

**SEYİR HALİNDE KURAL İHLALİ YAPAN  
ARAÇLARIN PLAKALARININ BELİRLENMESİ**

**2012  
YÜKSEK LİSANS TEZİ  
BİLGİSAYAR MÜHENDİSLİĞİ ABD**

**Ahmet Nusret ÖZALP**

**SEYİR HALİNDE KURAL İHLALİ YAPAN ARAÇLARIN  
PLAKALARININ BELİRLENMESİ**

**Ahmet Nusret ÖZALP**

**Karabük Üniversitesi  
Fen Bilimleri Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalında  
Yüksek Lisans Tezi  
Olarak Hazırlanmıştır**

**KARABÜK**

**Nisan 2012**

Ahmet Nusret ÖZALP tarafından hazırlanan “SEYİR HALİNDE KURAL İHLALİ YAPAN ARAÇLARIN PLAKALARININ BELİRLENMESİ” başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Yrd.Doç.Dr.Salih GÖRGÜNOĞLU

Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile Bilgisayar Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 18/04/ 2012

Ünvanı, Adı SOYADI (Kurumu)

İmzası

Başkan : Yrd. Doç. Dr. İlhami Muharrem ORAK (KBÜ)

Üye : Yrd. Doç. Dr. Salih GÖRGÜNOĞLU (KBÜ)

Üye : Yrd. Doç. Dr. Mustafa AKTAŞ (KBÜ)

...../...../2012

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile Yüksek Lisans derecesini onamıştır.

Prof. Dr. Nizamettin KAHRAMAN

Fen Bilimleri Enstitüsü Müdürü

*“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”*

Ahmet Nusret ÖZALP

## **ÖZET**

**Yüksek Lisans Tezi**

### **SEYİR HALİNDE KURAL İHLALİ YAPAN ARAÇLARIN PLAKALARININ BELİRLENMESİ**

**Ahmet Nusret ÖZALP**

**Karabük Üniversitesi**

**Fen Bilimleri Enstitüsü**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Tez Danışmanı:**

**Yrd. Doç. Dr. Salih GÖRGÜNOĞLU**

**Nisan 2012, 59 sayfa**

Trafik kazalarının bir sebebi de sürücü hatalarıdır. Hata yapan sürücülerin kullandığı araçların plakalarının doğru bir şekilde tespit edilmesi, hızlarının belirlenmesi önem arz eder. Bu çalışmada, öncelikli olarak görüntü işleme prensipleri hakkında bilgi verilmekte, araçların plakalarının belirlenmesi gerçek zamanlı hızlarının ölçülmesi, istenen güzergah üzerindeki araç sayısının hesaplanması ve araç yoğunluğunun belirlenmesi gerçekleştirilmiştir.

Araçların hızlarının ölçülmesinde çok çeşitli yollar kullanılmaktadır. Bunların en yaygın olanı trafik araçları üzerinde bulunan sabit yada mobil radar teknolojileridir. Günümüzde gerçek zamanlı görüntü alan aygıtlar vasıtasıyla kurulan sistemlerle de hız ölçüm işlemi yapılmaktadır.

Araç sayısının belirlenmesi, trafik yoğunluğunun öngörülmesi noktasında önemlidir. Böylece trafik denetimlerinin yapılacağı istikametlerin belirlenmesi, bu denetimlerin

ne şekilde hangi zaman aralıklarında yapılacağına ilişkin sonuçlar elde edilebilecektir. Günümüzde manuel yada elektronik sistemlerle yapılan bu işlemlerin, gerekli donanım alt yapısı hazırlanarak bilgisayar ortamında yapılması, bu tezde geliştirilen yazılımla gerçekleştirilmiştir.

**Anahtar Sözcükler** : Plaka tanıma, hız tespiti, araç sayımı, trafik yoğunluğu.

**Bilim Kodu** : 902.1.014

## **ABSTRACT**

**M.Sc. Thesis**

### **DETERMINING THE LICENSE PLATES WHICH BREAKS THE TRAFFIC RULES WHILE MOVING**

**Ahmet Nusret ÖZALP**

**Karabük University  
Graduate School of Natural and Applied Sciences  
Department of Computer Engineering**

**Thesis Advisor:**

**Asst.Prof.Dr.Salih GÖRGÜNOĞLU**

**April 2012, 59 pages**

One of the reasons of traffic accident is driver's failure. It is essential that Driver who makes mistake while driving should be identified correctly and their speed should be determined. There is a knowledge about principal, determining licence numbers; accounting their real time speed counting the number of the vehicles in the location and vehicle density is actualized in this study. There are many ways used for accounting speed. The most common one is located on the traffic vehicles that are fixed or mobile radar technology.

Nowadays, Thanks to real-time equipments you can evaluate speed. Determining the number of the cars, and projecting the traffic density is essential. So, Determining the direction of the traffic control, some results will be obtained in this controls.

Nowadays these processes are done by manual or electronic systems, in this thesis required hardware infrastructure is prepared in order to do in computer environment.

**KeyWords** : Plate recognition, speed detection, vehicle counting, traffic density.

**ScienceCode** : 902.1.014



## TEŞEKKÜR

Bu tez çalışmasının planlanmasında, araştırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteğini esirgemeyen, bilgilendirmeleriyle çalışmamı bilimsel temeller ışığında şekillendiren Sayın Hocam Yrd.Doç.Dr. SalihGÖRGÜNOĞLU'na sonsuz teşekkür ederim.

Sahip olduğu bilgi ve birikimi ile eğitim ve mühendislik alanında yüksek lisans eğitimi sürecince yardımcı olan Sayın Yrd. Doç. Dr. Baha ŞEN'e teşekkür ederim.

Lisans tezimde olduğu gibi bu çalışmamda da göstermiş olduğu sabır ve desteği için sevgili eşim Nesrin ÖZALP'eşükranlarımı sunarım.

## İÇİNDEKİLER

	<u>Sayfa</u>
KABUL .....	ii
ÖZET .....	iv
ABSTRACT .....	vi
TEŞEKKÜR .....	viii
İÇİNDEKİLER.....	ix
ŞEKİLLER DİZİNİ .....	xi
ÇİZELGELER DİZİNİ .....	xiii
SİMGELER VE KISALTMALAR DİZİNİ.....	xiv
BÖLÜM 1. ....	1
GİRİŞ .....	1
BÖLÜM 2. ....	5
KARAYOLLARI VE TRAFİK KAZALARI.....	5
2.1. TRAFİK KAVRAMI.....	5
2.2. TRAFİK KAZALARI .....	5
2.3. TRAFİK KAZALARININ SEBEPLERİ.....	6
2.4. MEYDANA GELEN KAZA SONUÇLARI .....	8
BÖLÜM 3. ....	11
PLAKA TANIMA SİSTEMLERİ.....	11
3.1. GÖRÜNTÜ İŞLEME .....	12
3.2. GÖRÜNTÜ İŞLEME TEMELLERİ.....	12
3.2.1. Görüntü Tanımı.....	12
3.2.2. Renk Modelleri .....	14
3.3. RENK DÖNÜŞÜMLERİ.....	16
3.4. GERÇEK RENK VE GRİ RENK SKALA DÖNÜŞÜMLERİ .....	16
3.5. SİYAH BEYAZ GÖRÜNTÜ DÖNÜŞÜMÜ .....	17

	<b><u>Sayfa</u></b>
3.6.GÖRÜNTÜ İŞLEME İLE PLAKA BÖLGESİNİN BULUNMASI.....	20
3.6.1. Süzgeç Algoritmaları.....	20
3.6.2. Komşuluk Bölgelerinin Belirlenmesi.....	21
3.6.3. Görüntü Pekiştirme .....	22
3.6.4. Karakter Tanıma .....	23
3.7.PLAKA TANIMA AŞAMALARI .....	23
3.8.YAPAY SİNİR AĞLARI .....	27
3.8.1.Yapay Sinir Ağı Modeli Çok Katmanlı Algılayıcılar .....	27
3.8.2. Öğrenme Kuralı .....	29
3.8.2.1. İleri Doğru Hesaplama .....	29
3.8.2.2. Geri Doğru Hesaplama .....	30
3.8.2.3. Ara Katman ile Çıktı Katman Ağırlıkların Değiştirilmesi.....	31
3.8.2.4. Çok Katmanlı Algılayıcılar Ağının Çalışması .....	33
3.8.2.5. Çok Katmanlı Algılayıcılarda Ağın Eğitilmesi .....	34
 BÖLÜM 4. ....	 36
PLAKA TANIMA HIZ ÖLÇME VE ARAÇ YOĞUNLUĞUNUN TESPİTİ.....	36
4.1.PLAKA TANIMA VE UYGULANAN YÖNTEM.....	36
4.2.SEYİR HALİNDEKİ ARACIN TESPİTİ VE HIZ ÖLÇÜMÜ .....	37
4.2.1.Gerçek Zamanlı Araç Tanımlama ve Hareket Belirleme .....	41
4.2.2.Yol Bölümleme (LaneMasking) İşlemi.....	42
4.2.3.Hareket Arka Plan Çıkarma İşlemi .....	42
4.2.4.Blob Algoritması Uygulama İşlemi .....	44
4.2.5.Araç Hareketi.....	45
4.2.5.1.Araç Tabanlı Hareket Algılama İşlemi .....	45
4.3. DENEY SONUCUNDAKİ VERİLERİN DEĞERLENDİRİLMESİ.....	48
 BÖLÜM 5. ....	 55
SONUÇLAR.....	55
 KAYNAKLAR.....	 57
ÖZGEÇMİŞ.....	59

## ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 2.1.Trafik kazalarının illere göre dağılımı.....	7
Şekil 2.2.Trafik kazası ölü sayılarına göre dağılımı .....	7
Şekil 2.3.Trafik kazası yaralı sayılarına göre dağılımı .....	8
Şekil 3.1.Sayısal resim ve görüntü matrisi.....	13
Şekil 3.2.Renkli görüntü .....	13
Şekil 3.3.Olgusal renk bölgesi.....	15
Şekil 3.4.Çekirdek süzgeç fonksiyonu.....	21
Şekil 3.5.Çekirdek fonksiyon uygulama mantığı .....	21
Şekil 3.6.Konvolüsyon işlemi.....	23
Şekil 3.7.Plaka ön işlem aşamaları .....	24
Şekil 3.8.Plaka kontrol aşaması.....	25
Şekil 3.9.Plaka bilgisinin okunması ve kayıt edilmesi aşaması .....	25
Şekil 3.10.Görüntü ön işlem sonuçları.....	25
Şekil 3.11.Geriye yayılım algoritması .....	28
Şekil 4.1.Uygulama ekran çıktısı.....	38
Şekil 4.2. Araç hareketinin çıkarılması.....	40
Şekil 4.3.Gerçek zamanlı görüntü alma .....	41
Şekil 4.4.Hareketli araçların belirginleştirilmesi.....	41
Şekil 4.5.Hareket tespiti .....	43
Şekil 4.6.Orta nokta bulma fark ve tolerans değerleri .....	43
Şekil 4.7.Sobel algoritması sonucu elde edilen görüntü.....	44
Şekil 4.8.Logedge bulma algoritması sonucu elde edilen görüntü.....	44
Şekil 4.9.Sobel algoritması yatay ve dikey maskeleri .....	45
Şekil 4.10.Uygulamada kullanılan ipcam .....	46
Şekil 4.11.Mobile dvr kayıt cihazı.....	47
Şekil 4.12.Hata oranı tespiti .....	50
Şekil 4.13.Blob algoritması ve contour algoritması sonuçları .....	51
Şekil 4.14.Ölçüm alanının belirlenmesi.....	52
Şekil 4.15.Ölçüm sonuçlarının karşılaştırılması.....	52

	<b><u>Sayfa</u></b>
Şekil 4.16.Hareket eden nesnelerin id işlemleri .....	53
Şekil 4.17.Araç sayılarının trafik yoğunluğu verisinde kullanılması .....	54

## ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
Çizelge 2.1. Yıllara göre genel kaza istatistikleri.....	9
Çizelge 2.2. 2009-2010 yıllarında trafikte uygulanan ceza sayıları .....	9
Çizelge 4.1. Nesne algı süre ölçümleri .....	40
Çizelge 4.2. Uygulamada kullanılan ipcam özellikleri.....	46
Çizelge 4.3. Mobil dvr kayıt cihazı teknik özellikleri .....	47
Çizelge 4.4. Hız ölçüm sonuçları.....	51
Çizelge 4.5. Safranbolu Karabük karayolu araç sayısı tespiti.....	53

## SİMGELER VE KISALTMALAR DİZİNİ

### SİMGELER

$T$	: Eşik Değeri
$f_i$	: $i$ renk sayısına sahip toplam piksel sayısı
$N$	: Toplam piksel sayısı
$p_i$	: Görüntüdeki her bir gri seviye renk değerinin bulunma olasılığı
$\mu_1$	: $T$ 'nin altında kalan sınıfların ortalama olasılık değeri:
$\mu_2$	: $T$ 'nin üstündesınıfların ortalama olasılık değerleri:
$\mu_T$	: Toplam olasılık değeri
$\sigma^2$	: Sınıf varyans değeri
$I(u, v)$	: Gerçek görüntü noktaları
$\hat{I}(u, v)$	: Filtre uygulanan görüntü noktaları
$H(i, j), H$	: Görüntüye uygulanacak olan kernel
$A_{kj}$	: $k$ girdisi bağlantı ağırlık değeri
$C_k^i$	: İşlem elemanı çıktısı
$NET_j^a$	: Net girdi değeri
$\beta_j^a$	: Eşik değeri
$E_m$	: İşlem elemanı hata oranı
$TH$	: Toplam Hata
$\lambda$	: Öğrenme Katsayısı
$\alpha$	: Momentum Katsayısı
$\delta_m$	: $m$ çıktı ünitesinin hata değeri

- $f'$  (NET) : Aktivasyon türevi  
 $\delta_j^a$  : Hata Terimi  
 $A(i, j)$  : Görüntü matrisi  $i$  ve  $j$ .ci elemanları  
 $f(x,y)$  : Görüntü modeli  
 $V_{ort}$  : Ortalama Hız  
 $X_{son}, X_{ilk}$  : Yer değiştirme  
 $t$  : Zaman  
 $n$  : Toplam piksel sayısı  
 $\Delta A^a$  : İşlem elemanının çıktı katmanındaki ağırlık değişim miktarı



## KISALTMALAR

TV	: Televizyon
RGB	: Red, Green, Blue (Kırmızı, Yeşil, Mavi)
DPI	: Dot Per Inch
PPI	: Pixels Per Inch
CMYK	: Cyan, Magenta, Yellow, Key
HSI	: Hue, Saturation, Intesity
HSB	: Hue, Saturation, Brightness
CMY	: Cyan, Magenta, Yellow
YIQ	: Görüntü Bilgi ve Renkliliği
YUV	: Açıklık ve Renk Alanı
NTSC	: National Television Standarts Committee
C.P.U.	: Merkezi İşlem Birimi
GOOCR	: Open Source Optical Character Recognition
TÜVTURK	: Araç Muayene İstasyonları
SQL	: Structured Query Language
YSA	: Yapay Sinir Ağları
XOR	: Exclusive OR
HVS	: Human Visual System
CIE	: Commision Internationale de L'Eclairage
LPR	: License PlateRecognition
OCR	: Optical Character Recognition
HLS	: Hue Lightness Saturation
OPOS	: Otomatik Plaka Tanıma Sistemi
OGS	: Otomatik geçiş sistemi
SOM	: Self Organizing Map
FBA	: Fast Backpropagation Algorithm
MSE	: Ortalama Karesel Hata
MERNİS	: Merkezi Nüfus İdaresi Sistemi
CCTV	: Close Curcuit TV Systems
IPCAM	: Internet Protocol Network Camera
ASCII	: American Standard Code for Information Interchange

MJPEG	: Moving Pictures Experts Group
IEEE	: The Institute of Electrical and Electronics Engineers
MBPS	: Megabit per second.
WEP	: Wired Equivalent Privacy
WPA	: Wi-Fi Protected Access
WPA2	: Wi-Fi Protected Access 2
WVGA	: Wide Video Graphics Array
FSB	: Front Side Bus
RAM	: Random Access Memory
CMOS	: Complementary Metal Oxide Semiconductor
QVGA	: Quarter Video Graphics Adapter
DVR	: Digital Video Recorder
VGA	: Video Graphics Adapter
LCD	: Liquid Crystal Display
ID	: Identity
FPS	: Frame Per Second
EDS	: Elektronik Denetim Sistemi

## BÖLÜM 1

### GİRİŞ

Görüntü İşleme, bilgisayar alanında hemen her ana bilim dalı ile tamamlayıcı ve ayırt edici özellikleri ile günümüzde önemli bir çalışma alanı olarak karşımıza çıkmaktadır. Günlük hayatta karşımıza çıkan anlık görüntülerin, tasarlanan sisteme giriş nesnesi olarak kullanılması suretiyle, o girdi doğrultusunda yeni veriler elde edilmesi işlemi olarak tanımlanabilir. Dijital ortama alınan verilerin istenen kriterler doğrultusunda işlenir hale gelmesi aşamalarında çok çeşitli algoritma ve yaklaşımlar kullanılır. Bu çalışmada öncelikle grafik ve görüntü üzerinden araçların plakalarının okunması; video görüntüleri üzerinden hareket eden nesnelerin tespit edilmesi; seyir halinde tehlikeli seyir eden araçların hızlarının tespit edilmesi işlemleri yapılmıştır. Bu işlemlerde temel görüntü işleme kuralları, günümüz programlama dillerine uyarlanarak, sistemlerin verimlilikleri arttırılmıştır. Bunun yanında araçların seyir ettiği istenilen bir yol üzerinde çevrimiçi olarak kamera vasıtasıyla yolun yoğunluğunun tespiti, yoldan geçen araç sayılarının sayılması suretiyle yapılmıştır.

Böylece elde edilen araç sayısından yolun yoğunluk analizi yapılarak, yoldaki trafik durumunun gerçek zamanlı olarak izlenmesi sağlanmıştır. Bu veriler yolun tadilat sürecini, yoldaki yoğunluğa bağlı olarak trafik kaza olasılığı gibi sonuçlara ulaşılması noktasında bir karar destek sistemi olarak karşımıza çıkmıştır. Görüntü işleme aşamaları plaka tanıma sistemi ile başlayıp, araç sayısı işlemi ile tamamlanmıştır. Otomatik Araç Tanıma konusunda ilk gelişme optik tarayıcı sistemlerin kullanılmasıyla 1970'lerde ABD'de gerçekleşmiştir. Daha sonraki yıllarda mikro elektronikteki gelişmeler çalışmaları yoğunlaştırmış, indüktif döngü, radyo frekansları, kızılötesi ve mikrodalga sistemleri araç gözetleme ve trafik kontrolünde kullanılmıştır. Araç tanıma teknolojisinin yol ücretlendirilmesi için denenmesi ilk olarak Hong Kong'da 1983-1985 tarihleri arasında uygulanmıştır. Motorlu araç plaka tanıma sistemleri, otomatik araç tanıma ve trafik akış kontrolü

amacıyla kullanılan sistemlerdir. Paralı otoyol ve köprü gişeleri, hastane, askeri tesis, otopark girişleri ve trafik ışık ihlalleri, şerit ihlalleri, hatalı sollama vb. tespitinde uygulama bulmaktadır. Önce gönüllü bir araç grubu elektronik plakalarla donatılmıştır. Sonra bu plakalara sahip araçlar yoldan geçtiğinde araçların fotoğrafları kapalı devre TV(Televizyon) ile alınmış ve kontrol merkezine iletilmiştir. Araç Tanıma Sistemine yönelik çalışmalar özellikle Avrupa’da altı ülkeyi kapsayan “Prometheus” adı verilen bir proje ve Japonya’da da buna benzer bir projeye devam etmektedir. Bu programlar araçları takip, elektronik ceza kesme, plaka tanıma ve araç tanımayı içermektedir [1].

Günümüzde devam eden bu tür projelerle güvenliği, verimi ve konforu arttırmak, ekonomik çözümler üretmek, kirliliği azaltmak amaçlanmaktadır. İlk plaka okuma sistemleri çok küçük başarı yüzdeleriyle çalışmaktaydılar. Bu sebeple pratikte kendilerine bir kullanım alanı bulamamışlardır. Bu sistemler dış etkenlerden (Güneş ışığı, trafik ışıkları, farlar, bozuk plakalar, plaka numaralarındaki hatalar v.b.) kolay etkilenmekte, donanımsal yetersizliklerden dolayı da sık sık problemler ortaya çıkarmaktaydı.

Zaman içinde gelişen yazılım ve donanımlar plaka tanıma sistemlerinin daha tatminkâr sonuçlar vermesine ve hayatın içine daha gerçekçi şekillerde girmesine yol açmıştır. Günümüzde sanayi ve endüstrinin birçok dalında benzer teknolojilerin kullanıldığı görülebilmektedir. Gittikçe güvenilir hale gelen bu sistemler artık günlük yaşantımıza kadar girmeyi başarmıştır. Hatta birçok sorunun çözümünde manyetik kartlar ve okuyucularla beraber kullanılmaktadırlar. Fakat plaka tanıma sistemleri, ilgili tüm gereksinimleri diğer ürünlere ihtiyaç duymadan karşılayabilecek seviyeye ulaşmıştır.

Eğer değerlendirme evresi herhangi bir nedenden ötürü kusursuz olarak gerçekleşmezse yazılım belirli kıstaslar içerisinde akışı engellemek adına bazı hataları tolare edebilmektedir. Örnek olarak giriş kaydı olan bir aracın çıkış zamanında olması gereken zamanlamaya göre çok küçük bir sapma meydana gelirse program bu sorundan dolayı hata vermeyip akışın devam etmesi için küçük

toleransları gösterebilmektedir. Bu küçük zeki yaklaşımla sistem her küçük hatada tıkanmayıp devamlılığı sağlamayı amaçlamaktadır.

Son çeyrek yüzyılda araç sayısının artması ve trafikte oluşan sorunlar, otomatik araç tanıma ve trafik akışının kontrolü üzerine çalışmalar yapılmasını gerektirmiştir. Bunun yanında özellikle giriş çıkış kontrollerinin, güvenlik önlemleri açısından öneminin artması, motorlu araçların gerçek zamanlı takibini de beraberinde getirmiştir. Asayiş ve trafik hizmetlerindeki kalite politikalarının arttırılması, artan araç sayısını da göz önüne aldığımızda kurumları bilişim teknolojilerinden daha fazla alanda yararlanmalarını sonucuna getirmiştir. Artık sadece güvenlik güçlerinin değil, kamuda, özel sektörde, insanların ortak kullanım alanlarında da motorlu araçların kontrolünde plaka tanıma sistemlerinin kullanımı yaygınlaşmıştır.

Görüntü işlemenin bir alt dalı olarak kabul edilen plaka tanıma sistemlerinde plakanın bulunduğu bölgenin tespit edilmesinde çeşitli yöntemler kullanılır. Bunun yanında yapay zekânın temel prensiplerinde de yararlanılmaktadır. Özellikle bilgisayar tabanlı uygulamalarda, kendine yer bulan Yapay Zekâ prensipleri lineer olmayan sistemlerin çözüm kümelerinin bulunmasında kendine yer bulmaktadır. Plaka tanımlama sistemlerinde görüntü işleme aşamalarının ardından, karakter analizi aşamasında yapay zekâ devreye girer.

Bu çalışmada mevcut karakter tanıma analizleri test edilerek, verimliliğinin arttırılması noktasında testler yapılmıştır. Mevcut sistemlerin ticari uygulamalarında, tanıma işlemlerinde kullanılan yazılım ve donanım aygıtlarının maliyetlerinin yüksek olması dolayısıyla sistemin uygulanmasında maliyet bir problem olarak karşımıza çıkmaktadır. Çözünürlüğü yüksek bir donanımsal kamera aygıtı ile yine yapılandırma ve performansı yüksek bir bilgisayar sistemi ile çalışan plaka tanıma sistemlerine alternatif olarak; düşük çözünürlüklü bir web kamera ve ortalama bir bilgisayar konfigürasyonunda çalışan yazılım ile bu sağlanmıştır. Özellikle tanımlanan plakaların saklanması noktasında tasarlanan veri tabanı sistemi ile verilerin güvenli bir şekilde saklanması ve hızlı bir şekilde okunduğu görülmüştür.

Gerçek zamanlı araçların hızlarının tespit edilmesi genelde radar sistemleri ile sağlanmaktadır. Mobil ve sabit olarak kullanılan bu sistemlerde, özellikle Amerika'da motorize polislerde daha mobilitesi yüksek sistemler kullanılmaktadır. Bu sistemlerin en büyük eksikliği sistemin manuel ayağının insan eliyle yapılması şartıdır. Gözlem ve tespit noktasında kullanılan cihazlar insan kanalıyla kullanılmakta ve kontrol edilmektedir. Gelişen teknolojiye bağlı olarak, artık yüksek hızda veri iletim bantlarına bağlı kamera sistemleri ile bu işlemler çok rahat yapılabilmektedir. İnsan merkezli yapılan bu tespit çalışmaları tasarlanan mimarilerle otomatik hale getirilebilmektedir. Bu durum insan faktörünü aradan çıkararak, doğruluk ve devamlılık noktalarında artılar katmaktadır. Çünkü insan merkezli olan sistemlerde genel olarak, birey bazlı problemler ortaya çıkabilmektedir.

Kamera ile tek bir merkezden tespit işlemi yapılarak, trafikte problem riski doğan noktalar analiz edildikten sonra gerekli önlemler alınabilir. Tespit işlemi sırasında, araçların tespit noktasında olma zorunlulukları kalkarak, tasarruf edilmesi sağlanabilir. Seyir halinde tehlikeli hareket eden sürücülerin belirlenmesi ile ülkemizde ve dünyadaki en büyük terörlerden olan trafik terörü riskini azaltma amacı güdülmüştür.

Gerçek zamanlı olarak, gerekli donanımsal mimarilerin tasarlanması ve montaj ile de bu tehlikeli seyir eden araçlar plaka tespit yoluyla belirlenebilir. Bu çalışmada bunun alt yapısının hazırlanması ve uygulanması sağlanmıştır. Video görüntüleri ile araçların kamera menzili içerisinde araç hızlarının tespit edilmesi işlemleri tamamlanmıştır.

## BÖLÜM 2

### KARAYOLLARI VE TRAFİK KAZALARI

Sosyal bir varlık olan insan, yüzyıllar boyunca doğayı anlamaya, yaşamını kolaylaştırmak için çalışmış; günümüzdeki teknolojik gelişmelerle de her geçen gün bu amacına ulaşmaya devam etmektedir. Artık zamanın değerini daha iyi anlayan insanoğlu için ulaşım faaliyetlerini önemi artmıştır. Kara ve denizyolları ile başlayan ulaşım ve seyahat faaliyetlerine günümüzde havayolu da eklenmiştir. Artan nüfusa paralel olarak, karayolu üzerinden yapılan bu yer değiştirmelerin sayısı, diğer alternatiflerine göre daha yoğundur. Bu yoğunluk beraberinde trafik kazalarını getirmektedir. Bu kazaların büyük bir kısmı can ve mal kaybıyla sonuçlanmaktadır. Yapılan bu çalışma ile karayollarında meydana gelen aşırı hız yapan araçların belirlenmesi amacı güdülmüştür.

#### 2.1. TRAFİK KAVRAMI

2918 Karayolları Trafik Kanunu'nun 3. Maddesine göre trafik; "İnsanların, hayvanların ve araçların karayolları üzerindeki hal ve hareketleridir." Dünya üzerinde artan nüfus sayısı, teknolojik gelişmelere paralel olarak artan gelişmişlik düzeyi trafikte seyir eden araçların sayısını arttırmaktadır. Artan araç sayısına bağlı olan trafik yoğunluğu beraberinde maddi ve manevi kayıplı kazaları da getirmektedir. Bu durum hem insanların yaşam kalitelerini hem de ülkelerin ekonomilerini olumsuz yönde etkilemektedir.

#### 2.2. TRAFİK KAZALARI

Trafikte, insanların kullanıma açık olan araziler, köprüler, alanlar karayolu kavramı olarak karşımıza çıkmaktadır. Karayolların bakım onarım çalışmalarına bağlı olarak, trafiğin güvenli ve düzenli şekilde akmasının sağlanması önemli bir parametredir.

Karayolu üzerinde hareket halinde olan araç yâda araçların sebep olduğu sonucunda yaralanma ve ölümlle sonuçlanan olaylara “Trafik Kazası” denir. Emniyet Genel Müdürlüğü’nün yapmış olduğu istatistikî çalışmalara bakıldığında, trafik kazalarının başlıca aktörünün insan olduğu görülmektedir. Diğer sebepler ise aşağıdaki gibidir [2].

1. İnsan Faktörü %66
2. Sürücü %27
3. Yolcu %1
4. Araç Faktörü %5
5. Yol Faktörleri %1

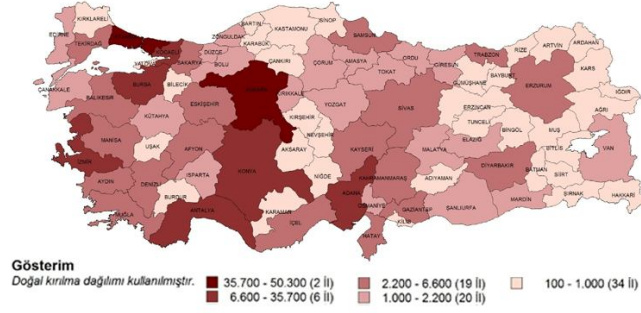
Burada gözden kaçırılmayacak kadar yüksek bir oranda insan faktörünün trafik kazalarında başrolü oynaması, insan kaynaklı sebeplerin araştırılmasını gerektirmektedir. Yine Emniyet Genel Müdürlüğü tarafından yapılan araştırmada insana bağlı sebeplerin başında trafikte yeterli tecrübe ve bilgi birikime sahip olmayan sürücüler gelmektedir.

### **2.3. TRAFİK KAZALARININ SEBEPLERİ**

Eğitimli bireylerin yetiştirilmesi ve trafikte sorumluluk duygusu ile hareket eden sürücülerin arttırılması için bir çalışma yapılması gerektiği açıktır. İnsana dayalı faktörlere bakıldığında aşırı hızın da kazaların nedenleri arasında olduğunu görüyoruz. Bu çalışmada işte bu noktada trafikte yapılabilecekler üzerinde durulmaktadır. Seyir halinde trafikte tehlikeli hareket eden araçların belirlenerek en kısa sürede ilgili merkezlere bildirilip, trafik kazasına sebep vermeden uyarılmasına dönük bir amaç güdülmüştür. Bunun yanında trafik yoğunluğunun yüksek olduğunun tespiti edilen noktalarda denetim raporları hazırlanarak, denetim faaliyetlerinin o noktalara kaydırılması gerektiren karar destek sistemleri geliştirilmesi amaçlanmıştır. Kazalara sebep olan sebepler aşağıdaki gibidir [2]. Şekil 2.1’de Türkiye’de trafik kazalarının illere göre dağılımı verilmiştir.

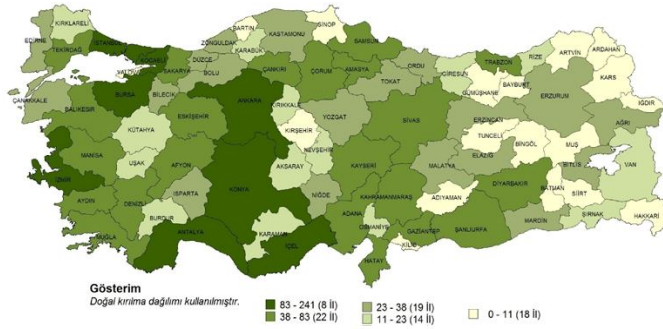


1. Acemilik,
2. Dikkatsizlik,
3. Uzun süre uykusuzluk
4. Hatalı Sollama
5. Aşırı Hız
6. Fazla yük taşımak
7. Alkollü Araç Kullanımı
8. Bazı ilaçları kullandıktan sonra araç kullanmak
9. Trafik kurallarını dikkate almamak
10. Rutin araç bakımlarını yaptırmamak



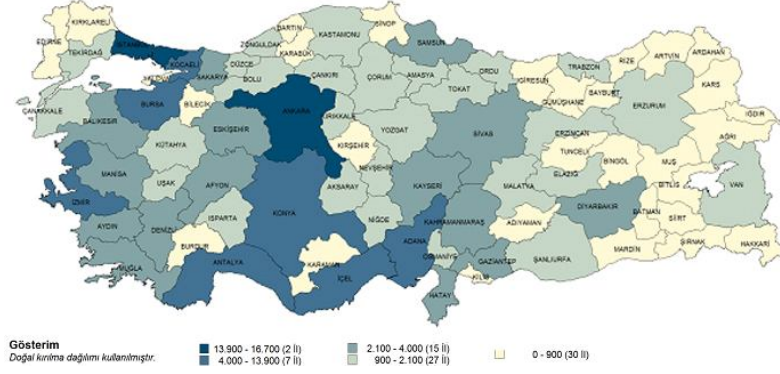
Şekil 2.1. Trafik kazalarının illere göre dağılımı [2].

Emniyet Genel Müdürlüğü tarafından yapılan istatistikî çalışmalardan görüldüğü üzere, ölüm ve yaralanma ile sonuçlanan kazaların meydana geldiği coğrafi bölgelere bakıldığında arazinin düz ve nüfus yoğunluğunun fazla olduğu bölgeler yer almaktadır. Şekil 2.2’de ölümlle sonuçlanan trafik kazalarının dağılımı görülmektedir.



Şekil 2.2. Trafik kazası ölü sayılarına göre dağılımı [2].

Düz arazilerde yüksek hız yapan araçların karıştığı kazaların oranı da mevsimsel olarak bakıldığında gün geçtikçe arttığı görülmektedir. Şekil 2.3’de bölgelere göre ortaya çıkan yaralanma sayılarının dağılımı görülmektedir.



Şekil 2.3. Trafik kazası yaralı sayılarına göre dağılımı [2].

#### 2.4. MEYDANA GELEN KAZA SONUÇLARI

Karayolu içersinde, insan faktörünün başlıca rol oynadığı kazalar her yıl milyarlarca dolarlık maddi zararlar ülke ekonomilerini olumsuz yönde etkilemektedir. Yapılan eğitim çalışmaları, yazılı ve görsel medya aracılığı ile yapılan uyarı uygulamalarına rağmen, hala aşırı hızla hareket eden, insan hayatını hiçe sayan sürücülerin sayısı oldukça fazladır. Burada en temel caydırıcı unsur para cezalarıdır. Uygulanan para cezaları arasında en yüksek sayı ve tutarla aşırı hız gelmektedir.

Elde edilen veriler ışığında görülüyor ki, insan gibi değerli bir varlığın kendi yapmış olduğu aygıtlar yüzünden her yıl dünyada ve ülkemizde hayatını kaybetmesi üzücü bir durumdur.

Bunun yanında sakat kalan insan sayısı da dünya nüfusu ile karşılaştırıldığında küçümsenmeyecek seviyededir. Maddi sonuçları da ülkelerin ekonomilere olumsuz etki etmektedir. Çizelge 2.1’de yıllara göre genel kaza istatistikleri verilmiştir. Çizelge 2.2’de ise 2009-2010 yıllarında trafikte uygulanan ceza sayıları verilmiştir.

Çizelge 2.1. Yıllara göre genel kaza istatistikleri [2].

<b>GENEL KAZA İSTATİSTİKLERİ</b>			
<b>Yılı</b>	<b>Kaza Sayısı</b>	<b>Ölü Sayısı</b>	<b>Yaralı Sayısı</b>
2001	442.960	4.386	116.202
2002	439.958	4.169	116.045
2003	455.637	3.959	117.551
2004	537.352	4.427	136.437
2005	620.789	4.505	154.086
2006	728.755	4.633	169.080
2007	825.561	5.007	189.057
2008(2)	950.120	4.236	184.468
2009(2)	1.053.346	4.324	201.380
2010(2)	1.104.388	4.045	211.496

Çizelge 2.2. 2009-2010 yıllarında trafikte uygulanan ceza sayıları [2].

<b>Türkiye Geneli Uygulanan Trafik Cezaları</b>			
	<b>2009</b>	<b>2010</b>	<b>FARK %</b>
Alkol Kontrolü	124.570	142.610	14,50%
Aşırı Hız Kontrolü	1.391.719	1.373.282	-1,30%
Cep Telefonu Kontrolü	189.155	221.245	17,00%
Emniyet Kemer Kontrolü	438.176	436.494	-0,40%
Koruma Başlığı-Gözlük Kontrolü	67.928	77.250	13,70%
Takograf Kontrolü	119.588	122.743	2,60%
Yaya Kontrolü	22.250	19.161	-13,90%

Trafikte insan hayatını hiçe sayan, kendisini bırakın karşıdaki insana yaşam hakkı vermeyen, aşırı hız yapan sürücülerin tespiti ve uygun görüldüğünde ceza ile karşılık bulan eylemlerinin sonuçlandırılması amacı güdülmüştür. Bu çalışmada kazaların

birinci aktörü olan insan ve insana bağı olan hıza bağı kazaların azaltılmasına dönük bir yaklaşım geliştirilmiştir.

Yapılan çalışmalarda donanımsal ve yazılımsal olarak yüksek maliyetlerin karşımıza çıkması, kurumların bazen ekonomik yetersizliklerle bu tür sistemlere adaptasyonunu zorlaştırmaktadır. Buna alternatif olarak daha düşük maliyetle daha yüksek verimle bu tür sistemlerin yaygınlaştırılmasına dönük araştırma geliştirme yapılmıştır.

## BÖLÜM 3

### PLAKA TANIMA SİSTEMLERİ

Gelişen teknolojiye bağlı olarak, bilgisayar sistemleri insan hayatının tüm alanlarında kendine yer bulmuştur. Özellikle günlük yaşamın düzenlenmesi, yaşam kalitesinin artırılması, insan vurgusu yapan uygulamalar her zaman ön planda olmuştur. Özellikle motorlu ve motorsuz araçların yaşamın bir parçası olmasıyla da insan hayatının önemi, bir trafik terörünün ortaya çıkmasıyla bir kez daha altının çizdirilmesine yol açmıştır. Her yıl ülkemiz dâhil tüm dünyada milyonlarca trafik kazası maddi ve manevi kayıplarla sonuçlanmaktadır. Bu kayıpların azaltılmasında alınan birçok önlemlerin olmasına rağmen insan faktörünün ağır basmasıyla maalesef ortadan kaldırılamamaktadır. Plaka Tanıma Sistemleri özellikle araç trafiğinin, güvenliğinin, yol kontrol ve akış istatistiklerinin tutulmasında kullanılan bir alandır. Bunun yanında giriş çıkış kontrollerinde de yaygın olarak kullanılan bir yapıdır.

Plaka Tanıma Sistemleri çevrimiçi ve çevrimdışı olarak iki şekilde tasarlanmaktadır. Çevrimiçi sistemlerde genelde video görüntüleri üzerinden tanımlama algoritmaları uygulanmaktadır. Çevrimdışı sistemlerde ise alınan fotoğraf görüntüleri ile plaka tanımlama işlemleri yapılır. Trafikte özellikle köprü, otoyol giriş ve çıkışlarının kontrolü, trafikte kural ihlali yapan araçların belirlenmesi, apartman site giriş çıkışlarının güvenliğinin sağlanması gibi uygulama alanlarında çevrimiçi yöntemler kullanılmaktadır.

Tetiklenen sistemlerin sonucu olarak alınan görüntü örneklemelerine ihtiyaç duyulan alanlarda ise çevrimdışı mimarileri uygulanmaktadır.

Bu çalışmada, fotoğraf görüntüleri üzerinden tanımlama işlemi yapılmıştır. Trafikte kural ihlali yapan araçların belirlenmesi aşamasında, hız verisine bağlı olarak araçların hızlarına bağlı olarak, 3-5 kare fotoğraf verisi alınarak, bu verilerin plaka

tanımlama işlemlerine tabi tutulup tanımlanan plaka bilgisinin tasarlanan bir veri tabanında tutulması amaçlanmıştır. Plaka verisi üzerinden kayıt sırasında sürücü bilgilerine ulaşıp bilgilerin girilmesi suretiyle bir veri bulutu oluşturulmaktadır. Böylece plaka merkezli sürücü hareketlerinin analizi sağlanabilmektedir.

Bu işlemler sırasıyla görüntü bilgisinin düşük maliyetli kameradan alınarak, yazılımla plaka alanın belirlenmesi, plaka bilgisinin okunması ve kayıt edilmesi suretiyle çalışmaktadır. Sonrasında sürücü bilgilerini içeren bir veri tabanında bu bilgiler tutulmaktadır. Plaka Tanıma Sistemlerinde görüntü işleme tekniklerinin temellerinden yararlanılmaktadır.

### **3.1. GÖRÜNTÜ İŞLEME**

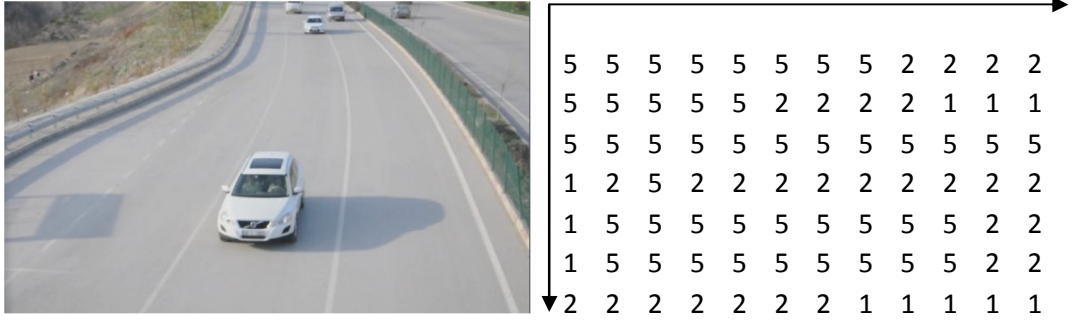
Plaka tanıma sistemlerinde temelde görüntü işleme yöntemlerinden yararlanılmaktadır. Dolayısıyla numune olarak alınan resim bilgisinde, resim bölgesini işlenmesi aşamasında hata oranının minimize edilmesi önemlidir. Bu aşamada seçilecek yöntem devreye girmekte, tercih edilen yöntemin doğru sonuçlar elde edilmesi noktasında ayırt edici olmasına dikkat edilmelidir.

### **3.2. GÖRÜNTÜ İŞLEME TEMELLERİ**

Görüntü İşleme ile ilgili olarak geliştirilen temel kuralların bilinmesi önemlidir. Temel kural en doğru şekilde imaj bilgisinin tanımlanmasıdır. Bu aşamada imajın renk bölgelerine ayrılıp işlenmesi gerekmektedir.

#### **3.2.1. Görüntü Tanımı**

Görüntü bilgisinin oluşmasının temeli, yansıyan ışığın kendisine duyarlı en yakın yüzeye düşmesi suretiyle, o bölgeye iz bırakmasıyla oluşur. Bu yüzey bazen bir dijital ya da analog kamera ya da benzer bir cihaz olabilir. Gelişen teknoloji ile yaygınlaşan dijital kameralarda, bu yüzey gelen yansıyan ışığın şiddetine göre bir sayısal değerle tutulur. Tüm bu sayısal değerler bir araya gelerek temelde görüntünün oluşmasını sağlar.



Şekil 3.1. Sayısal resim ve görüntü matrisi.

Elde edilen görüntüdeki resim bilgilerin sayısallaştırılması işlenmesini kolaylaştırılmaktadır. Özellikle iki renge indirgemede bu durum 0 ve 1 ile ifade edildiği için plaka tanıma sistemlerinde tercih edilen ve kullanılan bir yöntemdir. Gri renk skalasına indirgenen imaj bilgilerinde ise bu sefer resmin ifadesi 0 ile 255 arasında sayısal değerlere denk gelmektedir. Elde edilen görüntü matrisi Şekil 3.1.'de görüldüğü üzere  $m \times n$  matrisine dönüştürülmektedir. Eğer elde edilen görüntüdeki resim bilgisi renkli şekilde işlenecekse bu sefer imaj RGB denilen kırmızı, yeşil ve mavi şeklindeki üç farklı matrisle ifade edilmektedir. Bu üç matris ifadesi sonuçta gerçek renk bileşeni denilen sistemin oluşmasını sağlamaktadır [3].



Şekil 3.2. Renkli görüntü.

Şekil 3.2'de renkli görüntü bilgisi gösterilmiştir. Görüntü modeli olarak belirtilen yapı, iki boyutlu ışık ve yoğunluk fonksiyonudur. Model  $f(x,y)$  ile ifade edilir.  $(x,y)$  imgenin koordinat bilgisini;  $f(x,y)$  fonksiyonun değeri ise alınan bu koordinat noktasındaki pikselin ışık değerini ifade eder. Sayısal görüntü denilen sayısallaştırılmış görüntü modellerinin elde edilmesi iki aşamada gerçekleştirilir.

İşaret değerinden belirli aralıklarla örnekleme alınır. Örnekleme ardından kuantalama yapılarak, genlik seviyelerinin sadece belirli değerleri alınır. Örnekleme aşamasında çözünürlük kavramı karşımıza çıkar. Çözünürlük işlenen resim bilgisinin piksel sayısı ile hesaplanır. Birimi dpi (dot per inch) ve ppi (pixels per inch ) ile ifade edilir [3].

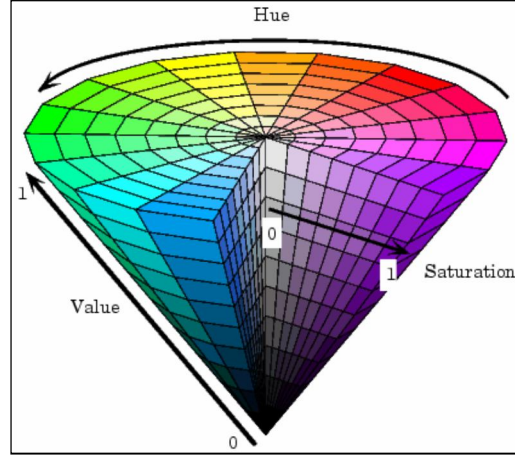
### **3.2.2. Renk Modelleri**

Renk bilimi insanoğlunun dünyayı anlaması ve algılaması noktasında ortaya çıkmış, gelişen teknolojik gelişmelere bağlı olarak bir bilim dalına dönüşen bir alan olmuştur. Isaac Newton kendi adını verdiği renk çemberi ile renk tonu ve doygunluk kavramları ile renklerin tanımlanması gerektiğinin üstünü çizmiştir. Renklerde parlaklığın yapısal bir nitelik olamayacağı tanımlamıştır. İnsan beyni çalışma mantığı olarak, renk algısını, ton, doygunluk ve parlaklık değerlerine göre algılar.

Renk modeli RGB (Red, Green, Blue ), CMYK (Cyan, Magenta, Yellow, Key ) gibi yardımcı renkle belirlenebilir. Bunun yanında HSI (Hue, Saturation, Intesity), HSB (Hue, Saturation, Brightness) ile ışık ve renk bileşeni ile de ifade edilir. RGB ile CMY arasında dönüşüm yapılabilir. CMY (Cyan, Magenta, Yellow) modelinde siyah rengin elde edilmesi tüm bileşenlerin en yüksek oranda kullanılması ile elde edilir. Bu yönü ile renkli yazıcılarda CMY renk modeli kullanılır.

Olgusal renk bölgesinde ise karşımıza, renk tonu denilen, renklerin tanımlanmasını sağlayan bir bileşen çıkar. Bu bileşen sayesinde kırmızı, mavi, yeşil vb. renkler beyin tarafından işlenebilir. Satürasyon (doygunluk), daha saf ve gerçek gözükmesini sağlayan beyaz olmayan renk kısımlarının seviyesini gösterir. Doygunluğu düşük resimlerde fazla miktarda beyaz renklerin karmaşıktığı görürüz. Alınan ışık seviyesinin yoğunluğunu ölçebildiğimiz parametre ise parlaklık olarak tanımlanabilir. Şekil 3.3'de olgusal renk bölgesi gösterilmiştir.





Şekil 3.3. Olgusal renk bölgesi [3].

Temelde RGB renk bölgelerinin kendi içinde oransal olarak belli tonlarda bozulması suretiyle oluşan bir renk bölgesi olan olgusal renk bölgesi; lineer bir dönüşüm ile tanımlanabilir. Munsell renk bölgesi olarak adlandırılan yaklaşık bin beş yüz renk örnekleme içeren bölgede, eşit aralıklarda örnekleme seçilmiştir. Travis ( Ford ve Robert 1998 ) ortaya koyduğu sonuçlarda RGB ve HSI ( Hue, Saturation, Intensity ) arasında lineer bir dönüşüm olduğunu belirtmiştir.

Olgusal renk bölgesi, RGB renk bölgesinin bozulması ile elde edilmiştir. Genellikle RGB bölgesinin lineer dönüşümü olarak ifade edilebilirler. Olgusal renk bölgesine ait olan, Munsell renk bölgesi sıralanmış 1500 adet renk örneği içerir. Söz konusu örnekler algısal olarak birbirine eşit olan örnekler arasından seçilmiştir.

$$\begin{aligned}
 R' &= \frac{\max(R, G, B) - R}{\max(R, G, B) - \min(R, G, B)} \\
 G' &= \frac{\max(R, G, B) - G}{\max(R, G, B) - \min(R, G, B)} \\
 B' &= \frac{\max(R, G, B) - B}{\max(R, G, B) - \min(R, G, B)}
 \end{aligned} \tag{3.1}$$

Renk tonunun hesaplanabilmesi için R, G, B, değerlerinin hesaplanması unutulmamalıdır. Eşitlik 3.1'de R, G, B değerlerinin nasıl hesaplandığı verilmiştir.

R' : Kırmızı renk ton değeri

G' : Yeşil renk ton değeri

B' : Mavi renk ton değeri

### **3.3. RENK DÖNÜŞÜMLERİ**

Görüntü işlemede işlem sırasında yapılan dönüşümlerin fazla olması istenen sonuca ulaşmada problemleri ortaya koyabilir. Örneğin resimde bulunan istenen bölgenin çıkarılmasında gri tonlama işleminin her işlem sonrası yapılması bazen işlem fazlalığına sebebiyet vermektedir.

Bunun yerine dönüşümler kullanılırken işlem sayısının minimum seviyede tutulması performansı arttıracaktır. Bu sebepten işlenecek resimlerin siyah beyaz tona dönüştürülerek binary ifade edebilme noktasına getirmek çoğu zaman avantajlı bir durumdur. Gri-Skala dönüşümleri bu noktada işlevsellikleri ile yaygın olarak kullanılmaktadır.

Uygulamanın yapısına göre renk modeli seçilir. Ana renklerin seçimi ile daha fazla renk gösterilebilir. Renk analizi söz konusu ise HSI daha uygundur. Çünkü insan göz algılamasına yakın bir yapıda çalışır. Resim ve video aktarımı, kalıcı yâda geçici olarak saklanması içinde, YIQ (bilgi ve renklilik) ve YUV (açıklık, renk alanı) gibi renk modelleri kullanılır.

### **3.4. GERÇEK RENK VE GRİ RENK SKALA DÖNÜŞÜMLERİ**

Bu yöntemde elimizdeki RGB verilerini, NTSC ( National Television Standarts Committee ) yapısı karşılıklarının bulunup doygunluk ve resim renk değerlerini minimum değeri olan sıfıra yakınlştırarak yapabiliriz. Eşitlik 3.2'de dönüşüm matrisi gösterilmektedir.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0,3 & 0,5 & 0,1 \\ 0,6 & -0,2 & -0,3 \\ 0,2 & -0,5 & 0,3 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.2)$$

I ve Q değerlerinin sıfıra yaklaştırılarak eşitlenmesiyle Y değeri hesaplanarak grileme yapılmış olur. Gri skalaya geçilerek de resmin işlenmesi aşaması kolaylaştırılmış olur. I ve Q renk fark işaretleridir.

$$Y = 0,3R + 0,5G + 0,1B \quad (3.3)$$

$$I = a ( R-Y ) \quad (3.4)$$

$$Q = b ( B -Y ) \quad (3.5)$$

### 3.5. SİYAH BEYAZ GÖRÜNTÜ DÖNÜŞÜMÜ

Daha önceki konularda değinildiği üzere, bir resim bilgisinin işlenebilmesi için temel yöntem resmin siyah beyaz dönüşümünün yapılmasıdır. Diğer renk aralıklarında ortam ışık seviyesinin farklılık arz edeceği gerçeği resmin işlenmesinde problemlere yol açabilmektedir. Parlaklık özellikle ortam koşullarından doğrudan etkilenen bir parametredir.

1979 yılında ortaya çıkan OTSU Algoritması ile resmin eşik seviyesi bulunarak, yapılan işlemlerde doğruluğun artırılması sağlanmaktadır. Ortaya çıkan eşik değeri [ 0-1] aralığında tanımlanan bir parlaklık seviyesiyle gri tonlaması yapılan resim siyah beyaz formatına çevirilir. Bu algorithmada temel mantık, kendi içinde sınıflandırılmış ve ağırlıklaştırılmış varyansı minimize edecek şekildedir. Buradaki sınıflar birbiri arasındaki varyansı maksimize ederek, gri resimleri 0 -256 aralığında değer alan piksel karşılıkları ile çalışır.

Bu yöntem kullanıldığında ortaya çıkan başlıca kabuller aşağıdaki gibidir.

1. Uzaysal evrede uyumluluk aranmaz.
2. Histogram ve resim bimodaldır.
3. Resim içinde tek tip parlaklık olduğu kabul edilir ve parlaklık değişimlerinin nesneden nesneye farklılık arz edeceği kabulü ile hareket edilir.
4. Sabit elde edilen istatistikî veriler ışığında farklı yapılar uyumu sağlanabilir.

Bu yöntemde aşağıdaki işlem basamakları izlenir:

Eşik değeri “T” Otsu’ya göre hesaplanır. Bu yöntemle görüntü üzerindeki gürültü olarak yer alan birçok piksel eleme işlemine tabi tutulur. Bu yöntemin amacı görüntüyü sınıflayacak en iyi eşik değerini elde etmektir. Eşik değeri Otsu metoduna göre şu şekilde hesaplanmaktadır [4]: Görüntü üzerindeki her bir gri seviye renk değerinin bulunma olasılığı Eşitlik 3.6’da verildiği gibi hesaplanır.

$$P_i = f_i / N \quad (3.6)$$

$f_i$  : i renk sayısına sahip toplam piksel sayısı

$N$  : Toplam piksel sayısı

$p_i$  : Görüntü üzerindeki her bir gri seviye renk değerinin bulunma olasılığı

Muhtemel eşik değeri T’ye göre, T’nin altında ve üstünde kalan toplam  $p_i$  değerleri  $w_1$  ve  $w_2$  Eşitlik 3.7 ve Eşitlik 3.8’e göre hesaplanır. [4].

$$w_1 = \sum_{i=1}^t p_i \quad (3.7)$$

$$w_2 = \sum_{i=t+1}^l p_i \quad (3.8)$$

T'nin altında ve üstünde kalan sınıfların ortalama olasılık değeri  $\mu_1$  ve  $\mu_2$  Eşitlik 3.9 ve Eşitlik 3.10'a göre hesaplanmaktadır.

$$\mu_1 = \sum_{i=1}^t i p_i \quad (3.9)$$

$$\mu_2 = \sum_{i=t+1}^1 i p_i \quad (3.10)$$

$$\mu_t = w_1 \mu_1 + w_2 \mu_2 \quad (3.11)$$

$\mu_t$  toplam olasılık değeri Eşitlik 3.11'e göre hesaplanmaktadır.

Bu eşitliklerden yararlanılarak sınıf varyansının değeri Eşitlik 3.12'ye göre hesaplanır.

$$\sigma^2 = w_1 (\mu_1 - \mu_t)^2 + w_2 (\mu_2 - \mu_t)^2 \quad (3.12)$$

$\sigma^2$  : Sınıf varyans değeri

Maksimum renk değeri tespit edilerek, max renk değeri kadar T değeri iterasyon yoluyla Eşitlik 3.13'deki gibi bulunur.

$$T = \text{Max}(\sigma^2) \quad (3.13)$$

### **3.6. GÖRÜNTÜ İŞLEME İLE PLAKA BÖLGESİNİN BULUNMASI**

Çevrimiçi ya da çevrimdışı olarak alınan resim bilgisi üzerinde plakanın yerinin doğru ve hızlı bir şekilde bulunması için çeşitli algoritmalar kullanılmaktadır. Bu işlemin plakanın yerinin bulunmasından sonraki plakanın okunması ve kayıt edilmesi aşamasında ise yapay sinir ağlarından yararlanılmaktadır.

Yapay sinir ağları unlineer sistemlerde yaygın olarak kullanılması, plakadaki karakterlerin yapılacak bir eğitimle tanımlanması aşamasının da lineer olmayan bir işlem olmasını gözler önüne sermektedir. Burada önceden tanımlanmış bir alanda yerel minimum noktalarının hesaplanması suretiyle çalışan bir algoritmadan söz etmek mümkündür.

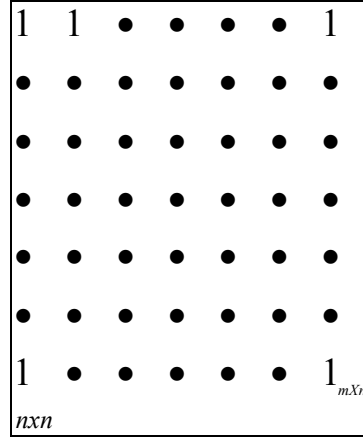
#### **3.6.1. Süzgeç Algoritmaları**

Süzgeç algoritmalarında kullanılan bir kernel süzgeç vardır. Bu kernel süzgeç ile işleme tabi tutulacak plaka bilgisinin olduğu resim ile korelasyona tabi tutularak işlem sonuçlandırılır.

Bu işlem sırasında en önemli aşama doğru korelasyonun uygulanmasıdır. Çekirdek merkezinin buradaki rolü çekirdek merkezinin hesaplanacak noktaya yerleştirilmesiyle başlar. Bu aşamadan sonra merkez noktaya komşu noktalar kernel içindeki değerler ile ağırlıklı olarak toplanır.

Şekil 3.4’de kernel süzgeç fonksiyonu gösterilmiştir.

Buradaki işlemle ilişkili olarak kavulasyon ise çekirdeğin sabit bırakılıp diğer noktaların ağırlıklandırılmasıdır. Şekil 3.5’de kernel fonksiyon uygulama mantığı gösterilmiştir.



Şekil 3.4. Çekirdek süzgeç fonksiyonu.

Görüntü Matrisi	12	24	2	22	11	$8^8$	$15^1$	$7^6$	Çekirdek Merkezi
	15	5	8	41	4	$14^3$	$16^5$	$12^7$	
	24	6	17	4	7	$17^4$	$22^9$	$15^2$	Korelasyon Çekirdeği
	4	18	10	8	8	12	20	16	
	10	17	4	5	5	14	10	22	
	7	19	8	6	23	17	3	27	
	8	20	6	8	20	18	7	25	
	45	7	32	7	10	12	9	10	

Şekil 3.5. Çekirdek fonksiyon uygulama mantığı.

### 3.6.2. Komşuluk Bölgelerinin Belirlenmesi

Görüntüde birbirleriyle komşuluğu olan bölgelerin gruplanması suretiyle, komşuluğu olmayan bölgelerden ayırt edilmesi işlemidir. Görüntü matrisindeki iki nokta birbiri ile temas halinde ise komşudurlar. En genel durum üzerinden bakılırsa A matrisinin A(i, j) elemanı incelenecek olursa; civarında bulunan tüm noktalar A(i, j) noktasının komşusudur. Bu çalışma ikili (siyah-beyaz) resim ile yapıldığından, bölgelerin alanları üzerinden kıyaslamalar yapılabilir. Komşuluk bölgesinin alanı, pembe ile belirlenmiş alanın altında kalan noktaların toplamıdır. Görüntü işlemede, aktif olan noktalar (“1”) üzerinden işlem yapılır. Resimde görülen, pembe alanındaki noktaların her birinin değerinin “1” olması durumunda bu alan içindeki noktalar birbiri ile komşudur denir. Bu alanın değeri, toplam dokuz adet nokta içerdiğinden “9”dur.

### 3.6.3. Görüntü Pekiştirme

Görüntü üzerindeki herhangi bir piksel değeri, o pikselin çevresindeki piksellerin ağırlıklı ortalaması ile bulunur. Buna konvolüsyon denir. Komşuluk gri seviyeleri bir konvolüsyon çekirdeği veya Görüntü Maskesi adı verilen matrisin katsayısına göre ağırlıklandırılır [5]. Eşitlik 3.14'e göre gerçek görüntü noktaları hesaplanır.

$$I'(u, v) = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} H(i, j)I(u - i, v - j) = I * H \quad (3.14)$$

$I(u, v)$  : Gerçek görüntü noktaları

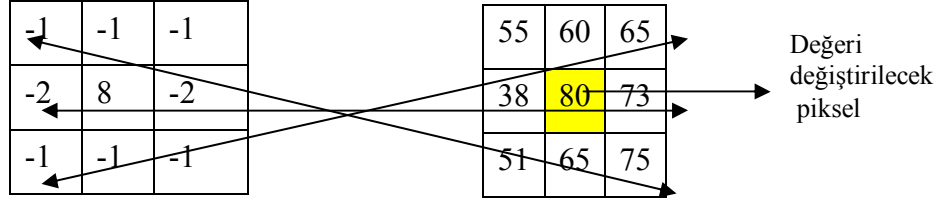
$I'(u, v)$  : Filtre uygulanan görüntü noktaları

$H(i, j)$ ,  $H$  : Görüntüye uygulanacak olan kernel

Böylece piksellerin komşuluğundan faydalanarak, bir filtre çekirdeğinin (konvolüsyon maskesi) resim üzerinde gezdirilmesi ile yapılır. Konvolüsyon işleminde uygulamanın yapısına göre görüntü maskesi seçimi yapılır. 3x3 ve 5x5 boyutlarında seçim yapılabilir.

Konvolüsyon işlemi ile işlenecek resimle ilgili olarak dikkat edilmesi gereken hususlar vardır. Örneğin kenar bölgelerinin işlememe, kenar bölgelerini kesmeye dikkat edilmelidir. Şekil 3.6'da konvolüsyon işlemi gösterilmiştir.





Şekil 3.6. Konvolüsyon işlemi.

Eşitlik 3.15 ve 3.16’da konvolüsyon işlemi hesaplama yöntemi ve sonuç değerleri gösterilmiştir.

$$\begin{aligned}
 G(x,y) &= k(-1,-1) f(x+1,y+1) + k(1,-1) f(x-1,y+1) \\
 &+ k(-1,0) f(x+1,y) + k(0,0) f(x,y) + k(1,0) f(x-1,y) \\
 &+ k(-1,1) f(x+1,y-1) + k(0,1) f(x,y-1)
 \end{aligned} \tag{3.15}$$

$$\begin{aligned}
 G(x,y) &= (-1*75) + (-1*65) + (-1*51) + (-2*73) + (8*80) + (-2*38) \\
 &+ (-1*65) + (-1*60) + (-1*55) \\
 &= 67
 \end{aligned} \tag{3.16}$$

#### 3.6.4. Karakter Tanıma

Yerel noktalara göre yapılan ayrıştırma sonrasında artık karakterlerin tanımlanması aşamasına gelinmiştir. Burada plaka üzerindeki karakterler yapay sinir ağları yada korelasyon yöntemleri ile tanınır ve ilgili aşamada işlenmek üzere hazırlanır.

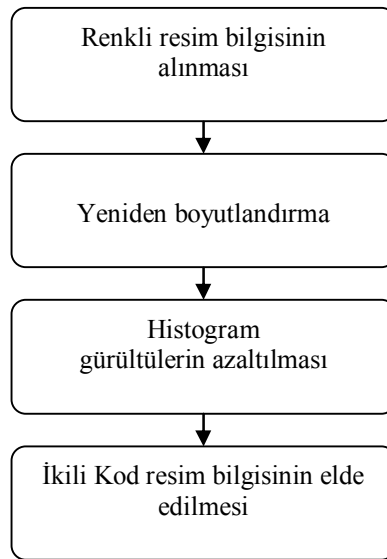
### 3.7. PLAKA TANIMA AŞAMALARI

Bu çalışmada öncelikli olarak hareketli görüntü üzerinden veri işleme işlemleri yapılsa da çevrimdışı alınan plaka bilgisinin okunmasında plaka tanımlama sistemleri algoritmalarından yararlanılmıştır. İlk olarak plaka ayrıştırma yapısı oluşturulmak üzere, verilerin işlenmesi yöntemi izlenmiştir. Mevcut karakter tanıma sistemleri incelenerek, performansları üzerinde çalışmalar yapılmıştır. Açık Kaynak Yazılım Kütüphanelerinden biri olan PUMA.Net aracı kullanılmıştır. GOOCR ( Free Optical

Character Recognition ) ile de açık kaynak kodlu sistemlerin kullanılabilirliği görülmüştür. Mevcut piyasa şartlarında ticari yönü ağır basan plaka tanıma sistemlerine alternatif olarak, maliyeti düşük kullanılabilirliği yüksek bir sistem ortaya koyma üzerine çalışmalar yapılmıştır. Geliştirilen yazılımlar, Visual Studio yazılım geliştirme ortamında geliştirilmiştir.

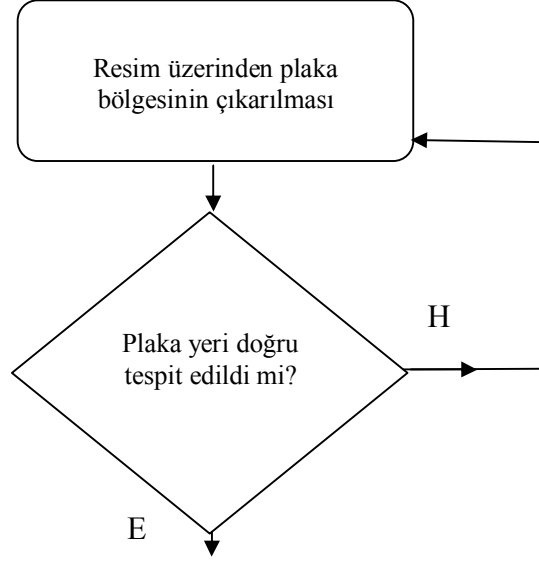
Plaka tanıma aşamalarında ilk kısım ön işlem aşamasıdır. Burada veriler minimize edilerek, dış etkenlerden kaynaklanan sistemde istenmeyen durumların önüne geçilmeye çalışılmıştır. Dış etkenlerden dolayı ortaya çıkacak olan distorsiyonların önlenmesinde filtrelerden yararlanılmıştır.

Yapılan bu ön işlem elemanı ile elde edilen görüntünün yeniden boyutlandırılması yoluna gidilmiş, bu boyutlandırılan kısım da yapay sinir ağları kullanılarak tanımlama ve kayıt işleminin yapılması sağlanmıştır. Burada en önemli kısımlardan biri YSA (Yapay Sinir Ağları) eğitilen nöronların parametreleri ( öğrenme katsayısı, momentum katsayısı gibi ) doğru bir şekilde ayarlanması önemli bir noktayı oluşturmaktadır. Eğer dış etkenlerin bozucu etkisi fazla ise, sistemin güvenilirliğini azaltıcı etken bir bozulma girişi oluyorsa o noktada YSA verilerinin eğitim aşamasında nöron sayısının ihtiyaca göre artırılması ya da filtre aşamasının yeniden değerlendirilmesi gerekebilmektedir [6]. Şekil 3.7’de plaka ön işlem aşamaları ayrıntılı şekilde gösterilmiştir.

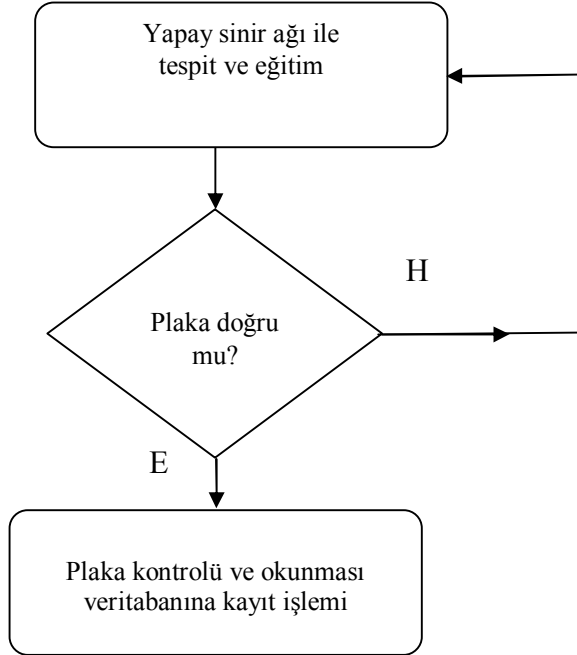


Şekil 3.7. Plaka ön işlem aşamaları.

Plakanın belirlenmesinde dış etkinlerin azaltıldığı ve plaka alanının ölçeklendirilip belirlendiği ön işlem aşamasından sonra alınan örnekleme ile plaka bölgesinin doğru şekilde belirlenip belirlenmediğinin kontrolünün yapılması gerekmektedir. Şekil 3.8’de plaka kontrol aşaması gösterilmiştir.

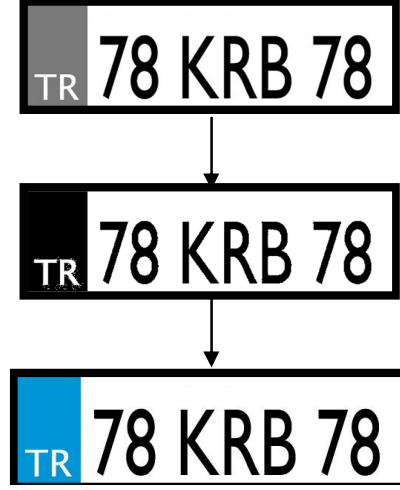


Şekil 3.8. Plaka kontrol aşaması.



Şekil 3.9. Plaka bilgisinin okunması ve kayıt edilmesi aşaması.

Bu aşama sonucu elde edilen veri neticesinde diğer aşamaya geçilip geçilmeyeceğinin kararı verilmektedir. Eğer plaka bölgesi tespit edilmediyse tekrar örnekleme alanı yeniden alınmalıdır. Böylece karakter analizine sokulacak alanın konumlandırılması aşaması tamamlanmaktadır [7]. Şekil 3.9'da plaka bilgisinin okunması ve kayıt edilmesi aşaması gösterilmiştir.



Şekil 3.10. Görüntü ön işlem sonuçları.

Ülkemizde araçlar üzerinde kullanılan plakalar, beyaz zemin üzerine siyah karakterler yazılmaktadır. Plakanın sol kısmında mavi zemin üzerinde Avrupa standartları gereği ülke kodu; sonraki kısımda il trafik kodunu veren 2 karakterli sayısal ifade; sonrasında ise alfabetik karakter ve sayısal değer vardır. Günümüzde trafiğe çıkan araçların muayenesini üstlenen TÜVTURK (Araç Muayene İstasyonu) tarafından araç muayenesi sırasında ülke kodu üzerine yapıştırılan muayene periyodunu bildiren hologram logo da tanımlamada dış etken olarak ele alınmıştır. Şekil 3.10'da görüntü ön işlem sonuçları gösterilmiştir.

Sistemin çalışmasında ilk olarak plaka verisi resim üzerinden alınır. İlgili karakterlerin olduğu plaka bölgesinin belirlenmesi YSA (Yapay Sinir Ağları) ile eğitim gerçekleştirilerek yapılır. Son aşamada plaka verisinin okunup veritabanı kaydı oluşturulur. Veri tabanında SQL ( Structured Query Language ) ile hazırlanan içersinde plaka bilgisi, araç sahibi bilgisi, Türkiye Cumhuriyeti kimlik numarası bilgisi, adres bilgisi, muayene tarihi gibi bilgileri yer almaktadır.

Ayrıca e-devlet uygulamalarından biri olan MERNİS ( Merkezi Nüfus İdaresi Sistemi ) ile adaptasyonu sağlanarak, araç sahibinin kontrolü de sağlanabilmektedir. Bu kısım hız ihlali yapan araçların plaka bilgilerinin okunduğunda bu plakanın kime ait olduğunun ve diğer ayrıntı bilgilerinin alınmasında yardımcı olacaktır. Özellikle oluşturulacak olan veritabanı ile uzun vadede bir veri bloğu oluşturulabilir.

### **3.8. YAPAY SİNİR AĞLARI**

Yapay Sinir Ağları'nın sistemde doğrudan kullanıldığı aşama budur. OCR (Optical Character Recognition ) ile tanımda kullanılacak karakterlerin yapısı irdelenmiştir. Amaç harf ve sayıların geriye yayılım algoritması ile karakterlerinin başarılı bir şekilde ayrıştırılmasıdır.

#### **3.8.1. Yapay Sinir Ağı Modeli Çok Katmanlı Algılayıcılar**

Modellenen bir sistemde en temel verileri girdi ve çıktılar oluşturur. Yapay Sinir Ağları'nın doğrusal olmayan sistemler üzerinde uygulanmasında “Çok Katmanlı Algılayıcılar” kullanılmaktadır. Ele alınan sistemin doğrusal olup olmadığını anlamak için mantıksal XOR ( Exclusive OR ) yapısını incelemek gerekir. XOR probleminin çözülmeye çalışılmasıyla Çok Katmanlı Algılayıcılar tanımlanmıştır. Rumelhart ve arkadaşları tarafından bu modelin bir diğer adı Geriye Yayılım Algoritmasıdır [8]. Tanıma, sınıflandırma, genelleme yapılması gereken problemlerin çözüm kümesi için bu algoritma kullanılır. Yapısı incelendiğinde Delta Öğrenme Kuralı adı verilen bir yöntemi kullandığı görülür. Başlangıçta basit algılayıcı modellerinin öğrenilmesi için geliştirilse de; asıl amaç tasarlanan ağdaki çıktı ile ağdan ortaya çıkan sonuç arasındaki hata oranını minimize etmektir. İşte bu tasarlanan yapıya ”Hata Yayma Ağı” denir [8].

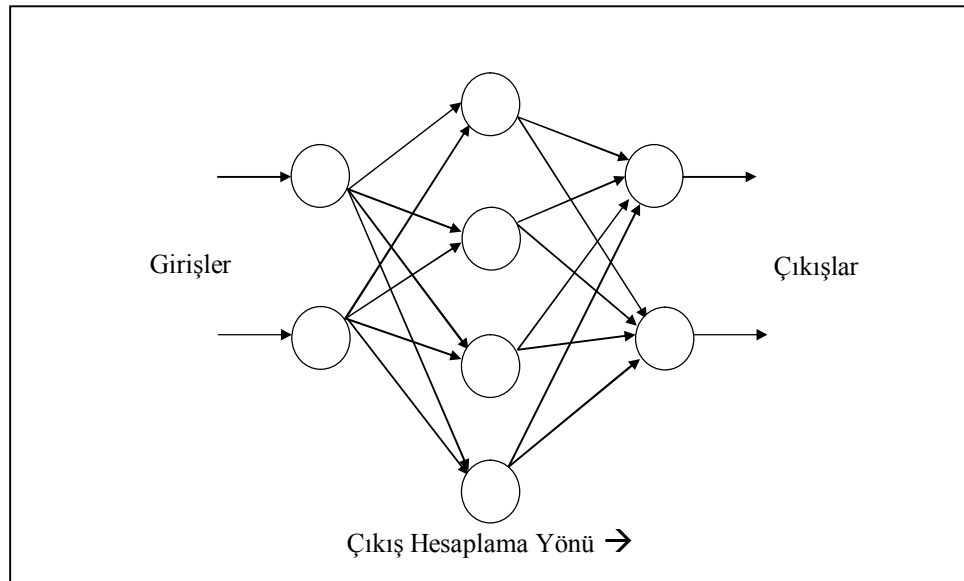
Sistem üç ana yapıdan oluşur.

1. Girdi Katmanı
2. Ara Katman
3. Çıktı Katmanı

Girdi Katmanı, dış dünyadan gelen her türlü girdi olarak tanımlanır. Bu katmanda amaç bilgi işlem değildir. Gelen her verinin bir sonraki aşamaya aktarılması temel amaçtır. Giriş sayısı birden fazla olabilir. Her işlem girişinin bir çıktısı olacağı unutulmamalıdır. Ara Katman, girişten alınan bilgileri çıktı katmanına aktarma görevi üstlenir. Çok Katmanlı bir Algılayıcıda birden fazla ara katman olabilir. Bu katmandaki her bir işlem elemanı bir sonraki katmandaki işlem elemanı ile bağlantılıdır.

Çıktı Katmanı, ara katmandan gelen girdi verilerinin karşılığı gösteren katmandır. Çıktı Katmanında, birden fazla işlem elemanı vardır. Burada gözden kaçmaması gereken olay, her işlem elemanının sadece bir tane çıktısı olduğudur.

Öğrenmeli Öğrenme stratejisini benimseyen Çok Katmanlı Algılayıcılarda, ağ mevcut örnekler ışığında problem uzayına karşılık gelecek bir uzay üretilir [9]. Şekil 3.11’de geriye yayılım algoritması gösterilmiştir.



Şekil 3.11. Geriye yayılım algoritması.

### 3.8.2. Öğrenme Kuralı

Bu modelde temel öğrenme kuralı, eğitim sırasındaki girdilere karşılık üretilmesi gereken çıktılardır. Ağın temel görevi, girişe karşılık gelen çıkışın üretilmesidir.



$$\zeta_j^a = \frac{1}{1 + e^{-(Net_j^a + \beta_j^a)}} \quad (3.18)$$

$\beta_j^a$  : Eşik değeri

Çıktı katmanından elde edilen değerler, hesaplanan ağırlık ileri işlemi ile tamamlanmış olur.

### 3.8.2.2. Geriye Doğru Hesaplama

Ağda işleme alınan girdiler için, ağ tarafından üretilen çıktılar karşılaştırılır. Bu iki veri arasındaki fark hata olarak tanımlanır. Uygulanan sistem üzerinde ortaya çıkan hatayı azaltmak temel amaçtır. Bu sebeple bu hesaplama yönteminde, ağırlık değerlerine dağıtılan hata ile sonraki aşamadaki hata azaltma yoluna gidilir. Eşitlik 3.19 ile işlem elemanı hata oranı hesaplanması gösterilmiştir.

$$E_m = B_m - \zeta_m \quad (3.19)$$

$E_m$  : İşlem elemanı hata oranı

Bu değer aslında işlem elemanı için oluşan hatadır. Buradaki asıl hatayı (toplam hatayı) bulmak için tüm hataların toplanması gerekir. Sonucu sıfır çıkmaması için negatif çıkma olasılığı olan hata verileri ele alındığında, ağırlıkların karelerinin hesaplanması yoluna gidilir. Eğitim aşamasında hata oranının azaltılması amacı ile hareket edilir. Toplam hata hesaplanması Eşitlik 3.20 de gösterilmiştir [10].

$$TH = \frac{1}{2} \sum_m E_m^2 \quad (3.20)$$

TH : Toplam Hata



Hatanın azaltılması için, hataya sebep olan işlem elemanlarına hata değerinin dağıtılması gerekir. Ağırlığı değiştirilen bir işlem elemanı anlamına gelen bu yapıda, iki durum söz konusudur. Birincisi ara katman ile çıktı katmanı arasındaki ağırlık değerlerinin değiştirilmesidir. Diğer durum ise girdi katmanı ile ara katmanı ağırlıklarının değiştirilmesidir.

### 3.8.2.3. Ara Katman ile Çıktı Katmanı Ağırlıklarının Değiştirilmesi

Herhangi bir t zamanında, ağırlığın değişim miktarının hesaplamak istersek, ara katmandaki j işlem elemanını çıktı katmanındaki m işlem elemanına bağlayan bağlantının ağırlığındaki değişim miktarı  $\Delta_{A_{jm}}^a$  şeklinde ifade edilir. Böylece ağırlığın değişim miktarı Eşitlik 3.21’de görüldüğü gibidir.

$$\Delta_{A_{jm}}^a(t) = \lambda \delta_m \dot{C}_j^a + \alpha \Delta_{A_{jm}}^a(t-1) \quad (3.21)$$

$\lambda$  : Öğrenme Katsayısı

$\alpha$  : Momentum Katsayısı

$\delta_m$  : m çıktı ünitesinin hata değeri

Bu formülasyonda, momentum katsayısı Çok Katmanlı Algılayıcı ağının öğrenme zamanında yerel bir noktada sabit kalmaması için ağırlık değişim değerinin belirgin bir oranda bir sonraki aşamadaki değişime eklenmesini, öğrenme katsayısı ise ağırlıkların değişim miktarını verir [10].Eşitlik 3.22’de m çıktı ünitesinin hata değeri hesaplanması gösterilmiştir.

$$\delta_m = f'(\text{NET}) \cdot E_m \quad (3.22)$$

$f'(\text{NET})$ : Aktivasyon türevi

Sigmoid fonksiyonun kullanılmasında Eşitlik 3.23’de gösterildiği gibidir.

$$\delta_m = \zeta_m (1 - \zeta_m) \cdot E_m \quad (3.23)$$

Değişim miktarı hesaplandıktan sonra yeni t iterasyonundaki değerler Eşitlik 3.24'deki gibidir.

$$A_{jm}^a(t) = A_{jm}^a(t-1) + \Delta A_{jm}^a(t) \quad (3.24)$$

Bu durum eşik değer ünitesinin ağırlıklarının değiştirmek gerekir. Bunun için de değişim miktarı hesaplanır. Çıktı katmanında bulunan işlem elemanlarının eşik değer ağırlıkları  $\beta^c$  şeklinde gösterilir. Çıktı sabit olarak 1 olması ile değişim miktarı Eşitlik 3.25'deki gibi hesaplanır.

$$\Delta \beta_m^c(t) = \lambda \delta_m + \alpha \Delta \beta_m^c(t-1) \quad (3.25)$$

Ara katman ile çıktı katman arasındaki ağırlıkların değişiminde, her ağırlığın sadece çıktı katmanındaki bir işlem elemanının hata değeri dikkate alınır. Buradaki hatayı oluşturan girdi katmanı ile ara katman arasındaki ağırlıklardır. En son ara katmana gelen bütün veriler girdi katmanı yâda bir önceki katmandan gelir. Bu sebeple ara katman ile girdi katmanı arasındaki ağırlıkların değiştirilmesinde çıktı katmanındaki işlem elemanlarının tümünün hatasından etkilenmesi gerekir. Değişim miktarı  $\Delta A^i$  Eşitlik 3.26'deki gibi hesaplanır.

$$\Delta A_{kj}^i(t) = \lambda \delta_j^a \zeta_j^a + \alpha \Delta A_{kj}^i(t-1) \quad (3.26)$$

$$\delta_j^a = f(\text{NET}) \sum_m \delta_m A_{jm}^a \quad (3.27)$$

$\delta_j^a$  : Hata Terimi

Hata terimi Eşitlik 3.27 ve Eşitlik 3.28'deki gibi hesaplanır.

$$\delta_j^a = C_j^a (1 - C_j^a) \sum_m \delta_m A_{jm}^a \quad (3.28)$$

Hata hesaplandıktan sonra bu eşitlik doğrultusunda değişim miktarı bulunabilir. Ağırlık değeri ise, Eşitlik 3.29'deki gibi hesaplanır.

$$A_{kj}^i = A_{kj}^i(t-1) \Delta A_{kj}^i(t) \quad (3.29)$$

$$\Delta \beta_j^a(t) = \lambda \delta_j^a + \alpha \Delta \beta_j^a(t-1) \quad (3.30)$$

Eşik değer aralıkları Eşitlik 3.30'deki gibi hesaplanır. Benzer şekilde ara katman eşik değer ağırlıkları  $\beta_j^a$  ile gösterilerek yukarıdaki değişim miktarı  $\Delta \beta_j^a(t)$  hesaplanır. Ağırlıkların yeni değerleri t iterasyonuna göre hesaplandığında Eşitlik 3.31'deki gibi hesaplanır.

$$\beta_j^a(t) = \beta_j^a(t-1) + \Delta \beta_j^a(t) \quad (3.31)$$

Bu durumda tüm ağırlıklar değiştirilmiş olur. Bir iterasyon hem geri hem de ileri hesaplama sonucunda tamamlanmış olur. Bu durum öğrenme tamamlanana kadar devam eder [10].

#### 3.8.2.4. Çok Katmanlı Algılayıcılar Ağının Çalışması

Her algoritmada olduğu gibi Çok Katmanlı Algılayıcılar Ağının da bir çalışma mantığı vardır. Hata oranının azaltılması ve buna ilişkin hata değerinin hesaplanıp, ara katmanda bunun dağıtılması suretiyle çalışan yapı örneklerin toplanması ile işlemeye başlar. Ağın çözüm kümesini üretmesi için önceki örneklerin ağa gösterilmesi gerekir. Ağın eğitilme aşamasında test için oluşturulan set ağa

gösterilmez. Eldeki örneklerin birer birer öğretilmesi sağlanır. Örnekler ışığında yapılan öğrenme ile örneklerin gösterilmesinden önceki öğrenme faaliyetleri karşılaştırılarak öğrenme başarısı görülür.

Ağın topolojik yapısının belirlenmesi ile girdi sayısı, ara katman sayısı, ara katman başına düşen işlem eleman sayısı gibi değerler belirlenir. Ağın topolojisi çıkarıldıktan sonra, öğrenme parametreleri belirlenir. Öğrenme katsayısı, işlem elemanlarının toplamı, momentum katsayısı gibi parametrelerin belirlenmesi gerekir. İşlem elemanlarının birbiri ile bağlantısını sağlayan ağırlık değerlerinin belirlenmesiyle de başlangıçta hangi değerleri alacağına dönük atama işlemi yapılır. Girdi ve çıktı değerleri belirli bir düzene göre seçilir ve ağın öğrenmeye başlaması sağlanır. Sonrasında öğrenme sırasında ileri yönde hesaplama yapılır. Ortaya çıkan sonuçla beklenen değerlerle karşılaştırılıp, ortaya çıkan hataya göre ağırlıkların değiştirilmesi aşamasıyla çok katmanlı algılayıcıların çalışma prosedürü tamamlanır.

Öğrenme işlemi tamamlandığında sistemin durdurulmasını sağlayan bir mekanizmanın olması gerekir. Bu durum ortaya çıkan hatanın bir alt seviyeye azaltılması ile yapılır [10].

### **3.8.2.5. Çok Katmanlı Algılayıcılarda Ağın Eğitilmesi**

Yapay Sinir Ağları günümüzde günlük yaşamın her alanında özellikle lineer uzayda çözümü olmayan sektörlerde hayatımızda yer bulmaktadır. Winrow-Hoff öğrenme kuralı YSA'da geliştirilerek geriye yayılım algoritması ortaya çıkmıştır. Burada giriş katmanı, gizli katman ve çıkış katmanı olarak 3 ana katman yer almaktadır. Her katmanda bir önceki aşamadan gelen veri ile beslenerek elde edilen bilgilerin doğruluğu sağlanmıştır, eğitim sonucu elde edilen verilerin güvenilirliği arttırılmaktadır.

Plaka Tanıma Sisteminde bizim çıkış verisi olarak okuyacağımız plaka bilgisi maksimum 8 karakterli ve plaka bilgisi içerisinde alfabetik karakterlerden 22 adet; sayısal karakterlerden rakamsal olarak 10 karakterlik bir veri gelecektir. Toplamda 32 karakterlik bir veri için  $2^6$  lık bir çıkış nöronuna ihtiyaç vardır. Fakat mevcut

durumun yeterli gelmemesi durumunda ilave olacak karakterler göz önünde bulularak  $2^7$ 'lik bir çıkış verisinin olacağı kabul edilmelidir.

Burada girişten alınan vektörlerle çıkışta beklenen sonuç vektörleri Yapay Sinir Ağlarında eğitmen için kullanılacaktır. Bu algorithmada standart olarak, gradyan azaltılması suretiyle bir yol izlenmektedir.

Yöntem aşağıdaki gibi özetlenebilir:

1. YSA'yı eğitmek için eğitim verilerinin uygulanması,
2. YSA'nın çıkışı ile elde edilmek istenen çıkış arasındaki hatanın her bir çıkış nöronu için tespit edilmesi,
3. Her bir nöron için ağırlık değerlerinin tespiti ve yerel hataların hesaplanması,
4. Yerel hataları azaltmak için gerekli ağırlık ve kutup değerlerinin hesaplanması,
5. Aynı adımları sonraki tekrarlar da yenilemek [11].

Çok katmanlı, ileri beslemeli YSA yapıları, geri yayılım eğitim algoritması kullanmak için gayet uygundur. Geri yayılım algoritmasının temeli, YSA'ya ait ağırlık ve kutup değerlerinin güncellenerek, performans fonksiyonunun, gradyanın negatif tarafına düşürülmesidir.

Gradyan eksiltme algoritması kullanılırken uygulanabilecek iki yol vardır: artım yöntemi ( incremental mode ) ve topluluk yöntemi (batch mode). Artım yönteminde; her bir giriş YSA'ya uygulandıktan sonra ağırlıklar ve kutuplar güncellenir. Topluluk yönteminde ise; tüm girişler YSA'ya uygulandıktan sonra ağırlıklar ve kutuplar güncellenir.

## BÖLÜM 4

### PLAKA TANIMA HIZ ÖLÇME VE ARAÇ YOĞUNLUĞUNUN TESPİTİ

Artan nüfusa paralel olarak artan araç sayısı beraberinde trafik kazalarının oranını arttırmaktadır. Bu durumu iyileştirme anlamında trafikte seyir halinde takip edildiklerinin fikrinin verilmesi, kazaların yaşanmasında caydırıcı bir unsurdur. Hız verilerinin okunması, araç yoğunluğuna bağlı olarak trafik denetim rollerinin belirlenmesinde önem taşır. Yapılan uygulamada mevcut sistemlere destek olarak bir yazılım mekanizmasının kullanılması düşünülmüştür.

#### 4.1. PLAKA TANIMA VE UYGULANAN YÖNTEM

Optik Karakter Tanıma Sistemi Sistemin bu bölümünde geri yayılım algoritmasıyla eğitilmiş ileri beslemeli çok katmanlı YSA'ları kullanılmaktadır. Türkiye Cumhuriyeti sivil araç plakalarının rakamları için rakam YSA ve harfleri için harf YSA olmak üzere iki tane YSA oluşturulmuştur. Eğitim seti gerçek görüntülerden alınan karakterlerden oluşmaktadır. Eğitim esnasında geri yayılım algoritmasını için kullanılan parametreler devir sayısı 1100, momentum 0.2 ve her nörondaki signum fonksiyonun a'sı 1 şeklindedir. Geri yayılım algoritmasında ağırlıklar sıralı yöntemle güncellenmektedir.

Harf YSA 35 giriş, 20 gizli nöron ve 23 çıkış nöronuna, rakam YSA ise 35 giriş, 20 gizli nöron ve 10 çıkış nöronuna sahiptir. Hangi karakterin hangi yapay sinir ağına gideceği ise karakterlerin dizilişleri belirler. Tek satırlı Türkiye Cumhuriyeti sivil plakaları için aday plaka bölgesinin 7 karakterli olanları için ilk 2 ve son 2 karakteri her zaman rakam ve 3. karakteri her zaman harf olduğu için sırasıyla bu karakterler rakam ve harf YSA'larına gönderilir. 4. ve 5. sıradaki karakterler ise hem harf hem rakam yapay sinir ağına gönderilerek yapay sinir ağlarından alınan sonuca göre hangisinin değeri daha fazlaysa o yapay sinir ağı için sınıflandırılır.

Aday plaka bölgelerinin 8 karakterli olanlarının ise ilk 2 ve son 3 karakteri her zaman rakam, 3. ve 4. sü her zaman harf olduğu için sırasıyla bu karakterlerde rakam ve harf YSA'na gönderilmektedir. 5. karakter ise her iki YSA'na gönderilerek alınan sonuca göre sınıflandırılmaktadır.

Sınıflandırılan karakterlerin ASCII (American Standard Code for Information Interchange) karşılığı oluşturularak karakter dizisine aktarılmaktadır. Buraya kadar olan yöntemlerin sonucunda oluşturulan karakter dizisinde elman sayısı ikiden fazlaysa bu karakterlerin en iyisi seçilmektedir. En iyisini seçmek için 7 ve 8 karaktere sahip olan dizilerden hangisi karakter dizisinde daha fazla ise o elemanlara bakılır.

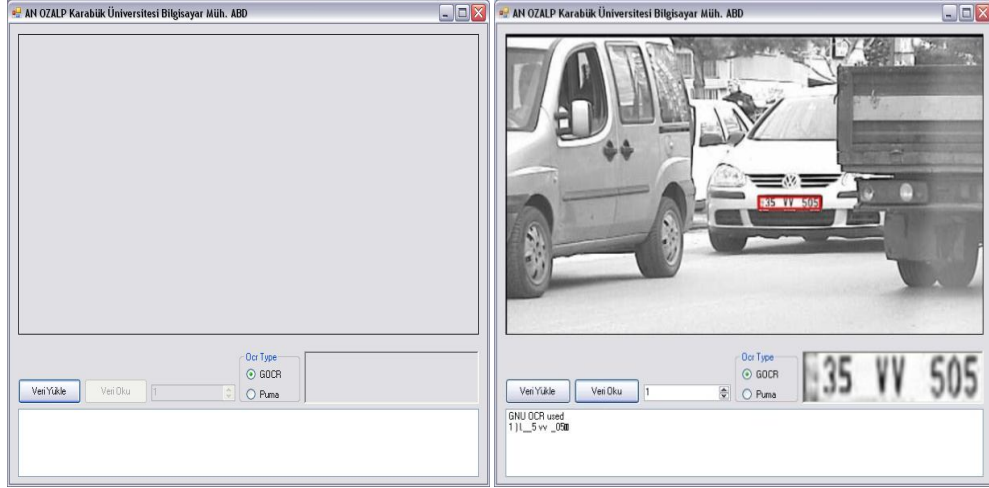
Örneğin 8 elemanlı olan diziler daha fazlaysa sadece 8 karaktere sahip olanlara bakılır. Seçme işleminde karakter dizilerinin ilk elemanlarından başlanarak son elemanlarına kadar her bir karakter dizisinin sütun hizasında ilgili karakterden hangisinden daha fazla ise o karakter sonuca yazılmaktadır.

Örneğin 78AN060, 78AN066 ve 78AC606 aday plaka bölgeleri sistemin sonunda bulunsun. Aday bölgelerde karakter dizisinin ilk sütununda 7, ikinci sütununda 8, üçüncü sütununda A, dördüncü sütununda N, beşinci sütununda 0, altıncı sütununda 6 ve yedinci sütununda 6 karakterleri baskındır. Bundan dolayı sonuç 78AN066 olarak sistemin çıktısı oluşturulmaktadır.

Şekil 4.1'de uygulama yazılımı ekran çıktısı gösterilmektedir. Bu ekran plaka tanıma işleminin yapıldığı kısımdır.

#### **4.2. SEYİR HALİNDEKİ ARACIN TESPİTİ VE HIZ ÖLÇÜMÜ**

Çalışmadaki amaç, standart CCTV (Close Curcuit TV Systems), IPCAM (Internet Protocol Network Camera) vb kameralarla geniş tabanlı bir izleme ve belirleme işlemi yapmaktır. Video ortamından alınan veriler ışığında araçların hızlarının tespit edilmesi suretiyle trafikte hız ihlali yapan araçların belirlenmesi amaçlanmıştır.



Şekil 4.1. Uygulama ekran çıktısı.

Bu kısımda görüntü işleme ve hareket eden nesnelerin algılanması, hız bilgilerinin tanımlanması karmaşık matematiksel algoritmalarla çözümlenmesi gereken bir kümedir. Bu işlemler:

1. Sahne Modelleme
2. Nesnelerin Geometrik şekillerle belirlenmesi
3. Görüntü üzerine Contour Algoritması uygulanması [12].

Trafikteki araçların belirlenmesinde izlene yol hareket eden nesnenin takip edilmesiyle başlar. Hareket edilen nesnenin takibinin sağlanması ardından kare değişimi üzerinden geliştirilen yazılımla hız verisinin alınması sağlanmıştır. 1/30 Fps (Frame Per Second) değerine sahip kamera ile yapılan ölçümlerde yaklaşık 200 m'lik alan içersinde ölçüm işlemi yapılmıştır. Burada toplam yer değiştirme hesaplaması üzerinden hız verisi hesaplanmıştır. Burada formülize edilen hız verisinde anlık hız, ortalama hız gibi formüllerden yararlanılmıştır. Ortalama Hız bilgisi toplam yer değiştirmenin toplam zamana bölünmesi ile hesaplanır. Eşitlik 4,1'deki formül ile ortalama hız hesaplanır. Burada  $V_{ort}$  ortalama hızı; X değerleri il ve son konumları; t değerleri ise ilk ve son zaman değerleri anlamına gelir. Eşitlik 4.1'e göre ortalama hız hesaplanmaktadır.



$$V_{\text{ort}} = \frac{x_{\text{son}} - x_{\text{ilk}}}{t_{\text{son}} - t_{\text{ilk}}} \quad (4.1)$$

Video işleme aşamasında;

1. Nesneye bağlı olarak contour çıkarma işlemi
2. Nesneye bağlı olarak hareket algılama işlemleri yapılmıştır.

Kameranın görüntü aldığı ölçüm alanın 155 m olarak ayarlanmıştır. Uygulama aracıyla 60, 70, 80, 90 ve 100 km/sa'lik hıza kadar geçiş yapılarak ölçüm yapılmıştır. Kamera Safranbolu Karabük Karayolu üzerinde bulunan üst geçit üzerine yerleştirilerek, gerçek zamanlı veri alışverişi sağlanmıştır.

Uygulama alanının uzunluğu ve bu aralıktan geçen aracın geçiş süresi bir arada kullanılmıştır. Bu değerlerde hızın alınan yolun zaman değişimine bölünmesi ile araç hızı her geçiş için hesaplanmıştır. Hesaplanan hız değerleri, aracın hız göstergesinde okunan ve zaman-hız ilişkisinden hesaplanan hızlarla karşılaştırılmıştır. Böylece hesaplanan hızlar arasında oluşan R2 değerleri incelenmiş ve böylece hesaplanan hızların doğruluğu bir kere daha araştırılmıştır. Uygulama süreci içerisinde yoldan geçen araçların hızları hesaplanırken, uygulama alanı için oluşturulan formül kullanılmıştır.

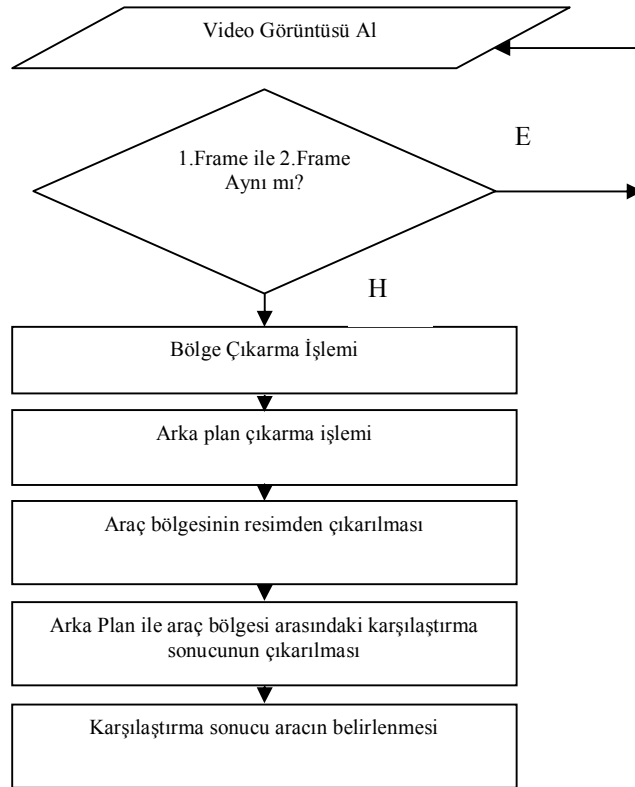
Uygulama ilk olarak, uzunluğu 70 m olan alanda yapılmıştır. Uygulama aracıyla bu alandan 50, 70 ve 90 km/saatlik hızlar ikişer defa uygulanmak üzere 5 defa geçilmiştir.

Uygulama aracının bu alandan geçiş süreleri görüntülerin analizi ile hesaplanmış ve araç geçişlerinin sayısı, bu geçişlerin süreleri ve uygulanan hızlar aşağıdaki tabloda gösterilmiştir. Buradaki uygulama hızı, uygulama aracının hız göstergesinde okunan değerdir [13]. Çizelge 4.1'de nesne algı süre ölçümleri verilmiştir.

Çizelge 4.1. Nesne algı süre ölçümleri.

Ölçüm	Süre (t)	Hız Değeri( km/sa)
1	5,04	50
2	3,42	60
3	2,74	70
4	4,78	80
5	3,74	90

Araç algılanması ve hareketinin takibi noktasında önemli bir noktada, hareketin algılandığı zaman aralığıdır. Bu değer ölçüm mesafesi, kamera açısı, kamera yakalama hızı gibi parametrelerle orantılıdır. Aşağıda bu aşamanın kodlarına yer verilmiştir. Şekil 4.2’de görülen algoritma çerçevesinde işlemler yapılmaktadır.



Şekil 4.2. Araç hareketinin çıkarılması.

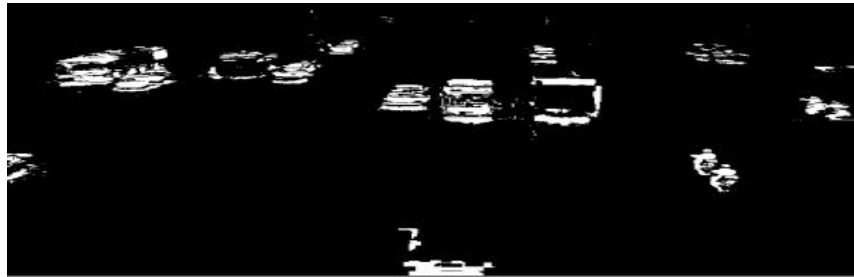
#### 4.2.1. Gerçek Zamanlı Araç Tanımlama ve Hareket Belirleme

Gerçek zamanlı yapılan araç hız verinin elde edilmesinde aşağıdaki yol izlenmektedir. Öncelikle araç hız ölçümü yapılacak olan yön belirlenmelidir. Yön ve hatta ölçüm yapılacak şeridin tanımlanması ölçüm sonuçlarının doğruluğunu arttıracaktır. Çok şeritli yollarda araçların hızlarının belirlenmesi için her şeride paralel olarak birer kamera uygulaması yapılmalıdır. Çoklu ölçüm ile aynı yâda hızları yüksek olan araçların belirlenmesi daha kolay olacaktır. Aşağıdaki görüntü işleme tekniklerinin çevrimiçi bir uygulamadaki performanslarının yükseltilmesi için çalışma yapılmıştır. Şekil 4.3’de gerçek zamanlı görüntü alma sonucu görülmektedir.

1. Ölçüm yapılacak olan şeridin belirlenmesi
2. Arka plan çıkarma
3. Contour Çıkarma
4. Contour Bağlama
5. Araç Hareket ve Hız verilerinin okunması



Şekil 4.3. Gerçek zamanlı görüntü alma.



Şekil 4.4. Hareketli araçların belirginleştirilmesi.

Şekil 4.4’de uygulama ekranı sonucu hareketli araçların belirginleştirilmesi sonucu görülmektedir.

#### **4.2.2. Yol Bölümleme (Lane Masking) İşlemi**

Lane Masking yönteminde; öncelikli olarak araç hareketlerinin yön bilgisinin alınması gerekir. Hareketin algılanması istenen şeride göre kamera açısının ayarlanması gerekmektedir. Ölçüm yapılacak olan doğrultunun belirlenmesi ardından arka plan eleme işlemi öncesinde hareketli nesnenin algılanmasında her karedeki değişim miktarına göre bir algoritma geliştirilmiştir. Burada kameranın Fps değerine göre bir algı söz konusudur. 1/30 değerinde bir Fps’ye sahip kameradaki hareket algılama duyarlılığına göre karelerdeki değişimin hesaplanmasında 1.frame, 3.frame, 5.frame şeklinde bir lineer artış gösteren mantık üzerinden gidilmiştir. Fps değeri yüksek donanım aygıtlarıyla yapıldığında ise hareketin algılanması ve hız verisinin ölçülmesi çok daha hassas ve doğru olacağı düşünülmektedir [14].

Hareket eden nesnenin belirlenmesinde, nesnenin hareket yönü ve hareket alanı önemlidir. Bununla ilgili olarak, nesnenin takibi için her framedeki hareket farkının algılanması gerekir.

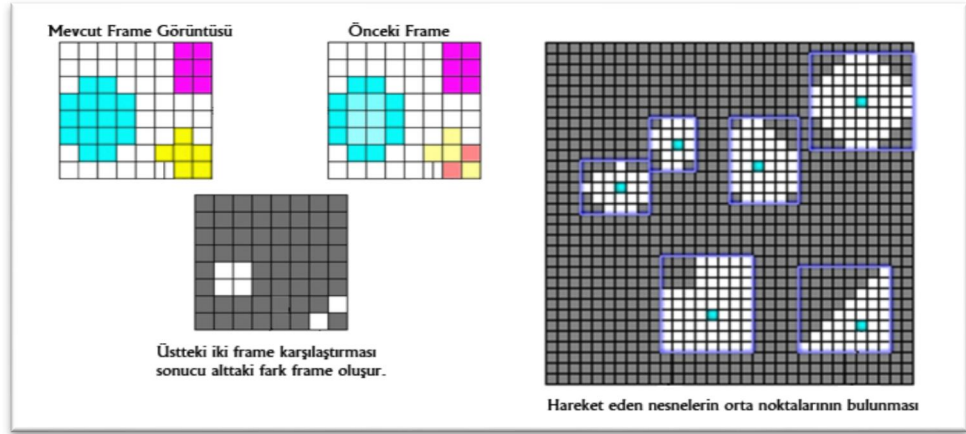
#### **4.2.3. Hareket Arka Plan Çıkarma İşlemi**

Sabit bütün nesnelere kaldırmak için kullanılır. Bu aşamadaki mevsimler faktörlere göre değişimler dikkate alınmalıdır. Sallanan ağaçlar vb gibi Arka Plan B (p), her RGB değerleri ortalama olarak hesaplanır. Ölçüm yapılacak alanda ilk olarak boş bir arka plan görüntüsünün alınarak, sonraki karelerde bu arka plana göre karşılaştırma yapmak suretiyle hareketli nesne belirlenmesi üzerine çalışılmıştır [15].



Şekil 4.5. Hareket tespiti.

Daha önce de bahsedildiği gibi, fps değeri yüksek olan bir kamera ile çekim yapıldığı varsayıldığında, her karedeki değişim miktarı daha hassas şekilde görülecektir. Hareket eden nesnenin belirlenmesinde temel olarak arka plan ayırt etme işlemi yapılır. Bu işlemin anlamı, video verisi içerisindeki frame'ler arasındaki farkın çıkarılmasıdır. Böylece nesnenin orta noktasının hareket noktaları arasında doğrusal bir ilişki oluşturulur. Her frame arası farklılığa bağlı olarak nesnenin hareketi algılanabilir. Bir sonraki frame bir önceki frame ile doldurularak, fark kümesi elde edilmiş olacaktır [16]. Şekil 4.5'de hareket tespiti için frame farkları gösterilmiştir. Şekil 4.6'da orta nokta bulma fark ve tolerans değerlerinin bulunması gösterilmiştir.



Şekil 4.6. Orta nokta bulma fark ve tolerans değerleri.

#### 4.2.4. Blob Algoritması Uygulama İşlemi

Temelde hareketli nesnelerin belirlenmesinde çeşitli algoritmalar kullanılır. Blob çıkarma da arka planının ortadan kaldırılarak hareketli nesnenin belirginleştirilmesi sağlanır. İlk planda boş bir arka plan ile karşılaştırma yapılarak, hareket eden nesnenin hareket durumu algılanabilir. Hareket eden nesnenin frame farklarından konumu tespit edilir. Frame içinde tespit edilen nesnenin orta noktası belirlenerek, nesne işaretlenip, hareket ID (Identity)'si verilir. Tüm işlemler bu ID üzerinden yapılır [17]. Şekil 4.7'de sobel edge algoritması sonucu elde edilen görüntü gösterilmiştir. Şekil 4.8'de ise log edge bulma algoritması sonucu elde edilen görüntü gösterilmektedir.



Şekil 4.7. Sobel edge algoritması sonucu elde edilen görüntü.



Şekil 4.8. Log edge bulma algoritması sonucu elde edilen görüntü.

Görüntü İşlem algoritmaları içerisinde en popüler olan algoritmalarından biri olan Sobel Algoritmasında amaç, verilen resim yada video görüntüsü üzerinde kenar bulma işleminin yapılmasıdır. Böylece görüntü içerisinde algılanması istenen nesnelerin belirlenmesinde kolaylık sağlanacaktır.

Sobel Algoritmasında temelde iki görüntü maskesi kullanılır. Bunlardan biri yatayda biri de dikeyde kenar bulmaya yarar. Bu çekirdek görüntü içersinde ışık yoğunluğunun değişiminin ani olduğu yerlerin bulunmasında kullanılır. Temelde bir türev mantığı ile çalıştığı görülür [18]. Şekil 4.9’da yatay ve dikey sobel maskeleri görülmektedir.

1	2	1
0	0	0
-1	-2	-1
Dikey Sobel Çekirdeği maskesi		

1	0	-1
2	0	-2
1	0	-1
Yatay Sobel Maskesi		

Şekil 4.9. Sobel algoritması dikey ve yatay maskeleri.

#### 4.2.5. Araç Hareketi

Hareket nesnesinin takibinin yapılması için hareket eden nesnenin orta noktalarının bulunması gerekir. Orta noktası bulunurken dikkat edilen nokta frame farklarının çıkarılmasıdır. Burada  $x_c$  ve  $y_c$  cismin hesaplanan orta noktalarını;  $\sum x_j$  cismin  $x$ 'deki göre toplam piksel sayısı;  $\sum y_j$  cismin  $y$ 'deki göre toplam piksel sayısını göstermektedir.  $n$  ise toplam piksel sayısıdır [19]. Eşitlik 4.2’de cismin orta noktaları hesaplanması gösterilmiştir.

$$x_c = \frac{\sum x_j}{n}, \quad y_c = \frac{\sum y_j}{n} \quad (4.2)$$

##### 4.2.5.1. Araç Tabanlı Hareket Algılama İşlemi

Gerçek zamanlı alınan görüntüye göre hız tahmini yapan bir yapı oluşturulmuştur. Burada alınan görüntüdeki frame değişimleri göre hesaplama yapılmıştır. Mevcut radar sistemlerinin çalışma prensibi; radyo dalgalarının bir kaynaktan gönderilip, geri yansıma süresi üzerinden hız verisinin alınması suretiyle çalışır. Alıcı devre geri dönen ekodan hedefin yön ve hız bilgisini, iki eko arasında geçen zamandan da hız bilgisi alınır. Eğer hız verisi alınacak olan nesne radar yörüngesinde radara doğru

geliyorsa, Doppler etkisiyle de hız bilgisi okunabilir [20]. Şekil 4.10'da uygulamada kullanılan ipcam gösterilmektedir.

- MJPEG : Moving Pictures Experts Group  
IEEE : The Institute of Electrical and Electronics Engineers  
Mbps : Megabit per second.  
WEP : Wired Equivalent Privacy  
WPA : Wi-Fi Protected Access  
WPA2 : Wi-Fi Protected Access 2



Şekil 4.10. Uygulamada kullanılan ipcam.

Çizelge 4.2'de uygulamada kullanılan ipcam özellikleri verilmiştir. 640X480 Piksel renkli CMOS (Complementary Metal Oxide Semiconductor ) sensörlü 2.8mm lense QVGA (Quarter Video Graphics Adapter)'da 30 fps; VGA (Video Graphics Adapter )'da 15 fps resim kare hızına sahip IP CAM kullanılmıştır.

Çizelge 4.2. Uygulamada kullanılan ipcam özellikleri.

Görüntü Algılayıcısı	300K 640x480 CMOS Sensör
Lens	F2.8 mm
Mini Aydınlatma	0.5 Lux
Lens Tipi	Cam Lens
Görüntü Sıkıştırma	MJPEG



Işık frekansı 50-60 Hz arasında kullanma şansı vermektedir. Ayrıca araç içi kamera kayıt DVR şekil 4.9’da gösterilen çizelge 4.3 ile de genel özellikleri belirtilen ( Digital Video Recorder ) cihazı olarak kullanılan 2.5 inch LCD (Liquid Crystal Display) ekrana sahip portatif kayıt cihazı da çevrimdışı kayıtlar için kullanılmıştır. Mobil özelliği sayesinde yüksek çözünürlüklü kayıt imkânı sağlaması ve maliyet noktasındaki avantajları bu cihazların kullanılmasına karar verilmesinde rol oynamıştır. Şekil 4.11’de görülen mobile dvr kayıt cihazı gösterilmektedir.



Şekil 4.11. Mobile dvr kayıt cihazı.

Çizelge 4.3. Mobil dvr kayıt cihazı teknik özellikleri.

Ekran Boyutu	2.5 inch
Görüntü Algılayıcısı	CMOS
Objektif	135 derece
Kayıt Çözünürlük	WVGA 1280x720 piksel 30 fps
Video Formatı	H.264

Çizelge 4.3’de uygulamada kullanılan dvr kayıt cihazı teknik özellikleri verilmiştir. Program geliştirme ve yazılım test performans ölçüm ve denemeleri Intel Pentium Mobile 760 (2.0Ghz/2MB/533FSB (Front side bus) ) işlemciye; 2 GB 533 DDR2 RAM (Random Access Memory) kapasiteli, 2MB L2 Tampon Belleği olan taşınabilir bilgisayar üzerinde yapılmıştır. Ayrıca taşınabilir bilgisayar üzerinde bulunan 1.3 MP çözünürlüklü kamera ile de alınan görüntülerin işlenmesi denemeleri de yapılmıştır.

### 4.3. DENEY SONUCUNDAKİ VERİLERİN DEĞERLENDİRİLMESİ

Yapılan ölçümler neticesinde elde edilen verilen ışığında yazılım çıktıları doğruluğu yüksek değerler vermektedir. Bu durum yazılım algoritmasının geliştirilerek kullanılabilme imkânını arttırmaktadır.

Bu çalışmada pasif plaka tanıma sistemi, araç hız ölçme, araç sayımı ve araç yoğunluğu tespit sistemleri gerçekleştirilmiştir. Sistem önden ve arkadan alınan araç görüntüsündeki plaka alanının çizgiye dayalı metot kullanarak belirlenmesine ve karakter tanıma kısmına bağlıdır. Çizgiye dayalı metot araba görüntüsünün enine kesitini gözlemlemeye dayanmaktadır. Enine kesit plaka numarası alanını kestiği zaman plaka alanı tespit edilir. Görüntü üzerinde plaka alanı bulunduktan sonra karakter ayrıştırma işlemi yapılmaktadır. Karakter ayrıştırma için yatay bölünme dikey bölünme ve karakter seçimi işlemleri uygulanmaktadır. Karakter ayrıştırma işlemi tamamlandıktan sonra şablon eşleme yaklaşımı ile karakter tanıma işlemi yerine getirilmektedir.

Gerçekleştirilen sistem 240 farklı araç görüntüsü üzerinde test edilmiştir. Sistemin başarı performansı sırası plaka bölgesinin tespitinde %95 plaka karakterlerinin bölünmesinde %94 ve plaka karakterlerinin tanınmasında %90 üstünde elde edilmiştir.

Çalışma içinde tasarlanan uygulamada kullanılan yaklaşık 240 plaka örneği üzerinde yapılan denemelerde plaka tanımadaki başarının çekilen resim açısı, gece gündüz zaman dilimi, resim ışık kalitesi, plaka yüzey okunabilirliği, yüzey dış etken şartları gibi etkilere bağlıdır[21].

Bu çalışmada gerçekleştirilen uygulamanın başarı performansı literatürdeki benzer 3 çalışma ile karşılaştırılmıştır. Birinci çalışmada: Plaka bölgesi çıkarılırken kenar belirleme lekeleme algoritmaları ayrıştırma bölümünde lekeleme, filtre ve bazı morfolojik algoritmalar plaka karakterlerinin tanınması için ise şablon eşleştirme kullanılmıştır.

İkinci çalışmada plaka bölgesinin bulunabilmesi için Gabor benzeri bir filtre, yatay yönde genişletme, eşikleme Bağlantı Bileşen Analizi kullanılmıştır. Tespit edilen plaka bölgesi birbirinden ayrı karakterlere bölünerek karakter tanıma sistemi için hazır hale getirilmiştir. Bundan sonraki fazda Self Organizing Map kullanılarak karakter tanınmıştır [22].

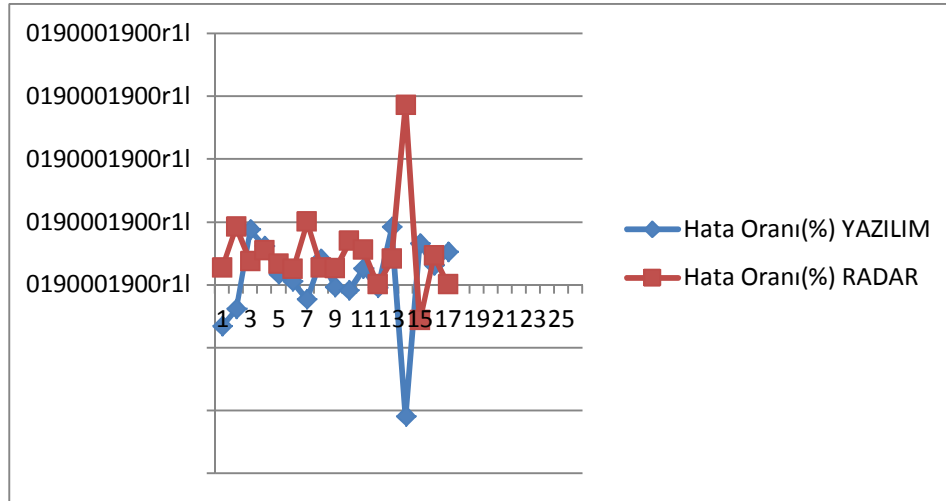
Sistem genel olarak literatürdeki çalışmalarda bulunan çalışmalarla karşılaştırıldığında benzer sonuçlar verdiği görülmektedir. Gerçek zamanlı uygulamalarda istenen hız ve verime sahip bir çalışma olmuştur. Benzer çalışmalarla karşılaştırıldığında başarı yüzdesinin yüksek olduğu görülmektedir.

Karakter tanıma da bir diğer etken farklı font ve yapıda plaka formatlarının da ülkemizde trafikte seyir halinde olmasıdır. Kalın, ince, birbirine yakın, TR logolu, TR logosuz, mavi logo yüzeyli, kırmızı logo yüzeyli gibi etkenlerde zaman zaman okumada başarıyı düşürmektedir. Ticari bir konu olması sebebiyle de sadece özel sektör tarafından araştırma geliştirme çalışmalarının yapılması konu ile ilgili çalışma sıkıntısını da beraberinde getirmektedir. Donanım kaynaklı yüksek maliyetlerin, yazılımsal olarak indirgenebilmesi, sistem kaynaklarının verimli kullanılması durumunda, bu sistemlerin farklı sektörlerde kullanılabilmesi sağlanabilir.

Araç hız tespiti çalışmasında, mevcut uygulamada yer alan yüksek maliyetli donanım ve iş gücü ile yapılmakta olan radar, EDS ve karayolları tarafından uygulanan hız tespit panolarına alternatif olarak, yazılım tabanlı hız ölçme işlemi yapılmıştır. Mevcut sistemlerle karşılaştırıldığında, öncelikle düşük donanım maliyeti, doğruluğu yüksek, kullanılabilirliği güvenli düzeyde olması sistemin artı noktalarını ortaya koymaktadır.

Araç hız tespiti ile trafikte tehlike arz eden araçların plakalarının tespit edilmesi fikri ile bu iki uygulama birleştirilerek, bundan sonraki çalışmalara referans teşkil edebilir. Burada her iki sistem arasında donanımsal bir sistem dizayn edilerek, sistemler arasında veri alışverişi sağlayan bir algoritma ile bu sistem adapte edilebilir. Araç hız ölçümü için geliştirilen yazılımla yapılan gerçek testlerde, radar ve gerçek değer arasındaki ölçüm oranları elde edilmiştir. Ülkemizde kullanılan

mevcut radar sistemleri bütçe yetersizlikleri gibi nedenlerden dolayı güncel teknolojiyi yakalayan şekilde ülkemizin her yanında kullanılamamaktadır. Bu durum hız ölçümünün yapılan saat aralığı ve gece gündüz gibi ölçüm zamanlarını etkilemektedir. Bölgemizde kullanılan radarlarda gece görüş sistemi adapte edilemediğinden gece ölçüm işlemi yapılamamaktadır. Bu durum gece hız sınırlarının aşılması ve dolayısıyla kaza oranını arttırmaktadır. Tasarlanan yazılıma ek olarak oluşturulacak olan çözünürlüğü yüksek, gece görüş özellikli, yakınlaştırma özelliği ve hareket sensörlü kameralarla hız ölçüm işlemi günün her saati otomatik olarak yapılabilir. Böylece mevcut durum iyileşmesi sağlanır. Bölgede meydana gelen kazaların büyük bir oranının akşam ve gece saatlerinde olduğu verilerle sabittir ve bu kazaların büyük bir kısmının temel sebebi yüksek hız ve dikkatsizlik olduğu da göz önüne alındığında tez kapsamında hazırlanan yazılım donanımsal gelişimiyle kullanılması sağlanabilecektir. Özellikle gerçek hız ve radar ölçüm sonuçları arasındaki ufak farklar kullanılabilirliğini arttırmaktadır [23]. Yapılan uygulamada elde edilen verilerin Çizelge 4.4 ile belirtilmiştir. Şekil 4.12’de ise hız hata oranı tespiti gösterilmiştir.

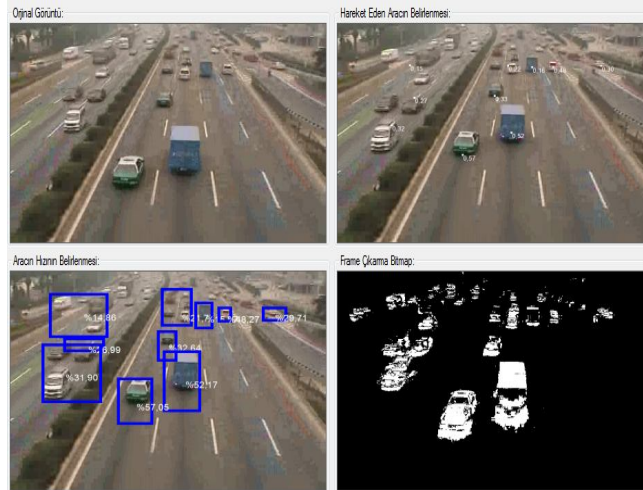


Şekil 4.12. Hız hata oranı tespiti.

Çizelge 4.4. Hız ölçüm sonuçları.

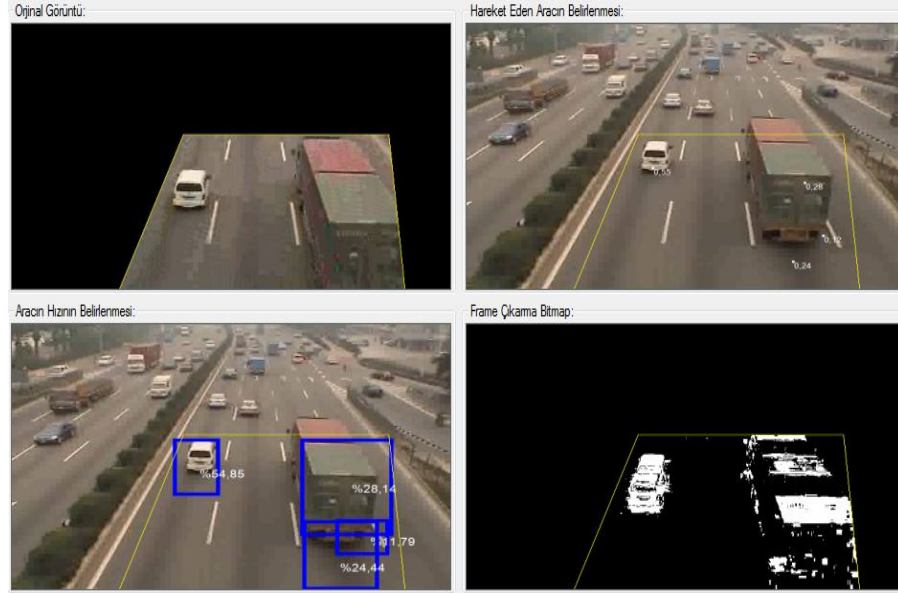
Ölçülen Değer(km/sa)		Gerçek Değer (km/sa)	Hata Oranı(%)	
75	73,58	76	-0,03289	0,0133
65	66,7	68	-0,01949	0,0462
54	57,52	55	0,043811	0,0185
74	78,42	76	0,030859	0,0270
62	63,52	63	0,008186	0,0161
80	81,2	81	0,002463	0,0125
40	41,52	42	-0,01156	0,0500
74	76,58	75	0,020632	0,0135
79	79,86	80	-0,00175	0,0127
58	59,74	60	-0,00435	0,0345
73	75,96	75	0,012638	0,0274
85	84,8	85	-0,00236	0,0000
49	52,4	50	0,045802	0,0204
35	36,2	40	-0,10497	0,1429
70	70,3	68	0,032717	-0,0286
86	89,37	88	0,01533	0,0233
80	82,14	80	0,026053	0,0000

Veriler ışığında uygulamadan elde edilen ölçüm sonuçları, yazılımın doğruluk oranını göstermektedir.



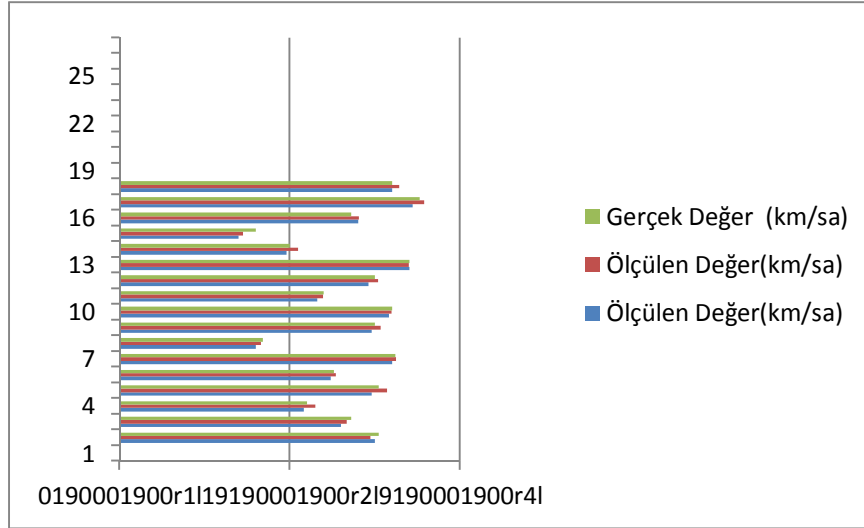
Şekil 4.13. Blob algoritması ve contour algoritması sonuçları.

Şekil 4.13’de uygulama sonucu elde edilen blob algoritması ve contour algoritması sonuçları gösterilmiştir.



Şekil 4.14. Ölçüm alanının belirlenmesi.

Şekil 4.14’de ölçüm işleminin doğruluğunu artırma adına yapılan ölçüm alanının belirlenmesi gösterilmektedir.



Şekil 4.15. Ölçüm sonuçlarının karşılaştırılması.

Şekil 4.15’de uygulama sonucu elde edilen sonuçların karşılaştırılması gösterilmektedir. Ölçülen değer ve gerçek değerler arasında farklar görülmektedir.



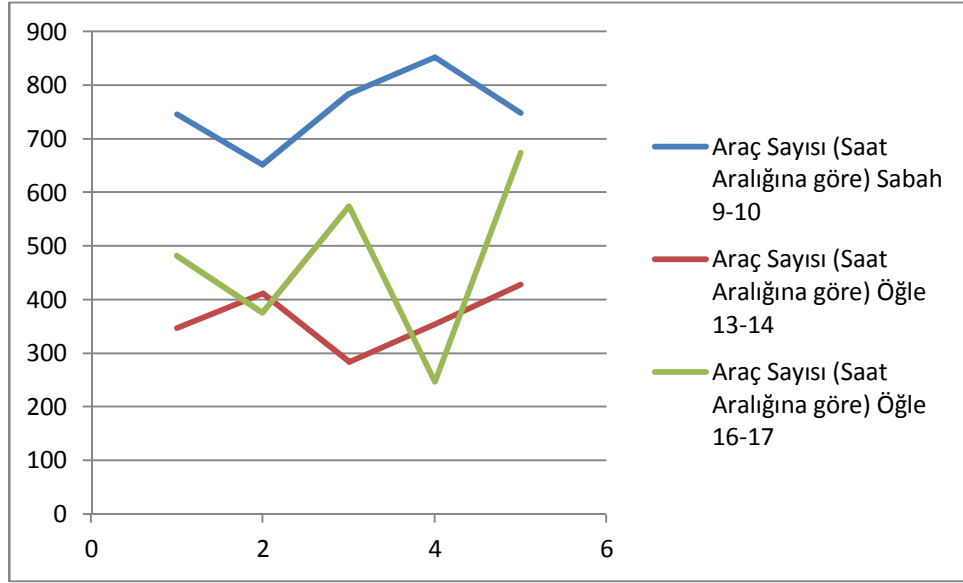
Şekil 4.16. Hareket eden nesnelere id işlemi.

Şekil 4.16’da hareket eden araçların id işlemine tabi tutulması gösterilmiştir. Bu işlem sonucunda Çizelge 4.5’de ölçüm sonuçlarından elde edilen araç sayıları belirtilmiştir.

Çizelge 4.5. Safranbolu-Karabük karayolu araç sayısı tespiti.

Günler	Araç Sayısı (Saat Aralığına göre)		
	Sabah 9-10	Öğle 13-14	Öğle 16-17
Pazartesi	746	347	482
Salı	652	412	376
Çarşamba	785	284	574
Perşembe	853	354	247
Cuma	749	428	675

Şekil 4.17’de araç sayılarının trafik yoğunluğu verisinde kullanılması gösterilmiştir.



Şekil 4.17. Araç sayılarının trafik yoğunluğu verisinde kullanılması.



## BÖLÜM 5

### SONUÇLAR

Plaka tanıma sistemleri, günümüzde otoyol kontrolleri, site ve apartman giriş çıkış kontrollerinde kullanılmaktadır. Bu çalışmada geliştirilen yazılım maliyet noktasında mevcut eş değerleri arasında daha ucuzdur. Hız ölçme işlemi güvenli sürüş ortamı oluşturma ve kontrol anlamında Emniyet Genel Müdürlüğü ve Karayolları Genel Müdürlüğü tarafından ihtiyaç duyulan bir konudur.

Mevcut durumda trafik ekipleri araç sayımını, araç hız tespiti sırasında matbu bir form aracılığı ile yazarak yapmaktadır. Bu formlar ile elde edilen veriler ışığında yerel, bölgesel ve ulusal denetim raporları çıkarılmaktadır. Bu raporlarla araç yoğunluğu, hız limit aşımı, kaza riski, kaza oranı, günlere ve saat aralıklarına göre araç sayılarına bakılmaktadır. Trafik denetim sıklıkları ve konumları belirlenmektedir. Trafik yoğunluğu ölçümü işlemi Karayolları Genel Müdürlüğü tarafından, yola döşenen kablolar aracılığı ile de tespit edilmektedir. Araç hız ölçüm işlemi ise trafik ekipleri tarafından sabit ve hareketli noktalar aracılığı ile yapılmaktadır.

Yapılan bu çalışma ile araç hız ve yoğunluk tespitleri gerçek zamanlı olarak ölçülmüştür. Uygulama sonucu elde edilen sonuçlar neticesinde, yazılımın kullanılabilirliğinin yüksek olduğu görülmektedir. Günün farklı zaman aralıklarında yapılan araç sayımı sonucunda trafik yoğunluğu tespiti başarılı bir şekilde yapılmıştır. Araç sayım ve raporlama işlemi geliştirilen uygulama ile otomasyonel hale getirilmiştir. Donanım ve gerekli network alt yapısı ile yazılım bütünleşik hale getirilmiştir. Manuel olarak yapılan bu işlemler günlük, haftalık, aylık ve yıllık olarak denetim raporlarına adapte edilebilir. Bu durum he iş gücü anlamında hem de diğer sistemlere bütünleşme anlamında yarar sağlayacaktır.

Gelecek alıřmalarda grnt iřleme teknikleri erevesinde geliřtirilen yazılımlar, donanımsal alt yapılarla desteklenip; zellikle gml sistemlerle btnleřik hale getirilebilir. Piyasa řartlarında kullanılabilir ve dođruluđu yksekrnler ortaya konabilecektir. Gelecek alıřmalarda, karayollarına konulacak olan dev kristal ekran ve kamera sistemleri ile hız yapan araların belirlenmesi řeffaf hale getirilebilir. Hız ve plaka verileri ekranda srclere gerek zamanlı olarak gsterilip, uyarma ve gerekli hallerde cezalandırma iřlemi yapılabilir. Bylece trafik kazalarının azaltılmasında geliřtirilen sistem,nemli bir yer tutabilir.

## KAYNAKLAR

1. Gillan W. J., "Prometheus-reducing traffic congestion by advanced technology, in roads and traffic", *International Road and Traffic Conference*, Berlin, Germany, 111-115 (2000).
2. Internet: Emniyet Genel Müdürlüğü Resmi Web Sayfası, "Emniyet Genel Müdürlüğü Son 10 yıllık Denetim Raporları", [http://www.trafik.gov.tr/istatistikler/10\\_yil\\_istatistik.asp](http://www.trafik.gov.tr/istatistikler/10_yil_istatistik.asp) (2012).
3. Çamaşircioğlu E., "Araç plaka algılama ve tanıma", Yüksek Lisans Tezi, *Ankara Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 35-40 (2007).
4. Liju D. and Ge Y., "An optimization-based approach to image binarization", *The Fourth International Conference on Computer and Information Technology CIT'04.*, Wuhan, China, 165-170 (2004).
5. Internet: Encyclopedia of mathematics, "Convolution of Function", [http://www.encyclopediaofmath.org/index.php/Convolution\\_of\\_functions](http://www.encyclopediaofmath.org/index.php/Convolution_of_functions) (2012).
6. Gokay A., "A constraint based real-time license plate recognition System", Yüksek Lisans Tezi, *Ortadoğu Teknik Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 24-30 (2007).
7. Yavuz G., "Plaka tanıma sistemi", Yüksek Lisans Tezi, *Sakarya Üniversitesi Fen Bilimleri Enstitüsü*, Sakarya, 23-56 (2008).
8. Wu C.M., Lei C.O., Chan H.W., and Tong S.K., " License plate recognition system", *The Fourth International Conference on Machine Learning and Cybernetics*, Guangzhou, China, 140-149 (2005).
9. Chang S.L., Chen L.S., Chung Y.C. and Chen S.W., "Automatic license plate recognition", *Intelligent Transportation Systems, IEEE Transactions*, 5 (1): 42-53 (2004).
10. Öztemel E., "Yapay Sinir Ağları 1.Baskı", *Papatya Yayıncılık*, İstanbul, 75-85 (2003).
11. Ghosh N., and Bhanu B., "Incremental unsupervised threedimensional vehicle model learning from video", *IEEE ITS*, 11 (2): 423-440 (2010).

12. Han P., Han W., Wang D. F. and Zhai Y. J., "Car license plate feature extraction and recognition based on multi-stage classifier", *Machine Learning and Cybernetics, International Conference*, Hebei, China, 128-132 (2003).
13. Draghici S., "A neural network based artificial vision System for plate recognition", *International Journal of Neural Systems*, Wayne State University, Detroit, USA, 113-126 (1997).
14. Internet: Quercus Technologies, "The Automatic Number Plate Recognition Tutorial", <http://www.anpr-tutorial.com/> (2012).
15. Zhang Y. and Zhang C., "New algorithm for character segmentation of license plate", *Intelligent Vehicle Symposium*, Ohio, USA, 106-109 (2003).
16. Naito, T., Tsukada T., and Yamamoto, S., "Robust license plate recognition method for passing vehicles under outside environment", *IEEE Trans. Vehicular Technology*, 9 (2): 2309-2319 (2000).
17. Nishiyama, K., Kato, K., and Hinenoya, T., "Image processing system for traffic measurement", *Proceedings of International Conference on Industrial Electronics, Control and Instrumentation*, Kobe, Japan, 1725 (1995).
18. Ponce P., Wang S. S., and Wang D. L., "License plate recognition final report", *Department of Electrical and Computer Engineering, Carnegie Mellon University*, Pensilvania, USA, 34-56 (2000).
19. Hauslen R., A., "The promise of automatic vehicle identification", *IEEE Transactions on Vehicular Technology*, 4 (2): 30-38 (1977).
20. Lin R., Cao X., Xu Y., Wei C., and Qiao H., "Airborne moving vehicle detection for urban traffic surveillance", *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems*, Beijing, China, 12-15 (2008).
21. Wu J., Liu Z., Li J., Gu C., Si M., and Tan F., "An algorithm for automatic vehicle speed detection using video camera", *Proceedings of 2009 4th International Conference on Computer Science & Education*, Suzhou, China, 193-196 (2009).
22. Yoginee B. and Kulkarni P.S., "An implementation of moving object detection, tracking and counting objects for traffic surveillance system", *2011 International Conference on Computational Intelligence and Communication Systems*, Gwalior, India, 143-148 (2011).
23. Glavtchev V., Ozcelik P., Ota M. J. and Owen D. J., "Feature-based speed limit sign detection using a graphics processing unit", *IEEE Intelligent Vehicles Symposium*, Baden, Germany, 195-200 (2011).

## ÖZGEÇMİŞ

Ahmet Nusret ÖZALP 1980 yılında Ankara/Beypazarı'nda doğdu. İlköğretim eğitimini Ankara/Ayaş'ta tamamladı. Liseyi Yenimahalle Anadolu Teknik, Teknik ve Endüstri Meslek Lisesi Bilgisayar Donanım Bölümünde birincilik ile 1999 yılında tamamladı. Aynı yıl Marmara Üniversitesi Teknik Eğitim Fakültesi Elektronik Bilgisayar Eğitimi Bilgisayar ve Kontrol Öğretmenliği bölümünü derece ile kazandı. 2003 yılında mezun olarak, Ağrı Teknik ve Endüstri Meslek Lisesi'nde Bilgisayar Öğretmeni olarak göreve başladı. Aynı zamanda Erzurum Üniversitesi MYO'da Programlama ve Bilgisayar Ağları dersleri verdi. 2004 yılında Safranbolu Kız Meslek Lisesi'ne tayin oldu. 2006 yılında Safranbolu Teknik ve Endüstri Meslek Lisesi'ne tayin olarak, burada Atölye Şefi olarak göreve başladı. Ulusal ve Uluslararası birçok bilişim yarışmalarında derecelerin yanında; ulusal yayın yapan bilişim dergilerinde makaleleri; ulusal ve uluslararası sempozyumlarda bildiri ve çalışmaları yayınlanmıştır. Cisco Yerel Akademi Sorumlusu olarak, 2008 yılından bu yana Cisco Eğitmeni olarak eğitim vermektedir.

Evli ve 1 çocuk babası olan Ahmet Nusret ÖZALP web-grafik tasarım, tenis, yüzme ve fotoğrafçılıkla ilgilenmektedir.

### **ADRES BİLGİLERİ**

Adres : Milli Eğitim Bakanlığı  
Safranbolu Milli Eğitim Müdürlüğü  
Safranbolu Teknik ve Endüstri Meslek Lisesi KARABÜK

Tel : (370) 712 12 69  
E-posta : ahmetnusret@hotmail.com