https://keras.io/about/> [Ziyaret Tarihi:01.03.2021].
- [56] https://keras.io/guides/sequential_model/ [Ziyaret Tarihi:01.03.2021].

- [57] Novak, K. J. and Chamlin, M. B., 2012, Racial threat, suspicion, and police behavior: The impact of race and place in traffic enforcement. *Crime & Delinquency*, 58(2), 275-300.
- [58] Tillyer, R. and Engel, R. S., 2013, The impact of drivers' race, gender, and age during traffic stops: Assessing interaction terms and the social conditioning model. *Crime & Delinquency*, 59(3), 369-395.
- [59] Blalock, G., DeVaro, J., Leventhal, S. and Simon, D. H., 2007, Gender bias in power relationships: evidence from police traffic stops.
- [60] Langton, L. and Durose, M. R., 2013, Police behavior during traffic and street stops, 2011. Washington, DC: *US Department of Justice, Office of Justice Programs, Bureau of Justice Statistics*.
- [61] Lundman, R. J. and Kaufman, R. L., 2003, Driving while black: Effects of race, ethnicity, and gender on citizen self-reports of traffic stops and police actions. *Criminology*, 41(1), 195-220.
- [62] Smith, M. R. and Petrocelli, M., 2001, Racial profiling? A multivariate analysis of police traffic stop data. *Police quarterly*, 4(1), 4-27.
- [63] Engel, R. S. and Calnon, J. M., 2004, Examining the influence of drivers' characteristics during traffic stops with police: Results from a national survey. *Justice Quarterly*, 21(1), 49-90.
- [64] Petrocelli, M., Piquero, A. R. and Smith, M. R., 2003, Conflict theory and racial profiling: An empirical analysis of police traffic stop data. *Journal of criminal justice*, 31(1), 1-11.
- [65] Schafer, J. A., Carter, D. L., Katz-Bannister, A. J. and Wells, W. M., 2006, Decision making in traffic stop encounters: A multivariate analysis of police behavior. *Police Quarterly*, 9(2), 184-209.
- [66] Gaines, L. K., 2006, An analysis of traffic stop data in Riverside, California. *Police Quarterly*, 9(2), 210-233.
- [67] <https://keras.io/api/optimizers/adam/> [Ziyaret Tarihi:12.05.2021].
- [68] <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/> [Ziyaret Tarihi:02.03.2021].
- [69] <https://www.raspberrypi.org/products/camera-module-v2/> [Ziyaret Tarihi:02.03.2021].
- [70] https://thingspeak.com/pages/learn_more [Ziyaret Tarihi:02.03.2021].

EKLER

EK-1 Veri gönderim ve plaka okuma yazılımının entegre hale getirilmesi.

```
import cv2
import pytesseract
from picamera.array import PiRGBArray
from picamera import PiCamera
import sys
import urllib
import urllib.request
import time
from time import sleep

camera = PiCamera()
camera.resolution = (640, 480)
camera.framerate = 30
myAPI = 'VGJQ4EVDC564****' #API KEY tanimi
baseURL = 'https://api.thingspeak.com/update?api_key=%s' % myAPI #analink tanimi
rawCapture = PiRGBArray(camera, size=(640, 480))
time.sleep(1)
text_tmp = " "
for frame in camera.capture_continuous(rawCapture, format="bgr",
use_video_port=True):
    image = frame.array
    cv2.imshow("Frame", image)
    #time.sleep(2)
    textont = None
    text = pytesseract.image_to_string(image)
    print(text)
    if (text != None) and (len(text)>5):
        #textont = text.replace(" ", "")
        textont = ".join(e for e in text if e.isalnum())
        time.sleep(.01)
        conn = urllib.request.urlopen(baseURL + '&field1=%s&field2=%s' % (textont,0))
        print(textont)
        conn.close()
    rawCapture.truncate(0)
    key = cv2.waitKey(1) & 0xFF
    text_tmp = textont
    if key == ord("q"):
        cv2.waitKey(0)
        break
cv2.destroyAllWindows()
camera.close()
```

EK-2 Rastgele seçilen plakaya ait riskli hareketlerin tüm geçmiş sürüşe oranının hesaplanması ve konsol ekranına yazdırılması.

```

from functools import partial
import json
import time
import random
import pandas as pd
import matplotlib as mpl
import matplotlib.pyplot as plt
import warnings
import tensorflow as tf
import tensorflow.keras as keras
import numpy as np
from keras.models import load_model
import json

df=pd.read_csv('yeni.csv')
df_2 = pd.read_csv('tplm_krsk_yeni_plaka.csv')
df_toplam = pd.read_csv('toplamlar_karisik.csv') #
yeni_plakalar = pd.read_csv('plakalar_eklenmis.csv')
df["Meslek"].unique()
df["Durum"].unique()
df_2["Durum"].unique()
X_n = df_2.drop(columns = ['Durum'])
meslek_adlari = ["Hastane_Calisani","MEB_Calisani",
"Kolluk_Kuvveti","Diger_Kamu_Calisani","Ogrenci","Esnaf_Isci","Pazarlamaci","E
mekli"]

model.load_weights('r_durum_son.h5')
gelen_plaka = 0
gelen_plaka_g = 0
df_gecici_2 = pd.DataFrame()
sayac = 0
test = [[0,1]]
test = np.array(test).astype(np.int)

while True:

    yeni_idx = random.randint(0,7941) #150
    gelen_plaka = yeni_plakalar.iloc[yeni_idx,0]

    gc1 = gelen_plaka[0:2]
    gc2 = gelen_plaka[2:3]
    gc3 = gelen_plaka[3:6]
    yaz_plaka = gc1 + " " + gc2 + " " + gc3
    print (yaz_plaka)

    tdurum_gplaka = X_n.loc[X_n['Plaka'] == gelen_plaka]
    t_durum2 = tdurum_gplaka.drop(columns = ['Plaka'])

```

```

t_durum2 = np.array(t_durum2).astype(float)
durum_gplaka = df_2.loc[df_2['Plaka'] == gelen_plaka , 'Durum']

if (gelen_plaka != gelen_plaka_g) and (gelen_plaka < 7945) :
    print (gelen_plaka)
    meslek_gplaka = df_toplam.loc[df_toplam['Plaka'] == gelen_plaka, 'Meslek']
    meslek = np.array(meslek_gplaka).astype(np.int)
    for i in range(len(t_durum2)):
        tahmin=model.predict(t_durum2[i].reshape(1,133))
        t_sayac = np.array(tahmin).astype(np.int)

        if t_sayac[0,1] > t_sayac[0,0]: #test[0,1]:
            sayac = sayac+1
            print("Tahmin sonucu:", tahmin)
            gelen_plaka_g = gelen_plaka
            yuzde = (sayac/len(t_durum2))*100
            print("PLAKA: " , gelen_plaka)
            print("TOPLAM RISKLI HAREKET: " , sayac , "/60")
            print("DURDURMA YUZDESI:" , yuzde)
            print("Meslek: " , meslek[0], " " , meslek_adlari[meslek[0]])

    else:
        print("Tanimsiz Plaka")
        sayac = 0
        time.sleep(2)

```

EK-3 ThingSpeak'ten plaka bilgisinin alınması ve riskli sürüşlerin toplam harekete oranının hesaplanarak konsol ekranına yazılması.

```

from functools import partial
import urllib.request
import json
import time
import random
import pandas as pd
import matplotlib as mpl
import matplotlib.pyplot as plt
import warnings
import tensorflow as tf
import tensorflow.keras as keras
import numpy as np
from keras.models import load_model
import json

df=pd.read_csv('yeni.csv')
df_2 = pd.read_csv('tplm_krsk_yeni_plaka.csv')
df_toplam = pd.read_csv('toplam_karisik.csv')
df["Meslek"].unique()
df["Durum"].unique()
df_2["Durum"].unique()
X_n = df_2.drop(columns = ['Durum'])
meslek_adlari = ["Hastane_Calisani","MEB
Calisani","Kolluk_Kuvveti","Diger_Kamu_Calisani","Ogrenci","Esnaf_Isci","Pazarl
amaci","Emekli"]
model.load_weights('r_durum_son.h5')
gelen_plaka = 0
gelen_plaka_g = 0
df_gecici_2 = pd.DataFrame()
sayac = 0
test = [[0,1]]
test = np.array(test).astype(np.int)
while True:
    TS =
    urllib.request.urlopen("https://api.thingspeak.com/channels/1311320/feeds.json?api_k
ey=HY4X0PIU01W7*****&results=2")
    response = TS.read()
    data=json.loads(response)
    b = data['feeds'][1]
    gelen_plaka = b['field1']
    gelen_plaka = ".join(e for e in gelen_plaka if e.isalnum())
    gelen_plaka = gelen_plaka.replace(" ", "")

    gc1 = gelen_plaka[0:2]
    gc2 = gelen_plaka[2:3]
    gc3 = gelen_plaka[3:6]

```

```

yaz_plaka = gc1 + " " + gc2 + " " + gc3
print (yaz_plaka)
tdurum_gplaka = X_n.loc[X_n['Plaka'] == gelen_plaka]
t_durum2 = tdurum_gplaka.drop(columns = ['Plaka'])
t_durum2 = np.array(t_durum2).astype(float)
durum_gplaka = df_2.loc[df_2['Plaka'] == gelen_plaka , 'Durum']

if (gelen_plaka != gelen_plaka_g) and (gelen_plaka < 7945) :
    print (gelen_plaka)
    meslek_gplaka = df_toplam.loc[df_toplam['Plaka'] == gelen_plaka, 'Meslek']
    meslek = np.array(meslek_gplaka).astype(np.int)
    for i in range(len(t_durum2)):
        tahmin=model.predict(t_durum2[i].reshape(1,133))
        t_sayac = np.array(tahmin).astype(np.int)

        if t_sayac[0,1] > t_sayac[0,0]:
            sayac = sayac+1
            print("Tahmin sonucu:", tahmin)
            gelen_plaka_g = gelen_plaka
            yuzde = (sayac/len(t_durum2))*100
            print("PLAKA: " , gelen_plaka)
            print("TOPLAM RISKLI HAREKET: " , sayac , "/60")
            print("DURDURMA YUZDESI:" , yuzde)
            print("Meslek: ", meslek[0], " " , meslek_adlari[meslek[0]])
        else:
            print("Tanimsiz Plaka")
    sayac = 0
    time.sleep(2)

```

EK-4 Arayüz yazılımı.

```

from functools import partial
from functools import partial
import urllib.request
import json
import time
import random
import pandas as pd
import warnings
import tensorflow as tf
import tensorflow.keras as keras
import numpy as np
from keras.models import load_model
import json
from tkinter import *
import tkinter as tk

df=pd.read_csv('yeni.csv')
df_2 = pd.read_csv('tumduz_plakaveri.csv') # yeni plaka
df_toplam = pd.read_csv('tumduz_plakaveri.csv') #
yeni_plakalar = pd.read_csv('plakalar_eklenmis.csv')
df["Meslek"].unique()
df["Durum"].unique()
df_2["Durum"].unique()
X_n = df_2.drop(columns = ['Durum']) #tplm_krsk_yeni_plaka_durumsuz.csv

Y = df[['Durum']]
meslek_adlari = ["Hastane_Calisani","MEB
Calisani","Kolluk_Kuvveti","Diger_Kamu_Calisani","Ogrenci","Esnaf_Isci","Pazarl
amaci","Emekli"]

model.load_weights('r_durum_son.h5')

gelen_plaka = 0 #gelen plakanin ysa modeline girecek verilerini cek
gelen_plaka_g = 0
df_gecici_2 = pd.DataFrame()
sayac = 0
test = [[0,1]]
test = np.array(test).astype(np.int)

running = True # Global flag

def scanning():
    if running: # Only do this if the Stop button has not been clicked
        global yuzde, t_sayac, durum_gplaka , gelen_plaka, gelen_plaka_g, tavsiye, gc1,
gc2, gc3, yaz_plaka
        global Label, t_durum2, meslek, sayac,tahmin,meslek_gplaka,meslek_adlari,
data, b
        x = root.winfo_pointerx()

```

```

y = root.winfo_pointery()
abs_coord_x = root.winfo_pointerx() - root.winfo_rootx()
abs_coord_y = root.winfo_pointery() - root.winfo_rooty()

entry_show_csr_x.delete(0, tk.END)
entry_show_csr_x.insert(0, abs_coord_x)

entry_show_csr_y.delete(0, tk.END)
entry_show_csr_y.insert(0, abs_coord_y)

TS =
url = "https://api.thingspeak.com/channels/1311320/feeds.json?api_key=HY4X0PIU01W7GG60&results=2"
response = TS.read()
data = json.loads(response)
b = data['feeds'][1]
gelen_plaka = b['field1']

gc1 = gelen_plaka[0:2]
gc2 = gelen_plaka[2:3]
gc3 = gelen_plaka[3:6]
yaz_plaka = gc1 + " " + gc2 + " " + gc3
print(gelen_plaka)
tdurum_gplaka = X_n.loc[X_n['Plaka'] == gelen_plaka] #gelen plakaya ait tum
hareketler cekildi.durum yok
t_durum2 = tdurum_gplaka.drop(columns = ['Plaka']) #plaka silindi
t_durum2 = np.array(t_durum2).astype(float)
durum_gplaka = df_2.loc[df_2['Plaka'] == gelen_plaka , 'Durum'] #orjinal durum

if (gelen_plaka != gelen_plaka_g) : #and (gelen_plaka < 7945) :
    print(gelen_plaka)
    entry_show_bplaka.delete(0, tk.END)
    entry_show_bplaka.insert(0, yaz_plaka)
    meslek_gplaka = df_toplam.loc[df_toplam['Plaka'] == gelen_plaka, 'Meslek']
    meslek = np.array(meslek_gplaka).astype(np.int)
    for i in range(len(t_durum2)):
        tahmin = model.predict(t_durum2[i].reshape(1,133))
        t_sayac = np.array(tahmin).astype(np.int)

        if t_sayac[0,1] > t_sayac[0,0]: #test[0,1]:
            sayac = sayac+1
            #print(sayac)
            print("Tahmin sonucu:", tahmin)

        if sayac > 15 and sayac < 26:
            tavsiye = "Yogunluk Bazli Islem "
            create_yellow()
        elif sayac > 25:
            tavsiye = "Durdur"
            create_red()

```

```

        else:
            tavsiye = "Serbest"
            create_green()

        yuzde = (sayac/len(t_durum2))*100

        entry_show_plaka.delete(0, tk.END)
        entry_show_plaka.insert(0, yaz_plaka)

        entry_show_meslekno.delete(0, tk.END)
        entry_show_meslekno.insert(0, meslek_adlari[meslek[0]])

        entry_show_thareket.delete(0, tk.END)
        entry_show_thareket.insert(0, len(t_durum2))

        entry_show_risklih.delete(0, tk.END)
        entry_show_risklih.insert(0, sayac)

        entry_show_oneri.delete(0, tk.END)
        entry_show_oneri.insert(0, tavsiye)

        entry_show_cikarim.delete(0, tk.END)
        entry_show_cikarim.insert(0, tavsiye)

        ins_lbx = str(yaz_plaka) + " " + tavsiye + " " + meslek_adlari[meslek[0]]
        cr_lbx.insert(0,ins_lbx)

        print("PLAKA: " , gelen_plaka)
        print("TOPLAM RISKLI HAREKET: " , sayac , "/60")
        print("DURDURMA YUZDESI:" , yuzde)
        print("Meslek: " , meslek[0], " " , meslek_adlari[meslek[0]])
        sayac = 0
    else:
        print("Yeni Plaka yok ,Tanimsiz Plaka")
        gelen_plaka_g = gelen_plaka
        root.after(500, scanning)

def start():

    global running
    running = True

def stop():
    global running
    running = False

root = Tk()
root.title("NIYAZI GOKDOGAN")
root.geometry("1440x850")

```

```

root.resizable(False, False)

basla = Button(root, text="Tahmini Surdur", command=start)
durdur = Button(root, text="Programi Duraklat", command=stop)

basla.place(bordermode=OUTSIDE, height=50, width=100, x=0 , y=800)
durdur.place(bordermode=OUTSIDE, height=50, width=120, x=1320 , y=800)

Label(root, text="Gelen Plaka : ", bg="#3300CC", fg="white").place(x=0, y=0)
Label(root, text="MESLEK :", bg="#3300CC", fg="white").place(x=269, y=0)
Label(root, text="TOPLAM HAREKET:", bg="#3300CC", fg="white").place(x=560,
y=0)
Label(root, text="TAHMINI RISKLI TOPLAM HAREKET:", bg="#3300CC",
fg="white").place(x=860, y=0)
Label(root, text="TAVSIYE:", bg="#3300CC", fg="white").place(x=1250, y=0)
Label(root, text="GECMIS PLAKA VE TAVSIYE", bg="#3300CC", fg="white", font
= "Calibri 30").place(x=720, y=45)

entry_gplaka = tk.StringVar()
entry_show_plaka = tk.Entry(root, textvariable=entry_gplaka)
entry_show_plaka.place(x = 77, y=0)

entry_meslekno = tk.StringVar()
entry_show_meslekno = tk.Entry(root, textvariable=entry_meslekno)
entry_show_meslekno.place(x = 325, y=0)

entry_thareket = tk.StringVar()
entry_show_thareket = tk.Entry(root, textvariable=entry_thareket)
entry_show_thareket.place(x = 670, y=0)

entry_risklih = tk.StringVar()
entry_show_risklih = tk.Entry(root, textvariable=entry_risklih)
entry_show_risklih.place(x = 1058, y=0)

entry_oneri = tk.StringVar()
entry_show_oneri = tk.Entry(root, textvariable=entry_oneri)
entry_show_oneri.place(x = 1303, y=0)

entry_csrxx = tk.StringVar()
entry_show_csrxx = tk.Entry(root, textvariable=entry_csrxx)
entry_show_csrxx.place(x = 575, y=825)

entry_csrxy = tk.StringVar()
entry_show_csrxy = tk.Entry(root, textvariable=entry_csrxy)
entry_show_csrxy.place(x = 725, y=825)

entry_bplaka = tk.StringVar()
entry_show_bplaka = tk.Entry(root, textvariable=entry_bplaka)
entry_show_bplaka.place(x = 125, y=140, width = 250 , height = 100)

```

```

entry_bplaka = tk.StringVar()
entry_show_bplaka = tk.Entry(root, textvariable=entry_bplaka , font = "Calibri 75",
justify='center',background='SystemButtonFace')
entry_show_bplaka.place(x = 50, y=140, width = 400 , height = 100)

entry_cikarim = tk.StringVar()
entry_show_cikarim = tk.Entry(root, textvariable=entry_cikarim , font = "Calibri 30",
justify='center',background='SystemButtonFace')
entry_show_cikarim.place(x = 75, y=610, width = 350 , height = 100)

cr_lbx = Listbox(root, height = 15, width= 46,bg = "SystemButtonFace", font =
"Calibri 25")
cr_lbx.place(x=550, y=100)
def create_red():
    entry_red = tk.StringVar()
    entry_show_red = tk.Entry(root, textvariable=entry_red, background="red")
    entry_show_red.place(x = 75, y=250, width = 350 , height = 350)

def create_green():
    entry_green = tk.StringVar()
    entry_show_green = tk.Entry(root, textvariable=entry_green, background="green")
    entry_show_green.place(x = 75, y=250, width = 350 , height = 350)

def create_yellow():
    entry_yellow = tk.StringVar()
    entry_show_yellow = tk.Entry(root, textvariable=entry_yellow,
background="yellow")
    entry_show_yellow.place(x = 75, y=250, width = 350 , height = 350)

root.after(500, scanning) # After 1 second, call scanning
root.mainloop()

```