

**T.C.
MALTEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**GÖRÜNTÜ İŞLEME ALGORİTMALARININ EŞ ZAMANLI
SÜREÇLERE AYRILARAK KABLOSUZ AĞ ÜZERİNDEN
GERÇEKLENMESİ VE PERFORMANS ANALİZLERİ**

YÜKSEK LİSANS TEZİ

Başar TURAN

**Tez Danışmanı
Yrd. Doç. Dr. Erdal GÜVENOĞLU**

İSTANBUL – 2017

**T.C.
MALTEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI**

**GÖRÜNTÜ İŞLEME ALGORİTMALARININ EŞ ZAMANLI
SÜREÇLERE AYRILARAK KABLOSUZ AĞ ÜZERİNDEN
GERÇEKLENMESİ VE PERFORMANS ANALİZLERİ**

YÜKSEK LİSANS TEZİ

Başar TURAN

**Tez Danışmanı
Yrd. Doç. Dr. Erdal GÜVENOĞLU**

İSTANBUL – 2017

Bu tez çalışması, Maltepe Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun / / tarih ve / sayılı kararıyla oluşturulan jüri tarafından ***Bilgisayar Mühendisliği Yüksek Lisans Tezi*** olarak kabul edilmiştir.

JÜRİ

Yrd. Doç. Dr. Erdal GÜVENOĞLU

Danışman

Prof. Dr. A. Mesut RAZBONYALI

Üye

Yrd. Doç. Dr. Nurşen TOPÇUBAŞI

Üye

ÖZET

Yüksek Lisans Tezi, Görüntü İşleme Algoritmalarının Eş Zamanlı Süreçlere Ayrılarak Kablosuz Ağ Üzerinden Gerçeklenmesi ve Performans Analizleri, T.C. Maltepe Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı.

Günümüzde görüntü işlemenin uygulama alanları geçmişte olduğundan daha fazladır ve gün geçtikçe bu çeşitlilik artmaktadır. Temel olarak görüntü işleme tıp ve biyolojide, astronomide, endüstriyel uygulamalar ve mühendislikte, güvenlik, savunma ve hukukta, bankacılıkta, ticarete, sanatta, coğrafyada, biyolojide, fizikte, oyun programlamada, uzay bilimleri başta olmak üzere hemen hemen tüm alanlarda kullanılmaktadır. Dijital görüntü işleme çok kapsamlı bir konu olup birçok matematiksel algoritmaya ihtiyaç duymaktadır. Dijital görüntünün işlenmesinde, geliştirilmek istenen birçok algoritma çok hızlı sistemlere ihtiyaç duymaktadır. Çünkü geliştirilen algoritmalar genel olarak çok fazla işlem yüküne sahiptir. Diğer yandan gelişen teknolojiyle birlikte kablosuz ağ üzerinden görüntü işlemenin sunduğu sayısız avantajın artacağı ve bu teknolojinin uygulama alanlarının daha da genişleyeceği düşünülmektedir. Bu bakımdan görüntü işleme algoritmalarının en son teknolojiye göre uyarlanması oldukça önemlidir.

Bu tez çalışmasında; görüntü işleme algoritmalarının eş zamanlı süreçlere ayrılarak kablosuz ağ üzerinden gerçekleştirilmesi ve performans analizlerinin incelenmesi ele alınmıştır. Çalışmanın ikinci bölümünde literatürdeki benzer çalışmalar özetlenmiş; üçüncü bölümde ise görüntü işlemede kullanılan görüntü, görüntü türleri, piksel, görüntü formatları, görüntü yakalama ve görüntü işleme gibi temel kavramlar tanımlanmış; görüntü işleme aşamaları ve seviyeleri, uygulama alanları, yöntemleri ve görüntü işleme algoritmaları, kablosuz ağ tanımı, çeşitleri, cihazları ve avantajları üzerinde durulmuştur. Dördüncü bölümde ise; görüntü işleme algoritmalarının kablosuz ağ üzerinden gerçekleştirilmesi konusu çerçevesinde masaüstü ve web tabanlı/online görüntü işleme uygulamaları, kablosuz ağ üzerinden görüntü işlemenin kapsamı, uygulama alanları ve avantajları, görüntü işlemede kullanılan hazır arayüzler,

Raspberry Pi ve kablosuz kullanımı ile bu çalışmaya ilişkin uygulamanın aşamaları, test sonuçları ve performans analizleri sunulmuştur.

Çalışmada geliştirilen uygulamanın performans analizlerine göre; kablosuz ağdaki cihaz sayısının artması, algoritmaların toplam görüntü işleme süresini kısaltmaktadır.

2017 yılında yazılmış olan bu tez 110 sayfadan oluşmaktadır.

Anahtar Kelimeler: Görüntü İşleme, Görüntü İşleme Algoritmaları, Kablosuz Ağ, Performans Analizi, Raspberry Pi.

ABSTRACT

Master Thesis, Performance Analysis and Realisation of Image Processing Algorithms over Wireless Networks by Separating Them into Synchronous Procedures. T.C. Maltepe University, Graduate School of Natural and Applied Sciences, Department of Computer Engineering.

Nowadays, the application area of the image processing is more than it was in the past and this variety is increasing day by day. Basically, image processing is used in almost all areas of medicine and biology, astronomy, industrialications and engineering, security, defense and law, banking, commerce, art, geography, biology, physics, game programming, space science. Digital image processing is a very comprehensive subject and requires many mathematical algorithms. In the processing of digital images, many algorithms that are desired to be developed require very fast systems. Because the algorithms developed in general have a lot of processing overhead. On the other hand, with the developing technology, it is thought that the numerous advantages offered by image processing over wireless network will increase and the application areas of this technology will be expanded. In this respect, adaptation of image processing algorithms to the latest technology is very important.

In this thesis, analysis and realisation of image processing algorithms over wireless networks by separating them into synchronous procedures are discussed. In the second Chapter of the study, similar studies in the literature are summarized; in the third Chapter, the basic concepts such as image, image types, pixels, image formats and image capturing used in image processing are defined; image processing steps and levels, application areas, methods and image processing algorithms, the definition, types, devices and advantages of wireless networks are emphasized. In the fourth chapter; the desktop and web-based/online image processing applications, scope and application areas of image processing over wireless network, ready-made interfaces, Raspberry Pi and its wireless use, and the phases of the implementation of this study, its test results and performance analyzes are presented.

According to the performance analysis of the application developed in the study; the increase in the number of devices in the wireless network shortens the total image processing time of the algorithms.

This thesis which was written in 2016 consists of 110 pages.

Keywords: Image Processing, Image Processing Algorithms, Wireless Network, Performance Analysis, Raspberry Pi.

TEŐEKKÖR

Yüksek lisans tezini ve öğrenimim süresince benden bilgi, tecrübe ve desteklerini esirgemeyen saygıdeğer tez danışmanım Sn. Yrd. Doç. Dr. Erdal GÜVENOĐLU' na ve hayatımın her aşamasında sevgi, ilgi ve varlıklarıyla yanımda olan aileme teşekkürü borç bilirim.

Başar TURAN

İÇİNDEKİLER

ÖZET.....	i
ABSTRACT.....	iii
TEŞEKKÜR.....	v
ŞEKİLLER LİSTESİ	x
ÇİZELGELER LİSTESİ	xii
1. GİRİŞ	1
2. LİTERATÜR ARAŞTIRMASI.....	3
3. MATERYAL VE YÖNTEM	13
3.1. Görüntü İşleme ve Temel Kavramlar	13
3.1.1. Görüntü	13
3.1.2. Görüntü türleri	13
3.1.3 . Piksel	14
3.1.4 . Görüntü formatları.....	15
3.1.4.1 Jpeg.....	16
3.1.4.2 Bmp	16
3.1.4.3 Tiff.....	16
3.1.4.4 Gif.....	17
3.1.4.5 Png.....	17
3.1.4.6 Psd	18
3.1.4.7 Pict.....	18
3.1.5 . Görüntü yakalama.....	18
3.1.6 . Görüntü işleme	19
3.1.7 . Görüntü işleme aşamaları	19
3.1.8 . Görüntü işleme seviyeleri.....	19
3.1.9 . Görüntü işlemenin uygulama alanları.....	20
3.1.10 . Görüntü işleme yöntemleri.....	20
3.1.10.1 Gürültü giderme ve görüntü düzenleme	22
3.1.10.2 Renk düzeltme ve dönüştürme.....	22
3.1.10.3 Kenar ve köşe bulma.....	25
3.1.10.4 Geometrik dönüştürme	28

3.1.10.5	Sayısal karıştırma veya optik karıştırma.....	29
3.1.10.6	İleri Görüntü İşleme.....	29
3.2	Görüntü İşleme Algoritmaları.....	30
3.2.3	Kenar belirleme algoritmaları.....	30
3.2.3.1	Gradyent kenar belirleme algoritmaları.....	30
3.2.3.2	LoG (Laplacian of Gaussian) algoritması	31
3.2.3.3	Canny kenar belirleme algoritması.....	32
3.2.4	Köşe saptama algoritmaları	37
3.2.4.1	Hough algoritması	37
3.2.4.2	Harris and Stephens köşe belirleme algoritması	38
3.2.4.3	Susan köşe belirleme algoritması	39
3.2.5	Gürültü giderme algoritmaları	45
3.2.6	Morfolojik İşlemler	47
3.2.6.1	Genişletme ve aşındırma işlemleri	48
3.2.6.2	Açma ve kapama işlemleri	50
3.2.6.3	Diğer morfolojik işlemler	52
3.2.7	Diğer algoritmalar.....	52
3.2.7.1	Görüntü netleştirme algoritmaları	52
3.2.7.2	Gri renk skalasına dönüştürme ve renk düzeltme algoritmaları	53
3.2.7.3	Plaka okuma ve karakterlerini tanıma algoritmaları	54
3.2.7.4	Yüz tanıma algoritmaları	54
3.3	Kablosuz Ağlar	54
3.3.3	Kablosuz ağ tanımı	55
3.3.4	Kablosuz ağ çeşitleri.....	55
3.3.4.1	Kablosuz yerel alan ağları (WLAN).....	55
3.3.4.2	Kablosuz kişisel ağlar (WPAN).....	56
3.3.5	Kablosuz ağ cihazları.....	58
3.3.6	Kablosuz ağların avantajları	58
4	GÖRÜNTÜ İŞLEME ALGORİTMALARININ KABLOSUZ AĞ ÜZERİNDEN GERÇEKLENMESİ	60
4.1	Görüntü İşleme Uygulamaları	60
4.1.3	Masaüstü tabanlı uygulamalar	60
4.1.4	Web tabanlı/çevrimiçi uygulamalar	61

4.2 . Kablosuz Ağ Üzerinden Görüntü İşleme	63
4.2.3 . Kapsamı ve uygulama alanları	63
4.2.4 . Avantajları.....	63
4.2.5 . Görüntü işlemede kullanılan hazır arayüzler.....	65
4.2.6 . Raspberry Pi ve kablosuz kullanımı	65
4.3 . Test Sonuçları	70
4.3.3 . Uygulamaya ilişkin temel bilgiler	70
4.3.3.1 Uygulamada kullanılan temel araçlar	70
4.3.3.2 Uygulamada kullanılan yazılım dilleri ve arayüz.....	70
4.3.3.3 Uygulamanın desteklediği algoritmalar	70
4.3.3.4 Uygulamanın desteklediği resim formatları	71
4.3.3.5 Uygulamanın desteklediği resim dosya boyut sınırı	71
4.3.4 . Uygulama aşamaları	71
4.3.5 . Performans analizleri	78
5. SONUÇ VE ÖNERİLER.....	83
KAYNAKLAR.....	86
ÖZGEÇMİŞ	92

KISALTMALAR

GA	: Genetik Algoritmalar
GIF	: Graphic Interchange Format
GPU	: Graphics Processing Unit
IEEE	: The Institute of Electrical and Electronics Engineers (Elektrik ve Elektronik Mühendisleri Enstitüsü)
IRDA	: Infrared (Kızılötesi)
JPEG	: Joint Photographic Experts Group
LAN	: Local Area Network (Yerel Alan Ağı)
PNG	: Portable Network Graphics
TBBA	: Topolojik Bağımsız Bileşen Analizi
TIFF	: Tagged Image File Format
USAN	: Univalve Segment Assimile Nucleus
Wi-Fi	: Wireless Fidelity (Kablosuz Bağlantı Alanı)
WLAN	: Wireless Local Area Network (Kablosuz Yerel Alan Ağı)
WWAN	: Wireless Wide Area Network (Kablosuz Geniş Alan Ağı)
YSA	: Yapay Sinir Ağları

ŞEKİLLER LİSTESİ

Şekil 3.1. Optik-Analog-Sayısal Görüntü	14
Şekil 3.2. Örnek renkli sayısal görüntüde seçilmiş alanın piksellerinin görünümü... 14	14
Şekil 3.3. Görüntüye histogram eşitleme uygulanması.....	24
Şekil 3.4. Görüntünün kontrastının değiştirilmesi	25
Şekil 3.5. Kenar ve köşe gösterimi	26
Şekil 3.6. Görüntünün Boyutlarının En/Boy Oranı Korunarak Küçültülmesi	28
Şekil 3.7. Görüntünün Boyutlarının Bağımsız Olarak Büyütülmesi	28
Şekil 3.8. Görüntünün Tek Eksen Etrafında Döndürülmesi.....	29
Şekil 3.9. İki Görüntünün Birleştirilmesi	29
Şekil 3.10. Görüntüye LoG algoritması uygulanması.....	32
Şekil 3.11. Normal görüntünün Gaussian filtresi kullanılarak yumuşatılması.	33
Şekil 3.12. Kenar belirlemede piksel değerlerinin ortalamasının alınması.....	36
Şekil 3.13. Kenar belirleme.....	36
Şekil 3.14. Normal görüntünün kenar belirleme filtresi kullanılarak kenarlarının belirlenmesi.....	37
Şekil 3.15. Koordinatların Köşeleri Oluşturması.....	38
Şekil 3.16. Harris and Stephens Köşe Saptama Algoritması ile bir aracın köşelerinin bulunması	39
Şekil 3.17. Beyaz arka zeminde verilen koyu renkli bir dikdörtgenden oluşan basit bir şekil üzerinde farklı alanlarda bulunan dört dairesel maske.....	40
Şekil 3.18. Lokal alanları (USAN'ları) maskelerin beyaz kısımları olarak gösterilmiş olan benzer renkteki dört dairesel maske	41
Şekil 3.19. Kenar ve köşe iyileşmesini gösteren, test görüntüsünün küçük bir kısmı verilen USAN alanının üç boyutlu bir grafiği.....	42
Şekil 3.20. Kenar ve köşe iyileşmesini gösteren, gürültülü gerçek bir görüntünün küçük bir kısmı verilen USAN alanının üç boyutlu bir grafiği.....	43
Şekil 3.21. Gürültünün benzer değerleri içermeye yeterince küçük olması durumunda USAN fonksiyonunda dikkate alınmaması.....	44
Şekil 3.22. Kırpma ve eşikleme dönüşüm fonksiyonları	46

Şekil 3.23. Normal görüntünün kırılması.....	46
Şekil 3.24. Normal görüntüye eşikleme yapılması	46
Şekil 3.25. Morfolojik yapısal öge örnekleri	48
Şekil 3.26. Genişletilmiş ikili görüntü	48
Şekil 3.27. Genişletme işleminin temel mantığı	49
Şekil 3.28. Aşındırılmış ikili görüntü.....	49
Şekil 3.29. Aşındırma işleminin temel mantığı.....	50
Şekil 3.30. Açma işlemi uygulanmış ikili görüntü.....	51
Şekil 3.31. Kapama işlemi uygulanmış ikili görüntü.....	51
Şekil 3.32. Normal görüntüye gri seviyeli görüntüye dönüştürülmesi	53
Şekil 4.1 Raspberry Pi.....	66
Şekil 4.2 Raspberry Pi 3.....	67
Şekil 4.3 Raspberry Pi Bağlantı Noktaları	68
Şekil 4.4 Paralel Görüntü İşleme Uygulamasının Başlangıç Görünümü.....	72
Şekil 4.5 Resmin Yüklenmesi	73
Şekil 4.6 Ağdaki Cihazların Bulunması	73
Şekil 4.7 Algoritmanın Seçilmesi ve Görüntü İşlemenin Başlatılması	75
Şekil 4.8 Resmin Parçalarının Cihazlarda İşlenmesi ve Görüntülenmesi.....	76
Şekil 4.9 Resmin Cihazlarda Ayrı Ayrı İşlenmiş Parçalarının Birleştirilmesi ve Sonuç Resminin Görüntülenmesi	77
Şekil 4.10 Resmin Cihazlarda Canny Algoritması ile Ayrı Ayrı İşlenmiş Parçalarının Birleştirilmesi ve Sonuç Resminin Görüntülenmesi Performans Analizi	78
Şekil 4.11 Canny Algoritmasının Performans Analizleri	79
Şekil 4.12 Resmin Cihazlarda Harris Algoritması ile Ayrı Ayrı İşlenmiş Parçalarının Birleştirilmesi ve Sonuç Resminin Görüntülenmesi Performans Analizi	80
Şekil 4.13 Harris Algoritmasının Performans Analizleri	81
Şekil 4.14 Canny ve Harris Algoritmalarının Kablosuz Ağda 1, 2 ve 3 cihaz Bulunması Durumlarında Görüntü İşleme Performans Analizleri.....	81

ÇİZELGELER LİSTESİ

Çizelge 4.1. Canny ve Harris Algoritmalarının Kablosuz Ağda Karşılaştırılması.....82

1. GİRİŞ

Görüntü işleme konusu son yılların en önemli araştırma alanlarından biridir [1]. Askeri endüstriden sualtı görüntülemeye, robotikten astronomiye (gök bilimi), fizik, sanat, biyomedikalden uzaktan algılama uygulamalarına; coğrafi bilgi sistemlerinden, gözlem ve tahmin uygulamalarına, hayvancılık ve petrol aramaya, gazete ve fotoğraf endüstrisinden trafik, radar, tıp, güvenlik, suç (kriminal) laboratuvarlarına kadar pek çok alanda kullanılmakta olan görüntü işleme; Yapay Sinir Ağları (YSA), dalgacık dönüşümü, yönlendirme süzgeçleri, Genetik Algoritmalar (GA), bulanık mantık, rastgele alan süzgeçleri, tekrarlamalı (iteratif) hücresel resim işleme algoritması, görüntü iletimi gibi pek çok alanda uygulama alanı bulmaktadır. Bunun yanı sıra ürün pazarlama, reklam, yazıları süsleme, çeşitli resim koleksiyonları oluşturma gibi uygulamalarda, tarih, sanat, eğlence, bilim, endüstri, hayvanlar âlemi gibi pek çok alanda görüntülere gerek duyulmaktadır. Teknoloji dünyasındaki hızlı gelişmelere paralel olarak sayısal (dijital) görüntüler çok büyük bir artış göstermekte ve kapsamlarının gittikçe genişlemesi ile birlikte görüntü içeren veri tabanlarının önemi artmaktadır [2].

Dijital görüntü işleme çok kapsamlı bir konu olup birçok matematiksel algoritmaya ihtiyaç duymaktadır [3]. Dijital görüntünün işlenmesinde, geliştirilmek istenen birçok algoritma çok hızlı sistemlere ihtiyaç duymaktadır. Çünkü geliştirilen algoritmalar genel olarak çok fazla işlem yüküne sahiptir. Gerçek zamanlı görüntü işleme algoritmaları gerçeklemek için saniyede en az 15-25 çerçevenin işlenebilmesi gerekmektedir [1].

Bu çalışmada; görüntü işleme algoritmalarının eş zamanlı süreçlere ayrılarak kablosuz ağ üzerinden gerçekleşmesi ve performans analizlerinin incelenmesi ele alınmıştır. Çalışmanın birinci bölümünde görüntü işlemede kullanılan görüntü, görüntü türleri, piksel, görüntü formatları, görüntü yakalama ve görüntü işleme gibi temel kavramlar tanımlanmış; görüntü işleme aşamaları ve seviyeleri, uygulama alanları, yöntemleri ve görüntü işleme algoritmaları ele alınmıştır.

Kablosuz ađların incelendiđi ikinci blmde; kablosuz ađ tanımı, eřitleri, cihazları ve avantajları zerinde durulmuřtur.

Drdnc blmde ise; grnt iřleme algoritmalarının kablosuz ađ zerinden gereklenmesi konusu erevesinde masast ve web tabanlı/evrimii grnt iřleme uygulamaları, kablosuz ađ zerinden grnt iřlemenin kapsamı ve uygulama alanları, rnek uygulama ve alıřmalar, hazır arayzler ile grnt iřleme ve performans analizleri ele alınmıřtır.

2. LİTERATÜR ARAŞTIRMASI

Tatlı' nın [4] çalışmasında; görüntü işleme araçları ve algoritmaları kullanılarak mobil cihazlarda görüntü işleme uygulamaları geliştirmeye olanak sağlayan sunucu-işlemci mantığı ile çalışan bir model tasarlanmıştır; tasarlanan bu model aydınlatma hesabı için kullanılmıştır. Hazırlanan yazılımla cep telefonu gibi mobil alana kurulan sistem dâhili kamerası sayesinde aydınlatma tasarımı yapılacak mekânın en, boy, tavan ve duvar bilgilerinin fotometrik değerlerini alabilme özelliğine sahiptir. Elde edilen bu veriler, mobil arayüz ile sunucuya gönderilmektedir. Sunucuya gönderilen veriler görüntü işleme teknikleri kullanılarak toplam ışık akısı için gerekli formüllere yerleştirilir. Veri tabanından gelen armatür çeşitlerine göre kullanıcının seçtiği armatür ile hesaplanan toplam ışık akısından, armatürden kaç adet kullanılacağı ve yerleştirme konumları uluslararası standartlara göre hesaplanarak bir aydınlatma raporu gönderilir. Böylece aydınlatma tasarım programları mobil hale getirilmiştir. Geliştirilen sistemin mobil arayüzünde Java altyapısı ve OpenCV kütüphaneleri kullanılmıştır. Web altyapısında wcf web servisi ve asp.net kullanılmıştır. Sunucu alt yapısında Windows servisleri, EmguCV ve C#, veri tabanı olarak da MsSQL kullanılmıştır.

Dinçer' in [3] çalışmasında; dijital görüntü işleme konusunda görüntünün işlenmesi için farklı algoritmalar kullanılmış, WebCAM tarafından elde edilen veya kullanıcının bilgisayarında bulunan bir görüntü üzerinde belirlenen dijital görüntü işleme tekniklerinin uygulanması sağlanmıştır. Belirlenen görüntünün uygulanan dijital görüntü işleme tekniklerine göre farklı şekillerde gösterimi, farklı filtreleme yöntemlerine tabi tutulması ve kenar belirleme yöntemi ile son haline getirilmesi ve endüstriyel bir uygulama üzerinde gösterilebilmesi sağlanmıştır. Çalışmada, endüstriyel uygulama için MATLAB üzerinde yapılan görüntü işleme ve arayüz yazılımı dışında elektronik devre ve mekanik gövde tasarımı yapılmış; mikrokontrolör ve motor sürücü devresi ile iki adım motorunun ve bir selenoidin kontrollü kullanımı mekanik kısım ile birlikte sağlanmıştır. Bu sayede kullanıcı tarafından belirlenen görüntü, yine kullanıcının belirlediği parametreler ile son haline getirilerek uygulama düzeneğinde çizdirilmesi sağlanmıştır [1].

Badem'in [5] çalışmasında; kameradan yakalanan görüntülerdeki biber örüntülerin tanımlanarak biberin yönünün tespit edilmesi amaçlanmıştır. Bu amaçla görüntü işleme algoritması olarak sezgisel bulanık mantık kenar çıkarım ve otsu algoritması kullanılmıştır. İşlenen görüntüdeki biber örüntülerinin sınıflandırılması için de geri yayımlı çok katmanlı algılayıcı yapay sinir ağı modeli kullanılmıştır. Sistem performansını artırmak için de GPU (Graphics Processing Unit, Grafik İşleme Ünitesi) teknolojisinden faydalanılmıştır. Kameradan yakalanan görüntülerde, Atanossov tarafından geliştirilen sezgisel bulanık mantık algoritmasıyla biberlerin kenar çıkartımı sağlanmıştır. Sezgisel bulanık mantık kenar çıkartım işleminde, statik bir eşik değeri kullanıldığından algoritmanın, yakın renk tonlarında kenar bulamadığı görülmüştür. Görüntülerin sayısal değerinden otomatik eşik değer belirleme metodu Otsu yöntemi kullanılarak bu sorun ortadan kaldırılmıştır. Sezgisel bulanık mantık kenar çıkartım işlemlerinde, örüntü yakalama basamağında siyah zemin ve beyaz zemin olmak üzere iki farklı zemin rengi kullanılmıştır. Siyah zemin rengiyle yapılan denemelerde, kırmızıbiberin renk tonu ile siyah zemin renginin renk tonu birbirine çok yakın değerler olduğundan, kenar çıkartım algoritmasından istenilen sonuç elde edilememiştir. Beyaz zemin renginde ise renk tonu farkı yüksek olduğu için neredeyse tüm biber örüntüleri kesintisiz bir şekilde ortaya çıktığından çok başarılı sonuçlar elde edilmiştir. Geliştirilen sistemin performansını artırmak için CUDA NVIDIA GTX 480 GPU destekli ekran kartından faydalanılmıştır. Sistem geliştirilirken hem CPU tabanlı olarak çalışabilen, hem de GPU tabanlı olarak çalışabilen 2 uygulama geliştirilmiştir. GPU teknolojisi kullanılarak otsu algoritmasından en fazla 3 kat, Sezgisel Bulanık mantık algoritmasından en fazla 442 kat ve sistemin toplam cevap süresinden en fazla 79 kat hız elde edildiği görülmüştür. Bu hız oranlarının oldukça başarılı olduğu görülmektedir. Siyah zeminli örüntüleri tanımlama işlemi için kullanılan YSA test işleminde CPU tabanlı uygulamada %51 GPU tabanlı uygulamada da %50 doğru sonuç elde edilmiştir. Beyaz zeminli örüntülerde ise YSA test işlemlerinde CPU tabanlı uygulamada %90, GPU tabanlı uygulamada ise %84 başarı elde edilmiştir. Siyah zeminli örüntülerin sınıflandırma başarısının düşük olması nedeniyle uygulanabilirliği bulunmamaktadır. Fakat beyaz zeminli örüntülerin tanımlama işleminde %90 başarı olması nedeniyle oldukça kararlı bir sınıflandırma yapılabileceği görülmüştür. Bu nedenle sistemde kullanılacak örüntülerin zemin renginin beyaz olması gerektiği

görülmüştür. Çalışmada; sezgisel bulanık mantık kenar çıkartım algoritmasının zayıf yönü olan statik eşik belirleme yöntemi yerine, görüntünün sayısal değerlerinden dinamik eşik belirleme metodu olan Otsu yöntemi kullanılarak daha keskin bir görüntü çıkartma algoritması geliştirilmiştir. Geliştirilen bu algoritma diğer algoritmalarla beraber GPU kullanılarak onlarca kat hızlandırılmıştır.

Yüksel'in [6] çalışmasında; tıpta hasta teşhisini ve tedavisini kolaylaştırmak amacıyla kullanılan medikal görüntü verileri (Bilgisayarlı Tomografi'den - Manyetik Rezonans'dan elde edilen) olan DICOM (Digital Imaging and Communications in Medicine) dosyalarının incelenip 3 boyutlu (3D) modelinin ortaya çıkarılması amacıyla kullanıcı ara yüzü bir medikal program geliştirilmiştir. Üç boyutlu modelin oluşturulmasında mesh algoritmaları kullanılmış; hassas, optimum şartlarda doğru ve hızlı bir şekilde mevcut görüntüler incelenerek uygulama yazılımı geliştirilmiştir. Çalışmada Manyetik Rezonans (MR) ya da Bilgisayarlı Tomografi (CT) cihazlarından elde edilen görüntülerin hangisinin daha iyi sonuç vereceği araştırılmış ve karşılaştırılmış; uygun olan görüntü dosyalarıysa (MRI) uygulama yazılımında mevcut veriler olarak işlenmiştir. Geliştirilen uygulama yazılımı sayesinde medikal görüntüleme teknikleri kullanılarak elde edilen DICOM formatlı resim dosyaları, istenilen görüntü işleme algoritmalarından yararlanılarak, farklı uygulamalar için (filtreler, Hausdorff Hesaplamaları gibi) sonuçlar alınması sağlanmıştır.

Alçın'ın [7] çalışmasında; Yapay Sinir Ağı (YSA) tabanlı bir parmak izi tanıma sistemi geliştirilerek, bu problemin çözümüne katkı sağlanmaya çalışılmıştır. Sistemde Gabor filtre tabanlı bir öznitelik çıkarma sistemi ile parmak izi vektörleri üretilmiş; bu vektörler YSA tarafından alınarak karşılık gelen parmak izi tanımlayıcı çıkış vektörlerine dönüştürülmüştür. Sistemin testi için iki ayrı parmak izi veri bankasından alınan görüntüler kullanılmış; çalışma sürecinde filtrelemeler, görüntü işleme ile ilgili işlemler, YSA eğitimi ve test aşamalarında kullanılan süreçlerin tümü MATLAB ortamında gerçekleştirilmiştir. Test sonuçlarına göre, geliştirilen sistem kimlik tanımlama ve tanımadaki hızı ve hassasiyeti ile, alternatif yöntemlerle karşılaştırıldığında tatmin edici ilerlemeler sağlamıştır.

Karakoç'un [8] çalışmasında; görüntü içinde görüntü aramanın hızlı ve verimli bir biçimde gerçekleştirilebilmesi amaçlanmış, bunun için yapay zekâ yöntemlerinden yararlanılmıştır. Görüntü eşleme için yapay sinir ağları ve görüntünün ortalama renk değeri, görüntüdeki renk değerlerinin standart sapması, korelasyon ve görüntülerden kenar tespiti gibi görüntü işleme kütüphanelerine dayalı çeşitli fonksiyonlar; görüntü arama içinse genetik algoritmalar kullanılmıştır. Çalışmada, akıllı arama algoritmaları, hızlı görüntü eşleme yöntemleri ve paralel programlama tekniklerine dayanan bütünleşik bir yöntem önerilmiştir. Görüntü aramada paralel programlama tekniklerinin uygulanması ve arama uzayının daraltılması; görüntü eşlemede eşlemenin hızlandırılması amacıyla, zeki yöntemler önerilerek işlemler gerçekleştirilmiştir. Arama süresini en aza indirebilecek ve görüntü eşlemeyi en verimli şekilde yapabilecek algoritmalar tasarlanarak kaba kuvvet aramalara göre çok daha verimli olacak bir sistem oluşturulmaya çalışılmıştır. Genetik algoritmalar, bir görüntüyü bir başka görüntü içinde aramada en uygun sonuçları verebilecek şekilde; arama ve eşleme süreçleriyse sonuçları doğrulama ile geliştirilmiştir. Paralel programlama için ise algoritmalar "parçala ve fethet" mantığına dayalı olarak tasarlanmıştır. Akıllı algoritmalar ve paralel programlama fikrine dayanan hesaplamasal olarak akıllı bir yol içeren çalışmada; hazırlanan yazılımın uygulanabilir olması için çalışılmış; önerilen tekniklerin başarısı, çok sayıda düşük ve yüksek çözünürlüklü referans ve şablon görüntü üzerinde test edilmiştir. Sonuçlar, önerilen yöntemin eşleşen görüntüleri elde etmede başarılı bir şekilde toplam arama süresini düşürebildiğini göstermiştir.

Karabiber'in [1] çalışmasında; gerçek zamanlı görüntü işleme uygulamaları gerçekleştirmek için tasarlanan hızlı, kompakt ve bağımsız görüntü işleme sistemleri olan Hücresel Sinir Ağları (HSA) tabanlı ACE16k olarak isimlendirilen bir analog işlemciye sahip Bi-i Hücresel ve Q-Eye olarak isimlendirilen bir analog işlemciye sahip Eye-RIS görü sistemleri incelenerek bu sistemler üzerinde görüntü işleme uygulamaları geliştirilmiştir. Çalışmada, Bi-i Hücresel görü sistemi üzerinde gradyan eşikleme kenar belirleme, hareketli nesne sayma ve yeni bir bölütleme algoritması olmak üzere üç farklı görüntü işleme uygulaması için donanıma özel algoritma tasarlanmış ve gerçekleştirilmiştir. İlk olarak, HSA tabanlı ve gradyan eşikleme

yöntemini kullanan bir kenar belirleme algoritması ACE16k işlemcisi kullanılarak gerçekleştirilmiştir. Elde edilen deneysel sonuçlar görsel açıdan gradyan eşikleme kenar belirleme algoritmasının en çok bilinen ve kullanılan kenar belirleme algoritmalarından biri olan Sobel kadar iyi sonuçlar verdiğini göstermekle birlikte zamanlama açısından gradyan eşikleme algoritmasının üstünlüğünü göstermiştir. Daha sonra bölütleme, hareket algılama ve özellik çıkarma teknikleri kullanılarak görüntü içindeki hareketli nesnelerin sayısını hesaplayan hareketli nesne sayma algoritması gerçekleştirilmiştir. Son olarak, IPOT olarak isimlendirilen ikili pencere operatörleri tabanlı yeni bir bölütleme algoritması gerçekleştirilmiştir. Literatürde önerilen iki farklı HSA tabanlı bölütleme algoritmaları ile karşılaştırılarak önerilen IPOT bölütleme algoritmasının etkinliği gösterilmiştir. Eye-RIS görü sistemi üzerinde ise yeni bir hareketli hedefi takip etme algoritması gerçekleştirilmiştir. Bu algoritma bir robot üzerine yerleştirilen Eye-RIS sistemi kullanılarak test edilmiş; test sonucunda üzerinde Eye-RIS sistemi olan robotun rastgele hareket eden bir başka robotu takip ettiği gözlemlenmiştir. Önerilen algoritmaların analog sistemler üzerinde gerçekleştirilerek elde edilen deneysel sonuçlar bu sistemlerin gerçek zamanlı görüntü işleme uygulamaları geliştirmek için çok uygun bir platform olduğunu göstermiş; bununla birlikte deneysel sonuçlar geliştirilen kenar belirleme, hareketli nesne sayma, bölütleme ve hareketli hedefi takip etme algoritmalarının etkinliğini de ispatlamıştır [1].

Kurtulmuş'un [9] çalışmasında; şeftali meyvesinin verim haritalamasına yönelik olarak meyvenin erken gelişme döneminde ve doğal ortamından alınmış sıradan renkli görüntülerinden meyveleri tespit ederek sayabilecek algoritmaların geliştirilmesi ve en iyi algoritma performanslarının ortaya koyulması amaçlanmıştır; algoritmaların geliştirilmesi ve test edilmesi için görüntüler Bursa Barakfaki köyünde yerel bir çiftçiye ait Elegance Lady çeşidi şeftali bahçesinden alınmıştır. Çalışmada histogram eşitleme ve logaritma dönüşümü gibi görüntü işleme tekniklerinden yararlanılarak doğal koşullarda alınmış görüntülerin aydınlanma koşulları zenginleştirilmiştir. Görüntü işleme tekniklerinden yararlanarak geliştirilen algoritmalar renk, şekil ve doku bilgisini kullanılan öznitelik çıkarma yöntemleriyle görüntülerden çıkarmışlardır. Çalışmada olgunlaşmamış şeftali bitkisini renkli görüntülerde saptama

anlamında yeni olan öznitelik çıkarma yöntemleri kullanılmıştır. Çıkarılan özniteliklerle farklı sınıflandırıcıların performanslarını ortaya koymak amacıyla yedi adet sınıflandırıcı (Diskriminant analizi, K-en-yakın komşu, Naive Bayes, regresyon ağaçları, sınıflandırma ağaçları, yapay sinir ağları ve destek vektör makinesi) eğitilerek denenmiştir. Görüntülerde arka plan elemesi yapmak ve potansiyel meyve bölgelerini saptamak amacıyla üç farklı görüntü tarama yöntemi geliştirilmiştir. Algoritmaların meyve olarak sınıflandırdığı alt-pencereler blob analiziyle tekilleştirilip meyve sayıları tespit edilmiştir. Farklı meyve tarama yöntemleri, istatistiksel ve deneysel yollarla belirlenen farklı öznitelik birleşimleri, farklı yapay sınıflandırıcılarının kullanımıyla değişik algoritmalar türetilmiş, eğitim ve test setleri üzerinde denemeler gerçekleştirilmiştir. Geliştirilen algoritmaların performansları farklı aydınlanma koşullarını içerecek şekilde karşılaştırılmıştır. Çalışma kapsamında geliştirilen algoritmaların bazılarında %85’ler düzeyinde saptama başarısı elde edilmiştir. Geliştirilen algoritmalar doğal bahçe koşullarında alınmış görüntülerdeki aydınlanma değişimlerinden fazla etkilenmemişlerdir.

Yıldız’ın [10] çalışmasında; kumaş hata denetimi ile ilgili iki farklı çalışma gerçekleştirilmiştir. İlk çalışma hatalı kumaş görüntülerine, görüntü işleme algoritmaları uygulanmıştır. Hatalı alanın tespitinden sonra şekilsel ve histogram özellikleri çıkartılmıştır. Bu özellikler kullanılarak Fuzzy C-Means(FCM) algoritması ile kümeleme işlemi gerçekleştirilmiş; tespit edilen hatanın kumaş üzerindeki dikey ve yatay koordinatlarının belirlenmesi de gerçekleştirilmiştir. İkinci çalışma ise kaydedilen video üzerinde hatalı görüntünün tespiti yapılmıştır. Hata, K En Yakın Komşu (KEYK), Bayes Ağları (BA) ve Karar Ağaçları (KA) algoritmaları kullanılarak anlık olarak sınıflandırılmıştır. Ayrıca hatanın yeri, cinsi ve kumaşın hangi metresinde yer aldığı belirlenmiştir. İlk çalışmada hatalı görüntüler üzerinde Fourier Analizi, Gabor Filtreler ve Dalgacık Dönüşümü yöntemleriyle uygulamalar yapılmış; geliştirilen Fourier Tabanlı görüntü işleme yönteminin diğer algoritmalarından daha iyi sonuç verdiği gözlenmiştir. Poplin kumaş üzerinde delik, yırtık, yabancı iplik, makine yağı, nope, çözümlü kaçığı ve atkı kaçığı hataları belirtilen görüntü işleme yöntemleri ile tespiti sağlanmıştır. Denim ve kaşe kumaş üzerinde ise çözümlü ve atkı kaçığı hatalarının tespitinde geliştirilen algoritmalar etkisiz kalmıştır. Farklı türdeki kumaş ve hatalar

üzerinde şekilsel ve histogram özellikleri kullanılarak FCM ile yapılan uygulamada ortalama %87 kümeleme başarısı elde edilmiştir. Gruplandırılan hata sayısının az ya da fazla olmasının kümeleme performansını etkilediği tespit edilmiştir. Yapılan ikinci çalışmada 3 farklı kumaş türü üzerinde farklı hatalara sahip olan video üzerinde anlık hata tespiti, sınıflandırma ve yer tespiti gerçekleştirilmiş; hata sınıflandırma işlemi için anlık olarak kaydedilen video üzerinde gri seviye eş oluşum matrisi kullanılarak özellik çıkarımı yapılmıştır. Elde edilen özellikler KEYK, BA ve KA kullanılarak sınıflandırılmış; sınıflandırma işlemi için her bir hata türüne ait 6 adet görüntü, eğitim verisi olarak kullanılmıştır. KEYK algoritması ile yapılan uygulamada ortalama olarak %90'lık bir başarı elde edilmiştir. KEYK le yapılan uygulamalarda diğer yöntemlere göre daha iyi sonuçlar elde edilmiştir.

Akan'ın [11] çalışmasında; sayısal görüntü analizine dayanan bir kontrol sisteminin emülasyonu için gerçek zamanlı görüntü girişlerini bilgisayarın USB portu üzerinden, kontrol sürecine ilişkin gerçek zamanlı veri giriş/çıkışlarını da paralel portu üzerinden desteklenen MATLAB tabanlı bir sistem geliştirilmiştir. Sistem aynı zamanda, düşük ve orta seviyeli görüntü işleme algoritmalarının eğitimi için bir araç olacak şekilde uyarlanmıştır. Geliştirilen sistem, taşıt plakalarının tanınmasına göre çalışan bir otopark bariyer kontrol sisteminin emülasyonu için kullanılmıştır. Bu amaçla geliştirilen bir kural tabanlı taşıt tanıma algoritmasının testleri bu sistem üzerinde yapılmış ve sonuçların tatmin edici olduğu gözlenmiştir.

Arslan'ın [12] çalışmasında; bir Hücrel Sinir Ağı Çok Fonksiyonlu Makine olan Bi-i Hücrel görü sistemi incelenerek bu sistem üzerinde hareketli görüntü işleme uygulamaları geliştirilmiştir. Gerçek zamanlı görüntü işleme uygulamaları gerçekleştirmek için tasarlanan hızlı, kompakt ve bağımsız görüntü işleme sistemi olan Bi-i Hücrel görü sistemi, Analog Hücrel Sinir Ağları (HSA) tabanlı ve ACE16k olarak isimlendirilen bir işlemci ve DSP olarak isimlendirilen Sayısal İşaret İşlemcisi olmak üzere iki işlemci içermektedir. Çalışmada Bi-i Hücrel Görü sistemi üzerinde hareketli görüntü işleme konusunda çeşitli uygulamalar geliştirilmiştir. Bu uygulamaların ilki Prewitt ve Canny kenar belirleme yöntemlerinin Bi-i Hücrel Görü Sistemi ile Matlab motoru kullanılarak gerçekleştirilmesi ve iki yöntemin sonuçlarının

karşılaştırılması çalışmasıdır. Çalışma sonunda elde edilen sonuçlar Canny yönteminin görsel açıdan çok daha başarılı olduğunu ve Bi-i Hücresele Görü Sisteminin Matlab motoru ile oldukça uyumlu çalıştığını göstermektedir. İkinci uygulamada ise Bi-i Hücresele Görü Sistemi üzerinde Edge, Edge8 ve Sobel kenar belirleme teknikleri uygulanmış ve sonuçları değerlendirilmiştir. Üçüncü uygulama olarak yine Bi-i Görü Sistemi üzerinde hareketli bir görüntüdeki nesnelerin tespiti ve bu nesnelerin özelliklerinin belirlenmesi uygulaması gerçekleştirilmiştir. Bu uygulamadan yola çıkılarak tespit edilen nesnenin sınırları içine taşan diğer nesne kalıntılarının temizlenmesi uygulaması geliştirilmiştir. Bu uygulamanın morfolojik işlemler adımı hem ACE16k hem de DSP işlemci üzerinde ayrı ayrı uygulanmıştır. Elde edilen sonuçlar karşılaştırıldığında ACE16k işlemcisinin hız bakımından avantaj sağladığı gözlenmiştir. Çalışmada geliştirilen bir diğer uygulama da bir hareketli görüntü içinde belirlenen bir nesnenin hareketli görüntüden çıkarılması uygulamasıdır. Bu uygulamada da yine ACE16k ve DSP işlemciler kullanılmış ve ACE16k'in hız bakımından sağladığı avantaj ortaya konulmuştur. Son olarak hareketli görüntü içindeki hareketli nesnelerin izledikleri yolu çizen bir uygulama geliştirilmiştir. Söz konusu uygulama da etkileşimli olarak çalışan ACE16k ve DSP işlemcileri üzerinde çalışmaktadır. Bu uygulamanın Bi-i görü sisteminin hedef takip uygulamaları için de etkin şekilde kullanılabileceğini gösterdiği saptanmıştır. Bu uygulamalar için geliştirilen algoritmalar analog sistemler üzerinde gerçekleştirilerek elde edilen deneysel sonuçlar bu sistemlerin hareketli görüntü işleme uygulamaları geliştirmede çok uygun birer platform olduğu; ayrıca elde edilen deneysel sonuçlar geliştirilen nesnenin kalıntılardan temizlenmesi, bir nesnenin hareketli bir görüntüden silinmesi algoritmalarının etkinliğini de ispatlamıştır.

Karaduman'ın [13] çalışmasında; kamera ve lazer teknolojileri kullanılarak ve üç boyutlu görüntü işleme algoritmaları geliştirilerek temassız ray profil ölçümü yapılmıştır. Lazer kamera ile raydan alınan görüntüler işlenerek arızalı bölgelerin tespit edilmesi, yine lazer kamera ile alınan mesafe ve koordinat verileri ile arızanın olup olmadığı garantilenmiş; lazer teknolojileri ile elde edilen mesafe ve koordinat verileri ile lazer ışınların rayın şekline göre kırılması kamera ile elde edilen iki boyutlu görüntüye üçüncü bir boyut kazandırmıştır. Bu şekilde raya ait derinlik bilgisi elde

edilebilmiştir. Gerçekleştirilen uygulamalarda görüntüye dair gürültüler Gaussian ve medyan filtreleme teknikleri uygulanarak yok edilmiştir. Gürültüden arındırılan görüntülerden faydalı veri elde edebilmek için Watershed dönüşümü gibi dönüşümler, Roberts, Canny, Prewitt gibi kenar çıkarım teknikleri ve Boundary Detection gibi sınır algılama teknikleri uygulanmıştır. Çalışmanın, daha düşük maliyetlerle ray profil analizi yapılacağına dair bir başlangıç adımı sağlayacağı düşünülmektedir.

İlhan'ın [14] çalışmasında; dört rotorlu insansız hava araçları için görüntü işleme tabanlı akıllı kontrol algoritmalarının geliştirilmesi amaçlanmıştır; bu amaçla çalışmada öncelikle tip-2 bulanık sistemlerin incelemesi ve optimizasyon çalışması yapılmıştır. İkinci olarak dört rotorlu insansız hava aracının matematiksel modeli oluşturularak MATLAB simülasyonu ile doğrulama işlemleri gerçekleştirilmiş ve sonrasında tip-2 bulanık sistemlerle pozisyon kontrolü yapılmıştır. Tip-1 bulanık sistem, PD ve tip-2 bulanık sistem kullanılan pozisyon kontrolünde simülasyon sonuçları karşılaştırmalı olarak verilmiş ve başarılı sonuçlar alınmıştır. Çalışmada son olarak dört rotorlu aracın görüntü işleme tabanlı pozisyon kontrolü gerçekleştirilmiştir. Bu yaklaşımda dört rotorlu aracın üzerinde bulunan bir kameradan alınan görüntülerin işlenmesi ile aracın bir nesneyi takip etmesi için gerekli kontrol algoritmaları sunulmuştur. Üç eksenli (ileri-geri, sağ-sol, aşağı-yukarı) olarak gerçekleştirilen kontrol yaklaşımı için elde edilen simülasyon sonuçları, bir nesnenin tespit edilerek takip edilmesi için görüntü işleme tabanlı tip-2 bulanık kontrol algoritmasının doğru, sağlam ve başarılı sonuçlar verdiğini göstermiştir. Sonuç olarak çalışma; dört rotorlu bir insansız hava aracının modellenmesi ve simülasyonu, tip-2 bulanık sistemlerin gerçekleştirilmesi ve optimizasyonu, dört rotorlu insansız hava aracının tip-2 bulanık mantık tabanlı pozisyon kontrolü ve dört rotorlu insansız hava aracının görüntü işleme tabanlı pozisyon kontrolü olmak üzere dört temel katkı noktasına sahip olduğu ispatlanmış ve geliştirilen algoritmalar birçok simülasyon sonucu ile doğrulanmıştır.

Özgül'in [15] çalışmasında; gömülü sistemlerin yüz tanıma nasıl performans gösterdikleri araştırılmış, iki farklı arm-tabanlı gömülü sistem ve bir tane intel-tabanlı sistem performans ölçümleri ile karşılaştırılmıştır. Bir video dosyasının resim kaynağı olarak kullanıldığı çalışmada; ilk olarak bu üç sistemde yüz bulma çalışması uygulanmış ve zaman performansları ölçülmüş; daha sonra üç farklı yüz tanıma algoritması, Fisherfaces, Eigenfaces ve Local Binary Pattern Histograms (LBPH) seçilmiştir. Videodan alınmış resimlerden bir veri tabanı oluşturulmuş ve oluşturulan veri tabanı üzerinde seçilen üç algoritma, daha önce belirtilen üç sistem üzerinde çalıştırılmıştır. Zamanlama performansı ve algoritmaların doğruluk oranları karşılaştırılmış ve yüz tanıma söz konusu olunca gömülü sistemlerin nasıl davrandığı belirlenmeye çalışılmış; performans analizi yapılarak performansın nasıl artırılacağı araştırılmıştır.

Kurt'un [16] çalışmasında; ilk aşamada iki klinik protokol, beyin ve genel Spect (Single Photon Emission Computerized Tomography), beş farklı spect sisteminde karşılaştırılmıştır. Jaszczak fantomunun tomografik görüntüleri her bir sistemde iki protokolde de sistemde mevcut görüntü alma ve görüntü işleme parametreleri ile elde edilmiştir. Çalışmanın ikinci aşamasında ise Jaszczak fantomu ile standart bir protokolde her sistemde tomografik çalışmalar yapılmıştır. Kesit görüntülerini elde etmek için Projeksiyon verileri diğer sistemlere aktarılmıştır. Sonuçlar incelendiğinde; ilk aşamada alınan verilerde kontrast (zıtlık) değerlerinin sistemler arasında 1.78 ile 1.91 oranında farklılık gösterdiği gözlenmiştir. İkinci aşamada sistemler arası benzer kesit elde etme algoritmalarının doğruluğu araştırılmış; buradaki maksimum kontrast değişiminin 1.24 olduğu görülmüştür. Aynı zamanda benzer görüntü toplama ve kesit algoritmaları kullanılarak Spect sistemlerinin performansları da test edilmiş ve bu karşılaştırmada 1.36 oranında fark elde edilmiştir. Sonuç olarak; bir sistemden diğerine gidildiğinde üretici tarafından benzer protokollerde önerilen klinik protokoller arasında farklar olduğu gözlenmiş; kullanıcıların aynı amaca yönelik protokollerde standardizasyonu sağlayabilmesi için fantom çalışmaları ile bu protokollerde gerekli değişiklikleri yapması önerilmiştir.

3. MATERYAL VE YÖNTEM

Bu bölümde gerçekleştirilecek çalışmada hayati öneme sahip görüntü işleme ve görüntü işlemede kullanılan temel kavramlar, görüntü işleme algoritmaları ve kablosuz ağlara ilişkin temel bilgiler ve tanımlara yer verilmiştir.

3.1. Görüntü İşleme ve Temel Kavramlar

Bu bölümde görüntü, görüntü formatları, görüntü yakalama ve görüntü işleme kavramlarının temel tanımları yapılacaktır.

3.1.1. Görüntü

Görüntü; resim, fotoğraf veya videoyu oluşturan her karedeki resim, bilgisayar ortamında elle veya otomatik olarak yaratılmış gerçek veya sanal grafikler gibi görsellere verilen ortak addır [2].

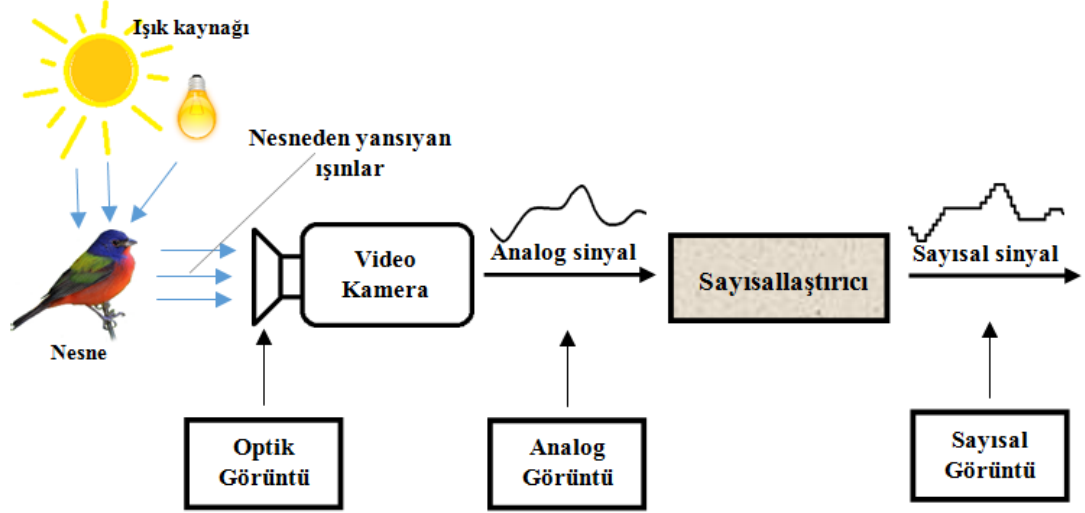
3.1.2. Görüntü türleri

Temel olarak üç tür görüntü vardır: Bunlar:

- Optik görüntü,
- Analog görüntü
- Sayısal görüntü.

Bu üç görüntü türü, görüntünün yakalanmasından sayısallaştırılmasına kadar geçen aşamalarda dönüşerek devam eder. Şekil’de görüleceği gibi; bir ışık kaynağından herhangi bir nesneye çarparak yansıyan ışınlar, cam, mercek gibi çeşitli kırıcı, bükücü, yansıtıcı ortamlardan (fotoğraf makineleri ve kameralar gibi) geçerek analog sinyaller aracılığıyla analog görüntüye dönüştürülür ve ardından sayısallaştırıcı ya da dijitalleştirici bir cihaz/araç vasıtasıyla sayısal/dijital görüntüye dönüştürülür [2].

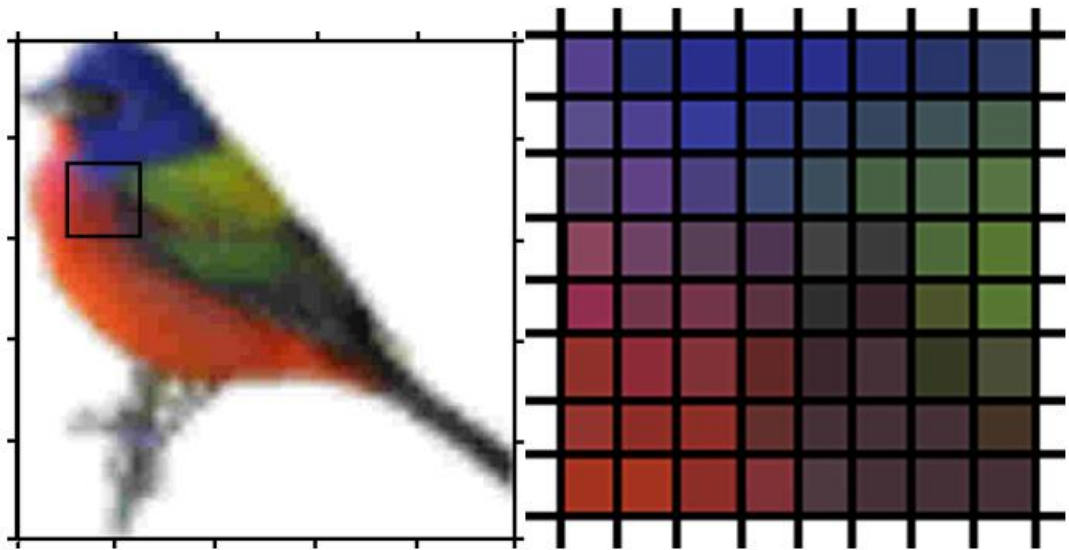
Şekil 3.1’ de Optik-Analog-Sayısal Görüntü gösterilmektedir[17].



Şekil 3.1. Optik-Analog-Sayısal Görüntü

3.1.3. Piksel

Temel yapısı Şekil 3.2’ de gösterilen bir dijital görüntü, satır ve sütun indisleri görüntü içerisinde herhangi bir noktayı tanımlayan elemanlardan meydana gelmiş bir matris olarak göz önüne alınabilir.



Şekil 3.2. Örnek renkli sayısal görüntüde seçilmiş alanın piksellerinin görünümü

Bu matriste her elemanın içerdığı dijital değer ya da değerler, o noktanın rengi hakkındaki bilgiyi barındırır. Bu dijital dizinin ya da matrisin her elemanına “görüntü elemanı” ya da “piksel” denir [18]. Bir diğer deyişle; bilgisayar ekranlarında oluşan görüntüler, farklı renklere sahip olabilen kare şeklindeki birimlerden oluşur. Çok yakından bakıldığında ya da görüntü büyütüldüğünde bu birim kareler fark edilebilir. Ekranda kontrol edilebilen bu en küçük birim karelere piksel adı verilir. Bitmap (piksel tabanlı) resim programları, ölçü birimi olarak piksel kullanırlar. Piksel yoğunluğu arttırılırsa görüntü netleşir. Çok fazla arttırıldığında ise dosyanın kapasitesinin artmasına; yazdırılırken zaman kaybı ve mürekkep sarfiyatına yol açar [19]. Şekil 3.2’de örnek olarak verilen 50x50 piksellik renkli sayısal bir görüntünün seçilmiş 8x8 piksellik belli bir alanının büyütülmüş hali görülmektedir. Soldaki görüntüde kare içinde gösterilmiş alan, sağda büyütülmüş olarak görülmektedir. Sağdaki 8x8 piksellik görüntü kesitinde her kare bir pikseli ifade etmekte olup her piksele atanmış renk değeri kendi sınırları içinde sabittir.

3.1.4. Görüntü formatları

Yaygın olarak kullanılan görüntü formatları şunlardır.

- JPEG (Joint Photographic Experts Group; jpg, jpeg),
- BMP (Bitmap; .bmp),
- TIFF (Tagged Image File Format; .tif),
- GIF (Graphic Interchange Format; .gif),
- PNG (Portable Network Graphics; .png),
- PSD (Photoshop Document; .pdf)
- PICT

3.1.4.1 Jpeg

JPEG (Joint Photographic Experts Group; jpg, jpeg) formatı, ileri teknoloji bir sıkıştırma özelliği olan görüntü formatıdır. Görüntünün daha küçük bir hale gelebilmesi için algılayıcı ve işlemciden gelen bütün görüntü verilerinin boyutunu küçültür [20]. JPEG sıkıştırma yöntemi, görüntünün algılanması için zorunlu olmayan detayları bulup atmakta ve dosyayı bu şekilde sıkıştırmaktadır. Kayıplı bir türdür. Kaybolan ayrıntılar ve sıkıştırma oranı arasında ilişki vardır. Fazla sıkıştırma fazla detay kaybı anlamına gelmektedir [21]. JPEG'den ne kadar sıkıştırma istendiği 0-100 arası bir faktör seçilmesiyle gerçekleştirilmektedir [19]. Sıkıştırma oranının yüksek seçilmesi halinde görüntü kalitesi çok düşük olabilmekte ve görüntüdeki istenilen detaylar kaybolabilmektedir [21].

3.1.4.2 Bmp

BMP (Bitmap Format; .bmp), Windows ve Microsoft'un PCX formatını değiştirerek geliştirdiği bir formattır. Paint programı görüntüleri bu formatta işlemektedir. BMP formatı 1-24 bit arasında değişen bir piksel derinliğini içerebilir. BMP, herhangi bir sıkıştırma yapmadan resmin özelliklerini tutan bir resim dosyası biçimidir. Sıkıştırma yapmadığı için PNG, JPEG gibi dosya biçimlerine göre çok daha fazla yer kaplamaktadır [19]. Bu formatta, görüntü içindeki pikseller değil görüntünün boyutları önemlidir. 24 bitlik 300*300 lük bir görüntü, 270000 byte yer kaplar. Görüntünün boş veya dolu olması sonucu değiştirmemektedir [21]. BMP görüntüleme formatındaki en büyük dezavantaj, dosya boyutunun yüksek alan kaplamasıdır. BMP formatındaki bir görüntünün büyük yer kaplamasının ana nedeni görüntüye ait ölçülere göre işlem yapmasından ileri gelmektedir [20].

3.1.4.3 Tiff

TIFF (Tagged Image File Format; .tif) formatı “Etiketlenmiş Görüntü Dosya Formatı” anlamına gelir. Bilgisayarlar arası ortak bir dosya formatıdır [19]. TIFF imajları herhangi bir sıkıştırma içermediği için büyük boyutta ve yüksek kalitededir [20]. Bu

nedenle, yüksek düzeyde görüntü kalitesine ihtiyaç duyulan uygulamalarda, görüntü “tiff” formatında kaydedilir ve görüntülerde kayıp olmaz. Çok sayıda renk derinliğini desteklemektedir. Dosya boyutu JPEG’e oranla oldukça yüksektir [21].

3.1.4.4 Gif

GIF (Graphic Interchange Format; .gif) grafik deęiřtirme formatı, 256 renk desteęine sahip iyi bir sıkıřtırma algoritması vardır. Farklı tiplerde görüntüler için oldukça güçlü bir görüntü formatıdır. Özellikle görüntüleme işleminde oldukça hızlıdır [19, 20]. Bu nedenle GIF dosyaları internet üzerinde oldukça yaygın kullanılan bir formattır. 256 renge kadarki (1 ila 8 bitlik) görüntülerde oldukça iyi sıkıřtırma sağlamaktadır. Animasyonlarda zamanlama ve farklı boyutlardaki resimleri bir arada tutma desteęi, saydam renk tanımlanması vardır. GIF dosyaları Bitmap, gri skala ve indekslenmiş renk sisteminde olabilmektedir. Gerçek renk desteęi yoktur. GIF resimleri sıralı (interlaced) ya da sırasız kaydedilebilmektedir [19]. Hareketli (animated) GIF’ler bir dosya içine birden çok GIF görüntüsü konularak oluşturulmaktadır. Bu şekilde çeşitli animasyonlar oluşturulabilmektedir [21].

3.1.4.5 Png

PNG (Portable Network Graphics; .png), taşınabilir aę grafikleri anlamına gelir. PNG kayıpsız Wave Table sıkıřtırma yöntemini kullanır. PNG dosyalarındaki saydamlık bilgileri alfa kanalı içerisinde saklanmaktadır [21]. Sıralı yükleme de olanaklıdır. Ayrıca sıkıřtırma için deęişik filtreleme algoritmaları sıkıřtırma öncesi kullanılabilir [19]. Fazladan bir alpha kanalıyla orijinal dosyanın katman özelliklerini de saklar (böyle kaydedildiğinde RGB yerine RGBA resmi olur. Gri seviye ve alfa kanalı bilgilerini içeren görüntüler de vardır. Çizim veya grafik içeren resimlerde iyi sonuçlar vermektedir [21]. PNG görüntü formatının önemli bir özellięi de, herhangi bir uygulamada kullanılan PNG görüntüsünün, uygulamayı destekleyen başka bir PNG formatıyla okunabilmesidir. Network sistemlerinde PNG, GIF’in üstünde bir performans sağlamaktadır. Özellikle deęiřtirilebilir şeffaflık

(transparency) ve görüntü parlaklığının platform kontrolü üzerinde oldukça etkilidir [20].

3.1.4.6 Psd

PSD (Photoshop Document) Photoshop uygulamasına özel bir formattır. PSD çok sayıda alfa kanalını, path'ı ve katmanı desteklemektedir. PSD dosyaları ikili dosya, indekslenmiş renk, gerçek renk RGB, CMYK, Lab biçimlerini destekler. Çalışması yapılan işlerin PSD'sinin saklanması alışkanlık haline getirilmesi, daha sonra yapılacak düzeltmelerde oldukça işe yarar olacaktır [19].

3.1.4.7 Pict

PICT formatı bütün programların ortak kullandığı dosya formatıdır. Bu format herhangi bir uygulama programına aktarıldığında resim bilgisi sayfaya dâhil olur. Görüntü diskten silinse bile baskı aracı sayfayı basılırken PICT dosyasını aramaz. Logo düzenleme ve yeniden oluşturma işlemleri sırasında PICT olarak kaydedilmiş görüntü, Illustrator ve FreeHand programlarına aktarıldığında, Illustrator programı isteğe bağlı olarak dosya aç komutuyla görüntüyü açar ve sayfaya Template (yarı saydam arka plan) olarak yapıştırır. Böylece, bu görüntü üzerinden vektörel olarak çizim yapılabilir [19]. Ancak renk ayırımında, bazı yazılımlarda sorunlara yol açması ve yapısı itibarıyla çabuk bozulması nedeniyle fazlaca kullanılmamaktadır [20].

3.1.5. Görüntü yakalama

Görüntü yakalama, gerçek yaşamdaki bir nesne için herhangi bir sahnenin fotoğraf makinesi veya kamera gibi donanımlarla fotoğrafının çekilmesidir. Görüntü yakalama, film ve benzeri hareketli görüntülerden programlar aracılığıyla da gerçekleştirilebilir [2].

3.1.6. Görüntü işleme

Görüntü işleme, giriş sinyali olarak bir “görüntü”nün kullanıldığı, sinyal işleme formlarından birisi olup; sayısal bir resme dönüştürülmüş görüntülerin çeşitli ihtiyaç ve amaçlar çerçevesinde farklı şekillerde işlenmesi, özellik ve görünümde değişikliğe uğraması sonucunda yeni bir görüntü elde edilmesi yöntemidir [2].

3.1.7. Görüntü işleme aşamaları

Görüntü işleme aşamaları, görüntünün hangi amaca yönelik işleneceğine bağlı olarak değişkenlik göstermekte birlikte, görüntü işlemede temel aşamalar; gürültü giderme ve görüntü düzenleme (görüntü kalitesini yükseltme, görüntü netleştirme vb.), renk düzenleme, (Parlaklaştırma, keskinlik ayarı, renk dönüştürme), kenar belirleme, köşe saptama, bölütleme (görüntünün nesne ya da alanlara bölünmesi), görüntü tanıma, sınıflandırma ile algılama (tanınan nesnelere ilişkin görüntü analizi yardımıyla görmeyle ilişkili bir kavram oluşturulması) olarak sıralanabilir [2, 17, 22, 23]. Bu seviyeler aynı zamanda birer görüntü işleme yöntemidir. Bu yöntemler Bölüm 2.5’de ayrıntılı olarak ele alınacaktır.

3.1.8. Görüntü işleme seviyeleri

Temel olarak görüntünün bilgisayarlaştırma ya da dijitalleştirme seviyeleri 3 kısımda incelenebilir. Bunlar; düşük, orta ve ileri seviyeli görüntü işlemedir. Görüntünün düşük seviyede işlenmesinde; gürültünün azaltılması, kontrastın ayarlanması, görüntünün keskinleştirilmesi gibi basit işlemler yer alır. Orta seviyede işlenmesinde: görüntünün bölütlenmesi (görüntünün nesne ya da alanlara bölünmesi), bilgisayarda bu görüntülerin uygun formlarda işlenebilmesi amaçlı olarak nesnelere tanınması ve sınıflandırılması gibi işlemler yer alır. Orta seviyede görüntü işlemede; girdi olarak görüntü alınırken, çıktı olarak bu görüntüdeki köşe ve kenarlar belirlenmiş, içerikteki tüm nesnelere tanınmış olur. Görüntünün yüksek seviyede işlenmesi ise algılama ile ilgilidir. Bu seviyede görüntü işlemede; tanımlanan bireysel nesnelere yönelik yapılan görüntü analizinin yardımıyla “görme” ye ilişkin bir kavram oluşturulur [17].

3.1.9. Görüntü işlemenin uygulama alanları

Günümüzde görüntü işlemenin uygulama alanları geçmişte olduğundan daha fazladır ve gün geçtikçe bu çeşitlilik artmaktadır. Temel olarak görüntü işleme tıp ve biyolojide (kemik kırıklarının tespiti, tümör tespiti, biyomedikal görüntüleme vb.), astronomide (uydu fotoğrafları vb.), endüstriyel uygulamalar ve mühendislikte (film endüstrisi, TV, materyal testi, uzaktan algılama, hareketli yüzeylerin incelenmesi, robotbilim, tekstil ve gıda endüstrilerinde süreç ve ürün denetimi, barkod okuma, belge işleme vb.), güvenlik, savunma ve hukukta (parmakizi tanıma, plaka tanıma sistemi, iris, yüz tanıma, sembol tanıma, el hareketi ve işaret tanıma, kâğıt para tanıma, resim ya da görüntü tanıma, hedef belirleme, GPR-mayın tarama, gece görüş, akıllı roket sistemleri vb.), sporda (sporçunun hızını bulma, topun gol çizgisini geçip geçmediğini tespit etme vb.), bankacılıkta, ticarete, sanatta, coğrafyada (hava ve uydu görüntülerinden hava tahmini vb.), biyolojide, fizikte (spektrometreler, elektron mikroskobu görüntüleri), oyun programlamada (bilgisayarda görü, 3-B modelleme vb.), uzay bilimlerinde (uydu, mikrodalga radar görüntüleri vb.), yani neredeyse tüm alanlarda kullanılmaktadır [7, 22-25].

3.1.10. Görüntü işleme yöntemleri

Görüntü işlemek için kullanılan yöntemler, ilgilenilen probleme uygun olarak geliştirilebilirler. Uygulamalar, çeşitli yöntemlerin birlikte kullanılmasını, birtakım ölçütlerin dâhil edilmesini ve işlem hızlarının artırılmasını gerektirebilirler. Görüntü işlemek için farklı alanların farklı yöntemleri kullanılabilir. Görüntü işlemek için kullanılan teknikler, problemlere göre başka yöntemlerle çalışılmasını gerektirebilirler. Görüntülerin modellenmesi, görüntülerden çeşitli verilerin çıkarılması, görüntü karşılaştırılması, benzerlik ölçümleri, görüntü aramaları, görüntü eşlemeleri gibi uygulamalar ya da İTGE sistemleri çok farklı yöntemlere gereksinim duyabilirler. Yöntemlerin seçimi, üzerinde çalışılan görüntülere, beklenen çıkışlara ve uygulama sürecine bağlıdır. Üzerinde çalışılan görüntüler çok farklı çözünürlükte veya bit derinliğinde olabilirler ve gürültü içerebilirler. Görüntü işleme uygulamaları, uygulamanın bilgisayar ortamında en uygun şekilde tasarlanmasını ve uygulama süreci

için etkili iyileştirme yöntemlerini gerektirebilir. Öte yandan, gerçekleştirilen uygulamaların gerçek hayatta uygulanabilir, kullanıcı etkileşimli ve çeşitli koşullara karşı sonuç verebilen bir yapıda olması gerekebilir. Tüm bu gereksinimler göz önüne alınarak görüntü işleme için gerekli altyapı oluşturulmalı ve işlemler bunlara dikkat ederek gerçekleştirilmelidir [8].

Görüntü işlemede, farklı amaçlar için kullanılan Fourier dönüşüm metodu, Histogram eşitleme, Medyan (Orta) filtreleri, Gabor süzgeç metodu, dalgacık dönüşümü metodu, Topolojik Bağımsız Bileşen Analizi (TBBA) vb. birçok yöntem bulunmaktadır [10]. Perihanoğlu [18], görüntü işleme yöntemlerini 3 grupta toplamıştır. Bunlar:

- Nokta işleme teknikleri
- Görüntü zenginleştirme yöntemleri
- Morfolojik işlemler

Negatif görüntü, görüntü histogramı, histogram eşitleme, parlaklık ayarlama, kontrast geliştirme ve eşikleme yöntemleri “nokta işleme tekniklerine girerken; görüntü yumuşatma, görüntü keskinleştirme ve kenar belirleme yöntemleri “görüntü zenginleştirme yöntemleri” arasına girmektedir. Genişletme ve aşındırma morfoloji operatörleri, açma ve kapama morfoloji operatörleri ve diğer morfoloji operatörleri ise morfolojik işlemler arasındadır [18]. Karakoç [8] görüntü işleme yöntemlerini yedi grup altında toplamıştır. Bunlar:

- Görüntü Dönüşümleri (Fast Fourier Dönüşümü, Hadamard Dönüşümü, Cosine Dönüşümü),
- Görüntü İyileştirme (Uzaysal Alanda İyileştirme, Frekans İyileştirmeleri),
- Görüntü Onarma (Bozulma Modelleri, Blok-Circulant Matrisleri, Ters Filtreleme),
- Görüntü Bölümleme (Süreksizlik Yakalama [Nokta yakalama, çizgi yakalama, kenar belirleme], Eşik/Alan Belirleme [Eşik belirleme, alan yakalama]),
- Görüntü Sıkıştırma (Kayıplı Sıkıştırma, Hata-Serbest Sıkıştırma),
- Görüntü Sunma (Sınır Tanımlama, Alan Tanımlama),

- Görüntü Algılama (Karar-Teori Metotları, Yapısal Metotlar).

Görüntü işleme yöntemleri, temel işlevleri çerçevesinde değerlendirildiğinde, başlıca görüntü işleme yöntemlerini beş grupta toplamak mümkündür. Bunlar [2]:

1. Gürültü giderme ve görüntü düzenleme
2. Renk düzeltme ve dönüştürme
3. Kenar ve köşe bulma
4. Geometrik dönüştürme
5. Sayısal karıştırma veya optik karıştırma

3.1.10.1 Gürültü giderme ve görüntü düzenleme

Görüntü işleme aşamalarının ilki ve görüntü işleme yöntemlerinin en önemlilerinden biri gürültü giderme ve görüntü düzenlemedir. Gürültü gidermede kırpma ve eşikleme (clipping and thresholding) yöntemleri uygulanmaktadır [24]. Bu yöntemler arasında en bilinenleri düzleştirme/yumuşatma (smoothing) filtresi olarak da anılan ve görüntüdeki gürültüyü gidererek daha yumuşak bir görünüm kazandıran Gaussian filtresi'dir [23].

Görüntü sıkıştırma ise; differential coding, variable length coding, Fourier Transform ve benzeri teknikleri içeren algoritmalar kullanılır [25]. Görüntü netleştirme algoritmaları Bölüm 3.2.5.1'de, gürültü giderme algoritmaları Bölüm 3.2.3'de detaylı olarak ele alındığından burada detaylandırılmayacaktır.

3.1.10.2 Renk düzeltme ve dönüştürme

Parlaklaştırma, keskinlik ayarı, histogram eşitleme, renk dönüştürme gibi işlemler renk düzeltme ve dönüştürme tekniklerine girmektedir. Her resimde gri ton dağılımı farklı olduğu için gri ton dağılım skalasında (histogram) homojenliği sağlamak üzere histogram eşitleme kullanılmaktadır [23]. Histogram eşitleme işleminde, resmin

kümülatif gri seviye dağılım skalası üzerinde normal dağılım uygulanmaktadır. Bu yeniden dağılım, gri seviye dağılımında dengeleme sağlamaktadır [26].

Histogram eşitleme kontrast zenginleştirme yöntemlerindedir. Görüntünün tümüne uygulanabileceği gibi sadece belli bir bölgesine de uygulanabilen histogram eşitlemesi ile görüntüde parlaklık en çok toplandığı alana en büyük kontrast zenginleştirme uygulanır. Tüm görüntüye uygulanırsa “global histogram eşitleme”, görüntünün belli bir bölgesine uygulandığında ise “lokal histogram eşitleme” adını alır. Bu yöntem histogramı dar olan görüntüler veya görüntü içindeki bölgeler için daha iyi sonuç verir [27, 28]. Görüntünün olasılık dağılım fonksiyonunun doğrusallaştırıldığı histogram eşitlemede aşağıdaki adımlar uygulanır [29, 30]:

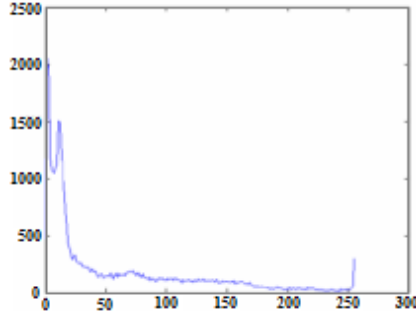
Adım 1: Orijinal görüntünün histogramı bulunur (Şekil 3.3b).
Adım 2: Histogramdan yararlanılarak kümülatif histogram bulunur. Kümülatif histogram, histogramın her değerinin kendisinden öncekiler ve kendisinin toplamı ile elde edilen değerleri içeren grafiklerdir.
Adım 3: Kümülatif histogram değerleri normalize edilip (toplam piksel sayısına bölünerek), yeni görüntüde olması istenilen maks. renk değerleri ile çarpılır, çıkan değer tam sayıya yuvarlatılır. Böylelikle gri seviye değerleri elde edilmiş olur.
Adım 4: Orijinal gri seviyeli değerleri ile üçüncü adımda elde edilen gri seviye değerleri birbirine karşılık düşürülür ve yeni histogram elde edilir (Şekil 3.3d) [18].



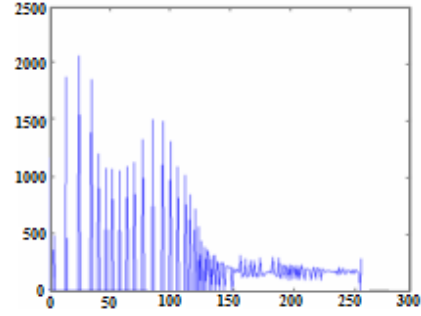
a. Normal (Orijinal) görüntü



b. Histogram eşitleme uygulanmış görüntü



c. Orijinal Görüntünün Histogramı



d. Histogram Eşitleme Uygulanmış Görüntünün Histogramı

Şekil 3.3. Görüntüye histogram eşitleme uygulanması

Kontrast geliştirme yönteminde ise görüntünün karşıtlık değerleri değiştirilir. İnsan gözünün gri değerlerini ayırt etme kapasitesi ortalama olarak otuz adet gri değeriyle sınırlıdır. Bir görüntüdeki en düşük ve en yüksek gri değerleri arasındaki farkın bu sınırdan düşük olması durumunda, bu görüntüdeki detayların insan gözü tarafından yeterince algılanamadığı söylenebilir. Bu tipteki görüntülerin histogramlarında gri düzeyli bir görüntünün dar bir aralığında çok sayıda pikselin yoğunlaştığı görülür. Bu görüntülere düşük kontrastlı görüntüler adı verilir. Dolayısıyla, kontrast kavramı bir görüntü içindeki grilik seviyesi farklarının genliğini ifade etmektedir. İnsan incelemesine uygun olmayan düşük kontrastlı görüntüler çeşitli dönüşüm fonksiyonları yoluyla insan için daha uygun duruma getirilebilirler. Farklı türdeki düşük kontrastlı görüntüler için farklı fonksiyonlar kullanmak gerekir. Koyu tonların ağırlıklı olduğu bir görüntüyü geliştirmek için gri değerlerini açık tonları gösteren yüksek değerlere doğru kaydırmak gerekir (Şekil 3.4) [31].



Şekil 3.4. Görüntünün kontrastının değiştirilmesi

Kontrast değiştirme algoritmalarında kullanılan formül şu şekildedir:

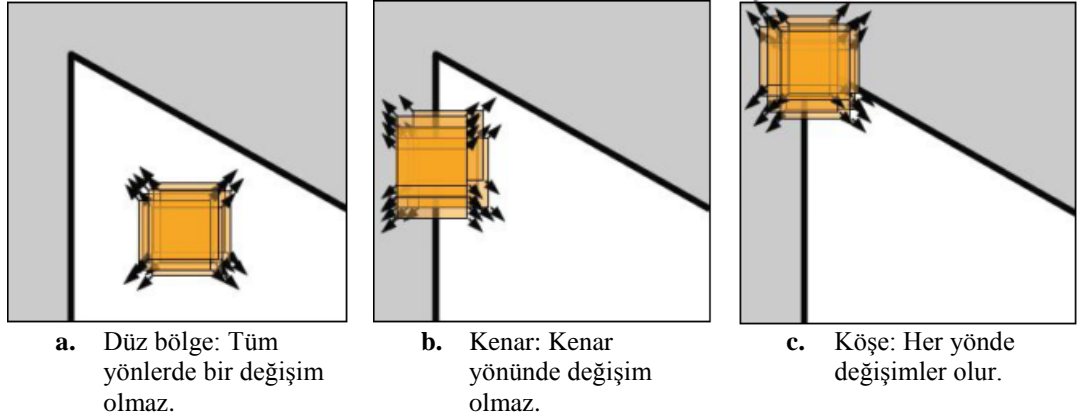
$$g(x, y) = T[f(x, y)] \quad (3.1)$$

$$= af(x, y) \quad s = ar \quad a > 1 \text{ ise karşıtlık artar.}$$

$$a < 1 \text{ ise karşıtlık azalır.}$$

3.1.10.3 Kenar ve köşe bulma

Kenar ve köşe bulma/belirleme, görüntü işleme ve görüntü tanıma sistemlerinde önemli bir araştırma alanıdır. Özellikle içerik tabanlı multimedya indeksleme ve geri getirmede, köşe bulma en güçlü tekniklerden biridir. Görüntü içinde köşe bulmak, bölütleme (segmentation), görüntü tanıma işlemleri gibi alanlarda kullanılmaktadır. Başarılı bir köşe belirleme aracı tüm gerçek köşeleri yüksek bir doğrulukla bulabilmelidir. Bununla birlikte köşe olmayan noktaları yok etmeli veya minimize etmelidir. Gürültüye hassas olmalıdır. Çözünürlük, yer değiştirme değişiminden ve ölçeklemeden etkilenmemelidir [32, 33]. Bir köşede, herhangi bir yönde hareket yoğunlukta büyük bir değişime sebep olur [22]. Şekil 3.5.'de Kenar ve köşe gösterimi bulunmaktadır[22].



Şekil 3.5. Kenar ve köşe gösterimi

Köşelerin önemli avantajlarından bir tanesi ölçekleme, döndürme gibi dönüşümlerden etkilenmemeleridir. Köşe açısı nesnenin döndürülmesi durumundaki değişimden etkilenmez. Köşenin bulunduğu yer ile nesnelerin pozisyonu belirtilebilir. Kenar belirleme ve nesne tanımda köşelerin bu özelliklerinden faydalanılabilir. Köşelerin aksine kenarlar yoğunluk değişimlerinden etkilenmezler ancak diğer görüntü dönüşümlerinden etkilenirler [34].

Köşe saptama araçları genel olarak geometri tabanlı ve şablon tabanlı olmak üzere ikiye ayrılabilir. Geometri tabanlı yaklaşım daha çok kullanılmaktadır. Geometrik özellikleri analiz ederek köşeleri belirleyen bu yaklaşımda görüntünün kenarları çıkarılır ve yerel maksimum eğim derecesine sahip noktalar köşe olarak bulunur. Bu tür yöntemlerin dezavantajı açı ve yerleşim gibi köşelerin özelliklerini kaydetmemeleridir. Şablon tabanlı yöntemler ise köşeleri belirlemek için görüntü şablonları kullanırlar. Bu işlemi de görüntü ve kayıtlı şablonları karşılaştırıp aradaki benzerliği ölçerek gerçekleştirirler. Bu yaklaşımın dezavantajı ise fazla miktardaki hesaplamalar nedeniyle işlem performansının düşük olabilmesidir [33, 35].

Köşe saptama işleminin nasıl yapılabileceğini görmek için Moravec'in geliştirdiği yöntem [36] incelendiğinde, bir görüntü üzerinde olası bir köşeyi içeren bir bölge ile bu bölgenin değişik yönlerde ötelenmiş hali arasındaki karelerin farkının toplamı (SSD: Sum of Squared Differences) hesaplanır [22].

$$E(x, y) = \sum_u \sum_v w(u, v) (I(u, v) - I(u - x, v - y))^2 \quad (3.2)$$

w, görüntü penceresi şeklinde ifade edilebilir: Değeri dörtgenin içi ise 1, aksi halde 0'dır. Eşik değerini aşan $\min\{E\}$ içerisindeki yerel maksimum değeri köşeyi ifade eder [22].

Harris ve Stephens [37], SSD denkleminin ikinci türevine dayanan bir hesaplama yaparak Moravec'in yöntemini geliştirmiştir. 2. seviye Taylor yaklaşımı kullanılarak (3.2) denklem aşağıdaki hale gelir [22]:

$$E(x, y) \approx \frac{1}{2}(x, y)A \begin{pmatrix} x \\ y \end{pmatrix} \quad (3.3)$$

A, kısmî türev terimlerini içerir, $I_x = \partial I / \partial x$, $I_y = \partial I / \partial y$.

$$A = \sum_u \sum_v w(u, v) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (3.4)$$

A'nın özdeğerleri λ_1 ve λ_2 aşağıdaki gibi hesaplanabilir (3.5):

$$\lambda_1 = \frac{1}{2}(a + c - \sqrt{(a - c)^2 + 4b^2}) \quad (3.5)$$

$$\lambda_2 = \frac{1}{2}(a + c + \sqrt{(a - c)^2 + 4b^2})$$

Kareköklerin hesaplanması karmaşık olacağı için, köşe karşılık fonksiyonu (corner response function) ile işlem yapılması daha pratik olacaktır [22]:

$$R = \lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2 \quad (3.6)$$

k, deneysel olarak bulunur.

R'nin değeri sonucun ne olacağını gösterir:

- Eğer R , pozitif büyük bir sayı ise, sonuç köşedir.
- Eğer R , negatif büyük bir sayı ise, sonuç kenardır.
- Eğer R 'nin mutlak değeri sıfıra yakın küçük bir sayı ise, sonuç düz bölgedir.

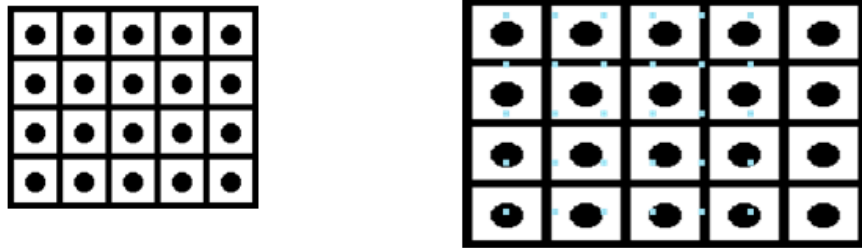
Bir eşik değeri uygulandıktan sonra, R 'nin yerel maksimumları köşeler olarak bulunmuş olur [22].

3.1.10.4 Geometrik dönüştürme

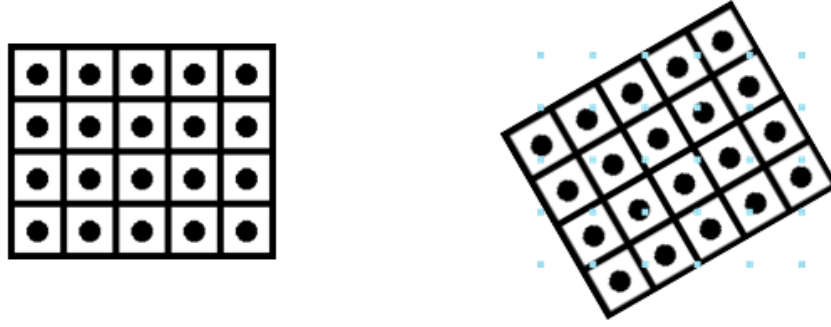
Geometrik dönüştürme; görüntünün küçültme (Şekil 3.6) ve büyütme (Şekil 3.7) işlemleriyle yeniden boyutlandırılması ve görüntünün döndürülmesi (Şekil 3.8) gibi yöntemlerdir [38].



Şekil 3.6. Görüntünün Boyutlarının En/Boy Oranı Korunarak Küçültülmesi



Şekil 3.7. Görüntünün Boyutlarının Bağımsız Olarak Büyütülmesi



Şekil 3.8. Görüntünün Tek Eksen Etrafında Döndürülmesi

3.1.10.5 Sayısal karıştırma veya optik karıştırma

Sayısal ve optik karıştırma iki veya daha fazla görüntünün birleştirilmesi yöntemidir. Bu yöntemde, görüntüleri birleştirmek için bir resimdeki seçilen pikseller diğerine kopyalanır [38]. Sayısal karıştırma işlemi Şekil 3.9’ da bir örnek ile gösterilmektedir.



a. Birinci Görüntü



b. İkinci Görüntü



c. İki Görüntünün Birleştirilmiş Hali

Şekil 3.9. İki Görüntünün Birleştirilmesi

3.1.10.6 İleri Görüntü İşleme

İleri görüntü işleme; görüntünün kalitesini yükseltme ve görüntü netleştirme (deblurring), sabit cisimleri tespit ederek iki boyutlu nesne tanıma, parazit oluşturma, istatistiksel bilgiler çıkarma, işlem ve filtre seçenekleri (Gaussian Filters), görüntü üzerinde birtakım bilgileri gizleme ve steganaliz, kayıpsız görüntü sıkıştırma, diğer matematiksel ve morfolojik işlemler ile görüntü anlama yöntemleri gibi ileri görüntü

işleme tekniklerini içerir. Görüntü arama için genetik algoritmalar ve görüntü eşleme için yapay sinir ağları ile çeşitli fonksiyonlar kullanılmaktadır [8].

3.2. Görüntü İşleme Algoritmaları

Görüntü işlemede farklı amaçlar için kullanılan yöntemler, aynı zamanda bu yöntemleri kullanarak oluşturulan algoritmalara da isim vermiştir. Fourier dönüşüm metodu, Histogram eşitleme, Medyan (Orta) filtreleri, Gabor süzgeç metodu, dalgacık dönüşümü metodu, Topolojik Bağımsız Bileşen Analizi (TBBA) vb. birçok algoritma bulunmaktadır [10]. Bu bölümde, görüntü işleme algoritmaları sekiz grupta altında incelenecektir. Bunlar:

- 1) Görüntü netleştirme algoritmaları
- 2) Gürültü giderme algoritmaları
- 3) Renk değiştirme ve düzeltme algoritmaları
- 4) Kenar belirleme algoritmaları
- 5) Köşe saptama algoritmaları
- 6) Plaka okuma ve karakterlerini tanıma algoritmaları
- 7) Yüz tanıma algoritmaları
- 8) Diğer algoritmalar

3.2.3. Kenar belirleme algoritmaları

Kenar bulma algoritmalarının başlıcaları; Gradyent Kenar Bulma Algoritmaları (Prewitt, Sobel ve Roberts Algoritmaları), LoG (Laplacian of Gaussian) Kenar Bulma Algoritması ve Canny Kenar Bulma Algoritması [6, 9, 11-13, 23, 39] olarak sıralanabilir. Bu algoritmalar aşağıda kısaca ele alınacaktır.

3.2.3.1 Gradyent kenar belirleme algoritmaları

Başlıca gradyent kenar belirleme algoritmaları; Prewitt Algoritması, Sobel Algoritması ve Roberts Algoritması'dır [39]. Bu algoritmaların performansını

karşılaştırmaya yönelik olarak yapılan çalışmalar; Sobel Kenar belirleme algoritmasının Roberts ve Prewitt algoritmalarına göre daha iyi sonuç verdiğini göstermektedir [12].

a) Prewitt Algoritması:

Prewitt Algoritması, Sobel Algoritması gibi düşey ve yatay keskinlik yakalar. Kernel matrisi Sobel'den farklıdır. Matrisin dizimi şöyledir: [1,1,1;0,0,0;-1,-1,-1] veya [1,0,-1;1,0,-1;1,0,-1] [40]. Prewitt Algoritması Sobel'e göre daha basit hesaplanır fakat sonuçta birtakım lekeler/gürültüler üretmektedir [6, 13].

b) Sobel Algoritması

Sobel Algoritması, 1. kısmi türeve dayanmaktadır [13]. Sobel operatörü yatay ve düşey yönde keskinlikleri yakalar. Kernel matrisi dizilimi şöyledir: [1,2,1;0,0,0;-1,-1,-1] veya [1,0,-1;2,0,-2;-1,-2,-1] [40].

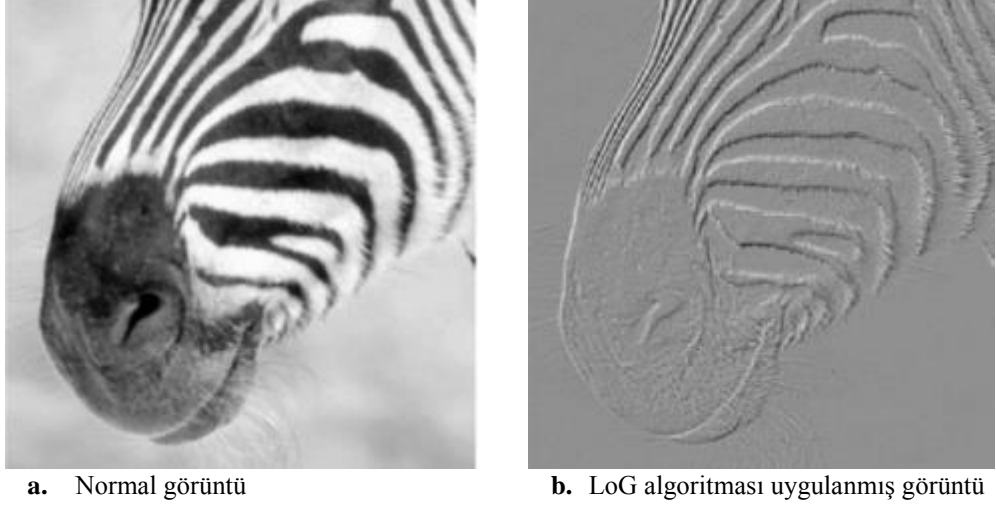
c) Roberts Algoritması

Bu filtre diagonal olarak kenar tarar. Kernel matrisi şöyledir: [1,1,0;1,0,-1;0,-1,-1] veya [2,1,0;1,0,-1;0,-1,-2] [40]. Roberts Algoritması, en eski ve basit kenar belirleme algoritmasıdır. Simetrik olmadığı ve 45°'lik kenarlarda sınırlamalar getirir. Buna karşılık Sobel'e kıyasla daha basit hesaplandığından, baskın faktörlerin alınmasında hızlı sonuç sağlamaktadır [6]. Bir diğer deyişle, sadece yatay ve düşey kenarları algılayabildiği için hızlı ve basittir. Bu nedenle gerçek zamanlı uygulamalarda kullanılabilir [13].

3.2.3.2 LoG (Laplacian of Gaussian) algoritması

“Marr-Hildreth algoritması” da denilen LoG (Laplacian of Gaussian) Algoritması [40], görüntüdeki keskin geçişleri bulmak için kullanılan türev bazlı algoritmalar [39]. Türev bazlı algoritmalar gürültüye çok duyarlı oldukları için, görüntüye laplasyeni

uygulamadan önce Gaussian algoritması gibi bir smoothing fitresinden geçirek gürültüleri azaltmak daha iyi sonuç vermektedir. Bu iki aşama sonucu elde edilen filtreleme işlemine LoG filtreleme metodu adı verilmektedir [39]. LoG, görüntüdeki kenarları ve oval bölgeleri bulan bir filtredir. Şekil 3.10' daki gibi bir çıktı üretir [41].



Şekil 3.10. Görüntüye LoG algoritması uygulanması

3.2.3.3 Canny kenar belirleme algoritması

Kenar bulmada son derece etkin bir algoritmadır. Önce görüntüdeki gürültü bir sigma değerine göre üretilen Gaussian çekirdekle konvolüsyonu alınarak azaltılır. Ardından, gradyent operatörünün uygulanmasıyla, kenar gradyentin büyüklük ve yönü hesaplanır. Kenarlar, non maxima baskılama uygulanarak inceltilir. Son olarak görüntü, ikili eşikleme uygulanarak istenmeyen ayrıntılardan arındırılır [40].

Canny kenar belirleme algoritması dört aşamada uygulanmaktadır [39].

- 1.Adım: Görüntüye bir Gaussian filtresi yardımıyla yumuşatma işlemi yapılır.
- 2.Adım: Kısmî türevler için sonlu-fark yaklaşımları kullanılarak gradyentin büyüklüğü ve yönü hesaplanır.
- 3.Adım: Görüntünün büyüklük değerinde maksimum olmayan noktaların bastırılması işlemi uygulanır.

- 4.Adım: Kenar pikselleri bulmak için çift eşikleme algoritması kullanılır.

Bu aşamalar aşağıda açıklanmıştır.

1.Adım: Yumuşatma

Görüntüye bir Gaussian filtresi yardımıyla yumuşatma işlemi uygulanır. $I [i, j]$ orijinal görüntü, $G [i, j, \sigma]$ Gaussian yumuşatıcı filtre ve σ Gaussian filtresinin standart sapması (yumuşatma derecesi) olmak üzere, orijinal $I [i, j]$ görüntüsü ve $G [i, j, \sigma]$ filtresinin konvolüsyonu sonucu elde edilen yumuşatılmış görüntü $S [i, j]$ ile ifade edilmektedir [12].

$$S [i, j] = G [i, j, \sigma] * I [i, j] \quad (3.7)$$



c. Normal görüntü



d. Gaussian filtresi uygulanarak yumuşatılmış görüntü

Şekil 3.11. Normal görüntünün Gaussian filtresi kullanılarak yumuşatılması.

2.Adım: Gradyanın Hesaplanması

Kısmî türevler için sonlu-fark yaklaşımlarının kullanılması yoluyla gradyanın büyüklük ve yönü hesaplanır. İlk olarak, $S [i, j]$ 'nin kısmî türevleri elde edilir [12]. Bunlar;

$$P [i, j] \approx (S [i, j+1] - S [i, j] + S [i+1, j+1] - S [i+1, j])/2 \quad (3.8)$$

$$Q[i, j] \approx (S[i, j] - S[i+1, j] + S[i, j+1] - S[i+1, j+1])/2 \quad (3.9)$$

şeklinde yazılır. x ve y kısmî türevleri 2×2 'lik kare matris üzerindeki sonlu farkların ortalaması alınarak hesaplanır. Buna göre gradyanın büyüklük değeri:

$$M(i, j) = \sqrt{P[i, j]^2 + Q[i, j]^2} \quad (3.10)$$

ve açısı:

$$Q[i, j] = \arctan(Q[i, j], P[i, j]) \quad (3.11)$$

olur [23].

3.Adım: Maksimum Olmayan Noktaların Bastırılması

Gradyan algoritmasında, gradyan alınarak kenar pikselleri belirlemek mümkündür. Canny algoritmasına göre kenar noktası, şiddeti gradyan vektörü yönünde lokal olarak maksimum olan bir nokta olarak düşünülebilir [23]. Bu durum oldukça kısıtlayıcı bir durumdur ve eşikleme yöntemiyle bulunan kenar piksellerinden oluşan çizgileri inceltmek için kullanılan bu işleme “Maksimum Olmayan Noktaların Bastırılması” adı verilir. Bu işlemden sonra elde edilen $N[i, j]$ görüntüsü aşağıdaki gibidir [12]:

$$N[i, j] = nms(M[i, j], \zeta[i, j]) \quad (3.12)$$

Yerel maksimum noktaları olarak kabul edilen noktalarda bu değer sıfırdır[23].

4.Adım: Eşikleme

Kenar pikselleri bulmak için çift eşikleme algoritması kullanılır. Canny kenar belirleme algoritması işaret gürültü oranını optimize edecek şekilde tasarlanmıştır. İlk adımda yumuşatma işlemi gerçekleştirilmesine rağmen, maksimum olmayan noktaları

bastırılmış olan $N [i, j]$ görüntüsünde, gürültüler nedeniyle bazı hatalı kenar noktalarının oluşması mümkündür. Bu türden hatalı kenar noktaların etkisi çok fazla değildir. Maksimum olmayan noktaları bastırılmış bu görüntüdeki hatalı kenar noktaları azaltmak için $N [i, j]$ 'ye bir eşik değeri uygulayarak eşik değerinin altındaki tüm noktaları sıfır yapılır [12].

Maksimum olmayan noktaları bastırılmış görüntüye eşikleme işlemi uyguladıktan sonra elde edilen görüntü $E [i, j]$ kenarları belirginleşmiş bir görüntüden ibarettir. Ancak, bu metotta uygun eşik değerini bulmak önemlidir. Uygun eşik değeri genellikle deneme yanılma yoluyla bulunmaktadır. Eğer eşik değeri çok küçük seçilirse, $E [i, j]$ görüntüsünde istenmeyen yanlış kenarlar oluşur, eşik değeri çok büyük seçildiğindeyse bazı kenarlar kaybolabilir. Bu nedenle, daha etkili bir eşikleme için çift-eşikleme yöntemi uygulanarak eşiklemede iki eşik değeri kullanılır. Belirginleşmiş görüntüyü elde etmek için $N [i, j]$ 'ye uygulanan çift eşiklemede eşik değerleri T_1 ve T_2 olarak alınır. Bu işlem sonucunda; $T_1 [i, j]$ ve $T_2 [i, j]$ eşiklenmiş görüntüleri elde edilir. T_2 görüntüsünde, kenar piksellerden oluşan hat üzerinde açıklıklar vardır. Fakat dışarıda kalan yanlış kenarların sayısı o kadar fazla değildir. Bu nedenle T_1 görüntüsü kullanılarak, T_2 kenar hattındaki açıklıklar kapatılarak optimum düzeltme sağlanmış olmaktadır [23].

Kenar bulma, resim üzerindeki piksel değerlerinin gruplandırılmasına dayanan bir yöntemdir. Bu yöntemde, öncelikle RGB değerleri birbirine yakın olan pikseller gruplanır ve bu piksellere tek bir değer atanır. Böylece resimde piksel öbekleri oluşturulur. Sonrasında bu piksel öbekleri arasındaki sert geçişler kenar olarak tanımlanır. Canny Edge Detection algoritması resim üzerindeki nesnelerin tanımlanması için kullanılır. Birbirini tamamlayan kenarlar bir nesneyi tanımlayabilir [12]. Şekil 3.12'de Kenar belirlemede piksel değerlerinin ortalamasının alınması gösterilmektedir[23].

5	7	6	4	152	148	149

Şekil 3.12. Kenar belirlemede piksel değerlerinin ortalamasının alınması

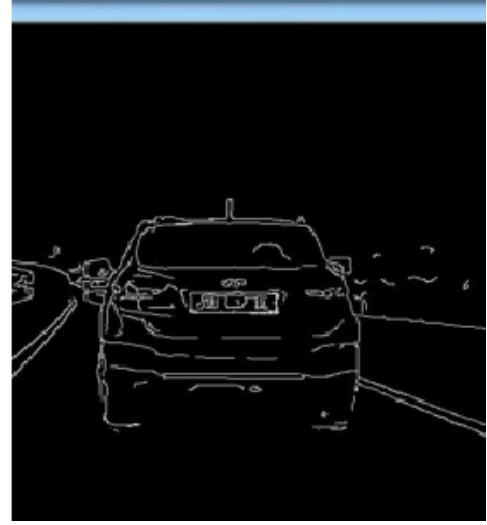


Şekil 3.13. Kenar belirleme

Şekil 3.13’de görüldüğü üzere 4-5-7-6 (ort 5,5) piksel değerlerini bir grup, 152-148-149 (ort 149) piksel değerlerini diğer bir grup olarak tanımlandığında bu iki grubun arasındaki ciddi ortalama farkı bir geçiş olduğu, buna bağlı olarak iki öbek arasında bir kenar bulunduğu sonucuna varılır[23]. Şekil 3.14’de verilen kenar belirleme uygulanmış görüntüdeki geçişler net bir şekilde görülmektedir. Şekilde görüldüğü gibi, görüntüye kenar belirleme işlemi uygulandığında görüntüdeki çocuğu çevreleyen kenarlar arka plan görüntüsünün herhangi bir etkisi olmaksızın net biçimde elde edilebilmektedir. Kenar belirleme işleminden elde edilen kenarların doğruluğu ilk görüntüdeki aydınlatmanın yeterli olup olmadığına önemli ölçüde bağlı olduğu bir gerçektir [23].



a. Normal görüntü



b. Kenar belirleme filtresi uygulanmış görüntü

Şekil 3.14. Normal görüntünün kenar belirleme filtresi kullanılarak kenarlarının belirlenmesi

3.2.4. Köşe saptama algoritmaları

Köşe saptama algoritmalarının başlıcaları Harris and Stephens Köşe Saptama Algılama algoritması, SIFT (Ölçekten Bağımsız Öznitelik Dönüşümü) köşe algılama algoritması, makine öğrenme merkezli FAST (Hızlandırılmış Segment Testi Özelliği) algoritması, SURF (Hızlandırılmış Gürbüz Öznitelikler) algoritması ve Hough algoritmasıdır [23, 42, 43]. Bu bölümde Hough Algoritması ve Harris-Stephens Köşe Saptama Algoritması ele alınacaktır.

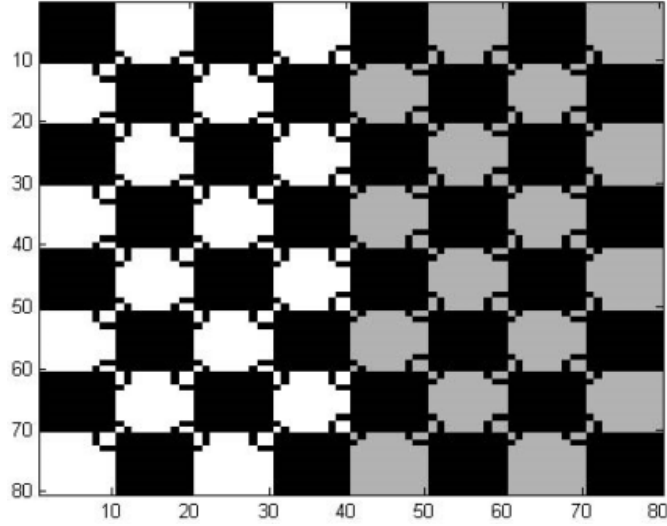
3.2.4.1 Hough algoritması

Hough dönüşümü, tespit edilen kenarların olası geometrik şekillere uygunluğunun oylaması mantığı ile çalışmaktadır. Görüntüdeki dikdörtgen biçimli kenarların Hough dönüşümü kullanılarak tespiti genel olarak aşağıdaki adımlar ile özetlenebilir [23]:

- Kaynak görüntü üzerinde kenarlar belirlenir.
- Bir eşikleme yöntemi kullanılarak görüntü ikili (siyah-beyaz) hale getirilir.

- Her kenar pikseli için noktanın üzerinde olabileceği olası geometrik şekillerin polar koordinattaki değerleri kullanılan bir akümülatör matrisi üzerinde her kenar piksel için olası şekillerin oylaması yapılır.
- Akümülatör değeri en yüksek olan şekiller en çok oy alan şekiller olduklarından görüntü üzerinde bulunma ya da belirgin olma olasılıkları yüksektir.

Bulunan şekiller isteğe bağlı olarak görüntü üzerine yazdırılabilir:



Şekil 3.15. Koordinatların Köşeleri Oluşturması

Şekil 3.15'deki resimde daire içine alınmış koordinatlar köşeleri göstermektedir. Bu resimde (10,70) ile (10,10) noktaları arasındaki çizgi “dikey çizgi”, (10,70) ile (70,70) arasındaki çizgi “yatay çizgi”, (10,10) ise köşe noktası olarak tanımlanabilir. Görüntüdeki aday dikdörtgenlere ait yatay ve dikey çizgiler köşe noktalarında birleşmektedir [23].

3.2.4.2 Harris and Stephens köşe belirleme algoritması

Canny Kenar Bulma algoritması uygulandıktan sonra görüntüdeki yatay ve düşey çizgilerin oluşturduğu aday dikdörtgenleri bulmak için bir köşe belirleme yöntemi kullanılması gerekmektedir. Harris and Stephens Köşe Saptama algoritması köşe belirlemede en çok kullanılan algoritmalarından biridir [23].



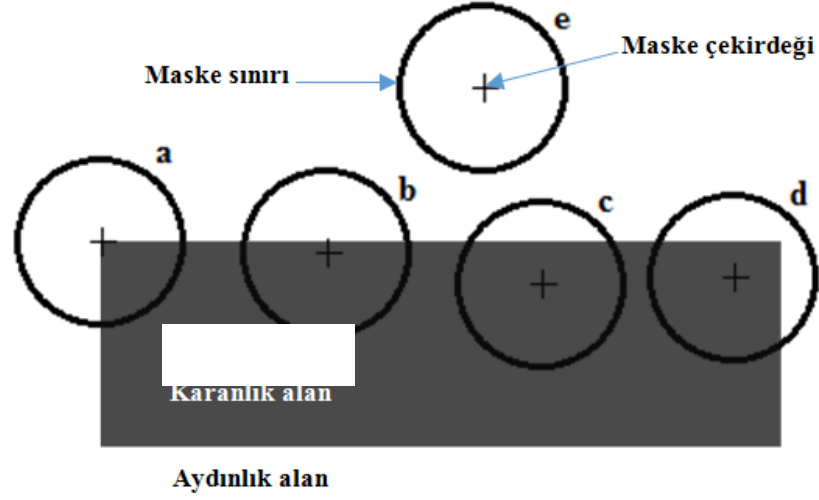
Şekil 3.16. Harris and Stephens Köşe Saptama Algoritması ile bir aracın köşelerinin bulunması

Harris ve Stephens Köşe Saptama algoritması ile tespit edilen köşe noktaları kullanılarak belirlenen dikdörtgen formuna yakın kapalı şekiller, aday araç ve plaka görüntüleri olarak kabul edilmektedir. Yatay ve düşey çizgiler belirlenirken yatayla $180^{\circ} \pm 15^{\circ}$ ve $90^{\circ} \pm 15^{\circ}$ açı yapan çizgiler seçilmektedir. Seçilen yatay ve düşey çizgilerin dışındaki çizgiler göz ardı edilmektedir. Aday görüntünün en üst tarafındaki ve en alt tarafındaki çizgiler aracın sınır çizgileri olarak kabul edilmektedir [23].

3.2.4.3 Susan köşe belirleme algoritması

SUSAN köşe belirleme algoritması, 1997 yılında Smith ve Brady [45] tarafından parlaklık karşılaştırmasına dayalı olarak önerilen bir köşe belirleme algoritmasıdır [44]. Bu algoritma, merkez piksele yakın yoğunlukta olan bir komşuluk içindeki piksellerin fraksiyonunu hesaplar [45]. Türev hesaplamayı önler, ancak görüntü gürültüye daha duyarlı olduğunda verimliliği düşer [42].

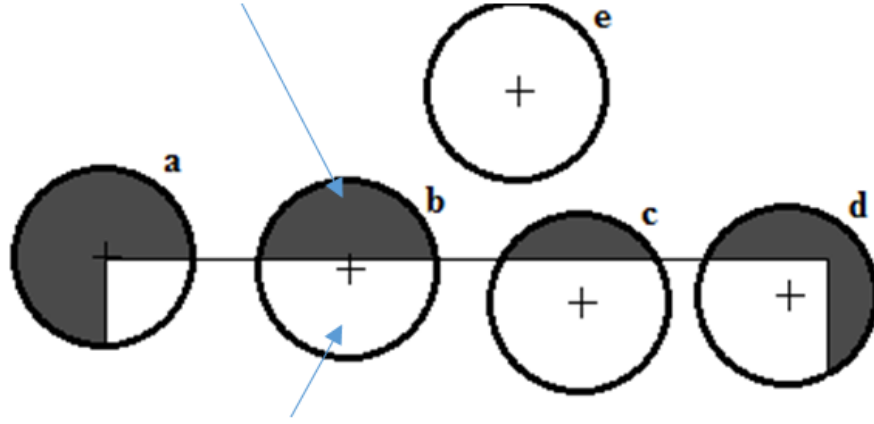
Bu algoritma şöyle açıklanabilir: Beyaz arka zeminde koyu bir dikdörtgen görülen, Şekil 3.17'yi ele alalım. Bu şekilde, ‘çekirdek’ olarak bilinen bir merkez piksele sahip olan dairesel bir maske, beş görüntü konumunda gösterilmektedir.



Şekil 3.17. Beyaz arka zeminde verilen koyu renkli bir dikdörtgenden oluşan basit bir şekil üzerinde farklı alanlarda bulunan dört dairesel maske

Bir maske içindeki her pikselin parlaklığı, bu maske çekirdeğinin parlaklığıyla karşılaştırıldığında, maskenin çekirdek ile aynı (veya benzer) parlaklığı olan bir alanı olarak tanımlanabilir. Maskenin bu alanı, “Univalve Segment Assimile Nucleus” için bir kısaltma olan ‘USAN’ olarak bilinir [45]. USAN’ın Türkçe karşılığı basit olarak “lokal alan” şeklinde tanımlanabilir. Şekil 3.18’de, Şekil 3.17’deki her maske, beyaz renkteki USAN’ı (lokal alanı) ile gösterilmiştir.

pikselleri çekirdekten farklı parlaklığa sahip olan maske kesiti

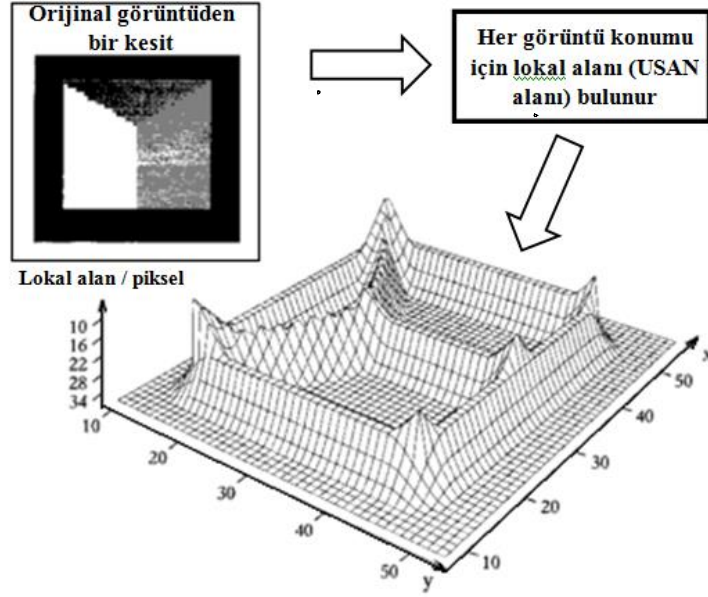


pikselleri çekirdekle aynı parlaklığa sahip olan maske kesiti

Şekil 3.18. Lokal alanları (USAN'ları) maskelerin beyaz kısımları olarak gösterilmiş olan benzer renkteki dört dairesel maske

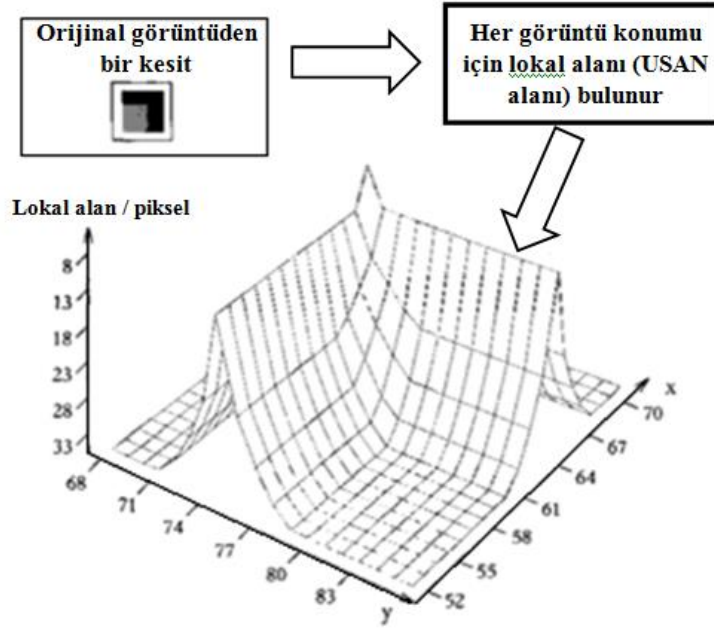
Her görüntü noktasının, benzer parlaklıkta bir yerel alan ile ilişkilendirmiş bu konsepti, SUSAN ilkesinin temelini oluşturmaktadır. Lokal alan veya USAN, resmin yapısı hakkında birçok bilgi barındırır. Bu algoritmanın küçük bir ölçekte etkin bir şekilde bölge bulucu özelliği vardır. Lokal alanın (USAN'ın) boyutundan, merkezi ve ikinci momentlerinden, iki boyutlu özellikler ve kenarlar tespit edilebilir. Özellik algılamaya yönelik bu yaklaşım, iyi bilinen yöntemlere göre birçok farklılığa sahiptir. En belirgin farklılığı, herhangi bir görüntü türevi kullanılması ya da gürültü azaltma işlemi gerektirmemesidir.

Bir USAN'ın (lokal alanın) alanı, söz konusu herhangi bir nokta etrafındaki bölgede bulunan görüntünün yapısına ilişkin en önemli bilgiyi verir. Şekil 1 ve 2'den görülebileceği gibi; USAN alanı, çekirdek görüntü yüzeyinin düz bir bölgesinde bulunduğu zaman maksimumdur; düz bir kenarın çok yakınında değeri yarıya düşer; ve bir köşenin içinde olduğunda ise bu değer daha da düşer. Kenarların ve iki boyutlu şekillerin temel belirleyicisi olarak kullanılan, USAN alanının bu özelliğidir. Şimdi de Şekil 3.19'da, bir test görüntüsünün küçük bir parçasının, çıktı olarak USAN alanını (kolak alanı) vermek üzere işlendiğini düşünelim.



Şekil 3.19. Kenar ve köşe iyileşmesini gösteren, test görüntüsünün küçük bir kısmı verilen USAN alanının üç boyutlu bir grafiği

Giriş görüntüsündeki her nokta, küçük bir dairesel maskenin çekirdeği olarak kullanılır ve ilişkili USAN bulunur. USAN'ın alanı, gösterilen üç boyutlu alan içerisinde kullanılır. USAN alanı, kenara yaklaştığında düşer (tam kenar noktasında minimuma ulaşır) ve köşelere yakın daha da düşer. Böylece görüntünün tam olarak köşelerinde minimum USAN alanını verir. Şekil 3.20, gürültülü gerçek bir görüntünün küçük bir kesitini ve USAN alanı işlemesiyle elde edilen çıktıyı göstermektedir.

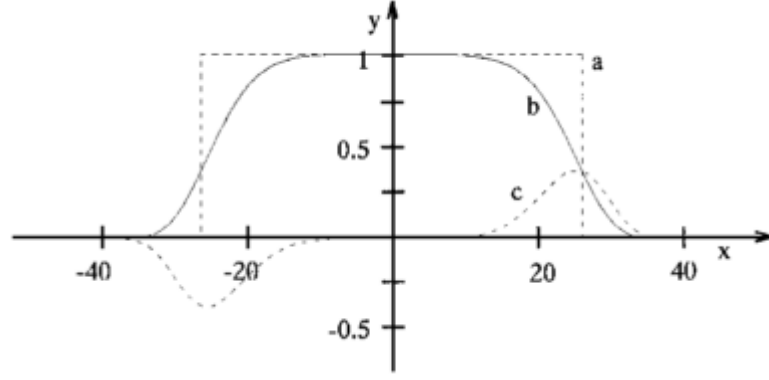


Şekil 3.20. Kenar ve köşe iyileşmesini gösteren, gürültülü gerçek bir görüntünün küçük bir kısmı verilen USAN alanının üç boyutlu bir grafiği

"Düz bölgeler" içerisindeki parlaklık değişimi, 256-gri tonlama seviyesinden 15-sıradır. Böylece kenar ve köşe iyileştirme yapılmış olmaktadır; son alan üzerinde ise hiçbir gürültü görünmemektedir. Bu argümanların göz önüne alınması ve Şekil 3.17-20'de gösterilen örneklerin ve sonuçların gözlemlenmesi, doğrudan SUSAN algoritmasındaki temel ilkenin formülünü vermektedir. Çıktı olarak ters çevrilmiş USAN alanını vermek üzere işlenmiş bir görüntü kenarları ve iki boyutlu özellikleri güçlü bir şekilde geliştirmiş olup, iki boyutlu özellikler kenarlardan daha güçlü bir şekilde geliştirilmiştir. Bu, SUSAN kısaltmasıyla ifade edilen "Smallest Univalued Segment Assimilating Nucleus"u verir.

SUSAN kenar ve köşe iyileştirme algoritmasının görüntü türevlerini kullanmaması, gürültülü görüntülerde performansının neden iyi olduğunu açıklamaktadır. Algoritma temel prensibinin doğrusal olmayan tepkisi, bütünleştirici etkisiyle birlikte güçlü gürültüyü reddeden bir özelliğe sahiptir. Bu, aynı şekilde bağımsız olarak dağıtılan Gauss gürültüsüne sahip bir giriş sinyali düşünüldüğünde kolaylıkla anlaşılabilir.

Gürültü, her 'benzer' değeri içermeye yeterince küçük olduğu sürece (bkz. Şekil 3.21) dikkate alınmaz.



a) Özgün benzerlik fonksiyonu (y ekseninde, birim yok) ile piksel parlaklık farkı (x ekseninde, gri seviyelerde). Bu örnek için piksel parlaklık farkı 'eşiği', ± 27 gri seviyede ayarlanmıştır. b) Daha dengeli bir işlev kullanılmaktadır. c) B-Sınır detektörü.

Şekil 3.21. Gürültünün benzer değerleri içermeye yeterince küçük olması durumunda USAN fonksiyonunda dikkate alınmaması

Alan hesaplamasında bireysel değerlerin integralinin alınması gürültünün etkisini daha da azaltır. SUSAN kenar ve köşe dedektörünün bir diğer üstün yönü de kontrol parametrelerinin diğer kenar algılama algoritmalarının çoğundan çok daha basit ve daha az rassal olması, dolayısıyla otomatikleştirilmesinin daha kolay olmasıdır. SUSAN gürültü azaltma algoritması, USAN'ın en iyi yerel komşuluk yumuşatmasını seçmek için kullanıldığı SUSAN prensibiyle ilgilidir [45].

Dolayısıyla, Smith ve Brady [45]; SUSAN köşe dedektörünün, görüntü türevlerini hesaplamadığı ve dolayısıyla gürültüyü yükseltmediği için gürültülü görüntülerde iyi performans gösterdiğini iddia etmektedir. Rosten ve Drummond [32] da; gürültü sonuçlarının, performansın gürültünün arttırılmaya başlamasıyla birlikte performansın oldukça hızlı bir şekilde düştüğünü gösterdiğini ancak gürültünün DoG dedektörü hariç diğer dedektörlerdekinden daha iyi seviyelere inip daha iyi performans gösterdiğini tespit etmiş ve Smith ve Brady'nin [45] iddiasını desteklemiştir. Shen ve Wang'ın [44] çalışmasında 45 ms'lik en iyi hız performansına sahip olduğu saptanan SUSAN algoritmasının köşe belirlemede oldukça iyi performans gösterdiği ancak sentetik görüntünün altındaki karenin ortasında yer alan X-Bağlantısını kaçırdığı;

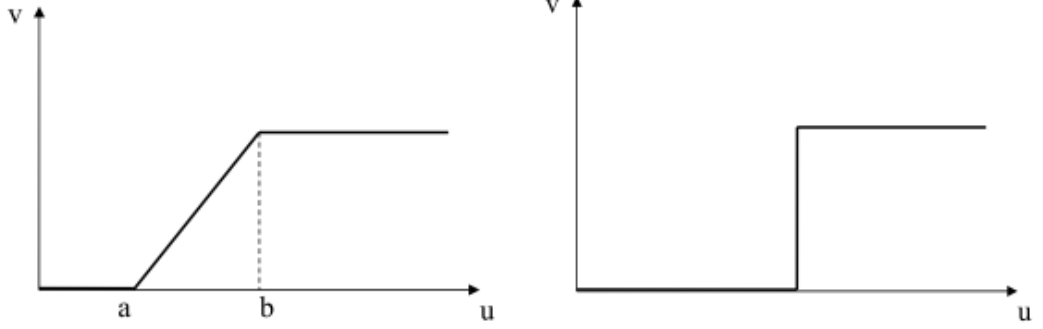
dolayısıyla SUSAN'ın bazı X-Bağlantılarında iyi çalışmadığı gösterilmiştir. Bu çalışmaya göre SUSAN, gürültüye karşı daha az duyarlı olmasına rağmen, düzgünce yumuşatılmış görüntülerde pek çok yumuşak köşeyi kaçırdığı için iyi sonuç verememiştir. Chen ve arkadaşlarının [46] çalışmasında ise; Harris köşe algılama algoritmasının genel olarak SUSAN köşe algılama algoritmasına üstün olduğu sonucuna varılmıştır.

3.2.5. Gürültü giderme algoritmaları

Gürültü gidermede temel olarak FIR, IIR, Adaptif ve Doğrusal Olmayan Algoritmalar kullanılmaktadır. FIR Algoritmalar (Sonlu Darbe Yanıtlı) non-rekürsif (geribeslemesiz) algoritmalar iken IIR Algoritmalar (Sonsuz Darbe Yanıtlı) Rekürsif (Geribeslemeli) algoritmalarıdır. Adaptif Algoritmalar; istenen bir işarete, öğrenme algoritmaları yardımıyla kendini adapte edebilirler. Doğrusal Olmayan Algoritmalar ise median filtre gibi doğrusal olmayan işlemleri gerçekleyebilirler [47]. Gürültü gidermede kullanılan algoritmalar kırpma ve eşikleme (clipping and thresholding) algoritmaları olarak da bilinmektedir [24].

Kırpma, karşılık esnetmenin özel bir durumudur. Giriş sinyalindeki renk değerleri sadece a ve b arasında olduğu durumlarda, gürültü temizlemek için kullanılır. Bir başka deyişle, aralıktaki değerlerin dışındaki renk değerleri önemli değil ise, kırpma kullanılabilir. İşlem sırasında belirli bir f_{min} ve f_{max} aralığındaki renk değerleri tüm skalaya yayılır. Örneğin 8 bit gri skalalı bir resimde; $f_{min}=80$ ve $f_{max}=140$ ise, buna kırpma uygulandığında, 80 ve altındaki renk değerleri tam siyaha çekilir, 140 ve üzerindeki değerler de tam beyaza çekilir. Böylece 80-140 arasındaki 60 değer, 256 seviyelik skalaya yayılır.

Kırpma işleminin bir değişik durumu da “eşikleme” olarak adlandırılır. Eşikleme yapıldığında imge iki bitlik gri skalaya dönüşür yani sadece “siyah” ve “beyaz” kalır. Başka bir deyişle, belirlenen eşik değerinin üstündeki değerlikli tüm pikseller beyaz yapılır; eşikğin altında kalanlar da siyah yapılır. Üçüncü bir alternatif olamaz. Şu şekilde de ifade edilebilir [24]: $f_{min}=f_{max}$ ="eşik değeri".



Şekil 3.22. Kırpma ve eşikleme dönüşüm fonksiyonları



a. Normal görüntü



b. Kırılmış görüntü

Şekil 3.23. Normal görüntünün kırılması



e. Normal görüntü



f. Eşikleme yapılmış görüntü

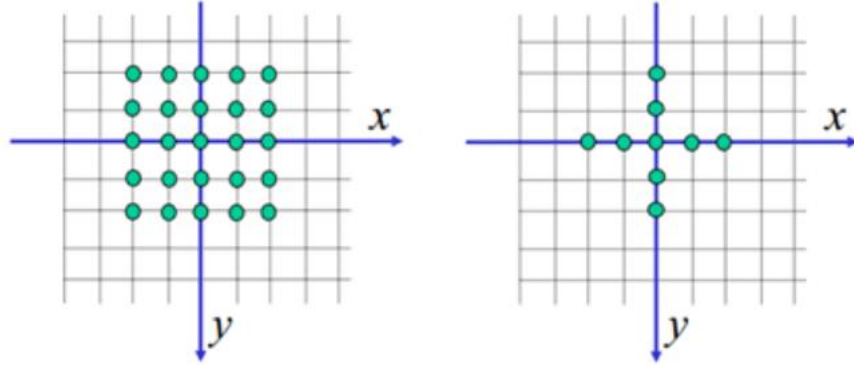
Şekil 3.24. Normal görüntüye eşikleme yapılması

Gürültü giderme algoritmalarına başlıca örnek olarak Gaussian algoritması [23] ve Farksal Gelişim Algoritması [48] verilebilir.

Kenar belirleme, gürültü yok etme, görüntü geliştirme gibi görüntü işleme uygulamalarında uzunlukları resim ve maske boyutları ile değişen ve genelde büyük boyutlu döngülerin kullanıldığı ve yoğun işlem gücü gerektiren konvolüsyon algoritmalarına başvurulmaktadır. Paralel hesaplama yaklaşımı böyle algoritmaların hızlandırılması için önemli bir potansiyel sunmaktadır [49].

3.2.6. Morfolojik İşlemler

Matematiksel morfoloji, görüntü işleme ve bilgisayar görüşünde kullanılan iyi bilinen tekniklerden biridir [50]. Bu kuramsal, şekil odaklı yaklaşım görüntüyü bir dizi ve genel olarak Yapısal Öge (SE:Structuring Element) olarak bilinen işlemin çekirdeği olarak başka bir set olarak ele alır. Genişletme, aşındırma, açma ve kapama gibi temel morfolojik işlemler, geometrik yapıların küme teorisine dayalı olarak analizinde ve işlenmesinde kullanılır. Açma ve kapama gibi geleneksel filtreler genellikle gürültüyü azaltmak için kullanılır, ancak ilgi alanı (ROI: Region Of Interest) segmentasyonunda ciddi sorunlar oluşturan bulanıklaştırma etkileri nedeniyle yaygın görüntü işleme uygulamalarında kullanımı yoktur. Morfolojik yeniden yapılandırmaya dayalı açma ve kapama filtreleri, konvansiyonel morfolojik açma ve kapama operatörlerinden ziyade şekil koruma açısından önerilmektedir [51]. Genişletme işlemiyle yapılan yeniden şekillendirme, ikili ve gri tonlamalı görüntülerde kendi jeodezi genişletme operatörleri tarafından tanımlanan morfolojik bir prosedürdür [52]. Yapısal Öge, görüntünün işlenmesinde ana görüntü üzerinde dolaştırılarak kullanılmakta olan matrisin morfolojiye özel hâle getirilmiş şeklidir (Şekil 3.25).

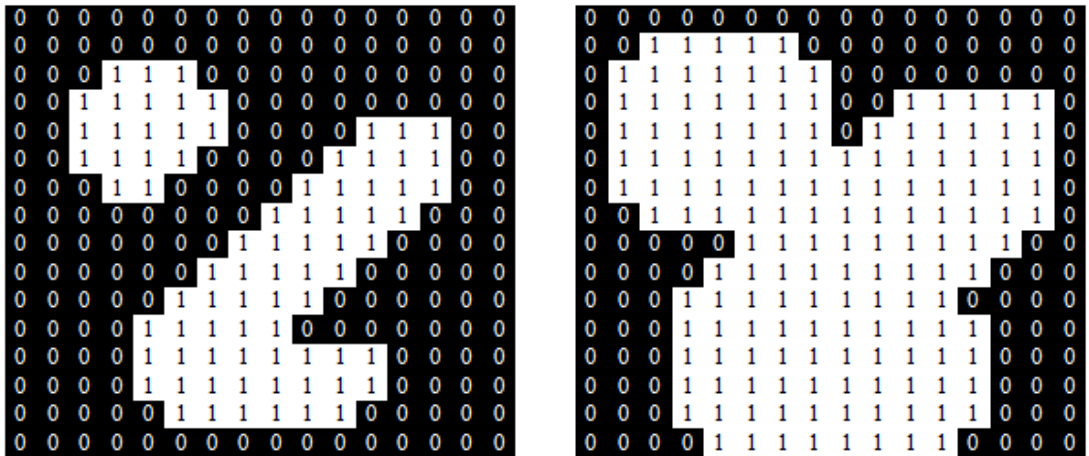


Şekil 3.25. Morfolojik yapısal öge örnekleri

Görüntü üzerinde dolaştırılan yapısal öge, uygulanmakta morfolojik işleme göre, görüntüde yer alan piksel değerleriyle yapısal ögenin değerlerini karşılaştırır. Bu sayede sonuçta elde edilecek görüntüde kullanılacak değer bulunmuş olur. Farklı şekil ve büyüklükte olabilen yapısal öğeler bir merkez noktasına sahiptir. İşlem sırasında orijinal görüntüdeki her piksel bu merkez noktaya oturtulur [53].

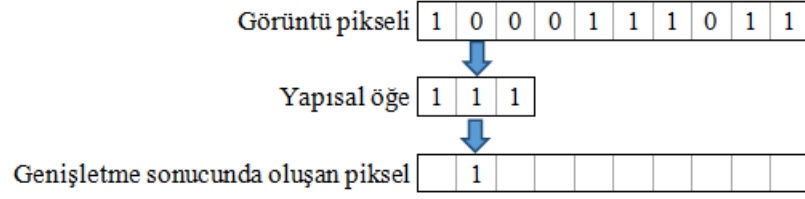
3.2.6.1 Genişletme ve aşındırma işlemleri

Genişletme işleminde, yapısal ögenin görüntü üzerinde dolaşması sırasında, yapısal ögenin merkeziyle objenin çakışması durumunda yapısal öge kadar genişletme gerçekleştirilmiş olur (Şekil 3.26).



Şekil 3.26. Genişletilmiş ikili görüntü

Bu işlem sonucunda her pikselin yapısal öge kadar büyütülmesi ve arka planın küçültülmesi sağlanır. Yapısal öğedeki orijin görüntü üzerinde ‘0’ değerine sahip bir pikselle karşılaştığında görüntüde bir değişiklik yapmaz; ancak ‘1’ değerine sahip bir pikselle karşılaşması halinde yapısal öge ile yapısal ögenin altındaki piksellere mantıksal ‘veya’ işlemi uygulanır (Şekil 3.27) [54].

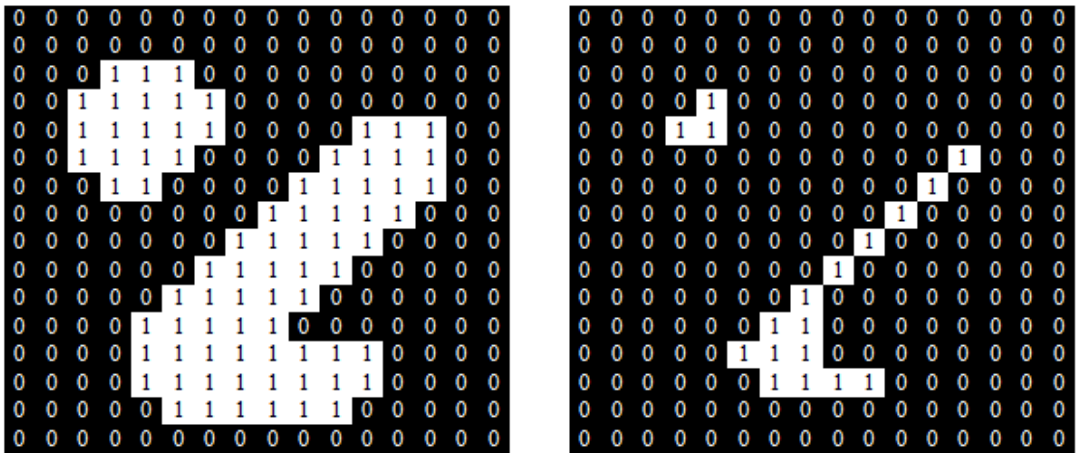


Şekil 3.27. Genişletme işleminin temel mantığı

A görüntüsünde B yapısal ögesiyle yapılan genişletme işlemi (δ) Formül 3.13’deki gibi gösterilir [18]:

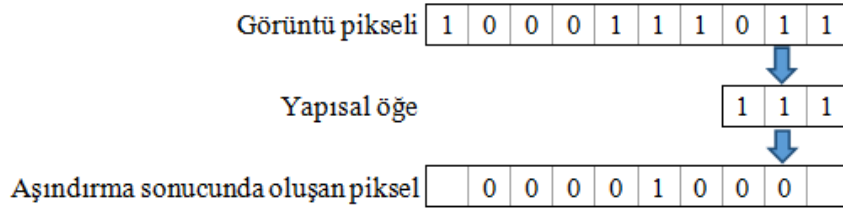
$$\delta(A) = \bigcap_{b \in B} A + b = A \oplus B \quad (3.13)$$

Aşındırma işlemi, ikili (binary) bir görüntüde bulunan nesnenin küçültülmesi veya inceltmesi için kullanılan morfolojik bir işlemdir. Aşındırma işlemiyle, yapısal ögenin orijini haricindeki yerlere arka plan değeri verilerek nesne üzerinde bir aşındırma işlemi gerçekleştirilir. Bu işlem 3.28 ‘de gösterilmektedir.



Şekil 3.28. Aşındırılmış ikili görüntü

Bu işlemde de yapısal öge görüntüde yer alan her piksel üzerinde tek tek dolaştırılır. Ancak bu işlemde yapısal ögenin orijin pikseli '1' değeriyle karşılaştığında yapısal öge içindeki piksellerin durumuna bakıldığı gibi yapısal öge içindeki '1' değerli piksellerin herhangi birinin altında görüntüde bir '0' değerinin bulunması halinde yapısal ögedeki diğer '1'lerinin altında yer alan piksellerle birlikte bu piksel de '0'a dönüştürülür (Şekil 3.29) [54].



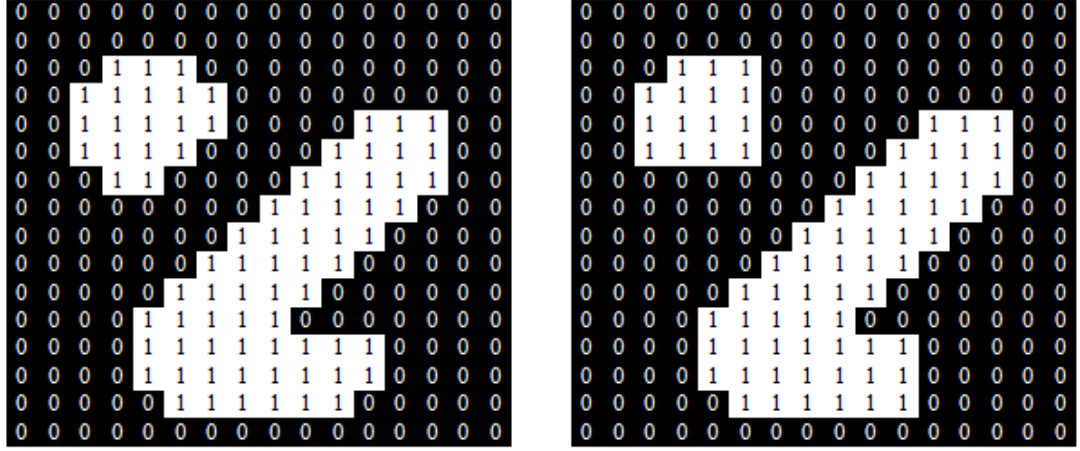
Şekil 3.29. Aşındırma işleminin temel mantığı

A görüntüsünde B yapısal ögesiyle yapılan aşındırma işlemi (ε) Formül 3.14'deki gibi gösterilir [18]:

$$\varepsilon(A) = \bigcap_{b \in B} A - b = A \theta B \quad (3.14)$$

3.2.6.2 Açma ve kapama işlemleri

Açma işlemi, aşındırma ve genişletme işlemleri sonrasında kullanılır. Bu işlemin temel prensibi, yapısal ögeye göre piksel sayısı daha az olan piksel gruplarının elenmesine, diğer piksellerin ise değiştirilmeden aynı yerinde kalmasına dayanır. Açma işlemi, nesnelere yer alan ince çizgileri ve ufak benekleri yok ederek nesnelere keskin hatları yumuşatır (Şekil 3.30).

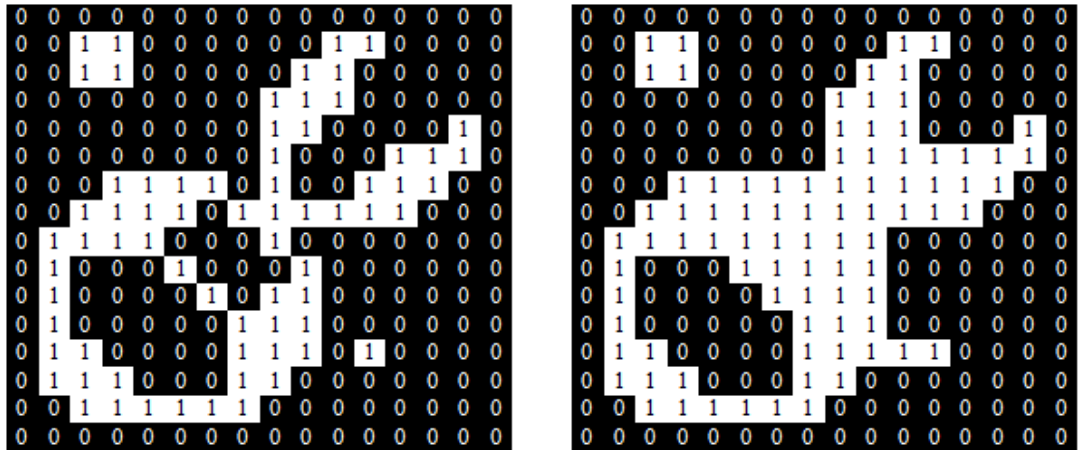


Şekil 3.30. Açma işlemi uygulanmış ikili görüntü

A görüntüsünde B yapısal ögesiyle yapılan açma işlemi Formül 3.15'deki gibi gösterilir:

$$A \circ B = (A \ominus B) \oplus B \quad (3.15)$$

Kapama operatörü, açma operatörünün tam tersi bir mantıkla çalışır. Genişletme ve aşındırma operatörleri sonrasında kullanılır. Kapama işlemindeki esas amaç boşlukların kapatılmasıdır (Şekil 3.31). Fakat bu operatör sadece objeye ilişkin boşlukların doldurulmasında başarılıdır. Kapama işlemiyle birlikte objelerdeki küçük girinti ve çıkıntılar ile küçük deliklerin yok edilmesi sağlanır. Bölgenin sınırları içinde kalan girintiler yok edilmiş, bölgenin dışına taşmış olan obje detayları ise özelliklerini kısmen kaybetse de görel olarak sağlam kalmış olurlar [55].



Şekil 3.31. Kapama işlemi uygulanmış ikili görüntü

A görüntüsünde B yapısal ögesiyle yapılan kapama işlemi Formül 3.16'daki gibi gösterilir [18]:

$$A \cdot B = (A \oplus B) \ominus B \quad (3.16)$$

3.2.6.3 Diğer morfolojik işlemler

Temel matematiksel morfoloji işlemleri arasında değerlendirilen genişletme, aşındırma, açma, kapama işlemlerinin yanısıra pekçok farklı uygulamada kullanılmak üzere geliştirilmiş morfolojik işlemler de bulunmaktadır. Bunların başlıcaları aşağıdaki gibi sıralanabilir [18]:

- İnceltme işlemi
- Kalınlaştırma işlemi
- İskelet (Orta Aks) işlemi

3.2.7. Diğer algoritmalar

Bu bölümde yukarıda ayrıntıları verilen görüntü işleme teknikleri dışında kalan ve gerçekleştirilen çalışmada aktif olarak kullanılan diğer görüntü işleme tekniklerine yer verilmiştir.

3.2.7.1 Görüntü netleştirme algoritmaları

Görüntü netleştirmede kullanılan algoritmaların başlıcaları; Wiener Algoritması, Düzenlenmiş (Regularized) Filtre, LucyRichardson Algoritması, Blind-Dekonvolüsyon Algoritması [39], Highpass filtresi, Sinc filtresi, Sobel operatörü filtresi ve görüntüde sınır belirlemede kullanıldığı gibi netleştirme amaçlı olarak da kullanılan Laplace algoritması [56] olarak sıralanabilir.

3.2.7.2 Gri renk skalasına dönüştürme ve renk düzeltme algoritmaları

Renkli görüntüler, birçok görüntü işleme algoritmasının sağlıklı sonuçlar vermesini engelleyebildiğinden görüntü içindeki renkli kısımlar gereksiz ayrıntılar olarak düşünülmektedir. Bu bakımdan renkli bir görüntünün işlenmesindeki en önemli ön aşamalardan biri de gri renk skalasına dönüştürmedir [23].

Renkli yani RGB (Red-Green-Blue: Kırmızı-Yeşil-Mavi) bir görüntüyü gri renk skalasına dönüştürme, esasen diğer algoritmaların üzerinde çalışacağı verinin sadeleştirilmesini sağlaması bakımından diğer algoritmalara temel teşkil eder. RGB moddaki bir görüntüde her piksel kırmızı, yeşil ve mavi renk tonunu belirten üç farklı byte ile temsil edilir ve bu üç byte'ın birleşimi o pikselin rengini oluşturur. RGB moddaki bir görüntünün griye dönüştürülmesinde bir piksel 3 byte yerine 1 byte ile temsil edilir hale getirildiğinden bu tek byte pikselin siyah beyaz ton skalasındaki tonunu belirtir ve bu tür byte' lardan oluşan resim gri modda olur [41]. Şekil 3.32'de normal bir görüntünün (a) gri seviyeye dönüştürülmüş hali (b) görülmektedir.



a. Normal görüntü



b. Gri seviyeye dönüştürülmüş görüntü

Şekil 3.32. Normal görüntüye gri seviyeli görüntüye dönüştürülmesi

Gri seviyeli görüntüler, renk bilgisi içermediğinden sadece parlaklık bilgisini tanımlarlar [3]. Gri seviyeli görüntüdeki her bir piksel 0 – 255 arası bir parlaklığa sahiptir. Burada 0 siyah rengi 255 ise beyaz rengi gösterir. Bu değerler arasındaki kısımlar ise gri tonları oluşturur. Bir gri tonlu görüntüde 256 tane farklı gri değer bulunur ve 256 gri değer bir byte olarak tanımlanır (1 Byte=8 Bit ve $2^8=256$) [21].

3.2.7.3 Plaka okuma ve karakterlerini tanıma algoritmaları

Plaka okuma ve karakterlerini tanıma algoritmaları, nesne tanıma, eşik belirleme v.b. yöntemlerle bir plakanın karakterlerini (harfler, rakamlar, semboller vb) tanıyan algoritmalarıdır. Bu algoritmalarda; araç ve plaka olma potansiyeli taşıyan aday görüntüler, en/boy oranı ve plaka içerme durumu göz önüne alınarak sınıflandırılmakta, plakaların bulunmasında köşe bulma algoritmalarından [23] ve plakalardaki karakterleri ayırt etmek için Blob Coloring (İkili Renklendirme) gibi algoritmalarından [57] yararlanılmaktadır.

3.2.7.4 Yüz tanıma algoritmaları

Yüz tanıma algoritmaları; verilen bir görüntüde yüz tanıma yöntemlerinin kullanılmasıyla oluşturulan algoritmalarıdır. Bu algoritmaların oluşturulmasında; sayısız farklı test görüntüsü için algoritmanın performansı değerlendirilmekte; bu da algoritmanın başarısını göstermektedir.

Birçok farklı yüz tanıma algoritması bulunmaktadır. Geleneksel algoritmalar, iki boyutlu (2D), 24x24 piksel boyutlu örnekler kullanırken, yeni yüz tanıma yöntemleri üç boyutlu (3D) ve/veya yüksek çözünürlüklü (HD) yaklaşımlar kullanmaktadır. Başlıca yüz tanıma algoritmalarına örnek olarak; Eigenfaces Algoritmaları, Fisherfaces Algoritmaları ve Local Binary Pattern Histograms (LBPH) verilebilir [15].

3.3. Kablosuz Ağlar

Bu bölümde kablosuz ağın tanımı, çeşitleri, kablosuz ağ cihazları ve kablosuz ağların avantajları ele alınacaktır.

3.3.3. Kablosuz ađ tanıml

Kablosuz ađ, kablolu ađların yanı sıra bir noktadan başka bir noktaya kablo hattı kullanmadan veri, ses ya da görüntü taşınmasını sađlayan iletişim ađlarına denir. Buna göre kablosuz ađları kablolu ađlardan ayıran en önemli noktanın, iletim ortamı olarak havanın kullanılması olduđu söylenebilir [58].

3.3.4 . Kablosuz ađ çeşitleri

Kablosuz Ađları, türlerine göre üç grupta toplamak mümkündür. Bunlar [19]:

- Kablosuz Yerel Alan ađları
- Kablosuz Kişisel Ađlar
- Kablosuz Geniş Alan Ađları

3.3.4.1 Kablosuz yerel alan ađları (WLAN)

Kablosuz yerel alan ađları (WLAN: Wireless Local Area Network), kablolu yerel alan ađlarına alternatif olarak veri alışverişinde kablolar yerine radyo frekanslarının ve kızılötesi ışınların kullanıldığı bir ađ sistemidir. Kablosuz yerel alan ađları, esasen Ethernet, Token Ring vb. ađ teknolojilerindeki fiziksel ve veri bağlantı katmalarının kablosuz versiyonudur. Bu bakımdan yeni bir özellik değildir, sadece eski teknolojinin yeni ortamda kullanılabilir hale getirilmesidir. 1997 yılında IEEE “The Institute of Electrical and Electronics Engineers” (Elektrik ve Elektronik Mühendisleri Enstitüsü) tarafından kablosuz iletişimin standartları belirlenmiştir. İlk olarak 802.11 kablosuz iletişim protokolü olarak belirlenen standartlar geliştirilerek, 1999’da 802.11b, 802.11a; 2003’de 802.11g ve 2009’da 802.11n standartları olarak yayınlanmıştır. Farklı şekilde, farklı alanlarda kullanılan 802.11 ađları, Wi-Fi “Wireless Fidelity” (Kablosuz Bağlantı Alanı) ađları olarak bilinmektedir. 802.11 standartları ile 2,4 Ghz’de 2 Mbps’a kadar iletişime imkân sağlanırken; 802.11n standartları ile hem 2,4 Ghz hem de 5 Ghz’de 144 Mbps’a hatta bazı eklentilerle 300 Mbps’a kadar iletişim mümkün olmaktadır [19, 59, 60].

3.3.4.2 Kablosuz kişisel ağlar (WPAN)

Ev ya da küçük iş yerlerinde birkaç bilgisayar ve çevre biriminden oluşan ağlara Kişisel Alan Ağları (PAN: Personal Area Networks) denmektedir. Kablo yerine kablosuz iletişim teknolojisi kullanılması durumunda ise Kablosuz Kişisel Alan Ağları (WPAN: Wireless Personal Area Networks) olarak adlandırılmaktadır. Bir başka ifadeyle WPAN'lar yakın mesafedeki elektronik cihazları kablosuz olarak birbirine bağlayan ağlardır. Bu tür sistemler diğer ağlara kıyasla daha düşük veri hızına ve daha kısa iletişim mesafesine sahiptirler. WPAN'ların hızları 1 Mbps ve menzilleri 10 metre civarındadır [61].

Kablosuz Kişisel Ağlar; Bluetooth, IRDA (Infrared: Kızılötesi) ve Home RF ağlarıdır. Bluetooth daha ziyade kişinin etrafındaki sayısal cihazlar arasında kablosuz bağlantı kurmak için geliştirilmiştir. HomeRF ise ev veya küçük işyerlerinde bir kablosuz ağ oluşturmak üzere tasarlanmıştır [61]. Kablosuz kişisel ağlar, kısa mesafeli bilgi işlenmesinde etkilidir. Çok farklı özelliklerde ofis makinelerinin haberleşmelerinin sağlanması hedeflenmiştir. IRDA yapısal olarak yetersiz özellikleri nedeniyle bluetooth'a yerini bırakmış durumdadır. Düşük maliyetli uygulamalardır [60].

a) Bluetooth

Bluetooth, dizüstü bilgisayarlar, cep bilgisayarları, modemler, LAN erişim noktaları ve telefonlar (cep, ev ve işyeri telefonları) gibi sayısal cihazlar arasında veri iletişimini sağlamak üzere oluşturulan endüstri konsorsiyumunun adıdır. Bluetooth teknolojisi 2.4 GHz bandında ilk olarak Ericsson Mobile Com. tarafından 1994 yılında geliştirilmiştir. Bluetooth, kısa mesafede bilgisayar, fare (Mouse), klavye, yazıcı, sayısal kamera ve telefon gibi cihazlar arasında kablosuz iletişimi sağlayan teknolojidir. Bluetooth aynı zamanda ağ bağlantısının çeşitli cihazlara dağıtılmasını da sağlar. Bluetooth ses iletimine de olanak tanımaktadır. Bluetooth uygun mesafedeki herhangi bir cihazı kablosuz olarak bir başka cihaza bağlayabilir. WLAN sistemleri 100 metre iletişim mesafesine sahip iken Bluetooth'un mesafesi yaklaşık 10 metredir [61].

b) IrDA (Infrared: Kızılötesi doğrudan erişim)

Kızılötesi (IR: Infrared), nispeten daha düşük seviyeli bir enerji olup duvar ya da diğer nesnelere geçmez. Radyo frekanslarıyla değil ışık darbeleriyle çalışır. Bu nedenle veri iletiminin olması için iki cihazın birbirini görmesi gerekir. Kişisel Sayısal Yardımcı (PDA), kişisel bilgisayar, uzaktan kumanda aygıtları gibi aygıtlar arasında bağlantı kurmak amacıyla kullanılabilir. Kızılötesi Doğrudan Erişim (IrDA) bağlantı noktası olarak bilinen özel bir iletişim bağlantı noktası, aygıtlar arasında bilgi alışverişi yapılmasını sağlamak için kızılötesi bağlantısı kullanır. Kızılötesi ile cihazlar arasında yalnızca birebir bağlantı gerçekleştirilebilir ve 900 MHz hızında veri iletimi gerçekleştirilebilir. Düşük güç tüketimi, radyo dalgalarından etkilenmemesi, kapalı ortamlarda izinsiz dinlemeye ve bozucu etkilere karşı tam bir güvenlik sağlaması ve herhangi bir lisans gerektirmemesi kızılötesi teknolojinin avantajlarıdır. Dezavantajları ise iletişim mesafesinin kısa olması (10-15 m), sinyallerin katı cisimleri geçememesi ve hava şartlarından etkilenmesidir [58].

c) Home RF

HomeRF, ev ve küçük işyerleri için geliştirilen kablosuz erişim standardıdır. Özellikleri Mart 1998'de kurulan Home Radio Frequency Working Group (HomeRF WG) isimli çalışma grubu tarafından ortak kablosuz erişim protokolü (Shared Wireless Application Protocol-SWAP) adı altında duyurulmuştur. HomeRF evde bulunan PC, kordonsuz telefon ve diğer cihazlar arasında ses ve veri iletişimini kablolama masrafına gerek kalmadan kablosuz olarak sağlamaktadır. HomeRF sistemi 2.4GHz ISM bandında çalışmakta; 50 metreye kadar mesafede veri iletişimi sağlamaktadır. HomeRF'in iletişim mesafesi işyeri uygulamaları için kısadır. Ancak ev uygulamaları için yeterlidir [61].

d) Kablosuz geniş alan ağları

Bir kampüs, bir şehir, bir ülke ya da dünya genelinde binlerce kilometre mesafe arasında kablosuz iletişimi sağlayan ağlara Kablosuz Geniş Alan Ağ (WWAN:

Wireless Wide Area Network) denilmektedir [58, 60]. İletişim yüksek düzey radyo dalgaları kullanılarak gerçekleştirilir. Anahtarlama merkezi olarak anılan bir merkez aracılığıyla WWAN'lar telefon hatları ya da sabit kablo yapıları üzerinden merkezî ağ sistemine bağlanabilirler [60]. En iyi örneği cep telefonu şebekeleridir. Bu ağlar, Kod Bölmeli Çoklu Erişim (CDMA) ya da Mobil İletişim için Küresel Sistem (GSM) gibi teknolojileri kullanır ve genellikle kamu kurumları tarafından düzenlenir [58].

3.3.5. Kablosuz ağ cihazları

Bir kablosuz ortama bağlanabilen tüm bileşenler ağın istasyonları olarak adlandırılır. Tüm istasyonlar kablosuz ağ kartları ile donatılmıştır. Kablosuz istasyonlar, erişim noktaları (AP–Access Point) ve kablosuz istemciler olarak iki ayrı kategoriye ayrılır. Erişim noktaları, yönlendiriciler ve baz istasyonlarıdır. Kablosuz istemciler ise dizüstü bilgisayarlar, masaüstü bilgisayarlar, kablosuz telefonlar, mobil cihazlar gibi kablosuz ağ kartı ile donatılmış diğer cihazlardır [58].

3.3.6. Kablosuz ağların avantajları

Kablosuz ağların kablolu ağlara göre avantajları şöyle sıralanabilir [60].

- Gezici satış, araştırma şirketleri ve gezici hastaneler, şantiyeler ve üniversite kampüsleri gibi mobil yapıdaki çalışma ortamlarının ihtiyaç duyduğu esnekliği sağlayamaz. Kablosuz ağlar, bu tür ortamlarda çalışanlara/bulunanlara, hareketlerinde esneklik (mobilite) imkânı sunar.
- Kablosuz ağların donanımı hızlı ve basittir. Kablosuz yerel alan ağlarının kurulumları basit olup herhangi bir uzmanlık gerekmez.
- Kablolu ağlar donanım yönünden kablo ile bağlanabilen parça ve cihazlarla kısıtlıdır. Kablosuz ağlar ise bu yönden esnek bir donanıma sahiptir. Ağa bağlanabilen her türlü cihaz ve parça ile kablosuz ağlarla yapılması mümkün olmayan işlerde çözüm sağlayabilir.
- Başlangıçta kablolu ağlara göre daha maliyetli olan kablosuz ağlar, ilerleyen zamanda daha fazla kişi tarafından tercih edilmesi ile birlikte bu alanda yapılan

maliyetlerin kiři bařına dūřen oranı yönünden kablolu ađlara göre daha ucuz hale gelmiřtir.

- Kablolu ađlarda, ađı kullanan kiři/cihaz sayısı kadar koblo bađlantısı yapılması gerekir. Kiři sayısı arttıkça kablo uzunlukları artar, veri iletim verimliliđi dūřer. Ayrıca sonradan ađa dahil edilen cihazlar için çekilen harici kablolar zamanla yıpranabilir, kırılabilir, iřlevsiz hale gelebilir. Bu durum hem maliyeti arttırır, hem de ađ yapısının genişletilebilirliğini iřlevsizleřtirir. Kablosuz ađların yapısı kolayca istenildiđi řekilde deđiřikliđe uđratılabildiđi için iki kiři arasındaki bir bađlantıdan binlerce bilgisayar, cep telefonu ve tabletle oluřturulan yapıya kadar çözümler sađlayabilir.

4. GÖRÜNTÜ İŞLEME ALGORİTMALARININ KABLOSUZ AĞ ÜZERİNDEN GERÇEKLENMESİ

Bu bölümde görüntü işleme algoritmalarının kablosuz ağ üzerinden gerçekleşmesi konusu çerçevesinde masaüstü ve web tabanlı/çevrimiçi görüntü işleme uygulamaları, kablosuz ağ üzerinden görüntü işlemenin kapsamı ve uygulama alanları, örnek uygulama ve çalışmalar, hazır arayüzler ile görüntü işleme ve performans analizleri ele alınacaktır.

4.1 . Görüntü İşleme Uygulamaları

Görüntü işleme uygulamalarını, masaüstü ve web tabanlı uygulamalar olarak iki bölümde inceleyebiliriz.

4.1.3 . Masaüstü tabanlı uygulamalar

Masaüstü tabanlı görüntü işleme uygulamalarının başlıcaları şunlardır [2]:

- Adobe Photoshop
- Adobe Illustrator
- GIMP
- PixBuilder Studio
- ArtWeaver
- x360 Görüntü İşleme Yazılımı
- GdPicture
- PhotoScape

4.1.4. Web tabanlı/çevrimiçi uygulamalar

Web tabanlı/Online görüntü işleme uygulamaları kullanıcıya tarayıcısı ve internet bağlantısı olan herhangi bir bilgisayar üzerinde çalışma imkânı sağlaması bakımından masaüstü uygulamalardan daha avantajlıdır. Çoğu durumda, kullanıcı çalışmasını online olarak kaydedebilir ve böylece masaüstü yazılım yükleme ve platform zorunluluğu ortadan kalkar. Web tabanlı/online görüntü işleme uygulamaların diğer avantajları ise aşağıdaki gibi sıralanabilir [2]:

- 1) Sadece bir tıklamayla fotoğrafların düzeltilmesi,
- 2) Sonuçların detaylı olarak ileri kontroller kullanılması yoluyla ince ayarlardan geçirebilmesi,
- 3) Kırpma, boyutlandırma ve döndürme gibi işlemlerin gerçek zamanlı olarak yapılabilmesi,
- 4) Sanattan eğlenceye kadar birçok özel efektin kullanılabilmesi sayesinde görüntülerin işlenerek istenen görüntülerin elde edebilmesi,
- 5) İçerik tabanlı görüntü erişim sistemleri sunabilmesi,
- 6) Görüntüler arasındaki ilişkilerin belirlenebilmesi,
- 7) Görüntülerin düzenlenebilmesi ve sınıflandırılabilmesi,
- 8) Görüntü sorgulamaları yapılabilmesi,
- 9) Görüntüdeki nesnelerin sınıflandırılabilmesi,
- 10) Kaynak görüntüdeki benzer nesnelerin tespit edilebilmesi.

En popüler web tabanlı/online görüntü işleme uygulamaları şunlardır [2]:

- Pixlr
- Splashap
- FotoFlexer
- SUMO Paint
- TinEye
- Wash
- Pixolu

- Wang
- iPiccy
- myImager

Diğer web tabanlı/online görüntü işleme uygulamaları ise aşağıda sıralanmıştır [2]:

- Matlabserver Web-enabled Image Processing Operators
- TinEye Reverse Image Search
- Sky Image Processor
- Catenary Online Image Processing
- Geoscience Australia Image Processing Online
- MyImager Online Image Editor
- Picnik Photo Editor
- Clark online image and photo editing
- ImageEditor Free Online Image Editor and Photo Editor
- Pixastic - Online JavaScript Photo Editor
- Shutterfly Online Photo Processing
- Medical-Image Processing
- Washington University CS Object and Concept Recognition for Content-Based Image Retrieval
- Izhuk Image upload and processing applet
- WebPicTool Free Online Photo Editor
- Free Online Photo Editor

4.2. Kablosuz Ağ Üzerinden Görüntü İşleme

Bu bölümde kablosuz ağ üzerinden görüntü işlemenin kapsamı ve uygulama alanları, örnek uygulama ve çalışmalar, hazır arayüzler ile görüntü işleme ve kablosuz ağ üzerinden uygulanan görüntü işleme algoritmalarının performans analizleri ele alınacaktır.

4.2.3. Kapsamı ve uygulama alanları

Kablosuz ağ üzerinden görüntü işlemenin çok çeşitli ve farklı uygulamaları bulunmaktadır. Yoğun kullanım alanlarının internet üzerinde gerçekleşmesiyle görüntü işleme kapsamında web tabanlı çeşitli proje, çalışma ve araştırma gibi uygulamalar gerçekleştirilebilir. Teknolojinin gelişmesi ve yaygınlaşması, internet erişim hızı ve altyapısına bağlı olarak uygulama geliştirmede önemli avantajlar sunmaktadır. Kablosuz ağ üzerinden görüntü işleme uygulamaları platformdan bağımsız olarak geliştirilebilmektedir. Görsel web araçlarını oluşturmak, uygulamaların görselleştirilmesi ve internet üzerinden sunulması da önemli avantajlarından. Gelişen teknolojiyle birlikte görüntülerin Google images, Yandex image search, TinEye, PicSOM, picsearch gibi online görüntü arama motorları yoluyla internet üzerinde aranması, ilgili görüntülerin ya da benzer görüntülerin web sayfalarında yer alıp almadığının sorgulanması ve sonuçlarının alınması da mümkündür [2].

4.2.4. Avantajları

Kablosuz ağ üzerinden görüntü işleme sayısız avantaj sunmaktadır. Bunların başlıcaları şu şekilde sıralanabilir:

- Çoklu ortam tabanlı veriler üzerinde, metin tabanlı uygulamaların sağladığından daha fazla işlem yapılmak istenebilir ve bu nedenle görüntülerin kullanılması ve işlenmesi gerekir.
- İnternet üzerinde görüntü içerikli işlemlerin icra edilebilmesi, aktarılıp sunulabilmesi ve benzeri paylaşım işlemleri görüntü işleminin internet üzerinde icra edilmesini gerektirir.
- Haberleşme, ticaret, kamu hizmetleri gibi birçok temel işlevin internet ortamı üzerinden yürütülmesi,
- Bilişim teknolojisindeki gelişmelerle birlikte günümüzde, dünyadaki gelişimler internet aracılığı ile anında izlenebilmekte, dolayısıyla bilginin hızla yayılması olanaklı kılınabilmektedir. Bir bilgi ve eğitim ortamı da olan internet ile içerik oluşturma
- Yalnızca metin tabanlı işlemleri icra etmek yerine görüntü tabanlı işlemlerin de gerçekleştirilmesi,
- Web tabanlı araçlar sayesinde değişik meslek gruplarından kişilere ve araştırmacılara ulaşabilinmesi, test olanağı sunulması ve problemlere çözüm getirilebilmesi,
- Görüntü işleme alanı ile daha çok kişinin tanışması,
- Geliştirilen uygulamaların çeşitli teknolojiler kullanılarak internet üzerinden kullanılabilir hâle getirilmesi ve bu yaklaşım ile çözümün daha hızlı ve kolay bir şekilde kullanıcıya internet üzerinden etkileşimli olarak sunulması,
- Gerçekleştirilen uygulamalarla görüntü işleme kapsamında web tabanlı içerik oluşturabilinmesi,
- Görüntü işlemeyi icra edebilmek için gerekli ortamların birleştirilerek internet üzerinden kullanılabilmesi ile birçok projenin daha çok kişiye, çok daha hızlı bir şekilde ulaşabilmesi,
- Geliştiriciler ve araştırmacılar için programlama, ilgili kullanıcılar için kullanım rahatlığı,
- Özellikle, etkileşimli olarak geliştirilen görüntü işleme kapsamındaki uygulamaların internet üzerinden kullanılması, anlatılmak istenen konuyu, bu

konu hakkında hiçbir bilgisi olmayan kullanıcılara dahi kolay bir şekilde sunabilmesi,

- İnternet üzerinde görüntüler işlenebilir olması
- Yardımcı program ve kütüphanelerin indirilmesi ile web tabanlı uygulamaların daha hızlı geliştirilebilmesi.

4.2.5 . Görüntü işlemede kullanılan hazır arayüzler

Görüntü işlemede kullanılan hazır arayüzler şunlardır [2]:

- PaintbrushJS: Javascript ve Canvas ile oluşturulmuş tarayıcı tabanlı görüntü işleme kütüphanesidir.
- Victor Image Processing Library: Geliştiricilere görüntü uygulamaları oluşturmak için pek çok fonksiyon sunan bir araçtır ve uygulamalara güçlü görüntü işleme, renk indirgeme, görüntüleme gibi imkânlar sunmaktadır.
- OpenCV: Görüntü işleme tekniklerini uygulayabilmede sayısız fonksiyon sunmaktadır.
- Pixastic: Bir JavaScript kütüphanesidir ve çeşitli işlemleri, filtreleri ve efektleri görüntülere uygulama imkânı tanır.
- Image Processing Toolbox for MATLAB: MATLAB için görüntü işleme aracıdır.
- Emgu CV: Bir OpenCV görüntü işleme kütüphanesidir.

Bunlara ek olarak masaüstü ve çevrimiçi/web tabanlı uygulamalar ile diğer araçlar da bulunmaktadır. Bu hazır arayüzler ile daha hızlı ve kolay görüntü işleme yapılabilmesi ve yeni görüntü işleme algoritmalarının geliştirilmesi mümkün olabilmektedir.

4.2.6 . Raspberry Pi ve kablosuz kullanımı

2009 yılında İngiltere’de bulunan ve aynı ismi taşıyan bir vakıf tarafından geliştirilen ve desteklenmekte olan Raspberry Pi [62]; öğrencilerin, amatörlerin ve hobicilerin

kullanımı için geliştirilmiş olan, yaklaşık olarak kredi kartı ebatlarında, tek bir elektronik devre kartından oluşmakta olan bir mini bilgisayar olarak tanımlanabilir [63]. Şekil 4.1’ de Raspberry Pi modülü görülmektedir.



Şekil 4.1 Raspberry Pi

Ethernet, HDMI, SD Kart okuyucu, ses çıkışı gibi pek çok gelişmiş çevre birimine sahip olan Raspberry Pi'nin üzerine Linux, Fedora gibi işletim sistemleri kurulabilmesi sayesinde çok yönlü bir mini bilgisayar yapısı elde edilmiştir. Raspberry Pi'nin, kendi üzerine kurulan linux işletim sistemi üzerinden python diliyle programlanması sayesinde giriş-çıkış pinleri, ethernet bağlantısı ve RCA Video bağlantısı gibi yapısal özellikleri yardımıyla farklı projelerin geliştirilebilmesi mümkün olabilmektedir [64].

Raspberry Pi' nin satışına 2012 yılı Şubat'ında başlanmış; ürünün piyasaya sürümü Raspberry Pi Model A ve Raspberry Pi Model B olmak üzere 2 farklı model altında gerçekleşmiştir. 'A modeli' nin tanıtımında 128MB Ram' e sahip olacağı belirtilmesine karşılık, satış öncesindeki bir değişiklik ile 256MB Ram ile piyasaya sürülmüştür. A modeli yalnızca bir adet USB girişine sahipken, B modelinde iki adet USB, bir adet Ethernet girişi bulunmaktadır. Her iki modelin de sahip olduğu USB girişleriyle standart tak-çalıştır USB fare ve klavyelerle sorun yaratmadan bir şekilde çalıştırılabilmektedir. 2012'nin Nisan ayı itibariyle Raspberry Pi' nin her iki modelinin

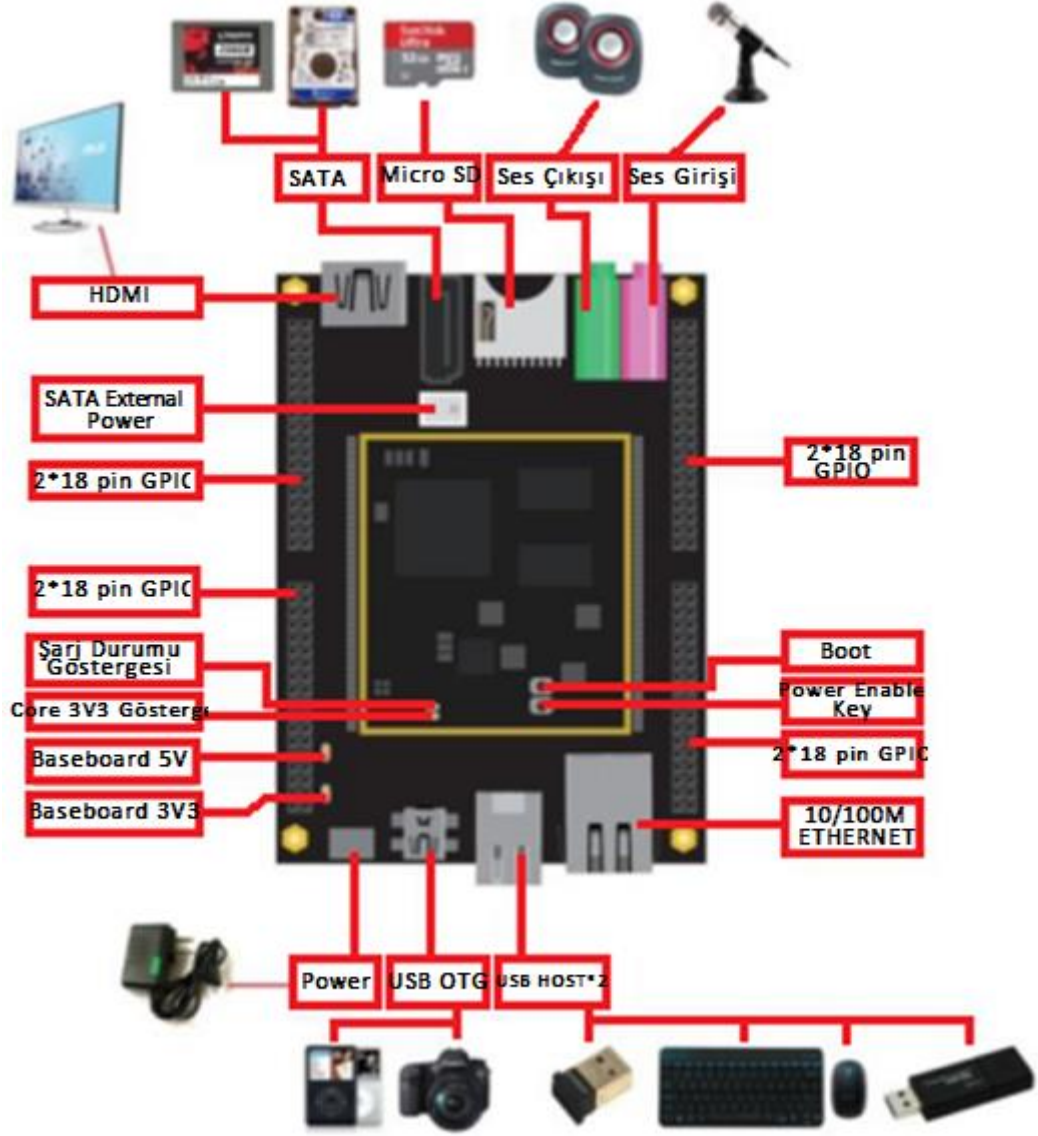
OpenCV üzerinden alınabilen kaynak kodlarla açık kaynak olarak kullanımı mümkün hale gelmiş; bu da dileyen herkesin tüm devre çizimleri ve teknik detaylarına ulaşabilmesini mümkün kılmıştır [65].

Raspberry Pi'nin klasik bilgisayarlardan en önemli farkı, ARM tabanlı ARM1176JZF-S 700 MHz bir işlemci (CPU) kullanmasıdır [65, 66]. Klasik bilgisayarlardaki serial çıkışlar yerine Raspberry'de pin çıkışları yer almaktadır [62]. Üzerindeki grafik işlemcisi HD görüntü kalitesi ve yüksek çözünürlük sağlayan VideoCore IV GPU olup başlangıçta 256MB belleğe sahipken sonraları bu bellek 512MB'a çıkarılmıştır [67]. Sabit disk olmayan Raspberry Pi'de, veri depolaması ve booting için SD kart kullanılmaktadır [66]. Raspberry Pi'nin en son modeli ise 2016 Mart ayında piyasaya sürülen Raspberry Pi 3'dür (Şekil 4.2).



Şekil 4.2 Raspberry Pi 3

Şekil 4.3'de Raspberry Pi bağlantı noktaları görülmektedir.



Şekil 4.3 Raspberry Pi Bağlantı Noktaları

Raspberry Pi 3'ün, yeni özellikleriyle birlikte hız açısından da ciddi bir gelişme kaydettiği görülmektedir. İşlemcisi 64-bit, dört çekirdekli 1,2GHz ARM Cortex A-53 olarak belirlenmiştir. Böylece 900MHz, 32-bit işlemciye göre büyük bir hız artışı sağlanmıştır. RAM ise hâlen 1GB olarak sunulmaktadır. Yine de Raspberry Pi 3'ün önceki versiyonu olan ve 2015 yılının Şubat ayında piyasaya sürülmüş olan Raspberry Pi 2' den %50 daha hızlı olduğu bildirilmiştir. Bu hız artışının da Raspberry Pi'nin eksiksiz bir bilgisayar olarak görülmesi açısından yeterli olacağı belirtilmektedir [63].

Ancak Raspberry Pi gibi geliştirme kartları üzerinde yapılan OpenCV derleme işleminin tamamlanma süresinin biraz uzun olacağı da bildirilmiştir [68].

Raspberry Pi'nin üzerindeki ekran kartı modülüyle herhangi bir monitör üzerinden görüntü alınabilmesi mümkün olduğu gibi kişisel bilgisayarlarda kullanılan çoğu yazılımı da bu platform üzerinde bulunabilmektedir [65]. Raspberry Pi üzerine kablosuz erişim özelliği Pi-Hut USB kablosuz erişim modülü eklenerek sağlanabilmektedir [69].

4.3 . Test Sonuları

Bu blmde, alıřma kapsamında grnt iřleme algoritmalarının eř zamanlı srelere ayrılarak kablosuz ađ zerinden gereklenmesine ynelik geliřtirilen uygulamaya iliřkin temel bilgiler, uygulama ařamaları, test sonuları ve uygulamanın performans analizleri sunulacaktır.

4.3.3. Uygulamaya iliřkin temel bilgiler

Bu kısımda uygulamada kullanılan temel aralar, kullanılan yazılım dili ve arayz, uygulamanın desteklediđi algoritmalar, resim formatları ve resim dosya boyutuna iliřkin bilgiler sunulacaktır.

4.3.3.1 Uygulamada kullanılan temel aralar

Uygulamada kullanılan temel aralar; bilgisayar, raspberry pi3 ve modemdir.

4.3.3.2 Uygulamada kullanılan yazılım dilleri ve arayz

Ana uygulamada C# dili, Raspberry Pi'de ise Java Dili kullanılmıř olup, OpenCV ktphanesinden yararlanılmıřtır.

4.3.3.3 Uygulamanın desteklediđi algoritmalar

Uygulamanın bařlangı versiyonunda Canny, Laplacian, Sobel, LoG, Harris ve Gaussian gibi kenar ve kře bulma algoritmaları kullanılabilmele birlikte diđer algoritmaların eklenmesi de mmkndr.

4.3.3.4 Uygulamanın desteklediği resim formatları

Uygulamanın başlangıç versiyonu jpg, bitmap, png ve tiff formatlarını desteklemektedir. Özel resim formatlarının codecleri bulunması halinde diğer resim formatlarının desteklenmesi de mümkündür.

4.3.3.5 Uygulamanın desteklediği resim dosya boyut sınırı

Uygulamada yüklenecek resimler için herhangi bir dosya boyutu yoktur. Ancak yine de dosya boyutları, kablosuz ağın bant genişliği ile sınırlı olduğu için uygulamada ağ bant genişliğini aşan dosya boyutuna sahip resimler desteklenmemektedir.

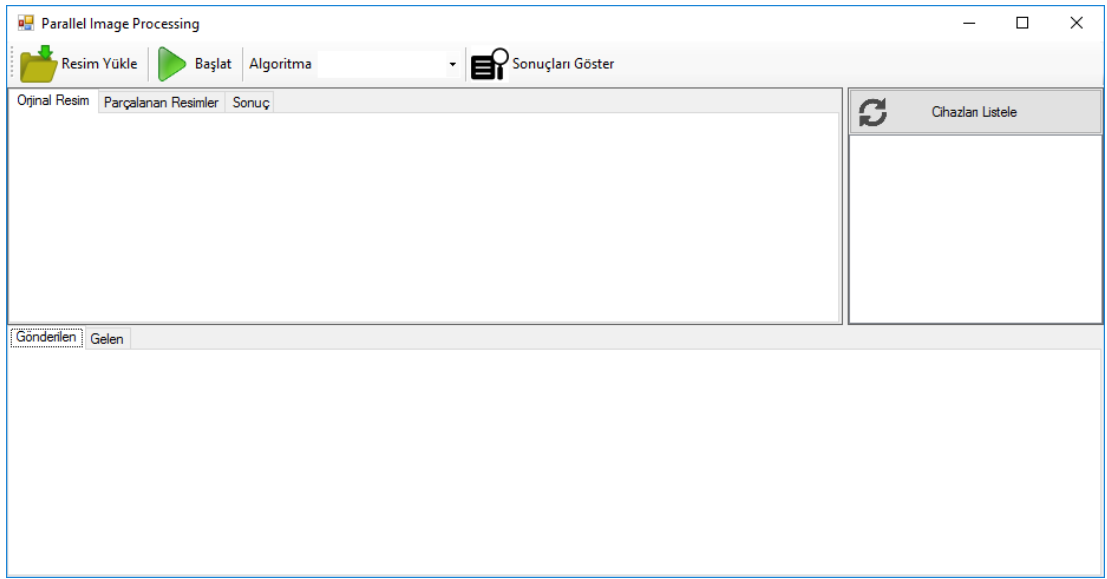
4.3.4. Uygulama aşamaları

Paralel görüntü işleme uygulamasının aşamaları kısaca şu şekilde özetlenebilir:

- Ana bilgisayarda bulunan uygulama ile bir resim seçilerek, seçilen resime hangi görüntü işleme algoritması uygulanması isteniyorsa bu algoritma seçilir (Örneğin; Canny kenar bulma algoritması).
- Daha sonra kablosuz ağ üzerinden ağda kaç cihaz olduğu ve cihazların ip adresleri Udp protokolü yardımıyla broadcast adı verilen yöntemle tespit edilir.
- Resim, ağda tespit edilen cihaz sayısı kadar eşit parçaya bölünerek bu cihazlara kablosuz ağ üzerinden gönderilir.
- Her cihaz seçilen algoritmayı kendisine gelen resim parçasına uygulayarak ağ üzerinden tekrar ana bilgisayara gönderir.
- Ana bilgisayardaki uygulama sayesinde, parçalanarak işlenmiş (seçilen algoritma uygulanmış) resimler, Tcp protokolü ile toplanıp aynı protokol ile tekrar birleştirilerek işlem tamamlanır.

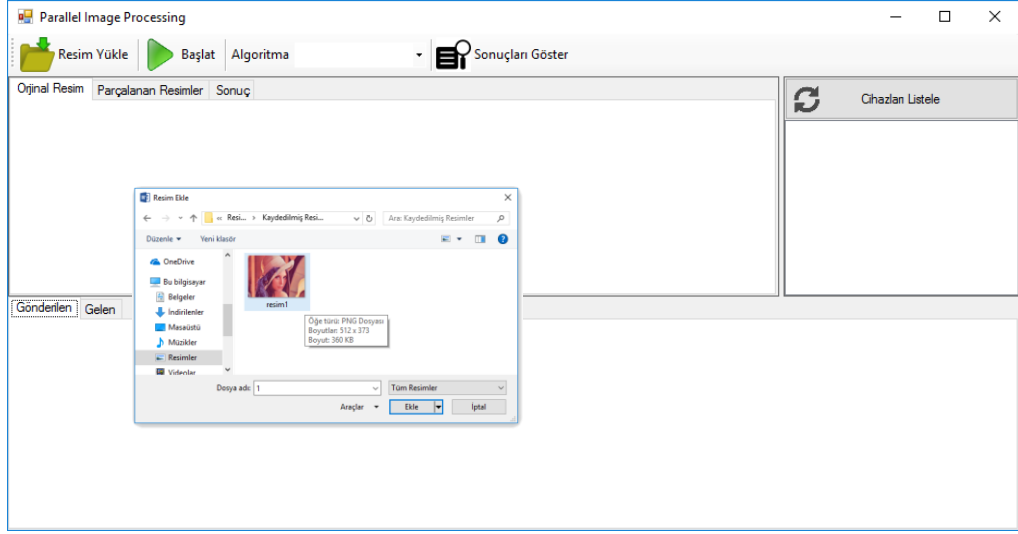
Sonuç olarak, algoritmanın resme uygulanması paralel işleme uygulaması sayesinde tek bir cihazın yaptığından daha kısa sürede gerçekleştirilebilmiştir.

Görüntü işleme algoritmalarının kablosuz ağ üzerinden birden fazla cihaz ile eş zamanlı süreçli gerçekleşmesini sağlayan uygulamanın arayüz görüntüleri ve yukarıda özetlenen aşamaların açıklamaları aşağıda sunulmuştur: Şekil 4.4’ de paralel görüntü işleme uygulamasının başlangıç görünümü sunulmuştur.



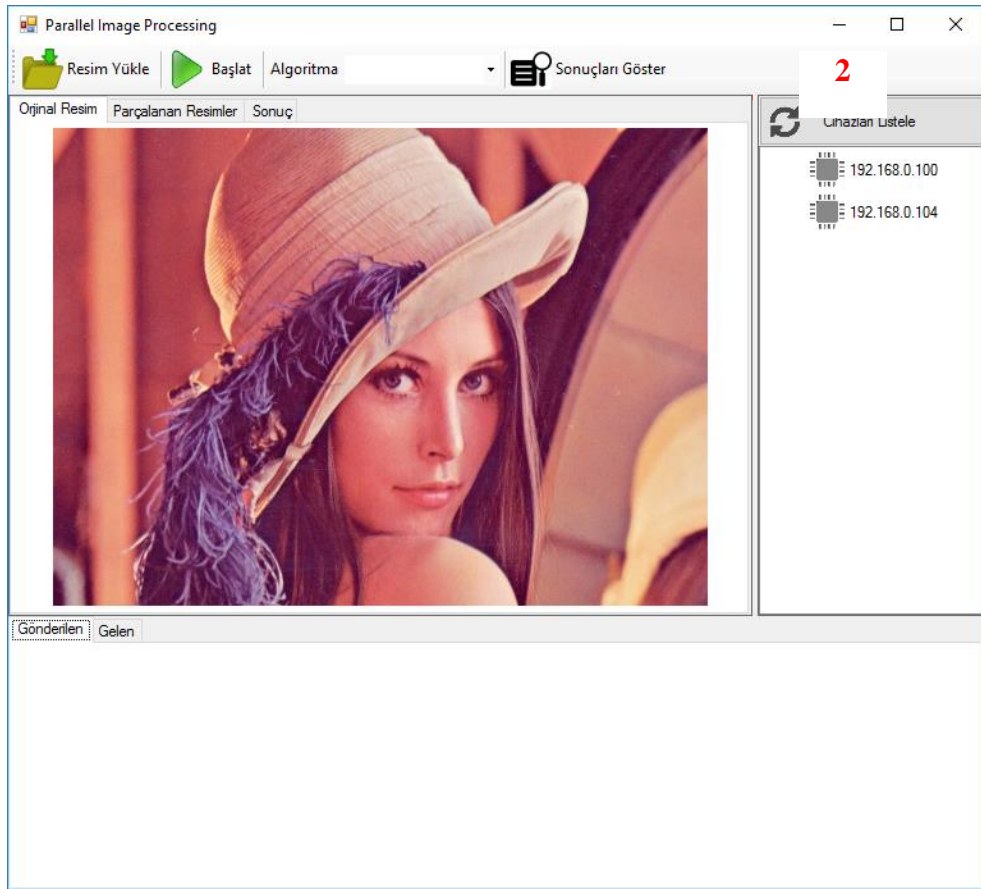
Şekil 4.4 Paralel Görüntü İşleme Uygulamasının Başlangıç Görünümü

“Resim yükle” butonuna basılarak ana bilgisayarda bulunan uygulama ile bir resim seçilerek yüklenir (Şekil 4.5).



Şekil 4.5 Resmin Yüklenmesi

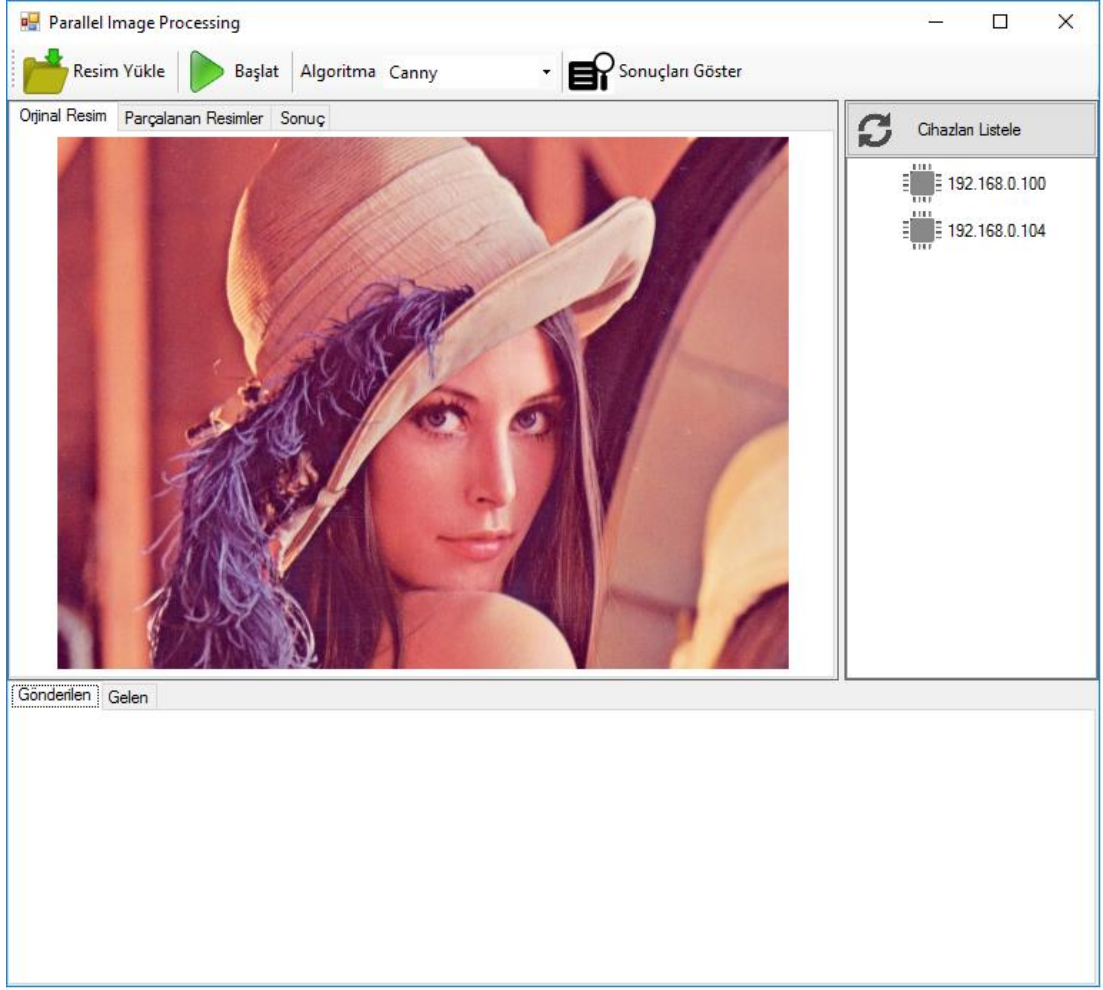
Resim yükledikten sonra “Cihazları Listele” butonuna basılarak ağdaki cihazlar bulunur (Şekil 4.6).



Şekil 4.6 Ağdaki Cihazların Bulunması

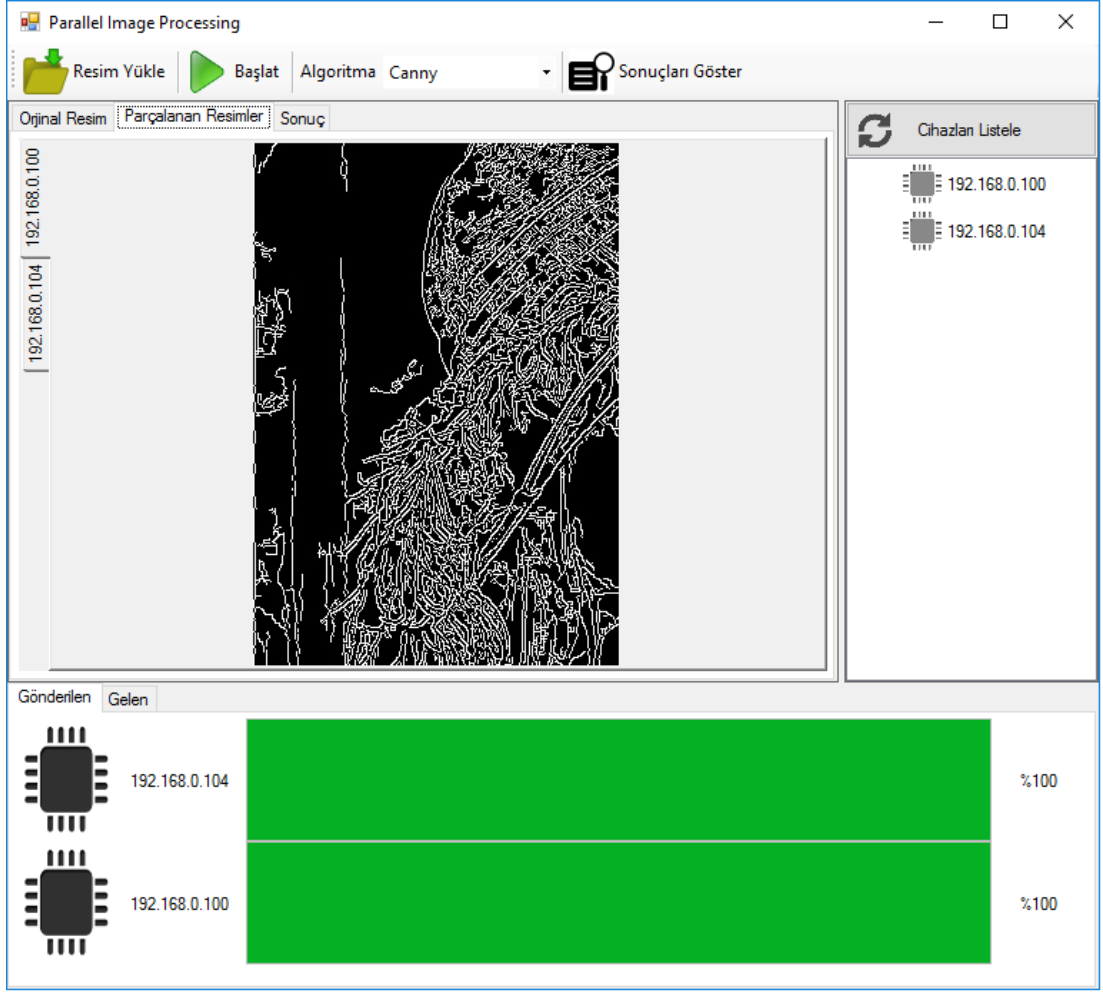
Şekil 4.6’ da kablosuz ağda 2 adet cihaz bulunduğu görülmektedir. Bu cihazlar yüklenen resmin işlenmesi sırasında kaç parçaya bölünerek hangi cihazlara gönderileceğini ifade eder. Temel amaç, kablosuz ağa bağlı olan tüm cihazları kullanarak algoritmanın işleme süresini kısaltmaktır. Buna göre yüklenen resmin işlenmesinde 2 adet cihaz kullanılacağı için resim 2 eşit parçaya bölünerek bu resim parçaları 192.168.0.100 ve 192.168.0.104 nolu cihazlarda ayrı ayrı işlenecektir.

Bir sonraki aşamada görüntünün işlenmesinde hangi algoritmanın kullanılacağını belirlenmesi gelir. “Algoritma” yazısının ve yanındaki boş alanın sağında yer alan “ok” a basılarak Combobox açılır kutu ile resme uygulanmak istenen kenar ve/veya köşe bulma algoritmalarından biri (burada Canny algoritması seçilmiştir) seçilir ve “Başlat” butonuna basılır (Şekil 4.7).



Şekil 4.7 Algoritmanın Seçilmesi ve Görüntü İşlemenin Başlatılması

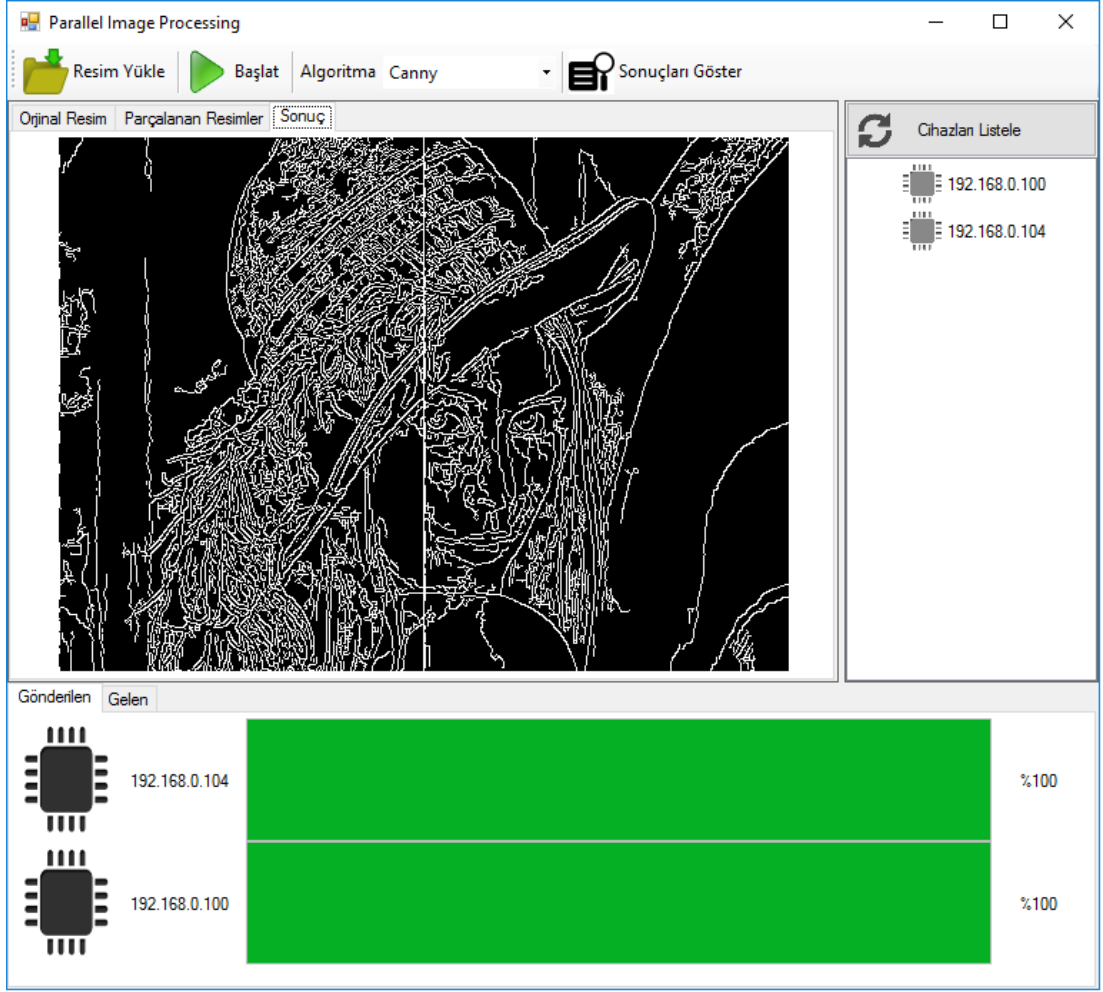
Böylece orijinal resmin 2 eşit parçaya bölünmüş görüntüleri ağ listesindeki sıralamaya göre ağdaki cihazlara eşit parçalar halinde gönderilmiş olur. “Parçalanmış Resimler” butonuna ve soldaki cihaz no’ lara basılarak resmin cihazlardaki işlenmiş parçaları görülebilir (Şekil 4.8).



Şekil 4.8 Resmin Parçalarının Cihazlarda İşlenmesi ve Görüntülenmesi

Şekil 4.8’ de uygulamanın alt kısmında cihazlara dağıtılan resim parçalarının Canny algoritmasındaki işleme durumlarına ilişkin işlem yüzdesi görülmektedir. Cihazlarda kendilerine dağıtılan resim parçasının işlenmesi tamamlandığında işlem yüzdesi %100 olarak görünür.

Resmin eşit 2 parçasının ağdaki 2 ayrı cihazda ayrı ayrı işlendikten sonra toplanarak tek bir parça haline getirilmiş sonuç resmi ise “Sonuç” butonuna basılarak görülebilir (Şekil 4.9).



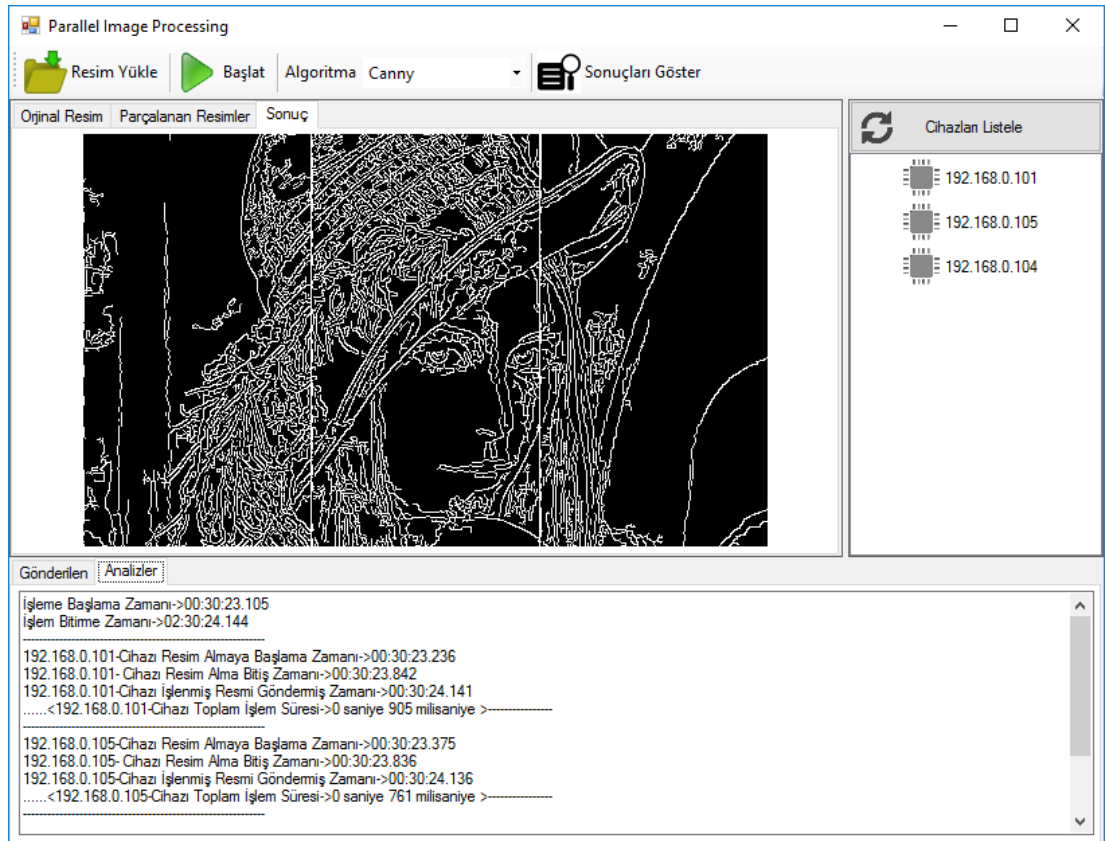
Şekil 4.9 Resmin Cihazlarda Ayrı Ayrı İşlenmiş Parçalarının Birleştirilmesi ve Sonuç Resiminin Görüntülenmesi

Böylece yüklenen orijinal resim, kablosuz ağa bağlı olan 2 cihaz aracılığıyla 2 eşit parçaya bölünerek ve verilen kenar ve/veya köşe algoritması cihazlarda ayrı ayrı işlenerek tekrar birleştirilmiş ve algoritmanın işleme süresi tek cihazla gerçekleştirilenden daha kısa sürede yapılabilmektedir.

4.3.5 . Performans analizleri

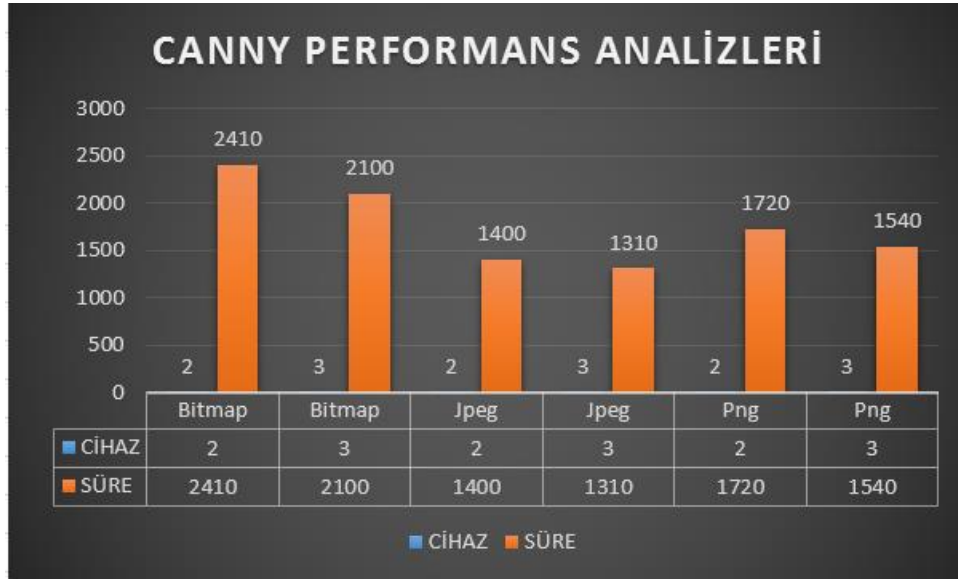
Bu bölümde, çalışma kapsamında kenar ve/veya köşe bulma görüntü işleme algoritmalarının eş zamanlı süreçlere ayrılarak kablosuz ağ üzerindeki cihazlar yardımıyla gerçekleştirilmesine ilişkin performans analizleri sunulacaktır.

Performans analizlerinde, uygulamadaki Canny ve Harris kenar ve köşe bulma algoritmalarının kablosuz ağdaki 2 ve 3 cihaz üzerinden aynı orijinal resmin Bitmap, jpeg ve png formatlarının 2 ve 3 eşit parçaya ayrılarak işlenmesi ve ana bilgisayarda toplanarak tek bir resim (sonuç resmi) haline getirilmesi işlemi için geçen toplam işlem süreleri ayrı ayrı bulunarak karşılaştırılmıştır.



Şekil 4.10 Resmin Cihazlarda Canny Algoritması ile Ayrı Ayrı İşlenmiş Parçalarının Birleştirilmesi ve Sonuç Resminin Görüntülenmesi Performans Analizi

Şekil 4.10’da; uygulamadaki Canny algoritması ile orijinal resmin kablosuz sistemde bulunan 3 cihaza eşit parçalar halinde gönderilmesi ve cihazlarda ayrı ayrı işlenen resim parçalarının ana bilgisayarda tekrar birleştirilmesiyle elde edilen sonuç resmi ve bu işlemin toplam süresinin bulunması amacıyla cihazlardaki sürelerinin hesaplanmasına ilişkin performans analizinden örnek bir kesit görülmektedir.

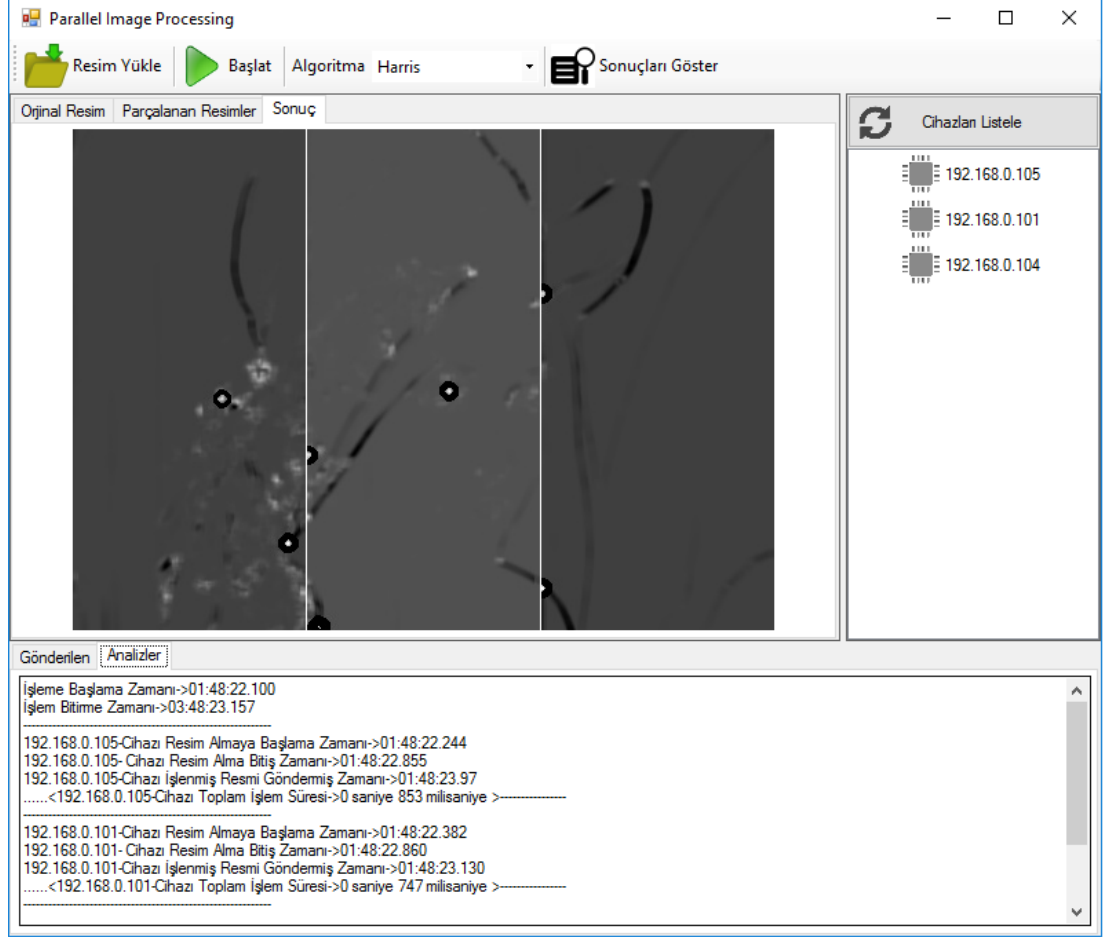


Şekil 4.11 Canny Algoritmasının Performans Analizleri

Şekil 4.11’ de ise; aynı orijinal resmin farklı formatlarının (bitmap, jpeg, png) sistemde 2 ya da 3 cihaz bulunması halinde Canny algoritması ile işlendiğinde geçen toplam süreye yönelik performans analizleri görülmektedir. Buna göre; Canny algoritması resmin jpeg formatında en hızlı (2 cihaz ile 1400ms, 3 cihaz ile 1310ms) sürede işlem yaparken, bitmap formatında en yavaş (2 cihaz ile 2410ms, 3 cihaz ile 2100ms) sürede işlem yapmaktadır. Resmin png formatının işlenmesi ise kablosuz ağda 2 cihaz bulunması halinde 1720ms, ağda 3 cihaz bulunması halinde ise 1540ms gibi orta hızda sürelerdedir.

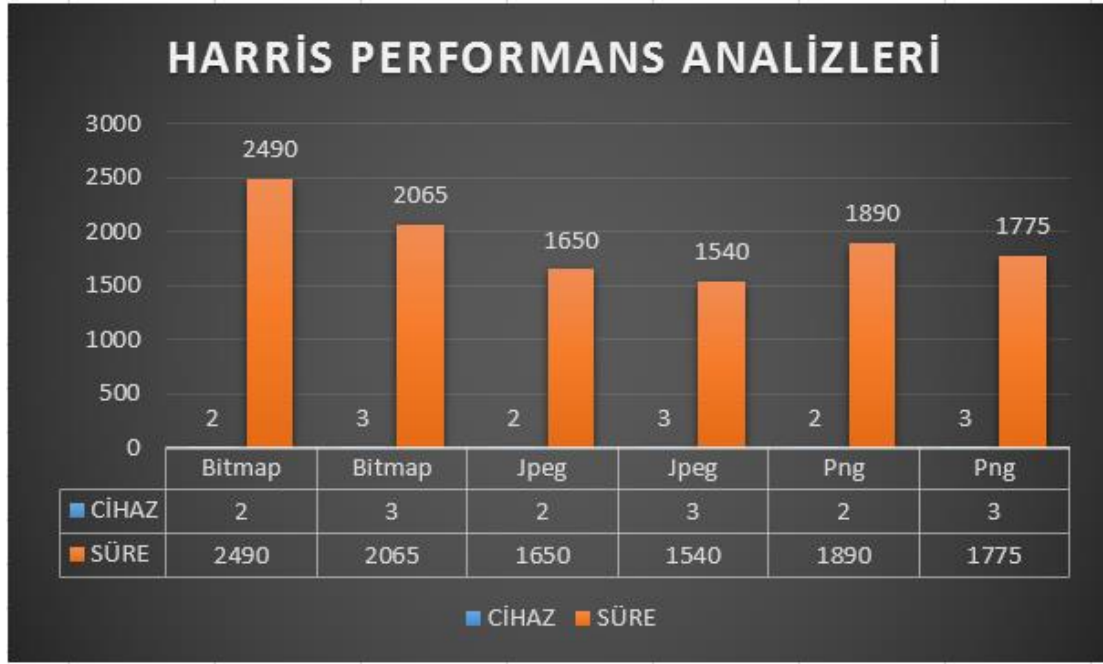
Şekil 4.12’de; uygulamadaki Harris algoritması ile orijinal resmin kablosuz sistemde bulunan 3 cihaza eşit parçalar halinde gönderilmesi ve cihazlarda ayrı ayrı işlenen resim parçalarının ana bilgisayarda tekrar birleştirilmesiyle elde edilen sonuç resmi ve bu işlemin toplam süresinin bulunması amacıyla cihazlardaki sürelerinin

hesaplanmasına ilişkin performans analizinden örnek bir kesit görülmektedir.



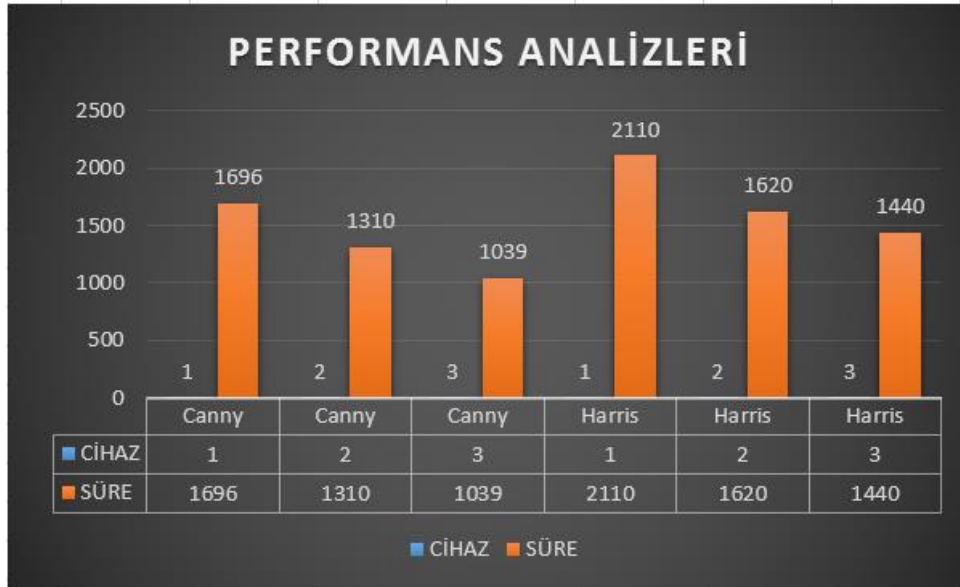
Şekil 4.12 Resmin Cihazlarda Harris Algoritması ile Ayrı Ayrı İşlenmiş Parçalarının Birleştirilmesi ve Sonuç Resminin Görüntülenmesi Performans Analizi

Şekil 4.13' de ise; aynı orijinal resmin farklı formatlarının (bitmap, jpeg, png) sistemde 2 ya da 3 cihaz bulunması halinde Harris algoritması ile işlendiğinde geçen toplam süreye yönelik performans analizleri görülmektedir. Buna göre; Harris algoritması resmin jpeg formatında en hızlı (2 cihaz ile 1650 ms, 3 cihaz ile 1540 ms) sürede işlem yaparken, bitmap formatında en yavaş (2 cihaz ile 2490 ms, 3 cihaz ile 2065 ms) sürede işlem yapmaktadır. Resmin png formatının işlenmesi ise kablosuz ağda 2 cihaz bulunması halinde 1890 ms, ağda 3 cihaz bulunması halinde ise 1775 ms gibi orta hızda sürelerdedir.



Şekil 4.13 Harris Algoritmasının Performans Analizleri

Şekil 4.14 ve Çizelge 4.1’ de Canny ve Harris algoritmalarının kablosuz ağda 1, 2 ve 3 cihaz bulunması durumlarında görüntü işleme performans analizleri sunulmuştur.



Şekil 4.14 Canny ve Harris Algoritmalarının Kablosuz Ağda 1, 2 ve 3 cihaz Bulunması Durumlarında Görüntü İşleme Performans Analizleri

Şekil 4.14 ve Çizelge 4.1'e göre; kablosuz ağda 1, 2 ve 3 cihaz bulunması halinde Canny Algoritması'nın toplam görüntü işlem süreleri sırasıyla 1696 ms, 1310 ms ve 1039 ms olurken; bu süreler Harris Algoritması'nda sırasıyla 2110 ms, 1620 ms ve 1440 ms' dir.

Çizelge 4.1. Canny ve Harris Algoritmalarının Kablosuz Ağda Karşılaştırılması

Algoritmalar	Cihaz Sayısına Göre Toplam İşlem Süresi (ms)			Cihaz Sayısı Artışına Karşılık Toplam Sürede Kısalma Oranı (%)		
	1	2	3	1→2	2→3	1→3
Canny	1696	1310	1039	22,76%	20,69%	38,74%
Harris	2210	1620	1440	26,70%	11,11%	34,84%

Çizelge 4.1'e göre; uygulamanın Canny Algoritması ile görüntü işleminde geçen toplam süre 2 cihaz kullanılması halinde ortalama %23, üç cihaz kullanılması halinde ise ortalama %21 (ağda 2 cihaz bulunması durumuyla karşılaştırıldığında) ve %39 (ağda 1 cihaz bulunması durumuyla karşılaştırıldığında) kısalmaktadır. Benzer şekilde; uygulamanın Harris Algoritması ile görüntü işleminde geçen toplam süre 2 cihaz kullanılması halinde ortalama %27, üç cihaz kullanılması halinde ise ortalama %11 (ağda 2 cihaz bulunması durumuyla karşılaştırıldığında) ve %35 (ağda 1 cihaz bulunması durumuyla karşılaştırıldığında) kısalmaktadır. Buna göre; kablosuz ağdaki cihaz sayısının artması her iki algoritma için de toplam görüntü işleme süresini kısaltmaktadır.

5. SONUÇ VE ÖNERİLER

Teknolojideki ilerleme ve internet altyapısındaki gelişmelerle birlikte sayısal ortamdaki verilerin herkese ulaşılabilir bir hâle gelmesi, veri miktarının ve görüntü veri tabanlarının çok büyük boyutlara ulaşması, internet üzerinde değişim, yenilik ve iyileştirmelerle beraber görüntü tabanlı çeşitli uygulamaları da beraberinde getirmiştir.

Görüntü işleme uygulamaları gerçekleştirmek isteyen geliştiriciler için pek çok hazır kütüphane ve yardımcı kaynak bulunmaktadır. Görüntü işleme kapsamında gerçekleştirilen uygulamalar internet ortamına aktarılarak veya internet üzerinde gerçekleştirilerek web tabanlı görüntü işleme uygulamaları, diğer durumlarda masaüstü tabanlı uygulamalar geliştirilebilmektedir. Görüntü işleme tekniklerinin gelişmesi, internet üzerinde bütünleşik olarak çalışabilen zeki uygulamaların çoğalması ve görüntü işleme kapsamında kullanılması ile görüntü işleme, internet üzerinde daha etkili ve aktif bir şekilde pek çok alanda uygulanabilir ve daha başarılı sonuçlar elde edilebilir. Günümüzde pek çok uygulamanın web ve hatta mobil tabanlı çalışabilecek şekilde getirilmesi ve geliştirilmesi ile bilgiye erişimin daha da esnek bir şekilde gerçekleştirilebildiği ve görüntü işlemenin de internet üzerinde daha esnek ve etkili bir şekilde uygulanabileceği önemli bir gerçektir. Sonuç olarak görüntü işleme, özellikle çoklu ortam için oldukça önemlidir. Metin tabanlı ve hatta ses tabanlı verilere bile görüntü işlemenin uygulanmasının mümkün olduğu görünmektedir.

Görüntü işlemede, farklı amaçlar için kullanılan Fourier dönüşüm metodu, Histogram eşitleme, Medyan (Orta) filtreleri, Gabor süzgeç metodu, dalgacık dönüşümü metodu, Topolojik Bağımsız Bileşen Analizi (TBBA) vb. birçok yöntem bulunmaktadır.

Görüntü işleme yöntemleri, temel işlevleri çerçevesinde değerlendirildiğinde; Gürültü giderme ve görüntü düzenleme, Renk düzeltme ve dönüştürme, Kenar ve köşe bulma, Geometrik dönüştürme, Sayısal karıştırma veya optik karıştırma şeklinde gruplanmaktadır. Görüntü işleme algoritmaları ise; Görüntü netleştirme algoritmaları, Gürültü giderme algoritmaları, Renk değiştirme ve düzeltme algoritmaları, Kenar

belirleme algoritmaları, Köşe saptama algoritmaları, Plaka okuma ve karakterlerini tanıma algoritmaları, Yüz tanıma algoritmaları vb. diğer algoritmalar olmak üzere oldukça çeşitlilik göstermektedir.

Görüntü işleme algoritmalarının eş zamanlı süreçlere ayrılarak kablosuz ağ üzerinden gerçekleşmesi ve performans analizlerinin incelenmesinin amaçlandığı bu çalışma kapsamında elde edilen sonuçlar aşağıda özetlenmiştir:

- Algoritmalarının işlem süresi, aynı orijinal resmin jpeg, bitmap ve png formatları için karşılaştırıldığında; jpeg formatında en kısa sürede işlem yapılırken, en uzun sürede görüntü işleme bitmap formatında olmaktadır.
- Canny algoritması Harris algoritmasına göre daha hızlı (daha kısa sürede) görüntü işlemektedir.
- Kablosuz ağdaki cihaz sayısının artması algoritmalar için toplam görüntü işleme süresini kısaltmaktadır (hızlandırmaktadır).

Bu çalışmada elde edilen sonuçlara dayanılarak ileride yapılacak benzer akademik çalışmalar için şu önerilerde bulunulabilir:

- Uygulamada, orijinal resmin eşit parçalara ayrılması, orijinal resmin ağdaki cihaz sayısı kadar dikey eşit parçalara ayrılması ile gerçekleştirilmektedir. Ancak orijinal resmin eşit parçalara ayrılarak işlenmesi sırasında kenar ve köşe algoritmaları tarafından, kesilmiş olan kısım, kenar ve köşe olarak algılanabilmektedir. Bu durum da parça resimlerin tekrar birleştirilmesi sırasında sonuç resminde birleştirilen kısımlarda bir çizgi görünmesine, dolayısıyla sonuç resminin parçalardan birleştirilmiş bir resim görüntüsü vermesine yol açmaktadır. İleride bu konuda yapılacak benzer akademik çalışmalarda; bu sorunun çözülebilmesi için algoritmada resimlerin kesilen kısımlarının ayrı cihazlarda işlenerek tekrar birleştirilmesi sonrasında sonuç resminde oluşan çizginin piksel genişliğinin hesaplanması ve görüntü işleme sonrası parça resimlerde kesilmiş kenarlarda oluşan çizgilerin bu piksel genişliği kadar kesilerek o şekilde birleştirilmesiyle ortadan kaldırılması önerilebilir.

- Uygulamadaki bir diğerk engelde orijinal resmin cihaz sayısı kadar eşit parçaya bölünmesinde dikey kesitlerin kullanılıyor olmasıdır. Bu durum, orijinal resmin en ve boy oranının 1' den çok farklı olması durumunda ve/veya kablosuz ağda çok sayıda cihaz bulunması halinde en ve boy oranı 1'den çok farklı resim parçaları oluşmasına ve bu parçaların kesilen kısımlarının tekrar birleştirme sırasında sonuç resminde cihaz sayısı kadar çizgi oluşturarak sonuç resmin çok parçalı resimlerden oluşmuş izlenimi vermesidir. İleride bu konuda yapılacak benzer akademik çalışmalarda; bu sorunun çözülebilmesi için uygulamanın orijinal resmin kablosuz ağdaki cihaz sayısına göre en ve boy oranı birbirine en yakın eşit parçalara sahip olacak şekilde bölünmesini, bunun için özellikle kablosuz ağda 4, 6, 8,... gibi çift sayıda cihaz bulunması halinde orijinal görüntünün yatay ve dikey kesitler kullanılarak eşit parçalara bölünmesini sağlayacak şekilde geliştirilmesi önerilebilir.
- Son olarak; ileride bu konuda yapılacak benzer akademik çalışmalarda; mevcut uygulamanın jpeg, bmp, tif, gif, png, pdf ve diğerk sık kullanılan resim formatlarını ve Hugh, Susan, Prewitt, Sobel ve Roberts Algoritmalarını destekleyecek şekilde geliştirilmesi; ayrıca uygulamanın kullanılan tüm algoritmalar, tüm resim formatları, farklı dosya boyutu ve piksel sayısına sahip resimler için karşılaştırmalı performans analizlerinin yapılması önerilebilir.

KAYNAKLAR

- [1] Karabiber F.,“Analog Hücresel Sinir Ağı İşlemcisi Kullanarak Gerçek Zamanlı Görüntü İşleme Uygulamaları”, Yayınlanmamış Doktora Tezi, İstanbul Üniversitesi, İstanbul ,2009.
- [2] Karakoç M.“Görüntü işleme, teknolojiler ve uygulamaları”, Akademik Bilişim 2012 (1-3 Şubat 2012), Uşak: Uşak Üniversitesi, 2012.
- [3] Dinçer S.,“Dijital Görüntü İşleme Teknikleri ile Matlab ve Mikrokontrolör Kullanılarak İki Boyutlu Görüntünün Çizdirilmesi”, Yayınlanmamış Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, İstanbul, 2009.
- [4] Tatlı M.,“Görüntü İşleme Araçları ile Mobil Ortamda Aydınlatma Tasarımı Yazılımı”, Yayınlanmamış Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi, Isparta,2013.
- [5] Badem H.,“GPU İle Hızlandırılmış Bulanık Mantık Algoritmalarının Görüntü İşlemede Kullanılması”, Yayınlanmamış Yüksek Lisans Tezi, Kahramanmaraş Sütçü İmam Üniversitesi, Kahramanmaraş, 2012.
- [6] Yüksel H.,“Medikal Görüntü İşleme İçin Fraktal Geometri Kullanarak Üç Boyutlu Modelleme ve Hızlı Prototipleme Yazılımının Geliştirilmesi”, Yayınlanmamış Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi, Isparta, 2010.
- [7] Alçın M., “Görüntü işleme esaslı parmakizi doğrulama”, Yayınlanmamış Yüksek Lisans Tezi, İstanbul: Marmara Üniversitesi, 2007.
- [8] Karakoç M., “Görüntü işleme teknikleri ve yapay zeka yöntemleri kullanarak görüntü içinde görüntü arama”, Yayınlanmamış Yüksek Lisans Tezi, Pamukkale Üniversitesi, Bursa, 2011.
- [9] Kurtulmuş F., “Olgunlaşmamış Şeftali Meyvesini Doğal Bahçe Koşullarında Alınmış Görüntülerde Görüntü İşleme Teknikleri ve Yapay Sınıflandırıcılarla Saptayarak Sayan Algoritmaların Geliştirilmesi”, Yayınlanmamış Doktora Tezi, Uludağ Üniversitesi. Bursa, 2012.
- [10] Yıldız K., “Kumaş Hatalarının Isıl Görüntüleme ve Görüntü İşleme Teknikleri ile Tespit Edilmesi”, Yayınlanmamış Doktora Tezi, Marmara Üniversitesi, İstanbul, 2014.
- [11] Akan H., “Sayısal Görüntü İşlemeye Dayalı Proses Kontrolü İçin Bir Sistem Tasarımı”, Yayınlanmamış Yüksek Lisans Tezi, Marmara Üniversitesi, İstanbul, 2007.

- [12] Arslan E., “Hücresel Sinir Ağı Sistemleri Kullanarak Hareketli Nesnelerin Görüntü İşleme Uygulamaları”,Yayınlanmamış Doktora Tezi,İstanbul Üniversitesi, İstanbul, 2011.
- [13] Karaduman G., “Demiryollarına Ray Profil Analizi İçin Üç Boyutlu Görüntü İşleme”, Yayınlanmamış Yüksek Lisans Tezi, Fırat Üniversitesi, Elazığ, 2013.
- [14] İlhan İ., “ Dört Rotorlu İnsansız Hava Aracı İçin Görüntü İşleme Tabanlı Akıllı Kontrol Algoritmalarının Geliştirilmesi”, Yayınlanmamış Yüksek Lisans Tezi, Fırat Üniversitesi, Elazığ, 2013.
- [15] Özdil A. “Evaluation of Face Recognition Algorithms on Embedded Systems”, Unpublished Master Thesis, Melikşah University, Kayseri, 2015.
- [16] Kurt A., “Nükleer Tıpta Planar ve Tomografik İncelemede Standart Fantom Kullanılarak Farklı Sistemlerin Görüntü İşleme Algoritmalarının Kıyaslanması”, Yayınlanmamış Yüksek Lisans Tezi, Ankara Üniversitesi, 2009.
- [17] Aydın İ., Bölüm:0 Sayısal Görüntü İşlemeye Giriş, Elazığ, 2013. http://web.firat.edu.tr/iaydin/bmu357/bmu357_bolum0.pdf (03.11.2016).
- [18] Perihanoğlu G.M., “Dijital görüntü işleme teknikleri kullanılarak görüntülerden detay çıkarımı”, Yayınlanmış Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, İstanbul, 2015.
- [19] MEB, “Kablosuz Ağ Sistemleri”, Ankara: Milli Eğitim Bakanlığı. 2011a. [http://www.megep.meb.gov.tr/mte_program_modul/moduller_pdf/Kablosuz Ağ Sistemleri.pdf](http://www.megep.meb.gov.tr/mte_program_modul/moduller_pdf/Kablosuz_Ag_Sistemleri.pdf), (25.10.2016).
- [20] Aydın S., “Atatürk Kitaplığı Süreli Yayınları Sayısallaştırma Çalışmalarının Teknik ve Yönetimsel Analizi”, Yayınlanmamış Yüksek Lisans Tezi, Marmara Üniversitesi, İstanbul, 2011.
- [21] Zeki İ. M., “Ürün yüzey hatalarının yayılım tabanlı görüntü işleme algoritmaları ile tespiti”, Yayınlanmamış Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, İstanbul, 2011.
- [22] Sevgen S., “Hücresel sinir ağları için kararlı şablon tasarımı ve görüntü işleme uygulamaları”, Yayınlanmamış Doktora Tezi, İstanbul Üniversitesi, İstanbul, 2009.
- [23] Kısa M., “Karayolunda seyreden araçların tanınması”, Yayınlanmamış Doktora Tezi, Selçuk Üniversitesi, Konya, 2014.
- [24] Özalp M., 2015, İmge İyileştirme, <http://documents.tips/documents/imge-iyilestirme-sunu.html> , (03.11.2016).
- [25] Yetik İ.Ş., ELE 473-573 Sayısal Görüntü İşleme - Projeler/Projects, 2015, <http://syetik.etu.edu.tr/ele573/projects.html> ,(23.11.2016).

- [26] Huang J., Kumar S.R., Mitra M., Zhu W.J., Zabih R., “Spatial color indexing and applications”, International Journal of Computer Vision, Volume:35, Number:3, pp.245–268. 1999, <http://www.cs.cornell.edu/~rdz/Papers/jing-IJCV99.pdf> , (01.12.2016).
- [27] Polat R., “Biyomedikal Görüntü Bölütleme”, Yayınlanmamış Yüksek Lisans Tezi, Fırat Üniversitesi, Elazığ, 2007.
- [28] Krutsh R. ve Tenorio, D. Histogram Equalization, Uygulama Notları, 2011.
- [29] Luhman T., Robson S. ve Kyle S., Close Range Photogrametry Principles, Methods and Applications, 2011.
- [30] Bayram B., Dijital Görüntü İşleme, Lisans Ders Notları, Yıldız Teknik Üniversitesi Jeodezi ve Fotogrametri Mühendisliği Bölümü, İstanbul, 2012.
- [31] Bellanger M., “Digital Processing of Signal: Theory and Practice”, John Wiley and Sons, USA, 2000.
- [32] Rosten E. & Drummond T., “Machine learning for high-speed corner detection”, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2006, 3951 LNCS, pp.430–443.
- [33] Chang X., Gao L. & Li Y., “Corner Detection Based on Morphological Disk Element”, In 2007 American Control Conference (July 11-13, 2007). Marriott Marquis Hotel at Times Square New York City, USA, 2007. pp.1994–1999, <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4282506> , (02.12.2016).
- [34] Zhang X., Lei M., Yang D., Wang Y. ve Ma L., “Multi-scale curvature product for robust image corner detection in curvature scale space”, Pattern Recognition Letters, 2007. 28(5), pp.545–554.
- [35] Nain N., Laxmi V., Jain, A.K., & Agarwal R., “Morphological Edge Detection And Corner Detection Algorithm Using Chain-Encoding”. In IPCV’06. 2006. pp.1-6.
- [36] Moravec H.P., “Obstacle avoidance and navigation in the real world by a seeing robot Rover”, Unpublished Doctoral Dissertation, CA, Stanford University, USA, 1980. https://www.ri.cmu.edu/publication_view.html?pub_id=22 , (01.12.2016).
- [37] Harris C. & Stephens M., “A Combined Corner and Edge Detector. Proceedings of the Alvey Vision Conference”, 1988, pp.147–151, <http://www.bmva.org/bmvc/1988/avc-88-023.html> , (14.11.2016).

- [38] Kalaycı T.E. “Görüntü İşleme”. KRDB Research Centre for Knowledge and Data, Faculty of Computer Science, Free University of Bozen-Bolzano. 2007, https://tekrei.github.io/dosyalar/sunum/gi_r.pdf , (23.11.2016).
- [39] Akar E. O., “Tarihi görüntülerin kalitesinin görüntü işleme teknikleri kullanılarak artırılması”, Yayınlanmamış Yüksek Lisans Tezi, Selçuk Üniversitesi, Ankara, 2006.
- [40] Siyah B., “Görüntünün Sınır Eğrisini Çıkaran Filtreler (Matlab)”, 2012. <http://www.bulentsiyah.com/goruntunun-sinir-egrisini-cikaran-filtreler-matlab/> , (27.10.2016).
- [41] Çil M.M., “Temel Görüntü İşleme Algoritmalarının FPGA Üzerinde Gerçeklenmesi”, Yayınlanmamış Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, İstanbul, 2015.
- [42] Fathima A.A, Karthik, R. & Vaidehi, V., “Image stitching with combined moment invariants and sift features1. Procedia Computer Science, 2013, Volume:19(Ant), pp.420–427. <http://dx.doi.org/10.1016/j.procs.2013.06.057> , (02.12.2016).
- [43] Varol A.H., 2014. Görüntü mozaikleme ile panoramik görüntü üretimi. Yayınlanmamış Yüksek Lisans Tezi, Başkent Üniversitesi, Ankara, <http://dspace.baskent.edu.tr:8080/xmlui/bitstream/handle/11727/2108/10053903.pdf?sequence=1&isAllowed=y> (23.11.2016).
- [44] Shen F. & Wang, H., “A Local edge detector used for finding corners”, The Third International Conference on Information, Communications and Signal Processing, Number:2, 2001, <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.104.429> , (22.12.2016).
- [45] Smith S. & Brady, J., “SUSAN—a new approach to low level image processing”, International journal of computer vision, Volume:23, Number:1, pp.45–78. 1997, <http://link.springer.com/article/10.1023/A:1007963824710> , (22.12.2016).
- [46] Chen J. Chen, J., Zou, L., Zhang, J., & Dou, L., “The Comparison and Application of Corner Detection Algorithms”, Journal of Multimedia, Volume:4, Number:6, pp.435–441. 2009.
- [47] Doğan B., “Tibbi Enstrumantasyon Tasarım ve Uygulamaları - Sayısal Filtreler”, 2012, <http://web.itu.edu.tr/~bdogan/dersler/tetu1314/TETU-SayisalFiltreler.pdf> , (27.11.2016).
- [48] Yiğit N., Karaboğa, N. & Gürer, B., “Adaptif Süzgeçlerde Farksal Gelişim Algoritması Kullanılarak Gürültü Giderme”, EMO IV. Öğrenci Kurultayı Sempozyumu (18-19 Ekim 2007), Ankara, TMMOB Elektrik Mühendisleri Odası, 2007, http://www.emo.org.tr/ekler/e13d1e382d895a5_ek.pdf , (19.11.2016).

- [49] Akgün D., “Paralel görüntü filtreleme için Çok çekirdekli bilgisayar üzerinde başarıml analizi”, İleri Teknoloji Bilimleri Dergisi, 2013, Cilt:2, Sayı:1, ss.76–83.
- [50] Vincent L., Morphological Grayscale Reconstruction in Image Analysis: Applications and Efficient Algorithms. IEEE Transactions on Image Processing, Volume:2, Number:2, pp.176–201. 1993.
- [51] Mukhopadhyay S. & Chanda B., Multiscale morphological segmentation of gray-scale images, IEEE Transactions on Image Processing, Volume:12, Number:5, pp.533–549. 2003.
- [52] Jaffery Z.A., Zaheeruddin & Singh L., Performance Analysis of Image Segmentation Methods for the Detection of Masses in Mammograms. International Journal of Computer Applications, Volume:82, Number:2, pp.44–50. 2013. <http://research.ijcaonline.org/volume82/number2/pxc3892100.pdf> , (01.01.2017).
- [53] Acar U. & Bayram B., Building extraction with morphology, RAST 2009 - Proceedings of 4th International Conference on Recent Advances Space Technologies, Number:1982, pp.33–38. 2009.
- [54] Fisher R., Perkins S., Walker A., & Wolfart E., Hypermedia image processing reference, 1996. <http://www.dsi.unive.it/~atorsell/Hipr.pdf> , (26.12.2016).
- [55] Girod B. & Wetzstein G., Digital image processing, 2007. http://web.stanford.edu/class/ee368/Handouts/Lectures/2015_Spring/1-Introduction_16x9.pdf , (28.12.2016).
- [56] Çelik A.R., “Görüntü işleme algoritmalarının FPGA donanımı üzerinde gerçekleştirilmesi”, Bildiriler: 17. Akademik Bilişim Konferansı (4-6 Şubat 2015). AB’15, Anadolu Üniversitesi, Eskişehir, Türkiye. 2015.
- [57] Çevik K.K. & Çakır A., “Görüntü İşleme Yöntemleriyle Araç Plakalarının Tanınarak Kapı Kontrolünün Gerçekleştirilmesi”, AKÜ Fen Bilimleri Dergisi, 2010, Sayı:1, ss.31–38.
- [58] MEB, “Kablosuz Ağlar”, Milli Eğitim Bakanlığı, Ankara, 2011b, <http://www.onurunurlu.com/dosyalar/agsistemleriyonlendirme/kablosuzaglar.pdf> , (26.11.2016).
- [59] Atalay A., “Kablosuz Ağlar”, 2009, <http://aliatalay.net/ag/kablosuz.pdf> , (25.10.2016).
- [60] Anon., “Kablosuz Ağ Çeşitleri”, Bir iyilik bilgi paylaşım platformu, 2010, <http://www.biriyilik.com/bilgisayar/kablosuz-ag-cesitleri-35381.html> , (29.10.2016).
- [61] Tekin F., “İnternet ve İleri Erişim Teknolojileri”, Yayınlanmamış Yüksek Lisans Tezi, Sakarya Üniversitesi, Sakarya, 2006.

- [62] Coşkun N., “İnsansız Kara Aracı Kontrol Sisteminin Tasarımı ve İmalatı”, Yayınlanmamış Yüksek Lisans Tezi , Abant İzzet Baysal Üniversitesi, Bolu, 2016.
- [63] Güçlü B., “Raspberry pi kullanarak görüntü işleme ile belirlenen hedefi bulma ve işaretleme”, 2016, <http://mekatronizm.com/raspberr-pi-kullanarak-goruntu-isleme-ile-belirlenen-hedefi-bulma-ve-isaretleme/> , (17.12.2016).
- [64] Akkuş S., “Gömülü Sistem Tabanlı, Kriptolu TCP/IP Veri Haberleşmesi Uygulaması”, Yayınlanmamış Yüksek Lisans Tezi, Marmara Üniversitesi, İstanbul, 2016.
- [65] Ersin Ç., “Arduino Mikrodenetleyici ve Güneş Enerjisi ile Çalışan Otomatik Bitki Sulama Sistemi”, Yayınlanmamış Yüksek Lisans Tezi, Süleyman Demirel Üniversitesi, Isparta, 2015.
- [66] Elektrikport, “Raspberry Pi Nedir? Arduino ile Farkları Nelerdir?”, 2013, <http://www.elektrikport.com/teknik-kutuphane/raspberry-pi-nedir-%20arduino-ile-farklari-nelerdir-/8305#ad-image-0> , (23.12.2016).
- [67] Yavuz M.T., “Paletli Mobil Manipülatör Tasarımı ve Modellemesi”, Yayınlanmamış Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, İstanbul, 2014.
- [68] Pişkin M., “Opencv ile görüntü işleme”, 2016, <http://mesutpiskin.com/blog/wp-content/uploads/2016/10/OpenCV-ile-G%C3%B6r%C3%BCnt%C3%BC-%C4%B0%C5%9Fleme.pdf> , (15.12.2016).
- [69] Arıman M., “Wireless Access Management Design in Software Defined Networks”, Unpublished Master’s thesis, Istanbul Technical University, Istanbul, 2016.

ÖZGEÇMİŞ

Başar Turan, 1988 yılında İstanbul'da doğdu. 2002 yılında öğrenime başladığı Elazığ Anadolu Teknik Lisesi Bilgisayar Bölümü'nü 2005 yılında tamamladı ve 2006 yılında Erzincan Üniversitesi Bilgisayar Teknolojileri ve Programlama Bölümü'ne girmeye hak kazandı. 2008 yılında bu bölümü bitirdikten sonra 2010-2014 Yılları arasında Haliç Üniversitesi Bilgisayar Mühendisliği Bölümü'nde öğrenim gördü. 2014 yılı itibariyle Maltepe Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği bölümünde Yüksek Lisans eğitime başlamıştır. 2015 yılının Temmuz ayına kadar Audio Firmasında Ar-Ge Mühendisliği yapmıştır. Başar Turan, şu an Arimas firmasında Proje Müdürü olarak profesyonel iş yaşantısına devam etmektedir.